

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA SUL-RIO-
GRANDENSE - IFSUL, *CAMPUS* PASSO FUNDO
CURSO DE TECNOLOGIA EM SISTEMAS PARA INTERNET**

VANESSA FERREIRA

**ANÁLISE DE FERRAMENTAS PARA GERENCIAMENTO DE
REQUISITOS DE SOFTWARE**

Prof. Me. André Fernando Rollwagen

PASSO FUNDO, 2014

VANESSA FERREIRA

**ANÁLISE DE FERRAMENTAS PARA GERENCIAMENTO DE
REQUISITOS DE SOFTWARE**

Monografia apresentada ao Curso de Tecnologia em Sistemas para Internet do Instituto Federal Sul-Rio-Grandense, *Campus* Passo Fundo, como requisito parcial para a obtenção do título de Tecnólogo em Sistemas para Internet.

Orientador: Prof. Me. André Fernando Rollwagen

PASSO FUNDO, 2014

VANESSA FERREIRA

**ANÁLISE DE FERRAMENTAS PARA GERENCIAMENTO DE
REQUISITOS DE SOFTWARE**

Trabalho de Conclusão de Curso aprovado em ____/____/____ como requisito parcial para a
obtenção do título de Tecnólogo em Sistemas para Internet

Banca Examinadora:

Prof. Me. André Fernando Rollwagen (Orientador)

Prof. Esp. Carmen Vera Scorsatto (Convidada)

Prof. Me. Evandro Miguel Kuszera (Convidado)

Prof. Dr. Alexandre TagliariLazzaretti
Coordenação do Curso

PASSO FUNDO, 2014

*À minha família,
que com carinho e compreensão
estiveram do meu lado me
estimulando em todos os momentos.*

AGRADECIMENTOS

Agradeço primeiramente ao meu pai Gil Ferreira que sempre com apoio me incentivou a alcançar meus objetivos, e aos meus irmãos que sempre estiveram do meu lado em todos os momentos.

Agradeço aos professores do Instituto Federal Sul-Rio-Grandense Campus Passo Fundo do Curso Superior de Tecnologia em Sistemas para Internet, que sempre receptivos me auxiliaram no decorrer do curso, e em especial o meu orientador Prof. Me. André Fernando Rollwagen que me apoiou e incentivou na execução desse projeto.

E por fim agradeço aos meus colegas e amigos que de alguma forma participaram dessa etapa, com estímulo e companheirismo, os levo em meu coração.

RESUMO

A qualidade de software é um requisito importante para os que desejam desenvolver softwares. Existem formas de gerenciar o software a ser projetado e os desenvolvedores devem escolher a maneira correta e eficaz que melhor se encaixe com as necessidades de seus projetos. Dessas formas de gerenciamento, são fornecidas ferramentas que auxiliam a controlar as alterações que ocorrem nos sistemas que se encontram em fase de desenvolvimento, de entrega ou de manutenção. Neste trabalho serão apresentadas ferramentas de gerenciamento que acompanham as mudanças que ocorrem nos requisitos, também serão verificadas se estas ferramentas suprem as necessidades do mercado. Dentre as ferramentas expostas, será elencada a que melhor gerencia requisitos indicando-a para empresas de pequeno porte.

Palavras-chave: Engenharia de Software; Engenharia de Requisitos; Gerenciamento de Requisito.

ABSTRACT

The quality of software is an important requirement for those wishing to develop software. There are ways to manage the software to be designed and developers should choose the correct and effective way that best fits the needs of your projects. These forms of management tools to help manage systems that are under development, delivery or maintenance is provided. In this project management tools that control the changes that occur in the requirements will be presented, and verify that the tools presented serve the needs of the market. After this analysis the solution that best solve these deficiencies is proposed.

Key words: Software Engineering; Requirements Engineering; Requirements Management.

LISTA DE FIGURAS

Figura 1 - Solicitação de Clientes.....	21
Figura 2 - Diagrama de Gantt	22
Figura 3 - Tela inicial do usuário de recurso	23
Figura 4 - Gerando relatórios nativos da ferramenta.....	24
Figura 5 - Lista de formulários customizados	25
Figura 6 - Inserir campos nos formulários	25
Figura 7 - Formulário de gestão de riscos	26
Figura 8 - Gerenciando requisitos	27
Figura 9 - Cadastro de caso de uso	28
Figura 10 - Subprojeto e rastreabilidade	29
Figura 11 - Rastreabilidade através dos requisitos	29
Figura 12 - Tela de rastreamento.....	30
Figura 13 - Escolher o projeto	31
Figura 14 - Criando novo projeto	31
Figura 15 - Guia de documentos do projeto	32
Figura 16 - Criando requisitos	32
Figura 17 - Lista de requisitos	33
Figura 18 - Configurando requisitos.....	33
Figura 19 - Gerenciamento de Requisitos	34
Figura 20 - Rastreabilidade.....	35
Figura 21 - Reutilização de requisitos	36

SUMÁRIO

1	INTRODUÇÃO	8
1.1	MOTIVAÇÃO	10
1.2	OBJETIVOS	10
1.2.1	Objetivo Geral	10
1.2.2	Objetivos específicos	11
2	REFERENCIAL TEÓRICO	12
2.1	ENGENHARIA DE SOFTWARE.....	12
2.2	ENGENHARIA DE REQUISITOS	13
2.3	GERENCIAMENTO DE REQUISITOS.....	15
2.4	QUALIDADE DE SOFTWARE	15
2.5	LEVANTAMENTO E ELABORAÇÃO DE REQUISITOS	16
3	METODOLOGIA	18
4	RESULTADOS E DISCUSSÕES	19
4.1	FERRAMENTAS DE GERENCIAMENTO.....	19
4.1.1	Plandora.....	19
4.1.3	Almirante	26
4.1.4	IBM <i>RationalRequisitePro</i>	30
4.1.5	<i>Serena Dimensiosn</i> RM.....	33
4.2	ANÁLISE DAS FERRAMENTAS DE GERENCIAMENTO DE REQUISITOS DE SOFTWARE.....	36
5	CONSIDERAÇÕES FINAIS.....	40
5.1	TRABALHOS FUTUROS	40
	REFERÊNCIAS	41

1 INTRODUÇÃO

Gerenciar o processo de produção de software é um quesito importante para que haja uma melhoria no seu desenvolvimento e durante todo seu ciclo de vida. Segundo Pressman (2011), empresas e desenvolvedores têm a preocupação em entregar aos seus clientes um sistema de qualidade e que diminua o custo e o tempo gasto com manutenções, porém a qualidade de software continua sendo um problema, na qual clientes culpam desenvolvedores por utilizar práticas descuidadas e entregar um software de baixa qualidade. Os desenvolvedores culpam os clientes por exigirem datas absurdas de finalização e contínuas mudanças forçando-os a entregar um software não validado completamente. Ambos estão com razão. Implementar um sistema que atenda às necessidades do cliente, entregando-o no prazo determinado, não é algo habitual, constantemente ocorrem problemas e contratemplos que não agradam às partes interessadas. Entre as atividades de gerenciamento de um software, destaca-se a devida documentação do que se especifica nos requisitos, ou seja, do que verdadeiramente o cliente deseja, para o melhor desenvolvimento.

O trabalho apresentado está relacionado à Engenharia de Software, com ênfase maior no gerenciamento e controle de requisitos. A pesquisa apresenta a importância do controle e gerenciamento de requisitos, a qual consiste em documentar softwares em desenvolvimento de modo a buscar maior eficácia no entendimento das necessidades expostas pelos clientes durante o levantamento de requisitos.

O tema que será abordado é o Gerenciamento de Requisitos, expondo a forma como é feito este processo de controle e gerenciamento, auxiliando na melhoria da qualidade do software. O estudo está focado na gerência dos requisitos, mais especificamente com o levantamento dos dados, que requer um envolvimento do desenvolvedor, dando maior atenção a este processo, demonstrando que é possível realizar um controle de requisitos com maior confiabilidade. Com o enfoque no gerenciamento de requisitos, busca-se demonstrar softwares existentes na área e desenvolver um software para gestão de requisitos utilizando software livre.

Segundo Lopes (2003, apud SANTOS, 2004) os requisitos podem apresentar alguns problemas quando não especificados corretamente como: os requisitos não estão de acordo com o que os clientes desejam; podem estar incompletos; as modificações são caras e o processo é lento; e não existe entendimento entre os envolvidos com o projeto. É comum que ocorra esse tipo de situação em qualquer fase do desenvolvimento, e não há uma solução

completa para resolver estes problemas. As técnicas de Gerenciamento de Requisitos podem reduzir estes problemas.

O problema é que muitos softwares disponibilizados no mercado gerenciam requisitos de modo a não suprir por completo as necessidades dos desenvolvedores durante a documentação e o desenvolvimento do sistema, não permitindo um controle eficaz de mudanças. Também existem ferramentas muito caras, que empresas e desenvolvedores optam por não utilizar em um software devido ao valor, e algumas ferramentas livres disponíveis não fornecem a documentação eficaz esperada.

De acordo com Sommerville (2011), é difícil demonstrar ao usuário um conjunto de requisitos que atenda as necessidades do cliente, pois estes usuários devem imaginar o funcionamento do sistema e como ele se ajusta com o trabalho diário, também para profissionais qualificados torna-se difícil. Portanto após qualquer alteração no documento de requisitos é inevitável a mudança no próprio sistema. Desta forma, busca-se responder como implementar um sistema, alcançando as necessidades do cliente, evitando custos elevados e perda de tempo com manutenções.

A análise de requisitos tem grande importância durante o desenvolvimento de um software, por isso é necessário encontrar uma maneira segura de realizar a coleta e a manutenção dos dados que serão necessários à base inicial do desenvolvimento. Também é preciso propiciar uma gerência dos requisitos do sistema de uma forma mais dinâmica, que facilite a troca e a inserção de novos requisitos. Obtendo-se uma ferramenta que auxilie no gerenciamento dos requisitos, a qual irá representar mais agilidade na manutenção do software e nas mudanças de que ele necessitar. Sommerville nos diz que:

O gerenciamento de mudanças é essencial, pois é necessário decidir se os benefícios da implementação de novos requisitos justificam os custos de implementação. A vantagem de usar um processo formal de gerenciamento de mudanças é que todas as propostas de mudanças são tratadas de forma consistente, e as alterações nos documentos de requisitos são feitas de forma controlada (2011, p. 79).

