FORTESI

Documento de Arquitetura de Software

Versão 1.7

Histórico da Revisão

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versão** | **Descrição** | **Autor** |
| 10/04/2015 | 1.0 | Elaboração inicial do documento | Lucas Castanheiro |
| 18/04/2015 | 1.1 | Caso de uso do documento | José Lucas |
| 24/04/2015 | 1.2 | Representação arquitetural do documento | Lucas Castanheiro |
| 30/04/2015 | 1.3 | Implantação e implementação | Raimundo Neto |
| 01/05/2015 | 1.4 | Revisão do documento | Raimundo Neto |
| 03/05/2015 | 1.5 | Diagrama de sequência | José Lucas |
| 04/05/2015 | 1.6 | Diagrama de pacotes e camadas | Lucas castanheiro |
| 25/052015 | 1.7 | Revisão do documento | Lucas Castanheiro |

Índice Analítico

1. Introdução 4

1.1 Finalidade 4

1.2 Escopo 4

1.3 Definições, Acrônimos e Abreviações 4

1.4 Referências 4

1.5 Visão Geral 4

2. Representação Arquitetural 5

3. Metas e Restrições da Arquitetura 6

4. Visão de Casos de Uso 6

4.1 Realizações de Casos de Uso 7

5. Visão Lógica 8

5.1 Visão Geral 8

5.2 Pacotes de Design Significativos do Ponto de Vista da Arquitetura 8

6. Visão de Processos 10

7. Visão de Implantação 10

8. Visão da Implementação 10

8.1 Visão Geral 10

8.2 Camadas 11

9. Visão de Dados (opcional) 12

10. Tamanho e Desempenho 12

11. Qualidade 12

Documento de Arquitetura de Software

# 1. Introdução

Este documento representa a arquitetura do sistema FORTESI, tendo como objetivo apresentar de forma transparente as funcionalidades e desempenhos do mesmo e de como ele está organizado os seus componentes.

## 1.1 Finalidade

Este documento oferece uma visão geral arquitetural abrangente do sistema, usando diversas visões arquiteturais para representar diferentes aspectos do sistema capturando e comunicando as decisões arquiteturais significativas que foram tomadas em relação ao sistema.

## 1.2 Escopo

Este documento auxilia e orienta na implementação do software como um todo, fornecendo uma visão ampla do mesmo que vem a ser desenvolvido para melhoras e otimizar as atividades diária de um advogado ou advocacia.

## 1.3 Definições, Acrônimos e Abreviações.

Crud – cadastrar, alterar, adicionar ou remover;

MVC – Modelo, Visão e Controle;

Java Web – Linguagem de Programação;

MySQL – Banco de Dados;

Fortesi – Gerenciador de processos jurídicos;

Releases – Um release do sistema consistirá do sistema (ou parte dele) pronto para uso e que agrega algum valor ao negócio do cliente.

## 1.4 Referências

Padrões de Projeto. Disponível em:

<http://www.devmedia.com.br/introducao-ao-padrao-mvc/29308>

Sistemas da mesma área. Disponível em:

<http://www.projuridico.com.br/>

## 1.5 Visão Geral

O restante do documento está organizado da seguinte forma:

2. Representação da arquitetura: Esta seção descreve qual é a arquitetura do software atual e como ela é representada.

3. Metas e restrições da arquitetura: Esta seção descreve os requisitos do software e os seus objetivos que tem impactos significativos na arquitetura, como segurança e privacidade, produto apresenta portabilidade, rapidez, reaproveitamento e interface de fácil entendimento para o usuário.

4. Caso de Uso: Representação das funcionalidades de situações do dia a dia.

5. Visão Logica: Representa os pacotes e classe dos subsistemas do FORTESI

6. Visão de Processos: Descreve os processos dentro do sistema como estão organizados e como são realizados.

7. Implantação: Descreve as configurações computacionais necessárias para o sistema.

8. Implementação: Descreve o método de desenvolvimento do software.

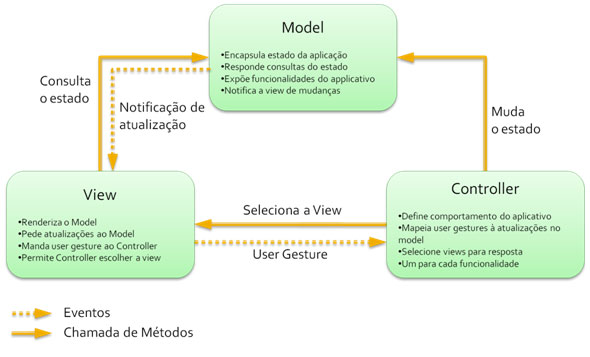
9. Visão Dados: Apresenta a forma de como os dados irão ser armazenados no sistema.

