



Universidade de Brasília - UnB
Faculdade UnB Gama - FGA
Requisitos de Software

Relatório 1

Autor: Nicácio Arruda, Pedro Sales, Ruan Nawe, Sabryna de
Sousa

Orientador: (Prof. Elaine Venson)

Brasília, DF
2016



Nicácio Arruda, Pedro Sales, Ruan Nawe, Sabryna de Sousa

Relatório 1

Universidade de Brasília - UnB

Faculdade UnB Gama - FGA

Orientador: (Prof. Elaine Venson)

Brasília, DF

2016

Lista de ilustrações

Figura 1 – SAF	12
Figura 2 – Programa MPS.BR	13
Figura 3 – Níveis de maturidade MR-MPS	14

Lista de tabelas

Lista de abreviaturas e siglas

MA	Metodologia Ágil
ER	Engenharia de Requisitos
SAF	Scaled Agile Framework
CMMI	Capability Maturity Model - Integration
MPS.BR	Melhoria do Processo de Software Brasileiro
MR-MPS	Modelo de Referência
MA-MPS	Método de Avaliação
MN- MPS	Modelo de Negócio
GRE	Gerência de Requisitos
DRE	Desenvolvimento de Requisitos
TI	Tecnologia de Informação

Sumário

1	INTRODUÇÃO	9
1.1	Visão Geral do Relatório	9
1.2	Contexto do Cliente	9
2	ABORDAGEM DE ER	11
2.1	Scaled Agile Framework (SAF)	11
2.2	Modelo de Maturidade	13
2.2.1	MPS.BR	13
2.3	Mapeamento do Contexto vs Abordagens	16
3	JUSTIFICATIVA	17
3.1	Panorama Ágil	17
3.2	Alterações e mudanças no ágil	18
3.3	Desenvolvimento Ágil x RUP	18
4	PROCESSO DE ENGENHARIA DE REQUISITOS	21
5	ELICITAÇÃO DE REQUISITOS	23
5.1	Técnicas de elicitação de requisitos	23
5.1.1	Entrevista	24
6	GERENCIAMENTO DE REQUISITOS	25
7	PLANEJAMENTO DO PROJETO	27
8	FERRAMENTAS DE GESTÃO DE REQUISITOS	29
8.1	Análise de Ferramenta de Gestão de Requisitos	29
	Referências	31
	APÊNDICES	33
	APÊNDICE A – PRIMEIRO APÊNDICE	35
	APÊNDICE B – SEGUNDO APÊNDICE	37

ANEXOS	39
ANEXO A – PRIMEIRO ANEXO	41
ANEXO B – SEGUNDO ANEXO	43

1 Introdução

“Entender os requisitos de um problema está entre as tarefas mais difíceis enfrentadas por um engenheiro de software”.(PRESSMAN, 2006, p.126). Tendo em vista a afirmação acima, infere-se que é necessário que a engenharia de requisitos forneça mecanismos apropriados para entender as necessidades dos clientes, analisar viabilidade, especificar a solução sem ambiguidades, validar, gerenciar as necessidades conforme suas alterações. Com base nas observações acima, este relatório tem como objetivo descrever de forma detalhada todos os mecanismos utilizados para levantar, especificar e detalhar todos os requisitos de um software, bem como os mecanismos utilizados nas fases de especificar solução, modelar processo, validar e gerenciar os requisitos.

1.1 Visão Geral do Relatório

Este relatório abrange os seguintes tópicos:

- Contextualização do Cliente: Conhecer a empresa, entender o problema e Descreve o contexto de negócio.
- Abordagem de engenharia de requisitos e Justificativa: Esclarecimento da abordagem escolhida e os motivos que levaram a equipe a escolher a metodologia.
- Processo de engenharia de requisitos: Descreve o processo que será realizado para o levantamento e documentação dos requisitos.
- Técnicas de elicitação de requisitos: Descreve as técnicas de elicitação de requisitos que serão utilizadas no trabalho 2.
- Gerenciamento de requisitos: Descreve os processos que serão realizados para o gerenciamento de requisitos e as ferramentas utilizada para esse objetivo.
- Planejamento do Projeto: Descreve o planejamento das atividades e as ferramentas

