

Rafael Vargas Mesquita dos Santos

**Desenvolvimento de um Sistema Especialista para a Análise do Cálculo do
Custo da Pós-Colheita do Café**

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

Orientador
Prof. Ana Cristina Rouiller

Lavras
Minas Gerais - Brasil
2005

Agradecimentos

Agradeço a Deus de todo o meu coração, pois a vida só tem sentido por causa Dele. Agradeço também a minha mãe, meu pai, e meu irmão, os quais formam a melhor família que eu poderia ter. Agradeço a minha orientadora e a meu co-orientador por me ensinarem não somente no projeto, mas também na formação pessoal. E por fim, agradeço a meus amigos, cada um deles com certeza têm parte neste trabalho (em especial aos integrantes da equipe de desenvolvimento do PÓS-CAFÉ).

Dedico este trabalho primeiramente a Deus. Ele como autor da vida me ajudou a escrever mais um capítulo da minha. Dedico também a meus pais e a meu irmão, que sempre fizeram questão de viver cada momento meu. Amo todos vocês!

Resumo

Este estudo mostra o desenvolvimento de um Sistema Especialista para analisar o cálculo do custo dos processamentos do café durante a fase de pós-colheita. Neste desenvolvimento foram aplicadas técnicas e padrões de Engenharia de Software. O sistema proposto tem como objetivo facilitar a decisão dos produtores, cooperativas e consultores da área, quanto aos processamentos os quais devem ser aplicados ao café.

Palavras-chave: sistemas especialistas, engenharia de software, pós-colheita do café

Sumário

1	Introdução	1
1.1	Caracterização do problema	1
1.2	Objetivos e Justificativa	2
1.3	Metodologia	3
1.4	Organização do trabalho	3
2	Sistemas Especialistas	5
2.1	Introdução	5
2.2	Definição	5
2.3	Arquitetura de um SE	6
2.4	Considerações finais	8
3	Engenharia de Software	9
3.1	Introdução	9
3.2	Definição	9
3.3	Modelos de processo de software	12
3.4	UML	15
3.5	Considerações finais	17
4	Metodologia de Desenvolvimento	19
4.1	Escolha da tecnologia	19
4.1.1	Linguagem de programação	19
4.1.2	Sistema operacional	20
4.1.3	Linguagem de modelagem	20
4.1.4	Banco de dados	21
4.2	Gerência de configuração	21
4.3	Arquitetura do sistema	22

4.4	Especificação dos requisitos	23
4.5	Modelagem do sistema	25
4.6	Implementação	27
4.7	Verificação	27
4.8	Divulgação e distribuição do sistema	28
4.9	Considerações finais	28
5	O Sistema PÓS-CAFÉ	31
5.1	Principais funcionalidades	34
5.1.1	Calcular o custo dos processamentos	34
5.1.2	Gerar gráfico comparativo entre os processamentos	35
5.1.3	Atualizar dados das máquinas agrícolas e o preço do café	37
5.1.4	Atualizar versão do sistema	37
5.1.5	Controlar chaves de registro	37
5.1.6	Gerar instalador do sistema	37
5.1.7	Módulo de ajuda	38
5.2	Considerações finais	38
6	Conclusões e Trabalhos Futuros	39
6	Referências Bibliográficas	41

Lista de Figuras

1.1	Possíveis processamentos do café abordados pelo sistema.	2
2.1	Arquitetura de um Sistema Especialista (SE).	6
3.1	Diagrama de colaboração ilustrando mensagens entre objetos de software.	17
3.2	Diagrama de classes de projeto de componentes de software.	18
4.1	Apresentação da arquitetura de três camadas.	22
4.2	Especificação das classes utilizadas no sistema.	23
4.3	Diagrama de colaboração para o requisito 'Logar no sistema'.	26
4.4	Especificação das classes utilizadas no sistema (modelo conceitual).	26
4.5	Modelagem do banco de dados do sistema.	27
4.6	Site do sistema.	29
4.7	CD correspondente à versão de demonstração do sistema.	29
5.1	Visão geral do sistema PÓS-CAFÉ.	32
5.2	Tela principal do sistema PÓS-CAFÉ.	33
5.3	Tela para escolher os processamentos do café.	34
5.4	Tela padrão de entrada de dados.	35
5.5	Tela específica de entrada de dados.	36
5.6	Tela correspondente ao gráfico gerado pelo sistema.	36
5.7	Tópicos correspondentes ao módulo de ajuda do sistema.	38

Capítulo 1

Introdução

1.1 Caracterização do problema

O setor de software no mundo tem impressionado pelo seu dinamismo recente. Ao contrário do previsto por alguns autores como Baumol, Blackman e Wolf [BAUMOL1991], ele tem apresentado elevado índice de crescimento de sua produtividade, graças a dois fatores. Em primeiro lugar, a emergência da Engenharia de Software permitiu a adoção de técnicas de desenvolvimento de software mais transparentes, reduzindo seu custo de manutenção, e, principalmente, não desperdiçando esforços já realizados [PONDÉ1997]. Em segundo lugar, o surgimento de máquinas com maior capacidade de processamento e armazenamento de informações possibilitou a substituição de mão-de-obra por equipamento, ao mesmo tempo em que permitiu a utilização de linguagens de mais alto nível.

No caso brasileiro, a reestruturação tem-se acelerado nos últimos anos, principalmente com a difusão do uso de microcomputadores e de comunicação via Internet, influenciando diretamente na economia nacional.

Por outro lado, a agricultura brasileira também é um setor econômico estratégico para a estabilização da economia, sendo o café um dos mais tradicionais produtos da agricultura brasileira, tendo grande influência no processo de industrialização da economia.

Todavia, para os produtores de café, consultores e cooperativas ligadas à área de pós-colheita do café, ainda é muito difícil decidir sobre quais os processamentos devem ser aplicados ao café durante a fase de pós-colheita do mesmo (a figura 1.1 mostra alguns dos possíveis processamentos). Muito deve ser considerado para

que decisões como essas possam ser tomadas com maior facilidade por parte dos produtores, consultores e cooperativas [BORÉM2004].

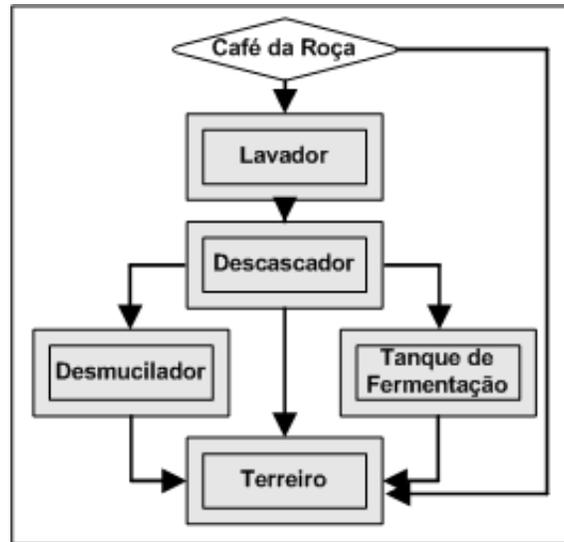


Figura 1.1: Possíveis processamentos do café abordados pelo sistema.

1.2 Objetivos e Justificativa

Este trabalho teve como objetivo a construção de um Sistema Especialista, voltado à área do agro-negócio, seguindo os conceitos da disciplina de Engenharia de Software - a qual se ocupa de todos os aspectos da produção de software.

O desenvolvimento desse sistema está embasado na teoria e técnicas de análise de decisão, utilizando padrões de Engenharia de Software, na finalidade de auxiliar os cafeicultores na escolha entre alternativos sistemas de processamento.

A atividade de desenvolvimento de software é cara. Cada vez mais, organizações que desenvolvem software tornam-se cientes da importância de se utilizar processos bem definidos e documentados, tendo em vista a necessidade de se construir produtos de qualidade real, para garantir o retorno de investimento [RUP2002].

Uma outra justificativa para este trabalho é que, apesar de existirem muitos

softwares relacionados ao custo de produção do café (softwares que não podem ser considerados Sistemas Especialistas), ainda não existia, no Brasil, nenhum software que fosse aplicado à fase de pós-colheita do café, e que considerasse um número tão grande de variáveis influenciadoras para o cálculo deste custo.

1.3 Metodologia

Antes do desenvolvimento do sistema, foi necessário estudar alguns conceitos relacionados à Sistemas Especialistas e Engenharia de Software. Isso porque, o PÓS-CAFÉ é um Sistema Especialista construído sobre padrões e técnicas de Engenharia de Software.

O desenvolvimento do sistema foi constituído das seguintes etapas:

- Definir o modelo de desenvolvimento de software a ser utilizado
- Escolher a tecnologia a ser utilizada
- Definir a arquitetura do sistema
- Elicitar e especificar os requisitos do sistema
- Modelar o sistema
- Implementar o sistema
- Testar o sistema
- Divulgar e distribuir o sistema

1.4 Organização do trabalho

Além deste capítulo introdutório o presente trabalho é composto por mais 5 capítulos.

No capítulo 2 deste trabalho, serão apresentados alguns conceitos relacionados à Sistemas Especialistas, na finalidade de facilitar o entendimento de alguns termos utilizados no conteúdo deste trabalho.

No capítulo 3, será apresentada a disciplina de Engenharia de Software. Isso porque, muitos padrões e técnicas dessa disciplina foram utilizadas durante o desenvolvimento do sistema PÓS-CAFÉ.

No capítulo 4, será realizada uma breve explanação das metodologias usadas para o desenvolvimento do trabalho e o procedimento metodológico para a realização do mesmo.

No capítulo 5, será apresentado o sistema propriamente dito. Uma breve descrição do sistema e suas principais funcionalidades fornecerão uma visão geral do PÓS-CAFÉ.

No capítulo 6, será apresentada uma conclusão sobre o trabalho realizado, bem como algumas propostas para possíveis trabalhos futuros. No final será descrita a bibliografia utilizada.

