

A ELICITAÇÃO DE REQUISITOS NO CONTEXTO DE UM PROJETO DE SOFTWARE

Leandro Moura Conceição*

RESUMO

O levantamento de requisitos é à base do desenvolvimento de projetos de software e tem um grande impacto no desenho e outras fases do ciclo de vida do produto. Se feito corretamente, pode ajudar a reduzir as alterações e mudanças dos requisitos. Além disso, a qualidade da elicitação determina a precisão de feedback do cliente sobre a integridade e validade dos requisitos. Esta fase é de alto valor para o projeto, é muito importante que o trabalho de elicitar seja feito tão próximo quanto possível da "perfeição". Considerando diferentes características de projetos de software, este trabalho propõe algumas regras gerais para a realização da Elicitação de requisitos com base em estudos bibliográficos, discussão e explicação dos processos e métodos relacionados aplicados nos diferentes tipos de projetos de software.

Palavras-chave: Engenharia de Requisitos, Elicitação de requisitos, Desenvolvimento de software.

OBJETIVOS

Este trabalho tem como objetivo primordial a explicitação dos conceitos relativos à Elicitação de Requisitos, bem como o estudo de algumas técnicas usadas na execução desta atividade. Além disso, objetiva-se dar uma visão geral do Processo de Engenharia de Requisitos, suas atividade e relatar a importância da elicitação neste contexto.

* Graduado em Tecnologia de Sistemas para Internet na FANESE, pós graduando em MBA de Engenharia de Software na FANESE. Email – leandromouradev@gmail.com

INTRODUÇÃO

É uma realidade que, a maior quantidade de erros e defeitos do software são resultados de uma pobre gestão e definição de requisitos (DIAS, 2015). Em palavras simples, isto significa que cerca da metade dos problemas encontrados poderiam ter sido evitados, simplesmente deixando claro, desde o início, o que o cliente espera do respectivo projeto.

Para melhorar esta situação, a Engenharia Requisitos é a ciência que se propõe a tratar, estabelece e documentar os requisitos de software. Na Versão 1.3 do Capability Maturity Model Integration Development (CMMI-DEV) a disciplina Requisitos de Software é dividida em duas áreas de processo: o primeiro é o desenvolvimento de Requisitos que inclui elicitação, definição, análise, especificação e validação; a segunda é a gestão Requisitos, que envolve o gerenciamento de requisitos desenvolvidos, incluindo o controle de mudança e verificação (OLIVEIRA, 2015).

A elicitação de requisitos é considerada a primeira fase de abstração do processo de compreensão do problema a ser resolvido do produto de software. É, essencialmente, uma atividade humana em que são identificadas as partes interessadas e o estabelecimento das relações entre fornecedor, cliente, usuários e equipe de desenvolvimento. O termo “elicitação” é utilizado para destacar que os bons requisitos não são só obtidos pelos clientes, mas sim pela habilidade de extrair informações do usuário. A seguir de acordo com SOMMERVILLE (2011), alguns problemas que ajudam a entender porque é difícil a elicitação de requisitos:

Problemas de escopo: Os limites do sistema são mal definidos e os clientes / usuários definem detalhes técnicos desnecessários que podem confundir, ao invés de esclarecer, os objetivos gerais do sistema.

Problemas de compreensão: Usuários não estão completamente seguros do que necessitam, têm uma compreensão deficiente das capacidades e limitações do seu ambiente de informação, e não tem um domínio completo do problema, eles têm dificuldades em comunicar suas necessidades para o engenheiro de software, e omite informações que considerem “Evidentes”, definem requisitos que entram em

conflito com as necessidades dos outros usuários, ou definem requisitos que são ambíguos ou inverificáveis.

Problemas de volatilidade: Mudança de requisitos ao longo do tempo.

Para ajudar a superar estes problemas, os engenheiros software devem executar de forma organizada a captura de requisitos.

A elicitação de requisitos é o princípio básico sobre o qual se constrói um projeto de software e tem um impacto muito alto na concepção e subsequentes fases do ciclo de vida do produto. Se feito corretamente, ele pode ajudar a reduzir as alterações e correções nos requisitos (FREITAS, 2015). Além disso, a qualidade da elicitação determina a precisão do feedback dos clientes sobre a integridade e validade dos requisitos . Uma vez que esta fase é crítica e de alto impacto no projeto, é muito importante que o trabalho de elicitar seja feito o mais próximo possível da perfeição.

