

**UNIVERSIDADE FEDERAL DE CAMPINA GRANDE**  
**CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA**  
**DEPARTAMENTO DE SISTEMAS E COMPUTAÇÃO**

**RELATÓRIO DE ESTÁGIO**

**DESENVOLVIMENTO DO MÓDULO CADASTRO DE  
PROCEDIMENTOS E SUPORTE TÉCNICO EM  
INFORMÁTICA**

**FELIPE BARBOSA ARAÚJO RAMOS**  
Estagiário

**TIAGO LIMA MASSONI**  
Orientador Acadêmico

**UIRÁ VASCONSELOS ALENCAR DE ASSIS**  
Supervisor Técnico

**Campina Grande – PB**

**01 de Julho de 2011**

**DESENVOLVIMENTO DO MÓDULO CADASTRO DE PROCEDIMENTOS E  
SUPORTE TÉCNICO EM INFORMÁTICA**

**APROVADO EM \_\_\_\_\_**

**BANCA EXAMINADORA**

---

Prof. Dr. Tiago Lima Massoni

ORIENTADOR ACADÊMICO

---

Prof<sup>a</sup> Dr. Joseana Macêdo Fechine

MEMBRO DA BANCA

---

Prof<sup>a</sup> Joseluze de Farias Cunha

MEMBRO DA BANCA



Biblioteca Setorial do CDSA. Maio de 2021.

Sumé - PB

## **AGRADECIMENTOS**

Agradeço primeiramente a Deus por me dar saúde, paz e todas as condições possíveis para exercer as tarefas do dia-a-dia.

Aos meus pais Adeziva Barbosa e Fernando Ramos que sempre me apoiaram e incentivaram a seguir os estudos, pois sem eles não estaria aqui.

Ao orientador acadêmico Prof. Dr. Tiago Lima Massoni, que me auxiliou e me apoiou durante o período do estágio.

Ao meu irmão Fernando pela paciência e compreensão que demonstrou durante todos esses meses.

A Alexandre, Luiz e Lenin, pelo apoio e dedicação, os quais foram fundamentais no desenvolvimento do trabalho.

Por fim, não posso deixar de agradecer ao Ministério Público da Paraíba (MPPB) pela grande oportunidade de colocar em prática conhecimentos adquiridos na graduação para auxiliar seus funcionários e participar da equipe de desenvolvimento da instituição.

## APRESENTAÇÃO

Como parte das exigências do curso de Ciência da Computação, da Universidade Federal de Campina Grande, para cumprimento da disciplina de estágio integrado, apresenta-se o relatório de estágio, DESENVOLVIMENTO DO MÓDULO CADASTRO DE PROCEDIMENTOS E SUPORTE TÉCNICO EM INFORMÁTICA.

O estágio foi realizado no Ministério Público do Estado da Paraíba, 2º Centro de Apoio Operacional às Curadorias. Rua Promotora Terezinha Lopes de Moura S/N, Liberdade, 58.410-064, Campina Grande, Paraíba.

O conteúdo do relatório está distribuído conforme descrição a seguir:

Seção 1 – Introdução.

Seção 2 – Ambiente de Estágio.

Seção 3 – Fundamentação Teórica e Tecnologias Utilizadas.

Seção 4 – Atividades do Estágio.

Seção 5 – Considerações Finais.

Referências Bibliográficas.

Apêndices.

## RESUMO

Neste relatório estão descritas as atividades de estágio realizadas no Ministério Público da Paraíba entre os meses de março e junho de 2011. Sendo estas atividades divididas entre planejadas e não planejadas. Dentre as não planejadas destacam-se o suporte técnico ao usuário, orientação na utilização de recursos tecnológicos da instituição, auxílio nas atividades relativas à instalação, atualização, customização e manutenção dos sistemas utilizados, suporte a rede, microcomputadores e periféricos em geral. Por outro lado, foram realizadas tarefas planejadas, que compreendem as atividades de desenvolvimento do módulo 3 (Cadastro de Procedimentos) que faz parte do sistema Sislegal. Sendo este uma das ferramentas utilizadas pelo projeto Nomelegal que visa reduzir o número de crianças sem o reconhecimento de paternidade em seu registro de nascimento, por meio do levantamento, em cada comarca, das pessoas que têm seu registro civil incompleto, notificando os respectivos responsáveis (geralmente as mães) para esclarecimento acerca das medidas que poderão ser utilizadas para a regularização da filiação. Assim, o sistema Sislegal é composto pelos seguintes módulos: cadastro das escolas existentes no município (particulares, municipais, estaduais, federais), de todos os níveis de ensino, cadastro da relação dos alunos sem maternidade ou sem paternidade reconhecida ou sem registro; cadastro da relação de procedimentos por escola originados a partir dos cadastros realizados o item anterior; registro das notificações expedidas para marcação das oitivas (audição, ato de ouvir as testemunhas ou as partes de um processo judicial); Módulo gerencial com a emissão de relatórios de casos resolvidos (paternidade aceita), casos não resolvidos e seus motivos, relação de escolas e relação de alunos.

# SUMÁRIO

## Conteúdo

AGRADECIMENTOS .....	3
APRESENTAÇÃO .....	4
RESUMO .....	5
SUMÁRIO .....	6
LISTA DE SIGLAS E ABREVIATURAS .....	8
LISTA DE FIGURAS .....	9
LISTA DE QUADROS .....	10
1 Introdução .....	12
1.1 Delimitação do Estágio .....	12
1.2 Objetivos .....	12
1.2.1 Objetivo Geral .....	12
1.2.2 Objetivos Específicos .....	13
2. Ambiente de Estágio .....	15
2.1 Identificação da Empresa .....	15
2.2 Infraestrutura de <i>Hardware</i> e <i>Software</i> .....	16
2.3 Dados do Orientador e Supervisor.....	16
3. Fundamentação Teórica .....	19
3.1 <i>Java Server Faces (JSF)</i> .....	19
3.2 <i>Hibernate</i> .....	20
3.3 <i>Spring</i> .....	22
3.3.1 Módulos .....	23
3.3.1.1 <i>Spring Core</i> .....	24
3.3.1.2 <i>Spring DAO</i> .....	24
3.3.1.3 <i>Spring ORM</i> .....	25
3.3.1.4 <i>Spring AOP</i> .....	25

3.3.1.5 <i>Spring Context</i> .....	25
3.3.1.6 <i>Spring Web</i> .....	25
3.3.1.7 <i>Spring Web MVC</i> .....	25
4. Atividades Realizadas .....	27
4.1 Cronograma .....	27
4.2 Atividades de Desenvolvimento .....	28
4.2.1 Estudo das Tecnologias .....	30
4.2.2 Configuração do Ambiente de Desenvolvimento.....	31
4.2.3 Desenvolvimento de um Protótipo.....	31
4.2.4 Desenvolvimento do Módulo Cadastro de Procedimento.....	31
4.2.5 Testes .....	34
4.3 Atividades de Suporte Técnico .....	34
4.3.1 Suporte Técnico ao Usuário .....	34
4.3.2 Instalação, Atualização, Customização e Manutenção de <i>Softwares</i> .....	35
4.3.3 Manutenção de Impressoras e Cartuchos.....	35
4.3.4 Manutenção da Rede .....	36
5. Considerações Finais .....	38
Referências Bibliográficas .....	40
APÊNDICE A - PLANO DE ESTÁGIO .....	42

