

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

**RELATÓRIO DE ESTÁGIO**

**EVOLUÇÃO DOS MÓDULOS DA CAMADA DE VISÃO  
DO SISRTM – SISTEMA DE ROTEIRO DE  
MANOBRAS**

FELIPE VIEIRA  
Estagiário

TIAGO MASSONI  
Orientador Acadêmico

ELOI ROCHA  
Supervisor Técnico

**Campina Grande – PB**

**Dezembro de 2010**

**EVOLUÇÃO DOS MÓDULOS DA CAMADA DE VISÃO DO SISRTM – SISTEMA  
DE ROTEIRO DE MANOBRAS**

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

**BANCA EXAMINADORA**

*Tiago L. Massoni*

---

Prof. Dr. Tiago Massoni

ORIENTADOR ACADÊMICO

*Joseana Macedo Fachine*

---

Prof.<sup>a</sup> Dr.<sup>a</sup> Joseana Macedo

MEMBRO DA BANCA

*Livia Maria R. Sampaio Campos*

---

Prof.<sup>a</sup> Dr.<sup>a</sup> Livia Sampaio

MEMBRO DA BANCA



Biblioteca Setorial do CDSA. Maio de 2021.

Sumé - PB

## AGRADECIMENTOS

Considerando-se este relatório de estágio como fruto de um trabalho desenvolvido durante quatro meses, agradecer pode não ser uma tarefa simples, e nem justa. Para não correr o risco da injustiça, agradeço de antemão a todos que de alguma forma participaram desse processo por diferentes razões, sejam elas através do fornecimento de diretrizes necessárias ao progresso harmonioso das atividades, seja dando todo o suporte psicológico requerido a quaisquer pessoas que realizam algum trabalho de caráter levemente duradouro.

Prosseguindo, gostaria de agradecer primeiramente a Deus, sem o suporte espiritual do qual essa tarefa não seria transcorrida da maneira esperada.

Dirijo meus próximos agradecimentos à empresa Smartiks Tecnologia da Informação, que na figura de seu membro superior e meu supervisor técnico Eloi Rocha (ao qual eu foco esse agradecimento em especial) me deram a oportunidade de desempenhar esse conjunto de tarefas de fundamental importância tanto a mim na figura de estagiário como à própria empresa, que usufrui do meu esforço para promover a ampliação do patrimônio intelectual da mesma.

É indispensável também não deixar de citar meus sinceros agradecimento ao Prof. Dr. Tiago Massoni, que no papel de meu orientador acadêmico forneceu uma série de orientações que fizeram com que a maioria das minhas dúvidas envolvendo eventuais detalhes do processo ao qual um estágio integrado é submetido fossem sanadas sem muitas dificuldades.

E por último, mas não menos importante, me sinto no dever de agradecer à professora Joseana Fachine, que na função de profissional docente da disciplina desempenhou um papel deveras significativo no intuito de fornecer-nos de alguma forma um ganho pessoal e profissional.

## APRESENTAÇÃO

Como parte das exigências estabelecidas pelo curso de Ciência da Computação da Universidade Federal de Campina Grande para cumprimento da disciplina de estágio integrado, apresenta-se neste documento o relatório de estágio, elaborado ao longo do período das atividades, iniciadas em 13 de Agosto e a serem concluídas no próximo dia 13 de Dezembro.

As atividades que compreendem o estágio foram realizadas na empresa Smartiks Tecnologia da Informação LTDA, cujo escritório situa-se na sala 06 da Fundação Parque Tecnológico da Paraíba, localizada em Campina Grande, na Rua Emiliano Rosendo Silva, nº 115, bairro Bodocongó. Vale-se salientar que, apesar desse ambiente representar a Smartiks em Campina Grande, era dado total apoio para que as tarefas fossem desenvolvidas também em outros locais que não o supracitado.

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

A Smartiks Tecnologia da Informação é uma empresa atuante no segmento de desenvolvimento de software, implementando soluções inteligentes para empresas de distribuição, transmissão e geração de energia elétrica. Entre os principais projetos realizados pela empresa está o sistema SisRTM, – sistema web desenvolvido em Java para gerência de Roteiros de Manobras – pertencente à CHESF (Companhia Hidro Elétrica do São Francisco). O SisRTM já se encontra em utilização pela CHESF e constatou-se que o sistema necessitava de novas funcionalidades, correções dos *bugs*<sup>1</sup> existentes, assim como uma melhoria no desempenho. Com relação à melhoria de performance, verificou-se que grande parcela do tempo observado durante a manipulação do sistema era caracterizado pelo recarregamento de páginas (conhecido como *refresh*<sup>2</sup> de tela). Dessa forma, grande parte das decisões de projeto que guiaram o desenvolvimento do estágio vieram como um mecanismo para melhorar a experiência do usuário eliminando ao máximo a necessidade do uso do *refresh*. Utilizando-se de alguns argumentos da mesma natureza dos supracitados foi proposta uma remodelagem da interface do sistema, de modo que a manipulação por parte dos agentes designados a tal tarefa fosse otimizada e, dessa forma, boa parte dos *bugs* verificados anteriormente no SisRTM fossem eliminados. A partir do momento que a interface do sistema foi modificada, verificou-se que a mesma dava respaldo à realização de novas tarefas que viriam a acrescentar ainda mais às funcionalidades preexistentes no sistema. Dessa forma, foram desenvolvidas atividades de implementação de novas funcionalidades ao SisRTM, não se contendo, assim, apenas à área de programação de interface. Esse relatório em especial cita de uma maneira detalhada todas as tarefas desenvolvidas como parte desse esforço de evolução da interface web do SisRTM.

---

<sup>1</sup> [http://searchsoftwarequality.techtarget.com/sDefinition/0,,sid92\\_gci211714,00.html](http://searchsoftwarequality.techtarget.com/sDefinition/0,,sid92_gci211714,00.html)

<sup>2</sup> <http://www.webopedia.com/TERM/R/refresh.html>

# SUMÁRIO

AGRADECIMENTOS.....	3
LISTA DE SIGLAS E ABREVIATURAS .....	8
LISTA DE FIGURAS .....	9
1. INTRODUÇÃO .....	11
1.1 O Problema .....	12
1.2 Objetivos do Estágio .....	13
2. AMBIENTE DE ESTÁGIO .....	16
2.1. A Empresa.....	16
2.2. Carga horária .....	16
2.3. Infraestrutura .....	18
2.4. Orientação Acadêmica .....	19
2.5. Supervisão Técnica.....	19
3. FUNDAMENTAÇÃO TEÓRICA.....	21
3.1 Processo de Software .....	21
3.1.1. <i>Planning Game</i> .....	22
3.1.2. Pequenas Entregas .....	23
3.1.3. Padrão de Código .....	23
3.1.4. Integração Contínua.....	24
3.2. Tecnologias Utilizadas .....	24
3.2.1. Java .....	24
3.2.2. Google Web Toolkit.....	25
3.2.3. Javascript .....	27
3.2.4. Hibernate .....	28
4. ATIVIDADES REALIZADAS.....	30
4.1. Entendimento da versão atual do SisRTM.....	32
4.2. Definição das atividades .....	32
4.3. Estudo da tecnologia GWT .....	33
4.4. Desenvolvimento.....	34

4.4.1 Finalização do desenvolvimento do módulo de Busca de Roteiros na nova camada de visão	34
4.4.2. Desenvolvimento da funcionalidade de listagem de roteiros em edição, um complemento do módulo de Busca na nova camada de visão.....	35
4.4.3. Desenvolvimento do módulo de Templates de Ação na nova camada de visão .....	37
4.4.4 Desenvolvimento do módulo de Usuários na nova camada de visão .....	38
4.4.5. Desenvolvimento do módulo de Origens na nova camada de visão .....	39
4.4.6. Refinamento dos módulos previamente implementados .....	41
5. CONSIDERAÇÕES FINAIS .....	44
6. REFERÊNCIAS BIBLIOGRÁFICAS .....	47
7. APÊNDICES.....	49
APÊNDICE A – Plano de Estágio .....	49



