IziMec

Documento de Arquitetura de Software

Versão <1.0>

Histórico da Revisão

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versão** | **Descrição** | **Autor** |
| **17/06/2021** | 1.1 | Criaçao | Rafael da Silva Reis |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Índice Analítico

1. Introdução [4](#__RefHeading___Toc18206175)

1.1. Finalidade [4](#__RefHeading___Toc18206176)

1.2. Escopo [4](#__RefHeading___Toc18206177)

1.3. Definições, Acrônimos e Abreviações [4](#__RefHeading___Toc18206178)

1.4. Referências [4](#__RefHeading___Toc18206179)

1.5. Visão Geral [4](#__RefHeading___Toc18206180)

2. Representação Arquitetural [4](#__RefHeading___Toc18206181)

3. Metas e Restrições da Arquitetura [4](#__RefHeading___Toc18206182)

4. Visão de Casos de Uso [4](#__RefHeading___Toc18206183)

4.1. Realizações de Casos de Uso [6](#__RefHeading___Toc18206184)

5. Visão Lógica [6](#__RefHeading___Toc18206185)

6. Visão de Processos [7](#__RefHeading___Toc18206188)

7. Visão de Implantação [7](#__RefHeading___Toc18206189)

8. Visão da Implementação [7](#__RefHeading___Toc18206190)

8.1. Visão Geral [7](#__RefHeading___Toc18206191)

8.2. Camadas [8](#__RefHeading___Toc18206192)

9. Visão de Dados (opcional) [8](#__RefHeading___Toc18206193)

10. Tamanho e Desempenho [8](#__RefHeading___Toc18206194)

11. Qualidade [8](#__RefHeading___Toc18206195)

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.

## Escopo

Estão contemplados no Documento de Arquitetura de Software os padrões de software, componentes de software, plataformas de desenvolvimento, plataformas de hardware, softwares de desenvolvimento, servidores de aplicação, servidores de banco de dados, sistemas operacionais, frameworks e APIs.São também descritos neste “Documento de Arquitetura de Software-DAS” a descrição dosfocos e sistemáticas arquiteturais, descrição das camadas de que é composto o modelo arquitetural, requisitos de segurança, requisitos de desempenho e requisitos de integrações.

## Definições, Acrônimos e Abreviações

Vide Glossario

## Referências

Documento de Visão de Negócio

Documento de Regras de Negócio

Glossário de Negócio

Documento de Especificação de Requisitos de Software

## Visão Geral

# Representação Arquitetural

Esse documento apresenta a arquitetura como uma série de visualizações: visualizações de caso de uso, visualização de processo, visualização da implementação e visualização de implantação. Essas visualizações são apresentadas como Rational Rose Models e utilizam o UML (Unified Modeling Language).