As práticas elicitação de requisitos incluem entrevistas, questionários, observação do trabalho do usuário, workshops, brainstorming, casos de uso, prototipagem; embora existam muitas outras características diferentes em projetos reais (OLIVEIRA, 2014). Com base na experiência de alguns pesquisadores em projetos reais, que estão resumidos a seguir, as abordagens usadas para desencadear requisitos em várias circunstâncias, reduzindo assim, a dificuldades dessa ciência e aumentando a eficiência da elicitação.

DESENVOLVIMENTO

Alguns tipos de projetos são relacionados e divididos a seguir para um maior esclarecimento da aplicabilidade do conhecimento.

1 Projeto para iniciar um novo sistema

Um projeto é identificado quando é manifestada na organização uma necessidade de construir um sistema. Isso poderia ocorrer dentro de uma unidade da empresa ou na área de TI.

Os requisitos representam o que o sistema de informações fará ou que funcionalidades conterão. Eles precisam ser explicados em um nível alto para que o

comitê de aprovação ou um responsável por projetos solicitados na empresa e a equipe de projeto compreendam o que a empresa espera do produto final (PRESSMAN, 2011). Os requisitos da empresa são os recursos e capacidades que o sistema de informações terá de incluir, como por exemplo, habilidade de agendar reuniões, ou habilidade de inserir informações de contatos e eventos realizados pela empresa.

Normalmente os requisitos podem ser funcionais ou não-funcionais. Um requisito funcional se relaciona diretamente a um processo que o sistema tem de executar ou às informações que ele precisa conter. Os requisitos funcionais fluem diretamente para as próximas etapas do processo de análise (casos de uso, modelos de processos e modelo de dados) porque definem as funções que o sistema precisa possuir.

Os requisitos não funcionais se referem às propriedades comportamentais que o sistema deve possuir como desempenho e usabilidade. Os requisitos não-funcionais podem influenciar o restante do processo de análise, mas frequentemente o fazem apenas de modo indireto, os requisitos não-funcionais são usados principalmente na fase de projeto, quando são tomadas as decisões sobre a interface com o usuário, o hardware e o software e a arquitetura subjacente do sistema.

1.1 Ambas as partes conhecem os requisitos.

Para estes projetos, geralmente usa-se um método de pesquisa que seria uma série de perguntas a fim de alcançar melhor compreensão, uma vez que tanto os entrevistadores como para os clientes do projeto são claras as necessidades de comunicação (DE OLIVEIRA, 2013). A "abordagem de pesquisa" refere-se a uma forma elicitador os requisitos, em que o entrevistador envia um questionário para os usuários de forma que alcance uma compreensão completa do projeto com base em demandas individuais e coletivas, necessárias para definir os problemas com mais precisão (DIAS, 2015).

O questionário é amplamente usado como ferramenta para elicitador de requisitos já que é uma ciência para análise estatística (DE SOUZA, 2015). Os questionários são utilizados quando:

- Há um grande número de pessoas.
- Precisa de respostas para problemas específicos bem definidos.
- Se quer um resultado específico.

Durante a preparação de questionários deve se ter em conta:

Manter o questionário menor quanto possível. Em vez de usar um questionário grande, melhor aplicar vários pequenos. Porque no caso de questionários grandes, geralmente, o usuário tende a se cansar depois de responder as primeiras 15 a 20 perguntas e não ser muito objetivo para responder o resto. Como regra comum um questionário não deve conter mais do que 10 a 15 perguntas. Estimar o tempo necessário para responder perguntas em um ambiente favorável que facilite objetividade do questionário (DIAS, 2015). Certifique-se de que as questões estão em um contexto livre de ambiguidades. Para ter certeza, aplicar o protótipo entre os entrevistados pode melhorar a compreensão das perguntas que foram estruturadas. Antes de fazer perguntas, temos que nos certificar de que as respostas são necessárias. Em um contexto específico, as respostas para muitas perguntas tornam-se informações sem sentido (BRONZE, 2014). Listar todas as questões possíveis separadamente. Uma vez que todos os requisitos e as perguntas estão prontos, crie um plano cartesiano, no qual os requisitos no eixo X e as perguntas sobre o eixo Y; Para cada pergunta, identificar quais requisitos estão sendo cumpridos. No final deste exercício são descartadas as perguntas que não estão associadas a um dos requisitos. Este método tem a vantagem de ser simples e claro, o que pode encurtar o tempo e reduzir os custos melhorar a eficiência de captura dos requisitos (LIMA, 2015).

