

UNIVERSIDADE ESTADUAL PAULISTA  
“JÚLIO DE MESQUITA FILHO”  
Instituto de Geociências e Ciências Exatas – IGCE  
Curso de Bacharelado em Ciências da Computação

RODRIGO CESAR ANTONIALLI

**SISTEMA DE TRATAMENTO DE DADOS DE GEOLOGIA (STRADGE)**

Trabalho realizado sob orientação do Prof. Dr. Ivan Rizzo Guilherme, DEMAC/IGCE  
Período: 11.2010 a 04.12.2010

Rio Claro-SP  
2010

## **SISTEMA DE TRATAMENTO DE DADOS DE GEOLOGIA (STRADGE)**

Trabalho de Conclusão do Curso, modalidade Trabalho de Graduação, apresentado, no 2º semestre de 2010, à disciplina ES/TG do Curso de Bacharelado em Ciências da Computação, período Noturno, do Instituto de Geociências e Ciências Exatas da Universidade Estadual Paulista “Júlio de Mesquita Filho”, campus de Rio Claro, para apreciação segundo as normas estabelecidas pelo Conselho do Curso, em 27.11.2007.

Aluno: Rodrigo Cesar Antonialli

Supervisor/Orientador: Ivan Rizzo Guilherme  
Departamento de Estatística, Matemática Aplicada e Computação

Rio Claro-SP  
2010

## AGRADECIMENTOS

Talvez esta seja uma das partes mais difíceis e agradáveis da elaboração do trabalho. É difícil listar todos aqueles que colaboraram direta ou indiretamente para o desenvolvimento deste trabalho, e ainda muito difícil transmitir o quão importante foram as pessoas que participaram deste processo. Ainda sim, é extremamente agradável saber que podemos contar com tantas pessoas ao longo da vida, para nos apoiar, apontar os nossos erros e o mais importante, nos fazer melhorar sempre.

Gostaria de transmitir meus agradecimentos ao meu orientador Prof. Dr. Ivan Rizzo Guilherme, por ampliar meus horizontes quanto aos diversos conceitos abordados no trabalho, e claro, pela paciência. Também gostaria de agradecer a todos os colegas de trabalho, em especial ao Caio M. Marques, e a tantos professores que contribuíram com seu conhecimento para o enriquecimento do trabalho e do autor como profissional e pessoa, durante toda a graduação.

Não poderia deixar de agradecer à Anne, aquela que escolhi, e pela qual fui escolhido, para compartilhar um pedaço da vida, pela compreensão, companheirismo, paciência, amor, carinho e tantas outras, durante o todo o tempo de graduação e de desenvolvimento deste trabalho.

Aos amigos, meus agradecimentos por compreenderem os momentos de *stress* (e muitas vezes compartilharem deles), mas acima de tudo por compartilhar os bons momentos.

Por fim, agradeço a toda minha família, em especial aos meus pais Marcos e Rosana, pois sem eles, nada do que sou hoje ou o que posso me tornar um dia, seria possível.

Agradeço a todos que colaboraram com este trabalho e com minha formação.

## LISTA DE FIGURAS

<b>FIGURA 1:</b> PARADIGMA "FIND-BIND-EXECUTE" DA SOA.....	13
<b>FIGURA 2:</b> REDE (OU DIAGRAMA) DE IGUAL ÁREA, TAMBÉM CONHECIDA COMO REDE DE SCHIMIDT-LAMBERT. ....	22
<b>FIGURA 3:</b> PRINCIPAIS DISCIPLINAS DO PU. FIGURA EXTRAÍDA DE (LARMAN, 2007).....	25
<b>FIGURA 4:</b> FASES E ITERAÇÕES DO PU. FIGURA EXTRAÍDA DE (LARMAN, 2007).....	25
<b>FIGURA 5:</b> COMPONENTES DA ARQUITETURA DO STRADGE. ....	29
<b>FIGURA 6:</b> COMPONENTE "CONTROLE USUÁRIOS" DA ARQUITETURA. ....	30
<b>FIGURA 7:</b> COMPONENTE "CONTROLE TRATAMENTO" DA ARQUITETURA. ....	31
<b>FIGURA 8:</b> COMPONENTE "CADASTRO DADOS" DA ARQUITETURA. ....	32
<b>FIGURA 9:</b> ARQUITETURA DO GEOSERVER. FIGURA OBTIDA DE (GEOSERVER, 2010).....	34
<b>FIGURA 10:</b> ARQUITETURA DO WPS IMPLEMENTADO PELO DEEGREE. FIGURA OBTIDA DE (DEEGREE, 2010).....	35
<b>FIGURA 11:</b> APLICAÇÃO DA ARQUITETURA PROPOSTA.....	37
<b>FIGURA 12:</b> MODELO DE DADOS DO STRADGE.....	41
<b>FIGURA 13:</b> DIAGRAMA DE CASOS DE USO PARA O STRADGE. ....	43
<b>FIGURA 14:</b> MODELO DE DOMÍNIO PARA O CASO DE USO CDU6. ....	46
<b>FIGURA 15:</b> DIAGRAMA DE SEQUENCIA DO SISTEMA (DSS) PARA O CASO DE USO CDU6. ....	46
<b>FIGURA 16:</b> DIAGRAMA DE SEQÜÊNCIA EXIBINDO A COLABORAÇÃO ENTRE OBJETOS DE SOFTWARE.....	49
<b>FIGURA 17:</b> DIAGRAMA DE CLASSES DO CASO DE USO CDU6.....	50
<b>FIGURA 18:</b> PÁGINA DE CADASTRO DE USUÁRIO.....	55
<b>FIGURA 19:</b> PÁGINA DE LOGIN. ....	55
<b>FIGURA 20:</b> PÁGINA COM FORMULÁRIO PARA IMPORTAÇÃO DE DADOS. ....	57
<b>FIGURA 21:</b> PÁGINA PRINCIPAL, ACESSADA APENAS POR USUÁRIOS AUTENTICADOS, QUE EXIBE OS CONJUNTOS DE DADOS DISPONÍVEIS. ....	58
<b>FIGURA 22:</b> INTERFACE COM MAPA DE UM CONJUNTO DE DADOS.....	59
<b>FIGURA 23:</b> LISTA DE PROCESSOS DISPONÍVEIS PARA O TRATAMENTO DOS DADOS. ....	60
<b>FIGURA 24:</b> EXEMPLO DOS PARÂMETROS DO PROCESSO DE GERAÇÃO DE UM DIAGRAMA DE IGUAL ÁREA.....	61
<b>FIGURA 25:</b> INTERFACE INTERATIVA COM AS OPÇÕES DE ELEMENTOS DE UM DIAGRAMA DE IGUAL ÁREA.....	62

<b>FIGURA 26:</b> A) FALHA NORMAL B) FALHA INVERSA C) FALHA TRANSCORRENTE. FIGURA EXTRAÍDA DE (TEIXEIRA ET AL, 2009). .....	<b>67</b>
<b>FIGURA 27:</b> PROJEÇÃO PARA OBTER UM DIAGRAMA DE IGUAL ÁREA. FIGURA EXTRAÍDA DE (MENDES E KULLBERG, 1992). .....	<b>68</b>
<b>FIGURA 28:</b> FUNÇÃO QUE CALCULA AS COORDENADAS DE UMA CURVA QUE REPRESENTA UM PLANO DE FALHA, DADOS O RUM E O MERGULHO DO PLANO .....	<b>71</b>
<b>FIGURA 29:</b> CURVA NO DIAGRAMA REPRESENTANDO UM PLANO COM RUMO DE 45 GRAUS E MERGULHO DE 30 GRAUS. ....	<b>71</b>
<b>FIGURA 30:</b> PONTO NO DIAGRAMA, REPRESENTANDO O PÓLO DE UM PLANO DE FALHA. ....	<b>72</b>
<b>FIGURA 31:</b> FUNÇÃO QUE CALCULA AS COORDENADAS DE UM PÓLO, DADO O RUMO E MERGULHO DE UM PLANO DE FALHA.....	<b>72</b>
<b>FIGURA 32:</b> ESQUEMA DO PROCEDIMENTO DO MÉTODO DOS DIEDROS RETOS.....	<b>73</b>

## LISTA DE TABELAS

<b>TABELA 1: OPERAÇÕES DO WMS.</b> .....	<b>16</b>
<b>TABELA 2: OPERAÇÕES DO WFS.</b> .....	<b>17</b>
<b>TABELA 3: OPERAÇÕES DO WPS.</b> .....	<b>19</b>
<b>TABELA 4: REQUISITOS DO STRADGE.</b> .....	<b>40</b>
<b>TABELA 5: RELAÇÃO DOS CASOS DE USO IDENTIFICADOS PARA O STRADGE.</b> .....	<b>41</b>
<b>TABELA 6: CASO DE USO CDU6: TRATAR DADOS.</b> .....	<b>44</b>
<b>TABELA 7: CONTRATO DE OPERAÇÃO CO1.</b> .....	<b>48</b>
<b>TABELA 8: CASO DE USO CDU1: CADASTRAR NO SISTEMA.</b> .....	<b>52</b>

## LISTA DE ABREVIACOES E SIGLAS

STRADGE	Sistema de Tratamento de Dados de Geologia
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
REST	Representational State Transfer
HTTP	Hypertext Transfer Protocol
WSDL	Web Service Definition Language
XML	Extensible Markup Language
UDDI	Universal Description, Discovery and Integration
URI	Uniform Resource Identifier
OGC	Open Geospatial Consortium
WMS	Web Map Service
WFS	Web Feature Service
WPS	Web Processing Service
GML	Geography Markup Language
CSV	Comma-Separated Values
JSON	JavaScript Object Notation
W3C	World Wide Web Consortium
JSP	Java Server Pages
HTML	HyperText Markup Language
DOM	Document Object Model
PU	Processo Unificado
API	Application Programming Interface
DSS	Diagrama de Sequencia do Sistema
WAR	Web application ARchive

## SUMÁRIO

LISTA DE FIGURAS	
LISTA DE TABELAS	
LISTA DE ABREVIATURAS E SIGLAS	
SUMÁRIO	

<b>1. INTRODUÇÃO</b>	<b>9</b>
1.1 ORGANIZAÇÃO DO TRABALHO	11
<b>2. ASPECTOS CONCEITUAIS</b>	<b>12</b>
2.1 TECNOLOGIAS UTILIZADAS	12
2.1.1 <i>Arquitetura Orientada a Serviços</i>	12
2.1.1.1 SOAP	14
2.1.1.2 REST	14
2.1.2 <i>Serviços OGC</i>	16
2.1.3 <i>Java</i>	19
2.1.4 <i>HTML5</i>	19
2.1.5 <i>WebGL</i>	20
2.1.6 <i>OpenLayers</i>	20
2.1.7 <i>Projeção Estereográfica</i>	21
2.2 SOFTWARES UTILIZADOS	23
2.3 METODOLOGIA	24
2.3.1 <i>O Processo Unificado</i>	24
<b>3. ARQUITETURA DO STRADGE</b>	<b>28</b>
3.1 ARQUITETURA PROPOSTA	28
3.1.1 <i>Camada de Dados e Processos</i>	29
3.1.2 <i>Camada de Serviços</i>	30
3.1.3 <i>Camada de Aplicação</i>	37
3.1.4 <i>Considerações Parciais</i>	38
<b>4. ANÁLISE, PROJETO E IMPLEMENTAÇÃO DO STRADGE</b>	<b>39</b>
4.1 CONCEPÇÃO	39
4.2 ELABORAÇÃO	45
4.2.1 <i>Iteração 1</i>	45
4.3 CONSTRUÇÃO	51
4.4 TRANSIÇÃO	53
<b>5. O SISTEMA DE TRATAMENTO DE DADOS DE GEOLOGIA</b>	<b>54</b>
5.1 AUTENTICAÇÃO E CADASTRO	54
5.2 MANIPULAÇÃO DE DADOS	56
5.2.1 <i>Importação de Dados</i>	56
5.2.2 <i>Conjuntos de Dados Disponíveis</i>	57
5.2.3 <i>Visualização de Dados</i>	58
5.3 EXECUÇÃO DE UM PROCESSO	59
5.3.1 <i>Seleção de um Processo</i>	60
5.3.2 <i>Parâmetros de um Processo</i>	60
5.3.3 <i>Execução e Resposta de um Processo</i>	61
5.3.4 <i>Salvar Resultados de um Processo</i>	62
<b>6. CONCLUSÕES</b>	<b>64</b>
<b>REFERÊNCIAS</b>	<b>66</b>
<b>APÊNDICE A</b>	<b>68</b>

## 1. INTRODUÇÃO

Atualmente, em um contexto geral, é grande a necessidade de publicar dados coletados por diversos grupos de pesquisa através de interfaces de Sistemas de Informação Geográfica (SIG), que apresentam mapas com informações relacionadas, sendo possível ter acesso a estas informações por meio do próprio mapa. Atualmente existem diversas soluções de sistemas capazes de atender esta necessidade.

Além de apresentar e compartilhar, há também a necessidade do desenvolvimento de sistemas capazes de processar dados, a fim de gerar produtos que auxiliem no estudo de uma determinada área. Também é possível encontrar soluções que atendam a esta necessidade. Entretanto, as soluções atuais, tanto para compartilhamento como para processamento de dados, atuam separadamente, sem considerar a relação dos dados publicados com os processos com eles realizados.

Para o compartilhamento de dados em ambiente SIG, utiliza-se atualmente sistemas desenvolvidos com uma Arquitetura Orientada a Serviços, onde os dados são publicados sob serviços especializados em manipulação de dados geográficos, mantendo informações a eles associadas.

Existe a tendência de que a publicação de dados possa ser relacionada com o processamento dos mesmos, a fim de se desenvolver ambientes mais completos para a realização de estudos e soluções de problemas, como pode ser evidenciado nos trabalhos de Ma et al. (2010) e de Yang et al. (2010).

Neste contexto, recentemente foi especificado também um serviço para padronizar a requisição de processos, utilizando o mesmo padrão dos serviços que manipulam dados geográficos. Tal serviço foi concebido para o processamento de dados geográficos, porém utilizando sua especificação, é possível requisitar qualquer tipo de processo.

Uma das áreas do conhecimento que apresenta cenários com as necessidades apresentadas acima, é a área de Geologia Estrutural. Esta área da Geociência estuda as deformações, a morfologia e os mecanismos de formação de estruturas originadas da crosta terrestre. Estuda também os mecanismos e processos de deformação e de seus produtos.

Dentre as técnicas de estudo da Geologia Estrutural está a Projeção Estereográfica, que possui como principal característica, facilitar o entendimento de situações geológicas complexas no espaço através de diagramas. Porém, mesmo com a simplificação das estruturas a serem representadas, o processo manual para geração de diagramas estereográficos requer diversos passos, tornando a tarefa não trivial. Entretanto, este processo pode ser descrito em algoritmos computacionais, tornando o tratamento de dados estruturais uma tarefa automatizada.

Por tratar de estudos relacionados a espaços geográficos, a necessidade pela publicação de dados em ambientes SIG torna-se evidente; a geração de diagramas a partir determinadas informações para auxiliar no estudo mostra a necessidade pelo processamento de dados.

É dentro deste contexto que este trabalho propõe a concepção de uma Arquitetura Orientada a Serviços, com a finalidade de compartilhar e permitir o processamento de dados de geologia.

Para validar a arquitetura proposta, será implementado o Sistema de Tratamento de Dados de Geologia (STRADGE), sistema no qual serão utilizados os métodos de geração de diagramas de projeção estereográfica.

## **1.1 ORGANIZAÇÃO DO TRABALHO**

Esta seção descreve a organização do trabalho apresentado neste documento.

No Capítulo 2, são apresentados aspectos conceituais relacionados ao trabalho e a metodologia de desenvolvimento utilizada, bem como as tecnologias e linguagens de programação adotadas no desenvolvimento do Sistema de Tratamento de Dados de Geologia (STRADGE).

No Capítulo 3, é apresentada a arquitetura proposta para o STRADGE.

No Capítulo 4, são apresentadas as etapas envolvidas no desenvolvimento do sistema, e no Capítulo 5, o sistema é descrito e suas funcionalidades são apresentadas.

Finalmente, no Capítulo 6, são apresentadas as conclusões obtidas ao final do trabalho.

## 2. ASPECTOS CONCEITUAIS

Nesta seção serão apresentados aspectos conceituais utilizados na concepção, no desenvolvimento e na implementação do STRADGE. As tecnologias e metodologias utilizadas e suas principais características serão apresentadas.

