UNIVERSIDADE FEDERAL DO MARANHÃO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
CURSO DE PÓS-GRADUAÇÃO EM ENGENHARIA DE ELETRICIDADE

SIMARA VIEIRA DA ROCHA

ELICITAÇÃO DE REQUISITOS BASEADA EM OBJETIVOS PARA POLÍTICAS DE SEGURANÇA E PRIVACIDADE EM COMÉRCIO ELETRÔNICO

São Luís
2005

SIMARA VIEIRA DA ROCHA

ELICITAÇÃO DE REQUISITOS BASEADA EM OBJETIVOS PARA POLÍTICAS DE SEGURANÇA E PRIVACIDADE EM COMÉRCIO ELETRÔNICO

Dissertação submetida ao Mestrado em Engenharia em Eletricidade da Universidade Federal do Maranhão como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação.

Orientador: Prof. Zair Abdelouahab.

São Luís
2005

SIMARA VIEIRA DA ROCHA

ELICITAÇÃO DE REQUISITOS BASEADA EM OBJETIVOS PARA POLÍTICAS DE SEGURANÇA E PRIVACIDADE EM COMÉRCIO ELETRÔNICO

Dissertação submetida ao Mestrado em Engenharia em Eletricidade da Universidade Federal do Maranhão como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação.

Aprovada em 30/ 09/ 2005

BANCA EXAMINADORA

Prof. Zair Abdelouahab, Ph.D
(Orientador)

Profa. Rossana Maria de Castro Andrade, Ph.D
(Membro da Banca Examinadora)

Prof. Maria Del Rosário Girard, Ph.D
(Membro da Banca Examinadora)

A meus Pais

AGRADECIMENTOS

A Deus, por ter me concedido o prazer de viver e também forças para buscar alcançar os meus objetivos.

A minha mãe, por ser meu exemplo diário de força, garra e determinação.

Ao meu pai, por sempre me incentivar a buscar novos conhecimentos.

Ao meu orientador, professor Zair Abdelouahab, pela imensurável paciência e dedicação dispensada à realização desse trabalho.

A todos os meus amigos por terem sido minha segunda família.

A todos aqueles que contribuíram direta ou indiretamente para realização desse trabalho.

“Understanding a problem is sometime as difficult as inventing the solution”
B. Russel

RESUMO

Este trabalho descreve um método para elicitação de requisitos baseado em objetivos para sistemas de comércio eletrônico, em conformidade com as políticas de segurança e privacidade existentes em um *site*. O método é originado pela integração das abordagens UWA [33] com a instanciação do método GBRAM [6] para o desenvolvimento de políticas e requisitos de sistemas de comércio eletrônicos seguros. O método resultante tem por objetivo garantir que as políticas de segurança e privacidade existentes nunca se tornem obsoletas com a adoção de novas funcionalidades a um *site*. Para tanto, provê meios para que os requisitos elicitados estejam em conformidade com as mesmas. Por outro lado, caso as organizações não tenham estabelecido suas políticas, a abordagem proposta sugere modelos através dos quais é possível a criação de tais políticas. Por fim, o método proposto ainda apresenta um modelo para o documento de especificação de requisitos, como forma de estabelecer um meio padrão para especificar requisitos de software, o qual poderá ser útil tanto para as equipes de desenvolvimento, na tentativa de facilitar a construção de sistemas, quanto para as equipes de análises, nas futuras manutenções ou acréscimo de funcionalidades ao *site*.

Palavras-Chave: Objetivos, Elicitação, Requisitos, Políticas de Segurança e Privacidade.

ABSTRACT

This work describes a method for the elicitation of requirements based on goals for electronic commerce systems in agreement with security and privacy polices of a site. The method integrates the UWA approach [33] with the GBRAM method [6] for developing requirements and policies for secure electronic commerce systems. The resulting method aims to guarantee that existent security and privacy policies do not become obsolete after the adoption of new functionalities to a site. For this reason, the method provides means to set the elicited requirements in conformity with these new functionalities. On the other hand, organizations that have not established their policies yet, the proposed approach suggests some models through which it is possible to create such policies. At last, the proposed method presents a model for the document of requirements specification in agreement with the approach described in this work, as way of establishing a standard means to specify software requirements that can be as useful for the developing teams, in the attempt of facilitating the construction of systems, as for the analyzing teams, in the future maintenances or increment of functionalities to a site.

Key-words: Goal, Elicitation, Requirements, Security and Privacy Policies.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE SIGLAS

UWA	- Aplicações Web Ubíquas
GBRAM	- Método de Análise de Requisitos baseados em Objetivos
ER	- Entidade Relacionamento
LEL	- Léxico Extendido da Linguagem
BMV	- Visão do Modelo Básico
RC	- Blocos de Requisitos
DRS	- Documento de Requisitos de Software
PFIRES	- Policy Framework for Interpreting Risk in e-Commerce Security
HoQ	- House of Quality
B2B	- Business to Business
B2C	- Business to Consumer
B2E	- Business to Employment
DoS	- Denial of Service

SUMÁRIO

1 INTRODUÇÃO

1.1 Descrição do problema

"A Engenharia de *Software* é o processo de desenvolvimento de um sistema baseado em computador, cujo principal objetivo é a construção de sistemas que apóiam os negócios das empresas e que atendam às necessidades e expectativas de seus usuários finais" [20].

Ainda segundo [20], entre as diversas fases do ciclo de vida do desenvolvimento de um *software*, a Engenharia de Requisitos é uma das fases mais cruciais e visa sistematizar o processo de definição dos requisitos, gerando especificações que descrevam de forma não ambígua, consistente e completa o comportamento do universo do domínio do problema.

Além disso, a Engenharia de Requisitos trata não apenas de conhecimentos técnicos, mas também gerenciais, organizacionais e econômicos, e está intimamente relacionada com a qualidade do software. Por esses motivos, ela tem sido muito estudada com o objetivo de melhorar a elicitação dos requisitos e obter requisitos mais confiáveis.

O modelo de processo mais comum da Engenharia de Requisitos divide as atividades em quatro grupos: elicitação de requisitos, negociação e análise de requisitos, documentação dos requisitos e validação dos requisitos.

De acordo com o *IEEE Standard Glossary of Software Engineering Terminology de 1997* (IEEE97), define-se requisito como uma condição ou capacidade necessária para um usuário resolver um problema ou alcançar um objetivo, onde se conclui que requisitos de *software* são derivados da necessidade que usuários têm de resolver algum problema.

Segundo [30, 31], os requisitos podem ser classificados em: requisitos de domínio, requisitos funcionais e requisitos não funcionais. Os requisitos de domínio são originados do domínio da aplicação do sistema e refletem as características desse domínio, podendo ser

funcionais ou não funcionais. Os requisitos funcionais descrevem os aspectos comportamentais de um sistema. Já os requisitos não funcionais especificam os aspectos não comportamentais de um sistema, capturando as propriedades e restrições sobre as quais um sistema deve operar, tais como: segurança, confiabilidade, disponibilidade, usabilidade e etc.

Na elaboração e especificação de um projeto de *software* é fundamental a observação e compreensão de quais requisitos são necessários para que este funcione bem, já que a qualidade do que será produzido dependerá fortemente de uma boa captura e especificação de requisitos. O objetivo é determinar não só o que o *software* deverá fazer, mas também os critérios de validação, os quais serão utilizados para avaliar se este cumpre o que foi previamente definido.

Porém, a obtenção desses requisitos nem sempre é feita de forma fácil, pois envolve a comunicação direta com o usuário, e entender o que o usuário deseja nem sempre é tarefa simples. Assim, a utilização de técnicas apropriadas para esse fim pode contribuir de maneira significativa para otimizar esse processo.

Por outro lado, com o advento das práticas de comércio eletrônico, a segurança e a privacidade da informação são, sem dúvida, uma das grandes preocupações das empresas sintonizadas com o seu tempo. Porém, a habilidade de determinar onde o negócio necessita de segurança e as quais características de segurança são apropriadas, dado um ambiente organizacional, é vital para o desenvolvimento de aplicações de comércio eletrônico.

O principal objetivo no desenvolvimento de uma política de segurança é estabelecer as expectativas organizacionais propostas para o uso do sistema e, também, os procedimentos para responder aos eventos de segurança. Dessa forma, devido à natureza dinâmica dos sistemas de comércio eletrônico, a criação de uma política de segurança é um trabalho progressivo e iterativo.

Em contrapartida, segundo [13, 27], reduzir ameaças a dados sensíveis é foco de muitos estudos endereçados a prover melhor segurança para a privacidade dos dados. Contudo, apesar de muitas organizações estarem cientes do problema dos acessos não autorizados a dados sensíveis, poucas têm estabelecido um programa de segurança eficiente para seus sistemas.

Todavia, quando novas tecnologias são adotadas, as políticas de segurança e privacidade organizacionais devem ser revistas e, geralmente, revisadas, para responder aos conflitos introduzidos por essas novas tecnologias. Portanto, há uma necessidade de criação de abordagens que permitam tanto o desenvolvimento de políticas de segurança e privacidade quanto garantam que elas não se tornem obsoletas pela adoção de novas funcionalidades a um *site*.

Este trabalho propõe a integração da abordagem UWA [33] com a instanciação do método GBRAM [6] para o desenvolvimento de políticas de segurança e privacidade, cujo objetivo é unir métodos da engenharia de requisitos, em especial a utilização de objetivos e cenários, para o desenvolvimento de aplicações de comércio eletrônico, em obediência às políticas de segurança e privacidade existentes em uma organização. Com a integração desses métodos, o método resultante visa suprir as principais deficiências descritas acima.

O UWA foi escolhido por centralizar o processo de elicitação dos requisitos a partir da identificação dos *stakeholders* e também por atribuir valores de entrega a cada objetivo capturado, o que poderá ser útil para resolução de conflitos e refinamento dos objetivos.

Por outro lado, a opção pela utilização do GBRAM instanciado para o desenvolvimento de políticas e requisitos para sistemas de comércio eletrônico seguros foi pelo fato de poder estender as fases de avaliação de riscos e avaliação de obediência, através da adição das técnicas propostas por [15, 32].

De acordo com [33], os objetivos são metas de alto nível que os *stakeholders* gostariam que a aplicação satisfizesse e os requisitos são metas de baixo nível que o sistema supostamente conhecerá e que poderão se diretamente entendidas e realizadas pelos desenvolvedores. Dessa forma, os objetivos genericamente representam os requisitos do sistema e, dependendo do nível de abstração, podem ser refinados por uma combinação de requisitos, de uma maneira recursiva.

A escolha pela utilização de objetivos deu-se, primeiro, pelo fato destes serem focados no porquê os sistemas são construídos, expressando a razão e a justificativa do propósito do sistema. Segundo, porque focando nos objetivos, ao invés dos requisitos específicos, permitem aos analistas se comunicarem com os *stakeholders,* usando uma linguagem baseada em conceitos com os quais eles estão mais familiarizados. Além disso, como os objetivos são tipicamente mais estáveis do que os requisitos, eles são uma fonte benéfica para a derivação dos requisitos [4].

Finalmente, os cenários, por serem descrições de comportamentos do sistema, podem sintetizar traços de comportamento de um sistema existente e também ajudar a descoberta de objetivos [22, 26].

1.2 Objetivos

Este trabalho visa criar um método para elicitação de requisitos baseado em objetivos para políticas de segurança e privacidade em comércio eletrônico possibilitando que, uma vez estabelecidas as políticas de segurança e privacidade de um *website*, elas podem continuar consistentes, mesmo com a adoção de novas funcionalidades ao *site*.

Para atingir esse objetivo, este trabalho empregará a seguinte metodologia:

- ✓ Investigar as principais abordagens existentes baseadas em cenários e objetivos;

- ✓ Investigar o domínio de *e-commerce*, enfatizando seus modelos e as políticas de segurança e privacidade;
- ✓ Propor um método para elicitação de requisitos em obediência com as políticas de segurança e privacidade;
- ✓ Mostrar a aplicabilidade do método em um estudo de caso.

1.3 Estrutura da dissertação

Este trabalho está organizado em cinco capítulos. O primeiro apresenta uma introdução ao tema, o qual descreve o problema a ser estudado, os objetivos pretendidos e a estrutura de organização do trabalho.

O segundo capítulo ilustra o estado da arte, considerando a fase da engenharia de requisitos de vários métodos que utilizam cenários e objetivos para a elicitação e modelagem de requisitos. Além disso, são apresentados os principais conceitos de comércio eletrônico, seus modelos mais importantes e também as definições e relevâncias das políticas de segurança e privacidade para tais sistemas.

Com a finalidade de facilitar e sistematizar o processo de obtenção dos requisitos para sistemas de comércio eletrônico foi criado, no capítulo três, um método para elicitação de requisitos baseado em objetivos, em obediência às políticas de segurança e privacidade.

O quarto capítulo se destina à aplicação do método proposto em um estudo de caso para elicitar os requisitos de uma livraria on-line.

Finalmente, o quinto capítulo faz as considerações finais, relatando as contribuições desse trabalho e as perspectivas de trabalhos futuros.

2 O ESTADO DA ARTE

O objetivo deste capítulo é descrever diversas abordagens que utilizam objetivos e cenários na fase de engenharia de requisitos. Além disso, são apresentados os principais conceitos referentes a comércio eletrônico.

Assim, na seção 2.1 são apresentadas algumas abordagens que utilizam cenários como meio para elicitar e modelar requisitos [1, 2, 3, 17, 18, 23, 26]. Já na seção 2.2 são descritas algumas abordagens baseadas em objetivos [4, 5, 7, 12, 33, 34] para capturar e modelar os requisitos. Ao término de cada seção é feito um resumo das principais características das abordagens descritas, como forma de evidenciar os principais pontos de cada uma delas.

Finalmente, nas seções 2.3, 2.4, 2.5, 2.6 e 2.7 são apresentados os principais conceitos envolvendo comércio eletrônico [28, 29], suas modalidades, políticas de segurança e privacidade, e a instanciação do método GBRAM [6] para a formulação de tais políticas, respectivamente.

2.1 Abordagens que utilizam cenários para elicitação e modelagem de requisitos

2.1.1 Modelo de cenários de Júlio César Leite et al

Segundo essa abordagem [17, 18], à idéia de cenário comumente existente na literatura, foram adicionados quatro conceitos principais, a saber:

- ✓ Um cenário começa pela descrição das situações do macrosistema e sua relação com o sistema mais externo, isto é, primeiro deve-se considerar as interfaces do macrosistema e, posteriormente, deve-se descrever as interfaces do software com seu macrosistema;

- ✓ Um cenário evolui à medida que se processa a construção do software;
- ✓ Cenários são naturalmente relacionados ao Léxico Estendido da Linguagem (LEL) e à Visão do Modelo Básico (BMV) da estrutura dos requisitos;
- ✓ Um cenário descreve situações com ênfase na descrição do comportamento. Desse modo, similarmente ao BMV, cenários usam descrições de linguagem natural como sua representação básica.

Assim, essa abordagem apresenta uma proposta que adiciona à estrutura de requisitos uma visão do cenário, a qual constitui-se das seguintes visões: visão do modelo léxico, visão do modelo básico, visão de cenário, visão de hipertexto e visão de configuração.

A visão do modelo léxico é a representação dos símbolos na linguagem do domínio do problema, denominada LEL, a qual baseia-se na idéia de entender a linguagem do problema sem se preocupar com o entendimento do problema.

Dessa forma, o LEL é uma representação da linguagem natural que auxilia na captura do vocabulário da aplicação, cujo foco principal é registrar os sinais (palavras ou frases) que são peculiares ao domínio. O objetivo é descobrir a semântica dos símbolos de uma aplicação no contexto em que ela se aplica, descrevendo precisamente a semântica dos símbolos próprios da linguagem de aplicação. A descrição dessa semântica segue um sistema de representação no qual cada símbolo deve ser descrito através de noções e impactos, onde as noções são descrições do significado das palavras ou frases e os impactos são descrições das representações dessas palavras ou frases, isto é, uma descrição extra da informação no contexto em que ela se aplica.

Já a elicitação da linguagem de aplicação objetiva a identificação de termos que sejam próprios da linguagem em questão, ou seja, a identificação de palavras ou frases

peculiares ao meio social da aplicação em estudo, onde, após essa fase, procura-se associar cada palavra ou frase a um significado.

São consideradas três fases para a coleta de fatos: identificação de símbolos da linguagem, identificação da semântica de cada símbolo e identificação das regras de produção de uma gramática que gera a linguagem de aplicação.

A validação interliga as três fases descritas acima por meio de um ciclo de validação que é apoiado em três tipos de retroalimentação: necessidade de identificar novos símbolos, correções de noções e impactos, e correção ou inclusão de regras.

A visão do modelo básico é uma estrutura que incorpora sentenças sobre o sistema desejado, onde tais sentenças são escritas em linguagem natural seguindo padrões definidos.

A visão do modelo de cenário é uma estrutura composta de objetivo, contexto, recursos, atores e episódios. Objetivos, contexto, recursos e atores são sentenças declarativas. Episódios são um conjunto de sentenças em linguagem simples que torna possível a operacionalização da descrição dos comportamentos.

A figura 2.1 mostra uma estrutura de entidade-relacionamento que descreve esse modelo de cenário.

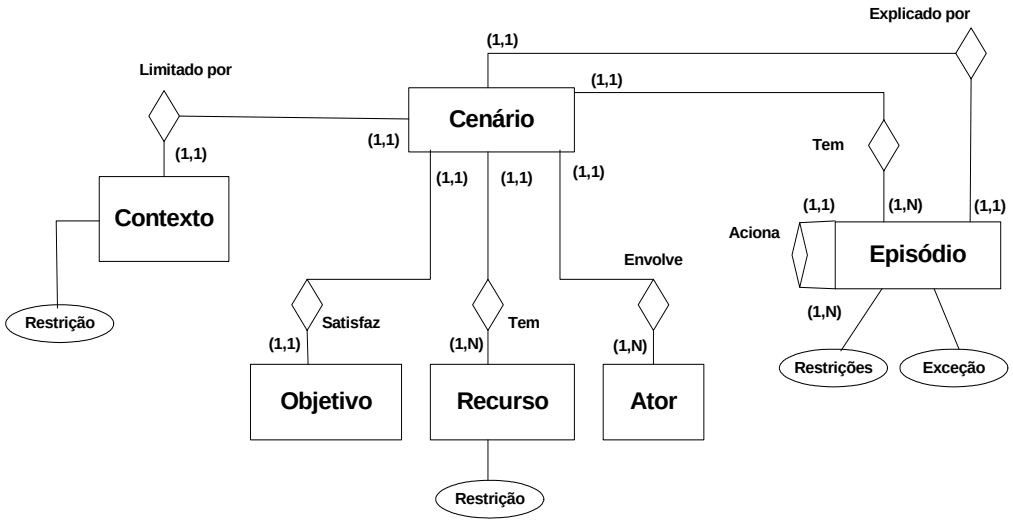

Figura 2.1 - Diagrama de ER do Modelo de Júlio César Leite et al [18].

A visão hipertexto é ortogonal à visão do modelo básico, à visão do modelo léxico e à visão do modelo de cenário. Porém, será focalizada somente a visão hipertexto a partir do ponto de vista do cenário, por ser o modelo mais enfatizado nessa seção.

Assim, existem muitos relacionamentos a serem explorados entre cenários e outros componentes da estrutura de requisitos. A partir de cada cenário é derivado um, ou mais nós hipertextos, que são relacionados por relacionamentos hipertextos, podendo-se ter um nó classe para cada tipo de entidade definida no modelo de cenários.

A razão para que nós hipertextos sejam derivados a partir de cenários é que podem ser construídas diferentes visões do mesmo cenário, de acordo com o perfil ou tarefa do usuário. Essa abordagem é usual em abordagens de projeto hipermídia moderna e permite, por exemplo, concentrar toda a informação desejada a partir de um cenário em um nó composto, contendo: título, objetivo, contexto, recursos como atributos e o conjunto de episódios como partes do nó, alcançáveis por relacionamentos estruturais.

Finalmente, a visão de configuração é essencial para manter a rastreabilidade dos produtos e suas visões. O LEL, o BMV e a visão do modelo de cenário estão sujeitos a um controle de versão e de configuração. Em um dado momento, uma visão de estrutura pode ser requisitada baseada na configuração atual ou em configurações passadas.

Cada versão do modelo mantém as seguintes informações: data, hora, usuário que faz a mudança, razões da mudança e tipo de mudança. Sendo que a consistência da configuração é garantida pelas restrições de consistência determinadas por cada modelo. Dessa forma, gerações de mudanças acionam um processo de checagem de consistência responsável pela uniformidade de uma dada configuração.

Do exposto nessa seção, observa-se que um aspecto negativo dessa abordagem é o fato da mesma apresentar-se como um modelo fechado, onde é necessário o conhecimento do

LEL e também de todo o seu processo de elaboração para que se possa fazer alguma extensão ao mesmo.

2.1.2 Abordagem CREWS-L'ECRITOIRE

Em [1, 2, 3, 26] é descrita a abordagem CREWS-L'ECRITOIRE, a qual propõe descobrir os requisitos por meio de um acoplamento objetivo-cenário e, assim, auxiliar diretamente na atividade da engenharia de requisitos. Dessa forma, o processo de elicitação de requisitos é organizado em duas fases: obtenção dos cenários e descoberta dos objetivos, os quais são executados de maneira iterativa.

A utilização do acoplamento bidirecional cenário-objetivo é uma das características principais dessa abordagem, pois permite mover-se, a partir dos cenários, para os objetivos e vice-versa. Na direção objetivo-cenário, à medida que um objetivo é descoberto, este é imediatamente operacionalizado por meio de um cenário. Já na direção inversa, ou seja, de cenários para objetivos, a abordagem conduz o processo de captura dos requisitos descobrindo novos objetivos através da análise de cenários textuais.

Outro aspecto importante provido por essa abordagem é a utilização dos relacionamentos *AND/OR* e também de refinamento entre os objetivos, que visa organizar um conjunto de requisitos em uma hierarquia de blocos de requisitos (RCs).

Finalmente, essa abordagem ainda provê um suporte metodológico, CREWS-L'Ecritoire, constituído na forma de guias de regras executáveis personificadas em um ambiente de software.

O núcleo dessa abordagem é o RC, definido como o par <objetivo, cenário>, o qual está organizado em três níveis de abstração: contextual, interação do sistema e interno do sistema. Sendo que cada um desses níveis corresponde a um tipo de RC.

No primeiro nível, o contextual, é feito a identificação dos serviços que o sistema deve prover para uma organização, assim como suas razões. Dessa forma, o RC contextual acopla um objetivo de projeto a um cenário de serviço (descreve o fluxo de serviços entre os agentes que são necessários para executar o objetivo do projeto).

Já o nível de interação concentra-se nas interações entre o sistema e seus usuários, que são necessárias para executar os serviços associados ao sistema no primeiro nível. Assim, um RC de interação combina um objetivo de serviço a um cenário de interação do sistema (descreve os fluxos de interações entre o sistema e seus usuários para executar o objetivo de serviço).

