



A IMPORTÂNCIA DAS TÉCNICAS DE LEVANTAMENTO DE REQUISITOS NO PROCESSO DE DESENVOLVIMENTO DE SOFTWARE

THE IMPORTANCE OF REQUIREMENT SURVEY TECHNIQUES IN THE SOFTWARE DEVELOPMENT PROCESS

Micheli Chichinelli¹

RESUMO: O passo fundamental para o sucesso, em qualquer processo de desenvolvimento de software, é a definição e análise de requisitos, porque, mesmo que um sistema tenha sido bem projetado e codificado, se ele não estiver bem especificado, com certeza, irá causar danos e transtornos para o cliente e para os desenvolvedores. Nesse sentido, o objetivo deste artigo é abordar a importância da Engenharia de Requisitos no processo de desenvolvimento de software, apresentando várias técnicas de levantamento de requisitos que possam contribuir para uma definição e análise de requisitos de maior qualidade.

Palavras-chave: Processo de desenvolvimento de software. Engenharia de requisitos. Técnicas de Levantamento de Requisitos.

ABSTRACT: *The key step to success in any software development process is the definition and analysis of requirements because, even if a system has been well designed and coded, if it is not well specified, it will certainly cause damage and customer and developer disorders. In this sense, the objective of this article is to address the importance of Requirements Engineering in the software development process, presenting several requirements surveying techniques that can contribute to the definition and analysis of higher quality requirements.*

Keywords: *Software development process. Requirements engineering. Requirement survey techniques.*

¹ Mestre em Engenharia de Produção, USP, 2002.

1 Introdução

Desenvolver um sistema, sem possuir um entendimento completo e correto dos requisitos pode trazer problemas, pois, certamente não atenderá a todas as necessidades dos usuários.

Uma má definição de requisitos nos estágios iniciais do processo de desenvolvimento pode resultar em altos custos de manutenção do sistema, pois diversas tarefas do ciclo de desenvolvimento de software deverão ser refeitas.

Para ALENCAR (1999), um processo de requisitos inadequado faz com que os usuários percam a confiança na equipe de desenvolvedores. Para a autora, trata-se de um processo difícil de ser bem realizado, mas de fundamental necessidade para o bom funcionamento do sistema desejado.

Diante dos vários problemas encontrados na fase de definição de requisitos, a Engenharia de Requisitos surge para tentar solucionar ou minimizar os mesmos. Neste sentido, o objetivo deste artigo é apresentar algumas técnicas de levantamento de requisitos que são utilizadas durante o processo de desenvolvimento de software. Estas técnicas, utilizadas muitas vezes de forma combinada, possibilitam uma definição de requisitos mais consistente e completa.

Neste contexto, para melhor entendimento sobre as técnicas de levantamentos de requisitos, será apresentada na próxima seção inicialmente a definição de requisitos.

2 Principais Definições de Requisitos

Antes de definir o Processo de Engenharia de Requisitos, inicialmente, é fundamental definir o que são requisitos.

Uma boa definição para requisitos é especificada por Leite (1994). Para o autor requisito é a condição necessária para a obtenção de certo objetivo, ou para preenchimento de certo objetivo.

Para Alencar (1999), requisitos são um conjunto de descrições de como o sistema pretendido deve se comportar ou um conjunto de propriedades, de atributos do sistema e/ou limitações do próprio processo de desenvolvimento do sistema.

Segundo Kotonya e Sommerville (1995), os requisitos refletem as necessidades dos clientes e usuários de um sistema. Eles devem incluir: justificção para a construção do

sistema; o que é pretendido que o sistema faça, e quais as restrições de projeto a serem observadas.

Uma definição bem simples para requisitos é dada por Macaulay (1996), onde requisitos são simplesmente algo de que o cliente necessita.

Depois das definições apresentadas, na próxima seção será caracterizado o Processo de Engenharia de Requisitos sob o ponto de vista de alguns autores.

3 Processo de Engenharia de Requisitos

Segundo Sommerville (2011), o processo de engenharia de requisitos tem como objetivo produzir um documento de requisitos acordados que especifica um sistema que satisfaz os requisitos dos *stakeholders*. Existem 4 atividades principais do processo de engenharia de requisitos.