1.2 Os clientes sabem os requisitos, mas os desenvolvedores não sabem.

Para estes projetos é necessário realizar algumas reuniões com os clientes para discutir as suas necessidades, com a finalidade de desenvolver um sistema que os satisfaçam. A prática às vezes demonstra que isso não é viável, porque os desenvolvedores não conhecem as necessidades, ou seja, não estão familiarizados o ambiente do projeto (BRONZE, 2014). Como resultado, não sabem guiar os clientes para discutir estes problemas e eles são limitados apenas a mencionar as

questões gerais, sem entrar em detalhes, porque os entrevistadores acreditam que o problema existente é comum e todos entrevistados deveriam conhecê-los. E às vezes acontece o pior, onde o entrevistador pode fazer perguntas aos usuários que eles mesmos não sabem o que significa, por não conhecer o assunto (MONTEIRO, 2014). Após a entrevista, os clientes têm a sensação de que explicaram claramente os problemas, enquanto os entrevistadores não compreenderam bem o que deveria ser requisito ou não.

Cada sistema de informação é, na verdade uma caixa preta. Onde se insere dados numa extremidade e na outra a informação surge. Se possível obter as duas coisas, significa que você tem uma ideia aproximada do sistema (LIMA, 2015). Para estes sistemas, recomenda-se:

Usar métodos de coleta de dados de entrada e de saída. Neste processo, são recolhidas todas as atividades feitas com os usuários, todo o material impresso como formulários e relatórios necessários para ser feita análise e compilação. A fim de conhecer o contexto do ambiente de trabalho. Esta atividade também pode funcionar em empresas sem sistemas de informatização. Recebendo esse material impresso, ajudará na maioria das necessidades de análise e classificação da informação (DUARTE, 2015). Portanto, também é um processo no qual os entrevistadores ficam familiarizados com o ambiente de trabalho e adquire bases para conversações e comunicações futuras com os usuários.

Reuniões de grupo. Após a realização do passo realizado acima, você tem uma compreensão muito profunda do sistema. As reuniões de grupo podem ser utilizadas normalmente dos seguintes casos:

- Quando o conhecimento é distribuído igualmente entre pequeno grupo de pessoas.
- Quando a uma incapacidade para atender todas as dúvidas das partes interessadas individualmente.
- Quando já foi realizada uma série de entrevistas e a equipe precisa ter todos os usuários para atestar o requisito geral.

Na reunião do grupo, cada participante pode expressar seus pensamentos. As respostas de um grupo são melhores do que as respostas de um indivíduo, para

estes casos. Reuniões de grupo também ajudam a reduzir o número de conflitos nos requisitos e mantém um equilíbrio no desenvolvimento (DE OLIVEIRA, 2013).

Geralmente essas reuniões são realizadas no espaço do cliente. Se as reuniões de grupo são mal geridas, tendem a causar os seguintes problemas:

Pequeno número de participantes assume o controle discussão, ou alguns dos participantes não estarão animados para falar. Para evitar estas situações, um analista é necessário para intermediar a discussão. Ele deve encorajar os participantes a entrar na discussão: em primeiro lugar, pedir-lhes para propor algumas questões que estão deixando mais dúvidas e pouco a pouco ir esclarecendo conjuntamente com usuário essa dúvida e, em seguida, converter lentamente em questões de fácil esclarecimento (DE OLIVEIRA, 2013). Também é importante retornar temas já discutidos para confirmar a sua compreensão, definir agenda de reuniões e utilização de pessoas que tomem nota do que já foi realizado.

1.3 Os desenvolvedores sabem os requisitos, mas os clientes não sabem.

A primeira coisa que se deve fazer é deixar que o cliente reconheça o que ele realmente quer. Os métodos de "protótipo" e "reuniões de discussão". Poderão ser utilizados nessas ocasiões.

Os protótipos de interfaces são baseados na compreensão que os desenvolvedores têm nos requisitos do cliente; então, é modelada uma interface da aplicação e a mesma é utilizada para a comunicação com os clientes (SANTOS, 2014). Com o método de protótipo, ambas as partes poderiam chegar alcançar gradualmente a confirmação dos requisitos do projeto. A seguir estão algumas etapas desse método.

Passo 1: Com o entendimento dos requisitos e ferramentas do projeto, os desenvolvedores trazem o protótipo da aplicação.

