



UNIVERSIDADE ESTADUAL DO SUDOESTE DA BAHIA  
DEPARTAMENTO DE CIÊNCIAS EXATAS

ANDERSON MARQUES DA SILVA FIGUEIRA

**ANÁLISE DAS TÉCNICAS DE LEVANTAMENTO DE REQUISITOS  
PARA DESENVOLVIMENTO DE SOFTWARE NAS EMPRESAS DE  
VITÓRIA DA CONQUISTA – BA**

VITÓRIA DA CONQUISTA – BA  
ABRIL DE 2012

ANDERSON MARQUES DA SILVA FIGUEIRA

**ANÁLISE DAS TÉCNICAS DE LEVANTAMENTO DE REQUISITOS  
PARA DESENVOLVIMENTO DE SOFTWARE NAS EMPRESAS DE  
VITÓRIA DA CONQUISTA – BA**

Monografia final de conclusão de curso apresentada ao Colegiado de Ciência da Computação – UESB como parte dos requisitos para obtenção do título de Bacharel em Ciência da Computação.

**Área de concentração:** Engenharia de Software

**Orientador (a):** Prof.<sup>a</sup> Ms. Maria Silva Santos Barbosa

**Coorientador:** Prof. Esp. Fabrício de Sousa Pinto

VITÓRIA DA CONQUISTA – BA

ABRIL DE 2012

ANDERSON MARQUES DA SILVA FIGUEIRA

**ANÁLISE DAS TÉCNICAS DE LEVANTAMENTO DE REQUISITOS  
PARA DESENVOLVIMENTO DE SOFTWARE NAS EMPRESAS DE  
VITÓRIA DA CONQUISTA – BA**

Monografia aprovada em \_\_\_\_/\_\_\_\_/\_\_\_\_ para obtenção do título de Bacharel em  
Ciência da Computação.

Comissão Examinadora

---

Prof.<sup>a</sup> Ms. Maria Silva Santos Barbosa  
Universidade Estadual do Sudoeste da Bahia

---

Prof. Esp. Fabrício de Sousa Pinto  
Faculdade de Tecnologia e Ciências de Vitória da Conquista

---

Prof. Ms. Francisco dos Santos Carvalho  
Universidade Estadual do Sudoeste da Bahia

VITÓRIA DA CONQUISTA – BA  
ABRIL DE 2012

*Aos meus pais e minha irmã  
pela enorme força e paciência  
durante este caminho e sempre.*

## **AGRADECIMENTOS**

A três pessoas, em especial, as quais não encontro palavras suficientes para agradecer o imenso amor recebido durante esta caminhada e por toda vida, Manoel Figueira e Maria de Fatima (meus pais) e Andreia Figueira (minha irmã). A força, a dedicação e a paciência foram apenas alguns dos atos de amor que essas três pessoas maravilhosas me deram durante este percurso. A conclusão desse curso só seria possível graças a vocês. Talvez seja pouco, mas dedicarei a minha vida a retribuir todos os valores e afetos que esses seres especiais me ensinaram. Amo vocês incondicionalmente e me sinto muito feliz e orgulhoso por fazer parte da família mais linda desse mundo. Obrigado por existirem!

A duas pessoas queridas, que compartilharam comigo dessa jornada, Leonardo (Leleco), um irmão que encontrei na vida e que me aconselhou diversas vezes e a Carolina, pela força e o carinho durante este percurso.

Aos amigos Thadeu, Celso, Guto e Vinícius, os quais foram e são parceiros de república e que de alguma maneira contribuíram com conselhos e resenhas, tornando esse caminho menos difícil.

Aos meus orientadores Maria Silva e Fabricio Sousa por todo ensinamento durante esta caminhada.

À galera do CETEP onde aprendi muito e construí diversas amizades.

A todos aqueles que de alguma forma me motivaram para que esse sonho se tornasse realidade.

À Deus, pois sei que estava ao meu lado em todos os momentos.

## LISTAS DE FIGURAS

Figura 1 – Diagrama de casos de uso .....	28
Figura 2 – Descrição contínua.....	29
Figura 3 – Modelando um diagrama de classes no Astah.....	34
Figura 4 – Interface do Rational RequisitePro.....	36
Figura 5 – Interface do OSRMT .....	37
Figura 6 – Interface do CaliberRM .....	38

## LISTA DE GRÁFICOS

Gráfico 01 – Experiência com levantamento de requisitos.....	41
Gráfico 02 – Participação da etapa de levantamento de requisitos em projetos .....	41
Gráfico 03 – Frequência do uso da técnica Entrevista .....	42
Gráfico 04 – Frequência do uso da técnica Questionário.....	42
Gráfico 05 – Frequência do uso de Diagramas .....	43
Gráfico 06 – Frequência do uso de Casos de Uso.....	44
Gráfico 07 - Frequência do uso de Cenários.....	44
Gráfico 08 – Frequência do uso de <i>User Stories</i> .....	45
Gráfico 09 – Frequência do uso de Reuniões em grupo .....	45
Gráfico 10 – Frequência do uso de <i>Brainstorming</i> .....	46
Gráfico 11 – Frequência do uso de Análise de documentos .....	46
Gráfico 12 – Frequência do uso de Protótipos .....	47
Gráfico 13 – Frequência do uso de <i>Group storytelling</i> .....	48
Gráfico 14 – Frequência da reutilização de requisitos.....	48
Gráfico 15 – Frequência do uso de Pontos de Vista .....	49
Gráfico 16 – Nível de importância na escolha de uma técnica de levantamento de requisitos.....	50
Gráfico 17 – Nível de importância do fator, preferência do cliente, na escolha da técnica pelo desenvolvedor .....	51
Gráfico 18 – Dificuldade dos usuários de visualizar os requisitos do sistema nas suas atividades.....	51
Gráfico 19 – Uso dos processadores de texto no levantamento de requisitos	52
Gráfico 20 - Uso de planilhas eletrônicas no levantamento de requisitos .....	52
Gráfico 21 – Uso de <i>templates</i> de documentos no levantamento de requisitos	53
Gráfico 22 – Uso das ferramentas de modelagem no levantamento de requisitos	54
Gráfico 23 – Uso das ferramentas de gerenciamento de requisitos.....	54
Gráfico 24 – Uso de quadros, <i>flip charts</i> , gravações de áudio e vídeo no levantamento de requisitos .....	55
Gráfico 25 – Como e quantas empresas validam requisitos identificados com os <i>stakeholders</i> após a primeira rodada de levantamento de requisitos.....	56

Gráfico 26 – Dificuldade do usuário de realizar e entender a validação de requisitos.....	57
Gráfico 27 – Maior formalismo da documentação, maior dificuldade do cliente compreender .....	58
Gráfico 28 – Menor formalidade, maior dificuldade de entendimento do analista	58
Gráfico 29 – Principais dificuldades dos desenvolvedores no levantamento de requisitos.....	60



## LISTA DE ABREVIATURAS

<b>CASE</b>	<i>Computer Aided Software Engineering</i>
<b>CMM</b>	<i>Capability Maturity Model</i>
<b>GPL</b>	<i>General Public License</i>
<b>IBM</b>	<i>International Business Machines</i>
<b>IEEE</b>	<i>Institute of Electrical and Electronics Engineers</i>
<b>ISO</b>	<i>International Organization for Standardization</i>
<b>JAD</b>	<i>Joint Application Development</i>
<b>JUDE</b>	<i>Java and UML Developer Environment</i>
<b>OSRMT</b>	<i>Open Source Requirements Management Tool</i>
<b>PDF</b>	<i>Portable Document Format</i>
<b>RAD</b>	<i>Rapid Application Development</i>
<b>UML</b>	<i>Unified Modeling Language</i>

## RESUMO

Esse trabalho de conclusão de curso tem por objetivo apresentar uma análise da utilização de técnicas de levantamento de requisitos no desenvolvimento de software pelas empresas de Vitória da Conquista, Bahia. Alguns estudos mostram que aplicação de uma determinada técnica apropriada facilita a elicitación de requisitos, e principalmente o relacionamento com o cliente. Caso contrário, acarretará na insatisfação do mesmo. Para a verificação desse fato, a pesquisa utilizou de questionários para coleta de dados que foram distribuídos em 18 empresas da cidade mencionada. A tabulação e análise dos dados foram feitas utilizando *Microsoft Excel 2010*. Os resultados mostraram que muitas técnicas são aplicadas na aquisição de requisitos, entretanto, técnicas que contam com a participação de diversos *stakeholders* não são comumente utilizadas ou são desconhecidas por grande parte dos desenvolvedores. Outro dado relevante foi com relação à dificuldade do desenvolvedor entender o que cliente realmente necessita, a qual foi opinada por 94,4% dos pesquisados. Concluiu-se que apesar do emprego de algumas técnicas de levantamento de requisitos, há a necessidade de um melhor estudo por parte dos desenvolvedores sobre outras técnicas que possam adequar e facilitar a elicitación de requisitos e o relacionamento com o cliente.

**Palavras chaves:** elicitación, engenharia de requisitos, levantamento de requisitos, técnicas de levantamentos de requisitos, software, *stakeholders*, cliente.

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	12
<b>2</b>	<b>ENGENHARIA DE REQUISITOS</b>	15
2.1.	REQUISITOS	16
2.2.	REQUISITOS DE SISTEMA E DE USUÁRIO	17
2.2.1.	PROCESSO DE ENGENHARIA DE REQUISITOS	18
2.2.2.	ELICITAÇÃO DE REQUISITOS	19
2.2.3.	ANÁLISE E NEGOCIAÇÃO	20
2.2.4.	DOCUMENTAÇÃO DE REQUISITOS	20
2.2.5.	VALIDAÇÃO DOS REQUISITOS	21
2.2.6.	GERENCIAMENTO DE REQUISITOS	22
<b>3</b>	<b>TÉCNICAS DE LEVANTAMENTO DE REQUISITOS</b>	23
3.1.	PONTOS DE VISTA	23
3.2.	ENTREVISTA	24
3.3.	QUESTIONÁRIO	24
3.4.	OBSERVAÇÃO	25
3.5.	JOIN APPLICATION DEVELOPMENT	26
3.6.	PROTOTIPAÇÃO	26
3.7.	CASOS DE USO	27
3.8.	CENÁRIOS	29
3.9.	ANÁLISE DE DOCUMENTOS	30
3.10.	<i>BRAINSTORMING</i>	30
3.11.	<i>STORYTELLING/GROUP STORYTELLING</i>	31
3.12.	<i>USER STORIES</i>	32
3.13.	REUSO DE REQUISITOS	32
<b>4</b>	<b>FERRAMENTAS QUE AUXILIAM NA AQUISIÇÃO E GERÊNCIA DE REQUISITOS</b>	33
4.1.	<i>TEMPLATES</i> DE DOCUMENTOS	33
4.2.	ASTAH	34
4.3.	<i>RATIONAL REQUISITEPRO</i>	34
4.4.	OSRMT	36
4.5.	CALIBERRM	37
<b>5</b>	<b>ANÁLISE DAS TÉCNICAS DE LEVANTAMENTO DE REQUISITOS PARA DESENVOLVIMENTO DE SOFTWARE NAS EMPRESAS DE VITÓRIA DA CONQUISTA – BA</b>	39
5.1.	METODOLOGIA DA PESQUISA	39
5.2.	EXPERIÊNCIA COM LEVANTAMENTO DE REQUISITOS E FREQUÊNCIA DO USO DE TÉCNICAS PARA ESSA ATIVIDADE	40

5.3. FERRAMENTAS UTILIZADAS NO PROCESSO DE LEVANTAMENTO DE REQUISITOS	52
5.4. VALIDAÇÃO DOS REQUISITOS APÓS A PRIMEIRA RODADA DE LEVANTAMENTO DE REQUISITOS	55
5.5. FORMALIDADE NO LEVANTAMENTO DE REQUISITOS	57
5.6. DIFICULDADES NO LEVANTAMENTO DE REQUISITOS	58
<b>6. CONCLUSÕES</b>	61
6.1. TRABALHOS FUTUROS	63
<b>REFERÊNCIAS</b>	64
<b>APÊNDICE – QUESTIONÁRIO</b>	68

## 1 INTRODUÇÃO

O desenvolvimento de software está em iminente expansão no Brasil e no mundo. É possível notar que com crescimento deste produto, houve o aumento de novas técnicas de levantamento de requisitos, fato preponderante para a produção de software. O levantamento de requisitos é uma das tarefas essenciais da engenharia de requisitos, parte que integra todo o processo de criação de um software.

A importância da engenharia de requisitos é perceptível, pois ela é a fase inicial do processo da engenharia de software, que estuda como coletar, entender, armazenar e gerenciar os requisitos. Além disso, se preocupa em compreender a veracidade dos requisitos de sistema e a documentação dos mesmos (THAYER, 1997). Sommerville (2007) complementa que durante o processo de engenharia de requisitos é que será definido o que sistema deve fazer, quais são as suas propriedades emergentes desejáveis e essenciais e as restrições quanto a sua operação e o seu processo de desenvolvimento.

É na fase de elicitação e análise de requisitos, a primeira atividade na engenharia de requisitos, que damos ênfase na busca e aquisição de requisitos para o desenvolvimento do software, utilizando técnicas e métodos adequados para obtê-las. Entretanto essa atividade não ocorre apenas uma única vez, seu processo é iterativo, ou seja, nas demais atividades do processo de engenharia de requisitos ela poderá ser empregada (BELGAMO, 1999).

Os clientes, usuários, desenvolvedores, gerentes de projeto e todos os demais envolvidos com o projeto do software são precursores dessa atividade. É nesta fase que todos eles estabelecem as informações necessárias que darão suporte ao engenheiro de software a produzir os requisitos adequados para o sistema em desenvolvimento

Inúmeras ferramentas de engenharia de requisitos são utilizadas para auxiliar e facilitar os desenvolvedores na aquisição de requisitos como, por exemplo, o Astah, ferramenta que propicia a construção de uma variedade de modelos gráficos que representam os aspectos informacional, funcional e comportamental do sistema (PRESSMAN, 2006). Há, também, ferramentas de gestão de requisitos como o RequisitePro e o CaliberRM que facilitam na

organização dos requisitos obtidos, na exclusão de requisitos ambíguos e/ou desnecessários; e na reutilização destes por outros projetos.

Com todas as facilidades e benefícios que a engenharia de requisitos traz consigo, Sommerville (2007) destaca que muitas empresas ainda não utilizam técnicas e métodos adequados para o desenvolvimento de software, o que ocasiona em atrasos e pouca confiabilidade no produto.

As dificuldades são resultantes de diversos fatores como: os clientes e/ou usuários não sabem o que querem; os engenheiros de requisitos, sem experiência, não entendem o que o cliente necessita; diversos usuários têm diferentes requisitos e o engenheiro não leva em consideração a categorização deles e fatores políticos e socioeconômicos podem influenciar nos requisitos do sistema por parte dos interessados (SOMMERVILLE, 2007).

Essa complexidade no levantamento de requisitos pode estar relacionada, como mostra os trabalhos “*Update a Systematic Review about Selection of Software Requirements Elicitation Techniques*” (DIESTE; LOPEZ; RAMOS, 2008) e “Um processo de elicitação de requisitos com foco na seleção da técnica de elicitação” (ASSIS; BARBOSA, 2009), às deficiências na escolha de técnicas e métodos adequados para o processo de levantamento de requisitos por parte dos desenvolvedores.

Com tais dificuldades mencionadas em relação à atividade de levantamento de requisitos, este trabalho de conclusão de curso tem por objetivo fazer uma análise de como é feito o levantamento de requisitos nas empresas de desenvolvimento de software. O escopo escolhido para ser feito a análise foi às empresas de desenvolvimento de software da cidade de Vitória da Conquista, localizada no sudoeste da Bahia, território brasileiro. A região, citada, foi escolhida por existirem poucos estudos relacionados a esse assunto, fator determinante para sua preferência. Outro item que influenciou na escolha dessa região é pela facilidade de locomoção até as empresas, pois existem inúmeras empresas que produzem software no Brasil e no mundo, o que poderia tornar a pesquisa inviável pela dificuldade de locomoção até elas.

Inicialmente foi feita uma pesquisa das empresas desenvolvedoras de software de Vitória da Conquista. A essas empresas foi entregue um questionário que visava indagar questões relacionadas ao uso, importância, frequência e tipos

de técnicas de levantamento de requisitos usadas por essas empresas. Além disso, foi averiguado como se dá a participação dos clientes nos projetos de software e as dificuldades relacionadas a esse fato. Os resultados relevantes foram avaliados e mostrados através de gráficos com sua devida análise.

O trabalho de conclusão de curso será organizado da seguinte maneira: o Capítulo 1 corresponde à introdução, o Capítulo 2 faz uma abordagem geral sobre a engenharia de requisitos, o Capítulo 3 comenta sobre as diversas técnicas de levantamento de requisitos, o Capítulo 4 aborda algumas ferramentas utilizadas para obtenção e gerência de requisitos, o Capítulo 5 traz o resultado das análises feitas nas empresas de Vitória da Conquista e, por último, a conclusão do trabalho mencionado.

## 2 ENGENHARIA DE REQUISITOS

A criação e sucesso de um software dependem e muito de uma engenharia de requisitos bem elaborada e definida. Pressman (2006) destaca que ela ajuda os engenheiros de software a compreender melhor o problema que eles vão trabalhar para resolver. Além disso, inclui um conjunto de tarefas que poderão ajudar na compreensão do impacto do produto sobre o negócio, do que realmente o cliente deseja e dos usuários finais que vão interagir com o software.

Uma das definições mais aceitas para a engenharia de requisitos é a de Zave (1997), ele diz que a engenharia de requisitos está relacionada com a identificação das metas a serem atingidas pelo sistema, o qual será desenvolvido, assim como a operacionalização dessas metas em serviços e restrições.

Outros autores definem a engenharia de requisitos de forma mais técnica. Thayer e Dorfman (2000) estabelecem como a ciência e a disciplina preocupada com a análise e documentação dos requisitos, a qual faz a análise e especificação de requisitos. Possui também mecanismos adequados para facilitar as atividades de análise, documentação e verificação.

O IEEE (1984) exprime que a engenharia de requisitos é um processo de obtenção, refinamento e verificação das necessidades do cliente para o desenvolvimento de um software, que tem por finalidade especificar de forma completa e correta os requisitos.

Goguem (1997) e Sommerville (2007) acrescentam ao conceito de engenharia de requisitos alguns aspectos que influenciam o seu processo, como fatores políticos, sociais, econômicos, e psicológicos. Esses fatores são evidenciados quando o desenvolvedor interage com o cliente e/ou usuário na busca dos requisitos do sistema, tendo que o mesmo levar em consideração esses diversos fatores.

Entre tantas definições, aspectos, visões e conclusões do que é a engenharia de requisitos, em suma, os autores da área concordam que a obtenção dos requisitos, atividade preponderante da engenharia de requisitos, não sendo feita de forma adequada, poderá ocasionar em sistemas falhos, complexos, e que não atendem ao objetivo proposto pelo cliente, podendo acarretar na inutilização do sistema, e na insatisfação do mesmo.