## LISTA DE SIGLAS E ABREVIATURAS

<b>CHESF</b>	Companhia Hidro Elétrica do São Francisco <sup>3</sup>
<b>DSC</b>	Departamento de Sistemas e Computação
<b>JVM</b>	<i>Java Virtual Machine</i>
<b>PaqtcPB</b>	Fundação Parque Tecnológico da Paraíba
<b>RTM</b>	Roteiros de Manobras
<b>SisRTM</b>	Sistema de Roteiro de Manobras
<b>UFCG</b>	Universidade Federal de Campina Grande
<b>XP</b>	<i>Extreme Programming</i> <sup>4</sup>
<b>GWT</b>	<i>Google Web Toolkit</i> <sup>5</sup>
<b>GXT/EXT-GWT</b>	<i>Extended Google Web Toolkit</i>
<b>TI</b>	Tecnologia da Informação
<b>GB</b>	Gigabytes
<b>GHz</b>	Gigahertz
<b>AJAX</b>	<i>Asynchronous Javascript And XML</i>
<b>API</b>	<i>Application Programming Interface</i>
<b>DOM</b>	<i>Document Object Model</i>
<b>SQL</b>	<i>Structured Query Language</i>
<b>HQL</b>	<i>Hibernate Query Language</i>
<b>SGBD</b>	Sistema de Gerenciamento de Banco de Dados
<b>POO</b>	Programação Orientada a Objetos
<b>XML</b>	<i>Extensible Markup Language</i>

---

<sup>3</sup> <http://www.chesf.gov.br/>

<sup>4</sup> <http://www.extremeprogramming.org/>, <http://improveit.com.br/xp/>

<sup>5</sup> <http://www.sencha.com/products/gwt/>

## LISTA DE FIGURAS

Figura 1.1 - Estrutura de um roteiro de manobras .....	13
Figura 4.1 - Tela de Busca de Roteiro de Manobras .....	35
Figura 4.2 - Tela de Listagem de Roteiros em Edição.....	36
Figura 4.3 - Tela de Gerenciamento de Templates de Ação.....	38
Figura 4.4 - Tela de Gerenciamento de Usuários.....	39
Figura 4.5 - Hierarquia de Origens do SisRTM.....	40
Figura 4.6 - Tela de Gerenciamento de Origens.....	41
Figura 4.7 - Tela de Login Remodelada.....	42

SEÇÃO I

INTRODUÇÃO

## 1. INTRODUÇÃO

O estágio a ser descrito neste documento compila todas as atividades desenvolvidas no período compreendido entre 13 de Agosto de 2010 e 13 de Dezembro do mesmo ano, intervalo no qual foram fornecidos serviços à empresa Smartiks Tecnologia da Informação<sup>6</sup>.

É de comum conhecimento que um dos grandes fatores responsáveis pela má qualidade da mão de obra especializada nos dias de hoje é justamente a ausência de oportunidades de experiências profissionais, e isso é uma característica não somente da área de Tecnologia da Informação, como também de vários outros setores em geral. Profissionais por vezes possuem um embasamento teórico bastante considerável, mas é notável a ausência de situações nas quais tais conceitos são postos em prática.

Nesse panorama global se encaixam perfeitamente as oportunidades de estágio: chances preciosas de serem desenvolvidas soluções no mercado real, fornecendo tanto um ganho particular para própria empresa, que usufrui de um postulante à empregado sedento de vontade de mostrar a qualidade de seus serviços, quanto (e mais importante ainda) um ganho pessoal bastante considerável por parte do estagiário, o qual por vezes não possui experiência de como são aplicados os mecanismos de negócio das micro, pequenas e médias empresas e que encontram nesta possibilidade uma perspectiva bastante promissora de entrar de vez no tão temido mercado de trabalho. É evidente que nem sempre essa primeira experiência pode ser a idealizada pelo jovem empregado, mas até mesmo nesse tipo de situação o estágio vem com uma carga deveras positiva de aprendizado: derrubar certos mitos existentes à respeito da entrada em conglomerados corporativos.

Foi dessa forma que a oportunidade de estágio foi tratada: como uma fonte de experiência na qual várias ocorrências do dia-a-dia eram observadas,

---

<sup>6</sup> <http://http://www.smartiks.com/website/>

desde as atividades de desenvolvimento previamente conhecidas, até o cotidiano do funcionamento de uma microempresa de TI, caracterizada por um mecanismo de delegação de tarefas e tomadas de decisões bastante intenso.

Este tratamento, porém, vem como um objetivo extra, fruto da diretriz principal considerada pelo estagiário em questão, que foi a de fornecer os serviços que ao fim do período de estágio gerariam os resultados esperados por parte dos envolvidos naquele processo no que diz respeito à evolução da interface web do sistema SisRTM da Smartiks Tecnologia da Informação.

### **1.1 O Problema**

Os centros de supervisão e controle de redes de transmissão e distribuição de energia elétrica vêm, constantemente, modernizando-se nos últimos tempos. Entretanto, a gerência de tais centros permanece sendo considerada uma tarefa árdua; entre as principais razões destacam-se a vasta dimensão geográfica e as particularidades inerentes a cada equipamento da rede. Um dos problemas que merece uma atenção especial consiste na dificuldade em gerenciar os Roteiros de Manobras. Entende-se por Roteiros de Manobras documentos que expressam quais ações devem ser realizadas durante a execução de manobras no sistema elétrico, por exemplo, quais as ações são necessárias para a liberação ou normalização de um equipamento. A estrutura de um roteiro de manobras pode ser observada na Figura 1.1.

É neste intuito que surge o sistema SisRTM. Desenvolvido pela Smartiks Tecnologia da Informação, essa aplicação web vem com a função de permitir um maior controle dos Roteiros de Manobra geridos pela CHESF. Apesar de ser considerado um sistema extremamente operacional no sentido do vasto uso por parte dos operadores da companhia, foi declarada como prioridade a necessidade de uma série de melhorias, de forma a tornarem a plataforma ainda mais ampla no sentido de permitir um controle cada vez maior das instalações elétricas de todo o país. O ponto mais discutido durante as reuniões da empresa trazia à tona a rejeição com relação à necessidade do recarregamento das janelas efetuado sempre que era realizada uma requisição ao servidor em busca de dados a serem exibidos na interface. Esse *refresh* na

maioria das vezes alterava a posição atual da tela, além de diminuir o desempenho da aplicação, o que era considerado como um comportamento deveras inaceitável levando-se em consideração as situações nas quais a aplicação era utilizada.

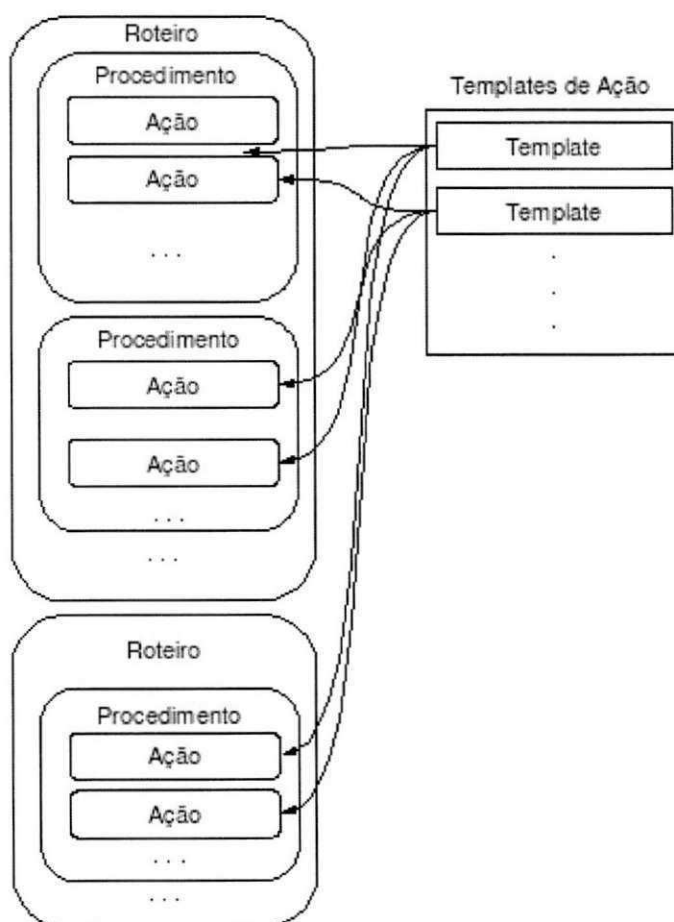


Figura 1.1 - Estrutura de um roteiro de manobras

## 1.2 Objetivos do Estágio

O objetivo geral deste estágio foi evoluir o SisRTM, corrigindo os problemas levantados pelos operadores da aplicação. A seguir, seguem os objetivos específicos para alcançar o objetivo geral descrito:

- Entender a estrutura do sistema em relação a sua arquitetura, ferramentas e tecnologias utilizadas;
- Estudar as ferramentas e tecnologias utilizadas;
- Remodelar um parte das telas do antigo sistema empregado na CHESF utilizando a tecnologia GWT, que excluirá de vez a necessidade da ocorrência de *refreshes* da tela
- Implantar novas versões do sistema sempre que possível para validar o que foi implementado.

