

**UNIVERSIDADE FEDERAL DE SANTA CATARINA**  
**PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO**

**PROGRAMA COMPUTACIONAL BASEADO EM CONCEITOS FINANCEIROS E**  
**CONTÁBEIS PARA O GERENCIAMENTO DE PEQUENAS EMPRESAS**

Dissertação submetida à Universidade Federal de Santa Catarina  
para a obtenção do Grau de Mestre em Ciência da Computação

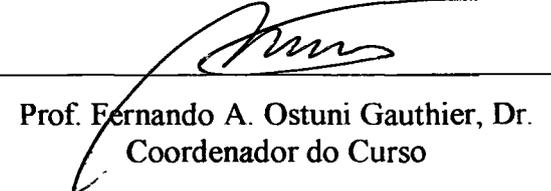
**MAXIMILIANO Z. PEZZIN**

Florianópolis, junho de 2001

MAXIMILIANO Z. PEZZIN

PROGRAMA COMPUTACIONAL BASEADO EM CONCEITOS FINANCEIROS E  
CONTÁBEIS PARA O GERENCIAMENTO DE PEQUENAS EMPRESAS

Esta Dissertação foi julgada adequada para obtenção do Título de "Mestre em Computação", Especialidade em Ciência da Computação e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação

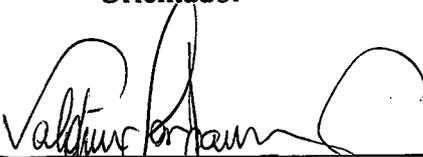


Prof. Fernando A. Ostuni Gauthier, Dr.  
Coordenador do Curso

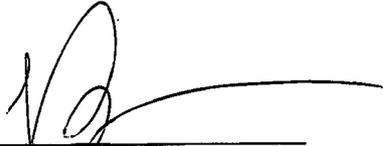
Banca Examinadora:



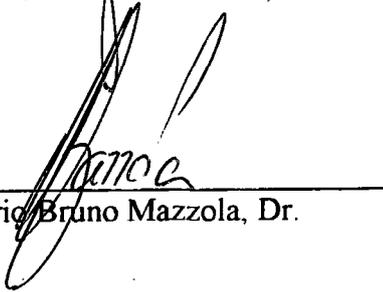
Prof. Antônio-Cezar Bornia, Dr.  
Orientador



Profa. Valdirene Gasparetto, M Eng.  
(Co-orientadora)



Prof. João Bosco da Mota Alves, Dr.



Prof. Vitorio Bruno Mazzola, Dr.

## DEDICATÓRIA

Dedico este trabalho a Deus, pela graça da vida e "perfeição" a mim concedidas, e à minha esposa, pelo carinho e esforço dispensados à realização dos meus estudos

## AGRADECIMENTOS

Ao Prof. Dr. Antônio Cezar Bornia, pelo acompanhamento e orientação neste trabalho.

Agradeço ainda à Doutoranda Valdirene Gasparetto, pelo apoio e auxílio no desenvolvimento.

À minha família, e em especial à minha esposa, Norah, pelo constante apoio e incentivo.

À UnC-Concórdia, pelo auxílio financeiro concedido a minha especialização.

A todos os funcionários do IME, que com o seu trabalho contribuíram, direta ou indiretamente, para a realização deste trabalho.

E principalmente a DEUS, por Ter criado todos os acima citados, aos quais reitero meus agradecimentos mais sinceros.

## SUMÁRIO

ÍNDICE DE FIGURAS .....	VIII
ÍNDICE DE TABELAS .....	IX
ÍNDICE DE QUADROS .....	XI
LISTA DE ABREVIATURAS .....	XII
RESUMO .....	XIII
ABSTRACT .....	XIV
<b>CAPÍTULO 1 - Introdução</b> .....	<b>1</b>
1.1 Considerações Iniciais .....	1
1.2 Objetivos .....	3
1.2.1 Objetivo Geral .....	3
1.2.2 Objetivos Específicos .....	3
1.3 Justificação .....	4
1.4 Motivação ao Desenvolvimento .....	4
1.5 Limitações do Trabalho .....	5
1.6 Estrutura da Pesquisa .....	5
<b>CAPÍTULO 2 – Projeto e Desenvolvimento de Software</b> .....	<b>6</b>
2.1 Sistemas de Informação (SI) .....	6
2.1.1 Tecnologia de informações (TI) .....	7
2.1.2 A engenharia da informação (EI) .....	8
2.2 Interface Homem Computador .....	9
2.2.1 As técnicas de Racionalização da interface homem-computador .....	10
2.2.2 Psicologia cognitiva e semiótica .....	11
2.2.3 Análise ergonômica .....	11
2.2.4 Tendências para as novas interfaces .....	12
2.3 Engenharia e Arquitetura de Software .....	14
2.3.1 Engenharia de software (ES) .....	14
2.3.1.1 O ciclo de vida do software .....	14
2.3.1.2 Cielô de vida clássicô (Cascatã) .....	14
2.3.1.3 O ciclo de vida em espiral .....	15
2.3.1.4 O Gerenciamento das falhas do software .....	16
2.3.2 Qualidade de software .....	18
2.3.3 Gerência do desenvolvimento de software .....	20
2.3.4 O desenvolvimento de pequenos projetos .....	23
2.3.5 A prototipação do software .....	23
2.3.6 O modelo SEI CMM .....	24
2.3.7 Personal Software Process (PSP) .....	25
2.3.8 Metodologias de desenvolvimento .....	26
2.3.9 A unificação dos módulos funcionais .....	27
2.3.10 Metodologia de testes de software .....	27
2.3.11 Gerenciamento de configuração .....	29
2.3.12 O Contexto de desenvolvimento dos programas atuais .....	30
2.4 Banco de Dados .....	30
2.4.1 Conceitos de banco de dados .....	30
2.4.2 Representação física do banco de dados .....	31

2.4.3	Visões do banco de dados .....	32
2.4.4	A estrutura interna do Sistema de Banco de Dados (SBD) .....	33
2.4.5	As Vantagens do BD perante à arquitetura tradicional .....	33
2.4.6	Normalização de dados .....	34
2.4.7	A integridade dos dados .....	36
2.4.8	O modelo de dados relacional .....	36
2.4.9	O SQL .....	37
<b>CAPÍTULO 3 - Informações Contábeis e Financeiras nas Pequenas Empresas .....</b>		<b>39</b>
3.1	Noções de Contabilidade Básica .....	39
3.1.1	Relatórios contábeis e financeiros .....	39
3.1.1.1	Balanço Patrimonial (BP) .....	39
3.1.1.2	Demonstração de Resultado do Exercício (DRE) .....	41
3.1.1.3	Demonstração do Fluxo de Caixa (DFC) .....	42
3.1.2	Os regimes de contabilidade .....	43
3.1.3	O plano de contas .....	44
3.1.4	A metodologia das partidas dobradas .....	45
3.1.5	Os estoques .....	46
3.2	A gerência da informação nas pequenas empresas .....	47
3.3	A contabilidade na gerência da pequena empresa .....	48
3.4	O Pequeno empresário e o contabilista .....	49
3.5	O Apoio computacional para o gerenciamento da pequena empresa ..	51
3.5.1	Informatização nas pequenas empresas .....	51
3.5.2	Características do software voltado para a gestão .....	51
3.5.3	O software indicado para a pequena empresa .....	53
<b>CAPÍTULO 4 – Projeto Lógico Proposto .....</b>		<b>55</b>
4.1	Definição dos Objetivos .....	55
4.2	Definição das Metodologias Utilizadas .....	55
4.2.1	A metodologia de desenvolvimento e a sua justificativa .....	55
4.2.2	Descrição dos passos de desenvolvimento .....	55
4.2.3	Fases de desenvolvimento .....	55
4.3	Descrição dos Módulos Funcionais .....	56
4.3.1	Senha e entrada .....	56
4.3.1.1	Escolha de usuário e senha .....	57
4.3.1.2	Sistema de backup e compactação .....	57
4.3.1.3	Definição da empresa e do BD padrão .....	57
4.3.2	Cadastro de clientes .....	58
4.3.3	Cadastro de fornecedores .....	59
4.3.4	Cadastro de códigos .....	59
4.3.4.1	Codificação financeira .....	59
4.3.4.2	Plano de contas contábil .....	60
4.3.5	Controle financeiro .....	61
4.3.5.1	Controle de contas financeiras – (Financeiro – Contábil) .....	61
4.3.5.2	Controle de caixa e movimentações .....	64
4.3.5.3	Controle de banco e movimentações .....	66
4.3.5.4	Controle de cheques emitidos e pré-datados .....	68
4.3.6	Relatórios .....	69

4.3.6.1	Relatórios básicos (financeiros)	72
4.3.6.2	Relatórios totalizados (financeiros)	72
4.3.6.3	Relatórios da tabela cruzada (financeira)	72
4.3.6.4	Relatório mensal de totalizações financeiras	73
4.3.6.5	Relatórios contábeis	73
4.3.6.6	Relatórios de utilização do programa	75
4.3.6.7	Modelo de restrições utilizado	76
4.3.7	Gerenciamento de Estoques	76
4.3.7.1	Controle de estoques	76
4.3.7.2	As Notas fiscais de entrada e de saída	78
4.3.8	Cadastro de eventos e dados contábeis	79
4.3.9	Configurações do programa	83
4.3.9.1	Cadastro de usuários e permissões	83
4.3.9.2	Definição de nível de complexidade dos módulos do programa	84
4.3.9.3	Configuração do programa	84
4.3.9.4	Reinicialização de banco de dados	84
4.3.9.5	Manutenção do banco de dados	84
4.3.10	Ajudas	85
4.3.10.1	Ajudas por Tópicos	85
4.3.10.2	Atualizando o Programa	86
4.3.10.3	Registrando o Programa	86
4.4	Requisitos do Programa	86
4.4.1	Requisitos de hardware	86
4.4.2	Requisitos de software	87
4.4.3	Softwares auxiliares	87
4.5	Diagrama de Fluxo de Dados (DFD)	87
4.5.1	Diagrama de fluxo de dados de contexto	88
4.5.2	Diagrama de fluxo de dados de primeiro Nível	89
4.6	A Ferramenta de Desenvolvimento	90
4.7	Desenvolvimento do Protótipo	90
4.7.1	Considerações do desenvolvimento	90
4.7.2	Quanto ao cumprimento das metas	90
4.7.3	Quanto aos testes realizados	91
4.8	Considerações Finais	92
<b>CAPÍTULO 5 – Conclusões e Recomendações</b>		<b>93</b>
5.1	Conclusões	93
5.2	Recomendações para trabalhos futuros	94
<b>REFERÊNCIAS BIBLIOGRÁFICAS</b>		<b>95</b>
<b>BIBLIOGRAFIA</b>		<b>97</b>
<b>ANEXO 1 – Principais telas do programa</b>		<b>98</b>

## ÍNDICE DE FIGURAS

Figura 1 – O modelo da engenharia da informação .....	8
Figura 2 – Os componentes da engenharia da informação .....	9
Figura 3 - A problemática da comunicação entre o usuário e o computador .....	10
Figura 4 - O ciclo de vida clássico dos sistemas de <i>software</i> .....	14
Figura 5 – O ciclo de vida espiral .....	16
Figura 6 – O índice desejável de falhas do <i>software</i> no tempo .....	17
Figura 7 - O índice real de falhas do <i>software</i> no tempo .....	17
Figura 8 – Gráfico comparativo do esforço no desenvolvimento de <i>software</i> .....	20
Figura 9 – Estrutura interna do banco de dados relacionais e a visão do usuário .....	31
Figura 10 – Visões do banco de dados .....	32
Figura 11 – Os módulos do protótipo desenvolvido .....	70
Figura 12 – DFD de contexto do programa proposto .....	88
Figura 13 – DFD de primeiro nível do programa proposto .....	89
Figura 14 – Tela de Senhas e Entrada .....	98
Figura 15 – Tela do Menu .....	99
Figura 16 – Tela de cadastros de clientes / fornecedores em complexidade máxima ....	99
Figura 17 – Tela de cadastros de clientes / fornecedores em complexidade mínima ....	100
Figura 18 - Tela de controle financeiro em complexidade máxima .....	101
Figura 19 – Tela de controle Financeiro em complexidade mínima .....	101
Figura 20 – Tela de controle de caixa em complexidade mínima .....	102
Figura 21 – Tela de controle de caixa em complexidade máxima .....	102
Figura 22 – Tela de Relatórios Totalizados .....	103
Figura 23 – Tela de Relatórios Contábeis .....	104
Figura 24 – Tela de Cadastro de Usuários e Permissões .....	104

## ÍNDICE DE TABELAS

Tabela 1 – Tendência atuais e futuras dos métodos computacionais .....	13
Tabela 2 – Características dos <i>softwares</i> atuais .....	53
Tabela 3 – Tabela de senhas e permissões .....	57
Tabela 4 - Tabela de clientes e fornecedores .....	58
Tabela 5 – Tabela de códigos financeiros .....	60
Tabela 6 – Tabela de códigos contábeis .....	61
Tabela 7 – Tabela de movimentações financeiras .....	62
Tabela 8 – Tabela de movimentações contábeis de crédito .....	63
Tabela 9 – Tabela de movimentações contábeis de débito .....	64
Tabela 10 – Tabelas de caixas .....	65
Tabela 11 – Tabelas de movimentações de caixa .....	65
Tabela 12 – Tabela de saldos diários de caixa .....	66
Tabela 13 – Tabela das contas bancárias .....	67
Tabela 14 – Tabela de movimentações bancárias .....	67
Tabela 15 – Tabela de saldos diários de bancos .....	68
Tabela 16 – Tabela de cheques emitidos .....	68
Tabela 17 – Tabela de padrões de relatórios financeiros .....	70
Tabela 18 – Tabelas de unidades dos cenários .....	71
Tabela 19 – Tabelas de grupos dos cenários .....	72
Tabela 20 – Tabelas de contas dos cenários .....	72
Tabela 21 – Tabelas de padrões de relatórios contábeis .....	75
Tabela 22 – Tabelas de padrões de relatórios contábeis - movimentos .....	75
Tabela 23 – Tabela de acontecimentos do programa .....	75
Tabela 24 – Tabela de produtos do estoque .....	77
Tabela 25 – Tabela de movimentações de produtos .....	77
Tabela 26 – Tabela de notas fiscais de entradas e saídas .....	78
Tabela 27 – Tabela e movimentos de notas fiscais de entrada e saída .....	79
Tabela 28 – Tabela de ações da empresa .....	80
Tabela 29 – Tabela de reservas de lucros da empresa.....	80
Tabela 30 – Tabela de cursos de organizações da empresa .....	80

	X
Tabela 31 – Tabela de lucros .....	81
Tabela 32 – Tabela de imobilizados .....	81
Tabela 33 – Tabela de empréstimos a receber .....	81
Tabela 34 – Tabela de empréstimos a pagar .....	82
Tabela 35 – Tabela de despesas do exercício seguinte .....	82
Tabela 36 – Tabela de contas de capital .....	82
Tabela 37 – Tabelas de aplicações financeiras .....	83

## ÍNDICE DE QUADROS

Quadro 1 - O balanço patrimonial.....	40
Quadro 2 - A demonstração de resultado do exercício .....	41
Quadro 3 - O demonstrativo do resultado financeiro .....	42

## LISTA DE ABREVIATURAS

BD	-	Banco de Dados
BP	-	Balanço Patrimonial
CMM	-	<i>Capability Maturity Model</i>
DRE	-	Demonstração de Resultado do Exercício
DFC	-	Demonstração do Fluxo do Caixa
DLPA	-	Demonstrações de Lucros e Prejuízos Acumulados
DOAR	-	Demonstração de Origens e Aplicações dos Recursos
EI	-	Engenharia da Informação
ES	-	Engenharia de <i>Software</i>
HW	-	Hardware (equipamento)
IHC	-	Interface Homem-Computador
PL	-	Patrimônio Líquido
PSP	-	<i>Personal Software Process</i>
SEI	-	<i>Software Engeneering Institute</i>
SBD	-	Sistema de Banco de Dados
SGBD	-	Sistema Gerenciador de Banco de Dados
SI	-	Sistemas de Informação
SQL	-	<i>Structured Query Language</i>
SW	-	<i>Software</i> (programa)

## RESUMO

A importância social e econômica das pequenas empresas justifica o estudo de técnicas de gerenciamento específicas a este segmento empresarial.

Nestas empresas, o desconhecimento técnico, somado a restrições de recursos financeiros, dificulta o acesso do pequeno empresário a ferramentas de apoio administrativo, dentre as quais os recursos computacionais, tais como métodos de processamento de dados e de geração de informes gerenciais.

Assim, o pequeno empresário acaba por desconsiderar relevantes informações referentes à sua empresa, tanto na área financeira como na área contábil.

O objetivo deste trabalho é a elaboração de um programa que auxilie o administrador de uma pequena empresa, baseando-se em informações contábeis e financeiras.

Com este interesse são caracterizadas as principais informações gerenciais aplicáveis ao contexto da pequena empresa, e desenvolvido um modelo de manipulação da informação segundo as necessidades deste ramo empresarial.

São realizados estudos das principais técnicas de desenvolvimento de software voltadas à criação de uma ferramenta de apoio ao gerenciamento financeiro e contábil em uma pequena empresa, considerando a problemática envolvida neste cenário.

Finalmente, é desenvolvido o protótipo, no qual são aplicados os conceitos estudados, tanto no contexto da informação gerenciada, quanto dos conceitos computacionais abordados.

## ABSTRACT

*The social and economic importance of the small companies justifies the study of specific techniques of management to this enterprise path.*

*In these companies, the unfamiliarity technician, added the restrictions of financial features, makes it difficult the access of the small entrepreneur the tools of administrative bracket, amongst which the computational features, such as methods of generation and data processing of you inform managemental.*

*Thus, the small entrepreneur finishes for disrespecting excellent referring information to its company, as much in the financial area as in the countable area.*

*The objective of this work is to develop an optimized model of management of information, countable financiers and, with the interest of assisting the administrators of small companies, in the manipulation of these its information.*

*With this interest the main applicable managemental information to the context of the small company are characterized, and developed a model of manipulation of the information according to necessities of this enterprise branch.*

*Studies of the main techniques of development of software come back to the creation from a tool of bracket to the financial and countable management in a small company, considering problematic are carried through the involved one in this scene.*

*Finally, the archetype is developed, in which is applied the studied concepts, as much in the context of the managed information, how much of the boarded computational concepts.*

# CAPÍTULO 1 - Introdução

## 1.1 CONSIDERAÇÕES INICIAIS

Chiavenato (apud Borinelli, 1998) afirma que "as pequenas empresas constituem o cerne da dinâmica da economia dos países, são impulsionadoras dos mercados, geradoras de oportunidades e proporcionam empregos mesmo em situações de recessão."

Borinelli (1998) explica que "os grandes e novos fenômenos sócio-econômicos denominados de abertura de mercados, globalização, terceirização, entre outros, têm feito surgir uma grande gama de pequenas e médias empresas. A todo momento, empregados se desligam de grandes corporações e dedicam-se a empreender o seu próprio negócio."

Ao considerar este contexto, evidencia-se a importância das pequenas empresas, tanto social como economicamente, bem como se expõe um cenário caracterizado por uma competitividade crescente.

Segundo Resnik (1990: 6), "as condições decisivas para o sucesso de uma pequena empresa não são teorias que podem ser encontradas em livros, pois não são fáceis de serem prognosticadas [...] no entanto, o que existe é valioso, pois compreende uma abordagem à administração eficiente que ajuda a decidir o que é importante e onde devem ser aplicados os recursos de maneira mais produtiva."

Basil (1982: 12) afirma que geralmente uma pequena empresa tem limites quanto a experiências e à utilização de técnicas avançadas de gerenciamento, o que justifica a elevada taxa de fracasso destas pequenas empresas.

Estas técnicas de gerenciamento dizem respeito à manipulação da informação da empresa, principalmente as informações financeiras.

Beraldi (2000) diz existir a necessidade da utilização da informação contábil-financeira, por parte dos administradores, como uma ferramenta de apoio à administração.

Esta necessidade é também citada por Borinelli (1998), ao afirmar que "seria de grande utilidade que a informação, especialmente a contábil-financeira, fosse encontrada em posição de destaque no contexto das empresas. Porém o quadro com que o administrador tem se deparado é o de uma contabilidade desvalorizada, especialmente na visão empresarial."

A importância da informação financeira e contábil é também citada por Carmo (1999), quando escreve que "a motivação necessária para que as pequenas empresas

busquem e adquiram recursos de informatização é a sua sobrevivência em um mercado cada vez mais globalizado e competitivo, onde informações gerenciais são consideradas indispensáveis.”

Entretanto a esta informação está agregado um custo que cria uma certa resistência e dificulta a implementação destes métodos no âmbito das pequenas empresas, devido também, segundo Basil (1982: 10), à falta de preparo na utilização destas informações.

Cria-se, desta forma, um cenário empresarial que diferencia grandes e pequenas empresas, baseando-se no domínio (utilização) ou não destas informações gerenciais.

Este cenário é exposto por Beraldi (2000) ao afirmar que “ as pequenas empresas no geral não possuem sistemas informatizados, ou seja, seus controles são feitos quase que exclusivamente por meio de papeladas intermináveis.”

Assim, o desafio imposto à pequena empresa é o de adaptar-se a um ambiente computadorizado, onde a informação gerencial, financeira e contábil, seja utilizada como uma ferramenta de auxílio à administração na tomada de decisões.

A informatização, segundo Beraldi (2000), traz consigo diversos benefícios: “Uma pequena empresa com um sistema totalmente informatizado, funcionando eficiente e eficazmente, proporcionará grandes vantagens, seja em relação ao tempo otimizado, à organização, à facilidade de obtenção de informações, à previsão ou a outros aspectos que contribuirão para o sucesso da pequena empresa.”

Desta forma, a informatização agregaria eficiência e eficácia às pequenas empresas, melhorando assim sua competitividade e, indiretamente, sua lucratividade.

No entanto, este processo de informatização deve ser realizado com cautela e apoio técnico, pois, segundo Beraldi (2000), a utilização de recursos ou ferramentas inadequadas pode acabar agravando ainda mais os problemas da empresa após a informatização.

Um problema a ser enfrentado é o da utilização de informações contábeis, e não só das informações financeiras, visto a carência no repasse destas informações por parte dos contabilistas, fato relatado por Frey (1998), Resnik (1990:141) e Carmo (1999).

Em seguida devem ser definidos os passos a serem seguidos no processo de informatização que, segundo Resnik (1990: 33), são diferenciados de acordo com o tamanho da empresa.

Segundo Basil (1982: 10), a maior parte da literatura disponível sobre o assunto é voltada à administração de grandes empresas, havendo poucas considerações relativas ao gerenciamento das pequenas empresas.

Há a necessidade de diferenciação entre as pequenas e grandes empresas, pois tanto os modelos de implementação quanto as ferramentas administrativas (programas, métodos, ...) a serem utilizadas dependerão do porte das mesmas.

Existem diversos critérios para identificar o porte de uma empresa, tais como: número de empregados, receita anual, patrimônio, número de estabelecimentos existentes, etc., critérios estes adotados por órgãos governamentais (IBGE, Receita Federal), SEBRAE, bancos, entidades de classe, entre outros.

A Lei nº 2.211-B de 1996, no artigo 2º define a Empresa de pequeno porte “ ... a pessoa jurídica e a firma mercantil individual que, não enquadrada como microempresa, tiver receita bruta anual superior a R\$ 244.000,00 (duzentos e quarenta e quatro mil reais) e igual ou inferior a R\$ 1.200.000,00 (um milhão e duzentos mil reais).”

A Receita Federal, por sua vez, preocupa-se com o faturamento das empresas para proceder a classificação. A Lei n.º 9.317 instituiu o SIMPLES, classificando como empresa de pequeno porte, a pessoa jurídica que tenha auferido, no ano calendário, receita bruta superior a R\$ 120.000,00 (cento e vinte mil reais) e igual ou inferior a R\$ 720.000,00 (setecentos e vinte mil reais).

Contudo ao focar o gerenciamento das informações financeiras no contexto da pequena empresa, evidencia-se a adequação da definição relativa à receita ou faturamento da empresa, pois este modelo apresenta maior pertinência quanto à análise de informações contábeis e financeiras da mesma, isto por tratar-se de informações correlatas.

## **1.2 OBJETIVOS**

### **1.2.1 OBJETIVO GERAL**

O objetivo geral do trabalho é elaboração de um programa que auxilie o administrador de uma pequena empresa, baseando-se em informações contábeis e financeiras.

### **1.2.2 OBJETIVOS ESPECÍFICOS**

- Realizar um estudo referente aos aspectos do gerenciamento de informações contábeis e financeiras, evidenciando a importância destas informações.
- Abordar as principais técnicas relacionadas ao desenvolvimento de um programa voltado à pequena empresa e a sua administração.

- Disponibilizar métodos (técnicas) diferenciados de acesso à informação, que auxiliem o empresário na utilização de informações gerenciais utilizadas internamente na empresa.
- Elaborar um protótipo do programa proposto.
- Validar os métodos e o protótipo com base nos requisitos apresentados.

