Especificação de Design de Software

## **Introdução**

Este documento registra as decisões de design para implementação do Sistema <Nome do Sistema>, assim como o raciocínio por trás de cada uma delas. O registro do design é feito a partir de diferentes visões de design.

<Esta seção deve ser complementada com a descrição de qualquer suposição ou dependência relevante sobre o software ou sobre seu uso. Isso pode incluir suposições sobre software ou hardware relacionado, sistemas operacionais ou navegadores sobre o qual o software deve ser executado, características relevantes dos usuários, dentre outros.>

## **Tecnologias**

<Na presente seção devem ser elencadas e descritas todas tecnologias a serem usadas para o desenvolvimento do produto, desde a determinação das linguagens de programação até a definição de eventuais frameworks, SGBD’s, dentre outros.>

## **Visões do sistema**

Nesta seção o design do sistema é descrito, considerando as visões <listar as visões contempladas. A decisão sobre quais visões usar depende muito do sistema. Alguns exemplos são as visões de implementação (diagrama de pacotes e/ou de componentes), lógica (diagramas de classes e, se necessário, de estados da UML, e diagrama entidade-relacionamento), física (diagrama de implantação da UML) de processos (diagramas de sequência da UML ou diagrama de processos de negócio da notação BPMN) e de casos de uso (descrição expandida de casos de uso e diagrama de casos de uso), que formam o Modelo das Visões Arquiteturais 4+1 de Krutchen. Neste projeto recomenda-se o uso da visão lógica >. Cada diferente visão enfoca um diferente aspecto do sistema, representando-o por meio de modelos específicos.

<Para cada visão escolhida, uma subseção deve ser criada aqui, seguindo o modelo da Subseção 3.1.>

## **<Nome da visão>**

<Aqui deve ser representado o sistema projetado, utilizando o diagrama mais apropriado. A sugestão é que seja utilizado o Modelo das Visões Arquiteturais 4+1 de Krutchen. Mesmo que seja usado um outro conjunto de visões, é essencial que sempre sejam registradas duas informações no mínimo:

* O diagrama em si;
* A justificativa para as decisões tomadas, especialmente se há designs alternativos.

Se for usado o Modelo das Visões Arquiteturais 4+1 de Krutchen, na visão de implementação, é necessário descrever os subsistemas/módulos do software, seus relacionamentos, dependências e interfaces. Se pertinente, complemente com a descrição de como as tecnologias estão associadas a estes diversos elementos de design arquitetural. Pode ser usado um diagrama de pacotes ou um diagrama de componentes da UML (*Unified Modeling Language*).

Com relação à visão lógica do sistema, a ideia é capturar o modelo de objetos do sistema. Para tanto, devem ser usados diagramas de classe da UML e, se necessário, diagramas de estado. Alternativamente, pode ser documentada a visão de dados a persistir na aplicação, por meio de um Diagrama Entidade-Relacionamento.

O design relacionado à visão de processos visa representar os aspectos de concorrência e de sincronização dos componentes do sistema. Deve ser voltado apenas para as principais funções do sistema, representando a lógica da interação entre os componentes do sistema que as implementam. Pode ser usado o diagrama de sequência da UML.

A visão física, por sua vez, pretende representar o mapeamento do software para o hardware que será usado para executá-lo, refletindo a distribuição dos componentes do sistema. Para representar esta visão, pode ser usado um diagrama de implantação da UML.>

## **Visão da interface com o usuário**

<Insira aqui a descrição da interface com o usuário, com a apresentação do mockup do sistema e da descrição detalhada do fluxo de interação (ou de navegação) com o usuário.

Também é de suma importância registrar o raciocínio por trás de cada decisão de design feita sobre a interface com o usuário, em especial se diferentes alternativas de design tiverem sido consideradas.>