

DANIEL GUIMARÃES DO LAGO

**SOAC: PROPOSTA DE UM SISTEMA ONLINE DE AUXÍLIO AO
CAFEICULTOR COM FOCO NA MOBILIDADE**

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

**LAVRAS
MINAS GERAIS – BRASIL**

2007

DANIEL GUIMARÃES DO LAGO

**SOAC: PROPOSTA DE UM SISTEMA ONLINE DE AUXÍLIO AO
CAFEICULTOR COM FOCO NA MOBILIDADE**

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

Área de Concentração:

Desenvolvimento de *software* para Dispositivos Móveis

Orientador:

Dr. Luiz Henrique Andrade Correia

**LAVRAS
MINAS GERAIS – BRASIL
2007**

DANIEL GUIMARÃES DO LAGO

**SOAC: PROPOSTA DE UM SISTEMA ONLINE DE AUXÍLIO AO
CAFEICULTOR COM FOCO NA MOBILIDADE**

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

Aprovada em 16 de janeiro de 2008.

Prof. Dr. Luiz Henrique Andrade Correia

Prof^ª. Dra. Marluce Rodrigues Pereira

Prof. Dr. André Luiz Zambalde

**LAVRAS
MINAS GERAIS - BRASIL**

A Deus,
Ao meu pai, Maurício;
À minha mãe, Vânia;
Aos meus avôs,
Tomé (em memória) e
Joaquim (em memória);
Às minhas avós,
Teresinha e Lolita;
Aos meus irmãos,
Davi, Lucas, Luísa e André.

AGRADECIMENTOS

Agradeço a *Deus* pelo mérito desta conquista.

Aos meus familiares, pelos ensinamentos de justiça, honestidade e trabalho.

À minha mãe, Vânia, pela firmeza e amor irrestritos.

Ao meu pai, Maurício, pelo caráter, honestidade e discernimento.

Ao tio Emanuel, pela amizade.

À minha avó Teresa, pela serenidade e estímulo; à avó Lolita, pela bondade; aos meus irmãos, pelo carinho.

Ao meu querido avô, Tomé, que se projetou em minha mente como um grande exemplo a ser seguido. À tia Mary, minha saudade; ao avô Joaquim, meu respeito.

Ao meu orientador Luiz Henrique, pela disponibilidade e dedicação.

Ao amigo Igor Lima, cujo auxílio foi fundamental para conclusão deste trabalho.

Aos mestres do DCC/UFLA, que contribuíram para essa vitória e, em especial, Luiz Henrique, Guilherme, André Saúde, Marluce, Joaquim, Ahmed, Rudini, Luciano, Samuel, Monserrat, Olinda, Mário, Deive, André Zambalde, Renata e Heitor.

Aos mestres de outros departamentos da UFLA, principalmente os de Química, que influenciaram positivamente em minha formação.

Aos funcionários do DCC/UFLA pela amizade e informações precisas, especialmente Clayton, Ângela e Deivson.

Aos meus amigos que, de forma direta ou indireta, muito colaboraram para que eu transpusesse essa trajetória.

SOAC: Proposta de um Sistema Online para Auxílio ao Cafeicultor com foco na Mobilidade

RESUMO

Este trabalho visa a elaboração de um processo de estudo e desenvolvimento de um *software* para auxílio ao gerenciamento da cultura cafeeira. O estudo constituiu-se na pesquisa das necessidades de administradores rurais da área cafeeira e, de acordo com os dados coletados, iniciar o desenvolvimento do *software* sugerido. Este *software* servirá como ferramenta para o acesso remoto de administradores rurais a um banco de dados centralizado, obtenção e armazenamento de informações organizadas e, desta forma, permitir a mobilidade destes administradores. É foco deste trabalho também permitir que tal acesso seja realizado através de computadores *desktops*, *laptops*, *handhelds* e telefones celulares, possibilitando a computação pervasiva.

Palavras-chave: Software, Redes de Computadores, Administração Rural, Mobilidade, Computação Pervasiva/Ubiqua.

SOAC: Proposal of Online System to Aid the Coffee Producer with focus in Mobility

ABSTRACT

The objective of this work is to elaborate a process of study and development of a software to help the management of coffee production. This study will search the needs of rural managers of coffee area and, according to the collected data, start the development of the suggested software. This software will be a tool that allows the remote access of rural managers to a centralized database and management of organized information, thus allowing the mobility of these managers. It is focus of this work to allow such access through desktop, laptop, handheld computers and cell phones; thus making possible ubiquitous computing.

Keywords: Software, Computer Networks, Rural Management, Mobility, Ubiquitous Computing

SUMÁRIO

RESUMO.....	vi
ABSTRACT.....	vi
LISTA DE FIGURAS.....	x
LISTA DE SIGLAS.....	xiii
1. INTRODUÇÃO.....	1
1.1. CONTEXTUALIZAÇÃO E MOTIVAÇÃO.....	2
1.2. OBJETIVOS.....	5
1.3. ESTRUTURA DO TRABALHO.....	6
2. REFERENCIAL TEÓRICO.....	7
2.1. GERENCIAMENTO DO CAFÉ.....	7
2.2. INFORMÁTICA NA CAFEICULTURA.....	8
2.3. BANCO DE DADOS.....	10
2.4. SISTEMA GERENCIADOR DE BANCO DE DADOS (SGBD).....	12
2.5. MySQL.....	13
2.6. MOBILIDADE.....	15
2.7. PORTABILIDADE.....	16
2.8. XHTML (eXtensible Hypertext Markup Language).....	17
2.9. PHP (PHP Hypertext Preprocessor).....	19
2.10. CSS (Cascading Style Sheets).....	21
2.11. JAVA.....	23
2.12. J2ME (Java 2 Micro Edition).....	26
3. METODOLOGIA.....	29
3.1. TIPO DE PESQUISA.....	29
3.2. PROCEDIMENTOS METODOLÓGICOS.....	29
3.2.1. LEVANTAMENTO E ANÁLISE DE REQUISITOS.....	30
3.2.2. MODELAGEM DO SISTEMA.....	33
3.2.3. DESENVOLVIMENTO DO SISTEMA.....	37
4. SOAC: DESCRIÇÃO DO SISTEMA.....	45
4.1. REGISTRO INICIAL DE USUÁRIO.....	45

4.2. MENU PRINCIPAL.....	48
4.3. CADASTROS DE INSUMOS.....	51
4.3.1. TIPOS DE INSUMOS.....	51
4.3.2. INSUMOS.....	53
4.3.3. LOJAS DE INSUMOS.....	56
4.4. CADASTROS DE FUNCIONÁRIOS E PARÂMETROS SALARIAIS.....	59
4.4.1. FUNCIONÁRIOS.....	59
4.4.2. PARÂMETROS DE SALÁRIOS.....	62
4.5. CADASTROS E CONTROLE DE DESPESAS.....	65
4.5.1. AQUISIÇÕES DE INSUMOS.....	66
4.5.2. OUTRAS DESPESAS.....	70
4.5.3. PAGAMENTOS DE FUNCIONÁRIOS.....	72
4.6. CADASTRO DE RECEITAS.....	76
4.6.1. PRODUTOS.....	77
4.6.2. VENDAS DE PRODUTOS.....	78
4.6.3. OUTRAS RECEITAS.....	81
4.7. GERAÇÃO DE RELATÓRIOS.....	82
4.7.1. RECIBOS.....	82
4.7.2. RELATÓRIOS SIMPLIFICADOS.....	87
4.8. UTILITÁRIOS.....	89
4.8.1. AGENDA.....	90
4.8.2. LISTA TELEFÔNICA.....	92
4.8.3. ALTERAÇÕES DE DADOS DE LOGIN.....	94
4.9. SOAC: DECISÕES DE DESENVOLVIMENTO.....	95
5. SoacCel: DESCRIÇÃO DO SISTEMA.....	100
5.1. SoacCel: VISÃO GERAL.....	100
5.2. SoacCel: LOGIN.....	102
5.3. SoacCel: MENU PRINCIPAL.....	104
5.4. CONSULTA POR FUNCIONÁRIOS.....	106
5.5. RELATÓRIOS SIMPLIFICADOS.....	108
5.6. AGENDA.....	110
5.7. LISTA TELEFÔNICA.....	112

5.8. DECISÕES DE DESENVOLVIMENTO.....	113
6. CONCLUSÕES.....	115
6.1. CONSIDERAÇÕES FINAIS.....	115
6.2. TRABALHOS FUTUROS.....	115
7. REFERÊNCIAS BIBLIOGRÁFICAS.....	117

LISTA DE FIGURAS

Figura 1: Interação do Sistema Gerenciador de Banco de Dados com o SOAC.....	14
Figura 2: Interação entre PHP (Middleware), MySQL (SGBD), Apache (servidor de HTTP [ex: páginas XHTML]) e o Navegador do usuário do sistema.....	20
Figura 3: Demonstração Hierárquica do J2ME com o J2SE.....	27
Figura 4: Diagrama de Casos de Uso do SOAC.....	34
Figura 5: Modelo do Banco de Dados utilizado pelo SOAC e SoacCel.....	35
Figura 6: Registro de Usuário visto de um computador desktop.....	46
Figura 7: Menu Principal visto de um computador desktop.....	49
Figura 8: Cadastro de Tipos de Insumos visto de um computador desktop.....	52
Figura 9: Cadastro de Tipos de Insumos após inserção do tipo "Herbicida".....	53
Figura 10: Cadastro de Insumos visto através de um computador desktop.....	54
Figura 11: Cadastro de Insumos após inserção do insumo "Round Down".....	55
Figura 12: Cadastro de Lojas de Insumos visto através de um computador desktop.....	57
Figura 13: Cadastro de Lojas de Insumos após inserção da loja "Coocidade".....	58
Figura 14: Cadastro de Funcionários visto através de um computador desktop.....	60
Figura 15: Cadastro de Funcionários após inserção de "João da Silva".....	62
Figura 16: Cadastro de Parâmetros de Salários visto através de um computador desktop..	64
Figura 17: Cadastro de Aquisições de Insumos visto através de um computador desktop..	67
Figura 18: Cadastro de Aquisições de Insumos após inserção da aquisição de 1x Round Down obtido da loja Coocidade, para demonstração do sistema.....	69
Figura 19: Cadastro de Outras Despesas visto através de um computador desktop.....	70
Figura 20: Cadastro de Outras Despesas após inserção de uma despesa.....	71
Figura 21: Cadastro de Pagamentos de Funcionários visto através de um computador desktop.....	73
Figura 22: Cadastro de Pagamentos de Funcionários visto através de um computador desktop: Foco nos dados do pagamento.....	74
Figura 23: Cadastro de Pagamentos de Funcionários visto através de um computador desktop após a inserção de um pagamento, foco na listagem dos pagamentos sumarizada.....	75

Figura 24: Cadastro de Produtos visto através de um computador desktop.....	77
Figura 25: Cadastro de Produtos visto através de um computador desktop após inserção do produto café.....	78
Figura 26: Cadastro de Vendas de Produtos visto através de um computador desktop.....	79
Figura 27: Cadastro de Vendas visto através de um computador desktop, após inserção de uma venda.....	80
Figura 28: Cadastro de Receitas visto através de um computador desktop.....	81
Figura 29: Geração de Recibos visto de um computador desktop.....	84
Figura 30: Recibos Gerados visto de um computador desktop.....	86
Figura 31: Geração de Relatórios Simplificados visto de um computador desktop.....	87
Figura 32: Relatórios Simplificados gerados visto de um computador desktop.....	88
Figura 33: Utilitário Agenda visto de um computador desktop.....	91
Figura 34: Utilitário Lista Telefônica visto de um computador desktop.....	92
Figura 35: Utilitário Lista Telefônica após ter efetuado uma procura.....	93
Figura 36: Utilitário Alteração de Dados de Login visto de um computador desktop.....	95
Figura 37: SoacCel: Diagrama de Fluxo.....	101
Figura 38: SoacCel: Exibição do Login visto de um Emulador de Telefone Celular.....	103
Figura 39: SoacCel: Exibição de Mensagem de Erro de Login visto de um Emulador de Telefone Celular.....	104
Figura 40: SoacCel: Exibição do Menu Principal visto de um Emulador de Telefone Celular.....	105
Figura 41: SoacCel: Exibição da Consulta por Funcionários vista de um Emulador de Telefone Celular.....	106
Figura 42: SoacCel: Exibição da Consulta por Funcionários vista de um Emulador de Telefone Celular, após efetuar a consulta.....	107
Figura 43: SoacCel: Geração de Relatórios Simplificados vista de um Emulador de Telefone Celular.....	108
Figura 44: SoacCel: Exibição da Geração de Relatórios Simplificados vista de um Emulador de Telefone Celular, após a geração.....	109
Figura 45: SoacCel: Exibição da Agenda vista de um Emulador de Telefone Celular.....	110
Figura 46: SoacCel: Exibição da Agenda vista de um Emulador de Telefone Celular, informando os compromissos em datas próximas a atual.....	111

Figura 47: SoacCel: Exibição da Lista Telefônica vista de um Emulador de Telefone Celular.....112

LISTA DE SIGLAS

CLDC – *Connected Limited Device Configuration*; Configuração de Dispositivos Limitados Conectados

HTML – *Hypertext Markup Language*; Linguagem de Marcação de Hipertexto

KVM – *K Virtual Machine*; Máquina Virtual K

J2ME – *Java 2 Micro Edition*; Edição Micro do *Java 2*

MIDP – *Mobile Information Device Profile*; Perfil de Informações de Dispositivos Móveis

PHP – *PHP Hypertext Preprocessor*; Pré-processador de Hipertexto *PHP*

SOAC – Sistema Online de Auxílio ao Cafeicultor com foco na Mobilidade

SoacCel – Sistema Online de Auxílio ao Cafeicultor para Celular

SQL – *Structured Query Language*; Linguagem Estruturada de Consultas

UML – *Unified Modeling Language*; Linguagem de Modelagem Unificada

1. INTRODUÇÃO

A atividade cafeeira tem fundamental posição no mercado mundial em virtude do café ser um produto de consumo básico. No Brasil, é notável sua importância.

O café é importante para o Brasil desde a época do império. Segundo dados da EMBRAPA (2007), é possível afirmar que mesmo nessa época a produção brasileira de café atingia 40% da produção mundial e chegou a contribuir isoladamente com 70% do valor de nossas exportações, no período de 1925 até 1929. Nos anos de 1987, até 2000, cerca de 25% da produção mundial de cerca de 100 milhões de sacas de café ao ano, eram provenientes do Brasil. Em 1997, esse produto gerou mais de três bilhões de dólares em receitas cambiais para o país. Embora o café tenha, ao longo do tempo, diminuído sua participação nas exportações brasileiras, devido à diversificação de produtos exportados, constitui-se, ainda hoje, um expressivo gerador de divisas.

Hoje, o Brasil possui uma área plantada de 2,7 milhões de hectares, com aproximadamente seis bilhões de pés, sendo concentrada a maior parte só no Estado de Minas Gerais. Com uma exportação média de 28 milhões de sacas, é o principal exportador e responde por mais de um terço de toda a produção mundial. O setor é responsável pela geração de sete milhões de empregos diretos e indiretos no país e por uma riqueza anual de 10 bilhões de reais.

Com relação às tecnologias aplicadas ao gerenciamento da cultura cafeeira, é perceptível a evolução, mas ainda estão bastantes subdesenvolvidas. Pesquisas demonstram que a maioria dos produtores rurais brasileiros encaram positivamente a idéia de se utilizar *softwares* para o auxílio do gerenciamento rural: segundo uma pesquisa realizada por Simião (UNIVAG, 2007) constatou que apenas 18,6% dos produtores pesquisados utilizam-se dos recursos da informática e, destes, só 5,7% utilizam-se de *softwares* voltados à administração rural e observou-se que a maioria dos produtores (75,7%) demonstrou interesse em utilizar programas de administração rural. Um dos motivos alegados pelos produtores, em não utilizar *softwares* para auxílio administrativo, está relacionado à falta de pessoal capacitado para operacionalizar os programas

disponíveis. A demanda é por produtos, *softwares*, que realizem controle de todas as atividades, abrangendo do pré-plantio até a colheita e comercialização de seus produtos. Além destas características, foi constatado que o *software* deve ser de fácil operação, com apoio e manutenção constantes.

1.1. CONTEXTUALIZAÇÃO E MOTIVAÇÃO

Com a evolução tecnológica observada, atualmente, nas mais diversas áreas do conhecimento humano, as empresas vêem a necessidade de melhorar sua capacidade de comunicação e gerência de informações constantemente.

No setor agrário, são verificadas dificuldades no que diz respeito à comunicação precisa, especialmente nos casos das pequenas empresas. Por exemplo: um administrador rural necessita viajar, porém possui obrigações, como: pagamento de empregados, venda de produtos, etc. Não sendo possível conciliar ambas operações simultaneamente, deixa uma pessoa encarregada para essa segunda tarefa, a qual depende de dados sobre o procedimento da colheita do produto e dos pagamentos dos funcionários; noções da quantia de insumos que deverão ser comprados; alguma ação a ser tomada como contrair empréstimos bancários para provisionar recursos para o pagamento da colheita, entre outras atividades.

Notam-se diversas dificuldades quanto à comunicação necessária entre o administrador rural e a pessoa responsável pelos pagamentos, pois a referida pessoa não tem as informações necessárias, de forma precisa, como é requerida pelo administrador. Aqui se percebe a necessidade de uma ferramenta tecnológica que poderia auxiliar esse tipo de trabalho.

Ainda, analisando o cenário anterior, é interessante a existência de um *software* que utilize as redes de computadores para a comunicação entre o administrador rural e

terceiros. Tal *software* poderia executar da seguinte maneira: à medida em que a pessoa responsável pelos pagamentos dos empregados fosse cadastrando a quantia do produto colhido e os pagamentos efetuados, um administrador, em qualquer local do mundo, que acessasse à Internet poderia obter informações armazenadas em um banco de dados. Este *software* também poderia auxiliar as tomadas de decisões e facilitar o gerenciamento da empresa.

Na Universidade Federal de Lavras, é notável crescente preocupação com a elaboração de ferramentas que auxiliem o setor do café. No departamento de Ciência da Computação, por exemplo, várias monografias foram desenvolvidas com foco neste setor.

Ferreira (2001) foi provavelmente o primeiro graduando de Ciência da Computação na Universidade Federal de Lavras a demonstrar preocupação com o setor cafeeiro. Criou um *software* com base construída em um banco de dados capaz de lidar com incertezas e auxiliar a tomada de decisões que antes era feita apenas por seres humanos. Para tornar isto possível, tratou as informações dadas pelo usuário utilizando *lógica fuzzy* (lógica utilizada para tratamento de incertezas, também chamada de *lógica nebulosa*) para inferir respostas aproximadas do conhecimento inexato passado, e desta forma tomar a decisão mais adequada.

Santiago (2002) também se preocupou com o setor cafeeiro. Desenvolveu um *software* que encontra padrões no banco de dados da tabela de negociações e cotações da *commodity* café da BM&F (Bolsa de Mercadorias & Futuros). O programa gera informações sobre o andamento da *commodity*, que são de grande interesse para um profissional com objetivos em facilitar a tomada de decisões.

Castro (2003), focando a área de Recuperação de Informação aliada à Inteligência Artificial, implementou um sistema para recuperação de informações na *web* utilizando agentes inteligentes para auxiliar o agronegócio do café. Basicamente, os dados com os quais ele se preocupou no desenvolvimento do sistema foram informações econômicas e conjunturais, que apresentam condições favoráveis ou desfavoráveis em relação aos preços

do café, a disponibilidade e custo dos insumos e da mão-de-obra, o custo e a disponibilidade de crédito, valores das bolsas etc; informações do ambiente, com dados sobre as condições climáticas, períodos de colheitas, entre outros; informações sobre o manejo da cultura, informando sobre problemas e soluções sobre o manejo da cultura cafeeira, como problemas do solo, sistemas de cultivo, cultivares, espaçamento, adubação etc.

Santos (2005) demonstrou também preocupação com este setor carente. Fazendo parte de um grupo de desenvolvimento de *software* para esta área, elaborou uma monografia retratando o trabalho efetuado juntamente com uma equipe para o desenvolvimento de um Sistema Especialista para analisar o custo dos processamentos do café na fase pós-colheita, com o objetivo de facilitar as decisões dos produtores, cooperativas e consultores da área quanto a quais processamentos devem ser aplicados ao produto.

Como se pôde perceber, nesta universidade muitos trabalhos foram desenvolvidos para o setor cafeeiro, mas não houve preocupação com o setor produtivo em nenhum *software* desenvolvido em monografias prévias.

Realizando uma extensiva pesquisa na Internet e nos *softwares* existentes para a área, é possível verificar também que são raros os *softwares* que abordam o setor produtivo de forma eficiente. Essa ineficiência pode ser provada com um exemplo desta própria universidade: a Universidade Federal de Lavras contém em sua estrutura o Centro de Inteligência em Mercados (CIM). O objetivo do CIM é o desenvolvimento de produtos e serviços tecnológicos para diversos segmentos e áreas, atuando tanto no âmbito público como no privado. Tendo notado esta deficiência, o CIM está desenvolvendo em um de seus projetos a Plataforma de Gestão de Custos, especialmente criada para o setor cafeeiro. O objetivo é criar uma ferramenta que gerencie os custos de produção e auxilie o controle da produção. Segundo informações obtidas do próprio CIM constatou-se que após o término do desenvolvimento desta ferramenta ela será disponibilizada na Internet por meio de acesso restrito, onde mediante o pagamento de uma periódica taxa o cliente do sistema

poderá ter acesso aos recursos da plataforma desenvolvida (CIM, 2007).

A proposta do presente trabalho é proporcionar aos administradores rurais um Sistema Online de Auxílio aos Cafeicultores (SOAC), objetivando o que nenhum trabalho anteriormente focou: a mobilidade no gerenciamento de produção, recursos humanos e informações úteis pertinente ao setor produtivo cafeeiro.

1.2. OBJETIVOS

Observando as deficiências apresentadas na seção anterior, foi possível perceber que ainda há uma escassez de *softwares* que auxiliem no controle da produção do setor cafeeiro, principalmente aqueles que foquem na mobilidade do administrador rural.

O principal objetivo do presente trabalho é averiguação e análise das principais necessidades do setor produtivo cafeeiro, por meio de pesquisas informais com administradores rurais e, com base nos dados coletados, desenvolver um *software*, que sirva como ferramenta de auxílio às tomadas de decisões, controle de receitas e despesas e que realize projeções pertinentes para facilitar o gerenciamento da empresa. Ainda, no âmbito principal deste projeto, este *software* deve permitir a mobilidade, mais especificamente que o acesso ao sistema seja possível não apenas através de computadores *desktops*, mas também *laptops*, *handhelds* e *telefones celulares* para dessa forma contribuir com a computação pervasiva.

O sistema é uma aplicação *web* e foi batizado como SOAC (Sistema Online de Auxílio ao Cafeicultor), e existe também a versão que possibilita o acesso via telefone celular, batizado de SoacCel. Nos capítulos posteriores os mesmos serão detalhados.

Como objetivos secundários deste trabalho, pode-se destacar o desenvolvimento de um sistema que organize recursos e informações.

1.3. ESTRUTURA DO TRABALHO

O presente trabalho encontra-se organizado em 5 capítulos, ordenados com a seguinte divisão:

No capítulo 1 é apresentada a construção do trabalho realizado, descrevendo o contexto, motivações e objetivos.

O capítulo 2 discute os conceitos de administração rural voltada à cafeicultura, recursos os quais esse projeto utilizará e apresentar conceitos das ferramentas as quais serão utilizadas durante a pesquisa e desenvolvimento do sistema proposto.

O capítulo 3 apresenta a metodologia utilizada na elaboração desse trabalho, o tipo de pesquisa e os procedimentos metodológicos.

No capítulo 4 é mostrado um *software* que foi desenvolvido como resultado deste projeto, o SOAC, bem como suas características e funcionalidades.

O capítulo 5 apresenta outro *software* desenvolvido para a pesquisa visando a mobilidade para acesso via telefones celulares, o SoacCel, e também suas funcionalidades.

No capítulo 6 estão presentes as considerações finais do presente projeto, conclusões e propostas para trabalhos futuros.

2. REFERENCIAL TEÓRICO

Este projeto é baseado no conhecimento dos recursos aplicados ao gerenciamento do café necessários ao empresário rural.

O objetivo deste capítulo é mostrar um embasamento teórico usado no desenvolvimento do sistema. Portanto, nas próximas seções serão abordados conceitos sobre gerenciamento do café, banco de dados, sistema gerenciador de banco de dados, mobilidade, portabilidade, desenvolvimento *web*, servidores e desenvolvimento para dispositivos móveis.

2.1. GERENCIAMENTO DO CAFÉ

Como o objetivo deste trabalho é o desenvolvimento de um *software* com o foco na área cafeeira, é de fundamental importância noções sobre o gerenciamento do café. Nesta seção serão abordados aspectos que permitem identificar quais são as principais necessidades dos cafeicultores relacionadas ao setor produtivo que podem ser satisfeitas por um *software*.

É importante conhecer os objetivos nos quais o projeto irá atuar. Para escolher o propósito de aplicação do mesmo, deve-se considerar que existem basicamente três divisões no sistema de produção rural, nas quais devem ser centradas nossa atenção. Andrade (1999, p. 4) enumera estas divisões da seguinte forma: os insumos são as entradas de fora da propriedade; processo sendo a produção, dentro da propriedade; e como saídas, fora da propriedade, o processamento e a distribuição.

Andrade (1999, p. 46) também alude que um dos grandes problemas do fazendeiro-empresário rural é saber qual é a melhor utilização que se deve dar aos recursos escassos: saber o que e quanto produzir na propriedade. De um modo geral, o empresário tem que decidir sobre o seguinte:

- "melhor" maneira de comercializar os produtos;
- quais produtos a serem produzidos economicamente;
- "melhores" técnicas de produção;
- quantidade adequada dos recursos a ser usada;
- "melhor" combinação de recursos;
- "melhor" distribuição de recursos.

Observada a cadeia produtiva do café superficialmente, será abordado como está a aplicação de informática na cafeicultura e mostrar exemplo de *software* atualmente utilizado para esta finalidade.

2.2. INFORMÁTICA NA CAFEICULTURA

Uma das coisas relevantes para o presente trabalho é identificar o quanto a informática é aplicada na cafeicultura. Nesta seção será mostrada como a informática está sendo utilizada neste setor bem como um *software* conhecido já existente com foco neste setor, o *OpenFarm*¹.

O desenvolvimento do *software* proposto foi baseado em informações obtidas através de administradores rurais da região sul do estado de Minas Gerais. Sobre esta região, Jesus *et. al* (1999) afirma em estudo realizado através de entrevistas junto a empresas/propriedades rurais da citada região que 31,1% utilizam o computador como instrumento de apoio às suas atividades técnicas, administrativas e gerenciais. O perfil básico dessas empresas é o seguinte: atuam na área de pecuária de leite e/ou cultura do café; são administradas sob base familiar ou capitalista; possuem entre 20 e 30 funcionários fixos e área de 300 hectares. Os motivos que levaram as empresas à

¹ Mais Informações: http://www.matheussantanalima.x-br.com/projetos/openfarm/new_doc.html (acessado em janeiro/2008)

informatização foram: facilidade de acesso e domínio da informação (45,4%), grande volume de informações a serem processadas (36,4%) e redução de custos (18,2%). Outras empresas (68,9%) ainda não fazem uso desta ferramenta, principalmente devido à falta de conhecimento na área e ao alto custo dos sistemas. O perfil técnico dessas empresas é muito semelhante (pecuária de leite e/ou cultura do café, administração familiar ou capitalista). Entretanto, possuem um número reduzido de funcionários fixos, área e produção.

Os principais problemas encontrados referem-se especificamente à coleta de dados (42,4%), treinamento básico em informática (30,3%) e alto custo dos sistemas (27,3%). Todos os equipamentos utilizados são compatíveis IBM-PC e encontram-se instalados nas fazendas há mais de um ano.

Um dos principais *softwares* utilizados por administradores rurais é o *OpenFarm*. Segundo Lima (2007), *O programa OpenFarm busca auxiliar no gerenciamento financeiro de propriedades rurais, por meio do controle de ganhos e gastos do produtor rural. O sistema permite ao usuário calcular, entre outros, seus gastos com insumos agrícolas, pagamento de funcionários, água e energia elétrica.*

A partir desses dados, o software elabora relatórios para que o produtor avalie os custos da produção em relação ao quanto ele vende. Assim, é possível saber qual o impacto dos gastos com água, em um determinado mês, por exemplo, no custo de produção daquele período. O programa faz, ainda, uma análise mensal ou anual de rendimento da produção.