### **1.3 JUSTIFICAÇÃO**

A realização de estudos sobre o desenvolvimento de ferramentas que auxiliem o gerenciamento de pequenas empresas justifica-se pela importância deste segmento.

Beraldi (2000) explica que “ a tecnologia da informação é um recurso de difícil acesso a pequenas empresas, seu custo de aquisição é considerado alto, e existe uma grande dificuldade de adequação e utilização destes recursos.”

Russ (2000) afirma que o custo da especificação de um sistema, por vezes, inviabiliza o projeto detalhado das necessidades da empresa, o que a impele a utilizar uma ferramenta inadequada ou ineficiente, do ponto de vista do administrador.

As pequenas empresas, geralmente, contam com serviços terceirizados de contabilidade, sendo sua escrituração contábil de responsabilidade de um contador autônomo ou de algum escritório de contabilidade.

Frey (1998) atesta a importância da obtenção de informações gerenciais internas (na própria empresa) ao afirmar que “ ... pode-se concluir que não há um efetivo comprometimento do contabilista com o aspecto gerencial da contabilidade; este fato pode estar aliado principalmente a três fatores: o número muito grande de clientes por contabilista; insatisfação com a remuneração recebida; e o fato de a contabilidade não ser a única fonte de renda, tornando-a algo acessório.”

Desta forma, justifica-se este trabalho pela necessidade da definição de técnicas e métodos necessários à utilização das informações contábeis e financeiras no contexto da pequena empresa, de uma forma otimizada e direcionada a este segmento empresarial.

### **1.4 MOTIVAÇÃO AO DESENVOLVIMENTO**

Beraldi (2000) afirma que, apesar de algumas pequenas empresas possuírem computadores, a maioria não se utiliza dos benefícios da informatização devido à inadequação dos recursos de software.

A análise de alguns softwares de gerenciamento expõe um cenário que denota um direcionamento claro do gerenciamento da informação voltado aos aspectos financeiros da empresa, desprezando-se, na maioria das vezes, as informações contábeis.

Isto se deve, provavelmente, à complexidade agregada a utilização destas informações. Este quadro, somado à problemática do relacionamento entre empresário e contabilista, gera um ambiente onde a informação contábil é praticamente ignorada.

Desta forma, a proposta é a disponibilização de um programa voltado à necessidade da pequena empresa, que auxilie na gerência destas informações.

## **1.5 LIMITAÇÕES DO TRABALHO**

Ao considerar o enfoque gerencial do estudo, não são realizados apontamentos relativos a aspectos legais da informação, ou seja, não são seguidas normas governamentais ou fiscais na manipulação de dados ou na extração de informações, sendo o programa proposto, voltado à obtenção de informações destinadas à gerência interna.

Não serão implementadas técnicas de análise das entradas de dados, quanto a sua coerência, ficando esta responsabilidade a cargo do usuário.

Não serão realizadas análises referentes à utilização das informações extraídas pelo programa, ou mesmo de implementações do programa proposto.

## **1.6 ESTRUTURA DA PESQUISA**

No Capítulo 2 tem-se a revisão de literatura quanto ao desenvolvimento do protótipo, onde é apresentada a fundamentação teórica a ser utilizada, baseando-se em estudos relativos as áreas correlatas ao programa que é proposto.

No Capítulo 3 são definidas as principais informações a serem geridas em uma empresa, assim como são analisadas as necessidades específicas de informações para uma pequena empresa. Realizam-se ainda análises de alguns programas comerciais de gerenciamento, avaliando sua aplicabilidade à pequena empresa.

No Capítulo 4 é apresentado o projeto lógico do sistema proposto, sendo definido o banco de dados, os diagramas de fluxo de dados, bem como especificada a funcionalidade de cada um dos módulos do sistema.

Finalmente, no Capítulo 5 tem-se as conclusões, onde são analisadas as informações manipuladas e o desenvolvimento do protótipo. Por fim, são sugeridos trabalhos futuros que propõem uma continuação do presente trabalho.

## **CAPÍTULO 2 – PROJETO E DESENVOLVIMENTO DE SOFTWARE**

Neste capítulo será realizada a revisão de literatura em duas partes. A primeira parte retrata conceitos relativos ao projeto e desenvolvimento de software. A segunda parte apresenta conceitos contábeis aplicáveis ao contexto das pequenas empresas.

Considerando os objetivos do projeto, foram definidas as principais áreas de fundamentação teórica a serem estudadas, sendo elas:

- Sistemas de Informação;
- Interface Homem-Computador;
- Engenharia e Arquitetura de Software;
- Banco de Dados.

### **2.1 SISTEMAS DE INFORMAÇÃO (SI)**

Inicialmente deve-se diferenciar dado e informação. Segundo Martin (1991:12), “informação é o conhecimento derivado dos dados, sendo dados os fatos registrados”.

Hamacker (1998) apresenta a importância da informação afirmando:

“Atualmente estamos inseridos num contexto de extrema mutabilidade, onde a concorrência se torna cada vez mais acirrada. O consumidor torna-se cada vez mais exigente em relação ao custo e a qualidade do produto e dos serviços associados. O mercado agora é global e mais sensível, sendo as alterações no mesmo mais rápidas e representativas resultando como imperativa a importância da informação como instrumento de impacto decisivo nas perspectivas de rentabilidade e competitividade das empresas”

De acordo Martin (1991: 17), a informação deve ser utilizada como uma ferramenta, sendo considerada como um fator diferencial de competitividade, mas, para tanto, exige-se que seja utilizada nos processos de planejamento, organização e controle da empresa.

Ainda segundo Martin (1991: 17), um erro ao desenvolver um programa ou sistema computacional com a finalidade de resolver um problema ou situação em especial, é ter-se a idéia de que se deve apenas armazenar os dados e gerar relatórios que exponham classicamente os dados armazenados, não visando à geração de informações.

Para Hamacker (1998), uma informação é considerada útil quando tem três características fundamentais agregadas:

- pertinência ao contexto;
- disponibilidade em tempo hábil;
- precisão na dimensionalidade.

Carmo (1999) afirma serem três as funções básicas de um SI na organização de uma empresa, as quais têm a capacidade de transformar informação em conhecimento:

- Resolução de problemas, mediante o equacionamento e a proposta de soluções para apoiar o dirigente da empresa a atuar como agente transformador;
- Produção do conhecimento, através da obtenção de informações que seriam de difícil acesso por outros procedimentos;
- Tomada de consciência, propiciando a sensibilização para um problema da organização e o desenvolvimento da consciência da coletividade sobre a sua solução a curto e médio prazo.

Os SI bem sucedidos, segundo Hamacker (1998), adicionam valor aos seus custos:

- auxiliando os processos da organização interna;
- melhorando a qualidade das informações;
- auxiliando a gerência, especialmente na tomada de decisões;
- criando e solidificando vantagens comparativas.

### **2.1.1 TECNOLOGIA DE INFORMAÇÕES (TI)**

Vasudevan (1996) e Fish (1996) apresentam estudos referentes às mutações sofridas pela informação e sua adaptação quanto às inovações tecnológicas. A importância de tais estudos é tão relevante quanto maior for a relevância da informação analisada.

O fator chave do estudo, segundo Vasudevan (1996), é a implementação das novas tecnologias, sendo necessária a seqüência de alguns passos básicos: definir técnicas, observar e coletar dados.

Há um consenso entre Vasudevan (1996) e Fish (1996) quanto à forma e análise das tecnologias da informação, a formação e definição do processo:

- Documentar e interpretar as implementações nas suas ocorrências;
- Gerar um relatório de procedimentos com conclusões e recomendações;
- Desenvolver um modelo de implementação, a ser utilizado como guia.

Assim, a tecnologia da informação pode ser vista como sendo a evolução natural dos sistemas de informação, onde a este são agregadas novas tecnologias, o que, segundo Vasudevan (1996), ocorre de forma contínua.

### 2.1.2 A ENGENHARIA DA INFORMAÇÃO (EI)

A manipulação e gerência de dados implica em uma definição dos relacionamentos entre as diversas informações pertinentes ao contexto, tanto do ponto de vista do projetista como do usuário.

Segundo Martin (1991: 23), “a engenharia da informação é um conjunto interligado de técnicas automatizadas na qual são construídos modelos da organização, modelo de dados e modelos de processos em uma abrangente base de conhecimentos, a fim de serem usados para criarem e manterem sistemas de processamento de dados”.

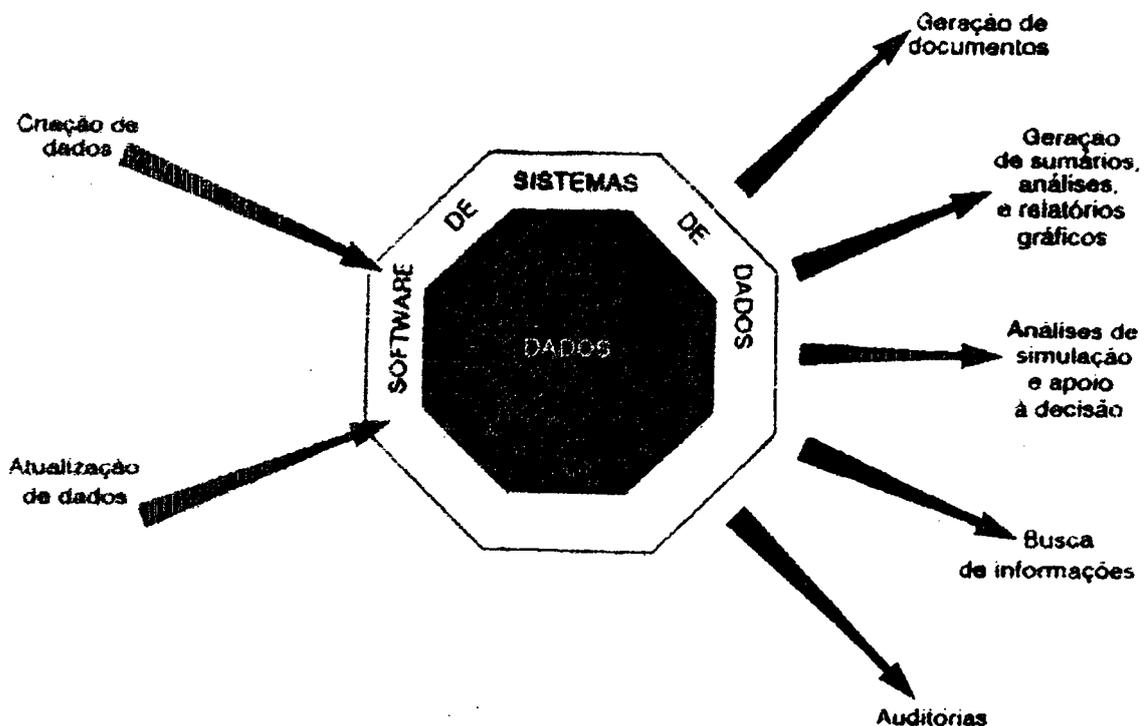


Figura 1 – O Modelo da Engenharia da Informação Fonte: Martin (1991: 37).

A figura 1 expõe a idéia geral de um sistema de gerência e administração de dados, expondo as entradas e saídas, bem como apresenta o conceito do processamento.

Outro fator que justificaria a aplicação de técnicas de engenharia da informação é a própria administração dos dados, vista sob o enfoque da criação de modelos estáveis de

dados. A figura 1, apresentada por Martin (1991: 37), evidencia a idéia do processamento dos dados de forma que o sistema compilará as informações e gerará documentos.

Os componentes de um sistema de informações são descritos na figura 2, proposta por Martin (1991: 39), onde são expostos os elementos que irão compor e atuar no sistema, de acordo com a entidade a qual está relacionada.

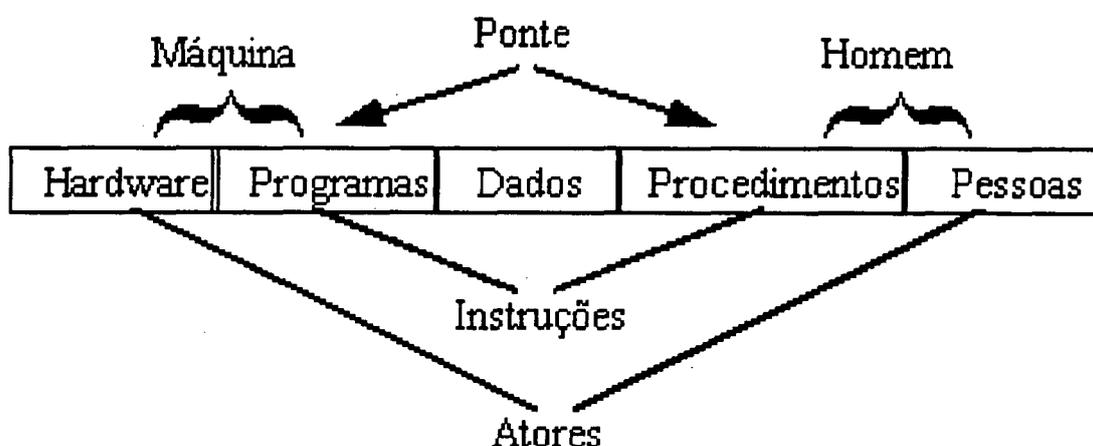


Figura 2 – Os componentes de um sistema de informações Fonte: Martin (1991: 39).

A figura 2 expõe os relacionamentos existentes entre os diversos componentes que compõem um SI. Fica evidenciada a necessidade de todos os componentes a fim de que seja caracterizada a gerência da informação.

Assim, segundo Martin (1991:12), dados + processamento = informações.

A conotação da ponte visa realizar a união entre os programas e o procedimento, ou seja, a definição formal das funções de gerenciamento da informação.

## 2.2 INTERFACE HOMEM COMPUTADOR

O estudo dos métodos de interface Homem-Máquina sempre acompanhou o desenvolvimento tecnológico, como afirma Heckel (1993: 12): “O importante não é a tecnologia em si, mas como ela é apresentada ao usuário”.

Segundo Heckel (1993: 49), “deve-se almejar uma interface otimizada, que extraia a necessidade de memorização e compreensão da mecânica envolvida no procedimento”.

Sandman (1998) considera a eficiência da comunicabilidade da interface como um fator de diferenciação entre sistemas.

Ainda segundo Heckel (1993: 14), diversos enfoques podem ser definidos quanto aos aspectos que tornam um software amigável: fatores humanos, ergonomia, facilidades de uso, psicologia cognitiva e a própria psicologia do usuário.

A figura 3 demonstra a importância da interface entre os procedimentos (humanos) e os programas (máquina).

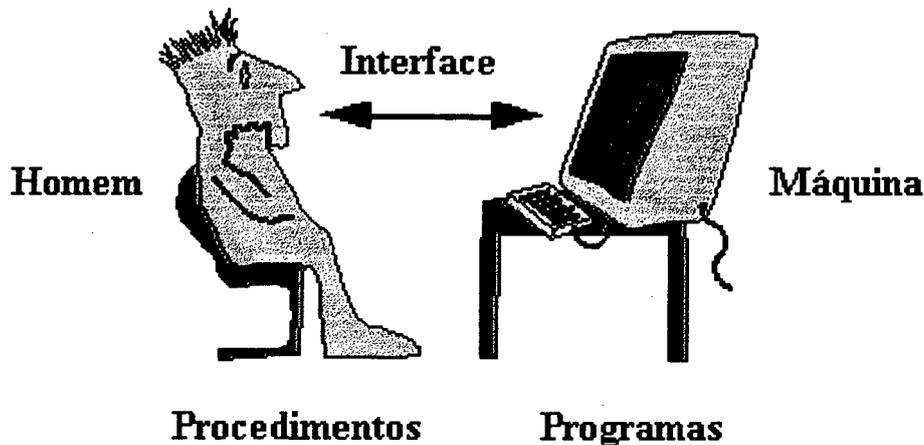


Figura 3 – A problemática da comunicação entre o usuário e o computador

A figura 3 reforça as idéias expostas por Martin (1991: 47), onde duas entidades, Usuário e Máquina, compreendem o sistema respectivamente como Procedimentos e Programas, sendo conectadas logicamente por dados, vistos agora como a interface.

Heckel (1993: 79) avalia a importância da interface do programa, dizendo que “... quando a preocupação com a aparência é maior do que a preocupação com o funcionalidade, o que se pode obter é a frustração do usuário”.

### 2.2.1 AS TÉCNICAS DE RACIONALIZAÇÃO DA INTERFACE HOMEM-COMPUTADOR

Heckel (1993: 18) analisa as principais técnicas de projeto da interface de software, descrevendo-as e inter-relacionando-as segundo suas características funcionais. Desta forma o autor apontou quais seriam as mais importantes técnicas a serem implementadas quando se deseja uma maior interatividade:

- ◆ Conhecimento sobre o assunto
- ◆ Conhecimento do público alvo
- ◆ Manutenção do interesse do usuário
- ◆ Comunicação visual
- ◆ Nível de conhecimento do usuário
- ◆ Linguagem do usuário
- ◆ Utilização de Metáforas
- ◆ Concentração do Usuário

- ◆ Comunicar-se só quando possível
- ◆ Diminuir a resistência do usuário
- ◆ Resolver os problemas
- ◆ Nunca frustrar o usuário
- ◆ Ajudar a resolver os problemas
- ◆ Responder as ações do usuário
- ◆ Liberar os aspectos mecânicos
- ◆ Consolidar os pensamentos
- ◆ Envolver o usuário
- ◆ Orientar o usuário
- ◆ Estruturar a interface
- ◆ Fazer algo confiável
- ◆ Atender todos níveis de usuário
- ◆ Aproximar-se do usuário
- ◆ Considerar a primeira impressão
- ◆ Criar um modelo mental
- ◆ Fazer algo simples
- ◆ É necessário ter visão

### 2.2.2 PSICOLOGIA COGNITIVA E SEMIÓTICA

Cybis (2000) analisa as principais variáveis referentes à psicologia cognitiva. Segundo o autor, a avaliação das interfaces humano-computador deve basear-se no estudo do comportamento humano e suas estruturas cognitivas.

Ao se projetar e desenvolver o sistema, devem ser considerados fatores, tais como: os modelos mentais, a memória conexionista, a percepção natural, bem como a heurística aplicada ao conhecimento humano.

Cybis (2000) define, ainda, a semiótica como a base da compreensão do relacionamento do trio: sinal, objeto e interpretação. A afirmação feita é a de que o entendimento, por parte do receptor, pode ser diferente da idéia imaginada pelo emissor, isto retrata uma compreensão errônea ou no mínimo uma ineficiência relativa ao processo de comunicação entre o receptor e o emissor.

A idéia apresentada é a de que o computador é basicamente uma ferramenta utilizada por um projetista, a fim de obter um objeto ou produto que atenda as exigências do usuário, visto que o que se pretende é criar um canal de alta comunicabilidade entre o usuário e os objetivos.

### 2.2.3 ANÁLISE ERGONÔMICA

Heckel (1993: 153) afirma que “primeiramente, deve-se realizar uma referência ao relacionamento entre a tarefa e a atividade a ser realizada, isto porque podem haver discrepâncias entre o que deve ser feito e como isto será feito.”

Heckel (1993: 153) comenta que a seqüência de passos deve ser apresentada ou exposta de forma amigável, evitando, de todas as formas, múltiplos entendimentos a respeito do objetivo de uma ou outra atividade.

Segundo Cybis (2000), “o software interativo é aquele que torna a representação proposta para sua interface compatível com aquelas esperadas pelo usuário em seu trabalho”.

Conseqüentemente, deve-se realizar um estudo relativo às atividades que estes mesmos usuários objetivam, a fim de que suas necessidades não só sejam supridas, mas que o usuário veja o programa como uma ferramenta, algo útil ao seu trabalho.

#### **2.2.4 TENDÊNCIAS PARA AS NOVAS INTERFACES**

Nielson (1996) expõe idéias relativas às tendências quanto à definição e criação de novos métodos relativos ao projeto da interface. As idéias citadas em seu trabalho apontam para a utilização de técnicas naturais ao usuário, tornando o aprendizado mais simples, gerando um sistema de fácil utilização, aumentando a satisfação e a aceitação do usuário.

De acordo com o pesquisador, podem ser apontadas as principais técnicas de interface atual, bem como prever quais seriam as técnicas a serem utilizadas no futuro.

Dentre as técnicas atuais apontadas por Nielson (1996), destacam-se:

- comandos que instruem o computador para realizar uma tarefa;
- estruturação funcional;
- estruturação por objetos;
- uma única aplicação ativa para o usuário;
- utilização de links na chamada de execuções.

Heckel (1993) baseia seus estudos e conclusões em experiências de softwares desenvolvidas por grandes empresas como IBM, Apple e Microsoft, considerando inclusive as tendências da época do desenvolvimento dos programas analisados e seu contexto de desenvolvimento.

Ao confrontarem as tendências futuras dos sistemas computacionais no desenvolvimento de sistemas, Nielson (1996) aponta as principais técnicas ou métodos, ao realizar comparações entre os sistemas atuais com as tendências de mercado ao desenvolvimento de sistemas. Uma comparação é apresentada na tabela 1:

<b>Enfoque</b>	<b>Sistemas Atuais</b>	<b>Tendências Futuras</b>
Atenção do Usuário	O usuário deve preocupar-se com aspectos mecânicos	O usuário deve preocupar-se somente com a tarefa, e não a aspectos mecânicos
Regras de Comportamento	O computador obedece os comandos de forma literal	O computador compreende os interesses do usuário e realiza suas ações
Controle da Interface	Os objetos estão disponíveis e controlados pelo usuário	A interface e seus objetos são invisíveis e controlados pelo sistema
Quanto à Sintaxe	Exige precisão e grandes definições na sintaxe	O sistema deve compreender o usuário, interagindo de forma amigável e flexível
Visibilidade dos Objetos	Os objetos disponibilizam-se a fim de que funções sejam executadas e manipuladas	Os objetos estão invisíveis, sendo manipulados e geridos por mecanismos implícitos
Seqüência de Interação	Os dados são inseridos de forma padronizada	Utilização de métodos não convencionais, fontes múltiplas.
Capacidade Informacional	Tipicamente o teclado e o mouse, pouca informação	Entradas baseadas em informações não convencionais.
FeedBack Informacional	Análise conferência da informação a nível léxico	A informação é reconhecida, tratada e analisada de acordo com a informação
Turn Take	São exigidas fases de conversação a fim de se obter uma informação final	As informações são encontradas de forma paralela e independente, aumentando a flexibilidade na busca
Local da Interface	Tipicamente a tela, com suas limitações	A idéia da interface é estendida a novas tecnologias e métodos de comunicação
Programação	Técnicas convencionais e baseadas na execução	Programação orientada a objetos, expansibilidade, visualização e compatibilidade

**Tabela 1 – Tendências Atuais e Futuras dos Métodos Computacionais, segundo Nielson (1996)**

A tabela 1 apresenta um relacionamento claro entre as principais técnicas atualmente utilizadas e as técnicas apresentadas por Nielson (1996), como tendências futuras a serem agregadas a sistemas.

Muitas semelhanças podem ser encontradas nas colocações de Nielson (1996) e Heckel (1993), o que reforça a idéia apresentada por Cybis (2000), de que um sistema deve ser voltado ao usuário, e que este não deve se preocupar com os aspectos mecânicos do desenvolvimento, mas com os dados inseridos e seus objetivos.

## 2.3 ENGENHARIA E ARQUITETURA DE SOFTWARE

### 2.3.1 ENGENHARIA DE SOFTWARE (ES)

Segundo MacConnel (2000), “a ES pode ser vista como uma abordagem de desenvolvimento de software elaborada com disciplinas e métodos bem definidos”.

Segundo Lindval (2000), os elementos da engenharia de software são:

- métodos: proporcionam os detalhes de como fazer para construir o software;
- ferramentas: dão suporte automatizado aos métodos;
- procedimentos: constituem o elo de ligação entre os métodos e ferramentas.

#### 2.3.1.1 O ciclo de vida do software

O ciclo de vida é definido por Pressman (1995: 32) como sendo o conjunto de atividades utilizadas na análise e na engenharia de sistemas.

Pressman (1995: 32) e Russ (2000) definem as etapas conhecidas como componentes de ciclo de vida de software ou Processo de Software:

1. natureza do projeto e da aplicação;
2. métodos e ferramentas a serem usados;
3. controles e produtos que precisam ser entregues.

#### 2.3.1.2 Ciclo de Vida Clássico (Cascata)

- modelo mais antigo e mais amplamente usado da engenharia de software ;
- modelado em função do ciclo da engenharia convencional;
- requer uma abordagem sistemática, seqüencial ao desenvolvimento de software.

As fases deste ciclo de vida são descritas por Pressman (1995: 33), na figura 4.