## LISTA DE SIGLAS E ABREVIATURAS

<b>MPPB</b>	MINISTÉRIO PÚBLICO DA PARAÍBA
<b>JSF</b>	<i>JAVA SERVER FACES</i>
<b>HTML</b>	HYPERTEXT MARKUP LANGUAGE
<b>UI</b>	<i>USER INTERFACE</i>
<b>JSP</b>	JAVA SERVER PAGES
<b>SQL</b>	<i>STRUCTURED QUERY LANGUAGE</i>
<b>HQL</b>	HIBERNATE QUERY LANGUAGE
<b>JDBC</b>	<i>JAVA DATABASE CONNECTIVITY</i>
<b>MVC</b>	MODEL-VIEW-CONTROLER
<b>FreeMaker</b>	<i>FERRAMENTA PARA GERAÇÃO DE TEXTO</i>
<b>JNDI</b>	JAVA NAMING AND DIRECTORY INTERFACE
<b>XP</b>	<i>EXTREME PROGRAMMING</i>
<b>SGBD</b>	SISTEMA DE GERENCIAMENTO DE BANCO DE DADOS
<b>SVN</b>	SUBVERSION

## LISTA DE FIGURAS

Figura 01	Organograma Administrativo do Setor de Informática	15
Figura 02	Regras de Navegação <i>JSF</i>	20
Figura 03	Classe <i>Java</i>	21
Figura 04	Conexão com o Banco de Dados	21
Figura 05	Mapeando Elementos	22
Figura 06	Utilizando <i>Hibernate</i>	22
Figura 07	Application Context	23
Figura 08	Módulos do <i>Spring</i>	24
Figura 09	Arquitetura <i>MVC</i> em Aplicações <i>Web</i>	29
Figura 10	Diagrama de Caso de Uso	29
Figura 11	Tela de <i>Login</i>	32
Figura 12	Tela Inicial	32
Figura 13	Página Principal de Procedimento	33
Figura 14	Tela Cadastrar Procedimento	33

## LISTA DE QUADROS

Quadro 01	Descrição das atividades e do tempo gasto	27
-----------	---	----

SEÇÃO I

INTRODUÇÃO

# **1. INTRODUÇÃO**

A participação de um graduando em um estágio é de fundamental importância na sua formação, pois esta é a oportunidade de colocar em prática os conhecimentos adquiridos na universidade, além de estar engajado em um ambiente de trabalho real possibilitando assim uma maior experiência ao ingressar no mercado de trabalho.

## **1.1 Delimitação do Estágio**

O estágio foi realizado no Ministério Público do Estado da Paraíba, 2º Centro de Apoio Operacional às Curadorias. Rua Promotora Terezinha Lopes de Moura S/N, Liberdade, 58.410-064, Campina Grande, Paraíba. As atividades foram realizadas entre os meses de março e junho de 2011, sendo a carga horária de 25 horas semanais. Assim, atingiu-se as 300 horas requeridas na disciplina.

## **1.2 Objetivos**

### **1.2.1 Objetivo Geral**

Desenvolver o módulo de cadastro da relação de procedimentos por escola originados que faz parte do sistema SisLegal, que visa a criação de um cadastro para as informações de nomes de escolas, diretores e relação de alunos e mães de alunos com os registros dos alunos e seus pais para o projeto Nome Legal (Planejamento estratégico do Ministério Público da Paraíba 2010-2016). Além de realizar o suporte técnico ao usuário, auxiliando os funcionários da instituição.

### 1.2.2 Objetivos Específicos

- Realizar um estudo das tecnologias utilizadas.
- Entender a arquitetura *Model-view-controler (MVC)* do sistema em que o módulo está inserido.
- Efetuar o desenvolvimento de um exemplo para aplicar e validar o estudo das tecnologias.
- Desenvolver o módulo de cadastro da relação de procedimentos que inclui o desenvolvimento da interface *WEB*, da lógica de negócio e criação das tabelas necessárias no banco de dados.

SEÇÃO II

# AMBIENTE DE ESTÁGIO

## 2 AMBIENTE DE ESTÁGIO

### 2.1 Identificação da Empresa

O Ministério Público da Paraíba é uma instituição do âmbito jurídico que está incumbido a defesa da ordem jurídica, do regime democrático e dos interesses sociais e individuais indisponíveis.

O organograma administrativo do setor de informática do MPPB está ilustrado na Figura 01.



**Figura 01 – Organograma Administrativo do Setor de Informática.**

A Diretoria de Tecnologia da Informação (DITEC) é composta pelo Departamento de Suporte e Redes (DESUR) e pelo Departamento de Desenvolvimento de Sistemas (DESEN), chefiados respectivamente por, Uirá Alencar e Jefferson Barbosa.

## **2.2 Infraestrutura de *Hardware* e *Software***

A sala onde foram realizadas as atividades de estágio está equipada com três computadores. Porém, as configurações de tais máquinas não eram suficientes para o uso das ferramentas utilizadas no desenvolvimento. Por isso, uma máquina pessoal foi utilizada, que tinha como configurações, processador Intel *Core 2 Duo*, HD de 120 GB e 2GB de memória RAM.

No âmbito de infraestrutura de *software* foi usada uma distribuição *linux*, já configurada com todas as ferramentas necessárias para o desenvolvimento do sistema.

## **2.3 Dados do Orientador e Supervisor**

### **Orientador Acadêmico**

**Nome:** Phd. Tiago Lima Massoni

**Endereço:** Departamento de Sistemas e Computação

Universidade Federal de Campina Grande

Avenida Aprígio Veloso, 882 – CEP: 58.109-970

Bodocongó, Campina Grande, PB – Brasil.

**Email:** tiagomassoni@gmail.com

### **Supervisor Técnico**

**Nome:** Uirá Vasconcelos Alencar de Assis

(Chefe de Suporte Técnico do Ministério Público da Paraíba)

**Endereço:** Rua Promotora Terezinha Lopes de Moura, S/N

CEP: 58.410-064

Liberdade, Campina Grande, PB – Brasil.

**Email:** [uira@mp.pb.gov.br](mailto:uira@mp.pb.gov.br)

SEÇÃO III

FUNDAMENTAÇÃO

TEÓRICA

## 3 Fundamentação Teórica

Nesta seção, serão abordados os conceitos da área de computação que foram utilizados para o desenvolvimento do sistema, bem como as tecnologias que foram necessárias durante a realização do estágio.

### 3.1 *Java Server Faces (JSF)*

É um *framework MVC* que possibilita o desenvolvimento de aplicações *web* de forma ágil [SAUVÉ, 2011]. É composto por 5 (cinco) módulos: componentes, eventos, validação e conversões, navegabilidade e *backbeans* [SAUVÉ, 2011].

Com o uso de *JSF* a parte da aplicação voltada para aparência do *software* é feita em *HyperText Markup Language (HTML)*, representada pelas páginas *web* [PITANGA, 2011]. O negócio é implementado em *Java*, cujos códigos estão nos *beans* [PITANGA, 2011].

*JavaServer Faces* oferece ganhos no desenvolvimento de aplicações *web* por diversos motivos:

- Permite que o desenvolvedor crie *UIs* através de um conjunto de componentes *UIs* predefinidos;
- Fornece um conjunto de *tags JavaServer Pages (JSP)* para acessar os componentes;
- Reusa componentes da página;
- Associa os eventos do lado cliente com os manipuladores dos eventos do lado servidor (os componentes de entrada possuem um valor local representando o estado no lado servidor);
- Fornece separação de funções que envolvem a construção de aplicações *web*.

São exemplificadas na Figura 02 regras de navegação *JSF*.

```
<a4j:commandButton value="Adicionar" action="#{usuarioMBean.salvar}"
rendered="#{usuarioMBean.estadoIncluir}" />
<a4j:commandButton value="Salvar" action="#{usuarioMBean.salvar}"
rendered="#{usuarioMBean.estadoEditar}" />
```

Figura 02 – Regras de Navegação *JSF*.