- a) **Estudo de Viabilidade:** É feita uma estimativa acerca da possibilidade de se satisfazerem as necessidades do usuário identificado usando-se tecnologias atuais de software e hardware. O estudo considera se o sistema proposto será rentável a partir de um ponto de vista de negócio e se ele pode ser desenvolvido no âmbito das atuais restrições orçamentais. O resultado deve informar a decisão de avançar ou não, com uma análise mais detalhada.
- b) **Elicitação e Análise de Requisitos:** Esse é o processo de derivação dos requisitos do sistema por meio da observação dos sistemas existentes, além de discussões com os potenciais usuários e compradores, análise de tarefas, entre outras etapas. Aqui 4 atividades são realizadas: a) *Obtenção/Descoberta de requisitos:* Essa é atividade de interação com os *stakeholders* do sistema para descobrir seus requisitos. Existem várias técnicas complementares que podem ser usadas para descoberta dos requisitos. b) *Classificação e organização de requisitos:* Essa atividade toma a coleção de requisitos não estruturada, agrupa os requisitos relacionados e os organiza em grupos coerentes. c) *Priorização e negociação de requisitos:* Inevitavelmente, quando os vários *stakeholders* estão envolvidos, os requisitos entram em conflito. Esta atividade está relacionada com a priorização de requisitos e em encontrar e resolver conflitos por meio de negociação de requisitos. Normalmente, os *stakeholders* precisam se encontrar para resolver as diferenças e chegar a um acordo sobre os requisitos. d)

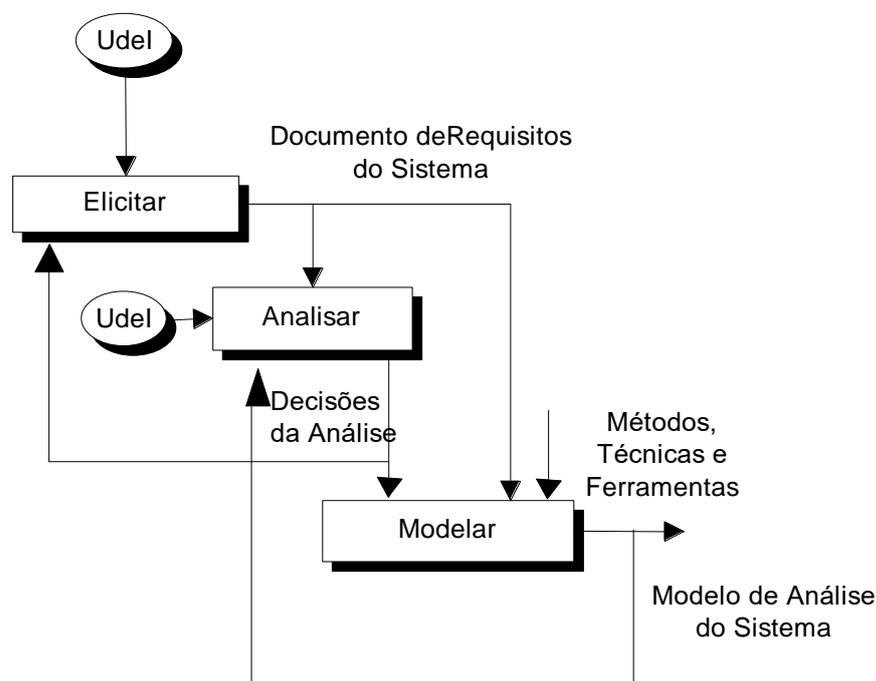
Especificação de requisitos: Os Requisitos são documentados. Documentos formais e informais podem ser produzidos.

- c) **Especificação de Requisitos:** É a atividade de traduzir as informações obtidas durante a atividade de análise em um documento que defina um conjunto de requisitos. Dois tipos de requisitos podem ser incluídos nesse documento. Requisito do usuário e Requisito do Sistema.
- d) **Validade de Requisitos:** Esta atividade verifica os requisitos quanto a realismo, consistência e completude. Durante esse processo, os erros no documento de requisitos são inevitavelmente descobertos. Em seguida, o documento deve ser modificado para correção desses problemas.

Já Leite (1994) especifica outra visão sobre o processo de Engenharia de Requisitos e o divide em três atividades: Elicitar, Modelar e Analisar. Esta divisão é muitas vezes difícil de se distinguir na prática, pois a ligação entre as partes é bastante forte. Por outro lado, isto ajuda, de forma eficiente, no entendimento do processo como um todo. Para o autor, produzir um documento de requisitos completo e consistente — produto da Engenharia de Requisitos — implica na necessidade inicial de entender melhor o contexto em que o problema se situa, ou seja, quais são os objetivos do produto a ser desenvolvido, as tarefas/atividades fundamentais para a “engenharia” deste produto e os limites do desenvolvimento. Assim, para especificar corretamente os requisitos é necessário definir o UdI- Universo de Informações (LEITE, 1994). O UdI é o contexto geral no qual o software deverá ser desenvolvido e, inclui todas as fontes de informação e todas as pessoas relacionadas ao software, as quais denominamos de agentes deste universo. O UdI é a realidade circunstanciada pelo conjunto de objetivos definidos por quem solicitou o software.