A principal fase do ciclo de desenvolvimento do sistema é a Análise de Requisitos, pois é onde se define e entende os requisitos, por tanto eles necessitam ser monitorados de acordo com as mudanças que ocorrem durante a produção do sistema. A Gerência de Requisitos ou o Gerenciamento de Mudanças de Requisitos define uma estrutura de tarefas para validar um documento de requisitos do sistema (SANTOS, 2004).

Desta forma o trabalho está dividido em três momentos. No primeiro momento, abordam-se as definições e conceitos necessários para iniciar o trabalho, ou seja, a teoria

necessária como embasamento para a resolução do problema. No segundo momento são descritos alguns softwares existentes, os quais controlam e gerenciam requisitos para que possa ser apontada a melhoria necessária, contribuindo para a qualidade na gestão de softwares.

No terceiro momento são apresentados os pontos positivos e negativos que as ferramentas de gerenciamento possuem, visando eleger a melhor ferramenta apropriada para pequenas empresas, dentre as expostas neste trabalho.

1.1 MOTIVAÇÃO

A necessidade dos desenvolvedores de software em entregar sistemas de qualidade aos seus clientes, têm se tornado um desafio cada vez maior. Ao desenvolver um sistema deve-se seguir as especificações e necessidades dos clientes, porém sabe-se que durante o processo de desenvolvimento ocorrem mudanças no sistema, essas mudanças podem ocorrer tanto no levantamento de requisitos quanto no fim do desenvolvimento, portanto se faz necessária uma documentação eficaz e que atenda as necessidades do sistema desenvolvido.

Para auxiliar na documentação de um sistema em desenvolvimento, busca-se um software que auxilie a gerenciar as mudanças que ocorrem nos requisitos, e que possa atender as necessidades de alterações em um projeto, com o objetivo de controlar e gerenciar os requisitos, proporcionando histórico das alterações, e que venha auxiliar utilizando-as em outros projetos, contribuindo e auxiliando em projetos futuros.

1.2 OBJETIVOS

1.2.1 Objetivo Geral

O objetivo desta monografia é analisar ferramentas de gerenciamento de requisitos, e eleger a ferramenta apropriada para a utilização em pequenas empresas de desenvolvimento de software.

1.2.2 Objetivos específicos

Os objetivos específicos são divididos em três fases:

- descrever alguns sistemas existentes no gerenciamento de requisitos;
- apontar os pontos fortes e fracos encontrados nos softwares existentes; e
- eleger a melhor ferramenta entre as estudadas para gerenciamento de requisitos em pequenas empresas;

2 REFERENCIAL TEÓRICO

Neste capítulo são abordados os conceitos de Engenharia de Software, Engenharia de Requisitos, Gerenciamento de Requisitos, Qualidade de Software, Levantamento e Elaboração de Requisitos, que são embasamento necessário a este trabalho.

2.1 ENGENHARIA DE SOFTWARE

A engenharia de software está ligada a todas as etapas da produção de software, desde o início da especificação do sistema até sua manutenção, quando se encontra em utilização. Voltada para aspectos técnicos de desenvolvimento, também inclui atividades de gerenciamento do projeto como o desenvolvimento de ferramentas, métodos e teorias que apoiam a produção do software. De maneira geral a engenharia obtém resultados de qualidade de acordo com o orçamento e com o prazo estimado (SOMMERVILLE, 2011).

Uma tecnologia em camadas, a engenharia de software está comprometida com a organização e a qualidade. Estas camadas são: ferramentas; métodos; processo; e foco na qualidade. O mais importante é o foco na qualidade que sustenta a engenharia de software. O processo permite um desenvolvimento racional e dentro do prazo. Os métodos envolvem atividades de modelagem e técnicas descritivas e gerem cada área da tecnologia. E por fim as ferramentas que fornecem suporte aos processos e métodos (PRESSMAN, 2011).

Conforme Paula Filho:

A engenharia de software se preocupa com o software como produto. Estão fora do seu escopo programas feitos unicamente para diversão do programador. Estão fora do seu escopo também pequenos programas descartáveis, feitos por alguém exclusivamente como meio para resolver um problema dessa pessoa, e que não serão utilizados por outros (2009, p. 5).

Os processos de desenvolvimento de software são atividades relacionadas que levam à produção de um software. Não existe o processo de software ideal, a maioria das empresas cria seu próprio processo de desenvolvimento visando um melhor aproveitamento da capacidade das pessoas e das características do sistema em desenvolvimento. Os processos de software devem incluir quatro atividades fundamentais a engenharia de software, essas atividades são: a especificação de software, definindo suas funcionalidades e restrições; projeto e implementação de software, produzindo o software atendendo as especificações; validação de software, validando para atender as necessidades dos clientes; e evolução de software, evoluindo de modo a atender as necessidades de mudança dos clientes. A engenharia de requisitos, tema abordado na seguinte seção, faz parte da atividade de

especificação de software que compreende a definição de serviços e restrições do sistema em desenvolvimento (SOMMERVILLE, 2011).

2.2 ENGENHARIA DE REQUISITOS

De acordo com Sommerville (2011), os requisitos são as descrições e restrições de um sistema que demonstram as necessidades do cliente para então desenvolvê-lo. O requisito pode ser declarado de uma maneira simples, que demonstre as necessidades do cliente, e também através de uma declaração formal, que possa documentar de uma forma mais completa o que é necessário para a implementação do sistema.

A engenharia de requisitos examina como o software deve ser realizado, as especificações que são necessárias para o projeto, quais as prioridades de trabalho, informações, e comportamentos que possui grande impacto no resultado final do desenvolvimento (PRESSMAN, 2011). Conforme relata Rocco (2001), a engenharia de requisitos deve propor métodos, técnicas e ferramentas que englobam todo o processo de definição dos requisitos.

Segundo Paula Filho (2009), muitos dos envolvidos no desenvolvimento de um software não entendem o motivo de haver as especificações dos requisitos. A engenharia de requisitos compreende um conjunto de técnicas, documentação e análise de requisitos. Sendo a documentação feita da forma satisfatória, a chance de os desenvolvedores entenderem os requisitos com facilidade é muito maior.

Requisitos envolvem ideias de como deve ser o sistema, que são fornecidas pelo cliente, ou seja, é a problemática que será solucionada pelos desenvolvedores de sistemas, os quais devem possuir total domínio de como solucionar o problema de maneira eficaz. De acordo com Pressman (2011), para o melhor entendimento dos requisitos, a engenharia de requisitos é um importante mecanismo, que tem seu início na comunicação entre os interessados no projeto e permanece durante toda a modelagem, adaptando-se com as necessidades do projeto e de todos os envolvidos que estão ligados ao desenvolvimento do software.

Ao realizar o levantamento de requisitos, deve-se levar em consideração os níveis de descrição dos mesmos, que informam os requisitos tanto na visão do cliente/usuário, quanto na visão do desenvolvedor. A diferença entre eles são os requisitos de usuário e requisitos de sistema.

É necessária a distinção dos diferentes níveis de declarações, separando em dois níveis, a descrição abstrata do sistema, denominada requisitos do usuário, na qual descreve-se, através de diagramas, os serviços e restrições que o software pretende desenvolver; e a descrição detalhada do sistema, denominada requisitos do sistema, os quais devem ser definidos com exatidão e precisão, sendo o que será documentado e posteriormente implementado no sistema, devem estar dispostas de maneira bem detalhada as funções, os serviços e as restrições para a melhor desenvoltura do software (SOMMERVILLE, 2011).

Segundo Niz (2001, apud SANTOS, 2004), os requisitos de sistema descrevem o ambiente e os objetivos que o software deverá alcançar, sendo que os requisitos de usuário requerem reuniões e encontros entre as partes envolvidas para definir as necessidades do sistema que possa satisfazer a todos os envolvidos e descrever de forma clara as necessidades, restrições e atividades que os usuários irão executar. Logo, requisitos do sistema são funcionalidades do software que capacitam usuários e satisfazem requisitos. Rocco (2001) afirma que os requisitos do usuário definem as tarefas sobre as quais os usuários devem estar informados de como realizá-las para o funcionamento do software, e que os requisitos do sistema compreendem necessidades funcionais, que especificam o comportamento do sistema, e não funcionais, que dependem do desenvolvimento do sistema, das restrições e das qualidades nele impostas.

Conforme Sommerville (2011), os requisitos do sistema se dividem em requisitos funcionais e requisitos não funcionais. Os requisitos funcionais descrevem o comportamento do sistema, os quais dependem do tipo de software, dos usuários e da maneira como foram abordados no processo de descrição, ou seja, descrevem detalhadamente quais funções o sistema deverá implementar. Nos requisitos não funcionais do sistema são definidas as restrições que o sistema deverá possuir em determinada função ou serviço, e surgem a partir de necessidades impostas pelos clientes, em diversas ocasiões se aplicam ao sistema como um todo.

De acordo com Paula Filho (2009), os requisitos são a definição dos critérios, para que sejam de alta qualidade, devem ser claros, completos, implementáveis e testáveis, os que não apresentarem estas qualidades são problemáticos e devem ser revisados. Em relação aos requisitos do sistema, dividem-se em requisitos funcionais, com a representação do comportamento que o sistema deve apresentar; e os requisitos não funcionais, que apontam como serão apresentados os comportamentos.

Os requisitos de software devem ser gerenciados, de forma a possuir maior precisão nas especificações e restrições do sistema a ser desenvolvido, portanto a seção a seguir apresenta o gerenciamento de requisitos.

2.3 GERENCIAMENTO DE REQUISITOS

É importante realizar a documentação do sistema, pois é através dessa documentação que será realizado o controle das mudanças que ocorreram no sistema durante o seu ciclo de vida. Segundo Rocco (2001, p. 17), “o objetivo principal da representação é propiciar uma melhor compreensão do domínio do problema e uma precisa definição do domínio da solução.”

Segundo Paula Filho (2009), alterações nos requisitos são inevitáveis, por melhor que tenha sido o levantamento e a análise, desenvolvedores e usuários podem encontrar defeitos nos requisitos já especificados para o desenvolvimento do software, além disso pode haver ambiguidade ou falta de entendimento do que realmente é desejado, os problemas com requisitos podem ser minimizados através das boas práticas de análise de requisitos, porém não os elimina por completo.

De acordo com Sommerville (2011), o gerenciamento de requisitos compreende em controlar as alterações nos requisitos do sistema, possibilitando avaliações dessas mudanças. Deve-se começar a ser planejado como será realizada a gerência de mudanças dos requisitos durante o processo de elicitação dos mesmos.