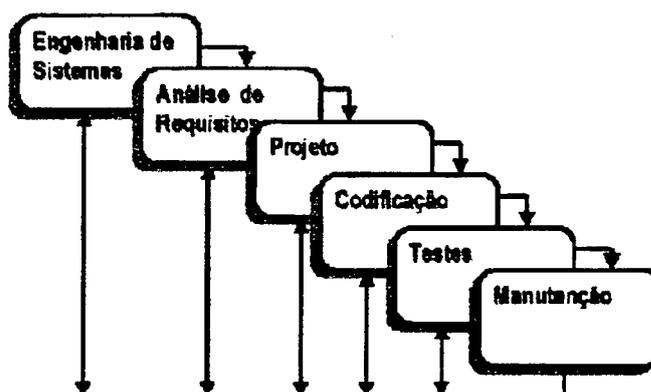


Figura 4 – O Ciclo de vida clássico dos sistemas de software Fonte: Pressman (1995: 33)

A figura 4 permite compreender a funcionalidade do ciclo de vida em suas diversas fases e mostra a possibilidade de se retornar a passos anteriores sempre que necessário.

Fase 1 – A análise e engenharia de sistemas - Envolve a coleta de requisitos em nível do sistema, sendo uma visão essencial quando o software deva ser interativo.

Fase 2 – Projeto - É a tradução dos requisitos do software representativos que podem ser avaliados quanto à qualidade, antes que a codificação se inicie.

Fase 3 – A Codificação - Tradução das representações do projeto para uma linguagem “artificial”, resultando em instruções executáveis pelo computador.

Fase 4 – Os testes - Concentram-se nos aspectos funcionais externos e nos aspectos lógicos internos do software, com todas as instruções sendo testadas.

Fase 5 – A manutenção - O software deverá sofrer mudanças depois que for entregue.

Alguns problemas do Ciclo de Vida Clássico são expostos por Pressman (1995: 34):

- projetos reais raramente seguem o fluxo que o modelo propõe;
- logo no início é difícil estabelecer explicitamente todos os requisitos;
- o cliente deve ter paciência, pois uma versão executável do software só fica disponível numa etapa avançada do desenvolvimento.

Russ (2000) afirma que, apesar das fragilidades do desenvolvimento, utilizando-se da Engenharia de Software, a aplicação destas técnicas é significativamente melhor do que uma abordagem casual ao desenvolvimento de software

Outra metodologia para a gerência de desenvolvimento de software é descrita por Pressman (1995: 38) como sendo o ciclo de vida espiral, e citada por MacConell (2000), como sendo o modelo ideal para o desenvolvimento de sistemas.

### 2.3.1.3 O ciclo de vida em espiral

Algumas características diferenciam o ciclo de vida em espiral, sendo elas:

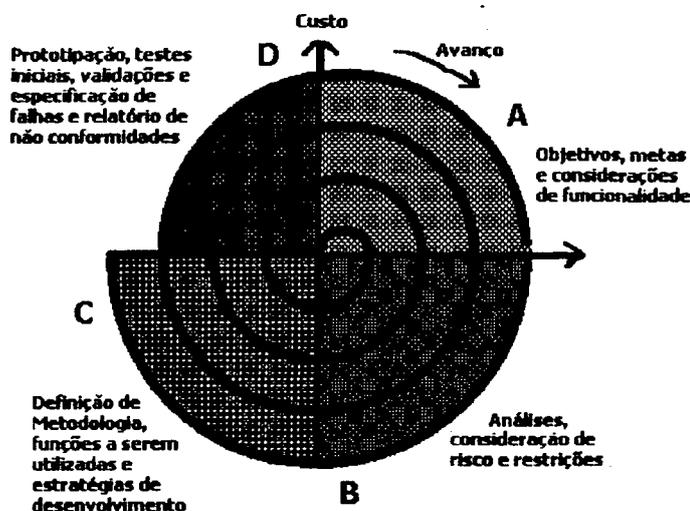
- engloba as melhores características do ciclo de vida clássico e da prototipação;
- segue a abordagem de passos sistemáticos do ciclo de vida clássico, incorporando-os numa estrutura interativa que reflete a real;
- usa a prototipação, em qualquer etapa, como mecanismo de redução de riscos.

As atividades do ciclo de vida em espiral podem ser analisadas separadamente.

O Ciclo de vida em espiral é composto de 4 fases contínuas e seqüenciais. A figura

5, apresentada por Pressman (1995: 39), denota graficamente o ciclo de vida em espiral.

Um modelo de desenvolvimento baseado em passos sequenciais e contínuos, onde o avanço do desenvolvimento do software ocorre progressivamente.



O Modelo Espiral Direcionado a Situação

Figura 5 – O ciclo de vida em espiral

A figura 5 explicita o processo de maturação do software relacionado aos ciclos de desenvolvimento, não sendo possível, no entanto, segundo Pressman (1995), MacConell(2000) ou Laitinen (2000), uma definição de um ponto de maturação preciso, a partir do qual o software seria considerado realizado.

- Fase 1 - Planejamento: determinação dos objetivos, alternativas e restrições;
- Fase 2 - Análise de Risco: identificação / resolução dos riscos;
- Fase 3 - Construção: desenvolvimento do produto no nível seguinte;
- Fase 4 - Avaliação do Cliente: avaliação e planejamento das novas fases.

MacConell (2000) avalia o modelo em espiral, validando-o e comentando que “... o modelo em espiral tem um alto nível de abstração, tem a capacidade de reduzir os riscos, porém é visto como complicado demais, sendo utilizado somente por especialistas”.

#### 2.3.1.4 O Gerenciamento das Falhas do Software

A problemática do estudo das falhas no software é analisada por Lindval (2000) e Laitinen (2000) onde são expostas as dificuldades na mensuração das falhas.

Pressman (1995: 15) afirma que a constância e contínua evolução dos

equipamentos de hardware aliada à competitividade natural do mercado impele, necessariamente, ao desenvolvimento de novos produtos.

Todo e qualquer produto desenvolvido tende a apresentar falhas que deverão ser solucionadas. A curva idealizada de descrição das falhas para o software é apresentada por Pressman (1995: 15) na figura 6.

A figura 6 denota uma característica típica do software, que é a impossibilidade de se extraírem totalmente as falhas do software, o que ocorre devido a alterações no projeto original, oriundas da manutenção natural do sistema.

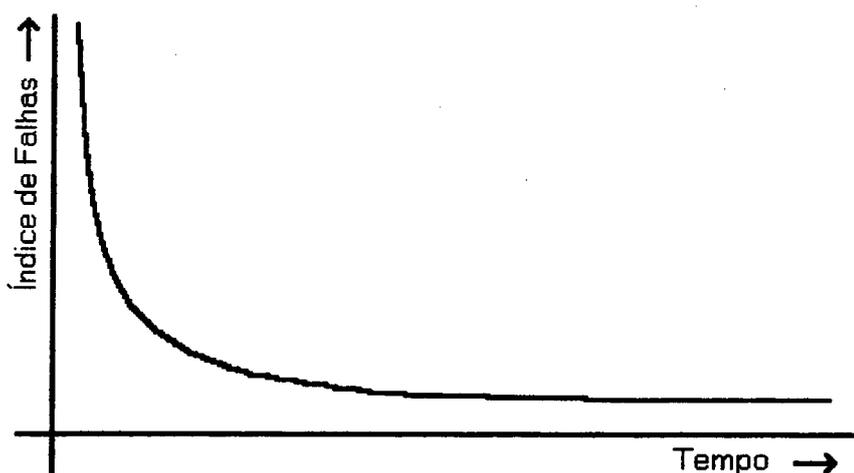


Figura 6 – Índice desejável de falhas do software no tempo Fonte: Pressman(1995:15)

Analisando a figura 6, pode-se concluir que com o tempo o sistema obterá um índice de falhas mínimo, e que este índice se estabilizaria. Isto na teoria.

Na prática, no entanto, a curva do índice é a apresentada por Pressman (1995: 15) como sendo a curva real das falhas, apresentada pela figura 7.

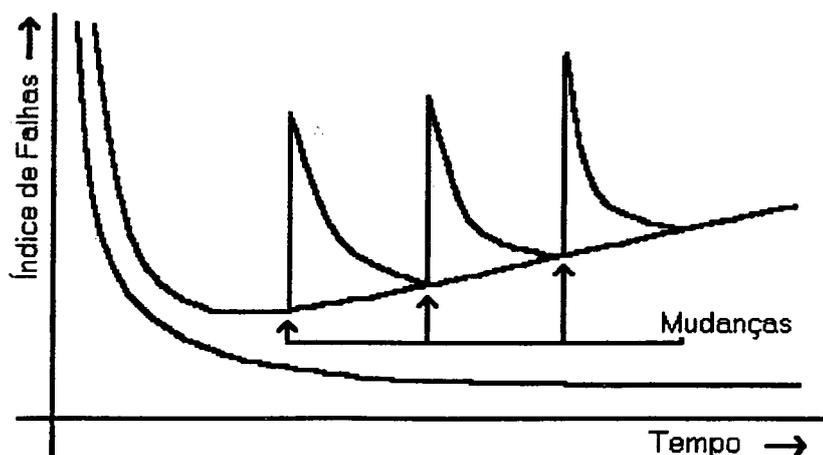


Figura 7 – Índice real de falhas do software no tempo, segundo Pressman (1995: 15)

A grande diferença é a consideração do fator de mudança do software ao processo, ou seja, a cada mudança solicitada o índice de falha mínimo eleva-se, deteriorando o software prematuramente devido às mudanças no software.

Na figura 7 tem-se uma representação real do índice de falhas que ocorre em um software ao longo de sua utilização. Note-se que nos instantes onde houve mudanças ou alterações, o índice de falhas eleva-se abruptamente, e que dificilmente o patamar de confiabilidade anterior é alcançado novamente.

Segundo Russ (2000), “dificilmente um sistema é concluído de forma que ajustes ou alterações não sejam necessárias. A grande diferença de confiabilidade estaria relacionada ao número de mudanças necessárias e aplicadas ao sistema”.

### 2.3.2 QUALIDADE DE SOFTWARE

Para Pressman (1995: 723), “... tem-se dado grande importância ao processo como forma de se garantir um software de melhor qualidade”.

Laitinen (2000) expõe que a qualidade é um termo que pode ser definido de várias formas, causando mal-entendidos, pois não tem um único sentido. Para cada conceito existem vários níveis de abstração; a visão popular é diferente do seu uso profissional.

Ainda segundo Laitinem (2000), a qualidade na visão popular difere-se da qualidade na visão científica, pois:

- é termo indefinível;
- ela pode ser sentida, discutida, julgada, mas não pode ser medida;
- um luxo, classe e elegância. Produtos caros e complexos têm melhor nível de qualidade. Confiabilidade e o número de reparos não é considerado.

Pressman (1995: 724) define a Qualidade de Software como sendo a “conformidade aos requisitos de desempenho e de funcionalidade que foram explicitamente definidos, aos padrões de desenvolvimento documentados e às características implícitas que são esperadas por todo software desenvolvido por profissionais”.

Garlan (1994) comenta que a gerência de processo objetiva a geração de produtos de acordo com o planejado e, ao mesmo tempo, melhorar a capacidade de produzir software com mais qualidade. Para tal, deve-se tomar cuidado na correta sequência dos seguintes passos:

1. Compreender o estado atual do processo;
2. Desenvolver uma visão do processo desejado;
3. Estabelecer ações para a melhoria do processo;
4. Gerar um plano para acompanhar estas ações;
5. Compreender os recursos para execução do plano;
6. Recomeçar a partir do Passo 1.

A qualidade enfatiza três importantes conceitos apontados por Belloquim (1999), e citados também por outros autores como sendo:

- Requisitos de Software formam a base a partir da qual a qualidade é avaliada. Falta de concordância com os requisitos é falta de qualidade;
- Padrões especificados definem um conjunto de Critérios de Desenvolvimento que orientam a construção do software. Se o critério não for seguido, faltará qualidade;
- Requisitos Implícitos que freqüentemente não são mencionados. Se o software atende aos requisitos explícitos, mas falha nos implícitos, a qualidade é suspeita.

Belloquim (1999) afirma ainda que o cumprimento das metas não pode ser considerado uma garantia de qualidade, vistos os demais fatores, tanto internos como externos, que influenciam indiretamente na qualidade, tais como:

- Realização correta das atividades;
- Robustez nas execuções em condições anormais;
- Extensibilidade quanto a possíveis alterações no projeto;
- Reusabilidade dos métodos e funções;
- Compatibilidade com outros sistemas;
- Eficiência na utilização dos recursos;
- Portabilidade entre tecnologias;
- Facilidade de utilização.

Já segundo Pressman (1995: 726), analisando as vantagens e desvantagens dos sistemas entre si, podem ser expostos quais seriam os fatores da Qualidade de Software que diferenciam os sistemas:

- *Manutenibilidade*: Posso consertá-lo?
- *Flexibilidade*: Posso mudá-lo

- *Testabilidade*: Posso testá-lo?
- *Portabilidade*: Poderei usá-lo em outra máquina?
- *Reusabilidade*: Poderei reutilizá-lo em outra máquina?
- *Interoperabilidade*: Poderei compor uma interface com outro sistema?
- *Corretitude*: Ele faz aquilo que eu quero?
- *Confiabilidade*: Ele se comporta com precisão o tempo todo?
- *Eficiência*: Ele rodará no meu hardware tão bem quanto possível?
- *Integridade*: Ele é seguro?
- *Usabilidade*: Ele foi projetado para o usuário?

A ISO 9000-3 em seu guia para a aplicação da ISO 9001 para o desenvolvimento, fornecimento e manutenção de software, criado em 1993, especifica requisitos mínimos para assegurar a qualidade de produtos e serviços, não definindo modelos ou impondo sistemas de qualidade.

### 2.3.3 GERÊNCIA DO DESENVOLVIMENTO DE SOFTWARE

Segundo MacConell (2000), “o desenvolvimento de programas pode ser considerado similar a criação de um produto convencional, tendo-se obviamente um direcionamento do enfoque para os problemas específicos oriundos das imposições do ambiente”.

O software, como qualquer produto que esteja por ser desenvolvido, exige o esforço e trabalho de uma equipe de desenvolvimento. A figura 8 denota a proporcionalidade entre as atividades do desenvolvimento do software exposto por Pressman (1995: 145).

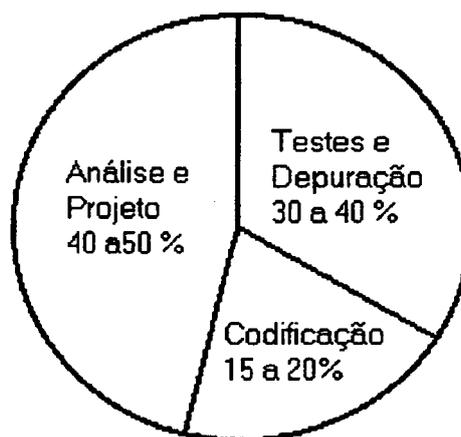


Figura 8 – O gráfico comparativo de esforço no desenvolvimento do software

O gráfico retrata o clássico modelo 40-20-40, analisando com maior ênfase os passos relativos ao projeto, colocando a codificação como sendo uma parte relativamente pequena perante o todo.

Segundo Pressman (1995: 147), as principais considerações quanto à criação e desenvolvimento do software são:

a) próprio caráter do Software - software é um elemento de sistema lógico e não físico.

Conseqüentemente, o sucesso é medido pela qualidade de uma única entidade e não pela qualidade de muitas entidades manufaturadas. O software não se desgasta, mas se deteriora!

b) falhas das pessoas responsáveis pelo desenvolvimento de Software:

- Gerentes sem nenhum *background* em software.
- Os profissionais da área de software têm recebido pouco treinamento formal em novas técnicas para o desenvolvimento de software.
- Resistência a mudanças.

c) mitos do Software propagaram desinformação e confusão:

Mitos do Software (administrativos):

- Já temos um manual repleto de padrões e procedimentos para a construção de software. Isso não oferecerá ao meu pessoal tudo o que eles precisam saber?
- Meu pessoal tem ferramentas de desenvolvimento de software de última geração; afinal lhes compramos os mais novos computadores.
- Se nós estamos atrasados nos prazos, podemos adicionar mais programadores e tirar o atraso.

Mitos do Software (do cliente):

- Uma declaração geral dos objetivos é suficiente para se começar a escrever programas - podemos preencher os detalhes mais tarde.
- Os requisitos de projeto modificam-se continuamente, mas as mudanças podem ser facilmente acomodadas, porque o software é flexível.

Mitos do Software (do profissional):

- Assim que escrevermos o programa e o colocarmos em funcionamento nosso trabalho estará completo.

- Enquanto não tiver o programa "funcionando", eu não terei realmente nenhuma maneira de avaliar sua qualidade.

A natureza e os objetivos dos softwares a serem desenvolvidos podem e são muito diferidos, de acordo com a aplicação.

Russ (2000) e Laitinen (2000) afirmam que a única diferença no desenvolvimento para um modelo 'pequeno' está relacionada à quantidade de informações a serem analisadas. Um modelo de desenvolvimento em espiral de pequenos projetos exigirá um cuidado especial, relativo à complexidade do modelo.

Assim o processo de desenvolvimento de software contém 3 fases genéricas, independentes do modelo de engenharia de software escolhido:

- Definição;
- Desenvolvimento;
- Manutenção.

**Definição:** "o que" será desenvolvido.

- **Análise do Sistema:** define o papel de cada elemento num sistema baseado em computador, atribuindo, em última análise, o papel que o software desempenhará.
- **Planejamento do Projeto de Software:** assim que o escopo do software é estabelecido, os riscos são analisados, os recursos são alocados, os custos são estimados e as tarefas e programação de trabalho definidas.

**Desenvolvimento:** "como" o software vai ser desenvolvido.

**Projeto de Software:** traduz os requisitos do software num conjunto de representações (algumas gráficas, outras tabulares ou baseadas em linguagem) que descrevem a estrutura de dados, a arquitetura do software, os procedimentos algorítmicos e as características de interface.

**Manutenção:** concentra-se nas "mudanças" que ocorrerão depois que o software for liberado para uso operacional

- **Correção:** mesmo com as melhores atividades de garantia de qualidade de software, é provável que o cliente descubra defeitos no software. A manutenção corretiva muda o software para corrigir defeitos.
- **Adaptação:** com o passar do tempo, o ambiente original, para o qual o software

foi desenvolvido provavelmente mudará. A manutenção adaptativa muda o software para acomodar mudanças em seu ambiente.

- **Melhoramento Funcional:** à medida que o software é usado, o cliente/usuário reconhecerá funções adicionais que oferecerão benefícios.

#### **2.3.4 O DESENVOLVIMENTO DE PEQUENOS PROJETOS**

Russ (2000) expõe as oito fases de desenvolvimento apresentadas como solução, para pequenos projetos de software, sendo elas:

- Restrições na atividade de análise;
- Análise do domínio;
- Análise da aplicação;
- Definição da arquitetura;
- Projeto da aplicação;
- Implementação do programa;
- Integração;
- Testes do sistema.

Evidencia-se a diferenciação entre o modelo apresentado por Russ (2000) e Pressman(1995: 191), principalmente no tocante às fases de desenvolvimento.

Russ (2000) afirma que as maiores diferenças ocorrem com vistas ao fato de que pequenos projetos tendem a ter menos ciclos de desenvolvimento até a finalização, o que gera uma diferenciação evidente quanto à maturação do sistema como um todo.

#### **2.3.5 A PROTOTIPAÇÃO DO SOFTWARE**

MacConell (2000) mescla diversos conceitos relativos ao desenvolvimento, abordando a prototipação de forma indireta ao considerar os principais e mais utilizados métodos e conceitos relativos a desenvolvimento de software.

Whittaker (2000) expõe a importância da prototipação do software como um método eficiente na busca de erros e falhas no desenvolvimento.

O processo de prototipação possibilita que o desenvolvedor crie um modelo do software que deve ser construído. Idealmente, o modelo (protótipo) serve como um mecanismo para identificar os requisitos de software.

As atividades básicas relacionadas à prototipação são descritas por Sisti (1994):

- a) Obtenção dos Requisitos: desenvolvedor e cliente definem os objetivos do software;
- b) Projeto Rápido: representação dos aspectos do software que são visíveis ao usuário;
- c) Construção do Protótipo: implementação do projeto rápido;
- d) Avaliação do Protótipo: cliente e desenvolvedor avaliam o protótipo;
- e) Refinamento dos Requisitos: cliente e desenvolvedor refinam os requisitos do software

Sisti (1994) afirma que o risco do desenvolvimento é tanto menor quanto melhores forem as especificações, as análises do problema, ou seja, a compreensão do problema como um todo.

Segundo Laitinen (2000), "... a chave é definir-se as regras do jogo logo no começo. O cliente e o desenvolvedor devem ambos concordar que o protótipo seja construído para servir como um mecanismo a fim de definir os requisitos."

### **2.3.6 O MODELO SEI CMM**

Segundo Rezende (1999: 145) e Pressman (1995: 171): "O SEI/CMM é o modelo hoje mais indicado de gerenciamento do Processo de Software".

SEI é o Software Engineering Institute, uma instituição vinculada à Carnegie-Mellon University. Este instituto foi fundado com o objetivo de pesquisar os problemas recorrentes encontrados nas tarefas de Engenharia de Software e propor soluções que permitam resolver estes problemas.

Segundo Belloquim (1999), "a idéia é criar uma infra-estrutura que permita a quem escreve software fazer um bom trabalho, isto é, criar software melhor, mais rápido, mais barato e com menos "bugs", que seja mais fácil de dar manutenção e tenha estendida sua funcionalidade. O mérito da SEI está justamente em perceber que métodos sozinhos não resolvem problema nenhum".

O modelo CMM é extremamente útil para uma organização de desenvolvimento poder avaliar a si mesma e estabelecer prioridades para melhorar seu processo de desenvolvimento e, conseqüentemente, aumentar a qualidade do software que desenvolve.

O Modelo CMM (Capability Maturity Model) - desenvolvido pelo SEI (Software Engineering Institute) para avaliar a maturidade de processo de desenvolvimento de software, concluído em 1991, considerando 5 níveis de maturidade:

- Inicial - nível 1 Processo disciplinado;
- Repetitivo - nível 2 Gerenciamento de configuração; Garantia de qualidade;

- Processo consciente - nível 3 (Definido) Revisões; Coordenação entre grupos;
- Processo previsível - nível 4 (Gerenciado) Gerenciamento de qualidade;
- Melhoria contínua do processo - nível 5 (Otimizado) Gerenciamento de mudança.

Os benefícios do uso da metodologia de desenvolvimento do modelo CMM são maximizados quando a empresa tem sua metodologia de desenvolvimento de sistemas "encaixada" no contexto mais amplo do Processo de Software.

Segundo Rezende (1999: 147) e Belloquim (1999), podem ser definidas e caracterizadas as principais peculiaridades da organização em cada um dos cinco níveis do CMM e as ações necessárias para passar ao próximo nível.

Segundo Pressman(1995), o modelo CMM é extremamente útil para que uma organização de desenvolvimento possa avaliar a si mesma e estabelecer prioridades para melhorar seu processo de desenvolvimento e, conseqüentemente, aumentar a qualidade do software que desenvolve.

### **2.3.7 PERSONAL SOFTWARE PROCESS (PSP)**

Segundo Rezende (1999: 156), "PSP é o processo de software usado por engenheiros de software, projetado para controlar e melhor gerenciar o trabalho".

Esta metodologia de desenvolvimento é recente, tendo sido desenvolvida por Watss S. Humphrey com a intenção de orientar o planejamento e desenvolvimento de módulos de software ou pequenos programas.

Ainda segundo Rezende (1999: 156), "a proposta do modelo PSP é fornecer um método de trabalho que ensine processos de software aos próprios desenvolvedores, partindo do princípio do seu envolvimento pessoal no próprio sucesso."

Os principais objetivos do modelo PSP são, segundo Rezende (1999: 156):

- Melhorar as estimativas, planejamento e acompanhamentos;
- Criar um comprometimento pessoal para a qualidade;
- Envolvimento total do engenheiro na melhoria contínua e constante do software;
- Organizar o backlog (pendências) de atividades, evitando sobrecargas.

### 2.3.8 METODOLOGIAS DE DESENVOLVIMENTO

Analisando os programas, seus objetivos e seu funcionamento, é possível definir os padrões de desenvolvimento utilizados, expondo as técnicas e as formas como os programas foram criados.

Dois métodos de desenvolvimento são apontados por Melton (1997):

#### Baseado em módulos

Melton (1997) comenta que os sistemas desenvolvidos baseados em módulos devem definir uma forma na qual a unificação deve ocorrer, demarcando também as variáveis e métodos de conectividade que deverão ser aplicados.

Garlan (1994) afirma que a arquitetura dos programas atuais não considera desde seu início os fatores de conectividade entre os diversos módulos do sistema.

O problema da unificação dos módulos do sistema, apontado por Melton (1997), exige um grau de organização da equipe de desenvolvimento, bem como aumenta a flexibilidade do desenvolvimento, pois exige, única e exclusivamente, que alguns padrões de conexão sejam rigorosamente seguidos, o que, em caso contrário, pode colocar em risco o projeto de desenvolvimento.

#### Desenvolvimento monolítico

Nesta metodologia de desenvolvimento, o programa é totalmente definido em sua fase de especificação e requisitos. O diferencial advém dos métodos de desenvolvimento e da implementação do sistema.