O último nível, interno, é focado no que o sistema internamente pode executar. Dessa forma, um RC interno combina um objetivo do sistema (expressa uma maneira possível de executar uma ação identificada em um cenário de interação) a um cenário interno.

Como mencionado acima, essa abordagem utiliza os relacionamentos *AND* (composição), que interligam aqueles RCs, os quais requerem um ao outro para definir o sistema funcionamento completamente; os relacionamentos OR (alternativa), que expressam as maneiras alternativas de executar o mesmo objetivo; e os relacionamentos de refinamento, que são utilizados para descrever os RCs em diferentes níveis de abstração, sendo direcionados pela parte do cenário que considera toda interação do cenário de nível i como um objetivo a ser executado no nível i + 1. Esses relacionamentos servem para interligar os RCs, formando uma hierarquia dos mesmos.

A figura 2.2 apresenta o modelo do processo da abordagem CREWS-L'Ecritoire, que é representado como um mapa, isto é, um grafo direcionado com intenções como nós, e estratégias como setas.

Como mostrado pela figura 2.2, o modelo de processo consiste em diferentes estratégias para suportar a elicitação de objetivos, às quais são associadas regras para descoberta de novos objetivos, implementadas em um protótipo de software L'Ecritoire.

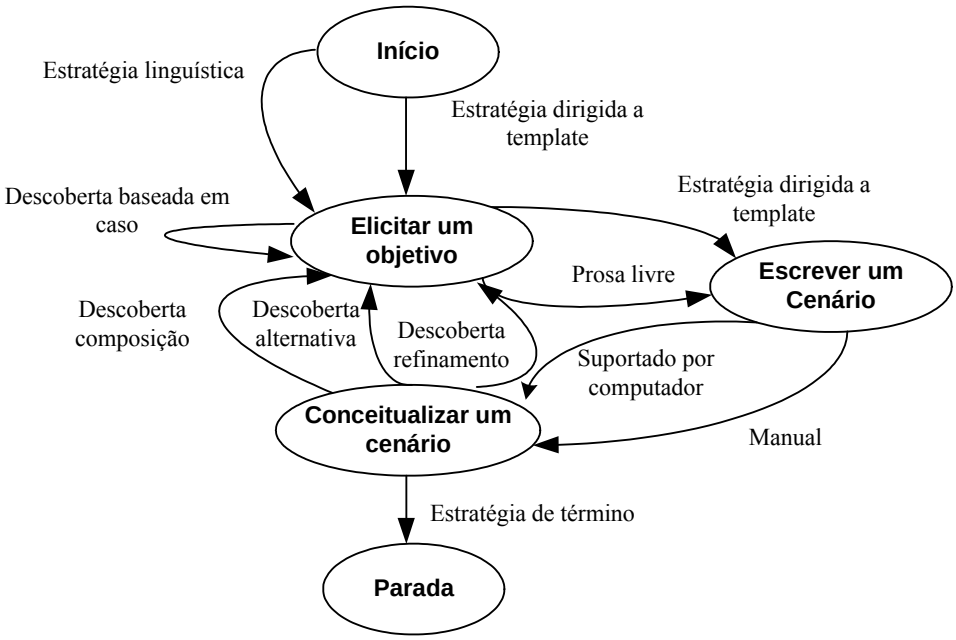

Figura 2.2 - Modelo de Processo da abordagem CREWS-L'Ecritoire [2].

A abordagem ainda prover um modelo de produto, conforme ilustra a figura 2.3, o qual representa os principais conceitos utilizados pela mesma, relacionados por *links* de composição, associação ou *is-a*.

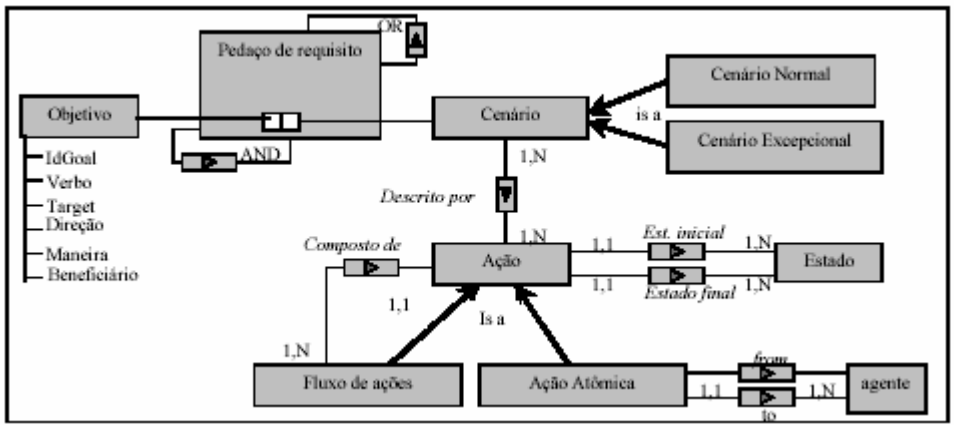

Figura 2.3 – Modelo de produto abordagem CREWS-L'Ecritoire [2].

Resumidamente, pode-se dizer que a abordagem CREWS-L'Ecritoire tem os seguintes objetivos:

- ✓ Obter cenários, através do uso de guias de estilos e conteúdos informais para a escrita destes. Também provê *templates* que visa auxiliar tanto na escrita quanto na limitação do tamanho dos mesmos;
- ✓ Organizar cenários, por meio da integração de cenários e requisitos dos sistemas em uma coleção de cenários;
- ✓ Transcrever os cenários informais para os formais;
- ✓ Prover suporte metodológico, através de um ambiente de software (L'Ecritoire), onde são implementadas as regras para a descoberta de objetivos e verificação de cenários;
- ✓ Descobrir requisitos/objetivos pela análise dos cenários de uma coleção de objetivo-cenário;
- ✓ Fragmentar os cenários, por meio da organização de coleções objetivo-cenário de uma forma hierárquica.

Como aspectos negativos dessa abordagem pode-se destacar o fato da não utilização do conceito de "caso de uso", nem seu relacionamento com os "objetivos". Além disso, não formaliza estratégia para a descoberta de objetivos iniciais.

Em [10] é apresentado e formalizado um método que integra a abordagem CREWS L'ecritoire com a abordagem de casos de uso descrita por Regnell et al e o método GBRAM baseado em objetivos, o qual visa suprir as deficiências descritas acima.

2.1.3 Análise de requisitos baseado em um modelo cíclico de averiguação

Em [23] é descrito um modelo que se constitui em uma estrutura formal e dinâmica para descrever discussões sobre requisitos, e refere-se à necessidade de suportar a comunicação durante o processo de elicitação de requisitos.

A figura 2.4 mostra o modelo cíclico de averiguação, onde os requisitos são incrementalmente elaborados por meio de discussões e confirmações, contribuindo, assim, para o refinamento dos mesmos.

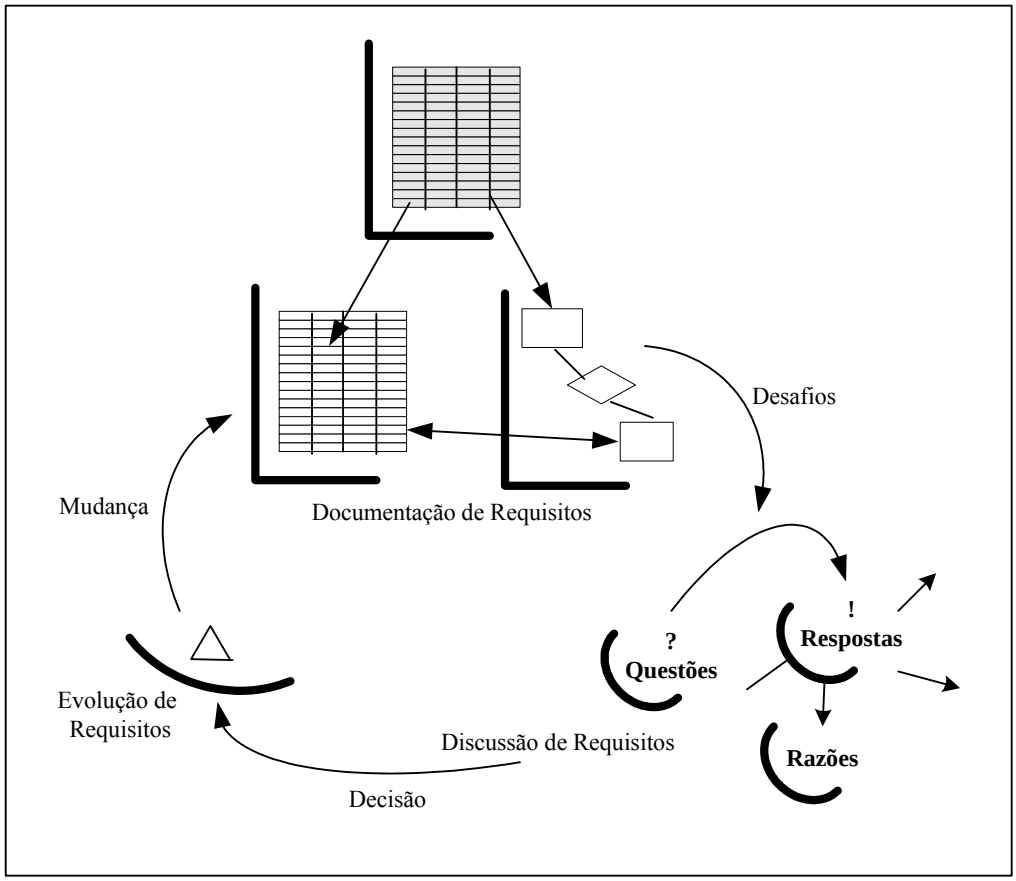

Figura 2.4 – Modelo cíclico de averiguação [23].

Esse modelo está dividido em três fases: documentação, discussão e evolução dos requisitos, conforme evidencia a figura acima.

A fase de documentação dos requisitos refere-se ao processo onde os *stakeholders* escrevem os requisitos propostos, sendo que cada requisito é especificado em forma de um nó hipertexto separado.

Nessa fase existem algumas maneiras de analisar requisitos. No caso de haver um documento de requisitos, este pode ser revisado. Caso não haja, deve-se começar a partir do nada para escrever requisitos baseados em entrevistas, documentação técnica para sistemas similares e outros.

Segundo essa abordagem, uma importante técnica que pode ser usada nessa fase é a análise de cenários, que é, simplesmente, o uso específico proposto do sistema. Mais precisamente, um cenário é uma descrição de uma ou mais transações, envolvendo o sistema requerido e o ambiente. Cenários podem ser documentados de diferentes maneiras, que vão depender do nível de detalhamento necessário. A forma mais simples é o caso de uso, o qual consiste meramente de uma pequena descrição com um número relacionado. Formas mais detalhadas são chamadas de *scripts* e são usualmente representadas como tabelas ou diagramas, envolvendo a identificação de uma ação e uma agente da ação. Por esse motivo, um *script* pode ser chamado de tabela de ação.

É importante ressaltar que cenários são úteis na aquisição e validação de requisitos, porém eles não constituem requisitos, uma vez que descrevem o comportamento do sistema apenas em situações específicas. Por outro lado, uma especificação descreve o que o sistema deve fazer em geral.

A segunda fase, discussão de requisitos, consiste do momento em que os *stakeholders* discutem os requisitos propostos, com anotações, e é composta de três elementos: questões, onde muitas discussões começam porque um *stakeholder* tem uma questão sobre o requisito; respostas, onde são propostas soluções para os problemas na forma de refinamentos candidatos ou revisões que respondem às questões, sendo que uma questão

pode gerar muitas respostas e estas servem para esclarecer os *stakeholders* e auxiliar a noticiar ambigüidades, requisitos perdidos e inconsistências; e razões, que consistem em respostas que requerem justificativas, quando os requisitos não são imediatamente óbvios.

Na terceira fase, evolução dos requisitos, os *stakeholders* relacionam as requisições de mudanças aos requisitos, na base de discussão, refinando-os quando os requisitos de mudanças são aprovados.

Observa-se que o objetivo da fase de discussão de requisitos é a confirmação de permanecer ou modificar o requisito, sendo que uma requisição de mudança pode conduzir de volta para uma discussão, a qual constitui sua razão, e também avançar para os requisitos modificados, uma vez que eles tenham sido executados.

Outro ponto importante a ser destacado é que, assim como na fase de discussão, a fase de evolução pode ocorrer gradualmente e informalmente, ou em discretas mudanças, seguindo uma revisão formal e procedimentos aprovados.

Como aspecto negativo, pode-se destacar que a natureza informal desse modelo o torna sujeito a ambigüidades, incertabilidades, dificuldades referentes à comunicação entre os envolvidos no processo, acordos sobre os requisitos e, também, sobre o gerenciamento de mudança dos mesmos [23].

2.1.4 Resumo das abordagens selecionadas que utilizam cenários como forma de elicitar e modelar requisitos

Esta seção apresenta um resumo das principais características das abordagens descritas nas seções anteriores, conforme mostrado nas tabelas 2.1.

Como critérios de resumo foram escolhidas as fases do método, as características mais importantes, os principais conceitos em que os mesmos se baseiam e também os produtos obtidos como resultado de cada um deles.

Método	Fases	Conceitos utilizados	Produtos obtidos	Características mais importantes	Limitações
Abordagem de Júlio César Leite et al	1 – Identificação dos símbolos da linguagem 2 – Identificação da semântica de cada símbolo 3 – Identificação das regras de produção de uma gramática que gera a linguagem de aplicação	• Cenário • Objetivo • Contexto • Recursos • Atores • Episódios • LEL	• Modelo Léxico • Modelo Básico • Modelo de Cenários	• O BMV é uma estrutura que incorpora sentenças sobre o sistema desejado • O LEL é uma representação da linguagem natural que auxilia a capturar o vocabulário da aplicação	• Modelo fechado, onde é necessário o conhecimento do LEL e todo seu processo de elaboração para que o mesmo possa ser extendido.
Abordagem CREWS-L'ECRITOIRE	1 – Elicitação de objetivos 2 – Escrita dos cenários 3 – Conceitualização dos cenários	• RC • Objetivos • Cenários normais • Cenários excepcionais • Agente • Estado • Ação	• Modelo de processo • Modelo de produto	• O RC constitui o núcleo da abordagem e estão organizados em três níveis de abstração: contextual, interação do sistema e interno do sistema. • Acoplamento bidirecional objetivo-cenário • Os relacionamentos AND, OR e de Refinamento interligam os RCs.	• Não apresenta estratégias eficazes para elicitação dos objetivos iniciais • Não focaliza o conceito de "casos de uso", nem seus relacionamentos com os objetivos.
Modelo Cíclico de Averiguação	1 – Documentação 2 – Discussão 3 – Evolução	• Cenários • Casos de Uso • *Scripts*/Tabelas de ação • Agente de ação • Ação	• Documentação dos requisitos por meio de cenários • Modelo cíclico de averiguação	• Suporta a comunicação durante o processo de requisitos • Esse modelo consiste de uma série de questões e respostas projetadas para capturar o processo de elaboração de requisitos e documentação através de refinamento	• Apresenta-se sujeito a ambigüidades e incertabilidades, por sua informalidade. • Dificuldades referentes à comunicação entre os envolvidos no processo

Tabela 2.1 – Resumo das abordagens baseadas em cenários

2.2 Abordagens que utilizam objetivos para elicitação e modelagem de requisitos

2.2.1 Meta-modelo KAOS

Em [12] é apresentado um metamodelo para elicitar requisitos iniciais e uma estratégia para conduzir o processo de aquisição de requisitos. Os requisitos considerados referem-se à parte a ser automatizada do sistema, seu ambiente físico e a maneira como essas partes tem que cooperar.

Essa abordagem foi desenvolvida no projeto KAOS e consiste em três componentes: o modelo conceitual, para adquirir e estruturar modelos de requisitos com uma linguagem de aquisição associada; um conjunto de estratégias para elaborar modelos de requisitos nesta estrutura; e um assistente automatizado para prover guias no processo de aquisição de acordo com as estratégias.

O modelo conceitual provê um número de abstrações em termos das quais modelos de requisitos são adquiridos. O metamodelo para aquisição de requisitos pode ser representado como um grafo conceitual, onde os nós representam as abstrações e as setas representam os *links* de estruturação.

A estratégia de aquisição define uma série de passos para a obtenção de componentes do modelo de requisitos como instâncias dos componentes do metamodelo. O assistente de aquisição ajuda a prover suporte automatizado na adoção de uma estratégia de aquisição a outra.

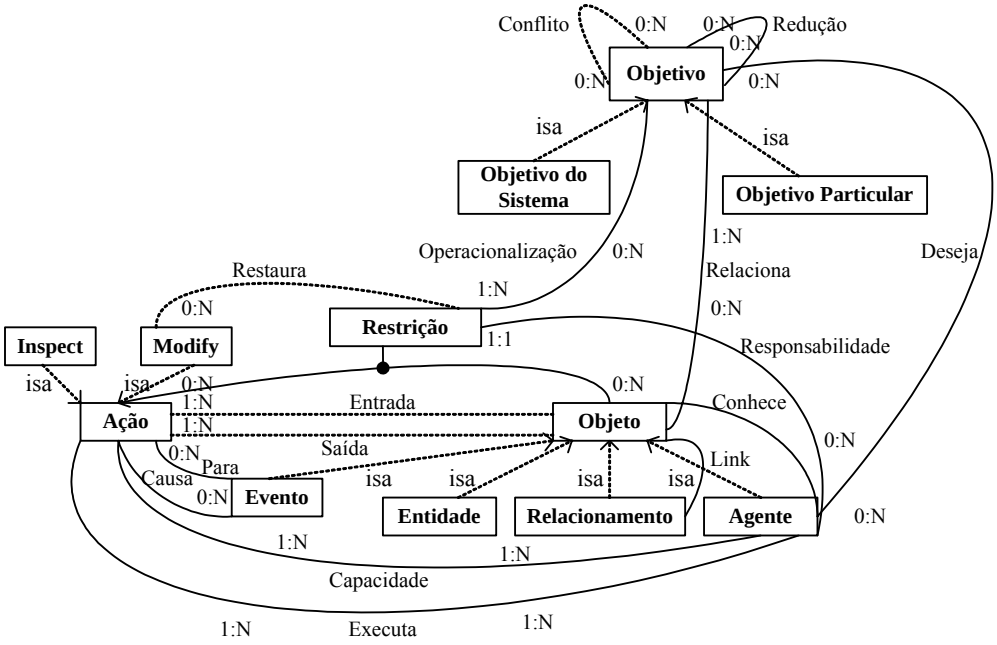

Figura 2.5 - Uma parte do metamodelo conceitual KAOS [12].

Essa abordagem envolve três níveis de modelagem, a saber:

- ✓ Nível meta: onde são elaboradas as abstrações independentemente do domínio;
- ✓ Nível domínio: onde são definidos conceitos específicos do domínio da aplicação;
- ✓ Nível instância: onde são definidas instâncias específicas dos conceitos do nível de domínio.

O metamodelo KAOS é um modelo conceitual para o nível meta, consistindo de conceitos de nível meta, relacionamentos, atributos e restrições. A linguagem de aquisição adotada é uma estrutura de dois níveis. Um nível mais externo, para declarar conceitos de nível de domínio em termos dos componentes do metamodelo, e um nível mais interno, para expressar valores para alguns meta-atributos.

O nível mais externo possui uma estrutura de entidade-relacionamento, produzindo a estrutura de banco de dados de requisitos. Já o nível mais interno corresponde a uma lógica de primeira ordem temporal.

Dentre as abstrações mais importantes envolvidas na parte do metamodelo (conforme figura 2.5) para a estratégia de aquisição direcionada a objetivos, destaca-se: um objeto, que é algo de interesse que pode ser referenciado nos requisitos; uma entidade, que é um objeto autônomo, onde suas instâncias podem existir independentemente de outras instâncias do objeto; um relacionamento, que é um objeto subordinado, isto é, sua existência depende da existência de objetos correspondentes ligados pelo relacionamento; um evento, que é um objeto instantâneo, onde suas instâncias existem apenas em um estado; ação, que é uma relação matemática sobre os objetos, onde aplicações destas definem transições de estados; restrições, que são formuladas em termos dos objetos e ações disponíveis para algum agente no sistema; e objetivos, os quais referem-se a um objetivo não operacional a ser executado pelo sistema composto (isto é, parte para ser automatizada, seu ambiente físico e a maneira como essas partes terão que cooperar), e são operacionalizados através das restrições.

O processo de aquisição é guiado por estratégias e modelos de domínio. As estratégias definem formas específicas de adquirir instâncias dos vários nós e *links* pertencentes ao grafo do metamodelo. Dessa forma, ele é direcionado para objetivo e consiste dos seguintes passos:

1. Adquirir estrutura do objetivo e identificar objetivos relacionados: neste passo são adquiridas instâncias do meta-conceito “objetivo” e do meta-relacionamento “redução”. A elaboração da estrutura do objetivo consiste em identificar os objetivos do sistema, sua categoria e padrões, associando-os com os objetivos fontes que eles reduzem; identificar os objetos referenciados pelos

objetivos, elaborando uma definição preliminar de seus aspectos; e identificar possíveis conflitos entre os objetivos do sistema.

2. Identificar agentes potenciais e suas capacidades: consiste em especificar as instâncias direcionadas a objetivos do meta-conceito “agente”, meta-relacionamento “capacidade” e meta-conceito “ação”.
3. Operacionalizar objetivos em restrições: os objetivos folha na estrutura objetivo, elaborados no passo um, são transformados em objetivos do sistema, formulados em termos de objetos e ações.
4. Refinar objetos e ações: neste passo, o analista define os objetos e ações, e completa a descrição dos objetos e ações já identificados. Além disso, ainda são definidos os relacionamentos de responsabilidade.
5. Derivar ações e objetivos para garantir restrições: aqui, ações e objetos são derivados para garantir que todas as restrições requeridas sejam satisfeitas.
6. Identificar responsabilidades alternativas: para cada restrição obtida no passo três, são identificados os vários links de responsabilidade. A identificação é baseada nas capacidades dos agentes determinados no passo dois.
7. Associar ações a agentes responsáveis: links de execução são efetivamente associados a agentes para as várias ações elaboradas nos passos dois e quatro, na base do link alternativo de responsabilidade estabelecido no passo seis. Uma ação é associada a um agente apenas se o mesmo tiver sido determinado entre os candidatos alternativos que tomam responsabilidade sobre as restrições que as ações garantem.

É importante ressaltar que o metamodelo KAOS suporta a noção de decomposição de objetivos, inclusive decomposição alternativa para comportamentos que podem satisfazer os

mesmos objetivos básicos, a qual inclui os relacionamentos de conflito entre objetivos, assim como os de redução *AND* e *OR*.