## 2.1. REQUISITOS

Um dos principais objetivos da engenharia de requisitos é encontrar os requisitos adequados para o sistema em desenvolvimento. Entretanto uma pergunta poderia ser questionada: o que é realmente requisito? Alguns autores definem requisitos da seguinte forma:

- Thayer e Dorfmann (2000) definem que requisito é uma característica do software necessária para que o usuário e/ou cliente possa encontrar a solução de um problema de forma a atingir um objetivo;
- Sommerville (2007) expressa que requisitos de um sistema são descrições dos serviços fornecidas pelo sistema e as suas restrições operacionais. Os requisitos demonstram as necessidades de um cliente de um sistema que ajuda a resolver um determinado problema, como por exemplo, controlar um dispositivo, enviar um pedido ou encontrar informações;
- Requisitos são como uma declaração completa do que o software irá fazer sem referir-se a como fazê-lo (LOPES apud SIDDIQI, 1996).
- Kruchten explica que um requisito é como uma condição ou capacidade que um software deve realizar (LOPES apud KRUCHTEN, 2000).
- Goguem (1997) acrescenta que requisitos são propriedades que um software deve ter para funcionar com êxito no ambiente em que for utilizado.

Diante das definições abordadas pelos autores acima, fica claro que os requisitos expressam as características necessárias que o software deve ter para a realização de tarefas, as quais satisfaçam os objetivos dos *stakeholders* (termo utilizado na engenharia de software para designar as pessoas que interagem ou são afetados pelo sistema em desenvolvimento como clientes, usuários, gerentes de projeto e dentre outros).

## 2.2. REQUISITOS DE SISTEMA E DE USUÁRIO

O conceito de requisitos é amplo e abstrato. Isso é demonstrado pelas inúmeras particularidades que o sistema possui, como funções, capacidades e propriedades. O cliente e/ou usuário, normalmente, não possui conhecimento técnico para entender essas tais particularidades, ele apenas objetiva que o software resolva o seu determinado problema. Entretanto, Sommerville (2007) destaca que no processo de aquisição de requisitos por parte dos engenheiros de requisitos e projetistas, ocorre muitas vezes um grave problema: a falta de clareza na descrição dos diferentes níveis de especificação de requisitos entre os participantes do processo de desenvolvimento do software.

A importância dos diferentes níveis de especificação de sistema é fundamental, pois facilitam a comunicação das informações relacionadas ao sistema em questão a diferentes tipos de leitores. Por isso, Sommerville (2007) divide os requisitos entre dois termos distintos: requisitos de sistema e requisitos de usuário.

Requisitos de sistema seriam as funções, os serviços e as restrições operacionais do sistema definido de forma detalhada. Deve existir um documento de requisitos de sistema de forma precisa, descrevendo exatamente o que será implementado.

Os requisitos de sistema de software são classificados de acordo com Sommerville (2007) em requisitos funcionais, não funcionais ou de domínio:

Requisitos funcionais são as atividades que o sistema deve fornecer, como o sistema deve reagir a determinadas entradas específicas e também verifica o comportamento do sistema em determinadas situações.

Requisitos não funcionais especificam as restrições sobre as atividades e funções fornecidas pelo sistema como de tempo; do processo de desenvolvimento; padrões e de qualidades gerais de um software, como custo, manutenibilidade, confiabilidade, usabilidade, desempenho, portabilidade e entre outras.

Requisitos de domínio são originados do domínio da aplicação do sistema e refletem as características desse domínio, podendo ser requisitos funcionais ou não funcionais.

Os requisitos de usuário são declarações, em uma linguagem natural com diagramas, de quais serviços o sistema deve oferecer e as restrições que ele deve realizar. Deve ser compreensível pelos usuários do sistema que não possuem conhecimento técnico detalhado, expressando, apenas, o comportamento externo do sistema. Recomenda-se evitar a utilização de características técnicas de projeto nos requisitos de usuário.

### 2.2.1. PROCESSO DE ENGENHARIA DE REQUISITOS

Para Kotonya (1998) processo é um conjunto organizado de atividades que transforma entradas em saídas, levando em consideração como uma atividade inicial do processo de desenvolvimento de software. Já a *International Organization for Standardization* (ISO, 1988) explica que processo é como “um grupo de atividades que se caracterizam por uma série de entradas específicas que agregam valor e fornecem uma série de saídas específicas para clientes externos e internos”.

As entradas incluiriam: informações sobre as funcionalidades do sistema; descrição das necessidades dos *stakeholders*; padrões usados na organização referente à prática de desenvolvimento do sistema, gerência de qualidade e entre outros; regulamentações e informações do domínio da aplicação. Estas informações são usadas para a realização do processo que resultará nas seguintes saídas: uma descrição dos requisitos do sistema que são compreensíveis pelos *stakeholders* e que houve um acordo entre eles; uma especificação dos requisitos; e modelos do sistema (SOMMERVILLE, 2007).

Há inúmeras propostas para modelos de processo de engenharia de requisitos. Pressman (2006) define as atividades da engenharia de requisitos em sete funções distintas: concepção, levantamento, elaboração, negociação, especificação, validação e gestão. Kotonya e Somerville (1997) descreve o processo em cinco fases: elicitação, análise, documentação e validação dos requisitos. Para Sommerville (2007) o processo de engenharia de requisitos constitui de quatro subprocessos: estudo de viabilidade, elicitação e análise, especificação e validação dos requisitos. Entretanto não existe um processo

considerado ideal. No geral as fases constituem a seguinte ordem: elicitação, análise e negociação, documentação, validação e gerenciamento dos requisitos.

### 2.2.2. ELICITAÇÃO DE REQUISITOS

A elicitação de requisitos ou, simplesmente, fase de levantamento de requisitos, compreende um conjunto de atividades em que os desenvolvedores utilizam para descobrir as necessidades dos *stakeholders*, buscando aplicar da melhor maneira os requisitos essenciais do sistema a ser desenvolvido. As necessidades, expectativas e recursos esperados do sistema serão levados em consideração de cada *stakeholder*. O intuito é prover de forma correta e completa entendimento do sistema.

A atividade de levantamento de requisitos é composta de quatro dimensões: o entendimento do domínio da aplicação, o entendimento do problema, o entendimento do negócio, e o entendimento das necessidades e das restrições dos *stakeholders*.

O levantamento envolve todo um conjunto de ações, visando capturar e registrar as informações que darão base para o entendimento do domínio do problema e a consequente especificação dos requisitos (KOURI, 2007).

A dificuldade para obtenção desse conhecimento é algo notório, pois os envolvidos têm diferentes experiências, conhecimentos e preconceitos. Outro fato que implica na dificuldade de adquirir o conhecimento é que os *stakeholders* podem omitir informações preponderantes, as quais, para eles, parecem óbvio, o que pode ocasionar em informações incompletas.

Para isso foram desenvolvidas inúmeras técnicas para elicitação de requisitos podendo ser usadas com o objetivo de colher informações como entrevistas, questionários, observação, cenários, análise social, etnografia e leitura de documentos (SOMMERVILLE, 2007). Além destas existem as técnicas em grupos em que é baseada no envolvimento dos *stakeholders* como *brainstorming* e reuniões de RAD (*Rapid Application Development*) e JAD (*Joint Application Development*).

### 2.2.3. ANÁLISE E NEGOCIAÇÃO

Após a fase de levantamento de requisitos, estes passarão pela fase de análise e negociação dos requisitos. Essa atividade na engenharia de software tem por objetivo descobrir inconsistências, pontos incompletos e os conflitos entre os requisitos obtidos, encontrando uma solução que estabeleça um acordo de mudanças que satisfaça a todos os *stakeholders* do sistema a ser desenvolvido. É importante enfatizar que essa fase pode ocorrer intercalada com a fase de elicitação de requisitos. Mas a atividade em si, ocorrerá, detalhadamente, logo após o esboço inicial dos requisitos produzidos.

Aspectos importantes que a fase de análise de requisitos pretende identificar são requisitos conflitantes, ambíguos, redundantes, esquecidos e duplicados. Os usuários devem discutir, priorizar e negociar esses problemas até que se obtenha um acordo com possíveis modificações e simplificações dos requisitos.

A negociação é o processo em que se tenta resolver conflitos entre usuários. Esse processo identifica as principais necessidades de cada usuário, atribuindo prioridades aos requisitos e, em seguida é feito a análise dos resultados para garantir que os requisitos mais críticos sejam atendidos.

### 2.2.4. DOCUMENTAÇÃO DE REQUISITOS

O documento de requisitos é um produto obtido através das atividades de desenvolvimento de requisitos que reúne necessidades e propósitos requeridos pelos *stakeholders*, o qual serve como um contrato entre os desenvolvedores e usuários.

Segundo a norma IEEE (1996) o documento de requisitos objetiva incluir declarações não ambíguas dos requisitos e ser completo. Outras características importantes do documento é que ele deve ser verificável, consistente, modificável, rastreável e usável durante todas as fases do ciclo de vida do requisito.

Dentre as principais metas a serem alcançadas pelo documento de requisitos, existem as essenciais, que visam à completude do documento. São elas: ser instrumento de acordo entre cliente e fornecedor sobre o que software deverá fazer; reduzir o tempo total do processo de criação de software; servir de

base de especificação de software e projeto; dar suporte nas fases de validação do software e gerenciamento do projeto; além de permitir o rastreamento e gerenciamento dos requisitos durante a fase de desenvolvimento e evolução do sistema.

### 2.2.5. VALIDAÇÃO DOS REQUISITOS

A validação de requisitos é a atividade do processo de engenharia de requisitos que certifica se o documento de requisitos é consistente com as necessidades dos usuários.

De acordo com Sommerville (2007) esta atividade dedica-se a mostrar que os requisitos obtidos definem realmente o sistema que o usuário necessita. Esta fase sobrepõe à análise, pois compreende a descoberta de problemas com os requisitos.

A atividade de validação de requisitos é fundamental para o processo de engenharia de requisitos porque ela procura diminuir os custos com a identificação de erros no documento de requisitos, quando eles são encontrados apenas na fase de desenvolvimento do sistema ou quando o sistema já está em operação.

Pressman (2006) diz que o principal mecanismo de validação de requisitos é a revisão técnica formal. A revisão é feita por equipe composta de engenheiros de software, clientes, usuários e outros interessados que analisam a especificação procurando por erros de conteúdo ou de interpretação, áreas com esclarecimentos não especificados, informações omissas, inconsistências, requisitos conflitantes e requisitos irrealísticos.

Algumas técnicas de validação de requisitos podem ser usadas para facilitar a validação do sistema como (SOMMERVILLE, 2007):

- Revisões de requisitos: os requisitos são examinados de forma sistemática pelos revisores os quais procuram solucionar problemas identificados;
- Prototipação: um modelo executável do sistema é mostrado aos clientes e usuários finais em que eles podem testar o sistema para ver se ele está adequado às suas necessidades;

- Gerações de casos de teste: são criados testes em cima dos requisitos obtidos para verificar se serão possíveis ou impossíveis de serem implementados.

#### 2.2.6. GERENCIAMENTO DE REQUISITOS

O gerenciamento de requisitos é uma atividade que é desenvolvida durante todas as fases do processo de engenharia de requisitos. Essa fase visa compreender e controlar as mudanças dos requisitos de sistema. É feito um acompanhamento dos requisitos individuais e dos requisitos dependentes ao longo do processo de desenvolvimento, de modo que seja possível avaliar o impacto das mudanças de requisitos (SOMMERVILLE, 2007).

Devido às diversas mudanças que podem ocorrer no desenvolvimento de um sistema, o gerenciamento de requisitos usa de um fundamento essencial, a rastreabilidade. A rastreabilidade é a propriedade de uma especificação de requisitos que mostra a facilidade de encontrar os requisitos relacionados (SOMMERVILLE, 2007).

Para um requisito ser rastreável ele deve possuir um identificador. Uma vez identificados os requisitos, tabelas de rastreamento são desenvolvidas. Essas tabelas relacionam os requisitos identificados a um ou mais aspectos do sistema ou do seu ambiente. Dentre as diversas tabelas de rastreamento possíveis podemos exemplificar as seguintes: por características (demonstra como os requisitos se relacionam a características importantes do sistema na visão do cliente); pela fonte (identifica a fonte de cada requisito); por dependência (mostra como os requisitos estão relacionados); por subsistemas (os requisitos são caracterizados pelos subsistemas que eles gerem) e por interface (mostra o relacionamento dos requisitos entre as interfaces internas e externas do sistema) (PRESSMAN, 2006).

### 3 TÉCNICAS DE LEVANTAMENTO DE REQUISITOS

No Capítulo 2 mostrou como a tarefa de obter requisitos é uma atividade relevante dentro da engenharia de requisitos. Para que isso ocorra, inúmeras técnicas de levantamento de requisitos são utilizadas para a facilitação da aquisição dos requisitos. Elas auxiliam na comunicação entre os clientes e projetistas do sistema, oferecendo meios adequados para obter requisitos diferentes e resolver problemas de requisitos divergentes.

Cada técnica tenta explorar características específicas do problema, mas podem existir diversos problemas com distintas características, então faz-se jus o uso de vários métodos para cada tipo de problema.

Esta fase leva em consideração principalmente o contexto social, diferente da fase de implementação, por exemplo, que é puramente tecnológica. Isso valida a necessidade de usar diversas maneiras para obter e resolver os problemas da aquisição de requisitos.

Nas seções seguintes serão apresentadas algumas técnicas de levantamento de requisitos.

#### 3.1. PONTOS DE VISTA

As abordagens orientadas a pontos de vista levam em consideração as percepções de cada *stakeholder* interessado no sistema que está sendo desenvolvido. Pressman (2006) argumenta que cada um deles tem uma visão diferente do sistema proposto, obtém diferentes benefícios com relação ao sucesso do desenvolvimento do software e também está exposto a diferentes riscos caso o sistema não venha atingir as expectativas aguardadas.

O engenheiro de requisitos tem a função de criar e organizar uma lista com os diferentes pontos de vista fornecidos de cada interessado à medida que for surgindo os requisitos do sistema.

Sommerville (2007) acrescenta que os pontos de vista podem ser divididos em três tipos genéricos: pontos de vista de interação (são pessoas ou sistemas que interagem com o sistema), pontos de vista indiretos (*stakeholders* que não utilizam o sistema diretamente, mas tem influência sobre os requisitos do sistema)



e pontos de vista de domínio (são características e restrições que afetam os requisitos de sistema).

Após a coleta dos diferentes pontos de vista dos interessados, os requisitos resultantes podem ser inconsistentes ou conflitantes. O engenheiro de requisitos tem a tarefa de categorizar essas informações e prevalecer os requisitos que agregam consistência ao sistema (PRESSMAN, 2006).

### 3.2. ENTREVISTA

A entrevista é uma das técnicas de levantamento de requisitos mais utilizada. Isso pode ser notado quando o engenheiro de requisitos discute com os usuários e/ou clientes no intuito de encontrar os requisitos ideais para o sistema a ser desenvolvido. Para isso, usa de questões, as quais, depois de uma filtragem, poderão se tornar em possíveis requisitos do sistema.

Basicamente existem dois tipos de entrevistas (Sommerville, 2007): as entrevistas fechadas, perguntas são definidas previamente e o *stakeholder* irá respondê-la da forma que foi concebida; e as entrevistas abertas em que não há um roteiro predefinido de questões, o engenheiro de requisitos explora diversos assuntos com a finalidade de obter uma maior compreensão sobre as necessidades do *stakeholder*.

As vantagens da utilização dessa técnica é que ela é mais flexível e tem o objetivo de adquirir informações de caráter subjetivo. Além disso, aproxima o engenheiro de requisitos do usuário do sistema, fazendo com que o usuário se sinta atuante do desenvolvimento do sistema. As desvantagens seriam o custo de tempo e esforço caso precise o engenheiro de requisitos entrevistar inúmeros usuários do sistema. Isso resultaria em uma maior dificuldade para tabular todos os dados obtidos (MARTINS, 2001).

### 3.3. QUESTIONÁRIO

Uma das técnicas de elicitação de requisitos que pode abranger um grande número de pessoas é o questionário. O seu uso é essencial quando se deseja obter informações de inúmeras pessoas. Além disso, justifica-se a sua

aplicabilidade 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.

Kendall (1992) argumenta que é uma técnica que permite adquirir informações de várias pessoas afetadas pelo sistema. Pode-se obter um *feedback* sobre problemas ou identificar possíveis melhorias em relação ao sistema.

O processo de criação do questionário não é tão simples como pode aparentar. É preciso usar uma metodologia para a formulação das questões, de acordo com o perfil de usuários que irão responder o questionário. O planejamento adequado sobre o conteúdo, formato, ordem, clareza e não ambiguidade das questões são fatores relevantes na construção do questionário. Outros itens a levar em consideração é prever antecipadamente dúvidas que podem surgir e evitar que ele seja muito extenso (BRAGA, 2008).

Apesar de esta técnica ser de extrema utilidade, ela possui uma desvantagem, a inflexibilidade, pois impossibilita a análise de questões subjetivas, as quais podem ser de grande importância para o entendimento do problema analisado (MARTINS, 2001).

### 3.4. OBSERVAÇÃO

A observação é uma técnica de grande utilidade para aquisição de requisitos. Ela possibilita a inserção do desenvolvedor no ambiente de trabalho em que o usuário ou grupo de usuários trabalham, o qual observa as tarefas executadas pelo usuário, sem interferir no ambiente. O intuito é obter requisitos através da análise das tarefas realizadas pelo usuário.

Uma das técnicas de observação utilizadas para elicitação de requisitos, muito empregada na área das ciências sociais, é a etnografia. Sommerville (2007) define como uma técnica de observação que pode ser usada para compreender os requisitos sociais e organizacionais implícitos de um sistema. Ou seja, o desenvolvedor observa o dia-a-dia com a finalidade de anotar as tarefas reais dos usuários envolvidos.

### 3.5. JOIN APPLICATION DEVELOPMENT

Nos anos 70 a IBM desenvolveu uma técnica para a eliciação de requisitos chamada *Joint Application Development* (JAD). Ela visa criar sessões de trabalho estruturadas, através de uma dinâmica de grupo e recursos visuais, em que os *stakeholders* envolvidos trabalham no intuito de encontrar os requisitos básicos até o *layout* de telas e relatórios do sistema a ser desenvolvido (CARVALHO, 2003). Além disso, a JAD promove a cooperação e o entendimento entre os desenvolvedores e usuários (ELLWANGER, 2005).

Os componentes de uma sessão JAD são entre 8 e 15 pessoas de diversas áreas envolvidas com a produção do software (BRAGA, 2008). Dentre elas a um facilitador que tem a função envolver os participantes nas reuniões, mediando às discussões e fazendo indagações a procura de requisitos não relatados. Tem uma contribuição muito importante, também, para com os usuários que participam das reuniões, fazendo com que estes compartilhem do desenvolvimento do sistema.

Essa técnica se baseia em quatro princípios básicos: dinâmica de grupo, a qual possui um facilitador para estimular a capacidade dos indivíduos envolvidos; técnicas audiovisuais visa aumentar a comunicação e o entendimento entre os participantes; manutenção do processo organizado e racional e a documentação padrão em que os participantes da reunião preenchem e assinam (CARVALHO, 2003).