SEÇÃO II

# AMBIENTE DE ESTÁGIO



## **2. AMBIENTE DE ESTÁGIO**

### **2.1. A Empresa**

A Smartiks Tecnologia da Informação pesquisa e desenvolve soluções inteligentes para empresas de distribuição, transmissão e geração de energia elétrica.

A missão da Smartiks consiste em criar soluções que reduzam o tempo de recomposição do sistema elétrico após falhas ou ocorrências. A equipe da Smartiks é composta por vários doutores e mestres que vêm se dedicando em encontrar soluções para o sistema elétrico há vários anos.

A Smartiks tem trabalhado em parceria com a Universidade Federal de Campina Grande (UFCG) e a Fundação Parque Tecnológico da Paraíba (PaqTcPB). Seu escritório é localizado na sala 06 do PaqTcPB, localizada em Campina Grande, na Rua Emiliano Rosendo Silva, nº 115, bairro Bodocongó.

### **2.2. Carga horária**

A seguir está feita a listagem das tarefas desenvolvidas pelo estagiário durante a execução do processo de estágio, fazendo-se um paralelo com o tempo decorrido na realização das mesmas, já a descrição das atividades será feita mais a frente na Seção 4. É importante frisar que foi planejado para as atividades de estágio um total de 360 horas a serem cumpridas no intervalo de quatro meses entre a data de 13 de Agosto de 2010 e 13 de Dezembro do mesmo ano.

1. Entendimento da versão em operação do SisRTM. **Tempo: 30h**
2. Estudo do framework gráfico a ser aplicado: GWT. **Tempo: 30h**
3. Finalização do módulo de Busca na nova camada de visão.  
**Tempo: 40h**
4. Desenvolvimento do módulo de Usuários na nova camada de visão. **Tempo: 30h**

5. Desenvolvimento do módulo de Templates na nova camada de visão. **Tempo: 40h**
6. Desenvolvimento do módulo de Origens na nova camada de visão. **Tempo: 40h**
7. Elaboração das demais telas do SisRTM. **Tempo: 60h**
8. Ajustes nos módulos previamente desenvolvidos. **Tempo: 60h**
9. Elaboração do plano de estágio. **Tempo: 10h**
10. Tarefas de contexto fora do SisRTM. **Tempo: 20h**

<i>Semana / Atividade</i>	<i>1</i>	<i>2</i>	<i>3</i>	<i>4</i>	<i>5</i>	<i>6</i>	<i>7</i>	<i>8</i>	<i>9</i>	<i>10</i>
13/08 – 20/08	X									
20/08 – 27/08	X	X								X
27/08 – 03/09		X						X		X
03/09 – 10/09		X	X					X		
10/09 – 17/09			X				X	X		
17/09 – 24/09			X				X	X		
24/09 – 01/10				X			X	X		
01/10 – 08/10				X	X		X	X		
08/10 – 15/10					X		X	X		
15/10 – 22/10					X		X	X		X
22/10 – 29/10						X	X	X		X
29/10 – 05/11						X	X	X		
05/11 – 12/11						X	X	X	X	X
12/11 – 19/11							X	X	X	X
19/11 – 26/11								X	X	

26/11 – 03/12						X	X	X	X	
03/12 – 10/12						X	X		X	

Quadro 2.1 – Cronograma final de atividades

### 2.3. Infraestrutura

O escritório da Smartiks é dotado de 2 máquinas, cada uma contendo um processador Intel Core 2 Duo 3.06 GHz, 320 GB de espaço em disco e 4 GB de memória principal, ambas com o sistema operacional Linux (distribuição Ubuntu 9.04) devidamente instalado em conjunto com o ambiente integrado de desenvolvimento *Eclipse*, dentre outros *softwares* necessários para evoluir o SisRTM.

Além disso, pode-se fazer uso das instalações fornecidas pelo PaqTcPB, que dão suporte ao uso de uma série de *commodities* tais como: ar condicionado, filtro de água, cafeteiras e alguns outros elementos que auxiliam no processo de desenvolvimento.

## **2.4. Orientação Acadêmica**

**Nome:** Tiago 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:** massoni@dsc.ufcg.edu.br

## **2.5. Supervisão Técnica**

**Nome:** Eloi Rocha Neto

**Endereço:** Fundação Parque Tecnológico da Paraíba  
Rua Emiliano Rosendo Silva, 115, sala 06  
Bodocongó, Campina Grande, PB – Brasil.

**Email:** eloi@smartiks.com

SEÇÃO III

FUNDAMENTAÇÃO

TEÓRICA

## 3. FUNDAMENTAÇÃO TEÓRICA

No transcorrer dessa seção serão descritos os conceitos que serviram de alicerce para o desenvolvimento das atividades de estágio. Na Seção 3.1 é apresentado o processo de software empregado e suas principais características, e na Seção 3.2 subsequente são listadas e explicadas as tecnologias que serviram de suporte à execução das tarefas desenvolvidas

### 3.1 Processo de Software

Um processo de desenvolvimento de *software* é um aglomerado de atividades que conduzem à elaboração de um produto de *software*. Estas atividades podem envolver o desenvolvimento do *software* desde o início utilizando uma linguagem padrão como *Java* ou *C* (SOMMERVILLE, 2007).

Apesar de já existir previamente, o SisRTM foi submetido a um processo de software baseado nas práticas mais difundidas do *Extreme Programming* (XP). XP é um processo de desenvolvimento que possibilita a criação de software de alta qualidade de maneira ágil, econômica e flexível. Vem sendo adotado com algum sucesso na Europa, nos Estados Unidos e mais recentemente no Brasil.

Cada vez mais as empresas convivem com ambientes de negócios que requerem mudanças frequentes em seus processos, as quais afetam os projetos de software. Os processos de desenvolvimento mais comuns são caracterizados por uma quantidade exagerada de atividades e artefatos que buscam proteger o software contra alterações, o que faz pouco ou quase nenhum sentido, uma vez que os projetos devem se adaptar a tais mudanças ao invés de evitá-las (TELES, 2009).

O XP foca os esforços da equipe de desenvolvimento em tarefas que geram resultados rapidamente na forma de software intensamente testado e

alinhado às necessidades de seus usuários. Além disso, simplifica e organiza o trabalho combinando técnicas comprovadamente eficazes e eliminando atividades redundantes. Por fim, reduz o risco dos projetos desenvolvendo software de forma iterativa e reavaliando permanentemente as prioridades dos usuários.

A seguir serão listados os conceitos característicos do XP mais usados durante o período de desenvolvimento das atividades de estágio.

### **3.1.1. *Planning Game***

Uma das filosofias mais importantes do XP gira em torno de duas questões chave: tentar prever ao máximo o que será finalizado levando-se em consideração uma determinada data e definir, sempre que possível, o que será realizado em seguida. De modo a se ter sempre o controle dessa problemática, existem outros dois pontos essenciais relacionados ao *planning game*.

#### **3.1.1.1. Plano de Release**

Por meio de um contrato firmado com os clientes, são listadas todas as funcionalidades desejadas pelos mesmos, e cabe então aos membros da equipe estimarem um custo a determinadas tarefas e elaborarem um plano contendo a ordem de implementação de tais incubências, além de um tempo estimado que será levado em consideração para a finalização das mesmas.

#### **3.1.1.2. Plano de Iteração**

É por meio desse documento que os desenvolvedores obtêm as diretrizes de acordo com as quais as entregas serão realizadas. O bloco de entregas é quebrado em um subconjunto de pequenas baterias de implementação (denominadas iterações) que em geral concentram suas durações entre 1 e 2 semanas.

### **3.1.2. Pequenas Entregas**

Um dos fatores responsáveis por mudanças de escopo de projetos acontece devido ao fato de a implementação do produto ser feita excluindo-se o cliente do processo. É intuitivo perceber que a promoção de mudanças em um produto final carrega consigo um grau de dificuldade bastante impactante no panorama geral, e como o XP é considerado um método de desenvolvimento ágil, ou seja, perfeitamente adaptável à ocorrência de mudanças, ele prega que um produto parcial seja sempre posto em contato com os envolvidos no processo, de modo que eventuais mudanças serão realizadas dentro do próprio escopo de implementação, o que pode se tornar bastante útil em projeto de contratos flexíveis.

Deixar o projeto sempre visível e acessível do ponto de vista do cliente mantém o projeto tangível, o que aumenta consideravelmente a probabilidade de sucesso do método.

### **3.1.3. Padrão de Código**

Uma vez que o XP é um projeto adequado ao desenvolvimento de projetos em equipe, é importante manter uma certa unificação do código de modo que qualquer integrante tenha total liberdade e capacidade de realizar alterações em quaisquer áreas da implementação. No processo de software em questão essa característica em destaque é suportada por intermédio da criação de um padrão de codificação que é posto em prática de uma maneira que seja quase impraticável perceber quem é o dono de uma determinada implementação dado o grau de homogeneidade desse artefato.