Como pode ser evidenciado pelo exposto nessa seção, o principal problema percebido nessa abordagem é o grande número de passos e abstrações durante o processo existente na mesma, os quais são difíceis de executar. Além disso, para iniciar o processo (passo um) é necessário gerar a estrutura objetivo, a qual é originária da identificação de subobjetivos de um objetivo, porém, como saber quando todos os objetivos requeridos foram identificados, se a abordagem não prover nenhuma estratégia para tanto?

Por outro lado, para executar o passo cinco, são oferecidas ao analista algumas regras de inferência para derivar e assegurar pré-condições, pós-condições e invariantes. Porém, fica difícil para o analista selecionar a regra manualmente e aplicá-la de modo correto.

2.2.2 O projeto UWA

Em [33, 34] é apresentado o projeto *Ubiquitous Web Applications* (UWA) que descreve uma *framework* geral cujo propósito é definir uma tecnologia e uma metodologia global para cobrir diferentes aspectos do trabalho de conduzir um projeto de desenvolvimento de software. Ele consiste de: um léxico comum e um conjunto de conceitos de projeto; um comum entendimento de diferentes atividades de projeto e suas interdependências mútuas; e um comum entendimento de diferentes aspectos metodológicos e tecnológicos.

O escopo do projeto UWA é desenvolver aplicações *web* ubíquas multi-dispositivos. Em particular seu foco é:

- ✓ Melhorar a qualidade do projeto, significando por qualidade, a integridade e a legibilidade do documento do projeto e suas semânticas;

- ✓ Melhorar a qualidade e a efetividade das aplicações, visando à usabilidade e ao comportamento da aplicação;
- ✓ Melhorar o processo do projeto, significando estruturá-lo de maneira mais organizada e mais suportada pelas ferramentas e, assim, mais produtivo;
- ✓ Melhorar a eficiência do ciclo global desenvolvimento-manutenção, ajudando os desenvolvedores a melhor lidar com os problemas de evolução e mudanças das aplicações (devido às mudanças de contexto e/ou mudanças de requisitos).

Para tanto, o UWA divide o projeto global do problema em subproblemas ou aspectos do projeto, tais como: definição dos requisitos, estabelecendo o que o sistema deverá fazer; projeto de hipermídia, definindo a informação e os aspectos de interação da aplicação; projeto de operações, determinando as operações que estarão disponíveis aos usuários pela aplicação; projeto de transações, estabelecendo as transações (seqüência de operações com propriedades adicionais específicas) que são fornecidas pela aplicação; e projeto customizado, definindo a adaptação da aplicação para as características do contexto e, em particular, para as características de dispositivos, de canais de conexão, de localização, etc.

Para cada um dos aspectos descritos acima, o UWA apresenta os seguintes elementos: um modelo, que captura o conjunto de conceitos e primitivas usados na construção do esquema, isto é, uma descrição da aplicação; uma notação, ou seja, um caminho lingüístico e visual para o projetista formular o esquema; um conjunto de *guidelines* e heurísticas, que ajudam o projetista a fazer as escolhas de projetos de maneira mais apropriada; e um conjunto de ferramentas que melhoram a eficiência do processo do projeto, reforçando sua coerência e consistência e, também, dando suporte à produção de uma documentação do projeto.

Por ser o foco desse trabalho, iremos detalhar apenas a parte referente à definição dos requisitos no projeto UWA. A figura 2.6 mostra o processo guia para a referida fase, o qual é composto de quatorze passos que serão detalhados mais adiante.

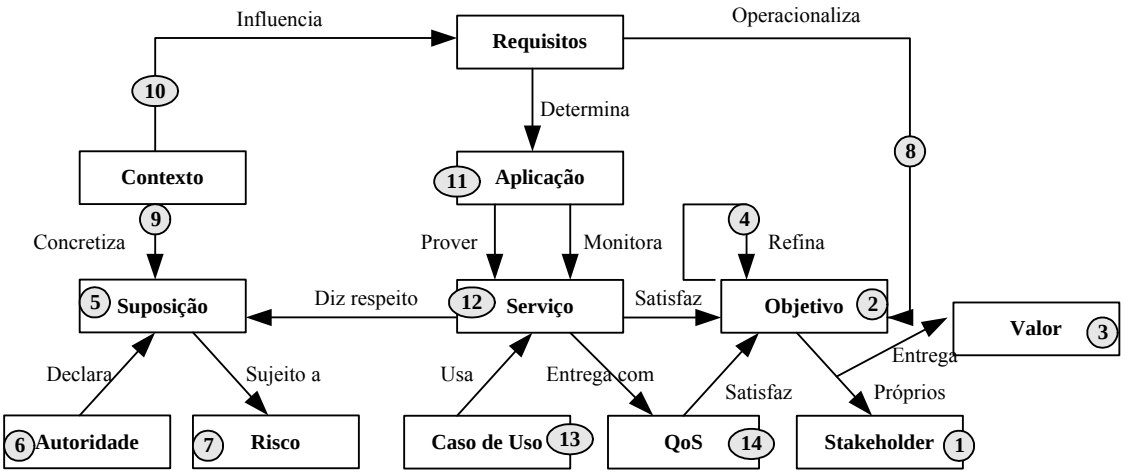

Figura 2.6 - Modelo de processo do projeto UWA [33].

A base dessa abordagem é a engenharia de requisitos orientada a objetivos e a distinção entre um objetivo (metas de alto nível que os *stakeholders* gostariam que a aplicação satisfizesse) e os requisitos (metas de baixo nível que o sistema supostamente conhecerá e que poderão ser diretamente entendidas e realizadas pelo desenvolvedores). Diferentemente das abordagens tradicionais orientadas a objetivos, o método UWA explicitamente captura os requisitos associados ao ambiente e a necessidade de projetar em tempo de execução mecanismos customizados. Ele usa a noção de suposição para denotar as propriedades indicativas do ambiente de operação da aplicação, que esta deve respeitar e que não pode ser alterada pelo sistema. Já a noção de contexto é empregada para denotar a concretização do ambiente e prover uma descrição gerenciável do mesmo.

O processo guia para a elicitação de requisitos na abordagem UWA é composto pelos seguintes passos:

Passo 1 – Identificação dos *stakeholders*: neste passo, todos os *stakeholders* são identificados, uma vez que eles são a maior fonte de requisitos do sistema, isto é, os objetivos que serão conhecidos pelo sistema para serem desenvolvidos.

Passo 2 – Elicitação dos objetivos do sistema: este passo consiste em identificar os requisitos do sistema de uma maneira direcionada a objetivos. Isso requer que todo *stakeholder* especifique o que ele gostaria que o sistema provesse, na sua própria perspectiva.

Passo 3 – Ligar um valor de entrega a cada objetivo: para cada objetivo estabelecido, os *stakeholders* responsáveis são perguntados sobre o estado de valor, isto é, o nível de benefício que eles ou a organização poderiam obter a partir da realização do objetivo. Tais informações serão particularmente úteis para o refinamento dos objetivos alternativos e resolução de conflitos.

Passo 4 – Refinamento dos objetivos: neste passo, os objetivos iniciais e de alto nível do sistema são refinados em objetivos mais detalhados e concretos para poderem ser imediatamente operacionalizados e realizados.

Passo 5 – Documentação das suposições: este passo envolve a identificação das suposições (propriedades do ambiente de operacionalização do sistema). Esse é o caso também das suposições que geralmente são especificadas quando os objetivos são refinados e esclarecidos.

Passo 6 – Identificação das autoridades: este passo consiste na identificação das autoridades, isto é, aqueles que estão numa posição capaz de declarar as suposições, podendo incluir os gerentes, especialistas de domínio e alguns *stakeholders*. Essas autoridades poderão ser claramente identificadas para permitir o rastreamento e também os riscos associados às suposições.

Passo 7 – Identificação dos riscos associados com as suposições: as suposições podem ser estabelecidas com menos do total de confiança, e, por esta razão, podem ser sujeitas a mudanças, ou seus *status* desconhecidos. Conseqüentemente, tais riscos necessitam ser especificados quando as suposições relevantes forem estabelecidas.

Passo 8 – Identificação dos critérios de operacionalização: um dos elementos inéditos dessa *framework* é o fato de que os requisitos podem ser operacionalizados em tempo de execução. Assim, sempre que uma mudança no ambiente é detectada, novos requisitos são operacionalizados a partir dos objetivos do sistema, os quais são imutáveis. Dados esses novos requisitos, a aplicação pode mudar por essa razão e, assim, produzir um novo serviço.

Passo 9 – Identificação das técnicas de concretização: este passo envolve o desenvolvimento de técnicas adequadas para monitoramento das propriedades do ambiente, as quais estão em constante mudança, e tais mudanças influenciam tanto os requisitos quanto os serviços providos.

Passo 10 – Identificação das técnicas de derivação dos requisitos: nesta fase os requisitos são estabelecidos como resultado da operacionalização dos objetivos, os quais estarão sujeitos às informações do contexto.

Passo 11 – Projeto da aplicação: este passo consiste em especificar um número fixo de variações pré-determinadas que são escolhidas em tempo de execução, baseados na mudança de contexto e que comporão a parte variável da aplicação (parte em que se faz a personalização).

Passos Restantes 12, 13, 14 – servem para medir e validar os serviços entregues aos usuários.

Como pode ser evidenciado pelo exposto nessa seção, observa-se que as idéias de ligar um valor de entrega aos objetivos e centrar o processo de elicitação dos objetivos a partir dos *stakeholders* podem ser particularmente úteis para resolver conflitos e também permitir o rastreamento dos requisitos.

Por outro lado, o grande número de passos a ser executados e a falta de técnicas e estratégias que ajudam a conduzir o processo de elicitação de requisitos tornam-no difícil de ser executado.

2.2.3 Método GBRAM

Em [4, 5, 7] é proposto o método *Goal-Based Requeriments Analysis Method* (GBRAM) o qual pressupõe que os objetivos não tenham sido previamente documentados ou explicitamente elicitados a partir dos *stakeholders*. Dessa forma, caberá ao analista examinar todas as fontes de informações disponíveis como: fluxos de processos ou de informações, transcrições de entrevistas com os *stakeholders*, declarações textuais de necessidades, etc, para determinar os objetivos do sistema a ser desenvolvido.

O GBRAM está dividido em duas fases: análise e refinamento de objetivos, e, como saída, produz o documento de requisitos de software (DRS), o qual visa tanto suportar a evolução e validação dos requisitos quanto prover uma comunicação não ambígua entre os *stakeholders*. Na figura 2.7 estão detalhadas as atividades que constituem cada uma das fases do GBRAM.

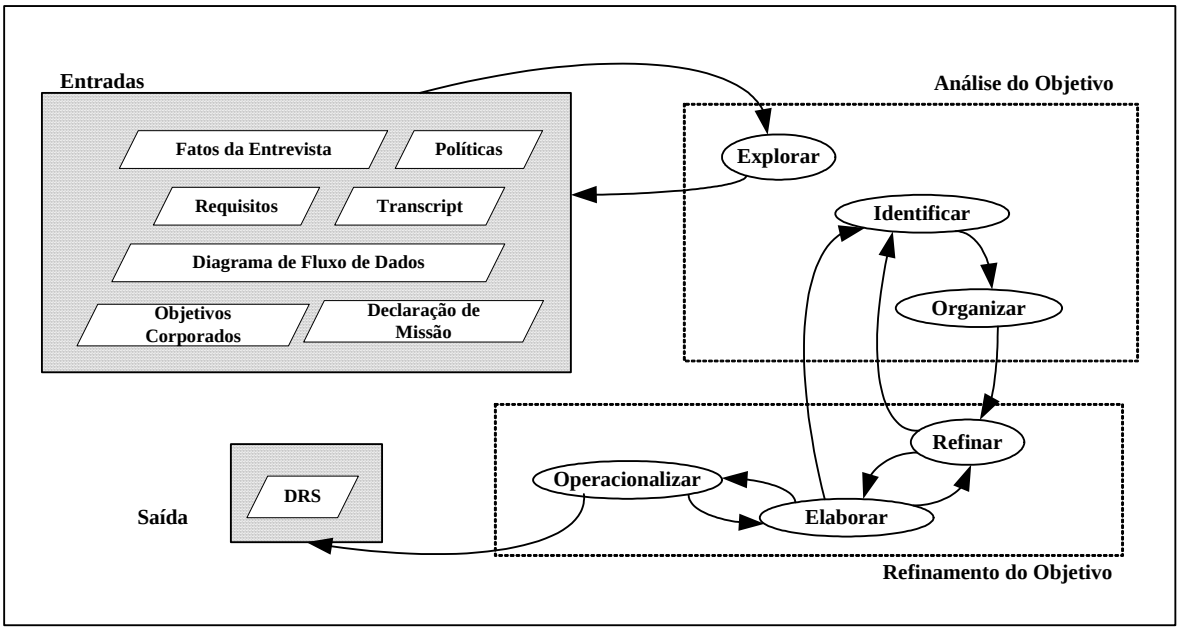

Figura 2.7 - Atividades do método GBRAM [5].

A fase de análise de objetivos apresenta as seguintes atividades: exploração da documentação existente para identificação dos objetivos iniciais; identificação dos objetivos, *stakeholders* e seus agentes responsáveis; e organização dos objetivos de acordo com as relações de dependência (quando um objetivo depende de outro para ser realizado) e classificação destes conforme as condições alvo (situações a partir das quais é possível classificar os objetivos em: manutenção e realização).

As técnicas providas pelo método para suportar a fase de identificação dos objetivos são: técnicas dirigidas a perguntas, onde cada declaração (ou fragmento da informação) é analisada e questionada através de perguntas do tipo: “Quais objetivos este fragmento da declaração exemplifica?”, “Quais objetivos esta declaração bloqueia ou obstrui?”; e a da localização de palavras de ação, a qual considera todas as palavras de ação ou determinados tipos de verbos, como por exemplo: alocar, realizar, executar, satisfazer, arranjar, melhorar, garantir, trilhar, entre outros, como meios para especificar objetivos.

As perguntas utilizadas para identificação dos *stakeholders* são: “Quem ou o que é reivindicado neste objetivo?”, “Quem ou o que se ganha ou perde com a realização ou prevenção desse objetivo?”. Já a pergunta feita para determinar os agentes (responsáveis pela realização ou satisfação dos objetivos dentro de uma organização ou sistema) é: “Quem ou quais agentes [são/deveriam ser/poderiam ser] os responsáveis por este objetivo?”.

Após a identificação dos objetivos, *stakeholders* e agentes, é feita à classificação dos objetivos de acordo com as condições alvo e também estabelecidas as relações de dependências existentes entre os mesmos. No GBRAM, os objetivos são classificados em objetivos de manutenção e de realização.

Os objetivos de manutenção são objetivos que devem ser satisfeitos enquanto sua condição alvo estiver presente (for verdadeira). Eles tendem a ser operacionalizados como ações ou restrições que impedem a ocorrência de certos estados e podem ser identificados através de perguntas como: “Este objetivo garante que alguma condição seja mantida verdadeira por todas as outras operacionalizações de objetivos?”, “Este objetivo afeta decisões em vários níveis dentro da organização?” e “Este objetivo requer um estado contínuo dentro do sistema?”. Além dessas perguntas, outro auxílio para descobrir tais objetivos consiste em utilizar palavras-chave como “prover” e “fornecer”, entre outras.

Os objetivos de realização mapeiam para ações que ocorrem no sistema e auxiliam na identificação dos requisitos funcionais necessários à satisfação dos *stakeholders* e clientes. Para descobrí-los pode-se utilizar questionamentos como: “A realização deste objetivo depende da realização de outro objetivo?” e “A realização de outro objetivo depende da conclusão deste objetivo?”.

Já as relações de dependência servem para ordenar os objetivos e existem entre pares destes, significando que um determinado objetivo é dependente de outro para sua realização,

podendo ser de dois tipos: dependência de precedência, a qual define que um objetivo G1 deve ser realizado antes do objetivo G2, sendo expresso por G1 < G2; e a dependência de contrato, onde para dois objetivos G1 e G2, o objetivo G2 deve ser realizado somente se o objetivo G1 ocorrer, sendo expresso por G1 → G2.

As atividades da fase de refinamento dos objetivos são: refinamento de um conjunto de objetivos com a diminuição do tamanho do conjunto (de objetivos), elaboração de cenários para descobrir os objetivos e requisitos escondidos, e operacionalização dos objetivos em requisitos operacionais.

Na etapa de elaboração dos objetivos são especificados os obstáculos a estes, levando-se em consideração as possíveis maneiras de falhas dos mesmos e como eles podem ser bloqueados, facilitando, assim, a antecipação dos casos excepcionais. Para identificar os obstáculos são utilizadas perguntas, tais como: "De quais outros objetivos ou condições este objetivo depende?", "Pode o agente responsável pela falha do objetivo realizá-lo?", "Pode a falha de outro objetivo causar completamente o bloqueio deste?", "Se este objetivo é bloqueado, quais são as conseqüências?", etc.

Após a especificação dos obstáculos aos objetivos, os analistas irão considerar os cenários possíveis para cada obstáculo. Para tanto, utilizam perguntas do tipo: "O que acontece se este objetivo não for executado?", "Por que este objetivo não foi executado?", "Quais circunstâncias sob a qual este objetivo ocorre?" e "Por que este obstáculo ocorre?".

Uma vez identificados os cenários, os objetivos devem ser operacionalizados e traduzidos em expressões de linguagem natural de requisitos no DRS. Assim, os objetivos operacionalizados, *stakeholders*, agentes responsáveis, restrições, obstáculos e cenários são mapeados em ações firmadas em um conjunto de esquemas-objetivos (modelos que especificam

os relacionamentos entre os objetivos e agentes em termos de eventos que causam uma mudança de estado).

Assim, o conjunto de esquemas-objetivos será mapeado para o DRS, incorporando todas as informações adquiridas durante as fases de análise e elaboração dos objetivos.

2.2.4 Resumo das abordagens selecionadas que utilizam objetivos como forma de elicitar e modelar requisitos

Esta seção apresenta um resumo das principais características das abordagens descritas nas seções anteriores, conforme mostrado nas tabelas 2.2, 2.3 e 2.4.

Como critérios de resumo foram escolhidas as fases do método, as características mais importantes, os principais conceitos em que os mesmos se baseiam e, também, os produtos obtidos como resultado de cada um deles.

Método	**Fases**	**Conceitos utilizados**	**Produtos obtidos**	**Características mais importantes**	**Limitações**
Metamodelo KAOS	1 – Adquirir estrutura do objetivo e identificar objetos relacionados 2 – Identificar agentes potenciais e suas capacidades 3 – Operacionalizar objetivos em restrições 4 – Refinar objetivos e ações 5 – Derivar ações e objetos para garantir restrições 6 – Identificar responsabilidades alternativas 7 – Associar ações e agentes responsáveis	• Objeto • Entidade • Evento • Ação • Ações *Inspect* • Ações *Modify* • Agente • Capacidade • Execução • Objetivos de sistema • Objetivos particulares • Restrições	Metamodelo KAOS	• Envolve três níveis de modelagem: nível meta, nível domínio e nível instância. • Possui um conjunto de estratégias para elaborar modelos de requisitos nesta estrutura • O metamodelo KAOS consiste em um modelo conceitual para adquirir e estruturar modelos de requisitos com uma linguagem de aquisição associada	• Número de passos difíceis de executar durante o processo • Não existe garantia que a aquisição de ações e objetos para um dado sistema irão corresponder as restrições • Dificuldades para gerar a estrutura objetivo

Tabela 2.2 – Resumo das principais características do Metamodelo KAOS

Método	Fases	Conceitos utilizados	Produtos obtidos	Características mais importantes	Limitações
Projeto UWA	1 – Identificação dos *Stakeholders* 2 – Elicitação dos objetivos do sistema 3 – Ligar um valor de entrega a cada objetivo 4 – Refinamento dos objetivos 5 – Documentação das suposições 6 – Identificação das autoridades 7 – Identificação dos riscos associados com as suposições 8 – Identificação dos critérios de operacionalização 9 – Identificação das técnicas de concretização 10 – Identificação das técnicas de derivação dos requisitos 11– Projeto da aplicação Passos Restantes 12, 13, 14 – servem para medir e validar os serviços entregues aos usuários.	• *Stakeholders* • Objetivos • Valor • Suposições • Contexto • Autoridades • Riscos • Concretização • Casos de Uso • Serviços			• Grande número de passos a ser executados • Faltas de técnicas e estratégias que ajudem a conduzir e formalizar o processo de elicitação de requisitos

Tabela 2.3 – Resumo das principais características do projeto UWA

Método	**Fases**	**Conceitos utilizados**	**Produtos obtidos**	**Características mais importantes**	**Limitações**
Método GBRAM	1 – Análise de objetivos • Exploração da documentação existente para identificação inicial dos objetivos • Identificação dos objetivos, stakeholders e seus agentes responsáveis. • Organização de objetivos de acordo com as relações de dependência e classificação dos objetivos 2 – Refinamento de Objetivos • Refinamento de um conjunto de objetivos com a diminuição do tamanho deste • Elaboração de cenários para descobrir objetivos e requisitos escondidos • Operacionalização dos objetivos em requisitos operacionais	• Objetivos de realização e de manutenção • *Stakeholders* • Agentes responsáveis • Relações de Dependência • Restrições • Cenários • Obstáculos dos objetivos	DRS (documento de requisitos de software)	• Possui duas fases principais: Análise e refinamento de objetivos • Classifica os objetivos em: objetivos de manutenção e de realização • Estabelece dois tipos de relações de dependência entre os objetivos: precedência e contrato	• Falta uma melhor sistematização na identificação dos objetivos iniciais

Tabela 2.4 – Resumo das principais características do método GBRAM

2.3 Comércio Eletrônico e suas modalidades

2.3.1 Modelos de comércio eletrônico

Comércio eletrônico pode ser definido como qualquer forma de transação de negócio na qual as partes envolvidas interagem eletronicamente, ao invés de compras físicas ou contato físico direto. Ou, numa definição mais direcionada à Ciência da Computação, segundo [19], comércio eletrônico é um conjunto de técnicas e tecnologias computacionais utilizadas para facilitar e executar transações comerciais de bens e serviços físicos ou virtuais.

O comércio eletrônico reúne três características que são consideradas decisivas para a sua exploração comercial: é fácil de usar, graças a sua interface de *browser;* tem alcance global, porque utiliza o *backbone* da rede mundial; e oferece um nível de segurança que garante confiabilidade das transações.

A seguir, serão apresentados os principais modelos de comércio eletrônico [29], apresentando as suas características, serviços oferecidos e tecnologia aplicada.

2.3.1.1 E-Business

O *e-business* pode ser considerado como qualquer processo de negócio que uma organização conduz através de uma rede de computadores. Em outras palavras, consiste em uma forma de fazer negócios onde uma organização mantém uma relação comercial com fornecedores, clientes e funcionários pela *Internet*.

