

Documento de Requisitos do Sistema
Methodology Explorer
Versão 3.0

Histórico de Alterações

Data	Versão	Descrição	Autor
17/11/2002	3.0	Fechamento do escopo com definição de todos os requisitos a serem implementados no TG de Carlos.	Carlos R. S. Júnior
12/08/2002	2.9	Reestruturação do documento. Avaliação dos requisitos anteriores e criação de novos visando adequação às novas funcionalidades desejadas.	Carlos R. S. Júnior
28/09/2001	2.8	Modificação na seção Descrição da interface com o usuário e nos casos de uso do sistema.	Suzana Mesquita
24/09/2001	2.7	Modificação do caso de uso Cadastrar e retirada de vários fluxos secundários	Suzana Mesquita
25/08/2001	2.6	Modificação do caso de uso Exportar	Suzana Mesquita
18/08/2001	2.5	Modificações gerais no documento	Suzana Mesquita
13/08/2001	2.4	Modificação na seção Descrição da interface com o usuário	Suzana Mesquita
23/07/2001	2.3	Modificações gerais no documento	Hermano Perrelli e Suzana Mesquita
14/04/2001	2.2	Modificação da definição da estrutura de árvore no tópico Interface com o usuário.	Suzana Mesquita
13/04/2001	2.1	Modificação do caso de uso RF09: Colar um Componente	Suzana Mesquita

Conteúdo

1. INTRODUÇÃO	4
1.1 VISÃO GERAL DO DOCUMENTO	4
1.2 CONVENÇÕES, TERMOS E ABREVIACÕES	4
1.2.1 Identificação dos requisitos	4
1.2.2 Prioridades dos requisitos	4
2. DESCRIÇÃO GERAL DO SISTEMA	5
2.1 ABRANGÊNCIA E SISTEMAS RELACIONADOS	5
3. REQUISITOS FUNCIONAIS (CASOS DE USO).....	5
3.1 CADASTRO.....	5
[RF001] Criar componente.....	5
[RF002] Excluir componente	5
[RF003] Alterar componente	6
3.2 INTERFACE.....	6
[RF001] Visualizar Componente.....	6
[RF002] Copiar componente.....	6
[RF003] Colar componentes.....	7
3.3 COMPILAÇÃO	7
[RF001] Compilar componente.....	7
3.4 IMPORTAÇÃO/EXPORTAÇÃO	7
[RF001] Anexar documentos.....	7
[RF002] Exportar metodologia.....	8
[RF003] Importar metodologia.....	8
[RF004] Salvar metodologia.....	8
[RF005] Gerar site de metodologia	9
4. REQUISITOS NÃO-FUNCIONAIS	9
[NF001] Usabilidade	9
[NF002] Desempenho	9
[NF003] Hardware e Software.....	9
5. REFERÊNCIAS	10

1. Introdução

Este documento especifica os requisitos do sistema *Methodology Explorer*, fornecendo aos desenvolvedores as informações necessárias para o projeto e implementação, assim como para a realização dos testes e homologação do sistema.

1.1 Visão geral do documento

Além desta seção introdutória, as seções seguintes estão organizadas como descrito abaixo.

1. **Seção 2 – Descrição geral do sistema:** apresenta uma visão geral do sistema, caracterizando qual é o seu escopo e descrevendo seus usuários.
2. **Seção 3 – Requisitos funcionais (casos de uso):** especifica todos os casos de uso do sistema, descrevendo os fluxos de eventos, prioridades, atores, entradas e saídas de cada caso de uso a ser implementado.
3. **Seção 4 – Requisitos não-funcionais:** especifica todos os requisitos não funcionais do sistema, divididos em requisitos de usabilidade, confiabilidade, desempenho, segurança, distribuição, adequação a padrões e requisitos de hardware e software.
4. **Seção 5 – Referências:** apresenta referências para outros documentos utilizados para a confecção deste documento.

1.2 Convenções, termos e abreviações

A correta interpretação deste documento exige o conhecimento de algumas convenções e termos específicos, que são descritos a seguir.

1.2.1 Identificação dos requisitos

Por convenção, a referência a requisitos é feita através do nome da subseção onde eles estão descritos, seguidos do identificador do requisito, de acordo com a especificação a seguir:

[nome da subseção. identificador do requisito]

Por exemplo, o requisito funcional [Recuperação de dados.RF016] deve estar descrito em uma subseção chamada “Recuperação de dados”, em um bloco identificado pelo número [RF016]. Já o requisito não-funcional [Confiabilidade.NF008] deve estar descrito na seção de requisitos não-funcionais de Confiabilidade, em um bloco identificado por [NF008].

Os requisitos devem ser identificados com um identificador único. A numeração inicia com o identificador [RF001] ou [NF001] e prossegue sendo incrementada à medida que forem surgindo novos requisitos.