O padrão implementado em si pode ser abstraído, o importante é que todos os pontos da implementação sejam familiares a cada membro do time.



### **3.1.4. Integração Contínua**

O desenvolvimento de novas funcionalidades é algo sempre iminente na maioria dos projetos de software, e o gerenciamento de versões pode se tornar uma tarefa bastante intangível caso as novas funcionalidades não sejam integradas à versão atual da aplicação.

Quanto maior o intervalo empregado antes da integração da implementação atual com a anterior, maiores são os riscos das ocorrências de conflitos de código. Baseando-se nessa problemática o XP sugere que a integração de código seja algo satisfatoriamente frequente na execução do processo e, dessa forma, os riscos gerados por versões incompatíveis sejam minimizados e a versão rodando nos domínios de cada integrante seja sempre a mais atualizada.

## **3.2. Tecnologias Utilizadas**

No escopo do projeto foram utilizadas várias ferramentas que deram suporte ao desenvolvimento das atividades de implementação. Tais tecnologias foram escolhidas previamente pensando-se as necessidades exigidas pelo projeto. As tecnologias Java, GWT, Javascript e Hibernate foram todas utilizadas no desenvolvimento das atividades de evolução da camada de visão do SisRTM, sendo listadas e brevemente descritas a seguir

### **3.2.1. Java**

*Java* é uma linguagem de programação orientada a objeto que foi criada a partir de um projeto de pesquisa de codinome *Green* pela *Sun Microsystems*. Esta linguagem era inicialmente chamada de *Oak*, nome dado por James Gosling, o fundador da linguagem, após olhar para uma árvore de carvalho fora da janela em seu local de trabalho na *Sun*. Tempos depois, foi descoberto que

já existia uma linguagem de programação com o nome escolhido, então, quando um grupo da *Sun* visitava uma cafeteria o nome *Java* foi sugerido. (DEITEL, 2007).

Diferentemente das demais linguagens, que são compiladas para um código nativo, a linguagem *Java* é compilada para um *bytecode*, que é executado pela sua própria máquina virtual: a *Java Virtual Machine (JVM)*. A linguagem de programação *Java* é a linguagem convencional da Plataforma *Java*, mas não sua única linguagem.

Uma aplicação *Java* é um programa de computador que executa quando se utiliza um comando chamado *java*, o qual inicia a *JVM*. A máquina virtual não conhece absolutamente nada sobre a linguagem de programação *Java*, e sim um formato binário, chamado de *class*. Este formato contém um conjunto de instruções da máquina virtual *Java*, também chamados de *bytecodes*. Além dos *bytecodes*, o formato *class* contém uma tabela de símbolos e quaisquer outras informações auxiliares (LINDHOLM, 1999).

### 3.2.2. Google Web Toolkit

O *framework* *GWT*, que vem do acrônimo *Google Web Toolkit*, é uma ferramenta para construção de interfaces *web* mais amigáveis e rápidas. O *GWT* foi feito para facilitar e agilizar o processo de desenvolvimento das interfaces de aplicações *web* utilizando *AJAX (Asynchronous Javascript And XML)*.

O desenvolvimento de aplicações que utilizam o conjunto de tecnologias que hoje são chamadas de *AJAX* já é um fato consumado no mercado de aplicações para *web*, mas todo esse movimento trouxe novos problemas, como o aumento considerável da quantidade de código *JavaScript* nesse tipo de aplicação. Essa volta do *JavaScript* à tona no mercado trouxe problemas da não tão distante guerra dos navegadores *web*, onde cada fornecedor (na época a *Microsoft* e a *Netscape*) implementavam a linguagem *JavaScript* de uma forma diferente (GEARY, 2009).

Hoje, com o grande apelo dessa nova tecnologia, surgiram diversos *frameworks* que prometem simplificar o desenvolvimento de aplicações utilizando AJAX e um destes *frameworks* é o GWT, desenvolvido dentro do Google e liberado pelos seus desenvolvedores como software livre (DINIZ, 2009)

Uma aplicação desenvolvida utilizando-se GWT é basicamente composta de duas partes:

- O Cliente
- O Servidor

#### **3.2.2.1. O Cliente**

É justamente nessa parte em que "a mágica acontece". É o local onde o compilador traduz o código Java para Javascript. Nesse local ocorrem as maiores limitações. Dispõe-se apenas de algumas classes da API Java como, por exemplo, os tipos básicos da linguagem, seus respectivos *wrappers*. A maior parte do código escrito no lado do cliente é derivada dos componentes, *Widgets*, fornecidos pela API. As classes disponibilizadas no pacote *com.google.gwt.user.client.ui* deste pacote contêm os componentes gráficos.

#### **3.2.2.2. O Servidor**

É nessa parte do sistema onde a liberdade é total, inclusive para usar as novas funcionalidades da API de Java. Toda a estrutura de acesso à banco de dados e qualquer outra tecnologia devem permanecer no servidor.

O Google *Web Toolkit* é uma plataforma muito interessante e sem dúvida será cada vez mais requisitada na medida que esta utiliza uma linguagem muito popular: Java. Embora alguns dos seus componentes ainda tenham alguns *bugs*, está constantemente a ser corrigida e aperfeiçoada, tornando-se cada vez melhor a cada versão disponibilizada.

Existe, além do framework original de Google, um extensão do mesmo denominada GXT ou EXT-GWT (*Extended GWT*) que nada mais é do que uma versão mais desenvolvida das partes gráficas da plataforma original. Em geral, os componentes gráficos implantados no SisRTM são pertencentes à extensão da ferramenta, visto que esta foi submetida a melhorias que corrigiram uma série de comportamentos inesperados dos elementos visuais.

### 3.2.3. Javascript

JavaScript é uma linguagem de script que teve sua fase áurea nos primórdios da Internet. Foi largamente usada para obter os mais variados e inusitados efeitos em páginas web. Àquela época quanto mais surpreendente fosse para o usuário o efeito ao entrar em uma página, mais o desenvolvedor do site era conceituado na comunidade. Com a guerra dos navegadores o conhecimento das técnicas de detecção do navegador via JavaScript era requisito básico de desenvolvimento.

Com a chegada dos Padrões Web e a consequente revisão dos conceitos de desenvolvimento de sites, novos rumos foram traçados. Rumos fundamentados em critérios de separação das camadas de desenvolvimento, de acessibilidade, usabilidade e portabilidade. Esses novos rumos negavam a maioria dos efeitos e principalmente critérios de desenvolvimento com JavaScript. A linguagem caiu em completo desuso e foi renegada e delegada a um plano inferior do desenvolvimento de aplicações web.

Os Padrões Web responsáveis pelo declínio da JavaScript também foram a sua redenção ao estabelecer as especificações para a linguagem e introduzir a padronização e o DOM – *Document Object Model* que possibilitam desenvolvimento de scripts não obstrutivos.

O Javascript é utilizado pela plataforma da GWT da Google, que possui um compilador capaz de converter código Java em *scripts*. Dessa forma, o desenvolvedor captura a capacidade de modelar páginas web como se

estivesse desenvolvendo uma aplicação para desktop, sem se apegar muito à sintaxe de Javascript.

### 3.2.4. Hibernate

O Hibernate é uma ferramenta utilizada para fazer a conversão de um modelo de banco de dados objeto/relacional para o esquema de Java, dando o a habilidade ao programador de manipular as operações envolvendo o banco de dados através da tão comum utilização de um esquema de objetos suportado pelo orientação a objetos fornecida por Java

De uma maneira mais clara, o Hibernate transforma os dados da estrutura lógica de um banco de dados em objetos definidos pelo desenvolvedor. Usando o Hibernate não há a necessidade de escrever muito código de acesso a banco de dados e de SQL, pois ele utiliza a sua própria HQL (*Hibernate Query Language*), acelerando a velocidade do seu desenvolvimento de uma forma fantástica. Vale lembrar que, apesar do fato do hibernate utilizar uma linguagem própria para realizar a persistencia dos dados, podemos mudar a qualquer momento o Sistema de Gerenciamento de Banco de Dados (SGBD) utilizado.

O Hibernate foi utilizado para mapear o bando de dados do SisRTM para classe da lógica, de modo que as interações com o banco de dados era feitas por intermédio da manipulação de objetos da lógica do sistema.

## SEÇÃO IV

---

ATIVIDADES

REALIZADAS

## 4. ATIVIDADES REALIZADAS

Neste capítulo são descritas todas as atividades compreendidas no período no qual foi realizado todo o esforço no intuito de fornecer uma série de melhorias ao Sistema de Roteiro de Manobras da Smartiks Tecnologia da Informação. O processo de desenvolvimento foi realizado em detrimento ao processo de *software* descrito na Seção 3 deste documento e considerando-se períodos de iteração de 1 semana.