Essa aplicação tem na retaguarda o suporte de um banco de dados e de um sistema de gestão empresarial que, juntos, formam as bases para que seja realizada a chamada interação dinâmica das informações, ou seja, a manipulação, o acesso, a atualização e a reposição dos dados essenciais para uma relação comercial por todos os componentes da cadeia produtiva envolvidos.

2.3.1.2 E-Commerce

O *e-commerce* (comércio eletrônico) é qualquer transação realizada através de uma rede de computadores, que envolva a transferência de propriedade ou direitos de uso ou serviços. As transações ocorrem dentro dos processos de *e-business* e são completadas quando há a concordância entre o comprador e o vendedor para transferir a propriedade ou direitos de uso ou serviços, inclusive as transações completadas com custo zero, como por exemplo, um *software* grátis para *download.*

Apesar de muitas vezes serem confundidos, o *e-commerce* difere do *e-business*, pois enquanto o último não envolve necessariamente uma transação comercial, é um tipo mais abrangente de negócio eletrônico; o outro deve envolver uma transação comercial, como um processo de compra e venda, por exemplo. Assim, embora sejam próximas, é comum o erro de utilizar essas duas expressões para o mesmo significado.

2.3.1.3 Business-to-Business

O modelo *business-to-business*, também conhecido como B2B, tem o foco corporativo e está relacionado com a interação comercial entre duas organizações na concretização de alguma transação de compra ou venda.

Uma transação nesta categoria de negócios é realizada de forma *on-line* e *real time,* sendo que as informações das empresas são trocadas e atualizadas nos respectivos bancos de dados para a tomada de decisões e início de processos administrativos.

Um exemplo típico dessa modalidade são os *sites* de leilões, onde há um mecanismo de formação de preços e designação de vencedores. O condutor do leilão pode ser comprador, chamados de leilões reversos, ou vendedor.

2.3.1.4 Business-to-Consumer

O modelo *business-to-consumer*, B2C, está, a maior parte das vezes, relacionado com esquemas de vendas ou estabelecimento de lojas virtuais. A demanda por esse tipo de solução está crescendo de forma assustadora em todo mundo, devido à facilidade de localização do item a ser adquirido, facilidade de obtenção de informações técnicas e também comodidade e liberdade que o processo dá a quem está exercendo a compra.

Para a empresa que está vendendo o produto, há vantagens como o baixo custo de manutenção de uma loja virtual, o baixo estoque de mercadorias, o alcance que a solução tem e a disponibilidade de atendimento.

2.3.1.5 Business-to-Employment

O *business-to-employment*, B2E, é o modelo em que o foco do negócio é o empregado e não mais o cliente, como no B2C, ou outros negócios, como o B2B. Esse modelo foi concebido pelas pessoas que trabalhavam com tecnologia da informação e que tinham como objetivo fazer com que seus negócios conseguissem atrair e reter uma equipe qualificada em um mercado competitivo, através de táticas de recrutamento eficientes, benefícios e oportunidades de educação, flexibilidade de horários, bônus e estratégias de melhorias para os empregados.

Mais especificamente, o termo B2E é freqüentemente utilizado para se referir ao portal B2E, algumas vezes chamado de portal pessoal, no qual existe uma página padrão para todos que acessam de dentro da organização. O portal B2E é algumas vezes considerado sinônimo de *intranet,* porém se diferencia no objetivo, pois este focaliza o desejo do empregado. Assim, ele é projetado para incluir não somente o que o empregado deveria achar em uma *intranet,* mas também qualquer informação pessoal e *links* que o mesmo possa

querer. A intenção é aumentar não só a eficiência, como também a satisfação do empregado, e dar sentido de comunidade dentro da organização.

2.4 Segurança e Privacidade

Com o advento das práticas de comércio eletrônico, a segurança da informação é, sem dúvida, uma das grandes preocupações das empresas sintonizadas com o seu tempo. Porém, a habilidade de determinar onde o negócio necessita de segurança e quais as características de segurança são apropriadas, dado um ambiente organizacional, é vital para o desenvolvimento de aplicações de comércio eletrônico.

Muitos são os motivos que levam uma organização a proteger as suas informações. Em primeiro lugar, deve-se considerar que criar, encontrar ou armazenar informações custa dinheiro, portanto a sua perda resulta prejuízo. Segundo, a informação é importante para a organização e seus negócios, assim como também é importante para os seus concorrentes, o que a torna alvo preferido para sabotadores, espiões industriais e vários tipos de golpistas ou *hackers.*

Segundo [6], reduzir ameaças a dados sensíveis é o foco de muitos estudos de métodos endereçados a prover melhor segurança para a privacidade de dados. Contudo, um balanço entre segurança e acessibilidade necessária para as operações normais do negócio deve ser considerado. Dessa forma, muitas organizações estão cientes do problema do acesso não autorizado aos dados pessoais, porém poucas tem estabelecido um programa efetivo de segurança para seus sistemas.

Assim, de acordo com [28], a segurança da informação, conforme mostrado na figura 2.8, tem como objetivo a preservação de cinco princípios básicos pelos quais se norteia a implementação desta prática:

- ✓ Confiabilidade – Garante a proteção da informação de acordo com o grau de sigilo de seu conteúdo, visando à limitação de seu acesso e uso apenas às pessoas para quem elas são destinadas;
- ✓ Integridade – Garante que o conteúdo da informação seja mantido na mesma condição em que foi disponibilizado, visando protegê-la contra alterações indevidas, intencionais ou acidentais;
- ✓ Disponibilidade – Garante que uma informação estará disponível para acesso no momento desejado;
- ✓ Autenticidade – Garante a identidade de quem está enviando a mensagem;
- ✓ Não Repúdio – Previne que alguém negue o envio e/ou recebimento de uma mensagem.

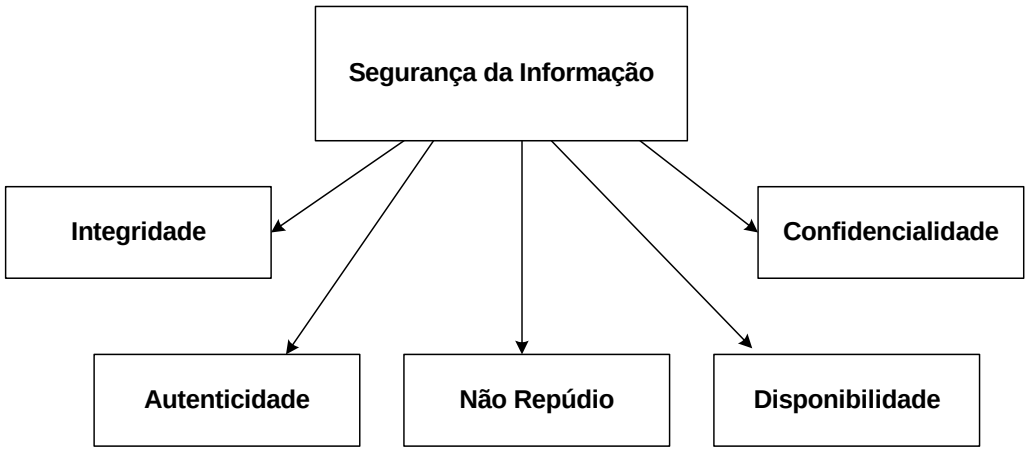

Figura 2.8 - Pilares da Segurança da Informação [28].

Atualmente, tem-se observado que as organizações estão extremamente preocupadas com a segurança nos sistemas de informação e redes de computadores, uma vez que são utilizadas variedades de fontes de ameaças como fraudes eletrônicas, espionagem, sabotagem, entre outras. E esses tipos de ameaças à segurança podem acarretar enormes prejuízos aos negócios.

Dessa forma, faz-se necessário garantir a confiabilidade e segurança de suas transações e combater os ataques causados por vírus, *hackers*, e ataques de "*denial of service (DoS)*", que estão se tornando cada vez mais comuns, mais ambiciosos e incrivelmente mais sofisticados.

Por outro lado, privacidade da informação não é um conceito de fácil definição, mas é geralmente definido como um direito moral ou legal, que descreve os interesses individuais que têm sustentação no espaço pessoal e livre de interferência de outras pessoas ou organizações [13]. Assim, privacidade afeta não só consumidores de comércio eletrônico como também usuários ou *stakeholders*, em outros domínios.

A privacidade da informação é impactada pelas funções organizacionais, tais como: comércio eletrônico, gerenciamento de banco de dados, técnicas de segurança, telecomunicações, sistemas colaborativos e implementação de sistemas [13]. Por essa razão, é preciso que os desenvolvedores de sistemas de comércio eletrônico estejam cientes desse fato e da necessidade da realização prévia de um plano de privacidade, através da determinação dos requisitos e projeto de software de aplicações de comércio eletrônico.

2.5 Política de Segurança

O primeiro passo na segurança de um sistema de comércio eletrônico é o desenvolvimento de uma política de segurança, a qual visa estabelecer quais usuários podem ser autorizados, como eles podem acessar o sistema e os dados, como os usuários sem autorização terão acesso proibido, e como os dados poderão ser tão bem protegidos dentro da organização, quanto fora dela.

O principal objetivo no desenvolvimento de uma política de segurança é especificar as expectativas organizacionais propostas para o uso do sistema e definir os procedimentos para prevenir e responder aos eventos de segurança. Além disso, objetivos,

como proteger dados sensíveis de acessos não autorizados e prevenir danos acidentais ou intencionais para *hardware* e *software*, entre outros, também devem ser atendidos.

Ao se falar de política de segurança para comércio eletrônico, não se pode esquecer de levar em consideração os riscos que são comuns a esse ambiente, uma vez que estão sujeitos, constantemente, à adição de vulnerabilidades. Dessa maneira, um risco ocorre quando uma ameaça explora uma vulnerabilidade para causar prejuízo ao sistema. Portanto, uma política de segurança deve prover a base para a implementação de controles de segurança, para reduzir riscos introduzidos por vulnerabilidades.

O conteúdo de uma política de segurança para aplicações de comércio eletrônico geralmente pode consistir de várias sub-políticas, dentre elas: política de identificação e autenticação, política de acesso remoto, política de criptografia, política de uso apropriado, política de senhas, política de integridade/segurança, política de privacidade e etc.

É importante ressaltar que, apesar da política de privacidade ser uma sub-política da política de segurança, para efeito de organização, neste trabalho, elas serão tratadas de maneira separadas.

2.6 Política de Privacidade

Uma política de privacidade pode ser definida como uma descrição compreensiva das práticas de um *website*, a qual deverá ser colocada em um local do *site* que pode ser facilmente acessado [13]. Dessa forma, toda organização envolvida em transações de comércio eletrônico tem a responsabilidade de adotar e implementar uma política para a proteção da privacidade das informações identificáveis individualmente.

Assim, o mínimo que se pode esperar de uma política de privacidade é que a mesma enderece os seguintes aspectos:

- ✓ Notícia/Conhecimento: refere-se às informações das práticas organizacionais que os consumidores deverão conhecer e/ou dar ciência, antes que uma informação seja coletada dele, como por exemplo: quais são as naturezas dos dados coletados, como estes são armazenados, quais os dados que são obrigatórios serem informados, etc;
- ✓ Escolha/Consentimento: diz respeito às práticas que visam garantir o direito do consumidor de decidir quais informações coletadas sobre ele poderão ser usadas e distribuídas a terceiros ou parceiros de um determinado *site*;
- ✓ Não Repudiação: descreverá os meios utilizados pelo *site* para garantir que alguém negue o envio e/ou recebimento de uma mensagem;
- ✓ Acesso/Participação: estabelece a maneira como os usuários poderão acessar as informações coletadas sobre os mesmos, para fazer correções, adições de dados, etc.

É importante destacar que na formulação de uma política de privacidade deve-se levar em consideração as vulnerabilidades que são comuns a ambientes de comércio eletrônico. Assim, a formulação de uma política de privacidade também é um processo progressivo e iterativo que requer constantemente atualizações, devido à natureza dinâmica desse tipo de ambiente.

2.7 GBRAM instanciado para o desenvolvimento de requisitos e políticas de sistemas de comércio eletrônico

Em [6] é apresentado uma instanciação do método GBRAM para o desenvolvimento de políticas e requisitos de sistemas de comércio eletrônico, conforme ilustrado pela figura 2.9.

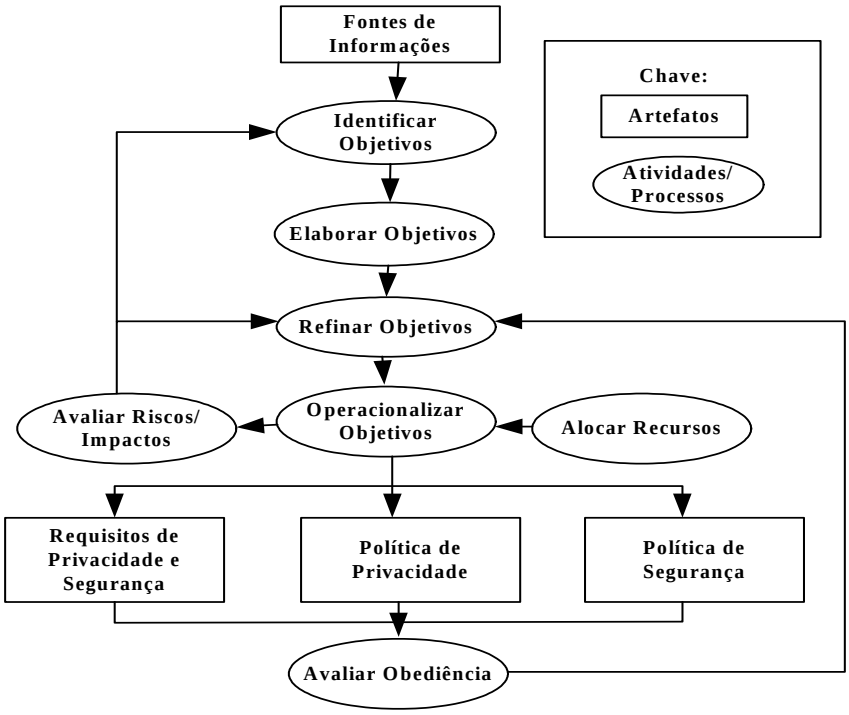

Figura 2.9 – GBRAM instanciado para formação de políticas [6].

Na instanciação do método, algumas fases do GBRAM foram mantidas, como: identificação, elaboração, refinamento e operacionalização dos objetivos. E acrescidas as fases de alocação de recursos, avaliação de riscos e impactos, e avaliação de obediência.

A fase de avaliação de riscos e impactos é baseado no *Policy Framework for Interpreting Risk in e-Commerce Security* (PFIRES) [21]. O PFIRES é um *framework* para interpretação de riscos em políticas de segurança nas aplicações de comércio eletrônico, o qual usa um modelo do ciclo de vida que consiste das seguintes fases: avaliação, planejamento, entrega e operação. Cada fase do modelo é marcada por um critério de saída, que deve ser conhecido antes da transição para a próxima fase. Assim, a avaliação de riscos é construída dentro desse ciclo de vida e as mudanças nas políticas são classificadas ao longo de uma "contínua mudança".

A fase de avaliação de obediência segue o *House of Quality* (HoQ) [15], que é uma abordagem para analisar e documentar grandes coleções de requisitos, a qual consiste de uma tabela onde a coluna da esquerda lista um conjunto de declarações de políticas da

organização, enquanto a linha do topo lista um conjunto de requisitos operacionalizados, cada um na sua própria coluna.

A tabela HoQ indica os relacionamentos que existem entre os requisitos e as políticas específicas. Os relacionamentos de cooperação são marcados com um "✓" e os relacionamentos de conflitos com um "×". Quando um conflito aparece entre políticas existentes e objetivos, os objetivos e/ou políticas são refinados, conforme mostrado na figura 2.9.

Esse capítulo procurou descrever algumas abordagens que utilizam objetivos e cenários na fase da engenharia de requisitos, e também os principais conceitos referentes a comércio eletrônico, suas modalidades e políticas de segurança e privacidade. O capítulo três apresentará um método que propõe a integração da abordagem UWA com a instanciação do método GBRAM para o desenvolvimento de políticas, cujo objetivo é unir métodos da engenharia de requisitos, em especial a utilização de objetivos, para o desenvolvimento de aplicações de comércio eletrônico, em obediência às políticas de segurança e privacidade existentes em uma organização.

3 MÉTODO PROPOSTO PARA ELICITAÇÃO DE REQUISITOS

O método proposto para elicitação de requisitos combina duas técnicas propostas pela abordagem UWA com alguns princípios do método GBRAM, instanciado para o desenvolvimento de políticas de segurança e privacidade.

Do projeto UWA, foram utilizados dois conceitos: a centralização da descoberta dos objetivos a partir dos *stakeholders* e a atribuição de valores de entrega a cada objetivo capturado.

Quanto ao GBRAM instanciado para o desenvolvimento de políticas de segurança e privacidade, foram empregados os seguintes princípios: avaliação de riscos e avaliação de obediência com as políticas.

O método resultante tem como finalidade elicitar requisitos para aplicações de comércio eletrônico. Dessa forma, para cada requisito capturado, será feita a avaliação de obediência, cuja finalidade é garantir que as políticas de segurança e privacidade existentes (ou criadas) em um *site* continuem consistentes, mesmo com a adoção de novas funcionalidades a este.

3.1. Fases do Método para elicitação de requisitos

O método proposto [25] está dividido em oito fases: identificar os stakeholders, identificar objetivos, atribuir valores aos objetivos, refinar objetivos, operacionalizar objetivos, avaliar riscos, criar políticas de segurança e privacidade (caso não existam) e avaliar obediência.

A figura 3.1 mostra o método proposto, onde os retângulos representam as fontes de informação, e as elipses as fases que constituem o mesmo.

A seguir serão explicadas cada uma das fases do método, as quais estão numeradas

de acordo com a seqüência em que devem ser executadas.

1 - Identificação dos *stakeholders*

Os *stakeholders* são definidos como os indivíduos que têm algum interesse no sistema, podendo incluir os usuários do sistema, proprietários, clientes, desenvolvedores, especialistas de domínio, gerenciadores, etc. Assim, a identificação e a documentação dos *stakeholders* é a chave para o rastreamento de requisitos, já que esse rastreamento é crítico, tanto para validar quanto para resolver conflitos entre os mesmos.

Porém, se os *stakeholders* não forem apropriadamente identificados, características do sistema não poderão ser descobertas, já que eles são as maiores fontes de requisitos do sistema. Por essa razão, esse método é iniciado pela identificação dos mesmos.

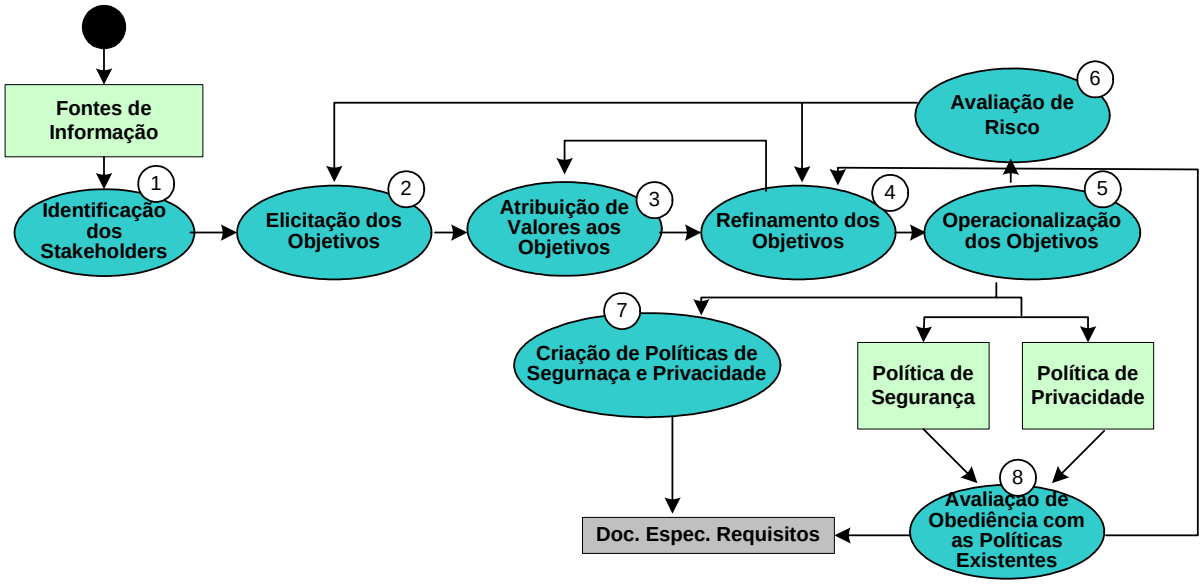

Figura 3.1 – Atividades do método proposto

A identificação dos *stakeholders* pode ser feita a partir da exploração da documentação existente ou através de perguntas como: “quem ou o que reivindica algum interesse no sistema?”, “quem ou o que se ganha ou perde com o desenvolvimento do sistema?”.

O método propõe, para cada *stakeholder* identificado, que o mesmo seja documentado da seguinte maneira: número do *stakeholder*, nome e tarefa que executa.

Para facilitar o entendimento do método proposto, vamos considerar um exemplo de uma loja de perfumes *on-line*. Com a aplicação das estratégias acima para a identificação dos *stakeholders* teríamos, por exemplo:

Número do *Stakeholder*: S1
Nome do *Stakeholder*: Fornecedor
Tarefa que executa: Fornece perfumes

Número do *Stakeholder*: S2
Nome do *Stakeholder*: Cliente
Tarefa que executa: Realiza compras

Um importante ponto a ser destacado é garantir que os *stakeholders* identificados estejam entusiasmados, dispostos e envolvidos no processo de captura de requisitos do sistema e, principalmente, tenham tempo disponível para realizar essa tarefa.

2 - Identificar os objetivos

Este passo consiste em identificar os requisitos do sistema de uma maneira direcionada a objetivos, uma vez que estes expressam as metas ou os desejos que os *stakeholders* gostariam que o sistema satisfizesse. Isso requer que cada *stakeholder* identificado na fase anterior estabeleça o que ele gostaria que o sistema provesse para si, na sua própria perspectiva, cabendo aos analistas de sistemas a coordenação dessas atividades, com o intuito de extrair o maior número possível de objetivos.

De acordo com [20], uma variedade de técnicas pode ser empregada para realizar a tarefa de captura de objetivos a partir dos *stakeholders*. Entre elas, podem-se destacar:

- ✓ Entrevistas: os analistas elaboram questionários com perguntas específicas para os *stakeholders*, com a finalidade de definir os objetivos gerais e

restrições que o software deverá ter. É importante ressaltar que essas entrevistas deverão ser feitas de forma clara e objetiva, evitando, assim, interpretações errôneas;

- ✓ Grupos de discussão: os analistas promovem, junto com um grupo de *stakeholders*, identificados anteriormente, "mesas-redondas" para incentivar os *stakeholders* a expressar, de maneira mais espontânea, todos os seus desejos com relação ao sistema a ser desenvolvido;
- ✓ *Story-boarding*: o analista utiliza-se de quadros com imagens que descrevam o que está sendo expresso pelos *stakeholders,* como forma de auxiliar no entendimento e melhorar a comunicação entre eles, possibilitando, assim, a extração de um número maior de objetivos.