A metodologia JAD possui duas grandes etapas: planejamento, cuja finalidade é levantar e especificar requisitos; e projeto, em que se lida com o projeto do software (CARVALHO, 2003).

As experiências com a metodologia JAD que resultaram em sucesso mostraram poucas mudanças de requisitos e uma redução de tempo e esforço gastos no levantamento de requisitos. Isso torna essa técnica relevante para o processo de aquisição de requisitos (BRAGA, 2008).

### 3.6. PROTOTIPAÇÃO

A prototipação é uma técnica que visa construir um protótipo inicial do sistema proposto. É uma versão inicial do sistema de software usado para

demonstrar conceitos, experimentar opções de projeto, conhecer mais sobre o problema e suas possíveis soluções. Usa-se, normalmente, uma metodologia de desenvolvimento rápido e iterativo para que os clientes e/ou usuários possam usar o sistema o mais cedo possível (SOMMERVILLE, 2007).

Os protótipos permitem que os usuários experimentem o software, podendo eles, usuários, verificarem como o protótipo auxilia no seu trabalho. Além disso, durante o seu uso, podem revelar erros e omissões nos requisitos propostos, podendo os usuários apresentar novos requisitos para o sistema (SOMMERVILLE, 2007).

Outros benefícios dos protótipos são a possibilidade de validação dos requisitos encontrados e a verificação da viabilidade do projeto proposto (é um meio que os *stakeholders* encontram para saber se o desenvolvimento do sistema completo será muito custoso, acarretando na inviabilidade do projeto). Também auxiliam na criação do projeto de interface com o usuário, pois os usuários, por exemplo, podem dizer sobre a dificuldade de utilizar o sistema em relação a encontrar funções, a complexidade dos menus, a linguagem complicada do software, botões que não representam o que efetivamente executam e dentre outras dificuldades com relação à interface.

Existem algumas desvantagens da utilização da prototipagem, dentre elas, está o risco de usar a estrutura do protótipo para a implementação da primeira versão do sistema (MARTINS, 2001) e quando o cliente é notificado que o protótipo precisa ser refeito para que os altos níveis de qualidade possam ser garantidos, o qual aparentemente estaria funcionando corretamente (PRESSMAN, 2006).

### 3.7. CASOS DE USO

A técnica de modelagem através de casos de uso está sendo utilizada cada vez mais pelos engenheiros de requisitos. Idealizada, na década de 70, pelo engenheiro de software sueco, Ivar Jacobson, tem sido de grande utilidade para a documentação dos requisitos funcionais de um sistema (BEZERRA, 2006).

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. Foi incorporada pela Linguagem de Modelagem Unificada (UML) que é representada pelo diagrama de casos de uso (BEZERRA, 2006).

Uma das características mais importantes dos casos de uso é que independentemente de sua forma, um caso de uso descreve o software ou o sistema do ponto de vista do usuário (PRESSMAN, 2006).

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). É demonstrado na Figura 1 um exemplo do diagrama de caso de uso.

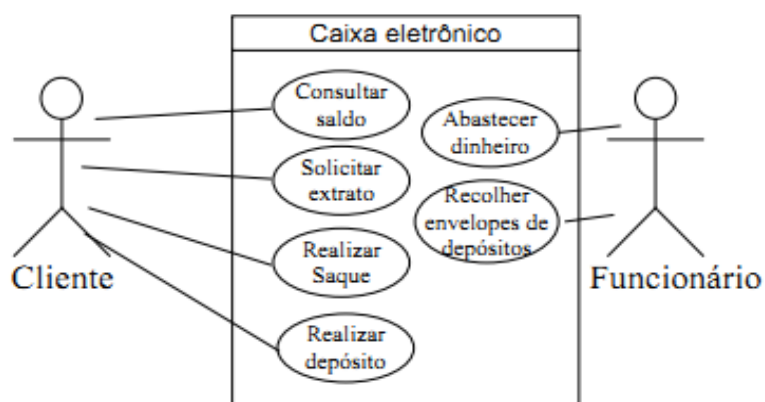


Figura 1 – Diagrama de casos de uso

Fonte: Furlan (1998)

A Figura 1 mostra um exemplo de diagrama de casos de uso, onde o cliente e funcionário são atores que interagem com o sistema do caixa eletrônico. O cliente faz as seguintes ações no sistema por meio dos casos de uso “consultar saldo”, “solicitar extrato”, “realizar saque” e “realizar depósito”. Já o funcionário realiza no sistema as tarefas “abastecer dinheiro” e “recolher envelopes de depósitos”.

Os casos de uso precisam ser descritos de forma narrativa entre as interações que ocorrem entre os elementos externos e o sistema. A descrição traz detalhes da execução do caso de uso que não pode ser demonstrado no diagrama de casos de uso.

Segundo (BEZERRA, 2006) as descrições comumente utilizadas são descrição contínua, numerada e particionada. Na Figura 2 podemos observar um exemplo de descrição contínua para o caso de uso “realizar saque”.

**Tabela 4-1 Exemplo de descrição contínua**

O Cliente chega ao caixa eletrônico e insere seu cartão. O sistema requisita a senha do Cliente. Após o Cliente fornecer sua senha e esta ser validada, o Sistema exhibe as opções de operações possíveis. O cliente opta por realizar um saque. Então o Sistema requisita o total a ser sacado. O Sistema fornece a quantia desejada e imprime o recibo para o Cliente.
--

Figura 2 – Descrição contínua

Fonte: Bezerra (2006)

As vantagens do uso dessa técnica, dentro da engenharia de requisitos, são de grande relevância para a obtenção de requisitos de um software. Ela possibilita um meio mais adequado de interação com o usuário, facilitando a comunicação. Agrega a ela, também, a parte descritiva, fundamental para a demonstração de detalhes intrínsecos ao diagrama, não podendo ser visualizado na sua forma pictórica.

### 3.8. CENÁRIOS

Um dos meios para se fazer um levantamento de informações visando desenvolver um novo sistema é através da construção de cenários. Essa técnica corresponde a uma descrição parcial do comportamento da aplicação. Ela pode ocorrer em um dado momento e em uma situação específica. Utiliza-se de uma coleção de narrativas de situações no domínio para levantar informações, identificar problemas e, com isso, resolver problemas antecipadamente (KOURI apud LEITE, 2000).

Os cenários devem ser elaborados durante as reuniões entre os *stakeholders*. Segundo Sommerville (2007), de maneira geral, o cenário deve possuir: uma descrição do que os usuários esperam do sistema no início do cenário; uma descrição do fluxo normal de eventos no cenário; uma descrição do que pode ocasionar em erro e como isso será tratado; relatar atividades que

podem ocorrer simultaneamente e; uma descrição do estado de sistema no fim do cenário.

Abaixo podemos observar uma estrutura definida por (KOURI apud LEITE, 2000, p. 44) para descrição de um cenário. É composta dos seguintes itens:

- **Título:** identifica o cenário;
- **Objetivo:** é definido como o objetivo pode ser alcançado;
- **Contexto:** descreve o estado inicial do cenário, suas pré-condições, o local (físico) e tempo;
- **Recursos:** são definidos os objetos passivos com os quais os atores lidam;
- **Atores:** pessoas, organizações e dentre outras que têm uma atribuição dentro do cenário;
- **Episódios:** é uma ação realizada por um ator onde participam outros atores utilizando recursos disponíveis. Episódios podem conter restrições e exceções. Um episódio, também, pode se referir a outro cenário.
- **Exceções:** tratamento para situação anormal ou de erro.

### 3.9. ANÁLISE DE DOCUMENTOS

Essa técnica é aplicada através da dedução de conhecimentos já expressos, ou melhor, escritos dentro do universo de informações. Para a modelagem de requisitos ela é muito útil, pois ajuda a definir os objetos que irão compor o modelo. Com a análise de documentos, é possível um contato com o vocabulário utilizado no domínio do problema e, também, facilitar a construção de um glossário de termos especializados, o qual auxiliará na definição dos objetos e que os *stakeholders* envolvidos no projeto possuam um conhecimento similar (KOURI, 2007).

### 3.10. BRAINSTORMING

*Brainstorming* é uma das técnicas de reuniões em grupo mais conhecida para levantamento de requisitos. Ela consiste em uma reunião de especialistas de diversos setores, sendo que cada componente tem a função de estimular ao outro

a criação de ideias, visando à resolução do problema em questão. As ideias obtidas não devem ser criticadas. Normalmente utiliza-se essa técnica na fase inicial do desenvolvimento quando pouco do projeto é conhecido e é vital a necessidade de novas ideias (BATISTA; CARVALHO, 2003).

A técnica *brainstorming* objetiva a geração de novas ideias através da liberdade dos participantes de criar, e esses de aceitar as ideias sugeridas. Uma sessão bem sucedida de *brainstorming* resulta em conjunto de boas ideias e os participantes sentem que cada um contribuiu de alguma maneira para a solução do problema. *Brainstorming* possui uma eficácia considerável para concepção de um sistema (BATISTA; CARVALHO, 2003).

### 3.11. STORYTELLING/GROUP STORYTELLING

*Group Storytelling* é uma técnica que se baseia na utilização de histórias em grupos de pessoas, incluindo organizações, como um método de comunicação para que os envolvidos possam compartilhar conhecimento. O *Group Storytelling* emprega o uso de técnicas de contagem de histórias que inspiram e motivam os participantes através de linguagens mais cotidianas, criando um entretenimento maior durante o processo de estruturação do conhecimento. Os participantes são estimulados a narrar histórias relacionadas aos fatos cujo conhecimento se quer elicitar. As histórias obtidas são compartilhadas com o grupo de pessoas envolvidas, o que resulta em compartilhamento de conhecimentos, agregação de novos conhecimentos e, por fim, em aprendizado (MIGON; JUNIOR, 2007).

A técnica é aplicada geralmente em um formato assíncrono e distribuído em que se podem utilizar ferramentas computacionais para permitir o acesso mais fácil aos participantes. Assim os integrantes não precisam estar no mesmo lugar ao mesmo tempo. Devido à dinâmica das organizações, como multinacionais, esses fatores auxiliam e facilitam na aplicação da técnica podendo ser expandida a várias filiais localizadas em cidades, estados e países diferentes (GONÇALVES, 2010).

O valor da técnica da *Group Storytelling* pode ser observada no que ela contribui para validação da história e a contribuição da sua construção por outros integrantes da empresa. Esse fator é de enorme relevância para coletar



informações autênticas para a construção do modelo de negócio (Migon e Junior, 2007).

### 3.12. USER STORIES

*User Stories* ou histórias de usuários tem o mesmo propósito dos casos de uso, mas não é a mesma coisa. *User Stories* são descrições feitas pelos clientes, como funções, as quais gostariam que o sistema realizasse. Elas também servem para estimar o plano de entrega das versões do sistema e para que sejam criados testes de aceitação de software. Estes testes indicam se o software cumpre mesmo o que cliente requisitou no seu plano inicial do sistema (SOUZA, 2004).

As histórias devem possuir detalhes de forma a minimizar o risco da produção do software, assim podendo estipular quanto tempo levará a implementação da descrição. Através dessas definições do usuário é possível preparar o plano de versões a serem demonstradas para o mesmo. O ponto central das *User Stories* está nas necessidades do usuário.

### 3.13. REUSO DE REQUISITOS

A engenharia de software baseada em reuso é uma estratégia de comparação de engenharia de software na qual o processo de desenvolvimento é voltado ao reuso do software existente. Segundo Sommerville (2007) somente nos últimos dez anos estão ocorrendo um maior desenvolvimento baseado em reuso.

O reuso de requisitos, por exemplo, de acordo com diversos autores, deve ser orientado através de métodos, esquemas e técnicas para que seja realizado da melhor maneira possível. Seguindo essa escrita, garante-se economia no tempo de elicitação, análise e validação dos requisitos, além de economizar dinheiro, pois mais de 80% dos requisitos são mais ou menos os mesmos em cada projeto (TONIOLO, 2011).

## 4 FERRAMENTAS QUE AUXILIAM NA AQUISIÇÃO E GERÊNCIA DE REQUISITOS

Em capítulos anteriores, foi mencionado que a tarefa de obter requisitos de forma correta é vital para o desenvolvimento de um sistema e para satisfação do cliente e/ou usuário. Um dos meios para facilitar essa atividade dentro da engenharia de software é o uso de ferramentas CASE (*Computer Aided Software Engineering*). Essas ferramentas auxiliam em atividades de engenharia de software, desde análise de requisitos e modelagem até programação e testes. São ferramentas automatizadas que objetivam facilitar o desenvolvedor de sistemas em várias etapas do ciclo de desenvolvimento de software.

Dentre as inúmeras ferramentas CASE já implementadas, será apresentado a seguir, como exemplo, algumas das ferramentas mais conhecidas para aquisição e gerência de requisitos.

### 4.1. TEMPLATES DE DOCUMENTOS

*Template* ou simplesmente “modelo de documento” é um documento sem conteúdo. Ele possui apenas uma apresentação visual e instruções em que locais se deve preencher o conteúdo adequado. Na engenharia de requisitos esta ferramenta pode ser utilizada para apresentar ao cliente o resultado final da análise e especificação de requisitos, além de outras atividades correlacionadas, para que o próprio possa validá-la. É nesse documento que fica declarado a concordância entre as partes (clientes, analistas e desenvolvedores) sobre o que deve ser desenvolvido.

Em geral esse modelo de documento deve ser basicamente textual, evitando o uso de termos técnicos, e ilustrados como modelos gráficos que demonstrem mais claramente a visão que os analistas tiveram dos problemas e dos requisitos para o desenvolvimento do sistema.

## 4.2. ASTAH

Com o descontinúo do projeto JUDE (*Java and UML Developer Environment*) em 2009, uma nova ferramenta de modelagem de diagramas UML foi desenvolvida, ASTAH. Ela é uma ferramenta muito simples de utilizá-la e podemos criar diagramas de casos de uso, de classes, de atividade, de sequência e dentre outros. No site do fabricante pode ser encontrada uma versão para teste, já que ela é uma ferramenta proprietária. É de grande serventia para os desenvolvedores de sistemas para levantamento de requisitos.

Abaixo na Figura 3 é possível observar a interface do software ASTAH na criação de um diagrama de classes:

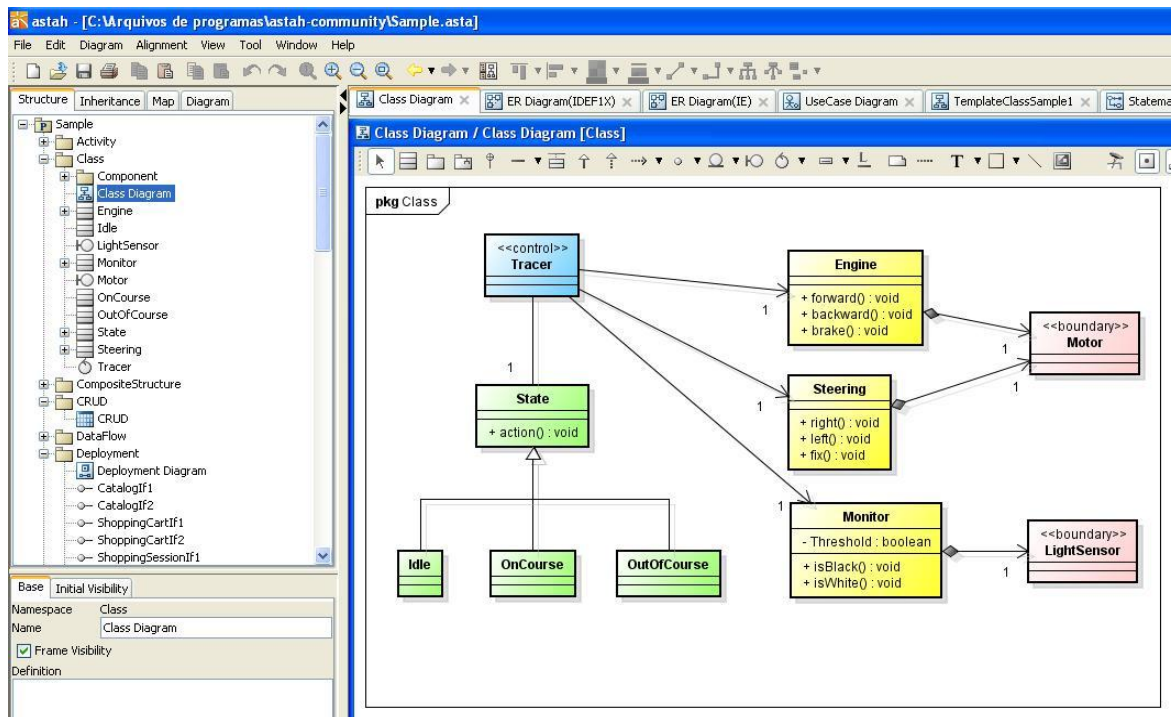


Figura 3 – Modelando um diagrama de classes no ASTAH

Fonte: <http://3.bp.blogspot.com> (2010)

## 4.3. RATIONAL REQUISITEPRO

A *IBM/Rational* desenvolveu uma ferramenta que objetiva o gerenciamento de requisitos de sistema chamada *Rational RequisitePro*. Ela possui um banco de dados centralizado onde são alocados os requisitos, permitindo o acesso de

forma mais adequada pela a organização. A ferramenta possui integração com a *Microsoft Word*, permitindo que os documentos sejam ligados ao banco de dados dinamicamente (THEILACKER, 2008).

Dentre as diversas características que a ferramenta possui podem-se citar as seguintes (THEILACKER, 2008):

- organização dos requisitos através de parametrizações;
- gerenciamento de características específicas dos requisitos, através de definição de atributos;
- controle de relacionamentos entre requisitos;
- poder visualizar o impacto nas alterações dos requisitos;
- controlar o histórico de mudanças dos requisitos;
- gerar informações estatísticas de requisitos;
- desenvolver um projeto originário com base em um existente no banco de dados, podendo usufruir dos mesmos tipos de documentos, tipos de requisitos, atributos, e tipos de segurança do projeto de origem;
- permite relacionar os requisitos a grupos de segurança;
- permite o acesso distribuído dos requisitos.