A gestão de requisitos é voltada para o controle de mudanças que ocorrem no decorrer do projeto, ou seja, essa necessidade de mudança ocorre desde o início do processo de desenvolvimento do software e segue durante todo o seu ciclo de vida. Portanto a gestão de requisitos são atividades que auxiliam a identificar e acompanhar as mudanças necessárias em certo momento de desenvolvimento do software (PRESSMAN, 2011).

2.4 QUALIDADE DE SOFTWARE

Como já relatado anteriormente, a qualidade de software é uma premissa muito importante que os desenvolvedores prezam ao implementá-lo. Conforme Pressman (2011, p. 360), “a qualidade de software pode ser definida como: uma gestão de qualidade efetiva aplicada de modo a criar um produto útil que forneça valor mensurável para aqueles que o produzem e para aqueles que o utilizam.”

A qualidade de software envolve as especificações de requisitos, estas devem estar em conformidade com as necessidades do cliente, porém a qualidade do software não é algo fácil de adquirir, por haver muitas modificações no processo de desenvolvimento do software, esse objetivo muitas vezes não é alcançado.

Na área de software existem diferentes maneiras de interpretar a garantia e o controle de qualidade, uma delas é a definição de padrões e procedimentos que reforçam a qualidade que deve ser alcançada, na outra é a utilização dos processos de garantia de qualidade do software, como os de verificação e validação. Os gerentes de qualidade devem estabelecer uma cultura de qualidade fazendo com que os responsáveis pelo desenvolvimento do software tenham como objetivo atingir um alto nível de qualidade do sistema (SOMMERVILLE, 2011).

Segundo Paula Filho (2009), para garantir a qualidade de software deve haver ações preventivas em relação aos problemas que podem surgir. Busca-se artifícios preventivos para reduzir a quantidade de defeitos que deverão ser revisados e removidos posteriormente, seja em um estágio mais avançado do projeto ou durante a vida útil do software.

Yourdon (1990) declarava que é importante desenvolver softwares de qualidade, e que utilizar mecanismos como a modelagem do sistema auxilia a manter o foco em características que têm maior importância ao desenvolvimento, a questionar manutenções e correções de requisitos mantendo o baixo custo e nível menos elevados de riscos, e a verificar se os analistas reconheceram o ambiente que o usuário necessita e o documentou de maneira a facilitar o entendimento do desenvolvedor.

2.5 LEVANTAMENTO E ELABORAÇÃO DE REQUISITOS

No levantamento de requisitos são definidas as funções e restrições que o sistema deverá possuir, são documentadas as necessidades do sistema para que haja a sua implementação. Segundo Zah90 (apud PRESSMAN, 2011), o levantamento de requisitos envolve resolução de problemas, elaboração, negociação e especificação, o que orienta as equipes a identificar problemas, propor soluções e especificar os requisitos da solução.

Segundo Sommerville (2011), a elicitación e análise de requisitos são atividades em que *stakeholders* trabalham com clientes e usuários finais para definir domínio, desempenho, restrições, serviços, dentre outras especificações. As atividades que compõem a elicitación e análise são a obtenção dos requisitos, onde ocorre a coleta dos mesmos; classificação e organização, a qual agrupa os requisitos que possuem algum tipo de relação; **priorização** e

negociação, dedicada a tratar os conflitos que podem ocorrer com os requisitos, definindo as prioridades de requisitos através da negociação; e a documentação de requisitos, onde são produzidos documentos formais ou informais.

De acordo com Paula Filho (2009), a definição dos requisitos é uma tarefa que consiste em definir listas de requisitos funcionais que identificam restrições, relacionamentos, modo de operações, diagramas, requisitos de ambiente e hipóteses do que será desenvolvido, ou seja, a descrição de todos os aspectos relevantes para a especificação dos requisitos. Os requisitos podem ser classificados de acordo com sua importância e estabilidade, sendo assim, a classificação da prioridade dos requisitos ocorre da seguinte maneira: requisitos essenciais, que devem conter nas especificações; requisitos desejáveis aumentam o valor do produto, a ausência deles pode causar problemas indesejáveis; e requisitos opcionais, serão cumpridos se houver disponibilidade de tempo, depois que os demais estiverem atendidos.

3 METODOLOGIA

Este trabalho desenvolveu-se através de pesquisas bibliográficas, que forneceram o embasamento inicial teórico envolvendo a qualidade e gerenciamento de requisitos. Para obter-se um gerenciamento de requisitos de qualidade, exige-se maior comprometimento dos desenvolvedores de software, sendo assim, existem ferramentas que auxiliam a obter esta qualidade. Dentre as inúmeras ferramentas disponíveis podemos citar: Plandora, Almirante, IBM *RationalRequisitePro*, *Serena Dimensions RM*, *XUSE meeting your requirements*, Borlan Caliber RM, OSRMT (*Open Source Requirements Management Tool*), *Spider-CL*, DotProject, SIGERAR, HP *Quality Center*, entre outras. Há vários artigos que apresentam ferramentas de gerenciamento de requisitos, pode-se citar o *Requirements Tools (ACHRAFI e LAM, 2014)* o qual apresenta várias ferramentas de gerenciamento disponíveis que podem ser utilizadas por desenvolvedores.

Neste projeto estão apresentadas as ferramentas Plandora, Almirante, IBM *RationalRequisitePro*, e *Serena Dimensions RM* as quais estão detalhadas suas principais funcionalidades, voltadas para o gerenciamento de requisitos. Estas ferramentas foram escolhidas entre as demais, por apresentarem documentação explicativa de como utilizá-las e que apresentavam as principais funcionalidades das ferramentas, versões online de demonstração e inclusive versões licenciadas por até trinta dias de uso. O detalhamento das ferramentas é realizado através da sua funcionalidade apresentada e como o usuário deve proceder para realizá-la, e também através de figuras demonstrativas.

Posterior à exposição das ferramentas, encontram-se os pontos positivos e negativos que se apresentam durante a especificação ou utilização da ferramenta esses pontos são descritos para que possa ser mais bem avaliada cada ferramenta. Sabe-se que cada empresa de desenvolvimento tem seu próprio modo de trabalho, por isso deve ser avaliado como cada empresa trabalha para ver se a ferramenta que será utilizada se encaixa com o cotidiano dos desenvolvedores, portanto visando atender as diferentes necessidades, o trabalho é finalizado com a indicação de uma ferramenta de gerenciamento de requisitos para empresas de pequeno porte.

4 RESULTADOS E DISCUSSÕES

Neste capítulo estão apresentadas e analisadas as ferramentas de gerenciamento de requisitos, seus pontos positivos ou negativos, e indicada a ferramenta de gerenciamento que melhor atender as necessidades de gerenciamento de requisitos e de desenvolvimento de empresas de pequeno porte.

4.1 FERRAMENTAS DE GERENCIAMENTO

O gerenciamento de requisitos é um fator importante para auxiliar a qualidade do software, os desenvolvedores de sistemas têm a preocupação de garantir um controle nas mudanças que ocorrem nos projetos durante o ciclo de vida. Existem várias ferramentas que fazem esse controle, a seguir serão apresentadas algumas destas ferramentas.

As ferramentas de engenharia de requisitos possuem uma variedade de mecânica e de modelos gráficos que apresentam características comportamentais, funcionais e informativas do sistema, e auxiliam na modelagem, validação e gestão de requisitos, formando uma base para os demais processos de software (PRESSMAN, 2011).

Segundo Santos (2004), no gerenciamento de requisitos ligado ao ciclo de vida do software, os requisitos que se classificam como técnicos e não técnicos devem estar em constante monitoramento, e manter a documentação desse monitoramento para servir de auxílio aos futuros planejamentos. Para isto, existem ferramentas que auxiliam na visualização e manutenção das alterações realizadas no projeto.

A seguir serão descritas as ferramentas de gerenciamento de requisitos Plandora, Almirante, IBM *Rational RequisitePro* e *Serena Dimensions RM*.

4.1.1 Plandora

A Plandora é uma ferramenta *open source*¹ que gerencia o controle de recursos de um projeto, permite a reutilização do código-fonte, fornece aos usuários um mapeamento histórico para o controle de todo o gerenciamento do projeto e o disponibiliza aos demais usuários para que tenham conhecimento dessas mudanças. Controla e planeja os custos do projeto, e indica o desempenho diário que foi realizado em um determinado projeto. Para

¹permitir que diferentes usuários foquem em diferentes partes do projeto, a ferramenta também fornece a criação de relatórios por projeto de uma forma bem dinâmica (PLANDORA, 2014).

No site da ferramenta está disponibilizado na *Wiki HOWTO* o seu funcionamento, como os clientes e desenvolvedores podem utilizá-la. A seguir serão demonstradas as funcionalidades mais importantes disponibilizadas pela ferramenta Plandora.

- **Trabalhode iterações** – a ferramenta permite que o líder do projeto possa criar para cada projeto uma entrada de iteração. A ferramenta Plandora utiliza o método ágil *Scrum*² para realizar as iterações, pois este permite alterar requisitos durante o ciclo de vida do projeto. O líder cria um *Sprint* (maneira como o método ágil³ *Scrum* denomina a iteração), e liga os requisitos necessários a essa iteração, os requisitos que se encontram sem ligação com o *Sprint* são facilmente editados ou excluídos até mesmo pelos clientes. Os requisitos que encontram-se em uma iteração não podem ser alterados para evitar qualquer tipo de confusão que venha ocorrer.
- **Clientes de projetos realizam solicitações sobre determinado projeto** – quando um cliente deseja realizar uma solicitação de alguma funcionalidade que este deseja que esteja presente no projeto, deve logar no sistema, e então aparecerá o formulário para requisitar a solicitação, então há alguns campos que o cliente deve preencher alguns campos obrigatórios como o projeto a que se destina a solicitação, a categoria da solicitação, a prioridade dessa solicitação, e a descrição do que está acontecendo ou do que deseja que seja feito no projeto. Isto permite que o cliente acompanhe todas as solicitações feitas no projeto, e ter acesso ao histórico de solicitações de modo on-line. É possível visualizar todas as ações tomadas em cada solicitação. A Figura 1 demonstra o formulário de solicitações dos clientes.