### 3.2 *Hibernate*

O *Hibernate* é um *framework* para o mapeamento objeto-relacional escrito na linguagem *Java* [JAVAFREE.UOL, 2011].

O *Hibernate* tem como objetivo diminuir a complexidade entre os programas *Java*, baseado no paradigma orientado a objeto, que precisam trabalhar com um banco de dados do modelo relacional. Em especial, no desenvolvimento de consultas e atualizações dos dados.

Sua principal característica é a transformação das classes em *Java* para tabelas de dados (e dos tipos de *Java* para os dos Sistemas de Gerenciamento de Banco de Dados escolhido). O *Hibernate* gera as chamadas *Structured Query Language* (SQL) e libera o desenvolvedor do trabalho manual da conversão dos dados resultante, mantendo a portabilidade do programa para quaisquer bancos de dados SQL, porém causando um pequeno aumento no tempo de execução. O *Hibernate* possui a sua própria linguagem de SQL chamada *HQL*, que é convertida para SQLs específicas de cada banco de dados, característica que faz dele um dos principais *frameworks* para independência de banco de dados [JAVAFREE.UOL, 2011].

Como as suas principais características já foram explicitadas anteriormente, serão expostos exemplos do funcionamento deste *framework* ilustrados nas figuras 03, 04, 05 e 06.

Na Figura 03, é ilustrada uma classe *Java* que contém dois atributos (linhas 2 e 3) que devem ser armazenados em tabelas de um banco de dados.

```

1. public class Usuario {
2.     private String usCod;
3.     private String usNome;
4.     public Usuario(){
5.     }
6.     public Usuario(String usCod, String usNome) {
7.         this.setUsCod(usCod);
8.         this.setUsNome(usNome);
9.     }
10. //... IMPORTANTE: colocar os métodos GETs e SETs...
11. }

```

**Figura 03 – Classe Java.**

Na Figura 04, pode-se observar a classe UsuarioDAO responsável pela conexão com a base de dados.

```

import ...
public class UsuarioDAO{
    private SessionFactory factory;

    public UsuarioDAO() throws Exception{
        factory = new
Configuration().addClass(Usuario.class).buildSessionFactory();
    }
    public void UsInserir(Usuario us) throws Exception {
        Session session = factory.openSession();
        session.save(us);
        session.flush();
        session.close();
    }

    public void UsAlterar(Usuario us) throws Exception {
        ...
    }
    public void UsExcluir(Usuario us) throws Exception {
        ...
    }
}

```

**Figura 04 – Conexão com o Banco de Dados.**

Na Figura 05, o elemento “*class*” (linha 5) mapeia uma classe para uma tabela. O elemento “*id*” (linha 6) mapeia um atributo da classe à chave primária da tabela (o *Hibernate* obriga o uso de *id* para todas as tabelas que refletem entidades do modelo Orientado a Objeto). O elemento “*property*” (linha 9) mapeia os atributos da classe para colunas da tabela.

```

1. <?xml version="1.0"?>
2. <!DOCTYPE hibernate-mapping PUBLIC "-//Hibernate Mapping DTD 3.0//EN"
3. "hibernate-mapping-3.0.dtd">
4. <hibernate-mapping>
5.   <class name="Usuario" table="tb_usuarios">
6.     <id name="UsCod" column="USCOD" type="string">
7.       <generator class="assigned"/>
8.     </id>
9.     <property name="UsNome" column="USNOME" type="string"/>
10.   </class>
11. </hibernate-mapping>

```

**Figura 05 – Mapeando Elementos.**

Por fim, na Figura 06 é apresentada a inserção de um registro em uma base de dados utilizando o *hibernate*.

```

...
        String log = "login";
        String nome = "Felipe";

        UsuarioDAO dao = new UsuarioDAO();
        Usuario usuario = new Usuario(log,nome);
        dao.usInserir(usuario);
        System.out.println("Registro inserido!");
...

```

**Figura 06 – Utilizando *Hibernate*.**

### 3.3 *Spring*

O *Spring* é um *framework open source* para a plataforma *Java*. Tem como princípio fundamental o conceito de Injeção de Dependências que torna uma simples biblioteca de classes diferente de um *framework* [CALÇADO, 2011].

A injeção de dependência é um padrão de desenvolvimento que visa desacoplar os componentes da aplicação [CARVALHO, 2006]. Os componentes são instanciados externamente a classe [CARVALHO, 2006]. Um gerenciador controla essas instâncias. Os componentes têm dependências entre si. Esse gerenciador, a partir de uma configuração, liga os componentes de forma a montar a aplicação.

A Injeção de dependência se relaciona com o padrão Inversão de Controle, mas não pode ser considerado um sinônimo deste, já que o segundo é um padrão de desenvolvimento de programas de computadores onde a seqüência (controle) de chamadas dos métodos é invertida em relação à programação tradicional, ou seja, ela não é determinada diretamente pelo programador [CARVALHO, 2006]. Este controle é delegado a uma infraestrutura de *software* muitas vezes chamada de *container* ou a qualquer outro componente que possa tomar controle sobre a execução [CARVALHO, 2006]. Esta é uma característica muito comum a alguns *frameworks*.

O *applicationContext.xml*, ilustrado na Figura 07, é onde são declarados os *beans* do *Spring*. O *framework* chama de *beans* todas as classes que gerencia. As classes precisam ser declaradas utilizando o elemento *<bean>*, seguindo o formato:

```
<bean id="identificador do bean" class="FQN da classe que implementa  
o bean" >  
    <property name="nome do atributo" ref="id do bean que satisfaz  
a dependência" />  
</bean>
```

Figura 07 – *Application Context*.

### 3.3.1 Módulos

A estrutura do *Spring* está apresentada na Figura 08.