Convém ressaltar que essas técnicas são complementares e podem ser ou não usadas em conjunto. Além disso, o método aqui proposto, não impede que outras técnicas, além das citadas acima, sejam empregadas, ficando a cargo do analista utilizar aquela que lhe for mais conveniente.

Para ilustrar essa fase, aplicando as técnicas descritas acima, a partir do *stakeholder* S2, foram encontrados os seguintes objetivos: G1 – Realizar compras, G2 – Escolher formas de pagamento, G3 – Montar carrinho de compras, G4 – Escolher endereço de entrega, G5 – Devolver produtos, etc.

Para auxiliar na descrição dos objetivos, o método proposto utilizará os cenários, os quais possibilitarão um detalhamento maior dos objetivos especificados e também ajudarão conduzir melhor a fase posterior, de operacionalização desses objetivos. Porém, os cenários descritos aqui não descreverão apenas os comportamentos normais, mas também os excepcionais ou variacionais [10].

Um cenário normal consiste em um conjunto de ações executadas de maneira que

o objetivo seja realizado. Caso existam situações não normais, onde mesmo assim o objetivo é realizado, está-se diante de cenários variacionais. Os cenários excepcionais referem-se a um conjunto de ações onde o objetivo não é realizado.

A nomenclatura utilizada para a descrição dos cenários será iniciada pela letra "C" seguida por um número que representa o tipo de comportamento do mesmo. Os comportamentos normais receberão o número do objetivo a qual o cenário pertence, a partir deste será incrementado em 1 para os demais cenários descobertos. Por exemplo, o cenário normal para o objetivo G3 será C3 e os excepcionais ou variacionais, se existirem, serão C3.1, 3.2, e assim por diante.

Dessa forma, para o objetivo G1, poder-se-ão ter alguns cenários, dentre eles:

Cenário: C1
Nome: Realizar compras de uma maneira normal
Objetivo: G1
Ator/Agente: Cliente
Pré-Condições: *If* cliente fez o *login* no sistema
If carrinho <> vazio
If endereço de entrega é igual ao constante no cadastro do cliente
Descrição:

1. O cliente faz o *login* no sistema
2. O sistema verifica que existem itens no carrinho de compras
3. O sistema exibe os dados cadastrais do cliente e o endereço de entrega, igual ao existente no cadastro do cliente
4. O sistema exibe um botão de confirmação e outro de alterar endereço de entrega do produto
5. O cliente confirma o endereço de entrega
6. O sistema mostra o valor total do pedido
7. O sistema solicita a escolha da forma de pagamento
8. O cliente escolhe a forma de pagamento
9. O sistema exibe uma mensagem "Entrega de pedido condicionado a confirmação do pagamento"

10. O sistema gera um número de pedido com situação “pendente”

Pós-Condições: *Do* pedido realizado com sucesso

Do situação do pedido = “pendente”

3 - Atribuir um valor a cada objetivo

Nesta fase, para cada objetivo identificado anteriormente, os *stakeholders* responsáveis são perguntados sobre os estados de valores que eles ou a organização poderiam obter a partir da realização do mesmo. Dessa forma, o valor que um *stakeholder* dá a um objetivo representa o nível de benefício que o mesmo irá obter quando da realização deste. Assim, o valor dado a um objetivo é relativo, uma vez que alguns objetivos podem ser mais valiosos para uns *stakeholders* do que para outros.

O método propõe que essa atribuição de valor seja da seguinte maneira: agrupar, para cada *stakeholder*, todos os objetivos especificados pelo mesmo; e numerar, em uma escala numérica ascendente, ou seja, do menor ao maior valor, todos os objetivos estabelecidos por este.

Seguindo o exemplo anterior, o *stakeholder* S2 especificou os seguintes valores para os objetivos G1, G2, G3, G4 e G5, respectivamente: 5, 3, 1, 2 e 4.

A importância da atribuição de um valor a cada objetivo definido será particularmente útil para resolução de conflitos e refinamento de objetivos.

Finalmente, após ser atribuído um valor a cada objetivo, é necessário que os mesmos sejam ordenados, isto é, se estabeleça a relação de dependência existente entre pares de objetivos. Uma relação de dependência especifica quando um dado objetivo depende de outro para poder ser realizado.

O método mantém as relações de dependências providas pelo método GBRAM, ou seja, a relação de dependência de precedência, a qual define que um objetivo deve ser realizado antes de outro; e a dependência de contrato, onde um objetivo somente é realizado

se um outro objetivo ocorrer, ambas explicadas anteriormente no estado da arte.

No exemplo ilustrativo, foi encontrada a seguinte relação de dependência de precedência: G1 < G5, ou seja, a devolução de um produto deve ser precedida de uma compra.

4 - Refinar objetivos

Após o estabelecimento de valores para cada objetivo e das suas relações de dependência, dá-se início à fase de refinamento de objetivos, a qual consiste em um processo manual realizado pelo analista de sistemas com o propósito de identificar inconsistências, redundâncias, unificar sinônimos e eliminar objetivos duplicados.

Esse passo é necessário porque os objetivos identificados inicialmente tendem a ser gerais e de alto-nível. Por essa razão, eles precisam ser refinados em objetivos mais detalhados e concretos para poderem ser operacionalizados ou realizados.

O foco dessa fase, além de resolver conflitos, é remover os objetivos sinônimos e redundantes, visando determinar quaisquer inconsistências que existam dentro do conjunto de objetivos capturados, para que os mesmos sejam operacionalizados na especificação de requisitos. Além disso, é importante ter-se em mente que o refinamento dos objetivos é usualmente feito para um propósito particular e a partir de uma perspectiva específica, para qual é razoável ser capturada.

Outro ponto a se destacar é a garantia que os sub-objetivos (objetivos derivados) possam suficientemente depurar os objetivos de alto nível e, assim, assegurar que a finalidade global dos objetivos gerais não seja comprometida nem perdida durante essa fase.

É importante ressaltar que, caso um *stakeholder* tenha atribuído valores iguais a dois ou mais objetivos estabelecidos por ele e estes conflitem, o analista deverá voltar ao passo anterior e pedir que o *stakeholder* estabeleça novos valores a tais objetivos, garantindo

assim que o propósito pretendido (resolução de conflitos) pela atribuição de valores não seja comprometido.

Como visto, essa fase pode precisar de vários níveis de refinamento e, por esta razão, incentiva-se o envolvimento dos *stakeholders* mais relevantes como forma de garantir um refinamento mais adequado aos propósitos do sistema.

Por ser um processo manual e envolver tanto *stakeholders* quanto analistas, a utilização das técnicas de grupo de discussão e *story-boarding* pode ser útil nessa fase.

5 - Operacionalizar objetivos

Após o refinamento dos objetivos é necessário que os mesmos sejam traduzidos para os requisitos, ou seja, as metas de baixo nível que o sistema terá que realizar para satisfazer os objetivos identificados nas fases anteriores. Assim, essa tradução consiste em descrever mais detalhadamente cada um dos objetivos e cenários, e mapeá-los em um conjunto de esquemas, que é uma forma de *template* estendido.

O método aqui proposto também utiliza um estilo informal para ajudar a conduzir essa fase, similar ao empregado pelo GBRAM, onde é utilizada uma estratégia dirigida a esquema, a qual é baseada na definição dos esquemas-objetivos, esquemas-casos-de-uso e esquemas-cenários, já descritos anteriormente. Porém, serão feitas algumas adaptações, as quais estão detalhadas abaixo.

O primeiro esquema utilizado será o esquema-objetivo, cuja finalidade é especificar os relacionamentos entre os objetivos e cenários. Para tanto, será especificado um modelo de objetivos incorporando todas as informações adquiridas nas fases anteriores. A sintaxe do esquema para o modelo de objetivos utilizado para definir objetivos consiste de: um número do objetivo, nome, descrição, *stakeholder*, valor, código do cenário e, se existirem, as pré-condições e pós-condições.

A seguir, será usado um esquema para o modelo de cenários e ações utilizado para especificar cenários e ações. A sintaxe do esquema consiste em: código do cenário, descrição, ator/agente, pré-condição e/ou pós-condição (se existir) e ação.

Finalmente, o esquema de ação será utilizado como forma de especificar cada ação de um cenário, sendo necessário pelo menos um esquema de ação para cada cenário. Porém, geralmente, existirão para cada cenário vários esquemas de ação, cuja sintaxe de cada cláusula é formada por: ação, tipo, entrada, número de seqüência e código do cenário.

A seguir foram descritos alguns esquemas-objetivos, esquemas-cenários e esquemas-ação, para o exemplo da loja de perfumes *on-line*.

Esquema-Objetivo

Nº do objetivo: G1
Nome: Realizar compras
Descrição: Consiste em comprar perfumes das mais variadas marcas espalhadas pelas diversas seções do *site*
Stakeholder: S2
Valor: 5
Cenários: C1, C1.1 e C1.2

Esquema-Cenário

Código do Cenário: C1.1
Descrição: Realizar compras sem itens no carrinho de compras
Ator/agente: Cliente
Pré-Condições: *If* cliente fez o *login* no sistema
If carrinho = vazio
If endereço de entrega é igual ao constante no cadastro do cliente
Pós-Condições: *Do* pedido realizado com sucesso
Do situação do pedido = "pendente"
Ações:

1. O cliente faz o *login* no sistema
2. O sistema verifica que existem itens no carrinho de compras

3. O sistema exibe a mensagem “Não existem itens no carrinho de compras, pelo menos um item deve ser escolhido”
4. O cliente escolhe um produto e manda adicionar no carrinho de compras
5. O sistema exibe os dados cadastrais do cliente e o endereço de entrega, igual ao existente no cadastro do cliente
6. O sistema exibe um botão de confirmação e outro de alterar endereço de entrega do produto
7. O cliente confirma o endereço de entrega
8. O sistema mostra o valor total do pedido
9. O sistema solicita a escolha da forma de pagamento
10. O cliente escolhe a forma de pagamento
11. O sistema exibe uma mensagem “Entrega de pedido condicionado a confirmação do pagamento”
12. O sistema gera um número de pedido com situação “pendente”

Esquema-Ação

Ação: O sistema verifica se existem itens no carrinho de compras
Tipo: Sistema
Entrada: Código do Produto
Nº de seqüência: 1
Código do Cenário: C.1

6 - Avaliar riscos

Em sistemas de comércio eletrônico, a fase de identificação de riscos é uma das mais críticas, devido ao ambiente dessas aplicações estarem constantemente sujeitos a adição de vulnerabilidades. Dessa forma, o risco ocorre quando uma ameaça explora uma vulnerabilidade para causar prejuízo ao sistema. Assim, para cada objetivo operacionalizado, deve-se identificar os riscos associados ao mesmo, ou seja, estabelecer aquelas ameaças que podem causar danos ao sistema.

Conforme mostra a figura 3.2, essa fase está subdividida em três subfases: identificar ameaças e vulnerabilidades, estimar a probabilidade de ocorrência e escolher uma

estratégia de atenuação de riscos. O método proposto utiliza algumas técnicas contidas em [16, 32], especialmente na primeira e segunda subfase, porém feitas algumas adaptações que serão explicadas abaixo.

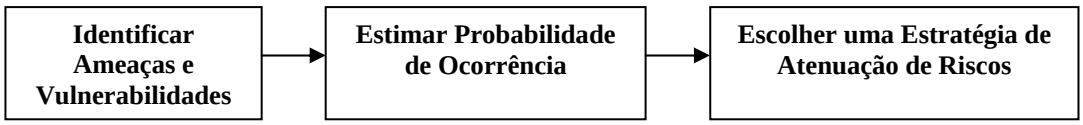

Figura 3.2 - Subfases da avaliação de riscos

A subfase "identificar ameaças e vulnerabilidades" é iniciada com a identificação do par (ameaça, vulnerabilidade) e também das fontes de ameaça, motivação e ações que podem levar a um ataque.

Já a subfase "estimar a probabilidade de ocorrência do risco" também é baseada em [32], onde a probabilidade de ocorrência do risco é classificada em três níveis: alto, quando a fonte de ameaça é altamente motivada e suficientemente capaz, e os controles, inicialmente criados para prevenir as vulnerabilidades, são ineficientes; médio, quando a fonte de ameaça é altamente motivada e suficientemente capaz, porém os controles podem prevenir com sucesso a exploração das vulnerabilidades; e baixo, quando falta motivação ou capacidade à fonte de ameaça, ou os controles estão no lugar para prevenir ou impedir que as vulnerabilidades sejam exploradas.

Finalmente, a subfase "escolher uma estratégia de atenuação de riscos" pode consistir em uma das ações, conforme mostra a figura 3.1: adição de um novo objetivo ou subjetivo para responder ao risco identificado ou um novo refinamento do objetivo, visando adicionar uma restrição para atenuação do mesmo.

Após o término dessa fase, o método proposto pode seguir dois caminhos: criar políticas de segurança e privacidade, caso a organização não tenha definido suas políticas ou avaliar obediência, cujo objetivo é garantir que os requisitos do sistema estejam de acordo com as políticas de segurança e privacidade existentes na organização. As seções seguintes

detalham cada uma dessas fases.

7 - Criar políticas de segurança e privacidade

Esta fase consiste em estabelecer políticas de segurança e privacidade para a organização. Assim, o método proposto inclui essa etapa como forma de garantir que tais políticas sejam criadas. Porém, para organizações com políticas de segurança e privacidade estabelecidas, a execução dessa fase será desnecessária.

Segundo [8], falta, à maior parte das políticas de segurança e privacidade criadas para *websites*, a clareza necessária para que os usuários possam fazer uso proveitoso das mesmas. Esse problema é principalmente devido à natureza monolítica (isto é, documentos muito extensos) e independente de contexto (documentos muito generalistas) de tais políticas.

Para a condução dessa fase, o método sugere a adoção de modelos como forma de estabelecer um meio padrão para construir políticas de maneira mais clara, menos ambígua e contendo, apenas, os aspectos mais relevantes, beneficiando, assim, tanto a organização, quanto os usuários de um *site* de comercio eletrônico.

A seguir, serão apresentados os modelos para a construção de políticas de segurança e privacidade para *sites* de comércio eletrônico, assim como um detalhamento de cada item existente nos mesmos. É importante ressaltar que, para efeito de organização, apesar da política de privacidade ser uma sub-política da política de segurança, serão tratadas de maneira separada nesse trabalho.

A figura 3.3 apresenta o modelo sugerido pelo método para a criação de uma política de segurança para um *site* de comércio eletrônico, cujo objetivo principal é garantir a confiabilidade (propriedades acessíveis somente pelas partes autorizadas), integridade (modificações feitas somente pelas partes e de formas autorizadas) e disponibilidade (propriedades sempre acessíveis às partes autorizadas). Abaixo estão detalhados todos os itens

do modelo.

Cabeçalho

<Título do documento>

<Autor(es)>

<Data da versão do documento>

<Número da versão do documento>

1. Introdução

1.1. Declaração de Propósito

2. Políticas

2.1. Política de identificação e autenticação

2.2. Política de criptografia

2.3. Política de integridade/segurança

2.4. Política de senhas

2.5. Política de uso

3. Apêndice

3.1. Glossário

3.2. Documentos Relacionados

Figura 3.3 - Modelo da Política de Segurança

A primeira parte do modelo é formada por um "Cabeçalho", o qual é composto pelo título do documento, nome(s) do(s) autor(es), data e número da versão da política de segurança.

A seção seguinte refere-se à "Introdução", que é constituída pela declaração de propósito, onde será feita uma breve descrição do porquê da política de segurança ser necessária e importante para o *site*.

Em seguida, a seção "Políticas" está dividida em cinco subseções, a saber: política de identificação e autenticação, de criptografia, de anúncio e educação, de senhas e de uso, as quais estão detalhadas abaixo.

No item referente à política de identificação e autenticação, será especificada a forma como o usuário deverá se identificar para ter acesso às áreas restritas do *site* e também a maneira como o *site* irá autenticar a validade das informações fornecidas pelos mesmos.

Em seguida, o item política de criptografia é destinado à descrição do tipo de tecnologia usada pelo *site* para criptografar as informações fornecidas pelos usuários. Já o item integridade/segurança descreverá os meios utilizados pelo *site* para preservar e garantir que os dados fornecidos não sejam acessados, alterados, modificados ou apagados por indivíduos não autorizados.

O item política de senhas é reservado para que o *site* esclareça seu usuário sobre a melhor maneira de criar uma senha e a importância das trocas constantes das mesmas. Finalmente, o item políticas de uso conterá uma descrição detalhada das regras de uso do *site,* assim como as responsabilidades atribuídas tanto ao usuário quanto à organização.

A seção final, "Apêndice", é constituída de duas subseções: glossário, que é destinado à colocação de qualquer definição ou explicação que poderá ajudar na leitura e no completo entendimento da política de segurança; e documentos relacionados, onde deverão ser listados todos os *links* com informações relevantes que poderão ser úteis para complementar a política de segurança criada.

A figura 3.4 apresenta o modelo sugerido pelo método para a criação de uma política de privacidade para um *site* de comércio eletrônico, cujo objetivo principal é descrever os tipos de informações coletadas pelo *website* e, também, como estas são manipuladas, armazenadas e usadas.

Semelhante ao modelo de segurança, o de privacidade também é iniciado pelo cabeçalho, que é igualmente composto pelo título do documento, nome(s) do(s) autor(es), data e número da versão da política de privacidade.

Cabeçalho

<Título do documento>

<Autor(es)>

<Data da versão do documento>

<Número da versão do documento>

1. **Introdução**
   - 1.1. Declaração de Propósito
2. **Princípios**
   - 2.1. Noticia/Conhecimento
   - 2.2. Escolha/Consentimento
   - 2.3. Não Repudiação
   - 2.4. Acesso/Participação
3. **Apêndice**
   - 3.1. Glossário
   - 3.2. Documentos Relacionados

Figura 3.4 - Modelo da Política de Privacidade**.**

A seção "Introdução" é composta pela declaração de propósito, onde será feita uma breve descrição do porquê da política de privacidade ser necessária e importante para o *site*.

A seção seguinte, "Princípios", irá conter os itens principais que devem estar presentes em uma política de privacidade, tais como: notícia/conhecimento, escolha/consentimento, não repudiação e acesso/participação.

Na parte referente à notícia/conhecimento, será especificada, de maneira detalhada, todas as informações coletadas dos usuários, como estas serão armazenadas e como poderão ser utilizadas pelo *site*. Já o item escolha/consentimento conterá a descrição das práticas, onde será dada a oportunidade ao usuário de escolher aceitar ou não o que está sendo estabelecido pelo *site,* como por exemplo o *download* de arquivos ou *software*, o preenchimento de formulários de pesquisas, etc.

Em seguida, o item não repudiação descreverá os meios utilizados pelo site para garantir que alguém negue o envio e/ou recebimento de uma mensagem. E finalmente, o item acesso/participação, estabelecerá a maneira como os usuários poderão acessar as informações coletadas sobre os mesmos para fazer correções, adições de informações, etc.

A última seção, "Apêndice", assim como no modelo da política de segurança, é constituída de duas subseções: glossário, que é destinado à colocação de qualquer definição ou explicação que poderá ajudar na leitura e no completo entendimento da política de privacidade; e documentos relacionados, onde deverão ser listados todos os *links* com informações relevantes que poderão ser úteis para complementar a política de privacidade criada.

Como dito anteriormente, essa fase só será executada se a organização não tiver suas políticas criadas. Caso contrário, a execução dessa fase tornar-se-á desnecessária.

8 - Avaliar obediência com as políticas existentes

Uma das fases mais críticas e também mais importantes dessa abordagem é a avaliação de obediência, a qual é necessária para garantir obediência entre a especificação de requisitos e as políticas de segurança e privacidade do *site*, minimizando, dessa forma, o risco de inconsistência entre esses requisitos resultantes e as políticas existentes. Assim, o método proposto garante que a especificação de requisitos do sistema, políticas de segurança e privacidade nunca se tornarão obsoletas devido à adoção iterativa dessa atividade.

Conforme mostra a figura 3.5, essa fase está subdividida em três subfases: avaliar as políticas existentes, identificar obediências e contradições, e desenvolver uma estratégia de ação, as quais serão detalhadas a seguir.

A subfase "avaliar as políticas existentes" é uma das tarefas mais trabalhosas e demoradas dessa fase, principalmente se é a primeira vez que se está executando o método.

Porém, é imprescindível que a mesma seja realizada para que possa ser dado continuidade ao processo. Assim, a estratégia proposta para conduzir essa subfase é baseada em HoQ [15], que é uma abordagem para documentar e analisar grandes coleções de requisitos.

À medida que as políticas vão sendo examinadas, vão-se extraindo as declarações de políticas do *site* e preenchendo a coluna da tabela equivalente à política examinada para posterior avaliação de obediência.

Convém ressaltar que, para efeito de organização, é importante que as políticas de segurança e privacidade sejam analisadas separadamente. Em seguida, a partir dos requisitos especificados, vai-se preenchendo a coluna equivalente aos requisitos elicitados.

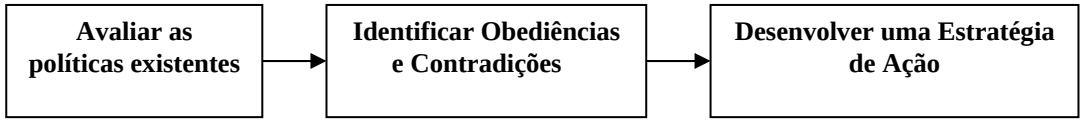

Figura 3.5 - Subfases da Avaliacão de Obediência

Após o término do preenchimento da tabela, é iniciada a subfase de "identificação de obediências e contradições", a qual consiste em estabelecer os relacionamentos existentes entre os requisitos e as políticas do *site*, onde os relacionamentos de cooperação são marcados com "✓" e os de conflitos com um "×". Para tanto, a tabela é examinada e preenchida de acordo com as relações encontradas.

Finalmente, a realização da subfase "desenvolver uma estratégia de ação", conforme ilustrado na figura 3.1, vai depender do tipo de relacionamento encontrado na tabela, podendo seguir um dos dois caminhos: o retorno à fase de refinamento de objetivos, será indispensável sempre que um requisito de um objetivo, que esteja sendo operacionalizado, conflite ou então seja redundante com as políticas existentes; e a atualização das políticas existentes, será imprescindível sempre que as mesmas não atendam os requisitos especificados.

Por fim, o método proposto sugere a adoção de um modelo para o documento de

especificação de requisitos, cujo objetivo é fornecer um meio padrão para especificar os requisitos elicitados, o qual poderá ser útil tanto para as equipes de desenvolvimento, na tentativa de facilitar a construção de sistemas, quanto para as equipes de análises, nas futuras manutenções ou acréscimo de funcionalidades ao *site*.