¹ O software deve ser disponibilizado e distribuído o código fonte inclusive na sua forma compilada. Se o código não for distribuído deve haver outra forma de obter o código fonte sendo esta através da rede ou através de um custo razoável de reprodução e não deve restringir nenhuma parte de reproduzir ou vender o código (OPEN SOURCE, 2014)

² Método ágil focado no desenvolvimento iterativo, projetado para o uso de equipes **localizadas em diferentes lugares**, onde os membros se encontram em rápidas reuniões. O desenvolvimento do software envolve equipes distribuídas e membros situados em diferentes lugares. O método *Scrum* divide-se em três fases, a primeira fase é o planejamento geral, a segunda fase são os ciclos de *sprint*, e a terceira fase sendo a última que encerra o projeto, completando a documentação com quadros de ajuda e avalia lições aprendidas (SOMMERVILLE, 2011)

³ São métodos de desenvolvimento em que os incrementos são pequenos criando e disponibilizando versões aos clientes em um período entre duas a três semanas, envolvendo-os no desenvolvimento do software com o objetivo de obter um rápido feedback da evolução dos requisitos, utilizando mais a comunicação informal (SOMMERVILLE, 2011).

Figura 1 - Solicitação de Clientes

The screenshot displays the Plandora software interface. At the top, the Plandora logo is on the left, and navigation links (Oscar Wilde, Show Survey, Show Repository, Show Reports, Options, Exit) are on the right. The main content is divided into two sections:

Requirement Form

Requester: oscar Requirement.Num.: 624 *Suggest.Date: [calendar icon]

Project: Service Desk

Category: Failure-Hardware-Other

Priority: Medium

Description: Failed hard disk.

Meta Field Example: This is a meta field example...

* The Suggested Date only express the deadline intended by requester and can be used, or not, by project leader into planning.

Discussion List

Posted By franz at Nov 20, 2010 5:30:51 PM
Don't Panic! Explain exactly what's happened?

Buttons: Save Changes, Clear, Main Form

Requirement List

13 item(s) found, displaying from 1 to 4 Filter: [input] [OK] [info icon]

Req.Num	Description	Requer.	Project	Priority	Category	Status	
690	As the marketing department I want to produce a product brochure to hand-out at our internal conference	oscar	Foo Project	Medium	User Story	Waiting Approve	[edit] [delete] [info]
677	As a manager I want to be able to view a report on all new newsletter subscribers for a given ...	oscar	Foo Project	Medium	User Story	Waiting Approve	[edit] [delete] [info]
645	As the system I must send out a confirmation email when a site visitor successfully registers for the newsletter.	oscar	Foo Project	Low	User Story	Waiting Approve	[edit] [delete] [info]
624	Failed hard disk.	oscar	Service Desk	Medium	Failure-Hardware-Other	Waiting Approve	[edit] [delete] [info]

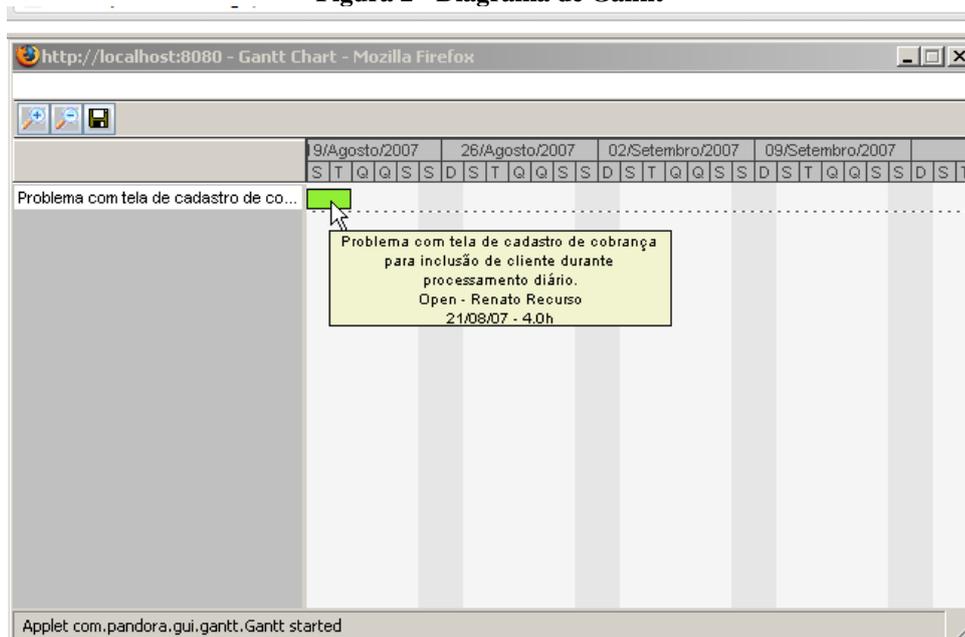
Buttons: Refresh Don't display Requirements closed since more than one week ago.

Plandora Project 2004-2010 v1.9.0

Fonte: Plandora,(2014)

- O líder do projeto trata as solicitações dos clientes**– A interface do líder do projeto é mais completa e possui algumas subdivisões. Nesta tela apresentada na Figura 2 ele pode observar as tarefas que foram criadas para que ele mesmo as execute; as solicitações dos clientes que estão pendentes de aprovação; lista de solicitações que o próprio líder requisitou semelhante a dos clientes; o líder tem acesso a informações dos novos pedidos como data, solicitante, o projeto que irá lhe auxiliar a decidir se esta solicitação será ou não aprovada; caso a solicitação seja aprovada deve ser criada uma tarefa para esta solicitação e repassada para o responsável que irá realizá-la, e agenda uma data de execução; podem ser criadas diversas tarefas para a mesma solicitação; quando a solicitação é mandada para execução, esta solicitação não se encontra mais na lista de solicitações pendentes; as tarefas podem ser modificadas ao longo do tempo e cabe ao líder ajustá-las no Diagrama de Gantt como demonstra a Figura 2; é possível também visualizar todas as tarefas referentes a um determinado projeto.

Figura 2 - Diagrama de Gannt



Fonte: Plandora,(2014)

- **Os recursos podem consultar tarefas e apontar horas trabalhadas** – qualquer usuário possui a responsabilidade por uma tarefa; todas as tarefas estão apresentadas no topo da tela inicial como mostra a Figura 3; as tarefas são organizadas com base no planejamento, mas é possível que o recurso escolha em qual trabalhar; na tela de edição possui o escopo, a data inicial, o estado e a duração da tarefa, aqui é possível deixar o solicitante ciente das ações que estão sendo tomadas em relação a requisição que foi realizada para criar a tarefa; após a execução da tarefa é possível fechar a solicitação e informar o tempo exato gasto para realizá-la; também permite documentar qual o procedimento realizado ao executar a tarefa, como por exemplo quais foram as dificuldades encontradas, soluções e lições, para que possa ser visto no projeto a acrescentar no conhecimento do projeto; para o solicitante, atualiza o histórico de sua solicitação mostrando que já está encerrada; o líder do projeto tem acesso a esta evolução das ações tomadas na tarefa através do diagrama *Gantt Chart*, e a mesma está atualizada automaticamente, e não é permitindo realizar alterações na tarefa através do diagrama.

Figura 3 - Tela inicial do usuário de recurso

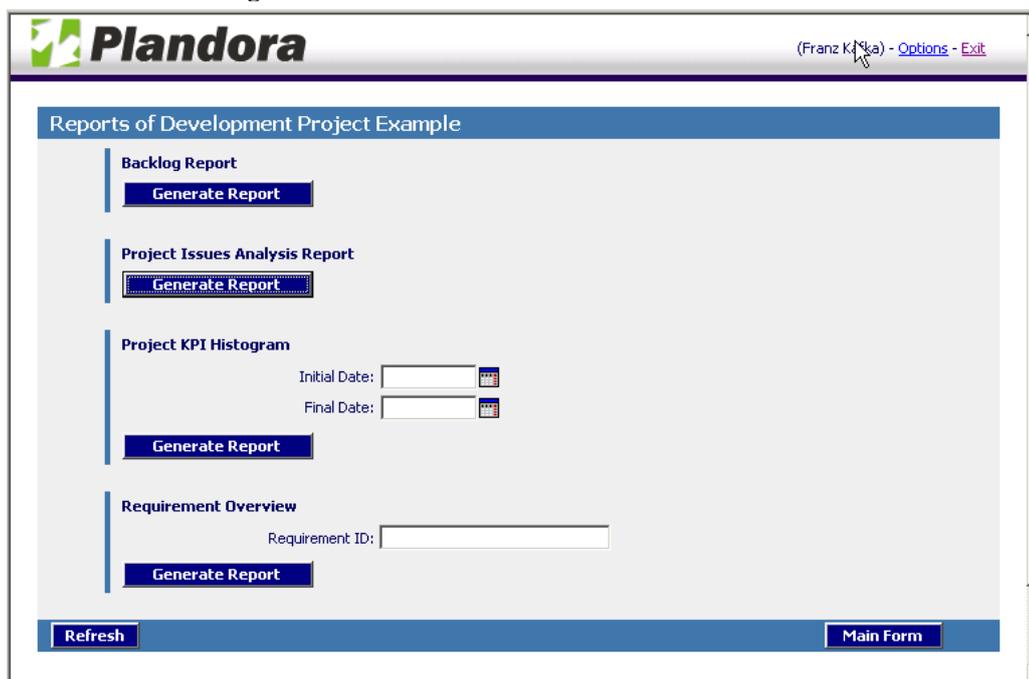
The screenshot shows the Plandora user interface. At the top, the Plandora logo is on the left, and the user's name '(Renato Recurso)' and navigation links 'Opções' and 'Sair' are on the right. The main content is divided into three sections:

- Minhas Tarefas:** Shows 1 item found. A table lists a task: 'Problema com tela de cadastro de cobrança' with start date '14/09/2007', duration '12h', status 'in-Progress', category 'Maintenance', and number of solutions '248'. Below the table is an 'Atualizar' button and a checkbox 'Não exibir tarefas que estejam fechadas há mais de uma semana.'.
- Minhas Solicitações:** Shows 'Nenhum item encontrado.' Below this is a table with columns: Num.Sol., Descrição, Projeto, Categoria, Prioridade, Status. Below the table is the text 'Nenhum registro encontrado para exibição', an 'Atualizar' button, and a checkbox 'Não exibir Solicitações que estejam fechadas há mais de uma semana.'.
- Meus Projetos:** Shows 1 item found. A table lists a project: 'Projeto Sistema Cobrança' with start date '18/08/2007', duration '27 dias', status 'Open', and profile 'Recurso'. Below the table are several icons for actions like print, refresh, and delete.

