

**Projudi**

**Documento de Arquitetura de Software**

**Versão 1.00**

Projudi	Versão: 1.0
Documento de Arquitetura de Software	Data: 05/05/2009
Documento de Arquitetura de Software - Projudi.doc	

## Histórico de Revisões

<b>Data</b>	<b>Versão</b>	<b>Descrição</b>	<b>Autor</b>
29/04/2009	1.0	Registro inicial	Paulo Eduardo Menon

Projudi	Versão: 1.0
Documento de Arquitetura de Software	Data: 05/05/2009
Documento de Arquitetura de Software - Projudi.doc	

# Sumário

1. Introdução ao Documento de Arquitetura de Software.....	4
1.1 Propósito.....	4
1.2 Escopo.....	4
1.3 Referências.....	5
1.4 Visão Geral do Documento de Arquitetura de Software.....	5
1.5 Metodologias de Desenvolvimento do Projudi.....	5
1.6 Representação Arquitetural do Documento de Arquitetura de Software.....	6
1.6.1 Visão de Casos de Uso.....	6
1.6.2 Visão Lógica.....	6
1.6.3 Visão de Processos.....	7
1.6.4 Visão de Implantação.....	7
2. Visão de Implantação – A Distribuição do Projudi.....	8
2.1 Máquinas do Projudi.....	8
2.1.1 Máquina tal.....	8
3. Visão de Casos de Uso.....	9
3.1 Cadastra Ação.....	9
3.2 Movimentação.....	9
3.2.1 Uso do Torque.....	9
4. Visão Lógica.....	10
4.1.1 Camada de Infra-Estrutura.....	10
4.2 Pacotes de Projeto Significantes da Arquitetura.....	10
4.2.2 Caracterização dos Domínios e Movimentos.....	11
4.2.3 Exceções do Projudi.....	11
5. Próximas Leituras.....	12

Projudi	Versão: 1.0
Documento de Arquitetura de Software	Data: 05/05/2009
Documento de Arquitetura de Software - Projudi.doc	

# Documento de Arquitetura de Software

## 1.Introdução ao Documento de Arquitetura de Software

O documento de arquitetura de software representa um dos artefatos do processo de engenharia do RUP<sup>1</sup> padronização criada pela Rational. Este processo de engenharia foi selecionado para direcionar o desenvolvimento do Processo Judicial Digital, cujo acrônimo utilizado ao longo do documento será Projudi.

Nesta seção, intitulada Introdução, e nas suas subseções, será apresentada uma visão geral do artefato Documento de Arquitetura de Software, definindo-se o seu propósito no contexto do processo, o seu escopo de representação, vinculado ao software em desenvolvimento, bem como as definições, os acrônimos, as abreviações, e as referências utilizadas. Para finalizar a seção, será apresentada uma visão geral do documento em relação ao Projudi.

A partir da seção dois, ter-se-á a representação detalhada da arquitetura do sistema, bem como as suas visões arquiteturais mais expressivas.

### 1.1Propósito

O propósito do documento de arquitetura é fornecer um molde de desenvolvimento que deve guiar as atividades de projeto e implementação do *software*. De fato, por representar as decisões significativas do projeto, o documento de arquitetura estabelece a estrutura colaborativa de elementos do sistema, os quais definem um modelo de construção que deve direcionar as demais atividades de produção técnica relacionadas ao projeto.

E exemplificar como esta a atual arquitetura utilizada no Projudi, descrevendo suas partes e visões significativas para termos uma visão geral do sistema.

### 1.2Escopo

O escopo do documento refere-se às características que deverão ser tratadas em relação ao sistema que se está desenvolvendo e que serão retratadas no documento de arquitetura de software. Desta forma, o Projudi representa um sistema que trata do registro e acompanhamento dos processos judiciais eliminando a parte física do papel, além de ter uma integração nacional das OABs e com consulta em tempo real.

---

<sup>1</sup> RUP representa um acrônimo para o termo em inglês *Rational Unified Process*.

Projudi	Versão: 1.0
Documento de Arquitetura de Software	Data: 05/05/2009
Documento de Arquitetura de Software - Projudi.doc	

### 1.3Referências

1. Especificações Suplementares [ES]
2. [Using the RUP to evolve a legacy system](#) [PK]
3. Philippe Kruchten
4. Design Patterns [GA]
5. Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides. Addison-Wesley
6. <http://db.apache.org/torque/> [TR]
7. Core J2EE Patterns [DA]
8. Deepak Alur, John Crupi, Dan Malks. Prentice Hall.