Embora o *OpenFarm* trabalhe fortemente com a área produtiva do setor rural, não é o objetivo deste *software* a mobilidade, ficando aqui a clara diferença entre este *software* e o sistemas propostos representados pelo SOAC e SoacCel.

Mostrado a utilização de informática no setor cafeeiro e sistemas utilizados para esta finalidade, é possível dar início a explicações acerca de recursos que o *software*

proposto deverá utilizar para auxiliar a gerência da administração cafeeira. A seção a seguir abordará o conceito de banco de dados e porque um banco de dados deverá ser utilizado pelo *software* proposto.

2.3. BANCO DE DADOS

Para o desenvolvimento deste *software*, após a preocupação já demonstrada com o gerenciamento do café, é hora de começar a definir as bases com relação ao armazenamento de dados do mesmo. É de vital importância conhecimentos relacionados a bancos de dados: o que são, o que fazem, a estruturação de um banco de dados confiável e sua construção, baseada em dados realmente úteis ao utilizador do *software*. Para o tratamento de redes e Internet, é necessário um banco de dados, que possua o modelo cliente-servidor. De acordo com Ramalho (1999, pg.9), *“na arquitetura cliente/servidor, o banco de dados fica residente em um computador chamado servidor e suas informações são compartilhadas por diversos usuários que executam as aplicações em seus computadores locais, ou clientes. Essa arquitetura propicia uma maior integridade dos dados, pois todos os usuários estão trabalhando com a mesma informação”*.

Neste banco de dados, serão criadas tabelas que contenham os dados pertinentes para um acesso externo. Ramalho (1999, pg. 13) nos diz que *“As tabelas são o coração do banco de dados. Nelas, que os dados ficam armazenados. Toda tabela possui um nome próprio e é composta por linhas e colunas. As colunas possuem nomes que as identificam e são proporcionais aos campos da tabela. As linhas correspondem aos registros. (...) Uma tabela é composta por linhas, que simbolizam os registros; e por colunas, que simbolizam os campos de cada registro”*.

ElsMari (2005) descreve que os bancos de dados e os sistemas de bancos de dados se tornaram componentes essenciais no cotidiano da sociedade moderna. No decorrer do dia, a maioria de nós se depara com atividades que envolvem alguma interação com os

bancos de dados. Por exemplo, quando vamos ao banco para efetuarmos um depósito ou retirar dinheiro, se fizermos reservas em um hotel, para a compra de passagens aéreas, se acessarmos o catálogo de uma biblioteca informatizada para consultar uma bibliografia, ou se comprarmos produtos – como livros, brinquedos ou computadores – de um fornecedor por intermédio de sua página *web*, muito provavelmente essas atividades envolverão um programa de computador que acessará um banco de dados. Até mesmo os produtos adquiridos em supermercados, em muitos casos, atualmente, incluem uma atualização automática do banco de dados que mantém o controle de estoque disponível nestes estabelecimentos.

Os bancos de dados e a sua tecnologia estão provocando um grande impacto no crescimento do uso de computadores. É viável afirmar que eles representam um papel crítico em quase todas as áreas em que os computadores são utilizados, incluindo negócios, comércio eletrônico, engenharia, medicina, direito, educação e as ciências da informação, para citar apenas algumas delas.

A palavra *Banco de Dados* é tão comumente utilizada que acaba sendo necessário defini-la. Segundo uma definição bastante genérica de Elsmari (2005), “*um Banco de Dados é uma coleção de dados relacionados. Os dados são fatos que podem ser gravados e que possuem um significado implícito.*” Por exemplo, considere nomes, números telefônicos e endereços de pessoas que você conhece. Esses dados podem ter sido escritos em uma agenda de telefones ou armazenados em um computador, por meio de programas como o *BrOffice.org Calc*. Essas informações são uma coleção de dados com um significado implícito, conseqüentemente, um banco de dados.

O termo *banco de dados* geralmente é mais restrito do que uma mera coleção de dados. Possui as seguintes características implícitas:

- Um banco de dados representa alguns aspectos do mundo real, sendo chamado, às vezes, de *mini mundo* ou de *Universo de Discurso (UoD)*. As mudanças no mini mundo são refletidas no banco de dados.

- Um banco de dados é uma coleção lógica e coerente de dados com algum significado inerente. Dados desorganizados, estruturados randomicamente não podem ser interpretados como um banco de dados.
- Um banco de dados é projetado, construído e povoado por dados, atendendo a uma proposta específica. Possui um grupo de usuários definido e algumas aplicações pré-concebidas, de acordo com o interesse deste grupo de usuários.

Resumidamente, um *banco de dados* possui algumas fontes das quais os dados são derivados, alguns níveis de interação com eventos do mundo real e um público efetivamente interessado em seus conteúdos.

Visto o conceito de banco de dados, é perceptível que o *software* proposto necessitará de lidar com tais bancos de dados. No entanto, existem ferramentas que fornecem acesso e manuseio facilitado para sistemas como o proposto. Na próxima seção serão abordadas estas ferramentas, que são os *Sistemas Gerenciadores de Banco de Dados*.

2.4. SISTEMA GERENCIADOR DE BANCO DE DADOS (SGBD)

Após a descrição sobre o que é um banco de dados, já se é possível entrar em uma nova definição: Sistema Gerenciador de Banco de Dados. É muito complexo e, atualmente, desnecessário, que um programa *web* utilize seu próprio sistema de gerenciamento de bancos de dados. Isso se deve ao fato de hoje termos um enorme número de sistemas que fazem isso.

Elsuari (2005) define um sistema gerenciador de banco de dados (**SGBD**) como sendo uma coleção de programas que permitem aos usuários criar e manter um banco de dados, ou seja, *um sistema de software de propósito geral* que facilita os processos de

definição, construção, manipulação e compartilhamento de banco de dados entre vários usuários e aplicações.

A construção de um banco de dados é o processo de armazenar os dados em alguma mídia apropriada, controlada pelo SGBD. A *manipulação* inclui algumas funções, como pesquisas em um banco de dados para recuperar um dado específico, atualização do banco de dados, para refletir as mudanças no mini mundo e gerar os relatórios dos dados. O *compartilhamento* permite aos múltiplos usuários e programas acessarem, de forma concorrente, o banco de dados. Outras funções importantes do SGBD são a *proteção* e a *manutenção* do banco de dados por longos períodos.

Na seção a seguir, será abordado sobre um dos mais conhecidos SGDBs da atualidade: o *MySQL*, que foi utilizado para ser o sistema gerenciador de banco de dados do SOAC e SoacCel. Serão abordadas peculiaridades do *MySQL* e os motivos que favoreceram sua designação para tal tarefa.

2.5. MySQL

Tendo definido o que é um Banco de Dados e um Sistema Gerenciador de Banco de Dados é hora de definir qual é o mais adequado para o *software* proposto. Entre os candidatos que se adequam ao sistema, um em especial é mais atrativo: o *MySQL*.

A figura 1 permite uma visão mais detalhadas sobre como é o funcionamento real do *MySQL* com um programa de aplicações que pode efetuar consultas. Neste funcionamento, o usuário interage com o SOAC, efetuando requisição de informações desejadas. O SOAC, por sua vez, interage com um SGBD, neste caso o *MySQL*, que administra todas as informações que foram requisitadas, efetuando as consultas pedidas e acessando os dados armazenados e, após isso, retornando os dados ao SOAC para serem mostrados ao usuário.

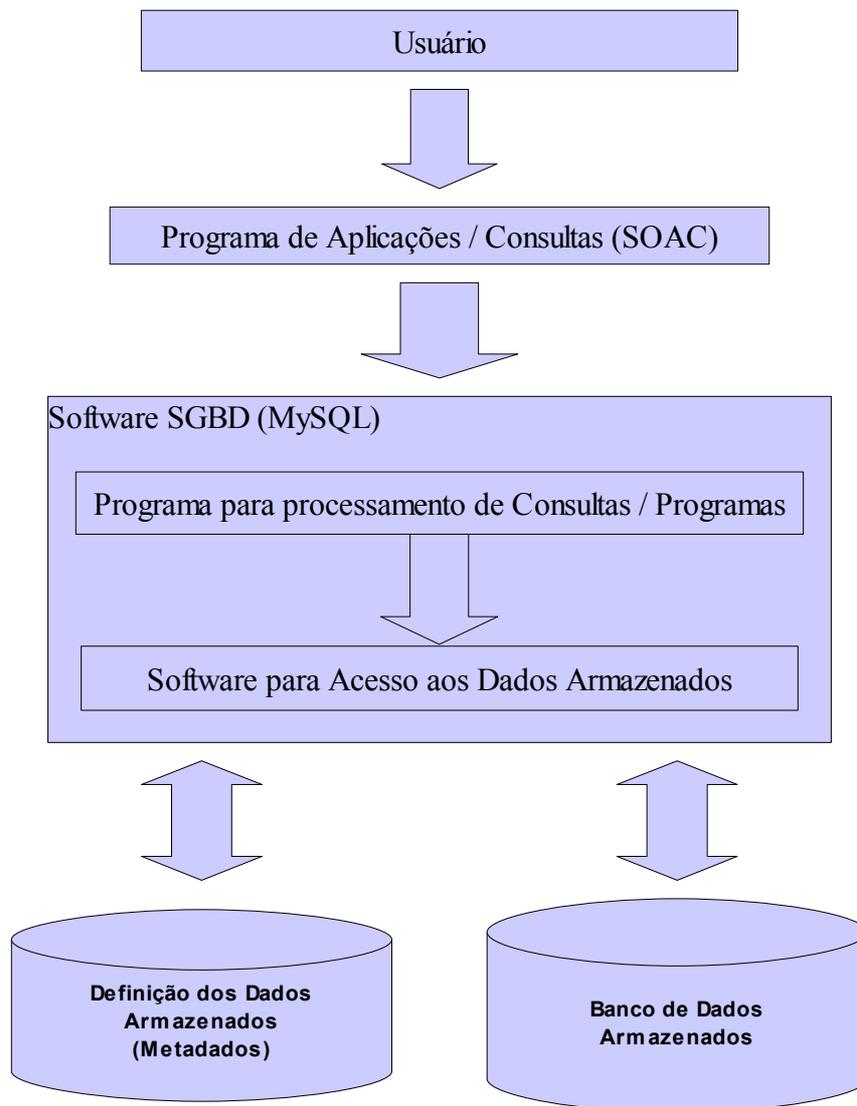


Figura 1: Interação do Sistema Gerenciador de Banco de Dados com o SOAC.

Existem várias justificativas para a escolha do *MySQL*. Bulger *et. al* (pg. 36) afirma que o *MySQL* possui bom custo benefício. O *MySQL* é livre para ser desenvolvido e pode ser utilizado em um ambiente de produção por um custo mínimo. É rápido e poderoso; pode não ser um banco de dados espetacular, mas ele é bastante completo no que diz respeito a servir conteúdo para *web* ou criação de um *site* comercial de tamanho moderado,

além de ser extremamente veloz comparado com a maioria dos outros bancos de dados existentes. Além destas características, ele encontra-se aperfeiçoando o tempo todo em uma velocidade incomum e com atualizações impressionantes. Por estas razões o *MySQL* foi escolhido para ser o sistema gerenciador de banco de dados.

Na seção a seguir será abordado um assunto de suma importância para o entendimento do funcionamento do SOAC e, especialmente, o SoacCel: a mobilidade. As noções de mobilidade, juntamente com o Sistema Gerenciador de Banco de Dados escolhido tornaram possível o desenvolvimento do *software* proposto.

2.6. MOBILIDADE

Muito foi escrito acerca de gerenciamento do café e sobre a base de dados que o sistema deverá utilizar. No entanto, como este é um sistema *web*, voltado para usuários que realmente necessitarão de acessá-lo dos mais diversos locais, é importante também definir os conceitos de mobilidade, que são empregados no presente trabalho.

As seguintes definições podem ser empregadas em mobilidade:

- Cobucci (2007) afirma que de forma mais abrangente, mobilidade é a propriedade do que é móvel ou do que obedece às leis do movimento, inconstância e dinamismo;
- Segundo Netto (2006), mobilidade é o termo utilizado para identificar dispositivos que podem ser operados à distância ou sem fio. Dispositivos que podem variar desde um simples *bip* até os mais modernos *Pockets*;
- Mobilidade, afirma MeioBit (2006), é a facilidade de acessar recursos de informática (entrada de dados, *web*, e-mail, consulta a dados) fora do ambiente principal de trabalho, podendo ou não estar em movimento. Também, deve ser considerado o uso do equipamento sem precisar de uma fonte de energia por pelo

menos quatro horas de uso e poder usá-lo em pé;

- De acordo com MobileLife (2005), mobilidade pode ser definida como a capacidade de poder se deslocar ou ser deslocado facilmente. No contexto da computação móvel, mobilidade se refere ao uso pelas pessoas de dispositivos móveis portáteis funcionalmente poderosos que ofereçam a capacidade de realizar facilmente um conjunto de funções de aplicação, sendo também capazes de conectar-se, obter dados e fornecê-los a outros usuários, aplicações e sistemas.

Como o sistema proposto será acessado através de dispositivos *handhelds* portáteis ou telefones celulares, este é um sistema que se enquadra nos padrões da mobilidade. A mobilidade se aplica a dispositivos móveis e, estes equipamentos podem ser transportados com facilidade. Na seção a seguir falaremos sobre um conceito que está fortemente atrelado a mobilidade nestes tipos de equipamentos: a portabilidade.

2.7. PORTABILIDADE

A portabilidade é definida como a capacidade que um dispositivo tem de ser transportado com certa facilidade. Hoje em dia, para ser considerado portátil, os dispositivos móveis têm que ser transportados facilmente na mão (MobileLife, 2005), mas essa definição de portabilidade pode-se alterar ao longo do tempo. Em outras palavras, algo que foi considerado portátil no passado pode não ser considerado portátil hoje. Por exemplo, os primeiros PCs portáteis, desenvolvidos na década de 80, eram tão grandes quanto pequenas valises, enquanto os primeiros telefones celulares eram do tamanho de *walkie-talkies*.

Segundo MeioBit (2006), um equipamento é dito portátil se reúne, de forma compacta, uma representação do que se possui no seu ambiente de trabalho ou em casa, mas de forma que ele possa ser facilmente movido de lugar sem necessidade de usar componentes múltiplos como monitor, teclado e mouse. O melhor exemplo é um *notebook*

de alta performance. Tem-se: tela, teclado, mouse, Internet sem fio e pelo menos 1 hora de bateria sem necessidade de um cabo de força e com recursos de processamento equivalentes a qualquer *desktop*.

Todos os conceitos apresentados até o presente momento retratam de bases nas quais o *software* proposto foi fundamentado. Na próxima seção será dado início a abordagem sobre as ferramentas que possibilitarão a construção do SOAC propriamente dito, a linguagem *XHTML*.

2.8. XHTML (eXtensible Hypertext Markup Language)

Foi apresentado como os dados serão armazenados e acessados no *software* proposto, mas apenas um banco de dados e as noções de mobilidade e portabilidade não são o bastante para um sistema *web*. Um fator que existe que não pode deixar de ser levado em conta é a forma como tal sistema será apresentado ao usuário.

Para publicação de documentos com marcação de hipertexto na *web*, a *World Wide Web Consortium* (W3C) desde o dia 26 de janeiro de 2000 recomenda a utilização do *XHTML*². *XHTML* é uma sigla para *eXtensible Hypertext Markup Language*, que é uma reformulação do *HTML* (*Hypertext Markup Language*) baseada em *XML* (*eXtensible Hypertext Markup Language*). Este padrão de edição de documentos basicamente combina as *tags* de marcação *HTML* com regras da *XML*; este processo de padronização tem em vista a exibição de páginas *web* em diversos dispositivos (televisão, *palm*, celular, etc). A intenção é melhorar a acessibilidade.

HTML é conhecido como sendo o formato padrão de hipertextos. Um hipertexto é um texto suporte que acopla outros textos em sua superfície cujo acesso se dá através dos links que têm a função de conectar a construção de sentido, estendendo ou

2 Fonte: <http://www.w3.org/TR/xhtml1/> (acessado em janeiro/2008)

complementando o texto principal. Um conceito de hipertexto precisa abranger o campo lingüístico, já que se trata de textos.

Em computação, hipertexto é um sistema para a visualização de informação cujos documentos contêm referências internas para outros documentos (chamadas de hiperlinks ou, simplesmente, links), e para a fácil publicação, atualização e pesquisa de informação. O sistema de hipertexto mais conhecido atualmente é a *World Wide Web*, no entanto a Internet não é o único suporte onde este modelo de organização da informação e produção textual se manifesta.

Em outras palavras, *XHTML* define como e de que forma desejamos que um determinado texto deverá aparecer na tela do usuário que está acessando o sistema, informando a formatação do mesmo, as imagens e tudo aquilo que quem acessa o sistema poderá ver.

O *XHTML* consegue ser interpretado por qualquer dispositivo, independentemente da plataforma utilizada, pois as marcações possuem sentido semântico para as máquinas. O *HTML* não consegue esta implementação. No entanto, não existem muitas diferenças entre o *HTML* e o *XHTML*. Para verificar se uma página *XHTML* está bem construída, o melhor método é validar o código através de uma aplicação *web* disponibilizada pela W3C.

Como se pôde perceber neste capítulo o *XHTML* permite a elaboração de textos para serem exibidos através de navegadores que utilizam a Internet. No entanto, essa simples elaboração de textos não permitem que a máquina do usuário que acessa o sistema tenha condições de acessar e obter dados diretamente do banco de dados ou manipular as páginas geradas conforme as necessidades do cliente e dados de entrada. Estes parâmetros necessitam ser alterados dinamicamente. Na seção seguinte será abordado a ferramenta que permite o SOAC fazer estas operações que o *XHTML* convencional não permite: o *PHP*.

2.9. PHP (PHP Hypertext Preprocessor)

Para tornar possível o acesso aos dados do banco de dados, através da Internet, é necessária a estruturação de uma interação do usuário com o sistema gerado por esse projeto. Para Corrêa (2006), interação é toda *comunicação mediada por computador*. Tal interação pode ser elaborada através de um *site* no servidor, que hospeda o banco de dados. Para criar tal *site*, carecemos de ferramentas e suporte para linguagens de programação robustas que possibilitem o acesso dinâmico ao banco de dados. Para isso, utilizaremos o *PHP*, uma linguagem voltada para a Internet, cuja sigla é um acrônimo recursivo que significa *PHP Hypertext Preprocessor* (Pré-processador de Hipertexto *PHP*).

Uma forma de definir linguagem de programação é como o elo de comunicação entre o programador e a máquina. No mundo, existem vários idiomas (inglês, espanhol, etc.), assim como no universo da computação há várias linguagens (C++, *Java*, Pascal, *PHP*). O programador equivale ao intérprete; apenas ele poderá compreender, manusear e decifrar o código. Portanto, a linguagem de programação é a ferramenta de trabalho do programador, que este, provavelmente, especializará numa determinada linguagem. A linguagem selecionada foi *PHP*.

Segundo Serrão *et. al* (2001, pg. 564) descreve que *PHP* é uma poderosa ferramenta que surgiu da necessidade de os programadores *web* utilizarem recursos capazes de permitir a inter-atividade com os utilizadores e gerarem conteúdo dinâmico.

Esta ferramenta, além de funcionar praticamente em todos os ambientes conhecidos, tem um custo de aquisição nulo. Aliada as funcionalidades proporcionadas pela interação com bases de dados, permite a concretização de uma plataforma robusta, capaz de suportar o desenvolvimento de aplicações orientadas à criar páginas dinâmicas.

O conteúdo dinâmico ao qual Serrão se refere quer dizer que para cada usuário que acessa o sistema pode estar vendo uma página que é gerada no momento de tal acesso. Isso

é importante para o *software* proposto, pois o mesmo necessita de obter recursos que são modificados o tempo todo em um banco de dados e por isso precisa modificar a forma como as informações são retornadas ao usuário.

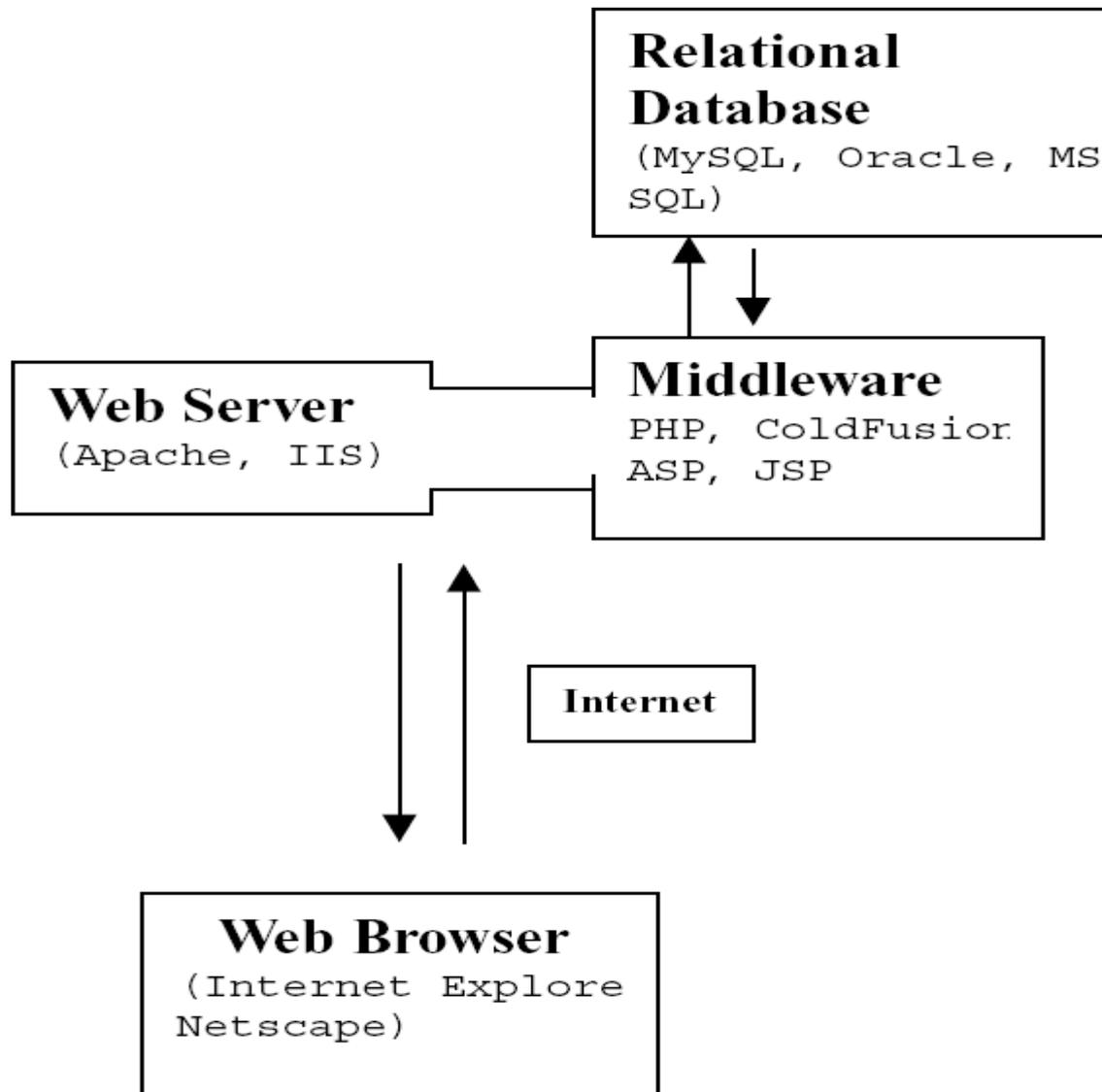


Figura 2: Interação entre PHP (Middleware), MySQL (SGBD), Apache (servidor de HTTP [ex: páginas XHTML]) e o Navegador do usuário do sistema.

O funcionamento de um servidor de PHP possui estrutura semelhante a apresentada na figura 2. Quando um navegador acessa um *site* através da Internet, os dados são provenientes de um servidor de páginas web. O PHP atua como sendo um *Middleware*

capaz de interagir entre o servidor *web* e um banco de dados, ou seja, um módulo que estende este servidor. Ele pode pré-processar as informações antes de serem enviadas ao navegador do usuário do sistema, bem como obter informações em um banco de dados neste processo.

Como características marcantes que influenciaram a escolha de *PHP* além das anteriormente citadas se destacam a velocidade e robustez, estruturação, portabilidade e a fraca tipagem providas por esta linguagem.

Por fim, Bulger *et. al* (2004, pg. 29) afirma não pensar existir uma melhor combinação de ferramentas do que *PHP* e *MySQL*. Os números falam por si próprios. A combinação *PHP* e *MySQL* são extremamente populares e a demanda de pessoas para utilização de tais ferramentas apenas aumenta.

Observado o funcionamento de *XHTML*, *PHP* e *MySQL* já se é possível desenvolver o Sistema Online para Auxílio ao Cafeicultor com Foco na Mobilidade para ser acessado via *browsers* de computadores pessoais. No entanto, é objetivo do sistema proposto o foco na mobilidade. E a mobilidade no caso do SOAC está atrelada a portabilidade. É objetivo deste sistema também ser acessível por dispositivos *móveis* e *portáteis*, como por exemplo *handhelds*. No entanto apenas utilizando as tecnologias apresentadas até o presente momento não se é possível criar um acesso otimizado ao sistema para estes dispositivos. Na seção seguinte será abordada a tecnologia que permite o acesso otimizado de dispositivos como *handhelds* ao SOAC, chamada de *CSS*.

2.10. CSS (*Cascading Style Sheets*)

Já foi definido e mostrado os recursos que o sistema utilizará para efetuar um acesso ao banco de dados e como serão mostrados as estruturas de formatação do mesmo com o usuário. No entanto, essa estruturação da interface e acesso ao banco de dados

apenas não são o bastante.

Um dos objetivos do SOAC é permitir que dispositivos portáteis do tipo *handheld* possam acessar o sistema *web*. Embora tais dispositivos possuam total acesso a Internet com um navegador semelhante ao utilizado em máquinas *desktop*, algumas atitudes com tais dispositivos devem ser tomadas devido à suas limitações.

Uma destas limitações é a largura de banda reduzida, ou seja, pouca quantidade de fluxo de dados entre o dispositivo e a Internet. Outra preocupação é o fato de ser muito desconfortável navegar em *sites* que exigem resoluções elevadas de tela, e as resoluções de tela destes dispositivos geralmente são bastante limitadas.

Várias alternativas possíveis podem ser tomadas para combater as limitações de dispositivos *handhelds* e os problemas citados. Uma delas é a elaboração de dois sistemas: um que seja utilizado apenas para *handhelds* e outro para computadores *desktops*, mas esta opção tem como problema despender um grande esforço para elaborar 2 sistemas. Uma outra opção seria a elaboração de um sistema universal, para resoluções de tela pequenas e com poucas imagens, tendo apenas o essencial; mas neste caso também teríamos o problema de o sistema não ficar amigável aos olhos do usuário. Uma terceira alternativa e provavelmente a mais interessante é a utilização de *CSS*.

CSS (Cascading Style Sheets) é uma linguagem de estilo usada para definir a apresentação de documentos escritos em uma linguagem de marcação, como no nosso caso o *XHTML*. Seu principal benefício é prover a separação entre o formato e o conteúdo de um documento.

Ao invés de colocar a formatação dentro do documento, o desenvolvedor cria um *link* (ligação) para uma página que contém os estilos, procedendo de forma idêntica para todas as páginas de um portal. Quando quiser alterar a aparência do sistema *web* desenvolvido basta portanto modificar apenas um arquivo.

Mais do que isso, *CSS* permitem definir uma diferente exibição dos estilos em uma página para diferentes tipos de mídia. Em outras palavras, a partir de uma mesma página passa a ser possível a visualização de várias formas diferentes em um *desktop* de um *handheld* e, ainda, que quando esta for impressa tenha outra formatação. Tudo isso em apenas uma página.

No caso do presente sistema, o *CSS* influenciará que uma mesma página seja visível de formas diferentes em computadores *desktop* de dispositivos móveis como *handhelds*.