Fonte: Plandora, (2014)

- Visualizar e customizar relatórios** – o relatório de cada projeto é acessado através do ícone de uma impressora na lista de meus projetos; a ferramenta dispõe de muitos relatórios nativos relacionados a todos os projetos, para cada tipo de relatório é gerado através de um botão uma consulta ao servidor, que retorna o arquivo do relatório, como mostra a Figura 4; Também é possível criar um relatório customizado, através do formulário de relatórios inclusive, é possível remover ou alterar os relatórios nativos; é informado o nome do relatório, tem o campo no qual é inserido a query SQL, a qual realiza a solicitação ao servidor, logo após seleciona-se o projeto que deseja vincular a esse relatório, tanto a todos os projetos como a um específico, e vincular o novo relatório a um formato de arquivo, e utiliza oiReport, um projeto *open source*, para criar a interface do relatório que não faz parte da ferramenta Plandora, por fim deve ser informado o local no qual se encontra o arquivo base que irá criar o relatório de acordo com especificado, depois disso o novo relatório customizado está disponível juntamente com os relatórios nativos para ser utilizado.

Figura 4 - Gerando relatórios nativos da ferramenta



The screenshot displays the Plandora software interface. At the top left is the Plandora logo, and at the top right, the user name '(Franz Kiska) - Options - Exit' is visible. The main content area is titled 'Reports of Development Project Example' and contains four report categories, each with a 'Generate Report' button:

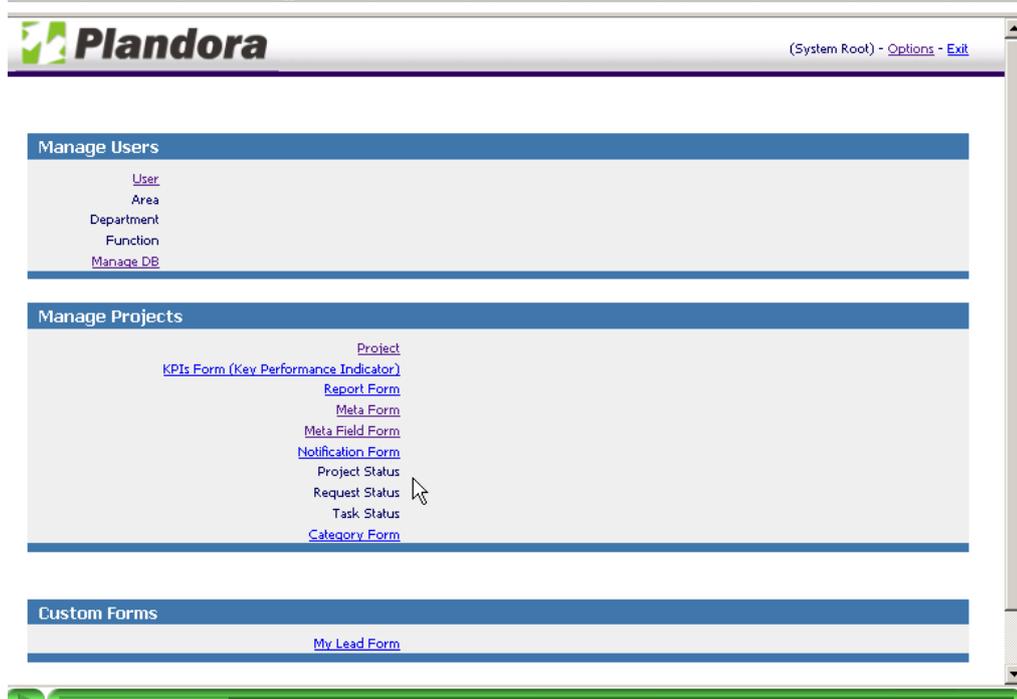
- Backlog Report**: A single 'Generate Report' button.
- Project Issues Analysis Report**: A single 'Generate Report' button.
- Project KPI Histogram**: Includes 'Initial Date:' and 'Final Date:' input fields with calendar icons, followed by a 'Generate Report' button.
- Requirement Overview**: Includes a 'Requirement ID:' input field, followed by a 'Generate Report' button.

At the bottom of the interface, there are two buttons: 'Refresh' on the left and 'Main Form' on the right.

Fonte: Plandora, (2014)

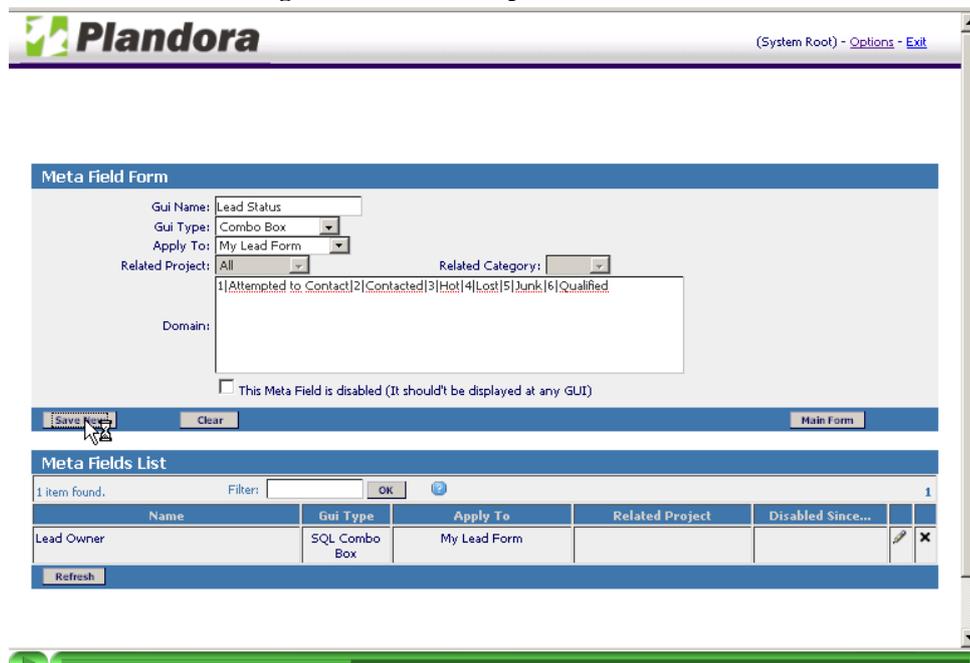
- **Criando formulários** – criar um formulário meta é útil para a lógica do projeto, após a criação do formulário aparecerá um novo link na lista de formulário como na Figura 5; mas é necessário inserir campos nesse novo formulário, ou seja, criar um campo no formulário de metas; o campo a ser inserido pode ser tanto em um formulário nativo como em um novo formulário que está sendo criado, como ilustra a Figura 6; em seguida informa o SQL que fará a consulta no servidor para retornar os campos necessários que já estão no sistema e criar o campo desejado; todos os capôs inseridos aparecem em uma lista de campos e informam qual formulário são pertencentes; e o novo formulário está criado de uma maneira prática e rápida; logo após os demais usuários também tem acesso a esse formulário que aparecerá em uma lista de formulários customizados, e poderá utilizá-lo.

Figura 5 - Lista de formulários customizados



Fonte: Plandora, (2014)

Figura 6 - Inserir campos nos formulários



Fonte: Plandora, (2014)

- **Gerenciar Riscos do Projeto** – através de um formulário de ocorrência de riscos, o que apresenta a Figura 7, a ferramenta disponibiliza uma forma de gerenciar os riscos encontrados e manter um histórico dessas ocorrências, a fim de identificar os riscos, classificar quanto ao seu impacto no projeto e definir soluções a esses riscos; como a

ferramenta disponibiliza um status do risco, dependendo da sua gravidade é sugerida uma ação para resolver este possível problema, de acordo com a evolução que irá tomando de acordo com cada risco, e as melhorias vão acontecendo a ferramenta muda o conteúdo na base de conhecimento, sendo útil para que os projetos futuros melhorem as lições aprendidas e tenham os mesmos riscos nos projetos seguintes.

Figura 7 - Formulário de gestão de riscos

Risk Form

Project: Development Project Example

Name: customer availability

Description: The customer is not available to periodic meeting. During the interviews he were interrupted many times to solve other issues.

Category: External Risk

Responsible: Franz

Probability: Probable Impact: High Tendency: Stable

Mitigation Plan: Send a email to Mr. Bing to emphasize the importance of interview meeting for the project successfull.

Contingency: Find out other alternatives to give information about the business rules.

Status: Identified

Relationship

New Relationship: Related with... Entity: OK

Save Changes Clear Project Form Main Form

Risk List

2 item(s) found, displaying from 1 to 2 Filter: OK

	Name	Category	Responsible			
214	customer availability	External Risk	Franz			
219	The F.R.K. application is not stable	Technical Risk	Franz			

Refresh

Fonte: Plandora, (2014)

4.1.2 Almirante

O Almirante é uma ferramenta de apoio ao processo de desenvolvimento que possui uma visão integrada. Cria um histórico que permite gerar dados comparativos para melhorar na construção de projetos. Os processos produtivos definidos pela Metodologia Celepar são apoiados pela ferramenta em relação a produtos gerados no desenvolvimento do software que são voltados para o registro de planejamento, execução e controle de processos produtivos (CELEPAR, 2014).

A ferramenta Almirante possui diversas funcionalidades, serão apresentadas as mais importantes, que a ferramenta dispõe, e estão voltadas para o gerenciamento de requisitos.

- **Registro de requisitos** – para cadastrar requisitos devem ser informados dados como: o projeto que o requisito pertence; a identificação do requisito informando se este é funcional ou não funcional; a prioridade com que este requisito deve ser atendido; se o requisito é suplementar, ou seja, se este se aplica ao projeto como um todo; uma breve descrição desse requisito; detalhar informações como riscos e restrições; informar quem identificou esse requisito, podendo ser cliente, técnico ou analista; informar se o requisito será implementado ou não e seu motivo; indicar a qualidade vinculada ao requisito não funcional; e permite relacionar o requisito aos demais já incluídos; conforme exibida na Figura 8. Se for necessário alterar um requisito, deve-se justificar o porquê dessa alteração, e a própria ferramenta se encarrega de gerar o histórico de requisitos.

Figura 8 - Gerenciando requisitos

*Identificação: Funcional Não Funcional Requisito suplementar: Sim Não

*Prioridade: Alto

*Descrição: Gerenciamento de requisitos (27 / 250)

*Detalhamento: Criar os seguintes mecanismos: (152 / 500)
 -acompanhamento do histórico de alterações;
 -informações de quem gerou o requisito e quando;
 -associação entre requisitos

*Fonte : Dani

Alocado ao proj.: Sim

Requisitos Associados

selecionar

Exibir	Descrição	Excluir
	Rastreabilidade de Requisitos	X
	Identificação de requisitos atendidos no caso de uso para uma determina iteração.	X
	Requisitos alocados e não alocados	X
	Atribuição de requisitos de sistema	X

Fonte: Celepar, (2014)

- **Cadastrar caso de uso** – o cadastro de caso de uso deve ser realizado relacionando-os aos requisitos para então promover a rastreabilidade dos mesmos. Os dados que serão informados são: identificar o projeto; informar o nome do caso de uso; informar o pacote a que pertence o caso de uso; informar uma breve descrição; informar a complexidade do caso de uso; informar quais são os requisitos atendidos; informar os

atores que devem interagir com o caso de uso. A Figura 9 demonstra a tela de cadastro de caso de uso.