Capítulo 2

Sistemas Especialistas

2.1 Introdução

Nesta seção será mostrada a tecnologia de Sistemas Especialistas (SE), que é a tecnologia utilizada no sistema Pós-Café.

Serão explicados alguns tópicos mais importantes relacionados aos SE's. Também serão dados alguns exemplos de SE's utilizados em várias áreas.

2.2 Definição

Segundo Rabuske [RABUSKE1995], os Sistemas Especialistas são sistemas computacionais que resolvem problemas de uma maneira bem parecida com um especialista humano. Eles possuem um conhecimento específico profundo sobre campos restritos do conhecimento.

O que caracteriza um Sistema Especialista (SE) é o seu alto grau de conhecimento sobre uma área específica, sendo então muito eficientes suas as respostas. Os SE's devem também ser capazes de informar ao usuário como obteve a resposta para um determinado problema, para deixá-lo a par de como tal resultado foi obtido

Existem diversos benefícios associados ao desenvolvimento de um SE como, por exemplo, distribuição de conhecimento especializado, memória institucional, flexibilidade no fornecimento de serviços, possibilidade de tratar informações a partir de conhecimentos incompletos ou incertos, entre outros.

Atualmente, os SE's são aplicados nas mais variadas áreas. Abaixo serão apresentados alguns exemplos, segundo Rabuske [RABUSKE1995]:

- Administração: FOLIO - ajuda administradores a determinar metas de investimento de clientes e selecionar portfólios que melhor preencham essas metas;
- Agricultura: POMME - ajuda no trato de macieiras;
- Geologia: PROSPECTOR- dá suporte de consultor na exploração mineral.

2.3 Arquitetura de um SE

Um modelo geral de um SE pode ser representado pela figura 2.1, porém, essa modelagem pode variar conforme sua aplicação.

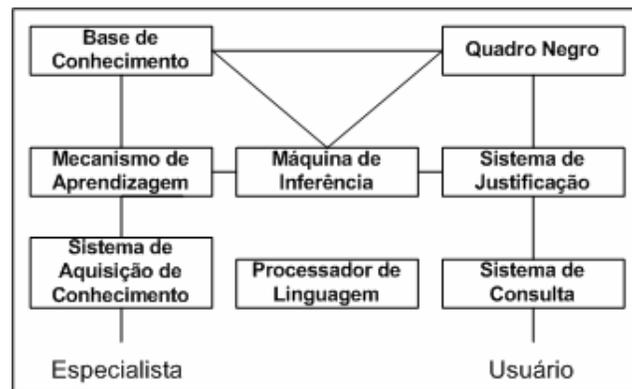


Figura 2.1: Arquitetura de um Sistema Especialista (SE).

Fonte: [RABUSKE1995]

Serão dados nos tópicos seguintes, detalhes sobre algumas partes do modelo apresentado.

A base do conhecimento

O grande poder de um SE está em sua base de conhecimento. É nessa base que está tudo necessário para que o sistema possa trabalhar para gerar resultados. Para

montar a base de conhecimentos são necessários dois passos: adquirir conhecimento e modelar o conhecimento adquirido.

Sistema de aquisição de conhecimento

O sistema de aquisição de conhecimento é responsável por dar conhecimento ao SE. Essa é a etapa mais difícil na construção de um SE. O conhecimento de um SE vem de um especialista humano. É necessário que a pessoa passe "tudo" que sabe sobre o assunto a ser tratado a fim de melhor representar o papel do especialista humano.

A aquisição do conhecimento é complexa porque é necessário saber lidar com o especialista, convencê-lo a passar o que sabe, a colaborar. É nessa fase que podem surgir informações incompletas, porque o especialista pode não ter informado tudo que era necessário, ter omitido algo.

A máquina de inferência

A máquina (ou motor) de inferência é que cuida da parte de tomada de decisão, analisando os dados da base de conhecimentos e da entrada do programa. Os motores de inferência possuem várias características (modo de raciocínio, estratégia de busca, resolução de conflitos e representação de incerteza).

O quadro negro

O quadro negro é a área de trabalho do Sistema Especialista. É nele que ficam armazenadas informações e fatos de suporte ao funcionamento do sistema, quando este efetua raciocínios.

Sistema de justificação

O sistema de justificação consiste na capacidade de questionamento fornecida aos usuários, seja para repetir uma dedução efetuada, seja para responder a outras questões que o sistema especificamente permita. A justificação é um requerimento obrigatório nos Sistemas Especialistas e geralmente tem a capacidade de responder às seguintes perguntas, segundo Rabuske [RABUSKE1995]:

- Como chegou a esta conclusão?

- Por que chegou a esta conclusão?
- Por que não chegou a outra conclusão?

Mecanismo de aprendizagem

O mecanismo de aprendizagem é a parte que cuida das alterações na base de conhecimento. As alterações podem ser simples como acréscimo de dados ou complexas como depuração da base, reordenação de prioridades, dentre outras.

O sistema de consulta

O sistema de consulta cuida da melhor interação entre o usuário e o programa. É esse módulo que permite ao usuário não ter conhecimento de como o sistema foi desenvolvido. Ele é um elo entre o programa e o usuário.

2.4 Considerações finais

O Sistema Especialista de análise do cálculo do custo da pós-colheita do café (PÓS-CAFÉ) é um sistema baseado em regras com o formato se A então B, em cujos elementos podem ser utilizados conectivos lógicos, além da teoria das probabilidades e dos conjuntos difusos. Para que o sistema funcione é necessário um motor de inferência que manipula as regras, de forma declarativa e não determinística¹. Segundo Waterman, este é o conjunto que se chama sistema baseado em regras [WATERMAN2002].

É o modelo estrutural mais conhecido e utilizado e que por muito tempo foi sinônimo de SE [WATERMAN2002]. Denominados sistemas de produção ou simplesmente regras de produção por Emil POST em 1943, são o método de representação do conhecimento na Inteligência Artificial (IA) mais utilizado. Isto se deve principalmente a sua facilidade de compreensão e programação, pois as regras fornecem uma maneira formal de representar diretivas e julgamentos próxima da linguagem natural e que permite descrever fatos dependentes do ponto de vista do contexto.

No próximo capítulo descreveremos alguns conceitos relacionados à Engenharia de Software. Uma das disciplinas estudadas para ajudar no desenvolvimento do sistema aqui proposto.

¹Que pode chegar ao mesmo resultado de maneiras diferentes

Capítulo 3

Engenharia de Software

3.1 Introdução

Nesta seção será apresentada a disciplina de Engenharia de Software (ES), da qual foram identificados padrões e técnicas de desenvolvimento de software utilizados no sistema PÓS-CAFÉ.

Serão explicados alguns tópicos mais importantes relacionados a ES. Também serão dados alguns exemplos de modelos de processos de software, bem como alguns conceitos sobre a UML (*Unified Modeling Language*).

3.2 Definição

O texto a seguir foi retirado do livro denominado 'Engenharia de Software' escrito por Sommerville [SOMMERVILLE2003].

Praticamente todos os países, hoje em dia, dependem de complexos sistemas com base em computadores. Cada vez mais os produtos incorporam, de algum modo, computadores e software de controle. Nesses sistemas o software representa uma grande e crescente proporção do custo total do sistema. Por isso, produzir software de um modo que apresente uma boa relação custo-benefício é essencial para o funcionamento das economias nacionais e internacionais.

A Engenharia de Software é uma disciplina da engenharia, cuja meta é o desenvolvimento de sistemas de software com boa relação custo-benefício. O software é abstrato e intangível, não é restrito por materiais nem regido por leis físicas ou

por processos industriais. De certa maneira isto simplifica a Engenharia de Software, por não haver limitações físicas para o potencial do software. Por outro lado, contudo, essa ausência de limitações naturais significa que o software pode facilmente se tornar bastante complexo e, conseqüentemente, muito difícil de ser compreendido.

A Engenharia de Software ainda é uma disciplina relativamente nova. A noção de 'Engenharia de Software' surgiu pela primeira vez em 1968, em uma conferência organizada para discutir a chamada 'crise do software'. Essa crise resultava diretamente da introdução (naquela época) do poderoso hardware de computador de terceira geração. Sua capacidade tornava viáveis aplicações de computador até então inimagináveis. O software resultante era bem maior e mais complexo do que os sistemas de software precedentes.

A experiência inicial de construção desses sistemas mostrou que uma abordagem informal do desenvolvimento de software não era o bastante. Projetos importantes sofriam atrasos, às vezes, de alguns anos. Por apresentarem custos muito maiores do que os inicialmente previstos, eles não eram confiáveis, eram de difícil manutenção e tinham desempenho inferior. O desenvolvimento de software estava em crise. Os custos de hardware caíam, enquanto os de software subiam rapidamente. Novas técnicas e novos métodos eram necessários para controlar a complexidade inerente aos grandes sistemas de software.

Essas técnicas se tornaram parte da Engenharia de Software e são amplamente, mas não universalmente, utilizadas. No entanto, ainda há problemas em produzir software complexos, que atenda às expectativas dos usuários e que seja entregue dentro do prazo e do orçamento estabelecido. Muitos projetos de software ainda têm problemas, e isso levou a alguns críticos, como Pressman [PRESSMAN1997], a sugerirem que a Engenharia de Software está em um estado de aflição crônica.

A medida que a nossa capacidade de produzir software aumentou, também cresceu a complexidade dos sistemas requeridos. Novas tecnologias que resultam da convergência de sistema de computadores e de comunicação, trazem novas questões para os engenheiros de software. Por essa razão e pelo fato de muitas empresas não aplicarem as técnicas de Engenharia de Software de maneira eficaz, ainda temos problemas. A situação não é tão ruim como os pessimistas sugerem, mas, com certeza, há espaço para melhorias.