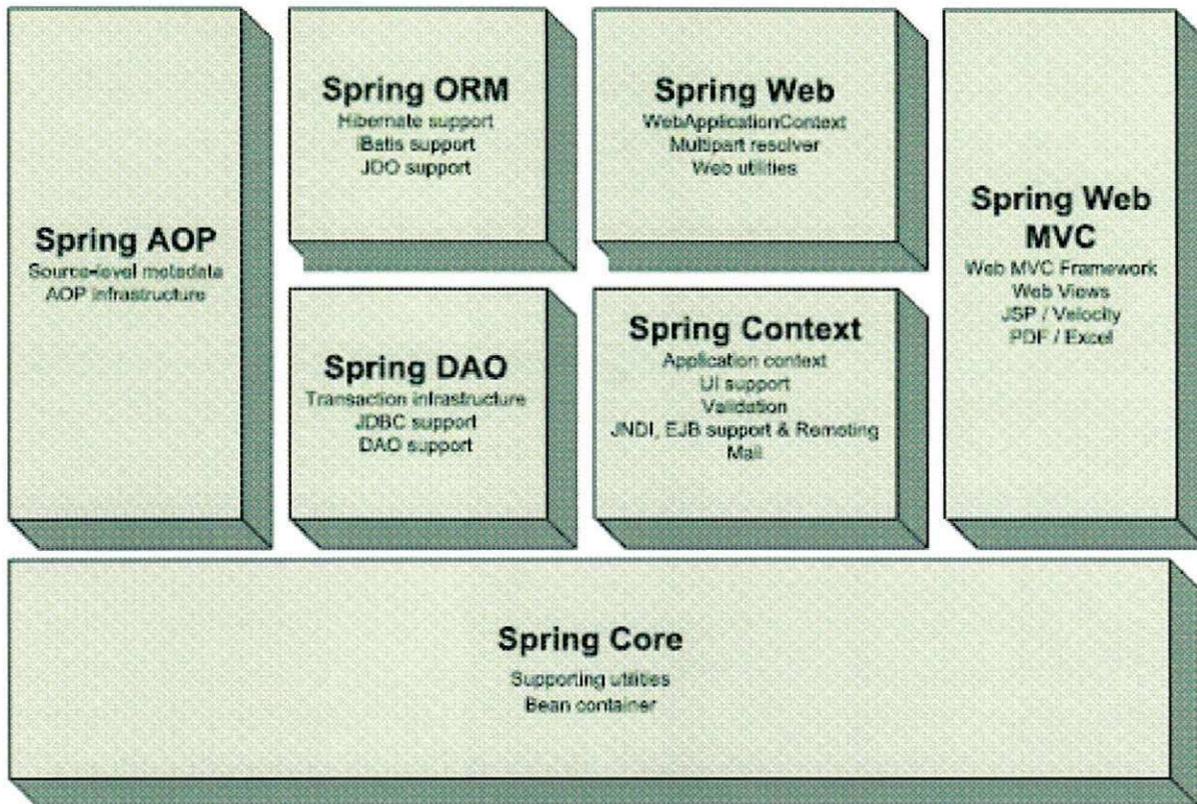


Figura 08 – Módulos do *Spring*.

### 3.3.1.1 *Spring Core*

Possui como elemento principal o *BeanFactory*, que é uma implementação do padrão *Factory*, que aplica Inversão de Controle para diminuir o acoplamento entre a especificação de dependências e o código da aplicação [CARVALHO, 2006].

### 3.3.1.2 *Spring DAO*

Elimina grande parte da codificação necessária para comunicação com o banco de dados [CARVALHO, 2006].

### **3.3.1.3 Spring ORM**

Permite a integração do *Spring* com *frameworks* para persistência de objetos, como o *Hibernate* [CARVALHO, 2006].

### **3.3.1.4 Spring AOP**

Provê suporte a uma implementação Orientada a Aspectos, possibilitando a programação de metadados que permite adicionar anotações no seu código fonte para especificar onde e como aplicar aspectos, como por exemplo, a definição de *pointcuts* [CARVALHO, 2006].

### **3.3.1.5 Spring Context**

Esse módulo aumenta o conceito de *BeanFactory*, adicionando eventos de ciclos de vida e de validação, além de fornecer serviços como integração com EJB, acesso JNDI e permitir o suporte para integração com outros *frameworks*, como por exemplo o *FreeMaker* [CARVALHO, 2006].

### **3.3.1.6 Spring Web**

Provê funcionalidades específicas para projetos *web*, como por exemplo, facilita as tarefas de manipulação de múltiplas requisições e vincula parâmetros de requisição para objetos de domínio. Permite integração com *Jakarta Struts* e *JSF* [CARVALHO, 2006].

### **3.3.1.7 Spring Web MVC**

O *Spring* pode ser facilmente integrado com outros *frameworks MVC*, mas esse módulo fornece uma implementação de *framework web* que utiliza inversão de controle para separar o controlador da lógica dos objetos de negócio [CARVALHO, 2006].

## SEÇÃO IV

ATIVIDADES

REALIZADAS

## 4 Atividades Realizadas

Nesta seção, serão detalhadas as atividades realizadas durante o período de estágio.

### 4.1 Cronograma

O cronograma proposto no Plano de Estágio, Apêndice A, foi seguido parcialmente. Mudanças no âmbito de metodologia de desenvolvimento se mostraram necessárias em virtude da indisponibilidade do gerente do projeto, Jefferson Barbosa, que é um membro importante para o desenvolvimento do sistema.

Após reunião com o chefe do departamento de desenvolvimento do MPPB observou-se que algumas atividades que fazem parte da metodologia de desenvolvimento *Scrum* [SCHWABER, 2004], que tinha sido definida no plano de estágio (Apêndice A), não poderiam ser seguidas. Assim, foi proposto pelos estagiários a mudança para a metodologia *eXtreme Programming (XP)* [PAPO, 2011], que é uma metodologia ágil própria para equipes pequenas e médias e que irão desenvolver *software* com requisitos vagos e em constante mudança [PAPO, 2011].

Uma síntese das atividades realizadas e do tempo demandado para sua execução está exposto no Quadro 1.

Quadro 1 - Descrição das atividades e do tempo gasto.

Atividade	Descrição da atividade	Tempo
1	Estudo da arquitetura do sistema e das tecnologias envolvidas ( <i>JSF, Hibernate e Spring</i> )	50h
2	Configuração do ambiente <i>Linux, Eclipse, TomCat</i> e do controle de versão	10h
3	Desenvolvimento do protótipo utilizando as tecnologias	25h

4	Desenvolvimento do Módulo Cadastro de Procedimentos	50h
5	Integração dos Módulos	25h
6	Elaboração de testes para o módulo Cadastro de Escolas	30h
7	Suporte técnico	80h
8	Elaboração do relatório	30h
Total		300h

## 4.2 Atividades de Desenvolvimento

Como o setor de desenvolvimento do MPPB está alocado no pólo de João Pessoa, foi necessário entrar em contato com a equipe da cidade, em especial com o chefe do setor.

Após um primeiro contato por telefone foi marcada uma reunião para definir em qual projeto os estagiários seriam alocados, como também quais tecnologias seriam usadas no desenvolvimento do mesmo.

Assim, ficou definido que a metodologia mais adequada ao perfil da equipe seria o *XP*, que é própria para equipes de menor porte [PAPO, 2011]. Já as principais tecnologias envolvidas no desenvolvimento do módulo Cadastro de Procedimentos foram determinadas como sendo *JSF*, *Hibernate* e *Spring*.

Também foi feito o *download* da arquitetura do projeto, por meio de *checkout* no *SVN*. Na Figura 09, está ilustrada a arquitetura *MVC* em aplicações *web*, sendo *model* a camada que contém a lógica da aplicação, *view* a camada de apresentação com o usuário e *controller* a camada de intermédio entre as duas camadas anteriores.

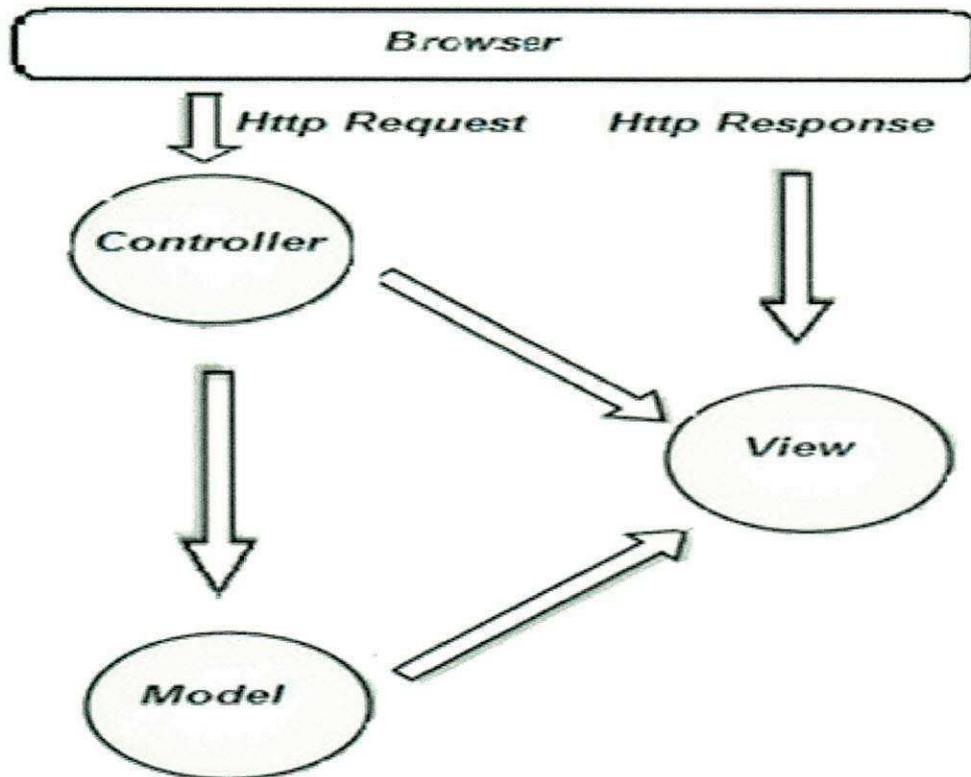


Figura 09 – Arquitetura MVC em Aplicações Web.