Passo 2: Os desenvolvedores apresentam os protótipos aos clientes e discutem com os mesmos, isso ajuda a descobrir novas necessidades ou alcançar um acordo.

Passo 3: De forma incremental, os desenvolvedores analisam esses requisitos e adicionam novos protótipos ou melhoram os já existentes.

Passo 4: Depois de várias reuniões tendo como base esses protótipos, os desenvolvedores finalmente conseguem alcançar a especificação dos requisitos dos usuários.

Com as reuniões de discussão. Depois de alcançar um acordo base sobre o problema "o que fazer" no processo acima, as partes iniciam reuniões para entender o funcionamento das regras de negócio, que consiste em "como implementar o modelo de prototipagem" (BATISTA, 2015). Este método discussão também pode ajudar a determinar os requisitos do projeto.

1.4 Ambas as partes não sabem os requisitos.

Esta condição apresenta grandes dificuldades em elicitare requisitos e é de maior risco no desenvolvimento do sistema (SOUZA, 2015). Nesta situação pode se usar os métodos da "Investigação" e "Cenários".

O método de investigação recolhe uma ampla gama requisitos através de:

- Investigação do trabalho de outras empresas do mesmo ramo.
- Observação de sistemas semelhantes.
- Revisão documental do sistema anterior, se houver.

Durante a investigação é necessário fazer algumas perguntas como:

Por que os usuários necessitam de um sistema? Que tipo de apoio o novo sistema dará ao cliente? Que efeito terá o novo sistema sobre a eficiência no trabalho? O que o cliente quer este sistema faça ou o que funções devem ser implementadas neste sistema? Que condições são necessárias para o funcionamento deste sistema?

Os cenários também podem ajudar a reunir informações para esses projetos. Com a elicitacão de requisitos baseada em cenários é possível mapear os requisitos que abordam o problema do sistema utilizando uma representacão com um conjunto de atores e casos de uso. Os atores representam entidades externas que interagem com o sistema. O primeiro passo da elicitacão é a identificacão dos atores (SANTOS, 2014). Uma vez identificado os atores por meio da ilustracão de cenários, se determina a funcionalidade que cada requisito vai abordar.

Um cenário é uma descrição da utilização do sistema que seria uma série de interações entre o sistema e usuário (LIMA, 2015). O trabalho com os clientes é necessário para atingir um conjunto completo de cenários, que é descrito em linguagem natural e utilização de forma prática o ambiente do sistema. O conjunto completo de cenários descrevem tudo o que se pretende fazer no sistema. Então os desenvolvedores formalizam os cenários de casos de uso, já que o cenário é uma das instâncias de um caso de uso, e um caso de uso especifica todos os cenários possíveis para uma determinada função, então se idealiza que um caso de uso só vai ser inicializado se ele tiver um determinado ator para realização do sistema (DA SILVA BISCAIA, 2014).

2. Projeto para atualizar sistemas existentes

Com o desenvolvimento da informatização, mais empresas têm o seu próprio sistema de gestão informações. Seus usuários expõem novos requisitos ao projeto devido a novas funções acrescentadas, as mudanças na lógica de negócios ou deficiências no sistema antigo, tornando cada vez mais difícil de satisfazer as necessidades das empresas (BRONZE, 2014). Nestes sistemas, os passos para elicitar os requisitos são:

Habituação com o sistema antigo: A razão que o sistema ainda existe até agora é porque pode satisfazer a maioria das necessidades do negócio, especialmente no *core* do negócio ou porque geralmente fazem atualizações cumulativas. Portanto, a maioria dos requisitos do software pode ser conseguida com a familiarização desse sistema. Também é aconselhável o conhecimento de mercados relacionados ao projeto, de forma a reduzir barreiras posteriores em comunicação com o do utilizador. Se houver documentos relevantes do sistema antigo, também poderão ser aproveitados (DE SOUZA, 2015).

Entrevistas: Logo após o passo anterior se obtém um bom conhecimento da maioria dos requisitos do sistema já que o convívio entra no contexto do projeto, a seguir, progredimos para a entrevista. A entrevista é uma das técnicas tradicionais mais simples de utilizar e que produz bons resultados na fase inicial de obtenção de dados. Convém que o entrevistador dê margem ao entrevistado para expor as suas idéias. É necessário ter um plano de entrevista para que não haja dispersão do

assunto principal e a entrevista fique longa, deixando o entrevistado cansado e não produzindo bons resultados (LIMA, 2015).