Na Figura 1, é apresentado o processo de engenharia de requisitos, segundo Leite (1994).

Figura 1 – Processo de Engenharia de Requisitos



Fonte: LEITE (1994).

Este processo tem início na fase de elicitação, identificadora dos fatos que compõem os requisitos do sistema, de forma a prover o mais correto e completo entendimento do que é demandado naquele software. Muitas técnicas utilizadas na elicitação provêm principalmente das Ciências Sociais como a Sociologia e a Psicologia. Já a modelagem objetiva descrever o problema de uma forma sistemática através da representação dos requisitos na forma de um modelo. Na análise, os requisitos são confrontados com a realidade para constatar sua correteza.

Já Locopoulos e Karakostas (1995 apud ALENCAR, 1999), formula o processo de Engenharia de Requisitos considerando três atividades principais: Elicitação, Especificação e Validação, que correspondem às três preocupações importantes na Engenharia de Requisitos: Compreensão, Descrição e Concordância de um problema.

Pode-se notar, que independente do processo de Engenharia de Requisitos utilizado, o resultado é um documento de requisitos do software. Este documento (algumas vezes chamado de especificação de requisitos de software) é a declaração oficial do que é requerido dos desenvolvedores do sistema.

Durante o processo de especificação de requisitos, surge a necessidade de estabelecer o tipo de requisito de que se esteja tratando, a fim de melhorar a compreensão

das necessidades do cliente. Neste sentido, na próxima seção será apresentada a classificação dos requisitos.

4 Classificação dos Requisitos

Os requisitos podem ser funcionais ou não funcionais. Os requisitos funcionais dizem respeito à definição das funções que um sistema ou um componente de sistema deverá fazer, ou seja, as entradas que deverão ser transformadas e as saídas que deverão ser produzidas.

Já os requisitos não funcionais, dizem respeito às restrições e comportamentos que o software deverá satisfazer.

Segundo Turine e Masieiro (1996), os requisitos não funcionais ou de qualidade incluem tanto limitações no produto (desempenho, confiabilidade e segurança) como limitações no processo de desenvolvimento (custos, métodos a serem adotados no desenvolvimento e componentes a serem reutilizados).

De forma geral, a diferença entre requisitos funcionais e não funcionais está no fato dos primeiros descreverem **o que** o sistema deverá fazer, enquanto os outros fixam restrições sobre **como** os requisitos funcionais serão implementados (ALENCAR, 1999).

Os requisitos funcionais e não funcionais, em princípio, devem ser distinguidos numa especificação de requisitos. Todavia, nem sempre é simples estabelecer a diferença entre o que seja um requisito funcional ou não funcional. Fazer uma separação bem definida fica um tanto difícil, pois alguns requisitos exprimem necessidades do sistema como um todo e não limitações, em funções individuais.

Os requisitos funcionais e não funcionais são derivados dos requisitos organizacionais, que por sua vez, são definidos em função do domínio da aplicação. Os requisitos organizacionais (DAVIS,1993; BUBENKO,1993) dizem respeito às metas da empresa, suas políticas estratégicas adotadas, os empregados da empresa com seus respectivos objetivos, os processos, informações, enfim representam a estrutura da organização.

De acordo com ALENCAR (1999), até pouco tempo, os projetistas e desenvolvedores ignoravam a importância de alguns aspectos organizacionais para o desenvolvimento de seus produtos. Só mais recentemente é que esses aspectos começaram a ser observados e foram considerados essenciais ao sucesso das implementações. Para a

autora, tem-se atualmente nos modelos, ausência de componentes intencionais (as razões, as motivações, os porquês).

Depois destas classificações, na próxima seção, serão apresentadas as algumas técnicas de Levantamento de Requisitos.

5 Técnicas de Levantamento de Requisitos

Um dos principais desafios no desenvolvimento de software é a definição dos requisitos do sistema. Para que um software atenda às reais necessidades do cliente, as técnicas de levantamento de requisitos são essenciais. Neste contexto, algumas técnicas de levantamento de requisitos são apresentadas a seguir.