O sistema é desenvolvido em um único módulo funcional, o que segundo Melton (1997), traz consigo vantagens e desvantagens em relação o desenvolvimento modular.

As vantagens podem ser definidas com base em uma maior consistência dos dados a serem gerenciados, isto por não existirem conexões entre as funções, visto que os comandos e funções atuam diretamente, sem a necessidade de serem conectados.

A desvantagem é a dificuldade de agregação de novas funções, pois isto poderia exigir a atualização completa do software, incluindo referências e banco de dados.

### 2.3.9 A UNIFICAÇÃO DOS MÓDULOS FUNCIONAIS

Melton (1997) descreve a questão da unificação arquitetural como a solução para o problema da conectividade, desde que os módulos, em sua criação, definam claramente as conexões e variáveis compartilhadas entre os módulos a serem implementados.

De acordo com as idéias propostas por Melton (1997), a unificação dos módulos criados dependerá basicamente da preparação realizada nos mesmos, a fim de que estes possam ser conectados no futuro; porém, é de se esperar que uma empresa que lance um produto incompleto não esteja preparada para lançar, ao menos a curto e médio prazo, os módulos restantes. Considerando-se a problemática da documentação, é de se esperar graves problemas no tocante à conectividade entre os módulos.

Já, segundo Russ (2000), os pontos problemáticos, na grande maioria dos casos, ocorrem quando, no desenvolvimento, ignoram-se aspectos básicos referentes ao problema para o qual o programa tem seu projeto definido:

- consideração quanto ao tamanho da organização;
- complexidade do projeto;
- atributos de qualidade;
- interações personalizadas.

### 2.3.10 METODOLOGIA DE TESTES DE SOFTWARE

Pressman (1995: 286) afirma: “A atividade de teste é o processo de executar um programa com a intenção de descobrir um erro. Um bom caso de teste é aquele que tem uma elevada probabilidade de revelar um erro ainda não descoberto. Um teste bem-sucedido é aquele que revela um erro ainda não descoberto”.

Quanto às atividades de teste, Pressman (1995: 788) afirma que:

- Objetivo: projetar testes que descubram sistematicamente diferentes classes de erros e façam-no com uma quantidade de tempo e esforço mínimos;
- Se a atividade de teste for conduzida com sucesso, ela descobrirá erros no software;
- A atividade de teste não pode mostrar a ausência de bugs; ela só pode mostrar se defeitos de software estão presentes;
- Se erros graves forem encontrados com regularidade a qualidade e a confiabilidade de software são suspeitas;
- Se erros facilmente corrigíveis forem encontrados a qualidade e a confiabilidade do software estão aceitáveis ou os testes são inadequados para revelar erros graves;

- Se não for encontrado erro a configuração de teste não foi suficientemente elaborada e erros estão escondidos no software.

Rezende (1999: 252) explica que “a atividade de testes é de fundamental importância no desenvolvimento de sistemas ou software, ela pode garantir o sucesso do mesmo, pois se todos os requisitos funcionais forem testados, provavelmente o cliente estará satisfeito.”

Whittaker (2000) analisa o procedimento de testes: “No projeto e na execução dos testes, deve-se considerar o software e suas funções, as entradas e saídas devem ser combinadas, bem como o ambiente onde o software irá ser executado.”

Desta forma, Whittaker (2000) relaciona quatro fases básicas nos testes que possibilitam a validação do sistema:

- modelagem do ambiente do software;
- a seleção dos cenários de teste;
- a validação dos cenários de teste;
- mensuração do progresso dos testes.

Outra metodologia de teste, de acordo com Pressman (1995: 791), pode ser realizada com os testes da caixa preta ou branca:

**Teste de Caixa Preta:** refere-se aos testes que são realizados nas interfaces do software.

São usados para demonstrar que as funções dos softwares são operacionais, que a entrada é adequadamente aceita e a saída é corretamente produzida; que a integridade das informações externas é mantida. Examina aspectos do sistema sem se preocupar muito com a estrutura lógica interna do software.

**Teste de Caixa Branca:** baseia-se num minucioso exame dos detalhes procedimentais. O

“status do programa” pode ser examinado em vários pontos para determinar se o “status” esperado ou estabelecido corresponde ao “status” real.

A grande diferença nos modelos apresentados por Whittaker (2000) e Pressman(1995) é relativa ao enfoque ou forma de abordar o problema, embora outros autores citem estes métodos como eficientes.

### **2.3.11 GERENCIAMENTO DE CONFIGURAÇÃO**

O SCM é uma atividade abrangente que é aplicada em todo o processo de engenharia de software para administrar as alterações durante o ciclo de vida do software.

Os itens apresentados por Pressman (1995: 921) são referidos também por outros autores, ainda que com pequenas alterações.

Estes passos compreendem todas as informações produzidas como parte do processo de engenharia de software, e são chamados coletivamente de configuração de software.

Abaixo são apresentadas as linhas básicas de definição do projeto do software, expostas por Pressman (1995: 921):

1. Especificação do Sistema
2. Plano de Projeto do Software
3. a) Especificação dos Requisitos de Software  
b) Protótipo Executável ou “em papel”
4. Manual preliminar do usuário
5. Especificação de Projeto
  - a) Descrição do projeto de dados
  - b) Descrição do projeto arquitetural
6. Listagem do código-fonte
7. a) Plano e Procedimento de Testes  
b) Casos de teste e resultados registrados
8. Manuais Operacionais e de Instalação
9. Programa Executável
  - a) Módulos - código executável
  - b) Módulos interligados
10. Descrição do banco de dados
  - a) Esquema e estrutura de arquivo
  - b) Conteúdo inicial
11. Manual feito de acordo com o usuário
12. Documentos de manutenção
13. Padrões e procedimentos para a construção do software

### **2.3.12 O CONTEXTO DE DESENVOLVIMENTO DOS PROGRAMAS ATUAIS**

Antonioni (1995) comenta que apesar de a maioria dos problemas apontados ser oriunda de falhas humanas, há uma tendência cada vez maior de minimizar estas falhas, visto o fator competitividade

Segundo Garlan (1994), um problema nos sistemas atuais é o seu superdimensionamento, onde o sistema é composto por um grande número de módulos. Em casos extremos, estes sistemas acabam por nunca serem concluídos, e alguns módulos que deveriam realizar algumas funções esperadas pelos usuários acabam por nunca serem concluídos.

Garlan (1994) afirma ainda que a análise dos programas permite concluir que os desenvolvedores e projetistas estão pouco interessados em saber quais os módulos serão utilizados, pois a idéia é a da criação de um modelo genérico, e não de um modelo específico, pois assim tem-se um sistema que pode ser comercializado para todos os tipos de empresas, independente de seu ramo de atividade.

## **2.4 BANCO DE DADOS**

### **2.4.1 CONCEITOS DE BANCO DE DADOS**

A necessidade de gerenciar os dados e informações do sistema exige estudo e definição formal dos métodos que serão utilizados no seu desenvolvimento.

Segundo Korth (1995: 1), “um sistema de banco de dados consiste em uma série de componentes funcionais, incluindo o gerenciador de arquivos, o gerenciador de buffer, o interpretador de consultas, o seletor de estratégias, o gerenciador de recuperação e o controlador de concorrência”.

Já Elmasri (1994: 2) diz que “um Banco de Dados é uma coleção de dados relacionados. Por dado pode-se pensar em fatos que podem ser registrados e que tenham informações implícitas”.

Ainda segundo Elmasri (1994: 2), “um SGBD é uma coleção de programas que permitem ao usuário criar e manter o BD. O SGBD é um software de propósito geral que facilita o processo de definir, construir e manipular o BD para vários aplicativos.”

A característica que permite uma independência dos dados e a operação dos mesmos é chamada de abstração dos dados. O BD deve prover uma representação conceitual dos dados que não aprofunde em detalhes dos dados armazenados.

Uma definição de Projeto Conceitual do Banco de Dados é descrita por Elmasri (1994: 24), “uma vez que todos os requisitos tenham sido coletados e analisados, o próximo passo é criar um esquema conceitual do BD, utilizando-se de conceitos de alto nível de modelagem. O esquema conceitual é uma descrição concisa dos requisitos de dados dos usuários e inclusive detalhes de descrição detalhados dos tipos de dados, relacionamentos e restrições do sistema”.

Date (1991: 3) apresenta uma visão para um sistema de banco de dados, afirmando que “um sistema de banco de dados não é nada mais do que um sistema de manutenção de registros por computador. O próprio banco de dados pode ser considerado uma espécie de sala de arquivo eletrônica, ou seja, um depósito de um conjunto de arquivos de dados computadorizados que oferecem recursos ao usuário”.

Rezende (1999: 236) expõe uma visão de banco de dados diferente dos demais autores citados anteriormente, voltada ao desenvolvimento de sistemas, afirmando: “É uma coleção de dados como em um arquivo convencional, porém mais aplicados a um determinado fim unificado, e muito efetivo no apoio à tomada de decisões.”

#### 2.4.2 REPRESENTAÇÃO FÍSICA DO BANCO DE DADOS

Como já foi dito nos itens anteriores, a idéia básica do banco de dados é abstrair o armazenamento e gerenciamento da informação ao usuário.

A diferença entre o conceito de banco de dados e a sua estruturação lógica dificulta a sua amostragem gráfica. Uma forma de expor a estruturação do banco de dados, baseando-se nos conceitos do Banco de Dados, é apresentada na figura 9.

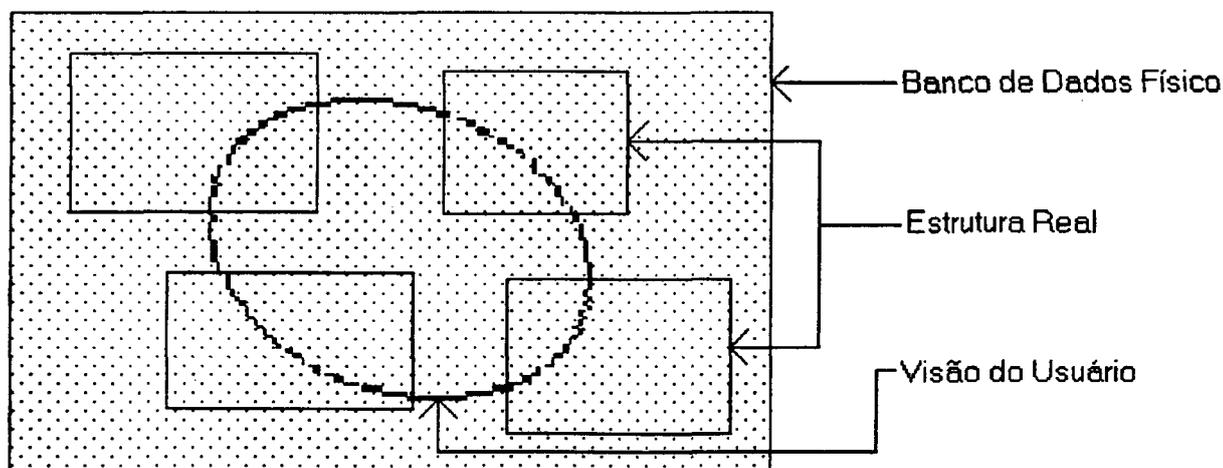


Figura 9 - Estrutura interna do Banco de Dados Relacionais e a visão do usuário

A figura 9 explicita graficamente as formas de gerenciamento atualmente utilizadas nos modelos relacionais bem como a interpretação que o usuário faz do modelo de armazenamento de dados que está utilizando.

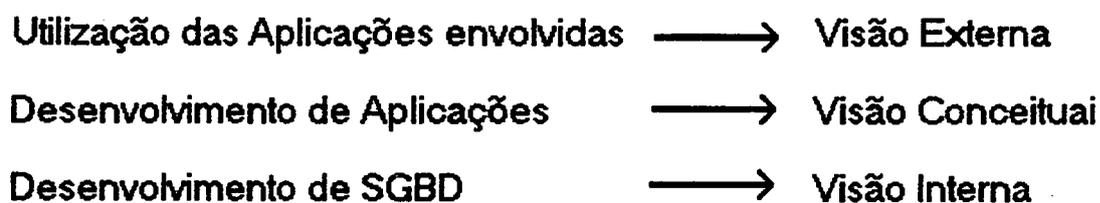
A figura 9 pode ser vista como uma mescla de idéias apresentadas por Date (1991: 39) e Korth (1995: 122) onde as informações estão agrupadas de uma forma muito mais organizada que a visão do usuário sobre o assunto.

### 2.4.3 VISÕES DO BANCO DE DADOS

Segundo Date (1991: 32), a idéia e concepção do Banco de Dados traz consigo a idéia da abstração e contextualização da informação, segundo o enfoque por que se tenha interesse, tendo-se, desta forma, uma arquitetura dividida em três níveis gerais:

- **VISÃO INTERNA** - É aquela vista pelo responsável pela manutenção do SGBD. Existe a preocupação com a forma de recuperação e manipulação dos dados do BD.
- **VISÃO CONCEITUAL** - É aquela vista pelo analista de desenvolvimento e pelo administrador das bases de dados. Existe a preocupação com a definição de normas para manipulação. Na visão conceitual, existem duas linguagens de operação que são:
  - Linguagem de definição dos dados (DDL) - Linguagem que define as aplicações, arquivos e campos que irão compor o banco de dados.
  - Linguagem de manipulação dos dados (DML) - Linguagem que define os comandos de manipulação e operação dos dados.
- **VISÃO EXTERNA** - É aquela vista pelo usuário que opera o aplicativo, através de interfaces desenvolvidas pelo analista, buscando o atendimento de suas necessidades.

O que se tem são diferentes visões, que retratam basicamente as atividades.



**Figura 10 – Visões do Banco de Dados**

São amostradas na figura 10 as diversas visões que são assumidas pelo BD, de acordo com enfoque ou interesse de análise. Obviamente a visão externa de um BD sugere simplesmente a utilização das aplicações e, indiretamente, dos dados que o constituem.

#### 2.4.4 A ESTRUTURA INTERNA DO SISTEMA DE BANCO DE DADOS (SBD)

Korth (1995: 19) relaciona os componentes funcionais de um SBD:

- Gerenciador de arquivos;
- Gerenciador de banco de dados;
- Processador de consultas;
- Pré-compilador da DML;
- Compilador de DLL.

Date (1991: 131) analisa as componentes internas do Sistema de Banco de Dados, tais como a indexação, técnicas de compressão, Hashing e metodologias de organização interna.

#### 2.4.5 AS VANTAGENS DO BD PERANTE À ARQUITETURA TRADICIONAL

Analisando funcionalmente o BD, as principais vantagens do Banco de Dados, segundo Korth (1995: 2) e Date (1991: 13), são:

*Redução ou eliminação de Redundâncias* - Possibilita a eliminação de dados privativos de cada sistema. Os dados que eventualmente são comuns a mais de um sistema são compartilhados, permitindo o acesso a uma única informação por vários sistemas.

*Controle de Concorrências* – Se diversos usuários atualizam o banco de dados concorrentemente, a consistência dos dados pode não mais ser preservada. Controlar a interação entre usuários simultâneos é atribuição do gerenciador de BD.

*Eliminação de Inconsistências* - Através do armazenamento da informação em um único local com acesso descentralizado e que os usuários utilizem uma informação confiável. A inconsistência consiste de um mesmo campo com valores diferentes em sistemas diferentes.

*Compartilhamento de Dados* - Permite a utilização simultânea e segura de um dado, por mais de uma aplicação ou usuário, independente da operação que esteja sendo realizada. Deve ser observado apenas o processo de atualização concorrente, para não gerar erros de processamento (atualizar simultaneamente o mesmo campo do mesmo registro). Os aplicativos são por natureza multiusuários.

*Restrições de Segurança* - Define para cada usuário o nível de acesso a ele concedido (leitura, leitura e gravação ou sem acesso) ao arquivo e/ou campo. Este recurso impede que pessoas não autorizadas utilizem um determinado arquivo ou campo.

*Padronização dos Dados* - Permite que os campos armazenados na base de dados sejam padronizados segundo um determinado formato de armazenamento (padronização de tabela, conteúdo de campos, etc) e o nome de variáveis seguindo critérios padrões pré-estabelecidos pela empresa.

*Manutenção de Integridade* - Exige que o conteúdo dos dados armazenados no Banco de Dados possua valores coerentes ao objetivo do campo, não permitindo que valores absurdos sejam cadastrados.

*Atender Necessidades Conflitantes* - Representa a capacidade que o administrador de Banco de Dados deve ter para solucionar "prioridades sempre altas" de todos os sistemas, tendo ele que avaliar a real necessidade de cada sistema para a empresa, para priorizar a sua implantação.

*Independência de Dados* - Representa a forma física de armazenamento dos dados no Banco de Dados e a recuperação das informações pelos programas de aplicação. Esta recuperação deverá ser totalmente independente da maneira como que os dados estão fisicamente armazenados. Este conhecimento do formato de armazenamento do campo é totalmente transparente para o usuário. A independência dos dados permite os seguintes recursos:

- a - Os programas de aplicação definem apenas os campos que serão utilizados, independente da estrutura interna dos arquivos
- b - Quando há inclusão de novos campos no arquivo, será feita manutenção apenas nos programas que utilizam esses campos.

## 1.6 NORMALIZAÇÃO DE DADOS

De acordo com Heuser (1998: 129), normalizar os dados em um BD consiste em definir o formato lógico adequado para as estruturas de dados identificados no projeto lógico do sistema, com o objetivo de minimizar o espaço utilizado pelos dados e garantir a integridade e confiabilidade das informações.

A normalização é feita através da análise dos dados que compõem as estruturas, utilizando o conceito chamado "Formas Normais (FN)". As FN são conjuntos de restrições que os dados devem satisfazer. Pode-se, por exemplo, dizer que a estrutura está na primeira forma normal (1FN), se os dados que a compõem satisfizerem as restrições definidas para esta etapa.

Date (1991: 407) define a normalização como uma forma de reduzir a redundância. “O objetivo geral do processo de redução é reduzir a redundância e, assim, eliminar certas anomalias de atualização. Deve-se, porém, assinalar que a normalização são apenas diretrizes; por vezes, há boas razões para não se normalizar tudo.”

Korth (1995: 190), analisando o contexto da normalização dos dados, cita algumas propriedades ‘indesejáveis’ apresentadas por um mau projeto, onde a normalização teria ocorrido de forma inadequada:

- repetição de informações;
- inabilidade para representar certas informações;
- perda de informação.

Para Heuser (1998: 129) a normalização completa dos dados é feita seguindo as restrições das primeiras quatro formas normais existentes, sendo que a passagem de uma FN para outra é feita tendo como base o resultado obtido na etapa anterior, ou seja, na FN anterior.

Para realizar a normalização dos dados, é primordial que seja definido um campo chave para a estrutura, campo este que permite identificar os demais campos da estrutura.

Heuser (1998: 130) e Date (1991: 381) expõem as formas normais:

*Primeira Forma Normal (1FN)* - Consiste em retirar da estrutura os elementos repetitivos, ou seja, aqueles dados que podem compor uma estrutura de vetor. Podemos afirmar que uma estrutura está normalizada na 1FN se não possuir elementos repetitivos.

*Segunda Forma Normal (2FN)* - Consiste em retirar das estruturas que possuem chaves compostas (campo chave sendo formado por mais de um campo), os elementos que são funcionalmente dependentes de parte da chave. Podemos afirmar que uma estrutura está na 2FN se ela estiver na 1FN e não possui campos funcionalmente dependentes da chave.

*Terceira Forma Normal (3FN)* - Consiste em retirar das estruturas os campos que são funcionalmente dependentes de outros campos que não são chaves. Podemos afirmar que uma estrutura está na 3FN se ela estiver na 2FN e não possui campos dependentes de outros campos não chaves.

*Quarta Forma Normal (4FN)* - Basicamente, tem-se a Quarta Forma Normal quando além de estar na forma 3FN, este também não contenha dependências multivaloradas. Cabe lembrar que o conteúdo é o mesmo, alterando-se somente a questão funcional da informação.

Rezende (1999: 238) apresenta as formas normais em um formato simples, e amostrando somente as 3 primeiras formas normais, as quais considera fundamentais ao projeto de um programa.

Após a normalização, as estruturas dos dados estão projetadas para eliminar as inconsistências e redundâncias dos dados, suprimindo desta forma qualquer problema de atualização e operacionalização do sistema.

A versão final dos dados poderá sofrer alguma alteração, para atender as necessidades específicas do sistema, a critério do analista de desenvolvimento durante o projeto físico do sistema.

#### **2.4.7 A INTEGRIDADE DOS DADOS**

Date (1991: 457) afirma que a integridade “refere-se a exatidão e precisão dos dados”.

Korth (1991: 16) afirma que “as restrições de integridade fornecem meios para assegurar que mudanças feitas no banco de dados por usuários autorizados não resultem na perda da consistência dos dados. Assim, as restrições de integridade resguardam o banco de dados contra danos acidentais.”

A problemática da integridade está associada à implementação dos testes e validação dos dados. O que se tem neste caso é uma sobrecarga no trabalho dos desenvolvedores, visto que caberá a estes o trabalho de garantir a integridade dos dados.

Considerando a realidade e o contexto do desenvolvimento atual dos programas, Date (1991: 466) afirma serem três as limitações quanto às garantias de integridade em um sistema:

- Limitações por domínio – quando um determinado atributo é definido em um determinado domínio;
- Limitações primárias e de chave externa;
- Dependências funcionais, multivaloradas e resultantes de junções.

#### **2.4.8 O MODELO DE DADOS RELACIONAL**

O modelo relacional compõe-se de três aspectos principais:

- Parte estrutural – composta por domínios e relações n-árias;
- Integridade dos dados – chaves primárias não-nulas e chaves externas;

➤ Manipulação de Dados – álgebra relacional e atribuição relacional.

Analisando os sistemas relacionais, Date (1991: 328) explica que, “o modelo relacional não é uma solução universal. Ele deve ser encarado apenas como uma necessidade mínima, no sentido de que qualquer banco de dados moderno deve, pelo menos, suportar os recursos prescritos pelo modelo”.

Cabe lembrar que nenhum SGBD atualmente implementa inteiramente todas as premissas e funções definidas no modelo relacional.

No entanto, o maior problema é a validação da integridade: “... a maioria dos sistemas falha quanto às regras de integridade, se não todos”. (Date 1991: 337).

Quanto à execução das regras de integridade relacional, Korth (1995: 168) expõe que a maioria dos sistemas relacionais atuais não suporta a noção de chaves primária e externa, e assim, naturalmente, não suporta da mesma forma as duas regras de integridade relacional:

- Integridade de entidade;
- Integridade referencial.

O Modelo de Dados relacional representa os dados contidos em um BD através de relações. Estas relações contêm informações sobre as entidades representadas e seus relacionamentos.

O Modelo Relacional, segundo Pressman (1995: 327), é baseado no conceito de matrizes, onde as chamadas linhas (das matrizes) seriam os registros e as colunas (das matrizes) seriam os campos. Os nomes das tabelas e dos campos são de fundamental importância para nossa compreensão entre o que estamos armazenando, onde estamos armazenando e qual a relação existente entre os dados armazenados.

#### 2.4.9 O SQL

No decorrer do desenvolvimento dos Bancos de Dados Relacionais, surgiram linguagens destinadas à sua manipulação. O Departamento de Pesquisas da IBM desenvolveu o SQL como forma de interface para o sistema de BD relacional denominado SYSTEM R, início dos anos 70. Em 1986 o American National Standard Institute (ANSI) publicou um padrão para o SQL. E assim, o SQL estabeleceu-se como linguagem padrão de Banco de Dados Relacional.

Segundo Date (1991: 111), “SQL é uma sub-linguagem de sistemas de dados. Como tal, inclui tanto uma linguagem de definição de dados (DDL) como uma linguagem de manipulação de dados (DML) ... o DDL opera nos níveis externo e conceitual.”

Korth (1995: 106) afirma que “a SQL estabeleceu-se claramente como a linguagem padrão de banco de dados relacional”.

O SQL nada mais é que uma série de comandos que permitem a definição dos dados, chamada de DDL (Data Definition Language), destinado a criação do banco de dados e das tabelas que o compõem, além das relações existentes entre as tabelas.

Outra característica muito importante disponível em SQL é sua capacidade de construção de visões, que são formas de visualizar os dados na forma de listagens independentes das tabelas e organização lógica dos dados.

Deve-se notar que a linguagem SQL consegue implementar estas soluções, somente pelo fato de estar baseada em BD que garante por si mesmo a integridade das relações existentes entre as tabelas e seus índices.

O SQL pode ser considerado, segundo Date (1991: 112), como um conjunto de quatro componentes principais:

- pré-compilador – realizar um pré-processamento para programas de aplicação que contenham SQL embutido;
- componente tradutor - que compila a string SQL e produz um plano de aplicação;
- supervisor de tempo de aplicação – que supervisiona o andamento dos programas SQL durante sua execução;
- gerenciador de dados armazenados – que cuida do BD real, armazenando e recuperando os registros à medida que são solicitados pelo plano de execução.