Houve um grande progresso desde 1968 e o desenvolvimento da Engenharia de Software melhorou de modo marcante o software que produzimos. Temos uma compreensão muito melhor das atividades envolvidas no desenvolvimento de software. Desenvolvemos métodos eficazes de especificação, projeto e implemen-

tação de software. Novas notações e ferramentas reduzem o esforço para produzir sistemas grandes e complexos.

Os engenheiros de software podem, com razão, se orgulhar de suas realizações. Sem softwares complexos, não teríamos explorado o espaço, não teríamos a Internet e as modernas telecomunicações e todos os meios de viagem seriam mais perigosos e dispendiosos. A Engenharia de Software tem contribuído muito, em sua curta existência, e estou convencido de que, à medida que essa disciplina amadurecer, a contribuição que a mesma fornecerá no século XXI será ainda maior [SOMMERVILLE2003].

A seguir serão apresentadas algumas perguntas frequentes sobre Engenharia de Software. Todas estas perguntas estão respondidas segundo a visão exposta por Sommerville [SOMMERVILLE2003].

- O que é software?

São os programas de computador e a documentação associada. Produtos de software podem ser desenvolvidos para um cliente específico ou para o mercado.

- O que é Engenharia de Software?

Engenharia de Software é uma disciplina da engenharia que se ocupa de todos os aspectos da produção de software.

- O que é um processo de software?

É um conjunto de atividades, cuja meta é o desenvolvimento ou a evolução do software.

- O que é um modelo de processo de software?

É uma representação simplificada de um processo de software, apresentada a partir de uma perspectiva específica.

- O que são métodos de Engenharia de Software?

São abordagens estruturadas para o desenvolvimento de software, que incluem modelos de sistemas, notações, regras, recomendações de projetos e diretrizes de processos.

- Quais são os atributos de um bom software?

O software deve proporcionar ao usuário a funcionalidade e o desempenho requeridos e deve ser passível de manutenção, confiável e de fácil uso.

- Quais são os principais desafios enfrentados pela Engenharia de Software?

Lidar com sistemas legados¹ computadores, sistemas aplicativos, instalações de redes ou outros equipamentos antiquados que continuam em uso, convivendo com sistemas mais modernos., atender à crescente diversidade e atender às exigências quanto a prazos de entrega reduzidos.

3.3 Modelos de processo de software

Um modelo de processo de software é uma representação abstrata de um processo de software. Cada modelo de processo representa um processo a partir de um perspectiva particular, de uma maneira que proporciona apenas informações parciais sobre o processo [SOMMERVILLE2003]. Nesta seção, apresenta-se uma série de modelos de processo muito genéricos (algumas vezes chamados de paradigmas de processo) e isso é feito a partir de uma perspectiva arquitetural. Em outras palavras, vemos a estrutura do processo, mas não os detalhes das atividades específicas.

Esses modelos genéricos não são definições definitivas de processos de software. Em vez disso são abstrações úteis, que podem ser utilizadas para explicar diferentes abordagens do desenvolvimento de software. Para muitos sistemas de grande porte, naturalmente, não existe apenas um processo de software que possa ser utilizado. Processos diferentes são utilizados para desenvolver diferentes partes do sistema. Como exemplos de alguns dos modelos de processo temos:

- O modelo em cascata [ROYCE1970];
- Desenvolvimento evolucionário [BOEHM1988];
- Desenvolvimento formal de sistemas;
- Desenvolvimento orientado a reuso;
- Desenvolvimento iterativo [BOEHM1988]:
 - Modelo de desenvolvimento espiral;
 - Modelo de desenvolvimento incremental.

¹Computadores, sistemas aplicativos, instalações de redes ou outros equipamentos antiquados que continuam em uso, convivendo com sistemas mais modernos.

Os processos com base no modelo em cascata e no modelo de desenvolvimento evolucionário são amplamente utilizados para o desenvolvimento de sistemas práticos. O desenvolvimento formal de sistemas foi utilizado com sucesso em muitos projetos, mas os processos com base neste modelo ainda são utilizados em poucas organizações. O reuso informal é comum em muitos processos, mas a maioria das organizações não orienta explicitamente seus processos de desenvolvimento de software pelo reuso. Contudo é provável que esta abordagem torne-se bastante influente durante o século XXI, uma vez que montar sistemas a partir de componentes reutilizáveis é essencial para o rápido desenvolvimento de software [SOMMERVILLE2003].

Segundo Larman [LARMAN1997], o modelo de desenvolvimento mais usado hoje é o modelo iterativo incremental. Neste modelo um ciclo de vida iterativo se baseia no aumento e no refinamento sucessivo de um sistema através de múltiplos ciclos de desenvolvimento de análise, de projeto, de implementação, e de teste.

O sistema cresce pelo acréscimo de novas funções em cada ciclo de desenvolvimento. Após uma fase preliminar de Planejar e Elaborar, o desenvolvimento continua numa fase Construir através de uma série de ciclos de desenvolvimento.

Cada ciclo trata um conjunto relativamente pequeno de requisitos, procedendo através de análise, de projeto, de implementação e de teste. O sistema cresce incrementalmente, a medida que cada ciclo é completado.

Isso contrasta com o clássico ciclo de vida em cascata, no qual cada atividade (análise, projeto e assim por diante) é executada uma única vez para o conjunto inteiro de requisitos do sistema. As vantagens do desenvolvimento iterativo incluem:

- A complexidade nunca se torna incontrolável.
- A realimentação é gerada cedo no processo, porque a implementação ocorre rapidamente para um pequeno subconjunto do sistema.

O processo adotado como base para o desenvolvimento do projeto é um modelo incremental e fortemente baseado no *Rational Unified Process* da *Rational Corporation* [RATIONALROSE], com a adição de algumas atividades e artefatos pertinentes às recomendações da ISO (*International Standard Organization*), em particular as recomendações incluídas na norma [ISO15504] e ao modelo CMM (*Capability Maturity Model for Software (SW-CMM)V1.1* do SEI (*Software Engineering Institute*). O *Rational Unified Process* (RUP) é um processo de Engenharia de Software desenvolvido pela Rational Software Corporation, cujas principais características são um desenvolvimento iterativo e incremental, orientado a objetos,

com foco na criação de uma arquitetura robusta, análise de riscos e utilização de casos de uso para o desenvolvimento. Seguindo a estrutura do RUP, o projeto será desenvolvido ao longo de 4 fases - concepção, elaboração, construção e transição - cada uma com objetivos específicos:

- Na fase de concepção, deve-se estabelecer o escopo e a viabilidade econômica do projeto. A atividade principal desta fase é:

Modelagem do Negócio - envolve o entendimento da estrutura e dinâmica da organização do cliente, garantindo que clientes, usuários e desenvolvedores tenham a mesma visão da organização para a qual será feito o desenvolvimento.

- Na elaboração, o objetivo é eliminar os principais riscos e estabelecer uma arquitetura estável a partir da qual o sistema poderá evoluir. As principais atividades desta fase são:

Elicitação de Requisitos - envolve a definição dos requisitos do sistema e de como gerenciar escopo e mudanças de requisitos.

Análise e Projeto - envolve a tradução dos requisitos numa especificação que descreve como implementar o sistema.

- Na fase de construção, um produto completo é desenvolvido de maneira iterativa até que esteja pronto para ser passado aos usuários. As atividades desta fase são:

Implementação - envolve o desenvolvimento de código: classes, objetos, etc., teste de unidades e integração de subsistemas.

Teste - envolve a verificação do sistema como um todo, com testes de integração e conformidade com os requisitos especificados.

- A fase de transição é quando o sistema é disponibilizado para o cliente. A principal atividade desta fase é:

Implantação - envolve a instalação do sistema e treinamento de usuários, assim como o planejamento e condução de beta testes².

No capítulo seguinte essas fases e atividades serão detalhadas.

²Testes realizados em cópias beta de um determinado sistema

3.4 UML

Como no desenvolvimento do sistema aqui proposto a linguagem de modelagem utilizada foi a UML, esta seção visa introduzir alguns conceitos relacionados a mesma. As informações a seguir referem-se ao texto de Larman [LARMAN1997].

UML é "uma linguagem para especificar, visualizar, construir os artefatos de sistemas de software..."[BOOCH2000]. Ela é um sistema de notação (incluindo a semântica para suas notações) dirigida à modelagem de sistemas, usando conceitos orientados à objetos.

UML é um padrão emergente, que está sendo aceito pela indústria, para a modelagem orientada a objetos. Ela começou com o esforço conjunto de Grady Booch e Jim Rumbaugh em 1994, para combinar seus dois métodos populares - os métodos Booch e OMT (*Object modeling Technique*). Mais tarde, Ivar Jacobson se juntou a eles (o criador do método OOSE, *Object Oriented Software Engineerig*). Em resposta a uma solicitação da OMG (*Object Management Group*, uma entidade de padronização estabelecida pela indústria) para definir uma linguagem e notação de modelagem padronizada, UML foi submetida como candidata em 1997.

O OMG aceitou UML, a qual também recebeu a aprovação de fato pela indústria, uma vez que seus criadores representam métodos de análise e / ou projeto de primeira geração muito populares. Muitas organizações de desenvolvimento de software e fornecedores de ferramentas CASE adotaram UML, e é muito provável que ela se torne um padrão mundial utilizado por desenvolvedores, autores e fornecedores de ferramentas CASE³.

Ainda segundo Larman [LARMAN1997], "UML" é a abreviação de Linguagem de Modelagem Unificada (*Unified Modeling Language*), uma notação (principalmente diagramática) para modelagem de sistemas, usando conceitos orientados a objetos.

Como deveriam ser alocadas as responsabilidades às classes de objetos? Como objetos deveriam interagir? Quais classes deveriam fazer o quê? Estas são questões críticas no projeto de um sistema. Algumas soluções boas e testadas para problemas de projeto podem ser (e têm sido) expressas como um conjunto de princípios de projeto exemplares.

Vários são os diagramas compreendidos na UML. Durante o desenvolvimento do sistema de análise do custo da pós-colheita do café os diagramas gerados foram: diagrama de casos de uso, diagrama de colaboração e diagrama de classes.

