



INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Gestão de Requisitos

Aplicação Colaborativa para a Gestão de Requisitos

Vasco Miguel Rodrigues

Dissertação para obtenção do grau de Mestre em
Mestrado em Engenharia Informática e de Computadores

Júri

Presidente: Prof. Doutor Pedro Manuel Moreira Vaz Antunes de Sousa

Orientador: Prof^a Doutora Maria do Rosário Gomes Osório Bernardo Ponces
de Carvalho

Vogais: Prof. Doutor Alberto Manuel Rodrigues da Silva

Novembro de 2008



Gestão de Requisitos

Vasco Miguel Rodrigues

Agradecimentos

Começo por agradecer à orientadora Prof. Maria Rosário Bernardo, por ter aceitado este desafio, pelos diversos conselhos transmitidos, ligação estabelecida com a organização onde foi desenvolvido o sistema apresentado e pelo acompanhamento fulcral na fase final de revisão e entrega da tese.

Gostaria também de agradecer ao professor José Borbinha pela total disponibilidade em me facultar o empréstimo de algumas das referências bibliográficas necessárias para o desenvolvimento cientificamente fundamentado de temáticas abordadas na tese e ainda ao Professor Miguel Mira da Silva pelo seus comentários construtivos que me permitiram melhorar o presente documento.

Quero ainda manifestar uma palavra de agradecimento ao professor José Silva Lopes pelo apoio em algumas áreas científicas e matemáticas, sem as quais não poderia ter dado início ao desenvolvimento da tese.

Pelo acompanhamento, flexibilidade e condições de trabalho oferecidas pela NOESIS, SA, gostaria de focar algumas pessoas que contribuíram directamente para o desenvolvimento desta dissertação de tese, são elas o Director de Quality Managment Eng. Nelson Ricardo dos Reis Tomás Pereira, pelo seu total apoio e partilha de experiências da área, ao Eng. Pedro Filipe Almeida Gonçalves pela colaboração prestada, assim como a todos os restantes elementos com os quais privei durante todo o processo.

Ainda uma palavra especial ao vários colegas de curso e amigos que contribuíram activamente com palavras de apoio e incentivo durante o desenrolar do trabalho. De sublinhar o apoio da minha colega Sofia Flores pelas críticas construtivas durante a redacção deste documento que muito contribuíram para a melhoria de qualidade deste.

Por fim gostaria de agradecer à minha família pelo apoio prestado e pela confiança e força transmitida.

Resumo

O presente documento resulta de um trabalho de pesquisa efectuado no contexto da tese de mestrado do autor, que tem por título: Gestão de requisitos. Esta tese de mestrado incide no desenvolvimento de uma aplicação que permita dar suporte à gestão de requisitos de projectos informáticos. Este tipo de aplicações enquadra-se numa categoria de ferramentas a que se denomina *case tools* (*computer-aided software engineering*).

As aplicações de gestão de requisitos têm como principal objectivo assistir e dar suporte ao desenvolvimento e manutenção de software. Dado este contexto, este artigo aborda os principais processos da engenharia de requisitos, de uma forma mais detalhada, a fase de gestão de requisitos. Além disso é realizada uma análise crítica a algumas das ferramentas mais utilizadas nesta área.

Partindo deste conhecimento são detectados uma série de problemas e posteriormente apresentado uma proposta para os resolver.

O produto final desta tese é uma aplicação *web* de baixo custo, que permite a gestão dos requisitos durante todo o ciclo de desenvolvimento de um projecto e consequentemente responder aos diversos desafios colocados por ferramentas deste tipo. O sistema desenvolvido tem como base uma proposta realizada pelo autor, que assenta numa solução orientada ao trabalho colaborativo onde é possível não só gerir os requisitos, mas também utilizar o sistema como um meio de aprovação de requisitos e pedidos de alteração a requisitos, através de um sistema de aprovações incorporado no sistema.

Palavras-chave

Engenharia de requisitos, Gestão de requisitos, Trabalho colaborativo, Gestão de alterações, Aplicações *Web*

Abstract

This document is the result of a research done in the context of a master's thesis of the author, which has the title: Requirements management. This master's thesis focuses on developing an application which can support the requirement management for software projects. Such applications fall into a category of tools called CASE tools (*computer-aided software engineering*).

Requirement Management tools are designed primarily to assist and provide support to software development and maintenance. Given this background, this paper addresses the key processes of requirements engineering, in a more detailed way the requirement management phase. Furthermore it is written a critical analysis to some of the most used tools in this area.

From this knowledge is discovered a number of problems and subsequently submitted a proposal that can help to resolve them.

The final product of this thesis is a low cost web application, which allows the management of requirements throughout the project life cycle and therefore meets the various challenges posed by such tools. The system developed is based in the proposal produced by the author, cooperative work oriented, where you can not only manage the requirements, but also use the system as a way to approve requirements and requests for change, through an approval system incorporated in the web application.

Índice

AGRADECIMENTOS	I
RESUMO	III
PALAVRAS-CHAVE	III
ABSTRACT	V
ÍNDICE.....	VII
LISTA DE SIGLAS E ABREVIATURAS.....	IX
ÍNDICE DE FIGURAS	XI
ÍNDICE DAS TABELAS.....	XIII
1 INTRODUÇÃO	1
1.1 CONTEXTO.....	1
1.2 PROBLEMA.....	2
1.3 PROPOSTA.....	2
1.4 ESTRUTURA DA DISSERTAÇÃO	4
2 ESTADO DA ARTE	7
2.1 INTRODUÇÃO.....	7
2.2 ENGENHARIA DE REQUISITOS.....	7
2.2.1 <i>O Processo da Engenharia de requisitos</i>	7
2.3 GESTÃO DE REQUISITOS.....	10
2.3.1 <i>Alterações dos requisitos</i>	11
2.3.2 <i>Change management</i>	12
2.3.3 <i>Rastreabilidade de requisitos</i>	14
2.3.4 <i>O suporte de ferramentas</i>	15
2.4 ANÁLISE DE FERRAMENTAS.....	17
2.4.1 <i>Open Source Requirements Management Tool</i>	18
2.4.2 <i>RTH</i>	18
2.4.3 <i>Requisite Pro</i>	19
2.4.4 <i>Telelogic DOORS Fastrack</i>	22
2.4.5 <i>Análise comparativa</i>	22
2.5 CONCLUSÃO	23
3 DESENVOLVIMENTO DA SOLUÇÃO.....	25
3.1 PROCESSO DE DESENVOLVIMENTO	25
3.2 PLANEAMENTO	26
3.3 ANÁLISE.....	27
3.3.1 <i>Diagramas de Casos de Utilização</i>	27

3.3.2	<i>Requisitos da Aplicação</i>	35
3.3.3	<i>Protótipo Funcional</i>	42
3.4	DESENHO	46
3.4.1	<i>Diagrama de Actividades</i>	46
3.4.2	<i>Modelo de Domínio</i>	50
3.4.3	<i>Diagrama de estados</i>	51
4	PLATAFORMA TECNOLÓGICA	55
4.1	INTRODUÇÃO	55
4.2	ARQUITECTURA DE SOFTWARE	55
4.3	TECNOLOGIA	57
4.3.1	<i>Camada de Apresentação</i>	57
4.3.2	<i>Camada de Negócio</i>	58
4.3.3	<i>Camada de Persistência</i>	58
4.4	CONCLUSÃO	59
5	TESTES REALIZADOS	61
5.1	INTRODUÇÃO	61
5.2	METODOLOGIA ADOPTADA	61
6	RESULTADOS OBTIDOS	63
6.1	APRESENTAÇÃO DA SOLUÇÃO FINAL	63
6.1.1	<i>Resumo do Projecto e sistema de aprovações</i>	63
6.1.2	<i>Matriz de Rastreabilidade</i>	64
6.1.3	<i>Pedido de alteração a requisitos</i>	65
6.1.4	<i>Comparação de Baselines</i>	66
6.1.5	<i>Definição e especificação do Requisito</i>	67
6.2	OPINIÕES DOS UTILIZADORES	68
6.3	COMPARAÇÃO COM OUTRAS FERRAMENTAS ANALISADAS	69
7	CONCLUSÃO	71
7.1	TRABALHO FUTURO	72
8	REFERÊNCIAS	ERRO! MARCADOR NÃO DEFINIDO.
9	ANEXOS	75
9.1	A.1: MODELO DA BASE DE DADOS	75
9.2	A.2: QUESTIONÁRIO REALIZADO	76

Lista de siglas e abreviaturas

ER – Engenharia de Requisitos

CASE - *Computer-Aided Software Engineering*

SOA - *Service Oriented Architecture*

SI – Sistemas de Informação

SCCM – *Software Change and Configuration Management*

AJAX – *Asynchronous JavaScript and XML*

XML – *Extended Markup Language*

XSD - *XML Schema Definition*

ADO - *ActiveX Data Objects*

RM – *Requirement Management*

QA – *Quality Assurance*

PHP- *Hypertext Preprocessor*

CI – *Configuration Item*

Índice de figuras

FIGURA 1: <i>INPUTS</i> E <i>OUTPUTS</i> DE UM PROCESSO DE ENGENHARIA DE REQUISITOS (2).....	8
FIGURA 2: MODELO DE ACTIVIDADES DO PROCESSO DE ER.....	9
FIGURA 3: CICLO DE ACTIVIDADES DA ENGENHARIA DE REQUISITOS (3).....	10
FIGURA 4: ESTADOS NUM PROCESSO DE <i>CHANGE MANAGEMENT</i> . (2).....	13
FIGURA 5:PROCESSO GENÉRICO DE ANÁLISE DA ALTERAÇÃO E DE CUSTO. (2).....	13
FIGURA 6: RASTREABILIDADE DE REQUISITOS E TIPOS DE ANÁLISE SUPORTADOS. (4)	15
FIGURA 7:INTEGRAÇÕES COM OUTRAS FERRAMENTAS CASE. (11)	17
FIGURA 8: PROCESSO EM CASCATA REVISTO.....	26
FIGURA 9: PLANO DO PROJECTO DESENVOLVIDO	27
FIGURA 10: DIAGRAMA DE ACTORES E OS SEUS PAPÉIS	28
FIGURA 11: DIAGRAMA DE CASOS DE UTILIZAÇÃO DO UTILIZADOR DO SISTEMA	29
FIGURA 12: DIAGRAMA DE CASOS DE UTILIZAÇÃO DO UTILIZADOR DO SISTEMA (PARTICIPAR EM DISCUSSÕES)	29
FIGURA 13: DIAGRAMA DE CASOS DE UTILIZAÇÃO DO ANALISTA	30
FIGURA 14: DIAGRAMA DE CASOS DE UTILIZAÇÃO DO ANALISTA (GERIR <i>BASELINES</i>)	31
FIGURA 15: DIAGRAMA DE CASOS DE UTILIZAÇÃO DO GESTOR DO PROJECTO.....	31
FIGURA 16: DIAGRAMA DE CASOS DE UTILIZAÇÃO DO <i>STAKEHOLDER</i>	32
FIGURA 17: DIAGRAMA DE CASOS DE UTILIZAÇÃO DO ADMINISTRADOR	32
FIGURA 18: DIAGRAMA DE CASOS DE UTILIZAÇÃO DO ADMINISTRADOR (GERIR PROJECTOS)	33
FIGURA 19: DIAGRAMA DE CASOS DE UTILIZAÇÃO DO ADMINISTRADOR (GERIR TIPOS DE REQUISITOS).....	33
FIGURA 20: DIAGRAMA DE CASOS DE UTILIZAÇÃO DO ADMINISTRADOR (GERIR ACESSOS).....	34
FIGURA 21: DIAGRAMA DE CASOS DE UTILIZAÇÃO DO ADMINISTRADOR (GERIR ATRIBUTOS)	34
FIGURA 22: DIAGRAMA DE CASOS DE UTILIZAÇÃO DA PLATAFORMA RM.....	35
FIGURA 23: ESTRUTURA DOS REQUISITOS DO SISTEMA.	36
FIGURA 24: PÁGINA INICIAL DO PROTÓTIPO FINAL DO SISTEMA.....	43
FIGURA 25: PÁGINA DE CONSULTA E EDIÇÃO DE UM REQUISITO DO PROTÓTIPO FINAL.	44
FIGURA 26: PÁGINA DA MATRIZ DE RASTREABILIDADE ENTRE REQUISITOS DO PROTÓTIPO.	44
FIGURA 27: PÁGINA DE ADMINISTRAÇÃO DE UM PROJECTO DO PROTÓTIPO.....	45
FIGURA 28: PROCESSO DE APROVAÇÃO DE UM REQUISITO.	47
FIGURA 29: PROCESSO DE GERIR VOTAÇÃO.	48
FIGURA 30: PROCESSO DE ALTERAR UM REQUISITO.	49
FIGURA 31: PROCESSO PARA EFECTUAR UM PEDIDO DE ALTERAÇÃO.	50
FIGURA 32: MODELO DE DOMÍNIO DO SISTEMA DE GESTÃO DE REQUISITOS.....	51
FIGURA 33: DIAGRAMA DE ESTADOS DE UM REQUISITO.....	52
FIGURA 34: ARQUITECTURA DE SOFTWARE DO SISTEMA IMPLEMENTADO	57
FIGURA 35: MODELO AJAX DE APLICAÇÕES WEB (22).	58
FIGURA 36: PÁGINA INICIAL DA APLICAÇÃO.	64

FIGURA 37: MATRIZ DE RASTREABILIDADE DO SISTEMA DESENVOLVIDO.	65
FIGURA 38: FORMULÁRIO DE PEDIDO DE ALTERAÇÕES A UM OU MAIS REQUISITOS.	66
FIGURA 39: PÁGINA DO SISTEMA QUE APRESENTA A COMPARAÇÃO ENTRE DUAS <i>BASELINES</i>	67
FIGURA 40: PAINEL DE CONSULTA E ALTERAÇÃO DE UM REQUISITO.	68

Índice das Tabelas

TABELA 1:PRINCIPAIS FACTORES PARA A MUDANÇA DE REQUISITOS. (2)	12
TABELA 2: COMPARAÇÃO FUNCIONAL DAS FERRAMENTAS ANALISADAS.	23
TABELA 3: REQUISITOS DE NEGÓCIO DA GESTÃO DE UTILIZADORES.	37
TABELA 4: REQUISITOS DE NEGÓCIO DA GESTÃO DE ACESSOS.....	37
TABELA 5: REQUISITOS DE NEGÓCIO PARA A GESTÃO DE PROJECTOS	38
TABELA 6: REQUISITOS DE NEGÓCIO PARA A GESTÃO DOS REQUISITOS.....	39
TABELA 7: REQUISITOS DE NEGÓCIO PARA A INTEGRAÇÃO COM FERRAMENTAS DE GESTÃO TESTES.	40
TABELA 8: REQUISITOS DE NEGÓCIO PARA A GERAÇÃO DE DOCUMENTAÇÃO.....	41
TABELA 9: REQUISITOS DE NEGÓCIO PARA A INTEGRAÇÃO COM FERRAMENTAS SCCM.....	41
TABELA 10: COMPARAÇÃO FUNCIONAL DA SOLUÇÃO DESENVOLVIDA COM OUTRAS FERRAMENTAS DO MERCADO.....	70

1 Introdução

Este capítulo introdutório serve para situar o leitor na temática relacionada com a presente dissertação de mestrado. Em primeiro lugar é realizada uma contextualização do trabalho realizado, onde é enquadrado o trabalho com a realidade actual, a que utilizadores e tipo de tarefas se destinam. Seguidamente é descrito o problema detectado após o estudo dos temas que a tese aborda e apresentada uma proposta, que segundo o ponto de vista do autor, permite ajudar a resolver este mesmo problema. Por fim é explicada a estrutura do presente documento, para que o leitor consiga efectuar uma leitura interessante e o mais proveitosa possível, com base na forma como a dissertação foi escrita.

1.1 Contexto

A importância de um sistema informático responder aos problemas e expectativas das organizações é actualmente um factor de extrema importância. Para alcançar este objectivo é necessário que exista uma sensibilização de toda a equipa que intervêm no processo de desenvolvimento, para importância dos requisitos de um sistema, pois é com base nestes que é feita a ponte entre as necessidades das organizações e a equipa de desenvolvimento. Os requisitos funcionam assim, como um guia para gestores de projecto, programadores, técnicos de sistemas e elementos de qualidade durante o desenrolar do projecto.

Dada a importância dos requisitos de um projecto informático é necessário que as empresas que se dedicam ao desenvolvimento de software procurem garantir a qualidade dos requisitos ao longo de todo o processo de desenvolvimento. Para isso deve ser adoptado um processo de engenharia de requisitos e realizada uma eficiente gestão dos requisitos durante o tempo de duração do produto informático. As ferramentas de gestão de requisitos assumem por isso um papel fulcral, pois permitem suportar grande parte das tarefas necessárias para uma eficiente gestão dos requisitos.

Esta dissertação de mestrado, realizada na área de qualidade da Noesis S.A., vai incidir sobre a área de gestão de requisitos. É realizado um estudo na área onde é identificado as problemáticas associadas à gestão de requisitos e em que medida uma ferramenta pode auxiliar esta tarefa. O objectivo final desta tese é a apresentação de uma solução que dê suporte à definição e gestão dos requisitos segundo as melhores práticas e procedimentos da área de qualidade. A Noesis vem contribuir com a sua vasta experiencia em projectos nesta área fornecendo um conhecimento prático. Esta informação é de grande importância para orientar o estudo na direcção certa, permitindo identificar melhor o possível problema e uma solução que melhor se adequa.

1.2 Problema

Esta tese parte da percepção de que as ferramentas de gestão de requisitos disponíveis no mercado não cobrem todas as funcionalidades necessárias para uma célere respostas às constantes necessidades de mudança organizacional que, necessariamente, se reflectem na indefinição e elevada taxa de alteração dos requisitos de sistemas de informação, bem como na urgência na obtenção de resultados. Se esta percepção tem fundamento, então existe a necessidade de desenvolver aplicações capazes de suportarem uma gestão de requisitos eficiente, com baixos custos e de fácil utilização.

Para provar que este problema é real o trabalho iniciou-se com o estudo realizado sobre a temática de gestão de requisitos, os processos que engloba, a sua elevada importância para o sucesso de um projecto e tendo ainda como base a análise de algumas das ferramentas utilizadas actualmente, foi possível detectar algumas lacunas e pontos de melhoria, que ainda no entender do autor, não permite a gestão mais eficiente dos requisitos, durante todo o ciclo de vida de um projecto informático.

Os principais problemas detectados neste tipo de ferramentas CASE são os seguintes:

- Os custos associados à compra de licenças, tempo de aprendizagem e formação de utilizadores são demasiado elevados, impossibilitando em grande parte dos casos a sua utilização em organizações de média e baixa dimensão.
- Ferramentas centralizadas não são direccionadas para projectos em que as equipas, *stakeholders* e utilizadores se encontram dispersos geograficamente;
- Falta de comunicação entre os diversos intervenientes no projecto, é a principal razão pela qual alguns dos requisitos, não correspondem às necessidades dos utilizadores;
- Pouca flexibilidade de integração com outras ferramentas CASE;
- Não procuram resolver uma forma de aprovação electrónica de requisitos e de pedidos de alterações ao projecto;
- Ferramentas com baixa usabilidade obrigam a tempos de aprendizagem muito elevados;
- Organizações necessitam de adaptar os seus processos e terminologias à ferramenta de gestão.

Os pontos de melhoria e problemas apontados não significam que a abordagem das soluções apresentadas pelo universo de ferramentas de gestão de requisitos esteja incorrecto, pretende-se no entanto reflectir sobre os problemas encontrados e com base nestes propor uma solução segundo um ponto de vista diferente

1.3 Proposta

A ferramenta a ser desenvolvida deve suportar as principais tarefas da disciplina de gestão de requisitos, desde o registo de alteração de requisitos, histórico de acções, suporte de diversos tipos e atributos de requisitos, criação de *baselines* e relatórios, definição de rastreabilidade e antevisão do impacto de alterações.

Um projecto informático interage com diversos intervenientes, num passado não muito distante era necessário a reunião destes de forma a realizar diversas tarefas da engenharia de requisitos em conjunto (como a validação e aprovação de requisitos). No entanto com a difusão da internet este conceito mudou e a distância geográfica deixou de ser um entrave para o progresso dos projectos. Desta forma é essencial que a ferramenta a implementar seja centrada na *Web*, desta forma o acesso à aplicação pode ser realizado a partir de qualquer local e em qualquer altura, quebrando qualquer barreira de geográfica existente entre os utilizadores da aplicação.

Através de uma solução *Web* para suportar a gestão dos requisitos, outros factores podem ser explorados de forma a tirar partido da facilidade de acessos à aplicação. Permitir o trabalho colaborativo dos diversos utilizadores, é uma das formas encontradas para resolver vários dos problemas de comunicação e de entendimento entre clientes, utilizadores, analistas, programadores e gestores. A aplicação *Web* deve por isso ser mais um canal de comunicação entre os vários intervenientes no projecto e desta forma permitir uma melhoria significativa da qualidade dos requisitos. Este objectivo pode ser alcançado através da criação de painéis de discussão associados a cada requisito do projecto, nos quais utilizadores participam de forma ordenada.

A aplicação *Web* desenvolvida deve também incorporar um sistema de aprovações. Com a introdução de um sistema deste tipo é possível efectuar aprovações de requisitos e pedidos de alterações (*change requests*) directamente na ferramenta. Esta funcionalidade adicional permite introduzir diversas vantagens ao processo de definição e gestão de requisitos, tais como:

- Conhecimento global de quais os requisitos aprovados, em aprovação e reprovados;
- Foco na análise dos requisitos reprovados;
- Maior eficiência do processo de aprovação e consequente processo de revisão;
- Substituição do papel pela informação electrónica;
- Melhorias de resposta a pedidos de alteração do projecto;
- Permite a participação na decisão de todos os intervenientes do projecto;
- Permite detectar requisitos críticos através do número de pedidos de alteração e de reprovação;
- Permite adquirir consenso formal com os *Stakeholders* através de um método menos intrusivo.

Devido à importância de integração deste tipo de ferramenta com outras aplicações CASE, o sistema deve fornecer mecanismos que facilitem a integração com outras aplicações. Para responder a este requisito a aplicação deve ser desenvolvida segundo o paradigma de uma arquitectura orien-

tada ao serviço (SOA), permitindo assim expor diversos serviços através da utilização de *Web services*. Estes serviços encapsulam uma determinada função de negócio reutilizável, comportando-se como um interface independente da forma de implementação do sistema. Estas propriedades dos serviços SOA permitem por isso uma fácil integração com qualquer tipo de aplicação.

A usabilidade da aplicação tem também um papel importante para o sucesso deste tipo de ferramenta. A razão da sua importância reside na facilidade de utilização do sistema, redução do tempo de aprendizagem e principalmente porque os diversos *stakeholders* do projecto não devem necessitar qualquer tipo de formação para utilizarem de forma adequada o sistema. O sistema deve ser por isso desenhado de forma a apoiar as actividades do utilizador. Com base nesta preocupação deve ser desenvolvido um protótipo funcional do sistema que deve ser avaliado segundo algumas das melhores práticas de avaliação de interfaces.

Para que seja possível seguir o estado de um requisito durante o ciclo de vida do projecto, esta ferramenta deve contemplar uma máquina de estados que permita classificar o requisito ao longo do tempo. A associação de um requisito a um estado é crucial para a gestão dos requisitos, pois permite perceber a evolução do requisito e o ponto de situação de cada um em particular. No entanto esta máquina de estados deve ser desenhada de forma a permitir a personalização dos seus estados, pretende-se desta forma adaptar a ferramenta a cada organização individual, ao contrário de outras abordagens que obrigam os utilizadores a adirem às terminologias e conceitos implícitos a cada estado.

Por fim a solução a desenvolver deve se apresentar-se como uma alternativa de baixo custo para a gestão dos requisitos. Seguindo princípios de desenvolvimento rápido através da utilização de tecnologias adequadas. Este ponto é importante para que pequenas e médias empresas possam usufruir de uma aplicação de suporte às tarefas de gestão de requisitos.

1.4 Estrutura da Dissertação

Esta dissertação de tese encontra-se organizada em sete capítulos:

- **Introdução:** Capítulo introdutório do documento onde é apresentado o contexto da dissertação de mestrado, os problemas detectados a que se pretende dar resposta através de uma proposta apresentada e a estrutura da dissertação.
- **Estado de Arte:** Neste capítulo é apresentado o estudo realizado na área de desenvolvimento da tese, nomeadamente o tema da engenharia de requisitos onde é focado o seu processo e as suas actividades principais. Posteriormente é demonstrada a importância da gestão dos requisitos para o aumento de qualidade dos projectos informáticos, onde é focado a problemática da gestão de alterações aos requisitos e a necessidade de definir ligações de rastreabilidade entre os requisitos. Prosseguindo com o estudo desta área é analisado a importância da utilização de uma ferramenta, a fim de auxiliar as tarefas de gestão dos requisitos. Por fim são analisadas algumas das ferramentas da área existentes no mercado e realizado um estudo comparativo entre estas.

- **Desenvolvimento da Solução:** Neste capítulo apresenta-se o processo de desenvolvimento utilizado para a implementação de uma ferramenta de apoio à definição e gestão de requisitos. Posteriormente é descrito o planeamento do projecto realizado, a fase de análise onde é definido o que o sistema deve fazer e também a fase de desenho da solução que define a solução a implementar.
- **Plataforma Tecnológica:** Nesta secção é apresentado as tecnologias e a arquitectura de software para suportar o desenvolvimento da solução proposta. É ainda apresentado as razões pelas quais foi desenvolvido um sistema de raiz e os motivos pelas quais foram realizadas estas escolhas tecnológicas.
- **Testes Realizados:** Aqui é exposto a metodologia utilizada para garantir o correcto funcionamento do projecto implementado, nomeadamente os testes realizados no decorrer do processo de desenvolvimento.
- **Resultados Obtidos:** Nesta parte do documento é apresentado o resultado final do trabalho que derivou do estudo deste tema, da análise e do desenho desenvolvido. Esta zona do documento apresenta ainda a avaliação da ferramenta segundo a recolha de opiniões de alguns utilizadores alvos e uma comparação da ferramenta com outras previamente analisadas.
- **Conclusão:** A conclusão do trabalho é descrita nesta secção, sendo ainda apresentado alguns pontos importantes para um trabalho futuro.