Na Figura 4 podemos observar a interface da ferramenta Rational RequisitePro:

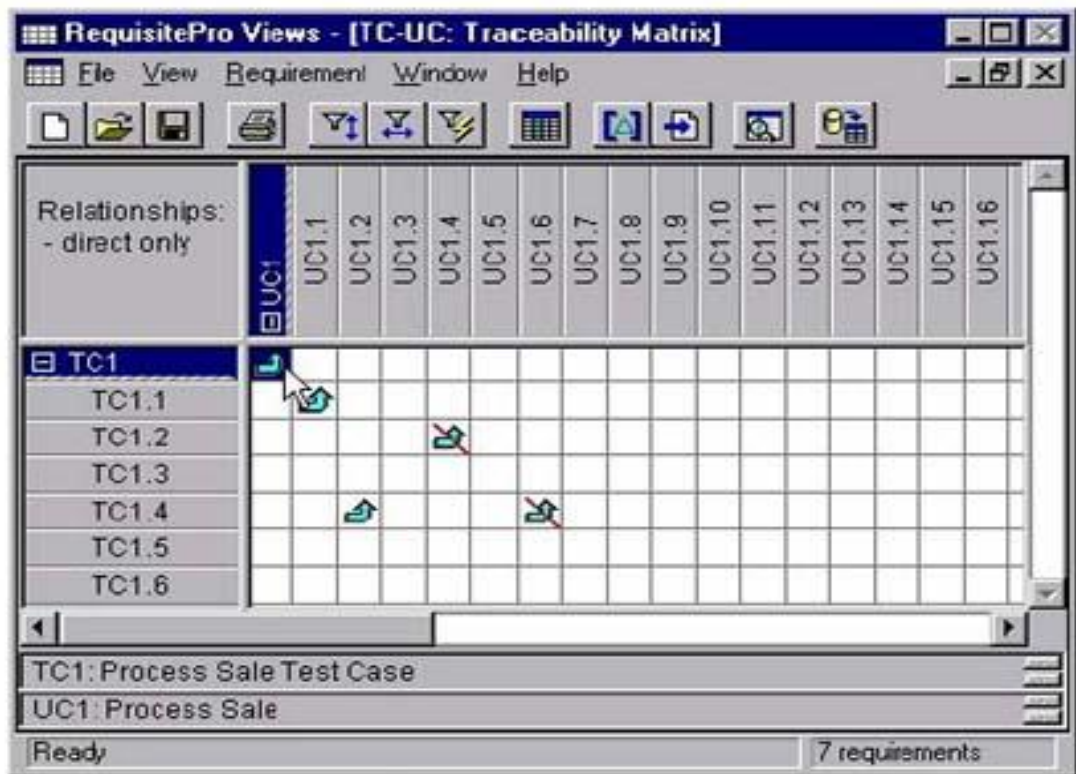


Figura 4 – Interface do Rational RequisitePro

Fonte: <http://www.ibm.com/developerworks/rational/library/835.html> (2003)

#### 4.4. OSRMT

A ferramenta OSRMT, “*Open Source Requirements Management Tool*” (ou, simplesmente, ferramenta de código aberto para gerência de requisitos) foi desenvolvida na linguagem Java, tendo como finalidade, apoiar o processo de gerência de requisitos. Licenciada sob os termos da GPL (*General Public License*) possui versão 1.5 como a mais estável e está disponível no site sourceforge.net.

O OSRMT tem como uma de suas metas principais permitir uma completa rastreabilidade do ciclo de vida de desenvolvimento de software em relação aos requisitos. Podemos destacar as seguintes funções da ferramenta: registro do autor, origem e motivo da necessidade de cada requisito; registro de casos de uso, status e origem de cada requisito; rastreabilidade (através de gráficos que identificam todas as dependências entre requisitos); definição e organização de artefatos e entrada de dados; e a possibilidade de gerar relatórios em formato PDF (*Portable Document Format*) de forma padronizada (YOSHIDOME, 2010).

Na Figura 5 é mostrada a interface do OSRMT:

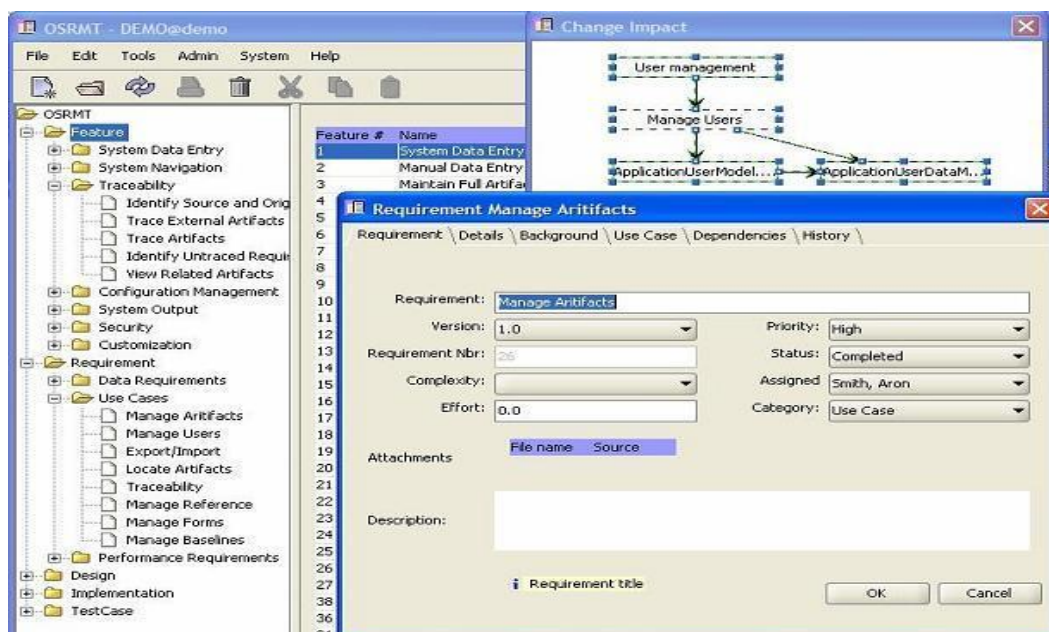


Figura 5 – Interface do OSRMT

Fonte: [http://download.cnet.com/OSRMT/3000-2383\\_4-10552248.html](http://download.cnet.com/OSRMT/3000-2383_4-10552248.html) (2006)

#### 4.5. CALIBERRM

O CaliberRM é uma ferramenta de gerenciamento de requisitos desenvolvida pela empresa *Borland*. Com a ferramenta os *stakeholders* envolvidos podem acompanhar todos os requisitos encontrados e modificados durante todo o ciclo de vida do aplicativo. Possui representação das rastreabilidades entre os requisitos através de matriz e diagramas (Theilacker, 2008).

Segundo *Borland* (2012) a ferramenta possui outras características dentre as quais estão:

- banco de dados centralizado e permite que sejam definidas matrizes e rastreabilidade entre os artefatos;
- permite relacionamentos bidirecionais, em que é possível identificar o requisito que prova impacto e aquele que sofre a ação;
- permite visualizar relacionamento indiretos entre requisitos;

- relacionamento de artefatos modificados são marcados como suspeitos, cabendo ao usuários decidir se esse relacionamento ainda é válido;
- permite o relacionamento de vários tipos de documentos aos requisitos;
- mantém o rastreamento entre requisitos de projetos distintos.

Na Figura 6 é possível visualizar a interface do CaliberRM:

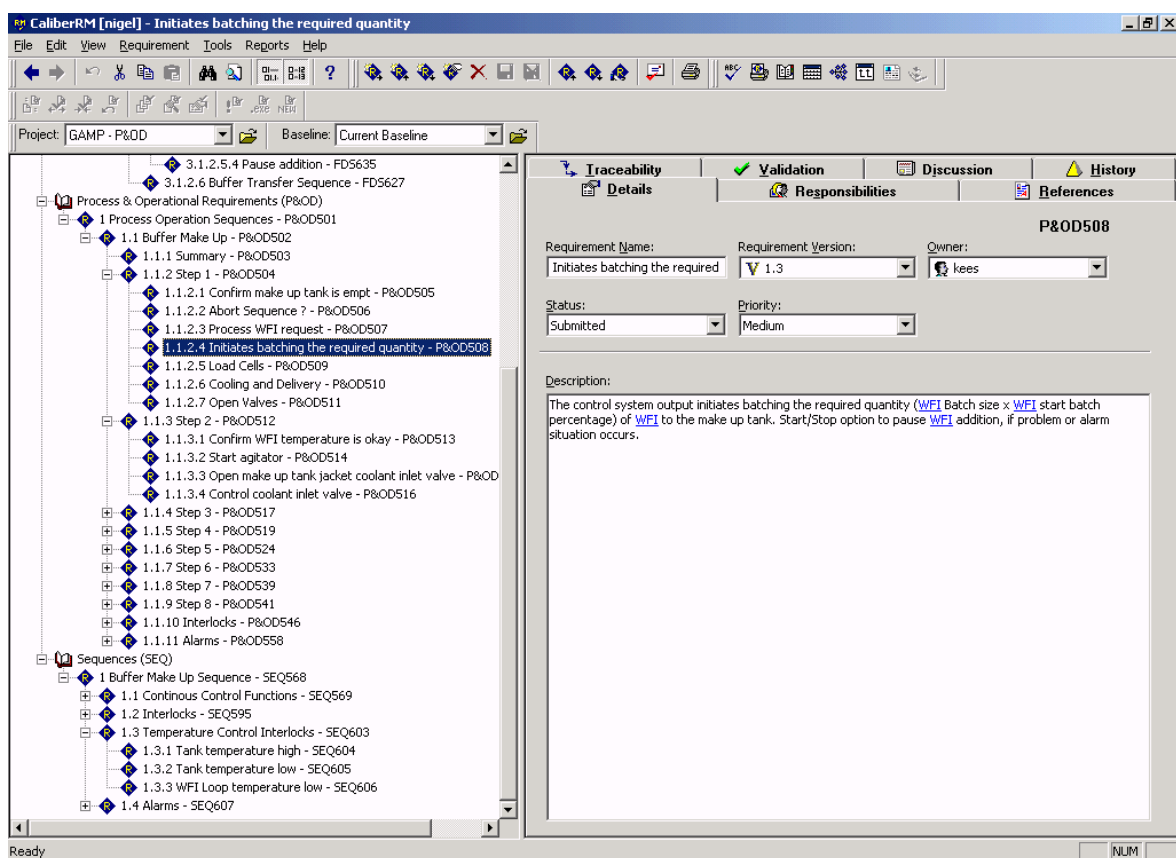


Figura 6 – Interface do CaliberRM

Fonte: <http://www.dthomas.co.uk/dtalm/products/borland-caliber.htm> (2012)

## **5 ANÁLISE DAS TÉCNICAS DE LEVANTAMENTO DE REQUISITOS PARA DESENVOLVIMENTO DE SOFTWARE NAS EMPRESAS DE VITÓRIA DA CONQUISTA – BA**

Durante o decorrer do trabalho salientou-se que a atividade de levantamento de requisitos é fator primário e essencial dentro da engenharia de software. Ela é quem dá o suporte para que as outras etapas do processo de desenvolvimento de software possam ocorrer.

Para mostrar a sua relevância no desenvolvimento de software, este capítulo tem o intuito de mostrar uma pesquisa feita nas empresas que produzem software em Vitória da Conquista, Bahia. O estudo visou verificar como é feita a atividade de levantamento de requisitos nessas empresas.

### **5.1. METODOLOGIA DA PESQUISA**

Para a realização da pesquisa foi escolhido o universo das empresas desenvolvedoras de software de Vitória da Conquista. Para encontrar as empresas foi utilizado sites de busca e informações de conhecidos, o qual possibilitou encontrar 20 empresas de desenvolvimento de software nessa região, o que não representa todo o universo da cidade de Vitória da Conquista. Isso pode ser explicado pela dificuldade de encontrar algumas empresas, por serem empresas individuais e se encontrarem em lugares remotos de difícil acesso, mudaram de endereço, estão em reforma, ou no seu estabelecimento não tinham informativos de que a empresa constava naquele local.

Foram utilizados questionários para a coleta dos dados, os quais foram entregues no próprio estabelecimento da empresa. O questionário foi respondido por um desenvolvedor escolhido aleatoriamente. O total de amostras observadas foi de 18 empresas, já que duas empresas não entregaram o questionário. O período para o recolhimento dos dados durou do dia 07/02/2012 ao dia 09/03/2012.

A pesquisa, em si, não tem a finalidade de fazer um estudo probabilístico na cidade mencionada e sim fazer uma análise com base nos dados coletados.



O questionário entregue às empresas possuía onze questões que indagavam aos desenvolvedores os seguintes assuntos:

- a experiência e a participação dos desenvolvedores na aquisição dos requisitos;
- técnicas e ferramentas utilizadas no levantamento de requisitos;
- fatores relacionados à importância na escolha de uma determinada técnica;
- como é feito a validação dos requisitos encontrados;
- o relacionamento dos desenvolvedores com os clientes e/ou usuários no desenvolvimento do software;
- as dificuldades encontradas no processo de levantamento de requisitos.

A ferramenta utilizada para tabulação dos dados e criação dos gráficos apresentados neste capítulo foi o *Microsoft Excel 2010*.

Nas seções seguintes serão apresentados os resultados obtidos e as análises feitas com base nos dados coletados.

## 5.2. EXPERIÊNCIA COM LEVANTAMENTO DE REQUISITOS E FREQUÊNCIA DO USO DE TÉCNICAS PARA ESSA ATIVIDADE

No estudo feito nas empresas, uma das questões abordadas foi com relação aos anos de experiência dos desenvolvedores na atividade de levantamento de requisitos. Com base nos dados coletados, pode-se observar no Gráfico 01 que os anos de experiência dos pesquisados correspondem a: 33% possuem de 3 a 5 anos; 28% possuem mais de 5 anos; 17% possuem de 1 a 2 anos; 11% de 2 a 3 anos e 11% possuem menos de 1 ano. Pode-se concluir através do Gráfico 01 que a maioria dos desenvolvedores de software de Vitória da Conquista tem mais de 3 anos de experiência.

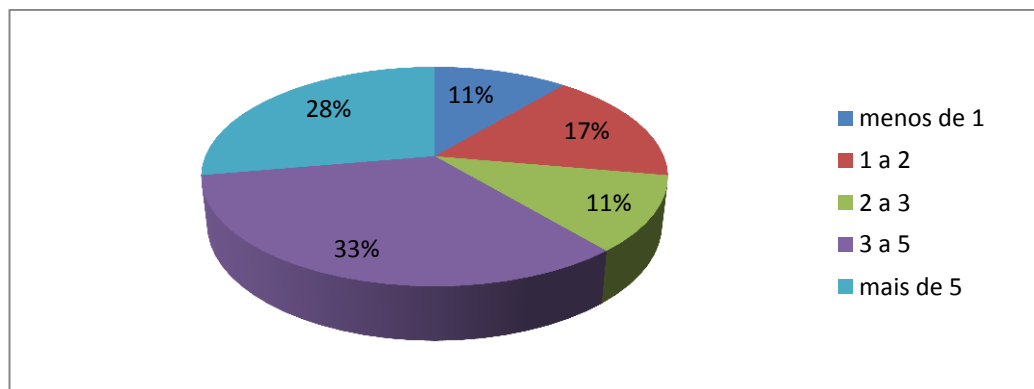


Gráfico 01 – Experiência com levantamento de requisitos

Fonte: Pesquisa direta (2012)

Outro quesito questionado na pesquisa foi com relação à participação dos desenvolvedores da empresa na etapa de levantamento de requisitos durante um projeto. O Gráfico 2 mostra que 28% deles participaram de mais de 12 projetos; 22% participaram de 4 a 6 projetos; 22% participaram de 1 a 3 projetos; 17% de 7 a 9 projetos; e 11% participaram de 10 a 12 projetos.

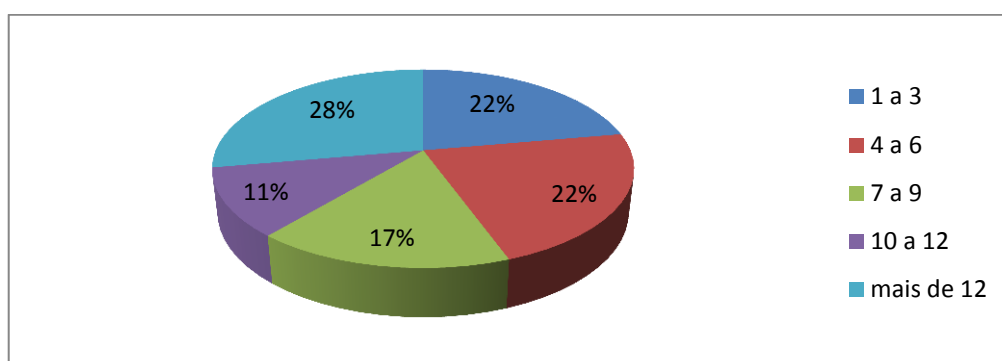


Gráfico 02 – Participação da etapa de levantamento de requisitos em projetos

Fonte: Pesquisa direta (2012)

Observando o Gráfico 02 nota-se que a maioria dos pesquisados participaram de mais de 6 projetos.

Um dos itens de grande relevância questionado aos desenvolvedores pesquisados, o qual norteia este trabalho, foi com relação à frequência da utilização de técnicas de levantamento de requisitos durante os projetos em desenvolvimento. Os resultados mais significativos serão mostrados a seguir.

O Gráfico 03 mostra que a técnica entrevista é usada sempre por 61% dos pesquisados; 28% usam frequentemente; 6% utilizam poucas vezes e 5% conhecem, mas nunca utilizaram. Verifica-se que 95% dos desenvolvedores utilizam essa técnica mesmo que em algumas ocasiões e que 100% deles conhecem esse tipo de técnica.

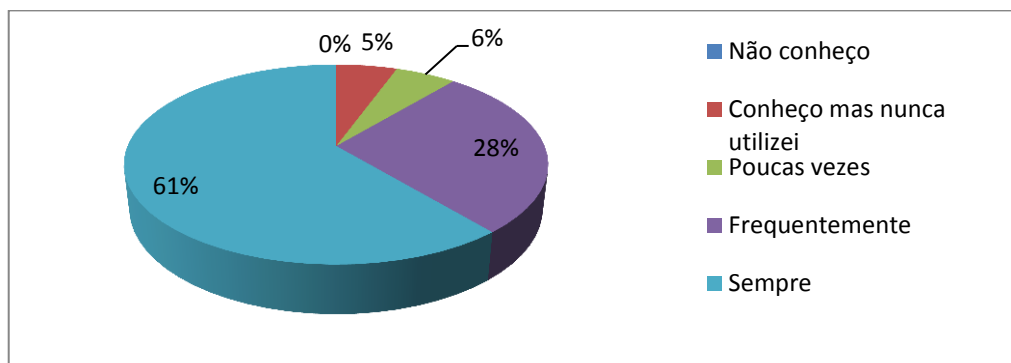


Gráfico 03 – Frequência do uso da técnica Entrevista

Fonte: Pesquisa direta (2012)

Em relação ao uso da técnica de levantamento de requisitos questionário, a qual é mostrada no Gráfico 04 nota-se que 39% conhecem, mas nunca utilizaram; 33% utilizam frequentemente e 28% utilizam poucas vezes. O que se pode notar, é que apesar de ser conhecida por 100% dos participantes é pouco utilizada pelos mesmos. Normalmente essa técnica é utilizada quando há a necessidade de coletar informações de muitos usuários afetados pelo sistema a ser desenvolvido (KENDALL, 1992), a qual não é comumente aplicada em Vitória da Conquista.