1.2 Contexto do Cliente

A empresa Eletrojun possui um sistema de compartilhamento de projetos como ilustrado no apêndice ” ”, esse sistema tem objetivo de reunir diversos estudantes no intuito de compartilhar projetos e ideias. O software desenvolvido pela Eletrojun está incompleto, há diversas funcionalidades que não estão implementadas, além de erros que devem ser corrigidos. Esse sistema tem um papel muito importante para a empresa, que é

o papel de unir todos os estudantes da Universidade de Brasília - Campus FGA, no intuito de desenvolver, criar, melhorar e divulgar projetos. A empresa espera que esse software possa ser o canal de comunicação entre os estudantes, de modo que, projetos possam ser desenvolvidos com o auxílio desse sistema. Esse site tem funcionalidades para gestão de outros projetos e necessita de uma ferramenta para compartilhamento de arquivos, esse sistema deve permitir que o usuário compartilhe ideias com outros usuários. Atualmente a Eletrojun não possui uma ferramenta para o gerenciamento de ideias, e sofre com diversos problemas administrativos, no que tange ao compartilhamento dessas ideias. A estrutura da Eletrojun é vertical, e segue uma hierarquia bem definida, esse tipo de estrutura pode ser favorável ao bom andamento do projeto, visto que, a equipe de Analista de Requisitos terá que se reunir apenas com diretores da empresa.

2 Abordagem de ER

A necessidade da busca de métodos que poderiam vir a ajudar na evolução do processo de construção de software, culminaram com o desenvolvimento da Engenharia de Software, disciplina que traz consigo métodos e técnicas que ajudam na administração da complexidade inerente do software. Contudo, apesar de inegáveis avanços, ainda existem pontos obscuros, sendo certamente requisitos de software um destes pontos. No ano de 1979, o Government Accounting Office (GAO) nos USA publicou um relatório no qual afirmava que menos de 20% da eficiência no alcance da satisfação dos clientes referentes ao produto final de software, resultaram no surgimento, desenvolvimento e definição de processos de desenvolvimento de software, com o objetivo de aumentar a produtividade e diminuir os riscos de um projeto(3). Dentre estes processos, encontram-se o processo cascata, o interativo, o espiral, o incremental, entre outros, que podem ser aplicados às soluções que melhor se encaixam em seus princípios e contextos. Para o presente projeto, foi proposta uma solução baseada nas Metodologias Ágeis (MAs): Abordagem que tem como objetivo principal garantir a entrega de produtos em um menor prazo, com maior qualidade e satisfazendo às necessidades dos clientes através da aplicação da produção enxuta para desenvolvimento de software, atacando os desafios emergentes de contextos dinâmicos (4). Baseado na análise do processo, foi determinada a utilização do processo SAF para o presente projeto, que será sucintamente descrito subsequentemente.