A seguir tem-se uma breve descrição de cada um desses diagramas gerados

³Softwares que auxiliam na modelagem de sistemas

pelo sistema, bem como um exemplo básico para facilitar o entendimento dos mesmos [LARMAN1997].

Definindo casos de uso

Compreender os requisitos inclui, em parte, compreender os processos do domínio e o ambiente externo (fatores externos que participam do processo). Estes processos do domínio podem ser expressos em casos de uso (descrições narrativas dos processos do domínio num formato de prosa estruturada). Casos de uso não são, na realidade, um artefato da análise orientada ao objeto; eles, simplesmente descrevem processos e podem ser igualmente eficazes num projeto que não usa a tecnologia orientada a objetos. Contudo, são um passo preliminar útil na descrição dos requisitos do sistema.

Por exemplo, no jogo de dados, apresentamos a seguir o caso de uso jogar um jogo.

Caso de uso: jogar um jogo

Atores: jogador

Descrição: Este caso de uso começa quando o jogador pega e lança os dados.

Se o total dos dados é sete, os dados vencem; caso contrário, o jogador ganha.

Definindo diagramas de colaboração

O projeto orientado a objetos se preocupa com a definição de especificações lógicas de software que atendem os requisitos funcionais, baseado na decomposição por classes de objetos. Um passo essencial, nesta fase, é a alocação de responsabilidade aos objetos e a ilustração de como eles interagem através de mensagens expressas em diagramas de colaboração. Os diagramas de colaboração mostram o fluxo de mensagens entre instâncias e invocação de métodos.

Por exemplo, suponha que é desejada uma simulação de software do jogo de dados. Este diagrama de colaboração na figura 3.1 ilustra o passo essencial do jogo, através do envio de mensagens a instâncias das classes Jogador e Dado.

Definindo diagramas de classes

Para definir uma classe várias questões devem ser respondidas:

- Como os objetos se conectam a outros objetos?

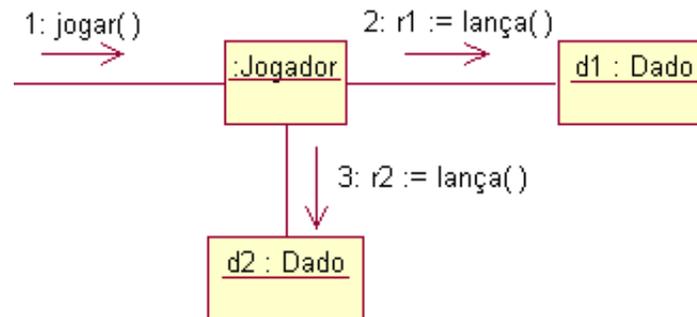


Figura 3.1: Diagrama de colaboração ilustrando mensagens entre objetos de software.

- Quais são os métodos de uma classe?

Para responder estas questões, inspecione os diagramas de colaboração, os quais sugerem as conexões necessárias entre objetos e os métodos que cada classe de software deve definir. Um diagrama que expressa estes detalhes de projeto é o diagrama de classes. Estes ilustram as definições de classes que devem ser implementadas em software.

Por exemplo, no jogo de dados, a inspeção do diagrama de colaboração conduz ao diagrama de classes da figura 3.2. Uma vez que uma mensagem jogar é enviada para uma instância de Jogador, Jogador requer um método jogar, enquanto Dado requer um método lançar.

No capítulo seguinte estes diagramas serão mencionados. Serão mostrados alguns exemplos desses diagramas gerados durante o desenvolvimento do sistema PÓS-CAFÉ.

3.5 Considerações finais

Este capítulo teve como objetivo introduzir alguns conceitos sobre Engenharia de Software. Através do conteúdo apresentado no mesmo percebe-se que, em geral, os engenheiros de software adotam uma abordagem sistemática e organizada em seu trabalho, uma vez que essa é, com frequência, a maneira mais eficaz de produzir software de qualidade. No entanto, a engenharia tem a ver, em grande

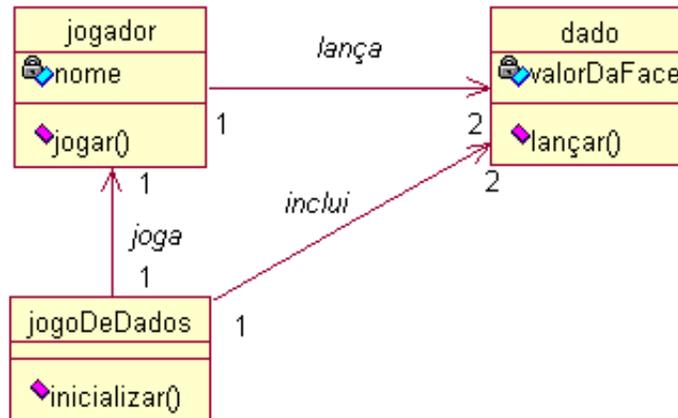


Figura 3.2: Diagrama de classes de projeto de componentes de software.

parte, com a questão de selecionar o método mais apropriado para um conjunto de circunstâncias, e uma abordagem mais criativa e informal para o desenvolvimento pode ser eficaz em algumas circunstâncias.

Com relação à UML, pode-se entender que esta é uma linguagem para modelagem; ela não guia um desenvolvedor em como fazer análise e projeto orientados a objetos, ou qual o processo de desenvolvimento a ser seguido. Conseqüentemente, os metodologistas continuarão a definir métodos, modelos e processos de desenvolvimento para a criação efetiva de sistemas de software, no entanto, agora, podem fazê-lo, usando uma linguagem comum - UML.

No capítulo seguinte será detalhado o processo de desenvolvimento utilizado. As atividades que foram realizadas para a construção do software serão descritas.

Capítulo 4

Metodologia de Desenvolvimento

Após um estudo realizado sobre conceitos e técnicas de Engenharia de Software e Sistemas Especialistas o PÓS-CAFÉ começou a ser desenvolvido.

O sistema foi desenvolvido, no UFLATEC (SWQUALITY), Universidade Federal de Lavras (Ufla). As etapas de desenvolvimento deste Sistema Especialista estão descritas de maneira detalhada nas seções seguintes.

4.1 Escolha da tecnologia

Nesta seção serão mostradas as ferramentas usadas para o desenvolvimento da aplicação, bem como o ambiente de desenvolvimento. A partir da seção 4.2 até a 4.7 serão detalhados os passos de desenvolvimento do sistema PÓS-CAFÉ. A seção 4.8 está ligada a gerência do produto, apresentando as diversas formas de divulgação do mesmo.

4.1.1 Linguagem de programação

A linguagem de programação escolhida para a implementação do sistema de análise do custo da pós-colheita do café foi o *Visual Basic*.

Os motivos que fizeram com que esta linguagem fosse escolhida, foram:

- Linguagem de fácil aprendizado: Uma vez que a maioria da equipe envolvida no desenvolvimento do sistema era uma equipe de pouca experiência

em desenvolvimento de software, foi de grande consideração essa característica da linguagem.

- Suporte a vários padrões gráficos, como jpg, gif, gif animado, pex, tif e gráficos 3D.
- Suporte a múltiplos *DBEngines*.
- Suporte a vários *resource files*.
- Suporte a polimorfismo.
- Suporte a um tipo especial de herança.

4.1.2 Sistema operacional

O sistema foi desenvolvido utilizando o sistema operacional *Windows 2000 Professional (Microsoft)*. A aplicação também foi testada em *Windows 98*, *Windows Me* e *Windows XP*.

Foi escolhido o sistema operacional windows por ser este de grande popularidade, facilitando assim a distribuição e divulgação do sistema.

4.1.3 Linguagem de modelagem

Foi utilizada para a modelagem do sistema PÓS-CAFÉ a linguagem UML (*Unified Modeling Language*), que é uma linguagem gráfica para visualização, especificação, construção e documentação de artefatos de sistemas complexos de software. A UML possui vários diagramas que facilitam na manutenção do projeto além de deixá-lo bem documentado para que alterações futuras possam ser facilmente feitas [BOOCH2000].

Para maiores detalhes sobre UML, veja [UML] e [BOOCH2000].

A modelagem da base de dados foi feita usando o modelo entidade relacionamento (ER), que é um modelo conceitual para modelagem de bancos de dados que utiliza conceitos como entidades, relacionamentos e atributos para modelar os dados. Para maiores informações veja [DATE2000].

Toda a modelagem do sistema, inclusive a modelagem do banco de dados, foi feita utilizando o programa *Rational Rose*.

4.1.4 Banco de dados

Foi escolhido o *Access* como banco de dados do sistema. Um dos principais motivos dessa escolha foi o fato de que utilizando o *Access* não se fazia necessário instalar o banco de dados na máquina do usuário do sistema, bem como o *driver* do *Access* (o *Windows* já possui o drive para o *Access*).

Outro fator importante na escolha do banco de dados é relacionado ao fato do *Access 2003* oferecer um conjunto eficiente de ferramentas que são sofisticadas o suficiente para desenvolvedores profissionais, além disso, são de fácil aprendizagem para novos usuários (fator muito importante, pois a equipe participante do desenvolvimento do sistema, como dito anteriormente, era uma equipe inexperiente em desenvolvimento de software).

4.2 Gerência de configuração

A gerência de configuração foi uma técnica da Engenharia de Software utilizada durante todo o desenvolvimento do sistema PÓS-CAFÉ.

Os três primeiros parágrafos desse item apresentam uma generalização do texto de Sanches [SANCHES2001], a respeito da gerência de configuração de software, para a gerência de configuração de um produto qualquer.

Durante o processo de desenvolvimento de um produto, é produzida uma grande quantidade de itens de informação que podem ser alterados durante o processo.

Para que não haja inconsistência nos itens de informação importantes para o projeto, a criação e alterações desses itens devem ser acompanhadas e controladas pelo gerente do projeto. O processo de gerência de configuração é um processo de ciclo de vida do software que permite que esse controle seja realizado.