1.2.2 Prioridades dos requisitos

Para estabelecer a prioridade dos requisitos, nas seções 4 e 5, foram adotadas as denominações “essencial”, “importante” e “desejável”.

5. **Essencial** é o requisito sem o qual o sistema não entra em funcionamento. Requisitos essenciais são requisitos imprescindíveis, que têm que ser implementados impreterivelmente.

6. **Importante** é o requisito sem o qual o sistema entra em funcionamento, mas de forma não satisfatória. Requisitos importantes devem ser implementados, mas, se não forem, o sistema poderá ser implantado e usado mesmo assim.

7. **Desejável** é o requisito que não compromete as funcionalidades básicas do sistema, isto é, o sistema pode funcionar de forma satisfatória sem ele. Requisitos desejáveis podem ser deixados para versões posteriores do sistema, caso não haja tempo hábil para implementá-los na versão que está sendo especificada.

2. Descrição geral do sistema

2.1 Abrangência e sistemas relacionados

O sistema *Methodology Explorer* é uma ferramenta para o processo de desenvolvimento de software. Fornece uma maneira intuitiva e eficiente para definir componentes adequados a uma empresa/projeto. Um componente é uma unidade da metodologia que pode ser manipulada isoladamente, por exemplo artefato, atividade etc.

Utilizando a ferramenta, o usuário - em geral, engenheiro de processos ou projetista de metodologias - poderá cadastrar novos componentes ou criar componentes a partir de outros já existentes. Além disso, poderá alterar, remover e consultar componentes já criados. Tais componentes podem ser exportados da ferramenta, gerando um documento texto, páginas HTML ou um arquivo PDF que podem ser visualizados sem utilizar a ferramenta.

A ferramenta conterá também testes de validação sobre os componentes criados. Estes são baseados no Rational Unified Process [2] (metodologia proposta pela empresa Rational Software Corporation [5]) e servem de ajuda aos usuários, evitando que este cometa pequenos erros.

Diante da facilidade de se definir metodologias, o *Methodology Explorer* contribui de modo decisivo para melhorar a qualidade do processo de desenvolvimento dos projetos de software de uma empresa.

3. Requisitos funcionais (casos de uso)

3.1 Cadastro

[RF001] Criar componente

Descrição do caso de uso: Este caso de uso permite que o usuário crie e armazene um novo componente no sistema.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

Entradas e pré-condições: não tem.

Saídas e pós-condição: um componente é cadastrado no sistema

[RF002] Excluir componente

Descrição do caso de uso: Este caso de uso permite que o usuário exclua um componente do cadastro de componentes do sistema. Um componente pode ser excluído de qualquer instanciamento de metodologia (árvore).

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

Entradas e pré-condições: recebe como entrada o componente que se deseja excluir

Saídas e pós-condição: o usuário consegue excluir o componente que deseja

[RF003] Alterar componente

Descrição do caso de uso: Este caso de uso permite que o usuário altere os dados de um componente.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

Entradas e pré-condições: recebe como entrada o componente que se deseja alterar.

Saídas e pós-condição: um componente é alterado no sistema.

3.2 Interface

[RF001] Visualizar Componente

Descrição do caso de uso: Este caso de uso permite que o usuário visualize os dados de um determinado componente (todos os seus atributos, exceto aqueles que são considerados suas propriedades).

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

Entradas e pré-condições: deve receber como entrada o componente que se deseja visualizar.

Saídas e pós-condição: o usuário visualiza o componente desejado

[RF002] Copiar componente

Descrição do caso de uso: Este caso de uso permite que o usuário copie um componente do cadastro de componentes do sistema. Ou seja, copia o componente de onde ele estava e manda a cópia para a área de transferência.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

Entradas e pré-condições: recebe como entrada o componente que se deseja copiar.

Saídas e pós-condição: o usuário consegue copiar o componente que deseja

[RF003] Colar componentes

Descrição do caso de uso: Este caso de uso permite que o usuário cole o componente armazenado na área de transferência do sistema no local indicado. O conteúdo da área de transferência continua inalterado. Aqui, local refere-se a uma pasta que contém componentes.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

Entradas e pré-condições: recebe como entrada o componente que se deseja colar e tem como pré-condição a necessidade de existência de alguma informação na área de transferência do sistema.

Saídas e pós-condição: o usuário consegue colar o componente no local desejado.

3.3 Compilação

[RF001] Compilar componente

Descrição do caso de uso: Este caso de uso permite que o usuário compile metodologias. Essa compilação permite que as metodologias sejam analisadas e comparadas entre si.

Prioridade: ☐ Essencial ☒ Importante ☐ Desejável

Entradas e pré-condições: deve receber como entrada as metodologias a serem compiladas.