2 Estado da arte

2.1 Introdução

A principal medida para avaliar o sucesso de um sistema informático reside no simples facto de este servir ou não os fins para o qual é destinado. A Engenharia de Requisitos tem como principal objectivo descobrir esses fins a partir da identificação dos *Stakeholders* e das suas necessidades, documentando-os permitindo desta forma a análise, comunicação e consequentemente o desenvolvimento da aplicação informática. (1)

Apesar das recentes e eficazes técnicas na área de Engenharia de Software, os projectos de desenvolvimento de sistemas, na maior parte dos casos, parecem estar destinados a atrasos consecutivos, agravamentos de orçamentos e em alguns casos não conseguem responder às verdadeiras necessidades do cliente. Na maioria dos casos estas falham não se devem a erros das equipas de desenvolvimento, mas sim em problemas derivados dos próprios requisitos e de todo o seu processo envolvente.

2.2 Engenharia de Requisitos

A engenharia de requisitos é um nome dado a um conjunto de actividades estruturadas no levantamento, documentação e manutenção dos requisitos. A utilização do termo engenharia implica que devem ser utilizadas de forma repetitiva e sistemática, técnicas para garantir que todos os requisitos de um sistema estão completos, consistentes e respondem às necessidades do cliente a que se destina.

As consequências de erros nos requisitos de um sistema são inúmeras. Estas consequências vão desde os típicos atrasos, aumento do custo dos projectos, elevados custos de manutenção e em casos de maior gravidade, o sistema final pode nem sequer responder às verdadeiras necessidades dos utilizadores. Assim, é importante que o processo de engenharia de requisitos seja realizado de uma forma adequada e eficiente (2).

No entanto existem alguns aspectos que fazem da Engenharia de requisitos um processo difícil. Estas dificuldades encontram-se essencialmente na falta de ideias claras dos clientes sobre o sistema de suporte que necessitam, no envolvimento de múltiplos *stakeholders* com diferentes objectivos e prioridades e ainda devido ao próprio negócio em si que se encontra em constantemente mudança. Estas dificuldades transformam o processo de ER num processo complexo em que a comunicação e a gestão da mudança assumem papéis fulcrais.

2.2.1 O Processo da Engenharia de requisitos

“If you can’t describe what you are doing as a process, you don’t know what you are doing.”

William Edwards Deming, management consultant, 1990-93

Um processo é um conjunto de actividades que transforma um input num designado output. Os processos são importantes para a partilha de conhecimento de complexos mecanismos para executar uma determinada tarefa. O processo da Engenharia de requisitos também tem os seus próprios inputs e outputs como é mostrado na Figura 1.

O processo de ER em si pode variar de organização para organização, no entanto as entradas e as saídas produzidas são similares. O processo pode ser pouco estruturado, no caso do conhecimento se encontrar centralizado nas pessoas que o realizam, ou bem estruturado, quando realizado de forma sistemática, baseado na aplicação de metodologias e *best practices*. (2)

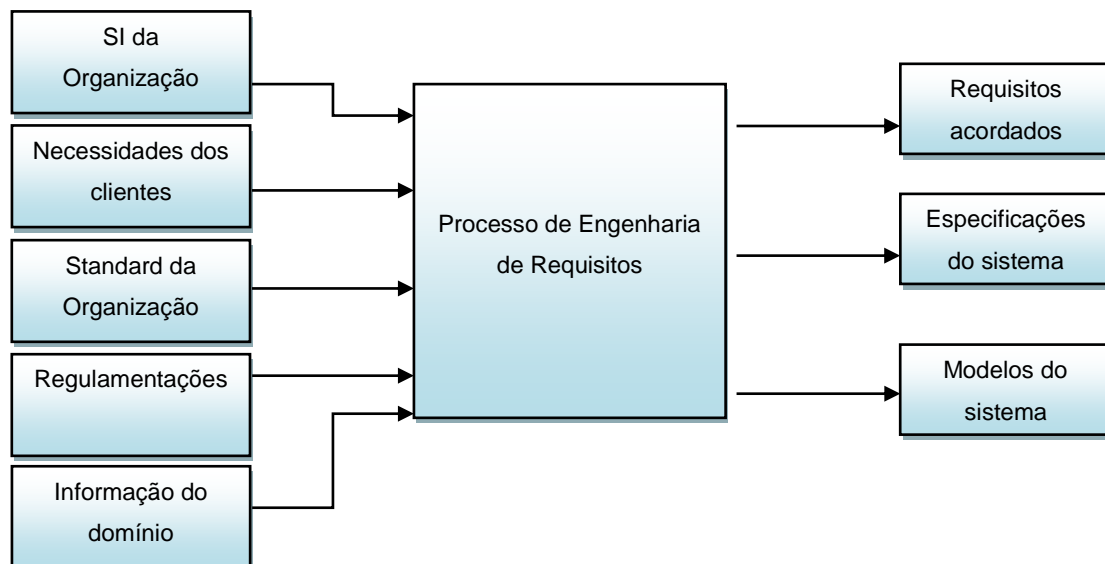


Figura 1: *Inputs e outputs* de um processo de Engenharia de Requisitos (2)

A um nível abstracto a maior parte dos processos de engenharia de requisitos podem ser descritos a partir do *coarse-grain activity model* apresentado na Figura 2. Este modelo apresenta quatro actividades distintas que interagem entre si durante o processo. (2)

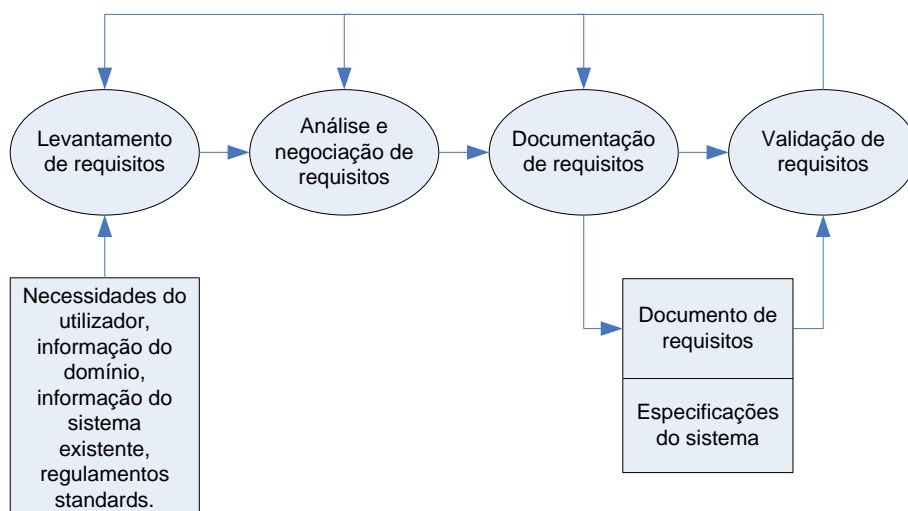


Figura 2: Modelo de actividades do processo de ER

Independentemente do processo utilizado existem quatro actividades fundamentais num processo de Engenharia de Requisitos:

- **Levantamento de requisitos:** Identificação das fontes de informação do sistema e a descoberta dos requisitos deste.
- **Análise e negociação de requisitos:** Análise detalhada dos requisitos, percepção de conflitos e dependências entre estes. Nesta fase são identificados ainda os requisitos incompatíveis com o orçamento estipulado para o projecto. Como o próprio nome indica, é necessária a comunicação com o cliente, para a aprovar os requisitos propostos.
- **Documentação de requisitos:** Escrita dos requisitos de forma a ser percebida tanto pelos stakeholders do projecto como pela equipa de desenvolvimento.
- **Validação de requisitos:** Verificação da consistência e integridade dos requisitos. Esta actividade pretende certificar que os requisitos descrevem de forma aceitável o sistema a implementar, antes de estes servirem de base para o desenvolvimento do sistema informático.

As fases apresentadas anteriormente podem ser separadas e conjugadas de diferentes formas, dependendo do tipo de projectos e do processo implementado por cada organização, no entanto todas elas derivam do modelo abstracto apresentado na figura anterior.

Estas actividades são muitas vezes apresentadas de forma sequencial, onde se começa com o levantamento e se finaliza com um documento de requisitos aprovados. Este documento serve depois como base para a implementação do sistema e é sobre este que são geridas as alterações que ocorrem durante o desenrolar do projecto.

Na realidade seja qual o for o detalhe do processo, a Engenharia de Requisitos é sempre um ciclo de actividades, tal como pode ser verificado na figura 3. As actividades são repetidas à medida

que os requisitos são derivados e a iteração continua durante a implementação e operação do sistema.

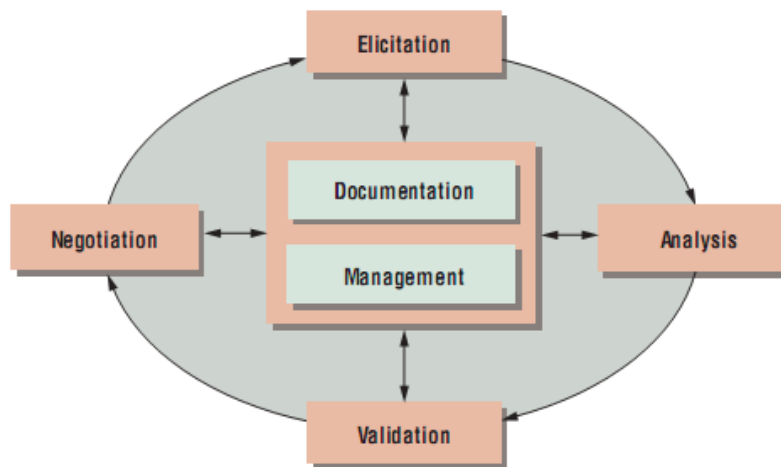


Figura 3: Ciclo de actividades da Engenharia de Requisitos (3)

O modelo de ciclo de vida apresentado na Figura 3, teve como principal influência o modelo *waterfall* da Engenharia de Software, proposto por Winston Royce em 1970. Nesse modelo partia-se do pressuposto que o processo de Engenharia de Requisitos e o desenvolvimento do sistema eram fases distintas, onde só era iniciado o desenvolvimento após o documento final estar produzido. Esse modelo pressupunha também que os requisitos de software não teriam mudanças significativas e daí a separação entre as duas fases (3).

No entanto, na prática, a separação entre estas duas fases é neste momento uma abordagem pouco realista. Actualmente, a mudança de requisitos é inevitável, pois o mundo empresarial, no qual as aplicações são utilizadas está em constante mudança – novos competidores e novos produtos a emergirem no mercado, reorganizações do negócio, reestruturações e a necessidade de reacção a novas oportunidades (3).

Devido a esta constante mutação de objectivos e prioridades, é necessário que exista uma boa gestão dos requisitos. Como é possível verificar no modelo apresentado (Figura 3), a gestão dos requisitos, exerce uma posição chave, juntamente com a documentação, no ciclo, pois é esta que permite responder às diversas mudanças que ocorrem durante o desenrolar de um projecto.

2.3 Gestão de Requisitos

Como vimos anteriormente a gestão de requisitos é uma actividade que pertence ao processo de Engenharia de Requisitos. Dado que é nesta actividade que se centra a tese do autor, esta vai ser apresentada de forma mais detalhada.

As alterações dos requisitos surgem em diversas fases do ciclo de desenvolvimento de um projecto. Existem no entanto requisitos que estão mais susceptíveis de alterações do que outros.

Os requisitos estáveis estão associados à essência do sistema e ao domínio da aplicação, devido a isso este tipo de requisitos não são tão susceptíveis de alterações.

No entanto existem também requisitos voláteis, que são específicos de uma instância do sistema para um cliente e para um contexto em particular. Existem pelo menos quatro tipos de requisitos voláteis: requisitos mutáveis, requisitos emergentes, requisitos de compatibilidades e requisitos consequenciais. (2)

2.3.1 Alterações dos requisitos

Durante o levantamento e desenvolvimento dos requisitos dos *stakeholders*, existe sempre períodos de intensas alterações dos requisitos. Nesta fase não é necessário implementar nenhum processo formal de controlo da mudança, pois trata-se de um período de grande dinâmica onde ainda não existe uma versão estável dos requisitos.

No entanto quando os requisitos de um projecto de software adquirem uma determinada estabilidade, deve ser utilizado um processo formal de controlo de alterações para que as alterações propostas sejam analisadas e seja calculado o impacto da alteração no projecto. Geralmente este nível de estabilidade de um requisito, só é alcançado quando os todos os requisitos já foram revistos e alcançam estado de aprovado. (4)

O facto de ser necessário realizar alterações a requisitos, nem sempre significa fracas práticas de engenharia. Estas mudanças podem resultar da conjugação de diversos factores como é visível na Tabela 1.

Factor de mudança	Descrição
Erros, conflitos e inconsistências nos requisitos	À medida que os requisitos são analisados e implementados, erros e inconsistências podem emergir. Estes problemas podem ser encontrados durante a fase de análise e validação dos requisitos, ou nas fases seguintes do processo de desenvolvimento.
Evolução do conhecimento do sistema pelos clientes/utilizadores	Durante o processo de desenvolvimento, os clientes e os utilizadores adquirem uma melhor percepção do que realmente necessitam para o sistema.
Problemas técnicos, de planificação ou de custo	Podem surgir problemas durante a implementação de um requisito. O custo ou o tempo de implementação pode ser demasiado elevado.
Alteração das prioridades do cliente	As prioridades do cliente podem mudar durante o desenvolvimento devido a diversos factores, como a entrada de novos competidores e novos produtos.
Alterações ao contexto do sistema	O meio no qual o sistema vai ser instalado pode mudar, causando alterações nos requisitos do sistema
Alterações na organização	A organização pode sofrer alterações na sua estrutura e nos seus processos criando novos requisitos do sistema.

Tabela 1: Principais factores para a mudança de requisitos. (2)

Devido a estes factores, a necessidade de realizar alterações aos requisitos, a um determinado momento do ciclo de desenvolvimento, é em grande parte dos casos inevitável e devido à sua frequência é actualmente encarado como um acontecimento natural. Desta forma é necessário que a gestão dos requisitos englobe um processo de gestão da mudança, o qual é conhecido por *change management*. (5)

2.3.2 *Change management*

A gestão da mudança envolve todos os procedimentos, processos e *standards* para gerir as mudanças aos requisitos de um sistema. Se não existir um processo formal para estas alterações, é impossível garantir, que as propostas realizadas para a alteração de requisitos respondem de facto aos objectivos de negócio do cliente (2) (6).

1. Políticas para a gestão da mudança

Para garantir uma consistente abordagem à gestão da mudança, é necessário definir uma série de políticas que devem cobrir os seguintes aspectos:

- O processo associada aos pedidos de mudança e a informação necessária para o processamento de cada um desses pedidos;
- O processo usado para analisar o impacto e os custos da mudança e a informação associada à rastreabilidade;
- Definir quem deve pertencer à equipa que considera formalmente os pedidos de mudança;
- Qual o software de suporte ao processo de controlo da mudança.

2. O processo da gestão da mudança

Este processo consiste num conjunto de actividades que são fundamentais para a aprovação da alteração de um requisito (ver Figura 4). Este processo tem como entrada a identificação do problema e como saída a revisão dos requisitos.

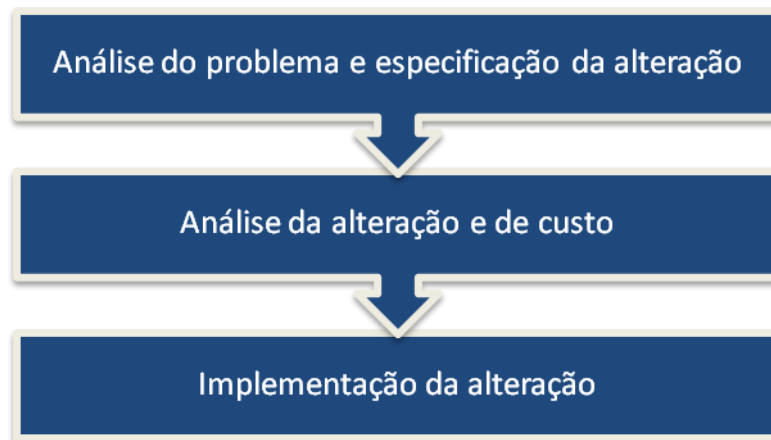


Figura 4: Estados num processo de *change management*. (2)

O processo específico para a primeira e última fase deste processo é sempre referente ao tipo de mudança a realizar. No entanto a fase de análise da alteração e de custo é um processo mais genérico, como é apresentado na Figura 5.

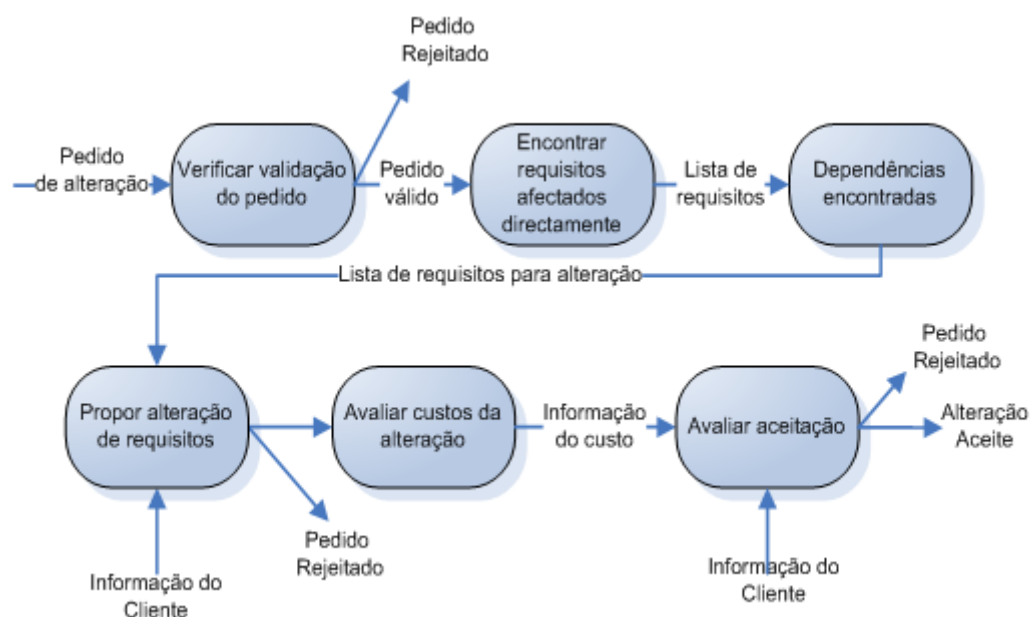


Figura 5: Processo genérico de análise da alteração e de custo. (2)

A partir da figura apresentada podemos verificar que um pedido de alterações pode ser rejeitado devido a diversos factores:

- Se o pedido de mudança for inválido;
- Se o pedido resultar em alterações não aceites pelos utilizadores finais;

- Se o custo de implementação da alteração for demasiado alto ou demorar muito tempo.

Durante o processo apresentado na Figura 5 é necessário realizar diversas análises aos requisitos, para determinar o verdadeiro impacto que uma alteração tem no projecto. Para que isto seja possível é imprescindível que exista uma forma de associar os requisitos entre si. A estas associações dá-se o nome de rastreabilidade entre requisitos.

A rastreabilidade assume por isso um papel elementar na gestão dos requisitos, devido à sua importância, esta temática é abordada numa secção independente. Nessa secção será possível perceber em que medida, estas associações fornecem indicadores importantes para a gestão dos requisitos do projecto, entre eles os indicadores necessários para a avaliação do impacto da alteração de um ou mais requisitos, a nível de custo e esforço no projecto.

2.3.3 Rastreabilidade de requisitos

No contexto da engenharia de requisitos a rastreabilidade consiste em perceber como requisitos de alto nível são transformados em requisitos de baixo nível. Um exemplo prático neste contexto é a ligação dos requisitos dos *stakeholders* com os requisitos de sistema. Desta forma é possível descrever e seguir a vida de um requisito, seja a partir da sua origem como através do retrocesso de um requisito de grão mais fino. (6)

A utilização de rastreabilidade permite desta forma obter os seguintes benefícios:

- ✓ Aumento da confiança do alcance dos objectivos estabelecidos;
- ✓ Capacidade para avaliar o impacto de alterações;
- ✓ Aptidão para seguir o progresso do projecto;
- ✓ Capacidade de balancear o custo/benefícios;
- ✓ Melhora a responsabilidade das organizações subordinadas.

De uma forma geral, as associações de rastreabilidade são de muitos para muitos, ou seja um requisito de nível baixo pode estar ligado a vários requisitos de um nível superior, e vice-versa. A Figura 6 apresenta a rastreabilidade de forma descendente entre os diversos contextos dos requisitos e entre a informação de testes. (7)

Vários tipos de análise de rastreabilidade (ver Figura 6) podem suportar os processos de engenharia de requisito

A análise de impacto é utilizada, como anteriormente já foi referido, para determinar quais os requisitos afectados quando um ou vários são sujeitos a alteração. Esta análise suporta o processo de *change management*. (4)

A análise de derivação trabalha na direcção oposta da análise de impacto. Esta verifica quais os requisitos de alto nível que dão origem ao seleccionado. Desta forma, requisitos de baixo nível que não possuam ligação a níveis superiores são potencialmente dispensáveis para o projecto. O processo suportado é neste caso a análise de custo – benefício. (4)

Por fim, temos a análise de cobertura que pode ser utilizada para determinar se todos os requisitos derivam em requisitos de outras camadas de níveis inferiores e se são abrangidos por testes. A falta destas associações é um forte indicador que determinado requisito pode não ser implementado ou testado. Este tipo de análise pode ainda servir de indicador do progresso do projecto e partindo do mesmo princípio o progresso dos planos de teste. (4)

Devido aos diversos tipos de análises que podem ser realizadas, a rastreabilidade é um conceito chave na engenharia de requisitos. Deve por isso ser sempre utilizado, pois fornece indicadores de grande importância que simplificam todo o processo de gestão dos requisitos.

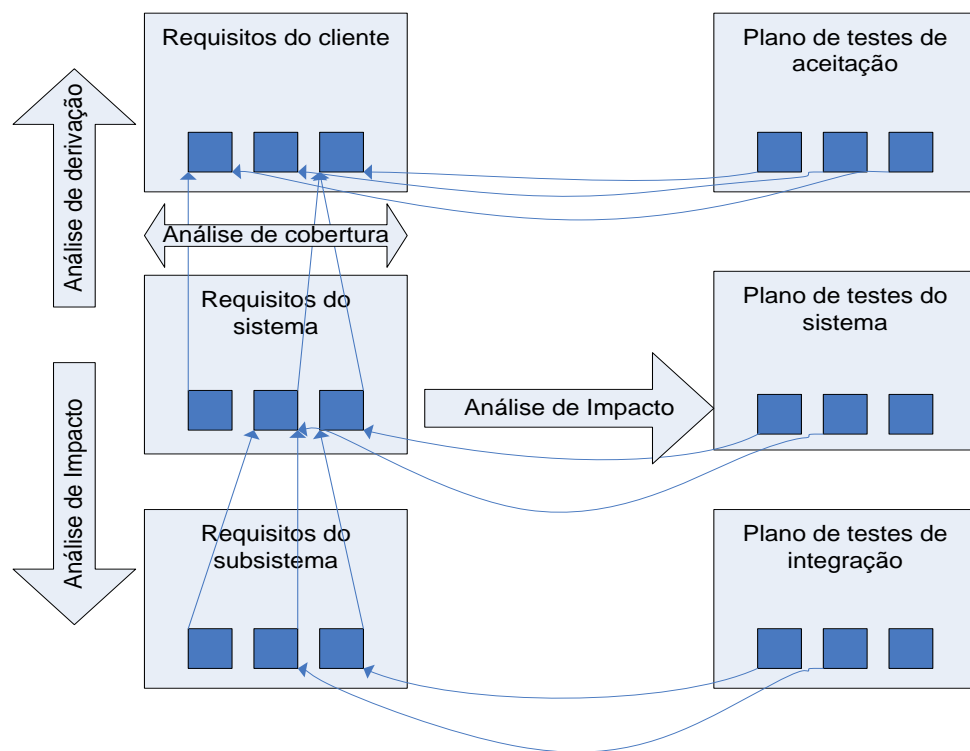


Figura 6: Rastreabilidade de requisitos e tipos de análise suportados. (4)

2.3.4 O suporte de ferramentas

Gerir a evolução dos requisitos ao longo do desenrolar do projecto é uma tarefa difícil de realizar de forma manual. Uma ferramenta apropriada pode facilitar significativamente muitos dos desafios que a gestão de requisitos necessita de responder. Uma ferramenta de gestão de requisitos funciona como um assistente automático durante o desenrolar do desenvolvimento do projecto. (8)

Este tipo de aplicações são classificadas como ferramentas do tipo CASE (*computer-aided software engineering*), pois são produtos que têm como principal objectivo suportar uma ou mais actividades do processo de engenharia de software. (9) (10)

Mesmo que um analista tenha realizado um trabalho excepcional durante o levantamento dos requisitos, a utilização de um assistente automático pode ajudar a gestão dos requisitos durante o processo de desenvolvimento. De seguida são apresentadas algumas das tarefas que este tipo de ferramentas pode ajudar a cumprir:

Gestão de versões e alterações. Durante o desenrolar de um projecto devem ser definidas *baselines* de requisitos (conjunto de requisitos que uma particular *release* abrange). O histórico de alterações de cada requisito pode ajudar a explicar decisões realizadas e permitir reverter para uma versão anterior se necessário.

Guardar atributos de requisitos. Deve ser guardado diversas informações ou atributos sobre cada requisito. Esta informação deve estar visível e susceptível de alteração, a todas as pessoas que trabalham sobre o projecto. As ferramentas de gestão de requisitos produzem de forma predefinida alguns desses atributos, como a data de criação e versão do requisito, estas permitem ainda que sejam definidos atributos adicionais de diversos tipos.