Um documento chamado 'Plano de Gerência de Configuração'¹ foi utilizado para garantir a padronização dos nomes de arquivos, classes, e demais artefatos gerados durante o desenvolvimento do projeto.

Uma ferramenta utilizada durante o desenvolvimento do projeto, afim de ajudar na gerência de configuração, foi o FreeVcs [FREEVCS]. Esta ferramenta auxilia organizando os arquivos, na finalidade de manter sempre versões atualizadas em todas as máquinas dos desenvolvedores.

O *FreeVCS* é uma ferramenta que tem como principal função controlar e gerenciar as versões do código fonte e de uma maneira fácil e manejável o desenvolvedor

¹Documento responsável por definir as responsabilidades e interfaces do projeto. Além de descrever como os artefatos do projeto devem ser nomeados, sinalizados e numerados.

irá sempre manter seus códigos fontes atualizados.

Um outro diferencial do *FreeVCS* é que ele mantém histórico de versões, com isso o desenvolvedor pode recuperar facilmente versões antigas do arquivo fonte, além disso ele permite também o cadastro de *BUGs*, lista de tarefas, desenvolvedores, etc.

4.3 Arquitetura do sistema

A arquitetura utilizada, comum para sistemas de informação que incluem uma interface de usuário e o armazenamento persistente de dados, é conhecida como a arquitetura de três camadas, conforme apresentado na figura 4.1. A seguir segue uma breve descrição da mesma:

- Camada de apresentação - camada que interage com o usuário, janelas, relatórios, etc.;
- Camada lógica da aplicação - camada que contém as regras do negócio, fórmulas, cálculos e tarefas necessárias para ajuda na tomada de decisão;
- Camada de armazenamento - camada de armazenamento persistente.



Figura 4.1: Apresentação da arquitetura de três camadas.

A figura 4.2 identifica os principais tipos de classes existentes no sistema PÓS-CAFÉ, oferecendo uma descrição para cada uma delas.

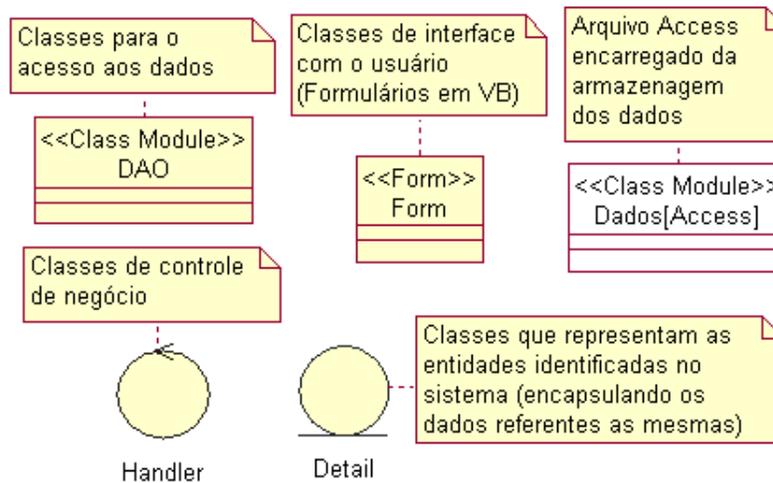


Figura 4.2: Especificação das classes utilizadas no sistema.

4.4 Especificação dos requisitos

Para a especificação dos requisitos do sistema foi utilizado o 'Documento de Requisitos'². Este artefato é comum dentro do UFLATEC (SWQUALITY), e utilizado para o detalhamento dos requisitos de todos os sistemas desenvolvidos na SWQUALITY.

Este documento especifica os requisitos do sistema de análise da pós-colheita do café, fornecendo aos desenvolvedores as informações necessárias para a execução de seu projeto e implementação, assim como para a realização dos testes e homologação³.

Todo conhecimento contido no sistema PÓS-CAFÉ estava inicialmente representado em planilhas *excel*. As fórmulas, índices de cálculo, entradas do produtor,

²Artefato utilizado para especificar os requisitos do sistema. Fornecendo uma visão geral do produto/serviço.

³Confirmação, ratificação, aceitação dos requisitos.

enfim, todas as variáveis envolvidas no cálculo do custo dos diversos processamentos do café.

Os requisitos foram especificados durante reuniões com o Prof. Dr. Flávio Meira Borém e o doutorando Carlos Henrique Rodrigues Reinato. Nestas reuniões o conhecimento contido nas planilhas *excel* era melhor detalhado, e as dúvidas em relação a algumas fórmulas eram sanadas.

A primeira etapa durante o processo de especificação de requisitos foi diferenciar as variáveis envolvidas no sistema. Separando-as em duas classes principais:

- Entradas do produtor: essas entradas representam os dados de entrada referentes às particularidades da produção de café de um produtor para que o custo de um determinado processamento seja calculado. Exemplo de entradas do produtor: quantidade da produção de café em sacas, porcentagem de café cereja, valor do abrigo do lavador, etc.
- Índices de cálculo: este termo designa alguns valores especiais, utilizados em fórmulas de cálculo para o custo dos processamentos, fruto de pesquisas desenvolvidas em relação à pós-colheita do café. Como exemplo de alguns dos índices de cálculo considerados tem-se: espessura do café cereja descascado, redução do volume do café no descascamento, etc. Os índices de cálculo do sistema terão valores máximos e mínimos a eles associados, afim de que, caso o usuário do sistema queira alterar seus valores, o mesmo possa ser instruído quanto aos valores aceitáveis para o índice em questão.

Em um segundo momento analisou-se a distinção entre perfis de usuários. Para isto foram identificados três tipos de usuários no sistema: usuário comum, usuário avançado e o usuário administrador.

- Usuário Comum: esse tipo de usuário tem acesso às funcionalidades do sistema de simulação de custos, mas sem a possibilidade de ajustar ou padronizar os índices de cálculo do sistema. O usuário comum não terá permissão para cadastrar tipos de terrenos, usuários e máquinas agrícolas no sistema.
- Usuário Avançado: esse tipo de usuário tem acesso às funcionalidades básicas do sistema, bem como, permissão para ajustar ou padronizar os índices de cálculo antes pré-estabelecidos. O usuário avançado não terá permissão para cadastrar novos usuários no sistema.
- Usuário Administrador: esse tipo de usuário é responsável por toda a parte administrativa do sistema. O usuário administrador poderá incluir novos

usuários, bem como atribuir perfis aos mesmos. Além disso, o administrador também terá permissão para ajustar ou padronizar os índices de cálculo do sistema. Esse usuário tem acesso a todas as funcionalidades do sistema.

Durante as outras reuniões realizadas as demais funcionalidades do sistema foram identificadas, e detalhadas por meio do 'Documento de Requisitos'.

4.5 Modelagem do sistema

A modelagem do sistema de análise do custo da pós-colheita do café (PÓS-CAFÉ) seguiu o padrão UML.

Os diagramas gerados pela modelagem do sistema foram:

- Diagramas de caso de uso: Descrevem o comportamento do sistema. Especificando os argumentos funcionais que determinarão a macro-arquitetura do sistema. Não participando da descrição da organização interna do software [LARMAN1997].
- Diagramas de colaboração: É muito utilizado para mostrar a interação entre objetos ativos e passivos em sistemas tempo real. Mais de uma execução de tarefa pode ser mostrada no mesmo diagrama. A figura 4.3 mostra um exemplo de um diagrama de colaboração desenvolvido no sistema.
- Diagramas de classes: Um diagrama de classes demonstra todas as classes identificadas para o desenvolvimento do sistema, bem como a interação entre elas. A figura 4.4 mostra o modelo conceitual.

Foi realizada também a modelagem do banco de dados. Esta modelagem pode ser observada com maiores detalhes na figura 4.5.

Além desses diagramas, também foram gerados 'Documentos de Casos de Uso'⁴. Estes documentos descreviam com maiores detalhes os requisitos identificados no documento de requisitos.

⁴Este artefato descreve com maiores detalhes os requisitos do sistema. Fornecendo uma descrição, a sequência de atividades, pré-condições e pós-condições relacionadas ao requisito.

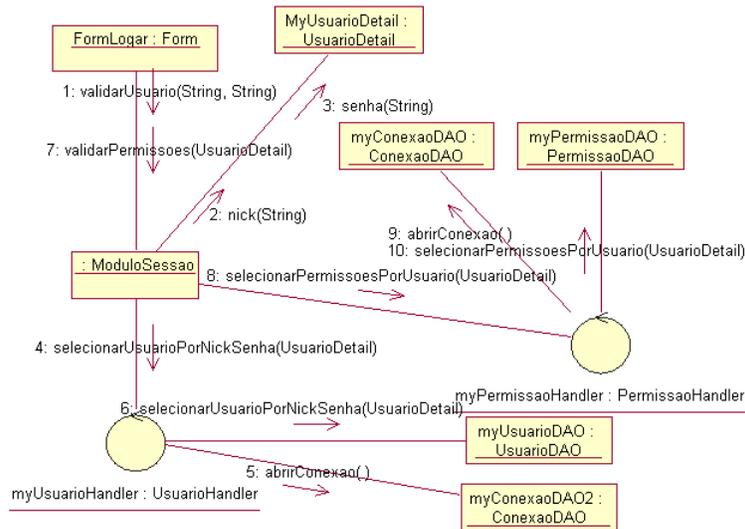


Figura 4.3: Diagrama de colaboração para o requisito 'Logar no sistema'.