Entrevista

A técnica de entrevista é simples e bastante eficiente na fase de levantamento dos requisitos, na qual o analista será o entrevistador que obtém os dados informados pelo cliente.

De acordo com Sommerville (2011), as entrevistas podem ser de dois tipos: a) *Entrevistas Fechadas*, em que os *stakeholders* (*partes interessadas*) responde a um conjunto predefinido de perguntas. b) *Entrevistas Abertas*, em que não existe uma agenda predefinida. A equipe de engenharia de requisitos explora uma série de questões com os *stakeholders* do sistema, e assim, desenvolve uma melhor compreensão de suas necessidades. Entrevistas são boas para obter uma compreensão global sobre o que os *stakeholders* fazem, como eles podem interagir com o novo sistema e as dificuldades que eles enfrentam com os sistemas atuais.

Questionário

Assim como a entrevista, a técnica de questionário contém perguntas a serem respondidas pelos usuários.

Segundo Figueira (2012), o questionário é uma técnica de elicitação de requisitos que pode abranger um grande de número de pessoas. O seu uso é essencial quando se deseja obter informações de inúmeras pessoas. O autor ainda justifica a aplicabilidade do

questionário quando há indisponibilidade física, dispersão das pessoas envolvidas no projeto ou, até mesmo, quando há necessidade de um levantamento estatístico das pessoas que utilizaram o sistema.

Um questionário pode ser composto por dois tipos: subjetivo e objetivo. Um questionário subjetivo contém perguntas que possibilitam ao usuário descrever sua opinião, ou seja, perguntas abertas. Porém, é importante tomar os devidos cuidados, pois a resposta do usuário pode se tornar de difícil interpretação. Já o questionário objetivo contém respostas claras e diretas, evitando diferentes respostas para usuários distintos. Esse questionário pode ser mais bem aplicado quando o analista tem conhecimento para listar as possíveis respostas.

Casos de Uso

A técnica de modelagem através de casos de uso está sendo utilizada cada vez mais pelos engenheiros de requisitos. A modelagem baseada em casos de uso possui uma notação gráfica simples e uma descrição em linguagem natural o que facilita a comunicação entre o desenvolvedor e o usuário.

De acordo com Melo (2010), um caso de uso descreve uma sequência de ações que representam um cenário perfeito e cenários alternativos, com o objetivo de demonstrar o comportamento de um sistema (ou parte deles), através de interações com atores. O caso de uso deve descrever uma rotina bem definida do sistema e deve ser totalmente compreensível tanto para a equipe de desenvolvimento quanto para os clientes que detêm o conhecimento do domínio do sistema.

Os modelos de casos de uso são compostos por atores, casos de uso e relacionamentos entre eles. Um caso de uso é a representação de uma sequência de interações entre um sistema e os atores. Os atores são agentes externos que utilizam esse sistema (BEZERRA, 2006).

Etnografia

De acordo com Sommerville (2011), a etnografia é uma técnica de observação que pode ser usada para compreender os processos operacionais e ajudar a extrair os requisitos de apoio para esse processo. O analista faz uma imersão no ambiente de trabalho em que o sistema será usado. O

trabalho do dia a dia é observado e são feitas anotações sobre as tarefas reais em que os participantes estão envolvidos. O autor argumenta que a etnografia ajuda a descobrir os requisitos implícitos do sistema que refletem as formas reais com que as pessoas trabalham, em vez de refletir processos formais definidos pela organização.

Por outro lado, Silva (2012), especifica que essa técnica não é apropriada para a descoberta de novos requisitos, pois não é possível observar um requisito que não existe até o momento. Uma outra desvantagem da técnica é que o usuário pode não agir naturalmente na execução das atividades com a presença de uma pessoa externa.