2.1 Scaled Agile Framework (SAF)

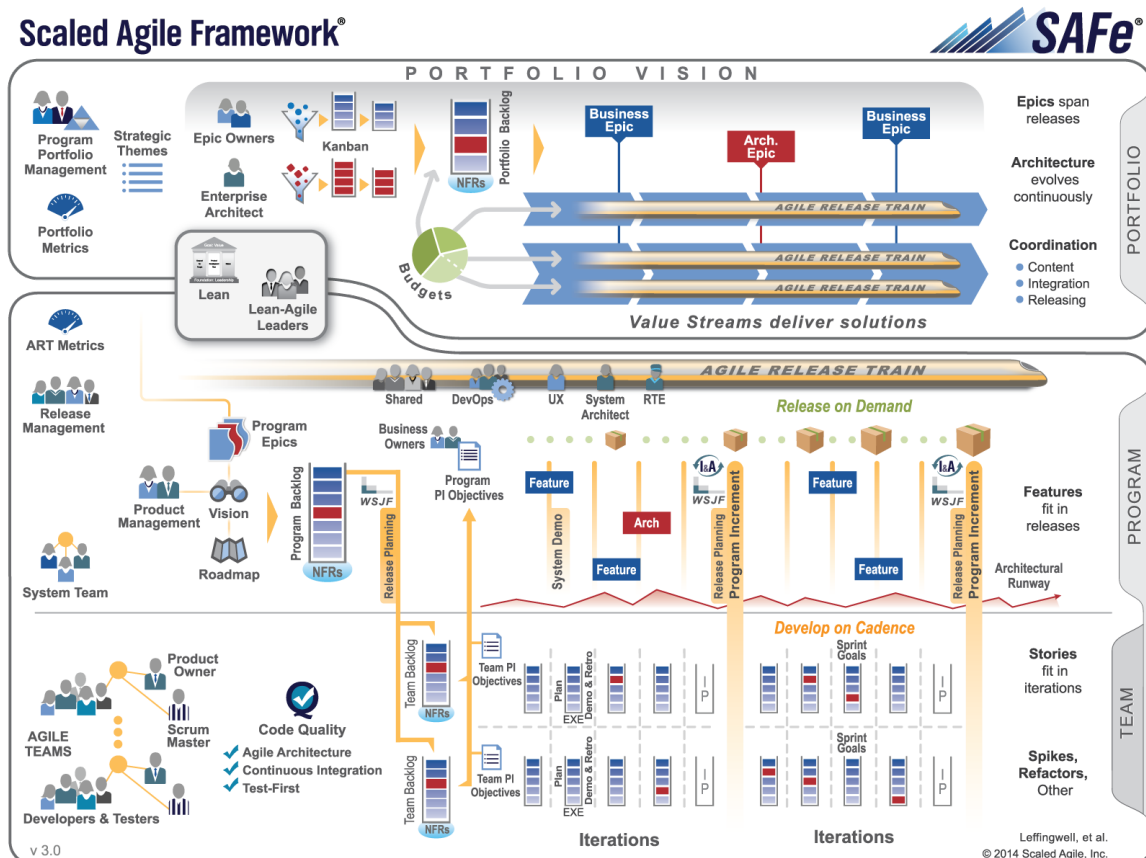
Tendo sua origem nas metodologias e filosofias ágeis, o SAF é um processo criado inicialmente para se adaptar ao contexto empresarial, trazendo a possibilidade de realizar entregas rápidas e com proximidade do cliente, alinhando constantemente com a expectativa do mesmo e adaptando constantemente o projeto às mudanças de requisitos do cliente. O SAF tem foco no produto em detrimento da documentação e nas pessoas envolvidas no processo, aproximando estas e trazendo o cliente como membro efetivo da equipe. Este tem papel fundamental na tomada de decisões no processo do desenvolvimento do projeto. Tais práticas corroboram uma maior adaptatividade às mudanças, tendendo a amenizar riscos no processo de execução do projeto, principalmente pela proximidade entre equipe e cliente (BOEHM, 2002) (1) O SAF é baseado nos princípios Ágeis, que se baseiam nos nove princípios seguintes:

- Adotar uma visão sistêmica
- Entendemos a economia da cadeia de valor

- Desenvolvemos sistemas iterativamente e incrementalmente
- Integramos e testamos frequentemente; adaptamos imediatamente
- Gerenciamos riscos e a eficácia através de ciclos de aprendizado rápido e síncrono
- Facilitamos o fluxo: limitamos o trabalho em execução, reduzimos o tamanho do lote e gerenciamos a duração da fila
- Sincronizamos entre os domínios o planejamento e a colaboração
- Baseamos marcos na avaliação objetiva dos sistemas de trabalho
- Desbloqueamos a motivação intrínseca dos trabalhadores do conhecimento
- Descentralizamos a tomada de decisão

A natureza particular dessa proposta é a ênfase que o framework dá ao gerenciamento de requisitos, trazendo novos conceitos nesta área, como histórias (user stories), temas (themes) e épicos (epics). Conceitos que ampliaram a forma de capturar requisitos, melhor adequados à ênfase em colaboração, flexibilidade e iterações mais curtas no projeto. (5)