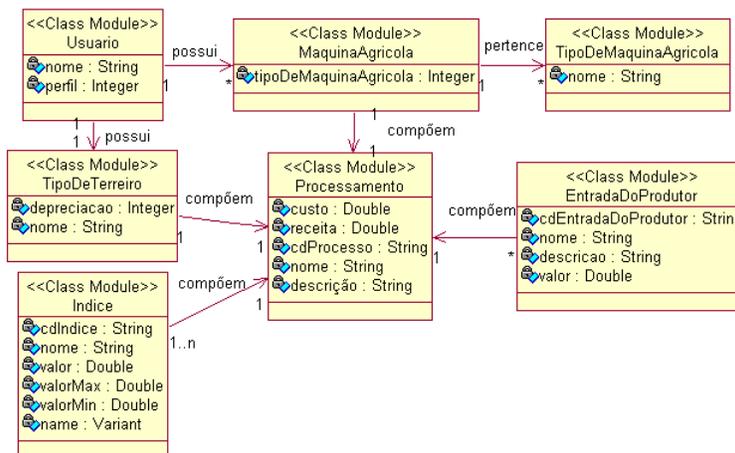


Figura 4.4: Especificação das classes utilizadas no sistema (modelo conceitual).

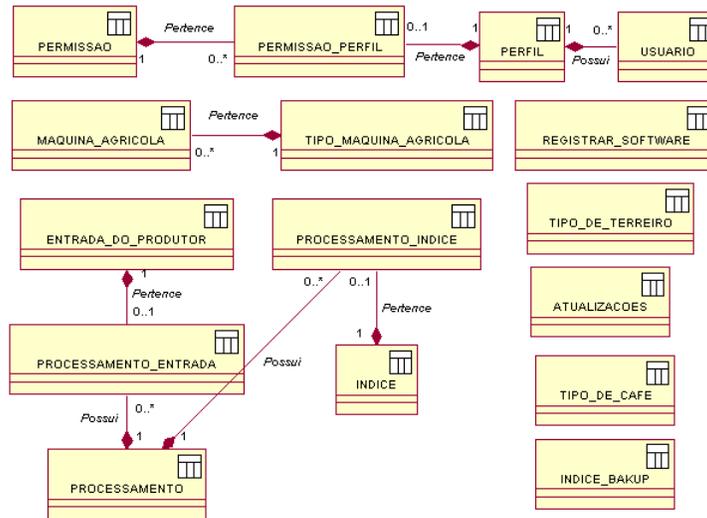


Figura 4.5: Modelagem do banco de dados do sistema.

4.6 Implementação

A implementação do PÓS-CAFÉ, como dito anteriormente, foi realizada em Visual Basic. Foi baseada na modelagem UML, realizada durante a atividade de desenvolvimento anterior. No próximo capítulo as principais funcionalidades do sistema serão melhor detalhadas.

4.7 Verificação

Durante o desenvolvimento do sistema foram realizados dois principais tipos de testes:

- Realizados pelos desenvolvedores do sistema: estes testes dizem respeito às verificações feitas pelos desenvolvedores afim de confirmar se os resultados obtidos pelo sistema correspondiam às especificações dos requisitos, descritos no 'Documento de Requisitos'. Todo estes testes eram documentados por meio do artefato denominado 'Plano de Testes'⁵.

⁵Este artefato tem como intuito documentar os testes realizados sobre um determinado sistema.

- Realizados por produtores de café: alguns produtores, considerados como potenciais auxiliares no desenvolvimento do sistema, receberam cópias do PÓS-CAFÉ, no intuito de que os mesmos retornassem sugestões, melhorias e possíveis alterações no sistema, afim de que o mesmo pudesse corresponder ao máximo possível com a realidade da pós-colheita do café.

4.8 Divulgação e distribuição do sistema

O sistema pós-café foi divulgado de diversas formas, sendo as principais:

- Entrevista com Prof. Flávio Meira Borém na TV Universitária (TVU - UFLA), destacando a importância do sistema para os produtores de café.
- Reportagem no jornal da UFLA.
- Apresentação de artigo relacionado ao sistema PÓS-CAFÉ no 30º congresso brasileiro de pesquisas cafeeiras, de 9 a 12 de novembro / São Lourenço - MG
- Criação de um site, conforme figura 4.6, para o sistema PÓS-CAFÉ. Este site está localizado no endereço <http://www.swquality.com.br/poscafe>.
- Distribuição gratuita de 50 cópias do PÓS-CAFÉ, versão de demonstração (demonstrada na figura 4.7), para produtores de café participantes de congressos relacionados à área.

4.9 Considerações finais

Durante o desenvolvimento do sistema percebeu-se a grande importância da escolha do processo base para o desenvolvimento do projeto. Os processos de desenvolvimento de software são métodos que ajudam na organização das atividades relacionadas com a criação, entrega e manutenção de sistemas de software.

Mais importante que seguir um processo ou método oficial é um desenvolvedor adquirir habilidade de criar um bom projeto, e que organizações sustentem o

Os testes a serem realizados deverão atender a 3 aspectos diferentes, a citar: teste funcional, teste de validação e teste de base de dados.

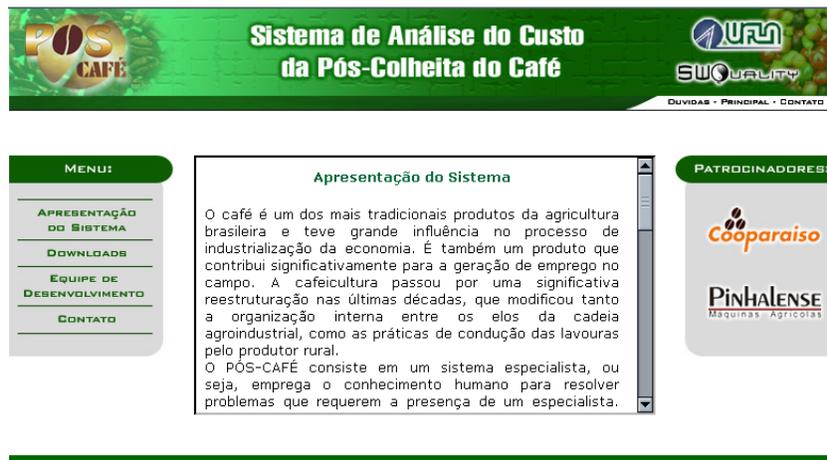


Figura 4.6: Site do sistema.



Figura 4.7: CD correspondente à versão de demonstração do sistema.

desenvolvimento desse tipo de habilidade. Ela surge do domínio de um conjunto de princípios e de heurísticas⁶ relacionadas com a identificação e a abstração de

⁶Regras poucos discutidas, de bom discernimento, que caracterizam a tomada de decisão a nível de especialista na área.

objetos adequados e da atribuição de responsabilidades aos mesmos.

O capítulo seguinte apresentará o sistema de análise do custo da pós-colheita do café (PÓS-CAFÉ). Uma breve descrição do sistema e um detalhamento de suas principais funcionalidades serão mostrados.

Capítulo 5

O Sistema PÓS-CAFÉ

O PÓS-CAFÉ consiste em um Sistema Especialista, ou seja, emprega o conhecimento humano para resolver problemas que requerem a presença de um especialista. Também foram usadas técnicas de Engenharia de Software para que o programa pudesse ser facilmente entendido por quem quiser aprimorá-lo ou aproveitar algo do seu desenvolvimento. Dentre as técnicas de Engenharia de Software utilizadas destacam-se: metodologias de desenvolvimento e o processo de desenvolvimento.

Este sistema se aplica à fase de pós-colheita do café e oferece informações de custo sobre alguns dos processamentos envolvidos, como: descascamento, despulpamento, desmucilamento, separação e secagem do café. O conhecimento por meio do qual o sistema foi desenvolvido é fruto de pesquisas realizadas por professores e alunos da Universidade Federal de Lavras (UFLA) na área de pós-colheita do café, tendo como intuito avaliar o custo da aplicação dos processamentos citados acima.

A principal contribuição do PÓS-CAFÉ, a partir da modelagem e sistematização desse conhecimento, é facilitar a tomada de decisão por parte dos produtores, cooperativas e consultores da área, quanto aos processamentos a serem aplicados durante a fase de pós-colheita do café. Esta decisão seria facilitada à medida que se conhecesse o custo dos processos de separação, descascamento, despulpamento e desmucilamento do café.

A idéia básica do sistema é possibilitar que o usuário escolha alguns dos possíveis processamentos do café para que seja realizada uma simulação de custos. Após a escolha dos processamentos, o usuário deve fornecer os dados relaciona-

dos à produção, de acordo com os formulários de entrada de dados gerados.

O sistema PÓS-CAFÉ processará e retornará relatórios que relacionam o custo dos processamentos e a margem líquida (valor em dinheiro, por saca de café, que o produtor irá ganhar ou perder - custo/benefício - quando o mesmo escolher por um tipo de processamento do café em relação a outro tipo) entre os mesmos. A partir dos relatórios informativos gerados pelo sistema o produtor terá facilitada a sua tomada de decisão quanto aos processamentos a serem aplicados a sua produção.

A figura 5.1 apresenta uma visão geral do sistema pós-café. Nela são identificados os principais módulos do sistema, bem como a interação entre os mesmos.

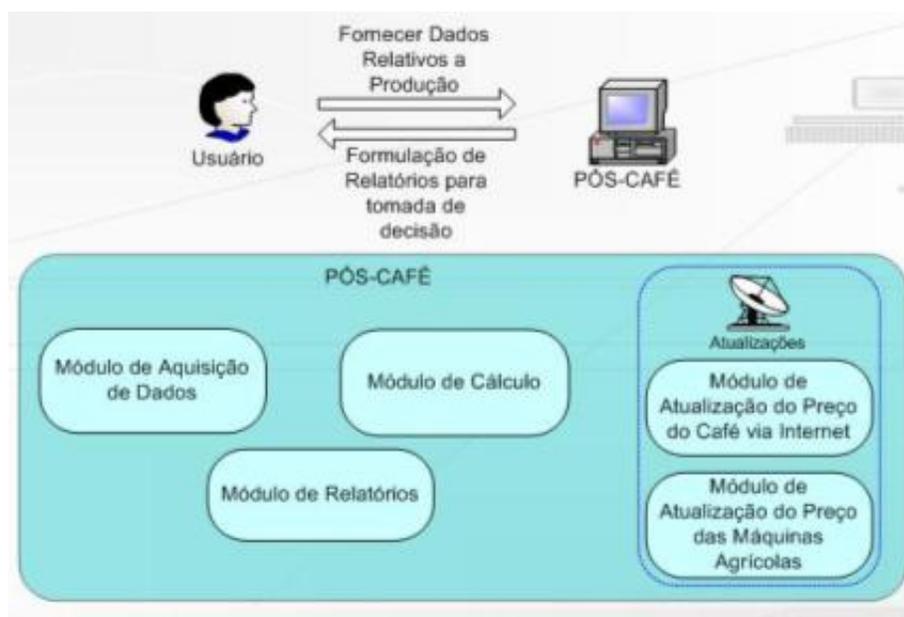


Figura 5.1: Visão geral do sistema PÓS-CAFÉ.