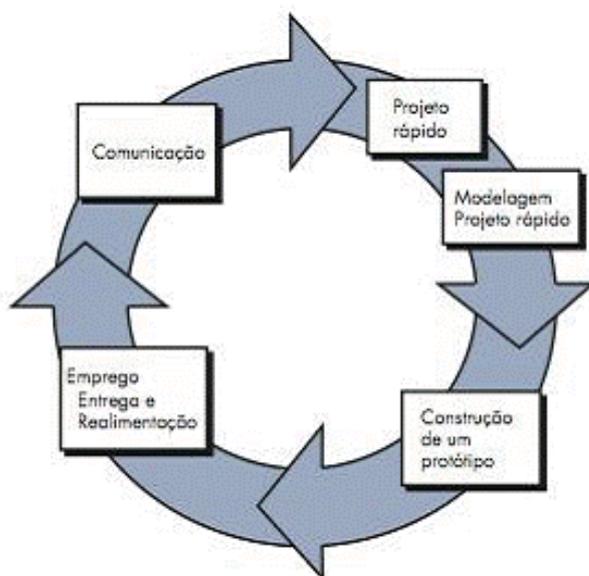
Prototipação

Conforme Pressman (2011) o usuário nem sempre sabe expressar qual sua necessidade, sendo assim o analista se depara com um problema: definir quais são os requisitos a serem levantados para o desenvolvimento do sistema. Uma técnica bem apropriada nesse caso é a prototipação, na qual protótipos de telas são criados contendo as possíveis entradas e saídas do sistema. As etapas da prototipagem conforme figura 2 são:

- a) **Comunicação:** O Engenheiro de software e o cliente encontram-se e definem os objetivos gerais do software, identificam as necessidades conhecidas e delinham áreas que necessitam de mais definições.
- b) **Plano Modelagem e Projeto rápido:** Uma iteração de prototipagem é planejada rapidamente e a modelagem (na forma de “projeto rápido”) ocorre.
- c) **Construção:** O projeto rápido leva à construção de um protótipo. Para elaborar o protótipo, o desenvolvedor tenta usar partes de programas existentes ou aplicar ferramentas (por exemplo, geradores de relatórios, gestores de janelas, etc) que possibilitam programas executáveis serem gerados rapidamente.
- d) **Implantação, Entrega e Feedback:** O protótipo é implantado e depois avaliado pelo cliente/usuário. O feedback é usado para refinar os requisitos.

A iteração ocorre à medida que o protótipo é ajustado para satisfazer às necessidades do cliente, e, ao mesmo tempo, permite ao desenvolvedor entender melhor o que precisa ser feito.

Figura 2: Prototipagem



Fonte: PRESSMAN (2011)

Técnica I*

Quando se procura compreender uma organização, percebe-se que os tipos de conhecimento encontrados nos diversos modelos existentes não são suficientes, uma vez que esses modelos descrevem apenas entidades, funções, fluxos de dados e estados do sistema, não expressando as razões envolvidas no processo, ou seja, o porquê executar uma determinada ação em vez de outra. Assim, faz-se necessário uma ontologia mais rica, capaz de reconhecer motivações, intenções, raciocínios sobre as características de um processo. De acordo com Yu (1995,1997), a técnica I* satisfaz essa necessidade e é utilizada para: (i) obter uma melhor compreensão sobre os relacionamentos da organização, entre os vários atores do sistema; (ii) entender as razões envolvidas nos processos de decisões; (iii) ilustrar as várias características de modelagem que podem ser apropriadas à Engenharia de Requisitos, principalmente na fase inicial da especificação dos requisitos (YU, 1997). Nesta técnica, uma compreensão mais aprofundada do processo pode ser obtida através de uma visão intencional e estratégica. Um processo precisa ser modelado como uma configuração de relacionamentos entre atores estratégicos e intencionais.

A técnica I* é composta por dois modelos: O Modelo de Dependências Estratégicas (DE) e o Modelo de Razões Estratégicas (RE). O Modelo de Dependências Estratégicas

(DE) descreve os relacionamentos de dependência entre os vários atores no contexto organizacional, e o Modelo de Razões Estratégicas (RE) descreve as motivações e preocupações dos *stakeholders*, e as razões que os levam a adotar uma configuração ou outra.

Método VORD - Definição de Requisitos Orientada a Pontos de Vista

Baseado na noção de pontos de vista, Kotonya e Sommerville (1995) desenvolveram um método para a Engenharia de Requisitos, denominado VORD (*Viewpoint-Oriented Requirements Definition*), o qual cobre o processo de Engenharia de Requisitos desde a descoberta dos requisitos iniciais até a modelagem detalhada do sistema. A discussão sobre o método se concentrará nos seus três primeiros passos iterativos:

- a) Identificação e Estruturação de Ponto de Vista
- b) Documentação de Ponto de Vista
- c) Análise e Especificação dos Requisitos de Ponto de Vista.

O primeiro passo, ocupa-se com a identificação e estruturação de pontos de vista relevantes no domínio de problema.

O segundo passo ocupa-se em juntar os pontos de vista identificados no passo 1 e, documentar o nome do ponto de vista, os requisitos com suas restrições e fontes. Requisitos de ponto de vista incluem um conjunto de serviços requeridos, requisitos de controle e um conjunto de requisitos não funcionais.