Alguns requisitos funcionais foram extraídos a partir de um diagrama de casos de uso exposto pelo gerente do projeto, que está ilustrado na Figura 10.

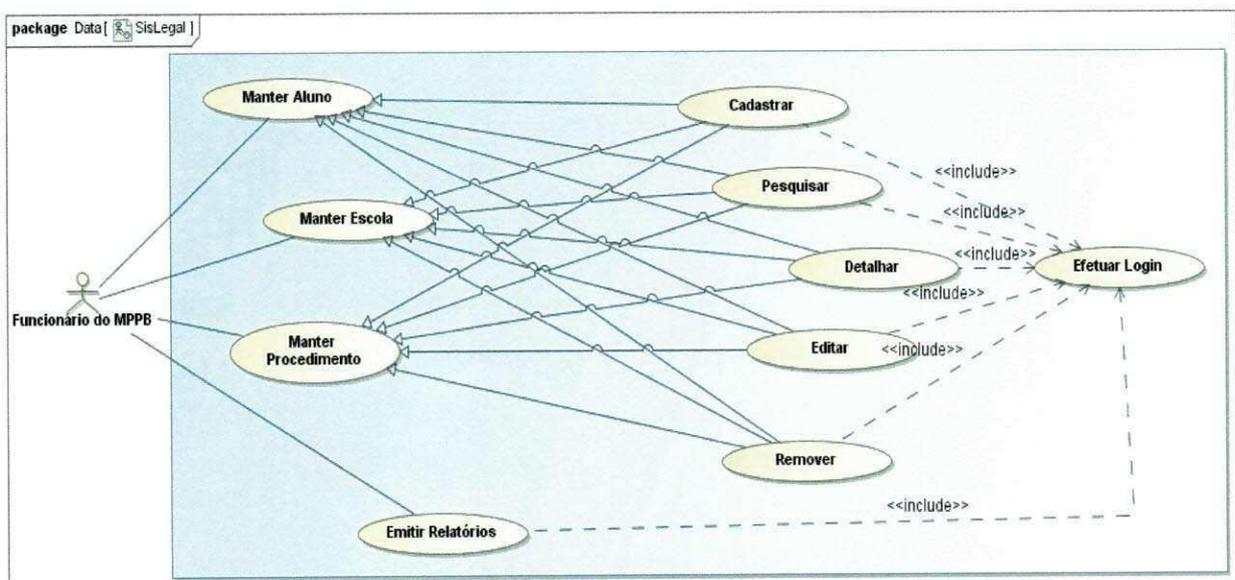


Figura 10 – Diagrama de Casos de Uso.

Os requisitos funcionais definidos estão elucidados a seguir.

- Criação/Cadastramento de um novo Procedimento: Deve ser disponibilizado um formulário de cadastro, este deve conter campos, a serem preenchidos;
- Edição: deve ser possível alterar as informações dos formulários já salvos no banco;
- Remoção: Deve ser possível a exclusão de formulários salvos;
- Detalhamento: Todas as informações do formulário devem ser visualizadas;
- Pesquisa: Deve ser possível pesquisar procedimentos.

Os requisitos não-funcionais se resumiram em:

- Ferramenta *Web*;
- As ferramentas utilizadas devem ser gratuitas;
- O código deve ser escrito em *Java*;
- O SGBD *Postgres* deveria ser utilizado;
- A interface deveria seguir o mesmo padrão dos demais sistemas do MPPB.

#### **4.2.1 Estudo das Tecnologias**

Antes de iniciar o desenvolvimento do módulo Cadastro de Procedimentos, foi preciso realizar o estudo das tecnologias utilizadas na implementação do projeto. Esta fase mostrou-se muito importante, pois possibilitou a adesão de conhecimentos novos, que somados aos adquiridos na graduação tornou possível as atividades de desenvolvimento.

## **4.2.2 Configuração do Ambiente de Desenvolvimento**

As atividades de desenvolvimento do sistema Sislegal demandaram a instalação de algumas aplicações nas máquinas que foram usadas para a execução das tarefas.

Como os computadores disponíveis na sala dos estagiários não continham configurações compatíveis com as exigidas pelas ferramentas usadas, foi necessário o uso de máquinas pessoais.

Os aplicativos utilizados foram *Eclipse IDE*, *Apache Tomcat*, *SGBD Postgres* e *SVN* para controle de versão.

## **4.2.3 Desenvolvimento do Protótipo**

Para validar o estudo das tecnologias e aumentar o grau de familiaridade com a arquitetura utilizada pelo Departamento de Desenvolvimento do MPPB, foi realizada a implementação de um protótipo com funcionalidades simples (adicionar, editar, excluir, detalhar), mas que seguia todo o padrão de desenvolvimento da instituição. Tornando mais fácil o início do desenvolvimento do módulo Cadastro de Procedimentos.

## **4.2.4 Desenvolvimento do Módulo Cadastro de Procedimento**

A seguir, serão descritas as atividades realizadas no desenvolvimento do módulo Cadastro de Procedimentos. A metodologia *XP* foi utilizada durante o período de implementação. Suas principais práticas foram: desenvolver *software* de forma incremental, expor código a todos os membros da equipe, manter o sistema simples, investir tempo em testes automatizados, interação entre os membros da equipe.

## Página de Login

Ao acessar o sistema, uma página é apresentada com os campos para usuário e senha que permitem a autenticação. Essa tela já havia sido implementada pelos desenvolvedores do Ministério Público e está ilustrada na Figura 11.

Ministério Público do Estado da Paraíba  
Procuradoria-Geral de Justiça

SisLegal

Login

Login da rede:

Senha:

Entrar

Diretoria de Tecnologia da Informação  
Departamento de Desenvolvimento de Sistemas

Figura 11 – Tela de Login.

## Página Inicial

Após a autenticação, o usuário terá acesso à página inicial do sistema, Figura 12. No menu ao lado esquerdo estão as opções de navegação.

Ministério Público do Estado da Paraíba  
Procuradoria-Geral de Justiça

SisLegal

Bem-vindo, Usuario Desenvolvimento! [sair]

Início      Página inicial

Aluno

Escola

Procedimento

Diretoria de Tecnologia da Informação  
Departamento de Desenvolvimento de Sistemas

Figura 12 – Tela Inicial.

## Página Principal de Procedimento

Pode-se observar na Figura 13, que esta tela permite pesquisar um procedimento por meio do seu tipo (Assinatura de TAC, Reunião, Oitiva), data e promotor responsável. Possibilitando mudanças posteriores. É possível

também clicar no botão Novo Procedimento para adicionar um procedimento novo.