É importante ressaltar que o estágio foi desenvolvido conjuntamente com outro estagiário, e aliado isso a uma série de fatores que exigiram uma remodelagem na distribuição de tarefas tem-se que foram necessárias algumas mudanças no conjunto de atividades a serem desempenhadas pelo estagiário e, assim, são visíveis algumas alterações no aglomerado de incubências desenvolvidas no período citado anteriormente. Tais diferenças podem ser visualizadas comparando-se as duas tabelas a serem apresentadas a seguir nos Quadros 4.1 e 4.2. Apesar da reformulação das atividades, o tempo sugerido para a realização das mesmas permaneceu inalterado, isso porque o tempo planejado anteriormente foi apenas redistribuído na nova estrutura de divisão sugerida.

O fato de o projeto SisRTM possuir 2 estagiários permitiu que a modelagem das novas telas do sistema fosse dividida entre os mesmos. Dessa forma, os esforços destes 2 estudantes permitiram que todas as interfaces do sistema de roteiro de manobras fosse traduzidas para o framework GWT. A divisão destas telas foi realizada pelo supervisor técnico conforme as necessidades observadas e as telas implementadas por este estagiário podem ser visualizadas nas próximas atividades. Essa característica específica desse projeto fez com que os planos de estágios de ambos os estagiários fossem muito parecidos, mas a diferença mais evidente é que cada um possuiu suas devidas incubências relacionadas a telas completamente independentes no sistema.

<b>Atividades</b>	<b>Ago</b>	<b>Set</b>	<b>Out</b>	<b>Nov</b>	<b>Dez</b>
Entendimento da estrutura do sistema	X				
Estudo da tecnologia (GWT)	X				
Refazer página de busca de roteiros	X				
Refazer página de visualização de roteiros	X				
Refazer página de edição de roteiros	X	X	X	X	
Desenvolvimento do módulo de sincronização de roteiros			X	X	X
Refazer demais páginas do sistema				X	X
Elaboração e defesa do relatório de estágio		X	X	X	X

Quadro 4.1: Cronograma Original

<b>Atividades</b>	<b>Ago</b>	<b>Set</b>	<b>Out</b>	<b>Nov</b>	<b>Dez</b>
Entendimento da estrutura do sistema	X				
Estudo da tecnologia (GWT)	X				
Refazer página de busca de roteiros	X	X			
Refazer página de busca por templates		X			
Refazer página de gerenciamento de usuários			X	X	
Refazer página de gerenciamento de origens			X		
Refazer demais páginas do sistema				X	X
Elaboração e defesa do relatório de estágio		X	X	X	X

Quadro 4.2: Cronograma Real

A seguir, será apresentada em forma de seções cada etapa realizada no ciclo do desenvolvimento do sistema.



#### **4.1. Entendimento da versão atual do SisRTM**

Essa primeira etapa componente do estágio integrado consistiu em uma das tarefas mais comuns levando-se em consideração a apresentação inicial de uma dada plataforma, que foi justamente a explanação dos principais conceitos que rodeavam o contexto de aplicação do SisRTM. Durando as primeiras semanas do período do estágio, a atividade em questão se caracterizou por algumas reuniões que transmitiram o contexto no qual o sistema era usado, tal qual sua importância em determinado panorama.

Para que tal processo fosse obedecido de maneira propícia, foi realmente necessário deixar claro alguns conceitos básicos do sistema, como a arquitetura da aplicação e os padrões de codificação ao qual o processo de desenvolvimento era submetido. Somando isso a alguns detalhes de implementação, essa incubência foi finalizada com êxito.

#### **4.2. Definição das atividades**

Analisado-se o escopo do projeto de estágio foi realizado um levantamento das possíveis tarefas a serem desempenhadas pelo estagiário em questão. Como o projeto em si consistia na reformulação da interface *web* de um sistema preexistente, pode-se dizer de maneira um tanto quanto breve que tais atividades se resumiam no mapeamento de um framework gráfico para outro. Isso que foi dito foi uma maneira extremamente breve de descrever o caráter do estágio antes de algumas mudanças ao qual o processo foi submetido, diz-se isso por que à medida que o sistema ia sendo evoluído, percebia-se a possibilidade de se inserir novas funcionalidades ao mesmo, e isso era um fator que caracterizava a formulação de novas tarefas que à priori não estavam no escopo original do estágio.

O controle das atividades era feito por intermédio de um gerenciador de projetos chamado Redmine<sup>7</sup>, o qual possui código aberto e contém calendários

---

<sup>7</sup> <http://redmine.smartiks.com>

e gráficos de Gantt que ajudam na representação visual dos projetos e seus *deadlines*. Com relação ao Redmine, a funcionalidade mais utilizada era a que consistia no gerenciamento de tarefas, aplicação essa que permitia que qualquer envolvido no projeto cadastrasse novas atividades a serem desempenhadas, fazendo comentários sobre a mesma e as atribuindo a um membro qualquer da empresa. Dessa forma, a qualquer momento podia-se visualizar o status atual do desenvolvimento dos deveres de cada projetista, o que é de extrema importância na implementação de qualquer atividade.

À medida que as tarefas eram realizadas elas tinham seus status alterados para *Closed* de tal forma que o supervisor técnico poderia analisar tais implementações e enumerar novas atividades a serem realizadas.

### **4.3. Estudo da tecnologia GWT**

Seguindo a cronologia de desenvolvimento das atividades chegamos ao ponto consistente no aprendizado da tecnologia escolhida para dar suporte às funcionalidades do novo SisRTM.

A tecnologia que foi definida para a implementação da nova camada de visão do sistema foi o GWT (Google Web Toolkit) o qual foi utilizado de maneira conjunta com sua extensão, o Ext-GWT (Extended GWT<sup>8</sup>), ambos caracterizados na Seção 3.

Uma vez que a documentação disponibilizada à respeito da ferramenta é bastante vasta, as 30 horas definidas para a realização dessa atividade foram extremamente satisfeitas durante o período de uma semana, de modo que ao fim desse intervalo a maioria dos conceitos do GWT - sejam eles relacionados à arquitetura do *framework* ou até mesmo aos próprios componentes gráficos suportados pelos mesmo – estava bem fixados nos conceitos do estagiário.

---

<sup>8</sup> <http://gwt-ext.com/>

Um outro fator que vale-se a pena se descrito, visto sua importância no processo de aprendizado da tecnologia, foi o fato de que a linguagem de programação ao qual o *framework* da Google é designado é uma plataforma bastante difundida no mercado e também muito transmitida no curso de Ciência da Computação da UFCG: Java (descrita na Seção 3). O fato de o estagiário já ter desenvolvido alguns projetos utilizando-se de determinada plataforma pode ser encarado como um ganho prévio de bagagem que fez com que essa tarefa em questão fosse finalizada com êxito, obedecendo o cronograma estabelecido.

#### **4.4. Desenvolvimento**

A seguir são descritas todas as atividades consistentes em esforços de implementação por parte do estagiário.

Essa descrição será feita abstendo-se da citação de detalhes mais inerentes à implementação, tentando focar um pouco mais na funcionalidade propriamente dita. Detalhes de implementação são abordados numa documentação disponível aos desenvolvedores no repositório da empresa.

##### ***4.4.1. Finalização do desenvolvimento do módulo de Busca de Roteiros na nova camada de visão***

O objetivo principal dessa primeira tarefa de implementação foi aplicar os conceitos resultantes do estudo das ferramentas, além de concluir a implementação de um dos componentes mais importantes da camada de visão que é o componente de busca por roteiros de manobras.

Como o estagiário em questão foi inserido nas atividades do projeto posteriormente ao seu início, já havia, após sua entrada, uma boa quantidade de artefatos de código disponíveis no repositório da empresa. Dessa forma, a elaboração de alguns componentes gráficos já havia sido iniciada, e uma dessas partes já implementada era justamente o elemento gráfico da busca de

roteiros, que ainda era carente de finalização no intuito de inserção de alguns elementos de interface restantes e implementação da lógica de outros desses componentes

Apesar de possuir um escopo bastante restrito, essa primeira atividade permitiu que o estagiário elucidasse as últimas dúvidas à respeito da camada de negócios dessa parte do sistema, além de sanar as primeiras dificuldades relacionadas à arquitetura da aplicação, último fator esse que veio a ser de fundamental importância para o desenrolar das atividades remanescentes.

Na Figura 4.1, é apresentando o resultado da primeira atividade de implementação e primeiro resultado visualizável compreendidos no escopo do estágio integrado.