Com todas as tecnologias demonstradas até aqui já torna possível a elaboração completa do SOAC e que este seja acessado via computadores pessoais e *handhelds*. No entanto, foi uma decisão de implementação torná-lo ainda mais portátil. Em outras palavras, foi decidido que o mesmo deveria ser portátil também para telefones celulares. Uma das decisões de implementação foi que os dados do SOAC para serem acessíveis a partir de tal dispositivo deveriam ter um *software* que fosse capaz de ser executado utilizando a tecnologia *Java 2 Micro Edition* nestes aparelhos, utilizando como linguagem de programação a linguagem *Java*. Nas seções seguintes será mostrado porque esta linguagem e tecnologias foram escolhidas.

2.11. JAVA

Conhecimentos básicos necessários para se ter uma idéia dos recursos que o sistema utilizará para ser acessado através de computadores *desktops* e de *handhelds já foram definidos*. No entanto, é um dos objetivos do trabalho que o sistema elaborado também seja acessível através de dispositivos móveis, como telefones celulares.

Ainda não existem telefones celulares a preços acessíveis que possam acessar a Internet com navegadores com boa capacidade de processamento, além de terem uma baixíssima resolução de tela comparados a *handhelds*. Em outras palavras, não existe uma

alternativa senão criar um *software* para funcionar em paralelo com o sistema SOAC para ter acesso pelo menos às ferramentas de consulta do sistema em questão.

Um novo *software* implica em uma nova linguagem de programação voltada para o dispositivo ao qual a programação será efetuada. No caso de dispositivos móveis, a empresa *Sun Microsystems*³ disponibiliza uma plataforma bastante eficiente, dentro das limitações de tais dispositivos, para *softwares* em geral, a *Java 2 Micro Edition (J2ME)*⁴, a qual será descrita adiante.

A plataforma *J2ME* roda com uma linguagem de programação denominada *Java*, a qual é o foco desta seção. Com relação à escolha da linguagem de programação para o *software*, deve-se levar em consideração o que dizem Coad e Mayfield (1998, pg. 15):

Os materiais de construção afetam profundamente as técnicas de projeto.

Os materiais de construção de uma casa afetam o projeto da casa. Os tecidos afetam o projeto das roupas. E, claro, os materiais de construção de programação (linguagens) afetam o projeto de software.

Em todas as áreas do esforço humano, novos materiais de construção são conseguidos por novos métodos de projeto.

Portanto, é necessário uma linguagem que gere um *software*, que seja capaz de ser executado em diferentes computadores pessoais, independente do sistema operacional que os mesmos utilizem. Com base nos fatos apresentados, foi realizada análise da linguagem de programação *Java*.

Oliveira (2002, pg. 6) afirma que *Java* é uma linguagem de programação desenvolvida pela Sun Microsystems e lançada em 1995, visando o mercado de bens

3 Fonte: <http://www.sun.com/> (acessado em janeiro/2008)

4 Fonte: <http://java.sun.com/javame/index.jsp> (acessado em janeiro/2008)

eletrônicos de consumo. Um objetivo estabelecido desde sua concepção foi o de ser uma linguagem segura, tanto no sentido de evitar algumas falhas comuns, que os programadores costumam cometer durante o desenvolvimento, como no sentido de evitar ataques externos. Isto é importante no mercado de bens eletrônicos de consumo por que ninguém gostaria de adquirir um produto que necessitasse desligar e religar para que voltasse a funcionar corretamente.

Estas propriedades despertaram o interesse para utilização de *Java* em outro ambiente que também necessitava de uma linguagem com este perfil: a Internet. A Internet também é um ambiente constituído por equipamentos de diferentes arquiteturas e necessita muito de uma linguagem que permita a construção de aplicativos seguros.

Pelo sistema proposto, é necessário o acesso à Internet. Deve-se considerar a dificuldade para implementação da referida funcionalidade. Oliveira (2002, pg. 9) informa a respeito disso com relação à linguagem de programação em estudo para este *software* que “*Java possui em seu núcleo básico classes para comunicação em rede por meio dos protocolos pertencentes à pilha de protocolos TCP/IP. A pilha de protocolos TCP/IP é a utilizada pela Internet e tornou-se o padrão de fato para comunicação entre computadores em uma rede heterogênea. Isto torna Java particularmente atrativa para o desenvolvimento de aplicações na Internet*”. Assim sendo, a linguagem de programação a ser utilizada no desenvolvimento desse *software*, será *Java*.

Falou-se até então da linguagem de programação que será utilizada, *Java*. Mas esta linguagem, por si própria não é capaz de ser compilada e executada em um dispositivo móvel como um telefone celular sem uma plataforma. Na seção a seguir será abordada a plataforma que suporta programas escritos na linguagem de programação *Java* para sua execução: a *Java 2 Micro Edition (J2ME)*.

2.12. J2ME (Java 2 Micro Edition)

Conforme demonstrado em capítulos anteriores, a linguagem utilizada para o desenvolvimento do SOAC para telefones celulares, denominado SoacCel será a *Java*. Neste capítulo será descrita a plataforma em cima da qual o *software* escrito em *Java* será executado; a plataforma *J2ME*.

Lima *et. al* (2007) diz que a plataforma *Java* vem se desenvolvendo continuamente, oferecendo suporte a uma crescente variedade de tipos de aplicativos. Como consequência, houve um grande aumento do seu tamanho, o que passou a conflitar com o objetivo do *WORA* (*write once, run anywhere*), uma vez que nem todo dispositivo oferecia suporte a processos que demandavam uma grande quantidade de memória. Como uma primeira tentativa de contornar o problema, a Sun introduziu, em 1997, uma variedade de plataformas *Java*, cada uma voltada a um segmento do mercado, sendo que a principal diferença entre elas era o subconjunto da *API* do *Java* a que ofereciam suporte.

Conforme a tecnologia foi se desenvolvendo, sentiu-se a necessidade de se unificar as plataformas orientadas a dispositivo e em 1999 surgiu o *Java 2 Micro Edition*, que é uma arquitetura que permite o uso da tecnologia *Java* em dispositivos com diversas características e funcionalidades. A plataforma *J2ME* é a edição da linguagem *Java* que foi projetada para dispositivos com memória, vídeo e poder de processamento limitados, como por exemplo telefones celulares.

A arquitetura da plataforma *J2ME* permite ao desenvolvedor conhecer informações específicas sobre as diferentes famílias de dispositivos e as *Application Program Interfaces* (*APIs*) disponíveis para cada uma delas. A máquina virtual *J2ME* foi projetada para ser menor e mais eficiente do que a máquina virtual *J2SE*, sendo um subconjunto desta. Em função disso, ocupa aproximadamente 60 kilobytes de memória em tempo de execução. A máquina virtual *J2ME* é chamada *KVM* numa alusão à pequena quantidade de memória exigida, que está na ordem dos kilobytes. É a base da arquitetura, localizada logo

acima do sistema operacional hospedeiro, seguida pelas configurações que dão suporte aos Perfis.

Segundo Lima *et. al* (2007), “*J2ME* é dividido em *configurations*, *profiles* e *API's* opcionais. Para entendermos melhor, pensemos da seguinte maneira, *Profiles* são mais específicos que *configurations*, e fazendo analogia a um velho exemplo temos uma abstração sobre o que é um carro e como ele é fabricado (*configuration*) e como um Ford é fabricado (*profile*), mais tecnicamente falando *profile* é baseado em *configuration* e ainda acima dos *profiles* estão as *API's* que na nossa analogia seria um modelo específico da Ford. Existem dois "*configurations*", um *configuration* é o *CLDC* (*Connected Limited Device Configuration*), que rege as configurações para aparelhos bem pequenos como celulares ou *PDA's*, o qual fica acima das diretrizes *J2ME* juntamente com *CDC* (*Connected Device Configuration*) o que rege as configurações para aparelhos um pouco maiores, mas mesmo assim pequenos”.

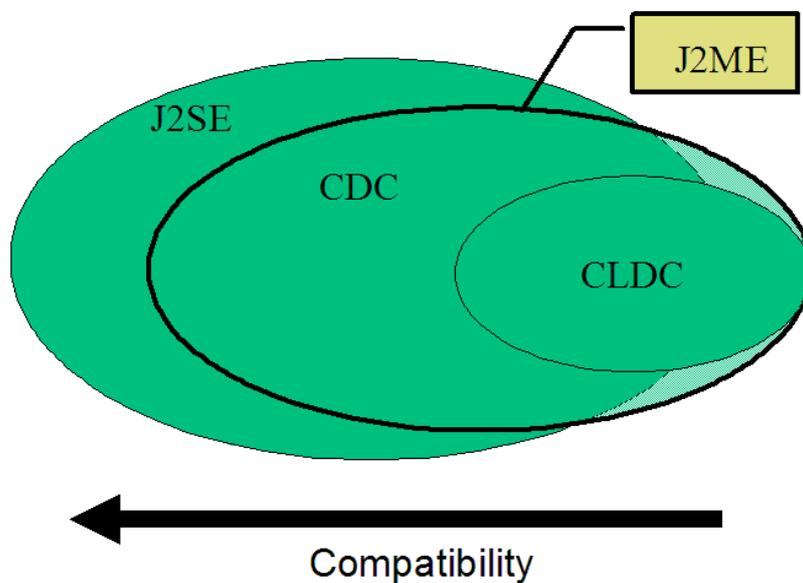


Figura 3: Demonstração Hierárquica do J2ME com o J2SE⁵.

A figura 3 é uma ilustração hierárquica representa adequadamente como a plataforma *J2ME* se comporta diante das plataformas *Java 2 Standard Edition*, bem como

5 Fonte: <http://www.devx.com/wireless/Article/21871/1954?pf=true> (acessado em janeiro/2008)

os recursos do *CLDC* se restringe ao *CDC* convencional

Portanto, para o desenvolvimento do SoacCel será utilizada a plataforma *J2ME*, com a configuração *CLDC* e o *software* desenvolvido será executado por um *KVM* em qualquer dispositivo de telefone celular com suporte a *Java*.

Terminado este referencial teórico, agora será mostrada a metodologia utilizada para elaboração do presente trabalho, detalhando o tipo de pesquisa e os procedimentos metodológicos nela aplicados.

3. METODOLOGIA

Neste capítulo será apresentado o tipo da pesquisa envolvida no projeto, bem como como foi efetuada a construção destes dois sistemas; SOAC (estático) e SoacCel (móvel).

3.1. TIPO DE PESQUISA

Para realização do presente trabalho, foram pesquisadas as necessidades dos administradores rurais no setor cafeeiro as quais precisavam ser atendidas por um *software* que focasse o setor produtivo e a mobilidade, tais como: compra de insumos, forma de como o pagamento de empregados é efetuado, gastos, dados da colheita, despesas extras, receitas e de como é obtido o retorno financeiro. O procedimento mencionado, que visou a coleta de dados, foi cumprido por meio de pesquisas informais realizadas com cafeicultores, realizadas em agosto de 2007. Foram entrevistados quatro administradores rurais das cidades de Campestre e Varginha, cidades estas localizadas na região sul do estado de Minas Gerais. Claro (1998) afirma que *a natureza qualitativa mede as variáveis que expressam atributos e qualidade, medindo a presença ou a ausência de algo, ou mesmo medir o grau em que algo está presente*. Portanto, pode-se afirmar que a natureza da pesquisa do presente trabalho é qualitativa.

Trata-se de pesquisa de caráter tecnológico, tendo-se em vista de que utiliza conhecimentos básicos e tecnologias existentes para criação do sistema proposto.

3.2. PROCEDIMENTOS METODOLÓGICOS

Esta seção irá abordar os procedimentos metodológicos utilizados para a realização do presente projeto, composto pela pesquisa, modelagem e desenvolvimento de *softwares* que foram os objetivos do presente trabalho. A seguir será apresentada uma noção geral

dos procedimentos metodológicos adotados e após isso estes serão detalhados em subseções, explicando detalhados em seções dedicadas.

Inicialmente houve a idealização e fundamentação dos princípios que este projeto deveria seguir. Com a definição destas bases, o primeiro procedimento metodológico efetuado foi a entrevista informal com cafeicultores do sul de Minas e para adquirir uma base nos dados identificar as necessidades destes possíveis futuros usuários do sistema.

Após esta fase foi feita uma modelagem inicial do projeto, na qual foram criadas diretrizes básicas para o desenvolvimento do mesmo. Posteriormente à parte de modelagem, foi executado o desenvolvimento do sistema e finalmente a análise dos resultados obtidos com o sistema.

3.2.1. LEVANTAMENTO E ANÁLISE DE REQUISITOS

As primeiras tarefas para possibilitar a construção dos *softwares* propostos foram o levantamento e a análise de requisitos do projeto. Esta tarefa foi subdividida em: levantamento de requisitos, análise de projetos com alguma semelhança já existentes, escrita do projeto, avaliação da implantação do projeto em *handhelds*, avaliação da implantação do projetos em telefones celulares, escolha da plataforma para o sistema e definição dos objetivos do sistema.

Inicialmente foram levantados requisitos com potenciais usuários do sistema através de pesquisas informais com os mesmos. Também foram criados esboços de qual seria o sistema, como o mesmo deveria funcionar e a viabilidade inicial do mesmo. Tendo cumprido todas estas exigências, foi dado início à fase seguinte, ou seja, a análise inicial do projeto.

Foi verificada a existência de sistemas que realizassem funções semelhantes ao

sistema proposto por este projeto. Tal verificação se fez necessária para evitar a construção de um sistema repetido ou até mesmo inútil e, também, para verificar como projetos e sistemas criados anteriormente poderiam de alguma forma contribuir para o desenvolvimento deste. Como não foram encontrados projetos com grau de semelhança que tornasse este projeto repetitivo ou o tornasse inútil, o desenvolvimento do projeto prosseguiu.

Após a fase de análise de projetos semelhantes, foi iniciada uma escrita inicial deste projeto, sobre como este seria desenvolvido, uma introdução e referencial teórico base sobre o mesmo, a idealização de uma metodologia e a elaboração de um cronograma estimado para ser seguido. Tal escrita permitiu definir os passos a serem seguidos para o desenvolvimento do mesmo e como seria uma idéia mais detalhada do funcionamento do sistema. Também permitiu verificar quais requisitos poderiam ser satisfeitos e aqueles que devido à complexidade ou limitação de tempo não poderiam.

Depois da escrita do projeto, e com uma definição mais profunda dos passos a serem seguidos ficou definido que o SOAC – Sistema Online de Auxílio ao Cafeicultor com foco na Mobilidade seria um sistema *web*. O servidor deverá pré-processar os dados antes de enviá-los ao usuário e também de acessar um banco de dados, para armazenamento dos dados. Pôde-se perceber durante o desenvolvimento do projeto que, embora o sistema já estivesse bem-estruturado com relação aos requisitos da plataforma, tinha algo que faltava no mesmo: foco na mobilidade. O foco na mobilidade se fez necessário pela natureza do usuário do sistema. É interessante que o usuário deseja acessar o sistema através de dispositivos portáteis em qualquer localização, pois administradores rurais estão em constante movimentação. Com essa preocupação análises extras de viabilidades de implementação e funcionamento do sistema para dispositivos portáteis, como *handhelds* e telefones celulares tornou-se necessária.

Na seqüência, foi verificada a possibilidade de tornar o sistema compatível e acessível através de *handhelds*, e as necessidades de modificações no sistema inicial para tornar isso possível. Também foram realizadas análises objetivando o funcionamento do

SOAC em telefones celulares. Nessas análises foi verificado que permitir o acesso para *handhelds* demandaria um esforço extra, mas não muito grande, para adaptar o sistema *web* SOAC para ser visualizado perfeitamente neste tipo de dispositivo. No entanto, notou ser praticamente impossível que um telefone celular comum acesse tal sistema diretamente via *web* através de um *browser* e necessitaria de uma plataforma própria para realização de tal tarefa.

Tendo preocupações com a plataforma, uma fina análise das opções existentes foi realizada com a finalidade de definir quais seriam as plataformas que suportariam o sistema para acesso através de máquinas *desktop*, *notebooks*, *handhelds* e telefones celulares; além de como seriam enviados os dados para tais clientes e qual seria o servidor do banco de dados utilizado pelo sistema. Como os mesmos programas são executados em uma máquina *desktop* e rodam igualmente em um *notebook*, foi possível verificar que a forma menos desgastante para um sistema *web* que atendesse essas duas classes de máquinas seria a utilização de um *software*, capaz de enviar hipertextos aos clientes. O *software* que mais se adequa à estas necessidades é o *Apache*. Após isso foi verificado que cada hipertexto que fosse enviado aos clientes deveria ser alterado dinamicamente, pois os dados se alterariam de igual forma. Dessa forma, foi necessário um pré-processador de hipertexto que, para esta função foi escolhido o *PHP*. Além disso seria necessário efetuar consultas a bancos de dados, e o banco de dados que notavelmente se adaptava para esta função, além de possuir considerável afinidade com *Apache* e o *PHP* foi o *MySQL*. Desta forma, a plataforma do sistema estava devidamente definida para *desktops* e *notebooks*.

Depois da análise da viabilidade de implantação do sistema SOAC para *handhelds*, foi possível constatar que o mesmo poderia acessar à Internet através de dispositivos como *GPRS* ou *EDGE*⁶, além de existirem *browsers* para este tipo de dispositivo. Foi constatado que a utilização de folhas de estilos – *CSS* – cuidadosamente desenvolvidas para estes dispositivos tornaria o sistema acessível e com uma navegação otimizada para os mesmos. No entanto, houve uma dificuldade encontrada no que diz respeito a utilização de telefones

6 Mais Informações: <http://eletronicos.hsw.uol.com.br/telefonica-3g1.htm> (acessado em janeiro/2008)

celulares para acesso ao sistema. Atualmente, poucos telefones celulares dispõem de uma tela de alta resolução e de um *browser* capaz de navegar satisfatoriamente em páginas *web*, e o acesso destes ao SOAC na realidade tornaria o sistema pouco amigável senão demasiadamente complexo. Como o objetivo do sistema foi focar a mobilidade e portabilidade tal tipo de dispositivo não poderia ser deixado de fora. Então surgiu aí a necessidade de utilizar uma plataforma devidamente desenvolvida para que celulares possam utilizar – a *J2ME*. Deste modo, um *software* específico seria construído em cima desta plataforma que viabilizasse a conexão de dispositivos celulares com os dados armazenados pelo sistema proposto. Tal *software* foi batizado como SoacCel (Sistema Online de Auxílio ao Cafeicultor para Celulares).

Após a realização de todas as análises anteriormente citadas, juntamente com a escolha de plataformas e cuidadosamente ter escolhido as bases em cima das quais o sistema iria trabalhar, foi realizada uma revisão da escrita no projeto e a criação definitiva e detalhada de quais seriam os objetivos do projeto.

3.2.2. MODELAGEM DO SISTEMA

Com base nas informações coletadas durante a idealização inicial do projeto, filtrados pelos objetivos do projeto e as devidas análises realizadas na fase anterior, deu-se início à modelagem do sistema.

Os dados obtidos através das pesquisas informais realizadas com os administradores rurais, foi possível elaborar um diagrama de casos de uso que seria utilizado como fundamento durante o desenvolvimento de todo o sistema, conforme a figura 4.

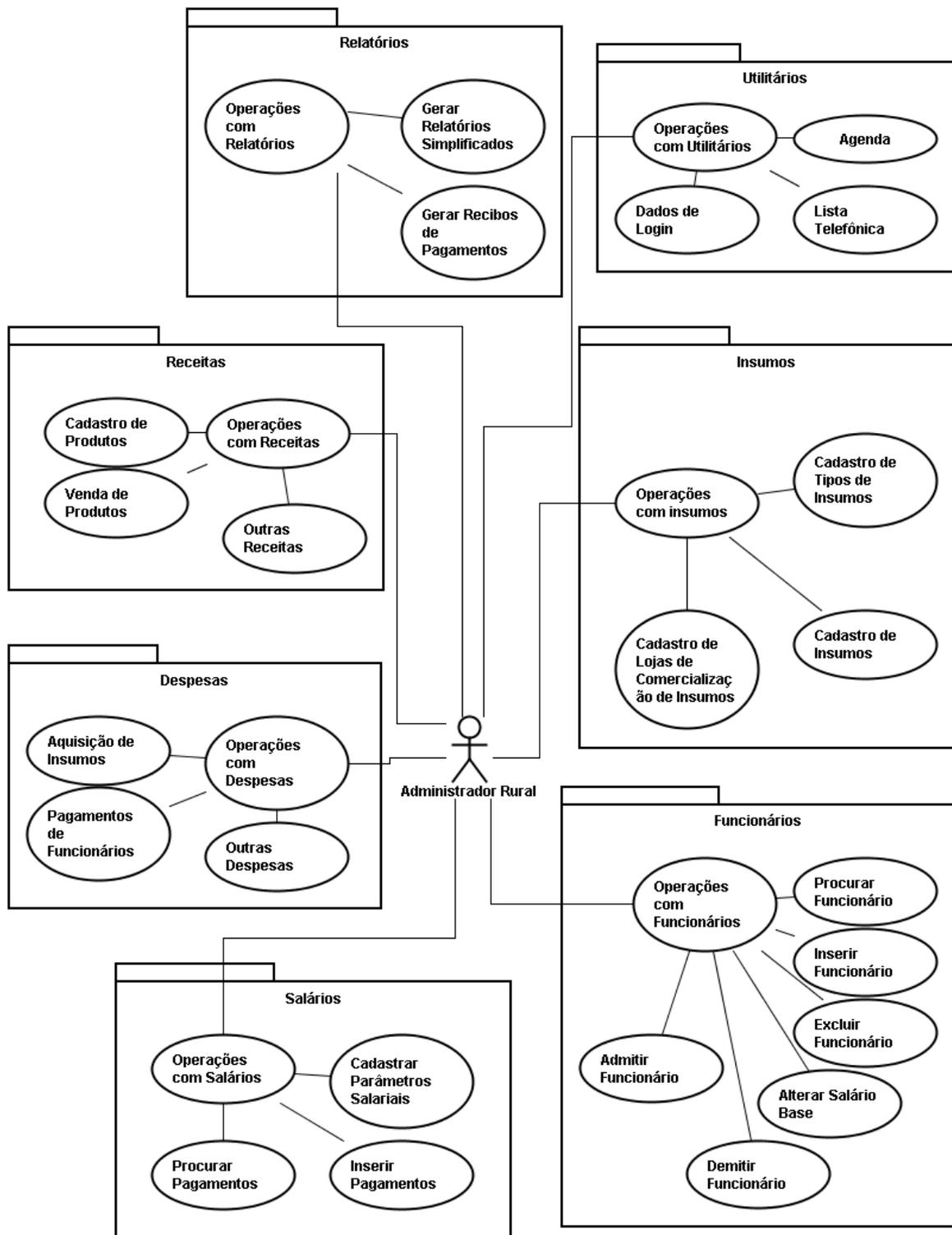


Figura 4: Diagrama de Casos de Uso do SOAC.

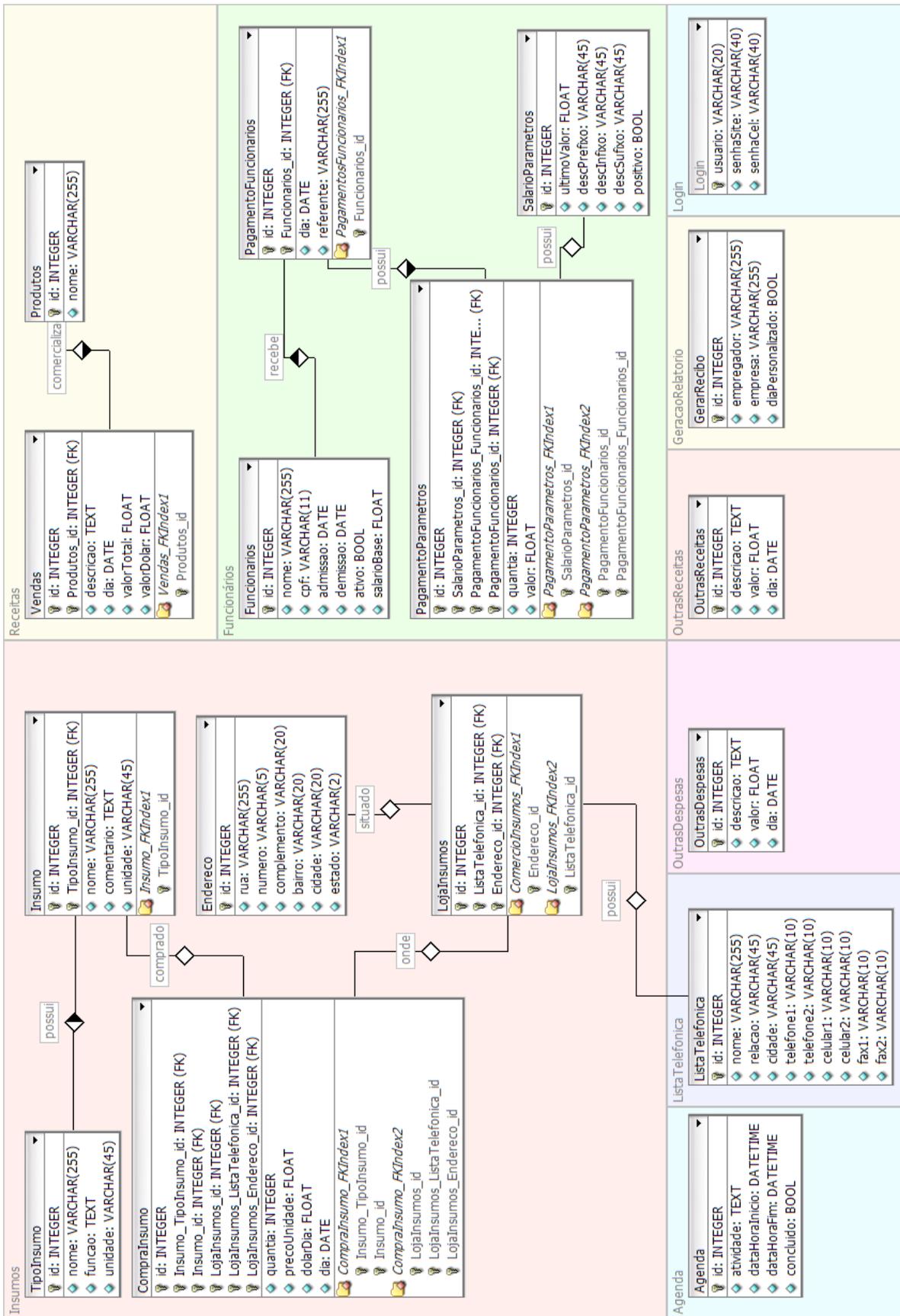


Figura 5: Modelo do Banco de Dados utilizado pelo SOAC e SoacCel.

É possível ter uma visão global das necessidades a serem implementadas no SOAC por meio de uma modelagem de casos de uso *UML* demonstradas no diagrama representado pela figura 4. A *UML (Unified Modeling Language)* é uma linguagem para especificação, documentação, visualização e desenvolvimento de sistemas, a qual por meio de seus diagramas é possível representar sistemas de *softwares* sob diversas perspectivas. O diagrama utilizado neste projeto representa as necessidades de um administrador rural nas quais o sistema proposto teve sua construção embasada. No diagrama mostrado temos o ator do sistema representado pelo administrador rural, e suas principais necessidades divididas em grupos funcionais denominados *Insumos*, *Funcionários*, *Salários*, *Despesas*, *Receitas*, *Relatórios* e *Utilitários* e cada um destes grupos com as necessidades detalhadas das operações desejadas pelo usuário que devem ser implementadas. Dada a complexidade e extensão de cada um dos casos, outros diagramas não serão mostrados neste projeto, porém o resultado deles, que é o objetivo do presente projeto, mostram as funcionalidades do Sistema Online de Auxílio ao cafeicultor. Este diagrama também passou por uma filtragem para tornar possível a realização deste em prazos plausíveis.

Para que o sistema pudesse ser construído o foco inicial deveria ser em quais dados são pertinentes ao mesmo, quais carecem de ser armazenados e administrados pelo sistema gerenciador de banco de dados. Tendo em vista isso, foi modelado o banco de dados, conforme mostrado na figura 5. Neste modelo é possível perceber o relacionamento entre as tabelas das funcionalidades do sistema, divididas por regiões. Para exemplificar estas relações, podemos entender que na região *Funcionários* um funcionário da tabela *Funcionarios* cadastrado no sistema recebe um pagamento da tabela *PagamentoFuncionarios* que, por sua vez, possui os parâmetros cadastrados em outra tabela denominada *TabelaParametros* que possui parâmetros salariais, armazenados na tabela *SalarioParametros*. Na região *Receitas* estão armazenada as comercializações de produtos devidamente cadastrados na tabela *Produtos* e, o resultado dessas comercializações, no caso as vendas, estão cadastradas na tabela *Vendas*. Este tipo de comportamento se repete para todas as partes do sistema, conforme visto na citada figura.