The screenshot shows the 'Pesquisar procedimento' (Search procedure) page in the SisLegal system. The header includes the logo of the Ministério Público do Estado da Paraíba, the text 'Ministério Público do Estado da Paraíba Procuradoria-Geral de Justiça', and the 'SisLegal' logo. A user greeting 'Bem-vindo, Usuario Desenvolvimento!' is visible in the top right. On the left, a navigation menu lists 'Início', 'Aluno', 'Escola', and 'Procedimento'. The main content area is titled 'Pesquisar procedimento' and contains a search form with the following fields: 'Informe um parâmetro de pesquisa', 'Tipo do procedimento:' (dropdown menu with 'Oitiva' selected), 'A partir da data:' (calendar icon with 'Mai 3, 2011' selected), 'Até a data:' (calendar icon with 'Jun 14, 2011' selected), and 'Promotor responsável:' (text input with 'Dr. Dmitri' entered). Below the form are buttons for 'Pesquisar' and 'Novo procedimento'. The search results area displays 'Nenhum registro encontrado' (No records found). At the bottom of the page, the footer reads 'Diretoria de Tecnologia da Informação Departamento de Desenvolvimento de Sistemas'.

Figura 13 – Página principal de Procedimento.

### Página de Cadastrar Procedimento

Nesta tela o usuário poderá cadastrar um novo procedimento, informando o tipo (Assinatura de TAC, Reunião, Oitiva), uma data e o nome do promotor responsável além de informações sobre o caso. Esta tela está ilustrada na Figura 14.

The screenshot shows the 'Cadastrar Procedimento' (Register procedure) page in the SisLegal system. The header and navigation menu are identical to Figure 13. The main content area is titled 'Cadastrar Procedimento' and contains a form with the following fields: 'Tipo do procedimento:' (dropdown menu with 'Reunião' selected), 'Data:' (calendar icon with 'Mai 9, 2011' selected), 'Promotor responsável:' (text input with 'Dr. Dmitri' entered), and 'Observações:' (text area). Below the form are buttons for 'Adicionar' and 'Voltar'. At the bottom of the page, the footer reads 'Diretoria de Tecnologia da Informação Departamento de Desenvolvimento de Sistemas'.

Figura 14 – Tela Cadastrar Procedimento.

#### **4.2.5 Testes**

Foi utilizado o *Junit* para testar o módulo 1 (Cadastro das escolas existentes no município (particulares, municipais, estaduais, federais), de todos os níveis de ensino) do sistema SisLegal, pois cada estagiário ficou encarregado de testar o módulo desenvolvido por outro colega. Tal atividade teve como principal intuito a validação do código desenvolvido, como também o auxílio a mudanças futuras, que podem ser validadas por meio de uma nova execução desses testes.

### **4.3 Atividades de Suporte Técnico**

As atividades de suporte técnico foram realizadas durante todo o período de estágio. Para evitar atrasos no desenvolvimento do sistema SisLegal, houve um revezamento por parte dos 4 (quatro) estagiários na execução de tais tarefas.

#### **4.3.1 Suporte Técnico ao Usuário**

Grande parte dos problemas que surgiram durante os meses de trabalho estavam relacionados com a pouca familiaridade por parte dos funcionários do MPPB no uso de *softwares*, que eram indispensáveis para o andamento das tarefas da instituição.

Assim, uma tarefa corriqueira dos estagiários consistia no auxílio aos funcionários da instituição no uso de algumas ferramentas. Como editores de texto, navegadores *web*, antivírus, leitores pdf, etc.

Para que as dificuldades não fossem recorrentes, os estagiários, além de resolver o problema do funcionário, também os ensinavam a utilizar de uma forma melhor o *software*. Aumentando assim o grau de conhecimento dos usuários.

### **4.3.2 Instalação, Atualização, Customização e Manutenção dos Softwares**

Sendo o MPPB uma instituição grande e com muitos funcionários, um trabalho bastante realizado foi a instalação de *software* e a manutenção dos já existentes, nas máquinas de vários funcionários.

Dentre as atividades efetuadas destaca-se a troca do *software* antivírus da instituição, que fez parte do processo de desinfecção das máquinas dos usuários. Compreendendo a desinstalação do *software* antigo (*Trend Micro*) e a instalação do programa novo (*Avira Antivir*), juntamente com a configuração do *Proxy*.

Também foi realizada a formatação do sistema operacional de algumas máquinas que tiveram problemas com vírus.

Outros *softwares* também foram instalados e atualizados, como, *BrOffice*, *Adobe Reader*, *drivers* de impressoras, etc.

### **4.3.3 Manutenção de Impressoras e Cartuchos**

Por haver uma grande demanda de impressão de documentos na instituição, diariamente foram abordados problemas relacionados a impressoras e cartuchos. Os mais comuns eram, atolamento de papel, problemas com a conectividade de impressoras compartilhadas, problemas com o *driver* da impressora, cartuchos danificados, etc.

Assim, um trabalho importante efetuado pelos estagiários foi a reciclagem de cartuchos, que compreendia em limpar, desentupir e encher cartuchos.

#### **4.3.4 Manutenção da Rede**

Também era função dos estagiários o suporte à rede. Porém, nenhum grande problema foi enfrentado durante o período do estágio. Os problemas mais corriqueiros eram relativos a configurações de *Proxy*, inserção e remoção de máquinas e impressoras na rede, cabos danificados, etc.