### 2.1 TECNOLOGIAS UTILIZADAS

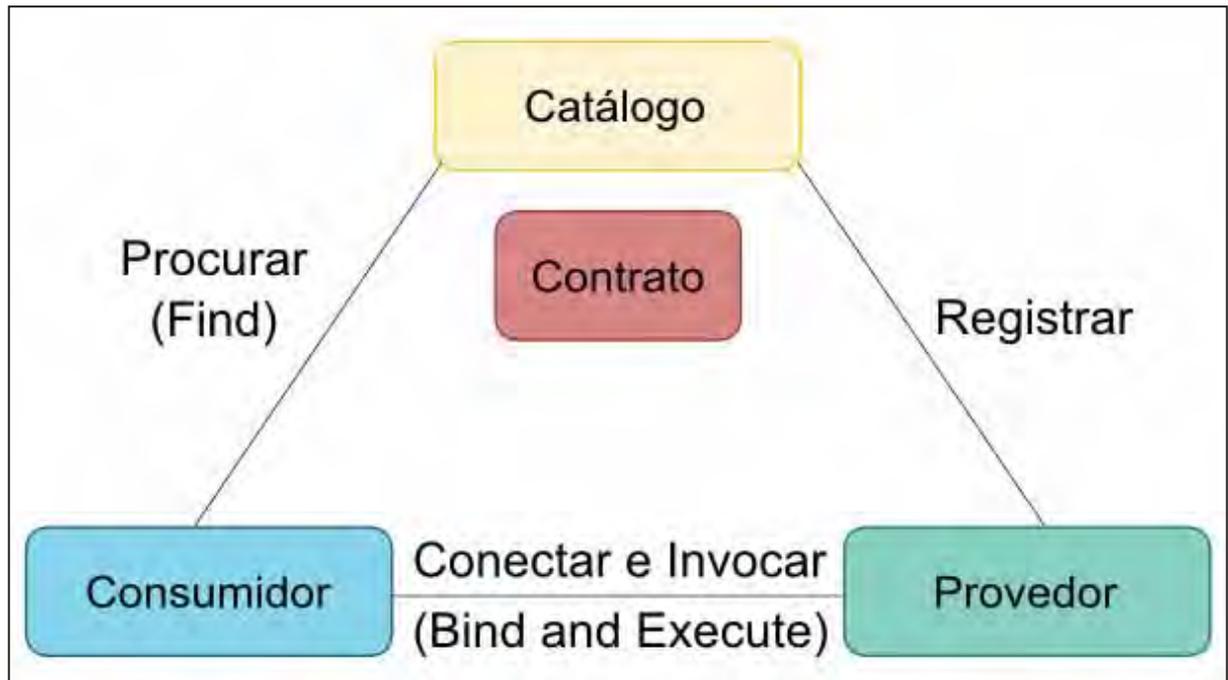
Os itens a seguir apresentam as tecnologias utilizadas na concepção, desenvolvimento e implementação do STRADGE.

As tecnologias apresentadas a seguir foram escolhidas para que o sistema pudesse atender aos requisitos propostos, bem como possibilitar sua expansão.

#### 2.1.1 Arquitetura Orientada a Serviços

A Arquitetura Orientada a Serviços (SOA - Service Oriented Architecture) é um estilo de arquitetura que determina que as funcionalidades de uma aplicação devem ser disponibilizadas em forma de serviços. Um serviço é uma função de um sistema que pode ser acessada por outro sistema e que deve ser auto-contido, isto é, seu estado não deve depender do estado de outros serviços. Um serviço pode ser composto por outros serviços, mas devendo sempre manter a característica de “caixa-preta” para os consumidores, ou seja, o modo como o serviço realiza sua tarefa deve ser invisível ao usuário, sendo que este deve preocupar-se apenas com a entrada e saída de dados (SOA WORK GROUP, 2010).

A SOA é baseada no paradigma conhecido como “*Find-Bind-Execute*”, que pode ser visualizado na **Figura 1**, a fim de que os serviços possam ser publicados, buscados e consumidos por qualquer sistema.



**Figura 1:** Paradigma "Find-Bind-Execute" da SOA.

Analisando a figura acima, identifica-se um consumidor (aplicação), que busca por um serviço que atende a determinados critérios em um catálogo de terceiros. Se o catálogo possui o registro do serviço, ele apresenta ao usuário um contrato, que especifica como o consumidor deverá interagir com o provedor de serviços, e o endereço de acesso ao serviço.

Utilizando o contrato e o endereço encontrados no catálogo, o consumidor pode conectar-se ao provedor de serviços e executar o serviço desejado.

Atualmente há dois padrões de implementação da SOA que são amplamente utilizados: o primeiro, baseado no protocolo SOAP (*Simple Object Access Protocol*) (SOAP, 2010), e o segundo, denominado REST (*Representational State Transfer*) (FIELDING, 2000). Ambos são brevemente descritos a seguir.

### 2.1.1.1 SOAP

Este padrão de implementação utiliza uma série de protocolos e documentos descritivos para padronizar a comunicação e a publicação de Web Services – serviços disponíveis e acessados através de uma conexão HTTP (*Hypertext Transfer Protocol*).

O SOAP é um protocolo baseado no HTTP que possui uma estrutura padrão, baseada em XML, para troca de mensagens entre um consumidor e um serviço. Esta estrutura é composta por três partes: um envelope, que é um elemento raiz que define o XML como uma mensagem SOAP e que contém os *namespaces* e os demais elementos; um cabeçalho, que contém informações para o processamento da mensagem, como por exemplo, autenticação e os tipos de dados contidos na mensagem; por fim, o corpo da estrutura, que contém a mensagem em si (SOAP, 2010).

Além do SOAP, o padrão utiliza o WSDL (*Web Services Description Language*), um documento que descreve um serviço de forma abstrata, como um conjunto de operações de um *endpoint*. Por ser baseado em XML, os documentos WSDL são passíveis de interpretação tanto por humanos quanto por máquinas, sendo portanto possível definir mecanismos de geração de consumidores a partir do próprio documento de descrição do serviço (WSDL, 2010).

O padrão ainda define um protocolo que especifica um método de publicação e descobrimento de diretórios de serviços, conhecido como UDDI (*Universal Description, Discovery and Integration*), também baseado em XML (OASIS, 2003).

### 2.1.1.2 REST

Inicialmente o termo REST foi empregado para descrever uma série de princípios de arquitetura para sistemas hipermídia distribuídos, como a World Wide Web (FIELDING, 2000).

Atualmente o termo é utilizado para descrever qualquer interface simples que utilize mensagens em formato texto, estruturadas em XML ou não, e o protocolo HTTP para comunicação, sem utilizar uma camada de abstração sobre o protocolo, como é o caso do SOAP.

O padrão REST estabelece uma série de conceitos e princípios. O principal conceito é o de Recursos (*Resources*), que são os elementos de informações acessíveis pelo protocolo HTTP. Pode-se entender um recurso como um serviço disponível em um servidor, bem como um conjunto de dados disponíveis através de um serviço.

Os princípios estabelecidos pelo REST são:

- Um protocolo cliente/servidor sem estados (*stateless*), ou seja, cada mensagem contém informação suficiente para o entendimento do pedido, não sendo necessário que nem o cliente, nem o servidor armazenem o estado de comunicações anteriores.
- Um conjunto de operações que se apliquem a todos os recursos de informações. Dentre as principais operações do HTTP, estão: POST, GET, PUT e DELETE. A operação POST envia dados junto a mensagem, para que estes sejam processados. A operação GET solicita um dado recurso, e as operações PUT e DELETE interagem sobre o recurso em si, adicionando ou removendo, respectivamente, um recurso do servidor.
- A identificação de cada recurso de forma única, através de um URI (Uniform Resource Identifier).
- A representação de estados dos recursos através de HTML ou XML, permitindo com que seja possível navegar de um recurso para outro através de ligações.

Os sistemas que seguem o padrão REST, como o proposto neste trabalho, são denominados *RESTful*. O termo *RESTful Web Services* designa serviços web que são implementados, disponibilizados e que devem ser consumidos seguindo os princípios do REST.

### 2.1.2 Serviços OGC

O sistema desenvolvido neste trabalho utiliza o padrão de implementação *RESTful* padronizados pelo OGC (*Open Geospatial Consortium*), uma entidade internacional, sem fins lucrativos que busca definir padrões para serviços baseados em geolocalização (OGC, 2010). Dentre os diversos serviços descritos por documentos de especificação do OGC, são utilizados neste trabalho apenas o WMS (*Web Map Service*), o WFS (*Web Feature Service*) e o WPS (*Web Processing Service*). Estes serviços são descritos a seguir:

#### a. WMS

O WMS, segundo o *Open Geospatial Consortium* (2010), fornece uma interface HTTP simples para requisição de imagens de mapas geo-referenciadas de uma ou mais base de dados distribuídas. Uma requisição WMS define uma camada geográfica e uma área de interesse para ser processada. Uma resposta do WMS é uma ou mais imagens de mapas geo-referenciadas. O WMS suporta diversos formatos de imagens para retornar, e também a transparência destas imagens, para que imagens de diferentes camadas possam ser sobrepostas (WMS, 2010). As operações definidas no padrão WMS são apresentadas na **Tabela 1**.

**Tabela 1:** Operações do WMS.

OPERAÇÃO	DESCRIÇÃO
GetCapabilities	Retorna uma lista de informações do servidor, bem como as operações e parâmetros válidos do WMS.
GetMap	Retorna a imagem requisitada pelo cliente.
GetFeatureInfo	Retorna os dados, incluindo a geometria e valores de atributos, para a localização de um pixel.
DescribeLayer	Indica o WFS ( <i>Web Feature Service</i> ) ou WCS ( <i>Web Coverage Service</i> ) para recuperar informação adicional sobre a camada.
GetLegendGraphic	Mecanismo geral para retornar símbolos de legendas.

## b. WFS

De acordo com o Open Geospatial Consortium (2010), o WFS provê uma interface simples para requisição, neste caso, de dados geo-espaciais através do HTTP. O WFS codifica e transmite informação no formato GML (*Geography Markup Language*), um padrão baseado em XML para representar dados geo-referenciados (GML, 2010).

Ao contrário do WMS que processa os dados em formato de imagem, o WFS não realiza nenhum tipo de processamento sobre os dados requisitados, apenas formata-os para que sejam trabalhados pelo cliente que os requisitou. Além do GML, dependendo da implementação, o WFS pode fornecer os dados em diversos formatos, sendo os mais comuns arquivos do tipo *shapefile*, utilizados por diversos softwares que trabalham com informações geo-referenciadas, CSV (*Comma Separated Values*) e JSON (*JavaScript Object Notation*) (WFS, 2010). As operações definidas no padrão WFS são apresentadas na

**Tabela 2.****Tabela 2:** Operações do WFS.

OPERAÇÃO	DESCRIÇÃO
GetCapabilities	Retorna uma lista de informações do servidor, bem como as operações e parâmetros válidos do WFS.
DescribeFeatureType	Retorna informação e atributos sobre um conjunto de dados particular.
GetFeature	Retorna os dados, incluindo a geometria e valores de atributos.
LockFeature	Evita que uma <i>featuretype</i> seja editada.
Transaction	Edita <i>featuretypes</i> existentes, criando, atualizando e excluindo.
GetGMLObject	Retorna elementos através do <i>gml:id</i> , possibilitando que o WFS seja requisitado através de um XLink. (Apenas a partir da versão 1.1.0).

As operações LockFeature e Transaction permitem com que requisições manipulem os dados do servidor. A operação GetGMLObject fornece uma interface para que o WFS possa ser requisitado através de um XLink, uma especificação do W3C (World Wide Web

Consortium) para criar links internos e externos dentro de documentos XML, e associar metadados a estes links.

### c. WPS

Por definição da Open Geospatial Consortium (2010), o WPS define regras para padronizar entradas e saídas de serviços de processamento de dados geoespaciais, como por exemplo, sobreposição de polígonos. Também define como um cliente por requisitar a execução de um processo (WPS, 2010).

Os padrões definidos pelo WPS baseiam-se em XML, e podem ser utilizadas pelo padrão SOAP, junto ao WSDL. Utilizando estes padrões o WPS facilita a publicação e descobrimento de serviços de processamento geoespaciais, além da criação de *workflows* com vários processos.

Os dados para processamento podem estar disponíveis no mesmo servidor que o WPS, ou serem acessados através da *Internet*.

Uma característica importante é que, apesar de sua criação ter como finalidade atender processamento de dados geoespaciais, ele permite a publicação de qualquer tipo de serviço. Outro ponto importante é o que WPS oferece suporte a vários formatos de entradas e saídas de dados. Os parâmetros tanto de entrada como de saída podem ser valores literais, como números e strings; um XML diretamente especificado no documento de requisição ou referenciado através de um XLink; e também dados em formato binário, como imagens.

No STRADGE, o WPS é utilizado para tratar dados que não possuem georeferenciamento, mas que estão associados a outros dados que possuem coordenadas geográficas, e a utilização do padrão permite maior integração entre os dados. As operações definidas no padrão WPS são apresentadas na **Tabela 3**.

**Tabela 3:** Operações do WPS.

OPERAÇÃO	DESCRIÇÃO
GetCapabilities	Retorna uma lista de metadados do WPS.
DescribeProcess	Retorna uma descrição de um ou mais processos, que inclui as entradas, saídas e formatos utilizados e pode ser utilizada para geração automática de uma interface para obtenção dos valores dos parâmetros.
Execute	Operação que permite um cliente executar um processo utilizando os parâmetros de entrada e receber as saídas do processo.

### 2.1.3 Java

A linguagem de programação orientada a objetos Java (DEITEL, 2005), foi utilizada em larga escala no desenvolvimento do sistema, tanto na implementação dos serviços, como em partes da aplicação, utilizando as tecnologias JSP (Java Server Pages) e Servlets, que utilizam Java para a criação de conteúdo web dinâmico, independentes de servidores e plataformas.

### 2.1.4 HTML5

A linguagem HTML (*HyperText Markup Language*), em sua quinta versão, trouxe diversos avanços no que diz respeito a interação com áudio e vídeo, e gráficos 2D e 3D à principal linguagem da web. Segundo a W3C (World Wide Web Consortium), nesta versão, novas funcionalidades foram introduzidas para auxiliar desenvolvedores de aplicações web, especialmente na definição de critérios para agentes em um esforço de melhorar a interoperabilidade (W3C, 2010).

O HTML5 foi utilizado na criação da interface da aplicação cliente do STRADGE, para que as novas funcionalidades da versão pudessem ser exploradas, e uma interface leve e compatível com a grande maioria dos *browsers* atualizados pudesse ser desenvolvida, além de garantir o suporte ao WebGL.

### 2.1.5 WebGL

O WebGL é uma API gráfica 3D, baseada no OpenGL ES 2.0, implementada em um *browser* sem o uso de *plugins*, através do elemento *Canvas* do HTML5 como uma interface DOM (Document Object Model).

O WebGL é utilizado através da linguagem Javascript, e até o presente momento foi implementado apenas nos navegadores Mozilla Firefox, Google Chrome, Safari e Opera, todos os navegadores os quais as empresas desenvolvedoras fazem parte do *WebGL Working Group* (KHRONOS GROUP, 2010).

No STRADGE, o WebGL foi utilizado para possibilitar maior interação do usuário com os resultados dos diagramas, e além disso, foi possível analisar que, por se tratar de uma implementação muito semelhante ao OpenGL, o WebGL pode ser utilizado para diversos fins, como a geração de vários modelos gráficos com interação, e isto, dentro de um navegador web, sem a necessidade de instalar plugins.

### 2.1.6 OpenLayers

O OpenLayers é uma biblioteca livre e *open source* Javascript, que permite, facilmente, adicionar um mapa dinâmico a um navegador (OPENLAYERS, 2010), fornecendo funções para trabalhar com a renderização de dados geográficos e requisições aos serviços do padrão OGC. Foi utilizada neste projeto para realizar requisições e tratar a apresentação de resultados dos serviços WMS e WFS.

### 2.1.7 Projeção Estereográfica

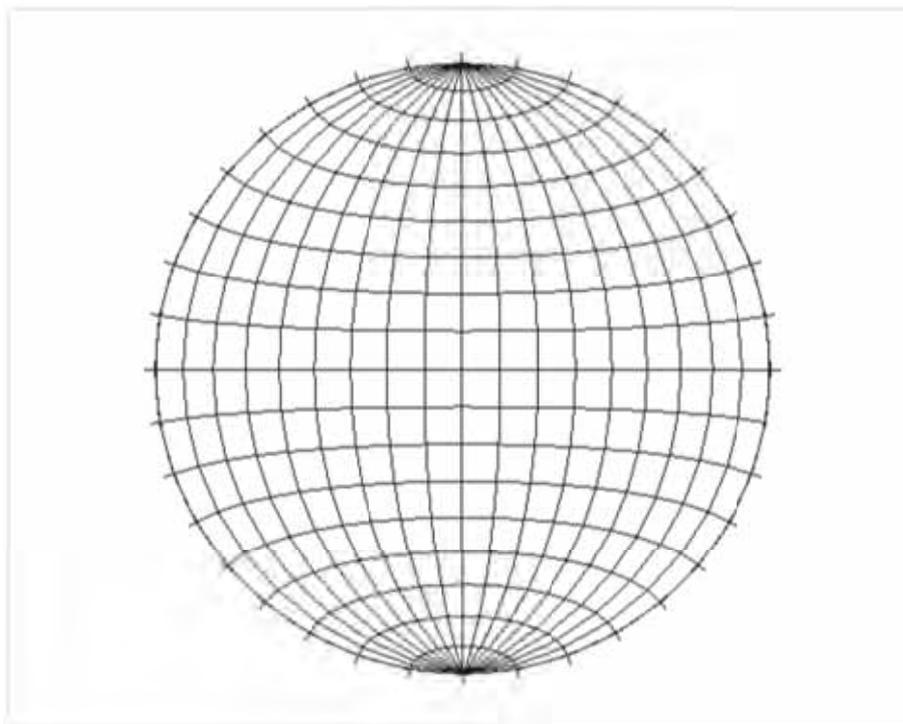
A projeção estereográfica tem como principal característica facilitar o entendimento de situações geológicas complexas no espaço através da representação bidimensional de feições planares e/ou lineares, e possibilita ainda a realização de inúmeras operações, indispensáveis em estudos de Geologia Estrutural, Cristalografia e Ciência dos Materiais, entre outras áreas da Geociência (CARNEIRO, 1996).