### 1.4Visão Geral do Documento de Arquitetura de Software

Nas próximas seções deste documento serão representados os detalhes significativos de construção arquitetural do Projudi, através da ordem a seguir.

Na seção 1.6, intitulada “Representação Arquitetural”, serão apresentadas as visões arquiteturais, como definidas no RUP, e a forma como essas visões se integram para fornecer uma perspectiva completa do sistema.

Na seção , com o título de “”, é apresentada uma breve descrição dos requisitos de software, que possuem impactos representativos na estrutura arquitetural do sistema, bem como algumas notas relacionadas aos motivos que definem esses requisitos como críticos para a aplicação. Além disso, discorre-se sobre requisitos não funcionais que podem ter influência significativa nas soluções arquiteturais do projeto, como performance e desempenho, e considerações sobre a estrutura de implementação do sistema.

Da seção 2 até a seção são documentados os detalhes de construção do sistema, como base nas visões apresentadas na seção dois. Isso possibilita que grupos distintos de interessados no sistema possam visualizar o *software* de acordo com sua importância, ao mesmo tempo em que permite a representação completa da estrutura significativa do projeto em desenvolvimento.

### 1.5Metodologias de Desenvolvimento do Projudi

O projeto engloba o paradigma de orientação a objetos e estruturada junto com o front end que se utiliza de scriptless JSP, HTML e Javascript. Na parte do domínio as classes com o padrão do Torque seguem a nomenclatura nome do caso de uso seguido do nome peer, herdando da API padrão do

Projudi	Versão: 1.0
Documento de Arquitetura de Software	Data: 05/05/2009
Documento de Arquitetura de Software - Projudi.doc	

Torque.

No caso do Projudi, os dois paradigmas de desenvolvimento estão sendo utilizados. Como este documento possui o objetivo de ser um documento de arquitetura geral para o projeto, tanto a parte do sistema que utiliza a orientação a objetos, como a parte da aplicação desenvolvida em linguagem estruturada e o front end, não terão representadas nesse documento utilizando UML, de acordo com as considerações do documento *Using the RUP to Evolve a Legacy System* [PK].

## 1.6 Representação Arquitetural do Documento de Arquitetura de Software

A arquitetura é à base da produtividade de software, contribuindo para a manutenção e extensibilidade das funcionalidades do sistema.

Este documento ilustra a arquitetura do Projudi a partir das visões arquiteturais definidas no RUP. Cada visão aborda um determinado conjunto de perspectivas específicas do desenvolvimento do sistema, o qual interessa mais diretamente a um grupo distinto de papéis do processo, e em fases e iterações diferenciadas.

Porém, do ponto de vista da arquitetura do sistema, todas as perspectivas representam categorias complementares de análise, as quais, vistas enquanto uma unidade, definem a estrutura completa da aplicação. Como essas visões arquiteturais constituem-se em categorias do RUP, as suas representações seguem as definições semânticas da UML, e estão voltadas a papéis específicos do processo.

Segue detalhamento de cada visão. Note que essa ordem é a comumente encontrada nos documentos de arquitetura da Rational®, entretanto não é a seguida neste documento para se tornar mais didático.

### 1.6.1 Visão de Casos de Uso

Na visão de Casos de Uso, os analistas, gerentes de projeto e outros interessados nas regras de negócio do sistema, terão uma amostra dos casos de uso de maior criticidade arquitetural, denominados de casos de uso arquiteturalmente significativos. Esses casos de uso são importantes, pois representam elementos do projeto que podem contribuir para o **insucesso** da implementação do sistema, devido aos seus riscos técnicos.

Além disso, são considerados casos de uso arquiteturalmente significativos, aqueles casos de uso cujo projeto possa representar uma estrutura comum de desenvolvimento, de tal forma que a realização de outros Casos de Uso siga as mesmas diretrizes, representando, dessa forma, um modelo arquitetural expressivo.

Essa visão é trabalhada na seção 3, “Visão de Casos de Uso” (p. 9).

### 1.6.2 Visão Lógica

Na visão lógica, ou visão de projeto, os projetistas e implementadores poderão entender a decomposição do sistema em pacotes, camadas e classes, suas responsabilidades, dependências e mecanismos, baseados em padrões de projetos, que permitem agregar qualidade ao sistema e possibilita maior flexibilidade e extensibilidade.

Projudi	Versão: 1.0
Documento de Arquitetura de Software	Data: 05/05/2009
Documento de Arquitetura de Software - Projudi.doc	

Essa visão é trabalhada na seção 4, “Visão Lógica”.

