Especificação de Requisitos de Software

## **Introdução**

Este documento registra os requisitos detalhados do Sistema <Nome do Sistema>, na forma de <requisitos textuais ou descrição de casos de uso> do produto.

## **Classes de usuários**

<Na presente seção devem ser descritas as várias classes de usuário relevantes para o sistema. Em um sistema voltado para o apoio a atividades de ensino, por exemplo, classes de usuário relevantes são: aluno, professor e coordenador de curso, dentre outros. Para cada classe de usuário relevante é importante que seja detalhado o tipo de responsabilidade que tem no sistema, eventuais restrições de acesso, bem como características que podem ajudar na operação do sistema, como seu nível educacional, conhecimento técnico, dentre outros.

Se há apenas uma classe de usuário, é importante ressaltar isso nesta seção, descrevendo também as características do usuário típico do sistema. Deve ser notado que as características a descrever dependem muito dos requisitos do sistema. Por exemplo, se o sistema deve apoiar o uso por parte de usuários que tenham alguma deficiência, então essa característica deve ser incluída.>

## **Definição de conceitos**

Nesta seção são descritos os principais conceitos relevantes para o domínio do sistema.

<A descrição dos conceitos do domínio deve ser feita via a criação de um glossário e/ou um modelo conceitual do domínio. O glossário, se existir, deve seguir o seguinte formato:

[**Nome do conceito (em negrito)**] - [descrição do conceito].

Se o modelo conceitual for incluído, deve ser construído como um diagrama de classes conceituais, segundo a notação UML (*Unified Modeling Language*), considerando sua versão mais recente à época da elaboração do documento. Além disso, o diagrama deve ser complementado com uma breve descrição textual de cada classe conceitual.

>

## **Requisitos de Software**

Nesta seção são descritos os <requisitos textuais ou descrição de casos de uso> do produto. Na Seção 4.1 são descritos os requisitos funcionais. Na Seção 4.2 são descritos os requisitos não-funcionais.

## **Requisitos funcionais**

<Se os requisitos forem representados no formato de casos de uso, será necessário complementar a descrição detalhada dos casos de uso com uma ou mais diagramas de caso de uso que representem graficamente os relacionamentos entre atores e casos de uso. Esse(s) diagrama(s) devem ser construídos segundo a notação UML, considerando sua versão mais recente à época da elaboração do documento.

Além disso, a descrição detalhada de cada caso de uso do sistema deve usar como referência o modelo descrito na Tabela 1. Deve ser notado que deverá ser criada uma tabela segundo o modelo citado para cada um dos casos de uso do sistema.>

| Nome do caso de uso | <Nome do caso de uso> |
| --- | --- |
| Identificador | <Identificador único do caso de uso nos artefatos do produto> |
| Pré-condições | <As condições que devem existir antes da execução do caso de uso, e sem as quais seus passos não podem ser executados, devem ser descritas aqui.> |
| Pós-condições | <As condições que devem existir após a execução do caso de uso devem ser descritas aqui.> |
| Fluxo principal | <Os passos do fluxo principal representam os passos de uma execução de sucesso do caso de uso. Devem ser descritos seguindo o esquema:   1. O usuário… 2. O sistema...   Deve ser lembrado sempre que um caso de uso descreve a interação entre um ou mais atores e o sistema. Portanto, não faz sentido que uma descrição detalhada de casos de uso contenha apenas passos do usuário, ou apenas passos do sistema.  Além disso, casos de uso essenciais não contêm descrições relacionadas a detalhes do sistema. Ou seja, não há referências a “botões”, “telas”, “campos”. Há referências a ações do usuário e do sistema, e informações que são trocadas entre eles.  Finalmente, é possível fazer reuso entre casos de uso, via relacionamentos de generalização, de inclusão (*include*) e de extensão (*extension*). A representação gráfica deste tipo de relacionamento deve respeitar as convenções da versão mais atual da UML à época de elaboração do presente documento. Já a representação textual, que deve ser feita na descrição detalhada deve respeitar as seguintes regras:   * Relacionamentos de extensão: incluir a expressão [Ponto de extensão: “Nome do caso de uso extensor” - aplicável até o passo Y], onde Y é o passo a ser retomado do caso de uso estendido; * Relacionamentos de inclusão: incluir a expressão [Via “Nome do caso de uso incluído”.> |
| Fluxos alternativos | <Os fluxos alternativos descrevem ações alternativas aos passos do fluxo principal. Cada fluxo alternativo deve ter um identificador único e deve indicar claramente o passo do fluxo principal a ele relacionado, segundo o esquema definido abaixo. Nesse esquema, X representa o número do passo do fluxo principal em que ocorreu o desvio para o fluxo alternativo em questão.  X - [Identificador único do fluxo alternativo e seu nome]:  X.1 - O usuário…  X.2 - O sistema…  …  X.n Retorna ao passo Y do fluxo principal.  Onde Y é o passo do fluxo principal a ser retomado após a execução do fluxo alternativo em questão.> |

**Tabela 1 - Descrição do Caso de Uso** <Identificador do caso de uso>

<Caso a descrição dos requisitos seja via requisitos textuais, a Tabela 1 deve ser retirada do documento. Adicionalmente, deverá ser adotado o seguinte esquema de representação, para cada requisito a ser descrito:

[Identificador único do requisito] - [Texto do requisito].

Também devem ser respeitadas as regras descritas a seguir. Cada requisito deve:

* Usar a estrutura “o sistema deve…” em alguma parte do texto do requisito, a fim de indicar de maneira objetiva que a responsabilidade pela funcionalidade é do sistema - ou seja, um requisito não deve descrever a capacidade esperada do usuário ou de seu operador;
* Ser necessário, ou seja, descrever uma capacidade, característica, restrição ou qualidade do sistema que, se removida, cria uma situação de deficiência;
* Usar termos objetivos e verificáveis, evitando termos e expressões subjetivos como “fácil”, “rápido”, “frequentemente”, “significante”, “se possível”, dentre outros. Em outras palavras, o requisito deve ser descrito de maneira tal que é possível demonstrar objetivamente que foi implementado no sistema;
* Estar escrito de maneira positiva (“o sistema deve”, em vez de “o sistema não deve”) e usando voz ativa;
* Ser livre de implementação, pois o requisito deve descrever o que o sistema deve fazer, sem dizer como, a menos que haja uma restrição real sobre o design do sistema;
* Ser claro e livre de ambiguidades, ou seja, deve ser possível interpretá-lo apenas de uma forma;
* Consistente, ou seja, sem conflitos com outros requisitos do sistema;
* Único, sem o uso de conjunções;
* Viável, ou seja, há tecnologia conhecida para implementá-lo.>

## **Requisitos não-funcionais**

<Os requisitos não-funcionais devem obedecer as mesmas regras de escrita de requisitos textuais descritos na Seção 4.1.>

## **Rastreabilidade de requisitos**

A rastreabilidade bidirecional entre os requisitos aqui descritos e os demais artefatos do sistema está definida em <inserir link para o documento de rastreabilidade do projeto>. Todos elementos rastreados, incluindo os requisitos, utilizam seus identificadores únicos como referência no documento de rastreabilidade.