A figura 3.6 apresenta a proposta para estruturação do documento de especificação de requisitos [24], de acordo com o método apresentado.

A parte inicial é composta de um "cabeçalho", o qual traz na primeira linha o título do documento, logo abaixo, o nome do autor ou autores, endereço, telefone, e-mail, data de produção da versão do documento de especificação, número da versão e, se esse documento for revisão de uma variante anteriormente produzida, a data da especificação anterior a esta.

A seção seguinte, após o cabeçalho, é a "introdução", a qual está dividida em três subseções: objetivo do documento, onde será descrito o propósito deste; escopo do produto, que estabelece uma visão sintética do objetivo do produto que será especificado e também o público a quem se destina; e uma visão geral do documento, com a finalidade de prover uma breve explanação do conteúdo do restante do documento, indicando sua estrutura básica. Caso alguma seção ou subseção prevista na proposta seja omitida ou alterada, a omissão ou alteração deverá ser justificada nesse ponto.

Em seguida, a seção "descrição geral do produto" aborda as funções que o sistema visa atender, além de estabelecer os aspectos técnicos que possam limitar o desenvolvimento do produto. Para isso, essa seção está subdividida em três itens: funções do produto, restrições gerais e requisitos adiados.

O item funções do produto contém as principais funções que ele desempenhará, assim como uma descrição sintética do objetivo de cada uma. E para efeito de organização, poderão ser representados em forma de tabela, cujo conteúdo será formado por um identificador da função, um nome e uma pequena descrição.

Cabeçalho

<Título do documento>

<Autor>

<Endereço>

<Data da versão do documento>

<Número da versão do documento>

1. **Introdução**
    - 1.1. Objetivo do documento
    - 1.2. Escopo do produto
    - 1.3. Visão geral do documento
2. **Descrição Geral do Produto**
    - 2.1. Funções do produto
    - 2.2. Restrições gerais
    - 2.3. Requisitos adiados
3. **Características Gerais**
    - 3.1. Documentação dos *stakeholders*
    - 3.2. Documentação dos objetivos
    - 3.3. Documentação das relações de dependências entre os objetivos
4. **Esquemas**
    - 4.1. Esquema-objetivo
    - 4.2. Esquema-cenário
    - 4.3. Esquema-ação
5. **Referências**
6. **Apêndice**
    - 5.1. Glossário
    - 5.2. Outras informações

Figura 3.6 - Documento de especificação de requisitos

Em seguida, o item referente às restrições gerais irá expor tanto os aspectos técnicos quanto gerenciais, os quais possam limitar as opções dos desenvolvedores, como por exemplo: restrições legais, de hardware, de linguagens de programação, de desempenho, de confiabilidade, etc.

Finalizando a seção de descrição geral, os requisitos adiados tratam dos requisitos identificados ao longo da elaboração da especificação de requisitos e que ficou decidido deixar para versões futuras do documento. Geralmente, tal atitude é tomada pela necessidade do cumprimento das estimativas de prazo previamente estabelecidas. Essa subseção é opcional e serve para registrar as idéias no momento de seu aparecimento e também para facilitar o surgimento de novas versões da especificação de requisitos.

A seção seguinte trata das "características gerais", que, para efeito de organização, foi dividida em três subseções: documentação dos *stakeholders*, que conterá a transcrição de todos os *stakeholders* identificados na primeira fase do método proposto; documentação dos objetivos, onde serão colocados todos os objetivos elicitados na segunda fase, e também o valor atribuído a estes pelos respectivos *stakeholders,* na terceira fase do método; e documentação das relações de dependência entre os objetivos, cuja finalidade será listar não só todas as relações de dependência como também classificá-las de acordo com a fase três.

Já a seção quatro, "esquemas", versa sobre os requisitos propriamente ditos. Para tanto, utiliza-se um conjunto de esquemas para fazer a tradução dos objetivos em requisitos. Dessa forma, a documentação dessa seção foi divida em três itens: esquema-objetivo, cuja finalidade é especificar os relacionamentos entre os objetivos e cenários; esquema-cenário, que contém o detalhamento dos cenários feitos na fase cinco do método; e esquema-ação, o qual especifica todas as ações de cada cenário operacionalizado.

As informações necessárias para que todas as fontes de dados citadas no documento de especificação de requisitos, inclusive aquelas obtidas através de atas de reuniões e memorandos, possam ser recuperadas, se necessário, devem estar descritas na seção "Referências".

Para finalizar esse documento, a seção "Apêndice" apresenta as informações que completem a especificação, tais como: o glossário de termos, usado durante a captura de

requisitos para facilitar na comunicação entre analistas e *stakeholders*, pois serve para definições de siglas, palavras e abreviações de termos usados; e outras informações, cuja finalidade será apresentar desde a lista de riscos encontrados na fase seis, às políticas criadas (na fase sete), até as tabelas de obediência preenchidas na fase oito.

Como esse documento é sugerido para especificar requisitos de acordo com os critérios estabelecidos pelo método proposto, seu preenchimento poderá ser feito de forma iterativa e incremental, a partir do início da execução do método, podendo ocorrer paralelamente a elicitação de requisitos.

3.2 Estudo comparativo com outros métodos existentes

A abordagem UWA é baseada na engenharia de requisitos orientada a objetivos e na distinção entre objetivos (metas que os *stakeholders* gostariam que o sistema satisfizesse) e requisitos (metas de baixo nível que o sistema supostamente deve conhecer e que podem ser diretamente entendidas e realizadas pelos projetistas).

No que concerne à elicitação e modelagem de requisitos, o UWA é composto por um modelo de processo que consiste nas seguintes fases: identificação dos *stakeholders*, elicitação dos objetivos do sistema, ligação de um valor de entrega a cada objetivo, refinamento dos objetivos, documentação das suposições, identificação das autoridades, identificação dos riscos associados com a aceitação, identificação dos critérios de operacionalização, identificação das técnicas de concretização, identificação das técnicas de derivação, projeto de aplicação, identificação dos serviços, e identificação e validação dos casos de uso.

A abordagem proposta engloba as fases de identificação dos *stakeholders*, identificação dos objetivos, atribuição de valores aos objetivos, refinamento dos objetivos, operacionalização dos objetivos, avaliação dos riscos, criação de políticas de segurança e

privacidade, e avaliação de obediência.

Ambas abordagens (UWA e a proposta) são orientadas a objetivos e na distinção entre objetivos e requisitos. Outro aspecto comum é que elas são centradas na identificação dos *stakeholders* e, a partir destes, as outras fases são conduzidas.

Além da fase de identificação dos *stakeholders*, duas outras fases são comuns às duas abordagens, atribuição de valores aos objetivos e refinamento dos objetivos. Porém, no método proposto essas fases são constituídas por uma série de técnicas e estratégias que ajudam a conduzir e formalizar o processo de elicitação de requisitos. Já no UWA, tais etapas são executadas de maneira informal, ou seja, não são estabelecidas nem formalizadas estratégias que ajudem a condução da etapa.

Um ponto diferencial da abordagem proposta é a utilização de uma escala numérica ascendente de valor, a qual é estabelecida por cada *stakeholder* para os objetivos que este especificou, cuja finalidade é possibilitar resolver possíveis conflitos que possam existir entre o conjunto de objetivos elicitados e também ajudar na fase posterior de refinamento de objetivos.

O método GBRAM engloba duas fases: análise dos objetivos, onde são identificados os objetivos, os *stakeholders* e os agentes responsáveis pela execução dos objetivos. Além disso, os objetivos são classificados em objetivos de manutenção e de realização, e são estabelecidas as relações de dependência entre os mesmos; e refinamento dos objetivos, onde além do refinamento, faz-se elaboração de cenários e operacionalização dos requisitos.

A abordagem proposta também mantém as relações de dependência de precedência e dependência de contrato providas pelo GBRAM, como forma de ordenar os objetivos. Outra semelhança é a fase de refinamento dos objetivos, onde são unificados os objetivos sinônimos e eliminados os redundantes.

Outra fase comum às duas vertentes é a operacionalização de objetivos. Porém, embora utilizem uma espécie de *template* estendido, o método proposto faz algumas adaptações para atender à natureza das aplicações de comércio eletrônico.

Uma diferença importante entre essas duas abordagens é que a proposta é centrada na identificação dos *stakeholders*, por acreditar que eles são fontes naturais de derivação dos objetivos e não vice-versa.

Outro ponto a se destacar é que, na instanciação do GBRAM para a criação de políticas de segurança e privacidade em sistemas de comércio eletrônico, foram adicionadas algumas fases: avaliação de riscos, alocação de recursos e avaliação de obediência. Sendo que as fases de avaliação de riscos e obediência também foram mantidas no método proposto. Porém, enquanto a instanciação do GBRAM utiliza o PFIRES [21], a abordagem apresentada nesse trabalho emprega os padrões propostos por [32], por serem bastante utilizados em grandes *sites* de comércio eletrônico. Dessa forma, foi adicionada a fase de avaliação de riscos, a identificação do par (ameaça, vulnerabilidade) e uma estimativa de probabilidade de ocorrência de risco para cada par.

O método proposto ainda se diferencia de outras abordagens por possuir uma fase para criação de políticas de segurança e privacidade, adotando modelos que visam estabelecê-las de maneira mais clara, menos ambígua e contendo, apenas, os aspectos mais relevantes, beneficiando, assim, tanto a organização quanto os usuários de um *site* de comércio eletrônico.

É importante ressaltar que, embora o método proposto tenha sido criado sob a ótica de elicitar requisitos de sistemas em obediência às políticas de segurança e privacidade em sites de comércio eletrônico, nada impede que o mesmo possa ser empregado para elicitar requisitos de qualquer tipo de sistema. Nesse caso, não serão necessárias as execuções das fases sete e oito.

Finalmente, a abordagem proposta ainda sugere um modelo para o documento de especificação de requisitos, como forma de estabelecer um meio padrão para especificar requisitos de software, cuja finalidade é facilitar as futuras manutenções do sistema e também ajudar na condução da fase posterior de validação dos requisitos elicitados.

A tabela 3.1 apresenta um resumo das principais características usadas na comparação entre os métodos utilizados e o método proposto.

Características	**UWA**	**GBRAM Instanciado**	**Método Proposto**
Distinção entre objetivos e requisitos	Sim	Sim	Sim
Centralização na identificação dos *stakeholders*	Sim	Não	Sim
Atribuição de valores aos objetivos	Sim	Não	Sim
Avaliação de ricos	Não	Sim	Sim
Criação de políticas de segurança e privacidade	Não	Não	Sim
Avaliação de obediência com as políticas existentes	Não	Sim	Sim
Estruturação do documento de especificação de requisitos	Não	Não	Sim

Tabela 3.1 – Resumo das características usadas na comparação entre os métodos

4 ESTUDO DE CASO

No estudo de caso considerado, tem-se uma livraria, que é uma empresa já consolidada no mercado há vários anos, com três lojas físicas na cidade de São Luís, Maranhão, onde são comercializados livros, CDs, DVDs, fitas de vídeo e etc.

Dentro da perspectiva de maximização das vendas, os donos perceberam que seus lucros poderiam ser aumentados se as vendas fossem disponibilizadas também pela *internet*. Além disso, para a empresa, o comércio eletrônico a tornaria pioneira nesse segmento no mercado local e também permitiria que seus produtos fossem comercializados, não só para todo país como também para todo o mundo.

Assim, por trabalhar com produtos pouco sujeitos às intempéries do mercado, mantém seus estoques dentro de um planejamento consistente. Para todos os pedidos efetuados, será fornecida ao cliente uma previsão de entrega, a qual dependerá da disponibilidade do produto no estoque e também dos fornecedores, os quais terão que obedecer aos prazos estabelecidos no *site* para cada região.

A totalidade da entrega dos pedidos será feita pelos correios, não importando onde o cliente esteja localizado, nem o tipo de pedido efetuado, sendo que o frete será cobrado do comprador, o qual terá conhecimento do valor a ser pago no momento da compra.

Todos os produtos comercializados no *site* serão passíveis de devolução por um prazo de noventa dias, a contar da data de emissão da nota fiscal, caso possuam algum defeito de fabricação e não apresentem indícios de mau uso.

Os pedidos serão controlados desde sua entrada até a entrega ao consumidor, sendo permitido a este consultar o andamento do pedido, a qualquer momento, pelo *site*.

Para tornar as compras mais atraentes, haverá diferenças de preços entre a loja virtual e as físicas na ordem de até 15%. E, nos meses de grande demanda, como dezembro (natal), janeiro (volta às aulas) e maio (dia das mães), essa diferença poderá chegar até 20%.

Ao realizar uma compra, será dada a opção ao cliente de receber a encomenda em casa ou em qualquer outro endereço especificado pelo mesmo. Nesse caso, o valor do pedido será recalculado de acordo com o CEP informado.

O *site* disponibilizará duas formas de pagamento: cartão de crédito e boleto bancário. Caso o cliente escolha a primeira opção, poderá fazer o parcelamento das compras de acordo com as restrições da administradora, as quais deverão estar explicitadas no *site*.

O boleto bancário será gerado pelo próprio sistema, após a finalização da compra e tem prazo para pagamento de três dias após a realização do pedido. Já a opção cartão de crédito, estará condicionada à liberação da administradora. Em ambos os casos, o envio das mercadorias constantes no pedido aos correios só acontecerá após a confirmação do pagamento.

O *site* deverá, ainda, disponibilizar uma lista com as perguntas mais freqüentes feitas pelos clientes, cujo objetivo será tanto auxiliar a esclarecer dúvidas de outros clientes como, também, caso seja necessário, melhorar o *site*.

De acordo com a descrição acima, o método proposto irá ser aplicado ao domínio dessa livraria como forma de mostrar sua aplicabilidade. Assim, inicialmente, foram aplicadas as estratégias dirigidas a perguntas: “quem ou o que reivindica algum interesse no sistema?“ e “quem ou o que se ganha ou perde com o desenvolvimento do sistema?“ para a identificação dos *stakeholders*.

Dessa forma, seguindo as estratégias contidas na primeira fase do método, identificamos os seguintes *stakeholder*s, os quais, para efeito de organização, estão dispostos em forma de tabela.

Código	S1
Nome	Cliente
Tarefa que Executa	✓ Cadastra/Atualiza/Exclui dados ✓ Realiza compras ✓ Faz devoluções

Tabela 4.1 – Características do *stakeholder* S1

Código	S2
Nome	Funcionário da livraria
Tarefa que Executa	✓ Cadastra/Atualiza/Exclui produtos ✓ Despacha produtos ✓ Controla recebimento de produtos

Tabela 4.2 – Características do *stakeholder* S2

Código	S3
Nome	Fornecedor
Tarefa que Executa	✓ Faz cotação de produtos ✓ Fornece produtos

Tabela 4.3 – Características do *stakeholder* S3

Código	S4
Nome	Gerente
Tarefa que Executa	✓ Solicita cotação de produtos ✓ Faz pedidos de compra ✓ Paga fornecedores

Tabela 4.4 – Características do *stakeholder* S4

Código	S5
Nome	Distribuidor
Tarefa que Executa	✓ Recebe pedidos ✓ Despacha pedidos feitos no *site*

Tabela 4.5 – Características do *stakeholder* S5

Seguindo os passos do método proposto e usando as técnicas contidas na segunda fase do mesmo, foram identificados os seguintes objetivos para cada um dos *stakeholders*,

conforme ilustrado na tabela 4.6.

Stakeholder	Objetivos	Descrição
S1	G1	Cadastrar clientes
	G3	Realizar compras
	G4	Escolher formas de pagamento
	G5	Fazer devolução de produtos
	G8	Acompanhar pedidos realizados
	G9	Enviar email para cada fase de acompanhamento do pedido
	G11	Montar lista de compras (à medida que os produtos vão sendo adicionados no carrinho de compras)
	G12	Permitir alterar dados cadastrais
	G19	Incluir itens no carrinho de pedidos
	G20	Fazer busca por palavras-chaves
S2	G7	Cadastrar produtos
	G12	Permitir alterar dados cadastrais
	G13	Fazer cotação de preços junto aos fornecedores
	G16	Despachar pedidos
S3	G14	Cadastrar fornecedor
	G15	Realizar cotação *on-line*
S4	G1	Cadastrar clientes
	G2	Autenticar usuários
	G6	Exibir lista dos produtos mais vendidos
	G10	Apresentar FAQ com as dúvidas mais freqüentes
	G12	Permitir alterar dados cadastrais
	G18	Fazer pedido de compra para reposição de estoque
	G20	Fazer busca por palavras-chaves
S5	G17	Entregar pedidos

Tabela 4.6 – Objetivos identificados por cada um dos *stakeholders*

Ainda de acordo com a segunda fase do método proposto, o passo seguinte é descrever cada objetivo elicitado na forma de cenários, visando um detalhamento maior desses objetivos. Convém ressaltar que os cenários descritos aqui representarão não só os comportamentos normais como também os excepcionais e variacionais.

Para facilitar a leitura desse estudo de caso, das páginas 86 à 102 estão descritos todos os cenários descobertos para cada um dos objetivos elicitados. E, a partir da página 102 é dado continuidade a aplicação das outras fases do método ao estudo de caso aqui apresentado.

Cenário: C1

Nome: Cadastrar cliente de uma maneira normal

Objetivo: G1

Ator/Agente: Cliente

Pré-Condições: *If* dados informados são válidos

Descrição:

1. O cliente entra na área de cadastro
2. O cliente informa seus dados pessoais, seu usuário e senha para acesso ao sistema
3. O sistema faz a validação de tamanho e conteúdo dos campos preenchidos
4. O sistema armazenará as informações na base de dados
5. O sistema emite uma mensagem “Dados cadastrados com sucesso”

Pós-Condições: *Do* cliente cadastrado

Cenário: C1.1

Nome: Cadastrar cliente com dados inválidos

Objetivo: G1

Ator/Agente: Cliente

Pré-Condições: *If* dados informados não são válidos

Descrição:

1. O cliente entra na área de cadastro
2. O cliente informa seus dados pessoais, seu usuário e senha para acesso ao sistema
3. O sistema faz a validação de tamanho e conteúdo dos campos preenchidos
4. Enquanto o sistema encontrar campos incompletos ou inválidos
5. O sistema exibe uma mensagem solicitando o preenchimento ou a correção das informações
6. O cliente corrige as informações
7. O sistema faz a validação
8. O sistema emite uma mensagem “Dados cadastrados com sucesso”

Pós-Condições: *Do* cliente cadastrado

Cenário: C2

Nome: Autenticar usuário de uma maneira normal

Objetivo: G2

Ator/Agente: Usuário (Cliente, Funcionário, Gerente e Fornecedor)

Pré-Condições: *If* código usuário = válido

If senha = válida

Descrição:

1. O usuário acessa a página de *login*
2. O usuário informa o *login* e a senha
3. O sistema verifica os dados na base de dados
4. O sistema autentica o usuário e o encaminha para a página seguinte

Pós-Condições: *Do* usuário autenticado

Cenário: C2.1

Nome: Autenticar usuário com código de usuário e/ou senha inválida

Objetivo: G2

Ator/Agente: Usuário (Cliente, Funcionário, Gerente e Fornecedor)

Pré-Condições: *If* código usuário = inválido e/ou

If senha = inválida

Descrição:

1. O usuário acessa a página de *login*
2. O usuário informa o *login* e a senha
3. O sistema verifica os dados na base de dados
4. O sistema emite a mensagem "Código do usuário ou senha inválida, por favor digite novamente"
5. O usuário digita novamente o código do usuário e a senha
6. O sistema autentica o usuário e o encaminha para a página seguinte

Pós-Condições: *Do* usuário autenticado

Cenário: C3

Nome: Realizar compras de uma maneira normal

Objetivo: G3

Ator/Agente: Cliente

Pré-Condições: *If* cliente fez o *login* no sistema

If carrinho <> vazio

If endereço de entrega é igual ao constante no cadastro do cliente

Descrição:

1. O cliente faz o *login* no sistema

2. O sistema verifica que existem itens no carrinho de compras
3. O sistema exibe os dados cadastrais do cliente e o endereço de entrega igual ao existente no cadastro do mesmo
4. O sistema exibe um botão de confirmação e outro de alterar endereço de entrega do produto
5. O cliente confirma o endereço de entrega
6. O sistema mostrar o valor total do pedido
7. O sistema solicita a escolha da forma de pagamento
8. O cliente escolhe a forma de pagamento
9. O sistema exibe uma mensagem "Entrega do pedido condicionado a confirmação do pagamento"
10. O sistema gera um número de pedido com situação "aguardando confirmação de pagamento"

Pós-Condições: *Do* pedido realizado com sucesso

Do situação do pedido = "aguardando confirmação de pagamento"

Cenário: C3.1

Nome: Realizar compra sem itens no carrinho de pedidos

Objetivo: G3

Ator/Agente: Cliente

Pré-Condições: *If* cliente fez o *login* no sistema

If carrinho = vazio

If endereço de entrega é igual ao constante no cadastro do cliente

Descrição:

1. O cliente faz o *login* no sistema
2. O sistema verifica que existem itens no carrinho de compras
3. O sistema exibe a mensagem "Não existem itens no carrinho de compras, pelo menos um item deve ser escolhido"
4. O cliente escolhe um produto e manda adicionar no carrinho de compras
5. O sistema exibe os dados cadastrais o cliente e o endereço de entrega igual ao existente no cadastro do cliente
6. O sistema exibe um botão de confirmação e outro de alterar endereço de entrega do produto
7. O cliente confirma o endereço de entrega

8. O sistema mostrar o valor total do pedido
9. O sistema solicita a escolha da forma de pagamento
10. O cliente escolhe a forma de pagamento
11. O sistema exibe uma mensagem “Entrega de pedido condicionado a confirmação do pagamento”
12. O sistema gera um número de pedido com situação “aguardando confirmação de pagamento”

Pós-Condições: *Do* pedido realizado com sucesso

Do situação do pedido = “aguardando confirmação de pagamento”

Cenário: C3.2

Nome: Realizar compra com endereço entrega diferente do existente no cadastro do cliente

Objetivo: G3

Ator/Agente: Cliente

Pré-Condições: *If* cliente fez o *login* no sistema

If carrinho <> vazio

If endereço de entrega é diferente do existente no cadastro do cliente

Descrição:

1. O cliente faz o *login* no sistema
2. O sistema verifica que existem itens no carrinho de compras
3. O sistema exibe os dados cadastrais o cliente e o endereço de entrega igual ao existente no cadastro do cliente
4. O sistema exibe um botão de confirmação e outro de alterar endereço de entrega do produto
5. O cliente altera o endereço de entrega
6. O sistema recalcula o valor do frete (se houver necessidade)
7. O sistema mostra o valor total do pedido
8. O sistema solicita a escolha da forma de pagamento
9. O cliente escolhe a forma de pagamento
10. O sistema exibe uma mensagem “Entrega de pedido condicionado a confirmação do pagamento”
11. O sistema gera um número de pedido com situação “aguardando confirmação de pagamento”