Ligar requisitos a outros elementos. Ligar requisitos individuais a outros componentes do sistema, permite assegurar que a equipa não negligencia inadvertidamente nenhum requisito durante a implementação. Deve ser possível definir ligações entre diferentes tipos de requisitos e entre requisitos de diferentes subsistemas. Quando é analisado o impacto de um pedido de alteração de um requisito, estas ligações revelam todos os outros elementos que podem também ser afectados.

Seguir os estados. Seguir o estado de cada requisito durante o desenvolvimento permite auxiliar a gestão do projecto. Se um gestor de projecto obtiver a informação da percentagem de requisitos que se encontram implementados e verificados, em implementação e ainda em análise ou revisão, então o gestor consegue ter uma visão correcta e actualizada do estado do projecto.

Ver subconjuntos de requisitos Ordenar, filtrar e realizar *queries* à base de dados para ver subconjuntos de requisitos com valores específicos.

Controlo de acessos. Especificar permissões de acesso a individuais ou a grupos de utilizadores. Acesso via Web permite a partilha de informação dos requisitos com todos os membros da equipa, mesmo que estes estejam geograficamente separados.

Comunicação com os Stakeholders. A maior parte das ferramentas de gestão de requisitos permite que membros da equipa discutam sobre assuntos relacionados com os requisitos de forma electrónica. Mensagens de correio electrónico são utilizadas para notificar os utilizadores afectados quando novas mensagens de discussão são escritas ou quando requisitos são alterados. (11)

Algumas ferramentas fornecem ainda interfaces para sistemas de processamento de texto comerciais, como por exemplo o Microsoft Word. Através desta integração é possível importar requisitos convertendo a língua natural para a base de dados da ferramenta.

Este tipo de ferramentas deve ainda suportar a integração com outras ferramentas CASE. Apesar de não ser um requisito fundamental é sem dúvida uma mais-valia para qualquer tipo de ferramenta deste tipo. Como é visível na Figura 7, um sistema de gestão de requisitos pode interagir com diversas ferramentas que suportam o desenvolvimento de um projecto informático.

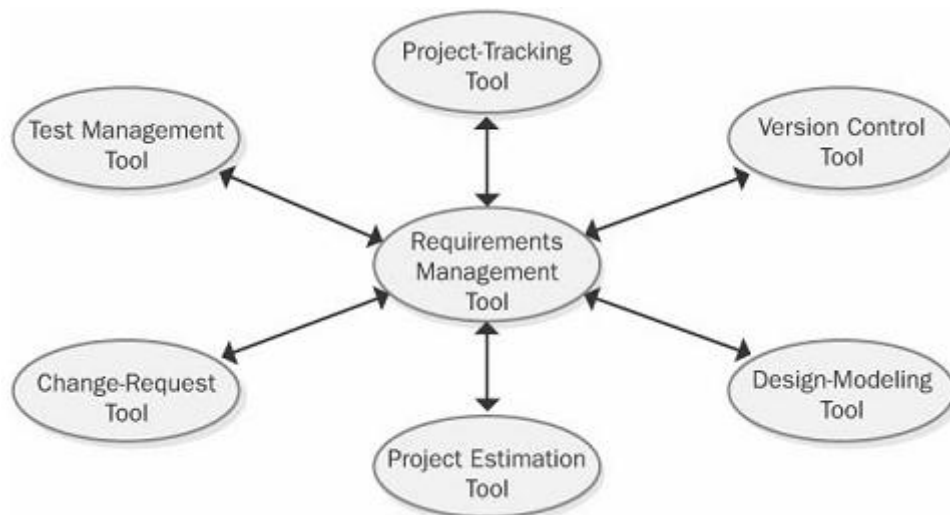


Figura 7: Integrações com outras ferramentas CASE. (12)

As diversas ferramentas CASE apresentadas no diagrama anterior têm um relacionamento directo com os requisitos controlados por uma ferramenta de gestão de requisitos. De forma a clarificar a importância destas integrações, podemos tomar como exemplo, a comunicação de informação com uma ferramenta de gestão de testes. Neste caso a associação dos testes com os requisitos permite analisar o progresso do projecto e detectar quais os requisitos que não têm testes associados. A falta desta associação é um indicador, como foi referido anteriormente, que o requisito pode não fazer sentido.

Como podemos verificar as ferramentas de gestão de requisitos permitem obter significativas melhorias no processo de Engenharia de Requisitos. Estas melhorias são fulcrais nos dias de hoje em que o tempo de desenvolvimento de um projecto é cada vez mais curto e a mudança de requisitos é em grande parte dos casos inevitável. No entanto, há que ter em consideração que nenhuma ferramenta de gestão de requisitos pode substituir um processo de Engenharia de Requisitos bem estruturado. É por isso necessário ter um processo bem definido para poder tirar partido de todas as vantagens que este tipo de ferramentas pode oferecer. (5) (11)

2.4 Análise de Ferramentas

Além do estudo do estudo realizado acerca das temáticas apresentadas nas secções anteriores, foi realizado uma análise a algumas das ferramentas comerciais e *open-source* existente no mercado (13). Com esta análise pretende-se avaliar as ferramentas individualmente e com isso identificar os pontos fortes e fracos de cada uma.

A partir desta informação será então possível realizar um juízo final sobre o estado actual das ferramentas de gestão de requisitos, e em que medida facilitam e agilizam este processo durante as diversas fases do desenvolvimento de um sistema informático.

2.4.1 Open Source Requirements Management Tool

Esta ferramenta de gestão de requisitos foi desenhada para garantir uma completa rastreabilidade durante o ciclo de vida de desenvolvimento do software para funcionalidades, requisitos, desenho, implementação e testes. Apresenta também um interface gráfico para derivação de requisitos, controlo de versões e atributos personalizáveis dos requisitos (principio, origem, riscos, esforço etc.) Esta ferramenta de código aberto é disponibilizada pela SourceForge.net (14).

Principais características e funcionalidades da ferramenta:

- ✓ Explorador em árvore de requisitos;
- ✓ Geração de uma matriz de rastreabilidade;
- ✓ Suporte ao anexo de ficheiros e hiperligações;
- ✓ Produz alguns relatórios de forma automática;
- ✓ Suporte a hierarquia de requisitos;
- ✓ Apresenta um diagrama de relacionamento entre requisitos;
- ✓ Permite a administração de grupos, utilizadores, acessos e projectos;
- ✓ Controlo de versões dos requisitos;

- ✗ Fraca Usabilidade;
- ✗ Só suporta quatro tipos de requisitos;
- ✗ Não suporta integração com aplicações externas;
- ✗ Não suporta a personalização de novos tipos de requisitos;
- ✗ A ligação de requisitos directamente na matriz não é suportada;
- ✗ Fórum de discussão inexistente.

Esta aplicação tem como principal ponto forte a geração de um diagrama de impacto de alterações de requisitos. Por outro lado não disponibiliza diversas funcionalidades essenciais para uma boa gestão de requisitos, algo bastante visível quando comparado com outras ferramentas, entre elas a destacar a falta de suporte para integração com aplicações externas e a falta de um fórum de discussão, essencial para a comunicação entre membros de equipa e *stakeholders* do projecto.

2.4.2 RTH

Mais uma aplicação *open-source* disponibilizada pela *SourceForge.net*. Esta aplicação distingue-se pela utilização da tecnologia *PHP*, sendo por isso uma ferramenta *Web-based*. Esta ferramenta suporta algumas das tarefas de gestão de requisitos e de gestão de testes. (15)

Após a utilização da ferramenta foi possível realizar um levantamento das suas principais características. Os principais pontos positivos e negativos resumem-se nos seguintes:

- ✓ Página inicial apresenta as últimas alterações realizadas;
 - ✓ Controlo de versões dos requisitos;
 - ✓ Suporta além da gestão de requisitos, a gestão de testes, gestão de *releases*, e gestão de defeitos (*bugs*);
 - ✓ Possui um explorador de requisitos em árvore com suporte de hierarquia;
 - ✓ Disponibiliza um fórum de discussão por requisito;
 - ✓ Permite associar testes aos requisitos;
 - ✓ É possível anexar ficheiros aos requisitos;
 - ✓ Administração de projectos, utilizadores e acessos;
 - ✓ Geração automática de relatórios de testes e resultados.
-
- ✗ Fraca usabilidade
 - ✗ Só relaciona requisitos através de hierarquias;
 - Não suporta rastreabilidade entre requisitos
 - ✗ Não produz matrizes de rastreabilidade;
 - ✗ Não é possível personalizar os tipos e atributos dos requisitos;

O RTH é uma aplicação *Web*, este ponto só por isso é um ponto forte, pois permite disponibilizar a aplicação com facilidade aos diversos intervenientes no projecto, além disso a aprendizagem é simples, permitindo ao utilizador de uma forma fácil e rápida criar requisitos logo na primeira abordagem com a aplicação.

Os pontos fracos a destacar é o interface fraco, que dificulta ao utilizador a execução de tarefas mais complexas. A inexistência de associações entre requisitos, que só é suportada visualmente no caso de hierarquias, e consequentemente a falta de uma matriz de rastreabilidade, o que torna difícil a gestão de alterações durante um projecto. Todos estes factores fazem desta ferramenta uma má escolha para a gestão eficiente de requisitos.

2.4.3 Requisite Pro

Esta aplicação da *IBM* é uma das aplicações mais utilizadas na gestão dos requisitos. O *Rational Requisite Pro* distingue-se de outras aplicações devido à avançada ligação em tempo real com o *Microsoft Word*. (16)

O resultado da análise da ferramenta foi obtido com base na sua utilização, consulta de documentação e em outras análises realizadas à aplicação (11) (17). As principais características são as seguintes:

- ✓ Explorador de requisitos em árvore, organização por *packages*;
 - ✓ Boa compatibilidade com o *Microsoft Word*;
 - ✓ Controlo de versões dos requisitos;
 - ✓ Facilidade de criar documentação devido aos diversos templates oferecidos;
 - ✓ Integração com outras aplicações:
 - Sincronização com aplicações de teste (*Quality Center*);
 - *Microsoft Project*;
 - Outras ferramentas CASE da *Rational*;
 - ✓ É possível adicionar/remover/modificar atributos, tipos de documentos e tipos de requisitos;
 - ✓ Suporta a rastreabilidade entre os requisitos através do relacionamento de requisitos de diferentes tipos. É possível definir estes relacionamentos directamente na matriz de rastreabilidade apresentada;
 - ✓ Quando é alterado um requisito são sinalizados todos os requisitos que se encontram relacionados com este, permitindo desta forma uma gestão de alterações e cálculo de impacto mais eficiente;
 - ✓ Apresentação de requisitos em vistas do tipo tabela, que pode ser filtrada e “ordenada” de acordo com diversos parâmetros;
 - ✓ Suporte a hierarquia de requisitos;
 - ✓ Fórum de discussão que pode estar associado a um ou mais requisitos;
 - ✓ Permite a criação de *baselines* do projecto;
 - ✓ Suporta a definição de rastreabilidade entre projectos;
 - ✓ Arquivo de projectos.
-
- ✗ Só é possível tirar partido de todas as suas vantagens quanto utilizado em conjunto com o processador de texto comercial *Microsoft Word*;
 - ✗ *Design* fraco, o que aumenta o tempo de aprendizagem;
 - ✗ Alguns problemas de usabilidade detectados devido à integração forte com o processador de texto.

O *Requisite Pro* distingue-se das restantes ferramentas, pois utiliza um sistema centralizado nos documentos, devido a esta característica, esta aplicação tem uma forte integração com o processador de texto *Microsoft Word*. Num posto de trabalho com este processador de texto, basta a selecção de partes de texto para incluir directamente esse texto como um requisito na aplicação.

Esta ferramenta tem com principais aspectos positivos a facilidade de manipulação e edição da rastreabilidade dos requisitos, a flexibilidade é outro dos pontos a destacar devido a diversas personalizações que é possível realizar, a incorporação de um fórum de discussão por requisito que

permite dinamizar a discussão e melhorar a qualidade do requisito, e ainda a facilidade de importação de requisitos a partir de documentos Word. Por outro lado, possui alguns pontos fracos a considerar. A falta de associação requisito – documento, um interface pouco agradável ao utilizador e alguns problemas de usabilidade, que acentuam a curva de aprendizagem dos utilizadores.

Caliber RM

O *Caliber RM* é uma ferramenta disponibilizada pela *Borland*, também muito utilizada em todo o processo de gestão de requisitos. Este software distingue-se de outras ferramentas da área devido ao facto de ser uma aplicação muito completa e por permitir realizar um elevado número de personalizações e integrações. (18) (10)

As principais características da ferramenta resumem-se ao seguinte:

- ✓ Navegação dos requisitos é realizada através de um sistema em árvore;
- ✓ Boa usabilidade complementado por um bom *design*;
- ✓ Suporte de hierarquia e de diversos tipos de requisitos;
- ✓ Fóruns de discussão por projecto e requisitos;
- ✓ Agregação de diversas informações sobre o requisito num ponto único e separação da informação em diversos separadores contextualizados (*Details, User Attributes Approvals, Responsibilities, References, Traceability, Validation, Discussion e History*).
- ✓ Associa requisitos a utilizadores/grupos os quais notifica quando algum dos seus requisitos é alterado;
- ✓ É possível ligar requisitos a documentos;
- ✓ Suporta a rastreabilidade, entre requisitos (do projecto e de projectos externos), casos de teste, grupos de testes, testes individuais e passos de testes (após a integração com *CaliberRBT* ou *Mercury TestDirector*);
- ✓ Apresenta matrizes de rastreabilidade, que suportam a aplicação de filtros personalizados, definição de ligações directamente na matriz e sinalizam o impacto nos requisitos quando um requisito é alterado.
- ✓ Na área de administração é possível criar e editar atributos personalizados, e ainda a personalização de tipos de requisitos;
- ✓ Permite criar e comparar *baselines*;
- ✓ Funcionalidades típicas de gestão de projectos, utilizadores e glossário.
- ✓ Controlo de versões dos requisitos;
- ✓ Permite gerar diversos tipos de relatórios;
- ✓ Suporte para a produção de relatórios personalizados utilizando templates Microsoft Word.

Após o estudo e experimentação desta ferramenta, é possível concluir que esta aplicação é a melhor a nível de usabilidade e interface apresentado ao utilizador. Além disso tem outros pontos fortes, tais como a facilidade de manipulação dos requisitos directamente no explorador em árvore, a

facilidade de integração com outras ferramentas externas e a existência de um fórum de discussão por requisito.

No entanto esta aplicação peca por apresentar demasiada informação sobre um requisito ao utilizador, o que pode confundir um utilizador ou *stakeholder* menos experiente. Como consequência, é necessário algum tempo de aprendizagem para tirar total partido das diversas funcionalidades disponibilizadas.

2.4.4 Telelogic DOORS Fastrack

Esta ferramenta disponibilizada pela *Telelogic* pode ser acedida via Internet através de um comum Web Browser e dispõe de diversas funcionalidades orientadas para cada tipo de perfil de utilizador. (19)

Principais características da aplicação são:

- ✓ Na página inicial são apresentados dados estatísticos do estado do projecto, e informações que dependem das permissões de cada utilizador;
- ✓ Permite a importação de requisitos a partir de ficheiros Excel;
- ✓ Apresenta requisitos em esquema de árvore, tabelas, através de um gráfico relacional e ainda em matriz de Gantt;
- ✓ Suporta a associação entre requisitos;
- ✓ Gera diversos relatórios automáticos;
- ✓ É Possível aplicar diversos filtros à matriz de rastreabilidade;

- ✗ Não é possível associar requisitos directamente na matriz de rastreabilidade;
- ✗ Painel de discussão inexistente.
- ✗ Usabilidade fraca.

Esta ferramenta tem como principais pontos fortes as diversas vistas apresentadas sobre os requisitos (árvore, tabela, diagrama e matriz), a facilidade de manipulação dos requisitos através do explorador em árvore e a apresentação de dados do projecto personalizados segundo cada tipo de utilizador na página inicial.

2.4.5 Análise comparativa

Com base nas análises realizadas, é possível sistematizar e comparar as diversas funcionalidades numa tabela comparativa. Esta tabela permite ter uma visão geral das principais funcionalidades disponibilizadas e quais as ferramentas que permitem uma melhor gestão dos requisitos (ver Tabela 2).

Funcionalidades	Caliber RM	Req. Pro	OSRT	Doors	RTH
Explorador em árvore	A	C	D	C	D
Nível de Usabilidade	B	D	D	C	D
Facilidade de aprendizagem	C	D	D	D	C
Fórum de discussão	C	C	I	I	D
Suporte de hierarquia	A	B	C	C	C
Rastreabilidade entre requisitos	A	A	B	B	I
Cálculo de Impacto de alterações	C	C	I	C	I
Importação de requisitos	C	C	I	C	I
Associação requisito-documento	C	D	D	I	D
Personalização de atributos	A	A	C	C	I
Personalização de tipos	A	C	C	I	C
Gestão de <i>Baselines</i>	C	C	D	I	I
Guardar o histórico dos requisitos	B	B	C	B	C
Notificação de alterações	B	C	I	C	C
Suporte a integrações externas	C	C	I	I	I
Gerir utilizadores e projectos	C	C	D	C	D
Geração relatórios automáticos	B	C	C	C	I
Apreciação global	A	B	C	B	D

Legenda: A - Muito Bom; B - Bom; C - Razoável; D - Fraco; I - Inexistente;

Tabela 2: Comparação funcional das ferramentas analisadas.

De sublinhar que as classificações apresentadas na Tabela 2 são válidas segundo o enquadramento do trabalho e o problema descrito no capítulo 1.2.

Após a análise das diversas ferramentas apresentadas é possível concluir que estas diversificam bastante a sua abordagem ao problema, assim como a apresentação de uma solução para este.

Comparando as cinco ferramentas, considero que aquela que disponibiliza um maior número de funcionalidades, e por isso de certa forma superior às restantes, é a ferramenta apresentada pela *Borland*. No entanto existem no meu entender, alguns pontos em que esta também poderia ser melhorada, principalmente no que diz respeito ao tempo de aprendizagem necessário e ao excesso de informação apresentada por requisito que em grande parte dos casos não é utilizada durante o processo de gestão dos requisitos.

Segundo a análise realizada, as ferramentas *open-source* ainda têm algum caminho a percorrer na área de gestão de requisitos, dada a escassez de produtos deste tipo existentes actualmente (13). Apesar de serem uma boa solução para alguns problemas e tarefas específicas, ainda não existe uma aplicação robusta que possa responder de forma eficiente à grande parte dos desafios que esta disciplina apresenta. Existe por isso uma grande discrepância de funcionalidades, entre ferramentas de gestão de requisitos comerciais e *open-source* na área.

2.5 Conclusão

Nesta secção foi descrito a importância da gestão dos requisitos durante o ciclo de vida de um projecto de software, dado que ainda existem poucas opções de escolha. Esta gestão assume hoje um papel decisivo para o sucesso dos projectos de software. O aumento da complexidade das aplicações e outros factores externos, como alterações nas organizações, obrigam a que seja necessário realizar uma gestão eficiente dos requisitos, pois a alteração de requisitos durante o desenrolar do desenvolvimento de um projecto é em muitos casos inevitável.

A partir deste estudo e da análise efectuada às diversas ferramentas existentes, é possível perceber quais são as principais características destas ferramentas, quais as melhores abordagens a tomar, e quais os pontos fracos onde se pode melhorar.

Tendo como base este documento, pode-se concluir que uma solução ideal para a gestão de requisitos eficiente, simples e eficaz deve passar pelo seguinte:

- Utilização de um explorador em árvore dos artefactos, para que seja possível visualizar directamente hierarquias e a que tipos correspondem;
- Um interface simples mas de boa usabilidade de forma a aumentar a produtividade e a simplificar a aprendizagem;
- Não disponibilizar todos os atributos possíveis de cada requisito mas sim os que são essenciais e mais comuns. No entanto deve ser permitido criar novos tipos de atributos e desta forma abranger propriedades específicas de cada organização;
- Promover a crítica e a pro-actividade entre os diversos intervenientes no projecto através de fóruns de discussão;
- Controlar as versões dos requisitos;
- Permitir a associação entre requisitos de diversos tipos;
- Disponibilizar matrizes de rastreabilidade para facilitar identificação de dependências entre os requisitos e cálculo de impacto de alterações aos requisitos.
- Permitir uma fácil integração com aplicações externas, como aplicações de gestão de testes e de gestão de configurações;
- Gerar relatórios que facilitem a documentação dos requisitos e apresentem a situação actual do projecto;

3 Desenvolvimento da solução

Neste capítulo apresenta-se o processo de desenvolvimento utilizado para a implementação de uma ferramenta de apoio à definição e gestão de requisitos.

A utilização de uma metodologia de desenvolvimento seguiu as seguintes motivações:

- Impor um processo de desenvolvimento disciplinado, planeado e detalhado;
- Tornar o desenvolvimento do software mais estruturado e eficiente;
- Assegurar o desenvolvimento de uma solução de qualidade utilizando as melhores práticas de engenharia de software e de concepção de SI.

3.1 Processo de Desenvolvimento

O primeiro passo no desenvolvimento de software passa por escolher uma metodologia que se adequa ao projecto de software a desenvolver.

O conceito de processo de desenvolvimento de software é um conceito de âmbito muito vasto, este pretende designar uma sequência de actividades, normalmente agrupadas em fases e tarefas, que são executadas de forma sistemática e uniformizada, realizadas por intervenientes com responsabilidades bem definidas, que tendo como base um conjunto de tarefas, produzem um conjunto de saídas. Segundo Booch estas permitem atingir uma orientação sobre a sequência de realização de actividades, especificar modelos descritivos do sistema, dirigir as tarefas e providenciar critérios para a monitorização e avaliação dos modelos e actividades do projecto. (20) (21)

O processo de desenvolvimento adoptado neste projecto foi o processo em cascata revisto (ver Figura 8). Este processo baseado no modelo em cascata tradicional, prevê a possibilidade de a partir de qualquer tarefa do ciclo se poder regressar a uma tarefa anterior de forma a contemplar alterações funcionais e/ou técnicas que possam surgir durante o projecto, em virtude de um maior conhecimento decorrente do desenrolar deste.

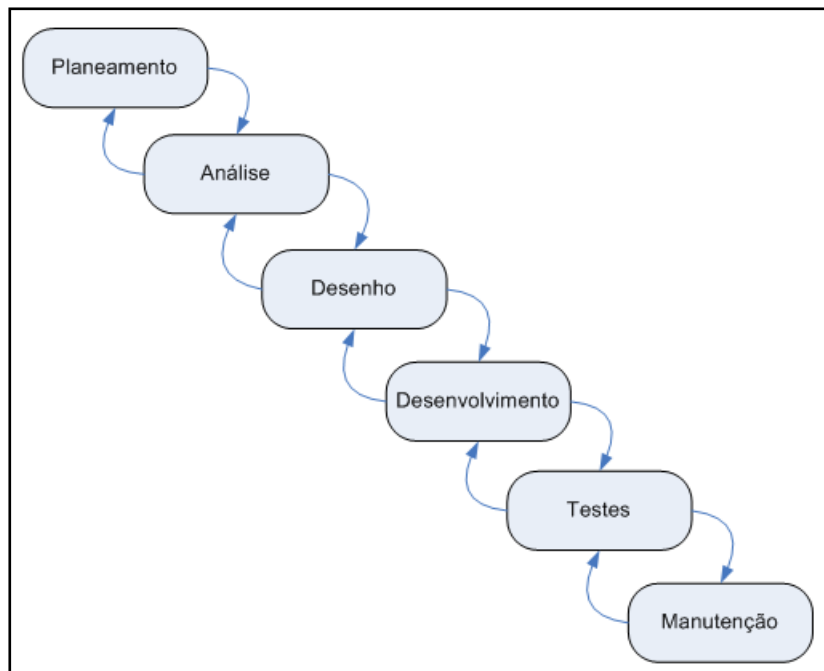


Figura 8: Processo em cascata revisito

A justificação para a adopção deste modelo para o desenvolvimento da ferramenta prende-se com o facto do trabalho se realizar num contexto empresarial em que com o decorrer do projecto poderia ser necessário realizar algum tipo de alteração sugerida pela organização ou mesmo pelo autor da tese de forma a responder a qualquer tipo de factor gerador de mudança aos requisitos do sistema.

Dado que este modelo apresenta o problema de possibilidade de extensão do período de tempo que decorre entre o início do projecto e a entrega da solução final, optou-se por desenvolver um protótipo do sistema logo nas primeiras fases do processo, de forma a restringir o número de alterações nas fases de desenvolvimento e testes da solução.

3.2 Planeamento

Antes de iniciar qualquer tipo de análise do problema ou qualquer outra actividade do processo de desenvolvimento, é necessário realizar um planeamento do projecto. Esta fase é importante uma vez que é necessário ter uma visão global do âmbito do projecto, de modo a elaborar um plano deste e analisar a sua viabilidade dentro do contexto da organização e do tempo disponível.

O plano global de desenvolvimento utilizado encontra-se descrito na figura seguinte.

ID		Nome da Tarefa	Duration	Start	Finish
1		Projecto	208,5 d	Thu 01-11-07	Fri 29-08-08
2		Planeamento	4 d	Thu 01-11-07	Wed 07-11-07
3		Definição do âmbito do Projecto	1 d	Thu 01-11-07	Fri 02-11-07
4		Elaborar Plano do projecto	2 d	Fri 02-11-07	Tue 06-11-07
5		Revisão e aprovação do plano	1 d	Tue 06-11-07	Wed 07-11-07
6		Análise	20 d	Wed 07-11-07	Wed 05-12-07
7		Especificação de Casos de Uso	6 d	Wed 07-11-07	Thu 15-11-07
8		Especificação de Requisitos	6 d	Thu 15-11-07	Fri 23-11-07
9		Construção de Protótipo Funcional	4 d	Fri 23-11-07	Thu 29-11-07
10		Validação de Protótipo Funcional	1 d	Thu 29-11-07	Fri 30-11-07
11		Revisão de Requisitos e protótipo	2 d	Fri 30-11-07	Tue 04-12-07
12		Especificações do sistema aprovadas	1 d	Tue 04-12-07	Wed 05-12-07
13		Desenho	12 d	Wed 05-12-07	Fri 21-12-07
14		Diagrama de Actividades	4 d	Wed 05-12-07	Tue 11-12-07
15		Modelo de Domínio	2 d	Tue 11-12-07	Thu 13-12-07
16		Modelo de Dados	3 d	Thu 13-12-07	Tue 18-12-07
17		Diagrama de Estados	2 d	Tue 18-12-07	Thu 20-12-07
18		Aprovação das especificações técnicas	1 d	Thu 20-12-07	Fri 21-12-07
19		Desenvolvimento	119 d	Thu 03-01-08	Thu 26-06-08
76		Testes	10 d	Thu 26-06-08	Thu 10-07-08
77		Testes de Sistema	5 d	Thu 26-06-08	Thu 03-07-08
78		Testes de Aceitação	5 d	Thu 03-07-08	Thu 10-07-08
79		Manutenção	19 d	Fri 01-08-08	Fri 29-08-08
80		Suporte	19 d	Fri 01-08-08	Fri 29-08-08
81		Fim de Suporte	0 d	Fri 29-08-08	Fri 29-08-08

Figura 9: Plano do projecto desenvolvido

3.3 Análise

A fase de análise encontra-se no domínio do problema. Esta fase tem como *output* final uma lista de requisitos de vários tipos que especificam a forma como um sistema deve funcionar ou seja o que o sistema “deve fazer”.