Uma vez devidamente realizada a modelagem dos dados necessários ao sistema e a forma como os mesmos seriam armazenados, foi possível então iniciar a modelagem do *software* SOAC, em princípio desenvolvido para acesso através de computadores pessoais e logo após isso as modificações no mesmo para ser acessado através de dispositivos *handhelds*.

Em paralelo com a modelagem do SOAC foi possível a realização da modelagem do SoacCel, uma vez que as bases que ambos necessitavam era apenas a modelagem do banco de dados. A modelagem do SoacCel serviu para definir como o sistema seria acessado através de telefones celulares e como seria a interface deste com o usuário final.

Tendo realizadas todas as modelagens, um processo de finalização e revisão para encontrar possíveis falhas foi realizado em ambas.

3.2.3. DESENVOLVIMENTO DO SISTEMA

Uma vez feita toda a modelagem de como o sistema iria funcionar, foi dado início ao desenvolvimento do mesmo. A arquitetura escolhida foi uma arquitetura em quatro camadas, tanto para o SOAC quanto o SoacCel. No SOAC a camada de apresentação é o próprio navegador do usuário, que interage com a camada *web* representada pelas informações que o *Apache* irá processar, deste modo obtendo informações processadas da camada de aplicação representada pelo *PHP* que por sua vez coleta dados da camada de dados, neste caso armazenados no *MySQL*. No SoacCel há mudanças apenas na camada de apresentação, que neste caso é representada pelo próprio SoacCel. O desenvolvimento dos sistemas foi dividido em três fases: o desenvolvimento do banco de dados e estruturas de acesso, o desenvolvimento para computadores pessoais e *handhelds* e o desenvolvimento para telefones celulares. Adiante está o detalhamento de cada uma destas grandes fases, mostrando de forma um pouco mais detalhada o que foi feito em cada uma destas.

Inicialmente será mostrado o que foi realizado durante o desenvolvimento de banco de dados e estruturas de acesso em detalhes. Mais a frente, no capítulo referente ao SOAC propriamente dito será informado detalhadamente o que é cada funcionalidade do sistema desenvolvido, pois não é o objetivo deste: nesta seção será abordado apenas como cada funcionalidade foi realizada em nível estrutural.

Com a modelagem do banco de dados devidamente feita, passou à parte da implementação do que foi modelado. Assim como na modelagem, todo o sistema depende que o banco de dados seja previamente feito para otimizar este processo. Conforme dados que foram coletados na parte de idealização e análise inicial do projeto, e modelados no processo de modelagem do banco de dados, foram criados as *SQLs* seguido das *Inserções Iniciais* dos dados que entrarão no banco de dados. *SQL* é uma linguagem de programação que permite que o programador interaja com o banco de dados informando ao mesmo que tipos de dados ele armazenará. As *inserções iniciais* são as primeiras inserções que uma tabela presente no banco de dados recebe assim que é criada; em alguns casos é necessário efetuar tais inserções. O desenvolvimento destes *SQLs* utilizados para geração das tabelas iniciais que será utilizado pelo sistema foram feitos todos em paralelo.

Uma vez tendo feita as *SQLs* e gerado as tabelas iniciais do banco de dados, devidamente inicializadas, foi necessário desenvolver estruturas de acesso à esse banco de dados para que o SOAC possa funcionar adequadamente. A primeira estrutura realizada é uma estrutura de abertura e fechamento de conexão com o banco de dados, a qual permite com que um sistema desenvolvido em *PHP* através de um *driver* previamente instalado no *Apache* permita estabelecer a conexão entre este e o sistema gerenciador de banco de dados (*MySQL*). Após o desenvolvimento destas estruturas iniciais, foi necessário desenvolver estruturas intermediárias em *PHP* de interação entre o SOAC e o sistema gerenciador de banco de dados, a saber: constantes, que permitem o sistema conhecer o endereço do banco de dados ao qual ele irá conectar e quais são os dados fundamentais para este estabelecimento de conexão; funções básicas que permitem o tratamento, envio e recebimento de pedidos de qualquer comando *SQL* ao sistema gerenciador de banco de dados, como consultas, inserções, alterações e exclusões de dados em tabelas; bem como

as estruturas necessárias para os sistemas do SOAC como *login*, estruturas de cadastramento de insumos, funcionários, pagamentos e também as estruturas que o SoacCel utilizará para efetuar o acesso ao banco de dados através de um dispositivo móvel.

Vale ressaltar que o desenvolvimento das estruturas de acesso aos dados foi realizado em paralelo com os sistemas SOAC e SoacCel propriamente ditos, à medida em que eram sendo necessárias. Essa foi uma estratégia adotada pois a simples elaboração das estruturas não permitia que elas interagiriam sem conflitos com o sistema e nem permitia que estas pudessem ser testadas.

Tendo visto a forma como foram desenvolvidas as estruturas de armazenamento e interação entre os sistemas SOAC e SoacCel com o banco de dados, iniciou-se então a parte mais complexa do desenvolvimento: o sistema SOAC propriamente dito. Com relação ao SOAC, detalhando o que foi dito anteriormente, é um sistema *web* que consiste em um portal contendo um *site* com diversas funcionalidades colhidas e tratadas na idealização e análise do projeto, e neste *site* contém páginas que são agrupadas de acordo com as funcionalidades destas requeridas.

A elaboração da página principal foi realizada por meio de estruturas e a definição das constantes e funções principais utilizadas pelo SOAC e algumas do SoacCel que foram detectados na modelagem como necessárias às diversas partes do sistema.

Desenvolvidas as funções principais e definidas as constantes adequadas ao sistema, foi criada a estrutura visual básica, que é a interface *web* do SOAC com o usuário. A principal preocupação neste desenvolvimento foi que a interface fosse bastante amigável e, na medida do possível, que possibilitasse a elevada intuitividade do sistema para a pessoa que o está utilizando.

Uma vez com as constantes, funções principais e estruturação visual básica do SOAC definidas, foi a vez do sistema de cadastramento de *login* para computadores pessoais. Esse sistema será explicado de modo mais detalhado no próximo capítulo, mas o

que o mesmo faz é o recolhimento de informações básicas na primeira vez que o sistema for acessado, como nome do usuário do sistema, uma senha para a utilização do SOAC e uma senha para a utilização do SoacCel, de forma que permita proteção contra acesso de pessoas indevidas ao sistema.

Paralelo ao desenvolvimento das funções principais e constantes, foi possível desenvolver também as bases folhas de estilos que possibilitassem o acesso para computadores pessoais (*desktops* e *notebooks*) e *handhelds*. Essas bases sofreram alterações durante todo o desenvolvimento do sistema para se adaptarem de forma otimizada à tela do dispositivo-alvo, e na realidade foram modificadas durante todo o ciclo de criação do SOAC.

Após a estruturação básica da página principal e da folha de estilos, deu-se início ao conteúdo propriamente dito do SOAC.

Visando a máxima facilidade de uso e o tipo de usuário com o qual estamos lidando provavelmente não ter ampla experiência na utilização de computadores, ficou estabelecido durante o processo de modelagem que a forma ideal de interação entre o sistema e o usuário seria que todas as funcionalidades do sistema ficassem disponíveis à vista em uma página principal de acesso que será mostrada em capítulo posterior, a qual seria um formulário principal de opções com todas as funcionalidades representadas em botões, e cada uma destas que relacionassem de forma considerável entre si permanecendo em grupos adequadamente distribuídos na tela.

Neste formulário principal de opções nós temos basicamente 6 grupos de funcionalidades, cada grupo contendo 2 ou 3 funcionalidades para melhor disposição. Todos os grupos começaram a ser desenvolvidos em paralelo, com exceção do grupo *Controle de Relatórios*, que dependia de outras funcionalidades estarem prontas para seu funcionamento.

Nesta seqüência, como o objetivo é mostrar apenas a evolução do sistema não será

mostrada com profundidade o que é a funcionalidade de cada grupo, pois isso faz parte da explicação do sistema, que é referente ao capítulo seguinte.

O grupo de Controle de Insumos buscou permitir ao administrador rural poder gerenciar informações sobre tipos de insumos que ele pode adquirir, como herbicidas; os insumos propriamente dito, como o nome do produto que representa um herbicida; e locais de comercialização de insumos que podem potencialmente vender insumos ou não. De posse destas informações será possível obter facilmente dados sobre como contactar estes locais para efetuar a compra/venda de insumos.

O grupo de controle de funcionários e salários visou o cadastramento e controle de funcionários e parâmetros relativos a pagamentos salariais dos mesmos.

O grupo de controle de despesas concentrou todas as informações do sistema pertinentes ao controle de despesas, como o registro de aquisição de insumos, pagamentos de funcionários e outras despesas. Devido ao forte entrelaçamento entre os dados deste controle e os demais grupos do sistema, este foi provavelmente o grupo de maior complexidade para ser implementado.

Referente ao grupo de cadastro de receitas, foi similar ao grupo de cadastramento de despesas, mas relativo a receitas, neste estão contidos dados do sistema condicionados ao cadastro de produtos que o administrador rural pode vender, vendas destes produtos e outras receitas.

A parte de controle de relatórios é uma das partes mais importantes do sistema. Esta funcionalidade do sistema visa o processamento e geração de recibos e relatórios para o SOAC e, devido ao forte entrelaçamento entre esta e outras funcionalidades, também acabou por ser de custosa complexidade sua elaboração.

O grupo de utilitários objetiva ser uma ferramenta com informações auxiliares ao administrador rural que não puderam ser cadastradas em outras partes do sistema, contendo

funcionalidades como agenda, lista telefônica e alteração dos dados para efetuar o *login* no sistema. Como é possível notar, tais utilitários não fazem parte de um núcleo de cadastramento de informações direta ou indiretamente ligadas à gastos e receitas, tampouco relatórios ou recibos. Este foi mais um motivo para a criação de um grupo separado para tais funcionalidades.

Vale detalhar que apesar de não parecer, assim como em considerável parte do SOAC, aqui também há um entrelaçamento de informações entre as funcionalidades. Por exemplo, a funcionalidade lista telefônica armazena automaticamente informações relativos a locais de aquisição de insumos cadastrados na seção pertinente ao mesmo do sistema.

No entanto foi desenvolvido também um segundo *software*, previamente citado, que permite o acesso via telefones celulares com suporte à plataforma *J2ME*: o SoacCel. Agora serão elucidadas as partes que não se referem diretamente ao funcionamento interno do sistema, que assim como o SOAC será abordado no próximo capítulo.

Embora o autor desta monografia tenha experiência utilizando a linguagem de programação *Java* na plataforma *J2SE (Java 2 Standard Edition)*, voltada para computadores pessoais, há várias diferenças sobre como as plataformas *J2SE* e *J2ME* operam.

Foi estudado como um celular interage com um *software* desenvolvido para esta plataforma através da linguagem *Java*, as configurações de dispositivos limitados (*CLDC*) bem como a interpretação que é feita pela máquina virtual “*Java*” (*KVM*) nestes dispositivos. Ainda, neste período foi analisado formas de como o sistema desenvolvido poderia acessar a Internet e obter dados compartilhados com o SOAC. Ainda, foi verificado a fundo quais funcionalidades seriam úteis para que um celular tivesse.

Tendo em vista a dificuldade de operar um celular para operações consideravelmente extensas, citando como exemplo o cadastramento de funcionários ou

aquisições e, que este dispositivo na prática seria utilizado objetivando praticamente consultas, foi definido que o SoacCel seria um sistema que interagiria com o SOAC no sentido então de apenas de efetuar as consultas mais importantes para o usuário do sistema.

Após a filtragem fina de quais funcionalidades o SoacCel deveria possuir, foi dado início ao projeto de desenvolvimento. A primeira preocupação neste processo foi a forma como o SoacCel deveria acessar os dados administrados pelo sistema gerenciador de banco de dados. Uma possível forma seria a conexão direta entre o SoacCel e o banco de dados, através de um *driver*, e todo o processamento e intermédio de informações entre o usuário do sistema, tratamento dos dados de entrada e comunicação com o banco de dados seria feito através do celular. Esta é uma boa alternativa, porém, no caso para o qual o SoacCel se propõe pode exigir um processamento desnecessário do celular para lidar com conexões a bancos de dados, *statements* e *resultsets*, além de um possível consumo elevado de memória.

Realizadas pesquisas mais refinadas foi possível verificar que para lidar com este problema se é possível fazer algo mais simples. O acesso a *web* para obtenção de dados do banco de dados é inevitável, então por que não utilizar apenas o protocolo de comunicação *HTTP* entre o servidor *web* e o celular? Isso é possível, e todo o processamento que é pesado para o celular se torna simples para tal servidor. *PHP* possui em sua estrutura a capacidade de obter variáveis enviadas através do próprio pedido de requisição de página utilizado pelo protocolo *HTTP*, interpretá-las e utilizá-las de forma favorável no sistema. Essa estrutura para leitura de variáveis passadas através do protocolo *HTTP* pelo *PHP* é conhecida como *Super Array GET*. Desta forma, se é possível fazer com que um dispositivo celular acesse o servidor *web*, enviar e requisitar dados, tudo isso através de uma única requisição de página. Uma vez com os dados recebidos pelo servidor *web* que possua o pré-processador *PHP* instalado, o mesmo age normalmente para obter os dados do banco de dados como é feito no SOAC, e processa os dados retornando ao celular os dados processados da forma adequada desejada por este dispositivo.

Uma vez idealizado como seria realizado esta transferência entre celular e servidor,

foi dado início ao desenvolvimento do SoacCel. Inicialmente, foi elaborado as classes e métodos que permitiam *web* através do protocolo *HTTP* via celular.

Tendo realizado a parte que comunicava no servidor *web* a partir do celular, foi necessário configurar o processamento das informações recebidas e tratamento das informações que seriam enviadas para o telefone celular, em formato de relatório.

Paralelo ao desenvolvimento do tratamento de informações recebidas e enviadas no servidor *web*, e em iguais datas de início e encerramento, foi elaborado um sistema de validação *login* (para validação no servidor *web*) feito no próprio dispositivo celular. Tal sistema de validação de *login* também utiliza o mesmo sistema de criptografia de senhas do SOAC (*SHA1*, o qual será falado mais detalhadamente no próximo capítulo), mas o processo de *hashing* para obtenção do *digest* neste caso é realizado pelo próprio telefone celular com bibliotecas que acompanham o *J2ME*.

Após ter desenvolvido as bases para conexão *web* através do celular e os devidos tratamentos no servidor, assim como o sistema de *login* começou então a elaboração das funcionalidades do sistema SoacCel propriamente ditas. Estas funcionalidades, por serem processadas pelo servidor *web*, reaproveitaram consideravelmente o código já existente do SOAC para serem desenvolvidas.

Na prática, foi possível desenvolver todas as funcionalidades do sistema principal em paralelo, funcionalidades estas que compreendiam a tela, o motor e a obtenção de dados para os sistemas de consulta disponíveis para o SoacCel, a saber: consultas de funcionários, agenda, geração de relatórios simplificados e consulta de lista telefônica. Neste período também foi desenvolvido um sistema universal para retorno de dados consultados no servidor *web*, que trata todas as requisições pedidas.

Findado o desenvolvimento do sistema e a elaboração da monografia, foi possível fazer uma análise de resultados do sistema. O código-fonte do trabalho desenvolvido está disponível na Internet no endereço www.comp.ufla.br/~xguimax.

4. SOAC: DESCRIÇÃO DO SISTEMA

Neste capítulo será abordado um dos resultados obtidos após o desenvolvimento deste projeto, no caso o *software* SOAC, juntamente com uma descrição detalhada de cada parte do funcionamento do mesmo.

Com os procedimentos metodológicos da construção do SOAC, Sistema Online para Auxílio ao Cafeicultor, devidamente demonstrado, foi possível se ter uma noção global do que é o sistema, mas este ainda não foi demonstrado nem explicado. Serão mostrados os principais módulos do sistema, divididos em *Registro Inicial de Usuário*, *Menu Principal*, *Cadastrros de Insumos*, *Cadastrros de Funcionários*, *Cadastrros de Controle de Despesas*, *Cadastro de Receitas*, *Geração de Relatórios*, *Utilitários* e ao final serão mostradas as *Decisões de Desenvolvimento* que foram tomadas durante o processo de construção do *software*.

É o objetivo deste capítulo também a explicação detalhada, tela a tela, do funcionamento do sistema para o usuário. Alguns detalhes de implementação específicos serão abordados em cada uma das telas, outros detalhes e preocupações globais aplicados a toda a implementação serão ditos na última seção deste capítulo.

Inicialmente será apresentado como o SOAC se comporta logo após sua instalação, efetuando um registro do usuário que fará o uso do sistema e como este procedimento é feito.

4.1. REGISTRO INICIAL DE USUÁRIO

Para a utilização do sistema SOAC, devido a natureza deste conter dados que podem ser sigilosos como será mostrado a frente, é importante que se tenha um controle de acesso ao mesmo. A medida tomada na modelagem para efetuar este controle de acesso é a utilização de um sistema de *login* com senha para permitir que o usuário do sistema se

identifique e somente mediante esta condição tal usuário ter acesso às demais partes do sistema.

Uma decisão de implementação foi efetuar o registro do primeiro usuário logo após o primeiro acesso ao sistema, requisitando a este os dados que o autenticarão no SOAC conforme mostrado na figura 6.

SOAC - Sistema Online de Auxílio ao Cafeicultor

Para utilização do SOAC - Sistema Online de Auxílio ao Cafeicultor, você necessita de cadastrar um nome de usuário, uma senha para acessar o sistema através do site e uma senha para acessar o sistema através do celular.

Informe os dados abaixo para completar o registro:

Login:	<input type="text" value="admin"/>
Senha (Site):	<input type="password" value="* * * *"/> #
Confirmar Senha (Site):	<input type="password" value="* * * *"/> #
Senha (Celular):	<input type="password" value="* * * *"/> #
Confirmar Senha (Celular):	<input type="password" value="* * * *"/> #

Registrar

SOAC - Sistema Online de Auxílio ao Cafeicultor
Desenvolvido por: Daniel Guimarães do Lago
Universidade Federal de Lavras

Figura 6: Registro de Usuário visto de um computador desktop.

Essa será a tela que permitirá o cadastro do *login* e senha através dos quais o usuário do sistema poderá acessá-lo, bem como uma senha que pode ser diferente da senha do *site* para acesso via celular. Uma vez cadastrados estes parâmetros, toda vez que o sistema for acessado será requerida uma autenticação do usuário, onde este deverá fornecer um nome de usuário e senha condizentes com o que foi fornecido previamente.

O registro deste *login* e senha também poderão ser realizados se o servidor que hospeda o sistema estiver sendo acessado por meio de um dispositivo *handheld*. Além disso, houve preocupações extras de segurança no desenvolvimento desta tela além das descritas no último tópico deste capítulo.

A primeira preocupação foi que um mesmo usuário tivesse uma opção de escolher uma senha diferente de acesso ao *site* via um *browser* ou via telefone celular. Tal preocupação foi devido ao fato de se descoberta a senha de uma das formas de acesso ao *site*, a outra permaneceria intacta. Isso ajuda a garantir a integridade do sistema, tendo em vista que para telefones celulares apenas a consulta de dados é permitida, enquanto através do *browser* o usuário pode alterar os dados do sistema. Em outras palavras, se um usuário tiver a senha do sistema para acesso via telefone celular roubada, a pessoa que obteve esta senha não conseguirá modificar os dados do sistema.

Além disso, houve preocupação com o armazenamento da senha. A senha não foi armazenada como uma *string* crua no banco de dados, o que permitiria que qualquer pessoa que tivesse acesso ao sistema gerenciador de banco de dados obtivesse a senha. A mesma passa por um algoritmo denominado *SHAI*.

O *SHAI* (RFC 3174, 2007) é um algoritmo utilizado para criptografia monodirecional para dados que sofre *efeito avalanche* mais seguros do mundo atualmente. O que o *SHAI* basicamente faz, é obter uma entrada de qualquer tamanho, realizar pesadas operações matemáticas em cima da mesma e obter de forma monodirecional uma palavra de tamanho 40 (para caracteres hexadecimais). Monodirecional quer dizer que com uma entrada o algoritmo produz uma saída, mas com a saída não é mais possível obter-se a entrada. Desta forma, é garantido que apenas quem saiba a senha consiga acessar o SOAC, mesmo que outra pessoa consiga obter de alguma forma obter informações acerca dos dados referentes a *login* e senha. O *efeito avalanche* sofrido pelo *SHAI* dificulta utilização de técnicas dedutivas para tentativas de descoberta de senha. Essa garantia se dá ao fato que entradas muito semelhantes, porém diferentes, produzam resultados completamente diferentes entre si após sofrerem a criptografia gerada pelo *SHAI*.

Relativo a opção entre tornar este sistema que permitisse um único *login* ou vários *logins*, foi escolhido que o mesmo deveria ser monousuário, pois imagina-se que quem vai utilizar frequentemente o sistema deve ser um único administrador rural ou pessoas em

quem o mesmo tenha total confiança e não teria motivos para não fornecer, e também por facilidade de implementação e das restrições que tornar este sistema completamente multiusuário acarretaria.

Uma consideração importante que deve ser ressaltada aqui é que devido ao fato das informações contidas neste sistema serem de caráter sigiloso, seria interessante que o servidor o qual rodasse o SOAC utilizasse um sistema de envio/recebimento de dados utilizando protocolos de comunicação seguros, como SSL ou TLS (RFC 2246, 1999). O SSL (*Secure Sockets Layer*) e o TLS (*Transport Layer Security*) são protocolos padrões para transmissões seguras através da Internet, criptografando os dados que serão enviados entre o servidor e computador do usuário.

Uma vez que o usuário do sistema tenha registrado um *login* e *senha*, será requisitado toda a vez que o mesmo acessar o SOAC ou SoacCel o *login* e *senha* cadastrados.

4.2. MENU PRINCIPAL

Quando um usuário registrado efetua o *login* no SOAC, ele passa a ter acesso ao menu principal do SOAC. Este menu principal permite que o usuário navegue entre todos os recursos que o sistema provê, através de um tela projetada para ser bastante intuitiva e de fácil entendimento.

O menu principal foi dividido em grandes grupos de cadastros, e cada grupo contém opções referentes aos objetivos deste. Dessa forma, é propiciada uma organização lógica dos dados referentes a cada grupo de cadastros, pois é considerada uma boa prática de usabilidade de interface homem-máquina.

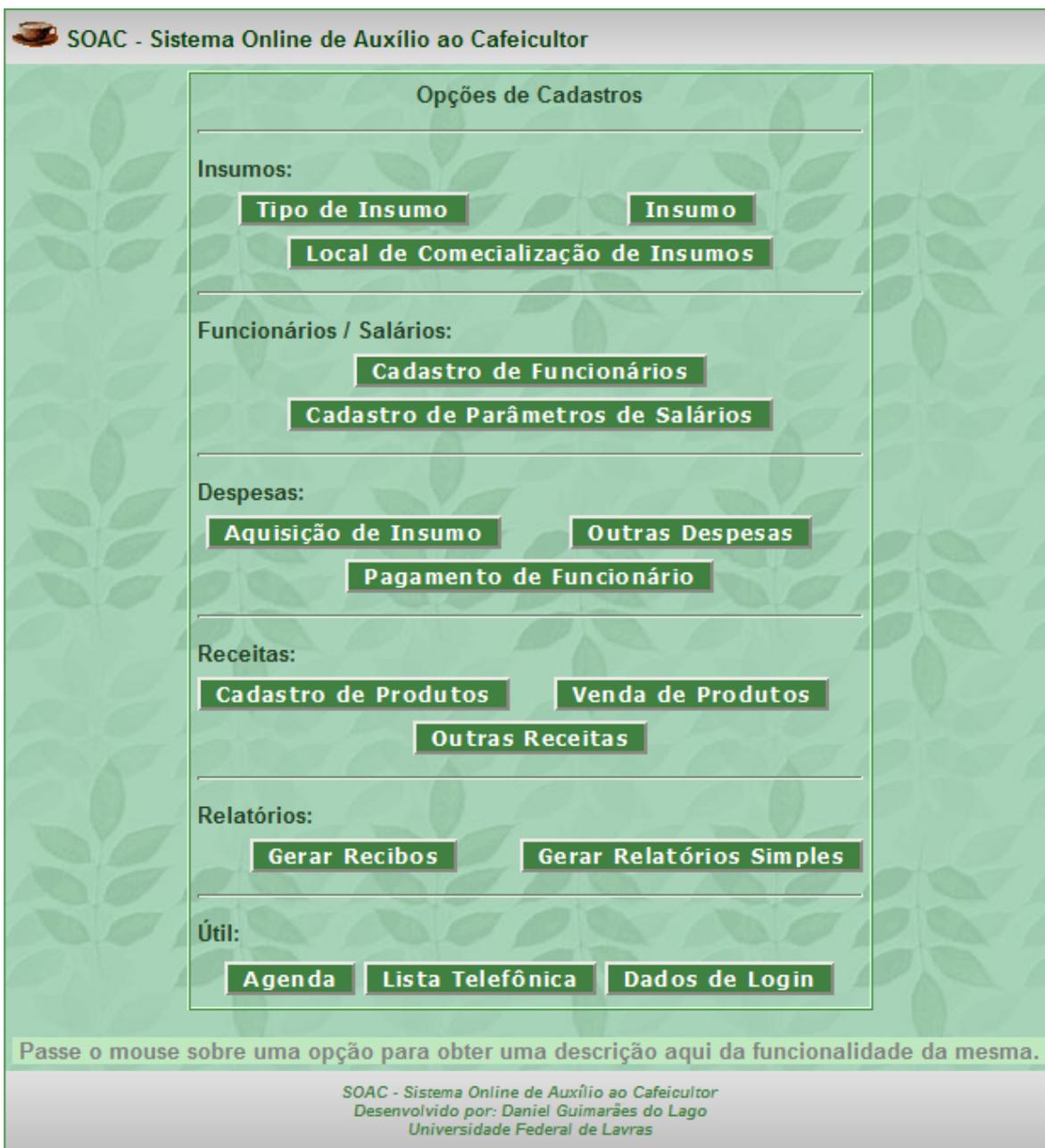


Figura 7: Menu Principal visto de um computador desktop.

Os principais grupos de cadastros são:

- Insumos
- Funcionários / Salários
- Despesas
- Receitas

- Relatórios
- Utilitários

Na figura 7 é mostrado o modelo da tela do menu principal, sendo este o que é o efetivamente visível para usuários do sistema quando o acessam através de algum *browser* por um computador pessoal.

Assim como em todas as telas do sistema, houve preocupação nesta tela com relação ao usuário. O padrão esperado de um usuário como administradores rurais com relação à utilização de computadores com sistemas como o SOAC é que estes tenham pouca experiência com tais máquinas. Entre as ações tomadas para tentar maximizar o entendimento do sistema por tais usuários, além do agrupamento lógico de funcionalidades do sistema, foi uma explicação sucinta, porém substancial de cada opção disponível. Nessa interface, quando o usuário passa o ponteiro do mouse sobre qualquer botão que possua uma funcionalidade, é exibida na barra de ferramentas uma descrição simplificada falando acerca do conteúdo de cada funcionalidade, dando uma noção melhor do que o sistema é capaz de fazer pelo usuário.

No sistema também foi modelado que quando o ponteiro não estiver sobre nenhuma parte da tela capaz de interagir com o usuário diretamente, que a barra de *status* também exibisse uma informação de retorno descrevendo que aguarda que se o usuário passar o mouse sobre algum objeto na tela com capacidade de interação, seria mostrado ao usuário o que este objeto pode fazer no formato de um texto na barra de status com informações.

Será mostrado detalhadamente o que cada um destes grupos permite ao usuário e como deve ser feita nos tópicos referentes ao mesmo. Assim como em *browsers* para computadores pessoais e *laptops*, para obter uma otimização da navegação houve também preocupação com relação a exibição do formulário principal de opções em *browsers handhelds*, buscando tornar as telas acessadas através de *browsers* de computadores pessoais e *handhelds* semelhantes, dentro das limitações impostas pelo *hardware* de cada uma das máquinas.