Figura 9 - Cadastro de caso de uso

*Nome: AceitarProgramacaoFase

Pacote: Fase Novo pacote ⓘ

*Descrição: (135 / 500) Este caso de uso tem por finalidade permitir a aceitação, aceitação com restrição ou a não aceitação (devidamente justificada) da fase.

*Complexidade: Complexo ⓘ

*** Requisitos Atendidos**

selecionar

Página 1 de 1

Exibir	Nome	Excluir
<input type="checkbox"/>	Critérios de aceitação da fase	<input type="checkbox"/>

Página 1 de 1

Atores

Ator: -- Selecione --

Nome	Nível Complexidade	Características Ator	Excluir
Lider de Fase	Complexo	Humanos	<input type="checkbox"/>

(*) Campos de preenchimento obrigatório.

Fonte: Celepar, (2014)

- **Iteração** – deve ser informado: o projeto; um nome para a iteração ou subprojeto; quais casos de uso fazem parte desta iteração; relacionar os requisitos que já estão vinculados ao caso de uso. Também é exibida a rastreabilidade do caso de uso. Como demonstrado na Figura 10.

Figura 10 - Subprojeto e rastreabilidade

Fonte: Celepar, (2014)

- Rastreabilidade** – a ferramenta almirante disponibiliza quatro formas de acessar a função rastreabilidade: a primeira forma é a rastreabilidade através do caso de uso, ou seja, o foco dela é em um caso de uso específico; a segunda é a função de rastreabilidade através do requisito selecionado, focando em um requisito específico; a terceira se dá através da aba de gerenciamento de requisitos exibida na Figura 11; e a quarta forma se dá através da tela de rastreamento conforme a Figura 12, na qual pode ser selecionado o tipo da rastreabilidade e o subprojeto que deseja consultar.

Figura 11 - Rastreabilidade através dos requisitos

Situação das tarefas		Gerenciamento de Requisitos		
Projeto	Requisitos	Casos de Uso	Rastreabilidade	
Administrador Pinhão				
Evolução do Documentador				
Scriba Pinhão - Evolução: Multi Repositório				
SISTEMA ALMIRANTE				
SISTEMA ALMIRANTE VERSÃO 2				
Sistema Sentinela Versão 2				

Fonte: Celepar, (2014)

Figura 12 - Tela de rastreamento

SubProjetos			
<input checked="" type="checkbox"/> Primeira Entrega			
Situação: Em Andamento			
Data Inicial Prevista:	04/05/2009	Data Início:	
Data Final Prevista:	14/08/2009	Data Fim:	
Casos De Uso:			
<input checked="" type="checkbox"/> UCS_PLANEJARINICIOPROJETO - (Complexidade: Complexo) - (Pacote:)			
Visualize detalhes do Caso de Uso			
Requisitos			
Requisito Vinculado ao Caso de Uso		Alocado	Associado ao Requisito
Planejamento/Replanejamento do Início do Projeto		SIM	
Tarefas Vinculadas ao Caso de Uso			
Fase	Tarefa	Situação	Executor
ANÁLISE	ESPECIFICAÇÃO DE CASO DE USO - UCS_PLANEJARINICIOPROJETO		Danielle Mayer
CONSTRUÇÃO	COMPONENTES CODIFICADOS - UCS_PLANEJARINICIOPROJETO		Marcos Antonio Chiarello
	TESTES DE INTEGRAÇÃO - UCS_PLANEJARINICIOPROJETO		Danielle Mayer
	TESTES DE INTEGRAÇÃO - UCS_PLANEJARINICIOPROJETO		Marcos Antonio Chiarello
PROJETO	ESPECIFICAÇÃO DE CASO DE USO - UCS_PLANEJARINICIOPROJETO		Danielle Mayer
	DESCRIÇÃO DE CASO DE TESTE DE INTEGRAÇÃO - UCS_PLANEJARINICIOPROJETO		Danielle Mayer
TRANSIÇÃO			

Fonte: Celepar, (2014)

4.1.3 IBM RationalRequisitePro

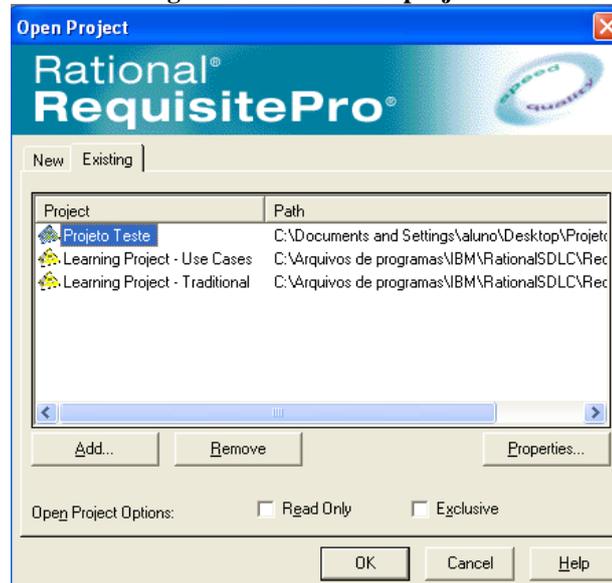
A IBM *RationalRequisitePro* é uma ferramenta que faz o gerenciamento de requisitos, permite que múltiplos usuários tenham acesso ao software e que estes façam o gerenciamento e o rastreamento das mudanças que ocorrem no projeto, também permite que o gerenciamento dos requisitos seja feito no Microsoft Word. Permite ao desenvolvedor fazer um controle das mudanças, proporcionando um maior conhecimento dos riscos do projeto, e, através da evolução do projeto, os requisitos são modificados e devidamente documentados. Também permite o acesso remoto aos usuários através do rastreamento dos requisitos para que possam fazer as modificações a distância, de uma forma que todos os usuários e administradores tenham o conhecimento das mudanças que ocorrem no projeto, proporcionando a sua visibilidade de uma maneira mais ampla (IBM CORPORATION SOFTWARE GROUP, 2014).

A seguir serão demonstradas as principais funcionalidades que a ferramenta disponibiliza.

- **Escolher e criar novos projetos** – na tela inicial da ferramenta deve ser escolhido em projeto deseja trabalhar, como demonstra a Figura 13; também é possível criar novos projetos, clicando na aba New, escolhendo o tipo do *template* do projeto, ou até

mesmo criando um *template* novo que se encaixe com as necessidades do projeto que será criado, ilustrado na Figura 14.

Figura 13 - Escolher o projeto



Fonte: elaborada pelo autor (2014)

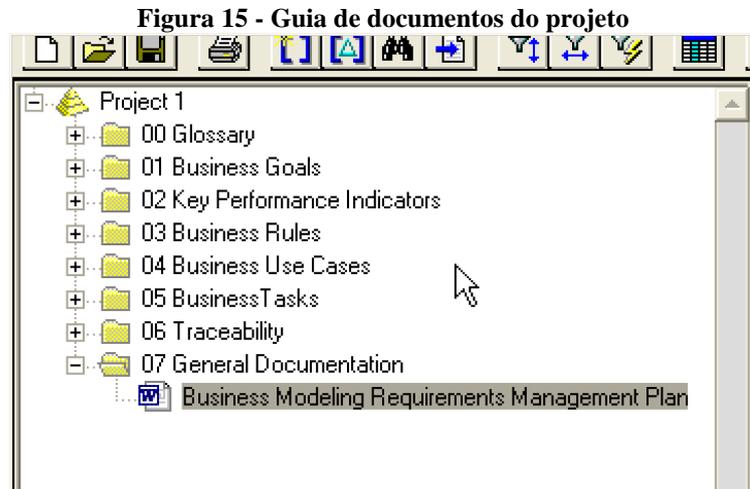
Figura 14 - Criando novo projeto



Fonte: elaborada pelo autor (2014)

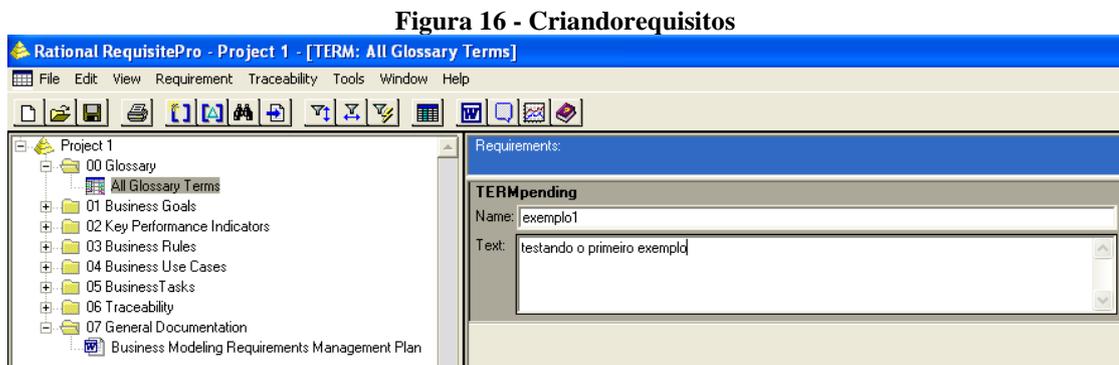
- **Documentos** – todos os documentos de edição criados para trabalhar são abertos com o MS Office Word. Na guia de documentos do projeto como apresenta a Figura 15. São encontrados todos os documentos que o *template* fornece para gerenciar o projeto, e também fornece o arquivo *Business Modeling Requirements Management Plan* (Plano de Gerenciamento de Requisitos Modelagem de Negócios), neste arquivo

encontram-se informações sobre como trabalhar com cada documento, como é estabelecida a rastreabilidade, e quais são os tipos de requisitos.