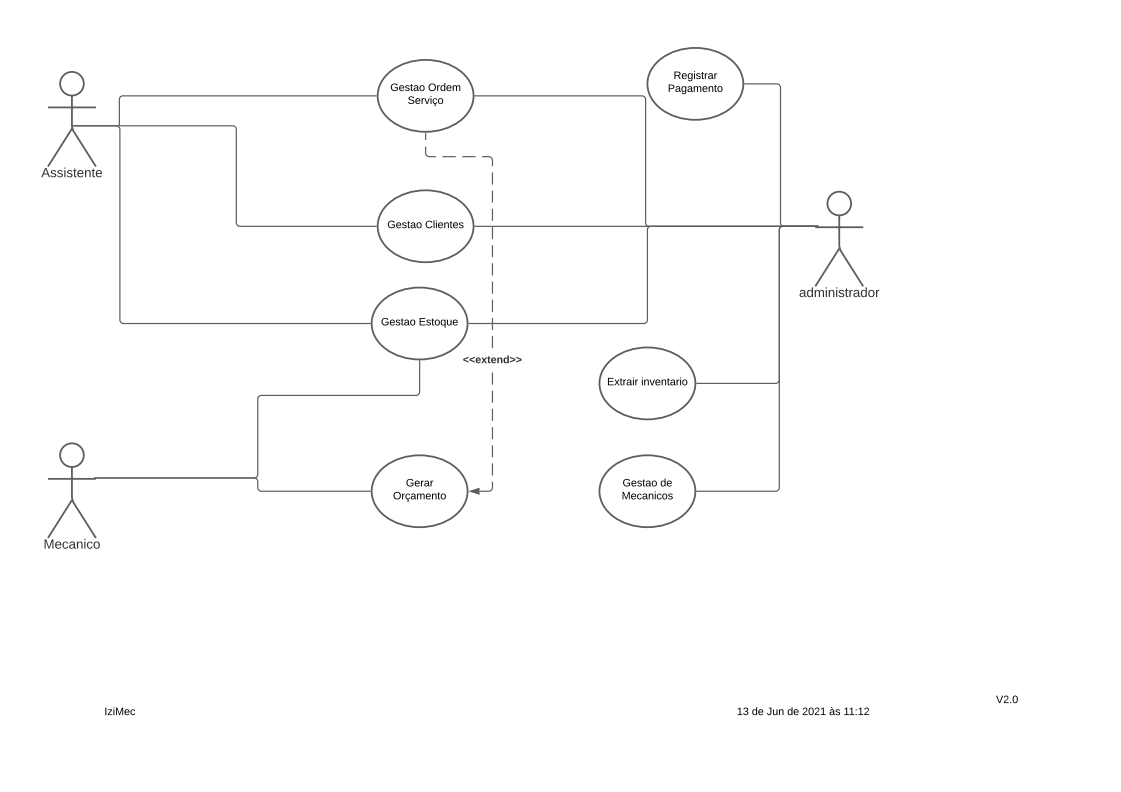
# Metas e Restrições da Arquitetura

O sistema deve estar hospedado em um servidor da mesma linguagem de programação e versão do software, e ter acesso a uma rede estável para a conexão com os usuários.

* Todos os requisitos de desempenho e de carregamento, conforme estipulados no Documento de Visão , devem ser considerados como a arquitetura sendo desenvolvida.

# Visão de Casos de Uso

Uma descrição da visualização de casos de uso da arquitetura de software. A Visualização de Caso de Uso é uma entrada importante na seleção do conjunto de cenários e/ou casos de uso que são o foco de uma interação. Ela descreve o conjunto de cenários e/ou os casos de uso que representam alguma funcionalidade central e significativa. Também descreve o conjunto de cenários e/ou casos de uso que possuem cobertura arquitetural substancial (que exercita vários elementos de arquitetura) ou que enfatizam ou ilustram um determinado ponto complicado da arquitetura. Os casos de uso nesse sistema são demonstrados no Diagrama abaixo.



## Realizações de Casos de Uso

1. Gestão Ordem de Serviços

Esse Caso de Uso ocorre quando o assistente recebe uma atribuição do cliente e registra no sistema IziMec. Podendo ser alterado, editado ou excluído pelo autor assistente.

1. Gestão Clientes

Esse caso de uso Ocorre quando o assistente registra os dados do cliente no sistema Izimec. Podendo ser alterado ,editado ou excluído pelo autor.

1. Gestão Estoque.

Esse caso de uso ocorre quando o ator assistente registra um produto no estoque e quando o ator mecânico registra uma retirada de peça do sistema IziMec.

1. Gerar Orçamento

Esse caso de uso ocorre quando é criada uma Os no sistema e podendo ser criado um orçamento que será preenchido pelo ator mecânico

1. Registrar Pagamento

Esse caso de uso ocorre quando o Ator Administrador deseja registrar um pagamento efetuado pelo cliente.

1. Extrair Inventario

Esse caso ocorre quando o Ator administrador deseja extrair relatórios gerenciais do inventario da oficina