As seguintes diretrizes podem ser de grande auxílio na direção de entrevistas bem sucedidas com o usuário: desenvolver um plano geral de entrevistas, certificar-se da autorização para falar com os usuários, planejar a entrevista para fazer uso eficiente do tempo, utilizar ferramentas automatizadas que sejam adequadas, tentar descobrir que informação o usuário está mais interessado e usar um estilo adequado ao entrevistar.

Para planejar a entrevista é necessário que antes dela sejam coletados e estudados todos os dados pertinentes à discussão, como formulários, relatórios, documentos e outros. Dessa forma, o analista estará bem contextualizado e terá mais produtividade nos assuntos a serem discutidos na entrevista.

É importante determinar um escopo relativamente limitado, focalizando uma pequena parte do sistema para que a reunião não se estenda por mais de uma hora. O usuário tem dificuldade de concentração em reuniões muito longas, por isso é importante focalizar a reunião no escopo definido.

Após a entrevista é necessário validar se o que foi documentado pelo analista está de acordo com a necessidade do usuário, que o usuário não mudou de opinião e que o usuário entende a notação ou representação gráfica de suas informações.

A atitude do analista em relação à entrevista é determinar seu fracasso ou sucesso (DE OLIVEIRA, 2013). Uma entrevista não é uma competição, deve-se evitar o uso excessivo de termos técnicos e não conduzir a entrevista em uma tentativa de persuasão. O modo como o analista fala não deve ser muito alto, nem muito baixo, tampouco indiretamente, ou seja, utilizar os termos: ele disse isso ou aquilo na reunião para o outro entrevistado. O modo melhor para agir seria, por exemplo, dizer: O João vê a solução para o projeto dessa forma. E o senhor André, qual é a sua opinião? Em uma entrevista o analista nunca deve criticar a credibilidade do entrevistado. O analista deve ter em mente que o entrevistado é o perito no assunto e fornecerá as informações necessárias ao sistema.

Para elaborar perguntas detalhadas é necessário solicitar que o usuário:

- Explique o relacionamento entre o que está em discussão e as demais partes do sistema;
- Descreva o ponto de vista de outros usuários em relação ao item que esteja sendo discutido;
- Descreva informalmente a narrativa do item em que o analista deseja obter informações;
- Perguntar ao usuário se o item em discussão depende para a sua existência de alguma outra coisa, para assim poder juntar os requisitos comuns do sistema, formando assim um escopo conciso.

Pode-se utilizar a confirmação, para tanto o analista deve dizer ao usuário o que acha que ouviu ele dizer. Neste caso, o analista deve utilizar as suas próprias palavras em lugar das do entrevistado e solicitar ao entrevistado confirmação do que foi dito (MONTEIRO, 2014). Através de entrevistas com diferentes usuários se conhecem os requisitos que alteram ou aumentam a demanda, em termos gerais, quando os usuários estão muito dispostos a falar com os entrevistadores e os mesmos explicam de forma clara e detalhada sobre o que é necessário ao sistema, essas partes que eles descrevem são exatamente o que o sistema antigo não podia satisfazer no seu trabalho, e que devidamente elaborado na elicitação dificilmente traria dúvidas para o desenvolvimento de um novo sistema (DA SILVA BISCAIA, 2014).

3 Recomendações

Estes métodos não são mutuamente exclusivos e podem ser aplicados separadamente ou de forma integrada. De acordo com as características do projeto. Existem outros métodos, que também podem ser adaptados quando os requisitos dos sistemas são capturados e alguns detalhes permanecem sem solução. Seja qual for o método aplicado, existem algumas regras que devem ser consideradas. Sommerville (2011) sugere um conjunto de orientações para a elicitação de requisitos:

- Avaliar a viabilidade técnica e comercial para o sistema proposto.
- Identificar pessoas que ajudam a você especificar os requisitos e entender seus ideais organizacionais.

- Definir o ambiente técnico, por exemplo, a arquitetura computacional, o sistema de operacional, em que irá operar o sistema.

- Identificar "restrições de domínio", isto é, as características do ambiente de negócios específico para o domínio de aplicação que limitam a funcionalidade ou o desempenho do sistema do produto a ser desenvolvido.

- Definir um ou mais métodos de elicitación de requisitos, por exemplo, entrevistas, reuniões de grupo, questionários, etc.

- Solicitar a participação de muitas pessoas para que se definam requisitos a partir de diferentes pontos de vista, assegurando de identificar a base ideal de cada requisito registrado.