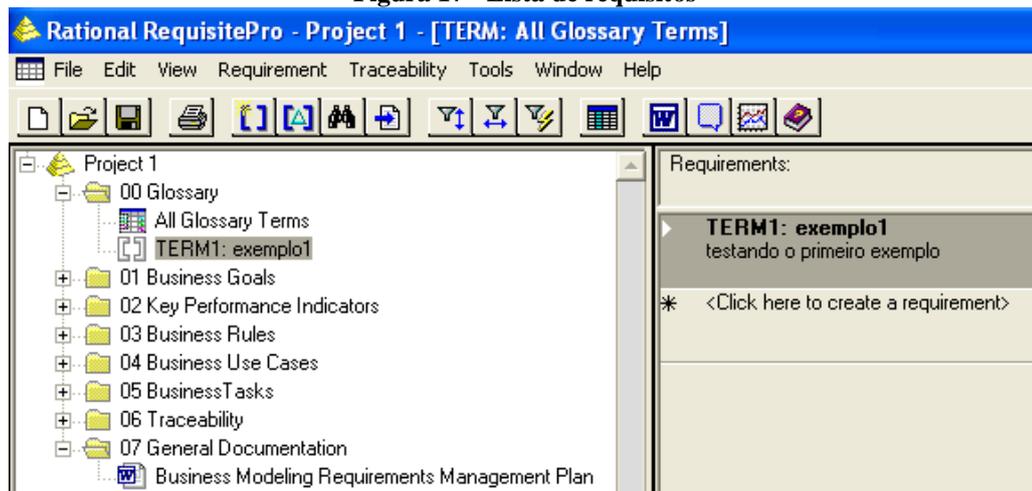
Fonte: elaborada pelo autor (2014)

- **Criar requisitos** – no momento de criar um requisito deve ser informado um nome para este requisito e uma breve descrição do mesmo, conforme a Figura 16. Após do requisito criado será possível acessá-los através de uma lista de requisitos como apresenta a Figura 17.



Fonte: elaborada pelo autor (2014)

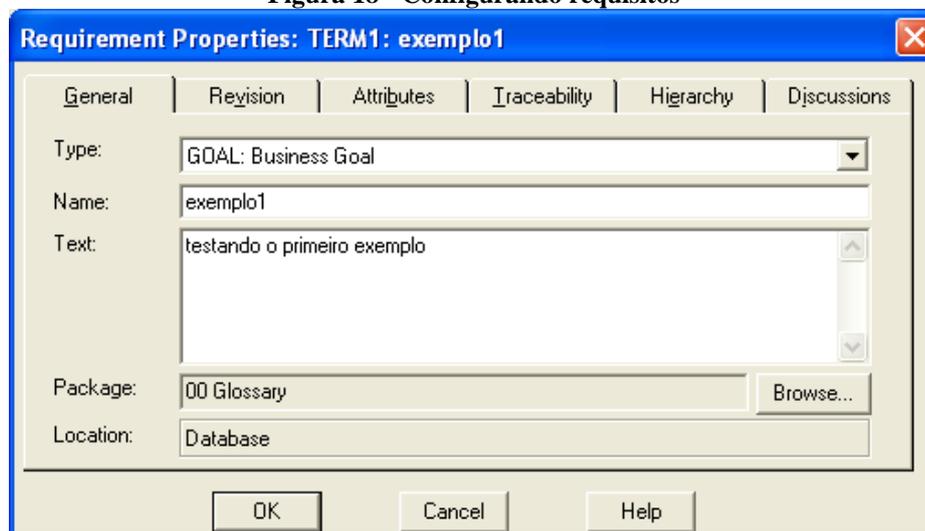
Figura 17 - Lista de requisitos



Fonte: elaborada pelo autor (2014)

- **Configurando requisitos** – após criar os requisitos, a ferramenta possibilita configurar estes requisitos, através de um *menu* de propriedades, no qual exibe uma tela para alterar dados referentes aos requisitos, como expõem a Figura 18. Pode ser inserindo informações como, tipo, o pacote a que se refere o requisito, atributos, hierarquia, localização, todas essas informações para integrá-lo com o projeto.

Figura 18 - Configurando requisitos



Fonte: elaborada pelo autor (2014)

4.1.4 *Serena Dimension RM*

O *Serena Dimensions RM* é um software caracterizado pela solução de gerenciamento requisitos e fornece vários recursos que abrangem todo o desenvolvimento do sistema que será criado. A partir da definição dos requisitos e através da web, permite que os

usuários possam criar, editar, comentar e gerenciar simultaneamente os requisitos. Trabalha com o *Microsoft Office Word*, que permite fazer as alterações off-line, sendo que também importa os documentos Word para atualizar no sistema, sem que seja necessário refazer as alterações online. O Serena também fornece a rastreabilidade e a reutilização de requisitos, controla de uma maneira mais eficaz os requisitos e permite que um requisito já definido possa ser reutilizado, isso ocorre porque os requisitos são armazenados em um único repositório central, o que permite a fácil localização dos requisitos já existentes (SERENA, 2014).

Algumas funcionalidades do *Serena Dimensions RM* estão apresentadas a seguir:

- **Gerenciar requisitos** – a Figura 19 mostra a funcionalidade de gerenciamento dos requisitos que a ferramenta disponibiliza. O gerenciamento é realizado através da web, eliminando várias versões em *desktop* ou dispositivos do usuário. O usuário pode editar simultaneamente requisitos, comentários, criar e votar em enquetes que são criadas para auxiliar no gerenciamento, e gerenciar mudanças, toda essa funcionalidade é realizada em um repositório central online.

Figura 19 - Gerenciamento de Requisitos

Option	% of Votes	Votes
Yes	50%	1
No	0%	0
Need more information	50%	1
Total:		2

Fonte: Serena, (2014)

- **Rastreabilidade** – a ferramenta disponibiliza aos seus usuários a rastreabilidade completa dos requisitos de negócio, fornecendo uma visão imediata de status de

exigência, as relações da exigência, a mudança do requisito, e as dependências do requisito, e permite acompanhar as necessidades durante todo o tempo da implementação. A Figura 20 demonstra essa funcionalidade.

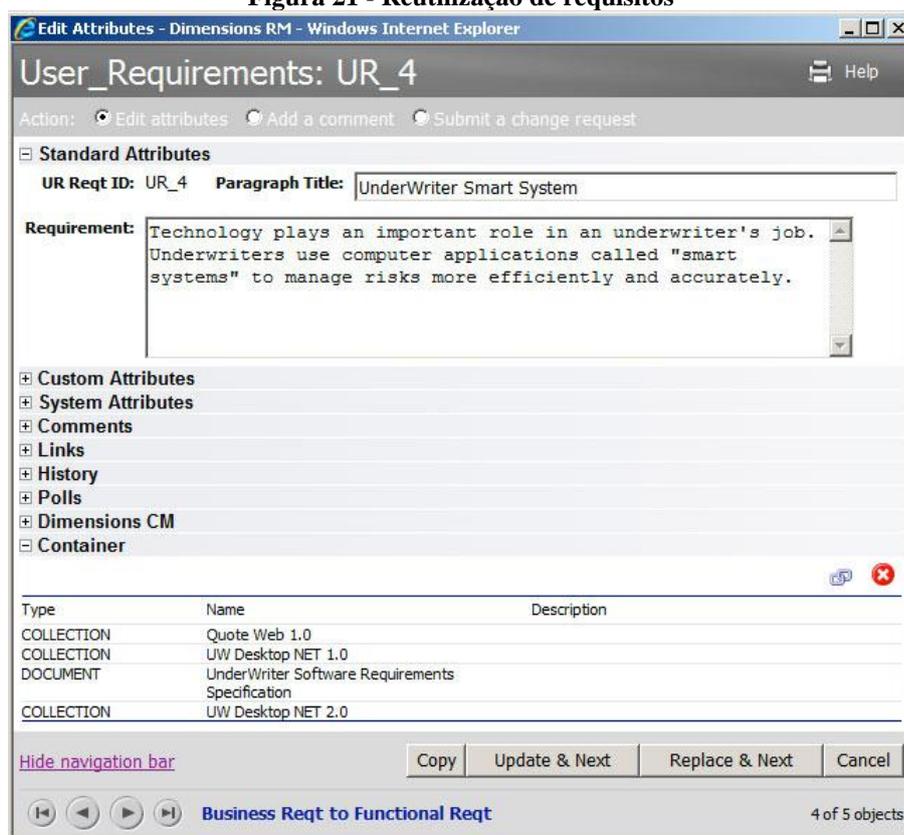
Figura 20 - Rastreabilidade

The screenshot displays the 'Marketing Requirements Trace' view in the Serena Dimensions RM application. The interface is organized into three main columns: Marketing Requirements, Product Requirements, and Component Requirements. Each column lists requirements with their respective IDs and titles. The Marketing Requirements column includes items such as 'MRKT_000001: Ephoto will be an online photo album' and 'MRKT_000002: Support Advantix formats'. The Product Requirements column lists 'PROD_000001: Try to use Td/Tk' and 'PROD_000003: Default window size'. The Component Requirements column shows 'COMP_000001: Utilize Td/Tk' and 'COMP_000017: Windows Desktop used'. The interface also features a navigation menu at the top, a search bar, and a footer with database and project information.

Fonte: Serena, (2014)

- **Reutilizar requisitos** – o software *Serena Dimensions RM* disponibiliza a reutilização dos requisitos, pois estes são armazenados em um repositório central auxiliando a identificar e reutilizar requisitos já definidos anteriormente, ou seja, quando o usuário está criando algo que existe no sistema. A Figura 21 demonstra as funcionalidades de reutilização de requisitos que a ferramenta dispõe.

Figura 21 - Reutilização de requisitos



Fonte: Serena, (2014)

4.2 ANÁLISE DAS FERRAMENTAS DE GERENCIAMENTO DE REQUISITOS DE SOFTWARE

Nas ferramentas detalhadas nesta pesquisa, foram encontrados pontos positivos e negativos que os usuários podem se deparar no momento da instalação ou até mesmo durante a utilização destas ferramentas. A seguir serão elencados alguns desses pontos:

Ao analisar a ferramenta Plandora identificaram-se pontos positivos como:

- É uma ferramenta livre com interface de usuário amigável e fácil utilização;
- O líder do projeto tem total acesso a toda a tarefa que está sendo executada no projeto;
- Os usuários podem trabalhar em suas tarefas e informar como está sendo executada a tarefa e quando esta tarefa foi realmente finalizada;
- A ferramenta também trabalha com o método ágil *Scrum* permitindo maior interação entre os desenvolvedores;
- Permite gerenciar riscos e criar diferentes formatos de relatórios para melhor visualizar o desempenho do projeto e proporcionar histórico dessas tarefas realizadas nos projetos.