Korth (1995: 107) expõe o SQL analisando suas características funcionais e apresentando-o como um conjunto de diversas partes:

- DDL – Linguagem de definição de dados;
- DML – Linguagem de manipulação de banco de dados interativo;
- Linguagem de manipulação de dados embutidos;
- Definições de Visões;
- Autorização;
- Integridade.

## **CAPÍTULO 3 - INFORMAÇÕES CONTÁBEIS E FINANCEIRAS NAS PEQUENAS EMPRESAS**

Neste capítulo são inicialmente identificadas as principais informações contábeis e financeiras aplicáveis no gerenciamento de empresas. Em seguida estes conceitos são analisados no contexto da pequena empresa, sendo realizada também uma análise de alguns softwares gerenciais que se propõem a este fim.

### **3.1 NOÇÕES DE CONTABILIDADE BÁSICA**

#### **3.1.1 RELATÓRIOS CONTÁBEIS E FINANCEIROS**

Marion (1989: 51) afirma que “o relatório contábil é a exposição resumida e ordenada de dados colhidos pela contabilidade”.

Marion (1989: 51) cita a Lei da S.A. que estabelece os principais relatórios contábeis a serem elaborados com base na escrituração contábil ao final de cada período:

- Balanço Patrimonial ou BP;
- Demonstração de Resultado do Exercício ou DRE;
- Demonstrações de Lucros e Prejuízos Acumulados ou DLPA;
- Demonstrações das Origens e Aplicações de Recursos ou DOAR.

Na seqüência são detalhados o BP e a DRE, que constituem-se nos principais demonstrativos financeiros, sendo que não serão estudados detalhes do DLPA e o DOAR.

Segundo Marion (1989: 403), “... além das Demonstrações Financeiras e Contábeis estabelecidas por Lei (BP, DRE, DOAR e DLPA), há outros relatórios padronizados de grande relevância. É o caso da Demonstração do Fluxo de Caixa (DFC), que, embora não exigido por Lei, é de grande utilidade interna na entidade.”

##### **1.1.1 Balanço Patrimonial (BP)**

Iudicibus (1990: 26) e Marion (1989: 53) afirmam que o Balanço Patrimonial reflete a posição financeira em um determinado momento, geralmente ao fim do ano fiscal ou de um período prefixado.

Segundo Ross (1997: 27), o “balanço é um instantâneo do valor contábil da empresa numa certa data, como se a empresa tivesse ficado momentaneamente imóvel ... é um modo conveniente de organizar e sintetizar o que uma empresa possui (ativos), o que a empresa deve (exigibilidades), e a diferença entre eles (patrimônio líquido)”

Neves (1997: 283) afirma que o BP "... e a demonstração que tem por objetivos expressar os elementos financeiros e patrimoniais de uma entidade, através da apresentação ordenada de suas aplicações de recursos (ativo) e das origens dos recursos."

A estrutura de BP apresentada no quadro 1 é aceita como padrão, sendo utilizada por diversos autores, Marion (1989:78), Iudícibus (1990:32), Neves (1997: 283) e Ross (1998: 38); trata-se, no entanto, de uma convenção de apresentação dos dados.

Ativo	Passivo + PL
Ativo Circulante	Passivo Circulante (até 1 ano)
Contas a Receber	Contas a Pagar
Estoques	Impostos
Investimento Temporário	Encargos
Contas a Receber	Empréstimos
Realizável a Longo Prazo	Exigível a Longo Prazo (> 1 ano)
Permanente	Patrimônio Líquido
Investimento	Capital
Imobilizado	Investimentos
Diferido	

**Quadro 1 – O Balanço Patrimonial. Fonte: Iudícibus (1990: 32).**

No quadro 1 é demonstrada a estrutura de um balanço patrimonial, tendo a esquerda o ativo e a direita o passivo e o patrimônio líquido.

Fazem parte do ativo todos os bens (máquinas, terrenos, estoques, ...) e direitos (contas a receber, duplicatas a receber, ações, créditos, ...) de propriedade da empresa.

Já o passivo é a obrigação exigível, isto é, no momento em que a dívida vencer será exigida (reclamada) a sua liquidação.

Marion (1989: 54) afirma que "o passivo evidencia toda a obrigação (dívida) que a empresa tem com terceiros: contas a pagar, fornecedores de matéria-prima, impostos a pagar, financiamentos, empréstimos, ... é uma obrigação exigível, isto é, no momento em que a dívida vencer, será exigida a liquidação da mesma."

O Patrimônio Líquido (PL) evidencia recursos dos proprietários aplicados na empresa, tanto o capital inicial, como os investimentos posteriores.

Em síntese, do lado esquerdo tem-se os bens e direitos (ativos) e do lado direito as obrigações com terceiros (passivo) e com proprietários (PL).

Para a elaboração do balanço patrimonial há a necessidade de se apurar o resultado do exercício. O resultado do exercício, segundo Marion (1989: 51), será "... o grande indicador global de eficiência: o retorno resultante do investimento dos donos da empresa (lucro ou prejuízo)."

### 3.1.1.2 Demonstração de Resultado do Exercício (DRE)

Um conceito sobre DRE é exposto por Marion (1989: 106): “Em cada final de período, todas as despesas são transferidas para a DRE, sendo que, no próximo período, apurar-se-á despesa e receita começando-se do zero, ou seja, não se acumulam despesa e receita de um período para o outro.”

Ross (1990: 42) afirma que “... a demonstração do resultado mede o desempenho da empresa em um certo período, geralmente um trimestre ou um ano.” Para Iudícibus (1990: 38), “a DRE é um resumo ordenado das receitas e despesas da empresa em determinado período.”

No quadro 2, proposto por Iudícibus (1998: 48), tem-se um modelo simplificado de DRE:

<b>Demonstração do Resultado do Exercício</b>	
<b>Receita Bruta</b>	
(-)	Deduções
	IPI e ICMS
	Abatimentos
<b>Receita Líquida</b>	
(-)	Custo dos Produtos Vendidos
<b>Lucro Bruto</b>	
(-)	Despesas Operacionais
	De vendas
	Administrativas
	Financeiras (-) Receita
<b>Lucro Operacional</b>	
(+/-)	Despesas/Receitas não operacionais
<b>Resultado do Exercício</b>	

**Quadro 2 – O Demonstrativo de resultado do exercício. Fonte Iudícibus (1990: 38).**

No exemplo citado no quadro 2 tem-se a estrutura e os principais componentes necessários à elaboração da DRE.

Caso o resultado do exercício seja positivo, indicará o lucro, caso seja negativo, representará que houve prejuízo no período analisado.

A estrutura da DRE pode variar de acordo com as necessidades e interesses da empresa, com maior ou menor nível de detalhamento de acordo com suas necessidades.

### 3.1.1.3 Demonstração do Fluxo de Caixa (DFC)

“O fluxo de caixa é o instrumento de projeção que possibilita ao administrador financeiro determinar as necessidades financeiras a curto e longo prazos da empresa, apontando o momento em que irão ocorrer as entradas e saídas de caixa” (Zdanowicz, 1986: 27).

A importância do DFC, segundo Marion (1989: 404), advém do fato que “... a DFC vem esclarecer situações controvertidas na empresa, como, por exemplo, através da comparação com a DRE, o porquê de a empresa ter um lucro considerável e estar com o caixa baixo, não conseguindo liquidar todos os seus compromissos. Ou ainda, embora seja menos comum, o porquê de a empresa ter prejuízo neste período em questão, embora o caixa tenha aumentado”.

Ainda, segundo Zdanowicz (1986: 27), “... é através do fluxo de caixa que o administrador financeiro pode detectar a capacidade da empresa de gerar capital necessário para sua expansão, modernização ou realocização”.

Segundo Iudícibus (1990: 93), “a Demonstração de Fluxo de Caixa indica a origem de todo o dinheiro que entrou no Caixa, bem como a aplicação de todo o dinheiro que saiu do caixa em determinado período e, ainda, o Resultado do Fluxo Financeiro. Assim, como a DRE, a DFC é uma demonstração dinâmica e também está contida no Balanço que, por sua vez, é uma demonstração estática.”

O fluxo de caixa é estruturado conforme é mostrado no quadro 3.

<b>Demonstração do Fluxo de Caixa</b>	
Saldo final do exercício anterior	
(+) Receitas	
Receitas do período	
(-) Despesas	
Pagamentos e Fornecedores	
Compra de Imobilizados	
Amortizações	
Pagamento de Despesas Operacionais	
Pagamento de dividendos	
Saldo final do exercício corrente	

**Quadro 3 – O Demonstrativo de resultado financeiro. Fonte: Iudícibus (1990: 48).**

Marion (1989: 404) comenta, em relação ao potencial de planejamento do DFC, que “... através do planejamento financeiro o gerente saberá o momento certo em que contrairá empréstimos para cobrir a falta de fundos, bem como quando aplicar no mercado

financeiro o excesso de dinheiro, evitando, assim, a corrosão inflacionária e proporcionando maior rendimento à empresa”.

O quadro 3 apresenta a estrutura do DFC, bem como evidencia suas diferenças em comparação ao DRE (quadro 2), pois desconsidera contas tais como depreciação, que existiram, mas não provocaram desembolso de caixa.

Segundo Iudícibus (1990: 101), a vantagem do DFC é sua maior capacidade de considerar movimentações nas contas de patrimônio, fornecendo uma visão mais próxima da realidade da empresa. Isto apesar de não considerar despesas que tenham existido mas que não tenham provocado desembolso de caixa.

Ross (1998: 46) afirma, ainda, não haver uma padronização referente à apresentação do fluxo de caixa, variando de acordo com o interesse e as necessidades de informações.

Segundo Borinelli (1998), as informações financeiras são as mais utilizadas pelos administradores das pequenas empresas, isto devido principalmente a sua facilidade e simplicidade de manipulação.

Uma diferença da DFC em relação ao BP e o DRE refere-se ao regime da contabilidade. Enquanto os demonstrativos contábeis trabalham com o regime de competência, a DFC trabalha com o regime de caixa.

### **3.1.2 OS REGIMES DE CONTABILIDADE**

Iudícibus (1990: 92) diz que “a contabilidade se utiliza de duas maneiras distintas para a apuração do resultado. Estas duas maneiras distintas de apurar o resultado denominam-se Regimes de Contabilidade.”

- Regime de Competência;
- Regime de Caixa.

O regime de competência é, segundo Iudícibus (1990: 92) e Marion (1989: 94), universalmente aceito e recomendado pela legislação do imposto de renda e para utilização em demonstrativos contábeis.

As regras básicas para a contabilidade neste regime são:

- A receita será contabilizada no período em que for gerada;
- A despesa será contabilizada como tal no período em que for consumida, incorrida ou utilizada, independentemente do seu pagamento.

Já o regime de caixa é, segundo Iudícibus (1990: 93) e Marion (1989: 95), uma forma simplificada de contabilidade, geralmente utilizada para o controle financeiro simples. É normalmente utilizado em pequenas empresas ou em entidades sem fins lucrativos. As regras básicas para a contabilidade neste regime são:

- A receita será contabilizada no momento de seu recebimento;
- A despesa será contabilizada no momento do pagamento, ou seja, quando sair dinheiro do caixa.

### **3.1.3 O PLANO DE CONTAS**

Iudícibus (1990: 235) afirma que “o plano de contas é o agrupamento ordenado de todas as contas que são utilizadas pela contabilidade dentro de uma determinada empresa. Portanto, o elenco de contas é considerado indispensável para os registros de todos os fatos contábeis. Cada empresa, de acordo com sua atividade e seu tamanho ... deve ter seu próprio plano de contas.”

Segundo Marion (1990: 236), “o plano de contas é numerado ou codificado de forma racional, o que facilita a contabilização através de processos mecânicos ou processos eletrônicos.”

O objetivo do plano de contas é realizar o agrupamento de todas as contas que serão utilizadas pela contabilidade dentro da empresa. Sua indispensabilidade advém da necessidade de organizar e padronizar os registros de todos os fatos contábeis.

Devem ser consideradas todas as atividades que a empresa venha a realizar no período. Pode-se no entanto alterar o plano de contas durante o ano fiscal, necessitando-se para isto definir quais as contas que serão agregadas, inserindo-as ao contexto do plano.

Segundo Marion (1989: 150), apesar de haver a possibilidade de utilizar o plano de contas de forma genérica, em todos os tipos de empresa, aconselha-se adaptar o plano à situação e realidade da empresa em questão, pois desta forma obtém-se um plano direcionado e especificamente criado para uma situação em especial, o que o torna mais útil à empresa.

Na prática, a pretensão é a definição de um plano de contas que padronize e organize os procedimentos e análises dos dados disponibilizados, bem como facilite a compreensão e obtenção das informações relacionadas aos dados em questão.

Marion (1989: 150) e Iudícibus (1990: 236) comentam ainda as possibilidades (e vantagens) de utilizar um plano de contas mínimo (ou simplificado), onde seriam

utilizadas somente as contas necessárias para um gerenciamento básico, possibilitando ainda a eliminação da codificação.

Uma forma de codificar o Plano de Contas é seguir os passos propostos por Marion (1990:236):

- Inicia-se com a unidade 1 para as contas do ativo, 2 para contas do passivo, 3 para o PL, 4 para as contas de receita e 5 para as contas dedutivas no resultado;
- Em seguida adiciona-se um segundo número que representa o grupo de contas do ativo, do passivo e assim por diante;
- terceiro dígito significa a conta do grupo.

O número de níveis do plano de contas dependerá da empresa, variando geralmente de 4 a 6 níveis, não sendo o número de níveis um fator determinante na qualidade do plano utilizado.

A codificação agiliza o registro dos eventos contábeis, ao utilizar um código que representará os débitos e créditos, no entanto deve-se definir métodos a serem utilizados que padronizem e viabilizem o armazenamento destes eventos contábeis de forma correta.

Neves (1997: 56) define o lançamento ou evento contábil: “É o registro de um fato contábil, sendo representado pela partida, ou seja, a expressão contábil que apresenta um fato pelo seu registro”.

#### **3.1.4 A METODOLOGIA DAS PARTIDAS DOBRADAS**

O utilização de partidas dobradas, segundo Iudícibus (1990: 250), é importante pois “... o método consiste no fato de que para qualquer operação há um débito e um crédito de igual valor ou um débito (ou mais débitos) de valor idêntico a um crédito (ou mais créditos). Assim, não há débitos sem créditos correspondentes.”

Neves (1997: 29) justifica as partidas dobradas ao afirmar que “a essência deste método, universalmente aceito, é que o registro de qualquer operação implica que um débito em uma ou mais contas deve corresponder a um crédito equivalente (de mesma monta), em uma ou mais contas, de forma que, se somados os valores debitados tem-se o mesmo valor da soma dos valores creditados”.

Marion (1989: 225) apresenta o formato de registro chamado de Diário Bicolunado, semelhante ao método de partidas dobradas apresentado por Neves (1997: 29) e Iudícibus (1990: 250).

As diferenças do Diário Bicolunado referem-se a aspectos informativos da informação contábil, sendo tecnicamente equivalentes ao método das partidas dobradas.

Segundo Iudícibus (1990: 250), a grande vantagem da técnica de partidas dobradas é a possibilidade de detecção de erros nos lançamentos contábeis, tanto no momento de cadastro como no instante de análise dos demonstrativos contábeis.

### **3.1.5 OS ESTOQUES**

Analisando o Quadro 1, exposto no item 2.2.2.2, destaca-se a necessidade da consideração dos estoques quando da elaboração do BP.

A importância da análise e consideração dos estoques da empresa, segundo Iudícibus (1997: 163), advém do fato de que "... os estoques representam um elemento altamente relevante do ativo de muitas empresas e um ponto crítico para a determinação do resultado do período. Sua avaliação a valores de saída, momento de realização líquida, é justificável nos estoques destinados à venda, embora possa estar destinada a dificuldades de estimativas dos custos e despesas adicionais."

O termo estoque é utilizado para designar propriedades relativas a:

- produtos para venda direta;
- em processo de produção;
- materiais a serem consumidos para a produção de produtos a serem vendidos.

Segundo Marion (1989: 271), "o grupo de contas Estoques assume grande importância no contexto do Balanço Patrimonial e seus efeitos são imediatamente sentidos no Patrimônio Líquido. Isto justifica a necessidade de demonstrar as movimentações na DRE, principalmente nos casos das empresas onde ocorram movimentações constantes dos produtos."

### **3.2 A GERÊNCIA DA INFORMAÇÃO NAS PEQUENAS EMPRESAS**

A popularização da informática, segundo Frey (1998), possibilitou a informatização de empresas de pequeno porte, disponibilizando o acesso a ferramentas de gerenciamento da informação antes só disponíveis a grandes corporações.

Segundo Beraldi (2000), esta informatização exigirá uma mudança na forma e nos métodos empresariais, sendo necessária uma abordagem baseada em roteiro básico, a fim de que o processo transcorra com naturalidade e eficiência:

Etapa 1 – Aprender sobre informatização

Etapa 2 – Contatar com pessoas especializadas

Etapa 3 – Analisar o processo atualmente utilizado na empresa

Etapa 4 – Analisar os processos que podem e devem ser informatizados

Etapa 5 – Analisar a utilização de uma rede interna

Etapa 6 – Definir claramente os objetivos da informatização

Etapa 7 – Definir quais serão os custos e benefícios envolvidos

Etapa 8 – Estabelecer um cronograma para a informatização

Etapa 9 – Preparar os funcionários à informatização

Etapa 10 – Definir quais serão os softwares e hardwares necessários

Beraldi (2000), relaciona algumas vantagens agregadas à informatização nas pequenas empresas:

- Melhora as informações utilizadas na tomada de decisões;
- Melhora o controle interno das operações;
- Ajuda o gerente a testar algumas decisões antes de colocá-las em prática;
- E principalmente, aumenta a competitividade e a produtividade.

Carmo (1999) afirma que “em geral nas pequenas empresas, a tomada de decisão para a formulação de uma estratégia empresarial está a cargo de uma pessoa, o seu dono. Neste contexto as empresas possuem conhecimentos reduzidos das oportunidades e ameaças do mercado externo, ignorando estes fatores na tomada de decisão”.

Ainda segundo Carmo (1999), “... esta realidade das pequenas empresas favorece a centralização das decisões, tornando muito característica a relação da empresa com seus funcionários e clientes, em função do estilo e personalidade do dirigente”.

Esta visão de gerência centralizada é também exposta por Frey (1998), ao afirmar que as informações e decisões da pequena empresa centralizam-se no proprietário.

Para fins gerenciais, os relatórios utilizados pelo gestor da pequena empresa devem, segundo Kassay (apud Frey, 1998), observar algumas características, tais como:

- a) Simplicidade
- b) Facilidade de obtenção
- c) Relevância
- d) Atualidade
- e) Possibilidade de efetuar simulações

A importância destes relatórios gerenciais está relacionada a obtenção de informações administrativas, visto a necessidade de apresentar de forma organizada os dados armazenados no sistema.

Desta forma, as informações administrativas proporcionam um sistema eficaz de gerenciamento da empresa, principalmente ao tratar-se de informações contábeis e financeiras.

### **3.3 A CONTABILIDADE NA GERÊNCIA DA PEQUENA EMPRESA**

Segundo Dantas (1996: 80), “a contabilidade é indispensável e exerce um papel central em qualquer empresa bem administrada. Ela é inseparável da administração.”

Resnik (1990: 137) cita que “ a falta de um sistema eficaz de contabilidade não é apenas um problema contábil, é um problema administrativo.”

A importância de um sistema que processe a informação da pequena empresa, transformando-a em conhecimento, é explicada por Carmo (1999):

“Um sistema de informações que contemple não somente as informações necessárias a administração, mas que possa diagnosticar e controlar aspectos fundamentais para o sucesso da empresa representa um suporte fundamental para a gestão e planejamento, proporcionando condições favoráveis para alcançar maior produtividade, eficiência e agilidade na tomada de decisões gerenciais e estratégicas da organização”.

A complexidade da manipulação dos eventos contábeis, e da própria contabilidade, é apontada por Dantas (1996: 80) como a razão para que pequenos empresários desconsiderem a contabilidade como uma ferramenta administrativa.

Dantas (1886:80) afirma ainda que “a maioria dos pequenos empresários evita a contabilidade por considerá-la uma tarefa incômoda, [...], no entanto, operar sem contabilizar todas as despesas e receitas é navegar sem bússola; na maioria das vezes a viagem termina num desastre ou não atinge o destino desejado.”

A informação gerencial, segundo Resnik (1990: 137), tem a capacidade de expor a situação financeira da empresa, tendo maior significância quando considerada no contexto da empresa de pequeno porte, devido a sua restrita disponibilidade de recursos.

Conforme Frey (1998), a maioria das pequenas empresas preocupa-se somente em realizar a apuração de impostos e contribuições, ignorando a escrituração contábil.

Resnik (1990:161) centraliza a atenção no gerenciamento de algumas informações contábeis básicas, tais como o controle de caixa, o controle de estoques e as contas a pagar e a receber.

Para Frey (1998), o planejamento empresarial deve basear-se na análise das informações gerenciais que, por sua vez, são obtidas a partir dos registros contábeis e financeiros da empresa.

Evidencia-se a necessidade da utilização de mecanismos de busca de informações especializadas, oriundas de métodos direcionados ao seu contexto, tais como:

- Criação de Relatórios totalizados e específicos;
- Mecanismos de auxílio a organização e extração de relatórios;
- Utilização de *software* especializados, tais como planilhas de cálculo e editores de texto, que otimizem o formato da informação gerada.

### **3.4 O PEQUENO EMPRESÁRIO E O CONTABILISTA**

Frey (1998) justifica a importância do relacionamento entre o contabilista e o pequeno empresário ao afirmar que “o hiato existente entre o pequeno empresário e o contabilista nos remete à hipótese de que o empresário não conheça o potencial da contabilidade como um sistema de informação, e que também tenha havido constrangimento por parte do contabilista, para responder a esta questão. “

Marion (1989: 30), ao tratar da utilização das informações contábeis, afirma que “a função básica do contabilista é produzir informações úteis aos usuários para a tomada de decisões. Ressalte-se, entretanto, que, em nosso país, em alguns segmentos da nossa economia, principalmente na pequena empresa, a função do contabilista foi distorcida, estando voltada exclusivamente para satisfazer às exigências legais”.

Carmo (1999) afirma que “a função do contabilista, principalmente nas pequenas empresas, deve ser mais consultiva, não apenas a de elaborar uma escrituração. As empresas de maior porte possuem uma contabilidade gerencial, que supre as lacunas deixadas pela contabilidade elaborada para atender a legislação fiscal e tributária.”

O contabilista pode, de acordo com Resnik (1990:138), fornecer informações gerenciais, uma vez que a maioria utiliza-se de métodos informatizados no processo da escrituração. Estas informações são imprescindíveis para o gerenciamento racional e eficaz por parte do empresário.

Resnik (1990:141) aponta problemas do suporte do contabilista à empresa:

- número de clientes é grande, e a disponibilidade de tempo para o atendimento é limitado em poucas horas por mês;
- Alguns contabilistas não acreditam na eficácia dos relatórios financeiros como uma ferramenta administrativa;
- Muitos contabilistas não apresentam preocupação com a adequação da contabilidade ao contexto de seu cliente;
- Poucos contabilistas de pequenas empresas têm interesse ou inclinação para ajudar a estabelecer controles contábeis básicos baseados em números;
- Embora um contabilista experiente possa ser uma fonte inestimável de conselhos práticos para sua empresa, poucos utilizam-se na prática destas recomendações;
- Muitos contabilistas de pequenas empresas não apresentam conhecimentos realmente profissionais, tanto em aspectos tributários quanto relativos ao planejamento empresarial baseado em suas informações gerenciais.

A análise do relacionamento do contabilista com o empresário permite concluir que o empresário deve concentrar-se em buscar as informações gerenciais necessárias à correta compreensão de sua empresa, sejam elas oriundas do contabilista ou de mecanismos internos.

## **3.5 O APOIO COMPUTACIONAL PARA O GERENCIAMENTO DA PEQUENA EMPRESA**

### **3.5.1 INFORMATIZAÇÃO NAS PEQUENAS EMPRESAS**

Beraldi (2000) afirma que o processo de informatização de uma pequena empresa deve ocorrer com vista à utilização das informações que esta esteja utilizando, a fim de que os processos possam ser informatizados (otimizados) trazendo um mínimo de alterações nos procedimentos da empresa e melhorando a utilização de seus recursos.

Segundo Hamacker (1998), o administrador deve concentrar-se na análise das informações realmente relevantes ao seu gerenciamento.

Esta ideia é defendida por Basil (1982: 101), ao referir-se à escolha de métodos e rotinas utilizadas na informatização. Desta forma, considera-se o problema da informatização dos métodos atualmente utilizados da empresa, alterando-se somente a forma de tratamento, e não o método de trabalho ou os dados manipulados.

Segundo Hamacker (1998), as informações gerenciais utilizadas em empresas de pequeno porte estão direcionadas à gerência financeira, tipicamente relacionadas ao controle das contas financeiras, caixa e bancos.