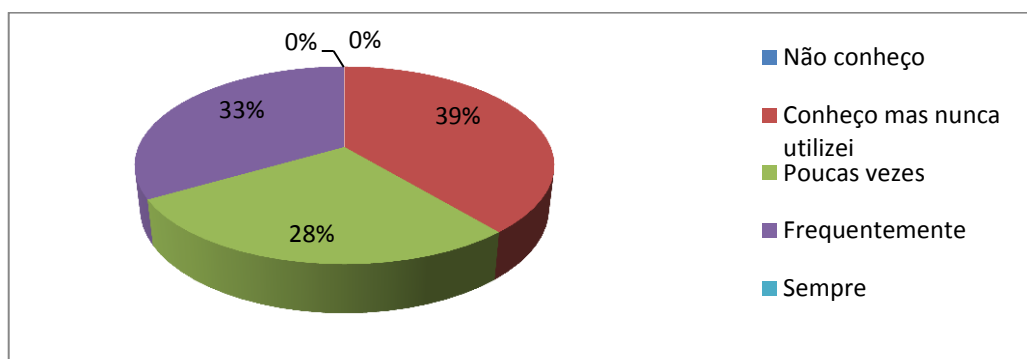


Gráfico 04 – Frequência do uso da técnica Questionário

Fonte: Pesquisa direta (2012)

Na aquisição de requisitos usam-se, também, Diagramas para modelar as informações que os *stakeholders* envolvidos no projeto explanam. Uma das linguagens mais conhecidas e utilizadas para este fim, mencionada no Capítulo 3, é a UML. No Gráfico 05 observa-se a aplicabilidade dessa técnica pelas empresas e constata-se que 39% utilizam frequentemente; 28% conhecem, mas nunca utilizaram; 22% usam sempre e 11% há empregam poucas vezes. Analisando os dados, pode-se observar que a maioria dos pesquisados aplicam esse tipo de técnica sempre ou frequentemente e que 100% conhecem essa técnica.

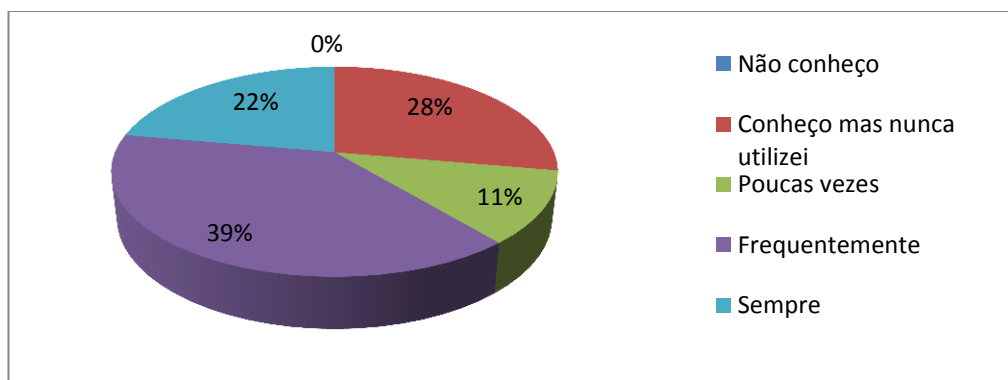


Gráfico 05 – Frequência do uso de Diagramas

Fonte: Pesquisa direta (2012)

Observando o Gráfico 06 que apresenta a frequência da empregabilidade dos Casos de Uso pelos desenvolvedores, nota-se que 33% usam frequentemente; 28% utilizam poucas vezes; 28% conhecem, mas nunca utilizaram e 11% empregam esta técnica sempre. Analisando o gráfico, evidencia-se que a maioria dos pesquisados aplicam esta técnica pelo menos em algumas ocasiões e que 100% conhecem a técnica.

Fazendo uma relação entre os dois Gráficos 05 e 06 percebe-se que os desenvolvedores, em sua maioria, utiliza um tipo de diagrama para fazer o levantamento de requisitos.

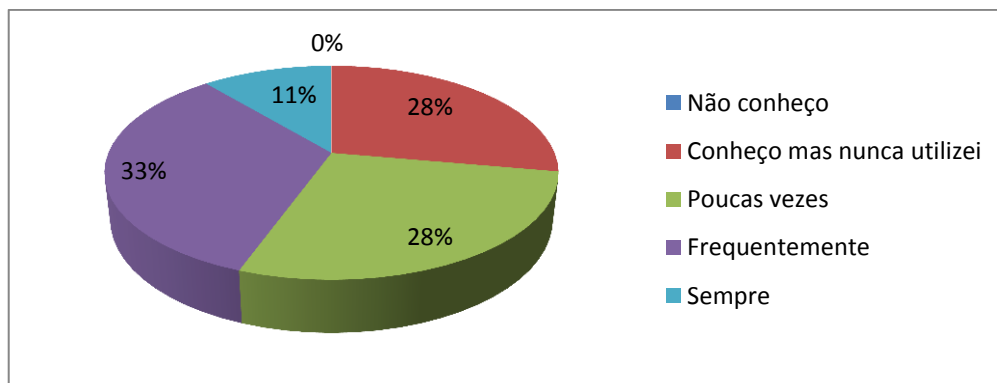


Gráfico 06 – Frequência do uso de Casos de Uso

Fonte: Pesquisa direta (2012)

Algumas técnicas que se valem de descrições narrativas para coletar requisitos de software não são de grande conhecimento por parte dos pesquisados. Pode-se notar que as técnicas Cenários e *User Stories*, mostrada nos Gráficos 07 e 08, respectivamente, não tiveram um alto índice de conhecimento e de utilização. O Gráfico 07 demonstra que 44% não conhecem a técnica Cenários; 28% conhecem, mas nunca utilizaram; 22% aplicam poucas vezes e 6% usam frequentemente. No Gráfico 08 nota-se que 39% não conhecem a técnica *User Stories*; 28% conhecem, mas não utilizam; 22% usam frequentemente e 11% poucas vezes. Então, conclui-se que a maioria dos desenvolvedores desconhece ou nunca utilizou as duas técnicas.

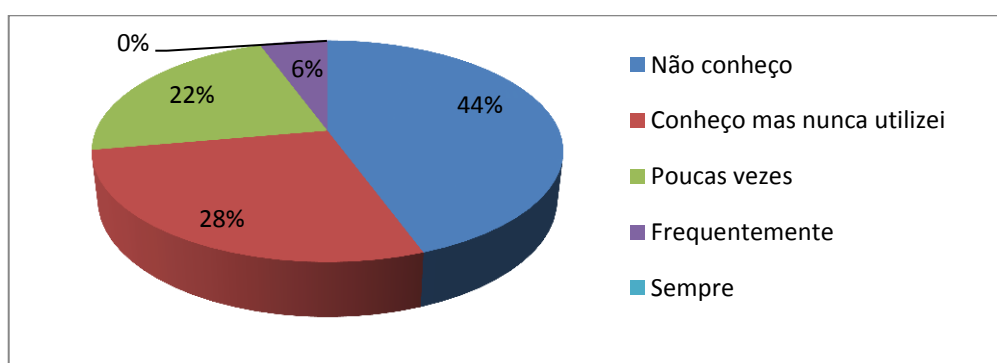


Gráfico 07 – Frequência do uso de Cenários

Fonte: Pesquisa direta (2012)

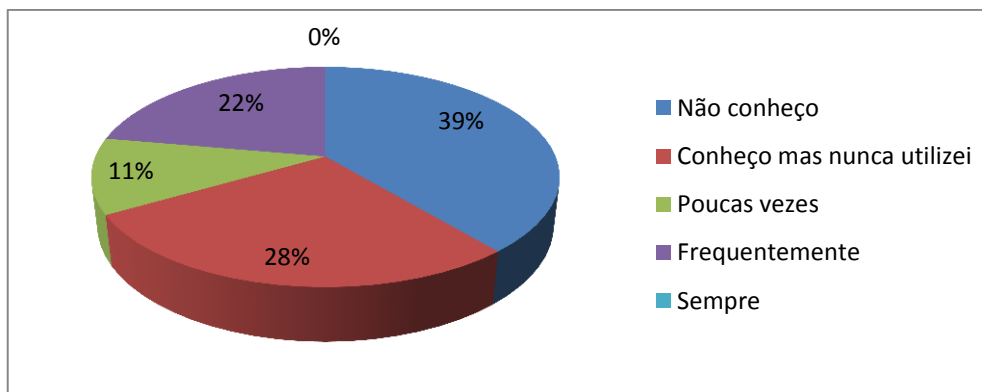


Gráfico 08 - Frequência do uso de User Stories

Fonte: Pesquisa direta (2012)

As reuniões em grupo são comumente utilizadas pelas empresas. Verifica-se no Gráfico 09 que 44% dos desenvolvedores a empregam frequentemente; 28% utilizam sempre; 17% usam poucas vezes e 11% conhecem, mas nunca utilizam. Entretanto, uma técnica de reunião em grupo com metodologia específica para levantamento de requisitos, a *Brainstorming*, mencionada no Capítulo 3, não possui uma frequência de utilização por parte dos desenvolvedores. No Gráfico 10 se pode observar que 39% dos pesquisados conhecem, mas nunca utilizam; 28% desconhecem essa técnica; 17% utilizam poucas vezes; 11% usam frequentemente e 5% utilizam sempre.

Pode-se inferir desses dados que a maioria dos desenvolvedores usam as reuniões para obter requisitos, no entanto a técnica *Brainstorming* não é usualmente aplicada pela maioria dos desenvolvedores nessas reuniões, apesar de 72% dos desenvolvedores conhecerem esta técnica.

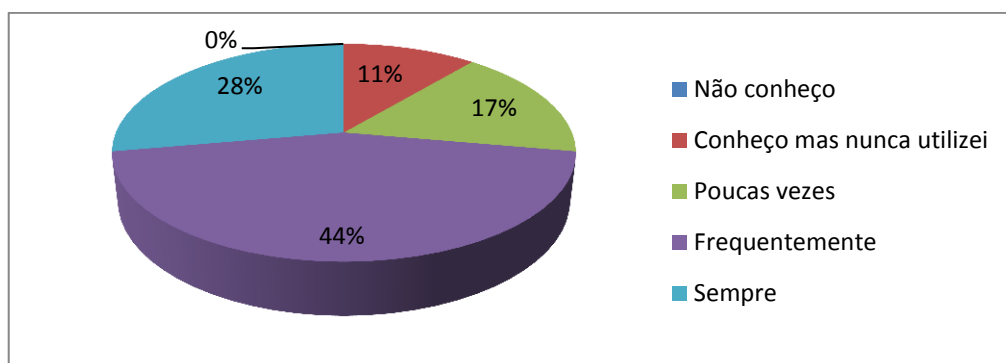
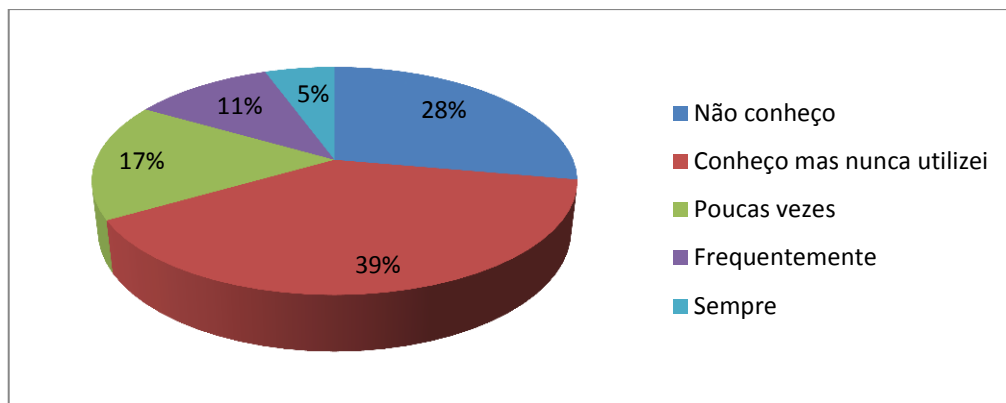


Gráfico 09 – Frequência do uso de Reuniões em grupo

Fonte: Pesquisa direta (2012)

Gráfico 10 – Frequência do uso de *Brainstorming*

Fonte: Pesquisa direta (2012)

A análise de documentos é uma técnica bem aceita por parte dos pesquisados. No Gráfico 11 é possível notar que 45% dos desenvolvedores usam frequentemente essa técnica; 33% utilizam sempre; 11% usam poucas vezes e 11% conhecem, mas nunca utilizam. Observa-se que 78% dos desenvolvedores utilizam essa técnica frequentemente ou sempre. Isso mostra que as empresas de desenvolvimento de software de Vitória da Conquista buscam informações já expressas como fonte de requisitos para a criação dos sistemas.

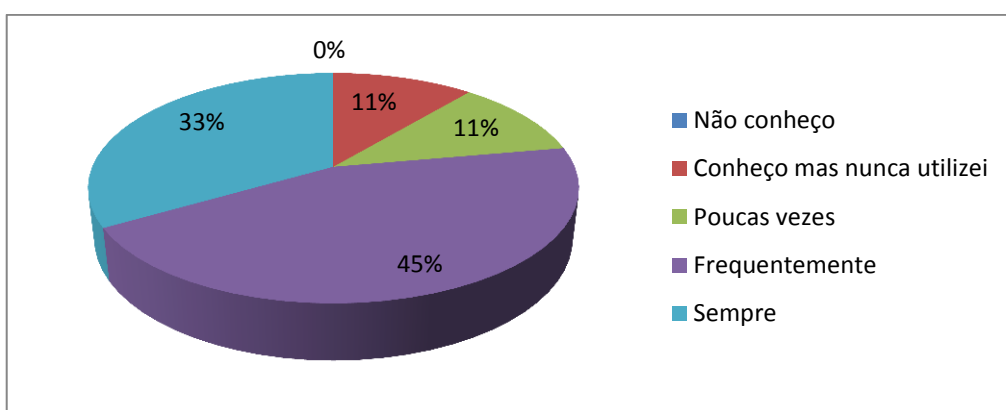


Gráfico 11 – Frequências do uso de Análise de documentos

Fonte: Pesquisa direta (2012)

A prototipagem, técnica que visa à construção de uma versão inicial do sistema sem muitas funções, é conhecida por 95% dos pesquisados, mas é pouco utilizada pelos desenvolvedores de Vitória da Conquista. No Gráfico 12

observar-se que 39% conhecem, mas nunca utilizam; 39% usam poucas vezes; 11% empregam frequentemente e 6% usam sempre.

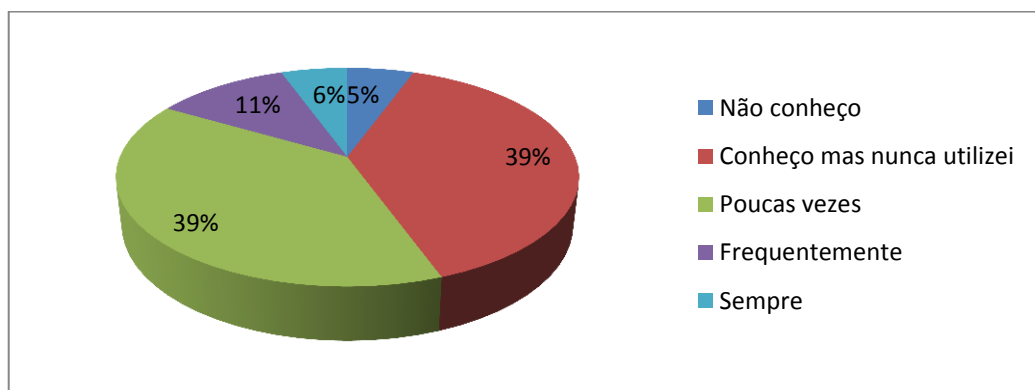


Gráfico 12 – Frequência do uso de Protótipos

Fonte: Pesquisa direta (2012)

Uma das técnicas pesquisadas que obteve o maior percentual de desconhecimento entre os desenvolvedores foi a *Group storytelling*. Essa técnica que visa a contagem de histórias, em grupo, por parte dos integrantes propiciando o compartilhamento de conhecimento é desconhecida por 83% dos pesquisados como mostra o Gráfico 13. Além disso, 11% declaram que conhecem, mas nunca utilizaram e que 6% usam poucas vezes. Com base no gráfico, observa-se que 94% desconhecem ou não empregam essa técnica em suas atividades de levantamento de requisitos.

Apesar de ser uma técnica com crescimento ascendente dentro do paradigma de entretenimento digital (POZZER, 2005), os resultados mostram que as empresas de software de Vitória da Conquista não compartilham desse conhecimento no seu processo de desenvolvimento de software.



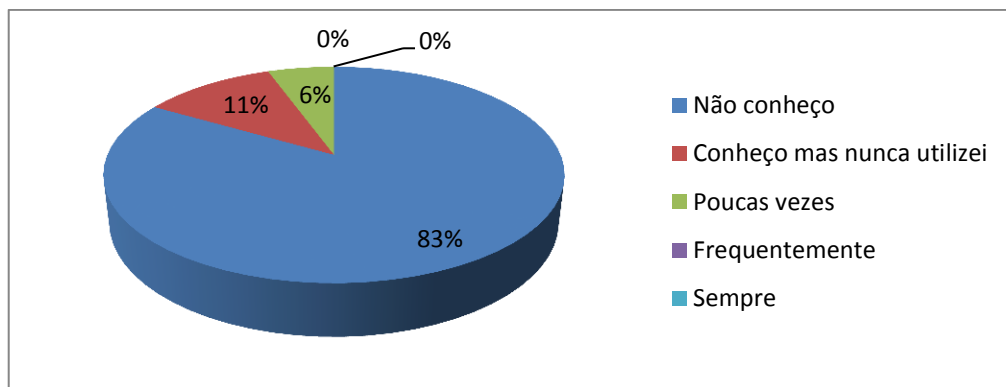


Gráfico 13 – Frequência do uso de *Group storytelling*

Fonte: Pesquisa direta (2012)

A frequência da reutilização de requisitos no processo de levantamento de requisitos, outra técnica observada na pesquisa, pode ser verificada no Gráfico 14. 33% dos desenvolvedores afirmam usarem a técnica frequentemente; 28% utilizam poucas vezes; 22% desconhecem a técnica e outros 17% dizem conhecer, mas nunca utilizam. Um dado relevante observado no gráfico é que a maioria dos pesquisados 61% aplicam essa técnica frequentemente ou ao menos em algumas ocasiões.

Esse resultado é justificado pelo fato de Sommerville (2007) afirmar que nos últimos anos há indício maior do uso da técnica, o que corresponde com os dados do gráfico.

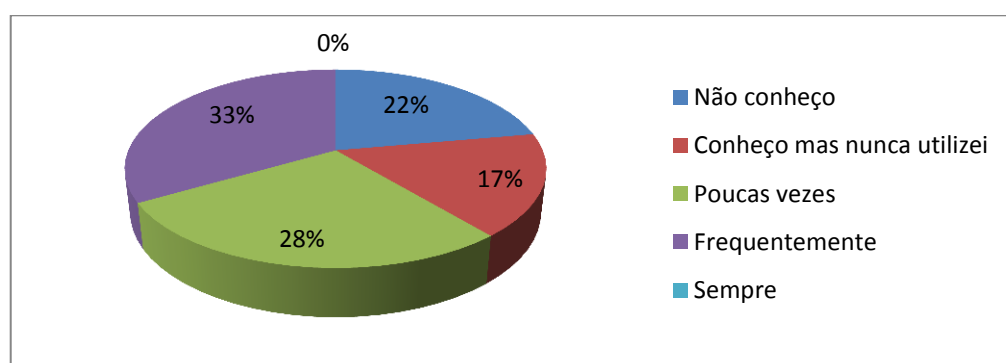


Gráfico 14 – Frequência da reutilização de requisitos

Fonte: Pesquisa direta (2012)

Outro aspecto interessante apresentado na pesquisa foi com relação à técnica Ponto de Vista. O Gráfico 15 mostra que 33% desconhecem essa técnica;

22% usam frequentemente; 22% utilizam poucas vezes; 17% conhecem, mas nunca utilizam e 6% usam sempre. O resultado demonstra que quase 1/3 dos pesquisados não sabem como essa técnica funciona e que 50% desconhecem ou conhecem, mas não utilizam. Isso leva a crer que os desenvolvedores podem em muitas ocasiões, não considerar os pontos de vistas de diversos *stakeholders* envolvidos com o desenvolvimento do sistema.