Os pontos negativos encontrados na ferramenta Plandora foram:

- É uma ferramenta web de código livre, que trabalha com a arquitetura cliente-servidor, exigindo para sua instalação um servidor web e um banco de dados, no site da ferramenta são sugeridos, o *Tomcat* como servidor web e o MySQL como banco de dados, porém deixa claro que podem ser utilizados outros servidores web, desde que sejam compatíveis, e que a ferramenta somente foi testada com a utilização do MySQL;
- Muitas funcionalidades que a ferramenta Plandora disponibiliza são executadas somente pelo líder do projeto, por exemplo, somente o líder do projeto tem a possibilidade de delegar tarefas aos demais desenvolvedores, ou seja, cada nova solicitação referente ao projeto, passa pela aprovação do líder, sendo que esse processo poderia ser de forma mais direta;
- Outros usuários alteram dados apenas de tarefas de sua responsabilidade sem ter acesso às demais atividades, sendo que diferentes usuários possam fazer parte de uma mesma equipe de desenvolvimento e necessitar trabalhar nas tarefas de outros companheiros de equipe.

Na ferramenta Almirante, após detalhada as suas funcionalidades foram encontrados os seguintes pontos positivos:

- Ferramenta completa no registro de requisitos, na qual é possível detalhar especificações e restrições no momento de registro dos requisitos;
- É possível cadastrar os casos de uso relacionando os requisitos que são atendidos pelos casos de uso, facilitando a visualização das funcionalidades que o sistema em desenvolvimento fornecerá aos seus usuários;
- Permite realizar iterações através dos casos de uso cadastrados na ferramenta;
- E rastrear os requisitos de diferentes maneiras, sendo possível realizar a rastreabilidade através dos casos de uso ou até mesmo apenas de um requisito em específico.

A ferramenta Almirante apresenta os seguintes pontos negativos:

- Esta ferramenta é de difícil acesso, pois não há informação de como obtê-la. No próprio site do desenvolvedor, possui pouca informação da mesma. Foram tentadas formas de contato para se ter alguma informação sobre como adquirir a ferramenta e não foi obtido respostas até a conclusão do projeto.

- Dentre os problemas encontrados durante a análise da ferramenta, destacam-se as iterações entre os requisitos que a ferramenta disponibiliza, pois a iteração só se dá através de casos de uso cadastrados, se o usuário do sistema desejar realizar as iterações apenas através dos requisitos sem necessariamente ligar a um caso de uso específico, não é possível o fazer.

A ferramenta *IBM RationalRequisitePro* foram analisada e encontrados os seguintes pontos positivos:

- Disponibiliza a qualquer usuário criar projetos no qual deseja trabalhar, assim como escolher em qual projeto deseja trabalhar;
- Todos os documentos de trabalho que a ferramenta disponibiliza são no formato do Microsoft Office Word, possibilitando ao usuário do sistema trabalhar no projeto sem necessariamente estar acessado o sistema;
- Permite criar os requisitos especificando como será as restrições e detalhes que este requisito possuirá.

E os pontos negativos encontrados na ferramenta são:

- A *IBM RationalRequisitePro* é uma ferramenta paga, cujo valor varia de 1290,00 USD a 5330,00 USD;
- A ferramenta não permite que o usuário trabalhe em mais de um projeto simultaneamente, pois no momento em que inicia sua utilização é solicitado ao usuário que indique em qual projeto deseja trabalhar, ou se deseja criar um novo projeto, o que acaba não proporcionando uma interface amigável ao usuário;
- Outro ponto importante de salientar sobre a ferramenta, é na tela de inserção de novos requisitos, nesta tela, há apenas as opções de nome e descrição do requisito, sendo que as demais configurações que são importantes em um requisito são acessadas posteriormente ao momento de inserção, sendo assim toda a vez que é criado um novo requisito o usuário deve lembrar que tem de configurá-los, e o sistema não deixa isso claro;
- A ferramenta também não possibilita inserir o usuário que está realizando a inserção ou configuração de um requisito, sendo assim é possível que prejudique a identificação de quem está realizando as alterações no projeto.

Na ferramenta *Serena Dimensions* RM foram encontrados os seguintes pontos positivos:

- Todo o gerenciamento de requisitos é realizado via web permitindo que os usuários interajam de forma simultânea na edição de dados;
- Permite rastrear os requisitos proporcionando maior controle das mudanças no decorrer da implementação do sistema;
- Possibilita ao usuário reutilizar requisitos já cadastrados, pois a ferramenta disponibiliza um único repositório central onde todos os requisitos então cadastrados, e cada vez que o usuário for cadastrar um requisito já existente ou com características semelhantes, é apresentada a sugestão de utilizar o requisito cadastrado, mas o usuário decide se irá utilizar o requisito que está cadastrado ou se prefere cadastrar outro com especificações ou restrições diferentes do cadastrado anteriormente.

Pontos negativos encontrados na análise da ferramenta *Serena Dimensions* RM:

- Sua licença é paga, não foi informado valores da licença de utilização;
- Para acessar informações mais detalhadas sobre a ferramenta o usuário que deseja adquiri-la deve acessar um sistema o qual permite comunicação direta aos seus clientes, e possuir um email corporativo, pois os emails com contas como *gmail*, *hotmail* e *Yahoo* não são aceitos pelo sistema.
- A ferramenta quer eliminar a grande quantidade de versões que os usuários venham possuir em seus desktops, porém permite que os usuários trabalhem *off-line* através do Microsoft Office Word.

Através da análise das ferramentas de gerenciamento de requisito e relatando seus pontos positivos e negativos a ferramenta com melhores características para gestão de requisitos em empresas de pequeno porte de desenvolvimento de software é a Plandora, pois além de proporcionar o gerenciamento de requisitos, a ferramenta também disponibiliza gerenciar o projeto como um todo. Também por ser uma ferramenta de livre acesso no qual os desenvolvedores podem adequar a mesma da maneira que melhor se encaixar com seu modo de trabalhar.

5 CONSIDERAÇÕES FINAIS

A utilização de ferramentas de gerenciamento de projetos tem se tornado cada vez mais corriqueira, e encontrar ferramentas que proporcionam este gerenciamento de maneira eficaz e que se encaixe completamente com a maneira de trabalho utilizada pelos desenvolvedores é algo difícil de encontrar. Cada empresa de desenvolvimento adota a metodologia que melhor se encaixa com o andamento do trabalho no seu cotidiano. Portanto, torna-se complexo encontrar ferramentas que suprem da melhor forma as necessidades dos desenvolvedores.

As ferramentas de gerenciamento de requisitos disponíveis se encaixam em parte com a necessidade dos seus interessados, portanto a necessidade de um software que consiga atender as diferentes necessidades dos usuários, e que estes possam adaptá-lo de maneira a fornecer funcionalidades que melhor se encaixe com seus projetos.

Um sistema de gerenciamento de requisitos de plataforma livre pode suprir estas necessidades, voltado para as especificações e restrições dos requisitos, na qual os desenvolvedores trabalharam em paralelo com o projeto de software controlando e gerenciando as mudanças necessárias.

Entendendo que cada organização tem sua metodologia de trabalho, a ferramenta Plandora é a que melhor se encaixa com suas características a empresas de pequeno porte, possuindo funcionalidades que representam várias etapas de desenvolvimento de um software, tornando-se uma ferramenta dois em um, gerenciando requisitos e o com funcionalidades que também gerenciam o projeto em si, e sendo uma ferramenta *open source* proporciona maiores facilidades em sua utilização.

5.1 TRABALHOS FUTUROS

Como trabalhos futuros sugere-se a análise, projeto e implementação de um software de gerenciamento de requisitos, visando extinguir os pontos negativos encontrados nas ferramentas e apresentados neste trabalho. O software a ser implementado deve ser *open source*, de licença GNU *General Public License* (GPL) que de acordo com o GNU *Operation System* (2007) é uma licença *copyleft* livre para softwares e outros tipos de obras, garantindo a liberdade de compartilhar e modificar todas as versões de um programa.

REFERÊNCIAS

ACHRAFI, Rabi; LAM, Dora. *Requirements Tolls*. Disponível em: <<http://volere.co.uk/tools.htm>> Acesso em 25 jun. 2014.

Celepar – Companhia de Informática do Paraná – Curitiba. Disponível em: <<http://www.frameworkpinhao.pr.gov.br/modules/conteudo/conteudo.php?conteudo=14>> Acesso em: 03 jun. 2014.

GNU Operation System – GPL Free Software – GNU General Public License. Disponível em: <<http://www.gnu.org/copyleft/gpl.html>> Acesso em 26 jun. 2014.

IBM Corporation Software Group. IBM Rational RequisitePro. Disponível em: <<http://www-03.ibm.com/software/products/en>> Acesso em: 30 mai. 2014

OPEN SOURCE. Open Source Initiative. The open source definition. Disponível em: <<http://opensource.org/osd>> Acesso em jun. 2014

PAULA FILHO, Wilson de Pádua. *Engenharia de Software Fundamentos Métodos e Padrões*. 3ed. Rio de Janeiro: Editora LTC – Livro Técnicos e Científicos Editora Ltda, Integrante GEN – Grupo Editorial Nacional, 2009.

Plandora Project – América – Campinas, 2004-2012. Disponível em: <<http://www.plandora.org/>> Acesso em: 20 mai. 2014.

PRESSMAN, Roger. S. *Engenharia de Software*. 7. ed. São Paulo: Editora McGraw-Hill, 2011.

ROCCO, Giovanni Ely. *Um Modelo de Estruturação de Requisitos para o Método Fusion*. Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul, Programa de Pós Graduação em Computação. Porto Alegre, 2001. Disponível em: <<http://www.lume.ufrgs.br/bitstream/handle/10183/3109/000332308.pdf?sequence=1>> Acesso em: 20 abr. 2014.

SANTOS, José Henrique Amaral dos. *Gerência de mudanças de requisitos: uma proposta de aplicação a um estudo de caso*. Universidade Federal do Rio Grande do Sul, Programa de Graduação e Pós-Graduação em Computação Porto Alegre, 2004. Disponível em: <<http://www.lume.ufrgs.br/bitstream/handle/10183/4118/000452937.pdf?sequence=1>> Acesso em: 17 abr. 2014.

Serena Dimensions RM. 2014. Disponível em: <<http://www.serena.com/index.php/en/products/dimensions-rm/>> Acesso em: 25 jun. 2014.

SOMMERVILLE, Ian. *Engenharia de Software*. 9ed. São Paulo: Editora Prentice Hall, 2011.

YOURDON, Edward. *Análise Estruturada Moderna*. 3 ed. Rio de Janeiro: Elsevier Editora Ltda, 1990.