Figura 1 – SAF



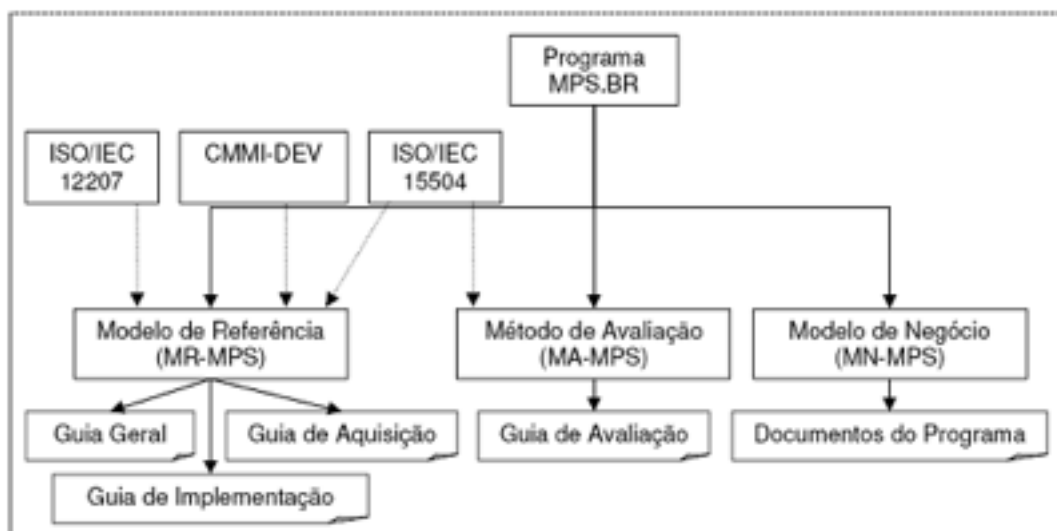
2.2 Modelo de Maturidade

O modelo de maturidade foi desenvolvido para auxiliar a execução de atividades que envolvam processos. Ele funciona como um guia em que a organização deve-se espelhar, realizar um plano, a fim de chegar em um nível de excelência. A exigência da contínua melhora dos projetos é o que torna como necessidade este gerenciamento de maturidade. (6) Pode-se entender que ao utilizar o modelo de maturidade, a organização envolvida terá vários benefícios como: mais produtividade das equipes, redução de custos, riscos do projeto minimizados, aumento do retorno de investimento e qualidade no produto entregue. (7)

2.2.1 MPS.BR

O MPS.BR (Melhoria do Processo de Software Brasileiro) tem o objetivo de melhorar a qualidade dos processos de software, principalmente das micro, pequenas e médias empresas. Ele baseia-se no CMMI (Capability Maturity Model - Integration) e é constituído pelas normas: NBR ISO/IEC 12207 – Processo de Ciclo de Vida de Software, pelas emendas 1 e 2 da norma internacional ISO/IEC 12207 e pela ISO/IEC 15504 – Avaliação de Processo. (10) Ele está dividido em três modelos: Modelo de Referência (MR-MPS), Método de Avaliação (MA-MPS), Modelo de Negócio (MN-MPS) como pode ser visto na (Figura 2).