Dado que o âmbito do projecto é bastante abrangente, decidiu-se realizar o particionamento do trabalho em partes mais pequenas, o que permitiu o levantamento mais eficiente dos diversos requisitos do sistema. O método utilizado para realizar esse particionamento foi a utilização de diagramas de casos de utilização que representam a visão do sistema segundo a perspectiva dos seus utilizadores.

Todos os diagramas representados neste relatório foram criados recorrendo à ferramenta *Enterprise Architect*, da *Sparx Systems*. Para a modelação optou-se pela linguagem *Unified Modeling Language version 2.0* (UML), dado que as suas propriedades e características, que fazem desta uma escolha natural para a especificação, visualização e documentação de sistemas de informação.

3.3.1 Diagramas de Casos de Utilização

Este tipo de diagramas permitem capturar os requisitos funcionais, através de uma aproximação focada na utilização do sistema. (22) A utilização desta abordagem ao problema permitiu alcançar o seguinte:

- Particionamento do problema;
- Adquirir as visões dos diferentes actores do sistema e das tarefas e papéis que têm que cumprir;
- Descoberta de novos requisitos que até esta fase não tinham sido percepcionados através da utilização e análise de outras ferramentas.

Seguidamente são apresentados alguns dos diagramas concebidos para suportar a definição do problema.

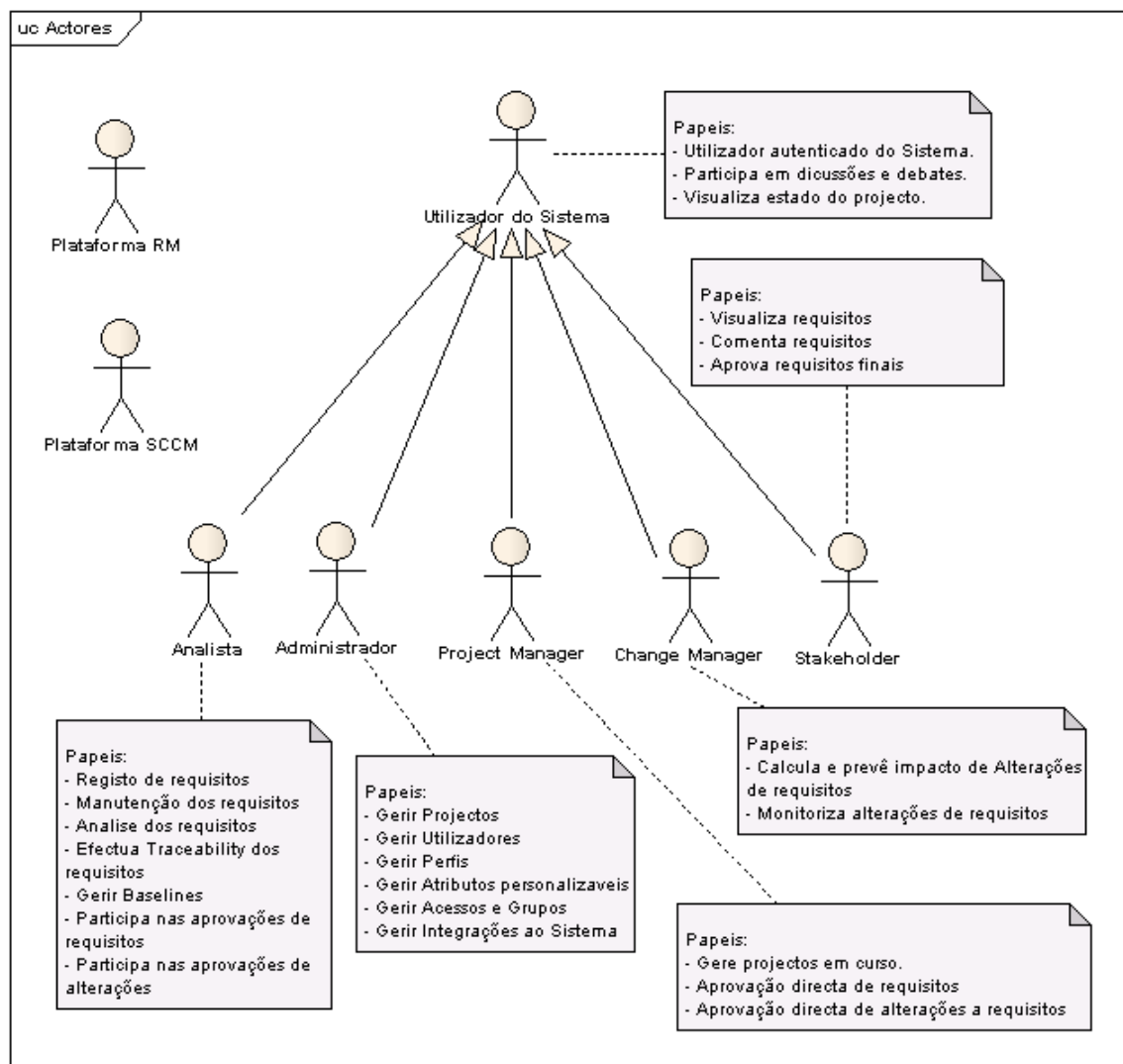


Figura 10: Diagrama de Actores e os seus papéis

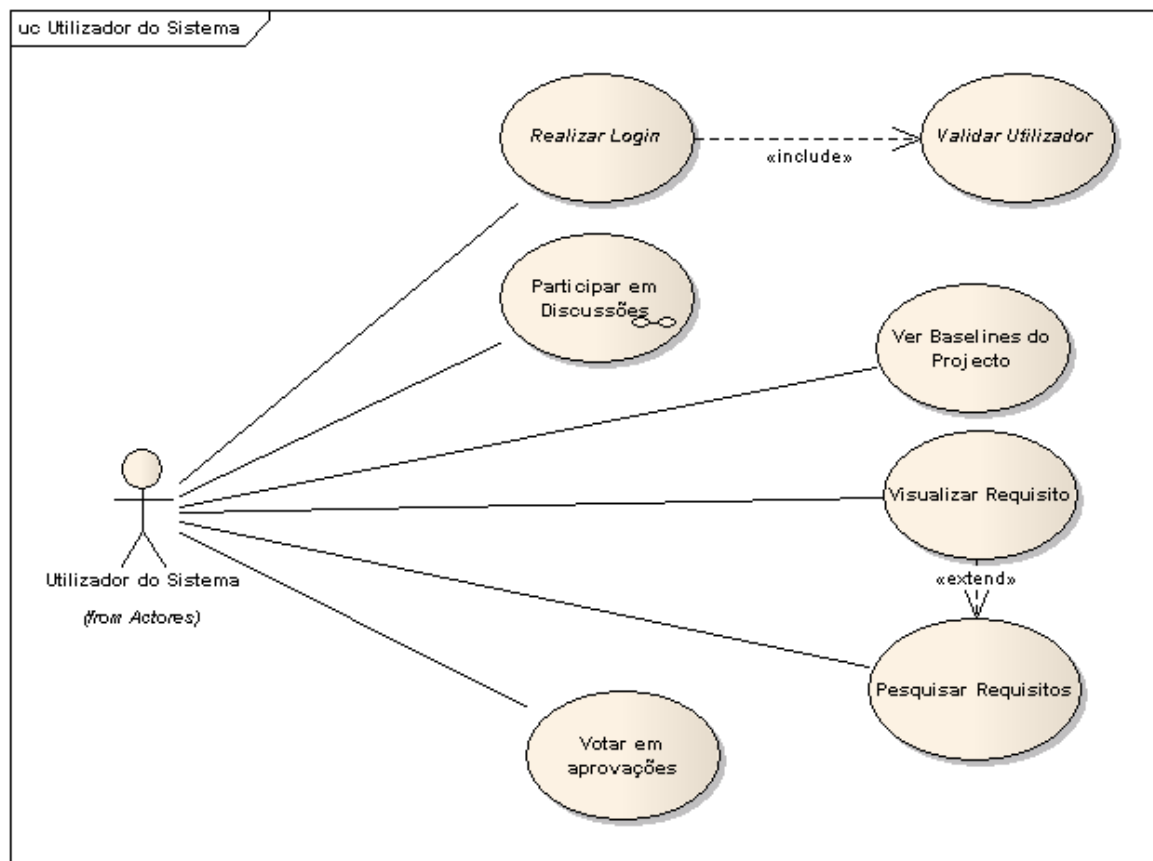


Figura 11: Diagrama de Casos de utilização do Utilizador do Sistema

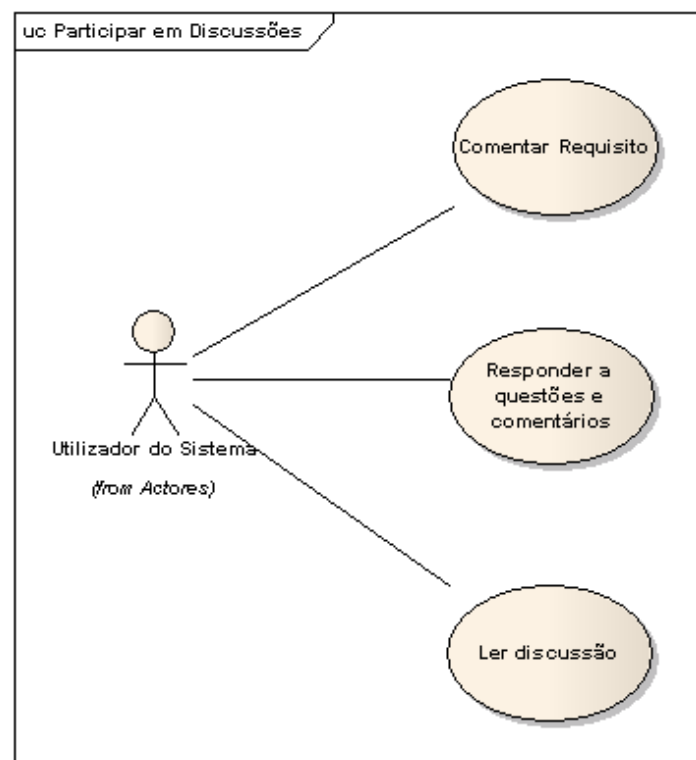


Figura 12: Diagrama de Casos de utilização do Utilizador do Sistema (Participar em Discussões)

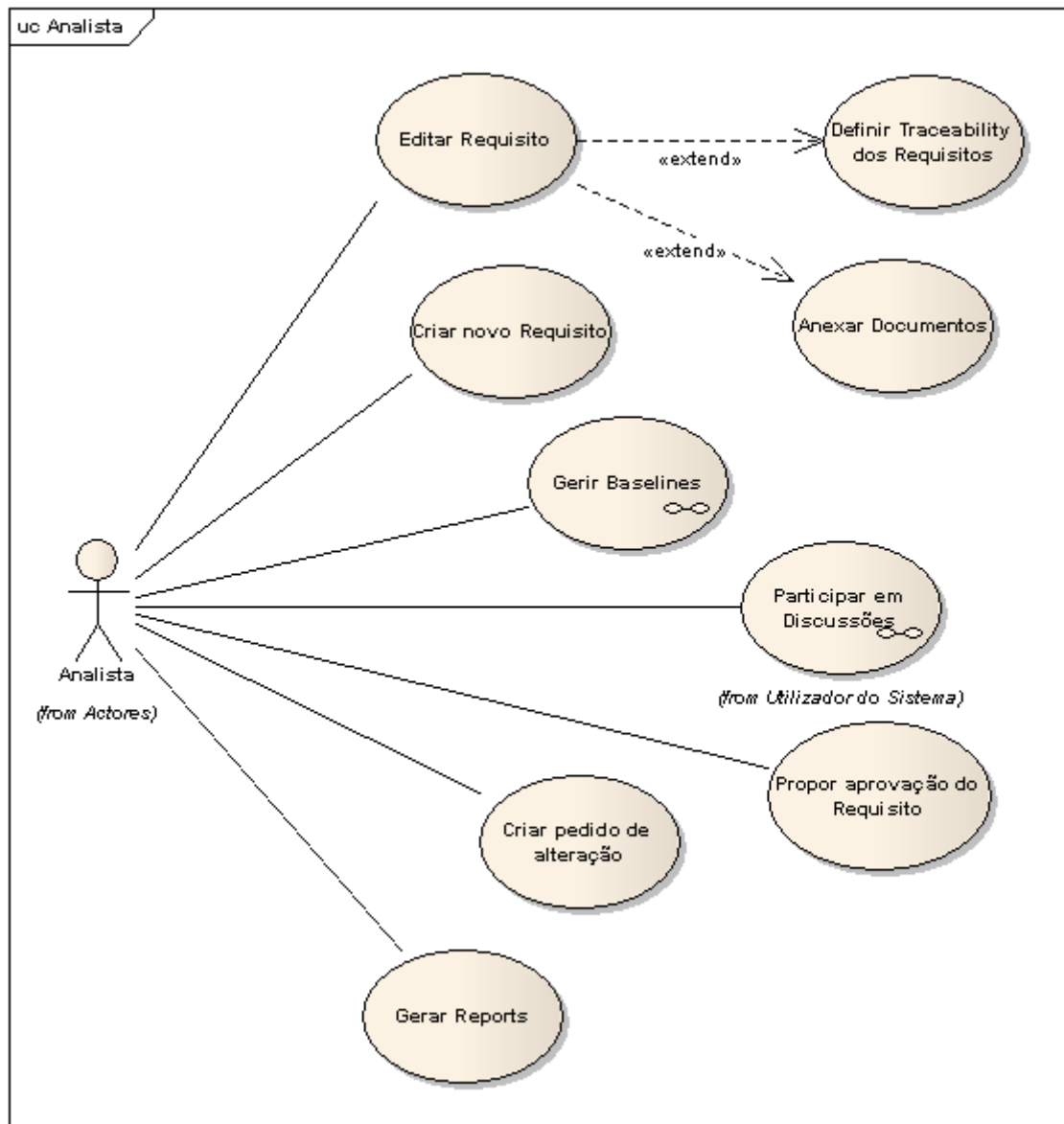


Figura 13: Diagrama de Casos de utilização do Analista

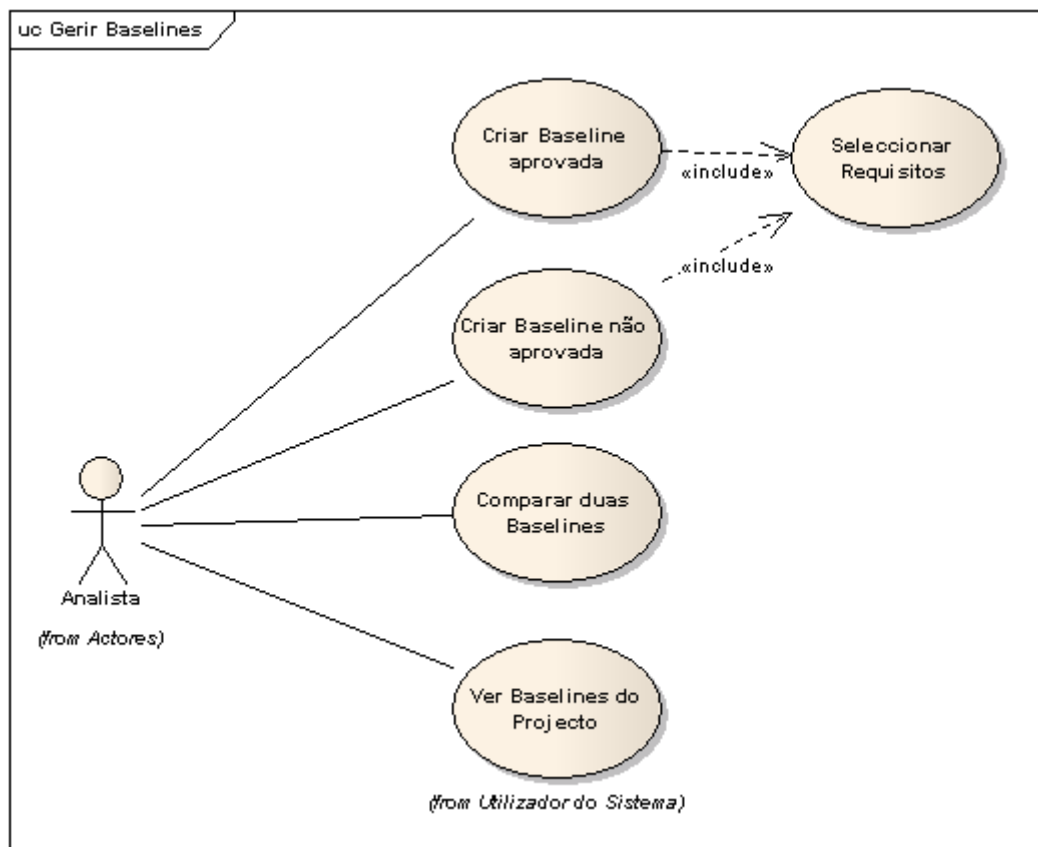


Figura 14: Diagrama de Casos de utilização do Analista (Gerir *Baselines*)

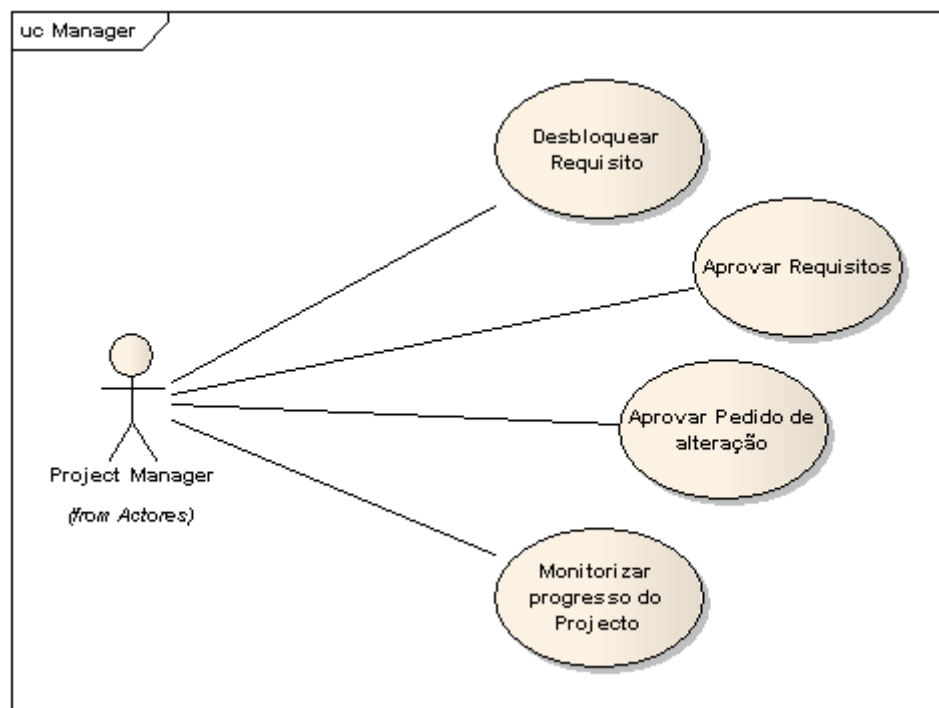


Figura 15: Diagrama de Casos de utilização do Gestor do Projecto

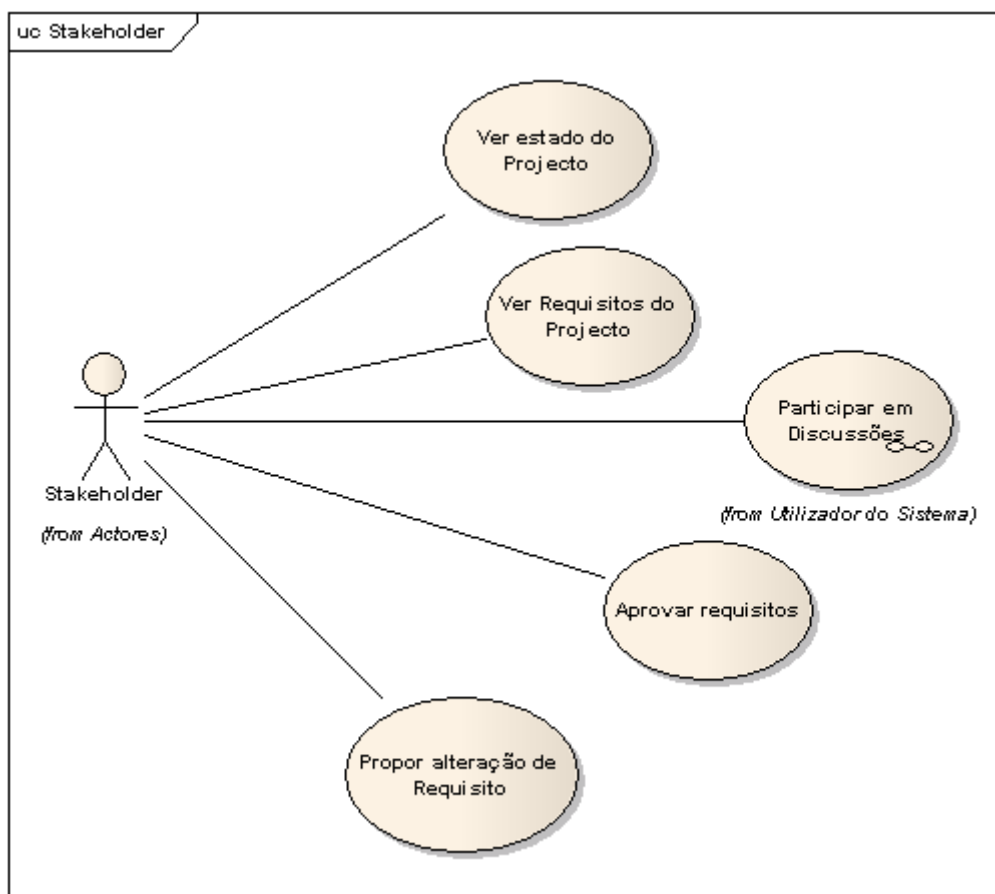


Figura 16: Diagrama de Casos de utilização do *stakeholder*

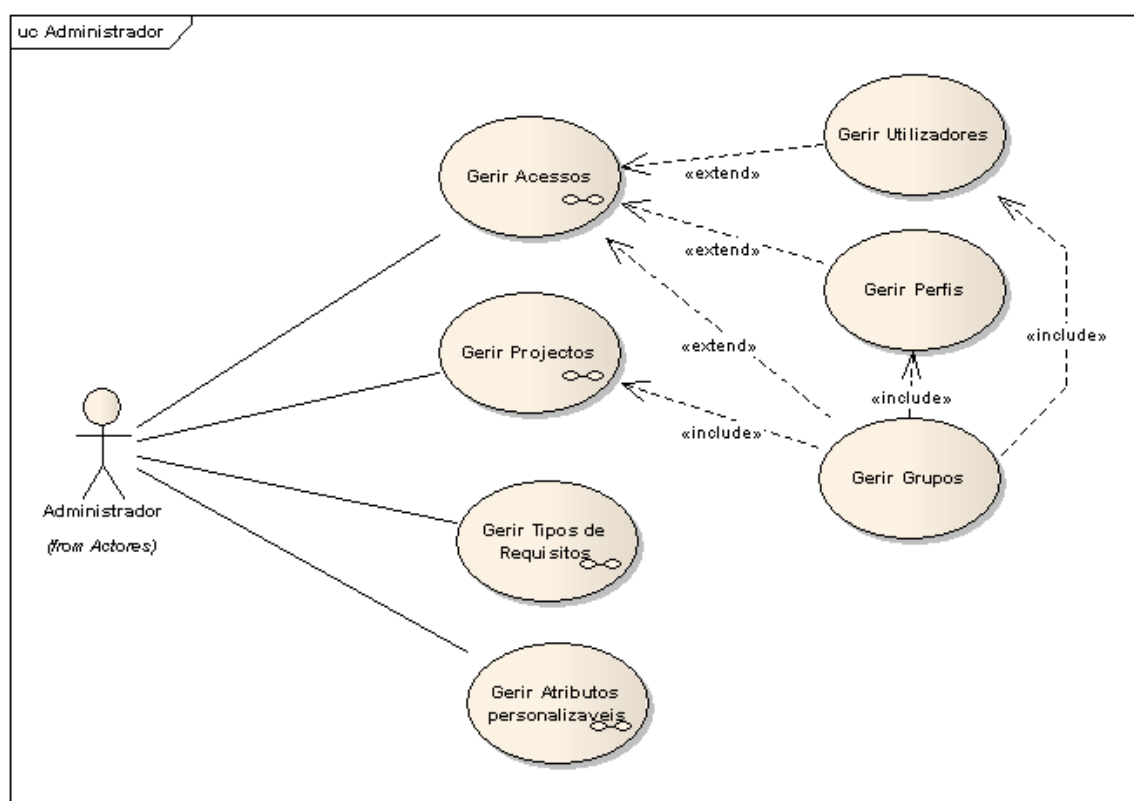


Figura 17: Diagrama de Casos de utilização do Administrador

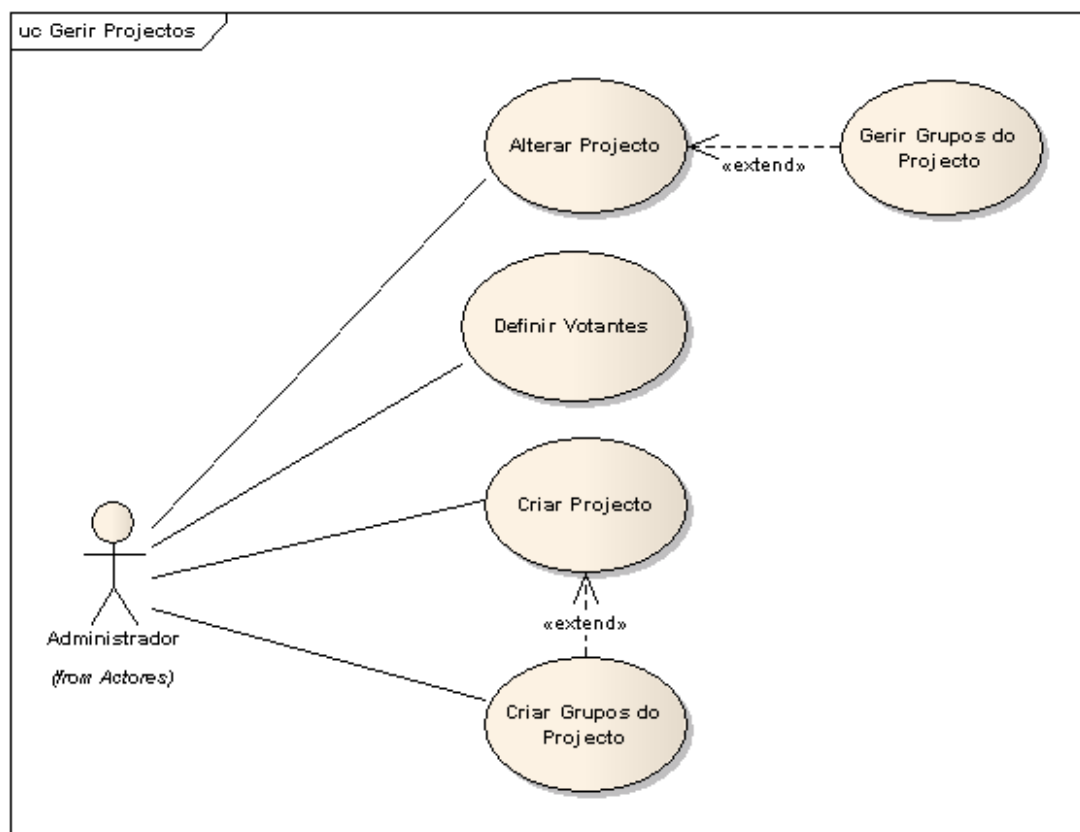


Figura 18: Diagrama de Casos de utilização do Administrador (Gerir Projectos)