4.3. CADASTROS DE INSUMOS

Neste tópico será abordado o primeiro grupo de opções que é exibida ao usuário na tela, que é o grupo de operações referentes ao cadastramento de insumos. Este grupo visa dar ao usuário um controle melhor sobre insumos e dados sobre aquisições de insumos desejados pelos usuário.

Foi objetivado neste grupo possíveis dinâmicas futuras de insumos e adaptabilidade do sistema sem necessidade de intervenção de administradores do sistema ou programadores para tal tarefa, permitindo maior controle ao usuário.

4.3.1. TIPOS DE INSUMOS

O tempo passa, e uma das conseqüências que estamos vivendo deste pressuposto é que a tecnologia também evolui com o tempo. Há alguns anos não existia adubos químicos para a produção de café (fertilizantes), nem substâncias de combate a pestes (pesticidas), ervas daninhas (herbicidas) ou fungos (fungicidas). Na figura 8 é mostrada uma das funcionalidades do SOAC, o cadastramento de tipos de insumos.

Em outras palavras, com o decorrer do tempo os tipos de insumos agrícolas evoluíram e surgiu uma enorme gama de tais tipos. Com o decorrer dos anos é possível, senão provável, que novos tipos de insumos surjam. Se não houver preocupação com tal evolução em um sistema que pode não sofrer muitas alterações futuras, é interessante que o controle ao cadastramento de tais tipos de insumos caibam ao usuário.

Para facilitar o entendimento do funcionamento do SOAC com insumos, para este e para os tópicos seguintes vamos admitir que até o presente momento não houvesse ainda sido inventados produtos químicos à base de glifosato, que basicamente são defensivos

agrícolas que servem para combater ervas daninhas. Embora já batido esse tipo de insumo, o nosso foco não é esse: é demonstrar como um novo tipo de insumo pode ser inserido e o sistema adequado ao novo que foi criado. Tal tipo de insumo, mesmo que seja considerado uma novidade e o sistema não tenha sido projetado para lidar com ele, pode ser inserido ou modificado facilmente através desta funcionalidade do sistema. Essa funcionalidade busca primariamente facilitar a classificação de insumos cadastrados no sistema, conforme será visto no próximo tópico.

The screenshot shows the SOAC (Sistema Online de Auxílio ao Cafeicultor) interface. At the top, there is a header with a coffee cup icon and the text "SOAC - Sistema Online de Auxílio ao Cafeicultor". Below this, the main heading is "Tipos de Insumo". The primary instruction is "Incluir novo tipo de insumo:". A form is provided with three input fields: "Nome do tipo de insumo:" with the example "ex: Herbicida" and the value "Herbicida"; "Função deste tipo de insumo:" with the example "ex: Combate à ervas daninhas." and the value "Combate ervas daninhas."; and "Unidade de comercialização do insumo:" with the example "ex: litros" and the value "litros". A green button labeled "Inserir Tipo de Insumo" is positioned below the form. A light green banner below the form contains the text "Passe o mouse sobre uma opção para obter uma descrição aqui da funcionalidade da mesma." At the bottom of the interface, a message states "Nenhum tipo de insumo está cadastrado atualmente." The footer includes the text "SOAC - Sistema Online de Auxílio ao Cafeicultor", "Desenvolvido por: Daniel Guimarães do Lago", and "Universidade Federal de Lavras".

Figura 8: Cadastro de Tipos de Insumos visto de um computador desktop.

Um detalhe ainda não mencionado até o presente momento é com relação a exibição dos tipos de insumos existentes ao usuário. Note, em ambas ilustrações mostradas, que existe um espaço para listagem de tipos de insumos cadastrados atualmente no sistema ao final da tela. Nesta listagem, todos os insumos cadastrados serão mostrados, como exemplificado na figura 9, após o inserção do tipo de insumo “herbicida”.

 **SOAC - Sistema Online de Auxílio ao Cafeicultor**

Tipos de Insumo

Incluir novo tipo de insumo:

Nome do tipo de insumo:
ex: *Herbicida*

Função deste tipo de insumo:
ex: *Combate à ervas daninhas.*

Unidade de comercialização do insumo:
ex: *litros*

Inserir Tipo de Insumo

Passe o mouse sobre uma opção para obter uma descrição aqui da funcionalidade da mesma.

Tipos de insumos cadastrados:

Nome: HERBICIDA	Unidade Comercializada: LITROS	excluir
Função: COMBATE ERVAS DANINHAS.		

SOAC - Sistema Online de Auxílio ao Cafeicultor
Desenvolvido por: Daniel Guimarães do Lago
Universidade Federal de Lavras

Figura 9: Cadastro de Tipos de Insumos após inserção do tipo "Herbicida".

4.3.2. INSUMOS

Foi visto no tópico anterior informações relevantes acerca do cadastramento de tipos de insumos e adaptabilidade do SOAC à criação de novos tipos de insumos. Tipos de insumos podem surgir em intervalos bastante espaçados de tempo, no entanto, insumos

propriamente dito surgem diariamente.

A ilustração representada pela figura 10 foi utilizada para demonstrar uma funcionalidade do grupo de cadastros de insumos: utilizando ainda o exemplo citado no tópico anterior, referente à criação de um novo tipo de insumo (herbicida), suponhas que uma empresa denominada *Mansanta* (neste caso um exemplo genérico meramente ilustrativo) criou um insumo do tipo previamente citado sob o nome de *Round Down*. É importante para o usuário do sistema, se for utilizar este insumo em sua produção, que adapte o sistema para reconhecer tal insumo.

SOAC - Sistema Online de Auxílio ao Cafeicultor

Insumos:

Incluir novo insumo:

Nome do insumo:
ex: Round Down

Tipo do insumo:

Comentário acerca deste insumo:
ex: Fabricado pela XYZ Corporation.

Unidade de comercialização do insumo:
ex: litros

Round Down

HERBICIDA

Fabricado pela Mansanta.

LITROS

Inserir Insumo

Passa o mouse sobre uma opção para obter uma descrição aqui da funcionalidade da mesma.

Nenhum insumo está cadastrado atualmente no banco de dados.

SOAC - Sistema Online de Auxílio ao Cafeicultor
Desenvolvido por: Daniel Guimarães do Lago
Universidade Federal de Lavras

Figura 10: Cadastro de Insumos visto através de um computador desktop.

Como se pode ver na figura 10, a parte utilizada referente ao “tipo de insumo” foi exatamente a mesma parte previamente cadastrada na funcionalidade exibida no tópico

anterior, neste caso um tipo “herbicida”. É um detalhe técnico, mas é importante ressaltar que a usabilidade do sistema foi um fator de tamanha preocupação que até mesmo o campo referente à “Unidade de Comercialização de Insumo” possui por *default* o mesmo valor obtido deste parâmetro existente em “Tipo de Insumo”. Na figura 11 está de mostrada a tela pós-inserção de insumo.

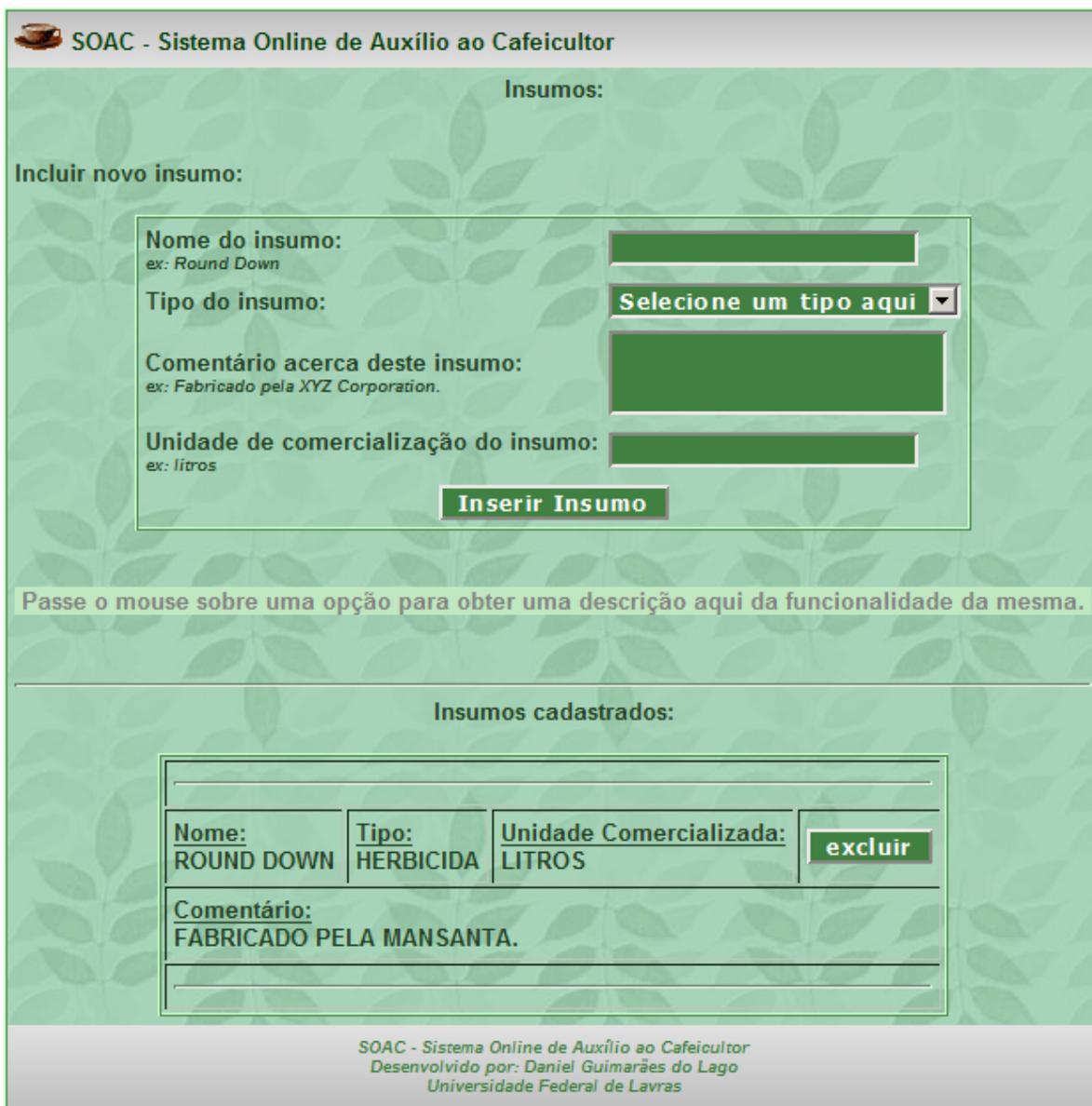


Figura 11: Cadastro de Insumos após inserção do insumo "Round Down".

Com relação à exibição dos resultados dos insumos inseridos, a mesma política de

listagem de tipos de insumos foi tomada: todos os insumos do sistema são exibidos em uma listagem após a opção de inserção de insumos. Embora o número de insumos seja superior ao número de tipos de insumos, dificilmente um administrador rural utilize tantos tipos de insumos que prejudique substancialmente a navegação do mesmo pelo sistema.

Deve-se ressaltar que todas as listagens, tanto a de insumos como a de tipos de insumos são realizadas em ordem alfabética, buscando o melhor desempenho possível da interação entre o usuário e o sistema. Todos os “tipos de insumos” cadastrados previamente no sistema estão disponíveis na listagem *drop-down* para facilitar o usuário a selecionar o tipo de determinado insumo, e assim como a listagem de insumos, estão também disponíveis em ordem alfabética.

4.3.3. LOJAS DE INSUMOS

Foi demonstrado até o tópico anterior como o usuário fará a inserção e consulta de dados através do SOAC para visualização e inserção de insumos e tipos de insumos. No entanto, esse tipo de informação é simplesmente inútil se não estiver atrelado à compra de um insumo em uma loja. É óbvio deduzir que se um usuário cadastrou um insumo no sistema ele vai querer também cadastrar lojas que comercializem insumos e uma forma de acessar facilmente informações para entrar em contato com tais lojas.

A capacidade de lidar também com dados pertinentes a tais lojas é uma das funcionalidades do SOAC. Nesta funcionalidade, o usuário poderá cadastrar informações pertinentes a lojas que comercializem insumos em geral, e obter os dados das mesmas em uma listagem. Vamos ilustrar esta funcionalidade em ação inserindo um dado bastante comum: grande parte dos administradores rurais hoje são filiados a cooperativas. Nossa inserção baseia-se numa cooperativa denominada *Coocidade* localizada no *centro* da cidade de *Pequenópolis* em *Minas Gerais*. Um administrador rural tem bastante contato com tal local e deseja cadastrá-lo no SOAC. Esta situação está ilustrada pela figura 12 e a

interface gerada após a devida inserção está mostrada na figura 13.

SOAC - Sistema Online de Auxílio ao Cafeicultor

Loja de Insumos:

Incluir nova Loja:

Nome da Loja: <small>ex: Coocidade</small>	<input type="text" value="Coocidade"/>
Rua: <small>ex: José Papaguaio</small>	<input type="text" value="Don João VII"/>
Número: <small>ex: 123</small>	<input type="text" value="456"/>
Complemento: <small>ex: Sala 53</small>	<input type="text" value="Sala 13"/>
Bairro: <small>ex: Jardim Eustáquio</small>	<input type="text" value="Centro"/>
Cidade: <small>ex: Ciudadópolis</small>	<input type="text" value="Pequenópolis"/>
Estado (Sigla): <small>ex: MG</small>	<input type="text" value="MG"/>
Telefone 1: <small>DDD + Número, apenas dígitos. 3512345678</small>	<input type="text" value="3187654321"/>
Telefone 2: <small>DDD + Número, apenas dígitos. 3512345678</small>	<input type="text" value="3112345678"/>
Celular 1: <small>DDD + Número, apenas dígitos. 3512345678</small>	<input type="text"/>
Celular 2: <small>DDD + Número, apenas dígitos. 3512345678</small>	<input type="text"/>
Fax 1: <small>DDD + Número, apenas dígitos. 3512345678</small>	<input type="text"/>
Fax 2: <small>DDD + Número, apenas dígitos. 3512345678</small>	<input type="text"/>

Passa o mouse sobre uma opção para obter uma descrição aqui da funcionalidade da mesma.

Lojas de Insumos Cadastradas:

SOAC - Sistema Online de Auxílio ao Cafeicultor
Desenvolvido por: Daniel Guimarães do Lago
Universidade Federal de Lavras

Figura 12: Cadastro de Lojas de Insumos visto através de um computador desktop.

Assim como o número de insumos existentes, o número de lojas de insumos, as quais um administrador rural entra em contato geralmente não é muito grande, e foi definido para o desenvolvimento do sistema que o mais interessante ao usuário final a visão das mesmas em uma listagem seqüencial. Ainda, se este número crescer demasiadamente devido a utilização desta área para outros fins, o acesso aos dados de determinada loja poderá ser feito através de uma funcionalidade entrelaçada a esta: a lista telefônica. Basicamente, todas as lojas cadastradas nesta funcionalidade, terá seus dados inseridos também de forma automática na lista telefônica. Esta não é uma inserção duplicada, é um compartilhamento de tabelas com dados semelhantes para finalidades correlacionadas.

3512345678

Fax 1:
DDD + Número, apenas dígitos.

3512345678

Fax 2:
DDD + Número, apenas dígitos.

3512345678

Inserir Loja de Insumos

Passa o mouse sobre uma opção para obter uma descrição aqui da funcionalidade da mesma.

Lojas de Insumos Cadastradas:

COOCIDADE	
Telefone (1): 3187654321	Telefone (2): 3187654321
Celular (1): Não cadastrado.	Celular (2): Não cadastrado.
Fax (1): Não cadastrado.	Fax (2): Não cadastrado.
Endereço: Rua DON JOAO VII, número 456 SALA 13, bairro CENTRO. Cidade: PEQUENOPOLIS, MG	

excluir

SOAC - Sistema Online de Auxílio ao Cafeicultor
Desenvolvido por: Daniel Guimarães do Lago
Universidade Federal de Lavras

Figura 13: Cadastro de Lojas de Insumos após inserção da loja "Coocidade".

4.4. CADASTROS DE FUNCIONÁRIOS E PARÂMETROS SALARIAIS

Nesta seção será descrito o grupo de funcionalidades do Sistema Online de Auxílio ao Cafeicultor com Foco na Mobilidade responsável pelo controle de funcionários e parâmetros salariais. Este grupo visa a criação de embasamento e controle para o grupo de funcionalidades relacionado com despesas.

4.4.1. FUNCIONÁRIOS

Para criar uma funcionalidade que permita operações de cadastro de funcionários para um sistema como o SOAC, deve-se ter em vista o enorme dinamismo no corpo de trabalho de empresas rurais. Em épocas normais um grupo fixo de trabalhador é mantido, porém, em épocas de colheita esse número de trabalhadores multiplica-se, devido a mão-de-obra extra necessária.

É necessário para os administradores rurais obter informações constantemente de trabalhadores atualmente ativos, porém esporadicamente é necessário também a obtenção de informações de um enorme número de trabalhadores inativos. Visando a elaboração do sistema para lidar com este tipo de situação, deve-se projetá-lo de forma que não prejudique a usabilidade devido um número grande de trabalhadores inativos.

Além dos problemas anteriormente citados, foi constatado que os principais dados os quais uma empresa rural necessita são as datas de admissão e demissão de um funcionário, bem como um documento que o identifique, como um Cadastro de Pessoa Física e, eventualmente, informações sobre o salário base do funcionário em questão para eventuais operações.

SOAC - Sistema Online de Auxílio ao Cafeicultor

Cadastro de Funcionários

Operação:
 Procurar funcionário Inserir funcionário

Dados do funcionário:

Nome: João da Silva

CPF: 12345678901

Admissão: 21 11 2007
dia dd mês mm ano aaaa

Demissão: 21 11 2007
dia dd mês mm ano aaaa

Salário Base: R\$ 800 , 00

Ativo

Inserir

Passe o mouse sobre uma opção para obter uma descrição aqui da funcionalidade da mesma.

Funcionários Ativos:

Nenhum funcionário ativo está cadastrado no banco de dados.

SOAC - Sistema Online de Auxílio ao Cafeicultor
Desenvolvido por: Daniel Guimarães do Lago
Universidade Federal de Lavras

Figura 14: Cadastro de Funcionários visto através de um computador desktop.

Tendo analisado todas essas necessidades e limitações, foi projetado a tela referente ao cadastro de funcionários. Na figura 14 está uma ilustração da mesma, vista de um computador pessoal *desktop*, neste exemplo mostrando a inserção de *João da Silva*, com CPF nº 12345678901, como sendo um funcionário ativo que começou a trabalhar em 21 de novembro de 2007 com um salário base de R\$ 800,00 (oitocentos reais).

Na tela mostrada, é possível perceber que o espaço que nas telas para cadastros de insumos eram usados para listagens em geral, nesse caso sofreu uma filtragem. Foi

demonstrado anteriormente que há um grande *turn over* de funcionários em empresas rurais, pois a colheita influencia fortemente nas necessidades de tais empresas. Por esta razão, apenas uma listagem básica de funcionários foi mantida, e adicionado novos recursos ao sistema.

Tanto no sistema acessado através de computadores pessoais, como *desktops*, e portáteis, como *handhelds*, houve a adição de uma caixa de *rádio* perguntando ao usuário qual a *operação* que o mesmo deseja fazer.

Como se pode notar, nessa tela há duas opções ao usuário: *procurar* e *inserir* funcionários. Uma política adotada no SOAC é que estas opções sejam todas visíveis de uma mesma página, modificando dinamicamente a página acessada pelo usuário utilizando *javascript* para facilitar a interação e também reduzir o consumo de banda que seria necessário de banda para que *handhelds* ao acessar esses dados necessitando carregar páginas inteiras.

Relacionado às operações possíveis, para efetuar uma inserção, o usuário deve clicar no botão de *rádio* inserir, fornecer os dados do funcionário que ele deseja inserir no sistema, definir se ele está ativo caso ele esteja ativo e trabalhando na empresa ou indicar a data na qual o mesmo tenha sido demitido, para cadastramentos de funcionários que trabalharam previamente mas que se deseja ter o cadastro dele.

Exemplificando uma operação de inserção, vamos adotar que o usuário deseja inserir um funcionário com o nome de João da Silva no sistema, que possui salário base de R\$ 800,00, que este funcionário está em atividade e todas as informações mostradas na ilustração anterior. Após a inserção deste funcionário no sistema, o mesmo será visível através da listagem de funcionários ativo, conforme demonstrada na figura 15.

Para efetuar uma busca, o usuário deve deixar marcado o botão de *rádio* procurar e fornecer o CPF do funcionário desejado. Caso o usuário não possua o CPF do funcionário para a operação de busca, ele poderá optar por digitar o nome parcial do funcionário que a

pesquisa retornará todos os funcionários que têm nome parcial semelhantes ao nome informado para a operação. A listagem de funcionários encontrada é retornada na tela e também é semelhante à listagem de funcionários ativos.

The screenshot displays a web interface for employee registration. At the top, there is a form for 'Salário Base' (Basic Salary) with a value of R\$ 0,00 and a checked 'Ativo' (Active) checkbox. Below this is a note: 'Nota: A busca será pelo CPF se informado, senão por parte do nome. Se nenhum for informado, serão procurados funcionários admitidos entre as datas de admissão e demissão especificadas.' A green 'Inserir' (Insert) button is positioned below the note. A light green banner below the form reads: 'Passe o mouse sobre uma opção para obter uma descrição aqui da funcionalidade da mesma.' The main section is titled 'Funcionários Ativos:' and contains a detailed view for 'ID do funcionário 1' with the name 'JOAO DA SILVA'. It shows the 'CPF: 12345678901', 'Salário Base: R\$ 800,00' with an 'editar' (edit) button, and 'Admissão: 21/11/2007' with a date picker showing '21' for the day, '11' for the month, and '2007' for the year. Below the date is the text 'Em atividade.' and a 'demitir' (dismiss) button. At the bottom of the interface, the footer text reads: 'SOAC - Sistema Online de Auxílio ao Cafeicultor Desenvolvido por: Daniel Guimarães do Lago Universidade Federal de Lavras'.

Figura 15: Cadastro de Funcionários após inserção de "João da Silva".

4.4.2. PARÂMETROS DE SALÁRIOS

Um dos objetivos do SOAC, além das já citadas mobilidade e portabilidade, foi a adaptabilidade. Adaptabilidade nesse caso quer dizer que este *software*, mesmo que não seja atualizado durante um grande período de tempo, pode manter-se útil para o usuário de

forma que o próprio usuário consiga criar dentro de seu sistema recursos que o mesmo necessitar utilizar, ou seja, permitir com que o próprio usuário adapte o *software* às suas necessidades.

Provavelmente a principal demonstração de adaptabilidade do SOAC encontra-se nessa funcionalidade. Parâmetros salariais permitem que, mesmo quando um tipo de parâmetro para pagamento de salários de funcionários não existam no sistema, o usuário – administrador rural – possa cadastrar novos parâmetros. Por exemplo: vamos supor que no futuro funcionários rurais passem a receber salários também pela irrigação de árvores do produto cultivado (no nosso caso, mais afundo, o café); e passem a ser pagos pela quantidade de plantas irrigadas. Será possível, de forma simples, inserir os novos parâmetros de pagamentos salariais para efetuar o pagamento destes funcionários sem nenhuma necessidade de atualização do sistema no sentido de ter que reinstalá-lo completamente ou parcialmente, uma vez que esta função poderá ser facilmente desempenhada pelo administrador rural.

Logo, uma vez tendo cadastrado os funcionários, se desejar obter um controle também sobre os pagamentos efetuados e as despesas é interessante que seja realizado o cadastramento de parâmetros salariais. Esta funcionalidade está diretamente entrelaçada com outras, como a funcionalidade do grupo de despesas de pagamento de funcionário e a funcionalidade do grupo de relatórios de geração de recibos. A figura 16 demonstra esta funcionalidade.

Nota-se, na tela representada pela figura 16, poucas opções mas que são todo o motor de outras funcionalidades. Nessa tela o administrador rural pode inserir novos parâmetros salariais. Os parâmetros salariais existentes são indicados em uma listagem logo após a opção de inserção. Na listagem, também é possível excluir parâmetros cadastrados. A seguir será explicado o que significa cada item deste parâmetro.

Basicamente, em toda a estrutura do sistema de pagamento de funcionários, cada pagamento salarial é constituído por parâmetros salariais e cada parâmetro é constituído

por 3 campos: *prefixo*, *infixo* e *sufixo*. Os três campos são requisitados para o usuário preencher.



Parâmetros de Salários

Incluir novo parâmetro de salário:

Prefixo:
ex: Pela colheita de

Infixo:
ex: sacos de café a R\$

Sufixo:
ex: utilizando colheita manual

Positivo (este valor incrementará o salário final do funcionário).
ex: se for colheita efetuada por funcionário, marque esta caixa; se for adiantamento, desmarque-a.

Passe o mouse sobre uma opção para obter uma descrição aqui da funcionalidade da mesma.

Parâmetros de salários cadastrados:

Positivo: SIM	Prefixo: PELA COLHEITA DE	Infixo: MEDIDAS DE CAFE A R\$	Sufixo:
Formato: PELA COLHEITA DE [QUANTIA] MEDIDAS DE CAFE A R\$ [VALOR]			
(soma ao salário do funcionário [QUANTIA] X [VALOR])			<input type="button" value="excluir"/>
Positivo: SIM	Prefixo: PELA COLHEITA DE	Infixo: MEDIDAS DE CAFE A R\$	Sufixo:
Formato: PELA COLHEITA DE [QUANTIA] MEDIDAS DE CAFE A R\$ [VALOR]			
(soma ao salário do funcionário [QUANTIA] X [VALOR])			<input type="button" value="excluir"/>

Figura 16: Cadastro de Parâmetros de Salários visto através de um computador desktop.

Um detalhe que ainda não foi mencionado previamente é que, existe um campo *Positivo* nesta funcionalidade. Este campo serve como uma identificação se este parâmetro salarial incrementará ou decrementará o salário do funcionário. Se ele estiver marcado, esse será um parâmetro salarial que será somado ao salário final do indivíduo, caso contrário será subtraído. Isso é necessário muitas vezes. Para ilustrar a necessidade deste campo como parâmetro salarial, imagine a seguinte situação: um administrador rural possui um funcionário que mora na fazenda em uma casa de colono. Não é justo ao administrador rural ter que arcar com algumas despesas deste funcionário, como por exemplo, despesa de energia elétrica consumida por este. No entanto, em muitas propriedades rurais as contas das tarifas de energia não são direcionadas aos funcionários, e sim aos administradores. Se o administrador rural julgar necessário por algum motivo abater os gastos de energia elétrica do salário de um funcionário, ele poderá criar um parâmetro tendo o campo *infixo* como Pelo consumo de, o campo *sufixo* como Kilowatts * hora a R\$ e o campo *sufixo* deixando sem preencher, e desmarcando o campo positivo. Neste caso, supondo que o funcionário tenha consumido 50 kilowatts * hora a R\$ 0,69 cada, no sistema de pagamento e recibo a mensagem mostrada será *Pelo consumo de 50 Kilowatts * hora a R\$ 0,69 = R\$ 34,50*.

Com relação à exibição dos parâmetros existentes, levou-se em conta que geralmente poucos parâmetros de salários existem, então foi tomado como estratégia de implementação deste sistema que, assim como os cadastros de tipos de insumos, a listagem seria mostrada ao final do menu referente a funcionalidade de inserção. Ainda, pode ser feita facilmente a exclusão de um parâmetro de salário qualquer simplesmente apertando o botão *excluir* intencionalmente colocado à direita de cada parâmetro na listagem.

4.5. CADASTROS E CONTROLE DE DESPESAS

Até o presente momento foram descritas algumas funcionalidades do SOAC relativas ao cadastramentos referentes a insumos, funcionários e parâmetros de salários. No

entanto, embora estes cadastramentos já sejam interessantes para auxiliar o administrador rural a contactar empresas fornecedoras de insumos e os empregados ativos em sua propriedade, essas funcionalidades são muito limitadas perante ao que um administrador rural realmente necessita.

Mas então, o que pode ser útil para um administrador rural além dos fajutos controles acima citados? Uma resposta para isso com certeza seriam funcionalidades que permitam operar despesas. Com os grupos funcionais anteriormente mostrados, é possível fazer um inter-relacionamento e entrelaçamento de dados para permitir uma funcionalidade extra: um sub-sistema de cadastramento de despesas.