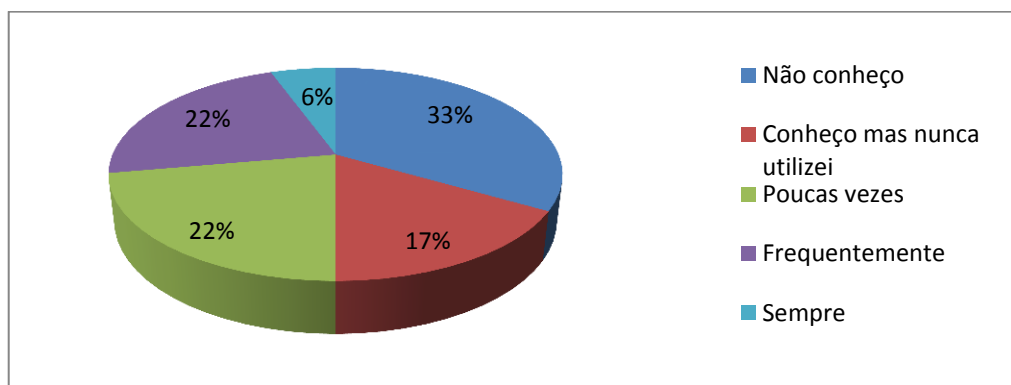


Gráfico 15 – Frequência do uso de Pontos de Vista

Fonte: Pesquisa direta (2012)

Um dos questionamentos feitos aos desenvolvedores e que tem um valor significativo para a pesquisa, é sobre que tipo de fatores influencia o desenvolvedor na escolha de uma determinada técnica de levantamento de requisitos. Dos itens apresentados aos pesquisados oito deles obtiveram um destaque maior, sendo considerado por mais de 70% dos pesquisados como muito importante ou importante. Observando o Gráfico 16 os fatores que mais influenciam os desenvolvedores são: familiaridade, uso anterior com sucesso, rapidez, custo, flexibilidade, utilidade, facilidade de uso e preferência da organização. Vale ressaltar que os itens rapidez, custo e utilidade são considerados por mais de 90% dos desenvolvedores como muito importante ou importante.

Esses dados podem ser correlacionados com os do uso frequente das técnicas entrevistas, diagramas, reuniões em grupo e análise de documentos, as quais se podem observar nos Gráficos 3, 5, 9 e 11, respectivamente. O que se nota é que os oito fatores mencionados influenciam diretamente na escolha

dessas técnicas, já que a maioria dos pesquisados a usam frequentemente ou sempre.

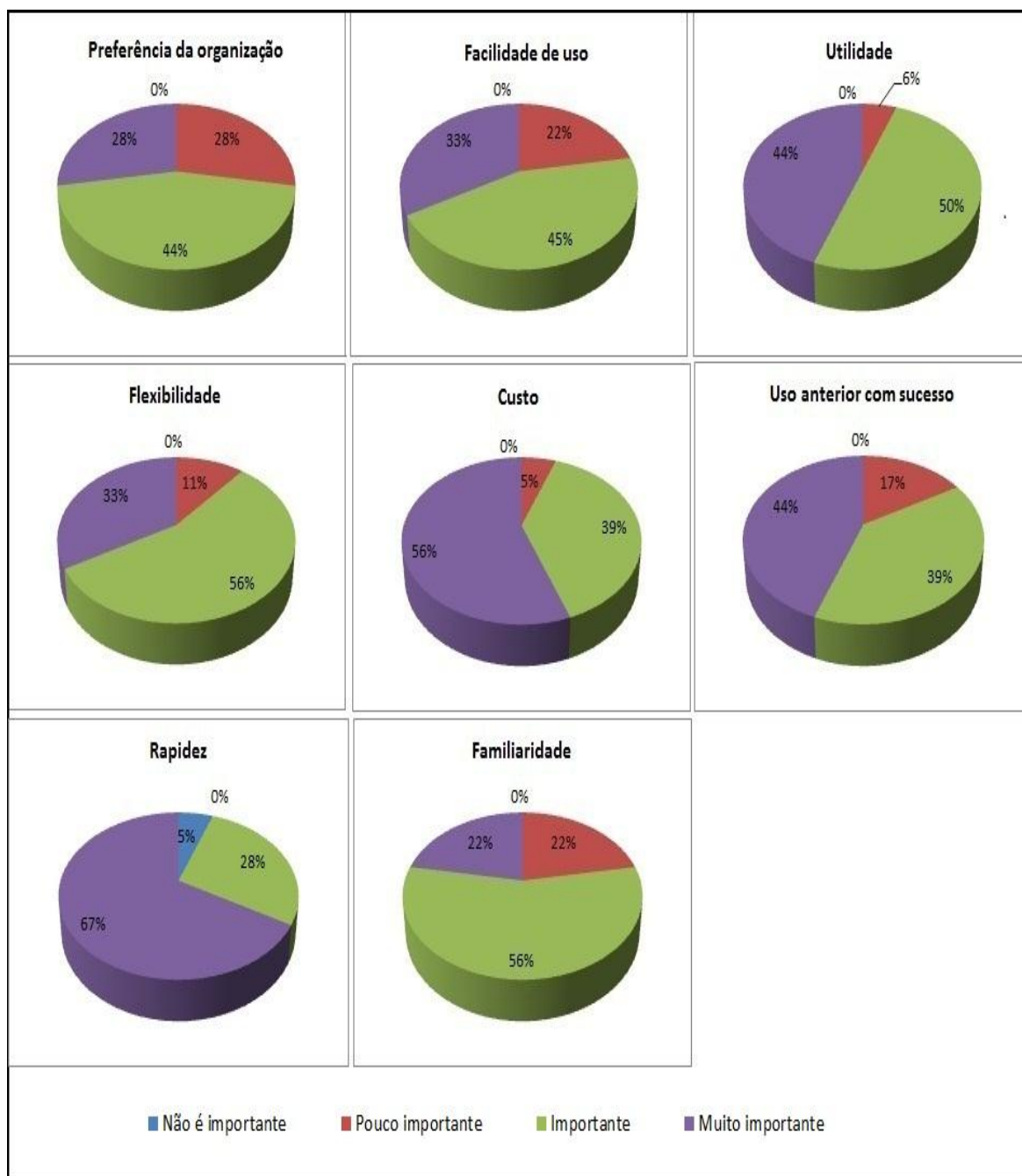


Gráfico 16 - Nível de importância na escolha de uma técnica de levantamento de requisitos

Fonte: Pesquisa direta (2012)

Um detalhe a salientar é que um dos fatores na escolha de uma determinada técnica obteve um resultado dividido entre os desenvolvedores. O fator,

preferência do cliente, foi considerado por 50% dos entrevistados como muito importante ou importante e 50% consideraram como pouco importante ou que possui nenhuma importância, como pode ser visto no Gráfico 17.

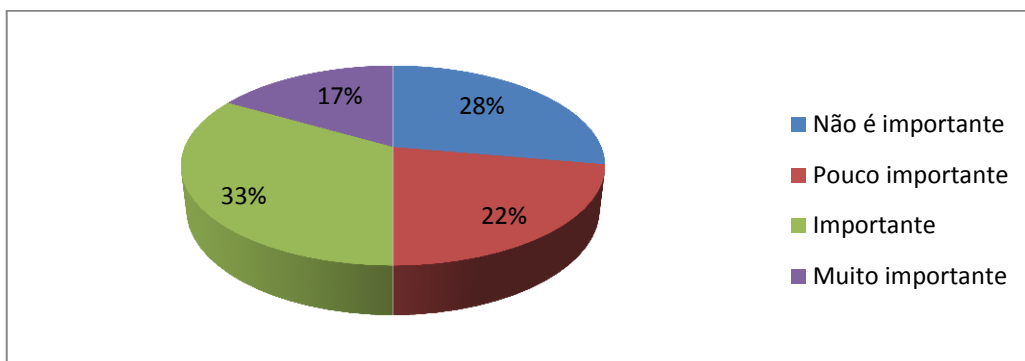


Gráfico 17 – Nível de importância do fator, preferência do cliente, na escolha da técnica pelo desenvolvedor.

Fonte: Pesquisa direta (2012)

Entretanto, essa divisão de opiniões entre os pesquisados, pode ser explicada pelo Gráfico 18, o qual demonstra que 72% dos desenvolvedores concordam que o usuário tem dificuldade de visualizar os requisitos dos sistemas nas suas atividades e que apenas 28% discordam desse fato. O que se pode inferir desses dados é que apesar da divisão de opiniões em que a preferência do cliente influencia na escolha da técnica pelo desenvolvedor, há uma tendência dos desenvolvedores escolherem a técnica a ser utilizada, pelo fato da dificuldade dos usuários em expressar o que realmente desejam.

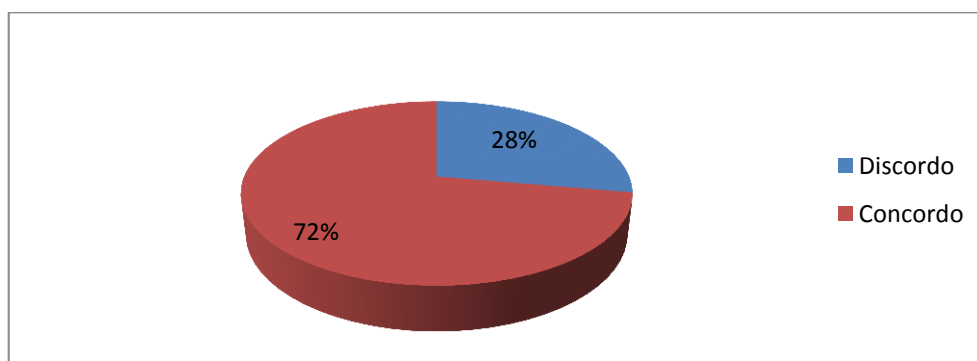


Gráfico 18 - Dificuldade dos usuários de visualizar os requisitos do sistema nas suas atividades

Fonte: Pesquisa direta (2012)

### 5.3. FERRAMENTAS UTILIZADAS NO PROCESSO DE LEVANTAMENTO DE REQUISITOS

Algumas ferramentas que auxiliam os desenvolvedores no levantamento de requisitos foram questionadas aos desenvolvedores quanto a sua frequência de utilização. Os resultados mais relevantes serão mostrados nesta seção.

A ferramenta que teve o maior percentual de utilização entre os pesquisados foi o processador de texto. Observar-se no Gráfico 19 que 39% dizem usar sempre; 33% usam frequentemente; 17% utilizam às vezes, 6% usam raramente e 5% nunca utilizaram.

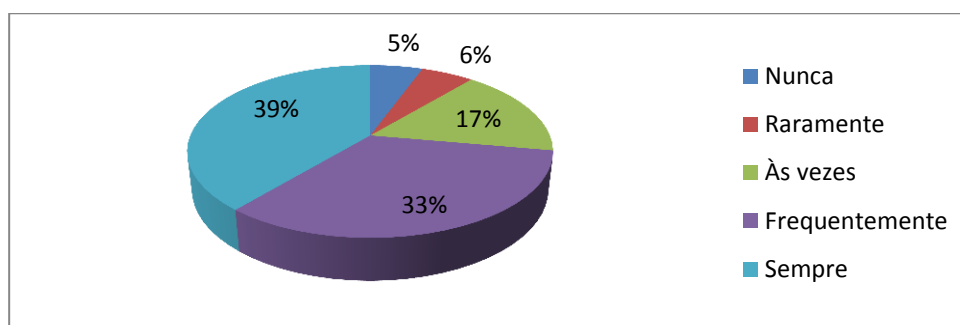


Gráfico 19 – Uso dos processadores de texto no levantamento de requisitos

Fonte: Pesquisa direta (2012)

De acordo com o Gráfico 20, as planilhas eletrônicas são utilizadas por uma grande parcela dos desenvolvedores apenas em algumas ocasiões. Os dados do gráfico mostram que 44% usam às vezes; 22% utilizam sempre; 22% empregam frequentemente; 6% usam raramente e 6% não utilizam.

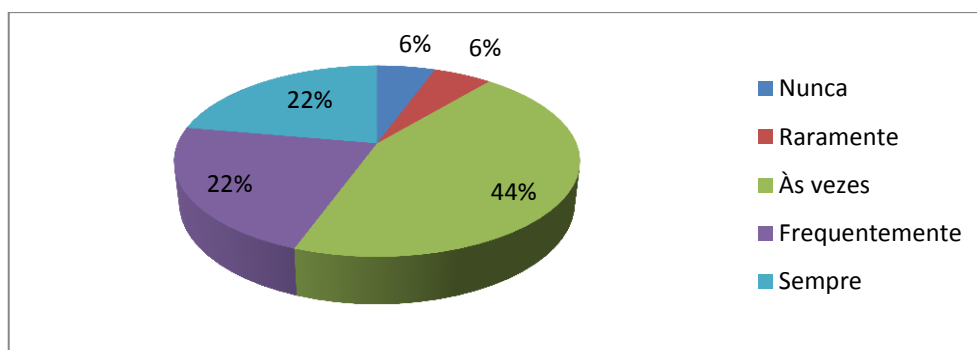


Gráfico 20 – Uso das planilhas eletrônicas no levantamento de requisitos

Fonte: Pesquisa direta (2012)

O uso de *templates* de documentos como ferramenta de auxílio na busca por requisitos de software não é muito utilizada pela maioria dos pesquisados. No Gráfico 21 nota-se que 56% nunca usaram a ferramenta; 33% dizem utilizar raramente; 6% frequentemente e 5% às vezes. Esse resultado pode ser explicado pelo fato dos *stakeholders* envolvidos no projeto preferirem validar os requisitos de software através de diagramas ou protótipos, como mostra o Gráfico 25.

Também podemos justificar o não uso desse tipo de ferramenta, basicamente textual, pelo fato de fazer parte da documentação do software e os desenvolvedores terem uma grande dificuldade em fazer esse tipo de prática, devido à política organizacional da empresa, pelos prazos curtos e pela inexistência dessa ferramenta na organização (NUNES, 2005).

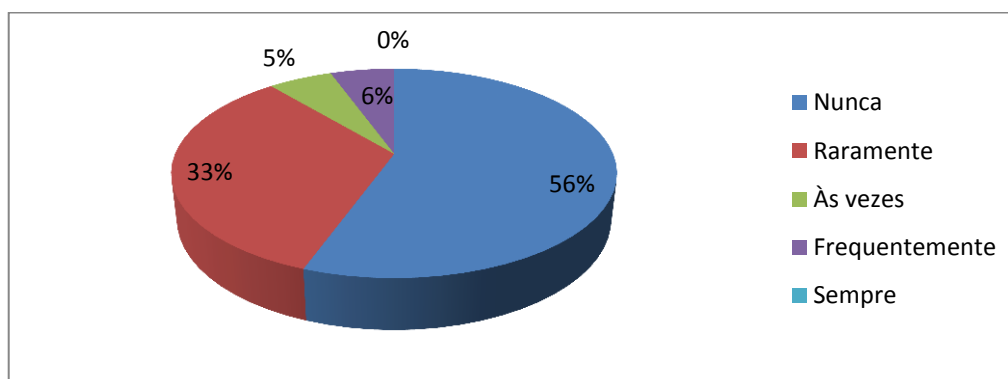


Gráfico - 21 – Uso dos *templates* de documentos no levantamento de requisitos

Fonte: Pesquisa direta (2012)

As ferramentas de modelagem se mostram como uma das mais utilizadas pelos desenvolvedores. Os dados coletados demonstram, no Gráfico 22, que 33% utilizam frequentemente; 17% usam sempre; 17% empregam às vezes; 17% usam raramente e 16% nunca utilizaram. Isso mostra que 50% dos desenvolvedores empregam esse tipo de ferramenta como auxílio no levantamento de requisitos. Observando os Gráficos 5 e 6 nota-se o porquê do uso desse tipo de ferramenta, pois a maioria dos desenvolvedores aplicam os diagramas, como exemplo os Casos de Usos, na obtenção de requisitos, os quais possuem ferramentas para auxiliar na sua criação.

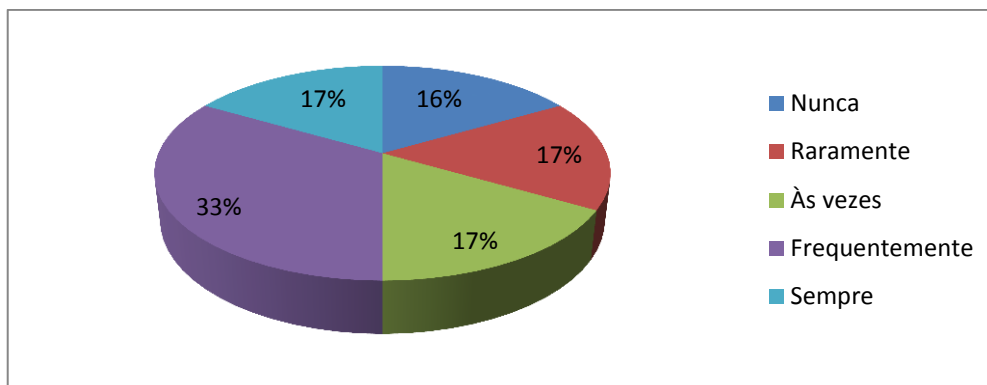


Gráfico 22 – Uso das ferramentas de modelagem no levantamento de requisitos

Fonte: Pesquisa direta (2012)

Em relação às ferramentas de gerenciamento de requisitos, o não emprego desse tipo de aplicação nas tarefas de levantamento de requisitos é algo significativo, como mostra o Gráfico 23. 56% dos desenvolvedores nunca utilizaram esse tipo de ferramenta; 28% usam às vezes; 11% empregam raramente e 5% utilizam frequentemente. O que se pode deduzir com relação a esses dados é que a maioria dos desenvolvedores pesquisados podem não utilizar um processo de desenvolvimento de software de maturidade organizacional como o CMM (*Capability Maturity Model*), por exemplo, que prega por melhores práticas no desenvolvimento de software e a implantação de gerenciamento de requisitos é uma das primeiras etapas para alcançar a maturidade organizacional (CARVALHO, 2001).