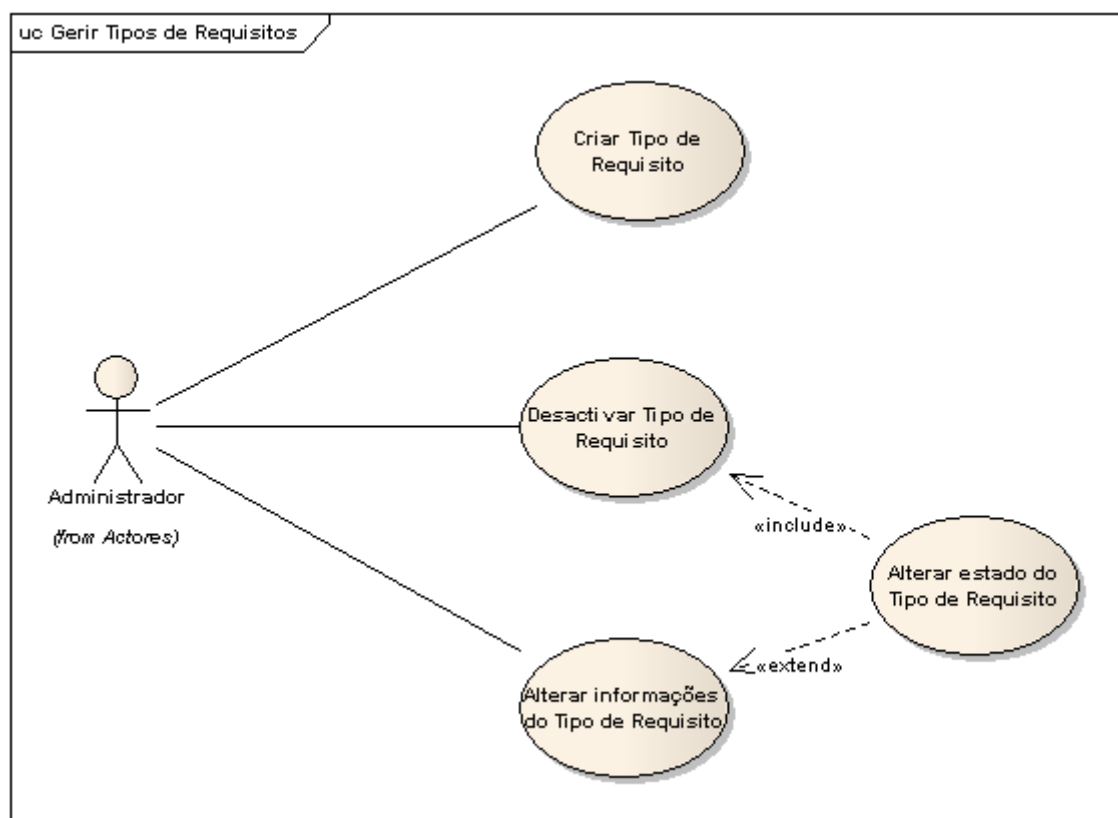


Figura 19: Diagrama de Casos de utilização do Administrador (Gerir tipos de Requisitos)

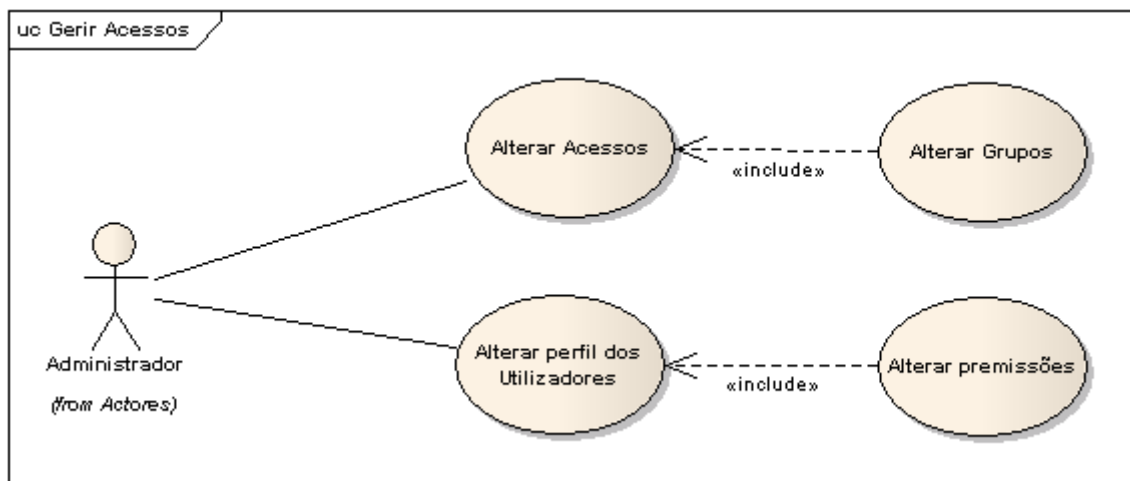


Figura 20: Diagrama de Casos de utilização do Administrador (Gerir Acessos)

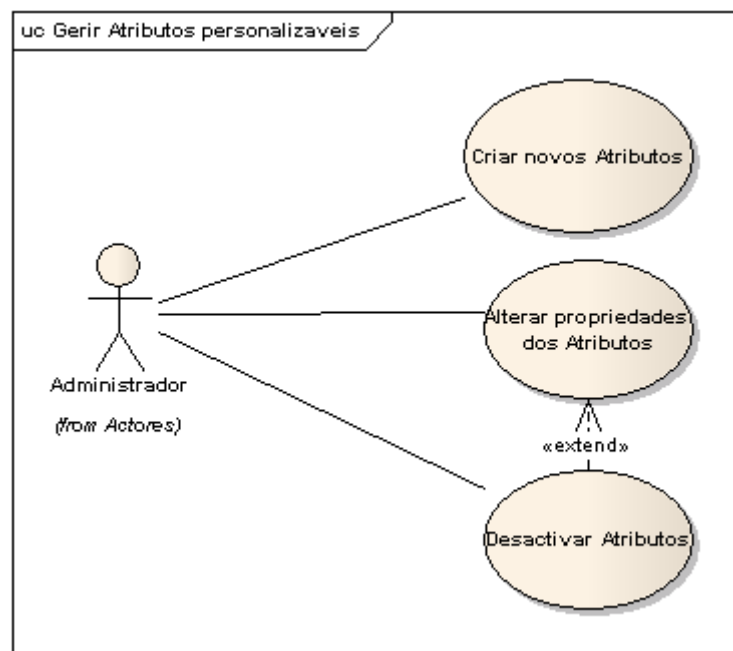


Figura 21: Diagrama de Casos de utilização do Administrador (Gerir Atributos)

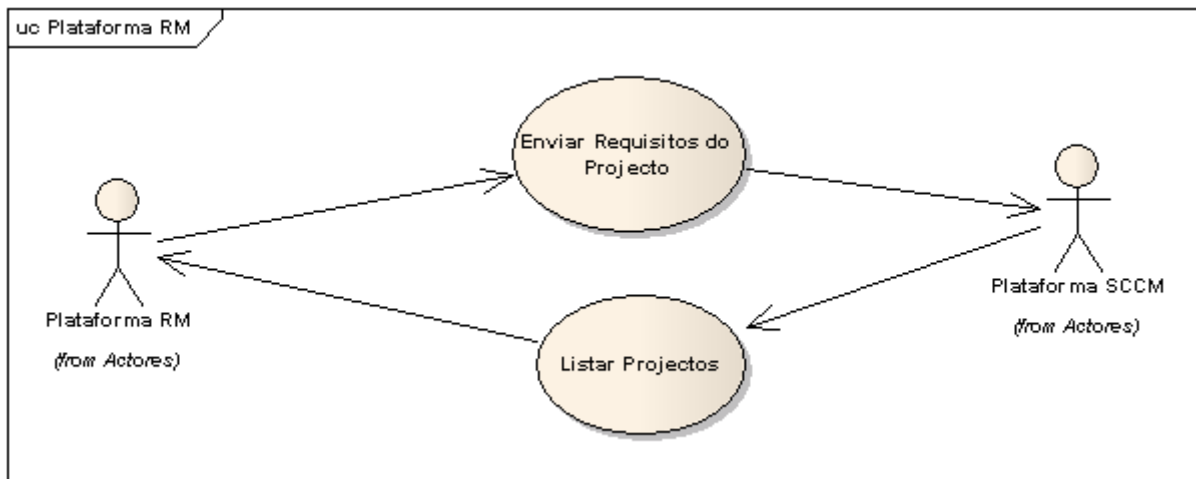


Figura 22: Diagrama de Casos de utilização da Plataforma RM

3.3.2 Requisitos da Aplicação

Através dos diagramas de casos de uso apresentados na secção anterior e tendo ainda como base o estudo realizado sobre a temática abordada pelo trabalho foi possível especificar os requisitos do sistema a desenvolver.

Devido ao processo de desenvolvimento utilizado, é previsível que venham a ocorrer alterações aos requisitos no decorrer do projecto (algo comum na maior parte dos projectos realizados actualmente). Para que fosse possível gerir os diversos requisitos ao longo do tempo foi utilizada a ferramenta RTH, uma das ferramentas analisadas anteriormente (em 2.4.2) de código aberto, para auxiliar esta tarefa de gestão.

Com a utilização desta ferramenta CASE foi possível não só gerir os requisitos do projecto, mas também adquirir ainda uma maior sensibilidade acerca das necessidades dos utilizadores deste tipo de ferramentas, o que no meu entender foi também um factor importante para melhorar a qualidade do sistema desenvolvido.

Foram levantados três tipos de requisitos, os requisitos de negócio (requisitos de alto nível), requisitos funcionais e não funcionais. A figura seguinte apresenta a estrutura em árvore dos requisitos subdividida por tipos e posteriormente por subsistemas do projecto.

NOE01-07005-Plataforma de RM - REQUIREMENTS - FOLDER VIEW

logged in as vrodrigues
 2008-08-13 15:10:55 (BST) [Switch Project](#)

[Home](#) | [Requirements](#) | [Release](#) | [Manage](#) | [User](#) | [Logout](#)

[Requirements](#) | [Folder View](#) | [Add Requirement - Record](#) | [Add Requirement - File](#) | [Notifications](#)

Doc Type	Status	Area Covered	
<div>▼</div> Functionality <div>▼</div>	<div></div> Assigned Release <div>▼</div>	<div>▼</div> Priority <div>▼</div>	Search

- [-] Business Requirements
 - [+] [BR-Gestão de Utilizadores](#)
 - [+] [BR-Controlo de Acessos](#)
 - [+] [BR-Gestão de Projectos](#)
 - [+] [BR-Gestão de Requisitos](#)
 - [+] [BR-Integração com Gestão de Testes](#)
 - [+] [BR-Gerar Documentação](#)
 - [+] [BR-Integração com SCCM](#)
- [-] Functional Requirements
 - [+] [FR-Gestão de Utilizadores](#)
 - [+] [FR-Controlo de Acessos](#)
 - [+] [FR-Gestão de Projectos](#)
 - [+] [FR-Gestão de Requisitos](#)
 - [+] [FR-Integração com Gestão de Testes](#)
 - [+] [FR-Gerar Documentação](#)
 - [+] [FR-Integração com SCCM](#)
- [-] Non-Functional Requirements
 - [+] [NFR-Tecnologias](#)
 - [+] [NFR-Usabilidade](#)
 - [+] [NFR-Segurança](#)

Figura 23: Estrutura dos requisitos do sistema.

Seguidamente apresenta-se os requisitos de alto nível levantados e subdivididos pelos seguintes subsistemas:

- Gestão de Utilizadores
- Controlo de Acessos
- Gestão de Projectos
- Gestão de Requisitos
- Gerar Documentação
- Integração com a Gestão de Testes
- Integração com SCCM

Optou-se por não incluir a descrição detalhada dos requisitos, para não prolongar em demasia esta secção. No entanto após cada bloco de requisitos é incluído uma breve descrição dos requisitos apresentados.

Gestão de Utilizadores

ID	Requisitos de Negócio
1	A aplicação deve permitir criar, remover e editar Grupos de Utilizadores.
2	A aplicação deve permitir a criação de perfis de segurança.
3	O sistema deve permitir a alteração dos dados de um utilizador
4	Deve ser possível associar Utilizadores a Grupos.
5	Deve ser possível criar novos Utilizadores do sistema.
6	Deve ser possível remover Utilizadores.
7	O sistema deve suportar a associação de Grupos a perfis de segurança.
8	Os perfis de segurança devem poder ser removidos e alterados.

Tabela 3: Requisitos de negócio da gestão de utilizadores.

O sistema a desenvolver deve suportar os requisitos típicos de gestão de utilizadores e de grupos de utilizadores. No contexto deste sistema um grupo é o nome dado a um conjunto de utilizadores que são associado a um projecto (exemplo: clientes, analistas, programadores). Desta forma um utilizador pode exercer o papel de programador num projecto e de analista em outro. Um utilizador só deve por isso ter acesso a um projecto e aos seus requisitos, se estiver associado a um grupo desse mesmo projecto.

Os perfis de segurança que são associados aos grupos de utilizadores, que definem as permissões de escrita e leitura que cada utilizador tem sobre os vários tipos de requisitos, permitindo desta forma controlar a edição e visualização dos requisitos do projecto.

Gestão de Acessos

ID	Requisitos de Negócio
9	A aplicação só deve permitir o acesso de Utilizadores autenticados.
10	Os Utilizadores do tipo administrador devem ter acesso a um painel de administração.
11	O sistema deve disponibilizar a opção de recuperação da palavra-chave.
12	O Utilizador autenticado deve poder fazer o <i>logout</i> em qualquer altura da aplicação.
13	Deve ser possível criar novos Utilizadores do sistema.
14	A aplicação deve limitar a alteração e criação de requisitos aos utilizadores com permissões de escrita sobre um determinado tipo de requisito.

Tabela 4: Requisitos de negócio da gestão de acessos.

De forma a controlar o acesso dos utilizadores à aplicação, só utilizadores devidamente autenticados podem aceder à mesma.

O sistema deve também restringir o acesso ao painel de administração, a todos os utilizadores que não sejam do tipo administrador. No painel de administração deve ser possível gerir projectos, utilizadores, perfis, tipos de requisitos e atributos específicos dos requisitos.

Todos os requisitos devem estar associados a um tipo, com base nesta associação e com o perfil do grupo do utilizador autenticado é possível controlar quais os requisitos que um utilizador pode criar, editar e visualizar. Desta forma é possível garantir a participação controlada e ordenada dos diversos intervenientes no projecto.

Gestão de Projectos, Tipos e Atributos dos Requisitos

ID	Requisitos de Negócio
15	O sistema deve permitir criar e alterar projectos.
16	Deve ser possível importar requisitos de outros projectos.
17	O sistema deve permitir criar <i>baselines</i> do projecto.
18	O sistema deve permitir criar, editar e remover atributos dos requisitos.
19	A aplicação deve suportar a criação e edição de tipos de requisitos.

Tabela 5: Requisitos de negócio para a gestão de projectos

Os utilizadores do tipo administrador podem criar e editar os projectos da aplicação. Para criar um projecto é necessário definir não só o nome e a descrição deste mas também o gestor do projecto, o gestor de alterações (“*change manager*”), definir grupos de utilizadores e ainda quais os utilizadores que podem participar nas votações de aprovação de requisitos e de pedidos de alteração a requisitos.

Quando é criado um novo projecto deve ser ainda possível adicionar requisitos iniciais ao projecto, de forma a promover a reutilização de requisitos.

A gestão de atributos personalizáveis permite adaptar a informação de cada requisito às necessidades de cada organização e de cada projecto, pois cada organização e projecto tem as suas particularidades. Por cada atributo criado deve ser definido não só o seu nome e descrição mas também o tipo de dado, que pode ser booleano, inteiro ou texto.

Seguindo o mesmo conceito dos atributos, os tipos de requisitos também podem ser personalizáveis, ou seja é possível criar e editar tipos de requisitos através da definição do nome, da abreviatura e descrição do tipo de requisito.

Por fim, é possível ainda criar *baselines* do projecto. No contexto desta ferramenta uma *baseline* é uma “imagem” de uma versão de cada requisito no repositório do projecto. Esta ferramenta deve permitir criar *baselines* fechadas (englobam todos os requisitos do projecto) e abertas (podem só incluir parte dos requisitos do projecto)

Gestão de Requisitos

ID	Requisitos de Negócio
20	O sistema deve permitir a criar novos requisitos.
21	A aplicação deve permitir a alteração de requisitos.
22	A aplicação deve permitir a associação entre requisitos de diversos tipos.
23	O sistema deve apresentar uma matriz de rastreabilidade dos requisitos.
24	Deve ser possível aplicar filtros à matriz de rastreabilidade.
25	Deve ser possível efectuar pesquisas de requisitos por nome e identificador único.
26	Deve ser possível anexar ficheiros a um requisito.
27	O sistema deve realizar o controlo de versões do requisito.
28	O sistema deve permitir controlar o estado de um requisito ao longo do tempo.
29	O sistema deve suportar a navegação dos requisitos através de um menu em árvore.
30	A página inicial da aplicação deve apresentar um sumário do estado dos requisitos.
31	O sistema deve detectar alterações ao requisito e assinalar as suas ligações como "sujas".
32	O sistema deve permitir definir hierarquias entre os requisitos.
33	Deve ser possível visualizar os requisitos numa estrutura em tabela.
34	Deve ser possível aplicar filtros à tabela de requisitos.
35	O sistema deve bloquear a edição de requisitos aprovados.
36	Um requisito só deve passar para o estado aprovado depois da aprovação dos utilizadores votantes.
37	Um requisito bloqueado só pode ser desbloqueado após a aprovação pelos utilizadores votantes.
38	Um requisito só deve passar para o estado "Aprovado", após a sua aprovação pelos Utilizadores.
39	O sistema deve apresentar quais as aprovações em curso.
40	O sistema deve permitir aos utilizadores votantes votar a favor ou contra a aprovação de requisitos e de pedidos de alteração.
41	Requisitos e pedidos de alteração só devem ser aprovados com todos os votos a favor.
42	O Sistema deve listar todos os requisitos aprovados e reprovados.
43	O Sistema deve listar todos os pedidos de alteração aprovados e reprovados

Tabela 6: Requisitos de negócio para a gestão dos requisitos.

No conjunto de requisito apresentado na Tabela 6 são definidos os requisitos que permitem gerir os requisitos da aplicação. O sistema deve permitir a criação de novos requisitos associados a um determinado tipo (requisitos de negócio, funcional, não funcional ou outro personalizado pelo administrador da aplicação). Para cada requisito deve ser guardado o seu nome, descrição, tipo,

estado, utilizador que criou o requisito, utilizador que realizou a última alteração, data de criação e data de última alteração.

A definição das ligações de rastreabilidade entre requisitos deve ser permitida. Sempre que um requisito é alterado o sistema deve alertar o utilizador de todos os requisitos que podem ter sido afectados (tendo como base as ligações entre os requisitos).

Para analisar o impacto de alterações aos requisitos deve ser apresentado uma matriz de rastreabilidade que permite visualizar de uma forma global as ligações entre os requisitos do projecto.

O sistema suporta ainda o versionamento dos requisitos, desta forma sempre que é realizada uma alteração a um requisito é criada uma nova versão deste.

A aplicação deve ainda suportar a aprovação de requisitos através de um sistema de votação incorporado. Sendo assim, um sistema só deve passar para o estado aprovado após a aprovação deste pelos utilizadores votantes. Um requisito ao ser aprovado é bloqueado não permitindo a alteração tradicional do requisito.

Integração com Gestão de Testes

ID	Requisitos de Negócio
44	O sistema deve ter um adaptador que permita a integração com ferramentas de testes.
45	O sistema deve suportar a associação de um projecto com um projecto de testes.
46	O adaptador de Integração deve exportar os testes a realizar ao projecto.
47	O sistema deve permitir associar testes a requisitos.
48	O adaptador do sistema deve ser configurável.
49	O sistema deve realizar a sincronização automática com a ferramenta de testes.
50	É desejável que exista um interface para configurar o adaptador de integração.
51	O sistema deve testar a integração correcta do adaptador à ferramenta externa.
52	O sistema deve permitir activar e desactivar a integração com ferramentas de gestão de testes.

Tabela 7: Requisitos de negócio para a integração com ferramentas de gestão testes.

O conjunto de requisitos apresentados na tabela anterior define a forma de integração idealizada para integrar a ferramenta de gestão de requisitos com uma ferramenta externa de gestão de testes. No entanto, por questões de limitação de tempo estes não foram implementados.

Gerar Documentação

ID	Requisitos de Negócio
52	O sistema deve gerar um relatório do sumário dos requisitos de um projecto.
53	A aplicação deve gerar um relatório do estado actual do projecto.
54	O sistema deve gerar um relatório detalhado dos requisitos do projecto.
55	A aplicação deve suportar a geração de um relatório de comparação de <i>baselines</i> .
56	O sistema deve gerar um relatório do histórico dos requisitos de um projecto.
57	A aplicação deve permitir exportar um relatório para o formato Excel.
58	A aplicação deve permitir exportar um relatório para o formato PDF.
59	O sistema deve possibilitar a geração dos relatórios personalizáveis
60	O sistema deve permitir criar, editar e remover relatórios personalizáveis.

Tabela 8: Requisitos de negócio para a geração de documentação.

Na Tabela 8 é apresentado um grupo de requisitos que define as funcionalidades que a ferramenta deve comportar em relação à exportação de informação sobre os requisitos da ferramenta.

Todos os relatórios gerados do sistema devem poder ser exportados para os formatos xls e PDF. Os requisitos que dizem respeito a possibilidade de criar *reports* personalizados não foram implementados dado o limite de tempo associado ao projecto. O *report* do estado actual dos requisitos deve apresentar os requisitos segundo a sua hierarquia.

Integração com SCCM

ID	Requisitos de Negócio
61	O sistema deve permitir o envio de requisitos de <i>baselines</i> aprovadas.
62	O sistema deve notificar o SCCM de novas <i>baselines</i> aprovadas.
63	O sistema deve associar um projecto a uma aplicação SCCM.
64	O sistema deve permitir activar e desactivar a integração com o SCCM.

Tabela 9: Requisitos de negócio para a integração com ferramentas SCCM.

Quando é criada uma *baseline* aprovada, esta passa a estar disponível para consulta de uma ferramenta SCCM através de uma interface orientada aos serviços. No entanto, esta consulta só pode ser realizada se a ferramenta de gestão tiver a opção de integração com uma ferramenta SCCM activa. Através deste método é possível enviar automaticamente os requisitos para uma ferramenta que realiza o controlo do processo de desenvolvimento, de configurações e releases associadas aos requisitos desenvolvidos.

Este ponto deve ser alvo de uma explicação prévia do que são ferramentas de SCCM, quais as vantagens da integração entre estes dois tipos de aplicações e ainda o contexto em que surgiu a necessidade de realizar esta integração em particular.

Como foi referido no capítulo 2.3.4, uma ferramenta de gestão de requisitos pode interagir com diversas ferramentas que participam em diferentes fases do desenvolvimento de um sistema informático. Uma das ferramentas com a qual este tipo de ferramentas pode interagir são ferramentas de controlo de versões. As ferramentas de Software Change and Configuration Management (SCCM) são ferramentas que visam controlar todo o processo de desenvolvimento de software, sendo que uma das suas tarefas é o controlo de versões de software.