10. Tamanho e Desempenho: Descreve o dimensionamento do software e os impactos no desempenho.

11. Qualidade: Descreve as qualificações do software.

# 2. Representação Arquitetural.

A representação arquitetural do FORTESI baseia-se no modelo (MVC) Modelo, Visão e Controle, neste modelo a arquitetura encontra-se representada por cinco visões.



As funcionalidades do sistema estão compreendidas da seguinte forma,

* Visão de Caso de Uso Mostra as funcionalidades do sistema e as funções utilizadas em uma rotina de trabalho de um advogado, desde a interação, entradas fornecidas e chamamento dos métodos para o retorno ao usuário.
* Visão logica apresenta alguns pacotes, camadas e classes mais significativas e impactantes para a arquitetura do software.
* Visão de Processos mostra todas as funcionalidades do sistema, como estão integradas e tem parte significativa na arquitetura o que uma impacta na outra.
* Visão de Implantação As necessidades básicas em hardware, o sistema tem funcionalidades em computadores com sistema operacional Windows a partir da geração Xp.
* Visão de Implementação Linguagem Java web com banco de dados MySQL.

# 3. Metas e Restrições da Arquitetura

A meta principal desse documento de Arquitetura do software Fortesi é de impor um maior entendimento entre os desenvolvedores e mostrar quais elementos serão implementados dentro de cada iteração detalhadamente. Nossa proposta de arquitetura envolveu fatores como a finalidade do sistema e o tipo de usuário. Características a ser adotada nesse documento são:

* Modularidade: o sistema deve ser desenvolvido em camadas, havendo uma interface de comunicação bem definida entre as mesmas.
* Reusabilidade: a arquitetura do sistema deve ser tal que permita a utilização de classes e componentes em outros projetos, favorecendo o tempo de produção e a qualidade do produto gerado.

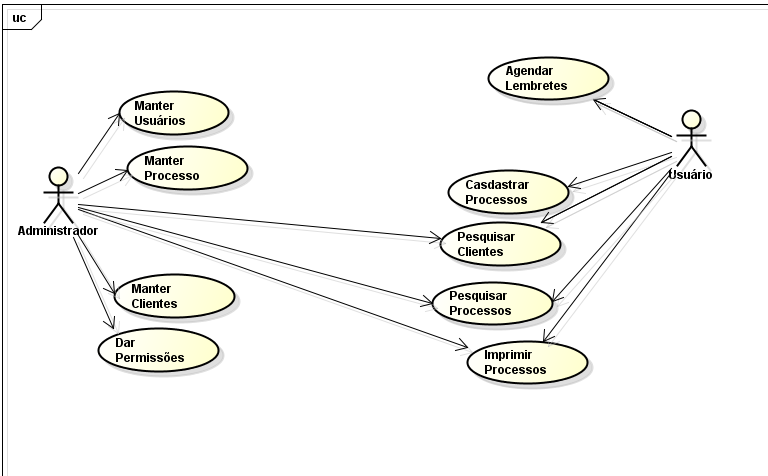
As restrições são:

* Os prazos de lançamento dos releases do sistema devem ser mostrados nas datas corretas.
* O sistema deve seguir as regras de negócio previamente analisadas e discutidas, sem alterar o funcionamento atual do negócio.

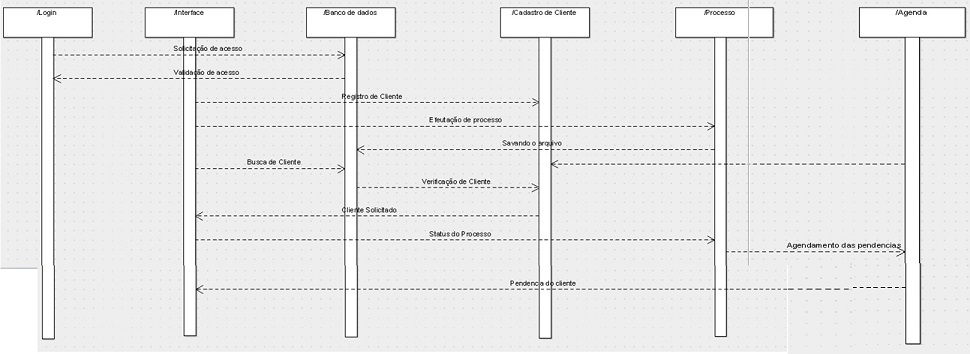
# 4. Visão de Casos de Uso

Abaixo estão apresentados os casos de uso que representam a funcionalidade do Fortesi.