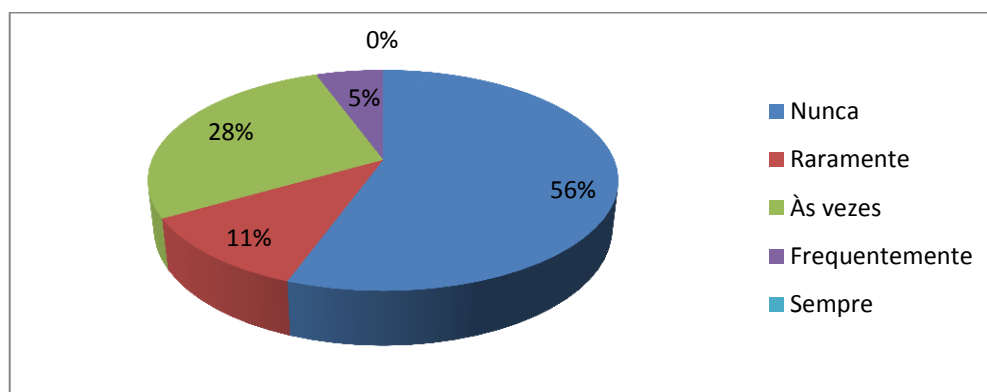


Gráfico 23 – Uso das ferramentas de gerenciamento de requisitos

Fonte: Pesquisa direta (2012)

Durante a pesquisa verificou-se a prática do uso de quadros e *flip charts* no levantamento de requisitos, e notou-se que 78% dos desenvolvedores nunca utilizaram esse tipo de auxílio, como mostra o Gráfico 24. Outro dado significativo é com relação às gravações de áudio e vídeo, as quais não são utilizadas por 67% dos desenvolvedores demonstrada no Gráfico 24. Pode-se inferir desses gráficos que apesar das reuniões em grupo ser um método frequente na obtenção de requisitos, dados apresentados pelo Gráfico 5, esses instrumentos não são comumente utilizados durante essas reuniões.

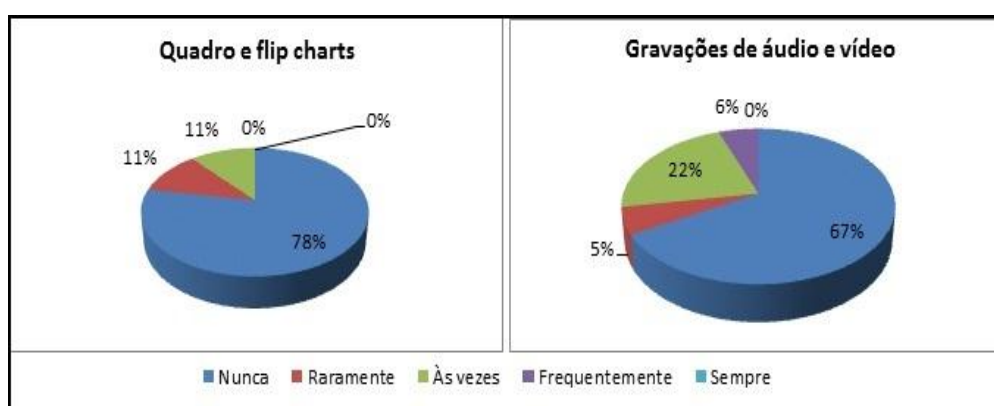


Gráfico 24 – Uso de quadros, *flip charts*, gravações de áudio e vídeo no levantamento de requisitos

Fonte: Pesquisa direta (2012)

#### 5.4. VALIDAÇÃO DOS REQUISITOS APÓS A PRIMEIRA RODADA DE LEVANTAMENTO DE REQUISITOS

O Gráfico 25 mostra como é feito a validação dos requisitos identificados, junto aos *stakeholders*, após a primeira fase de busca de requisitos. Nessa questão os pesquisados poderiam escolher mais de uma opção como meio de validação. Os resultados mais significativos mostram que nove pesquisados utilizam protótipos desenvolvidos pela equipe técnica para validação, o que representa 50% dos desenvolvedores; seis validam através de diagramas, representando 33,3% e cinco avaliam os documentos, como listas de requisitos fornecidos pela equipe técnica, representando 27,8% dos pesquisados.



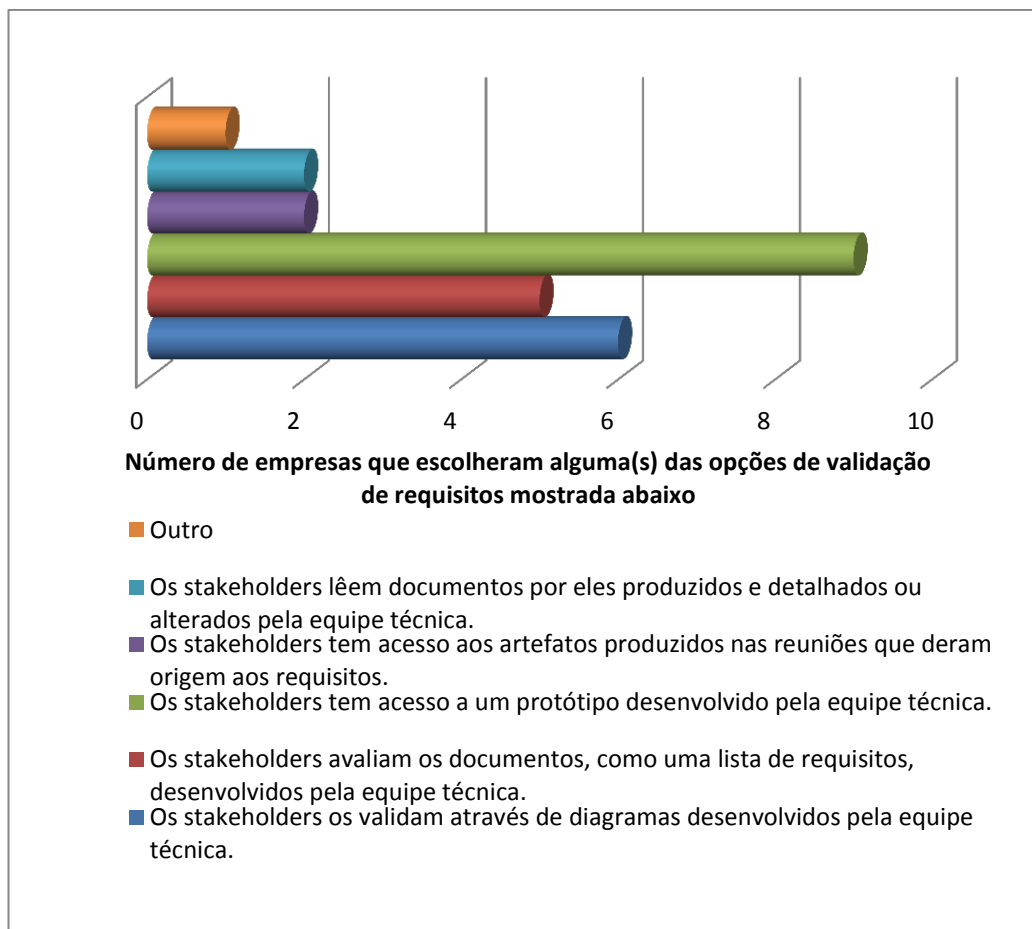


Gráfico 25 – Como e quantas empresas validam requisitos identificados com os *stakeholders* após a primeira rodada de levantamento de requisitos.

Fonte: Pesquisa direta (2012)

Fazendo uma análise do Gráfico 25 e correlacionando com o Gráfico 12 se pode notar uma divergência de dados quanto ao aspecto da frequência da utilização da técnica prototipagem e uso dela para validar requisitos. Os resultados mostram que de acordo com o Gráfico 12, poucas vezes os desenvolvedores usam essa técnica para levantamento de requisitos. Entretanto, 50% dos mesmos dizem validar os seus primeiros requisitos com o auxílio da prototipagem junto aos *stakeholders*. O que se pode deduzir desses dois fatos é que a prototipagem é utilizada de forma arbitrária, ou seja, não obedece aos seus conceitos e métodos específicos, a qual é utilizada por muitos desenvolvedores, não como um tipo técnica apenas, mas como artifício para mostrar os requisitos implementados para os clientes poderem validá-los.

Diferentemente dos protótipos o uso de diagramas e análise de documentos, Gráficos 5 e 11 seguem a mesma lógica do Gráfico 25 em que a maioria dos desenvolvedores utilizam essas técnicas frequentemente para levantar requisitos e muitos deles, também, usam para validar os requisitos encontrados junto aos *stakeholders*.

Outro aspecto a ser levado em consideração é sobre o usuário não ter paciência em validar os requisitos e não entenderem de forma clara o seu significado. O Gráfico 26 mostra que 83% dos desenvolvedores concordam com essa afirmação e 17% discordam.

Fazendo uma correlação com o Gráfico 25 se pode deduzir que com a dificuldade do usuário de ter paciência de realizar, entender e validar requisitos, os desenvolvedores tendem a usar os protótipos como meio para facilitar esse processo.

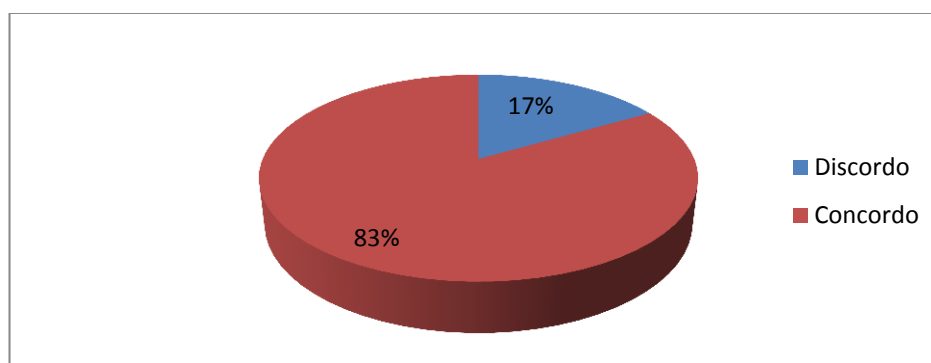


Gráfico 26 – Dificuldade do usuário de realizar e entender a validação de requisitos.

Fonte: Pesquisa direta (2012)

## 5.5. FORMALIDADE NO LEVANTAMENTO DE REQUISITOS

Aspectos sobre a formalidade entre o analista de desenvolvimento e o cliente foram abordados na pesquisa. Analisando os dados do Gráfico 27, 78% dos desenvolvedores concordam que quanto maior for o formalismo da documentação, maior será a dificuldade em obter o entendimento do cliente e 22% discordam dessa afirmação. Outro questionamento feito foi quanto a menor formalidade do analista com o cliente, mais difícil será a sua compreensão. 61% discordaram dessa afirmação e 39% concordaram como mostra o Gráfico 28.

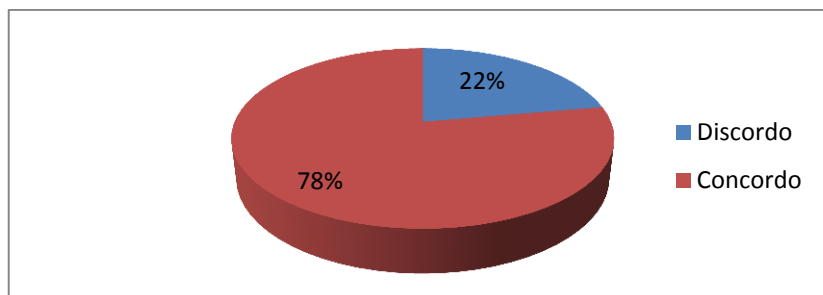


Gráfico 27 – Maior formalismo da documentação, maior dificuldade do cliente compreender.

Fonte: Pesquisa direta (2012)

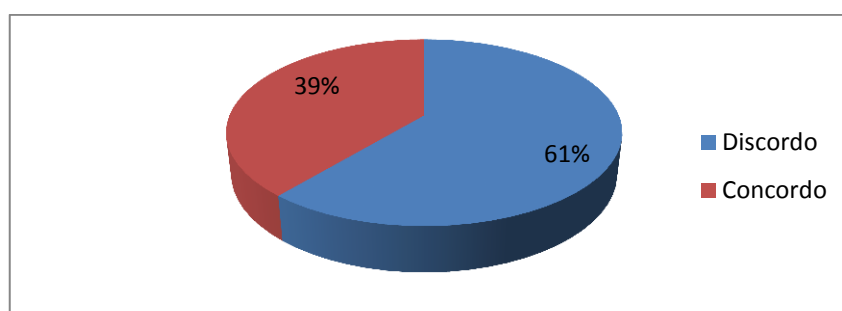


Gráfico 28 – Menor formalidade, maior dificuldade de entendimento do analista.

Fonte: Pesquisa direta (2012)

Observando o resultado dos dois gráficos pode-se inferir que a formalidade prejudica essencialmente o cliente, já que ele não possui conhecimento técnico formal para interagir com o desenvolvedor, ficando a critério do mesmo usar uma linguagem mais cotidiana tanto na documentação quanto no relacionamento com o cliente.

## 5.6. DIFICULDADES NO LEVANTAMENTO DE REQUISITOS

Um dos questionamentos feitos aos desenvolvedores foi sobre as principais dificuldades encontradas no levantamento de requisitos. Eles tinham a opção de escolher mais de um tipo de dificuldade encontrada ou citar outra que não estava dentro das opções. Os resultados mais significativos podem ser observados no Gráfico 29, o qual mostra que dezessete dos desenvolvedores escolheram que a maior complexidade no levantamento de requisitos está no fato do cliente ter dificuldade de expressar o que realmente deseja. Isso representa em

percentagem em 94,4% dos pesquisados que escolheram essa alternativa. Onze desenvolvedores creditam a dificuldade às mudanças nos requisitos, representando 61,1% como opção escolhida. Dificuldades como validar o que o cliente solicitou ou o analista ter que exercer diversas funções durante essa atividade foi escolhida por sete desenvolvedores, representando 38,8% dos pesquisados, os quais marcaram essa opção. Outro item representativo escolhido por seis desenvolvedores foi a existência de muitos *stakeholders* com interesses distintos, o que ocasiona numa maior complexidade no levantamento de requisitos, e representa 33,3% dos pesquisados que assinalaram essa alternativa.

É possível notar, que a maior complexidade do levantamento de requisitos por parte dos desenvolvedores pesquisados é com relação ao cliente informar o que realmente ele necessita, para que o software seja desenvolvido. Isso pode ser resultado do uso de formalismo dos desenvolvedores na interação com o cliente, o que dificulta a exposição da informação pelo cliente, como mostra o Gráfico 27. Ou se pode inferir que são utilizadas poucas técnicas de levantamentos de requisitos apropriadas para cada tipo de cliente ou pelo alto índice de desconhecimento de determinada técnica para diversos tipos de situações com o cliente, como mostra os Gráficos 4, 7, 8, 10, 12, 13 e 15.

Outra questão a ser levantada é com a dificuldade que os *stakeholders* encontram com as mudanças dos requisitos do software, alternativa escolhida por 61,1%. Esse fato pode ser explicado observando o Gráfico 18, o qual mostra que 72% dos pesquisados dizem que os clientes têm dificuldade de visualizar requisitos nas suas atividades. Isso pode ocasionar em mudanças de requisitos durante o levantamento de requisitos, como demonstra o Gráfico 29.

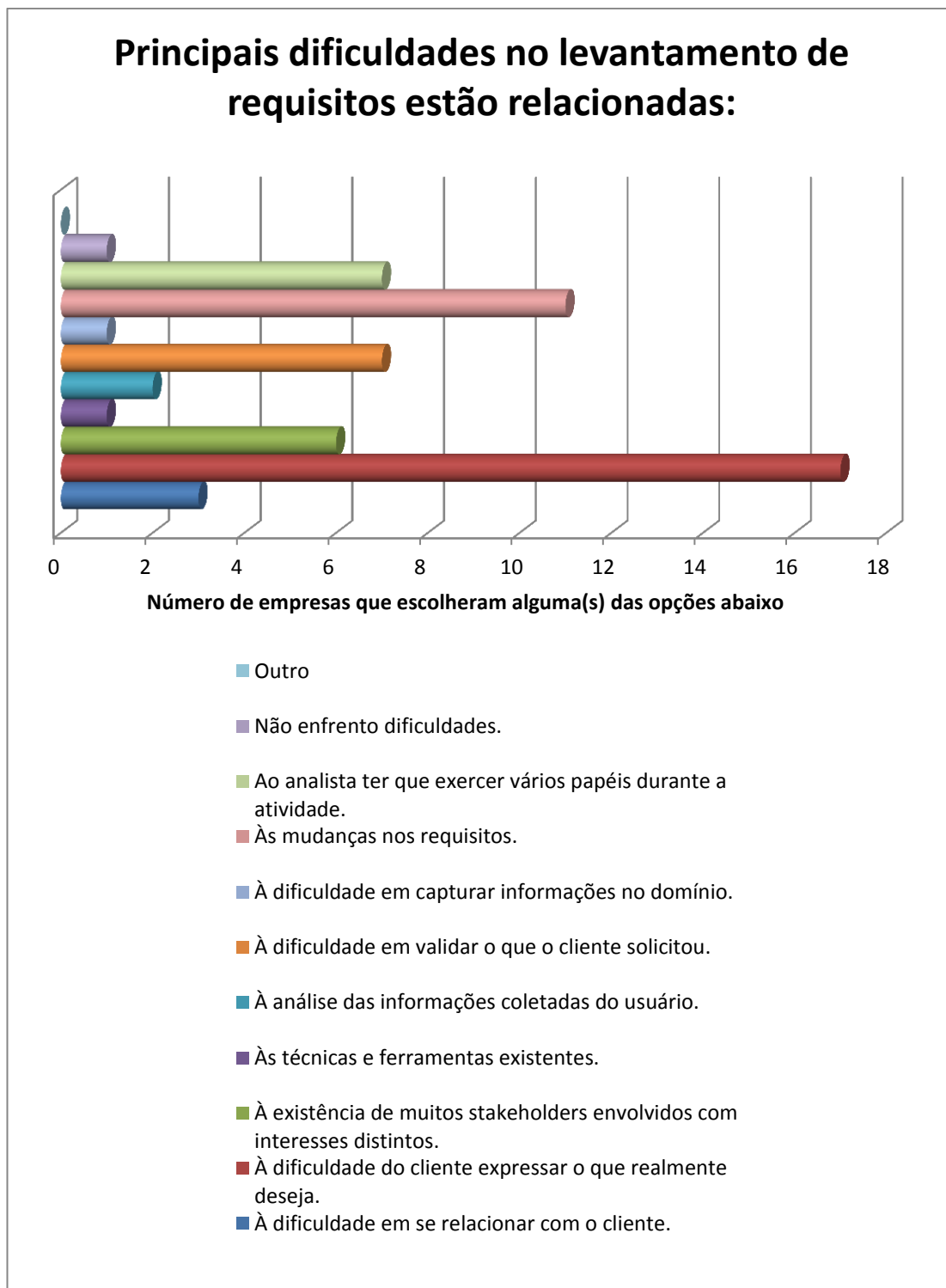


Gráfico 29 – Principais dificuldades dos desenvolvedores no levantamento de requisitos.

Fonte: Pesquisa direta (2012)

## 6. CONCLUSÕES

Primeiramente, a pesquisa teve como enfoque a relação que o desenvolvedor tem com essa atividade de levantamento de requisitos. Fatores como experiência na área e participação em projetos foram questionados aos pesquisados. Os resultados mostraram que a maioria dos desenvolvedores possui uma boa experiência na atividade de levantamento de requisitos. Entretanto, apesar da experiência, grande parte dos pesquisados não participaram de muitos projetos durante a elicitaco de requisitos.