- Identificar os requisitos ambíguos como candidatos a protótipo.
- Criar cenários para ajudar a clientes / usuários a identificar melhor os requisitos.

CONCLUSÃO

A chave para o sucesso em um projeto de software é a elicitación de requisitos. Dada à variedade de métodos de elicitación e diferentes níveis de maturidade dos requisitos de cada projeto, deve ser adaptada uma estratégia adequada para realizá-lo, e é primordial também que o analista possua perfil adequado, já que o mesmo precisa mais do que apenas a capacidade de desenhar fluxograma e outros diagramas técnicos, ele tem a função de projetar e analisar sistemas de ótimo desempenho, mas para que isso aconteça ele tem que compreender os conceitos abstratos, comunicar-se bem, etc. Já que o conjunto da técnica da elicitación e os conceitos formados pelo analista torne-se o mais próximo possível da realidade. Este artigo descreve alguns métodos para aplicação, de acordo com o tipo de projeto realizado. Selecionar o método correto pode encurtar o tempo, reduzir custos e melhorar a eficiência da elicitación de requisitos.

REFERÊNCIAS

DIAS, Mailton Galdino. Aplicação da filosofia lean em uma empresa de desenvolvimento de software. 2015.

OLIVEIRA, Fabrício de Almeida. Estudo e avaliação da área de processo gestão de requisitos de acordo com a norma cmmi nível 2 na empresa swquality. 2015.

SOUZA, Cibely Cobo da Silva. Medidas de requisitos de software: um estudo exploratório de aplicabilidade. 2015.

FREITAS, Sheila Cristiana de. A utilização de mapas conceituais como apoio ao ensino de análise de requisitos de software. 2015.

DUARTE, André Luís Faria; VIEIRA, Paulo Roberto da Costa; SILVA, Antonio Carlos Magalhães da. AVALIAÇÃO DE ESCALA PARA MENSURAÇÃO DA EFICÁCIA DE SISTEMA DE INFORMAÇÃO: UM ESTUDO COM ANÁLISE FATORIAL CONFIRMATÓRIA. **GEINTEC-Gestão, Inovação e Tecnologias**, v. 5, n. 2, p. 1986-1999, 2015.

DE SOUZA, Caroline Battistello Cavalheiro; SAVI, Rafael. Design centrado no usuário e o projeto de soluções educacionais. **Revista E-Tech: Tecnologias para Competitividade Industrial-ISSN-1983-1838**, v. 1, n. 1, 2015.

LIMA, Felipe Augusto Pastore de. Desenvolvimento enxuto de software. 2015.

BATISTA, JONATHAS ANTUNES. DESENVOLVIMENTO DE PROTÓTIPO DE SOFTWARE PARA GESTÃO DO CONHECIMENTO: aplicado à uma fábrica de software. **Projetos e Dissertações em Sistemas de Informação e Gestão do Conhecimento**, v. 3, n. 2, 2015.

DONATO, Fabiana Juvêncio Aguiar. a aplicação de uma abordagem sistemática, disciplinada e quantificável, para o desenvolvimento, operação e manutenção do software”. 2015.

BRONZE, Reinaldo et al. Levantamento de requisitos de software, uma análise comparativa. **Sistemas de Informação & Gestão de Tecnologia.**, n. 6, 2014.

MONTEIRO, Alanna Camylla Coêlho. Requisitos não-funcionais para o desenvolvimento de software educacional. 2014.

OLIVEIRA, Romeu Ferreira de. Um método semi-automatizado para eliciação de requisitos de acessibilidade Web. 2014.

DA SILVA BISCAIA, Rosiane et al. ESPECIFICAÇÃO DO ESCOPO DE SISTEMA DE SOFTWARE A PARTIR DA UTILIZAÇÃO DA ENGENHARIA DE REQUISITOS. 2014.

SANTOS, Fabio S. VALIDAÇÃO DE REQUISITOS ATRAVÉS DA PROTOTIPAÇÃO DE SOFTWARE. **CONNECTION LINE**, n. 9, 2014.

DE OLIVEIRA, João Paulo N. et al. Desafios no Gerenciamento de Conflitos em Projetos de Desenvolvimento Distribuído de Software. 2013.

SOMMERVILLE, Ian. Engenharia de Software, n.9, 2011.

PRESSMAN, Roger S. Engenharia de Software - Uma Abordagem Profissional, n 7, 2011.