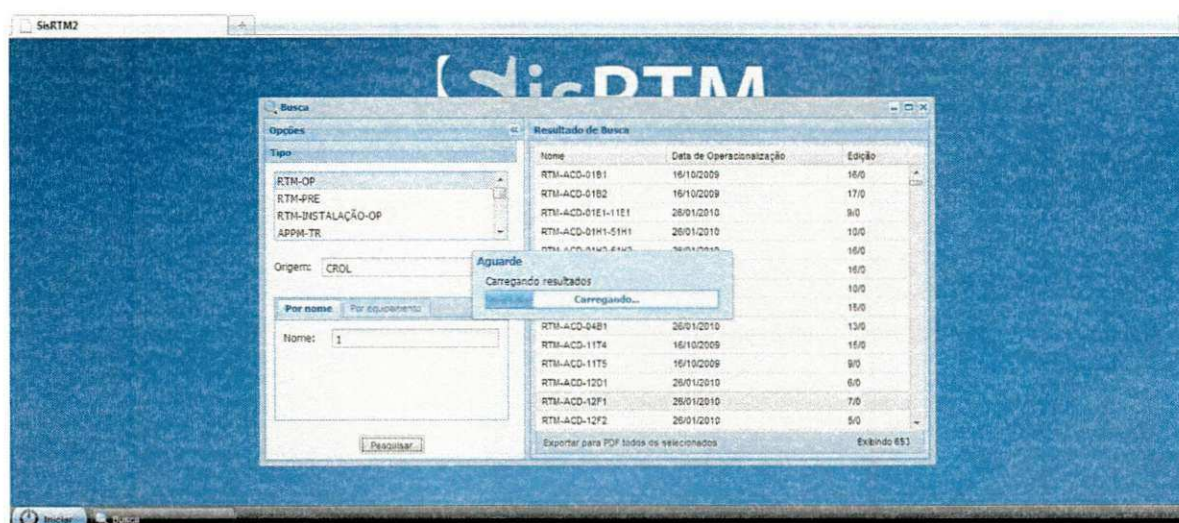


Figura 4.1 – Tela de Busca de Roteiro de Manobras.

#### ***4.4.2. Desenvolvimento da funcionalidade de listagem de roteiros em edição, um complemento do modulo de Busca na nova camada de visão***

Essa segunda tarefa de implementação já possui um caráter novo justamente pelo fato de ser o primeiro componente gráfico da nova camada de visão do SisRTM que foi desenvolvido desde o início pelo estagiário. No que

diz respeito ao fator de aprendizado, essa atividade deixou estabelecido o entendimento dos conceitos da arquitetura do sistema por parte do desenvolvedor.

Em se tratando do panorama da aplicação, o antigo sistema SisRTM dava suporte aos seus usuários para fazer uma listagem de todos os roteiros de manobras disponíveis para edição, isso por que nem todos os roteiros cadastrados na base de dados estão disponíveis para operações de alteração de conteúdo. Roteiro operacionais, por exemplo, que são documentos sob uso frequente dos profissionais da CHESF e, portanto, não podem ser editados a qualquer momento, correndo risco de comprometer a integridade do funcionamento das atividades da companhia, o que, se tratando de uma instituição da magnitude da CHESF, pode ser uma ocorrência catastrófica.

Analisando um pouco o contexto dessa funcionalidade, não é muito difícil perceber que essa listagem nada mais é que um resultado de uma busca na base de dados. Utilizando-se dessa abordagem, a implementação dessa nova tela reaproveitou boa parte dos componentes gráficos do módulo de busca, o que fazendo-se uso de algumas práticas da programação orientada a objetos, permitiu a criação de uma nova funcionalidade sem a necessidade da elaboração de código desnecessário.

Na Figura 4.2, pode ser visualizado o resultado dessa atividade.



The screenshot shows a window titled 'Roteiros em Edição' from the SisRTM application. The window contains a table with three columns: 'Nome', 'Data Última Modificação', and 'Versão'. The table lists 14 rows of route information. At the bottom of the window, there are buttons for 'Exportar para PDF todos os selecionados' and 'Editar', and a status bar indicating 'Exibindo 7/17'.

Nome	Data Última Modificação	Versão
RTM-AGD-14L4	03/12/2009	1
RTM-AGD-05B1	07/10/2009	12
RTM-AGD-05B2	07/10/2009	12
RTM-AGD-05E1	07/12/2009	7
RTM-AGD-05E1-CB	07/03/2008	1
RTM-AGD-05E4-15E4	07/12/2009	2
RTM-AGD-05T1	07/02/2006	3
RTM-AGD-05T2	07/02/2006	2
RTM-AGD-15D1	07/12/2009	3
RTM-AGD-15D2	07/12/2009	6
RTM-AGD-15D3	07/12/2009	6
RTM-AGD-15D4	07/12/2009	4
RTM-AGD-15D5	07/12/2009	3
RTM-AGD-15E1	07/12/2009	6

Figura 4.2 – Tela de Listagem de Roteiros em Edição.

#### **4.4.3. Desenvolvimento do módulo de Templates de Ação na nova camada de visão**

No contexto do SisRTM, um roteiro de manobras é composto por uma série de ações responsáveis pela liberação e normalização de equipamentos da rede elétrica. Ao realizar uma edição de um RTM, é possível inserir novas ações neste, de maneira que o documento se torne cada vez mais efetivo dentro de uma determinada problemática. Utilizando um editor de texto convencional, todas as ações são inseridas livremente mediante alteração direta do documento. Agora, utilizando o SisRTM, para uma ação ser inserida, é necessário escolher primeiramente a qual template ela está relacionada e informar o conteúdo dos parâmetros. Nesse contexto, um template consiste em um texto parametrizado. Por exemplo: se o texto da ação é "CMD – Abrir 15D1", então para que essa ação seja inserida, é necessário escolher o template "\${1} – Abrir \${2}", onde \${1} e \${2} são parâmetros do template, que, nesse caso, são representados por CMD e 15D1, respectivamente.

Uma das telas planejadas durante a fase de levantamento de atividades do projeto foi a de gerenciamento de templates. Foi feito o uso da palavra gerenciamento para englobar as funções de listagem, adição, remoção e edição de templates.

Essa pode ser considerada o primeiro componente um pouco mais complexo do SisRTM, pois engloba uma série de opções que devem ser fornecidas ao usuário.

Como nesse ponto do período de estágio o estagiário já possuía um certo grau de conhecimento do *framework* gráfico, o desenvolvimento dessa tarefa transcorreu de maneira bastante propícia e nenhuma grande dificuldade foi encontrada e, desta forma, a atividade forneceu resultados bastante consideráveis, os quais podem ser visualizados na Figura 4.3.



Figura 4.3 – Tela de Gerenciamento de Templates de Ação.

#### 4.4.4. Desenvolvimento do módulo de Usuários na nova camada de visão

O SisRTM possui um mecanismo no qual cada usuário cadastrado no sistema é dotado de um conjunto de ações relacionadas que podem ser desempenhadas. Um usuário do tipo administrador do sistema, por exemplo, possui respaldo para a execução de todas as operações possíveis fornecidas pela aplicação, enquanto um visualizador de roteiros não possui, por exemplo, a permissão de adicionar um novo usuário na base de dados. Dessa forma, o administrador do sistema possui o poder de adicionar novos utilizadores ao sistema, cada qual com suas respectivas responsabilidades. Essa funcionalidade do sistema é provido por intermédio de uma tela de gerenciamento de usuários, que dá suporte a operações de consulta, edição, remoção e adição de novos usuário.

O módulo em questão também aproveitou a implementação de módulos anteriores, mais necessariamente o pacote de gerenciamento de templates, que continha elementos gráficos que seriam úteis ao componente em destaque. Dessa forma, mais um vez os conceitos de Programação Orientada a Objetos (POO) foram utilizados para fornecer mais funcionalidades com menos código.

Posteriormente, na fase de refatoramento e correção de bugs essa tela foi levemente alterada, de forma que a lista de usuários desenvolvida anteriormente, mostrando o login do usuário, seu email e seu nome completo, passaria a mostrar também a lista de papéis desempenhados por aquele utilizador dentro do sistema. O resultado dessa tela após essas duas atividades pode ser analisado na Figura 4.4.