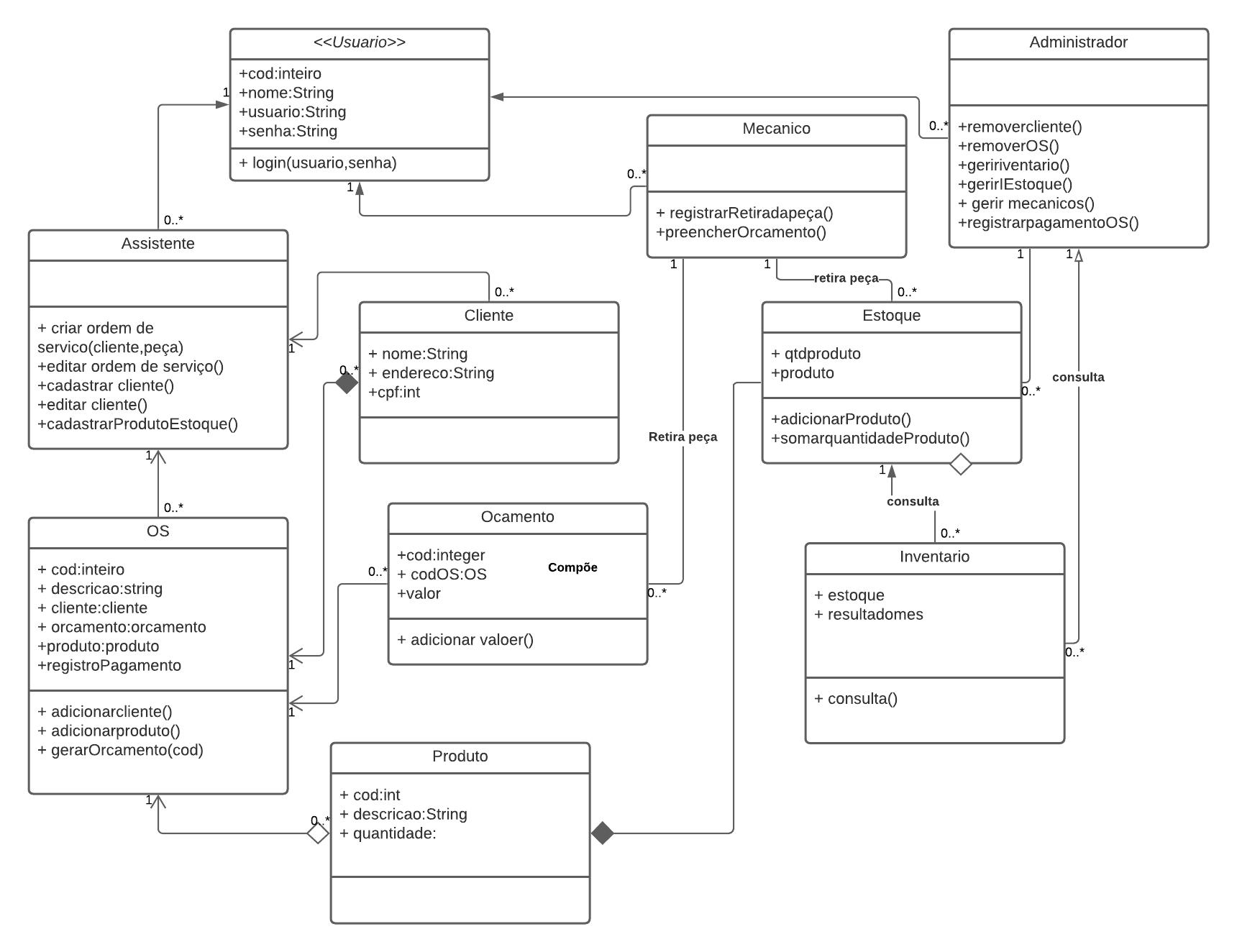
1. Gestão de Mecânicos

Esse caso de uso ocorre quando o Ator Administrador deseja registrar, alterar ou excluir um novo mecânico ao Sistema.

# Visão Lógica

No Diagrama de classes é uma representação estática utilizada na área da programação para descrever a estrutura de um sistema, apresentando suas classes, atributos, operações e as relações entre os objetos .

Logo abaixo temos a representação do Diagrama de classe do sistema IziMec.



# Visão de Processos

[Esta seção descreve a decomposição do sistema em processos leves (threads simples de controle) e processos pesados (agrupamentos de processos leves). Organize a seção em grupos de processos que se comunicam ou interagem. Descreva os modos principais de comunicação entre processos, como transmissão de mensagens e interrupções.]

# Visão de Implantação

[Esta seção descreve uma ou mais configurações da rede física (hardware) na qual o software é implantado e executado. Ela é uma visão do Modelo de Implantação. No mínimo, para cada configuração, ela deve indicar os nós físicos (computadores, CPUs) que executam o software e suas interconexões (barramento, LAN, ponto a ponto, etc.) É incluído também um mapeamento dos processos da **Visão de Processos** nos nós físicos.]

# Visão da Implementação

[Esta seção descreve a estrutura geral do modelo de implementação, a divisão do software em camadas e os subsistemas no modelo de implementação e todos os componentes significativos do ponto de vista da arquitetura.]

## Visão Geral

[Esta subseção nomeia e define as diversas camadas e o seu conteúdo, as regras que determinam a inclusão em uma camada específica e as fronteiras entre as camadas. Inclua um diagrama de componentes que mostre os relacionamentos entre as camadas. ]

## Camadas

[Para cada camada, inclua uma subseção com o respectivo nome, uma lista dos subsistemas localizados na camada e um diagrama de componentes.]

# Visão de Dados (opcional)

[Uma descrição da perspectiva de armazenamento de dados persistentes do sistema. Esta seção será opcional se os dados persistentes forem poucos ou inexistentes ou se a conversão entre o Modelo de Design e o Modelo de Dados for trivial.]

# Tamanho e Desempenho

[Uma descrição das principais características de dimensionamento do software que têm um impacto na arquitetura, bem como as restrições do desempenho desejado.]

# 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.]