Pós-Condições: *Do* pedido realizado com sucesso

Do situação do pedido = “aguardando confirmação de pagamento”

Cenário: C4

Nome: Escolher forma de pagamento do tipo boleto bancário

Objetivo: G4

Ator/Agente: Cliente

Pré-Condições: *If* forma de pagamento escolhida = boleto bancário

Descrição:

1. O sistema lista as formas disponíveis para pagamento
2. O cliente escolhe a opção pagamento com boleto bancário
3. O sistema informa que o cliente terá três dias para efetuar o pagamento
4. O sistema exibe o botão gerar boleto bancário
5. O cliente manda gerar o boleto bancário
6. O sistema informa que essa opção de pagamento tem até 48h, após efetivação do pagamento, para ser compensada

Pós-Condições: *Do* forma de pagamento = boleto bancário

Cenário: C4.1

Nome: Escolher forma de pagamento do tipo cartão de crédito

Objetivo: G4

Ator/Agente: Cliente

Pré-Condições: *If* forma de pagamento escolhida = cartão de crédito

Descrição:

1. O sistema lista as formas disponíveis para pagamento
2. O cliente escolhe a opção pagamento com cartão de crédito
3. O sistema solicita a digitação das informações do cartão
4. O cliente informa a operadora, o número do cartão, a validade do cartão, o código de segurança e a quantidade de parcelas
5. O sistema informa que essa opção de pagamento está condicionada à liberação da administradora

Pós-Condições: *Do* forma de pagamento = cartão de crédito

Cenário: C5

Nome: Fazer devolução de produtos de uma maneira normal

Objetivo: G5

Ator/Agente: Cliente e Funcionário da livraria

Pré-Condições: *If* situação do pedido = entregue

If usuário fez o *login* no sistema

Descrição:

1. O cliente faz *login* no sistema
2. O cliente entra no formulário de devolução de produtos
3. O cliente informa o número do pedido, o produto e o motivo da devolução
4. O sistema verifica que a situação do número do pedido informado é "entregue"
5. O sistema notifica o cliente dos procedimentos para devolução
6. O sistema gera um número de pedido de devolução com situação "não recebido"
7. O cliente despacha o produto para a livraria
8. O funcionário da livraria recebe o produto
9. O funcionário da livraria atualiza a situação da devolução para "recebido"
10. O funcionário envia o novo produto
11. O funcionário atualiza a situação da devolução para "enviado"

Pós-Condições: *Do* situação da devolução = "enviado"

Cenário: C5.1

Nome: Fazer devolução de produtos cuja situação do pedido é "não entregue"

Objetivo: G5

Pré-Condições: *If* situação do pedido = não entregue

If usuário fez o *login* no sistema

Descrição:

1. O cliente faz *login* no sistema
2. O cliente entra no formulário de devolução de produtos
3. O cliente informa o número do pedido, o produto e o motivo da devolução
4. O sistema verifica que a situação do número do pedido informado é "não entregue"
5. O sistema notifica o cliente que a devolução não pode ser realizada porque o pedido não foi entregue

Cenário: C6

Nome: Exibir lista de produtos mais vendidos

Objetivo: G6

Ator/Agente: Gerente

Pré-condições: *If* gerente fez *login* no sistema

Descrição:

1. O gerente faz *login* no sistema
2. O sistema faz a autenticação do gerente
3. O gerente agenda a freqüência que ele deseja atualizar a lista dos produtos mais vendidos e como as mesmas deverão ser organizadas
4. O sistema verifica o agendamento
5. O sistema processa as informações das vendas de acordo com o agendamento
6. O sistema organiza o *ranking* de vendas de acordo com o informado

Cenário: C7

Nome: Cadastrar produtos de uma maneira normal

Objetivo: G7

Ator/Agente: Funcionário da livraria

Pré-condições: *If* funcionário fez *login* no sistema

If dados do produto são válidos

Descrição:

1. O funcionário faz *login* no sistema
2. O sistema faz a autenticação do funcionário
3. O sistema abre o formulário de cadastro do produto
4. O funcionário entra com os dados do produto e a classificação do mesmo (livro, CD, DVD)
5. O sistema faz a verificação de tamanho e conteúdo dos campos
6. O sistema inclui as informações na base de dados e exibe a mensagem “Produto cadastrado com sucesso”
7. O sistema exibe o produto na seção indicada

Pós-Condições: *Do* produto cadastrado

Cenário: C7.1

Nome: Cadastrar produtos com dados inválidos

Objetivo: G7

Ator/Agente: Funcionário da livraria

Pré-condições: *If* funcionário fez *login* no sistema

If dados do produto são inválidos

Descrição:

1. O funcionário faz *login* no sistema
2. O sistema faz a autenticação do funcionário
3. O sistema abre o formulário de cadastro do produto
4. O funcionário entra com os dados do produto e a classificação do mesmo (livro, CD, DVD)
5. O sistema faz a verificação de tamanho e conteúdo dos campos
6. Enquanto o sistema verificar que os dados estão incompletos ou inválidos
7. O sistema exibe a mensagem "Dados inválidos ou incompletos, por favor digite novamente"
8. O funcionário corrige e/ou completa as informações
9. O sistema inclui as informações na base de dados e exibe a mensagem "Produto cadastrado com sucesso"
10. O sistema exibe o produto na seção indicada

Pós-Condições: *Do* produto cadastrado

Cenário: C8

Nome: Acompanhar pedidos de uma maneira normal

Objetivo: G8

Ator/Agente: Cliente

Pré-condições: *If* cliente fez *login* no sistema

If número de pedidos do cliente <> vazio

Descrição:

1. O cliente entra na área "meus pedidos"
2. O cliente faz *login* no sistema
3. O sistema faz a autenticação
4. O sistema lista o histórico dos pedidos do cliente com as suas respectivas situações
5. O cliente clica no pedido que deseja detalhamento
6. O sistema exibe o detalhamento do pedido selecionado

Cenário: C8.1

Nome: Acompanhar pedidos sem que haja qualquer pedido realizado

Objetivo: G8

Ator/Agente: Cliente

Pré-condições: *If* cliente fez *login* no sistema

If número de pedidos do cliente = vazio

Descrição:

1. O cliente entra na área “meus pedidos”
2. O cliente faz *login* no sistema
3. O sistema faz a autenticação
4. O sistema verifica que não existem pedidos realizados para o cliente
5. O sistema exibe a mensagem “Não existem pedidos realizados”

Cenário: C9

Nome: Enviar *emails*

Objetivo: G9

Ator/Agente: Sistema

Pré-condições: *If* cliente optou por receber correspondências do *site*

Descrição:

1. O sistema verifica, para cada pedido realizado, se houve mudança na situação do pedido
2. O sistema verifica se o cliente cadastrou *email* e optou por receber correspondências do *site*
3. O sistema envia o *email* informando a ocorrência
4. O sistema marca na base de dados que o *email* já foi enviado para aquela situação

Pós-Condição: *Do email* enviado com sucesso

Cenário: C10

Nome: Cadastrar uma pergunta na FAQ de uma maneira normal

Objetivo: G10

Ator/Agente: Gerente

Pré-condições: *If* gerente fez *login* no sitema

If pergunta não existe na FAQ

Descrição:

1. O gerente faz *login* no sitema
2. O gerente consulta os emails enviados com dúvidas ou perguntas
3. O gerente responde os *emails* e, de acordo com a relevância, inclui na FAQ

4. O sistema verifica que a pergunta não existe na FAQ
5. O sistema inclui a pergunta na base de dados e exibe a mensagem “Pergunta cadastrada com sucesso”
6. O sistema exibe a pergunta na FAQ

Pós-Condição: *Do* pergunta cadastrada

Cenário: C10.1

Nome: Cadastrar uma pergunta já existente na FAQ

Objetivo: G10

Ator/Agente: Gerente

Pré-condições: *If* gerente fez *login* no sitema

If pergunta já existe na FAQ

Descrição:

1. O gerente faz *login* no sitema
2. O gerente consulta os *emails* enviados com dúvidas ou perguntas
3. O gerente responde os *emails* e, de acordo com a relevância, inclui na FAQ
4. O sistema verifica que a pergunta já existe na FAQ
5. O sistema exibe a mensagem “Pergunta já existe na FAQ”

Pós-Condição: Do pergunta não cadastrada

Cenário: C11

Nome: Montar carrinho de compras de uma maneira normal

Objetivo: G11

Ator/Agente: Cliente

Pré-Condições: *If* produto selecionado

Descrição:

1. O cliente navega pelo *site*
2. O cliente seleciona um produto e clica no botão adicionar no carrinho de compras
3. O sistema verifica que um produto foi selecionado
4. O sistema solicita o CEP para a entrega, a quantidade de produtos e a forma de envio (encomenda normal ou SEDEX)
5. O cliente informa o CEP, a quantidade de produtos e a forma de envio
6. O sistema calcula o valor do frete para o produto escolhido bem como o prazo de entrega

7. O sistema exibe um botão de continuar comprando e outro de finalizar a compra
8. O cliente clica em um dos botões

Pós-Condições: *Do* número de itens do carrinho <> vazio

Cenário: C11.1

Nome: Montar carrinho de compras sem selecionar um produto

Objetivo: G11

Ator/Agente: Cliente

Pré-Condições: *If* produto não selecionado

Descrição:

1. O cliente navega pelo *site*
2. O cliente clica no botão adicionar no carrinho de compras
3. O sistema verifica que nenhum produto foi selecionado
4. O sistema exibe a mensagem “pelo menos um item deve ser adicionado ao carrinho de compras”

Cenário: C12

Nome: Alterar dados cadastrais de uma maneira normal

Objetivo: G12

Ator/Agente: Usuário (Cliente, Funcionário, Gerente e Fornecedor)

Pré-Condições: *If* dados informados são válidos

Descrição:

1. O usuário entra faz *login* no sistema
2. O sistema faz a autenticação
3. O usuário clica no botão “alterar dados cadastrais”
4. O sistema abre um formulário, de acordo com o tipo de usuário (cliente, funcionário, fornecedor), com os dados existentes
5. O usuário altera as informações desejadas
6. O sistema faz a validação de tamanho e conteúdo dos campos preenchidos
7. O sistema armazena as informações na base de dados
8. O sistema emite uma mensagem “Dados cadastrados com sucesso”

Pós-Condições: *Do* dados alterados com sucesso

Cenário: C12.1

Nome: Alterar dados cadastrais com informações inválidas

Objetivo: G12

Ator/Agente: Usuário (Cliente, Funcionário, Gerente e Fornecedor)

Pré-Condições: *If* dados informados não são válidos

Descrição:

1. O usuário entra faz *login* no sistema
2. O sistema faz a autenticação
3. O usuário clica no botão "alterar dados cadastrais"
4. O sistema abre um formulário, de acordo com o tipo de usuário (cliente, funcionário, fornecedor), com os dados existentes
5. O usuário altera as informações desejadas
6. O sistema faz a validação de tamanho e conteúdo dos campos preenchidos
7. Enquanto o sistema encontrar campos incompletos ou inválidos
8. O sistema exibe uma mensagem solicitando o preenchimento ou a correção das informações
9. O usuário corrige as informações
10. O sistema faz a validação
11. O sistema armazena as informações na base de dados
12. O sistema emite uma mensagem "Dados cadastrados com sucesso"

Pós-Condições: *Do* dados alterados com sucesso

Cenário: C13

Nome: Fazer cotação de preços junto aos fornecedores

Objetivo: G13

Ator/Agente: Funcionário da livraria

Pré-Condição: *If* gerente fez *login* no sistema

Descrição:

1. O funcionário faz *login* no sistema
2. O sistema faz a autenticação
3. O funcionário abre um formulário que contém a lista de produtos (por tipo) que estão com estoque baixo
4. O funcionário seleciona os produtos e os fornecedores com os quais deseja fazer cotação de preços

5. O sistema envia email aos fornecedores com as especificações dos produtos desejados
6. O sistema atualiza a situação desses produtos para “em cotação”

Pós-Condições: *Do* situação do produto = “em cotação”

Do email enviado ao fornecedor

Cenário: C14

Nome: Cadastrar fornecedor de uma maneira normal

Objetivo: G14

Ator/Agente: Fornecedor

Pré-Condições: *If* dados informados são válidos

Descrição:

1. O fornecedor entra na área de cadastro do *site* destinada a ele
2. O fornecedor informa seus dados pessoais, seu usuário, senha para acesso ao sistema e também os produtos com os quais trabalha
3. O sistema faz a validação de tamanho e conteúdo dos campos preenchidos
4. O sistema armazenará as informações na base de dados
5. O sistema emite uma mensagem “Dados cadastrados com sucesso”

Pós-Condições: *Do* fornecedor cadastrado

Cenário: C14.1

Nome: Cadastrar fornecedor com dados inválidos

Objetivo: G14

Ator/Agente: Fornecedor

Pré-Condições: *If* dados informados não são válidos

Descrição:

1. O fornecedor entra na área de cadastro do *site* destinada a ele
2. O fornecedor informa seus dados pessoais, seu usuário, senha para acesso ao sistema e os produtos com os quais trabalha
3. O sistema faz a validação de tamanho e conteúdo dos campos preenchidos
4. Enquanto o sistema encontrar campos incompletos ou inválidos
5. O sistema exibe uma mensagem solicitando o preenchimento ou a correção das informações
6. O fornecedor corrige as informações
7. O sistema faz a validação

8. O sistema emite uma mensagem “Dados cadastrados com sucesso”

Pós-Condições: *Do* fornecedor cadastrado

Cenário: C15

Nome: Realizar cotação *on-line* de uma maneira normal

Objetivo: G15

Ator/Agente: Fornecedor

Pré-Condições: *If* fornecedor fez *login* no sistema

If existem produtos a serem cotados

Descrição:

1. O fornecedor recebe *email* informando que existem produtos para serem cotados
2. O fornecedor entra no *site* e efetua o *login*
3. O sistema faz a autenticação
4. O sistema verifica que existem produtos a serem cotados pelo fornecedor e abre um formulário com os mesmos
5. O fornecedor preenche as informações de preço, valor, prazo de entrega e forma de pagamento para cada produto existente na lista
6. O sistema exibe a mensagem “Cotação realizada com sucesso”
7. O sistema atualiza a situação dos produtos constante na lista para “cotados”
8. O sistema envia um *email* ao gerente informando que existe cotação a ser analisada

Pós-Condições: *Do* situação do produto = cotado

Do email enviado ao gerente

Cenário: C15.1

Nome: Realizar cotação sem que haja produtos para serem cotados

Objetivo: G15

Ator/Agente: Fornecedor

Pré-Condições: *If* fornecedor fez *login* no sistema

If não existem produtos para serem cotados

Descrição:

1. O fornecedor entra no *site* e efetua o *login*
2. O sistema faz a autenticação
3. O sistema verifica que não existem produtos a serem cotados pelo fornecedor
4. O sistema exibe a mensagem “No momento não existem produtos para cotação”

Cenário: C16

Nome: Despachar pedidos de uma maneira normal

Objetivo: G16

Ator/Agente: Funcionário

Pré-Condições: *If* funcionário fez *login* no sistema

If existem pedidos com confirmações de pagamentos

Descrição:

1. O funcionário faz *login* no sistema
2. O sistema faz a autenticação
3. O funcionário entra na área de despachar pedidos
4. O sistema verifica que existem pedidos para ser despachados
5. O funcionário emite a relação dos pedidos que tiveram confirmação de pagamento
6. O funcionário seleciona os produtos no estoque que constituem um pedido
7. O funcionário emite etiquetas com o endereço de entrega e identifica os pedidos
8. O funcionário gera uma listagem, por data, dos pedidos que estão sendo enviados para os Correios
9. O funcionário atualiza a situação do pedido para “enviado para os Correios”

Pós-Condições: *Do* situação do produto = “enviado para os correios”

Cenário: C16.1

Nome: Despachar pedidos sem confirmação de pagamento

Objetivo: G16

Ator/Agente: funcionário

Pré-Condições: *If* funcionário fez *login* no sistema

If não existem pedidos com confirmações de pagamentos

Descrição:

1. O funcionário faz *login* no sistema
2. O sistema faz a autenticação
3. O funcionário entra na área de despachar pedidos
4. O sistema verifica que não existem pedidos para ser despachados
5. O sistema exibe a mensagem “Não existem pedidos para serem despachados”

Cenário: C17

Nome: Entregar pedidos

Objetivo: G17

Ator/Agente: Distribuidor, Funcionário

Descrição:

1. O distribuidor recebe uma lista e os produtos para ser entregues
2. O distribuidor confere os produtos
3. O distribuidor verifica a opção escolhida (encomenda normal ou SEDEX) e encaminha o pedido em malote ou avião
4. Ao ser entregue o pedido, o distribuidor envia uma listagem à livraria com os pedidos entregues
5. O funcionário da livraria atualiza situação do pedido para "entregue"

Pós-Condições: *Do* situação do pedido = "entregue"

Cenário: C18

Nome: Fazer pedido de compras para reposição de estoque

Objetivo: G18

Ator/Agente: Gerente

Descrição:

1. O gerente recebe um *email* avisando que existem cotações para ser analisadas
2. O gerente faz *login* no sistema
3. O sistema faz a autenticação
4. O sistema lista os produtos das cotações vencedoras
5. O gerente escolhe os produtos da cotação vencedora e monta o pedido de compra
6. O sistema atualiza a situação do produto cotado para "feito pedido de compra"
7. O sistema envia um *email* ao fornecedor informado que foi feito um pedido de compra

Pós-Condições: *Do* situação do produto = "feito pedido de compra"

Cenário: C19

Nome: Incluir itens no carrinho de compras

Objetivo: G19

Ator/Agente: Cliente

Descrição:

1. O cliente navega pelo *site*
2. O cliente seleciona o produto desejado
3. O cliente clica no botão incluir no carrinho de compras

4. O sistema mostra o valor do produto e solicita que o cliente informe o CEP de entrega e a quantidade de produtos
5. O sistema calcula o valor do frete de acordo com os dados informados
6. O sistema exibe o valor total do pedido (produto + valor de frete)

Pós-Condições: *Do* produto incluído no carrinho de compras do cliente

Cenário: C20
Nome: Fazer busca de produtos por palavras-chaves
Objetivo: G20
Ator/Agente: Cliente
Descrição:

1. O cliente navega pelo *site*
2. O cliente digita a palavra que deseja buscar
3. O sistema procura por todos os produtos que contem a palavra informada
4. O sistema lista, por ordem alfabética, todos os itens encontrados

Após a descrição de todos os cenários, dá-se início a fase três do método, que é a atribuição de valores a cada um dos objetivos estabelecidos. Assim, a partir dos *stakeholders* identificados têm-se:

Stakeholder	Objetivos Especificados	Valor
S1	G1	3
	G3	10
	G4	8
	G5	9
	G8	7
	G9	4
	G11	2
	G12	6
	G19	1
	G20	5
S2	G7	2
	G12	1
	G13	3
	G16	4
S3	G14	1
	G15	2
S4	G1	5
	G2	4
	G6	1
	G10	6
	G12	3
	G18	7
	G20	2
S5	G17	1

Tabela 4.7 – Valores atribuídos aos objetivos pelos *stakeholders*

Ainda de acordo com a fase três do método, o passo seguinte é a ordenação dos objetivos de acordo com as relações de dependência encontradas entre os mesmo. Dessa forma, foram encontradas as seguintes relações de dependências de precedência:

$G1 < G12$

$G3 < G5$

$G13 < G15 < G18$

$G3 \rightarrow G4$

Um exemplo de dependência de precedência ocorre entre os objetivos G1 e G12, onde para que um cliente possa alterar seus dados cadastrais é necessário que ele já esteja cadastrado junto à livraria Vinícius de Moraes.

Outra relação de dependência de precedência foi encontrada entre os objetivos G3 e G5, ou seja, uma devolução de produto deve ser precedida de uma compra. Por fim, verifica-se que para um pedido de compra possa ser realizado (G18) é necessário que os fornecedores já tenham cotado os produtos (G15), que por sua vez, deve ter recebido a lista de produtos a cotar (G13).

Já a relação de dependência de contrato encontrada pode ser exemplificada pelos objetivos G3 e G4, isto é, o ato de realizar uma compra dispara automaticamente a escolha de uma forma de pagamento.

A estratégia de refinamento, contida na fase quatro do método, foi aplicada sobre o exemplo proposto. Assim, os objetivos G11 - Montar lista de compras e G19: - Incluir itens no carrinho de pedidos, eram sinônimos, não havendo necessidade de manter os dois, ficando somente o G11.

Após a fase de refinamento, dá-se início a operacionalização dos objetivos, que consiste em traduzi-los nos requisitos através da utilização de um conjunto de esquemas, descritos na fase cinco do método. A seguir foram descritos alguns esquemas-objetivos, esquemas-cenários e esquemas-ação do exemplo proposto.