De acordo com Frey (1998), nas pequenas empresas a contabilidade é colocada em segundo plano, sendo considerada informação de pouca utilidade, aplicável somente ao ambiente burocrático (fiscal) da empresa.

Esta situação é justificada por Basil (1982: 92) ao afirmar que a pequena empresa concentra sua análise em informações aplicáveis ao seu gerenciamento imediato, o que de certa forma induz a utilização de métodos e técnicas 'rápidas' de gerenciamento, tais como os controles financeiros, visto que os controles contábeis são considerados complexos e dispensáveis neste cenário.

Em decorrência deste contexto, a pequena empresa acaba por utilizar *software* voltado ao gerenciamento exclusivamente financeiro, ignorando os aspectos contábeis.

### **3.3.2 CARACTERÍSTICAS DO SOFTWARE VOLTADO PARA A GESTÃO**

Alguns *softwares* comerciais estão disponíveis, inclusive via internet, com o interesse de auxiliar o gerenciamento de pequenas empresas.

Estes programas disponibilizados apresentam algumas características genéricas, referentes ao seu funcionamento e sua operacionalidade.

Assim, a análise destas características, possibilita a avaliação destes *softwares*, segundo alguns fatores de qualidade apontados por Pressman (1995) e Heckel (1993).

No item 2.3.2, a qualidade do software é descrita como indefinível, onde a medição é relativa a um aspecto ou circunstância em especial. Neste sentido, Antonioni (1995) afirma que a qualidade é algo de difícil definição.

Pressman (1995: 724) expõe que a conformidade com os requisitos e objetivos pode ser considerada um fator de diferenciação de qualidade.

Desta forma, alguns programas foram adquiridos, via internet, instalados e analisados, a fim de determinar quais seriam suas principais características de funcionamento e a sua aplicabilidade ao contexto das pequenas empresas.

Dentre os programas analisados, alguns não obtiveram êxito em seu processo de instalação ou apresentavam falhas em sua execução, fatos que impediram sua análise.

No entanto, diversos programas foram instalados e analisados, com o intuito de definir algumas características que fossem genéricas a estes programas.

Alguns dos softwares analisados foram:

Software	Fabricante	Objetivo – Enfoque
Contábil	Sadig	Gestão empresarial
Administração	Datacempro	Gestão empresarial
Gerenciamento	Gemini	Gerência
Contábil	Hábil	Gestão empresarial
Contas	Portal Software	Gerência Financeira

**Tabela 2 - Características dos Softwares atuais**

O grande número de variáveis a serem consideradas quando da análise de um programa, expostas por Antonioni (1995), somadas a dificuldade de definir métricas de qualidade, expostas por Pressman (1995: 725), direcionam a análise dos programas a aspectos subjetivos.

Desta forma, podem ser descritos, de forma genérica, algumas características dos programas disponibilizados, enfatizado os pontos fortes e as fraquezas destes programas.

Ao considerar que os *softwares* analisados visam suprir tanto pequenas como médias empresas, o cenário criado é o de uma gerência voltada ao gerenciamento financeiro. Assim, os *softwares* disponíveis, em sua maioria, voltam sua atenção a aspectos financeiros, deixando o gerenciamento contábil em segundo plano.

Outra característica comum aos programas citados na tabela 2, refere-se ao grande número de módulos existentes nos diversos programas, o que pode ser visto como uma desvantagem, principalmente ao se considerar usuários com pouca experiência.

Este grande número de módulos baseiam-se no interesse do fabricante em desenvolver um produto aplicável a diversas situações, o que do ponto de vista da qualidade é uma característica negativa, pois desconsidera a utilização de um foco na resolução do problema.

Da mesma forma, um grande número de módulos traz consigo problemas de unificação, descritos em 2.3.8 e 2.3.9, referentes não só à conectividade entre os módulos mas também a aspectos de manutenção do programa.

Finalmente, há de se considerar a problemática da complexidade do programa, exposta tanto por Heckel (1993: 12), relativa a eficiência da interface, quanto por Pressman (1990), ao considerar a funcionalidade do programa.

Assim, podem ser definidas algumas características fundamentais, ou indispensáveis, de softwares voltados à gerência de pequenas empresas.

### **3.5.3 O SOFTWARE INDICADO PARA A PEQUENA EMPRESA**

A análise das características apontadas em 3.3.2, somadas às necessidades expostas nos tópicos 3.1 e 3.2, permitem a definição de algumas características desejadas em *softwares* para pequenas empresas:

- Simplicidade de uso – O programa deve ser simples e direto, trabalhando e gerenciando informações realmente relevantes ao contexto do problema;
- Interface Otimizada – O programa deve ser voltado ao usuário, atendendo suas necessidades e restrições;
- Informação otimizada - Deve-se trabalhar exclusivamente com as informações necessárias, evitando trabalho desnecessário;
- Desenvolvimento orientação ao usuário – Os módulos devem ser desenvolvidos de forma que solucionem os problemas do usuário, e não os do desenvolvedor;
- Gerência financeira e contábil – Apesar do enfoque, claramente financeiro, é importante disponibilizar ferramentas contábeis. Deve-se ainda, definir métodos especializados de acesso e manipulação destas informações contábeis.
- Controle de estoques e notas fiscais de entrada e saída – A gerência e controle dos estoques e suas movimentações são de suma importância;

- Controle de clientes e fornecedores – Refere-se aos dados e informações básicas das pessoas físicas e jurídicas relacionadas à empresa;
- Relatórios e estatísticas gerenciais – O registro e a escrituração dos eventos da empresa são somente a entrada dos dados, há de se prover métodos eficientes de apresentação das informações da empresa, basicamente relatórios.

A análise das características apresentadas no item 3.3.2, que mesclam as técnicas e métodos de desenvolvimento do software, apresentadas no capítulo 2, e as informações financeiro-contábeis de pequenas empresas, apresentadas no capítulo 3, permitem a elaboração de um projeto lógico que atenda as necessidades gerenciais de uma pequena empresa.

De posse do projeto lógico, o desenvolvimento do protótipo implicará na codificação do projeto em uma linguagem de programação, baseando o desenvolvimento nas técnicas expostas no capítulo 2.

## **CAPÍTULO 4 – PROJETO LÓGICO PROPOSTO**

### **4.1 DEFINIÇÃO DOS OBJETIVOS**

O objetivo do projeto que será exposto nos próximos itens é o gerenciamento de informações financeiras e contábeis no contexto administrativo de pequenas empresas.

Conjuntamente a este cenário de apoio a manipulação dos dados, busca-se a definição de métodos de gerenciamento da informação de forma mais especializada.

### **4.2 DEFINIÇÃO DAS METODOLOGIAS UTILIZADAS**

#### **4.2.1 A METODOLOGIA DE DESENVOLVIMENTO E A SUA JUSTIFICATIVA**

O modelo de desenvolvimento utilizado foi a prototipação (item 2.3.5), sendo aplicada a metodologia de desenvolvimento monolítico (apresentado em 2.3.8).

A escolha do modelo de prototipação deve-se à forma que o programa foi desenvolvido: projeto e implementações rápida. Outra razão para a prototipação é a sua utilização para a validação da metodologia de conciliação do financeiro e a contabilidade.

A utilização da prototipação permite a validação dos módulos, dos métodos de tratamento da informação, bem como quanto a integração entre os diversos módulos.

#### **4.2.2 DESCRIÇÃO DOS PASSOS DE DESENVOLVIMENTO**

Inicialmente foi realizada a análise de requisitos (itens expostos em 3.1, 3.2, e 3.3), definição dos módulos funcionais (item 4.3), definição dos banco de dados (item 4.3) e a codificação em si, com a criação das telas e dos código de cada formulário.

Posteriormente foram realizados os testes e validações dos módulos desenvolvidos, sendo aplicado tanto os testes de caixa branca quanto de caixa preta.

#### **4.2.3 FASES DE DESENVOLVIMENTO**

Houve 4 etapas básicas de desenvolvimento, ao final das quais o protótipo foi considerado concluído, instante no qual é iniciada a etapa de manutenção:

- Desenvolvimento do Banco de Dados;
- Desenvolvimento das telas e módulos;
- Implementação de técnicas de racionalização da interface funcional e arquivo de ajuda;
- Execução de testes e montagem de banco de dados de exemplo.

O desenvolvimento ocorreu em quatro fases:

- Fase 1 – Criação do Banco de Dados
- Fase 2 – Desenvolvimento das telas e módulos
- Fase 3 – Racionalização da Interface
- Fase 4 – Criação de um banco de dados de exemplo

### 4.3 DESCRIÇÃO DOS MÓDULOS FUNCIONAIS

A figura 11 apresenta um modelo geral do protótipo, sendo apresentados os módulos bem como a conexão lógica da chamada de módulos.

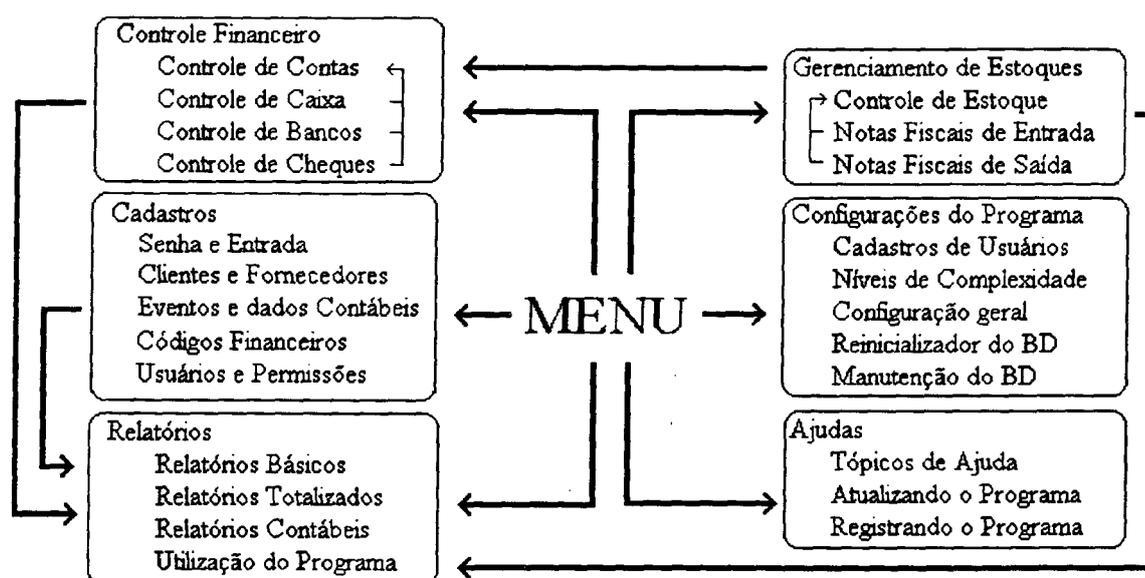


Figura 11 – Os módulos do protótipo desenvolvido

Na figura 11 tem-se os diversos módulos funcionais do protótipo, podendo ser diferenciados seis grandes grupos funcionais; controle financeiro, relatórios, cadastros, gerenciamento de estoques, configuração do programa e ajudas.

A seguir serão descritos em detalhes estes módulos, sendo apresentadas suas principais características funcionais, restrições e o BD do módulo, no anexo 1, tem-se o programa e manuais detalhados do protótipo.

No anexo 1, são amostradas algumas das principais telas do programa.

#### 4.3.1 SENHA E ENTRADA

Sempre que o sistema for acessado ou reconfigurado o usuário deverá informar sua senha e assim terá ou não autorização de acessar o sistema.

#### 4.3.1.1 Escolha de Usuário e Senha

O sistema é projetado para múltiplos usuários, sendo que cada usuário terá acesso somente à módulos que lhe forem autorizados. Desta forma, é possível organizar as informações e dar acesso somente aos módulos aos quais este tenha realmente autorização.

O gerenciamento dos usuários e suas senhas exige um agente superior de gerenciamento. O usuário denominado *supervisor* definirá os usuários, uma senha inicial e os módulos que este usuário terá acesso.

A tabela 3 é uma tabela codificada do Banco de Dados que tem por idéia armazenar controlar o acesso ao programa e a cada um de seus módulos. Cada usuário terá uma senha, sendo a partir da entrada da senha que o usuário terá acesso ao sistema.

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Código	Autonumeração		Controle	
Nome	Texto	120	Usuário	Codificado
Senha	Texto	120	Usuário	Codificado
Permissão	Texto	120	Usuário	Codificado
Bloqueio	Booleano		Usuário	Codificado
Data	Data / hora		Usuário	Codificado

**Tabela 3 – Tabela de Senhas e Permissões**

#### 4.3.1.2 Sistema de Backup e Compactação

Pressman (1995) aponta o backup (ou cópia de segurança) como sendo algo fundamental, sendo em alguns casos considerado imprescindível.

Desta forma, há de se prover uma metodologia de armazenamento de todos os dados do programa. Basicamente, o sistema copiará o banco de dados em disquete ou mesmo em outro computador via rede.

Intencionalmente, a função de backup será sugerida ao usuário antes do início das atividades, no instante em que o usuário estiver por entrar no sistema.

#### 4.3.1.3 Definição da Empresa e do Banco de Dados padrão

O sistema pode gerenciar diversas empresas utilizando para tal um BD padrão ou um banco de dados específico. O usuário poderá criar empresas alocando seus dados em um BD exclusivo. Cada empresa terá assim, seus dados, usuários e configurações próprias.

O usuário poderá inclusive alocar seu banco de dados em uma localização alternativa, na máquina onde trabalha, em outra máquina na rede ou mesmo em disquete.

### 4.3.2 CADASTRO DE CLIENTES

O cadastro de clientes tem diversas utilizações no sistema:

- Armazenamento de dados do clientes;
- Fornece os dados do cliente para diversas partes do sistema;
- Fornece dados adicionais;
- Impede o cadastramento duplicado de clientes.

No próprio módulo de cadastro de cliente há a possibilidade, variando o nível de complexidade, de visualizar as movimentações de produtos ou financeiras (contábeis ou gerenciais) do cliente escolhido, de forma rápida.

O usuário poderá optar por 5 níveis de visualização:

- Clássico em campos;
- Clássico em tabelas;
- Tabelas;
- Amostragem do Financeiro;
- Amostragem do Financeiro e Produtos.

A idéia de visualizar informações adicionais é algo pessoal, onde o usuário escolherá o melhor modelo a ser usado, de acordo com suas necessidades.

O banco de dados dos Clientes é apresentado na tabela 4:

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Código	Autonumeração	X	Controle	
Nome	Texto	90	Usuário	
Endereço	Texto	90	Usuário	
Cidade	Texto	40	Usuário	
Estado	Texto	3	Usuário	
E-mail	Texto	50	Usuário	
Fone1	Texto	12	Usuário	
Celular	Texto	13	Usuário	
CGC_CPF	Texto	15	Usuário	
RG_IE	Texto	15	Usuário	
Data	Texto	12	Usuário	
Tipo	Texto	10	Controle	Cliente / Fornecedor

**Tabela 4 – Tabela de Clientes e Fornecedores**

É aberta a possibilidade ao usuário de agregar ao sistema campos no banco de dados que julgue necessário. No entanto, os campos essenciais deverão ser mantidos. O usuário somente poderá acrescentar campos, utilizados como observações e para gerenciamento adicional.

Finalmente, é oferecida ao usuário a possibilidade de realizar buscas em vários campos da tabela, simultaneamente, de forma que encontre-se um registro em especial ou um grupo de usuários com uma determinada característica. Por exemplo, é possível encontrar o cliente *Carlos Silva* ou mesmo os *Silva* ou os que tenham *ilv* no nome.

### **4.3.3 CADASTRO DE FORNECEDORES**

O cadastro de fornecedores funcionalmente é idêntico ao cadastro de clientes.

O que difere é a origem dos dados, sendo diferenciado o acesso quanto ao tipo de conta (financeira e contábil) gerada. As informações amostradas relativas a movimentações de produtos e financeiras são genéricas, ou seja, são amostradas TODOS os registros, independente da origem da conta ser de cliente ou fornecedor.

O banco de dados dos Fornecedores é o mesmo utilizado pelos clientes, sendo o campo Tipo o campo onde ocorre a diferenciação do registro entre fornecedores e clientes.

### **4.3.4 CADASTRO DE CÓDIGOS**

Segundo Ross (1997), a utilização de códigos é uma forma eficiente de organizar as contas da empresa, possibilitando a definição de atividades e diferenciando os dados.

Assim abre-se a possibilidade de agrupar diversas contas em um formato que garantirá um nível organizacional adequado a elaboração dos relatórios que serão descritos no item 3.4.6.

#### **4.3.4.1 Codificação Financeira**

A codificação e criação de campos de códigos financeiros garante uma maior flexibilidade na extração de informações do sistema.

Os campos de codificação financeira são:

- Unidade - Centro de Custo, Unidade de Negócio ou Centro de Atividade;
- Grupo - Descrição Funcional;
- Conta – Descrição Analítica.

Os campos são inter-relacionados, ou seja, os campos são cadastrados na ordem unidade, grupo e conta. No entanto, no momento de extrair as informações a abertura e forma de organizar a busca é totalmente aberta, podendo-se realizar buscas em algum grupo ou mesmo conta de forma totalmente independente.

Assim, o usuário é quem definirá as informações que pretenderá gerenciar, o que indiretamente indicará a profundidade de conhecimento que terá sobre a empresa.

Cabe citar que os campos utilizados no sistema variam de acordo com o nível de complexidade definido pelo usuário, de acordo com o descrito nos tópicos a seguir. Assim, o usuário poderá definir quais serão os códigos que pretenderá utilizar no gerenciamento.

O banco de dados dos códigos financeiros está descrito na tabela 5

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Unidade	Texto	20	Usuário	
Grupo	Texto	20	Usuário	
Conta	Texto	20	Usuário	

**Tabela 5 – Códigos Financeiros**

A mesma tabela é utilizada para os três códigos financeiros, sendo o agrupamento das contas gerenciado pelo próprio programa.

#### 4.3.4.2 Plano de Contas Contábil

O plano contábil é simplificado, ou seja, não são abordados tópicos referentes a Imposto de renda ou depreciações (de forma direta).

O plano de contas contábeis se utilizará do cadastramento de dados gerenciais (item 3.4.5.1), sendo independente funcionalmente do plano financeiro.

O programa sugere os campos essenciais ao gerenciamento contábil básico, disposto a partir do plano de contas contábil simplificado (item 2.2.2.6), permitindo, no entanto, que novos campos sejam adicionados ao gerenciamento contábil.

O plano contábil é sugerido, propositalmente, em seu formato mínimo, ou seja, é proposto um plano contábil mínimo, que permita a realização de uma contabilidade gerencial, deixando aberta a possibilidade de definir e criar novos códigos contábeis.

A tabela de armazenamento dos códigos contábeis é exposta na tabela 6. Esta tabela armazena diversas informações de cada conta contábil. Dentre as informações armazenadas estão o saldo e o dia do fechamento de saldo (final do exercício anterior).

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Codreduzido	Número	20	Controle	
Classe	Texto	20	Usuário	Definição formal
Contamae	Texto	20	Usuário	Conta nível superior
TA	Texto	2	Usuário	Título / Analítico
Tipo	Texto	2	Usuário	Balanco / Relatório
Nível	Número	X	Usuário	Nível Hierárquico
Nome	Texto	50	Usuário	A conta em si
Dia	Data/hora	X	Sistema	
Saldo	Moeda	X	Sistema	

**Tabela 6 – Tabela de Códigos Contábeis**

O usuário já encontrará disponibilizados registros referentes ao plano de contas mínimo do sistema, ou seja, o plano de contas já estará cadastrado, sendo definido pelo sistema, não havendo a possibilidade do usuário alterá-lo.

#### **4.3.5 CONTROLE FINANCEIRO**

O objetivo do programa desenvolvido é a extração de informações de forma inteligente, baseados nos relatórios do sistema. Porém, para que sejam extraídas estas informações, torna-se indispensável que dados sejam inseridos de forma padronizada.

Desta forma, foram separados quatro grandes áreas de controle, de acordo com a funcionalidade e área que se está sendo gerenciada, sendo elas descritas a seguir.

##### **4.3.5.1 Controle de contas financeiras – (Financeiro – Contábil)**

O gerenciamento financeiro é separado em dois grandes grupos, Financeiro e Contábil. O enfoque do controle financeiro é orientado a administração, trabalhando com os três códigos básicos descritos na tabela 5.

O controle financeiro é básico e essencial ao programa, sendo o sistema contábil ‘atrelado’ ao gerencial, de forma que alguns dados são inclusive compartilhados, tais como datas e valores, bem como os nomes e observações das contas geradas.

O controle financeiro, como exposto no item 4.3.4, trabalha com informações básicas, atendo-se a somente os dados ditos imprescindíveis ao gerenciamento contábil.

Isto se faz necessário pois está sendo relacionado um modelo financeiro, que não comportaria eventos contábeis que não provocassem desembolso de caixa.

Esta situação é explicada em detalhes no capítulo 3, ao justificar que o gerenciamento financeiro ignora eventos não sujeitos a desembolsos de caixa.

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Movi	Autonumeração	X	Controle	
Codigo	Número	X	Controle	Reduzido da classe
Dia	Data / hora	X	Controle	
Classe	Texto	20	Usuário	
Valor	Moeda	X	Usuário	
Obs	Texto	150	Controle	

**Tabela 9 – Tabela de Movimentações Contábeis de Débito**

A tabela 9 tem o modelo de escrituração dos movimentos contábeis de débito.

#### 4.3.5.2 Controle de caixa e movimentações

O controle de caixa tem por objetivo armazenar informações referentes às movimentações financeiras que envolveram o caixa da empresa (ou os caixas, dependendo da forma de organização, tais como unidades de negócios ou centros de custos).

A conciliação de caixa refere-se ao registro de toda e qualquer movimentação de caixa, garantindo que as datas dos movimentos registradas sejam exatamente as ocorridas.

O usuário definirá se deseja ou não utilizar a conciliação de caixa, sendo esta indispensável para o gerenciamento contábil.

O programa permite ainda que sejam definidos diversos caixas, a serem utilizados em cada unidade de negócio. Desta forma, o que se tem é a possibilidade de manter um controle de múltiplos caixas (uma para cada Unidade de Negócio) ou mesmo um caixa global, que trabalhará com todas as unidade simultaneamente.

A conciliação temporal do caixa (correto registro da data do evento) é um problema extra a ser resolvido, visto que é comum o problema de uma movimentação ser registrada com uma data diferente da ocorrida, o que impossibilitaria a criação de um livro caixa com fechamento de saldo diário.

Desta forma, há a necessidade de criar um método de controle de caixa que permita a totalização de saldos (diários) mesmo para lançamentos posteriores a data de totalização.

Por fim, todas as movimentações de caixa são armazenadas de acordo com a unidade de referência. É possibilitado ao usuário que este procure por uma determinada movimentação, baseando-se no valor ou na data da movimentação.

A utilização do controle de caixa exige a manipulação de três tabelas de banco de dados, uma tabela de cadastros de caixas, uma que registra os movimentos de caixa e uma para o armazenamento dos saldos finais dos caixas nos dias em que houverem movimentos.

As tabelas de caixas e movimentos são amostradas nas tabelas 10 e 11:

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Ncaixa	Autonumeração	X	Controle	
Caixa	Texto	30	Usuário	
Saldo	Moeda	X	Controle	

**Tabela 10 – Tabela de Caixas**

Cada registro da tabela 10 corresponde a um caixa gerenciado pelo programa. Caso o usuário considere necessário, poderá criar novos caixas, a fim de organizar suas movimentações de acordo com a unidade ou centro de custo.

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Codigo	Autonumeração	X	Controle	
Dia	Data / Hora	X	Controle	
Hora	Data / Hora	X	Controle	
Caixa	Texto	25	Usuário	
Registro	Número	X	Controle	Oriundo do financeiro
Valor	Moeda	X	Usuário	Valor movimentado
Tipo	Texto	1	Controle	Entrada ou Saída
Saldo	Moeda	X	Controle	Saldo final operação
Unidade	Texto	20	Usuário	
Grupo	Texto	20	Usuário	
Conta	Texto	20	Usuário	
Diamov	Data / Hora	X	Controle	Conciliação Temporal
Obs	Texto	150	Controle	
Obs2	Texto	150	Controle	

**Tabela 11 – Tabela de Movimentações de Caixa**

Analisando as tabelas 11 e 12, percebe-se que grande parte dos campos são utilizados pelo programa (controle). O fato destes campos serem gerenciados pelo programa evidencia a facilidade de operação do caixa, ao eximir o usuário da preocupação com a manipulação destas informações.

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Código	Autonumeração	X	Controle	
Dia	Data / Hora	X	Controle	
Saldo	Moeda	X	Controle	Saldo atual final
Ncaixa	Número	X	Controle	
Estado	Texto	10	Controle	Indica dia aberto

**Tabela 12 – Tabela de Saldos Diários de Caixa**

A tabela 12 armazena os saldos finais dos caixas ao final dos dias em que houverem movimentações no caixa. No entanto, algumas movimentações podem ocorrer após o expediente e deveriam ser consideradas neste mesmo dia.