Actualmente existe uma elevada preocupação por parte das organizações em controlarem todo o processo de desenvolvimento. Esta preocupação também era partilhada pela organização que apoiou o desenvolvimento desta tese. Assim, aliado ao facto de durante o desenrolar desta tese de mestrado, decorrer em simultâneo uma outra dissertação de mestrado que teve como produto final uma ferramenta de SCCM. Existiu uma interacção de ambas as partes a fim de levantar os requisitos de maior importância para que esta integração.

As principais vantagens de integrar a plataforma de gestão de requisitos com uma ferramenta de SCCM são as seguintes:

- Transição automática das *baselines* aprovadas para desenvolvimento para a ferramenta de controlo do processo de desenvolvimento;
- Comunicação automática de alterações de requisitos:
 - Notificação de requisitos alterados;
 - Notificação de novos requisitos de um projecto;
 - Notificação da remoção de requisitos do projecto (cancelados).
- Armazenamento dos requisitos como um CI da ferramenta de SCCM;
- Optimização do processo de desenvolvimento de software e de comunicação de alterações ao projecto durante o processo.

3.3.3 Protótipo Funcional

Após o levantamento de uma primeira versão dos requisitos para uma aplicação de gestão de requisitos, decidiu-se criar um protótipo funcional do sistema com o objectivo de validar junto de potenciais utilizadores os requisitos até à altura levantados, limitar o projecto através da definição de uma meta atingível no tempo disponível, capturar novos requisitos descobertos pela utilização do protótipo por parte dos utilizadores e ainda avaliar a usabilidade do interface proposto pelo protótipo funcional.

O protótipo funcional foi criado recorrendo à ferramenta Microsoft Visio e a um conjunto de formas disponibilizadas pela GUUI Web Prototyping Tool, que permite criar protótipos interactivos

de sites e aplicações Web. A razão da escolha destas ferramentas para a criação do protótipo funcional deveu-se à facilidade de criação do protótipo através do simples *drag-and-drop* de controlos nas páginas e de ligação entre estas, o que permitiu criar com rapidez um protótipo funcional do sistema.

Nesta fase do projecto foram realizadas reuniões com potenciais utilizadores de aplicações de gestão de requisitos. A abordagem tomada nesta fase foi a reunião de utilizadores chave de aplicações deste género, o que permitiu obter um maior número de críticas e sugestões segundo diferentes pontos de vista de cada utilizador. Os utilizadores que testaram o protótipo foram os seguintes:

- Uma experiente gestora de projectos informáticos;
- Dois experientes consultores na área da qualidade de software;
- Diversos estudantes de engenharia de informática e computadores.

Com base nas críticas e sugestões dos diversos utilizadores o protótipo foi evoluindo, o que se traduziu em três versões do mesmo. Neste documento é apresentado a versão final do protótipo após a avaliação dos utilizadores e consequente alteração deste (ver figuras abaixo).



Figura 24: Página inicial do protótipo final do sistema.

[Help](#)
[Sair](#)

Home	Requirements	Baselines	Reports	Administration
------	--------------	-----------	---------	----------------

Add New Requirement
Requirements List
Traceability matrix
Change Request

Project1

Requirements

Business Requirements

- 1- Register User
- 3- Req 3
- 2- Register User

Functional Requirements

Non-Functional Requirements

Search

Geral
Traceability
History
Discussion

Name:

Description:

Lorem ipsum onsectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat.

Type: Business Requirements

State: New

Custom2:

Files: [Register User - Use case](#)
[Interface](#)

ID: 1

Version: 0.3

Last Update: 1-11-2007

By: util1

Custom1:

[Mensagens de Erro/Alerta]

Add File
New Clhid
Save Requirement

Figura 25: Página de consulta e edição de um requisito do protótipo final.

[Help](#)
[Sair](#)

Home	Requirements	Baselines	Reports	Administration
------	--------------	-----------	---------	----------------

Add New Requirement
Requirements List
Traceability matrix
Change Request

Project1

Requirements

Business Requirements

- 1- Register User
- 3- Req 3
- 2- Register User

Functional Requirements

Non-Functional Requirements

Search

Traceability Matrix

	11- Req FR	15- Req FR	11- Req FR				
1-Register User		🔴					
3- Req 3	🟢						
2- Register User	🟢						

Filter

From

Requirement

Type Business Requirements

State New

To

Requirement

Type Functional Requirements

State New

[Mensagens de Erro/Alerta]

Apply Filter

Figura 26: Página da matriz de rastreabilidade entre requisitos do protótipo.

Figura 27: Página de administração de um projecto do protótipo.

A partir do *feedback* obtido pelos utilizadores do protótipo, alguns novos requisitos foram capturados. A lista de requisitos apresentada na secção anterior já reflecte os requisitos que derivaram de sugestões e críticas dos utilizadores que experimentaram este protótipo, no entanto é importante mencionar aqui as principais sugestões dadas pelos utilizadores que permitiram aperfeiçoar a ferramenta desenvolvida.

- Necessidade de importar requisitos de projectos anteriores, de forma a permitir a reutilização de requisitos com base em projectos bem sucedidos no passado;
- Apresentar o estado actual dos requisitos do projecto, através da apresentação da percentagem de requisitos em cada estado, desta forma é possível controlar de forma mais eficiente a evolução global do projecto;
- Permitir a exportação de relatórios para o formato xls;
- Permitir a aplicação de filtros à tabela de requisitos e à matriz de rastreabilidade. Os principais filtros sugeridos foram o filtro por tipo e estado dos requisitos.

Além dos requisitos funcionais recolhidos a partir da interacção dos utilizadores com o protótipo, foi ainda realizada uma avaliação da interface da aplicação a implementar. Esta avaliação foi realizada através da avaliação da satisfação dos utilizadores relativamente ao protótipo apresentado, facilidade de execução das principais tarefas de gestão de requisitos e ainda através da avaliação do protótipo segundo as dez heurísticas de usabilidade desenvolvidas por Jakob Nielsen (23).

A partir da avaliação heurística do protótipo foram detectadas algumas deficiências do interface apresentado. Os problemas detectados incidiram principalmente na falta de ajuda e documentação

associada à aplicação, na facilidade de reconhecimento e identificação de acções e na ajuda ao utilizador no reconhecimento e recuperação de erros. Com base nestes problemas detectados foram realizadas alterações aos protótipos, nomeadamente inclusão de um painel de ajuda, associação de ícones auto explicativos de tarefas a botões e hiperligações e a definição de uma zona de apresentação de mensagens de erros.

Após a alteração do protótipo, de forma a reflectir os novos requisitos funcionais e de usabilidade detectados, o protótipo foi bem aceite pelos utilizadores que culminou com uma aprovação formal dos requisitos do sistema a desenvolver.

3.4 Desenho

Nesta fase do projecto é realizada a definição da solução. Com base nos resultados produzidos pela tarefa de análise procedeu-se à especificação técnica e das características do sistema a desenvolver.

3.4.1 Diagrama de Actividades

Os diagramas de actividade são reconhecidos como adequados à modelação da visão funcional de um sistema pois permitem descrever a lógica dos seus processos ou das suas funções. Com base neste conceito considerou-se adequado descrever o comportamento de alguns dos principais processos que a ferramenta suporta. (20)

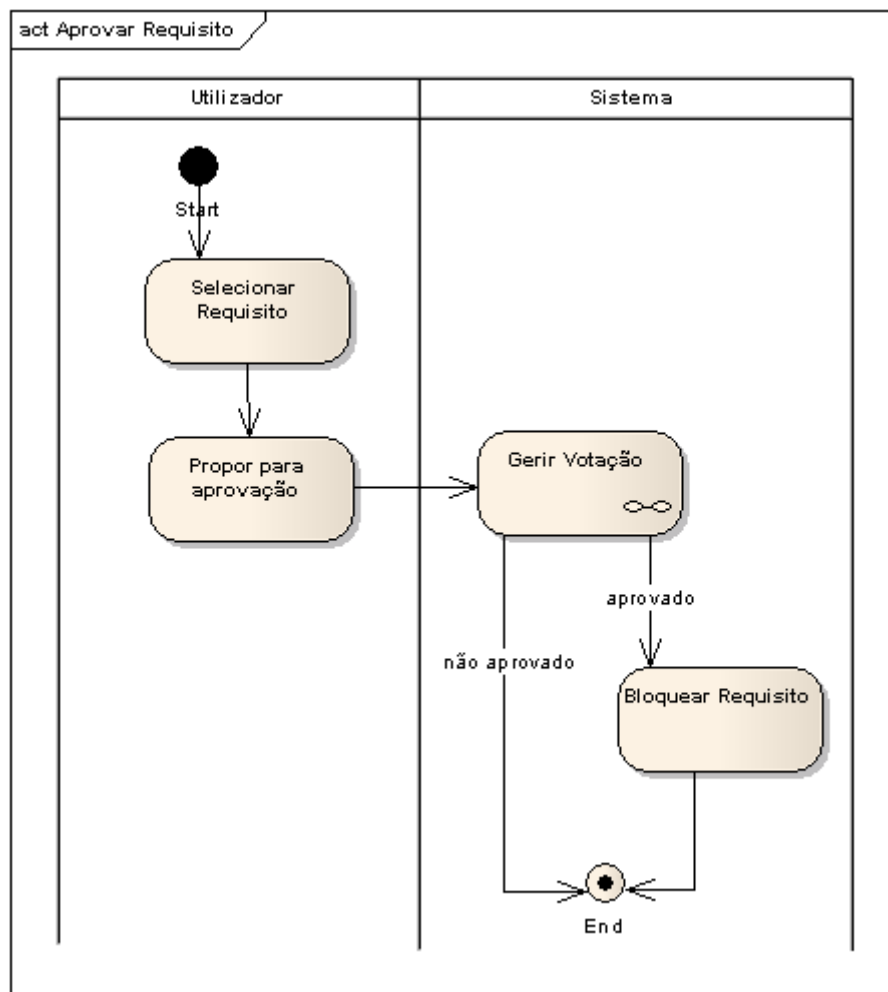


Figura 28: Processo de Aprovação de um Requisito.

O diagrama anterior representa o comportamento do sistema para que seja aprovado um requisito. Note-se que o utilizador ao propor para aprovação o requisito, acciona o processo de votação do sistema, representado pela actividade composta “Gerir Votação”. Caso o *output* da votação seja positivo o requisito passa a estar aprovado e é bloqueado.

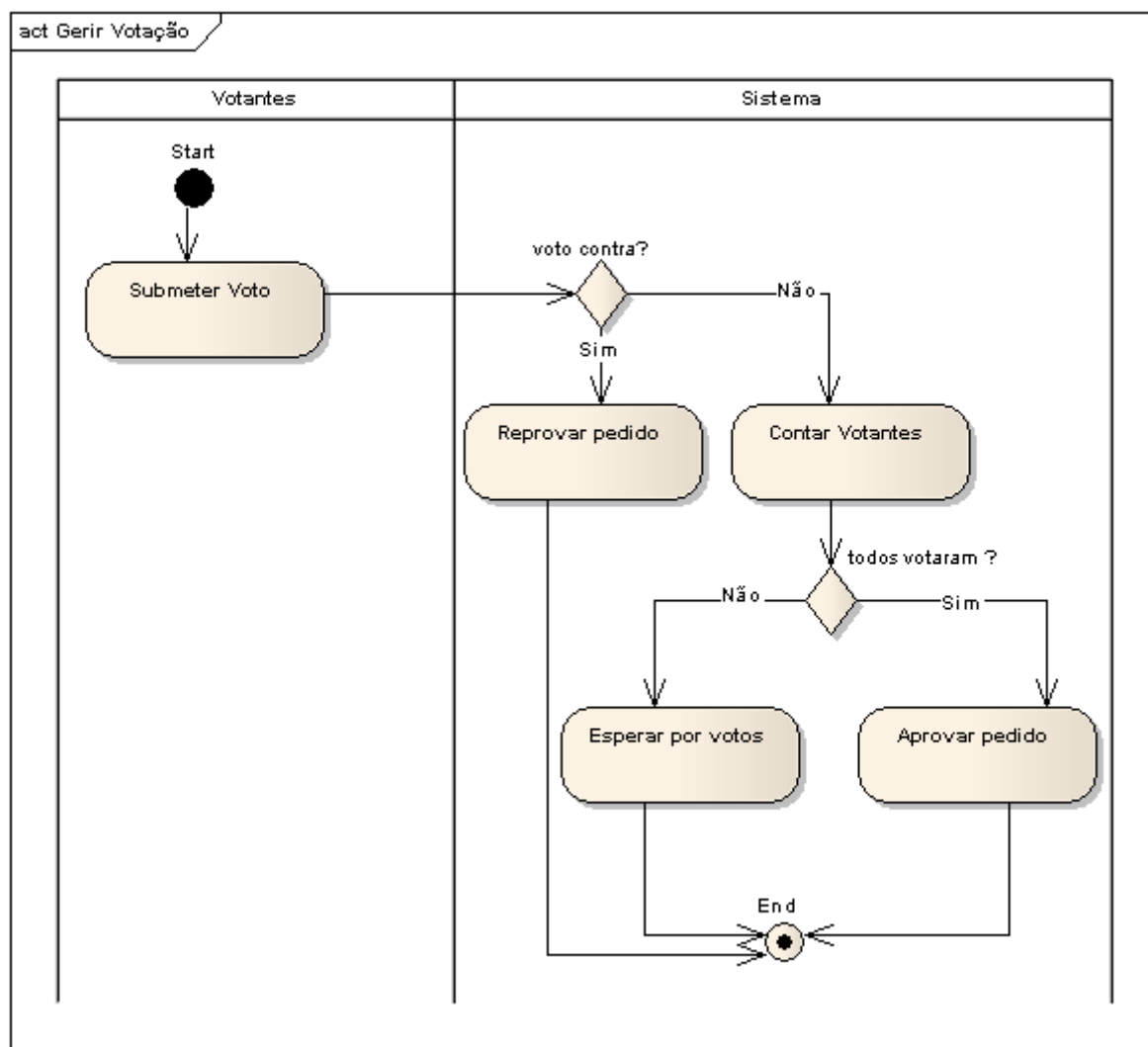


Figura 29: Processo de gerir votação.

A Figura 29 ilustra o processo de votações do sistema. Como pode ser observado o sistema só aprova um pedido de aprovação (que pode ser um pedido de alteração ou de aprovação de um requisito) se todos os utilizadores votantes votarem a favor da sua aprovação.

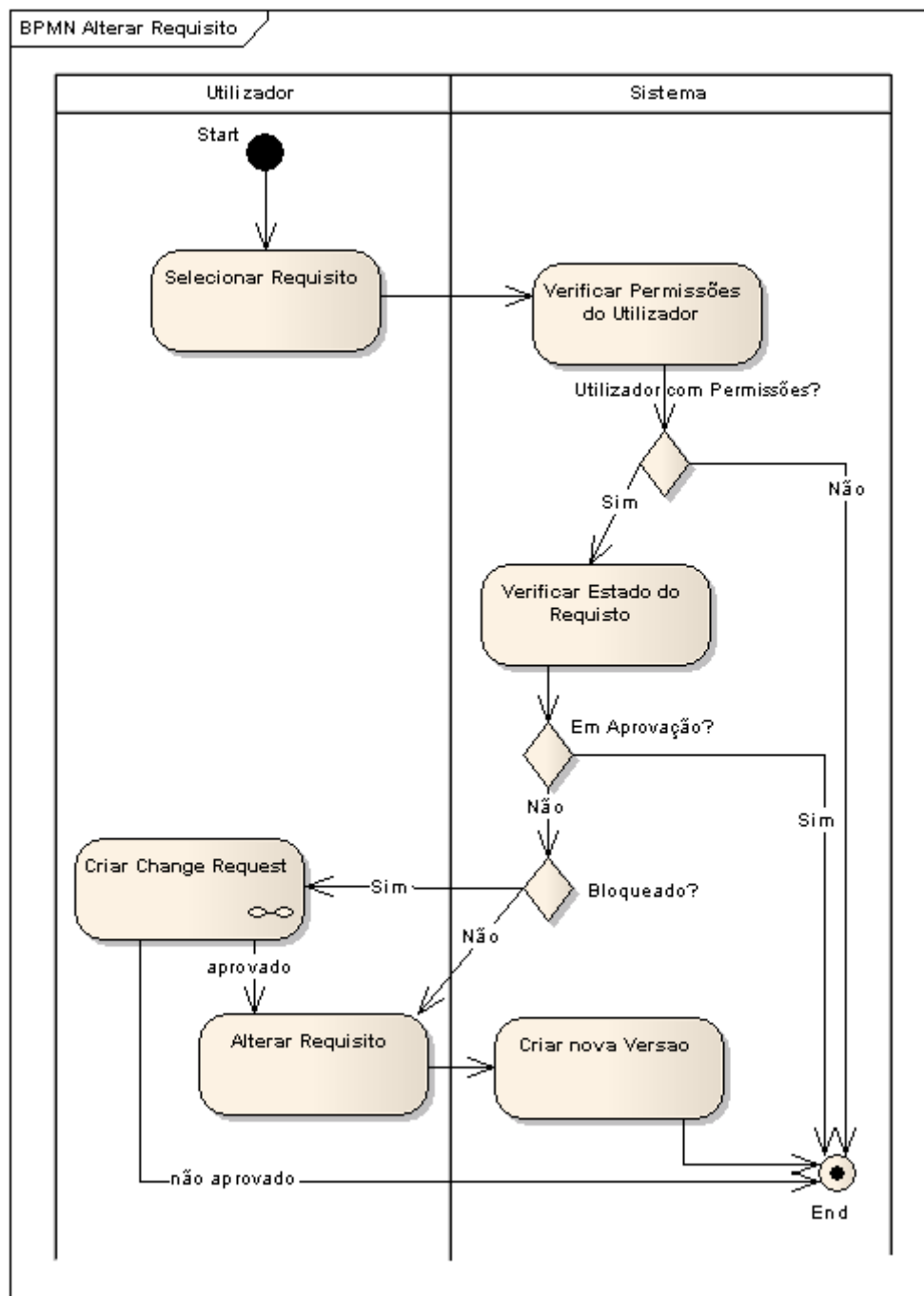


Figura 30: Processo de Alterar um requisito.

O processo de alterar um requisito representado pela Figura 30, explica o controlo realizado sobre um requisito de forma a só permitir a edição de requisitos sobre os quais um utilizador tenha permissões de escrita e ainda se o requisito não estiver nem em aprovação ou bloqueado (o que implica que já tenha sido aprovado anteriormente).

O diagrama sublinha ainda a importância de efectuar um pedido de alteração de um requisito, dando inicio a um processo de análise e aprovação do pedido por parte dos outros utilizadores do sistema (ver Figura 31). Se o pedido de alteração for aprovado todos os requisitos associados ao pedido de alteração são desbloqueados e o utilizador com permissões pode alterar o requisito.

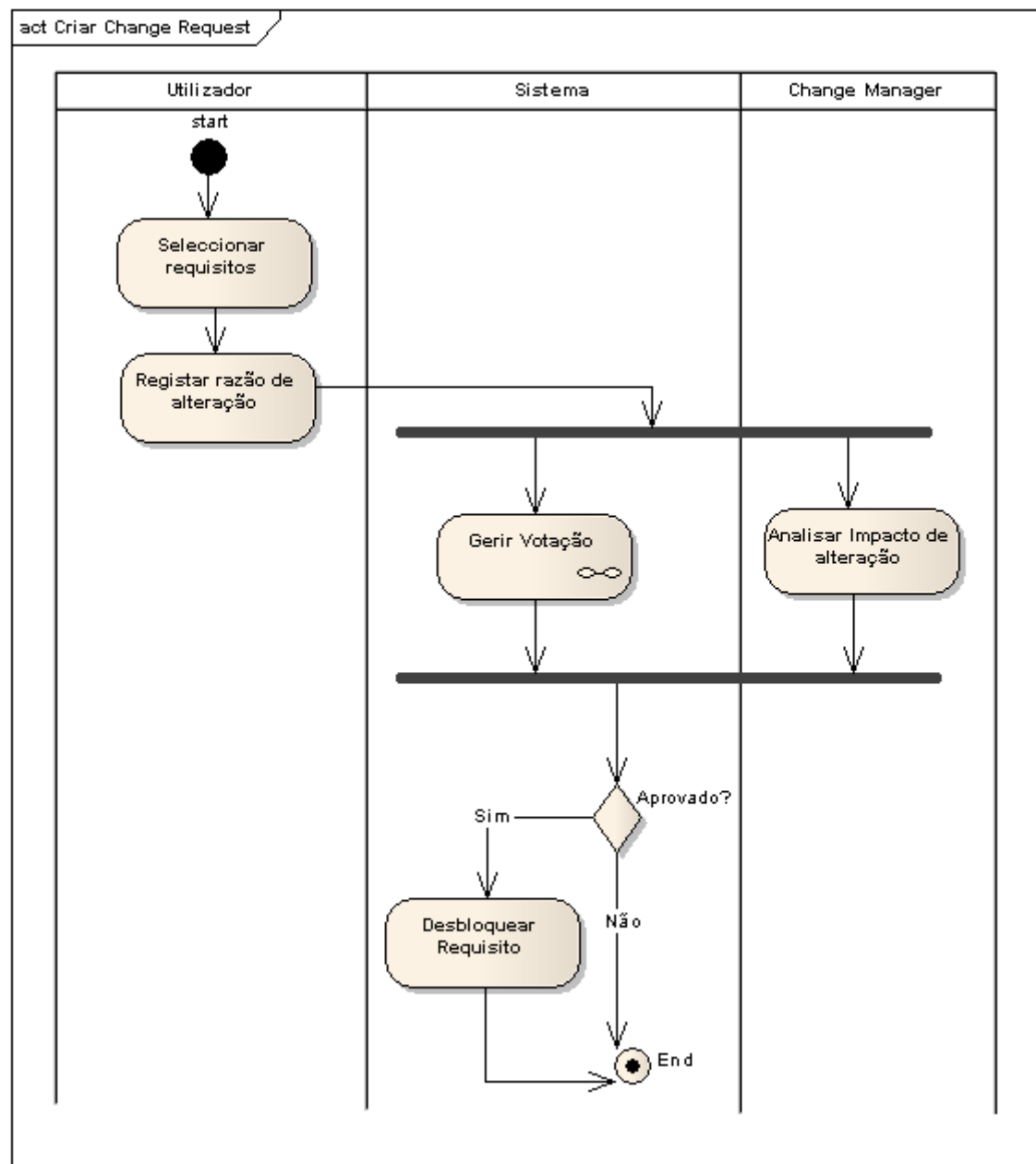


Figura 31: Processo para efectuar um pedido de alteração.

3.4.2 Modelo de Domínio

Nesta fase optou-se também por produzir um modelo de domínio de acordo a temática de gestão de requisitos e os requisitos do sistema produzidos. Este modelo permite descrever as várias identidades envolvidas no sistema e a suas relações. Este tipo de modelo providencia uma visão estrutural do sistema, fornecendo uma documentação dos conceitos chave e do vocabulário do sistema. (20)

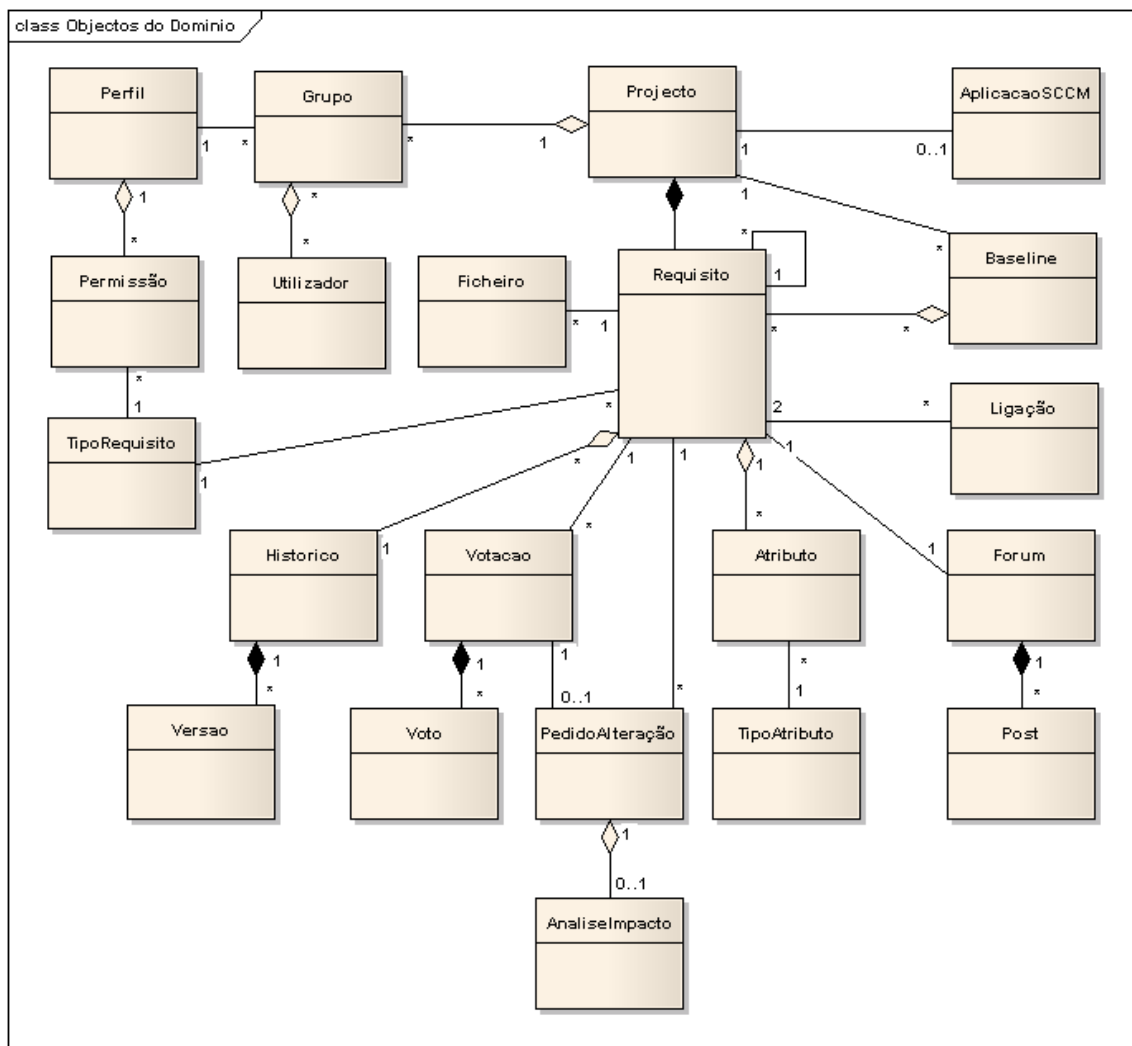


Figura 32: Modelo de Domínio do sistema de gestão de requisitos.