Esquema-Objetivo

Nº do objetivo: G1

Nome: Cadastrar Cliente

Descrição: Consiste em cadastrar os clientes que acessam o *site* para que os mesmos possam realizar suas compras

Stakeholders: S1 e S4

Valor: 3 e 5

Cenários: C1, C1.1

Esquema-Cenário

Cenário: C1

Nome: Cadastrar usuário de uma maneira normal

Objetivo: G1

Ator/Agente: Cliente

Pré-Condições: If dados informados são válidos

Descrição:

1. O cliente entra na área de cadastro
2. O cliente informa seus dados pessoais, seu usuário e senha para acesso ao sistema
3. O sistema faz a validação de tamanho e conteúdo dos campos preenchidos
4. O sistema armazenará as informações na base de dados
5. O sistema emite uma mensagem “Dados cadastrados com sucesso”

Pós-Condições: Do cliente cadastrado

Cenário: C1.1

Nome: Cadastrar usuário com dados inválidos

Objetivo: G1

Ator/Agente: Cliente

Pré-Condições: If dados informados não são válidos

Descrição:

1. O cliente entra na área de cadastro
2. O cliente informa seus dados pessoais, seu usuário e senha para acesso ao sistema
3. O sistema faz a validação de tamanho e conteúdo dos campos preenchidos
4. Enquanto o sistema encontrar campos incompletos ou inválidos
5. O sistema exibe uma mensagem solicitando o preenchimento ou a correção das informações
6. O cliente corrige as informações
7. O sistema faz a validação
8. O sistema emite uma mensagem “Dados cadastrados com sucesso”

Pós-Condições: Do cliente cadastrado

Esquema-Ação

Ação: O sistema valida as informações do cliente

Tipo: Sistema

Entrada: nome, CPF, endereço, usuário e senha de acesso

Nº de seqüência: 1

Código do Cenário: C1, C1.1
Ação: O cliente redigita as informações
Tipo: Cliente
Entrada: nome, CPF, endereço, usuário e senha de acesso
Nº de seqüência: 2
Código do Cenário: C1.1

Após a operacionalização dos requisitos, é iniciada a fase seis do método proposto, que consiste na avaliação de riscos. Uma boa fonte para a identificação de risco de uma aplicação *Web* é a análise de seus arquivos de *log*. Porém, como a aplicação em questão não existe, será tomado como base os objetivos já capturados. Assim, seguindo as estratégias proposta pelo método, obtém-se a tabela abaixo:

Ameaça	**Vulnerabilidades**	**Fonte de Ameaça**	**Motivação**	**Ações que podem levar a um ataque**	**Probabilidade**	**Estratégia**
Acesso não autorizado	✓ Senhas de empregados demitidos não são excluídas do sistema ✓ Deixar o usuário administrador sem senha	*Hacker, Cracker*	✓ Desafio ✓ Ego ✓ Curiosidade	✓ Cavalos de Tróia ✓ Bugs no sistema ✓ Abertura no Firewall	✓ Médio	✓ Manter o anti-vírus atualizado ✓ Limitar acesso ao firewall a somente o adiminstrador
Firewall permite *telnet*	✓ Permitir acesso remoto	*Cracker*	✓ Adquirir acesso de administrador	✓ Usuário *guest* com esse tipo de permissão	✓ Baixo	✓ Configurar o *firewall* de modo a não permitir acesso remoto
Sistemas de adivinhação de senhas	✓ Permitir nome de usuário e/ou senhas nulas ✓ Utilizar nomes de usuários e senhas óbvias	*Hacker, Cracker*	✓ Obter acesso para os mais variados fins	✓	✓ Alto	✓ Bloquear o acesso temporariamente após 3 tentativas mal sucedidas ✓ Uso de bloqueadores *antispam* ✓ Criticar usuários e senhas óbvias
Roubo de *cookies*	✓ Navegador não configurado de forma correta ✓ Gravação de *cookies* não criptografados	*Hacker, Cracker*	✓ Explorar dados privados de um usuário	✓ Injeção de scripts no lado do cliente ✓ Uso do Eavesdropping (escuta secreta)	✓ Alto	✓ Utilização de SSL para criptografar todo o tráfico ✓ Desabilitar o recebimento automático de *cookies*, exceto para *sites* confiáveis
Sessões com *timeout* muito grande	✓ Permitir que o usuário continue executando uma página mesmo que esta tenha ficado um longo período sem uso	*Hacker, Cracker*		✓ Controle de uma sessão por terceiros	✓ Baixo	✓ Impor limites de tempos para uma sessão permanecer ativa
Clonagem do número de cartão de crédito	✓ Armazenar o número do	*Hacker, Cracker*	✓ Ganho financeiro	✓ Uso de programas como *keylogger*	✓ Médio	✓ Usar o número do cartão apenas durante o processo da compra

	cartão			para capturar informações digitadas		✓ Uso de certificação digital entre as duas extremidades (cliente-servidor)
Cavalo de Tróia	✓ Máquinas com anti-vírus e antispam desatualizados	*Hacker, Cracker*	✓ Roubo de dados confidenciais ✓ Obtenção de ganhos financeiros ✓ Etc.	✓ Uso de *trojans* como Subseven e o Milenium	✓ Alto	✓ Manter anti-vírus e antispam atualizados
Negação de Serviço *(Denial of Service ou Distributed Denial of Service)*	✓ Servidor configurado para aceitar um número limite de requisições por sessão	*Cracker*	✓ Concorrências de mercado ✓ Denegrir a imagem de uma empresa	✓ Utilização de programas que gerem um número alto de requisições para os servidores ✓ Utilização de máquinas Zumbis	✓ Alto	✓ Configurar os servidores *web* de modo a evitar esse tipo de ataque
Utilização de *Backdoors*	✓ Configuração incorreta de um *firewall*	*Hacker, Cracker*	✓ Tornar a acessar uma máquina que já foi invadida	✓ Cavalo de tróia ✓ *Firewall* permitindo acesso a portas altas ✓ Uso de ferramentas como: *BackOrifice* e *NetBus*	✓ Alto	✓ Configuração do Firewall de modo a manter controle de acesso às portas altas

Tabela 4.8 – Avaliação de Riscos e Vulnerabilidades

É importante ressaltar que, como previsto na fase seis, dependendo das estratégias adotadas para a atenuação de riscos, pode ser necessária novamente a execução de uma ou mais fases anteriores do método (conforme figura 3.1). Como exemplo, o cenário C2.1 deveria ser reescrito, acrescentando-se o seguinte teste "*if* número de tentativas = 3", bloquear temporariamente o usuário.

Analisando os objetivos elicitados e usando os modelos apresentados na fase sete, já se têm condições de fazer um esboço das políticas de segurança e privacidade, o qual só será completado após a conclusão da fase avaliação de obediência, uma vez que é possível encontrar conflitos entre tais políticas e os requisitos operacionalizados.

Outro ponto importante a se destacar é que a execução dessa fase só foi necessária porque, no exemplo proposto, as políticas de segurança e privacidade não existiam. Caso contrário, poderia se passar diretamente para a fase oito, avaliação de obediência.

Abaixo estão descritos os esboços das políticas de segurança e privacidade:

Política de Segurança
Simara Rocha
27/04/2005
Versão: 1

1. Introdução

1.1. Declaração de Propósito

A livraria Vinícius de Moraes tem total compromisso e respeito com o cliente quanto à segurança de seus dados durante todo o processo de compra.

Para demonstrar esse compromisso, abriu esse espaço para esclarecer a você, cliente, nossa conduta diante das informações pessoais que você nos confidencia. Assim, ressaltamos que tais informações fornecidas no processo de compra são criptografadas e totalmente processadas por computador, sem qualquer intervenção humana.

A presente política poderá ser alterada a qualquer tempo, já que este documento

não cria qualquer vínculo contratual entre a Vinícius de Moraes e seus clientes ou terceiros. Porém, toda e qualquer alteração feita será vinculada neste espaço.

2. Políticas

2.1. Política de identificação e autenticação

Para facilitar as futuras compras, a livraria Vinícius de Moraes criou um sistema de identificação, onde o cliente faz um cadastro informando seus dados pessoais, ou seja, nome, CPF, endereço, telefone e CEP, e opcionalmente, data de nascimento e *email*. Além disso, o cliente deverá criar um usuário e uma senha de acesso, que será solicitada sempre que este entrar em qualquer área restrita do *site*.

Dessa forma, sempre que o cliente desejar fazer uma compra ou até mesmo consultar históricos dos pedidos já realizados, basta informar apenas o usuário e a senha de acesso, não havendo, portanto, necessidade de preencher novamente um cadastro.

2.2. Política de criptografia

Todas as informações que passam pelo nosso processo de compra, ou seja, dados pessoais, forma de pagamento escolhida ou qualquer outra informação fornecida a esta livraria, são automaticamente codificadas por um sistema tecnológico específico antes de ser transmitida. Observe o ícone "cadeado fechado" localizado na barra de *status* do seu navegador. Esse é o símbolo da criptografia das informações.

2.3. Política de integridade/segurança

Para proteger suas informações de quaisquer violações, a Vinícius de Moraes utiliza o padrão da indústria de Internet para segurança (SSL). Este software permite encriptar (codificar) toda a sua informação pessoal, incluindo cartão de crédito, tornando totalmente impossível alguém obter esta informação pela rede.

2.4. Política de senhas

Mesmo trabalhando com o que há de mais moderno em matéria de tecnologia para proteger seus dados, lembre-se que seu usuário e senha são de uso pessoal e intransferível, não devendo ser informados, em hipótese alguma, a terceiros.

2.5. Política de uso

A livraria Vinícius de Moraes se responsabilizará pela garantia da segurança de todas as informações prestadas por nosso cliente, sendo que são de inteira responsabilidade dos mesmos os dados pessoais fornecidos, o número de cartão de crédito (eventualmente utilizado), bem como a alterações dos dados cadastrais.

Política de Privacidade

Simara Rocha
27/04/2005
Versão: 1

1. Introdução

1.1. Declaração de Propósito

A livraria Vinícius de Moraes tem total compromisso e respeito com o cliente quanto à privacidade de seus dados durante todo o processo de compra.

Para demonstrar tal compromisso, abriu esse espaço para esclarecer a você, cliente, nossa conduta diante das informações pessoais que você nos confidencia. Assim, ressaltamos que tais informações fornecidas no processo de compra são criptografadas e totalmente processadas por computador, sem qualquer intervenção humana.

A presente política poderá ser alterada a qualquer tempo, já que este documento, já que o este documento não cria qualquer vínculo contratual entre Vinícius de Moraes e seus clientes ou terceiros. Porém, toda e qualquer alteração feita será vinculada neste espaço.

2. Princípios

2.1. Noticia/Conhecimento

A livraria Vinícius de Moraes apenas coleta seus dados nas seguintes situações: no processo de compras, no cadastro de nossas promoções, ao responder uma pesquisa e no cadastro de *email*, para receber as novidades do *site*.

Tais informações poderão, entretanto, ser agrupadas conforme determinados critérios e utilizadas como estatísticas genéricas, objetivando um melhor entendimento do perfil do consumidor.

2.2. Escolha/Consentimento

O recebimento de *emails*, participação de promoções ou pesquisas, bem como *downloads* de arquivos ou *softwares* que, porventura, existam no *site*, serão sempre condicionados ao consentimento do cliente, o qual será informado, de maneira clara e objetiva, o propósito cada um. Convém ressaltar, que tais autorizações podem ser revogadas a qualquer momento pelo mesmo.

2.3. Não Repudiação

A Vinícius de Moraes é certificada pela Verisign, a maior autoridade em segurança na *internet*, o que garante que nossos clientes podem fornecer tranqüilamente seus dados no processo de compra.

2.4. Acesso/Participação

A qualquer momento o nosso cliente pode ter acesso ao seu cadastro junto a Vinícius de Moraes, bastando para tanto, informar o usuário e a senha de acesso. E, assim, adicionar, alterar ou até mesmo excluir seu cadastro existente em nosso *site*.

3. Apêndice

3.1. Glossário

- ✓ O SSL é um protocolo de segurança que criptografa os dados enquanto estão sendo transmitidos pela *internet*. A criptografia é feita em segundo plano, sem qualquer interação do usuário, portanto, não precisa sequer digitar senhas.

Após o preenchimento inicial dos modelos, dá-se início a fase oito do método, que é a avaliação de obediência, cuja finalidade é verificar se as políticas existentes (ou criadas) estão de acordo com os requisitos elicitados anteriormente. Como exemplo, as tabelas 4.9 e 4.10 trazem, respectivamente, uma avaliação preliminar das políticas de segurança e privacidade.

Declarações da Política de Segurança		**REQUISITOS**		
		Utilizar SSL para criptografar as informações	Manter armazenado apenas os dados pessoais	Impedir cadastro de usuários e/ ou senhas óbvias
	Informações de compras são codificadas	✓		
	Coleta de dados do cliente para compra	✓	×	
	Criação de usuário e senha de acesso	✓	✓	✓

Tabela 4.9 – Avaliação de obediência com a política de segurança

Declarações da Política de Privacidade		REQUISITOS			
		Dados pessoais visíveis somente ao perfil de usuários	Suspender acesso após 3 tentativas	Personalizar perfil de usuário	Solicitar *email* para o envio de mensagens ofertas
	Autenticação é requerida para acesso as páginas restritas do *site*	✓	✓		
	Agrupamento de informações pessoais para fins estatísticos			×	
	Participação em promoções ou pesquisas				✓
	Alteração das informações cadastrais	✓		✓	✓

Tabela 4.10 – Avaliação de obediência com a política de privacidade

Como pode ser evidenciada pela tabela 4.9, a declaração "coleta de dados do cliente para compra" é conflitante com o requisito "manter armazenado apenas os dados pessoais". Isso significa que é preciso atualizar a política de segurança, ou seja, deixar claro que o número do cartão de crédito não é armazenado, sendo utilizado somente no momento da compra.

Por outro lado, a tabela 4.10 mostra que a declaração "agrupamento de informações pessoais para fins estatísticos" é conflitante com o requisito "personalizar perfil de usuário". Assim, é necessário atualizar a política de privacidade deixando claro que as estatísticas são feitas sobre informações genéricas e não objetivam a troca ou comercialização das mesmas.

Finalmente, por ser um exemplo didático, cuja finalidade é mostrar, passo a passo, a aplicabilidade do método proposto, não foi feito o preenchimento do documento de especificação de requisitos sugerido nesse trabalho. Porém, seu preenchimento poderá ser feito de forma paralela à execução das fases do método.

5 CONCLUSÃO

5.1 Contribuições do trabalho

Este trabalho fez um estudo de algumas abordagens baseadas em objetivos e cenários e apresentou uma proposta de um método para a fase da Engenharia de Requisitos voltado para sistemas de comércio eletrônico em obediência com as políticas de segurança e privacidade existentes em um *site*.

O principal foco foi garantir que tais políticas nunca se tornem obsoletas pela adoção de novas funcionalidades ao *site*. Assim, à medida que novas tecnologias ou funcionalidades são adotadas, os requisitos elicitados deverão estar em conformidade com as mesmas.

Outro ponto importante provido por essa abordagem é a habilidade de poder criar, através de modelos fornecidos pela mesma, as políticas de segurança e privacidade, caso não existam. Dessa maneira, essa fase foi inserida no método como forma de garantir que tais políticas fossem especificadas, uma vez que a adoção de modelos visou estabelecê-las de maneira mais clara, menos ambígua e contendo, apenas, os aspectos mais relevantes, beneficiando, assim, tanto a organização quanto os usuários de um *site* de comércio eletrônico.

Esse método foi originado a partir da integração das abordagens UWA [33] com o método GBRAM [6], instanciado para o desenvolvimento de políticas e requisitos para sistemas de comércio eletrônico, consistindo em um modelo de processo que foi aplicado a um estudo de caso para elicitar os requisitos de uma livraria *on-line*.

Ao método UWA foi adicionada uma escala numérica ascendente de valor, a qual é estabelecida por cada *stakeholder* para os objetivos por ele especificado, cuja finalidade é

possibilitar resolver possíveis conflitos que possam existir entre o conjunto de objetivos elicitados, e também ajudar na fase posterior de refinamento de objetivos.

No que concerne à instanciação do método GBRAM, foi adicionada a fase de avaliação de riscos, a identificação do par (ameaça, vulnerabilidade) e uma estimativa de probabilidade de ocorrência de risco para cada par, todos baseados em [32], por serem padrões largamente usados em grandes *sites.*

A abordagem proposta ainda sugeriu um modelo para o documento de especificação de requisitos como forma de estabelecer um meio padrão para especificar requisitos de software, cuja finalidade é facilitar as futuras manutenções do sistema e também ajudar na fase posterior de validação dos requisitos elicitados.

Por fim, o método proposto nesta dissertação possui limitações. Uma delas consiste em alcançar apenas os requisitos funcionais, uma vez que os não funcionais não foram considerados.

Outra limitação observada é a não existência de normas técnicas para o preenchimento dos modelos apresentados, o que poderia contribuir para a formalização do método proposto.

5.2 Trabalhos futuros

Sugere-se como trabalhos futuros a extensão desse método para requisitos não funcionais. Assim, o método englobaria tanto aspectos funcionais quanto não funcionais dos requisitos. Em [11] é apresentada uma estratégia para lidar com requisitos não funcionais desde as primeiras etapas do processo de desenvolvimento de software, a qual poderia ser integrada ao método.

Outra sugestão é a implementação de uma ferramenta de suporte ao desenvolvimento do método, a qual poderia ser desenvolvida para automatizar desde as

atividades de identificação dos *stakeholders*, elicitação dos objetivos, atribuição de valores aos objetivos, especificação dos cenários, operacionalização dos objetivos, preenchimento dos modelos das políticas de segurança e privacidade, até a geração do documento de especificação de requisitos.

Tornar a abordagem proposta reutilizável poderia ser uma outra extensão deste trabalho, uma vez que se poderiam obter ganhos de tempo, custo e esforços da equipe de desenvolvimento.

Finalmente, a criação de normas técnicas para os modelos apresentados nas fases cinco e oito do método poderia contribuir para o maior formalismo do mesmo.

REFERÊNCIAS

[1] ACHOUR, C. Ben. **Guiding Scenario Authoring**. Proc. Eighth European Japanese Conf. Information Modeling and Knowledge Bases, *pp. 181-200,* Ellivuori, Finland,May 1998.

[2] ACHOUR, C. B.; MUSTAPHA, T.; SOUVEYET, C. **Bridging the gap between users and requirements engineering: the scenario-based approach**. Computer Systems Science and Engineering, v14, n6, Nov, 1999, p 379-388.

[3] ACHOUR, C. B.; ROLLAND, C.; SOUVEYET, C., **A proposal for improving the quality of scenario collections**. Proceedings of the Fouth International Workshop on Requirements Engineering: Foundations of Software Quality, REFSQ´98, Pisa, Italy Presses Universitaires de Namur (eds, E. Dubois, A. L. Opdhal, K. Pohl), pp. 29-42, 1998.

[4] ANTÓN, I. Annie. **Goal-Based Requirements Analysis**. Second IEEE International Conference on Requirements Engineering (ICRE ‘96), Colorado Springs, Colorado, pp. 136-144, 15-18 April 1996.

[5] ANTÓN, I. Annie. **Goal Identification and Refinement in the Specification of Software-Based Information Systems.** Ph.D. Dissertation, Georgia Institute of Technology, Atlanta, GA, 1997.

[6] ANTÓN, I. Annie; EARP, B. Julia, **Strategies for Developing Policies and Requirements for Secure Eletronic Commerce System.** 1º Workshop on Security and Privacy in E-Commerce at CCS2000, November 2000.

[7] ANTÓN, I. Annie; POTTS, Colin. **The Use of Goals to Surface Requirements for Evolving Systems**, International Conference on Software Engineering (ICSE ‘98), Kyoto, Japan, pp. 157-166, 19-25 April 1998.

[8] BOLCHINI, D.; ANTÓN I. A.; STUFFLEBEAM, W. **I need it now: Improving Website Usability By Contextualizing Privacy Policies.** To appear: The 4th International Conference on Web Engineering (ICWE 2004), Munich, Germany, 28-30 July 2004.

[9]BOLCHINI, D.; PAOLINI, P. **Capturing Web Application Requirements through Goal-Oriented Analysis**. WER, pp. 17-28, 2002.

[10] CARVALHO, M.; ABDELOUAHAB, Z. **Um Método para Elicitação e Modelagem de Requisitos baseado em Objetivos.** WER, pp. 319-337, 2001.

[11] CYSNEIROS, L. M.; LEITE, J.C.S. P. **Requisitos não funcionais: Da Elicitação ao Modelo Conceitual**. Tese de Doutorado submetida na PUC-Rio de Janeiro, em fevereiro de 2001.

[12] DARDENNE, A., LAMSWEERDE, V. A. and FICKAS, S. **Goal-directed Requirements Acquisition**. Science of Computer Programming, 20 (1-2): 3-50, April 1993.

[13] EARP, B. J.; PAYTON C. F. **Information Privacy Concerns Facing Health Care Organizations in the New Millennium.** Submitted to Information Systems Research, April, 2000.

[14] GORDIJN, J. **Value based Requirements Engineering – Exploring Innovative e-commerce ideas**. 2002. Tese PhD, Vrije Universiteit, Amsterdam. Disponível em <http://www.cs.vu.nl/~gordijn/>.

[15] HAUSER, R. J.; CLAUSING, D. **The House of Quality.** Harvard Business Review, 32(5), pp. 63-73,1988.

[16] JAISINGH, J.; REES, J. **Value at Risk: A Metodology for Information Security Risk Assessment.** Purdue University, West Lafayette, IN, 2000.

[17] LEITE, Júlio César Sampaio Leite do Prado et al. **Enhancing a Requirements Baseline with scenarios**. In Third IEEE International Symposium on Requirement Engineering RE'97, Antapolis, Maryland, IEEE Computer Society Press, pp. 44-53, 1997.

[18] LEITE, Júlio César Sampaio Leite do Prado et al. **A Scenario Construction Process. In on Requirement Engineering.** Antapolis, Maryland, IEEE Computer Society Press, pp.38-61. 2000.

[19] MEIRA, W. J; MURTA, C. D.; RESENDE, R. **Comércio Eletrônico na WWW**. São Paulo: IME-USP, 2000.

[20] PRESSMAN, Roger. **Engenharia de Softwares**. 3ª ed., São Paulo: Editora Makron Books do Brasil Ltda., 1995. 1056p.

[21] Policy Framework for Interpreting Risk in eCommerce Security. CERIAS Technical Report, Purdue University, 1999.

[22] POTTS, C. **Scenic: A Strategy for Inquiry-Driven Requirements Determination.** Proceedings IEEE 4th International Symposium on Requirements Engineering (RE'99), Limerick, Ireland, 7-11 June 1999.

[23] POTTS, C., TAKAHASHI, K. and ANTON, I. A. **Inquiry-Based Requirements Analysis**. IEEE Software, 11(2): 21-32, March 1994.

[24] ROCHA, V. S. **Um Modelo para o Documento de Especificação de Requisitos baseado no Processo Unificado/UML.** Monografia de Conclusão de Curso, MA, 2001.

[25] ROCHA, Simara; ABDELOUAHAB, Zair e FREIRE, Eduardo. **Requirement Elicitation Based on Goals with Security and Privacy Policies in Electronic Commerce.** WER, pp 63-74, 2005.

[26] ROLLAND, C.; SOUVEYET, C.; ACHOUR, C. B. **Guiding Goal Modeling Using Scenarios.** IEEE Transactions on Software Engineering, 24(12), pp. 1055-1071, December 1998.

[27] SEINAUER, D.; KATZKE, S.; RADACK, S. **Basic Intrusion Protection: The First Line of Defense.** IT Professional (IEEE Computer Society), 1(1), pp. 43-48, 1999.

[28] SÊMOLA, M. **Gestão de Segurança da Informação – Uma visão executiva**. 3.Ed. Rio de Janeiro: Elsevier, 2003. 160p.

[29] SILVA, C. J; ROMANI, R.; MELO E. T. **Comércio Eletrônico: Modelos de Negócios na Internet**. Relatório Técnico, UNICAMP. Disponível em <www.dcc.unicamp.br/~ra015057/mp205/artigo.html>, 2002.

[30] SOMMERVILLE, I.; KOTONYA, G. **Requirements Engineering: Processes and Techniques**. Jonh Wiley & Sons, 1997.

[31] SOMMERVILLE, I.; SAWYER, P. **Requirements Engineering: a good practice guide**. Jonh Wiley & Sons, 1998.

[32] STONEBURNER, G.; GOGUEN, A.; FERINGA, A.. **Risk Management Guide for Information Technology Systems.** NIST Special Publication 800-30, July 2002.

[33] UWA Consortium, **Requirements and Design Specification for Banca 121 Pilot Application.** UWA Project Deliverable D11. Disponível em <www.uwaproject.org>, 2001.

[34] UWA Consortium, **Evaluation of UWA Design Methodology.** UWA Project Deliverable D13. Disponível em <www.uwaproject.org>, 2001.