* Login.
* Agenda.
* Crud de processos.
* Buscar informações do processo.



## 4.1 Realizações de Casos de Uso

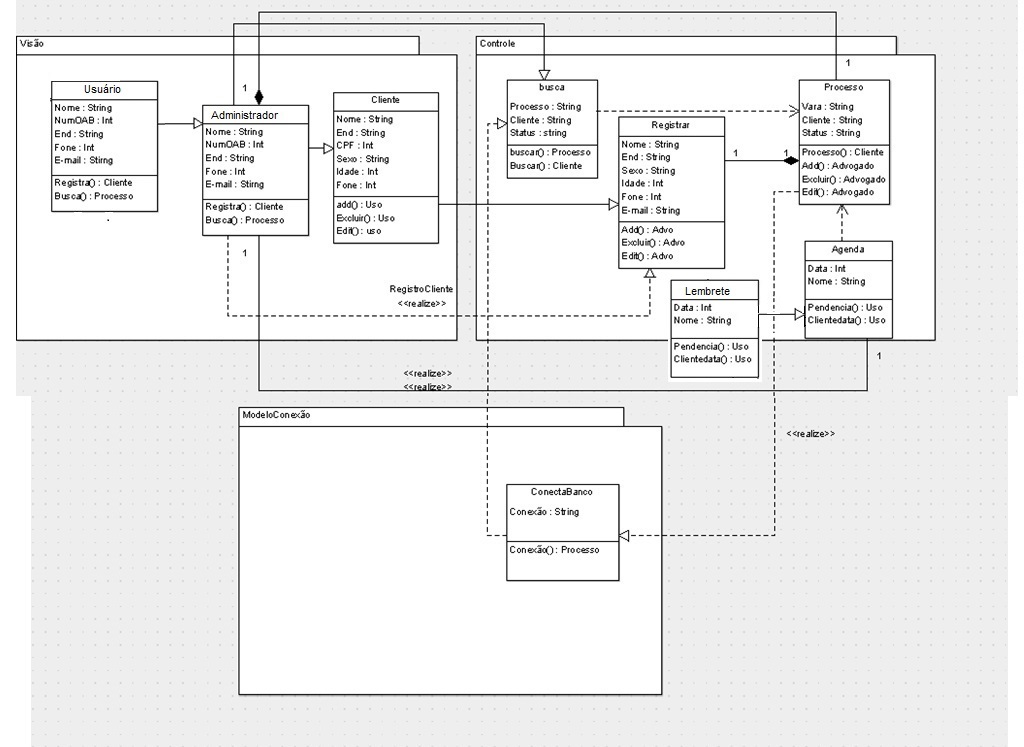


# 5.Visão Lógica

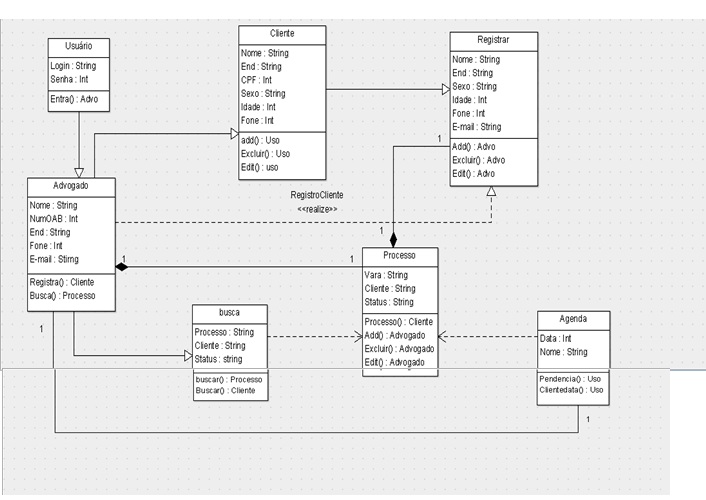
## 5.1Visão Geral

* No pacote visão cotem todas as classes que interligam a parte gráfica com as interfaces relacionando a interação com usuário.
* No pacote controle contem todas as classes que controlarão as partes funcionais do sistema.
* No pacote modelo contem às informações que se utilizará como modelo.

## 5.2Pacotes de Design Significativos do Ponto de Vista da Arquitetura



# 6.Visão de Processos



# 7. Visão de Implantação

Esta secção apresenta as configurações de rede física do sistema:

* Configuração do Computador: sistema operacional Windows de geração 7/8 e 8.1.
* Memória RAM de no mínimo 2Gb.
* Processador Intel 2.10 GHZ.
* HD Interno.