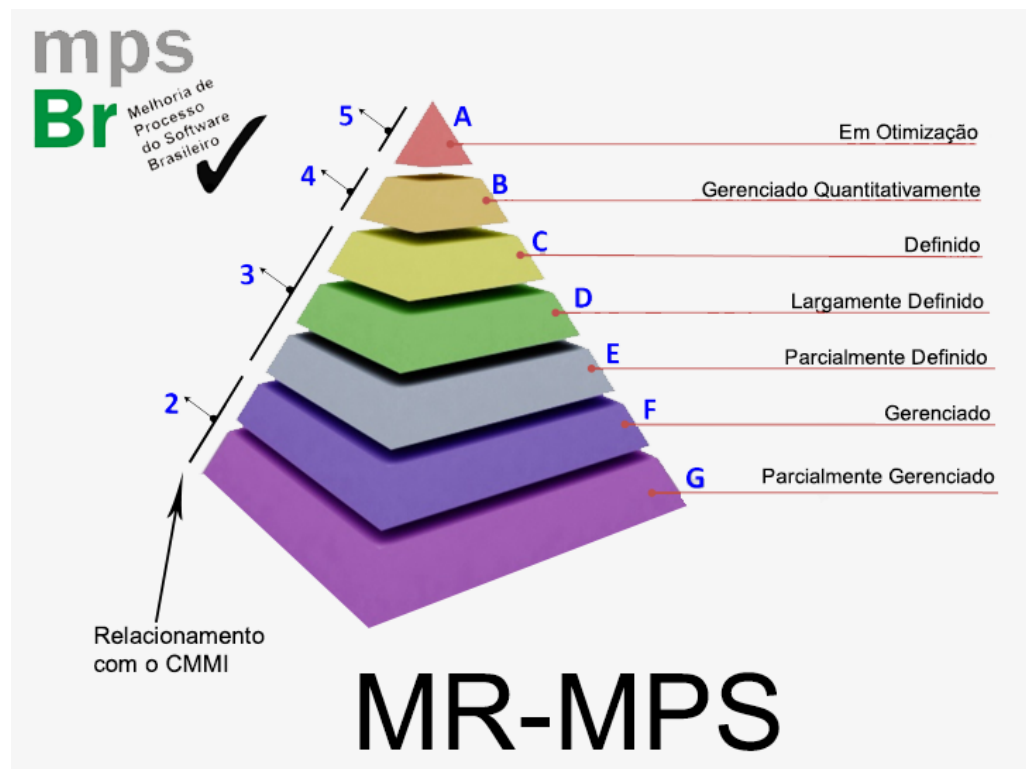
Figura 2 – Programa MPS.BR



- MR-MPS

O modelo de referência para melhoria do processo de software é dividido em vários níveis de maturidade como pode ser visto na (Figura 3).

Figura 3 – Níveis de maturidade MR-MPS



“Os níveis de maturidade estabelecem patamares de evolução de processos, caracterizando estágios de melhoria da implementação de processos na organização. O nível de maturidade em que se encontra uma organização permite prever o seu desempenho futuro ao executar um ou mais processos.” (10)

Em cada nível de maturidade da figura 3 são analisados os processos fundamentais, processos organizacionais e processos de apoio.

Processos Fundamentais:

- Aquisição
- Gerência de requisitos
- Desenvolvimento de requisitos
- Solução técnica
- Integração do produto
- Instalação do produto
- Liberação do produto

Processos Organizacionais:

- Gerência de projeto
- Adaptação do processo para gerência de projeto

- Análise de decisão e resolução
- Gerência de riscos
- Avaliação e melhoria do processo organizacional
- Definição do processo organizacional
- Desempenho do processo organizacional
- Gerência quantitativa do projeto
- Análise e resolução de causas
- Inovação e implantação na organização

Processos de apoio:

- Garantia de qualidade
- Gerência de configuração
- Validação
- Medição
- Verificação
- Treinamento

A engenharia de requisitos se encontra nos níveis D e G, Largamente Definido e Parcialmente Gerenciado, respectivamente.

– **Nível G**

Gerência do Projeto e Gerência de Requisitos compõem o nível G de maturidade. A finalidade do processo de Gerência de Requisitos é gerenciar os requisitos e componentes do produto do projeto. Além de identificar as inconsistências entre os requisitos, planos e produtos do projeto.(12)

Resultados esperados:

GRE 1 - Comunicação constante com os fornecedores de requisitos;

GRE 2 - Os requisitos são compreendidos;

GRE 3 - Requisitos são aceitos;

GRE 4 - Comprometimento com os requisitos; **GRE 5** -Rastreabilidade entre os requisitos, planos do projeto e produtos do trabalho; **GRE 6** - São corrigidas as inconsistências entre os requisitos, planos e produtos do projeto; **GRE 7** - Ao longo do projeto as mudanças nos requisitos são gerenciadas.

– **Nível D**

Neste nível acontece o Desenvolvimento de Requisitos, Integração do Produto, Solução Técnica, Validação, e Verificação. (12)

Os requisitos são estabelecidos em conformidade com o cliente.

