**Documento de Requisitos do Sistema**

**Sistema de Distriução de Vagas em Escolas Segundo as Preferências de Escolas e Alunos**

**Histórico de Alterações**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versão** | **Descrição** | **Autor** |
| 10/09/2017 | 1.0 | Conversão do Modelo PDF para DOCX e preenchimento de dados básicos | Arnaldo Carneiro |
|  |  |  |  |

**Conteúdo**

[1.Introdução 4](#__RefHeading__544_2138050620)

[1.1Visão geral do documento 4](#__RefHeading__744_364107869)

[1.2Convenções, termos e abreviações 4](#__RefHeading__746_364107869)

[2.1 Abrangência e sistemas relacionados 6](#__RefHeading__748_364107869)

[3.1Cadastro 6](#__RefHeading__750_364107869)

[3.2Interface 7](#__RefHeading__752_364107869)

1. Introdução

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

* 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. Convenções, termos e abreviações

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

* + 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. Prioridades dos requisitos

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

1. **Essencial** é o requisito sem o qual o sistema não entra em funcionamento. Requisitos essenciais são requisitos imprescindíveis, que têm que ser implementados impreterivelmente.
2. **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.
3. **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.
4. Descrição geral do sistema

2.1 Abrangência e sistemas relacionados

O sistema *NOME DO SISTEMA* 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 sistema *NOME DO SISTEMA* contribui de modo decisivo para melhorar a qualidade do processo de desenvolvimento dos projetos de software de uma empresa.

1. Requisitos funcionais (casos de uso)
   1. Cadastro

**[RF002] Excluir componente**

**Nome do caso de uso:** Criar componente

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

**Atores:**

**Regras de negócio:**

**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 instanciação 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**

**Nome do caso de uso:** Alterar componente

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

**Atores:**

**Regras de negócio:**

**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.

* 1. Interface

**[RF001] Visualizar Componente**

**Nome do caso de uso:** 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).

**Atores**:

**Regras de negócio:**

**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

**4. Requisitos não-funcionais**

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 adotar 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.

Nota: Acrescentar neste capítulo as figuras dos casos de uso com a respectiva referência e nome, exemplo:

**[RF001] Criar componente**