### **1.6.3 Visão de Processos**

A visão de processo exibe os principais processos e as principais *threads*<sup>2</sup> do sistema, mostrando a concorrência de tarefas em tempo de execução do sistema. Essa visão só é obrigatória se o sistema possuir um grau significativo de simultaneidade, ou nos casos onde o arquiteto deseja agrupar elementos lógicos do projeto, em processos distintos, para representá-los em processadores separados.

Essa visão é trabalhada na seção , “”.

### **1.6.4 Visão de Implantação**

Já a visão de implantação deve mostrar uma ou mais configurações do servidor de aplicação na qual o software será implantado e executado. Essa visão contém a descrição dos vários softwares que devem ser instalados, seja de sistema operacional, seja de banco de dados, seja de contêiner web, e a alocação de recursos físicos. Através da visão de implantação pode representar também a capacidade do servidor, para processamento e capacidade de armazenamento de dados e com a configuração da rede para verificar a largura de banda projetada entre as comunicações entre as pontas e o servidor central. Isto é necessário para os sistemas com graus representativos de distributividade, e como interprocessos podem interferir ou otimizar o desempenho total do sistema.

Essa visão é trabalhada na seção 2, “Visão de Implantação ”.

---

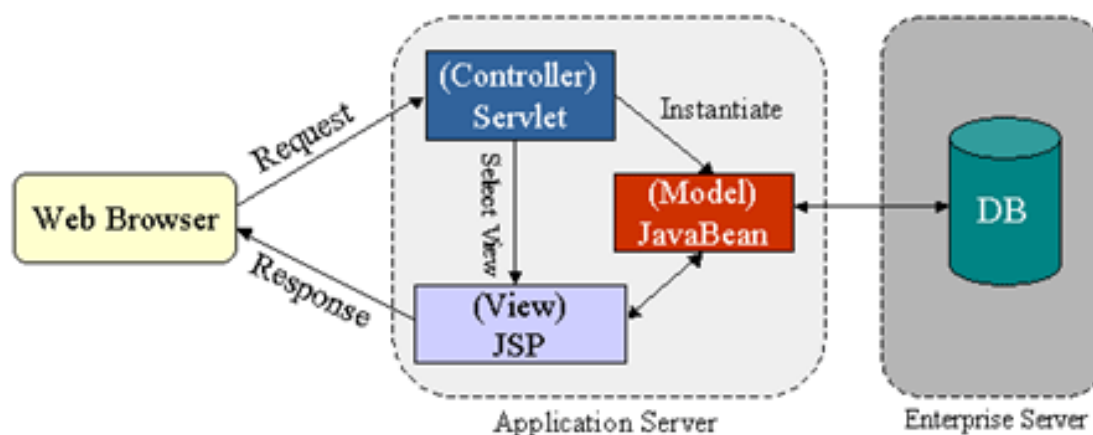
<sup>2</sup> Threads podem ser entendidas como um fluxo de controle dentro de um processo, e que compartilham as suas variáveis de instâncias.

Projudi	Versão: 1.0
Documento de Arquitetura de Software	Data: 05/05/2009
Documento de Arquitetura de Software - Projudi.doc	

## 2. Visão de Implantação – A Distribuição do Projudi

A visão de implantação é responsável por ilustrar a constituição física do sistema. São descritas as máquinas com capacidade de processamento, as aplicações que são executadas nessas máquinas e como é a comunicação do sistema.

A ilustra o esquema geral da arquitetura do Projudi.



### 2.1 Máquinas do Projudi

O Projudi será executado em uma plataforma de hardware distinta:

#### 2.1.1 Máquina tal

Plataforma de captação das operações das estações dos usuários. Executa o servidor de aplicações do Projudi (atualmente o Tomcat 5.5 e sistema operacional Red Hat 5.).



Projudi	Versão: 1.0
Documento de Arquitetura de Software	Data: 05/05/2009
Documento de Arquitetura de Software - Projudi.doc	

### 3. Visão de Casos de Uso

Esta seção foi elaborada considerando-se os casos de uso ou cenários de caso de uso que representam funcionalidades centrais e significativas do sistema e que experimentam elementos arquiteturais ou enfatizam ou ilustram um determinado ponto da arquitetura.

Esses casos de uso exercitam os mecanismos arquiteturais utilizados no Projudi e, por isso, servirão como provas de conceito para a averiguação das soluções implementadas.