Dentro do contexto de Geologia Estrutural, um diagrama de projeção estereográfica pode ser utilizado para diversas finalidades, entre as quais pode-se destacar a obtenção de uma direção preferencial de movimentação de uma estrutura; e também a obtenção das direções das forças de deformação que atuaram sobre uma estrutura, alterando sua forma. Conseqüentemente, para a obtenção destes resultados destacados no parágrafo anterior, é preciso utilizar um tipo de diagrama conhecido como Rede (ou Diagrama) de Igual Área, ou Rede (ou Diagrama) de Schmidt-Lambert (CARNEIRO, 1996).

O diagrama consiste em um círculo dividido em áreas iguais através de curvas, que formam a rede de fato. A rede geralmente é representada com curvas a cada dois graus, mas este número pode variar, geralmente para curvas a cada dez graus. O perímetro do círculo representa um plano horizontal arbitrário utilizado para medir o ângulo do rumo do mergulho, ou direção do plano de falha, variando portanto de 0 (zero) a 360 (trezentos e sessenta) graus.

Por ser uma visão da face superior de uma semi-esfera, o mergulho do plano passa a ser considerado da borda do círculo até o centro, sendo a borda equivalente a 0 (zero), equivalente a um plano horizontal, e o centro equivalente a 90 (noventa) graus, equivalente a um plano vertical.

Na **Figura 2** é apresentado um exemplo de uma Rede de Igual Área e na **Figura 27** do Apêndice A é apresentada a projeção de uma esfera sobre um plano, de forma a obter o diagrama.



**Figura 2:** Rede (ou Diagrama) de Igual Área, também conhecida como Rede de Schimidt-Lambert.

Este diagrama permite a representação de falhas geológicas presentes em determinada estrutura através de curvas (representando os planos de falha) e/ou pontos, também chamados de pólos (representando retas normais ao plano de falha). Exemplos destas curvas e pólos podem são apresentados na **Figura 29** e na **Figura 30** do Apêndice A, onde pode-se encontrar também os algoritmos utilizados no sistema para sua obtenção, apresentados na **Figura 28** e na **Figura 31**, obtidas no site de Steven Dutch, da Universidade de Wisconsin (DUTCH, 1999).

Utilizando-se a representação dos pólos de planos de falha, é possível determinar uma orientação preferencial do movimento ocorrido com uma estrutura analisada.

A partir da representação de curvas (planos), foi desenvolvida uma técnica para representar e calcular as direções dos esforços (forças) principais atuantes nas estruturas, que causaram deformações e foram responsáveis pela geração de falhas.

Esta técnica é denominada Método dos Diedros Retos, e está detalhada no item A2.2 do Apêndice A, mas resumidamente utiliza uma série de sobreposições de áreas determinada por curvas para obter os resultados descritos acima.

A definição dos termos geológicos envolvidos com a projeção estereográfica podem ser encontrados no item A1 do Apêndice A.

## **2.2 SOFTWARES UTILIZADOS**

Nesta seção são apresentados os softwares utilizados no processo de desenvolvimento e na implementação do STRADGE.

Os softwares foram escolhidos de acordo com as necessidades dos requisitos do projeto e também de acordo com as funcionalidades disponibilizadas por cada software.

A seguir, a relação de softwares utilizados no projeto:

- i. Apache Tomcat 6.0.29: Servidor de aplicações Java utilizado para disponibilizar tanto o sistema desenvolvido, quanto os Web Services utilizados para o estudo de caso.
- ii. Eclipse – Helios Release: IDE (Integrated Development Environment) utilizado para a implementação do sistema em Java e compilação do WPS do Deegree.
- iii. Geoserver : Utilizado para publicação dos dados através de serviços WMS e WFS.
- iv. Deegree 3: Utilizado para publicação de processos através do WPS.
- v. PostgreSQL 8.4 com extensão PostGIS: SGBD (Sistema Gerenciador de Banco de Dados) utilizado para armazenar os dados do sistema. A extensão PostGIS implementa tipos e funções para suporte à dados geográficos.
- vi. Firefox 4 (beta): Navegador com maior suporte às tecnologias HTML5 e WebGL, utilizado para testes da implementação do sistema.

## 2.3 METODOLOGIA

A seguir, é apresentada a metodologia na qual foi baseada o processo de desenvolvimento do sistema.

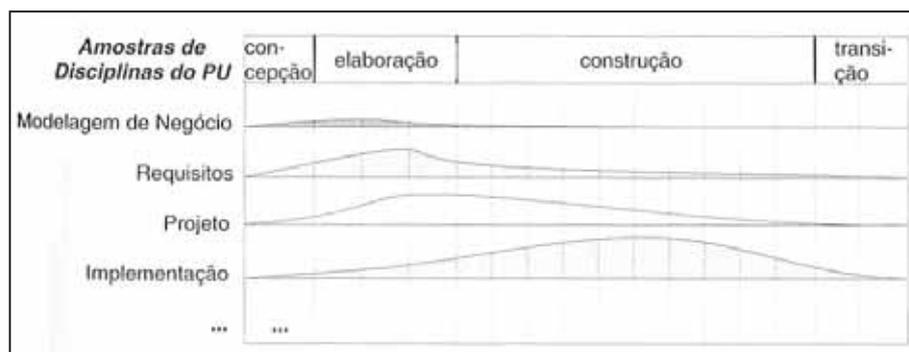
É importante salientar que o processo apresentado não será utilizado em sua plenitude, devido ao contexto de desenvolvimento e implementação do trabalho, feito por um único membro. Entretanto as principais características e boas práticas do Processo Unificado foram empregadas no processo de desenvolvimento do STRADGE.

### 2.3.1 O Processo Unificado

Um processo de desenvolvimento de software consiste em uma série de regras, padrões e seqüência de atividades que buscam garantir a qualidade do produto final e o cumprimento do contrato de desenvolvimento de software. O Processo Unificado (PU) surgiu como um processo iterativo, ágil e evolutivo, para desenvolvimento de sistemas orientados a objetos (LARMAN, 2007).

O PU divide as seqüências de atividades estabelecidas como disciplinas, com cada uma caracterizando um conjunto de artefatos (documentos, código, gráfico, entre outros).

As disciplinas mais importantes estabelecidas no PU são apresentadas na **Figura 3**.

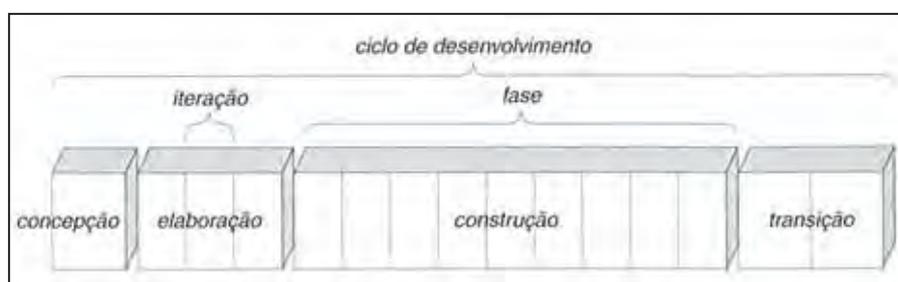


**Figura 3:** Principais Disciplinas do PU. Figura extraída de (LARMAN, 2007).

As disciplinas apresentam as seguintes definições:

- Modelagem de negócio: criação do artefato Modelo do Domínio, para visualizar conceitos importantes.
- Requisitos: criação de Casos de Uso e Especificação Suplementar, para captar requisitos.
- Projeto: criação do Modelo de Projeto para projetar objetos de *software*.
- Implementação: refere-se à programação e construção do sistema.

O ciclo de desenvolvimento do PU é dividido em quatro etapas: Concepção, Elaboração, Construção e Transição. A **Figura 4** apresenta as quatro fases com algumas iterações, e em seguida, estas fases são descritas brevemente.



**Figura 4:** Fases e iterações do PU. Figura extraída de (LARMAN, 2007).

### *Concepção*

A fase de concepção é uma etapa inicial curta na qual apenas alguns artefatos são produzidos. Artefatos estes que estão relacionados à disciplina de Modelagem do Negócio.

Nesta fase, apenas uma listagem dos requisitos deve ser elaborada, assim como um breve levantamento dos casos de uso, a fim de avaliar a viabilidade do projeto, definição de custos, identificação de riscos, entre outras observações relacionadas (LARMAN, 2007).

### *Elaboração*

A Elaboração é a fase que compreende o detalhamento e descoberta dos requisitos e casos de uso identificados na primeira fase.

É nesta fase também em que se define a arquitetura central do sistema, definindo artefatos como modelos de domínio, diagramas de sequência e classe.

Recomenda-se que nesta etapa também se inicie alguns testes de implementação da parte central e crítica da arquitetura para mitigar os riscos (LARMAN, 2007).

### *Construção*

Na fase de Construção, as partes centrais e críticas da arquitetura são finalizadas, e os elementos restantes, de menor risco, são implementados. Ao final desta fase, inicia-se a preparação para a fase de Transição (LARMAN, 2007).

### *Transição*

A última etapa do ciclo de desenvolvimento envolve testes finais e implantação do sistema desenvolvido (LARMAN, 2007).

Durante o desenvolvimento do trabalho proposto, foi detectado que, devido a certas características do sistema orientado a serviços, é preciso fazer certas adaptações à documentação sugerida pelo processo de desenvolvimento orientado a objetos. Ainda sim, foi possível modelar e documentar o sistema de forma adequada.

### 3. ARQUITETURA DO STRADGE

Nesta seção será apresentada a arquitetura proposta para o STRADGE. A arquitetura foi concebida a partir da necessidade do grupo de pesquisa do Laboratório de Geomodelagem, localizado no Departamento de Petrologia e Metalogenia (DPM) do curso de Geologia da UNESP de Rio Claro, por um sistema capaz de gerar os diagramas de projeção estereográfica.

Analisando um conjunto de dados pertencente a este grupo de pesquisa, e as finalidades dos resultados obtidos através das técnicas de projeção estereográfica, foram levantados requisitos gerais para a definição da arquitetura.

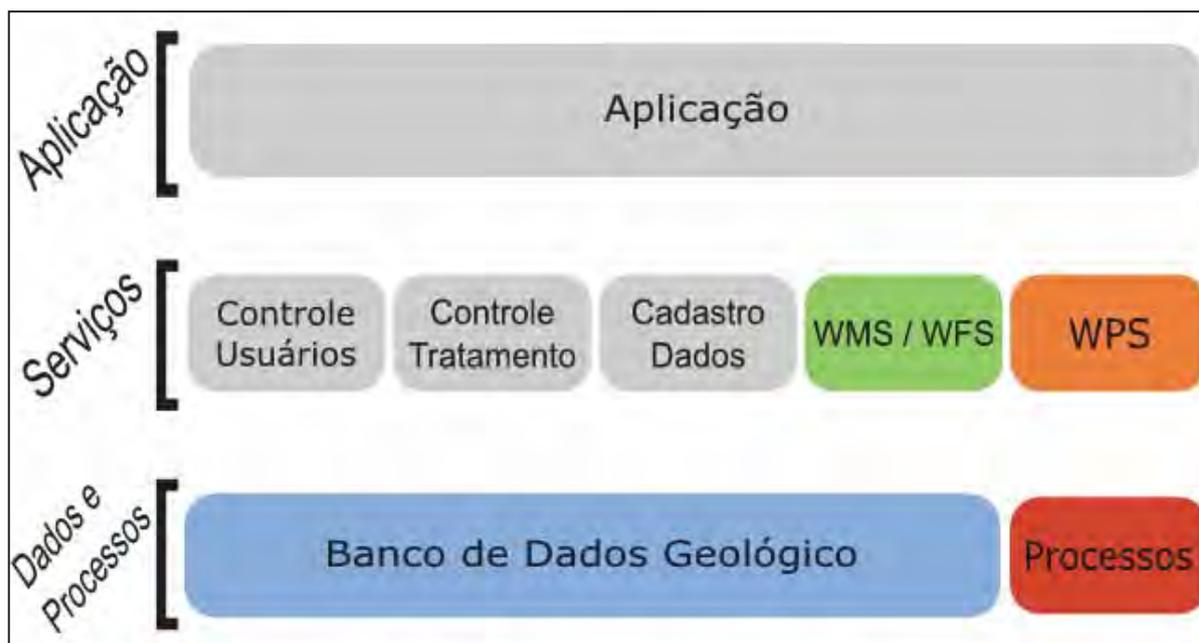
Os requisitos levados em consideração para desenvolver a arquitetura foram possibilitar a integração dos dados com os processos de geração dos diagramas e ainda a utilização dos resultados em um mesmo ambiente.

Além disso, o compartilhamento de informação entre usuários e a possibilidade de adaptar e expandir esta arquitetura também foram considerados, levando então a utilização de serviços.

#### 3.1 ARQUITETURA PROPOSTA

A Arquitetura do STRADGE tem como principal base a utilização de serviços RESTful (descritos no item 2.1.1) a fim de garantir uma camada de isolamento do conjunto de dados, para que estes possam ser recuperados e processados de acordo com as necessidades da camada de aplicação, sem que haja dependência ou acesso direto a camada de dados.

A **Figura 5** apresenta os componentes da arquitetura. Em seguida, uma descrição de cada componente é apresentada.



**Figura 5:** Componentes da arquitetura do STRADGE.

### 3.1.1 Camada de Dados e Processos

#### a) Banco de Dados Geológico

O Banco de Dados Geológico é implementado utilizando o SGBD (Sistema de Gerenciamento de Banco de Dados) PostgreSQL (POSTGRESQL, 2010) com sua extensão de suporte a objetos geográficos PostGIS (POSTGIS, 2010). O banco de dados geológico é utilizado para armazenar os dados das estruturas geológicas, as informações dos usuários, e o resultado dos processamentos.

Como pode ser observado na **Figura 5**, a camada de armazenamento de dados encontra-se totalmente isolada da camada de aplicação por meio dos componentes que implementam os serviços web REST.

## b) Processos

Os processos e seus arquivos de descrição são armazenados em um repositório contido na implementação do WPS, portanto, para adicionar, remover ou alterar um processo, é preciso re-compilar o WPS e implantá-lo novamente no servidor. O componente do WPS responsável por acessar este repositório é o chamado Process Provider.

### 3.1.2 Camada de Serviços

#### a) Controle de Usuários



**Figura 6:** Componente "Controle Usuários" da Arquitetura.

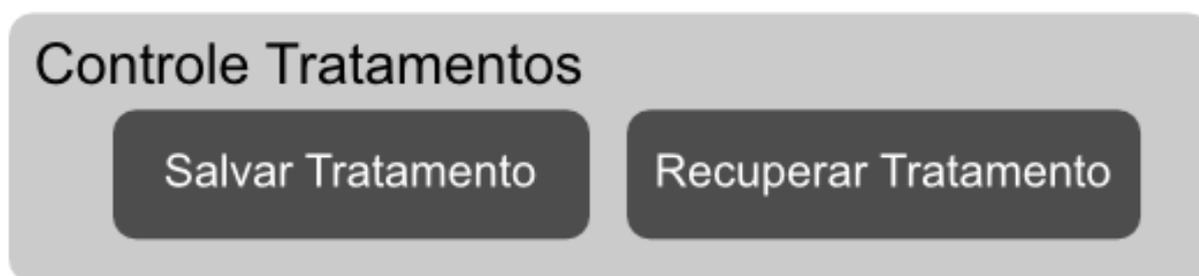
O componente de Controle de Usuários (**Figura 6**), é composto por serviços REST responsáveis por acessar e cadastrar informações de usuários no Banco de Dados Geológico.

Os serviços REST implementados no componente são responsáveis pelo cadastro de um novo usuário; por verificar se um nome de login fornecido durante o cadastro já foi utilizado ou não; pela verificação das credenciais de login e senha durante processos de autenticação; e também por verificar a propriedade de um dado conjunto de informações publicadas.

O serviço responsável por verificar credenciais de autenticação, apenas valida se um dado nome de usuário e senha são os mesmos cadastrados no Banco de Dados, e a camada Aplicação deve se responsabilizar pelo controle de sessão de cada usuário, de acordo com a validação do serviço, permitindo ou não o acesso a determinadas páginas.

Além de inserir um usuário no Banco de Dados Geológico, o serviço responsável pelo cadastro de novos usuários também cria um usuário equivalente no GeoServer, para que este possa utilizar as mesmas credenciais para controlar as permissões das publicações dos conjuntos de dados.

#### b) Controle de Tratamento

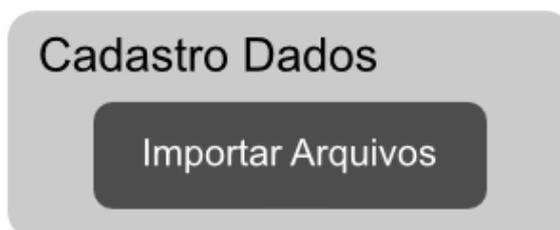


**Figura 7:** Componente "Controle Tratamento" da Arquitetura.

Este componente, ilustrado na **Figura 7**, implementa serviços que são responsáveis por salvar o resultado de tratamentos de dados realizados pelos usuários através do WPS e visualizados na interface composta pelo HTML5 e o WebGL, além de recuperar um determinado tratamento já executado para exibir ao usuário.

Há uma tabela específica no banco de dados para armazenar metadados do tratamento realizado, como qual conjunto de dados foi utilizado, quem realizou o tratamento e ainda qual o caminho para o diretório onde estarão armazenados os resultados em si. Optou-se por utilizar um esquema de diretório para garantir que qualquer tipo de resultado de processo possa ser armazenado, como por exemplo, imagens e ou seqüências de documentos de diversos formatos.