O módulo de aquisição de dados do sistema PÓS-CAFÉ recebe de um determinado usuário os dados relacionados às particularidades de sua produção como: quantidade em sacas, tempo de colheita em dias, etc.

Os dados que o produtor deverá preencher para realizar as simulações serão dependentes dos processamentos escolhidos pelo mesmo, ou seja, segundo os processamentos escolhidos será gerado um formulário de entrada de dados para o usuário, considerando os atributos importantes em relação aos processamentos.

O módulo de cálculo é encarregado de efetuar os cálculos necessários, baseado em índices, frutos de pesquisas, a fim de retornar o custo de cada um dos processos escolhidos (lavagem, descascamento, desmucilamento e despulpamento) e a margem líquida entre os mesmos.

O módulo de relatórios mostra os resultados relacionados ao custo dos processamentos calculados, bem como as margens líquidas entre os processamentos, apresentando-as graficamente, de forma a facilitar à compreensão dos usuários do sistema.

As atualizações dos valores de preço do café e de características das máquinas agrícolas poderão ser feitas através da Internet, a partir de bancos de dados atualizados.

O módulo de atualizações do preço do café tem como intuito atualizar os preços do café segundo as cotações da saca de café diárias.

Poderão ser realizadas atualizações das máquinas agrícolas, mas essas atualizações serão feitas apenas nas máquinas agrícolas cadastradas no servidor (computador) do sistema, ou seja, atualizações relativas às máquinas agrícolas pertencentes às empresas parceiras.

Na figura 5.2 pode ser visualizada a tela principal do sistema PÓS-CAFÉ.

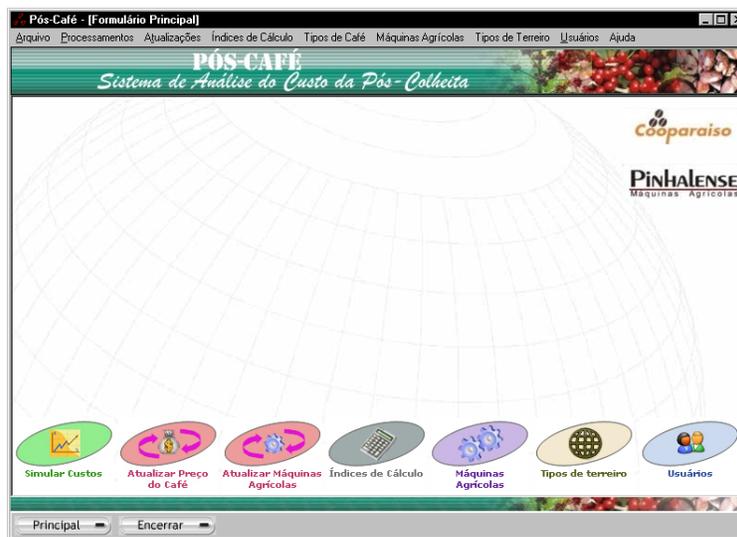


Figura 5.2: Tela principal do sistema PÓS-CAFÉ.

5.1 Principais funcionalidades

Nesta seção serão abordadas as principais funcionalidades do sistema PÓS-CAFÉ. Algumas telas do sistema serão mostradas para facilitar o entendimento das funcionalidades.

5.1.1 Calcular o custo dos processamentos

O calculo do custo dos processamentos compreende as seguintes etapas:

Primeiro o usuário do sistema deverá escolher os processamentos com os quais deseja efetuar as simulações. A figura 5.3 mostra a tela do sistema onde esta escolha dos processamentos será realizada.

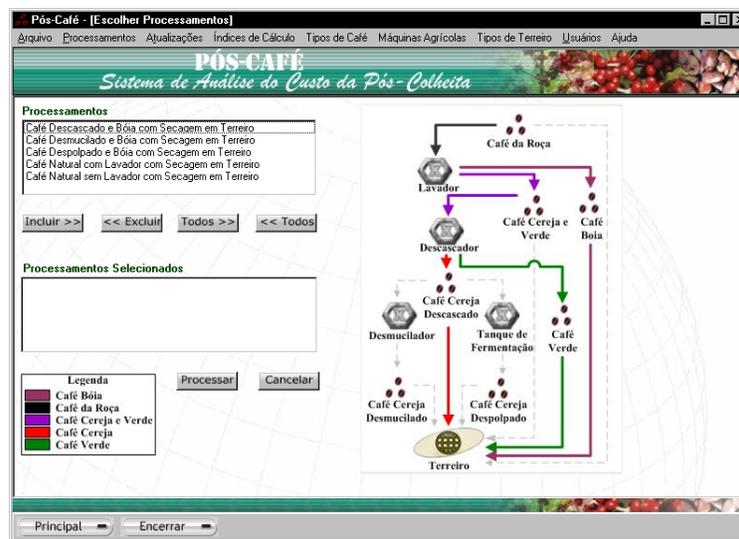


Figura 5.3: Tela para escolher os processamentos do café.

A seguir o usuário deverá fornecer alguns dados referentes à sua produção. Estes dados deverão ser informados no formulário de entrada padrão (tela do sistema com dados a serem informados, comuns a todos os processamentos), mostrado na figura 5.4, e no formulário de entrada específica (tela do sistema com dados a serem informados, específicos para cada processamento), mostrado na figura 5.5.

A partir de todas as informações fornecidas pelo usuário, o sistema está pronto

PÓS-CAFÉ - [Formulário de Entrada Padrão]

Arquivo Processamentos Atualizações Índices de Cálculo Tipos de Café Máquinas Agrícolas Tipos de Terreiro Usuários Ajuda

PÓS-CAFÉ
Sistema de Análise do Custo da Pós-Colheita

Processamentos Selecionados
 Café Descascado e Bóia com Secagem em Terreiro
 Café Desmucilado e Bóia com Secagem em Terreiro
 Café Despolpado e Bóia com Secagem em Terreiro
 Café Natural com Lavador com Secagem em Terreiro
 Café Natural sem Lavador com Secagem em Terreiro

Data das Últimas Atualizações
 Preço do Café: 6/12/2004 Fonte: Cooperaiço
 Máquinas Agrícolas: 24/11/2004 Fonte: Pinhalense

Entradas do Produtor
 Produção em Sacas: 1000
 Tempo de Colheita (Dias): 90
 Porcentagem de Café Cereja (%): 30
 Porcentagem de Café Verde (%): 15
 Tempo Máximo Disponível para o Funcionamento das Máquinas Agrícolas (Horas/Dia): 4
 Redução no Volume do Café Cereja após o Descascamento: 40
 Desbloquear

Valor da Saca do Café Padrão
 Café Padrão (R\$): 250
 Clique no botão para obter o valor da saca do B/C, tipo 6, bebida dura. Atualizar

Máquinas Agrícolas
 Lavador: Escolher
 Descascador: Escolher
 Desmucilador: Escolher

Tipos de Café (%)

Tipos de Café (%)	Ágio I Deságio
Café da Roça: 4	<input type="radio"/> <input type="radio"/>
Café Verde: 10	<input type="radio"/> <input type="radio"/>
Café Bóia: 0	<input type="radio"/> <input type="radio"/>
Café Cereja e Verde: 0	<input type="radio"/> <input type="radio"/>
Café Cereja Descascado: 15	<input type="radio"/> <input type="radio"/>
Café Cereja Desmucilado: 15	<input type="radio"/> <input type="radio"/>
Café Cereja Despolpado: 15	<input type="radio"/> <input type="radio"/>

Tanque de Fermentação
 Valor (R\$): 5000 Tanques para o processamento do café cereja despolpado.
 Vida Útil (Anos): 15

Volume de Café (Litros)
 Volume de Café por Saca: 480
 Volume de Café Bóia por Saca: 410
 Volume de Café Verde por Saca: 480

Energia Elétrica
 Valor da Tarifa (R\$/KWh): 0,32

Trabalhadores
 Valor do Dia Homem (R\$): 25

Selecione o Tipo de Terreiro
 Tipo de Terreiro: Terreiro de lama asfáltica
 Custo nº: R\$ 2,62
 Cadastrar Alterar

Confirmar Cancelar

Principal Encerrar

Figura 5.4: Tela padrão de entrada de dados.

para efetuar o cálculo dos custos. Todo o cálculo realizado está baseado no conhecimento contido no sistema PÓS-CAFÉ (por isso denominado Sistema Especialista).

5.1.2 Gerar gráfico comparativo entre os processamentos

A partir dos dados de entrada informados pelo usuário do sistema, serão simuladas várias situações possíveis durante a fase de pós-colheita do café, as quais tem o objetivo de, oferecendo uma visão geral das possibilidades, facilitar a tomada de decisão por parte do usuário do sistema.

O gráfico apresenta os diferentes valores da margem líquida calculados. Os outros eixos do gráfico representam: porcentagem de cereja (quantidade de café cereja existente na lavoura) e porcentagem do ágio (relação, em porcentagem, correspondente à diferença de preço entre as sacas de maior valor dos dois processamentos analisados). Além da porcentagem de verde (quantidade de café verde existente na lavoura) que também é considerada para o cálculo da margem líquida.