Para solucionar o problema de lançamentos posteriores a solução utilizada é o do fechamento do caixa após os lançamentos terem sido realizados, mesmo que isso só ocorra em algum dia posterior a que se refere o lançamento.

Assim, um saldo de finalização diária do caixa é registrado para utilizações posteriores, em relatórios, ou mesmo para a visualização dos saldos diários finais.

#### 4.3.5.3 Controle de banco e movimentações

O controle bancário não só é semelhante ao controle de caixa (descrito no item 4.3.5.2) como trabalha inclusive com informações equivalentes, ou seja, tem-se um modelo funcional semelhante, com níveis de complexidade que permitem ou não a conciliação.

Apesar de funcionalmente semelhante, o controle é totalmente independente.

Assim pode-se trabalhar somente com o controle bancário ou somente com o controle de caixas, ou (e preferencialmente) com os dois, visto que o programa provê a troca de informações entre os dois módulos.

As únicas diferenças entre os módulos de caixa e bancários são relativos a informação gerenciada. Diferenças estas verificáveis na tabela 13, onde um número de conta é utilizado para diferenciar diversas contas em uma mesma instituição bancária.

Igualmente ao sistema de caixas três tabelas são necessárias, uma para o cadastro das contas, outra para as movimentações das contas bancárias e uma para o armazenamento dos saldos totais diários de cada conta bancária.

As tabelas de informações bancárias são mostradas nas tabelas 13, 14 e 15.

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Nbanco	Autonumeração	X	Controle	
Banco	Texto	30	Usuário	
Nconta	Texto	15	Usuário	
Saldo	Moeda	X	Controle	

**Tabela 13 – Tabela das contas bancárias**

Da mesma forma que a tabela 11, a tabela 14 é utilizada sempre que houver alguma movimentação bancária, sendo a maior parte das informações gerenciadas pelo programa, de forma automática.

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Codigo	Autonumeração	X	Controle	
Dia	Data / Hora	X	Controle	
Hora	Data / Hora	X	Controle	
Banco	Texto	25	Usuário	
Registro	Número	X	Controle	Oriundo do financeiro
Valor	Moeda	X	Usuário	Valor movimentado
Tipo	Texto	1	Controle	Entrada ou Saída
Saldo	Moeda	X	Controle	Saldo final operação
Unidade	Texto	20	Usuário	
Grupo	Texto	20	Usuário	
Conta	Texto	20	Usuário	
Diamov	Data / Hora	X	Controle	Conciliação Temporal
Obs	Texto	150	Controle	
Obs2	Texto	150	Controle	

**Tabela 14 – Tabela de Movimentos Bancários**

A tabela 15 tem a mesma funcionalidade, dados e objetivos da tabela 12, diferenciando-se somente na origem dos dados.

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Código	Autonumeração	X	Controle	Somente leitura
Dia	Data / Hora	X	Controle	Somente leitura
Saldo	Moeda	X	Controle	Saldo atual final
Nbanco	Número	X	Controle	Somente leitura
Estado	Texto	10	Controle	Indica dia aberto

**Tabela 15 – Tabela de Saldos Bancários Diários**

#### 4.3.5.4 Controle de Cheques Emitidos e Pré-datados

O módulo de gerenciamento de cheques permite não só a impressão dos cheques (em formulários bancários padrão), como também possibilita o armazenamento, em banco de dados, de todos os cheques emitidos.

De forma a otimizar o processo de criação do cheque, o módulo financeiro gera, segundo solicitação do usuário, um novo registro no módulo de cheque com informações relativas ao registro financeiro que esteja sendo movimentado.

Há a possibilidade ainda de realizar buscas nos cheques emitidos, a partir de vários dos campos cadastrados.

A impressão dos cheques segue o padrão de cheques atualmente existentes no mercado financeiro.

A tabela 16 expõe os campos da tabela de banco de dados:

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Código	Autonumeração	X	Controle	Somente leitura
Dia	Data / Hora	X	Controle	Somente leitura
Quem	Texto	50	Usuário	Nominação
Valor	Moeda	X	Controle	Saldo atual final
Nbanco	Número	X	Controle	Somente leitura
Conta	Texto	10	Controle	Indica dia aberto

**Tabela 16 – Tabela de registro de cheques emitidos**

Um cheque pode ou não ser armazenado no banco de dados, de acordo com a necessidade ou interesse do usuário.

### 4.3.6 RELATÓRIOS

O programa deve dispor de métodos para a extração das informações do sistema.

Quanto maior for a especialização dos relatórios, na amostragem das informações, maior será a importância da disponibilização de recursos de auxílio à utilização destes relatórios.

O recurso utilizado é a definição de cenários, que representarão relatórios específicos, tais como as despesas de uma unidade ou as receitas de um grupo ou vários grupos.

Um gerenciador de relatórios tem a finalidade de organizar os relatórios mais utilizados, evitando assim, que a cada vez que um relatório específico (cenário) seja necessário, este tenha que ser montado, bem como diminui as chances de um relatório ser montado de forma diferente da esperada.

Uma metodologia de armazenamento de cenários será implementada a fim de permitir que diversos relatórios possam ser criados e gerenciados. Isto se faz necessário devido ao grande número de relatórios que poderão ser criados pelo usuário.

Assim, ao criar um relatório (cenário) será possível armazenar sua estrutura com um nome que lhe seja sugestivo, pois desta forma, sempre que este relatório se fizer necessário ele estará disponível e não consumirá tempo do usuário em refazê-lo, bem como obterá exatamente o relatório e as informações que espera.

Os relatórios são diferenciados de acordo com a complexidade e funcionalidade, bem como com o tipo de informação que cada um deverá apresentar.

O armazenamento dos cenários exigirá que sejam utilizadas diversas tabelas de banco de dados para armazenar cada um dos cenários distintamente.

Serão necessárias ao todo 4 tabelas para a definição de cada cenário:

- Banco de dados de Relatórios;
- Unidades escolhidas;
- Grupos escolhidos;
- Contas escolhidas.

A tabela de armazenamento dos relatórios é descrita na tabela 17.

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Nrel	Autonumeração	X	Controle	
Texto	Texto	50	Usuário	Nome do Cenário
C1	Booleano	X	Controle	
C2	Booleano	X	Controle	
C3	Booleano	X	Controle	
C4	Booleano	X	Controle	
C5	Booleano	X	Controle	
C6	Booleano	X	Controle	
C8	Booleano	X	Controle	
C9	Booleano	X	Controle	
Da	Texto	15	Usuário	Intervalo de datas
Db	Texto	51	Usuário	Intervalo de datas
Vb	Texto	15	Usuário	
Va	Texto	15	Usuário	
Vbp	Texto	15	Usuário	
Vpa	Texto	15	Usuário	
Rb	Texto	15	Usuário	
Ra	Texto	15	Usuário	
Co1	Texto	15	Usuário	
Co2	Texto	15	Usuário	
Co3	Texto	15	Usuário	
Co4	Texto	15	Usuário	
N1	Booleano	X	Usuário	
N2	Booleano	X	Usuário	
N3	Booleano	X	Usuário	
N4	Booleano	X	Usuário	
Cw1	Número	X	Controle	Coluna do relatório
Cw2	Número	X	Controle	Coluna do relatório
Cw3	Número	X	Controle	Coluna do relatório
Cw4	Número	X	Controle	Coluna do relatório
Cw5	Número	X	Controle	Coluna do relatório

Cw6	Número	X	Controle	Coluna do relatório
Cw7	Número	X	Controle	Coluna do relatório
Cw8	Número	X	Controle	Coluna do relatório
Cw9	Número	X	Controle	Coluna do relatório
Cw10	Número	X	Controle	Coluna do relatório
Mês	Texto	10	Usuário	Relatório anual
Meses	Texto	10	Usuário	Relatório anual
Ano	Texto	10	Usuário	Relatório anual
Pro	Booleano	X	Usuário	
Com	Booleano	X	Usuário	
Rea	Booleano	X	Usuário	

Tabela 17 – Tabela de Padrões de Relatórios Financeiros

Dentre os diversos campos do relatório, alguns são gerenciados pelo programa, tais como o gerenciamento das colunas e tipos de contas, sendo outros dados controlados pelo usuário, que definirá quais serão as restrições do relatório, utilizando-se para tal de uma tela de auxílio a elaboração dos diversos cenários criados.

As tabelas de definição de unidade, grupo e conta que farão parte de cada cenário são mostradas nas tabelas 18, 19 e 20.

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Nrel	Número	X	Controle	Somente leitura
Cod_unidade	Número	X	Controle	Somente leitura

Tabela 18 – Tabela de unidades dos cenários

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Nrel	Número	X	Controle	Somente leitura
Cod_grupo	Número	X	Controle	Somente leitura

Tabela 19 – Tabela de grupos dos cenários

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Nrel	Número	X	Controle	Somente leitura
Cod_conta	Número	X	Controle	Somente leitura

Tabela 20 – Tabela de contas dos cenários

Estas tabelas 18, 19 e 20 armazenam quais serão os campos que serão analisados nos registros financeiros na elaboração do relatório.

O usuário poderá definir quais serão os campos a serem restringidos ou considerados na montagem dos relatórios.

#### 4.3.6.1 Relatórios Básicos (financeiros)

Assim, tem-se, primeiramente, os relatórios básicos que apresentam de forma direta os registros selecionados de acordo com as restrições repassadas pelo usuário.

O relatório utiliza-se basicamente de um intervalo de entrada e das restrições do programa, permitindo a realização de relatório sobre todos os campos considerados importantes do ponto de vista do gerenciamento.

#### 4.3.6.2 Relatórios Totalizados (financeiros)

Agregando funções SQL, tem-se os relatórios de totalização, que permitem extrair informações dos registros financeiros a partir da criação de três níveis de totais, sendo definido pelo usuário quais serão os campos totalizados bem como a ordem que a totalização deverá ocorrer.

Todas as restrições definidas pelo usuário serão aplicadas à busca, ou seja, a totalização ocorrerá mediante a chave de busca definida pelo usuário.

O modelo totalizado possibilita o agrupamento de contas, realizando a totalização em até 3 níveis de códigos, em cada um dos nível de totalização.

A relatório totalizado utiliza-se de um a três campos de totalização e de um intervalo de busca. O relatório aplica as restrições na elaboração dos relatórios.

#### 4.3.6.3 Relatórios da Tabela Cruzada (financeira)

O relatório da tabela cruzada permite a realização de uma tabela onde dois campos do modelo financeiro são cruzados a fim de se obter o relacionamento financeiro entre os registros existentes, podendo ser considerada uma ferramenta auxílio à análise, pois possibilita a descoberta de discrepâncias e erros de digitação, de uma forma objetiva.

A tabela cruzada utiliza-se de dois campos e de um intervalo de busca. Baseia-se nas restrições do programa e não implementa nenhuma rotina de ordenação, excetuando os campos que estão sendo analisados.

A tendência é a da criação de uma tabela esparsa, onde poucas células da tabela venham a ser preenchidas a cada montagem da tabela, indicando os valores que sejam comuns aos dois campos que estejam sendo analisados no relatório.

#### 4.3.6.4 Relatório mensal de totalizações financeiras

Segundo Stair (1996), um modelo de relatório de aplicável na gerência empresarial é os que permitem a realização de comparativos entre as atividades da empresa, ou métodos que permitam comparar fases ou épocas da empresa.

Walton (1993) comenta que toda a estratégia e planejamento da empresa deve ser baseada nos históricos de movimentações da empresa.

Os relatórios mensais permitem a realização dos relatórios totalizados, mantendo a estrutura de 3 níveis de totalizações com o diferencial da separação dos valores de acordo com o mês em análise. O que se tem é uma totalização mensal em vários níveis, das contas que cumprem às restrições sugeridas pelo usuário.

Tem-se as mesmas restrições do relatório totalizado com a diferença que não é utilizado um intervalo de datas, e sim o usuário informa qual o mês e ano inicial e quantos meses de relatório deverão ser realizados. Utiliza-se das demais restrições do programa normalmente, sendo a única diferença o modelo temporal, que não são agora duas datas, mas um mês e ano e um determinado número de meses.

#### 4.3.6.5 Relatórios contábeis

Os relatórios contábeis possibilitam a montagem de relatórios de totalização de registros, baseados nos eventos contábeis cadastrados no programa, sendo montado e apresentado no formato de células (tal como planilhas eletrônicas).

Os dados que constarão no relatório são os cadastrados no módulo financeiro (item 4.3.5.1) e os cadastrados no módulo de cadastro de eventos e dados contábeis (item 4.3.8).

Os relatórios contábeis utilizam-se dos códigos contábeis dos eventos registrados, a fim de permitir a totalização dos valores.

O usuário poderá definir n relatórios (ver exemplo no programa), sendo que cada relatório pré definido é independente dos demais, tendo sua própria configuração (nome, linhas, colunas e códigos), e permitindo que os relatórios sejam padronizados, mantendo um mesmo formato (que tenha sido considerado ideal pelos usuários).

Os relatórios totalizam os eventos de acordo com seus códigos, utilizando-se de alguns códigos de totalização, considerando os tipos de movimentos contábeis:

n → indica que o valor totalizado calcula:  $\sum \text{débitos} - \sum \text{créditos} = \text{valor}$

i → indica que o valor totalizado calcula:  $\sum \text{créditos} - \sum \text{débitos} = \text{valor}$

d → indica que o valor totalizado calcula o  $\sum \text{débitos} = \text{valor}$

c → indica que o valor totalizado calcula o  $\sum \text{créditos} = \text{valor}$

Exemplificando a totalização de códigos, são citados alguns itens de totalização:

n1 → o valor calculado corresponde a diferença entre o somatório dos eventos contábeis de débito menos o somatório dos eventos contábeis de crédito que tenham sido realizados que possuam o código contábil iniciando por 1.

d3.1 → o valor calculado é o somatório de todos os eventos contábeis de débito que tenham código contábil iniciando por 3.1

c3.1.1 → o valor calculado é o somatório de todos os eventos contábeis de crédito que tenham código contábil iniciando por 3.1.1

Assim, a totalização dos códigos ocorre quando o conteúdo da célula do relatório inicia-se por n, d, i ou c, de acordo com o exposto nos exemplos de totalização descrito anteriormente. Em caso da célula não iniciar-se por outra letra, esta célula não será processada, sendo seu texto repassada integralmente para o relatório criado.

A totalização dos eventos contábeis em um período permite a montagem de relatórios específicos, que utilizem o resultado destas totalizações para a montagem dos relatórios imaginados e montados pelo usuário, que não sejam padronizados, ou definidos formalmente, tal como despesas operacionais menos despesas com funcionários.

A montagem de relatórios contábeis especializados, que trabalhem com determinados códigos de forma totalizada, possibilita que o usuário obtenha os relatórios padronizados bem como o BP e a DRE, e defina novos relatórios que trabalhem com somente alguns códigos que tenha interesse.

O programa disponibiliza em seu banco de dados padrão (de exemplo), relatórios que exemplificam o funcionamento dos relatórios, apresentando um modelo de BP, de DRE, de diferenças entre receitas e despesas entre outros.

As tabelas de banco de dados utilizadas para o armazenamento dos relatórios contábeis (estrutura), são apresentadas nas tabelas 21 e 22.

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Nrela	Número	X	Controle	No do relatório
Título	Texto	50	Usuário	Título do Relatório
Colunas	Número	X	Controle	Número de colunas
Linhas	Número	X	Controle	Número de linhas

**Tabela 21 – Tabela de Padrões de Relatórios Contábeis**

A tabela 21 armazena os cadastros dos relatórios contábeis, armazenando os textos e codificações de totalização, bem como o número de colunas e linhas de cada um dos modelos de relatórios contábeis que o usuário manipular.

Na tabela 22 são armazenados os movimentos, ou conteúdos de cada célula dos relatórios contábeis, contendo um texto ou um código de totalização (iniciado por n, i, c ou d)

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Nrela	Número	X	Controle	Indica o relatório
Texto	Texto	50	Usuário	Informação básica
Coluna	Número	X	Controle	
Linha	Número	X	Controle	

**Tabela 22 – Tabela de Padrões de Relatórios Contábeis - Movimentos**

#### 4.3.6.6 Relatórios de Utilização do Programa

Uma característica do programa é que todas as movimentações ocorridas no programa, independentemente do usuário, são registradas e em caso de necessidade podem ser relatadas.

O armazenamento das informações de acesso ao programa e a seus módulos ocorre automaticamente, sendo os dados referentes ao acesso armazenados em uma tabela do banco de dados, descrita na tabela 23.

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Codigo	Autonumeração	X	Controle	Somente leitura
Dia	Data / hora	X	Controle	Somente leitura
Hora	Data / hora	X	Controle	Somente leitura
Módulo	Texto	20	Controle	Somente leitura
Operador	Texto	20	Controle	Somente leitura

**Tabela 23 – Tabela de Acontecimentos do Programa**

O que se tem é o registro de todos os acessos a módulos. Os dados armazenados são o nome do usuário, a data/hora e módulo acessado. O relatório baseia-se na definição de um usuário, um intervalo de datas ou no módulo que foi acessado.

Esta tabela é totalmente gerenciada pelo programa, não havendo nenhuma possibilidade do usuário alterar seus dados ou informações, sendo útil quando da necessidade de verificar o acesso a algum módulo que tenha sido alterado ou acessado.

O supervisor poderá 'limpar' a BD, excluindo os últimos 100 registros (+ antigos).

#### 4.3.6.7 Modelo de restrições utilizado

Os relatórios do programa baseiam-se na definição de um intervalo temporal, um intervalo de valores e em uma lógica que permite que vários registros de vários campos sejam selecionados no banco de dados que se está analisando. O que se tem como modelo de restrição é a possibilidade de selecionar alguns registros de cada campo. Assim tem-se como um padrão de relatório:

*campo1.(registro1 OU registro2 OU registro3 ...) E campo2.(registro1 OU registro2 ...)*

Um exemplo deste método de restrição de relatório é apresentado a seguir:

*Unidade =("Escritório" ou "Granja" ou "Casa") e Grupo =("Impostos" ou "Salário")*

### 4.3.7 GERENCIAMENTO DE ESTOQUES

O gerenciamento do estoque possibilita a manutenção dos estoques atualizados, ou seja, o usuário, a qualquer momento poderá extrair relatórios sobre qualquer um dos produtos do estoque.

Da mesma forma, sempre que houverem entradas ou saídas do estoque esta movimentação atualizará o estoque automaticamente.

#### 4.3.7.1 Controle de estoques

Assim, a gerência do estoque envolverá tanto a manutenção de dados atualizados de estoque como registrará as movimentações, de entrada e saída.

Na tabela 24 é apresentado o modelo do estoque de produto, não havendo separações ou agrupamentos de produtos. Alguns campos como o ICMS, IPI, despesas e lucro são utilizados pelo programa para calcular o valor final dos produtos.

O estoque é utilizado também na elaboração do Balanço Patrimonial, como descrito no Quadro 1, os valores de todos os produtos comporão o valor físico do estoque.

O controle do cadastro de produtos e seus estoques atuais é descrito na tabela 24.

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Código	Autonumeração	X	Controle	
Produto	Texto	60	Usuário	
Unidade	Texto	5	Usuário	
Quantidade	Número	X	Usuário	
Unidade	Texto	5	Usuário	
P_compra	Moeda	X	Usuário	Preço de entrada
P_venda	Moeda	X	Usuário	R\$ venda sugerido
Calculado	Moeda	X	Usuário	R\$ venda Calculado
Icms	Moeda	X	Usuário	Imposto em %
Ipi	Moeda	X	Usuário	Imposto em %
Despesas	Moeda	X	Usuário	R\$ despesa unitaria
Lucro	Moeda	X	Usuário	Lucro em %
Pedido	Número	X	Usuário	No do pedido

**Tabela 24 – Tabela de Produtos**

Os dois valores apresentados para a venda (P\_compra e P\_venda) permitem que seja utilizado tanto um valor sugerido pelo usuário como um valor calculado, com base nas despesas e no lucro pretendido sobre a mercadoria.

A tabela 25 apresenta o formato do registro das movimentações de estoque:

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Código	Autonumeração	X	Controle	
Codnota	Número	X	Controle	
Codigoproduto	Número	X	Controle	
Dia	Data / hora	X	Usuário	
Quantidade	Número	X	Usuário	
Unidade	Texto	5	Usuário	
Valor	Moeda	X	Usuário	
Tipo	Texto	1	Controle	Entrada / Saída
Quem	Texto	50	Usuário	

**Tabela 25 – Tabela de Movimentações de Produtos**

O registro das movimentações de mercadorias (estoque) gera a possibilidade da montagem de relatórios que refletirão as entradas e as saídas de produtos, em intervalos de datas ou por um cliente ou fornecedor em especial.

#### 4.3.7.2 As Notas Fiscais de Entrada e de Saída

O objetivo do controle das notas fiscais é facilitar a entrada e saída dos produtos no estoque. Além de padronizar as movimentações, ajuda na organização nos momentos de aquisições e saídas de produtos, armazenando informações relativas às movimentações.

O armazenamento das notas fiscais, tanto de entrada como de saída, exige a utilização de duas tabelas de banco de dados, a primeira registra as notas fiscais e a segunda armazena informações dos movimentos das notas fiscais.

As tabelas das notas fiscais de entrada e saída são muito semelhantes, sendo apresentado um modelo, que será utilizado para ambas, na tabela 26.

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Código	Autonumeração	X	Controle	Usado pelo sistema
Nnota	Número	X	Usuário	Número da nota
Nome	Número	X	Usuário	Usuário
Dia	Data / hora	X	Controle	Dia da operação
Venda	Moeda	X	Controle	Valor total da nota
Obs	Texto	100	Usuário	Observações
Orçamento	Booleano	X	Controle	Indica orçamento
Entregue	Booleano	X	Controle	Foi entregue
Pago	Booleano	X	Controle	Foi pago
Baixado	Booleano	X	Controle	Foi baixado
Impresso	Booleano	X	Controle	Foi impresso
Financeiro	Booleano	X	Controle	Criada conta

**Tabela 26 – Tabela de Notas Fiscais de Entradas e Saídas**

Os registros são criados no momento do preenchimento das notas fiscais de entrada e de saída. Sendo utilizados inclusive para o lançamento de entrada e saída do estoque, o que poderá acontecer no momento do preenchimento da nota fiscal, ou posteriormente.

Já a tabela 27 expõe a formato do registro das movimentações de produtos, considerando novamente que o registro de entrada e de saída utilizam dados semelhantes.

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Código	Autonumeração	X	Controle	Usado pelo sistema
Nnota	Número	X	Controle	Número da nota
Produto	Número	X	Usuário	
Quantidade	Número	X	Usuário	
Unidade	Texto	5	Usuário	
Unitario	Moeda	X	Usuário	
Icms	Moeda	X	Usuário	
Ipi	Moeda	X	Usuário	
Baixado	Booelano	X	Controle	Movimento estoque

**Tabela 27 – Tabela de Movimentos de Notas Fiscais de Entradas e Saídas**

A tabela 27 apresenta o formato dos registros de movimentações de produtos, tanto de entrada como de saída.

#### **4.3.8 CADASTRO DE EVENTOS E DADOS CONTÁBEIS**

Os eventos contábeis no programa podem ser cadastrados (escriturados) no módulo financeiro, de acordo com o descrito no item 4.3.5.1.

A tela é projetada de forma a que apresente todos os itens utilizáveis no Balanço Patrimonial e na Demonstração do Resultado do Exercício.

Porém, alguns itens tais como patrimônio líquido e alguns itens constantes no permanente (tais como imobilizado, ações, cursos e depreciações) não são considerados nos movimentos financeiros.

Desta forma, torna-se necessário viabilizar ao usuário o cadastro destes itens contábeis de forma que estes constem nos relatórios tais como o Balanço Patrimonial e a Demonstração do Resultado do Exercício.

Diversas tabelas no banco de dados serão utilizadas para armazenar estes dados contábeis, elas serão expostas nas tabelas a seguir, e armazenarão informações referentes aos registros contábeis da empresa.

As informações destas tabelas auxiliares serão utilizadas na montagem dos relatórios contábeis da empresa.

A tabela 28 apresenta o formato dos dados contábeis das Ações da empresa.

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Código	Autonumeração	X	Controle	Usado pelo sistema
Dia	Data/hora	X	Usuário	Número da nota
Descrição	Texto	100	Usuário	
Quantidade	Número	X	Usuário	
Unitario	Moeda	5	Usuário	
Total	Moeda	X	Usuário	
Anotações	Memorando	X	Usuário	

Tabela 28 – Tabela de Ações da empresa

A tabela 29 apresenta o formato dos dados contábeis das Reservas de lucros:

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Código	Autonumeração	X	Controle	Usado pelo sistema
Dia	Data/hora	X	Usuário	
Descrição	Texto	100	Usuário	
Total	Moeda	X	Usuário	
Anotações	Memorando	X	Usuário	

Tabela 29 – Tabela de Reservas de Lucros

A tabela 30 apresenta o formato dos dados contábeis dos cursos e da organização interna da empresa.