Resultados esperados:

DRE 1 - São identificadas as necessidades, expectativas, restrições e requisitos de interface do cliente; **DRE 2** - Requisitos funcionais e não-funcionais são estabelecidos; **DRE 3** - Requisitos são refinados; **DRE 4** - Conceitos operacionais e cenários são desenvolvidos; **DRE 5** - As funcionalidades são desenvolvidas; **DRE 6** - Requisitos são avaliados para assegurar as necessidades dos interessados; **DRE 7** - Requisitos são validados.

- MA-MPS

É o método que permite a avaliação segundo o modelo MPS, seguindo o processo de avaliação e método de avaliação MPS, e características de qualificação dos avaliadores. (10)

O processo de avaliação é constituído por 4 etapas :

- Contratar a avaliação;
- Preparar a realização da avaliação;
- Realizar a avaliação;
- Documentar os resultados da avaliação.

- MN-MPS

O Modelo de Negócio tem o objetivo de credenciar as instituições que seguirem o Modelo de Referência (MR-MPS), Método de Avaliação (MA-MPS), tiverem experiência em processos de software e possuírem estratégias de implementação do modelo e treinamento dos consultores. (10)

2.3 Mapeamento do Contexto vs Abordagens

De acordo com a descrição e análise das abordagens, e as características da equipe, e do projeto, pode-se descrever alguns pontos chave que influenciam na decisão.

1. A empresa está dispondo de um gerente para auxiliar a equipe de desenvolvimento em todas as tarefas de levantamento de requisitos. 2. A equipe de engenharia de requisitos é integrada e comunicativa, características que facilitam a comunicação com o cliente. 3. A formação da equipe está disposta em 4 membros, dos quais 3 membros já trabalharam juntos utilizando metodologias ágeis, e todos os membros possuem conhecimentos básicos sobre métodos ágeis. 4. A distância entre o cliente e a equipe é pequena, pois as dependências do cliente encontram-se no mesmo local onde a equipe realizará o trabalho.

3 Justificativa

“Aviso! Não cometa o erro de assumir que a agilidade lhe dará licença para abreviar soluções. Processo é um requisito e disciplina é essencial” (Pressman)

Segundo Pressman, para projetos que adotam uma filosofia de desenvolvimento de software ágil, um ambiente ideal seria aquele em que engenheiros de software trabalham juntos na mesma equipe.

Visto que a abordagem ágil requer uma proximidade com o cliente, e a cliente tem uma disponibilidade de tempo para que possamos fazer reuniões semanais, o grupo chegou a conclusão que a abordagem ágil se adéqua melhor ao nosso contexto.

Além da proximidade com a cliente, devido a mesma ser uma estudante da mesma faculdade dos componentes do grupo, em nossa primeira reunião, ela demonstrou possuir conhecimentos da área de software, que foi um dos fatores determinantes para nossa escolha, deixando os componentes mais seguros quanto a metodologia adotada.

Outro fator importante analisado, foi que ela já orientou outro aluno da faculdade que era da Eletrojun para fazer uma parte do sistema, demonstrando uma certa experiência na descrição de requisitos, outro fator analisado e que contou como ponto positivo para a definição da abordagem.

3.1 Panorama Ágil

O manifesto foi feito em 2001 por 17 desenvolvedores que foram na contramão da produção de software tradicional. O manifesto passava a valorizar os indivíduos e interações mais que os processos e ferramentas, o software funcionando mais que a documentação abrangente, colaboração do cliente mais que negociação de contrato, e responder a mudanças mais que seguir um determinado plano.

O manifesto ágil surge com o intuito de sanar as fraquezas reais e perceptíveis da maneira tradicional de produção de software. Apesar da metodologia fornecer benefícios importantes, não é indicada em todas as situações, porque “There is no silver bullet” (Fred Brooks) não existe a bala de prata para a metodologia de desenvolvimento empregada no projeto, tudo vai ter que ser analisado para que se chegue a metodologia mais sensata ao projeto.

moda, e explica o que é ágil na sua concepção:

“Uma equipe ágil é aquela rápida e capaz de responder apropriadamente a mudanças. Mudanças têm muito a ver com desenvolvimento de software. Mudanças no software que está sendo criado, mudanças nos membros da equipe, mudanças devido a novas tecnologias, mudanças de todos os tipos que poderão ter um impacto no produto que está em construção ou no projeto que cria o produto.”