Um grupo de cadastramento de despesas aproveitando as funcionalidades já citadas pode auxiliar por exemplo em um controle sobre insumos adquiridos, pagamentos de funcionários com base nos parâmetros de salários da funcionalidade referente a isso e também o cadastramento de outras despesas, para o caso daquelas que não sejam diretamente referentes ao pagamento de funcionários ou aquisição de insumos.

O objetivo desta seção, portanto, é mostrar o grupo de cadastramento de despesas, composto por *Cadastramento de Aquisição de Insumos*, *Cadastramento de Pagamento de Funcionários* e *Cadastramento de Outras Despesas*.

4.5.1. AQUISIÇÕES DE INSUMOS

Essa funcionalidade do grupo de cadastramento de despesas permite ao usuário do sistema efetuar o cadastramento de aquisições de insumos. Mais do que um mero cadastramento, possui uma interface para o ágil acesso aos insumos adquiridos. Será demonstrado nesta subseção detalhes acerca de como isso é realizado.

Na interface representada pela figura 17, nota-se a elaboração de uma das partes

mais custosas da elaboração do sistema, pelos assuntos que neste tópico serão discutidos. Na parte superior desta interface, adotou-se o mesmo padrão utilizado pelo cadastro de funcionários, um *rádio* que permite selecionar a operação desejada relativo à aquisições de insumos. Duas operações aqui são permitidas: a procura e a inserção de aquisições.

The screenshot shows a web application window titled "SOAC - Sistema Online de Auxílio ao Cafeicultor". The main content area is titled "Cadastro de Aquisições" and contains a form with the following sections:

- Operação:** Two radio buttons are present: "Procurar aquisição" (unselected) and "Inserir aquisição" (selected).
- Dados do produto:** Two dropdown menus: "Produto" with the value "ROUND DOWN" and "Loja de Aquisição" with the value "COOCIDADE".
- Dados da aquisição:** Two input fields: "Quantidade adquirida" with the value "1 un." and "Preço por unidade" with the value "R\$ 18,00". Below these is a field for "Dólar hoje" with the value "R\$ 1,80".
- Data da aquisição:** A date picker showing "21" for the day, "11" for the month, and "2007" for the year. Below the date fields are labels: "dia dd", "mês mm", and "ano aaaa".
- Inserir:** A button located at the bottom of the form.

Below the form, there is a green banner with the text: "Passe o mouse sobre uma opção para obter uma descrição aqui da funcionalidade da mesma."

At the bottom of the interface, there is a section titled "Últimas Aquisições:" which is currently empty.

The footer of the application contains the text: "SOAC - Sistema Online de Auxílio ao Cafeicultor", "Desenvolvido por: Daniel Guimarães do Lago", and "Universidade Federal de Lavras".

Figura 17: Cadastro de Aquisições de Insumos visto através de um computador desktop.

Na operação de procura, será possível procurar aquisições de insumos fornecendo

alguns dados específicos. A busca será feita pelo produto e loja especificados. O produto e lojas possíveis estão em um menu *drop-down* cujos dados são obtidos dos cadastramentos previamente realizados em *lojas de comercialização de insumos e insumos*. Caso nenhuma produto ou loja sejam selecionados, os parâmetros não definidos serão ignorados durante a busca. Assim como na funcionalidade referente ao cadastro de funcionários, os resultados retornados da busca serão mostrados no início da tela.

Com a operação de inserção, é possível cadastrar uma inserção fornecendo os seguintes dados: *produto*, *loja de aquisição*, *quantidade adquirida*, *preço por unidade*, *dólar hoje* e *data de aquisição*. O produto se refere ao insumo que foi adquirido; todos os insumos aqui listados são aqueles que foram previamente cadastrados na funcionalidade referente ao cadastramento de insumos. A loja de aquisição refere-se à loja de comercialização de insumos; todos os itens desta listagem foram obtidos da funcionalidade de cadastramento das lojas de comercialização de insumos. A quantidade adquirida e preço por unidade são auto-explicativos. A opção *dólar no dia* foi uma das sugestões dadas por cafeicultores durante a fase da pesquisa. Este dado é muito importante, tendo em vista que grande parte dos insumos o valor de venda do mesmo é fortemente atrelado ao dólar, pelo fato dos que se encaixam nesta categoria possuírem seus preços baseados na moeda americana. Além disso, a *commoditie* café possui seu preço de mercado regido de forma dominante pelas exportações, ou seja, o dólar é um fator de alta influência no preço desta. Com estes dados se é possível fazer estimativas de gastos realizados em valores da moeda americana e, dessa forma, estimar com mais detalhamento e precisão os custos de produção não só na moeda nacional, bem como na moeda estrangeira. A *data de aquisição* deve ser preenchida com a data na qual o insumo foi adquirido; este parâmetro pode ser usado não apenas para dar uma noção do período quando tal insumo foi comprado, bem como ser utilizado pelo sistema de geração de relatórios presente no SOAC o qual será mostrado com maiores detalhes no tópico pertinente a este.

Uma vez que pelo menos um dado tenha sido inserido, ele será adicionado à listagem presente ao final da tela se ele for uma das últimas 20 *aquisições de insumos* em ordem cronológica do campo *data da aquisição* aquisições cadastradas no sistema, como

pode-se visualizar na figura 18.

The screenshot displays a web interface for recording acquisitions. At the top, there are input fields for the exchange rate ('Dólar hoje: R\$ 0,00') and the acquisition date ('Data da aquisição: 21/11/2007'). A note below these fields states: 'Nota: A busca será feita pelo produto e loja selecionados. Caso nenhum seja selecionado, a pesquisa ignorará este parâmetros.' An 'Inserir' button is positioned below the note. A light green banner below the form reads: 'Passe o mouse sobre uma opção para obter uma descrição aqui da funcionalidade da mesma.' Below this banner, a section titled 'Últimas Aquisições:' contains a table with one entry. The entry details include: ID da Aquisição: 1, Local da Compra: COOCIDADE, Produto: ROUND DOWN, Quantia: 1, Preço / Unidade: R\$ 18, Preço Total: R\$ 18, Data: 21/11/2007, and Dólar no Dia: 1.8. A 'Preço em Dólares: 10' is also shown, along with an 'excluir' button. At the bottom of the interface, the text reads: 'SOAC - Sistema Online de Auxílio ao Cafeicultor Desenvolvido por: Daniel Guimarães do Lago Universidade Federal de Lavras'.

Últimas Aquisições:			
ID da Aquisição: 1	Local da Compra: COOCIDADE		
Produto: ROUND DOWN	Quantia: 1	Preço / Unidade: R\$ 18	Preço Total: R\$ 18
Data: 21/11/2007	Dólar no Dia: 1.8	Preço em Dólares: 10	<input type="button" value="excluir"/>

Figura 18: Cadastro de Aquisições de Insumos após inserção da aquisição de 1x Round Down obtido da loja Coocidade, para demonstração do sistema.

Na listagem de aquisições também é possível excluir aquisições realizadas; essa funcionalidade é aplicável a aquisições que eventualmente tenham sido adicionadas acidentalmente ou que por algum motivo não necessitem mais estar nesta listagem. Mais do que isso, essa listagem fornece informações dinâmicas do preço total do produto (em reais) e também em dólares, permitindo ao usuário do sistema fazer a comparação instantânea entre duas listagens.

4.5.2. OUTRAS DESPESAS

Já foi mostrado que o SOAC é, relativo à funcionalidade de cadastro de despesas, capaz de lidar com aquisições de insumos, pagamentos de funcionários e despesas que não se encaixam nas duas anteriores. O objetivo deste recurso é o cadastramento de despesas que certamente um administrador rural vai ter que não são estas citadas.

A figura 19 apresenta uma interface de fácil preenchimento, utilizada pelo SOAC para o cadastramento de despesas que não se encaixam nas outras funcionalidades deste grupo de funcionalidades.

The screenshot shows the SOAC web interface for recording expenses. The title bar reads "SOAC - Sistema Online de Auxílio ao Cafeicultor". The main heading is "Cadastro de Outras Despesas". The interface includes a form with the following fields and options:

- Operação:** Two radio buttons: "Procurar despesa" (unselected) and "Inserir despesa" (selected).
- Dados da despesa:**
 - Descrição:** A text input field containing "Compra de 30 unidades de sacaria".
 - Data:** Three input fields for day, month, and year, containing "21", "11", and "2007" respectively. Below these fields are labels: "dia dd", "mês mm", and "ano aaaa".
 - Valor:** A currency input field showing "R\$ 15,00".
- Inserir:** A green button to submit the form.

Below the form, a message reads: "Passe o mouse sobre uma opção para obter uma descrição aqui da funcionalidade da mesma." Below this is a section titled "Últimas (Outras) Despesas Cadastradas:" with the message "Nenhuma despesa está cadastrada atualmente." At the bottom, the footer text reads: "SOAC - Sistema Online de Auxílio ao Cafeicultor", "Desenvolvido por: Daniel Guimarães do Lago", and "Universidade Federal de Lavras".

Figura 19: Cadastro de Outras Despesas visto através de um computador desktop.

Como se pode ver, esta tela herda um pouco algumas das idéias implementadas em outras telas do sistema, como seleção de operação entre procura e inserção através de um *rádio*, numa mesma tela alterada dinamicamente de acordo com a seleção do usuário.

Pela operação *procurar* do sistema, é possível fazer uma varredura em todas as despesas cadastradas nessa funcionalidade do sistema. Para isso, deve-se preencher no campo *data* a data à partir da qual a despesa foi cadastrada e, no campo *valor*, o teto inferior que será utilizado na busca.

Descrição:

Data:
dia dd mês mm ano aaaa

Valor: R\$,

Nota:
A busca procurará todas as despesas com pelo menos o valor especificado E a partir da data especificada.

Procurar

Passa o mouse sobre uma opção para obter uma descrição aqui da funcionalidade da mesma.

Últimas (Outras) Despesas Cadastradas:

ID: 1	Data: 21/11/2007
Descrição: COMPRA DE 30 UNIDADES DE SACARIA.	
Valor: R\$ 15	excluir

SOAC - Sistema Online de Auxílio ao Cafeicultor
Desenvolvido por: Daniel Guimarães do Lago
Universidade Federal de Lavras

Figura 20: Cadastro de Outras Despesas após inserção de uma despesa.

Na operação *inserir*, como o próprio nome intuitivamente mostra, permite inserir

uma “outra” despesa com os dados informados nos campos de entrada. No campo *descrição*, uma descrição detalhada da despesa deve ser fornecida, e os demais campos preenchidos intuitivamente conforme o que eles pedem: *data* representa a data na qual essa despesa ocorreu e *valor* representa o custo total da despesa.

Após inseridas, assim como na funcionalidade de cadastro de aquisições, as últimas despesas cadastradas serão mostradas em uma listagem na mesma tela de cadastro, contendo informações sumarizadas sobre cada uma delas. Para exemplificar esta listagem, imagine que como nas imagens anteriormente mostradas foi inserido uma despesa definida como “*Compra de 30 unidades de sacaria a R\$ 15,00*”. A figura 20 demonstra esta despesa vista do SOAC.

Nessa tela a opção para excluir elementos da listagem também é facilmente acessível, na própria listagem, para maximizar a usabilidade do sistema.

4.5.3. PAGAMENTOS DE FUNCIONÁRIOS

Essa é uma avançada funcionalidade do SOAC, que permite o pagamento de funcionários com base em parâmetros salariais definidos previamente pelo próprio administrador rural. Nela, o usuário do sistema poderá efetuar o pagamento de qualquer funcionário recebendo uma estrutura atraente, conforme pode ser vista na figura 21.

Esta interface é construída de acordo com os parâmetros salariais cadastrados no sistema. Na parte superior da mesma, possuímos a opção de *operação*, que permite definir se vamos efetuar uma consulta (*rádio procurar*) ou uma inserção (*rádio inserir*).

Logo abaixo da operação desejada, é pedido os dados do funcionário. Neste caso, apenas o nome do funcionário é requerido. O nome de todos os funcionários aptos a receber pagamentos (*ativos*) estão nessa listagem, no formato de um menu *drop-down*, e o

usuário pode digitar parcialmente o nome do funcionário para encontrar o nome do mesmo mais rapidamente. Também há um botão logo abaixo para listar todos os funcionários, *ativos e inativos*, para caso o usuário deseje efetuar o pagamento de um funcionário que não esteja em atividade.

Dados do pagamento:	Quantia	Infixo	Valor	Sufixo Total
Prefixo				
PELA COLHEITA DE	16	MEDIDAS DE CAFE A R\$	4,50	72
PELA COLHEITA DE	20	MEDIDAS DE CAFE A R\$	5,50	110
PELA COLHEITA DE	18	MEDIDAS DE CAFE A R\$	7,00	126
PELAS	0	HORAS EXTRAS A R\$	0,00	0.00
PELAS EMPREITADAS: CARPAS DE	0	CCF A R\$	0,00	0.00
PELO(S)	0	MES(ES) A R\$	0,00	0.00
PELOS	0	DIAS DOMINGOS E FERIADOS A R\$	0,00	0.00
PELOS	15	DIAS TRABALHADOS A R\$	30,00	450
SALARIO FAMILIA	1	R\$	30,00	30
ADIANTAMENTOS NESSE MES	0	R\$	0,00	0.00
HABITACAO	0	R\$	0,00	0.00
INSS	1	R\$	6,40	-6.4
TOTAL:				781.6

Figura 21: Cadastro de Pagamentos de Funcionários visto através de um computador desktop.

Após os campos que requerem dados do funcionário, existem mais dois campos. Um campo é o campo *referente a*, o qual basicamente se o usuário desejar inserir um pagamento poderá informar ao que este pagamento se refere, como por exemplo *Pagamento do Mês de Outubro*. Visando a máxima usabilidade, por padrão este campo já vem preenchido com a última referência a qual o mesmo aludiu.

Dados do pagamento:					
Prefixo	Quantia	Infixo	Valor		Sufixo Total
PELA COLHEITA DE	16	MEDIDAS DE CAFE A R\$	4	50	72
PELA COLHEITA DE	20	MEDIDAS DE CAFE A R\$	5	50	110
PELA COLHEITA DE	18	MEDIDAS DE CAFE A R\$	7	00	126
PELAS	0	HORAS EXTRAS A R\$	0	00	0.00
PELAS EMPREITADAS: CARPAS DE	0	CCF A R\$	0	00	0.00
PELO(S)	0	MES(ES) A R\$	0	00	0.00
PELOS	0	DIAS DOMINGOS E FERIADOS A R\$	0	00	0.00
PELOS	15	DIAS TRABALHADOS A R\$	30	00	450
SALARIO FAMILIA	1	R\$	30	00	30
ADIANTAMENTOS NESSE MES	0	R\$	0	00	0.00
HABITACAO	0	R\$	0	00	0.00
INSS	1	R\$	6	40	-6.4
TOTAL:					781.6

Figura 22: Cadastro de Pagamentos de Funcionários visto através de um computador desktop: Foco nos dados do pagamento

Por último, nesse primeiro agrupamento de campos, o campo *data do pagamento* é pedida para que o usuário o complete com a data na qual este pagamento foi efetuado. Por questões de usabilidade também a mesma já vem preenchida por padrão com a data atual do sistema.

Após esse primeiro agrupamento de campos, o usuário preencherá os campos previamente cadastrados como parâmetros de pagamentos salariais. Seguindo a idéia, quem está utilizando o sistema deve preencher basicamente 2 dados: *quantia* e *valor*. Isso foi projetado dessa forma para a máxima adaptabilidade possível do sistema. Num exemplo prático, veja a figura 22 para se ter uma noção do que quer dizer estes dados *quantia* e *valor*. Quantia diz respeito ao total de vezes que um parâmetro salarial será multiplicado; valor se refere ao valor base deste salário. Por exemplo, ao informar que um determinado funcionário colheu 3 medidas de café à R\$ 4,50, 3 será a quantia e 4,50 será o valor, o total será o R\$ 13,50 que é o resultado da multiplicação da quantia pelo valor.

Como visto em tópicos anteriores, a entrada padrão para dados de pagamentos de

funcionários consiste na seguinte idéia: *prefixo quantia infixo valor sufixo*, onde *prefixo*, *infixo* e *sufixo* são dados fixos fornecidos pelos parâmetros salariais. *Quantia* e *valor* são dados dinâmicos preenchidos durante o pagamento de funcionários. À medida em que estes parâmetros salariais forem sendo modificados durante o preenchimento dos campos deste grupo dos pagamentos de funcionários, um cálculo automático do valor total do parâmetro do salário e do valor total do pagamento do funcionário é mostrado na tela para maior entendimento e controle do que está sendo preenchido. O total de cada parâmetro salarial é o resultado matemático da *quantia* multiplicada pelo *valor* do que estiver sendo preenchido neste exato momento. A alteração dinâmica do valor total mostrado na tela é possível graças ao emprego de *javascript*.

Últimos pagamentos (ordem decrescente):

Código do Pagamento: 1 Data do Pagamento: 21/11/2007
 Pago a: JOAO DA SILVA
 Referente a: PAGAMENTO DO MES DE OUTUBRO

Prefixo	Quantia	Infixo	Valor	Sufixo	Total
PELA COLHEITA DE	16	MEDIDAS DE CAFE A R\$	4.5		72
PELA COLHEITA DE	20	MEDIDAS DE CAFE A R\$	5.5		110
PELA COLHEITA DE	18	MEDIDAS DE CAFE A R\$	7		126
PELOS	15	DIAS TRABALHADOS A R\$	30		450
SALARIO FAMILIA	1	R\$	30		30
INSS	1	R\$	6.4		-6.4
TOTAL:					781.6

[excluir](#)

SOAC - Sistema Online de Auxílio ao Cafeicultor
 Desenvolvido por: Daniel Guimarães do Lago
 Universidade Federal de Lavras

Figura 23: Cadastro de Pagamentos de Funcionários visto através de um computador desktop após a inserção de um pagamento, foco na listagem dos pagamentos sumarizada.

Para a operação de inserção, todos os grupos de parâmetros devem ser preenchidos pelo usuário e após isso o botão *inserir* deve ser pressionado. Uma vez que estes dados são inseridos eles estarão presentes no banco de dados para possíveis consultas e futura geração de recibos. Os últimos pagamentos em ordem decrescente também são mostrados de forma sumarizada na tela, como se pode ver na figura 23.

Como visto na tela mostrada, assim como nas listagens de diversas outras partes do sistema, é possível se excluir facilmente os itens (pagamentos efetuados) dessa listagem.

Com relação a operação de *procura*, foi definido que a forma mais fácil para o usuário obter informações de pagamentos de funcionários seria fornecendo apenas o nome do mesmo e o sistema retornando todas os pagamentos efetuados ao próprio. Para efetuar uma consulta o usuário do sistema necessita apenas fornecer o nome do funcionário na listagem *drop-down* para isso designada e pedir para o sistema procurar que os resultados serão retornados conforme este parâmetro.

4.6. CADASTRO DE RECEITAS

Foi identificado como o SOAC trabalha para auxiliar o administrador rural com relação à despesas. Mas, para um sistema que efetue este tipo de controle financeiro não deve ser constituído apenas de controle de gastos: o controle de ganhos também deve ser levado em conta. Por isso houve uma preocupação durante o desenvolvimento do SOAC com um grupo de funcionalidades de cadastramento de receitas.

O grupo de cadastro de receitas é composto por 3 funcionalidades: *cadastro de produtos*, *cadastro de venda de produtos* e *cadastro de outras receitas*; as quais serão abordados afundo nos tópicos a seguir.

4.6.1. PRODUTOS

O cadastramento de produtos é uma funcionalidade simples que age em cooperação com o cadastramento de vendas de produtos. O objetivo desta funcionalidade é reconhecer o que pode ser vendido para obtenção de receita. A figura 24 demonstra o objetivo deste cadastramento.



The screenshot shows a web interface for the SOAC system. At the top, there is a header with a coffee cup icon and the text "SOAC - Sistema Online de Auxílio ao Cafeicultor". Below this, the word "Produtos" is centered. The main area is titled "Incluir novo produto:". There is a text input field labeled "Nome do produto:" with the example "ex: Café" and the word "Café" entered. Below the input field is a green button labeled "Inserir Produto". A horizontal line separates this section from the next, which contains the text "Nenhum produto está cadastrado atualmente." At the bottom, there is a footer with the text "SOAC - Sistema Online de Auxílio ao Cafeicultor", "Desenvolvido por: Daniel Guimarães do Lago", and "Universidade Federal de Lavras".

Figura 24: Cadastro de Produtos visto através de um computador desktop.

Essa funcionalidade é bastante simples e apenas armazena um único dado sobre produtos que podem ser vendidos: o nome. Uma vez cadastrado os produtos desejados que podem ser vendidos, todos eles aparecem em uma listagem ao final da tela, como outras partes do sistema também fazem; conforme mostrado na figura 25.



Figura 25: Cadastro de Produtos visto através de um computador desktop após inserção do produto café.

4.6.2. VENDAS DE PRODUTOS

Uma vez cadastrado os produtos que podem ser vendidos, já pode utilizar o grupo de funcionalidades de receitas para realização de operações de cadastros com receitas propriamente dito.

Segundo o que foi levantado durante as entrevistas e também na modelagem do sistema, os parâmetros mais necessários para as vendas são o *produto* vendido (obviamente), uma *descrição* da venda, o *valor* total da venda, o valor do *dólar* no dia da venda e também a data da realização da venda, e assim foi feito, como mostra a figura 26.

 SOAC - Sistema Online de Auxílio ao Cafeicultor

Cadastro de Vendas

Operação:
 Procurar venda Inserir venda

Dados do produto:
Produto:

Dados da venda:
Descrição da venda:

Valor da Venda: , Dólar hoje: ,

Data da venda:

dia dd mês mm ano aaaa

Passe o mouse sobre uma opção para obter uma descrição aqui da funcionalidade da mesma.

Últimas Vendas:

SOAC - Sistema Online de Auxílio ao Cafeicultor
Desenvolvido por: Daniel Guimarães do Lago
Universidade Federal de Lavras

Figura 26: Cadastro de Vendas de Produtos visto através de um computador desktop.

Nessa tela temos o estilo semelhante ao de outras interfaces mostradas anteriormente: duas *operações* que o usuário pode escolher; *procurar* e *inserir* vendas; dados do *produto* que será vendido, neste caso, o nome; e dados da venda, com uma *descrição da venda*, o *valor da venda*, o valor do *dólar* no dia da venda e também a *data* da venda.

No caso de a opção de *operação* estar definida como sendo *procura*, ela verificará se foi fornecido um produto ao sistema para efetuar a pesquisa e, em caso positivo essa pesquisa será feita com este parâmetro. Caso contrário ela ignorará todos os parâmetros e retornará todas as vendas de todos os produtos.

R\$ 0,00 R\$ 0,00

Data da venda:

21 11 2007
dia dd mês mm ano aaaa

Nota:
A busca será feita pelo produto selecionado.
Caso nenhum seja selecionado, a pesquisa ignorará este parâmetros.

Inserir

Passa o mouse sobre uma opção para obter uma descrição aqui da funcionalidade da mesma.

Últimas Vendas:

ID da Venda: 1	Descrição: 30 SACOS DE CAFE VENDIDOS PARA A COOPERATIVA COOCIDADE.		
Produto: CAFE	Valor da Venda: 6000	Valor do Dólar: R\$ 1.8	Data: R\$ 21/11/2007

excluir

SOAC - Sistema Online de Auxílio ao Cafeicultor
Desenvolvido por: Daniel Guimarães do Lago
Universidade Federal de Lavras

Figura 27: Cadastro de Vendas visto através de um computador desktop, após inserção de uma venda.

Se a *operação* tiver sido definida pra inserir, ela lerá as informações prestadas pelo usuário e de forma semelhante às inserções do grupo de cadastros de despesas realizará a inserção do produto. Para exemplificar isso será simulada a inserção de uma venda de 30

sacos do *produto* café vendidos à *loja* Coocidade pelo *preço* total de R\$ 6.000,00 com o *dólar* no dia valendo R\$ 1,80. A listagem dos produtos vendidos será mostrada ao final da tela, e também com o acesso ao botão excluir facilitado, conforme a figura 27.

4.6.3. OUTRAS RECEITAS

Semelhante ao grupo de cadastro de outras despesas, foi necessário incluir no sistema um cadastro para outras receitas para efetuar o cadastramento de receitas que nenhuma outra parte do SOAC contemple, ou seja, neste caso que não seja receita proveniente de vendas de produtos.

The screenshot shows a web application interface for recording receipts. At the top, there is a header with a coffee cup icon and the text "SOAC - Sistema Online de Auxílio ao Cafeicultor". Below this is a sub-header "Cadastro de Outras Receitas". The main form area is titled "Operação:" and contains two radio buttons: "Procurar receita" (unselected) and "Inserir receita" (selected). Below the operation selection is a section titled "Dados da receita:" containing three fields: "Descrição:" with the text "Venda de 3 bois de 15 arrobas para João Silvestre.", "Data:" with a date picker showing "21" for the day, "11" for the month, and "2007" for the year, and "Valor:" with a currency field showing "R\$ 1400,00". A green "Inserir" button is located below the form. Below the form is a message: "Passe o mouse sobre uma opção para obter uma descrição aqui da funcionalidade da mesma." At the bottom of the form area, there is a section titled "Últimas (Outras) Receitas Cadastradas:" with the message "Nenhuma receita está cadastrada atualmente." The footer of the page contains the text: "SOAC - Sistema Online de Auxílio ao Cafeicultor", "Desenvolvido por: Daniel Guimarães do Lago", and "Universidade Federal de Lavras".

Figura 28: Cadastro de Receitas visto através de um computador desktop.

Não há muito o que descrever sobre o sistema de cadastramento de Outras Receitas, tendo em vista que ele é praticamente igual ao sistema de cadastramento de despesas, com a única diferença que ao invés de seus dados serem de gastos do administrador rural eles serão de ganhos. Na figura 28 está demonstrado como é feito este cadastramento.

Relacionado ao sistema propriamente dito, visualmente ele se comportará de forma semelhante ao cadastramento mostrado no tópico *Cadastro de Outras Despesas*.

4.7. GERAÇÃO DE RELATÓRIOS

Foram vistas todas as áreas principais relativas ao fluxo financeiro que o Sistema Online de Auxílio ao Cafeicultor provê controle. Embora já sejam partes suficientes para tornar o SOAC um sistema básico de auxílio à administração cafeeira na parte referente ao controle de gastos, não seriam muito úteis se faltassem duas funcionalidades.

A primeira funcionalidade a qual é interessante que um administrador rural tenha acesso, conforme notado na fase das entrevistas, seria um sistema para geração automatizada de recibos após o cadastro de todos os pagamentos de funcionários. A segunda funcionalidade seria a geração simplificada de relatório, que permitisse a impressão completa de todos os custos e receitas do sistema que estejam em um determinado padrão.

O objetivo desta seção será, portanto, abordar o grupo de geração de relatórios e suas funcionalidades.

4.7.1. RECIBOS

Nessa parte do SOAC foi implementado um sistema para geração de recibos

referentes aos pagamentos de administradores rurais efetuados no *cadastramento de pagamentos de funcionários*.

A idéia dessa funcionalidade é que, uma vez que tenham sido cadastrados todos os pagamentos de funcionários referentes a um mesmo padrão de pagamento (ex: *pagamento referente ao mês de outubro de 2007*), o administrador rural possa imprimir todos os recibos desta referência, ou de um grupo de referências recentemente inseridas, de uma vez só. Na figura 29 está uma ilustração que mostra esta parte do sistema.

Como se pode observar, essa é uma tela bastante simples que realiza essa funcionalidade. Nessa tela temos alguns parâmetros que devem ser preenchidos pelo administrador rural para a geração dos recibos desejados. Para gerar um conjunto de recibos, o usuário deverá entrar com o número de identificação do primeiro recibo que se deseja gerar o relatório e, também, com o número de identificação do último recibo para esta geração. Estes números de identificação são gerados e mostrados ao usuário no momento em que o mesmo cadastra um pagamento de funcionário recente ou na listagem ao final, que será detalhado ainda neste tópico. Tendo preenchido o número de identificação do recibo inicial e do último recibo, ao efetuar a geração de recibos todos os pagamentos com número de identificação contemplados neste intervalo serão gerados.

Uma vez selecionado o grupo de recibos que deva ser gerado, o usuário passa a ter que preencher dados da empresa. Nestes dados deverá colocar o nome da empresa responsável pela emissão dos recibos. Além disso, deverá ser preenchido também o nome do empregador responsável pelos pagamentos. Estes dados ficarão no cabeçalho do recibo que será mostrado mais a frente. Para máxima usabilidade, o sistema cadastra e preenche automaticamente os dados de *nome da empresa* e *empregador* com os últimos dados que foram utilizados para preencher estes campos.

