**Service Status**

Documento de Arquitetura de Software

Versão <1.0>

[Observação: O template a seguir é fornecido para uso com o Rational Unified Process (RUP). O texto em azul exibido entre colchetes e em itálico (style=InfoBlue) foi incluído para orientar o autor e deve ser excluído antes da publicação do documento. Um parágrafo digitado após esse estilo será automaticamente definido como normal (style=Body Text).]

[Para personalizar campos automáticos no Microsoft Word (que exibem um fundo cinza quando selecionados), escolha File>Properties e substitua os campos Title, Subject e Company pelas informações apropriadas para este documento. Depois de fechar a caixa de diálogo, para atualizar os campos automáticos no documento inteiro, selecione Edit>Select All (ou Ctrl-A) e pressione F9 ou simplesmente clique no campo e pressione F9. Isso deve ser feito separadamente para Cabeçalhos e Rodapés. Alt-F9 alterna entre a exibição de nomes de campos e do conteúdo dos campos. Consulte a ajuda do Word para obter mais informações sobre como trabalhar com campos.]

Histórico da Revisão

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versão** | **Descrição** | **Autor** |
| <dd/mmm/aa> | <x.x> | <detalhes> | <nome> |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Índice Analítico

1. Introdução [4](#__RefHeading___Toc212892805)

1.1 Finalidade [4](#__RefHeading___Toc212892806)

2. Metas e Restrições da Arquitetura [4](#__RefHeading___Toc212892807)

3. Suposições e Dependências [4](#__RefHeading___Toc212892808)

4. Requisitos Arquiteturalmente Significantes [4](#__RefHeading___Toc212892809)

5. Decisões, Restrições e justificativas [4](#__RefHeading___Toc212892810)

6. Mecanismos Arquiteturais [5](#__RefHeading___Toc212892811)

7. Camadas da Arquitetura [5](#__RefHeading___Toc212892812)

8. Visões da Arquitetura [5](#__RefHeading___Toc212892813)

9. Qualidade [5](#__RefHeading___Toc212892814)

Documento de Arquitetura de Software

# Introdução

[A introdução do **Documento de Arquitetura de Software** fornece uma visão geral do documento inteiro. Ela inclui a finalidade, o escopo, as definições, os acrônimos, as abreviações, as referências e a visão geral do **Documento de Arquitetura de Software**.]

## Finalidade

Este documento oferece uma visão geral arquitetural abrangente do sistema, usando diversas visões arquiteturais para representar diferentes aspectos do sistema. O objetivo deste documento é capturar e comunicar as decisões arquiteturais significativas que foram tomadas em relação ao sistema.

[Esta seção define o papel ou finalidade do **Documento de Arquitetura de Software**, na documentação do projeto como um todo, e descreve rapidamente a estrutura do documento. O público-alvo específico do documento é identificado, com uma indicação de como ele espera usar o documento.]

# Metas e Restrições da Arquitetura

[Descreva a filosofia da arquitetura. Identifique as questões que dirigem a arquitetura, como “O sistema será dirigido por funcionalidades complexas, integração com sistemas legados ou questões de desempenho? O sistema precisa ser robusto durante um longo tempo de manutenção ?

[Formule um conjunto de metas que a arquitetura tem de cumprir em sua estrutura e comportamento. Identifique questões críticas que a arquitetura deve resolver, tais como “há dependências de hardware que precisam ser isoladas o resto do sistema?” e “O sistema precisa funcionar de forma eficiente em condições não-usuais?”]

# Suposições e Dependências

[Liste as suposições e dependências que dirigem as decisões arquiteturais. Isto pode incluir áreas sensíveis ou críticas, dependências e interfaces com sistemas legado, a habilidade e experiência da equipe, a disponibilidade de recursos importantes, e assim por diante ]

# Requisitos Arquiteturalmente Significantes

[Insira uma referência ou link para os requisitos que exploram aspectos relevantes da arquitetura.]

# Decisões, Restrições e justificativas

[Liste as decisões tomadas em relação às abordagens arquiteturais. Estas decisões servirão como guias para definir as partes arquiteturalmente significantes do sistema. Justifique cada decisão ou restrição para que os desenvolvedores compreendam a importância de construir o sistema dentro desse contexto. Isto pode incluir uma lista de “Faça” ou “Não Faça” para guiar os desenvolvedores no desenvolvimento do sistema]

* Decision or constraint and justification
* Decision or constraint and justification

# Mecanismos Arquiteturais

[Liste os mecanismos arquiteturais, como mecanismos de persistência, comunicação e tratamento de erros, por exemplo, e descreva ocorrente estado de cada um. Inicialmente, cada mecanismo pode ser somente um nome e uma breve descrição. Eles evoluirão até que o mecanismo se torne um padrão ou uma colaboração de elementos de projeto que possam ser aplicados diretamente em algum aspecto do projeto.]

**Mecanismo Arquitetural 1**

[Descreva a finalidade, os atributos e funções do mecanismo arquitetural.]

**Mecanismo Arquitetural 2**

[Descreva a finalidade, os atributos e funções do mecanismo arquitetural.]

# Camadas da Arquitetura

[Descreva os padrões de arquitetura utilizados e como a arquitetura se manterá consistente e uniforme. Isto pode ser uma simples referência para um conhecido padrão arquitetural, como o padrão de divisão em camadas e uma descrição de como os componentes do sistema podem ser colocados juntos.]

# Visões da Arquitetura

[Descreva as visões arquiteturais usadas para descrever a arquitetura. Isto ilustra as diferentes perspectivas disponíveis para rever e documentar as decisões arquiteturais.]

Visões Recomendadas:

* **Lógica:** Descreva a estrutura e comportamento de porções arquiteturalmente significantes do sistema. Isto deve incluir a estrutura de pacotes, interfaces críticas, importantes classes e subsistemas e as relações entre estes elementos. Isto também inclui visões físicas e lógicas dos dados persistentes.
* **Operacional:** Descreva os nós físicos do sistema e os processos, threads e componentes que rodam em cada um desses nós. Esta visão não é necessária se o sistema roda num único processo e num único thread.
* **Casos de Uso:** Uma lista ou diagrama dos casos de uso que contém requisitos arquiteturalmente relevantes.

# Qualidade

[Uma descrição de como a arquitetura do software contribui para todos os recursos (exceto a funcionalidade) do sistema: extensibilidade, confiabilidade, portabilidade e assim por diante. Se essas características possuírem significado especial, como implicações de segurança, garantia ou privacidade, elas deverão ser delineadas claramente.]