SEÇÃO V

CONSIDERAÇÕES

FINAIS

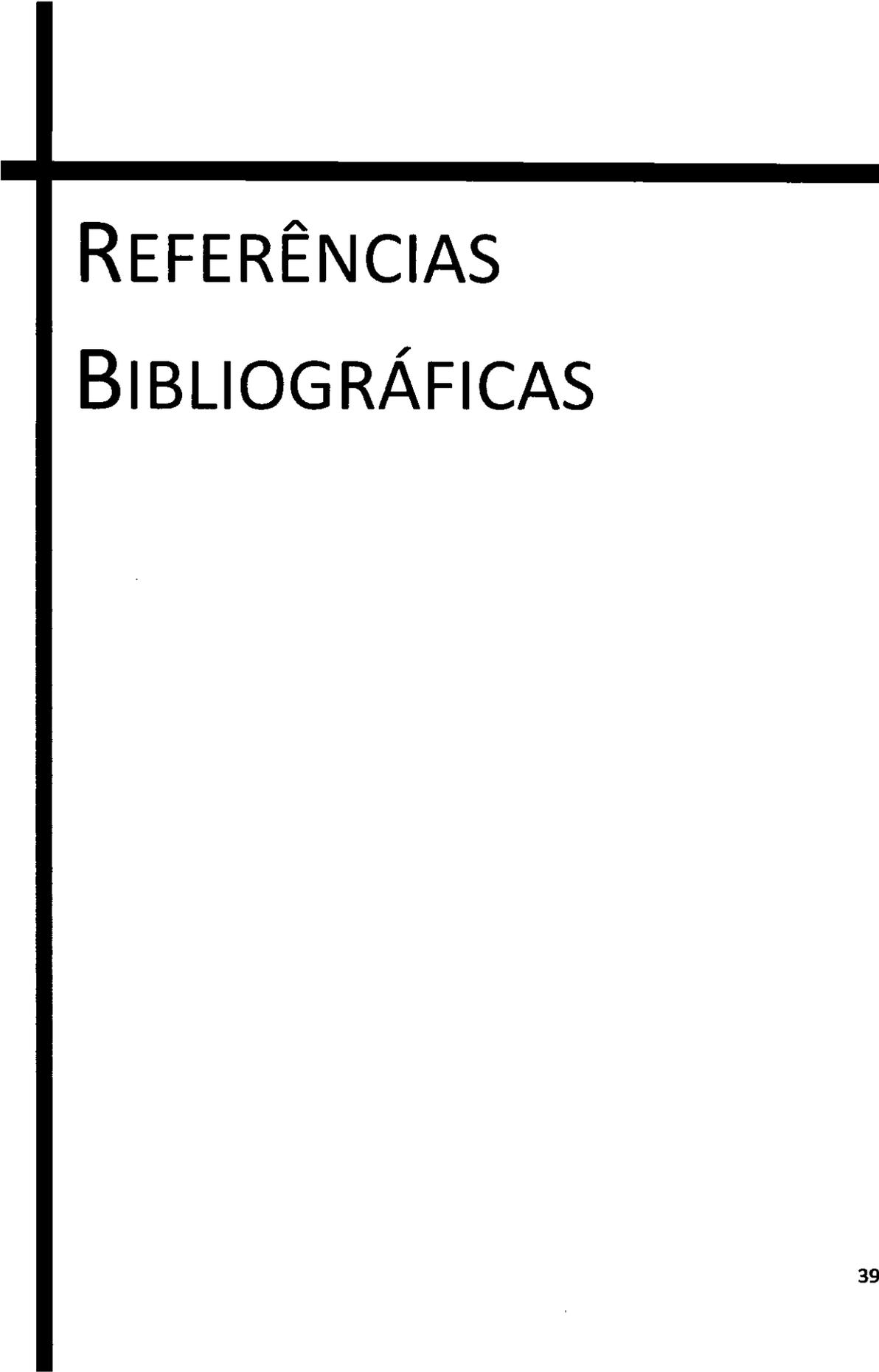
## 5 Considerações Finais

A participação de um graduando em um estágio é de suma importância na sua formação, pois possibilita colocar em prática todo o conhecimento obtido durante o curso superior. Além disso, a experiência adquirida com a convivência diária com funcionários, ambiente de trabalho, entre outros, é fundamental na hora de ingressar no mercado de trabalho.

Outro fator importante é que no estágio o universitário tem a possibilidade de aprender diariamente diante dos problemas que surgem, o que pode ser observado como um complemento de sua formação e um diferencial quando este tentar concorrer a uma vaga em um emprego.

Durante o período de realização das atividades algumas dificuldades surgiram, porém foram sanadas sem maiores problemas. Como por exemplo, a falta de familiaridade com algumas tecnologias que foram utilizadas no desenvolvimento e não são abordadas em disciplinas da graduação. Entretanto, como o curso de Ciência da Computação da UFCG possui algumas disciplinas que permitem o aprendizado de novas linguagens e ferramentas este problema não veio a prejudicar o andamento das atividades.

Em resumo, é possível observar que a base adquirida na universidade é bastante importante. Porém, é necessário colocar em prática esses conceitos adquiridos, e a participação em um estágio proporciona esta oportunidade. Assim, uma sugestão seria a criação de uma disciplina obrigatória de estágio, para que todos os alunos do curso tivessem tal oportunidade antes de ingressar em um mercado tão concorrido.



REFERÊNCIAS

BIBLIOGRÁFICAS

## Referências Bibliográficas

SAUVÉ, J. P. **Java Server Faces (JSF)**. Disponível em: <<http://www.dsc.ufcg.edu.br/~jacques/cursos/daca/html/jsf/jsf.htm>>. Acessado em: 18 de Junho de 2011.

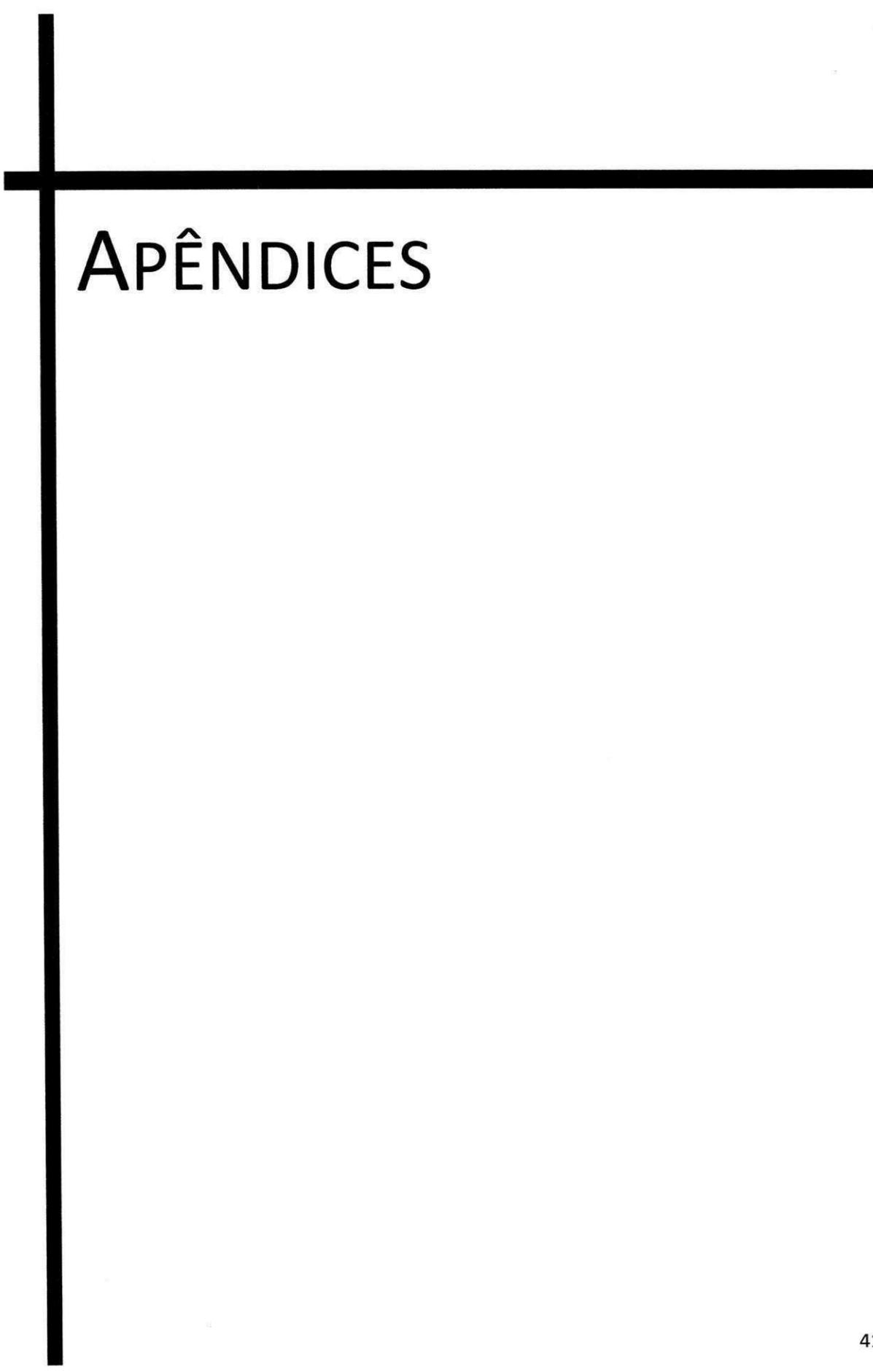
PITANGA, Talita. **JavaServer Faces: A mais nova tecnologia Java para desenvolvimento Web**. [S.L]. Disponível em: <<http://www.guj.com.br/content/articles/jsf/jsf.pdf>>. Acessado em: 17 de Junho 2011.