Já é mais que comprovado que o ágil é a metodologia de desenvolvimento de software que melhor se adapta a mudanças. O suporte para mudanças no ágil é incorporado em tudo que é desenvolvido, tratado pelo próprio Ivar Jacobson como o coração e a alma do software.

4 Processo de Engenharia de Requisitos

5 Elicitação de Requisitos

A palavra vai explicar a funcionalidade do levantamento de requisitos, apesar de ser um termo mais generalista, ela vai na essência do levantamento de requisitos. Essa parte é a que o profissional de TI, que geralmente desempenha o papel de engenheiro de requisitos, conversa com o cliente para entender sua necessidade, e o mesmo o orienta, para que suas necessidades sejam atendidas. É importante ressaltar que essa é uma fase muito crucial para todos os projetos, principalmente na área de software, pois nesse momento é que a solução do problema do cliente vai ser elaborada.

5.1 Técnicas de elicitação de requisitos

Ao analisarmos o nosso contexto para o levantamento de requisitos, percorremos e analisamos as diversas técnicas que são usadas atualmente, como o levantamento orientado a ponto de vista, etnografia, prototipagem, entrevista, questionário, brainstorming e JAD. Logo após analisarmos chegamos a conclusão que a técnica que se adequa melhor ao nosso contexto é a entrevista. Partindo do princípio que já existe uma parte do nosso software pronto, e que nossa cliente conhece bem os requisitos e tem noções de desenvolvimento de software, técnicas como JAD apesar de promover a cooperação, o trabalho em grupo e o entendimento, acabam não se adaptando, devido sua metodologia ser baseada na criação e na resolução de problemas, onde os desenvolvedores ajudam os usuários do sistema a criarem problemas e eles mesmos proporem soluções. O Brainstorming é uma técnica que a princípio suas ideias não parecem convencionais, mas são encorajadas, estimulando frequentemente os participantes para a elaboração de soluções criativas. Mas a exploração de novas ideias não se aplica ao nosso caso, porque a ideia do nosso software já existe. Já a técnica da aplicação de questionários é geralmente usada quando existe uma distância física entre os usuários, não se aplicando ao nosso caso, já que nossa cliente tem uma proximidade física do nosso grupo, e aplicação dos questionários se torna obsoleta e desnecessária. A prototipagem parte do pressuposto de fazermos protótipos para a validação do cliente, implementando os aspectos críticos do sistema, mas nesse caso já temos um protótipo já validado pelo cliente, técnica que se adotada, geraria um retrabalho. As técnicas de Workshops são usadas para promover a interação entre os stakeholders, não se aplicando novamente ao nosso contexto, sendo que nos dirigiremos a apenas uma representante da Eletrojun, apesar.

5.1.1 Entrevista

A entrevista visa identificar o problema, propor a solução, negociar os aspectos de abordagem, identificar um conjunto preliminar de requisitos do projeto.

- As entrevistas serão conduzidas com a participação dos engenheiros de software e outros interessados.
- A agenda de entrevistas deve possibilitar uma entrevista que cubra todos os pontos importantes, e abra espaço para o fluxo de novas ideias.
- Deve haver um controle da reunião através de um plano de entrevista, para que não haja uma dispersão do conteúdo pertinente ao entrevistador.

6 Gerenciamento de Requisitos

7 Planejamento do Projeto

8 Ferramentas de Gestão de Requisitos

8.1 Análise de Ferramenta de Gestão de Requisitos

Referências

Apêndices

APÊNDICE A – Primeiro Apêndice

Texto do primeiro apêndice.

APÊNDICE B – Segundo Apêndice

Texto do segundo apêndice.

Anexos

ANEXO A – Primeiro Anexo

Texto do primeiro anexo.

ANEXO B – Segundo Anexo

Texto do segundo anexo.