Com relao à frequncia de utilizao de tcnicas de levantamento de requisitos ficou evidente que algumas delas so de uso frequente por mais de 60% dos desenvolvedores, como entrevistas, diagramas, reunies em grupo e anlise de documentos. Outras possuem um desconhecimento ou so conhecidas, mas no utilizadas por um grande percentual de pesquisados, principalmente aquelas que usam tcnicas baseadas em narraes dos usurios, como, por exemplo, a *Group Storytelling* que obteve uma percentual de 83% de desconhecimento pelos desenvolvedores.

Outro detalhe a salientar com relao às reunies em grupo, apesar de ser muito utilizada pelos desenvolvedores, a tcnica *Brainstorming*, um tipo de reunio em grupo com metodologia especfica  desconhecida ou no utilizada por 67% dos pesquisados. Ferramentas que auxiliariam a *Brainstorming* como quadros e *flip charts* na interao com os *stakeholders* so minimamente usadas pelos desenvolvedores. O que se deduz que essas reunies utilizam de pouca metodologia para o levantamento de requisitos entre os *stakeholders* participantes.

A prototipagem se mostrou como uma tcnica pouca aplicada pelos desenvolvedores. Entretanto, ela  o meio mais empregado para validao dos requisitos entre os *stakeholders*, levando a crer que sua metodologia  mais utilizada como um facilitador para a aprovao dos requisitos juntos aos *stakeholders* e no como uma tcnica e metodologia especfica, propriamente dita, para levantamento de requisitos.

Um dos pontos de destaque da pesquisa foi com relao aos fatores que levam os desenvolvedores a escolherem determinada tcnica. Mais de 70% dos

pesquisados acreditam que estes fatores como familiaridade, uso anterior com sucesso, rapidez, custo, flexibilidade, utilidade, facilidade de uso e preferência da organização são determinantes para a aplicabilidade de uma técnica.

Com relação ao uso efetivo de ferramentas de levantamento de requisitos, a pesquisa mostrou que as ferramentas tradicionais como processadores de texto e de modelagem são as mais utilizadas por uma boa parcela dos desenvolvedores. No entanto, ferramentas mais recentes como *templates* de documentos e gerência de requisitos nunca foram utilizadas por 56% dos pesquisados.

Aspectos relacionados ao formalismo na interação entre os clientes e desenvolvedores se mostraram prejudiciais ao entendimento do cliente em relação aos requisitos que o próprio objetiva ter no software. Isso é acarretado, também, pela grande dificuldade que o cliente tem de visualizar os requisitos nas suas atividades, como mostra a pesquisa.

A pesquisa proporcionou verificar quais motivos tornam a tarefa de levantamento de requisitos complexa. Mudanças de requisitos e a dificuldade do cliente expressar o que realmente deseja foram os itens mais citados, sendo que este último foi mencionado por 94,4% dos desenvolvedores. Conclui-se então que existe uma dificuldade no relacionamento entre os desenvolvedores e os clientes. Isso pode ser resultado da escolha errônea da técnica a ser aplicada na interação com o cliente por parte do desenvolvedor.

Por fim, a análise dessa atividade nas empresas de desenvolvimento de software de Vitória da Conquista, mostrou que os desenvolvedores aplicam técnicas e ferramentas de levantamento de requisitos mais comuns durante o processo de desenvolvimento de software. Entretanto, as técnicas e ferramentas que possibilitam uma maior integração entre os *stakeholders* envolvidos com o projeto do software, precisam ser mais bem estudadas pelos desenvolvedores para que haja uma maior aplicabilidade no levantamento de requisitos, o que poderia resultar na minimização das dificuldades implícitas que existem nessa atividade.

## 6.1. TRABALHOS FUTUROS

Uma sugestão para trabalhos futuros seria um estudo de uma metodologia de levantamento de requisitos voltada para os clientes de software da região, a qual enfatizasse como escolher a melhor técnica para cada tipo de cliente, o que poderia diminuir a dificuldade do desenvolvedor em entender o que cliente necessita.

Outra sugestão seria uma análise das ferramentas que auxiliam na elicitacão de requisitos, observando quais delas são consideradas melhores para serem aplicadas na atividade de levantamento de requisitos.



## REFERÊNCIAS

BARBOSA, Glívia et al. **Um processo de elicitação de requisitos com foco na seleção da técnica de elicitação.** Encontrado em <http://www.lbd.dcc.ufmg.br/colecoes/sbqs/2009/014.pdf>. Acesso em 16/03/2012.

BATISTA, Edinelson A.; CARVALHO, Ariadne M. B. R. **Uma Taxonomia Facetada para Técnicas de Elicitação de Requisitos.** Encontrado em [http://www.inf.pucrio.br/wer/WERpapers/artigos/artigos\\_WER03/edinelson\\_batista.pdf](http://www.inf.pucrio.br/wer/WERpapers/artigos/artigos_WER03/edinelson_batista.pdf). Acesso em 01/02/2012.

BELGAMO, Anderson. **Estudo Comparativo sobre as Técnicas de Elicitação de Requisitos do Software.** Encontrado em: <http://walterdominguez.info/contextoconteudo/tema/requisitosdesistema/texto/tecnicaselicita%C3%A7%C3%A3orequisitos.pdf>. Acesso em 31/01/2012.

BENTES, Luciana N. et al. Uma Análise Avaliativa de Ferramentas de Software Livre no Contexto da Implementação do Processo de Gerência de Requisitos do MPS.BR. Encontrado em [http://scholar.google.com.br/scholar?cluster=2110837029317871398&hl=pt-PT&lr=lang\\_pt&as\\_sdt=0,5](http://scholar.google.com.br/scholar?cluster=2110837029317871398&hl=pt-PT&lr=lang_pt&as_sdt=0,5). Acesso em 01/02/2012.

BEZERRA, Eduardo. **Princípios de Análise e Projeto de Sistemas com UML.** 2. ed.. Rio de Janeiro: Campus, 2006.

BRAGA, Fabrício P. **Técnicas de Levantamento de Requisitos.** Encontrado em <http://www.followscience.com/library/exacts-and-informatics/computer-science/tecnicas-de-levantamento-de-requisitos-175>. Acesso em 01/02/2012.

CARVALHO, Ana E. S. **Uma Estratégia para Implantação de uma Gerencia de Requisitos Visando a Melhoria Dos Processos de Software.** Encontrado em [http://wer.inf.puc-rio.br/WERpapers/artigos/artigos\\_wer01/carvalho.pdf](http://wer.inf.puc-rio.br/WERpapers/artigos/artigos_wer01/carvalho.pdf). Acesso em 03/03/2012.

DIESTE, O.; Lopez, M. e Ramos, F. **“Updating Systematic Review about Selection of Software Requirements Elicitation Techniques”.** In Workshop on Engenharia de Requisitos (WER 2008), Barcelona, Espanha.

ELLWANGER, Fábio. **O processo de engenharia de requisitos visando rastreabilidade através do uso de princípios de *workflow*.** Santa Cruz do Sul – RS, 2005. 127f.. Trabalho de conclusão de curso (Bacharelado em Ciência da Computação) – Departamento de Informática, Universidade de Santa Cruz do Sul.

EMANUEL. Astah Community, [S.I], 2010. Encontrado em: <http://emanasblog.blogspot.com.br/2010/09/astah-comunity.html>. Acesso em 30/03/2012.

FURLAN, José D. **“Modelagem de Objetos através da UML”.** Makron, SP, 1998.

GOGUEN, Joseph A. “**Techniques for Requirements Elicitation**” in Software Requirements Engineering, IEEE-CS Press, Second Edition, 1997, p.p.110 –122.

GONÇALVES, João C. de A. R. **Story Mining: Elicitação de Processos de Negócio a partir de Group Storytelling e Técnicas de Mineração de Texto**. Rio de Janeiro, 2010. 175f. Dissertação (Mestrado em Informática) – Centro de Ciências Exatas e Tecnologia, Universidade Federal do Estado do Rio de Janeiro.

HOLDINGS, Dunstan T. **Enterprise Software Requirements Management System**. Reino Unido, 2012. Encontrado em: <http://www.dthomas.co.uk/dtalm/products/borland-caliber.htm>. Acesso em 01/04/2012.

IBM. Use Case Management with Rational Rose and Rational RequisitePro, [S.I.], 2003. Encontrado em: <http://www.ibm.com/developerworks/rational/library/835.html>. Acesso em 01/04/2012.

IEEE Std. 830, **IEEE Guide to Software Requirements Specification**, The Institute of Electrical and Electronics Engineers, New York, EUA, (1984).

IEEE Std. **12333-1996 IEEE Guide for Developing System Requirements Specification**. The Institute of Electrical and Electronics Engineers. Piscataway, NJ. Guide, EUA, 1996.

ISO/IEC **9126 Standard for information technology: software products evaluation: quality characteristics and guidelines for their use**. Geneva, 1988.

KENDALL, K.E.; Kendall, J.E. **Systems Analysis and Design**, Prentice Hall: 1992.

KOTONYA, G. e SOMMERVILLE, I. **Requirements Engineering – Processes and Techniques**. John Willy & Sons, 1997.

KOURI, Márcia G. **Definição de requisitos para um sistema de monitoramento de veículos no transporte rodoviário de cargas**. São Paulo, 2007. 165f.. Dissertação (Mestrado em Engenharia Elétrica) – Escola Politécnica, Universidade de São Paulo.

LOPES, Leandro T. **Um Modelo de Processo de Engenharia de Requisitos para Ambientes de Desenvolvimento Distribuído de Software**. Porto Alegre, 2004. 142f.. Dissertação (Mestrado em Ciência da Computação) – Faculdade de Informática, Pontifícia Universidade Católica do Rio Grande do Sul.

MARTINS, Luiz E. G. **Uma Metodologia de Elicitação de Requisitos de Software Baseada na Teoria da Atividade**. Campinas – SP, 2001. 182f..

Dissertação (Doutorado em Engenharia Elétrica) – Faculdade de Engenharia Elétrica e Computação, Universidade Estadual de Campinas.

MIGON Lilian B.; SILVA JR., Luiz C. L. **De histórias a processos: Utilização da técnica de Group Storytelling para apoio à eliciação de processos de negócios.** Encontrado em <http://www.ic.unicamp.br/~beatriz/wbpm2007/35314.pdf>. Acesso em 01/02/2012.

NUNES, VANESSA B. **Integrando gerência de configuração de software, documentação e gerência de conhecimento em um ambiente de desenvolvimento de software.** Vitória – ES, 2005. 200f.. Dissertação (Mestrado em Informática) – Departamento de Informática, Universidade Federal do Espírito Santo.

PRESSMAN, Roger S. **Engenharia de Software.** 6. Ed.. São Paulo: McGrawHill, 2006.

POZZER, Cezar T. **Um Sistema para Geração, Interação e Visualização 3D de Histórias para TV Interativa.** 156f.. Rio de Janeiro – RJ, 2005. Dissertação (Doutorado em Informática) – Departamento de Informática, Pontifícia Católica do Rio de Janeiro.

SMITH, Aron. **OSRMT.** 2006. Encontrada em: [http://download.cnet.com/OSRMT/3000-2383\\_4-10552248.html](http://download.cnet.com/OSRMT/3000-2383_4-10552248.html). Acesso em 01/04/2012.

SOMMERVILLE, Ian. **Engenharia de Software.** Tradução: Selma Shin Shimizu Melnikoff, Reginaldo Arakaki, Edilson de Andrade Barbosa. 8. ed. São Paulo: Person Addison-Wesley, 2007.

SOUZA, Ramon H. **Acesso remoto a horários de ônibus: uma aplicação para dispositivos móveis utilizando J2ME e MIDP 2.0.** Florianópolis – SC, 2004. 76f.. Trabalho de conclusão de curso (Bacharelado em Ciência da Computação) – Departamento de Informática e Estatística, Universidade Federal de Santa Catarina.

THAYER, Richard; DORFMAN, Merlin. **System and Software Requirements Engineering** - Second Edition. Los Alamitos: IEEE Computer Society Press Tutorial, 2000. 528p

THEILACKER, P. **Estudo comparativo das ferramentas case requisitopro e caliberrm voltadas para engenharia de requisitos.** Joinville – SC, 2008. 127f.. Trabalho de conclusão de curso (Bacharelado em Sistemas de Informação) - Sociedade educacional de santa catarina, Instituto Superior Tupy.

TONIOLO, Cristiano M. **Reuso de requisitos para famílias de produtos em sistemas embarcados.** Encontrado em <http://www.ibirapuera.br/pesquisa/revista/vol1/capa.pdf#page=32>. Acesso em 02/02/2012.

ZAVE, Pamela. **Classification of Research Efforts in Requirements Engineering.** ACM Computing Surveys. v. 29, n. 4. Dev. 1997. p. 315-321.

## APÊNDICE – QUESTIONÁRIO

### Pesquisa Conclusão de Curso – Levantamento de Requisitos

Olá sou Anderson Figueira, graduando em Ciência da Computação/UESB. Eu e os meus orientadores, Maria Silva Santos Barbosa e Fabrício Sousa Pinto, estamos desenvolvendo uma pesquisa relacionada a atividade de levantamento de requisitos em Vitória da Conquista/BA. Para tal, precisamos realizar um diagnóstico e solicitamos a sua contribuição preenchendo o formulário a seguir. Esta pesquisa é anônima e este formulário não realiza qualquer identificação de quem a responde. São 11 perguntas que levam cerca de 8 minutos para serem respondidas. Obrigado pela colaboração!

**\*Obrigatório**

**1) Quantos anos de experiência possui com levantamento de requisitos? \***

- menos de 1
- 1 a 2
- 2 a 3
- 3 a 5
- mais de 5

**2) Em quantos projetos participou da etapa de levantamento de requisitos? \***

- 1 a 3
- 4 a 6
- 7 a 9
- 10 a 12
- mais de 12

**3) Com que frequência utiliza as seguintes técnicas de levantamento de requisitos? \***

	Não conheço	Conheço mas nunca utilizei	Poucas Vezes	Frequentemente	Sempre
Entrevistas	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Questionários	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Diagramas (Ex: UML)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Cenários (1)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Protótipos	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Brainstorming (2)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Reuniões em grupo	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Observação	( )	( )	( )	( )	( )
Casos de uso (UML)	( )	( )	( )	( )	( )
Análise de documentos	( )	( )	( )	( )	( )
Storytelling/Group storytelling (3)	( )	( )	( )	( )	( )
Use Stories (4)	( )	( )	( )	( )	( )
Reuso de requisitos	( )	( )	( )	( )	( )
Pontos de Vista	( )	( )	( )	( )	( )

**Legenda:**

- 1- Coleção de narrativas de situações do domínio para levantar informações.
- 2- Reunião de especialistas de diversos setores, sendo que cada componente tem a função de estimular ao outro a criação de ideias para a resolução do problema em questão.
- 3- Grupos de pessoas contam histórias sobre as informações que se deseja obter. Utiliza-se de uma linguagem mais cotidiana. É geralmente usada em um ambiente distribuído em que os integrantes podem utilizar ferramentas computacionais para compartilhar conhecimento.
- 4- Os clientes descrevem funções, as quais gostariam que o sistema realizasse. Serve para estimar o plano de entrega das versões do sistema e para que sejam criados testes de aceitação de software.
- 5- Visões diferentes das pessoas envolvidas no projeto do sistema.

**4) Com que frequência utiliza estas ferramentas durante o levantamento de requisitos? \***

	Nunca	Raramente	Às vezes	Frequentemente	Sempre
Processadores de texto (Ex: Word)	( )	( )	( )	( )	( )
Planilhas eletrônicas (Ex: Excel)	( )	( )	( )	( )	( )
Templates de documentos (Ex: IEEE)	( )	( )	( )	( )	( )
Ferramentas de gerenciamento de requisitos (Ex: RequisitePro)	( )	( )	( )	( )	( )
Ferramentas de	( )	( )	( )	( )	( )

modelagem (Ex: Jude)					
Quadros e flip charts	( )	( )	( )	( )	( )
Gravações de áudio e vídeo	( )	( )	( )	( )	( )

5) Qual o nível de importância destes fatores na sua escolha sobre a técnica a ser utilizada? \*

	Não é importante	Pouco importante	Importante	Muito importante
Familiaridade	( )	( )	( )	( )
Uso anterior com sucesso	( )	( )	( )	( )
Rapidez	( )	( )	( )	( )
Esforço	( )	( )	( )	( )
Custo	( )	( )	( )	( )
Flexibilidade	( )	( )	( )	( )
Utilidade	( )	( )	( )	( )
Facilidade de uso	( )	( )	( )	( )
Preferência do cliente	( )	( )	( )	( )
Preferência da organização	( )	( )	( )	( )

6) Após a primeira rodada de levantamento de requisitos como você valida os requisitos identificados com os stakeholders (1)? \*

- ( ) Os stakeholders os validam através de diagramas desenvolvidos pela equipe técnica.
- ( ) Os stakeholders avaliam documentos, como uma lista de requisitos, desenvolvidos pela equipe técnica.
- ( ) Os stakeholders tem acesso a um protótipo desenvolvido pela equipe técnica.
- ( ) Os stakeholders tem acesso aos artefatos produzidos nas reuniões que deram origem aos requisitos.
- ( ) Os stakeholders leem documentos por eles produzidos e detalhados ou alterados pela equipe técnica.
- ( ) Outra: \_\_\_\_\_

**Legenda:**

- 1- Termo utilizado na engenharia de software para designar as pessoas que interagem ou são afetados pelo sistema em desenvolvimento como clientes, usuários, gerentes de projeto e dentre outros.

**7) Quanto maior o formalismo da documentação maior a dificuldade em obter o entendimento do cliente. \***

	<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	
Discordo plenamente	( )	( )	( )	( )	Concordo plenamente

**8) Quanto menor o formalismo, mais difícil o entendimento do analista. \***

	<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	
Discordo plenamente	( )	( )	( )	( )	Concordo plenamente

**9) O usuário tem dificuldade de visualizar requisitos de sistema nas suas atividades.**

	<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	
Discordo plenamente	( )	( )	( )	( )	Concordo plenamente

**10) O usuário não tem paciência de realizar a validação dos requisitos e não tem o entendimento correto e completo do seu significado. \***

	<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	
Discordo plenamente	( )	( )	( )	( )	Concordo plenamente

**11) As principais dificuldades no levantamento de requisitos estão relacionadas:**

- ( ) À dificuldade em se relacionar com o cliente.
- ( ) À dificuldade do cliente expressar o que realmente deseja.
- ( ) À existência de muitos stakeholders envolvidos com interesses distintos.
- ( ) Às técnicas e ferramentas existentes.
- ( ) À análise das informações coletadas do usuário.
- ( ) À dificuldade em validar o que o cliente solicitou.
- ( ) À dificuldade em capturar informações de domínio.
- ( ) Às mudanças nos requisitos.
- ( ) Ao analista ter que exercer vários papéis durante a atividade.
- ( ) Não enfrento dificuldades.
- ( ) Outra: \_\_\_\_\_