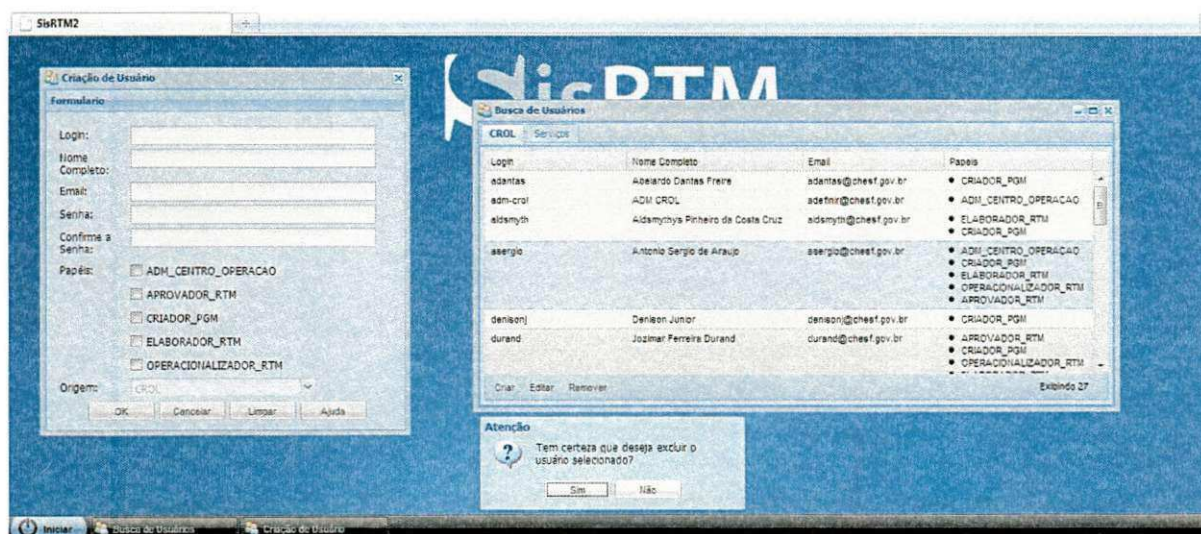


Figura 4.4 – Tela de Gerenciamento de Usuários.

#### 4.4.5. Desenvolvimento do módulo de Origens na nova camada de visão

Dentro da lógica de negócio do SisRTM, uma origem consiste no órgão com o qual um usuário está relacionado; podendo ser um centro de operação, serviço de operação ou instalação, ou seja, através da origem do usuário, é possível recuperar se o usuário está relacionado com um Centro de Operação, Serviço de Operação ou Instalação.

O antigo SisRTM possuía uma tela que consistia num componente de gerenciamento de origens – entendendo-se por gerenciamento aquelas três suboperações citadas anteriormente: criação, edição, remoção e recuperação – por meio do qual o usuário poderia navegar sobre as diversas origens



existentes na CHESF.

Em termos de regras de negócio, existe uma hierarquia de origens consistente no esquema que pode ser analisado na Figura 4.5.

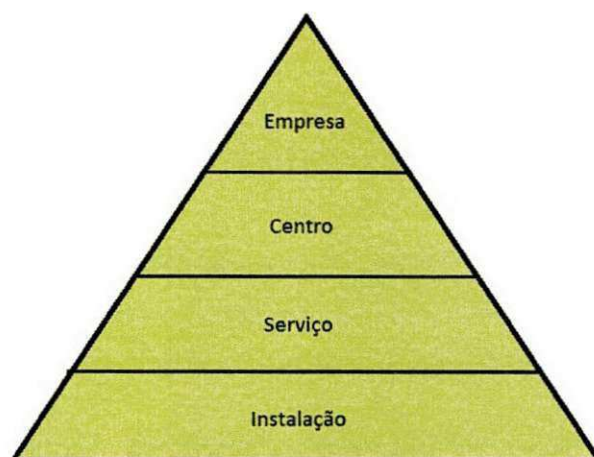


Figura 4.5 – Hierarquia de Origens do SisRTM.

Esse módulo em especial consistiu na elaboração da tela de gerenciamento de todas as origens existentes no SisRTM. Para a montagem desse componente foi necessário uma reunião com o intuito de idealizar o componente gráfico que seria mais coerente a ser utilizados, chegando-se a construção de um elemento gráfico do tipo *TreePanel*, semelhante à estrutura utilizada pelo sistema operacional Windows Microsoft para exibir a hierarquia de diretórios na aplicação Windows Explorer.

Apesar de exigir um pouco mais de cuidado no período de pré implementação, esse componente (Figura 4.6) não exerceu nenhuma influência negativa no processo de desenvolvimento e foi montado corretamente com a ajuda do outro estagiário da empresa.



Figura 4.6 – Tela de Gerenciamento de Origens.

#### **4.4.6. Refinamento dos módulos previamente implementados**

Desempenhadas todas as tarefas relacionadas à remodelagem do SisRTM, foi de fundamental importância testar todas as funcionalidades implementadas e reparar os eventuais defeitos. Para isso, foi implantada no ambiente da CHESF uma versão do sistema compilando todas as funcionalidades desenvolvidas pelos 2 estagiários da empresa. Após um curto período de testes, foi elaborado um documento contendo as impressões dos *stakeholders* e um conjunto de comportamentos considerados como possíveis bugs da aplicação.

De posse desses documentos, o supervisor técnico dividiu todos esses registros e catalogou novas tarefas a serem desenvolvidas pela equipe de desenvolvimento. É importante ressaltar que esse documento não continha apenas sugestões de reparos no SisRTM, mas também impressões dos usuários que geraram a necessidade da implementação de novas funcionalidades à aplicação, e essas novas funcionalidades também foram cadastradas como novas atividades a serem realizadas pelos estagiários da Smartiks.

De uma maneira geral, as tarefas não causaram grande impacto no nível de dificuldade dessa atividade em questão, especialmente por que as mesmas

foram devidamente delegadas àqueles que originalmente implementaram o componente a ser reparado. Eventualmente, uma ou outra incubência exigia uma remodelagem do *backend* da aplicação. Mas, uma vez entendido o modelo de codificação dessa parte do código, até mesmo esses deveres eram realizados sem obstáculos.

Um dos pontos que foi bastante discutido diz respeito ao comportamento associado à tela referente a um usuário não autenticado no sistema, várias opções foram testadas até que uma solução agradável fosse obtida, o resultado dessa tarefa pode ser visualizado na Figura 4.7.



Figura 4.7 – Tela de Login Remodelada.

Após suas respectivas implementações, todas as telas citadas anteriormente eram julgadas pelo supervisor técnico, que avaliava as funcionalidades implementadas e a usabilidade dos componentes. Após o julgo interno, a interface era exibida para os operadores da CHESF, que faziam suas observações, fornecendo o feedback da mesma e, quando necessário, uma série de ajustes nos elementos gráficos.

SEÇÃO V

CONSIDERAÇÕES

FINAIS

## 5. CONSIDERAÇÕES FINAIS

É de comum conhecimento que a lógica de negócio do mercado real possui um grau de cobrança bastante divergente daquele exercido durante o desenvolvimento de atividades de cunho acadêmico. Um fator bastante recorrente no desenvolvimento de software real é o desconhecimento dos clientes à respeito do esforço empregado na implementação de tal sistema. O mandatário do processo não tem discernimento algum de categorizar o grau de dificuldade relacionado a uma determinada atividade, e cabe à gerência de projeto lidar com essa peculiaridade da maneira mais adequada possível. Foi comum durante os trabalhos de implementação da nova camada de visão do SisRTM a disponibilidade de incubências com um adicional de complexidade alta e, sendo assim, foram necessárias etapas de planejamento definidas, de modo que não houvesse a criação de um escopo grande demais para a manipulação do estagiário.

Apesar de vir como uma característica à priori negativa, essa cobrança demasiada carrega consigo um efeito colateral que é justamente a elaboração de estratégias alternativas para a implementação de determinados comportamentos, e isso foi visível no projeto em destaque. A escolha do framework GWT veio com o intuito de satisfazer uma determinada funcionalidade considerada simples do ponto de vista do cliente, mas que exigiu uma total reformulação na estrutura empregada até então. Os aspectos positivos trazidos junto desse fator é que o aprendizado de uma nova plataforma por parte dos envolvidos promove um aumento considerável do seu conhecimento, e isso pode (e deve) se mostrar de fundamental importância nas oportunidades a serem obtidas posteriormente.

É importante perceber que uma série de dificuldades foi mostrada durante o período de desenvolvimento de estágio. Tais obstáculos caracterizaram-se pelo desconhecimento das tecnologias envolvidas no processo e também, por vezes, uma visão turva da arquitetura do sistema SisRTM. Dificuldades são partes comuns da realização de qualquer tipo de projeto, tal qual a idealização de novas soluções de acordo com o domínio da problemática e urgência da pendência. No caso em particular, esse pontos

foram superados por intermédio do estudo da plataforma aliado ao auxílio fornecido pela equipe de desenvolvimento do SisRTM.