#### 3.1 Cadastra Ação

O caso de uso “Cadastra Ação” é o principal caso de uso do sistema. Ele é responsável pelo registro dos processos judiciais no sistema, uma das operações mais utilizadas no Projudi. Uma variedade expressiva de casos de uso deve seguir os mesmos modelos de implementação do “Cadastra Ação”.

#### 3.2 Movimentação

O “Movimentação” representa uma funcionalidade disponibilizada no Projudi para fazer a movimentação do processo conforme o tipo de movimentação disponível, sendo a segunda parte mais importante do sistema.

##### 3.2.1 Uso do Torque

A principal característica desses casos de uso é o fato de o cliente ser uma aplicação WEB que acessa o servidor de aplicação através do protocolo https. Esse caso de uso possibilita verificar a arquitetura de web adotada, baseada no framework Torque, e auxilia o estabelecimento de padrões de projetos para isso.

Projudi	Versão: 1.0
Documento de Arquitetura de Software	Data: 05/05/2009
Documento de Arquitetura de Software - Projudi.doc	

## 4. Visão Lógica

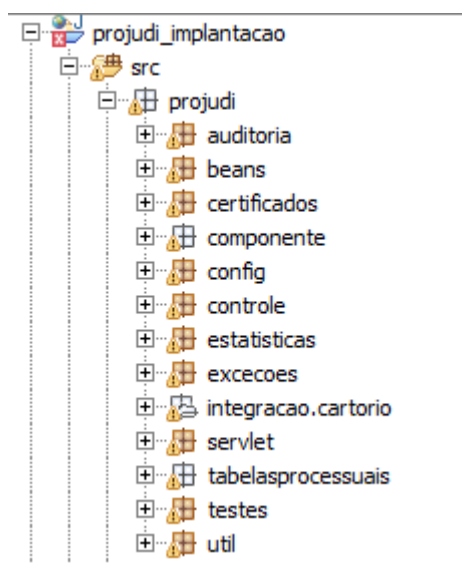
O objetivo da visão lógica é representar a estrutura de relacionamento entre os elementos arquiteturais da aplicação, definindo seus padrões e conceitos gerais, enfatizando principalmente o tempo de compilação. Nesta seção, inicialmente apresenta-se a estrutura de camadas arquiteturais do Projudi.

### 4.1.1 Camada de Infra-Estrutura

A camada de infra-estrutura contém a base de classes e serviços para ser usada na camada de casos de uso. Essa camada contém configurações específicas de cada estado, layout da página principal, o *contex.xml* com as configurações de conexão com o banco de dados e os *torque.properties* e *projudi.properties* com as configurações de link, email e parâmetros do sistema.

### 4.2 Pacotes de Projeto Significantes da Arquitetura

Seguindo a separação de camadas definida:



**Figura - Estrutura de Pacotes**

A organização em pacotes pode ser visualizada na Figura - Estrutura de Pacotes.

#### 4.2.1.1 Pacote de utilitários – projudi.util

Esse pacote possui um conjunto de classes utilitárias que facilitam o reaproveitamento de código no desenvolvimento do sistema. Essas classes incluem algoritmos de formatação de CPF e CNPJ, envio de email, cálculo de prazos, validação PIS. São classes leves e não devem depender de nada além da própria

Projudi	Versão: 1.0
Documento de Arquitetura de Software	Data: 05/05/2009
Documento de Arquitetura de Software - Projudi.doc	

API padrão do J2SE.

#### **4.2.2 Caracterização dos Domínios e Movimentos**

##### **4.2.2.1 O que são os domínios?**

Uma das principais características do paradigma de orientação a objetos é poder desenvolver estruturas de algoritmos, utilizando-se os mesmos termos definidos nas representações de negócio do sistema. No caso do Projudi, todos os termos significativos das regras de negócio foram modelados, no contexto do sistema, como classes de negócio da aplicação com a denominação de Peer conforme padrão do framework Torque. Assim, têm-se domínios como Processo, Movimentação, ArquivoDocumento, AreaRecurso, e outros termos específicos da regra de negócio que são mapeados a classes do sistema.

##### **4.2.3 Exceções do Projudi**

Discorrer sobre *TorqueException*.

Projudi	Versão: 1.0
Documento de Arquitetura de Software	Data: 05/05/2009
Documento de Arquitetura de Software - Projudi.doc	

## 5.Próximas Leituras

Este documento de arquitetura representa um dos principais artefatos para a documentação das definições gerais do projeto. Optou-se por subdividir o tratamento dos assuntos de arquitetura em volumes dedicados às aplicações do Projudi.

Recomenda-se a leitura dos demais volumes, de acordo com a necessidade. São eles:

- 
-