JAVAFREE. **Hibernate**. Disponível em: <<http://javafree.uol.com.br/wiki/hibernate>>. Acessado em: 18 de Junho de 2011.

CALÇADO, P. **Spring Framework**. Disponível em: <<http://blog.flexdev.com.br/wp-content/uploads/spring/apostila-spring.pdf>>. Acessado em: 16 de Junho de 2011.

CARVALHO, Marlon. **Spring Framework: Introdução**. Disponível em: <[http://imasters.com.br/artigo/4497/spring\\_framework\\_introducao](http://imasters.com.br/artigo/4497/spring_framework_introducao)>. Acessado em: 19 de Junho de 2011.

PAPO, J. P. **Metodologias Ágeis e Extreme Programming(XP) Uma Introdução**. Disponível em: <<http://www.spinsp.org.br/apresentacao/SpinXP.pdf>>. Acessado em: 23 de Junho de 2011.



# APÊNDICES

# APÊNDICE A – Plano de Estágio



UFPG - UNIVERSIDADE FEDERAL DE CAMPINA GRANDE  
CEEI - CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA  
DSC - DEPARTAMENTO DE SISTEMAS E COMPUTAÇÃO

## Plano de Estágio Integrado

Ministério Público

Desenvolvimento do Módulo Cadastro de Procedimentos e Suporte Técnico em  
Informática

Felipe Barbosa Araújo Ramos

Mat: 20721015

Prof. Phd Tiago Lima Massoni

Orientador Acadêmico

Campina Grande,

Março 2011

## **Informações Pessoais**

---

**Nome:** Felipe Barbosa Araújo Ramos

**Curso:** Ciência da Computação

**Matricula:** 20721015

**CPF:** 081.538.434-35

**RG:** 3272333 SSP/PB

**Endereço:** Rua Venâncio Martins Sampaio, nº 37, Centro.

Soledade – PB

CEP: 58155-000

**Email:** felipebar15@gmail.com

## **Ambiente de Estágio**

---

As atividades de estágio serão realizadas no Ministério Público do Estado da Paraíba, 2º Centro de Apoio Operacional às Curadorias. Rua Promotora Terezinha Lopes de Moura S/N, Liberdade, 58.410-064, Campina Grande, Paraíba.

O sistema será desenvolvido por quatro estagiários, onde cada um ficará encarregado por um módulo do projeto.

## **Supervisão**

---

### **Supervisor Acadêmico**

**Nome:** Phd. Tiago Lima Massoni

**Endereço:** Departamento de Sistemas e Computação

Universidade Federal de Campina Grande

Avenida Aprígio Veloso, 882 – CEP: 58.109-970

Bodocongó, Campina Grande, PB – Brasil.

**Email:** tiagomassoni@gmail.com

### **Supervisor Técnico**

**Nome:** Uirá Vasconcelos Alencar de Assis

(Chefe de Suporte Técnico do Ministério Público da Paraíba)

**Endereço:** Rua Promotora Terezinha Lopes de Moura, S/N

CEP: 58.410-064

Liberdade, Campina Grande, PB – Brasil.

**Email:** uira@mp.pb.gov.br

## Resumo do Problema do Estágio

---

A Paraíba conta hoje com um número muito alto de crianças sem o reconhecimento de paternidade em seu registro de nascimento. Esse tipo de situação pode causar diversos tipos de deficiências psicológicas interferindo na socialização e na vida dessas crianças. Além de ser importante do ponto de vista psicológico, a regularização do registro, também é um direito da criança.

Com a implantação deste projeto o Ministério Público da Paraíba visa reduzir o número de crianças sem o reconhecimento de paternidade em seu registro de nascimento, por meio do levantamento, em cada comarca, das pessoas que têm seu registro civil incompleto, notificando os respectivos responsáveis (geralmente as mães) para esclarecimento acerca das medidas que poderão ser utilizadas para a regularização da filiação.

O sistema SisLegal está dividido nos seguintes módulos:

- 1. Cadastro das escolas existentes no município (particulares, municipais, estaduais, federais), de todos os níveis de ensino;**
2. Cadastro da relação dos alunos sem maternidade ou sem paternidade reconhecida ou sem registro;
- 3. Cadastro da relação de procedimentos por escola originados a partir dos cadastros realizados o item anterior;**
4. Registro das notificações expedidas para marcação das oitivas (audição, ato de ouvir as testemunhas ou as partes de um processo judicial);
5. Módulo gerencial com a emissão de relatórios de casos resolvidos (paternidade aceita), casos não resolvidos e seus motivos, relação de escolas e relação de alunos;

## **Objetivo do estágio**

---

Desenvolvimento do módulo Cadastro de Procedimentos, que faz parte do sistema SisLegal que visa a criação de um cadastro para as informações de nomes de escolas, diretores e relação de alunos e mães de alunos com os registros dos alunos e seus pais para o projeto Nome Legal (Planejamento estratégico do Ministério Público da Paraíba 2010-2016).

## Metodologia

---

O Departamento de Desenvolvimento de Sistemas do MPPB, subordinado a Diretoria de Tecnologia da Informação, está utilizando para o desenvolvimento de seus sistemas metodologia ágil com práticas de *SCRUM* e *Kanban* para acompanhamento e transparência das tarefas (estórias ou requisitos do sistema) atribuídas ao desenvolvedor e práticas de RUP para coleta e documentação dos sistemas.

## **Atividades *Planejadas***

---

Devem ser desenvolvidas as seguintes atividades no estágio:

- Levantamento dos requisitos do sistema com a gestora do projeto;
- Análise e arquitetura do sistema;
- Implementação do módulo III (cadastro da relação de procedimentos por escola);
- Testes do módulo I (cadastro das escolas existentes no município);
- Implantação;
- Manutenção.

## ***Não Planejadas***

Suporte técnico em informática.

## Cronograma de atividades

---

O cronograma de atividades proposto é o seguinte:

Atividades planejadas	2-4/Março	1-2/Abril	3-4/Abril	1-2/Maio	3-4/Maio	1-2/Junho	3-4/Junho
Levantamento dos requisitos detalhados do sistema com a gestora do projeto		X					
Estudo das tecnologias utilizadas no departamento de Desenvolvimento de Sistemas - DITEC		X					
Análise da arquitetura do sistema			X				
Implementação do cadastro da relação de procedimentos			X	X	X		
Testes unitários, de integração e funcionais			X	X	X	X	
Treinamento							X
Implantação							X
Elaborar Relatório	X	X	X	X	X	X	X
Suporte Técnico	X	X	X	X	X	X	X
Defesa do Estágio							X

## **Resultados esperados**

---

Após a implantação, treinamento e correta alimentação do sistema por parte dos servidores e promotores envolvidos no projeto esperamos fornecer condições para que a promotora responsável possa apresentar relatórios gerenciais de acompanhamento dos vários casos de registro de paternidade cadastrados no sistema.

## Bibliografia

---

- [1] SCHWABER, K., Agile Project Management with Scrum ed. Microsoft Press, 2004. ISBN 978-0-735-61993-7.
- [2] KRUCHTEN, P., Introdução ao RUP : Rational Unified Process ed. Ciência Moderna, 2003. ISBN 8-57393-275-9

## **Aprovação**

---

---

Uirá Vasconcelos Alencar de Assis  
Supervisor Técnico

---

Tiago Lima Massoni  
Orientador Acadêmico

---

Joseana Macêdo Fchine  
Coordenadora da disciplina Estágio Integrado