3.4.3 Diagrama de estados

Um diagrama de estados permite ilustrar o comportamento interno de um determinado objecto, subsistema ou sistema global. Estes diagramas representam os possíveis estados de um objecto, as correspondentes transições entre estados e os eventos que fazem desencadear as transições. (20)

No contexto desta tese o diagrama de estados desenhado (ver Figura 33) representa a abordagem tomada para gerir um requisito ao longo do tempo. Este diagrama é importante pois apresenta o ciclo de vida de um requisito no sistema de gestão de requisitos implementado e os eventos e acções que desencadeiam a transição entre os diferentes estados.

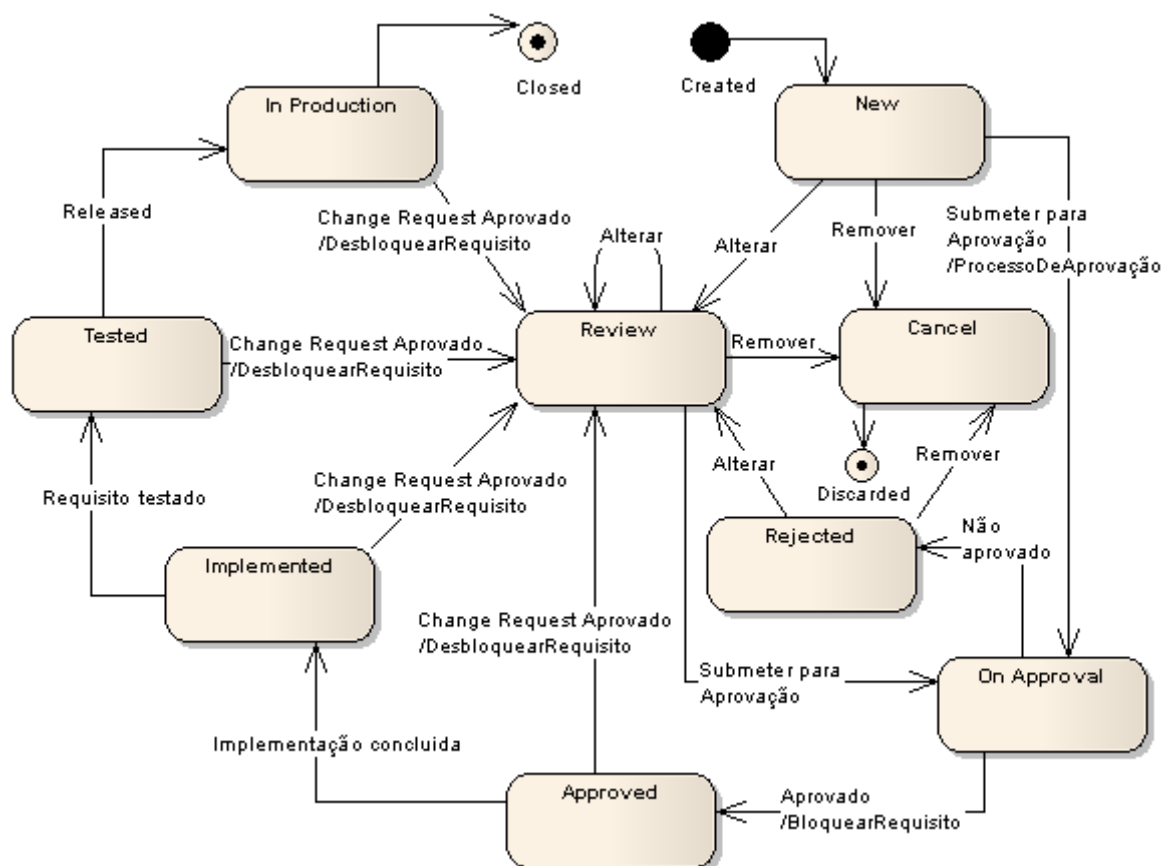


Figura 33: Diagrama de Estados de um requisito

A semântica intrínseca à máquina de estados apresentada é a seguinte:

- Um requisito ao ser criado passa para o estado “New” que indica que se trata um requisito novo.
- O sistema deve permitir a transição do requisito para um estado “Review” de forma a indicar que foram realizadas alterações ao requisito e que por isso se encontra em revisão ou análise.
- Quando um requisito passa para o estado “On Approval” é iniciado um processo de aprovação do requisito. Após a conclusão deste processo, se o requisito for aprovado transita automaticamente para o estado “Approved”, caso contrário (não aprovado) transita para o estado “Rejected”.
- Os requisitos que são removidos pelos utilizadores devem passar para o estado “Cancel”.
- Após a aprovação de um requisito só é possível a sua reedição se for criado e aprovado um pedido de alteração (Change Request) ao requisito. Com a aprovação de um Change Request o requisito é desbloqueado e deve transitar automaticamente para o estado “Review”.

A utilização desta máquina de estados associada ao requisito permite controlar a edição ordenada dos requisitos e gerir a evolução do requisito durante o desenrolar do projecto. No entanto,

a predefinição de uma máquina de estados associada a um requisito obrigaria a adaptação da organização à terminologia, conceitos e processos inerentes. Assim decidiu-se permitir a personalização dos estados dos requisitos através de um ficheiro próprio de configuração, aumentado desta forma a flexibilidade do sistema.

O ficheiro de configuração criado para este efeito deve permitir a definição de todos os estados que um requisito pode transitar. Este ficheiro possibilita definir o nome dos estados, qual o estado que acciona o processo de aprovação do requisito, quais os estados bloqueantes (restrição de edição do requisito) e ainda quais os estados associados à aprovação, revisão, remoção e rejeição do requisito.

4 Plataforma Tecnológica

Nesta secção são apresentadas as tecnologias escolhidas para suportar o desenvolvimento da solução proposta. É ainda descrito a arquitectura de software escolhida, baseada no modelo de três camadas, adequado para uma implementação eficiente de uma aplicação Web com fortes capacidades de portabilidade, manutenção e reutilização. A escolha das tecnologias e da arquitectura foram apoiadas nos requisitos que a ferramenta deve responder apresentadas nas secções anteriores.

4.1 Introdução

Quando se decide desenvolver qualquer projecto informático, é imperativo procurar as ferramentas mais adequadas de forma a garantir que os requisitos de um sistema são alcançados dentro do prazo estipulado, que permitam um desenvolvimento rápido e iterativo do projecto e possibilitem uma fácil expansão e alteração do sistema e das suas funcionalidades ao longo do seu período de vida.

Além da importância da escolha das ferramentas e das tecnologias adoptadas é também importante definir uma boa arquitectura de software que suporte as características do sistema, que permita um desenvolvimento incremental do projecto, promova a evolução e a reutilização dos diversos componentes do sistema.

A primeira decisão efectuada neste capítulo foi a necessidade de um desenvolvimento de raiz do sistema proposto. A razão desta opção *versus* a customização de uma solução de código aberto, incide essencialmente no facto de a única ferramenta que se enquadrava com as características da solução proposta era a ferramenta RTH disponibilizada pela *SourceForge*. No entanto após a análise da sua documentação e da tecnologia que lhe estava associada conclui-se que não seria uma escolha acertada devido ao seguinte:

- Tempo de aprendizagem necessário elevado para alterar a ferramenta.
- A abordagem da ferramenta era orientada segundo uma visão de uma ferramenta de gestão de testes e pouco focada na discussão e gestão dos requisitos
- A plataforma tecnológica da ferramenta dificultava e restringia a integração com ferramentas externas através de uma abordagem orientada ao serviço.
- Inexistência de documentação de suporte ao programador.

4.2 Arquitectura de Software

A definição da arquitectura de software é uma etapa de extrema importância no processo de desenvolvimento, uma vez que uma boa escolha permite um aumento de flexibilidade e de portabilidade do software a desenvolver.

A arquitectura escolhida foi baseada no modelo de três camadas, uma arquitectura cliente-servidor, de forma a permitir separar fisicamente cada uma das três camadas.

Este tipo de arquitectura tem como principais vantagens permitir o desenvolvimento modular do software, aumentar a reutilização e facilitar a alteração ou substituição de qualquer uma das camadas independentemente da tecnologia.

O modelo é composto pelas seguintes camadas:

- Camada de Apresentação (*Presentation Layer*): É chamada de GUI (*Graphical User Interface*), ou simplesmente interface. Esta camada interage directamente com o utilizador e é através dela que o utilizador consegue interagir com a base de dados, através da comunicação com a camada de negócio.
- Camada de Negócio (*Business Layer*): Faz a ligação entre a camada de apresentação e a camada de persistência. Contém os objectos de negócio e a lógica que controla a funcionalidade da aplicação. Nesta camada são expostos alguns serviços, que encapsulam determinadas funções de negócio, e permitem uma fácil integração com sistemas externos.
- Camada de Persistência (*Data Access Layer*): Camada que funciona como interface entre a aplicação e a base de dados. Esta camada recebe chamadas da camada de negócio e executa os pedidos à base de dados. Através desta camada é possível isolar a lógica da aplicação dos detalhes de uma específica solução de armazenamento.

A Figura 34 ilustra a arquitectura adoptada neste sistema.

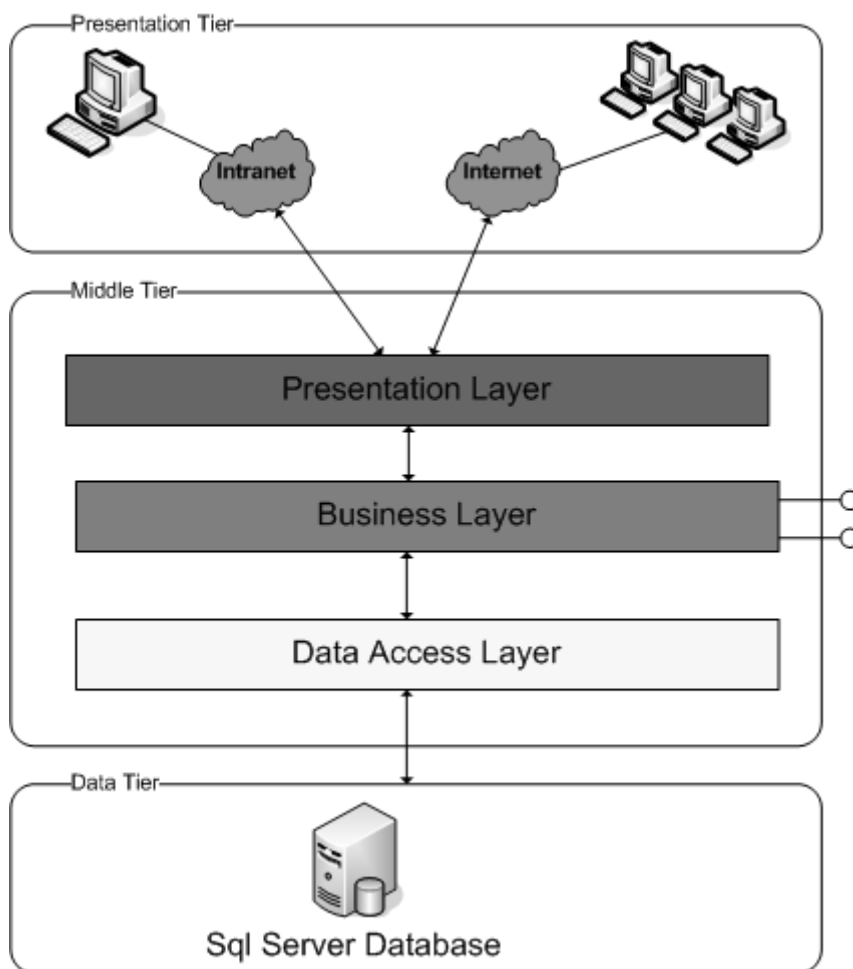


Figura 34: Arquitectura de Software do sistema implementado

4.3 Tecnologia

As tecnologias utilizadas para o desenvolvimento da solução são expostas de seguida, divididas segundo cada umas das camadas apresentadas no subcapítulo anterior. Estas tecnologias foram escolhidas por permitirem suportar todos os requisitos que a solução comporta, com base na experiência prévia e devido à existência de uma vasta documentação sobre cada uma destas, oferecendo desta forma uma grande rapidez de aprendizagem, reduzindo o custo e o tempo de desenvolvimento.

4.3.1 Camada de Apresentação

A tecnologia escolhida para suportar esta camada foi o *ASP.Net* da *Microsoft* suportada pela linguagem de programação *C#*, devido ao reduzido tempo de formação e consequente redução do tempo necessário para o desenvolvimento.

As tecnologias Javascript e XML também foram utilizadas nesta camada através da utilização de uma plataforma AJAX disponibilizada pela *Microsoft* com o nome de *AJAX Toolkit*.

AJAX é um acrónimo que descreve uma técnica para a criação de aplicações Web em que algumas partes da página são recarregadas dinamicamente. Sempre que é necessário actualizar

uma determinada zona, é criado um objecto *XMLHttpRequest*, responsável por estabelecer a comunicação com o servidor em *background*. Em seguida é utilizado o Javascript para tratar o documento XML recebido e actualizar o interface *web* apresentado ao utilizador (24). A figura seguinte apresenta um modelo genérico das aplicações web que utilizam a abordagem AJAX.

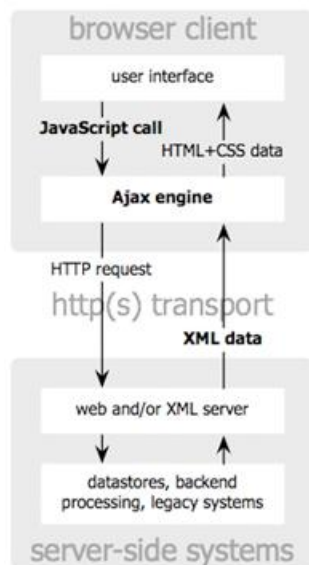


Figura 35: Modelo Ajax de aplicações web (25).

A razão da escolha desta técnica para o desenvolvimento do interface da aplicação web em detrimento das técnicas tradicionais deve-se às diversas vantagens que esta técnica proporciona. Estas vantagens prendem-se com a redução da banda larga utilizada para o *refresh* de conteúdos da página, um interface mais intuitivo e de resposta mais rápida a *inputs* do utilizador, o que torna a aplicação web mais interactiva com o utilizador, e ainda redução das conexões ao servidor. No entanto, existem algumas limitações e desvantagens baseadas na aplicação deste modelo, o que obriga uma utilização moderada desta técnica de forma a não reduzir a usabilidade da aplicação.

4.3.2 Camada de Negócio

Nesta camada foi adoptada a linguagem de programação C# devido à simplicidade de ligação entre a camada de apresentação, desenvolvida sobre a mesma tecnologia, e utilizando o mesmo pacote de desenvolvimento de software, o *Visual Studio 2008*.

Para expor os serviços desta camada intermédia foi utilizado a tecnologia *Web Services* que permite a integração da aplicação com aplicações externas independentemente da tecnologia dos sistemas.

4.3.3 Camada de Persistência

A tecnologia escolhida para suportar o armazenamento de dados foi o *Microsoft SQL Server*, as razões para esta escolha foi o baixo custo de manutenção, o facto de ser um tecnologia adoptada

por grande parte das organizações, independentemente da sua dimensão, e ainda por esta tecnologia estar disponível nas instalações da Noesis, desde o início do trabalho o que reforçou a escolha por esta solução.

Para o armazenamento de configurações do sistema foi utilizado a tecnologia XML, devido às diversas vantagens que esta tecnologia contempla, entre elas a portabilidade, facilidade de leitura (por máquinas e humanos) e a simplicidade de validação a partir de um XSD.

Por fim, o acesso ao repositório de dados foi efectuado a partir da tecnologia ADO.NET incluída na *Microsoft. Net Framework*, dado ser a opção mais indicada para um fácil e fiável aceder aos dados armazenados na base de dados.

Um estudo mais aprofundado das tecnologias que poderiam ser usadas nesta arquitectura está fora do âmbito desta dissertação de mestrado.

4.4 Conclusão

A plataforma tecnológica apresentada não pretende ser a escolha ideal para a generalidade de todas as aplicações web e equipas de desenvolvimento, mas antes a que melhor se adequa a este projecto específico dado os seus requisitos funcionais e restrições de tempo e tecnologias disponíveis.

5 Testes Realizados

Nesta secção é apresentado a forma como foi testada a aplicação ao longo do processo de desenvolvimento.

5.1 Introdução

O objectivo da realização de testes de uma aplicação é avaliar a adequada correcção e funcionamento de todos os componentes do sistema, principalmente os executáveis. Esta verificação consiste na confirmação que a codificação do sistema está conforme a especificação técnica produzida na tarefa de desenho, que por sua vez resulta dos requisitos especificados na fase de análise.

5.2 Metodologia adoptada

A execução de testes ao sistema desenvolvido foi realizado de forma gradual e de forma iterativa, no entanto, na fase final do desenvolvimento foram intensificados de forma a garantir o correcto funcionamento da aplicação em função dos requisitos e modelos especificados.

Foram realizados os seguintes tipos de testes:

- Testes unitários;
- Testes de integração;
- Testes de sistema;
- Testes de aceitação;

Os testes realizados à aplicação foram suportados por uma ferramenta interna da organização Noesis, baseada na ferramenta *open-source* *TestLink*. Esta ferramenta de testes permite criar e gerir casos de testes e ainda organizá-los em planos de teste. Esta ferramenta facilita por isso uma organização e gestão mais eficiente dos testes a realizados à solução desenvolvida.

6 Resultados Obtidos

Nesta secção é apresentado o resultado final da aplicação desenvolvida segundo os requisitos e modelos apresentados nas secções anteriores. Resta então validar se a análise realizada, que produziu esta aplicação, corresponde à realidade e se esta ferramenta permite de facto gerir de uma forma mais eficiente os requisitos de um projecto informático.

Para que seja possível verificar e validar que a solução desenvolvida responde aos problemas atrás identificados, seria necessário realizar um estudo a longo prazo onde a ferramenta daria o suporte necessário para a gestão dos requisitos de vários projectos informáticos. Infelizmente não surgiu nenhuma oportunidade no curto espaço de tempo disponível, para usar a ferramenta quer num cliente da Noesis, quer num projecto interno, o que impossibilitou a realização de uma avaliação mais exaustiva do desempenho da aplicação e o seu relacionamento com a melhoria de qualidade dos projectos por esta suportada.

Devido a esta restrição decidiu-se avaliar a ferramenta a partir da recolha das opiniões de utilizadores alvo. Esta recolha de opiniões foi realizada através dos seguintes métodos:

- Demonstração das funcionalidades da ferramenta a utilizadores;
- Experimentação da ferramenta por parte dos utilizadores;
- Análise das respostas a um questionário de forma a recolher o *feedback* e o nível de aceitação da ferramenta.

6.1 Apresentação da Solução Final

De seguida são apresentados *screenshots* da aplicação de forma a mostrar que a aplicação cumpre os requisitos especificados, e apresenta um interface semelhante ao protótipo de alta-fidelidade desenvolvido. É também realizada uma breve descrição da ilustração apresentada, das funcionalidades que lhe estão associadas e consequentes tarefas que suportam.

6.1.1 Resumo do Projecto e sistema de aprovações

Como é possível visualizar na Figura 36 a página inicial da aplicação apresenta diversas informações relevantes ao projecto. Após a selecção de um projecto, é apresentado a descrição geral do projecto em curso, a situação global dos requisitos relativamente ao estado em que se encontram, um painel onde é possível efectuar a votação relativamente à aprovação ou reprovação de requisitos e/ou pedidos de alteração em curso e ainda consultar quais os requisitos e pedidos de alteração aprovados e reprovados.

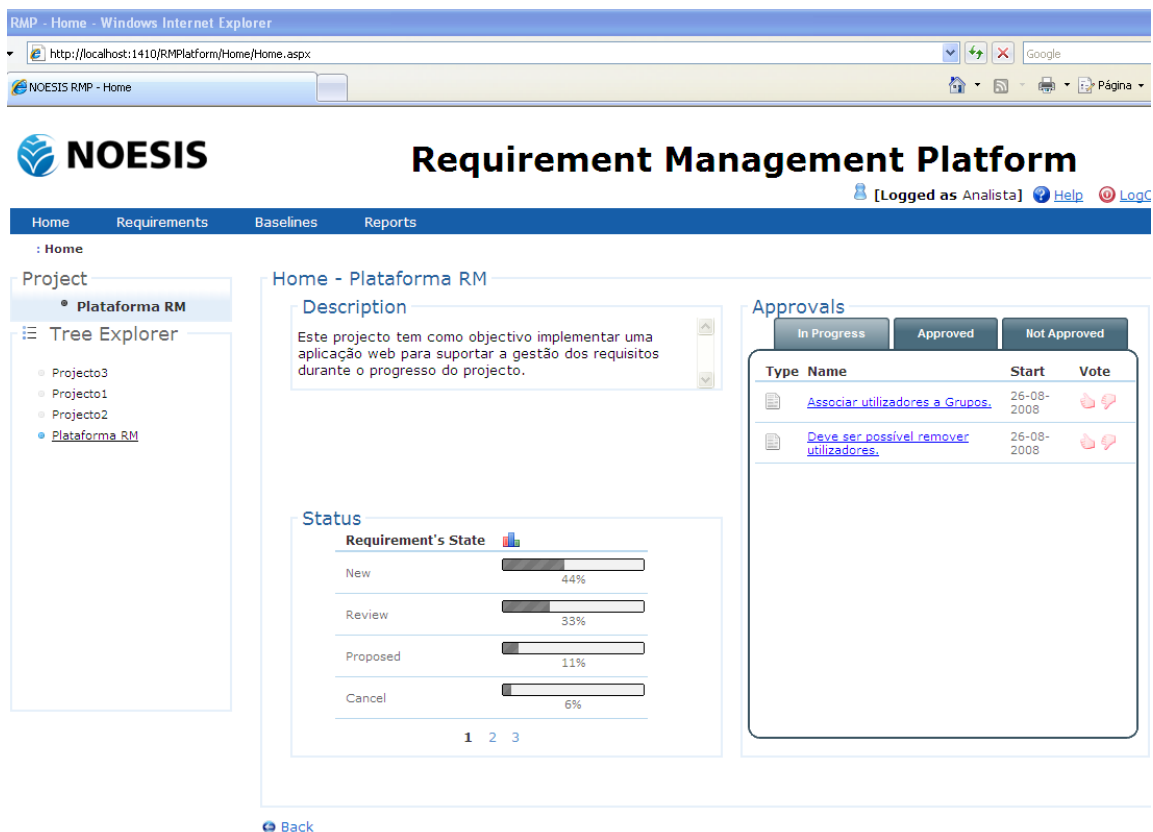


Figura 36: Página inicial da aplicação.

6.1.2 Matriz de Rastreabilidade

A ilustração seguinte apresenta a matriz de rastreabilidade desenvolvida. Esta permite não só consultar de uma forma global as dependências entre os requisitos, mas também adicionar, remover e limpar directamente na matriz ligações entre requisitos.

A matriz desenvolvida permite ainda a aplicação de filtros oferecendo desta forma diversas vistas sobre as ligações dos requisitos. Esta funcionalidade adiciona uma flexibilidade extra à matriz permitindo que o utilizador se foque unicamente nos requisitos que pretende gerir e analisar.

Com base neste ecrã, diversas actividades típicas da gestão de requisitos são suportadas, nomeadamente a análise de impacto de alterações a requisitos do projecto, manutenção da rastreabilidade entre os requisitos, limpeza de ligações sujas (ou seja validar ligações que o sistema automaticamente notifica de duvidosas após a alteração de um requisito) e ainda a visão global dos requisitos do projecto e as suas inter-dependências.

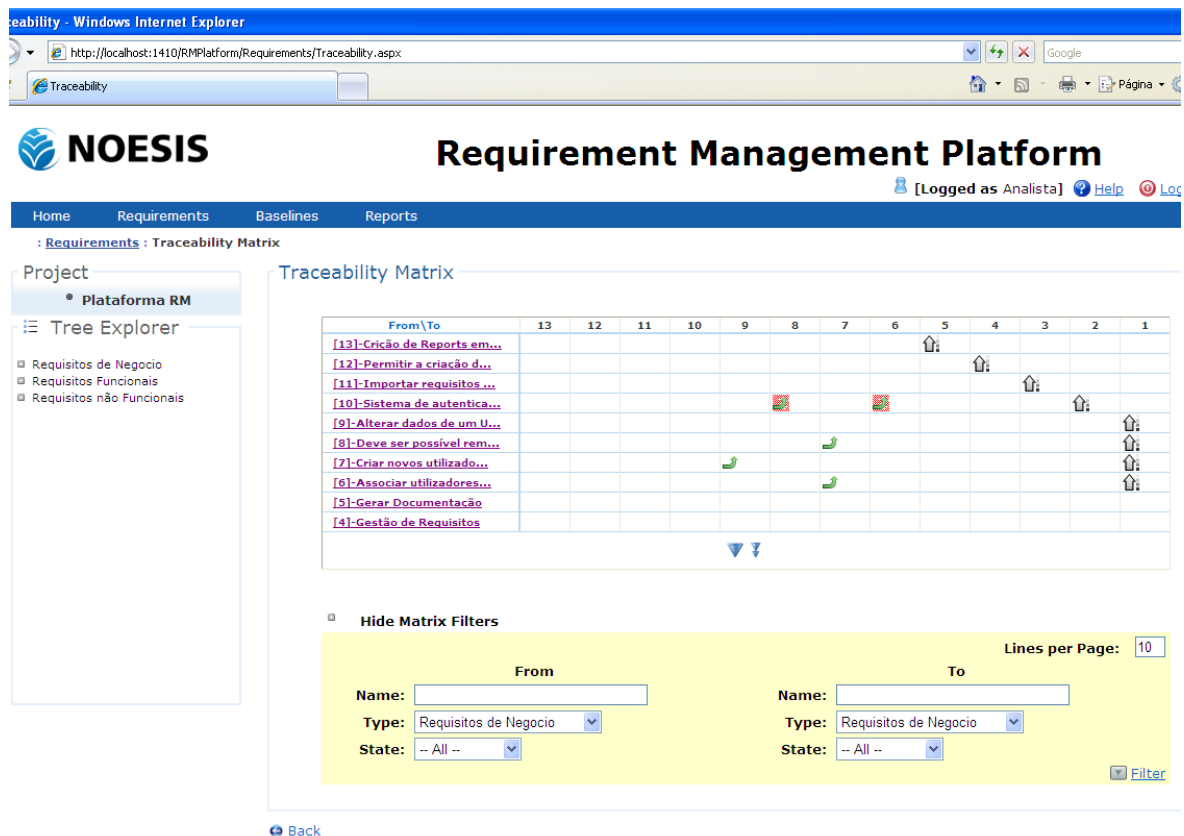


Figura 37: Matriz de rastreabilidade do sistema desenvolvido.