## c) Cadastro de Dados



**Figura 8:** Componente "Cadastro Dados" da Arquitetura.

O componente de cadastro (**Figura 8**) possui o papel fundamental de cadastrar informações das estruturas geológicas no Banco de Dados Geológico e automaticamente publicá-las no GeoServer (GEOSERVER, 2010), o servidor de dados geográficos que implementa, entre outros serviços, o WMS e o WFS.

Cada usuário pode cadastrar um número de informações indeterminado, desde que determine um nome único no sistema para identificar um conjunto de informações que serão inseridas no Banco de Dados Geológico.

Para cada conjunto de informações, o usuário pode definir se os dados são públicos, ou seja, podem ser visualizados e acessados por qualquer usuário autenticado no sistema, ou se os dados são privados, podendo ser visualizados e acessados somente por ele.

Assim que o usuário realiza o *upload* do arquivo que contém o conjunto de dados, o serviço de importação realiza a leitura do arquivo, armazenando as informações no Banco de Dados, realizando as verificações necessárias.

Em seguida, o serviço interage com o Banco de Dados Geológico transformando as coordenadas geográficas do conjunto para o formato interpretado pela extensão PostGIS. Além de gerar uma View para o conjunto, com a união das estruturas com suas falhas.

Uma View pode ser considerada como uma tabela virtual em um banco de dados, formada através de uma consulta definida pelo usuário, operando sobre colunas de tabelas reais. Geralmente as Views são utilizadas para deixar acessíveis junções de colunas sob um nome de tabela e otimizar consultas, fazendo com que um usuário ou aplicação não tenha que realizar a operação de junção a cada acesso ao banco de dados.

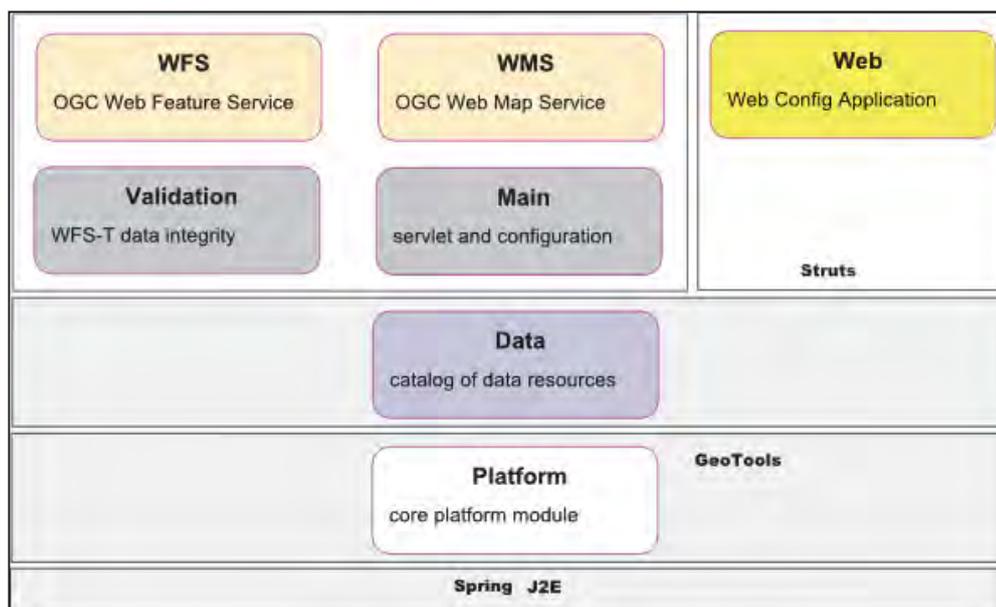
As Views criadas no Banco de Dados Geológico unem determinadas colunas da tabela de estruturas geológicas com suas falhas.

Apesar de uma estrutura poder ter uma ou mais falhas, gerando na View uma tabela com entradas repetidas para as colunas da tabela de estruturas, esta junção é necessária para a publicação no GeoServer, que nativamente não oferece suporte para os níveis de relacionamento de tabelas em um banco de dados.

Uma vez gerada a View de um conjunto, o serviço utiliza uma API REST do GeoServer para publicar, através de requisições a serviços REST do GeoServer, esta View e definir as permissões de usuários sobre a FeatureType (nome que o GeoServer atribui a uma Layer que representa as coordenadas de acesso a um conjunto de dados num repositório, sendo este, neste caso, um banco de dados espacial).

Finalizado o processo de cadastro e publicação, o conjunto de dados já está disponível para ser acessado através dos serviços WMS e WFS do GeoServer.

## d) WMS e WFS



**Figura 9:** Arquitetura do GeoServer. Figura obtida de (GEOSERVER, 2010).

O componente que implementa os serviços WMS e WFS, entre outros, é composto pelo GeoServer, como descrito anteriormente. A arquitetura deste servidor, ilustrada pela comunidade de desenvolvimento, pode ser vista na **Figura 9**.

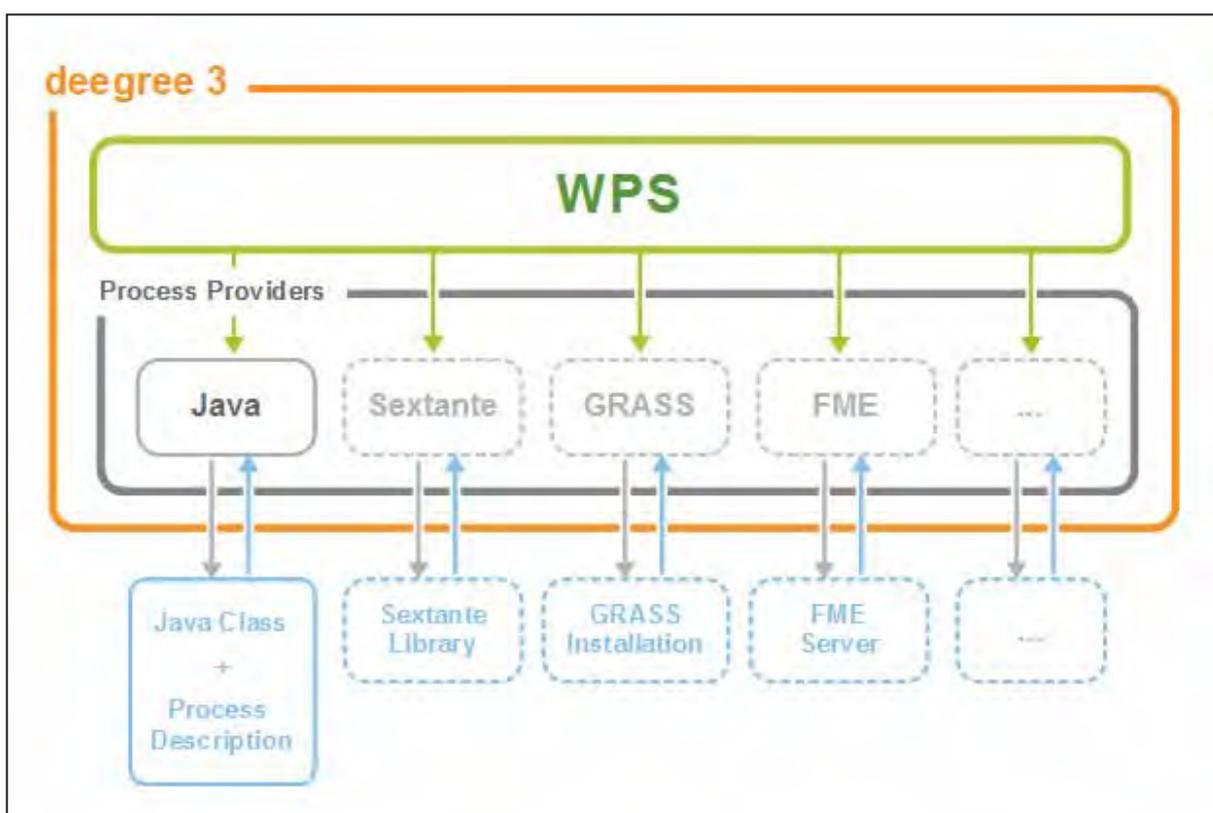
O GeoServer publica dados de qualquer fonte de dados geográfico utilizando padrões abertos (GEOSERVER, 2010), mais especificamente, os serviços padronizados pela OGC.

Na arquitetura em questão, a Aplicação realiza requisições aos serviços disponíveis no GeoServer, selecionando através de parâmetros quais os conjuntos de dados (FeatureTypes) deseja acessar e quais os formatos de saída. Também é possível definir filtros para recuperar apenas uma determinada seleção dentro de um conjunto.

A Aplicação utiliza o WMS do GeoServer para exibir um mapa contendo as estruturas cadastradas no Banco de Dados Geológico, dentro de um determinado conjunto de dados. Além disso, também é utilizado para exibir as informações a respeito de uma determinada estrutura selecionada no mapa, pelo usuário.

O WFS é utilizado para passar os dados como parâmetro de entrada para o WPS. E tanto a consulta ao WFS como ao WMS possuem várias opções de formatos de saída. Na Aplicação, o formato requisitado ao WFS é o GML3 (Geography Markup Language) (GML, 2010), um padrão XML para representar dados geográficos, que é utilizado como parâmetro de entrada para as requisições ao WPS.

e) WPS



**Figura 10:** Arquitetura do WPS implementado pelo Deegree. Figura obtida de (DEEGREE, 2010).

O componente de implementação do WPS é composto pelo Deegree (DEEGREE, 2010), um servidor semelhante ao GeoServer, mas que possui uma implementação do serviço em questão um pouco mais avançada, incluindo também uma documentação mais extensa sobre sua utilização. Apesar de já ser possível utilizar uma versão do WPS no GeoServer, a implementação do WPS ainda está em estágio inicial.

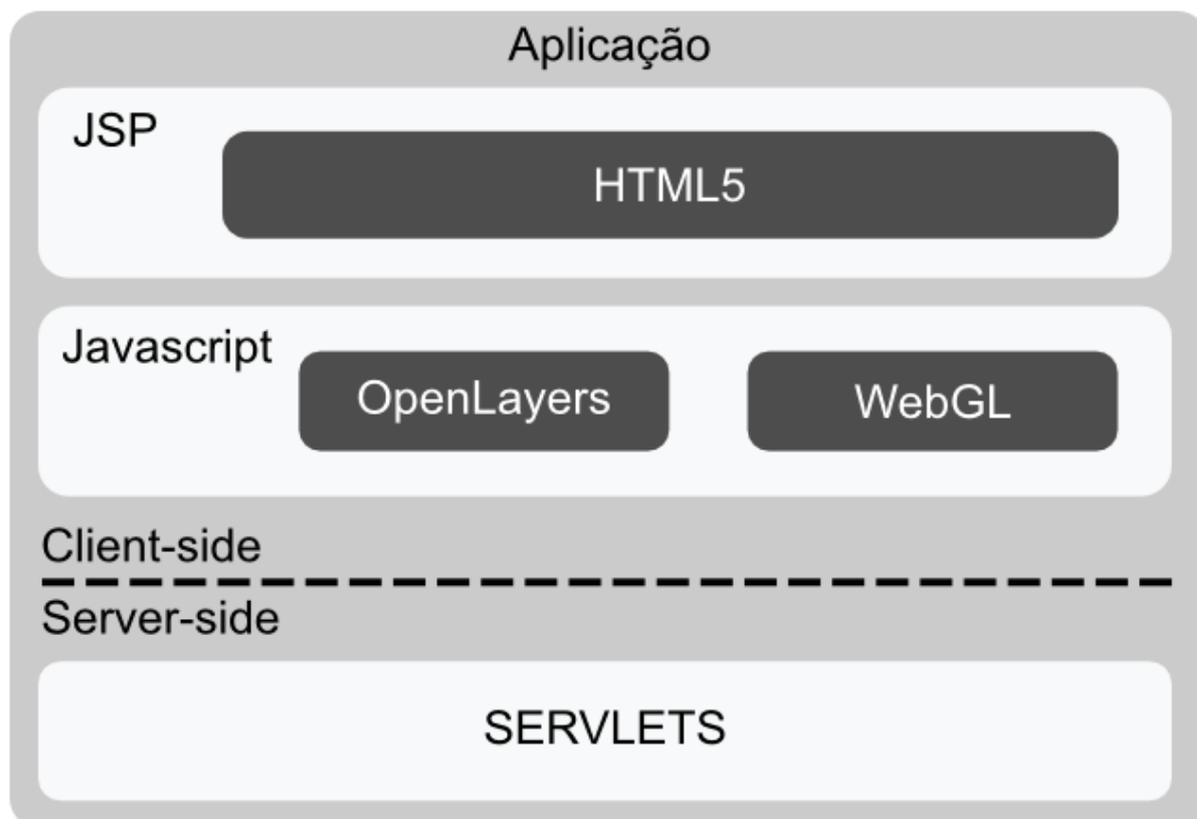
Neste componente, ilustrado pela comunidade de desenvolvimento na **Figura 10**, encontram-se os processos disponíveis sob o padrão WPS que podem ser acessados pela Aplicação, incluindo aqueles que trabalham sobre os dados de falhas das estruturas, para gerar os diagramas de Projeção Estereográfica.

Como citado anteriormente, o WPS pode receber diversos formatos de entrada, e neste caso, para os processos de geração de diagramas, os processos recebem um valor literal para definir o raio do diagrama a ser gerado e o resultado de uma consulta ao WFS do GeoServer no formato GML3, contendo os dados de uma determinada estrutura.

Uma vez executado o processo, o retorno para a Aplicação dá-se em formato XML, contendo as coordenadas dos elementos do diagrama, para que este possa ser exibido através de uma interface composta pelo HTML5 e o WebGL.

### 3.1.3 Camada de Aplicação

a) Aplicação



**Figura 11:** Aplicação da Arquitetura proposta.

Como é possível notar na descrição dos demais componentes, a Aplicação, ilustrada na **Figura 11**, é responsável por organizar o fluxo de requisições aos serviços e exibição de resultados.

Na implementação concretizada no desenvolvimento deste trabalho, a Aplicação é uma interface Web, desenvolvida principalmente com as tecnologias JSP e Servlet, que realiza um controle de sessão de usuários, garantindo verificação de permissão de acesso às páginas.

A interface Web também utiliza a tecnologia HTML5 para exibir as páginas no navegador, o OpenLayers para disponibilizar os mapas de cada conjunto de dados e

permitir a interação do usuário com os mesmos, e o WebGL, como citado anteriormente, para exibir os resultados dos processos de geração de diagramas de Projeção Estereográfica.

#### **3.1.4 Considerações Parciais**

Esta arquitetura garante a capacidade do usuário de processar dados de geologia estrutural e ainda compartilhar não só os dados originais, como também os resultados dos tratamentos aplicados, sendo possível que cada usuário defina a permissão de acesso para cada conjunto original ou de resultado de tratamento.

Pelo fato da Aplicação ter sido desenvolvida como uma interface Web, garante também a independência de plataforma.

Os serviços RESTful possibilitam, através da independência entre os dados e a aplicação, que qualquer tipo de aplicação possa realizar o workflow necessário para se obter os resultados desejados. E além disso, modificando as requisições e seqüências dos serviços chamados, é possível dar outras características e funcionalidades à aplicação.

## **4. ANÁLISE, PROJETO E IMPLEMENTAÇÃO DO STRADGE**

No processo de desenvolvimento do STRADGE foi utilizado o Processo Unificado, apresentado na subseção 2.3.1, assim para melhor entendimento das etapas do desenvolvimento, a estrutura desta seção está de acordo com as fases do ciclo de desenvolvimento do PU.

Considerando a complexidade do sistema e o objetivo de validar a proposta de atender às necessidades de processamento e compartilhamento de dados, apenas as partes mais relevantes para sua validação, que dizem respeito à manipulação de dados e execução de processos, foram desenvolvidas, e portanto, apenas os documentos de análise, projeto e implementação destas partes do sistema serão apresentados.

### **4.1 CONCEPÇÃO**

Na fase de concepção, foram levantados os requisitos do STRADGE, apresentados a seguir, baseados nas necessidades do grupo de pesquisa de geologia por um sistema capaz de realizar o processamento de um conjunto de dados armazenados.

Em um primeiro momento o levantamento deu-se com o estudo dos conceitos de geologia associados aos dados coletados, sob quais formatos estes costumam ser representados e como poderiam ser adequadamente armazenados pelo STRADGE.

Os dados armazenados pelo sistema são referentes à medidas de orientações de falhas geológicas contidas em estruturas de afloramento, que pode-se definir por corpos rochosos expostos à superfície terrestre.

Em seguida, foi realizado um estudo sobre a Projeção Estereográfica e sobre as técnicas desenvolvidas. O estudo ocorreu através de reuniões com especialistas na área de Geologia e com o auxílio da obra de Carneiro (1996). As técnicas de Projeção

Estereográfica foram escolhidas com base nas necessidades do grupo de pesquisa do Laboratório de Geomodelagem do DPM, da UNESP de Rio Claro.

Com as necessidades estabelecidas e os estudos necessários realizados, foram então definidos os requisitos do sistema, listados de forma resumida na **Tabela 4**.

**Tabela 4:** Requisitos do STRADGE.

<b>Requisitos do Sistema</b>
<b>1. Cadastro de Usuários no Sistema.</b>
<b>2. Autenticação de Usuários.</b>
<b>3. Importação de Dados para o Sistema.</b>
<b>4. Apresentação dos Conjunto de Dados.</b>
<b>5. Visualização de Mapas com dados relacionados.</b>
<b>6. Seleção de Processos de Tratamento.</b>
<b>7. Tratamento de Dados.</b>
<b>8. Possibilidade de salvar e compartilhar dados tratados.</b>

Os requisitos 1 e 2 tratam do controle de usuário que a aplicação deve realizar, pois apenas usuários devidamente identificados podem acessar o sistema.