Especificado o intervalo que deverá ser impresso, a empresa e o empregador responsável, deve-se especificar a data de impressão do recibo. O usuário tem duas opções: pedir para que, durante a geração do recibo, seja obtido do campo *data* do pagamento de

funcionários, ou pode pedir para especificar uma data para a geração dos recibos.

SOAC - Sistema Online de Auxílio ao Cafeicultor

Geração de Recibos

Gerar recibos para códigos (IDs) com base nos seguintes parâmetros:

Intervalo de geração:
ID Início: ex: 4 ID Fim: ex: 32

1 1

Dados da empresa:
Empregador: **NOME DO EMPREGADOR**
Empresa: **NOME DA EMPRESA**

Especificar data:
 Utilizar data do pagamento Utilizar data deste formulário

Data: **21** **11** **2007**
dia dd mês mm ano aaaa

Gerar Recibos

Passe o mouse sobre uma opção para obter uma descrição aqui da funcionalidade da mesma.

Últimos 200 pagamentos (ordem decrescente):

Código do Pagamento (ID): 1 Data do Pagamento: 21/11/2007 Valor: R\$ 781.6
Pago a: JOAO DA SILVA
Referente a: PAGAMENTO DO MES DE OUTUBRO

marcar inicio **marcar fim**

SOAC - Sistema Online de Auxílio ao Cafeicultor
Desenvolvido por: Daniel Guimarães do Lago
Universidade Federal de Lavras

Figura 29: Geração de Recibos visto de um computador desktop.

Uma vez preenchido todos os dados requeridos o usuário do sistema poderá efetuar a geração do relatório simplesmente clicando no botão *gerar recibos*. Um detalhe ainda mencionado superficialmente aqui é de onde o usuário vai efetivamente poder obter o número de identificação dos recibos a serem gerados. Uma opção é obter esse número através da própria funcionalidade que permite o *cadastro de pagamento de funcionários*, através da *operação* procura. Esse é um procedimento simples, mas para aperfeiçoar a usabilidade do sistema foi inserida uma listagem com os últimos 200 pagamentos de funcionários cadastrados, com dados sumarizados relativos à cada paramento e com duas opções possíveis para cada um deles, se assim o usuário desejar: um botão *marcar início* e outro botão *marcar fim*. Com estes botões, o usuário pode selecionar um grupo de recibos a serem gerados sem a necessidade de ficar efetuando consultas em outra funcionalidade, base ele clicar nestes botões que o número de identificação dos recibos gerados serão automaticamente preenchidos com as opções desejadas.

Após todos os parâmetros terem sido preenchidos, o usuário poderá efetuar a geração do grupo de recibos por ele especificada. Na tela, todos os recibos são mostrados como sendo uma pré-visualização do que será impresso, como se pode observar na ilustração 30, onde se nota que o estilo do recibo conta com um cabeçalho que possui o nome do empregador e da empresa, um código do recibo para facilitar posterior consulta ao banco de dados, a data de pagamento que foi especificada segundo parâmetros desejados pelo próprio usuário, o empregado a quem este pagamento foi realizado e ao que este pagamento se refere.

No corpo do recibo, existe uma listagem com descrição e valores cadastrados no pagamento de funcionários. Cada descrição baseia-se na regra já citada de *infixo quantia prefixo valor sufixo* (ex: *Pela colheita de 16 medidas de café a R\$ 4,50*) e o valor total da operação previamente mostrada. Para não abarrotar o preenchimento do recibo com dados inúteis, aqueles que tiverem valor 0 foram descartados. As operações de subtração do salário, como por exemplo o pagamento da energia elétrica consumida pelo funcionário em sua residência na propriedade rural caso o administrador a decida abater, são feitas automaticamente. Na última linha do corpo é impresso o valor final que o administrador

rural pagou ao funcionário.

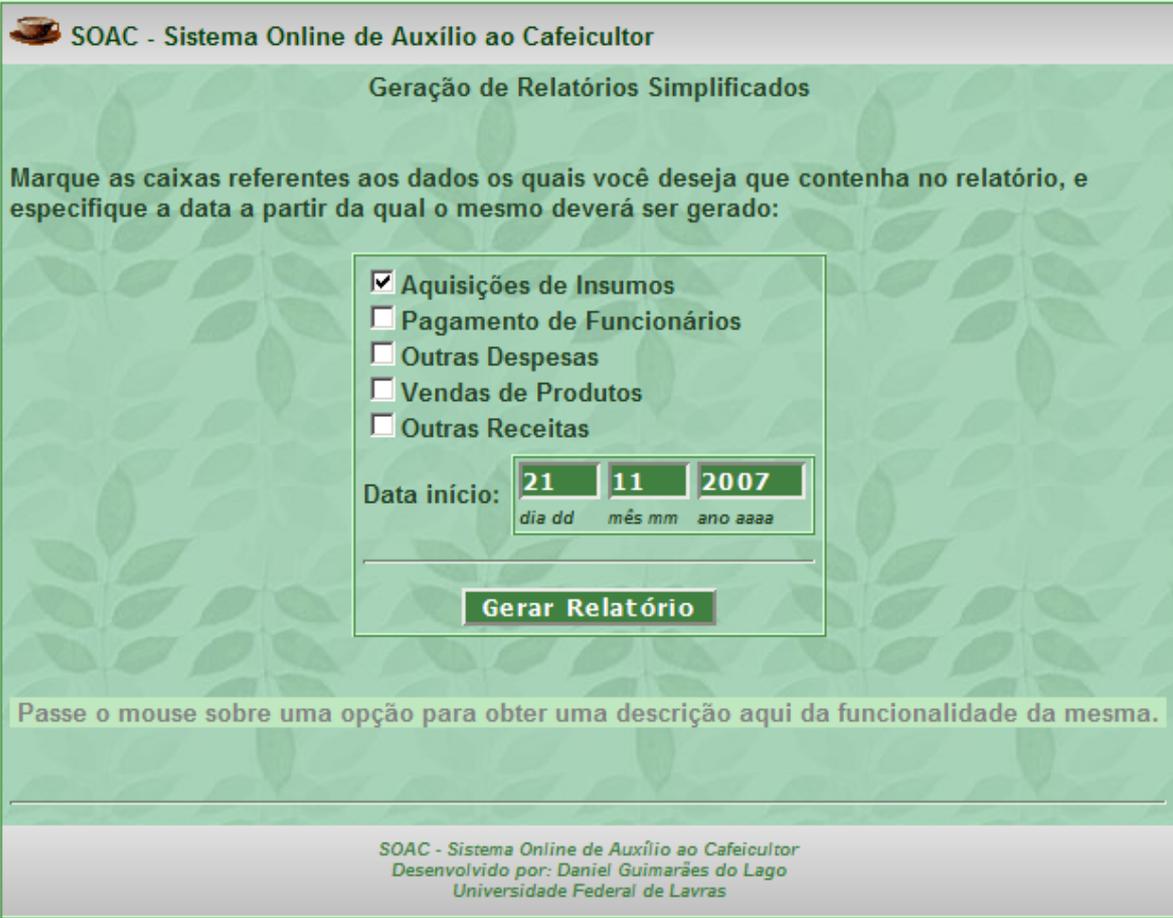
RECIBO DE SALÁRIO DO TRABALHADOR RURAL	
NOME DA EMPRESA	
Código (ID): 1	Data do Pagamento: 21/11/2007
Empregador: NOME DO EMPREGADOR	
Pago a: JOAO DA SILVA	
Referente a: PAGAMENTO DO MES DE OUTUBRO	
<i>Descrição</i>	<i>Valor (RS)</i>
PELA COLHEITA DE 16 MEDIDAS DE CAFE A R\$ 4,50	72,00
PELA COLHEITA DE 20 MEDIDAS DE CAFE A R\$ 5,50	110,00
PELA COLHEITA DE 18 MEDIDAS DE CAFE A R\$ 7,00	126,00
PELOS 15 DIAS TRABALHADOS A R\$ 30,00	450,00
SALARIO FAMILIA 1 R\$ 30,00	30,00
INSS 1 R\$ 6,40	-6,40
TOTAL:	781,60
21 de Novembro de 2007	
_____ JOAO DA SILVA	

Figura 30: Recibos Gerados visto de um computador desktop.

Ao final do documento, a data em formato extenso é preenchida, juntamente com o espaço para o funcionário assinar o recibo com mais duas testemunhas.

Embora na tela todos os recibos sejam mostrados seqüencialmente, quando o documento for enviado para a impressos, por motivos de organização e controle o SOAC preocupou-se em imprimir apenas um recibo por folha.

4.7.2. RELATÓRIOS SIMPLIFICADOS



The screenshot shows a web interface for the SOAC system. At the top, there is a header with a coffee cup icon and the text "SOAC - Sistema Online de Auxílio ao Cafeicultor". Below this, the main heading is "Geração de Relatórios Simplificados". The interface prompts the user to "Marque as caixas referentes aos dados os quais você deseja que contenha no relatório, e especifique a data a partir da qual o mesmo deverá ser gerado:". A central form contains five checkboxes: "Aquisições de Insumos" (checked), "Pagamento de Funcionários", "Outras Despesas", "Vendas de Produtos", and "Outras Receitas". Below the checkboxes is a date selection field labeled "Data início:" with three input boxes containing "21", "11", and "2007". Underneath these boxes are the labels "dia dd", "mês mm", and "ano aaaa". A green button labeled "Gerar Relatório" is positioned below the date field. At the bottom of the form area, there is a note: "Passe o mouse sobre uma opção para obter uma descrição aqui da funcionalidade da mesma." The footer of the page contains the text: "SOAC - Sistema Online de Auxílio ao Cafeicultor", "Desenvolvido por: Daniel Guimarães do Lago", and "Universidade Federal de Lavras".

Figura 31: Geração de Relatórios Simplificados visto de um computador desktop.

Uma outra funcionalidade do SOAC relativa a geração de relatórios, além da geração de recibos, é a geração de relatórios simplificados. O objetivo desta importante funcionalidade é a impressão de relatórios sumarizados de acordo com o que o usuário

requerer que o sistema faça. Ao acessar esta funcionalidade, será exibido ao usuário uma tela como ilustrada pela figura 31.

Este é um recurso que entrelaça com todos os outros recursos do sistema relativos ao controle financeiro. Para gerar um relatório sumarizado, o usuário deverá marcar as caixas que deseja que sejam mostradas no relatório, quantas caixas quiser. Após isso ele deve especificar a *data de início*.

Ao clicar no botão *Gerar Relatório* o sistema vai fazer uma varredura em todo o seu banco de dados e, com base nas caixas marcadas, gerar um relatório de todas as operações financeiras que foram iguais ou posteriores à *data de início* especificada no sistema. Uma vez gerado, ao usuário final um relatório semelhante ao seguinte, com as opções desejadas, será mostrado ao usuário como na figura 32.

SOAC - Sistema Online de Auxílio ao Caficultor		
Relatório gerado em 21 de Novembro de 2007		
<i>Despesas</i>		
Aquisições de Insumos		
<i>Data</i>	<i>Descrição</i>	<i>Valor (R\$)</i>
21/11/2007	1 x ROUND DOWN adquirido de COOCIDADE a R\$ 18,00 cada unidade (Dolar no dia: US\$ 1,80).	R\$ 18,00
TOTAL		R\$ 18,00

Figura 32: Relatórios Simplificados gerados visto de um computador desktop.

Essa tela após o processamento foi otimizada para impressão e facilitado o entendimento. O estilo dos relatórios gerados possuem no cabeçalho a data na qual foram gerados, um título para a seção *Despesas* (se aplicável) e um título para a seção *Receitas* (se aplicável). Dentro do relatório, propriamente dito, é impresso os campos com a data, descrição e valor de cada operação realizada e, ao final, o total por cada seção marcada na tela de geração de relatórios simplificados.

Como se pode perceber, esse é um relatório sumarizado que dá uma visão geral de todas as operações do sistema em datas desejadas pelo administrador rural e, portanto, é de um auxílio inestimável para o usuário.

4.8. UTILITÁRIOS

Conforme demonstrado até o presente momento, as principais funcionalidades do SOAC, para computadores pessoais e *handhelds*, que buscam auxiliar a cadeia produtiva da administração rural focada no café estão presentes como sendo suas funcionalidades principais.

No entanto, algumas funcionalidades extras que possam aproveitar da mobilidade projetada no sistema seriam bastante úteis. Ilustra-se a situação da seguinte forma: um administrador rural possui, uma agenda de tarefas que devem ser feitas, sendo estas organizadas em um arquivo no computador. Por algum motivo o mesmo viaja e necessita obter os compromissos agendados, porém este esqueceu de consultar o computador ou de passar os dados por exemplo para um *notebook*. É bastante razoável que se neste computador tiver seus compromissos agendados que ele, utilizando-se dos princípios da computação pervasiva, consiga acessar estes dados de qualquer lugar. Este exemplo demonstra o que seria um utilitário interessante.

Nesta seção aborda-se os utilitários que estão presentes no SOAC, constituídos pela *agenda, lista telefônica e alterações de dados de login*.

4.8.1. AGENDA

No SOAC, aproveitando seu foco em relação à mobilidade e conforme exemplificado no item anterior, é plausível que fosse desenvolvido um sistema que possuísse uma agenda acessível de qualquer lugar.

Uma funcionalidade deste tipo não só é plausível como é bastante útil – realmente um utilitário – para o administrador rural. A figura 33 apresenta o módulo da agenda que está integrado ao SOAC. Nesta interface se é possível cadastrar novos compromissos, especificando qual *Atividade* o administrador deseja agendar, bem como o dia e horário do início e fim da atividade.

Uma vez cadastrado que compromissos tenham sido cadastrados, estes serão armazenados em um banco de dados e, faltando 3 dias para o compromisso agendado, ele começará a aparecer na listagem presente nesta interface dos próximos compromissos. Essa listagem é mostrada em ordem cronológica de data dos compromissos agendados e, também se é possível marcar estes como concluídos ou excluí-los. Nessa listagem também são exibidos os compromissos atrasados que ainda não foram concluídos, como forma de alertar o usuário do sistema que existem atividades pendentes.

Além destas funcionalidades, a agenda também conta com uma listagem dos compromissos mais recentemente realizados para auxiliar o usuário do sistema a se situar no tempo.

 SOAC - Sistema Online de Auxílio ao Cafeicultor

Agenda

Incluir novo compromisso:

Atividade:

Início:
hora hh minuto mm dia dd mês mm ano aaaa

Fim:
hora hh minuto mm dia dd mês mm ano aaaa

Passe o mouse sobre uma opção para obter uma descrição aqui da funcionalidade da mesma.

Compromissos atrasados:

Data/Hora Início	Data/Hora Fim	Atividade	Concluir	Excluir
21/11/2007 21:16:00	21/11/2007 21:16:00	PREPARAR IMAGENS PARA APARESENTACAO DESTE SISTEMA EM ACAA.	<input type="button" value="concluído"/>	<input type="button" value="excluir"/>

Compromissos dos próximos 3 dias:

Data/Hora Início	Data/Hora Fim	Atividade	Concluir	Excluir

Últimos compromissos realizados:

Data/Hora Início	Data/Hora Fim	Atividade

SOAC - Sistema Online de Auxílio ao Cafeicultor
 Desenvolvido por: Daniel Guimarães do Lago
 Universidade Federal de Lavras

Figura 33: Utilitário Agenda visto de um computador desktop.

4.8.2. LISTA TELEFÔNICA

Outro utilitário que foi detectado como sendo importante para administradores rurais, se aproveitando da mobilidade dada pelo SOAC, foi a lista telefônica. Tendo em vista o enorme número de contatos que administradores rurais normalmente têm, uma lista telefônica é fundamental para tais pessoas. Aproveitando um sistema unificado de recursos como o SOAC então foi designado então como sendo interessante ter uma lista telefônica para suprir as necessidades deste setor.

SOAC - Sistema Online de Auxílio ao Cafeicultor

Lista Telefônica

Operação Desejada:

Procurar Contato Inserir Contato Editar Contato Apagar Contato

Procurar Contato

Nome:
Ex1: Coaxupé
Ex2: Joao

Cidade:
Ex1: Guaxupé

Relação:
Ex1: comercial

Procurar

Passe o mouse sobre uma opção para obter uma descrição aqui da funcionalidade da mesma.

SOAC - Sistema Online de Auxílio ao Cafeicultor
Desenvolvido por: Daniel Guimarães do Lago
Universidade Federal de Lavras

Figura 34: Utilitário Lista Telefônica visto de um computador desktop.

Durante a implementação da lista telefônica no SOAC houve forte preocupação com a usabilidade da mesma. Tendo em vista o enorme número de telefones

potencialmente cadastráveis nesta, foi desenvolvida uma interface toda especial para a mesma, como se pode ver na figura 34.

Essa tela de lista telefônica busca ser a mais simplificada e dinâmica possível para sua utilização. Quando um *rádio* de uma *operação desejada* é selecionado, toda a tela é dinamicamente modificada através de *javascript* para se adaptar a opção desejada.

Número do Registro: 2	
Nome completo: Cooxupe	
Relação com o contato: Comercial	
Cidade: Guaxupe	
Telefone 1: 3512345678	Telefone 2:
Celular 1:	Celular 2:
Fax 1:	Fax 2:

SOAC - Sistema Online de Auxílio ao Cafeicultor
Desenvolvido por: Daniel Guimarães do Lago
Universidade Federal de Lavras

Figura 35: Utilitário Lista Telefônica após ter efetuado uma procura.

Na figura 34 como exemplo, a operação *procurar* foi selecionada e os dados pertinentes a mesma para serem preenchidos está sendo mostrado ao usuário do sistema. As opções permitidas em uma lista telefônica são as operações de *procura*, *inserção*,

edição e *exclusão*. Neste tópico será abordado com detalhes a capacidade de cada uma dessas funcionalidades.

A operação de *inserção* permite que o usuário do sistema insira um novo contato. Os parâmetros utilizados para se inserir um contato são o nome do contato, a cidade, o tipo de relacionamento com o mesmo, dois telefones, dois celulares e dois faxes. Ainda nesta seção será demonstrado um exemplo de inserção realizada e o resultado da busca realizada no sistema para o usuário.

A operação de *procura*, o usuário preenche os campos nome, cidade e relacionamento para efetuar a busca. Os campos não preenchidos serão ignorados durante a varredura do banco de dados. Após efetuar uma procura qualquer, os dados são retornados para o usuário como exemplo na figura 35.

A operação *editar* permite editar um dado existente na lista. Para editar o dado, primeiro é necessário fornecer o número de identificação gravado da entrada na lista, que pode ser obtido normalmente pela operação de *procura*.

A operação *apagar* apaga uma entrada da lista telefônica com os dados informados. Assim como a operação *editar* o usuário do sistema deve fornecer o código de identificação da entrada para apagá-la do sistema.

4.8.3. ALTERAÇÕES DE DADOS DE LOGIN

Este é talvez o utilitário mais simples de todo o sistema SOAC. Através desta funcionalidade o usuário poderá alterar o nome do *login* e a senha para acesso ao sistema via computadores pessoais ou *handhelds*. A figura 36 ilustra esta funcionalidade.

Nessa tela o usuário deverá informar qual é o *login* para acessar o sistema, uma

senha, confirmar a senha digitada, uma senha para acesso via celular e confirmar também a digitação da senha para acesso via telefone celular.

SOAC - Sistema Online de Auxílio ao Cafeicultor

Alterar Dados de Login:

Entre com os novos dados:

Login:

Senha (Site): #

Confirmar Senha (Site): #

Senha (Celular): #

Confirmar Senha (Celular): #

Passa o mouse sobre uma opção para obter uma descrição aqui da funcionalidade da mesma.

SOAC - Sistema Online de Auxílio ao Cafeicultor
Desenvolvido por: Daniel Guimarães do Lago
Universidade Federal de Lavras

Figura 36: Utilitário Alteração de Dados de Login visto de um computador desktop.

As mesmas preocupações tomadas com o primeiro registro do usuário também foram tomadas nessa tela, como por exemplo o algoritmo de criptografia monodirecional *SHA1* para ocultar a senha real de qualquer pessoa que tente utilizar o sistema. Como o usuário já está logado para efetuar a alteração de sua senha no sistema não houve preocupação de pedir a confirmação da senha atual do mesmo para efetuar a alteração de seus dados cadastrais.

4.9. SOAC: DECISÕES DE DESENVOLVIMENTO

Durante o desenvolvimento do SOAC houve preocupações globais de

desenvolvimento que afetaram o sistema como um todo. Para não tornar repetitivo explicações sobre políticas adotadas em todas as telas, todas as decisões de implementações globais que foram tomadas estão neste tópico.

A primeira preocupação foi com a segurança. Diversas medidas foram tomadas com relação a segurança. Para evitar a simulação de sessões e acesso de máquinas não-autorizadas, *cookies* não foram utilizados. *Cookies* são informações armazenadas no navegador da máquina cliente do usuário que acessa um determinado *site* na Internet. As informações armazenadas por um *cookie* podem ser utilizadas para obtenção de informações da *sessão* que o usuário estabeleceu com o *site*, tais como dados digitados previamente que permitam o acesso autenticado do usuário ao sistema. Essa ação foi tomada não pelo fato de *cookies* serem por si inseguros, mas para evitar que terceiros que possam eventualmente usar a mesma máquina que acessa o SOAC (supondo que a máquina tenha vários usuários) utilizarem de *cookies* armazenados em cache para efetuarem o acesso não-autorizado ao sistema.

Outra medida tomada, foi evitar no máximo possível *Code Injection*. Shiftlett (2005) afirma que *Code Injection* é uma prática baseada no envio de dados “estranhos” ao servidor que hospeda um determinado *site* através da prática de *buffer overflow*, fazendo com que este servidor execute códigos mal-intencionados. Embora seja raríssimo pessoas que dominem tal prática, ela existe e pode ser muito perigosa. O *PHP* é imune a *Code Injection*. No entanto há vários *módulos*, que são basicamente extensões para que o *PHP* consiga realizar ações que normalmente não conseguiria fazer sozinho, que não são provados contra essa técnica. Para evitar problemas de segurança por *code injection* nenhum módulo adicional foi utilizado além do *PHP* e o necessário para conectá-lo com o *MySQL*.

Uma outra técnica utilizada por pessoas com acesso malicioso à Internet e, provavelmente a mais comum de se encontrar em *sites* com *PHP*, devido à imperícia dos seus programadores, é chamada de *SQL Injection*. Essa prática consiste em os dados preenchidos em campos de formulários, não adequadamente processados pelo *PHP*, serem

diretamente utilizados pelo SGBD e, por isso, executarem ações prejudiciais ao sistema. Uma grande estratégia usada para prevenir esse tipo de ação é tratar todos os dados de entrada utilizando caracteres de escape nos mesmos. No SOAC todos os dados preenchidos através de formulários passam por funções que efetuam o escapamento de todos os caracteres de entrada antes de serem processados, exceto os dados que são utilizados para os algoritmos de criptografia monodirecional, para não alterarem a senha antes de ser armazenada no banco de dados e por serem inúteis nesta situação pois a cadeia de caracteres de entrada será processada de qualquer maneira.

No desenvolvimento do SOAC também houve outras preocupações além das já demonstradas com segurança. Nos parágrafos a seguir serão mostradas as decisões de desenvolvimento do sistema relacionados a usabilidade.

É bem sabido que para uma navegação otimizada a quantidade de informações que serão exibidas na tela nunca devem exceder uma largura determinada. Para computadores *desktop* e *notebook* é um consenso que a largura máxima que cada tela deve ter não deve ser superior a 800 pixels. Em todas as telas do Sistema Online de Auxílio ao Cafeicultor essa preocupação foi levada em conta durante o desenvolvimento. A largura de telas em *handhelds* já se tornou um problema maior: não existe uma largura de tela padronizada em máquinas *handhelds*. Os *handhelds* atuais em média possuem telas de resoluções que variam de 176 pixels a 480 pixels, e com alguns padrões intermediários como por exemplo 320 pixels. Foi constatado ser impossível manter uma interface amigável do SOAC em *handhelds* de baixíssima resolução como 176 pixels; portanto o padrão de largura de tela adotado para maximizar a navegação no SOAC através deste tipo de dispositivo foi de 320 pixels.

Um fator preocupante também foi o grau de instrução relativo a utilização de equipamentos de informática por administradores rurais. Na fase de pesquisas foi verificado que a grande maioria das pessoas que se enquadram neste cargo possuem pouca ou nenhuma experiência na utilização de equipamentos de informática. Levando isso em consideração buscou-se ao máximo exibir mensagens simples e de fácil entendimento.

Mais do que isso, em todas as telas foi adicionada uma barra de status no próprio sistema, que informava o tempo todo o que o usuário poderia fazer. Por exemplo, se o usuário do sistema passasse o mouse em cima de qualquer campo, na barra de status é exibida uma mensagem informando ao utilizador qual é o tipo de dado esperado naquele campo, bem como o padrão de entrada. A utilização de cores legíveis e várias das técnicas pregadas para um bom sistema de interface homem-máquina foram feitas.

A padronização do sistema também foi levada em conta no SOAC. Para facilitar a navegação no sistema entre as plataformas de computadores pessoais, como *desktops* e dispositivos portáteis, como *handhelds*, a disposição de todas as telas foram intencionalmente dispostas da forma mais semelhante possível, dessa forma mantendo o máximo possível a intuitividade do usuário ao acessar o sistema.

Para acessibilidade do sistema em âmbito universal, para não haver conflitos de exibição errônea de caracteres, todo sistema foi desenvolvido com caracteres em formato *UTF-8*. Desta forma, pode-se acessar o sistema de qualquer local do mundo sem a geração de caracteres estranhos na tela e também sem conflitos entre o *PHP* e o *MySQL*, independentemente do sistema operacional que estes estejam utilizando.

Para maximização da legibilidade, compatibilidade com dispositivos móveis e facilitar a capacidade de efetuar as operações de busca nas mais diversas funcionalidades do SOAC, todos os caracteres de entrada passam também por um procedimento denominado no sistema como *tratar_string*. Este procedimento efetua a remoção de todos os acentos de caracteres de entradas e passam o valor de entrada para letras em maiúsculas. Desta forma, por exemplo, se o usuário digitar que deseja buscar através do *handheld* um funcionário com o nome *joao* se tiver um funcionário que foi cadastrado inicialmente no sistema como tendo o nome *João* este funcionário será retornado na busca.

Para facilitar a leitura dos retornos de buscas, e de listagens no sistema, todas elas são ordenadas de alguma forma, geralmente alfabeticamente mas eventualmente em ordem mais adequada para exibir seu conteúdo.

Por último, mas não menos importante: uma grande preocupação do SOAC foi com relação à banda consumida por *handhelds*. Geralmente os computadores portáteis denominados *handhelds* possuem dispositivos que permitem o acesso destes à Internet, citando por exemplo os que conectam a redes *GPRS* e o *EDGE*. Como essas tecnologias provêem banda limitada (no exemplo, 80 kbps na *GPRS* e 384 kbps na *EDGE*), foi um dos objetivos do trabalho reduzir o consumo da banda. Essa redução foi feita realizando a remoção da exibição de todas as imagens do sistema para *handhelds*. É óbvio que a transferência de dados depende das pesquisas que foram realizadas, mas em simulações realizadas com o sistema na média as maiores pesquisas retornavam ao *handheld* 60 KB de dados, ou seja, demorariam 6 segundos numa rede *GPRS* ou pouco mais de 1 segundo numa rede *EDGE*.

5. SoacCel: DESCRIÇÃO DO SISTEMA

Como já visto, o objetivo do presente trabalho foi a elaboração de um *software* com foco na mobilidade. Um dos equipamentos mais comuns nos dias de hoje e que mais permitem a mobilidade são os telefones celulares. Com base nisso, pensou-se durante o processo da modelagem do SOAC como seria possível torná-lo mais móvel ainda. A resposta encontrada foi torná-lo acessível também através de telefones celulares.

Na atualidade poucos telefones celulares possuem um *browser* integrado, e os que possuem tal ferramenta têm ínfima largura de banda. No entanto, já faz algum tempo que vários dos celulares fabricados estão vindo com suporte a tecnologias como *J2ME*. A solução encontrada para tornar a mobilidade do SOAC extensível ao celular então foi a criação de um outro *software*, denominado SoacCel, que torna possível execução neste tipo de dispositivo.