6.1.3 Pedido de alteração a requisitos

Após a aprovação de um requisito este é bloqueado e deixa de ser possível a sua reedição. No entanto, em grande parte dos casos com o decorrer do projecto é imprescindível efectuar alterações aos requisitos. Com base nesta realidade, o sistema obriga a criação de um pedido formal de alteração de um ou mais requisitos, o qual é analisado posteriormente por um gestor de alterações e submetido a aprovação dos utilizadores.

A Figura 38 apresenta o formulário onde o utilizador que efectua o pedido deve preencher de forma a iniciar o processo de pedido de alteração de requisitos. O utilizador deve seleccionar os requisitos que pretende incluir no pedido e especificar a razão e a descrição detalhada da alteração a realizar.

Após a criação do pedido é criado automaticamente uma votação que irá determinar a aprovação ou não do pedido. Os utilizadores podem consultar os pedidos de alterações em curso e terminados, os quais incluem além do pedido realizado pelo utilizador, também a análise de impacto realizada pelo gestor de alterações do projecto.

Change Request - Windows Internet Explorer

http://localhost:1410/RMPlatform/Requirements/ChangeRequest.aspx

Change Request

NOESIS Requirement Management Platform

[Logged as Analista] Help LogOut

Home Requirements Baselines Reports

: Requirements : Change Request

Project

- Plataforma RM
- Tree Explorer
- Requisitos de Negocio
 - 5 - Gerar Documentação
 - 4 - Gestão de Requisitos
 - 3 - Gestão de Projectos
 - 2 - Controlo de Acessos
 - 1 - Gestão de Utilizadores
 - Requisitos Funcionais
 - Requisitos não Funcionais

Change Request

Request for Change Analysis Impact

Created by: Zé Manuel

Name: Alteração do Sistema de Autenticação

Reason: O Sistema obriga a autenticação de todos os utilizadores, mesmo quando acedida a partir da intranet da organização.

Descrição: O Sistema deveria permitir a validação de utilizadores através do Active Directory.

Requirements: [Done](#)

Number	Version	Nome	Remove
10	3	Sistema de autenticação.	Remove

Cancel Create

Back

Figura 38: Formulário de pedido de alterações a um ou mais requisitos.

6.1.4 Comparação de *Baselines*

Uma tarefa comum realizada na gestão de requisitos é a comparação dos requisitos ao longo do tempo. De forma a suportar esta tarefa o sistema inclui as funcionalidades de criação e comparação de *baselines*.

A figura seguinte apresenta um painel que mostra as alterações realizadas nos requisitos entre duas *baselines* criadas. Após a selecção das duas *baselines* a comparar é automaticamente apresentado ao utilizador as alterações que ocorreram entre as duas imagens do projecto.

O sistema permite ainda inverter a comparação entre as *baselines* e definir o número de elementos por página. Ainda a sublinhar a utilização de cores que facilitam a análise do utilizador em relação às alterações que ocorreram nos requisitos (verde: novo requisito; cinza: requisito sem alterações; laranja: requisito modificado).

net Explorer

orm/Baselines/BaselineComparator.aspx

Google

Página Ferram

Requirement Management Platform

[Logged as Analista] Help LogOut

Baselines Reports

Baseline Comparator

Baseline Intermédia			Baseline Inicial
N.º	Version	Name	Result
3	1	Gestão de Projectos	No Changes
7	1	Criar novos utilizadores do sistema	No Changes
13	4	Criação de Reports em diversos formatos	Changed: [Version > 1] To [Version > 4]
14	1	Gestão de Utilizadores	No Changes
8	1	Deve ser possível remover utilizadores.	No Changes
20	2	Permitir a alteração dos projectos	New
12	1	Permitir a criação de novos requisitos	No Changes
18	1	Segurança	No Changes
6	2	Associar utilizadores a Grupos.	No Changes
17	1	Usabilidade	No Changes

1 2

Items per page: 10

[Change order](#)

[Back](#)

Figura 39: Página do sistema que apresenta a comparação entre duas *Baselines*.

6.1.5 Definição e especificação do Requisito

A Figura 40 apresenta a informação relativa a um requisito. Nesta página é possível navegar pelos requisitos do projecto através de uma árvore criada para esse efeito. Após a selecção do requisito pretendido é apresentado ao utilizador várias informações relativas ao mesmo, o que permite geri-lo com eficácia e simplicidade.

A árvore apresentada no lado esquerdo da ilustração reflecte o tipo de cada requisito em particular e apresenta ainda a hierarquia dos requisitos. A representação utilizada permite navegar pelos requisitos de uma forma organizada, contextualizada e simples.

Após a selecção de um requisito é apresentado ao utilizador diversas informações relevantes do requisito. O tipo de informação apresentada é separada por seis separadores, que permitem organizar de uma forma adequada toda a informação relativa ao requisito.

De seguida é apresentado em pormenor cada separador, a informação e as funcionalidades que cada um disponibiliza:

- Separador “Details”: Este separador apresenta a informação principal do requisito, nomeadamente o utilizador que criou o requisito, o identificador único do requisito e a sua versão, o seu nome, descrição e estado associado.
- Separador “Custom Attributes”: Nesta zona pode ser consultada e alterada a informação dos atributos que foram especificados no painel de administração.
- Separador “Files”: Este separador apresenta uma lista de documentos e ficheiros que estão associados ao requisito. É possível ainda adicionar novos ficheiros, remover e consultar os ficheiros listados.
- Separador “History”: Aqui é listado o histórico de versões do requisito, o estado que lhe estava associado e é ainda possível consultar essas mesmas versões.
- Separador “Discussion”: Este separador permite que qualquer utilizador participe na discussão do requisito. Para isso é possível adicionar novos comentários e responder a comentários realizados por outros utilizadores.

The screenshot shows the NOESIS Requirement Management Platform interface. The browser window title is 'Requirements - Windows Internet Explorer'. The address bar shows 'http://localhost:1410/RMPlatform/Requirements/Default.aspx'. The page has a blue header with the NOESIS logo and the title 'Requirement Management Platform'. Below the header is a navigation bar with links: Home, Requirements, Baselines, Reports. The main content area is titled 'Requirements' and shows a requirement with the title 'Deve ser possível remover utilizadores.' The requirement is displayed in a modal window with tabs: Details, Custom Attributes, Files, Traceability, History, Discussion. The 'Details' tab is active, showing the following information:

- Nº.: 8
- Version: 1
- Updated on: 26-08-2008 by ApprovalProcess
- Created on: 26-08-2008 by Analista
- Name: Deve ser possível remover utilizadores.
- Description: O Administrador poder remover os utilizadores do sistema, evitando desta forma a apresentação de utilizadores que já não se encontram a realizar projectos.
- Type: Requisitos de Negocio
- State: Approved

 At the bottom of the modal window are buttons: Cancel, RFC, and New Child. A 'Back' link is also visible below the modal window.

Figura 40: Painel de consulta e alteração de um requisito.

6.2 Opiniões dos Utilizadores

A forma encontrada para validar a aplicação desenvolvida nesta tese foi a recolha do *feedback* de utilizadores alvo após a experimentação do sistema produzido. Esta recolha de opiniões foi realizada através do *feedback* pessoal dos utilizadores do sistema e ainda através das respostas dadas por estes ao questionário que se encontra disponível para consulta no anexo A.2.

Participaram neste estudo quinze utilizadores dos quais, dois deles são consultores seniores na área de qualidade (QM), três alunos recém-licenciados em Engenharia Informática e Computadores, e os restantes alunos frequentam actualmente o Mestrado ou Licenciatura no curso de Informática e Computadores no Instituto Superior Técnico.

De uma forma geral os resultados obtidos e opiniões recolhidas foram bastante positivas. Os principais pontos a destacar foi a boa classificação por parte dos utilizadores relativamente ao tempo de aprendizagem e à usabilidade da ferramenta. A ferramenta obteve também alguns elogios relativamente à abordagem tomada para controlar e gerir as alterações aos requisitos. Em suma a ferramenta foi classificada a nível global com um 9 (de uma escala de 1 a 10) pelos utilizadores.

Os utilizadores com maior experiência na área de qualidade de software permitiram adquirir um melhor feedback relativamente à lógica da ferramenta, na medida em que esta permite auxiliar a resolução de grande parte dos problemas da gestão de requisitos. Estes destacaram o interface simples e de fácil aprendizagem, o modelo de aprovações do sistema, o suporte à análise de impacto de alterações e a elevada flexibilidade da ferramenta, nomeadamente a possibilidade de definir os estados para os quais os requisitos podem transitar.

Como aspectos negativos foram apontados alguns reparos acerca da acessibilidade da ferramenta em *browsers* mais antigos e menos utilizados pelo universo de utilizadores. Alguns utilizadores com maior experiência e sensibilidade nesta problemática consideram que a abordagem ao problema relativamente ao sistema de aprovações de requisitos é muito boa, mas no entanto, levantaram algumas preocupações relativamente à adopção deste mecanismo para a aprovação de requisitos, já que em grande parte dos casos essa aprovação formal não acontece.

6.3 Comparação com outras ferramentas analisadas

Tendo como base a análise anterior efectuada a algumas das ferramentas existentes no mercado e a solução implementada para responder à problemática da gestão dos requisitos, optou-se por comparar as ferramentas para entre si.

Funcionalidades	Caliber RM	Req. Pro	OSRT	Doors	RTH	RM
Explorador em árvore	A	C	D	C	D	A
Nível de Usabilidade	B	D	D	C	D	A
Facilidade de aprendizagem	C	D	D	D	C	A
Fórum de discussão	C	C	I	I	D	B
Suporte de hierarquia	A	B	C	C	C	B
Rastreabilidade entre requisitos	A	A	B	B	I	B
Cálculo de Impacto de alterações	C	C	I	C	I	A
Importação de requisitos	C	C	I	C	I	I
Associação requisito-documento	C	D	D	I	D	B
Personalização de atributos	A	A	C	C	I	B
Personalização de tipos	A	C	C	I	C	A
Gestão de <i>Baselines</i>	C	C	D	I	I	B
Guardar o histórico dos requisitos	B	B	C	B	C	B
Notificação de alterações	B	C	I	C	C	D
Suporte a integrações externas	C	C	I	I	I	B
Gerir utilizadores e projectos	C	C	D	C	D	B
Geração relatórios automáticos	B	C	C	C	I	C
Apreciação global	A	B	C	B	D	A

Legenda: A - Muito Bom; B - Bom; C - Razoável; D - Fraco; I - Inexistente;

Tabela 10: Comparação funcional da solução desenvolvida com outras ferramentas do mercado

É importante mencionar que a classificação apresentada deriva da análise realizada segundo o contexto e o problema particular apresentado no capítulo 1 deste documento.

7 Conclusão

Após um ano de investigação na área de gestão de requisitos, é possível concluir que actualmente já existe alguma preocupação das organizações sobre a problemática da gestão de requisitos. Devido à necessidade de responder a alterações dos requisitos, durante o decorrer do projecto, surgiram diversas aplicações que oferecem várias funcionalidades que permitem facilitar algumas das tarefas da gestão dos requisitos. No entanto, é preciso sublinhar que a utilização de uma ferramenta para a gestão de requisitos, não substitui a implementação de um processo de engenharia de requisitos bem definido.

A solução desenvolvida partiu de uma necessidade específica da Noesis SA que é partilhada por grande parte das pequenas médias empresa. Este tipo de organizações tem a necessidade de gerir os requisitos de projectos informáticos através de uma solução de baixo custo, de elevada flexibilidade e personalização de acordo com o projecto em desenvolvimento. Necessitam ainda de uma ferramenta com um elevado nível de usabilidade e de fácil aprendizagem, que fosse mais um veículo facilitador de comunicação entre os diversos intervenientes no projecto, de fácil integração com outras ferramentas CASE e que ofereça características que permitam um fácil acesso á aplicação independentemente da localização dos seus utilizadores.

Com base neste problema foi realizado um estudo aprofundado na temática de engenharia de requisitos. Este incluiu o estudo do processo genérico de engenharia de requisitos, a análise da importância da gestão dos requisitos durante o desenrolar do projecto, o estudo da importância da gestão de alterações (change managment) e rastreabilidade de requisitos e uma análise das principais tarefas que devem ser suportadas por este tipo de ferramentas. Por fim foi ainda realizada uma análise das principais ferramentas comerciais e *open-source*. Todo este estudo foi crucial para perceber a problemática associada a este tipo de ferramentas.

Através do estudo realizado foi produzida uma ferramenta que não contendo todas as funcionalidades quando comparada às ferramentas concorrentes a nível comercial, se distingue pelo seguinte:

- Oferece aos seus utilizadores as principais tarefas de gestão de requisitos de uma forma simples e de fácil aprendizagem;
- Disponibiliza um sistema de aprovações de requisitos que permite aprovar cada requisito em particular;
- Apresenta um novo meio de comunicação que permite a discussão dos requisitos, permitindo melhorar a qualidade dos mesmos;
- Preocupa-se com a gestão da mudança no contexto dos requisitos, oferecendo um método de aprovação dos requisitos aprovados que necessitam de ser alterados;
- Suporta a análise de impacto de requisitos através de uma matriz de rastreabilidade;

- Desenvolvida segundo o paradigma SOA, potencializando uma rápida e fácil integração com outras ferramentas CASE;

Em suma a ferramenta que resultou do estudo e investigação na área de engenharia de requisitos revelou-se ser uma ferramenta adequada para auxiliar a gestão dos requisitos ao longo do tempo dado os resultados obtidos pelo *feedback* de utilizadores alvo. Esta ferramenta disponibiliza algumas das tarefas essenciais para a gestão dos requisitos, o que é por si uma mais-valia para qualquer organização que se esteja a iniciar nas tarefas de gestão de requisitos.

7.1 Trabalho Futuro

Como trabalho futuro de investigação nesta área, sugere-se a continuação de implementação de algumas funcionalidades que foram deixadas de parte, nomeadamente um módulo que permita integrar a ferramenta com ferramentas de teste, permitir a associação de testes a requisitos, ainda permitir que a ferramenta produza um maior número de relatórios.

Para ficar definitivamente provada a contribuição desta ferramenta como forma de resolução dos problemas identificados, deveriam ser efectuados casos de estudo, em pequenas e médias empresas onde decorressem projectos de curto ou médio prazo a fim de analisar e comprovar a vantagens da ferramenta num ambiente real.

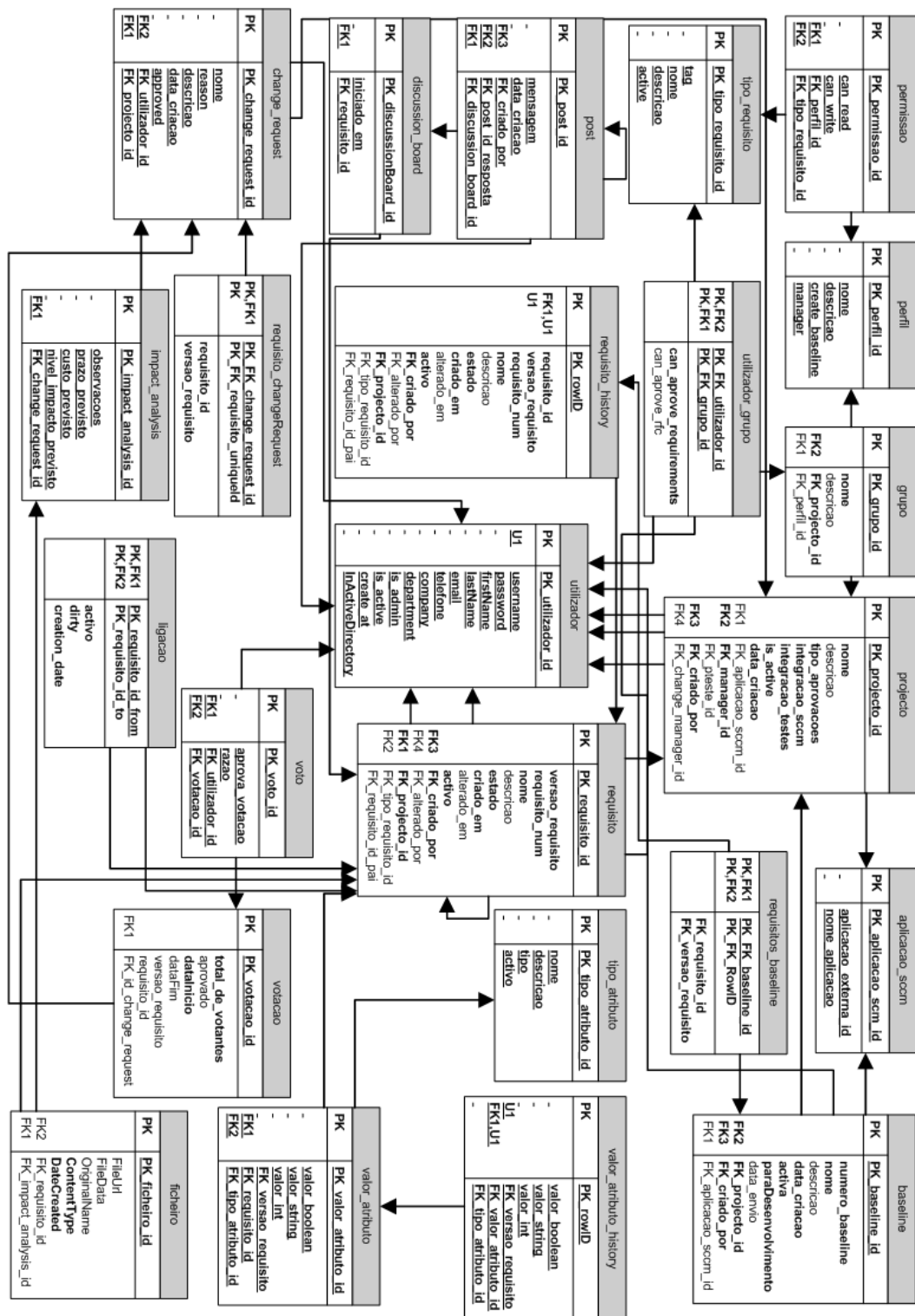
8 Referências

1. *Requirements Engineering: A Roadmap*. **Easterbrook, Steve e Nuseibeh, Bashar**. s.l. : ACM Press, 2000. International Conference on Software Engineering. pp. 35-46.
2. **Sommerville, Ian e Kotonya, Gerald**. *Requirements Engineering: Process and Techniques*. s.l. : Wiley, 1998.
3. *Integrated Requirements Engineering: A Tutorial*. **Sommerville, Ian**. 1, s.l. : IEEE Computer Society Press, January de 2005, Journal: IEEE Software, Vol. 22, pp. 16-23.
4. **Elizabeth, Hull, Kenneth, Jackson e Jeremy, Dick**. *Requirements Engineering*. 2nd Edition. s.l. : Springer, 2005.
5. *Requirements Lifecycle Management and Release Planning in Market-Driven Requirements Engineering Processes and Tools to support the RE Process*. **Carlshamre, Pär e Regnell, Björn**. Greenwich UK : IEEE CS Press, 2000. International Workshop on the Requirements Engineering Process: Innovative Techniques, Models,.
6. **Turbit, Neville**. *Requirements Traceability*. [White Paper] s.l. : The PROJECT PERFECT White Paper Collection, 2005.
7. **Ramesh, Balasubramaniam e Jarke, Matthias**. Toward Reference Models for Requirements. *Journal:IEEE Transactions on Software Engineering*. 2001, Vol. 27, 1, pp. 58-93.
8. **Young, Ralph R**. *The Requirements Engineering Handbook*. Boston : Artech House Publishers, 2004.
9. **Software Engineering Institute**. Computer-Aided Software Engineering (CASE) Environments. *Software Engineering Institute Web Site*. [Online] 2008.
http://www.sei.cmu.edu/legacy/case/case_what.html.
10. **Borland**. Effective Requirements Definition and Management. *A Borland White Paper*. s.l. : A Borland White Paper, 2006.
11. **Wieggers, Karl E**. Automating Requirements Management. *Process Impact*. 1999.
12. —. *Software Requirements : Practical techniques for gathering and managing requirements throughout the product development*. 2nd Edition. s.l. : Microsoft Press, 2003.
13. **The Atlantic Systems Guild**. Volere Requirements Resources. [Online] 1995-2007.
<http://www.volere.co.uk/>.
14. **SourceForge.net**. SourceForge.net: Open Source Requirements Management Tool. *SourceForge Web Site*. [Online] 2006-2007. <http://sourceforge.net/projects/osrmt>.
15. **SourceForge**. SourceForge.net: RTH. *SourceForge Web Site*. [Online] 2005-2008.
<https://sourceforge.net/projects/rth/>.
16. **IBM**. IBM - Requirements Management - Rational RequisitePro - Software. [Online] IBM, 2008.
<http://www-306.ibm.com/software/awdtools/reqpro/>.
17. **International Council on Systems Engineering**. INCOSE Requirements Management Tools Survey. [Online] 1996-2008. <http://www.paper-review.com/tools/rms/read.php>.
18. **Borland**. Borland® CaliberRM™. *Software Requirements Engineering from Borland*. [Online] 1994-2008. <http://www.borland.com/us/products/caliber/rm.html>.

19. **Telelogic.** Requirements Management for software development – DOORS Fastrack – Telelogic. *Telelogic Web Site*. [Online] 2008. <http://www.telelogic.com/products/doors/doors/index.cfm>.
20. **Silva, Alberto e Videira, Carlos.** *UML, Metodologias e Ferramentas CASE*. 2nd Edition. s.l. : Editora Centro Atlântico, 2005. Vol. I.
21. —. *UML, Metodologias e Ferramentas CASE*. 2nd Edition. s.l. : Editora Centro Atlântico, 2008. Vol. II.
22. **Cockburn, Alistair.** *Writing Effective Use Cases*. s.l. : Addison-Wesley, 2000.
23. **Nielsen, Jakob.** Heuristics for User Interface Design. *useit.com:Jakob Nielsen on Web Design*. [Online] 1995-2008. http://www.useit.com/papers/heuristic/heuristic_list.html.
24. **Abreu, Luis.** *Ajax com ASP.NET Curso Completo*. Lisboa : FCA - Editora de Informatica, Lda, 2007.
25. **Garrett, Jesse James.** Ajax: A New Approach to Web Applications. *Adaptive Path*. [Online] 18 de February de 2005. <http://adaptivepath.com/ideas/essays/archives/000385.php>.
26. **Robertson, Suzanne e Robertson, James.** *Mastering the Requirements Process*. 2nd Edition. s.l. : Addison-Wesley Professional, 2006.
27. **McConnel, Steve.** *Rapid Development: Taming Wild Software Schedules*. s.l. : Microsoft Press, 1996.
28. **Leffingwell, Dean e Widrig, Don.** *Managing Software Requirements: A Use Case Approach*. 2nd Edition. s.l. : Addison-Wesley Professional, 2003.
29. **Jones, David A., et al.** *Interfacing Requirements Management Tools In The Requirements Management Process -A First Look*. Requirements Working Group of the International Council on Systems Engineering. 1997. White Paper.
30. **Ludwig Consulting Services, LLC.** *Managing Requirements*. [Online] 2007. <http://www.jiludwig.com/>.
31. **IEEE Computer Society.** *Guide to the Software Engineering - Body of Knowledge*. [ed.] Alain Abran e Pierre Bourque. s.l. : Angela Burgess, 2004.
32. **Software Engineering Institute.** *CMMI® Web Site*. [Online] <http://www.sei.cmu.edu/cmmi/>.
33. **Software Engineering Institute (SEI).** *CMMI for Development Version 1.2* . 2006.

9 Anexos

9.1 A.1: Modelo da Base de Dados



9.2 A.2: Questionário Realizado

Com base na utilização da ferramenta Requirement Management Platform responda com sinceridade às seguintes questões.

1. Classifique a facilidade de execução das tarefas que lhe foram propostas.
 - a) Difícil
 - b) Razoável
 - c) Fácil

- 2) Considera que o sistema de aprovações da ferramenta é um bom método para validar os requisitos?
 - a) Sim
 - b) Não

- 3) Classifique de 1 a 10 o tempo necessário na aprendizagem da aplicação.

- 4) Considera importante a gestão de alterações dos requisitos durante todo o ciclo de vida do processo de desenvolvimento?
 - a) Sim
 - b) Não

- 5) Se respondeu Sim na resposta anterior classifique a solução proposta pela ferramenta para suportar os pedidos de alteração aos requisitos.
 - a) Fraca
 - b) Razoável
 - c) Boa
 - d) Muito Boa

- 6) Na sua opinião como classifica o nível de usabilidade da ferramenta?
 - a) Fraco
 - b) Razoável
 - c) Bom
 - d) Muito Bom

- 7) Utilizaria esta ferramenta para suportar a gestão e definição dos requisitos de um projecto informático
- a) Sim
 - b) Não

- 8) Qual a sua classificação global (de 1 a 10) relativamente à ferramenta que experimentou?
- _____

- 9) Considera que a ferramenta desenvolvida responde aos problemas apresentados a ferramentas deste tipo?
- a) Sim
 - b) Não

- 10) Na sua opinião:

10.1) Quais os pontos fortes da aplicação?

10.2) O que acha que deve ser melhorado?

Obrigado pelo tempo disponibilizado.