O requisito 3 aborda a importação de dados externos para o STRADGE, através de arquivos no formato XLS e *shapefile*. Além disso, também considera a publicação automática dos dados importados sob os serviços WMS e WFS.

Uma vez importados os dados, o requisito 4 garante que estes devem ser apresentados aos usuários em forma de listas, diferenciando os dados públicos e privados.

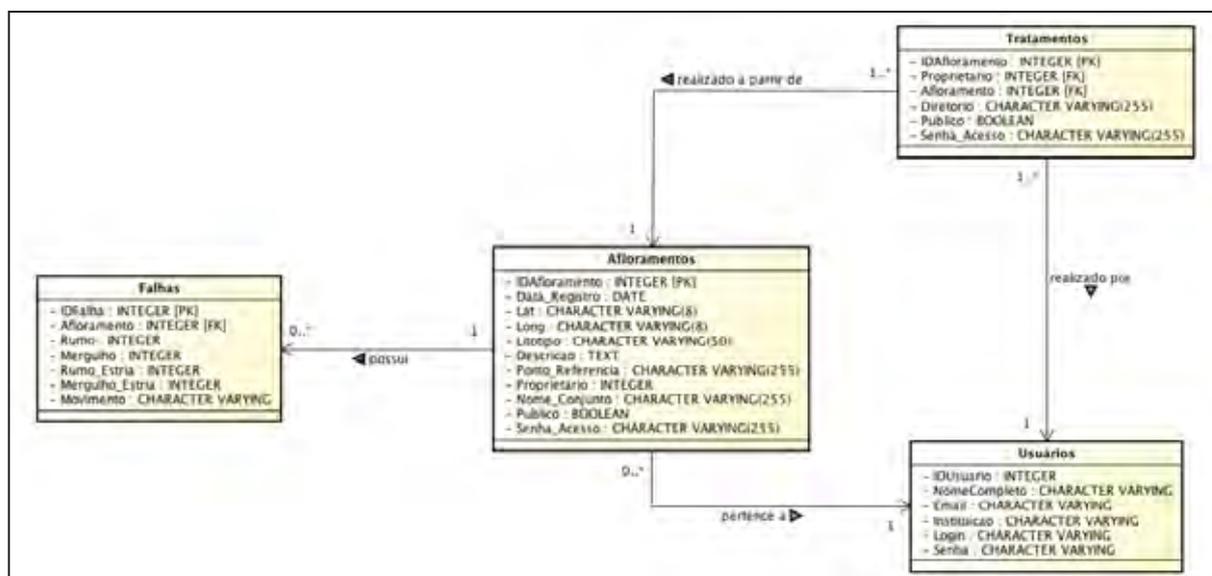
Os requisitos 5 e 6 tratam da visualização dos dados em um mapa dinâmico e da seleção de um processo de tratamento para estes dados, que no caso do STRADGE, são os processos de geração de diagramas.

O requisito 7 trata do procedimento para o tratamento de dados, utilizando os processos de geração de diagramas de projeção estereográfica, entre outros.

Por fim, o requisito 8 identifica a possibilidade de salvar e compartilhar o resultado (diagramas) dos processos executados.

Além dos requisitos iniciais do sistema, foi elaborado também um modelo de dados, apresentado na **Figura 12**. Este modelo, baseado em um modelo utilizado em um projeto do

Laboratório de Geomodelagem do DPM da UNESP de Rio Claro, foi definido para validar a arquitetura e servir de base para a implementação realizada no desenvolvimento do trabalho, portanto, conta apenas com alguns elementos.



**Figura 12:** Modelo de Dados do STRADGE.

Para atender aos requisitos do sistema, foram identificados os casos de uso, listados na **Tabela 5**. Os casos de uso destacados são considerados críticos (os mais importantes) para o desenvolvimento do sistema.

**Tabela 5:** Relação dos casos de uso identificados para o STRADGE.

Sigla	Caso de Uso
CDU1	Cadastrar no Sistema
CDU2	Autenticar-se no Sistema
CDU3	Importar Dados
CDU4	Listar Dados
CDU5	Visualizar Dados
CDU6	Tratar Dados
CDU7	Salvar Tratamentos
CDU8	Editar permissões de Acesso a Dados

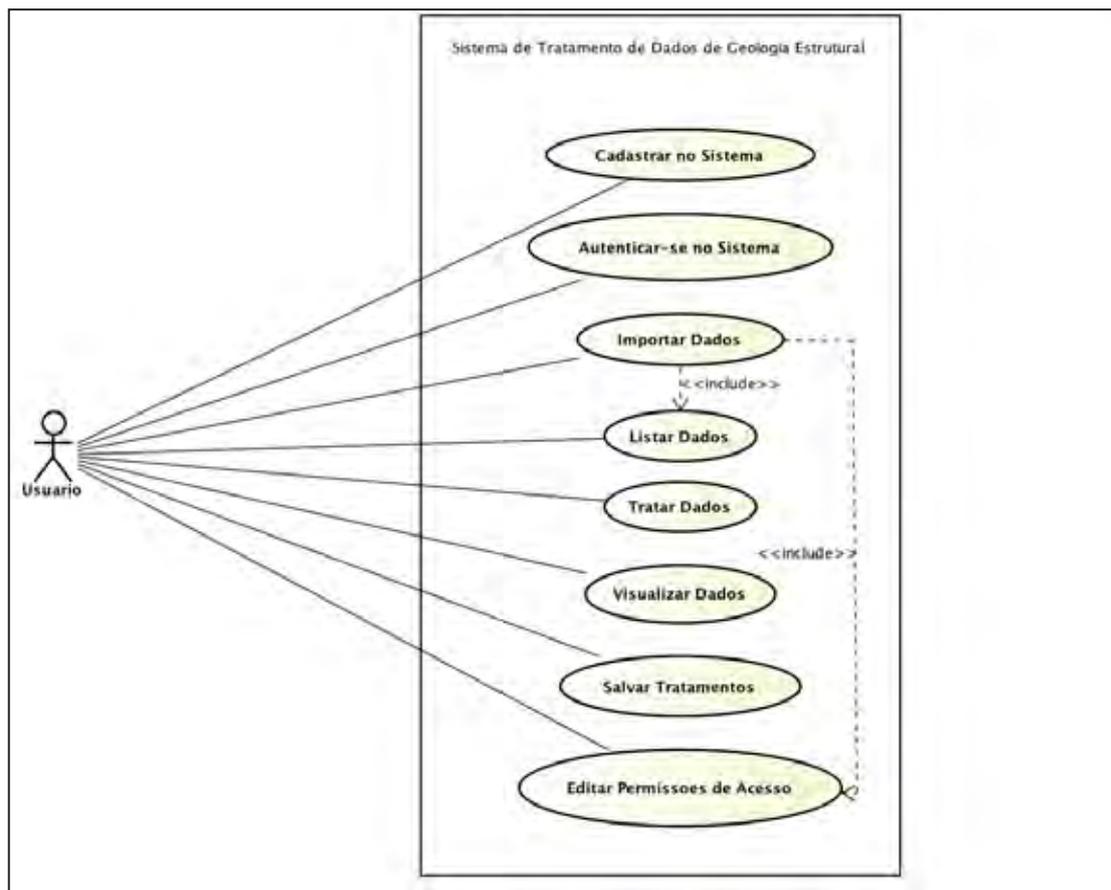
Em razão da complexidade do sistema, com exceção do caso de uso CDU8, todos os casos de uso foram desenvolvidos parcialmente visando validar a arquitetura. O caso de

uso CDU3 foi implementado parcialmente, suportando apenas um formato de arquivo para importação. Já os casos para os casos de uso CDU5, CDU6, e CDU7, apenas funcionalidades essenciais foram implementadas, para validar o funcionamento da arquitetura proposta.

A apresentação dos conjuntos disponíveis para a visualização (CDU3) e a visualização dos dados (CDU5) foram implementadas de forma simples, exibindo, respectivamente, apenas informações básicas dos conjuntos, separando-os em públicos e privados, e dos Afloramentos (estruturas geológicas).

O tratamento de dados (CDU6) é feito apenas para os processos de geração de diagramas de projeção estereográfica, e a opção de Salvar os Tratamentos (CDU7) apenas armazena os resultados gerados e exibe, para cada conjunto de dados, se já houve ou não um tratamento com aquele conjunto.

Na **Figura 13** é apresentado o diagrama de casos de uso do sistema.



**Figura 13:** Diagrama de Casos de Uso para o STRADGE.

O passo seguinte consiste na descrição detalhada dos casos de uso. Na **Tabela 6**, é apresentada a descrição detalhada do caso de uso CDU6.

Tabela 6: Caso de Uso CDU6: Tratar Dados.

<b>CDU6: Tratar Dados</b>	
<b>Escopo:</b>	<ul style="list-style-type: none"> <li>• STRADGE</li> </ul>
<b>Nível:</b>	<ul style="list-style-type: none"> <li>• Objetivo do usuário</li> </ul>
<b>Ator principal:</b>	<ul style="list-style-type: none"> <li>• Usuário</li> </ul>
<b>Interessados e interesses:</b>	<ul style="list-style-type: none"> <li>• Usuário: deseja tratar dados publicados.</li> </ul>
<b>Garantia de Sucesso (Pós-Condições):</b>	<ul style="list-style-type: none"> <li>• O Usuário seleciona conjunto de dados e o processo de tratamento, em seguida requisita ao Sistema para executá-lo.</li> </ul>
<b>Pré-Condições:</b>	<ul style="list-style-type: none"> <li>• Usuário deve estar autenticado no Sistema.</li> <li>• Dados devem ter sido inseridos no Sistema.</li> <li>• Um conjunto de dados deve ter sido selecionado para exibição.</li> </ul>
<b>Cenário de Sucesso Principal (Fluxo Básico):</b>	<ol style="list-style-type: none"> <li>1. Usuário seleciona uma estrutura, cujos dados deseja tratar.</li> <li>2. Sistema pergunta ao Usuário qual processo de tratamento deseja utilizar.</li> <li>3. Usuário seleciona o processo desejado.</li> <li>4. Usuário determina os parâmetros adicionais do processo.</li> <li>5. Sistema executa o processo selecionado com o conjunto de dados determinado pelo Usuário.</li> <li>6. Sistema encaminha Usuário para interface que exibirá o resultado do processo que foi executado, permitindo interação do Usuário e possibilidade de salvar o trabalho.</li> </ol>
<b>Extensões (Fluxo Alternativo):</b>	<ol style="list-style-type: none"> <li>1. Ocorre um erro com a seleção de dados do Usuário. <ol style="list-style-type: none"> <li>a. Sistema exibe mensagem de erro ao Usuário.</li> <li>b. Sistema solicita que o Usuário selecione os dados novamente.</li> </ol> </li> <li>2. Ocorre um erro quando o Sistema executa o processo selecionado. <ol style="list-style-type: none"> <li>a. Sistema exibe mensagem de erro ao Usuário.</li> <li>b. Sistema solicita que o Usuário tente realizar o procedimento novamente.</li> </ol> </li> </ol>
<b>Frequência de Ocorrência:</b>	<ul style="list-style-type: none"> <li>• Aleatória.</li> </ul>
<b>Problemas em aberto:</b>	<ul style="list-style-type: none"> <li>• Nenhum.</li> </ul>

Ao final da etapa de concepção do projeto, os artefatos produzidos foram os requisitos do sistema, a definição dos casos de uso e o detalhamento de cada caso de uso.

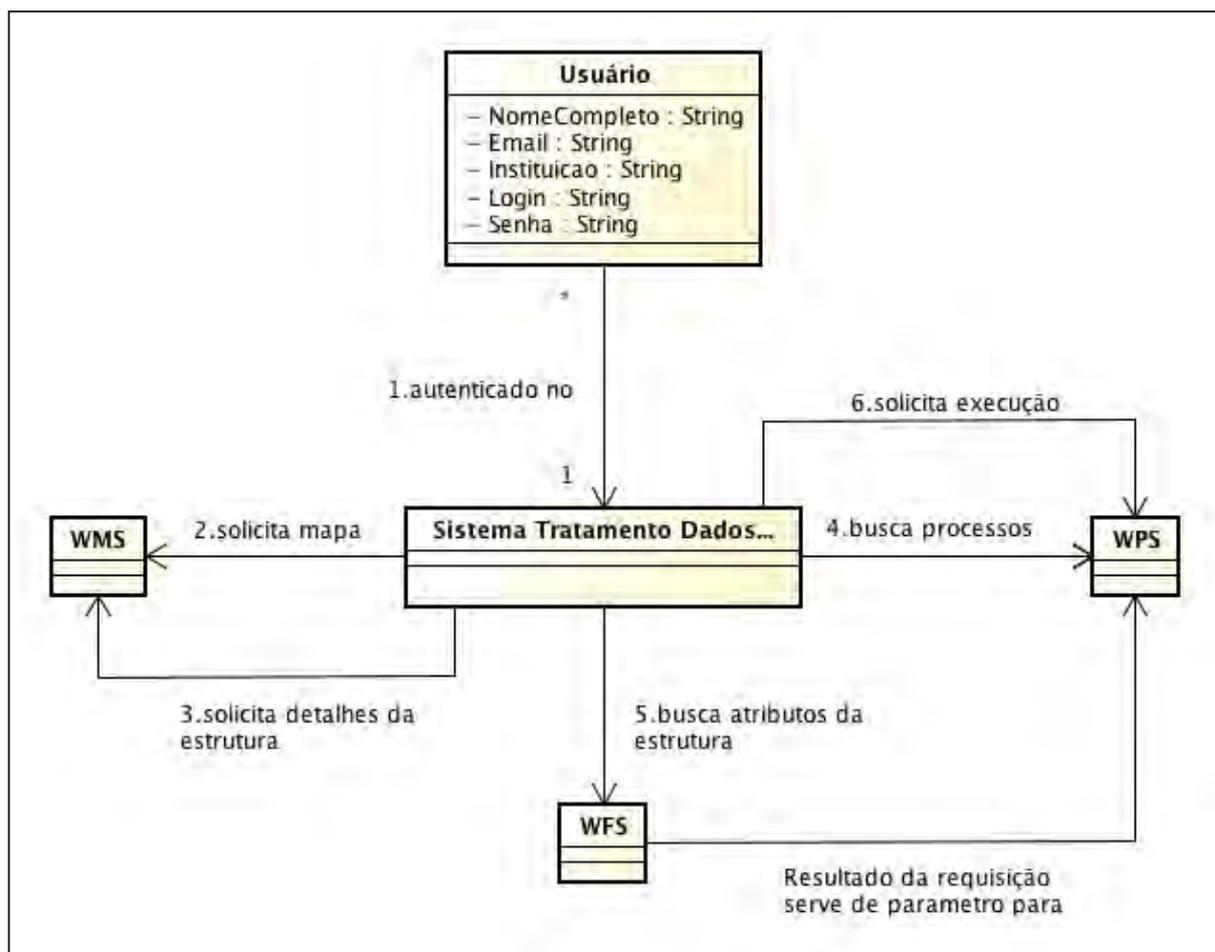
## 4.2 ELABORAÇÃO

O passo seguinte consiste em elaborar um Modelo de Domínio a partir da descrição detalhada de um caso de uso, um Diagrama de Seqüência do Sistema (DSS), definir os Contratos de Operações e em seguida, elaborar um Diagrama de Seqüência que mostra a colaboração entre os elementos de software; por fim, elaborar um Diagrama de Classes, que exibirá os elementos de software e seus pacotes, utilizados então para a implementação de código.

### 4.2.1 Iteração 1

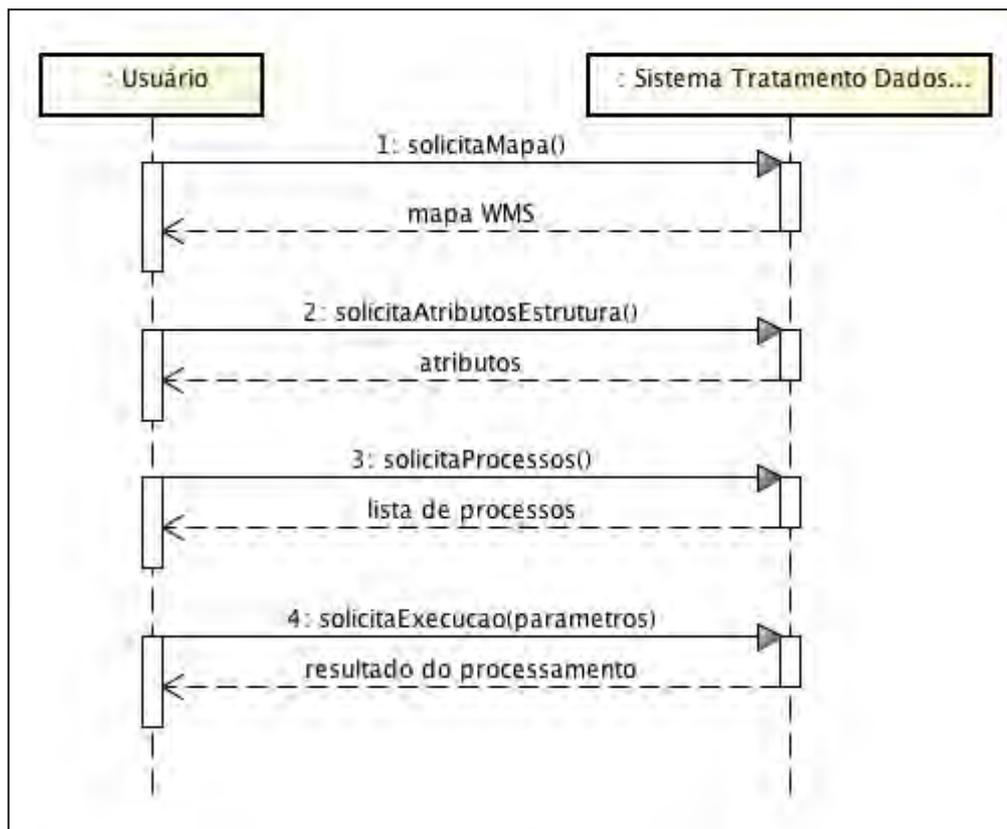
Nesta primeira iteração, o caso de uso CDU6 foi completamente projetado e implementado de forma básica, para garantir seu papel no sistema. Após algumas iterações, esta etapa passou por revisões, principalmente no que diz respeito ao controle de acesso de usuários, que foi projetado e implementado em uma iteração posterior.