O último passo ocupa-se em especificar os requisitos funcionais e não funcionais de forma apropriada. A notação usada depende do ponto de vista, do requisito e das fontes de requisitos associadas ao primeiro. Notações apropriadas variam desde linguagem natural (se a fonte de requisitos se ocupa com requisitos não técnicos), passando por equações (se a fonte de requisitos é um físico, por exemplo) até modelos de sistema expressos em notações formais ou estruturadas.

6 Conclusão

Várias são as razões para se estudar e aplicar a Engenharia de Requisitos, contudo o fato principal é a correlação existente entre os erros e inadequações do produto final, com as falhas ocorridas durante as etapas de análise e especificação dos requisitos.

Todas as técnicas de levantamento de requisitos apresentadas neste artigo são importantes na especificação dos requisitos. Todavia deve-se ressaltar que não existe uma técnica que seja melhor que outra, todas apresentam vantagens e desvantagens, dependendo das peculiaridades de cada cliente.

Ao escolher uma técnica ou uma combinação de técnicas para realizar o levantamento de requisitos, vários aspectos devem ser considerados na escolha, tais como: perfil e nível de comprometimento dos *stakeholders*, nível de conhecimento do negócio, nível de qualificação dos profissionais de levantamento de requisitos, objetivos da organização, entre outros.

O propósito do artigo não é realizar uma comparação entre as técnicas de levantamento de requisitos, mas apresentar um conjunto de técnicas, que quando combinadas, podem favorecer um levantamento de requisitos mais completo e consistente, possibilitando a criação de um software que atenda às reais necessidades do cliente.

Referências

- ALENCAR, F. M. R. **Mapeando a modelagem organizacional em especificações precisas**. Recife, 1999. 304f. Tese (Doutorado em Informática), Centro de Informática, Universidade Federal de Pernambuco.
- BEZERRA, E. **Princípios de Análise e Projeto de Sistemas com UML**. 2. ed..Rio de Janeiro: Campus, 2006.
- BUBENKO JR., J. A. Extending the scope of information modelling. In: INTERNATIONAL WORKSHOP ON THE DEDUCTIVE APPROACH TO INFORMATION SYSTEMS AND DATABASE, 4, 1993, Lloret de Mar, Catalonia. **Proceedings...** Departament de Llenguatges i Sistemes Informatics of the Universitat Politecnica de Catalunya, Catalonia, Barcelona: A. Olivé, p.73-98.
- DAVIS, A. **Software Requirements: Objects, Functions and States**. London: Prentice-Hall, 1993.
- FIGUEIRA, A. M. S. **Análise das Técnicas de Levantamento de Requisitos para Desenvolvimento de Software nas Empresas de Vitória da Conquista – BA**. 2012.71f Trabalho de Conclusão de Curso (Graduação), Universidade Estadual do Sudoeste da Bahia, Vitória da Conquista, Bahia,2012.
- KOTONYA, G.; SOMMERVILLE, I. **Requirements Engineering With Viewpoints**. Cooperative System Engineering Group - Technical Report CSEG/10/1995. Computing Department, Lancaster University, 1995.
- LEITE, J.C.S.P. **Engenharia de Requisitos**. Notas de Aula, PUC-RJ, 1994.
- LOUCOPOLOS, P.; KARAKOSTAS, V. **System Requirements Engineering**. London: McGraw-Hill, 1995.
- MACAULAY, L. **Requirements Engineering**. Berlin: Springer Verlag, 1996.
- MELO, A. C. **Desenvolvendo aplicações com UML 2.2: do conceitual à implementação**. 3. ed. Rio de Janeiro: Brasport, 2010.

PRESSMAN, R. S. **Engenharia de software**. 9.ed. São Paulo: McGraw-Hill, 2011.

SOMMERVILLE, I. **Engenharia de Software**. 9.ed. São Paulo: Pearson Prentice Hall, 2011.

TURINE, M. A. S.; MASIERO, P. C. **Especificação de Requisitos**: Uma introdução. Relatório Técnico. ICMSC:USP, 1996.

YU, E. S. **Modelling Strategic Relationships for Process Reengineering**. Toronto,1995. 166f. Tese (Doutorado em Informática) – Department of Computer Science, University of Toronto, Toronto.

YU, E. Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering. In: IEEE INTERNACIONAL SYMPOSIUM ON REQUIREMENTS ENGINEERING, 3, 1997, Annapolis. **Proceedings...** USA: Washington D.C., 1997, p. 226-235.