Por intermédio das atividades de estágio integrado foi possível entender um pouco mais da lógica de negócios aplicada ao mundo "real". É sempre importante em qualquer área presenciar situações de aplicação dos conhecimentos no dia-a-dia das pessoas, o que no exemplo do estagiário em especial fez com que este se sentisse cada vez mais preparado para tentar conduzir à frente novos desafios e até mesmo cogitar a eventual criação do seu próprio empreendimento.

Com relação à prática da disciplina de Estágio Integrado no curso de graduação em Ciência da Computação, esta pode servir como ferramenta bastante eficaz na modelagem de profissionais responsáveis na área, os quais serão rapidamente absorvidos pelo mercado de trabalho. Um ponto que pode assustar um pouco os candidatos a cursantes de tal disciplina é a carga burocrática demasiada ao qual o processo de estágio é submetido. Característica essa que, por vezes, fornece uma série de inconveniente que poderiam ser evitados caso esse conjunto de documentos fosse diminuído. Fora isso, é muito fácil perceber que a disciplina oferecida pelo departamento tem todos os pre-requisitos para ser bem aproveitada pelos alunos.



REFERÊNCIAS

BIBLIOGRÁFICAS

## 6. REFERÊNCIAS BIBLIOGRÁFICAS

TELES, Vinicius. **Extreme Programming, XP**: metodologia desenvolvimento ágil. Improve It, disponível em <http://improveit.com.br/xp>. Acesso em: 5 de dezembro de 2009.

IEEE COMPUTER SOCIETY. **SWEBOK®**: A project of the IEEE Computer Society Professional Practices Committee. Estados Unidos: IEEE Computer Society, 2004. LINDHOLM, Tim; YELLIN, Frank. **The Java™ Virtual Machine Specification**: Second Edition. Disponível em [http://java.sun.com/docs/books/jvms/second\\_edition/html/VMSpecTOC.doc.htm](http://java.sun.com/docs/books/jvms/second_edition/html/VMSpecTOC.doc.htm). Acesso em: 6 de dezembro de 2009.

DEITEL, Paul J.; DEITEL, Harvey M. **Java™**: How to program. 7.ed. Estados Unidos: Pearson Education, 2007.

SOMMERVILLE, Ian. **Software Engineering**. 8.ed. China: Pearson Education, 2007.

ROCHA, Eloi *et al.* **SISRTM - SISTEMA DE ROTEIRO DE MANOBRAS**. In: X EDAO – ENCONTRO PARA DEBATES DE ASSUNTOS DE OPERAÇÃO, 10, 2008, São Paulo.





# APÊNDICES

## 7. APÊNDICES

### *APÊNDICE A – Plano de Estágio*



UFPA – UNIVERSIDADE FEDERAL DO PARÁ

CEEI – CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA

DSC – DEPARTAMENTO DE SISTEMAS E COMPUTAÇÃO

## **Plano de Estágio**

***Smartiks Tecnologia da Informação LTDA***

**Felipe Vieira Falcão Mat.: 20811015**

Orientador: Tiago Lima Massoni

Agosto 2010

## **8. Informações Pessoais**

---

**Nome:** Felipe Vieira Falcão

**Matrícula:** 20811015

**Endereço:** Rua Olga de Azevedo Oliveira, 460  
Nova Brasília – Campina Grande, PB – Brasil

**Email:** felipe@smartiks.com

## **9. Ambiente do Estágio**

---

O ambiente de estágio é o escritório da empresa Smartiks, localizado na sala 06 da Fundação Parque Tecnológico da Paraíba, situada na Rua Emiliano Rosendo Silva - 115, no bairro de Bodocongó.

A carga horária semanal é de 20 horas, sendo parte dela cumprida no ambiente do estágio e parte em casa.

O trabalho será realizado num grupo de 4 (quatro) pessoas, sendo dois desenvolvedores, um líder técnico e um gerente.

***Supervisão Acadêmica***

**Nome:** 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:** massoni@dsc.ufcg.edu.br

***Supervisão Técnica***

**Nome:** Eloi Rocha Neto

**Endereço:** Rua Emiliano Rosendo Silva, 115, sala 06

Bodocongó, Campina Grande, PB – Brasil.

**Email:** eloi.rocha@gmail.com

**11. Resumo do Problema**

---

A Smartiks Tecnologia da Informação é uma empresa atuante no mercado de desenvolvimento de software, desenvolvendo soluções inteligentes para empresas de Distribuição, Transmissão e Geração de Energia Elétrica. Está situada na Rua Emiliando Rosendo Silva, 116, sala 06 no Bairro de Bodocongó em Campina Grande, e tem trabalhado em parceria com a Universidade Federal de Campina Grande (UFCG) e a Fundação Parque Tecnológico da Paraíba (PaqTcPB).

Entre os principais projetos realizados pela empresa, está o SisRTM - sistema web desenvolvido na linguagem de programação JAVA para gerência de Roteiros de Manobras. Entende-se por roteiros de manobras documentos operacionais que

descrevem as ações a serem realizadas para liberar e normalizar equipamentos da rede elétrica.

Após utilizar a versão anterior do SisRTM, a CHESF (Companhia Hidroelétrica do São Francisco) solicitou que algumas melhorias fossem realizadas no sentido de melhorar a performance de execução do sistema, principalmente no que diz respeito a otimizações nas operações envolvidas durante o processo descrito pelo uso do sistema.

## **12. Objetivos do estágio**

---

O principal objetivo do estágio é evoluir o SisRTM, refazendo toda a camada de apresentação do software para eliminar totalmente o uso de *refresh* (atualização da página), melhorando a performance do sistema e tornando a interface mais amigável.

Para resolver os problemas de desempenho da versão implantada na CHESF, será utilizada a tecnologia GWT no desenvolvimento da nova interface gráfica. O Google Web Toolkit (GWT) facilita esse processo, permitindo que os desenvolvedores criem rapidamente e mantenham aplicativos JavaScript complexos e de alto desempenho na linguagem de programação Java.

## **13. Metodologia**

---

O sistema será desenvolvido de maneira iterativa e incremental, seguindo práticas de desenvolvimento de software adaptadas para a nossa realidade e baseadas em processos de desenvolvimento existentes no mercado, tais como: eXtreme Programming [2] e easYProcess [1].

A equipe será responsável pelas práticas concernentes ao entendimento do sistema, manutenção e testes do sistema e implantação.

## 14. Atividades Planejadas

---

Devem ser desenvolvidas as seguintes atividades durante o estágio:

<b>Atividades</b>	<b>Horas estimadas</b>
Entendimento da estrutura do sistema	20
Estudo da tecnologia (GWT)	30
Refazer página de busca de templates de roteiros	60
Refazer página de busca de roteiros	70
Refazer página de visualização de roteiros	30
Refazer demais páginas do sistema	50
Desenvolvimento do módulo de sincronização de roteiros	70
Elaboração e defesa do relatório de estágio	30
<b>Total de Horas</b>	<b>360</b>

## 15. Resultados Esperados

---

Espera-se que, conforme o cronograma apresentado, ao fim do estágio, a nova versão do sistema esteja 100% funcional com as principais funcionalidades da versão original implementadas e o problema de desempenho resolvido (isto é, nenhum *refresh*).

Esta nova versão, denominada SisRTM2, e que já está em processo de desenvolvimento, será implantada em um ambiente de testes onde o usuário irá realizar todos os testes necessários para validar o produto. Uma vez que o novo sistema estiver operacional e aprovado em todos os testes, ele irá então substituir o antigo sistema da CHESF.

## 16. Cronograma de Atividades

---

As atividades propostas serão distribuídas da seguinte forma ao longo dos próximos 5 (cinco) meses:

Atividades	Ago	Set	Out	Nov	Dez
Entendimento da estrutura do sistema	X				
Estudo da tecnologia (GWT)	X				
Refazer página de busca de roteiros	X				
Refazer página de visualização de roteiros	X				
Refazer página de edição de roteiros	X	X	X	X	
Desenvolvimento do módulo de sincronização de roteiros			X	X	X
Refazer demais páginas do sistema				X	X
Elaboração e defesa do relatório de estágio		X	X	X	X

## 17. Bibliografia

---

[1] GARCIA, F. P. et al. easYProcess: um processo de desenvolvimento de software para uso no ambiente acadêmico. In: XII Workshop de Educação em Informática - XXIV Congresso da Sociedade Brasileira de Computação, 2004, Salvador.

[2] BECK, K., Extreme Programming Explained: Embrace Change. 2000, Reading, Massachusetts: Addison-Wesley.

## 18. Aprovação

---

---

Eloi Rocha Neto

Supervisor Técnico

---

Tiago Lima Massoni

Supervisor Acadêmico

---

Joseana Macêdo Fachine

Coordenadora da Disciplina Estágio Integrado