O primeiro artefato desenvolvido a partir da descrição do CDU6 foi um Modelo de Domínio em que são apresentadas as classes conceituais e seus relacionamentos. Na **Figura 14** é apresentado o Modelo de Domínio para o caso de uso CDU6.



**Figura 14:** Modelo de Domínio para o Caso de Uso CDU6.

Em seguida, foi elaborado um Diagrama de Seqüência, no qual é possível ter uma visão inicial do problema, no qual um usuário seleciona os dados de uma determinada estrutura para realizar um tratamento, define qual processo vai ser utilizado para a requisição da execução do processo escolhido, com os parâmetros adicionais definidos. Na **Figura 15** é apresentado o Diagrama de Seqüência Inicial para o caso de uso CDU 6.



**Figura 15:** Diagrama de Sequencia do Sistema (DSS) para o Caso de Uso CDU6.

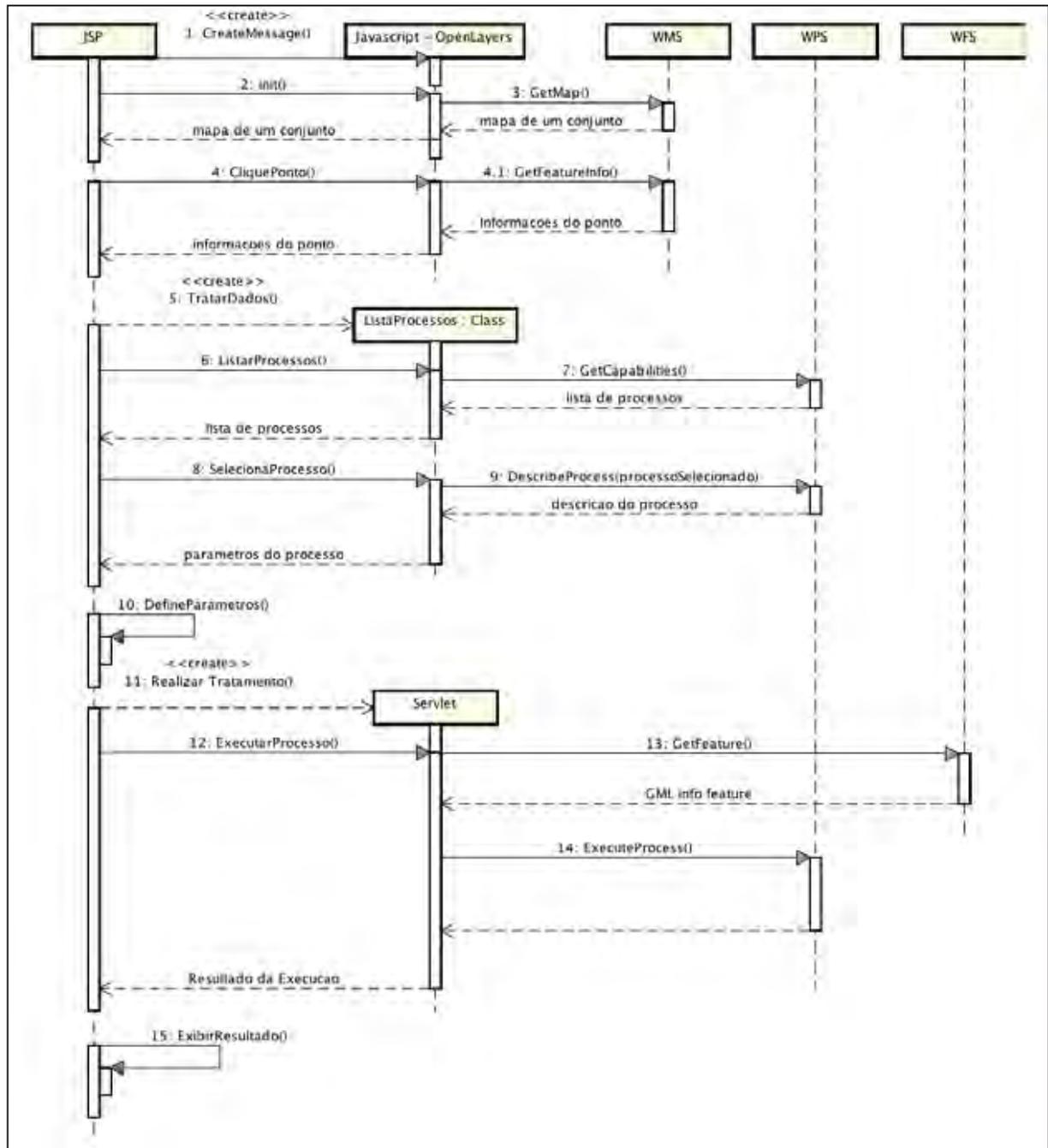
A partir da elaboração do DSS, é feita a formalização das operações necessárias para atingir o objetivo do caso de uso com sucesso, através de um Contrato de Operação. Na **Tabela 7** é apresentada como exemplo, a descrição do contrato para a operação “solicitaMapa”.

**Tabela 7:** Contrato de Operação CO1.

<b>Contrato de Operação CO1: solicitaMapa</b>	
<b>Operação:</b>	solicitaMapa
<b>Referencias cruzadas:</b>	-
<b>Pré-Condições:</b>	Usuário deve estar autenticado no sistema.
<b>Pós-Condições:</b>	Sistema apresentou uma interface com um mapa contendo a localização das estruturas do conjunto de dados selecionado.

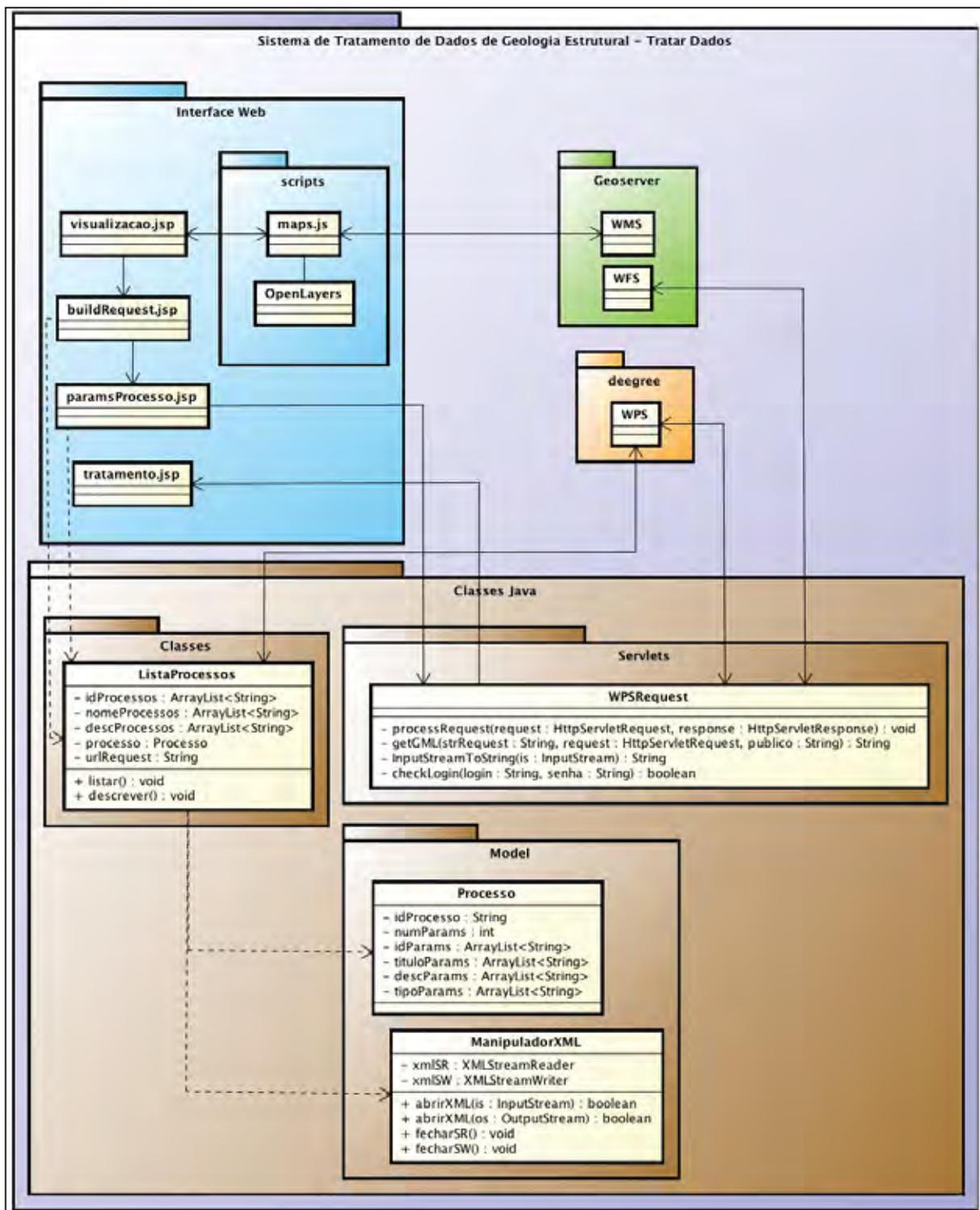
Após a formalização de todas as operações em Contratos de Operação, o artefato seguinte é o Diagrama de Seqüência, onde são apresentadas as interações entre os objetos de software.

No caso desta arquitetura orientada a serviços, os serviços são considerados objetos de software, e são representados como unidades (WMS, WPS, WFS) no diagrama da **Figura 16**.



**Figura 16:** Diagrama de Seqüência exibindo a colaboração entre objetos de software.

O último artefato é o Diagrama de Classes, onde são apresentados os objetos de software e seus pacotes detalhadamente. Os serviços são elementos externos à Aplicação, portanto não possuem detalhamento no diagrama apresentado na **Figura 17**.



**Figura 17:** Diagrama de Classes do Caso de Uso CDU6.

Ao final da primeira iteração, o código referente aos elementos do caso de uso CDU6 é escrito. Em seguida é dado início a próxima iteração, considerando outro caso de uso crítico.

Como citado anteriormente, os elementos implementados nesta iteração passaram por modificações após iterações seguintes, principalmente após a definição de controle de usuários, que inseriu etapas de autenticação em algumas interações entre os objetos de software.

### **4.3 CONSTRUÇÃO**

A fase de construção é a última fase para a implementação do sistema. Neste caso, foram implementados os casos de uso que não foram considerados críticos para o funcionamento do sistema, o CDU1, CDU2, e CDU8.

Como citado anteriormente, o caso de uso CDU8 não foi implementado por não ser considerado essencial na validação da arquitetura proposta, uma vez que a edição de permissão não influencia as demais funcionalidades do sistema.

No processo de implementação do caso de uso CDU1 e principalmente do caso de uso CDU2, Cadastro de Usuários e Autenticação, respectivamente, foi necessário introduzir elementos de verificação de permissão e Login devidamente efetuado nas outras partes do sistema.

Apenas para ilustrar o início da fase de construção, na **Tabela 8** é apresentada a descrição detalhada do caso de uso CDU1.

Tabela 8: Caso de Uso CDU1: Cadastrar no Sistema.

<b>CDU1: Cadastrar no Sistema</b>	
<b>Escopo:</b>	<ul style="list-style-type: none"> <li>• STRADGE</li> </ul>
<b>Nível:</b>	<ul style="list-style-type: none"> <li>• Objetivo do usuário</li> </ul>
<b>Ator principal:</b>	<ul style="list-style-type: none"> <li>• Usuário</li> </ul>
<b>Interessados e interesses:</b>	<ul style="list-style-type: none"> <li>• Usuário: deseja cadastrar-se no sistema.</li> </ul>
<b>Garantia de Sucesso (Pós-Condições):</b>	<ul style="list-style-type: none"> <li>• O Usuário cadastra-se no sistema, tornando possível acessar os dados públicos e/ou importar dados privados.</li> </ul>
<b>Pré-Condições:</b>	<ul style="list-style-type: none"> <li>• Usuário não deve estar cadastrado no Sistema.</li> </ul>
<b>Cenário de Sucesso Principal (Fluxo Básico):</b>	<ol style="list-style-type: none"> <li>1. Usuário acessa formulário de cadastro.</li> <li>2. Usuário preenche seus dados no formulário.</li> <li>3. Sistema confirma cadastro do Usuário.</li> <li>4. Sistema direciona Usuário para página de login.</li> </ol>
<b>Extensões (Fluxo Alternativo):</b>	<ol style="list-style-type: none"> <li>1. Login já cadastrado.               <ol style="list-style-type: none"> <li>a. Sistema exibe mensagem de erro ao Usuário.</li> <li>b. Sistema solicita que o Usuário escolha outro login.</li> </ol> </li> </ol>
<b>Frequência de Ocorrência:</b>	<ul style="list-style-type: none"> <li>• Aleatória.</li> </ul>
<b>Problemas em aberto:</b>	<ul style="list-style-type: none"> <li>• Nenhum.</li> </ul>

#### 4.4 TRANSIÇÃO

Finalmente, a etapa de transição abrange os passos necessários para a implantação do STRADGE em um servidor e como os usuários podem acessá-lo.

Considerando que o STRADGE é um sistema web, é necessário que haja a configuração de um servlet container, capaz de servir as páginas e serviços desenvolvidos utilizando a linguagem Java e que manipulam outras tecnologias. Também é necessário a instalação de um SGBD para armazenar todos os dados do sistema. O servlet container utilizado no servidor é o Apache Tomcat 6.0.29 e o SGBD é o PostgreSQL.

Considerando a arquitetura apresentada na seção 3, o primeiro passo é configurar uma base de dados utilizando o PostgreSQL para que seja possível armazenar os dados do sistema.

Em seguida, é preciso realizar a implantação do GeoServer e do Deegree no servidor Tomcat. Ambos os sistemas são disponibilizados pelos seus desenvolvedores como arquivos do tipo WAR (Web Application Archive), que são descompactados automaticamente pelo Tomcat.

Por fim, tanto os serviços relacionados aos módulos “Controle Usuários”, “Controle Tratamentos” e “Cadastro Dados”, como a interface web da Aplicação, apresentados na descrição da arquitetura na seção 3, também são implantados no servidor através de arquivos WAR, estes por sua vez gerados pela IDE Eclipse, após o processo de compilação.

Para acessar a interface web do STRADGE, o usuário deve possuir um navegador com suporte às tecnologias utilizadas e deve acessar o sistema pelo protocolo HTTP.

No momento do desenvolvimento do trabalho, o navegador que mostrou melhor suporte a todas as tecnologias utilizadas, principalmente suporte a WebGL, foi o Mozilla Firefox 4 (beta).

## 5. O SISTEMA DE TRATAMENTO DE DADOS DE GEOLOGIA

Nesta seção são apresentadas as funcionalidades implementadas do STRADGE. Apenas as principais funcionalidades, que validam a arquitetura, foram implementadas, tais como a autenticação e cadastro de usuários, a importação e visualização de dados geográficos e execução do método de projeção estereográfica. Quanto à interface gráfica, foi desenvolvida apenas para avaliar um protótipo, por isso não considera questões de usabilidade e padronização.

### 5.1 AUTENTICAÇÃO E CADASTRO

Para que um usuário possa utilizar o sistema, ele deve realizar o login com suas credenciais previamente cadastradas. Caso um usuário não esteja cadastrado, é possível que ele se cadastre através da interface apresentada na **Figura 18**. Na **Figura 19** é apresentada a página de login, a qual o usuário visualiza assim que acessa o sistema ou finalize seu cadastro.



The screenshot shows a web browser window with the URL `http://localhost:8081/STDGE/cadastro.jsp`. The page title is "STDGE - Sistema para Tratamento de Dados de Geologia Estrutural". The main content area contains a registration form with the following fields and labels:

- Nome Completo:
- E-mail:
- Empresa/Instituição:
- Login:
- Senha:
- Confirmar senha:

At the bottom of the form, there are two buttons: "Cancelar" and "Cadastrar".

Figura 18: Página de Cadastro de Usuário.



The screenshot shows a web browser window with the URL `http://localhost:8081/STDGE/`. The page title is "STDGE - Sistema para Tratamento de Dados de Geologia Estrutural". The main content area contains a login form with the following fields and labels:

- Login:
- Senha:

Below the password field is a "Login" button. Below the login form, there is a section for "Novo usuário:" with a "Cadastrar-se" button.

Figura 19: Página de Login.

## 5.2 MANIPULAÇÃO DE DADOS

As funcionalidades que manipulam dados abrangem sua importação para o sistema, a relação de conjuntos disponíveis para acesso e a visualização de um determinado conjunto em um mapa.