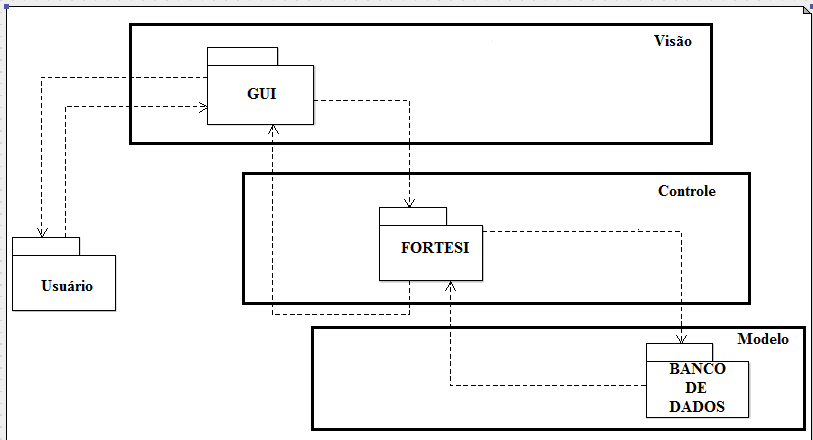
# 8. Visão da Implementação

## 8.1 Visão Geral

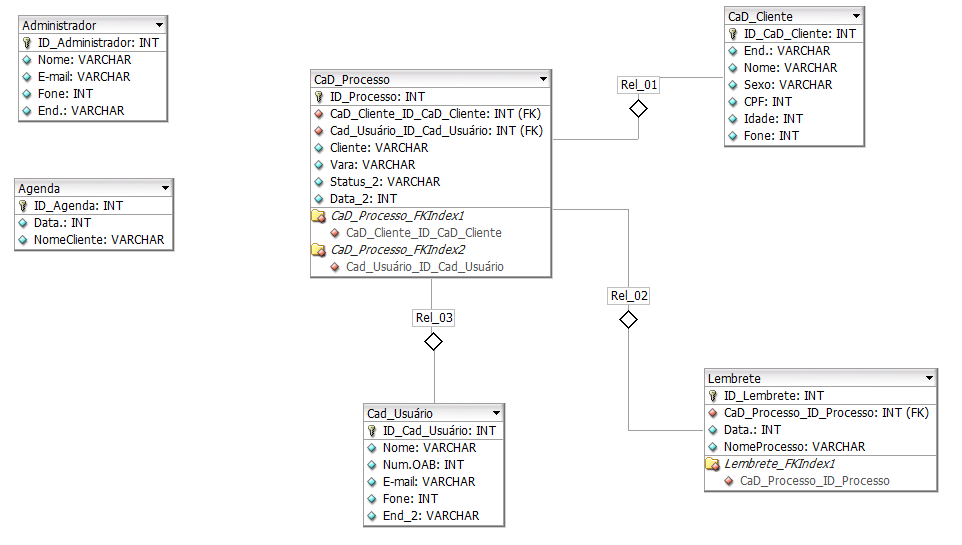
O sistema está organizado no padrão arquitetural MVC nas seguintes camadas:

* Visão – O usuário através da interação com a interface (GUI) fornece as entradas requeridas para a utilização do sistema, busca de processos, cadastro de cliente, cadastro de processo e visualização.
* Controle – O FORTESI interpreta as entradas requeridas pelo o usuário através da interface desenvolvida em Java Web.
* Modelo – O banco de dados gerencia as informações e conteúdos dentro, faz a solicitação, retira, adiciona, exclui e atualiza as informações necessárias retorna com uma saída que é repassada ao usuário através da interface.

## 8.2Camadas



# 9.Visão de Dados



# 10.Tamanho e Desempenho

O tamanho do software FORTESI é constate, mas tendo em vista que sua manutenção poderá trazer modificações que acarretaram na alteração do seu tamanho e também na questão do banco de dados que podem variar de tamanhos.

O desempenho do FORTESI apresenta uma interface de simples entendimento e agradável mantendo o usuário mais seguro das suas decisões mesmo que ele não tenha um conhecimento básico de informática ou do sistema.

# 11. Qualidade

A elaboração da arquitetura deve trazer uma interação bem sucedida e de fácil entendimento ao usuário, trazendo benefícios de qualidade como rapidez, confiabilidade, segurança e autenticidade que são algumas características fundamentais no desenvolvimento de software.