A figura 5.6 mostra como estas informações são apresentadas ao usuário do sistema.

Pós-Café - [Formulário de Entrada Específica]

Arquivo Processamentos Atualizações Índices de Cálculo Tipos de Café Máquinas Agrícolas Tipos de Terreiro Usuários Ajuda

PÓS-CAFÉ

Sistema de Análise do Custo da Pós-Colheita

Café Descascado e Bóia com Secagem em Terreiro

Tempo de Secagem (Dias)		Espessura do Café (Metros)		Mão-de-Obra (Número de Funcionários)	
Café da Floça	<input type="text" value="15"/>	Café da Floça	<input type="text" value="0,05"/>	Lavador	<input type="text" value="1"/>
Café Verde	<input type="text" value="18"/>	Café Verde	<input type="text" value="0,1"/>	Lavador e Descascador	<input type="text" value="1"/>
Café Bóia	<input type="text" value="8"/>	Café Bóia	<input type="text" value="0,05"/>	Lavador e Desmucilador	<input type="text" value="1"/>
Café Cereja e Verde	<input type="text" value="18"/>	Café Cereja e Verde	<input type="text" value="0,05"/>	Lavador e Tanque de Fermentação	<input type="text" value="1"/>
Café Cereja Descascado	<input type="text" value="8"/>	Café Cereja Descascado	<input type="text" value="0,03"/>	Número de funcionários trabalhando no processamento do café:	
Café Cereja Desmucilado	<input type="text" value="8"/>	Café Cereja Desmucilado	<input type="text" value="0,03"/>	Trabalhando no lavador, descascador e distribuindo o café.	
Café Cereja Despolpado	<input type="text" value="8"/>	Café Cereja Despolpado	<input type="text" value="0,03"/>		

Abriço
 Valor (R\$) Local de armazenamento das máquinas agrícolas envolvidas no processamento do café.
 Vida Útil (Anos)

Principal Encerrar

Figura 5.5: Tela específica de entrada de dados.

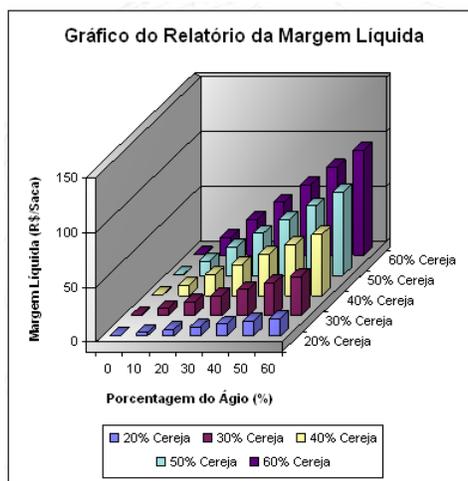


Figura 5.6: Tela correspondente ao gráfico gerado pelo sistema.

5.1.3 Atualizar dados das máquinas agrícolas e o preço do café

As atualizações dos valores do preço do café e das características das máquinas agrícolas poderão ser realizadas através da Internet, a partir de bancos de dados atualizados.

A principal finalidade dessa funcionalidade é a de garantir ao usuário do sistema dados mais atuais possíveis, tanto de preço do café quanto de características das máquinas agrícolas.

Estes dados serão mantidos em um banco de dados do sistema PÓS-CAFÉ, e serão atualizados periodicamente pelos patrocinadores do sistema.

5.1.4 Atualizar versão do sistema

Esta funcionalidade tem o sentido de atualizar a versão do sistema de um determinado usuário do sistema, por uma versão mais nova disponibilizada.

A principal finalidade desta funcionalidade é de possibilitar que os usuários do sistema possam ter acesso a possíveis correções feitas no sistema, bem como, a possíveis novas funcionalidades adicionadas ao sistema.

Desta forma, garante-se que, mesmo sendo encontrado algum erro no sistema, este pode ser solucionado e sua solução pode ser aproveitada por todos os usuários do sistema.

5.1.5 Controlar chaves de registro

O objetivo desta funcionalidade é a de garantir que apenas um usuário possa estar utilizando uma chave do sistema disponibilizada, evitando assim a pirataria do sistema.

Cada usuário do sistema possui uma chave para utilização do mesmo. Esta chave é única para cada usuário. Assim, qualquer um que tentar instalar o PÓS-CAFÉ pertencente a uma outra pessoa, terá seu sistema bloqueado, e será impedida de utilizar o PÓS-CAFÉ em seu computador.

5.1.6 Gerar instalador do sistema

Para a geração do instalador do sistema PÓS-CAFÉ foi utilizado o programa de geração de instalador denominado *Inno Setup* [INNOSETUP]. Todas as bibliotecas (dll's) e os componentes utilizados no sistema foram verificados e incluídos no instalador do sistema.

O instalador gerado abrange as seguintes versões do *Windows*: *Windows 98*, *Windows Me*, *Windows 2000* e *Windows XP*.

5.1.7 Módulo de ajuda

O módulo de ajuda do sistema funciona como um manual de utilização do sistema. Todas as funcionalidades do sistema estão detalhadamente explicadas neste módulo.

A ajuda do sistema PÓS-CAFÉ é sensível ao contexto, ou seja, para cada tela do sistema em que o usuário se encontra existe uma ajuda específica para auxiliar o mesmo. A figura 5.7 mostra os tópicos de ajuda existentes neste módulo.



Figura 5.7: Tópicos correspondentes ao módulo de ajuda do sistema.

5.2 Considerações finais

Este capítulo apresentou o sistema PÓS-CAFÉ e suas principais funcionalidades. A partir do mesmo consegue-se ter uma idéia geral do que é, na verdade, o sistema de análise do custo da pós-colheita do café (PÓS-CAFÉ).

O desenvolvimento do PÓS-CAFÉ foi um trabalho proposto pelo professor Flávio Meira Borém, devido à inexistência, exposta por ele, de programas que possibilitassem uma análise do custo dos processamentos do café durante a fase de pós-colheita.

Capítulo 6

Conclusões e Trabalhos Futuros

Os Sistemas Especialistas são ferramentas que permitem a distribuição do conhecimento, fazendo com que as pessoas, não especialistas em determinados assuntos, possam ser beneficiadas através do conhecimento de um especialista da área. Essa característica dos SE's, sem dúvida, é um dos principais benefícios oferecidos pelos mesmos.

A aplicação das técnicas da Engenharia de Software tais como metodologias de desenvolvimento e o processo de desenvolvimento, foram de suma importância na construção do sistema PÓS-CAFÉ, garantindo que o sistema seja completamente implementado e que atenda os requisitos propostos.

Os resultados obtidos através da utilização do Sistema Especialista PÓS-CAFÉ correspondem satisfatoriamente com a realidade do custo dos diferentes processamentos pelos quais o café pode passar durante sua fase de pós-colheita.

Versões de demonstração do sistema já foram distribuídas. Entretanto existe a proposta de melhorar ainda mais este sistema, incluindo no mesmo, o módulo de secagem mecânica do café. Assim, praticamente todos os processamentos pelos quais o café pode passar em sua fase de pós-colheita seriam abordadas pelo sistema.

Outra proposta de trabalho futuro é o desenvolvimento de um modelo de especificação de requisitos que considere toda dificuldade encontrada durante a fase de aquisição do conhecimento para um SE. Um modelo que possa facilitar este processo, fazendo com que o especialista e o desenvolvedor do SE possam falar a mesma língua.

Referências Bibliográficas

- [BAUMOL1991] BAUMOL, W., BLACKMAN, S., WOLF, E. *Productivity and American*. Cambridge, 1991.
- [BOEHM1988] BOEHM, B. *A Spiral Model for Software Development and Enhancement*, *Computer*, vol. 21, n. 5 1988
- [BOOCH2000] BOOCH, G; RUMBAUGH J.; JACOBSON, I. *UML, guia do usuário*. Rio de Janeiro, Campus, 2000.
- [BORÉM2004] BORÉM, F. M. *Processamento do café*. Lavras, 2004.
- [DATE2000] DATE, C. J. *Introdução a Sistemas de Bancos de Dados*. Rio de Janeiro, Campus, 2000.
- [FREEVCS] FreeVCS. <http://www.freevcs.org>. Visitada em 10/08/2004.
- [INNOSETUP] Inno Setup. <http://www.jrsoftware.org/isinfo.php>. Visitada em 08/07/2004.
- [ISO15504] ISO/IEC TR 15504 *Information Technology, Software Process Assessment* 1998
- [LARMAN1997] LARMAN, C. *Applying UML and Patterns, An Introduction to Object-Oriented Analysis and Design*. New Jersey, Prentice Hall, 1997.
- [PONDÉ1997] PONDÉ, J. *Competitividade da indústria de software*. Campinas, 1993.

- [PRESSMAN1997] PRESSMAN, R. S. *Software engineering: a practitioner's approach*. Nova York, 1997
- [RABUSKE1995] RABUSKE, R. A. *Inteligência artificial*. Florianópolis, UFSC, 1995.
- [RATIONALROSE] Rational Rose <http://www.rationalrose.com>. Visitada em 03/10/2004.
- [ROYCE1970] ROYCE, W. *Managing the Development of Large Software Systems*. 1970
- [RUP2002] Rational Unified Process 2002
- [RUSSEL1995] RUSSEL, S. J.; NORVING, P. *Artificial intelligence, A modern approach*. New Jersey, Prentice Hall, 1995.
- [SANCHES2001] SANCHES R. *Gerência de configuração*. São Paulo: Prentice Hall, 2001b. p. 59-64.
- [SOMMERVILLE2003] SOMMERVILLE, I. *Engenharia de Software*. 2003
- [UML] Object Management Group. <http://www.uml.org>. Visitada em 20/11/2004.
- [WATERMAN2002] WATERMAN, D. A. *A Guide to Expert System*. 1986