### 5.2.1 Importação de Dados

A opção de Importar Dados permite que um usuário registre seus próprios conjuntos de dados no sistema. Como pode ser observado na interface web apresentada na **Figura 20**, o usuário deve selecionar a fonte dos dados, definir um nome para o conjunto, fornecer o arquivo com os dados e definir se o conjunto é público ou privado.

Após o processo de importação, os dados estão armazenados no Banco de Dados do sistema e também publicados para serem acessados através dos serviços WMS e WFS.



The screenshot shows a web browser window with the URL `http://localhost:8081/STDGE/importarDados.jsp`. The page title is "STDGE - Sistema para Tratamento de Dados de Geologia Estrutural". The navigation menu includes "Principal", "Importar Dados", and "Sair". The main content area is titled "Selecione o formato de importação:" and contains two radio buttons: "Shapefile" and "XLS (Microsoft Excel) [Baixar exemplo](#)". Below this is a section "Defina qual nome usar a estrutura de dados:" with a text input field containing "nome do conjunto". The next section is "Selecione a origem para os dados:" with a "Selecionar arquivo..." button. The final section is "Defina o acesso aos dados:" with two radio buttons: "Acesso Público" and "Restrito". A "Enviar" button is located at the bottom left of the form.

**Figura 20:** Página com formulário para importação de dados.

## 5.2.2 Conjuntos de Dados Disponíveis

Uma vez que um usuário efetua a autenticação corretamente no sistema, são apresentados os conjuntos de dados aos quais aquele usuário possui acesso. Como é possível ver na interface web apresentada na **Figura 21**, há duas listas de conjuntos, a primeira lista contém os dados de acesso privado, ou seja, que apenas o usuário registrado pode acessar; e a segunda lista contém dados públicos, ou seja, conjuntos de dados cadastrados por outros usuários ou mesmo pelo usuário autenticado, uma vez que ele os tenha disponibilizado para acesso de terceiros.



**Figura 21:** Página principal, acessada apenas por usuários autenticados, que exibe os conjuntos de dados disponíveis.

Para cada conjunto de dados, é possível observar também a opção de “Visualizar”. Esta opção leva o usuário para um outra página onde é possível visualizar os dados de um determinado conjunto em um mapa dinâmico.

### 5.2.3 Visualização de Dados

Quando um usuário faz opção por visualizar um conjunto de dados, uma interface exibe um mapa que, como pode ser visto na **Figura 22**, apresenta uma camada básica com o contorno do território do Brasil, e uma camada que contém pontos referentes à localização de cada afloramento registrado naquele conjunto.



**Figura 22:** Interface com mapa de um conjunto de dados.

Se o usuário selecionar um afloramento exibido no mapa, um *pop-up* aparecerá contendo as informações básicas sobre o afloramento, como apresenta a **Figura 23**. Neste mesmo *pop-up* é exibido um link para o tratamento de dados (“Tratar Dados”), que se acessado pelo usuário passa as informações daquele ponto selecionado para a parte do sistema responsável por iniciar o tratamento de dados.

O primeiro passo para realizar um tratamento com os dados do ponto selecionado no mapa é escolher o processo que se deseja executar.

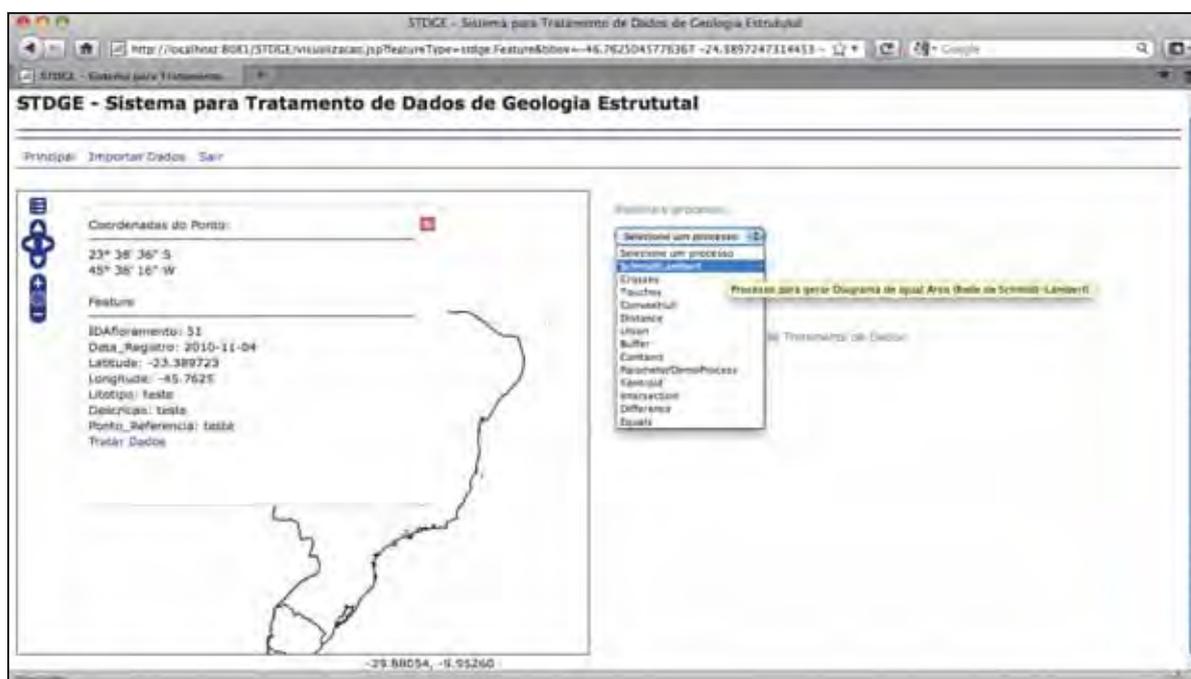
### 5.3 EXECUÇÃO DE UM PROCESSO

A seguir são apresentadas as funcionalidades relacionadas com a execução de um processo, tais como a seleção de um processo, a definição de seus parâmetros, a apresentação dos resultados e a possibilidade de salvar os resultados gerados.

### 5.3.1 Seleção de um Processo

Após a seleção de um afloramento e a exibição dos detalhes sobre ele, assim que o usuário selecionar o link “Tratar Dados”, uma opção surge ao lado do mapa para que seja escolhido qual processo deverá ser executado.

Um exemplo é apresentado na **Figura 23**, onde o usuário seleciona o processo de geração de diagrama de projeção estereográfica.



**Figura 23:** Lista de processos disponíveis para o tratamento dos dados.

### 5.3.2 Parâmetros de um Processo

Ao selecionar um processo, é solicitado ao usuário que informe quais são os parâmetros que aquele processo necessita.

Para a execução do processo denominado “SchmidtLambert”, são solicitados os parâmetros apresentados na **Figura 24**.



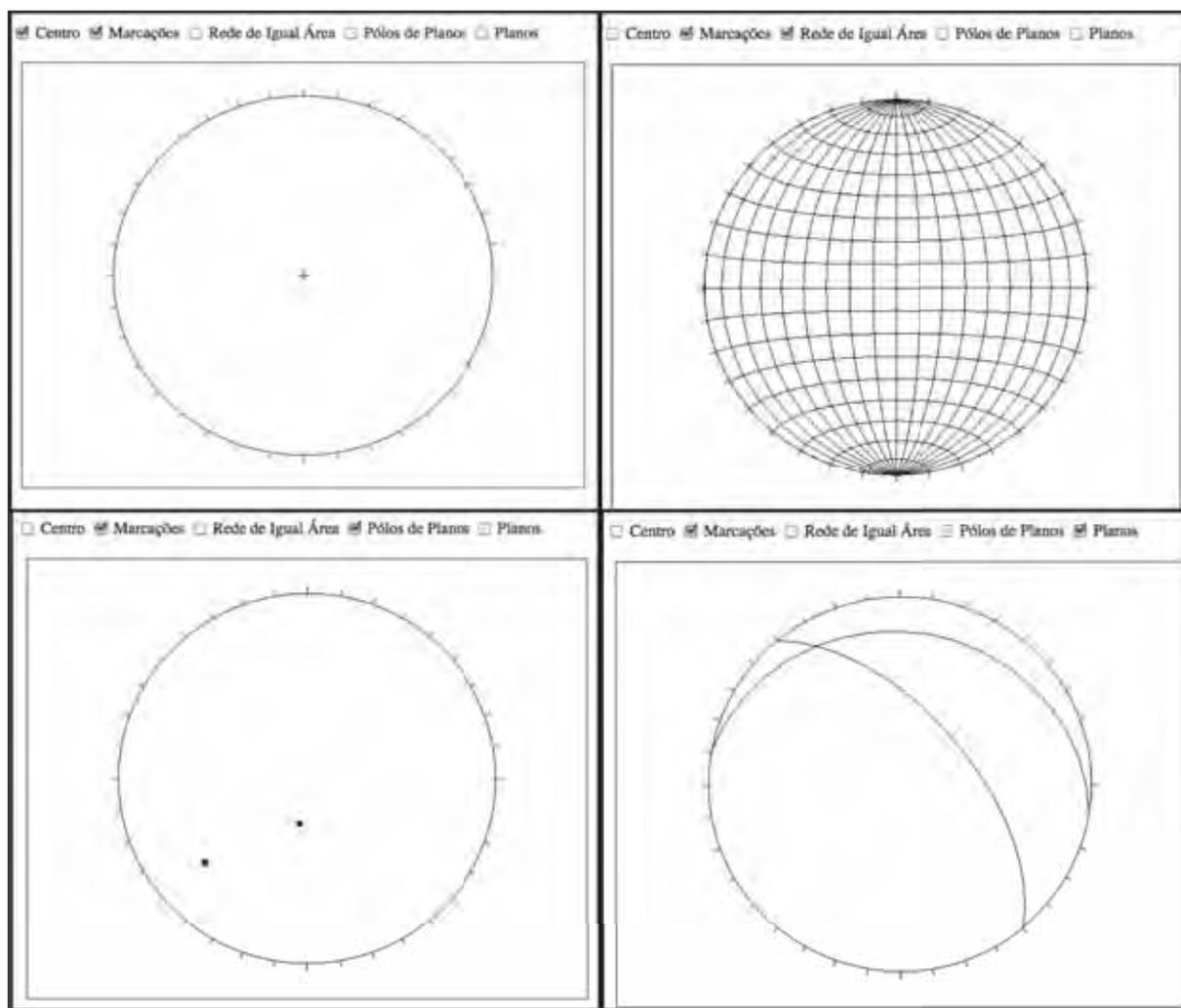
**Figura 24:** Destaque para a parte da interface que apresenta os parâmetros do processo de geração de um Diagrama de Igual Área.

Alguns parâmetros são automaticamente obtidos pelo sistema, geralmente os que dizem respeito aos dados cadastrados. Outros parâmetros requerem uma entrada do usuário, como é o caso do raio do círculo de um diagrama de projeção estereográfica, como é apresentado também na **Figura 24**.

### 5.3.3 Execução e Resposta de um Processo

Uma vez definidos o processo e seus parâmetros, basta o usuário selecionar o botão “Realizar Tratamento” que o sistema envia os parâmetros de entrada para o processo, através de uma requisição WPS, e apresenta, ao lado do mapa, o resultado da execução.

No caso de um processo de geração de um diagrama de projeção estereográfica, o resultado da execução é tratado e apresentado em uma interface interativa, na qual é possível que o usuário altere a visualização de elementos do diagrama, como é possível ver na seqüência apresentada na **Figura 25**.



**Figura 25:** Interface interativa com as opções de elementos de um diagrama de Igual Área.

### 5.3.4 Salvar Resultados de um Processo

A última funcionalidade a ser descrita é a opção que permite ao usuário registrar no sistema os resultados da execução de um determinado processo com um conjunto de dados selecionado.

No caso da implementação do processo de geração de diagramas de projeção estereográfica, o sistema armazena uma referência ao usuário que realizou a requisição de processamento; qual o conjunto de dados utilizado e as coordenadas dos elementos do diagrama; além de permitir comentários do próprio usuário.

Os dados podem ficar a disposição de outros usuários, dependendo da opção de permissão de acesso. A reutilização dos dados não foi implementada, uma vez que cada usuário pode utilizar os dados de diversas maneiras. Na implementação concretizada, é possível salvar o tratamento e visualizar quais conjuntos já possuem dados tratados.

## 6. CONCLUSÕES

Neste trabalho foi possível desenvolver estudos que possibilitaram a concepção, projeto e implementação de uma arquitetura que permite processar dados publicados sob serviços de manipulação de dados geográficos, e compartilhar não só os dados em si, como também os resultados da execução de processos. Utilizando a arquitetura proposta foi feita a implementação do STRADGE.

A utilização do padrão de serviço geográfico WPS como um padrão para a requisição de execução de processos, aliado aos padrões de serviços WMS e WFS, dá suporte para o processamento tanto de dados geográficos como também de dados associados. A adoção destes padrões de serviços no caso do STRADGE pode ser considerado como um avanço na visão de sistemas de processamentos de dados de geologia estrutural, pois integra processos e seus resultados à localização geográfica das informações, fundamental nos estudos da área.

Além de atender às necessidades no escopo da área de Geologia Estrutural, mais especificamente com a geração de diagramas de projeção estereográfica, é possível que a mesma estrutura de serviços e componentes definidos na arquitetura seja utilizada para atender necessidades semelhantes de qualquer outra área, considerando devidas adaptações.

Com relação à utilização de serviços, observou-se que há uma tendência cada vez maior na utilização de uma arquitetura orientada a serviços na solução de problemas, não só pelas vantagens técnicas, mas principalmente pela possibilidade de criar ambientes em que um grande número de usuários possam interagir; e também pela possibilidade de integração entre sistemas, através da comunicação entre serviços.

Um outro aspecto importante no desenvolvimento do trabalho foi a adoção e a utilização de softwares livres, como o Deegree e o Geoserver, sendo fundamental o contato com a comunidade de usuários para a troca de experiências e principalmente com a

comunidade dos próprios desenvolvedores, sempre solícitos a avaliar possíveis problemas e precisos ao apontar soluções, alternativas e esclarecimentos sobre seus sistemas, o que colaborou muito para o andamento do trabalho e sua concretização.

Como sugestões para trabalhos futuros neste contexto, surge a possibilidade de haver comunicação entre os usuários através do próprio sistema, para que os processamentos e as informações possam ser discutidas e avaliadas; e o controle individual de permissão de acesso a um conjunto de dados, para que os usuários possam formar grupos de trabalho. A implementação de novos processos de tratamento de dados de geologia deve ser considerada na implementação da arquitetura proposta.

## REFERÊNCIAS

CARNEIRO, C. D. R. **Projeção estereográfica para análise de estruturas**: Programas ESTER e TRADE. Fundamentos teóricos, exercícios e aplicações em microcomputador, laboratório e campo. 1. 66T. Campinas: co-edição CPRM/IG-UNICAMP/IPT-DIGEO, 1996. V. s/n. 184 p.

**DEEGREE** Disponível em: <[66TTP://www.deegree.org/](http://www.deegree.org/)>. Acesso em: 10 out. 2010.

DEITEL, H.; DEITEL, P. **Java**: Como Programar. 6. 66T. São Paulo: Prentice Hall, 2005. 1110 p.

DUTCH, S. **Spherical Projections**. Natural and Applied Science. University of Wisconsin. Green Bay. 1999. Disponível em: <[66TTP://www.uwgb.edu/dutchs/structge/sphproj.htm](http://www.uwgb.edu/dutchs/structge/sphproj.htm)>. Acesso em: 15 ago. 2010.

FIELDING, Roy Thomas. **Architectural Styles and the Design of Network-based Software Architectures**. 2000. 162 f. Dissertação (Doutorado) – University Of 66TTP66órnia, Irvine, 2000. Cap. 5. Disponível em: <[66TTP://www.ics.uci.edu/~fielding/pubs/dissertation/fielding\\_dissertation\\_2up.pdf](http://www.ics.uci.edu/~fielding/pubs/dissertation/fielding_dissertation_2up.pdf)>. Acesso em: 25 set. 2010.

**GEOSERVER** Disponível em: <[66TTP://geoserver.org/display/GEOS/Welcome](http://geoserver.org/display/GEOS/Welcome)>. Acesso em: 5 ago. 2010.

GREINER G., HORMANN K., **Efficient clipping of arbitrary polygons**, ACM Transactions on Graphics, Vol. 17, 1998, 71-83 p.

KHRONOS GROUP. **WebGL**: OpenGL ES 2.0 for the Web. Disponível em: <<http://www.khronos.org/webgl/>>. Acesso em: 20 out. 2010.

LARMAN, C. **Utilizando UML e padrões: uma introdução à análise e ao projeto orientados a objetos e ao desenvolvimento iterativo**. Tradução Rosana Vaccare Braga. 3. 66T. Porto Alegre: Bookman, 2007. 695 p.

MA, W. et al. **A 3D Visualization and Management Framework for Geological Disaster Monitoring and Early Warning based on Virtual Globes**. Disponível em: <[http://ieeexplore.ieee.org/xpl/freeabs\\_all.jsp?arnumber=5567614](http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5567614)>. Acesso em: 15 out. 2010.

MENDES, A. F.; KULLBERG J. C. **Drawing and use of auxiliary projection nets: the program STEGRAPH**. Centro de Estratigrafia e Paleobiologia da Universidade Nova de Lisboa, Portugal. 1992.

OASIS. **OASIS Standards**. Disponível em: <<http://www.oasis-open.org/specs/index.php#uddiv3.0.2>>. Acesso em: 30 set. 2010.