Neste capítulo será abordada a construção do SoacCel, como foi implementado cada parte do sistema dividido por telas, bem como uma visão geral explicando as decisões de implementação ocorridas no mesmo. Inicialmente será dada uma visão geral, após isso serão apresentados os módulos referente ao sistema de *Login e Menu Principal* seguido pelos módulos de consultas de *Funcionários, Relatórios Simplificados, Agenda, Lista Telefônica* e, por fim, as decisões tomadas durante o desenvolvimento deste sistema.

5.1. SoacCel: VISÃO GERAL

O SoacCel é uma versão simplificada do SOAC desenvolvida para telefones celulares que permite a realização de consultas que podem ser feitas normalmente pelo SOAC via celular. A decisão de permitir apenas consultas e não a inserção, alteração ou exclusão de conteúdos do banco de dados é que devido ao volume de dados que o administrador rural necessita digitar é inviável realizar esta digitação em grande quantidade via dispositivo celular. Na realidade, o objetivo do SoacCel é permitir apenas

consultas simples que podem ser necessárias eventualmente para um administrador rural. Por isso as demais operações foram descartadas durante a fase da modelagem do sistema.

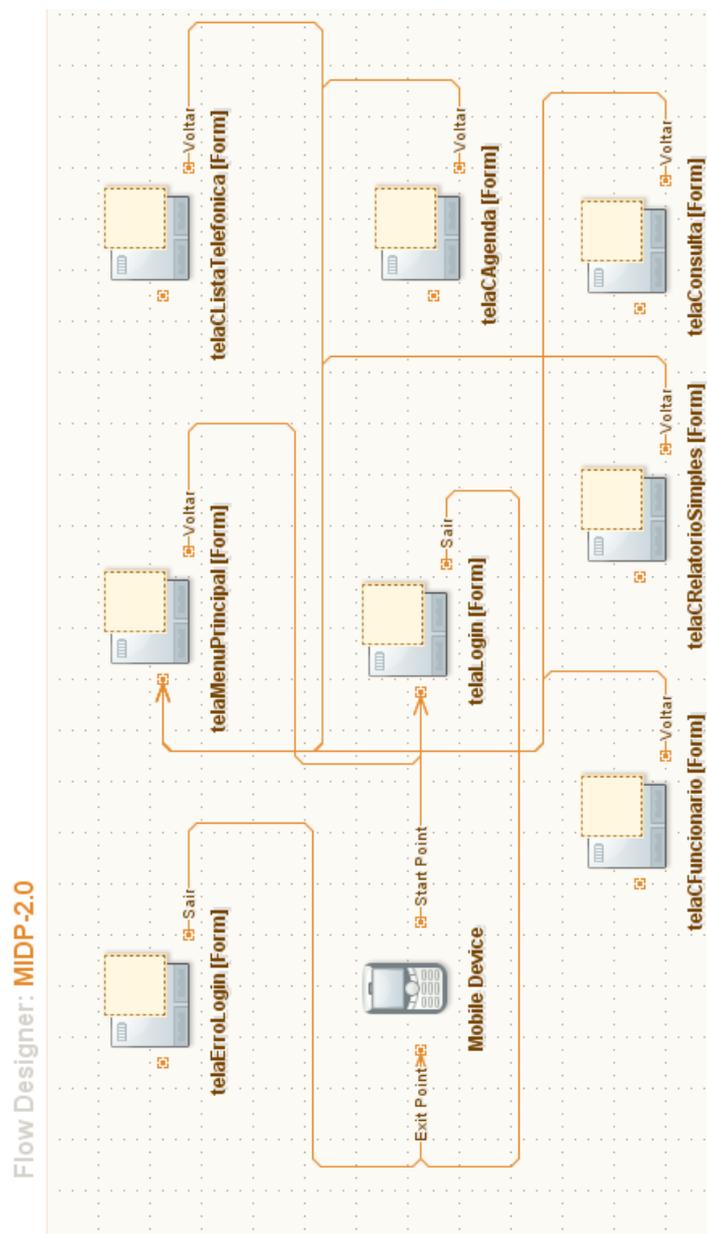


Figura 37: SoacCel: Diagrama de Fluxo.

A modelagem global do SoacCel envolveu, portanto, várias telas de consultas e telas de autenticação. Uma visão global do sistema pode ser melhor elucidada pelo

diagrama de fluxo mostrado na figura 37. Analisando este diagrama será feita agora uma descrição superficial do funcionamento do sistema.

Quando o dispositivo celular inicializa o SoacCel, a primeira tela que o usuário recebe é uma tela de *login*, onde o mesmo deverá efetuar a autenticação para informar que está autorizado a utilizar o sistema. Se o usuário não efetuar o *login* com sucesso, é exibida uma mensagem de erro ao mesmo e ele retorna a tela inicial de *login*; em caso de sucesso uma tela contendo um menu principal é exibida. Nesta tela de menu principal o usuário possui quatro alternativas básicas: *Consultar Aquisições*, *Gerar Relatório Simples*, *Consultar Agenda*, *Consultar Lista Telefônica*, nas quais o usuário poderá obter informações do servidor que hospeda o sistema.

Tendo mostrada uma visão superficial do sistema, agora serão mostrados em detalhes cada funcionalidade do SoacCel.

5.2. SoacCel: LOGIN

Assim que o usuário inicia a execução do sistema SoacCel, a primeira tela que é exibida ao mesmo é a tela de *login*, ilustrada na figura 38.

Como pode se perceber, nesta primeira tela alguns dados são requeridos do usuário. Estes dados são referentes aos dados cadastrados pelo usuário durante a primeira vez que ele acessou o SOAC através de um computador pessoal e efetuou o registro de um *login* e *senha* para tornar o sistema acessível via celular. Nessa tela deve ser informado pelo usuário do sistema o *login* cadastrado, a *senha* para acesso via celular e, também, o servidor que está atualmente rodando o SOAC. Uma vez feito isso, o mesmo sistema de criptografia monodirecional utilizado para codificar a *senha* no SOAC, o *SHA1*, é efetuada no *SoacCel*. A partir deste momento ele se conecta ao servidor do sistema e, em caso de sucesso no *login* a tela contendo o menu principal do sistema será exibida; caso contrário

uma tela de erro é mostrada e o usuário retorna à tela inicial do sistema.

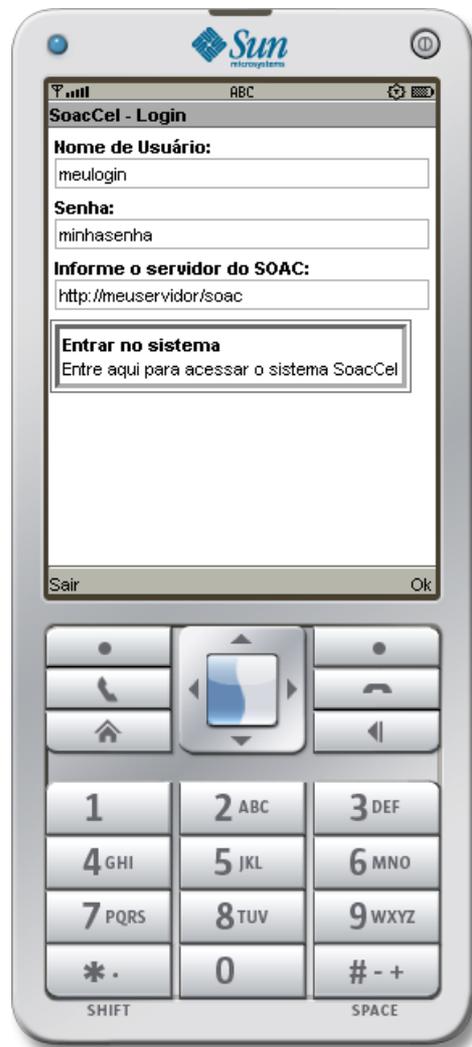


Figura 38: SoacCel: Exibição do Login visto de um Emulador de Telefone Celular.

A tela de erro que é retornada ao usuário em caso de falha no *login* do usuário é semelhante à figura 39.



Figura 39: SoacCel: Exibição de Mensagem de Erro de Login visto de um Emulador de Telefone Celular.

5.3. SoacCel: MENU PRINCIPAL

Uma vez tendo efetuado o *login* e se autenticado no sistema, o SoacCel exibe um

menu principal com todas as suas funcionalidades para que o usuário possa navegar entre elas e assim obter dados desejados. A tela que exibe o menu principal está demonstrada na figura 40.

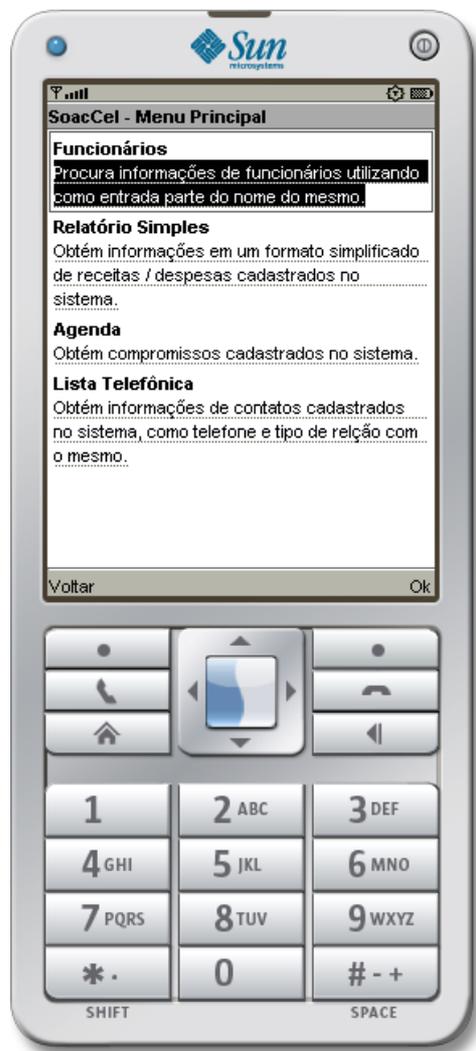


Figura 40: SoacCel: Exibição do Menu Principal visto de um Emulador de Telefone Celular.

Como é possível notar, na tela de menu principal estão todas as funcionalidades existentes no SoacCel: *Consulta de Funcionários, Geração de Relatório Simples, Agenda e*

Lista Telefônica. O usuário pode navegar entre estas opções para, através do servidor que hospeda o SOAC, obter os mesmos dados do sistema.

5.4. CONSULTA POR FUNCIONÁRIOS

Esta funcionalidade do SoacCel permite ao usuário efetuar uma consulta no sistema por funcionários cadastrados

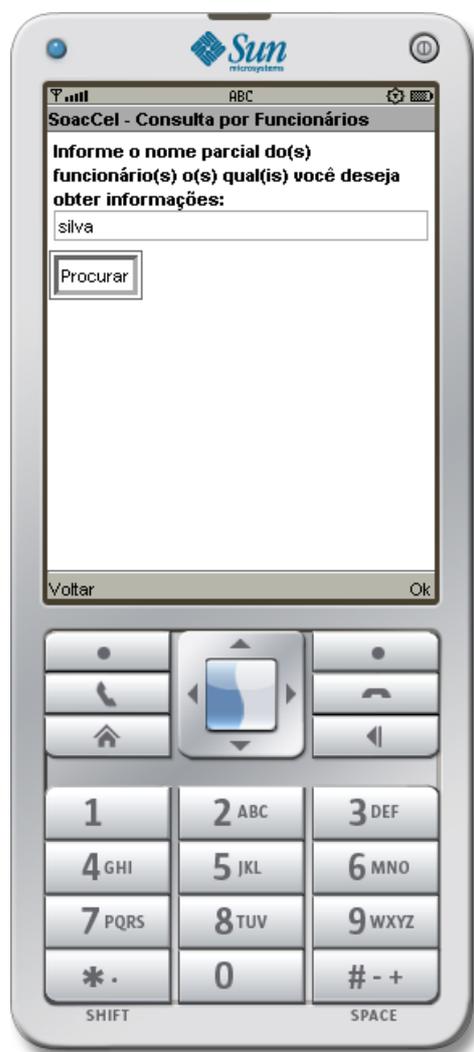


Figura 41: SoacCel: Exibição da Consulta por Funcionários vista de um Emulador de Telefone Celular.

Para exemplificar a funcionalidade mostrada, imagine que um cafeicultor está no meio de uma propriedade rural, e que (supondo que a mesma seja atendida por rede de telefonia celular) este administrador rural *necessite* de obter informações sobre um funcionário que ele se recorda apenas que o sobrenome seja *Silva*. Neste caso, ele utiliza a tela de consultas de funcionários e preenche os dados conforme as informações que este administrador rural tem disponível, conforme pode ser visto na figura 41.



Figura 42: SoacCel: Exibição da Consulta por Funcionários vista de um Emulador de Telefone Celular, após efetuar a consulta.

Após os dados serem fornecidos, o SoacCel realizará uma busca em seu banco de dados do SOAC e retornará informações das consultas, conforme mostrado na figura 42.

5.5. RELATÓRIOS SIMPLIFICADOS

Outra funcionalidade do SOAC também presente no SoacCel é a capacidade de gerar relatórios simplificados com base em todos os dados principais cadastrados.

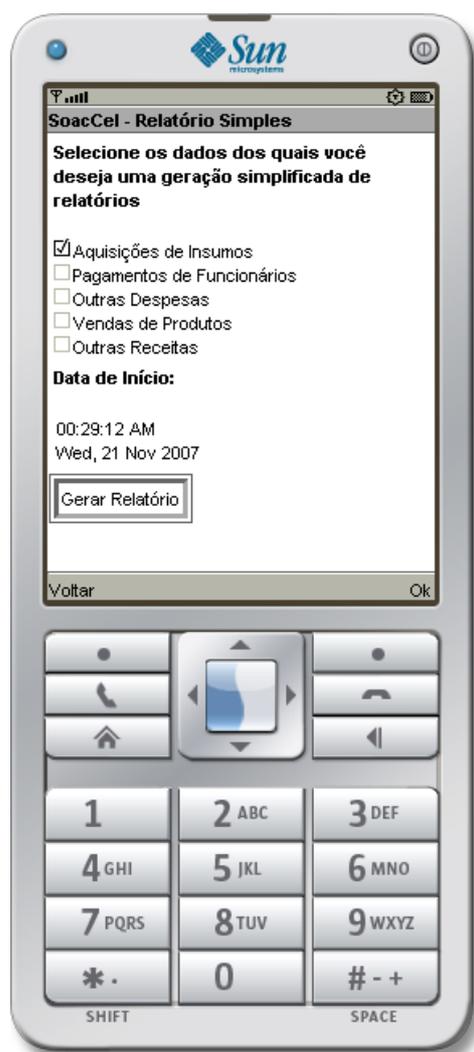


Figura 43: SoacCel: Geração de Relatórios Simplificados vista de um Emulador de Telefone Celular.

A tela representada na figura 43 demonstra essa geração de relatórios. Nessa interface, o usuário deverá marcar as caixas contendo as opções do relatório que o mesmo deseja que sejam exibidos na tela, bem como a data de início para geração de relatório, seguindo as mesmas regras do SOAC. A hora de início será desprezada. Após o usuário efetuar a consulta no sistema uma tela de retorno semelhante a figura 44 é mostrada.

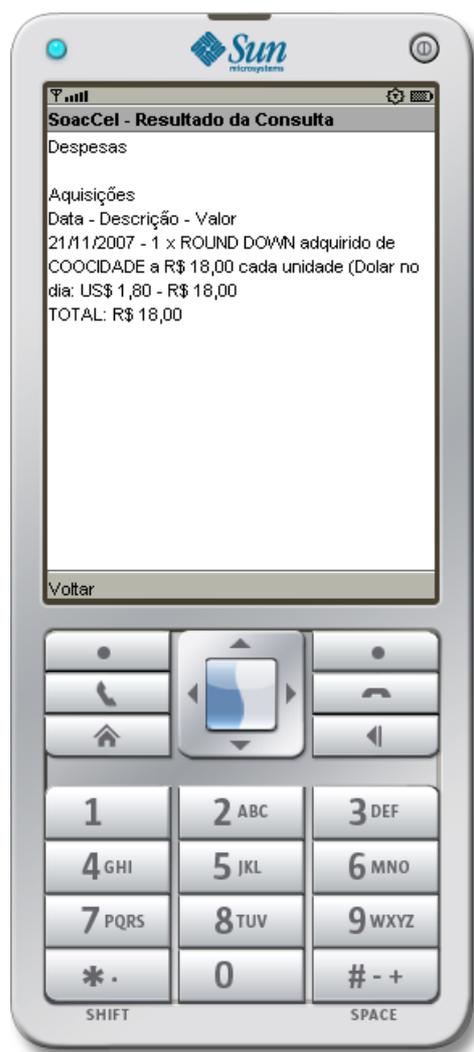


Figura 44: SoacCel: Exibição da Geração de Relatórios Simplificados vista de um Emulador de Telefone Celular, após a geração.

5.6. AGENDA

Dado que quando atividades são realizadas por administradores rurais eles dificilmente permanecem em um local onde computadores estejam disponíveis com fácil acesso, foi identificada como importante a portabilidade na funcionalidade *Agenda* do SOAC. Por isso houve preocupação de desenvolver o acesso à mesma no SoacCel conforme apresentado na figura 45.



Figura 45: SoacCel: Exibição da Agenda vista de um Emulador de Telefone Celular.

Por meio do SoacCel é possível obter informações da agenda em qualquer local. A agenda do SoacCel foi modelada para permitir o rápido acesso aos compromissos do dia atual, dia anterior e dia próximo ao atual. A tela que permite efetuar a consulta na agenda está como na figura 45.



Figura 46: SoacCel: Exibição da Agenda vista de um Emulador de Telefone Celular, informando os compromissos em datas próximas a atual.

Tendo esta tela mostrada, o usuário basta confirmar que deseja listar seus compromissos do dia atual, dia anterior e próximo dia que o SoacCel se conectará ao SOAC e fará a exibição adequada na tela como retorno da funcionalidade. A tela de retorno, para essa consulta em exemplo, é mostrada conforme figura 46.

5.7. LISTA TELEFÔNICA



Figura 47: SoacCel: Exibição da Lista Telefônica vista de um Emulador de Telefone Celular.

A última funcionalidade existente no SoacCel é a capacidade de acessar os telefones cadastrados no SOAC. Desta forma, é possível ter uma mesma base unificada de dados para conter este dado que possibilita a comunicação entre o administrador rural e outras pessoas. A tela que permite o usuário a efetuar consultas no SoacCel é demonstrada através da figura 47.

Semelhante às demais telas, a consulta obterá os dados que o usuário preencheu e com base neles a pesquisa será realizada. Assim como no resto do sistema, os dados não preenchidos serão ignorados.

5.8. DECISÕES DE DESENVOLVIMENTO

Assim como no SOAC, houve preocupações durante o desenvolvimento de todo o SoacCel. A primeira preocupação de implementação foi a seguinte: o banco de dados do servidor do SOAC inicialmente permitia apenas a abertura de conexão local. Pode-se abrir o banco de dados publicamente para um acesso global mas, tendo em vista que o *PHP* já está aberto para conexões externas, é possível utilizar mais este intermediador para tentar aumentar ainda mais a segurança do acesso. Para aumentar a segurança deste acesso, todos os acessos do SoacCel ao banco de dados foram intermediados através de uma página em *PHP* que interpreta as requisições para o banco de dados e retorna os dados interpretados.

Problemas com o fato dos dados armazenados serem *UTF-8* e o dispositivo celular padrão ter como sistema de exibição de caracteres sendo compatível com o *ISO8859-1* foram tratados retirando os acentos de todas as mensagens que são enviadas do banco de dados ao dispositivo celular antes dos dados serem armazenados no próprio banco de dados. Para evitar os problemas dos caracteres identificadores de um documento com o cabeçalho *UTF-8* foi colocado no celular um teste para saber se há caracteres nestes padrões e ignorá-los durante a leitura.

Houve preocupação também com o consumo de banda. A banda utilizada pelo sistema transfere apenas texto, na menor quantidade possível para que os dados possam ser interpretados.

6. CONCLUSÕES

Neste capítulo será abordada a conclusão obtida após a realização do presente trabalho, assim como as propostas de realização de trabalhos futuros.

6.1. CONSIDERAÇÕES FINAIS

O setor agrário, apesar dos avanços sofridos nos últimos anos, ainda demanda de forte tecnologia que permita a portabilidade e seja de fácil utilização.

O SOAC e SoacCel foram *softwares* elaborados que buscaram, dentro de suas limitações atender as principais necessidades na cadeia produtiva do setor agrário, enfocando principalmente o setor cafeeiro. Embora a modelagem inicial do projeto tenha enfatizado exclusivamente o setor, notou-se que, devido à construção com foco generalista, este *software* pode ser facilmente modificado para atender a necessidade de outros setores que não sejam necessariamente o do café.

O foco na mobilidade para este sistema permitiu demonstrar que é possível realizar a aplicação prática da computação pervasiva no setor rural, integrando aparelhos de telefones celulares, *handhelds*, *laptops*, computadores pessoais.

O desenvolvimento deste *software* auxiliará administradores rurais em suas tarefas, especialmente as do setor produtivo.

6.2. TRABALHOS FUTUROS

Ao término de desenvolvimento deste *software*, foi verificado que seria de grande valia que o mesmo evoluísse no sentido de lidar com ativos. Durante a fase de

levantamento de requisitos, alguns administradores rurais alegaram lidar com rendas secundárias além do café, por exemplo gado, e gostariam de ter um controle maior dentro do próprio SOAC sobre seu rebanho, como número de cabeças e idade dos mesmos, para facilitar a tomada de decisão não apenas com vendas de café mas também com outras alternativas.

Outra sugestão adaptar o SOAC a setores que não são apenas do café ou rurais. Essa sugestão é plausível pois na modelagem e desenvolvimento do SOAC e SoacCel houve um objetivo de torná-los o mais genérico possível para auxiliar a administração rural.

Pode-se também tornar interessante a inserção, edição e alteração de dados no SoacCel, especialmente com relação a agenda que o acompanha.

A aplicação em campo do mesmo por vários administradores rurais se faz adequada para o sistema desenvolvido para sua validação, uma vez que esta não foi realizada.

Como última sugestão, pode-se citar a incrementação do sistema para atender maiores necessidades dos produtores rurais e efetuar maior redução do consumo de banda em transferências de dados para celulares e *handhelds*.

7. REFERÊNCIAS BIBLIOGRÁFICAS

ANDRADE, J. G., **Gerenciando a Fazenda Cafeeira**, 1.ed., Lavras, Editora: UFLA/FAEPE, 1999, 83 p.

BULGER B., GREENSPAN J., WALL D., **MySQL/PHP Database Applications**, 2ª Edição, Editora Willey, Indianápolis, 2004. 819p.

CASTRO, F. L., **Um Sistema de Suporte ao Agronegócio do Café Utilizando Agentes Inteligentes para Recuperação de Informações na web**. Lavras – Minas Gerais, 2003. 75p. Monografia de Graduação – Universidade Federal de Lavras. Departamento de Ciência da Computação.

CIM – Centro de Inteligência de Mercados. Disponível em <<http://www.cim-agro.com.br>> . Acesso em novembro/2007.

CLARO, D. P., **Análise do Complexo Agroindustrial das Flores do Brasil**. 1998. Dissertação de Mestrado - Universidade Federal Lavras.

COAD, P., MAYFIELD, M., **Projeto de Sistemas em Java: Construindo Aplicativos e Melhores Applets**; tradução Altair Dias Caldas; revisão técnica Álvaro Rodrigues Antunes, 1.ed., São Paulo, Editora MAKRON Books, 1998, 0232 p.

COBUCCI, C. E. M., **SisGAM: Um sistema móvel para gestão acadêmica**, Monografia de Graduação, Lavras, 2007, 148 p.

CORRÊA, C. H. W., **A complexidade do conceito de interação mediada por computador: para além da máquina**, UNIREvista: Editora PUCRS, Vol. 1, nº 3, 2006. 9 p.

ELSMARI, R., **Sistemas de Banco de Dados** / Ramez Elmasri e Shamkant

B. Navathe; vários tradutores, 4ª Edição, Editora Addison Wesley, São Paulo, 2005.

EMBRAPA. **Histórico**, Disponível em <http://www22.sede.embrapa.br/cafe/unidade/historico.htm>. Acesso em agosto/2007.

FERREIRA, D. M., **Sistema de Produtividade na Colheita de Café Utilizando Incerteza em Banco de Dados**. Lavras – Minas Gerais, 2001. 52p. Monografia de Graduação – Universidade Federal de Lavras. Departamento de Ciência da Computação.

JESUS, J. C. S.; ZAMBALDE, A. L. Informática na Agropecuária: Hardware, Software e Recursos Humanos, **AGROSOFT 99**, II Congresso da SBI-Agro (Associação Brasileira de Agroinformática), 1999.

LIMA, I. G.; COSTA, H. A. X.; RESENDE, A. M. P.; **Apostila J2ME**, Mini-Curso de Introdução à Programação *J2ME*, ERIMG, Lavras, 2007. 39p.

LIMA, M. S.. **Projeto Livre OpenFarm 2.0**, Disponível em http://www.matheussantanalima.x-br.com/projetos/openfarm/new_doc.html >, Acesso em janeiro/2008.

MEIOBIT. **Mobilidade X Portabilidade**. [S.I.]: MeioBit, 2006. Disponível em: <http://www.meiobit.com/arq/007536.HTML>>. Acesso em abril/2006.

MOBILELIFE. **Mobile Life Oferece Mobilidade**. [S.I.]: MobileLife, 2005. Disponível em: <http://www.mobilelife.com.br/>>. Acesso em janeiro/2006.

NETTO, M. M. **Design para Dispositivos Móveis - O primeiro passo para**

um sistema bem sucedido, é a interação com o usuário. Disponível em: <<http://www.linhadecodigo.com.br/>>. Acesso em abril/2006.

OLIVEIRA, A. P., MACIEL, V. V., **Java na Prática**, Volume 1, 1.ed., Viçosa, Editora: UFV, 2002, 272 p.

RAMALHO, J.A., **SQL: A Linguagem dos Bancos de Dados**, 1. ed., São Paulo, Editora: Berkeley, 1999, 627 p.

RIBEIRO, G., **Entendendo a 3G**: HowStuffWorks, Inc, 2008. Disponível em: <<http://eletronicos.hsw.uol.com.br/telefonias-3g1.htm>>. Acesso em janeiro/2008.

RFC 2246 - **The TLS Protocol**: The Internet Engineering Task Force, 1999. Disponível em: <<http://www.ietf.org/rfc/rfc2246.txt>>. Acesso em janeiro/2008.

RFC 3174 - **US Secure Hash Algorithm 1 (SHA1)**: Faqs.org, 2007. Disponível em: <<http://www.faqs.org/rfcs/rfc3174.html>>. Acesso em dezembro/2007.

SANTIAGO, D. P., **Busca de Padrões de Negociações e Cotações da Commodity Café no Mercado de Futuros**. Lavras – Minas Gerais, 2002. 39p. Monografia de Graduação – Universidade Federal de Lavras. Departamento de Ciência da Computação.

SANTOS, R. V. M., **Desenvolvimento de um Sistema Especialista para Análise do Cálculo do Custo da Pós-Colheita do Café**. Lavras – Minas Gerais, 2005. 54p. Monografia de Graduação – Universidade Federal de Lavras. Departamento de Ciência da Computação.

SERRÃO, C., MARQUES, J., **Programação com PHP**, 1.ed., Editora: FCA, 2001, 564 p.

SHIFLETT, C., **Essential PHP Security**, 1 ed. Editora O'Reilly, 2005, 124 p.

SIMIÃO, S. A., **Utilização de Software de Administração Rural em Mato Grosso**, Revista ConnectiON Line, Editora UNIVAG, 2006. 8 p.

W3C – World Wide *Web* Consortium. <<http://www.w3.org/>>. Acesso em novembro/2007.

**Ficha Catalográfica preparada pela Divisão de Processos Técnicos
da Biblioteca Central da UFLA**

Lago, Daniel Guimarães do

SOAC: Proposta de um Sistema Online de Auxílio ao Cafeicultor com foco na Mobilidade / Daniel Guimarães do Lago – Minas Gerais, 2007. 18 p : il

Monografia de Graduação – Universidade Federal de Lavras. Departamento de Ciência da Computação.

1. Dispositivos Móveis. 2. Engenharia de Software. 3. Redes de Computadores. I. LAGO, D. G. II. Universidade Federal de Lavras. III. SOAC: Sistema Online de Auxílio ao Cafeicultor com foco na Mobilidade.