Saídas e pós-condição: os componentes das metodologias são compilados no sistema.

3.4 Importação/Exportação

[RF001] Anexar documentos

Descrição do caso de uso: Este caso de uso permite que anexar documentos gerais a componentes. Por exemplo, anexar o *template* do *Documento de Requisitos* ao fluxo de requisitos.

Prioridade: ☐ Essencial ☒ Importante ☐ Desejável

Entradas e pré-condições: deve receber como entrada o caminho absoluto para um arquivo no sistema de arquivos.

Saídas e pós-condição: O documento é anexado ao componente.

[RF002] Exportar metodologia

Descrição do caso de uso: Este caso de uso permite ao usuário a possibilidade de exportar uma metodologia num determinado formato, como XML, por exemplo. O usuário também tem a opção de escolher se o componente deve ou não ser exportado juntamente com seus anexos.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

Entradas e pré-condições: A entrada é uma metodologia a ser exportado e seus sub-componentes, ou seja, todos os componentes que um determinada metodologia.

Saídas e pós-condição: Os componentes são exportados para um arquivo em um determinado formato (como XML).

[RF003] Importar metodologia

Descrição do caso de uso: Este caso de uso permite que componentes de uma metodologia exportada sejam importados do sistema de arquivos e apresentados no *Methodology Explorer*. Os componentes, para serem importados precisam estar no mesmo formato utilizado no caso de uso [Importação/Exportação.RF002]. Importar um componente apenas permite manipular o componente dentro do Methodology Explorer. Para inseri-lo de fato, é preciso realizar o caso de uso [Importação/Exportação.RF004]

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

Entradas e pré-condições: A entrada é o caminho absoluto para um arquivo no sistema de arquivos.

Saídas e pós-condição: O componente importado será inserido na(s) árvore(s) de componentes adequada.

[RF004] Salvar metodologia

Descrição do caso de uso: Este caso de uso permite salvar as alterações realizadas nos componentes de uma metodologia.

Prioridade: ☐ Essencial ☒ Importante ☐ Desejável

Entradas e pré-condições: A entrada é uma metodologia.

Saídas e pós-condição: um componente é persistido no *Methodology Explorer*.

[RF005] Gerar site de metodologia

Descrição do caso de uso: Este caso de uso permite que um *site* seja gerado para uma metodologia já compilada. O site deve conter também os possíveis artefatos que foram anexados.

Prioridade: ☐ Essencial ☒ Importante ☐ Desejável

Entradas e pré-condições: Um componente metodologia é a entrada para o caso de uso que tem, como pré-condição, que a toda a metodologia já esteja salva.

Saídas e pós-condição: um site completo é gerado no sistema de arquivos contendo os arquivos HTML e os artefatos anexados à metodologia.

4. Requisitos não-funcionais

[NF001] Usabilidade

A interface com o usuário é de vital importância para o sucesso do sistema. Principalmente por ser um sistema que não será utilizado diariamente, o usuário não possui tempo disponível para aprender como utilizar o sistema.

O sistema terá uma interface amigável ao usuário primário sem se tornar cansativa aos usuários mais experientes. Em especial, o módulo de publicação HTML possuirá um wizard para ajudar o usuário.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[NF002] Desempenho

Embora não seja um requisito essencial ao sistema, deve ser considerada por corresponder a um fator de qualidade de software.

Prioridade: ☐ Essencial ☒ Importante ☐ Desejável

[NF003] Hardware e Software

Visando criar um produto com maior extensibilidade, reusabilidade e flexibilidade, deve ser adotado como linguagem principal de desenvolvimento Java seguindo cuidadosamente as técnicas de orientação a objetos. Entretanto, outras linguagens também poderão ser usadas quando indicações técnicas recomendem.

O uso da linguagem Java permite não especificar qual será o sistema operacional e a máquina em que o programa irá executar. No entanto, essa máquina deverá se comunicar com um sistema de banco de dados.

Prioridade: ☐ Essencial ☒ Importante ☐ Desejável

5. Referências

1. Furlan, J. D. **Modelagem de Objetos através da UML**. São Paulo, Makron Books, 1998.
2. Kruchten, P. **The Rational Unified Process – An introduction**. Addison-Wesley, 1998.
3. Página da disciplina Análise e Especificação de Requisitos. www.cin.ufpe.br/~if119.
4. Página da disciplina Metodologia e Desenvolvimento de Software www.cin.ufpe.br/~mds.
5. Página da empresa Rational Software Corporation www.rational.com.
6. Página do projeto de instanciação de ambientes de desenvolvimento de software convencionais e orientados a domínios (visitada em 18/01/2001) www.cos.ufrj.br/~taba.