OGC. OPEN GEOSPATIAL CONSORTIUM. **Welcome to the OGC Website**. Disponível em: <<http://www.opengeospatial.org/>>. Acesso em: 30 set. 2010.

**WMS. Web Map Service.** 2010. Disponível em: <<http://www.opengeospatial.org/standards/wms>>. Acesso em: 30 set. 2010.

**WFS. Web Feature Service.** 2010. Disponível em: <<http://www.opengeospatial.org/standards/wfs>>. Acesso em: 30 set. 2010.

**WPS. Web Processing Service.** 2010. Disponível em: <<http://www.opengeospatial.org/standards/wps>>. Acesso em: 30 set. 2010.

**GML. Geography Markup Language,** 2010. Disponível em: <<http://www.opengeospatial.org/standards/gml>>. Acesso em: 30 set. 2010.

**OPENLAYERS:** Free Maps for the Web. Disponível em: <<http://www.openlayers.org/>>. Acesso em: 10 ago. 2010.

**POSTGIS** Disponível em: <<http://postgis.refractor.net/>>. Acesso em: 15 ago. 2010.

**POSTGRESQL:** The world's most advanced open source database. Disponível em: <<http://www.postgresql.org/>>. Acesso em: 5 ago. 2010.

**SOA WORK GROUP. The Open Group:** Making Standards Work. Disponível em: <<http://opengroup.org/projects/soa/doc.tpl?CALLER=documents.tpl&dcat=&gdid=10632>>. Acesso em: 25 set. 2010.

**W3C. WORLD WIDE WEB CONSORTIUM.** 2010. **HTML5:** A vocabulary and associated APIs for HTML and XHTML. Disponível em: <<http://www.w3.org/TR/html5/>>. Acesso em: 20 set. 2010.

**WSDL. Web Services Description Language (WSDL) 1.1.** 2010. Disponível em: <[http://www.w3.org/TR/wsdl#\\_wsdl](http://www.w3.org/TR/wsdl#_wsdl)>. Acesso em: 20 set. 2010.

**SOAP. Latest SOAP versions.** 2010. Disponível em: <<http://www.w3.org/TR/soap/>>. Acesso em: 20 set. 2010.

**YANG, Q. et al. Research and Implementation of Geological Disaster Spatial Information Management Platform Framework Based on Geospatial Portal.** Disponível em:

<<http://ieeexplore.ieee.org/Xplore/login.jsp?url=http%3A%2F%2Fieeexplore.ieee.org%2Fiel5%2F5559273%2F5567473%2F05567825.pdf%3Farnumber%3D5567825&authDecision=-203>>. Acesso em: 15 out. 2010.

## APÊNDICE A

### A.1 DEFINIÇÕES

Definição dos termos geológicos utilizados no trabalho e detalhamento das técnicas de projeção estereográfica:

#### A.1.1 FALHAS GEOLÓGICAS

Uma falha geológica caracteriza-se pela região da superfície de uma rocha na qual pode-se observar o deslocamento entre blocos que compõem a rocha, sendo tal deslocamento paralelo à superfície da rocha.

O deslocamento entre os blocos forma um plano, denominado de plano de falha.

#### A.1.2 RUMO

O termo rumo, refere-se mais especificamente ao rumo do mergulho da falha. Indica a direção do mergulho do plano de falha, que é perpendicular à direção do plano de falha. A medida portanto, trata-se do ângulo na horizontal, variando de 0 (zero) a 360 (trezentos e sessenta), entre o norte e a direção do mergulho do plano de falha.

#### A.1.3 MERGULHO

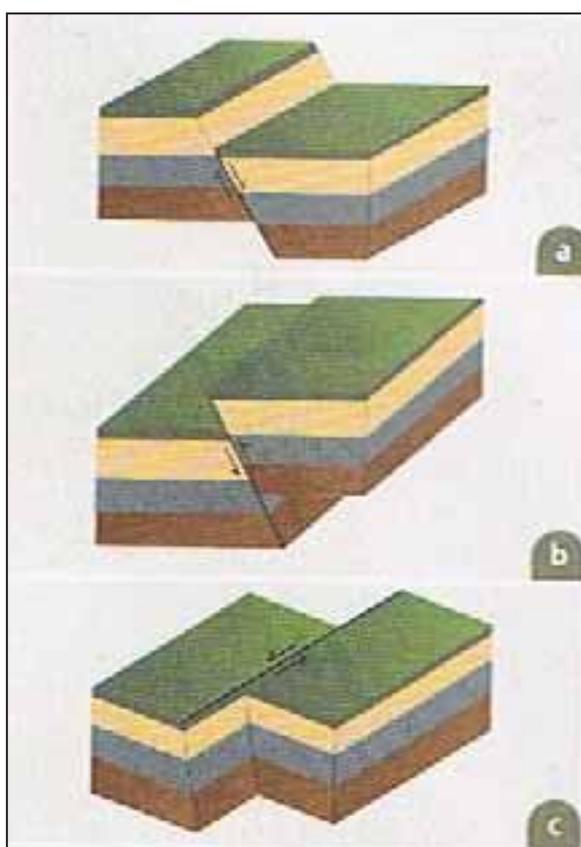
O mergulho é a medida angular, que varia de 0 (zero) a 90 (noventa) graus, entre o plano de falha e um plano horizontal arbitrário. O ângulo de mergulho é medido na mesma direção do rumo.

### A.1.4 ESTRIAS

Em geologia estrutural, estrias são as marcas deixadas na superfície da rocha durante o movimento de deslocamento de blocos. Estas marcas caracterizam-se por ranhuras paralelas entre si que, se analisadas, podem determinar o sentido do movimento

### A.1.5 SENTIDO DO MOVIMENTO

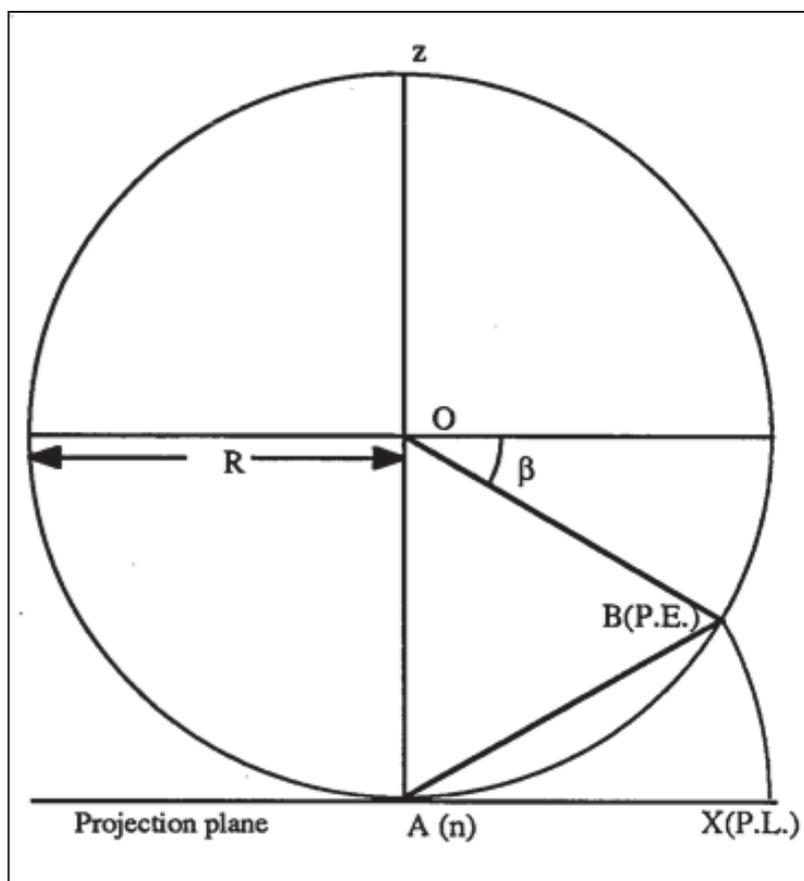
O sentido do movimento indica qual movimento ocorreu entre os blocos de rocha. As falhas podem ser classificadas de acordo com o sentido do movimento. Na **Figura 26** é apresentada a classificação das falhas e os movimentos entre blocos.



**Figura 26:** a) Falha Normal b) Falha Inversa c) Falha Transcorrente. Figura extraída de (TEIXEIRA et al, 2009).

## A.2 PROJEÇÃO ESTEREOGRAFICA

### A.2.1 REDE DE IGUAL ÁREA OU DIAGRAMA DE SCHMIDT-LAMBERT



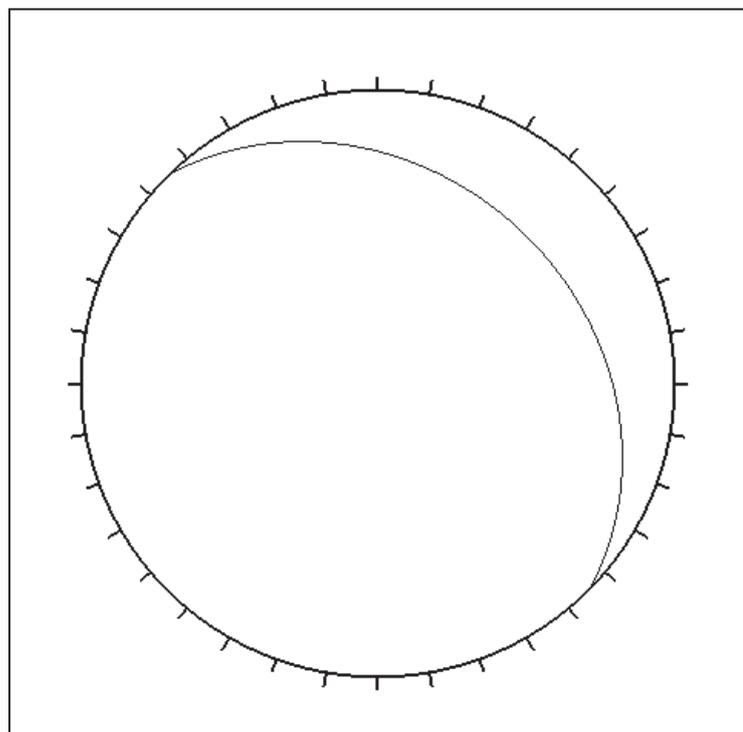
**Figura 27:** Projeção para obter um Diagrama de Igual Área. Figura extraída de (MENDES e KULLBERG, 1992).

```

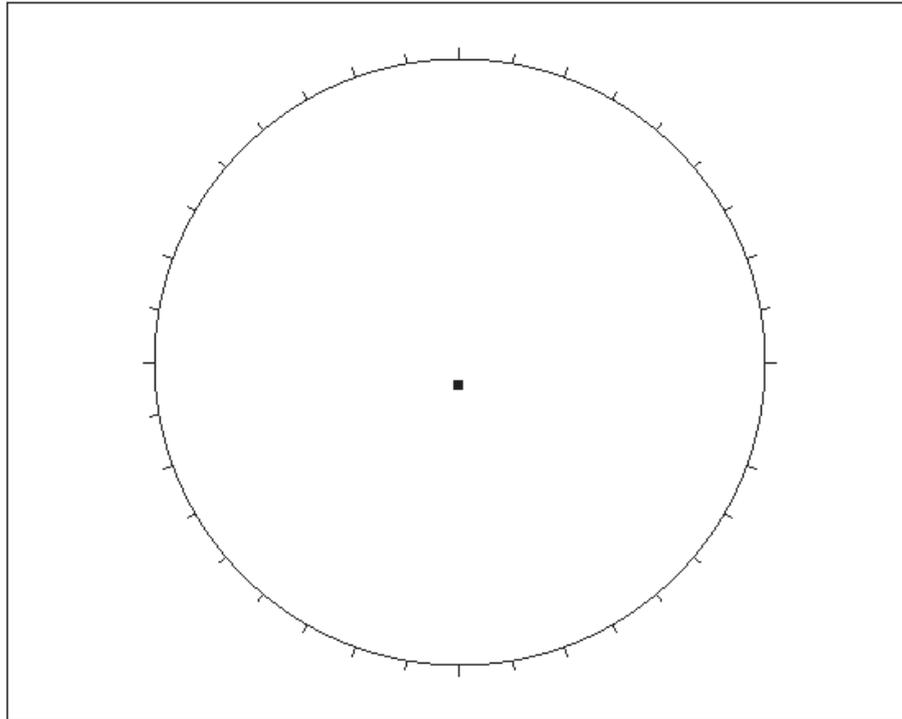
public ListaDEC calcularArcos(int rumo, int mergulho) {
    ListaDEC arco = new ListaDEC();
    rumo = 90 - rumo
    //Calculando coordenadas da curva:
    for (int k = -90; k <= 90; k+=10) {
        Coordenadas coord = new Coordenadas();
        //Plano
        double x = Math.cos(Math.toRadians(k)) * Math.sin(Math.toRadians(90 - mergulho));
        double y = Math.sin(Math.toRadians(k));
        double z = Math.cos(Math.toRadians(k)) * Math.cos(Math.toRadians(90 - mergulho));
        double ptX = x * Math.sqrt(2 / (1 + z));
        double ptY = y * Math.sqrt(2 / (1 + z));
        //Escala para raio
        ptX = (raioDiagrama / 1.41) * ptX;
        ptY = (raioDiagrama / 1.41) * ptY;
        if (ptY <= 7.03 && ptY >= -7.03) {
            coord.setX(((ptX * Math.cos(Math.toRadians(rumo))) - (ptY * Math.sin(Math.toRadians(rumo)))));
            coord.setY(((ptX * Math.sin(Math.toRadians(rumo))) + (ptY * Math.cos(Math.toRadians(rumo)))));
        }
        arco.add(coord);
    }
    return arco;
}

```

**Figura 28:** Função que calcula as coordenadas de uma curva que representa um plano de falha, dados o rum e o mergulho do plano.



**Figura 29:** Curva no diagrama representando um plano com rumo de 45 graus e mergulho de 30 graus.



**Figura 30:** Ponto no diagrama, representando o pólo de um plano de falha.

```

public Coordenadas calcularPolos(int rumo, int mergulho, String tipo) {
    Coordenadas polos = new Coordenadas();
    rumo = rumo - 180;
    if (tipo.equals(PLANOS)) {
        double rc = raioDiagrama * Math.sqrt(2.0) * Math.sin(Math.toRadians(mergulho / 2));
        polos.setX((rc * Math.sin(Math.toRadians(rumo))));
        polos.setY((rc * Math.cos(Math.toRadians(rumo))));
    }
    if (tipo.equals(LINEACOES)) {
        rumo = rumo + 180;
        double rc = raioDiagrama * Math.sqrt(2.0) * Math.sin(Math.toRadians(45.0 - mergulho / 2));
        polos.setX((rc * Math.sin(Math.toRadians(rumo))));
        polos.setY((rc * Math.cos(Math.toRadians(rumo))));
    }
    return polos;
}

```

**Figura 31:** Função que calcula as coordenadas de um pólo, dado o rumo e mergulho de um plano de falha.

## A.2.2 MÉTODO DOS DIEDROS RETOS

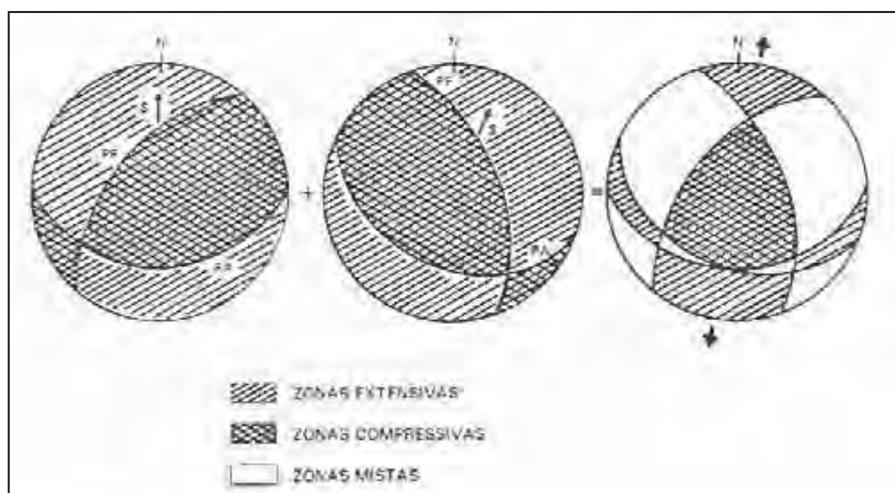
Este método foi desenvolvido por Angelier e Mechler em 1977, e é utilizado para determinar as direções dos eixos de um elipsóide que representa as principais forças exercidas sobre um corpo rochoso. Estas forças causaram deformação do corpo e também a formação das diversas falhas analisadas.

Para determinar as direções, o método segue uma seqüência de combinação: um plano de falha é representado no diagrama através de uma curva; uma outra curva, representando o rumo e o mergulho das estrias da mesma falha também é representada. Estas duas curvas dividem o diagrama em quatro espaços (diedros).

Para cada falha em uma estrutura, é feita a mesma representação. Em seguida, a cada par de representação de plano e estria, os diedros são sobrepostos, e de acordo com o sentido do movimento, as intersecções de cada região são classificadas como compressivas ou extensivas. Ao resultado, é sobreposta a terceira representação, novamente classificados os novos espaços e assim procede-se com todas as falhas.

Ao final do processo, os espaços destacados e classificados tendem a uma região pontual, com a qual se pode determinar o rumo e mergulho dos eixos do elipsóide que representa as forças.

Na **Figura 32** é apresentado um esboço do procedimento.



**Figura 32:** Esquema do procedimento do método dos diedros retos.