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Código	Autonumeração	X	Controle	Usado pelo sistema
Dia	Data/hora	X	Usuário	
Descrição	Texto	100	Usuário	
Nome	Texto	100	Usuário	
Depreciação	Moeda	X	Usuário	
Total	Moeda	X	Usuário	
Anotações	Memorando	X	Usuário	

Tabela 30 – Tabela de cursos e organização

A tabela 31 apresenta o formato dos dados contábeis das Lucros:

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Código	Autonumeração	X	Controle	Usado pelo sistema
Dia	Data/hora	X	Usuário	
Descrição	Texto	100	Usuário	
Total	Moeda	X	Usuário	
Anotações	Memorando	X	Usuário	

Tabela 31 – Tabela de Lucros

A tabela 32 apresenta o formato dos dados contábeis do Imobilizado:

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Código	Autonumeração	X	Controle	Usado pelo sistema
Dia	Data/hora	X	Usuário	Número da nota
Descrição	Texto	100	Usuário	
Quantidade	Número	X	Usuário	
Unitario	Moeda	X	Usuário	
Depreciação	Moeda	X	Usuário	
Total	Moeda	X	Usuário	
Anotações	Memorando	X	Usuário	

Tabela 32 – Tabela de Imobilizado

A tabela 33 apresenta os dados contábeis dos empréstimos a receber da empresa:

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Código	Autonumeração	X	Controle	Usado pelo sistema
Dinicio	Data/hora	X	Usuário	
Dfinal	Data/hora	X	Usuário	
Vinicio	Moeda	X	Usuário	
Vfmal	Moeda	X	Usuário	
Tipo	Texto	50	Usuário	
Banco	Texto	50	Usuário	
Contato	Texto	50	Usuário	
Anotações	Memorando	X	Usuário	

Tabela 33 – Tabela de Empréstimos a receber

A tabela 34 apresenta os dados contábeis dos empréstimos a pagar da empresa:

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Código	Autonumeração	X	Controle	Usado pelo sistema
Dinicio	Data/hora	X	Usuário	
Dfinal	Data/hora	X	Usuário	
Vinicio	Moeda	X	Usuário	
Vfinal	Moeda	X	Usuário	
Tipo	Texto	50	Usuário	
Banco	Texto	50	Usuário	
Contato	Texto	50	Usuário	
Anotações	Memorando	X	Usuário	

**Tabela 34 – Tabela de Empréstimos a pagar**

A tabela 35 apresenta o formato dos dados contábeis das despesas do exercício seguinte (DES) da empresa:

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Código	Autonumeração	X	Controle	Usado pelo sistema
Dcadastro	Data/hora	X	Usuário	
Ddespesa	Data/hora	X	Usuário	
Valor	Moeda	X	Usuário	
Razao	Texto	50	Usuário	
Observação	Texto	50	Usuário	
Anotações	Memorando	X	Usuário	

**Tabela 35 – Tabela de Despesas do Exercício Seguinte**

A tabela 36 apresenta o formato de escrituração dos dados contábeis das contas de capital:

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Código	Autonumeração	X	Controle	Usado pelo sistema
Dia	Data/hora	X	Usuário	
Descricao	Texto	100	Usuário	
Total	Moeda	X	Usuário	
Depreciacao	Moeda	50	Usuário	
Anotações	Memorando	X	Usuário	

**Tabela 36 – Tabela de contas de capital**

A tabela 37 apresenta o formato de escrituração dos dados contábeis das aplicações financeiras.

<b>Campo</b>	<b>Tipo</b>	<b>Tamanho</b>	<b>Utilização</b>	<b>Observação</b>
Código	Autonumeração	X	Controle	Usado pelo sistema
Dia	Data/hora	X	Usuário	
Descricao	Texto	100	Usuário	
Valor	Moeda	X	Usuário	
Razao	Texto	50	Usuário	
Observação	Texto	50	Usuário	
Anotações	Memorando	X	Usuário	

**Tabela 37 – Tabela de Aplicações Financeiras**

As tabelas citadas neste tópico são as responsáveis pelo registro de dados adicionados à contabilidade, manualmente pelo usuário. Os lançamentos destes dados podem ser feitos no módulo financeiro (4.3.5.1), no entanto, alguns lançamentos, tais como contas de patrimônio e depreciação, que não geram movimentações financeiras (desembolso de caixa ou de bancos) não podem ser lançados no financeiro.

#### **4.3.9 CONFIGURAÇÕES DO PROGRAMA**

O programa é apresentado de forma que já esteja preparado ao usuário inserir informações.

O que se tem é um banco de dados padrão, que contém informações referentes a uma empresa 'exemplo'. O usuário poderá, desta forma, analisar funcionamento do programa, tomando por base os dados de uma empresa exemplo.

Nesta situação hipotética tem-se dados que possibilitam a elaboração dos diversos relatórios que o sistema disponibiliza, havendo inclusive a possibilidade de acrescentar novos registros ao banco de dados.

##### **4.3.9.1 Cadastro de usuários e permissões**

Visto a potencialidade de funcionamento em ambiente multi-usuário, há a possibilidade de cadastrar-se múltiplos usuários, onde um agente de supervisão, chamado *supervisor* cria os usuários e fornece-lhes uma senha inicial, a qual pode ser alterada a qualquer momento pelo usuário, ou mesmo desabilitado pelo supervisor.

Cada usuário cadastrado terá acesso somente a módulos que lhe sejam designados pelo supervisor.

As restrições oferecidas pelo programa são opcionais, ou seja, sempre que um usuário é criado este terá acesso máximo a todos os módulos, ou seja, poderá acessar todos os módulos e realizar todas as operações de gerenciamento.

Caberá ao supervisor, definir quais os módulos a serem acessados pelo usuário, bem como as operações que este poderá acessar.

#### 4.3.9.2 Definição de Nível de Complexidade dos Módulos do Programa

O usuário poderá, a qualquer instante, alterar o nível de complexidade do programa, não sendo necessário, nem ao menos reiniciar o programa, nem mesmo do módulo em que tiver sido alterada a configuração. A tela de definição de complexidade permite alterar a configuração de qualquer módulo do programa.

#### 4.3.9.3 Configuração do Programa

Todas as informações referentes a empresa são apresentadas na forma clássica, onde as informações básicas a respeito da empresa. Estes dados da empresa serão utilizados na criação de relatórios e recibos.

Estas informações estarão codificadas, e serão consideradas o registro do programa, sendo que o funcionamento não será afetado no caso de inexistência do registro, mas os documentos gerados apresentarão uma mensagem de falta do registro.

#### 4.3.9.4 Reinicialização de Banco de Dados

A instalação inicial do programa disponibiliza um banco de dados de exemplo, mas, de acordo com a necessidade do usuário, o programa provê o recurso de 'zerar' o banco de dados, excluindo os registros existentes, deixando o programa pronto para a inicialização dos trabalhos.

O supervisor poderá escolher quais os módulos que serão inicializados e quais as tabelas serão zeradas, podendo realizar esta tarefa a qualquer momento de operação do programa. Somente o supervisor poderá realizar esta operação.

#### 4.3.9.5 Manutenção do Banco de Dados

De acordo com as necessidades, o usuário poderá inserir campos no BD.

Estes campos serão agregados na tabela original e servirão como campos auxiliares ou de observação. O usuário poderá acrescentar os campos manualmente ou utilizando o atualizador, bastando para tal executar uma versão mais recente do programa.

A atualização automática ocorre no momento em que uma versão mais recente do programa é executada. O banco de dados é verificado e todos os campos que não existam, ou que tenham sido renomeados são automaticamente criados no banco de dados, sem nenhuma intervenção por parte do usuário.

#### **4.3.10 AJUDAS**

Um paradigma relacionado a apresentação de um programa de computador é a questão do treinamento e a ajuda ao usuário.

Diversas técnicas são utilizadas pelas empresas desenvolvedoras com a finalidade de solucionar a problemática da exposição das potencialidades e funções disponibilizadas pelo programa criado. A metodologia convencional retratada por MacConell (2000) expõe a utilização de ajudas baseadas em índices, assistente, tutoriais e frames de auxílio.

No decorrer do desenvolvimento do programa serão utilizados diversas metodologias de auxílio ao usuário, tais como o Tooltipbox (caixa dinâmica que aparece junto ao cursor indicando a finalidade do objeto), arquivo de ajuda e o nome do objeto.

No entanto, a racionalização da interface ao usuário está intimamente relacionada ao desenvolvimento de uma tela amigável, do ponto de vista funcional.

Assim, a validação do próprio programa está diretamente relacionada a concepção de uma interface amigável, que exija poucos conhecimentos de informática, o que, 'dispensaria' a busca de soluções em metodologias de ajuda mais elaboradas.

Da mesma forma, o programa trabalha com conceitos básicos de contabilidade, de maneira a utilizar jargões e termos que sejam habituais ao usuário.

##### **4.3.10.1 Ajudas por Tópicos**

O principal método de auxílio a ser implementado será a ajuda em tópicos. Assim, ao clicar em F1, um arquivo contendo os principais tópicos será disponibilizado, para acesso aleatório do usuário. O arquivo de ajuda será desenvolvido em um programa auxiliar, específico para a confecção de arquivos de ajuda.

O objetivo do arquivo de ajuda é esclarecer ou solucionar prováveis dúvidas que o usuário possa encontrar no decorrer da utilização do programa.

O fato da ajuda apresentar-se de forma semelhante a outros programas comerciais, pode ser visto como uma grande vantagem ao considerar que este é um método amplamente difundido e aceito de ajuda ao usuário.

#### 4.3.10.2 Atualizando o Programa

A atualização do programa poderá ser realizada via internet.

Desta forma, será substituído o programa, o banco de dados, o arquivo de ajuda e demais arquivos de configuração necessários ao gerenciamento do programa.

#### 4.3.10.3 Registrando o Programa

O programa desenvolvido estará disponibilizado em um site na internet, e poderá ser instalado sem nenhum custo, não havendo nem mesmo a necessidade de registro.

Na operação básica, o programa poderá ser utilizado por um prazo de 18 meses não tendo nenhum custo ao usuário. Ao final deste período o programa solicitará, a cada período de 1 minuto que seja registrado, por meio de uma caixa de mensagem.

No instante em que o usuário registrar o programa, via e-mail, receberá uma senha de liberação do programa, a qual será gerada com base no nome da empresa, endereço, CGC e outras informações da empresa.

O usuário enviará por e-mail o nome da empresa e o CGC e receberá um e-mail de resposta com uma string de 150 caracteres, o qual, ao ser inserido no programa automaticamente libera-lo-á indefinidamente e sem restrições.

O usuário não poderá alterar este código (para repassá-lo para outra pessoa), pois as informações de sua empresa serão utilizadas nos documentos gerados pela empresa que esteja utilizando a cópia não registrada (corretamente).

## 4.4 REQUISITOS DO PROGRAMA

Algumas restrições são impostas, principalmente no tocante a necessidades de hardware e software mínimas ao funcionamento satisfatório do programa.

### 4.4.1 REQUISITOS DE HARDWARE

Não há uma definição de HW para o funcionamento do programa, sugere-se, no entanto, que a máquina tenha uma configuração mínima de um PC 486, 16 Mbytes de memória RAM, e espaço disponível em disco de 5 Mbytes. Mas certamente em máquinas com configurações 'melhores', o aumento da performance do programa será significativa.

Uma configuração considerada ideal é a de um Pentium II 300 ou K6-2 500 com 64 Mbytes, isto porque alguns relatórios necessitam de um poder de processamento considerável, dada a forma como os relatórios foram desenvolvidos. Cabe lembrar, no entanto, que trata-se apenas de uma questão de ‘tempo’ na geração do relatório.

O funcionamento em rede exige que seja instalado o programa nas máquinas que utilizarão o programa, e uma rede esteja operacional para o compartilhamento dos dados.

#### **4.4.2 REQUISITOS DE SOFTWARE**

O grande requisito de software necessário é o Sistema Operacional Windows 95 ou superior. Tendo-se a rede Microsoft Workgroup, a instalação do programa com um vínculo ou atalho de rede é requisito suficiente para a manipulação e gerenciamento dos dados em um ambiente de rede, de forma distribuída.

#### **4.4.3 SOFTWARES AUXILIARES**

A utilização de um SGBD tal, como o Microsoft Access, poderá ser útil, se considerada a possibilidade de manipular e gerenciar diretamente o banco de dados.

A utilização de editores de texto, tal como o Microsoft Word e o Microsoft Wordpad, para a visualização e armazenamento de relatórios auxiliaria o usuário e diminuiria os gastos com a impressão dos relatórios.

A utilização de planilhas eletrônicas, tais como o Microsoft Excel, auxiliam no gerenciamento e organização dos relatórios. Considerando-se que todos os relatórios podem ser exportados no formato de planilha eletrônica, as possibilidades de gerenciamento destes relatórios, bem como de sua visualização, expõem uma grande vantagem no tocante à organização dos relatórios e dos próprios dados gerenciados.

Portanto, sugere-se que o usuário se utilize do pacote Microsoft Office como ferramenta ADICIONAL para a manipulação dos dados e relatórios do programa.

### **4.5 DIAGRAMA DE FLUXO DE DADOS DFD**

Segundo Pressman (1995), o diagrama de fluxo de dados representa o fluxo dos dados dentro do programa.

Serão apresentados os dois principais fluxos, sendo o de 1º nível o diagrama de contexto, ou genérico. O segundo DFD trata da ‘expansão’ do DFD de 1º nível, sendo expandidas as funções e os dados do programa.

Os passos para a criação de um DFD são sugeridos por Rezende (1999: 182):

- Identificar e relatar os requisitos funcionais;
- Identificar as entidades externas;
- Associar fluxos de dados a cada entidade;
- Identificar as consultas, associando-as às entidades externas envolvidas;
- Desenhar o DFD;
  - Iniciar no canto superior esquerdo;
  - Deixar entidades externas nos cantos;
  - Origens à esquerda e destinos à direita;
  - Desenhar os fluxos, os processos necessários e os depósitos de dados;
  - Verificar se todas as entradas e saídas foram incluídas;
  - Associar manutenções necessárias e depósitos de dados;
  - Redesenhar o DFD, minimizando as interseções de fluxos de dados;
- Explodir ou derivar os processos complexos em níveis inferiores.

#### 4.5.1 DIAGRAMA DE FLUXO DE DADOS DE CONTEXTO

Segundo Pressman (1988: 180), o DFD é utilizado para particionar um sistema, sendo a principal ferramenta de análise estruturada e da especificação estruturada. O DFD de contexto visa apresentar os componentes de forma sucinta, como exposto na figura 12.

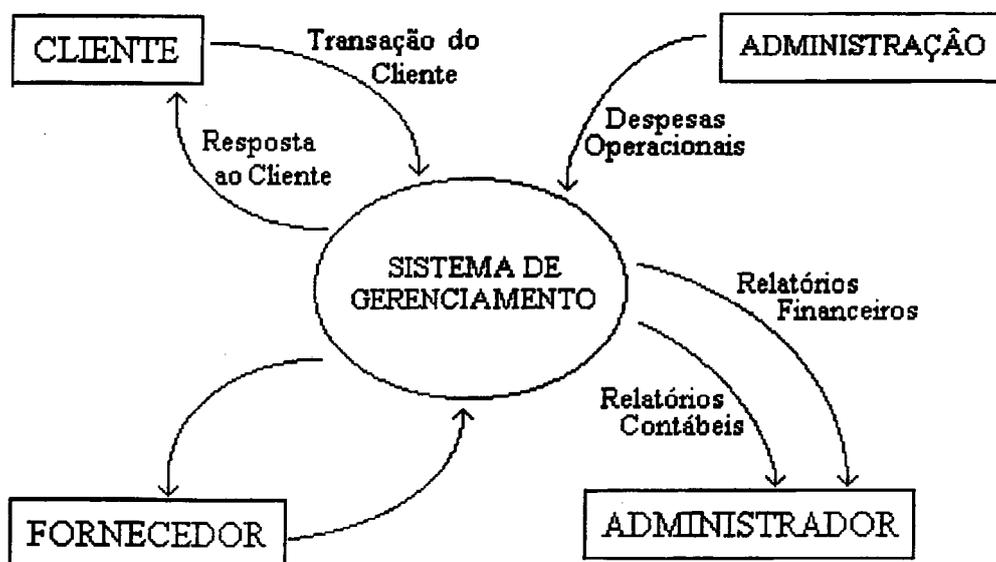


Figura 12 – DFD de Contexto do Programa Proposto

No DFD apresentado na figura 12, tem-se as entidades externas cliente e fornecedor. São também apresentadas a administração, que será a responsável pelo

gerenciamento das despesas operacionais e o administrador, que analisará os relatórios criados pelo programa,

#### 4.5.2 DIAGRAMA DE FLUXO DE DADOS DE PRIMEIRO NÍVEL

Segundo Pressman (1988: 185) , “a atividade de nivelamento do DFD consiste em detalhar as funções do DFD, particionando uma função em um conjunto de funções que serão funcionalmente equivalentes.”

A vantagem do particionamento é uma melhor compreensão das funções e atividades executadas na empresa. A figura 13 apresenta o particionamento do DFD apresentado na figura 11, sendo amostradas as funções do sistema de gerenciamento.

No DFD exposto na figura 13, são apresentados alguns módulos funcionais do programa. São apresentados os departamentos de compra e venda, bem como é evidenciado o papel centralizador da informação assumido pela contabilidade.

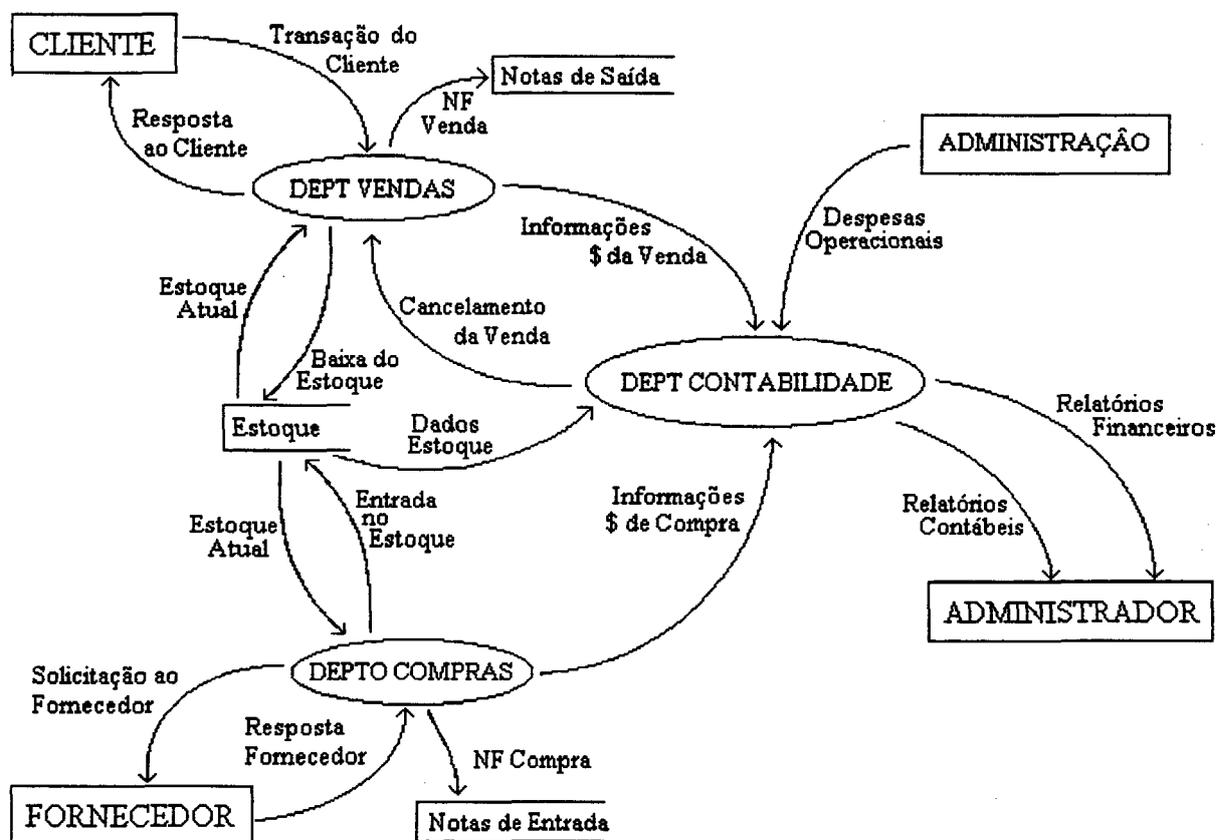


Figura 13 – DFD de primeiro nível do programa proposto

Percebe-se ainda que o administrador somente analisará as informações geradas, sendo utilizadas para a tomada de decisões por parte do administrador.

## **4.6 A FERRAMENTA DE DESENVOLVIMENTO**

A ferramenta utilizada no desenvolvimento foi Microsoft Visual Basic 6.0, sendo utilizado o Access 97 para o gerenciamento do Banco de Dados e o Vbhelp 2.0 para a criação do arquivo de ajuda.

## **4.7 DESENVOLVIMENTO DO PROTÓTIPO**

### **4.7.1 CONSIDERAÇÕES DO DESENVOLVIMENTO**

O protótipo foi desenvolvido em aproximadamente 410 horas de trabalho, sendo destinada a maior parte do tempo (60%) ao desenvolvimento das telas e da sua otimização. O restante do tempo foi destinado ao projeto lógico, o BD, os testes e as validações.

As alterações no BD não implicaram em alterações no modelo de gerenciamento.

Diversas foram as dificuldades enfrentadas no decorrer do desenvolvimento do protótipo de software, podendo ser citada a falta de críticas por parte do usuário final, como uma das principais dificuldades.

A otimização das telas exigiu a alteração dos métodos utilizados no acesso à informação, acarretando alterações nas metodologias de manipulação da informação. Esta atividade de otimização de telas exigiu um tempo superior ao previsto inicialmente.

Outro ponto a ser considerado é o da implementação das técnicas de otimização do acesso à informação, visto que quanto maior a facilidade gerada ao usuário, maior é a complexidade do algoritmo implementado.

Finalmente, a unificação dos módulos funcionais exigiu a realização de alterações nas estruturas dos módulos, devido, principalmente, às alterações decorrentes da otimização das telas e ao surgimento de novas instâncias do programa.

### **4.7.2 QUANTO AO CUMPRIMENTO DAS METAS**

Os resultados alcançados relativos ao gerenciamento financeiro e contábil permitem não só a utilização do programa em um nível de gerenciamento diferenciado, baseando-se em níveis de complexidade (questão da interface), bem como possibilita a utilização efetiva destas informações no âmbito da pequena empresa.

Finalmente, o protótipo proposto cumpre seu objetivo ao manipular as principais informações necessárias ao gerenciamento de pequenas empresas, de forma diferenciada e considerando a problemática envolvida no contexto deste segmento empresarial.

### 4.7.3 QUANTO AOS TESTES REALIZADOS

A metodologia de testes escolhida foi o Teste da Caixa Preta, descrita no tópico 2.3.10, onde são realizadas entradas e analisadas as saídas do protótipo.

No decorrer do desenvolvimento dos módulos, testes foram realizados a fim de validar o módulo em desenvolvimento.

Inicialmente os testes ignoraram o relacionamento do módulo testado com os demais módulos, sendo validados somente as funções no âmbito interno ao módulo. Em um segundo instante foram verificados os relacionamentos entre o módulo testado e os demais módulos os quais este interaja, sendo então analisadas as instâncias criadas.

Como eram de se esperar, falhas foram detectadas, tanto nos testes iniciais (internos) quanto na integração entre os módulos. Dependendo da falha ocorrida sua resolução exigiu a realização de alterações no código. A princípio as falhas encontradas eram tratadas exclusivamente internamente ao módulo, ignorando seu relacionamento com os demais módulos. No instante em que esta falha estivesse corrigida e o módulo estivesse funcionando corretamente, eram então realizados testes onde fossem verificados os relacionamentos entre os módulos os quais ocorresse alguma interação a nível de informação, tal como descrevem Melton (1997) e Pressman (1995: 286).

O relacionamento entre módulos com compartilhamento de informações exigiu, em alguns casos, a realização de novos testes no módulos relacionados ao módulo alterado. Pode-se exemplificar esta situação ao detectar alterações no formato dos relatórios devido a alteração do tamanho do campo de nome no cadastro de clientes.

O fato de trabalhar-se com um modelo de desenvolvimento monolítico (item 2.3.8) possibilitou uma grande coesão entre as informações gerenciadas, ou seja, o relacionamento da informação entre os módulos poderia ser analisada internamente ao código do programa, e não estaria condicionada ao acesso direto à informação.

Desta forma, os erros puderam ser minimizados, visto que o relacionamento funcional entre os diversos módulos estava condicionado às informações pertinentes ao módulo que estivesse sendo analisado. Esta vantagem é citada por Melton (1997) como a grande vantagem do modelo monolítico, tendo sido atestada no desenvolvimento.

Os procedimentos de testes e validações não foram documentados em detalhes, sendo que as alterações no projeto, tanto a nível de código quanto a nível de estrutura de banco de dados foram diretamente alteradas no projeto lógico.

Finalmente, realizou-se testes sobre o protótipo como um todo, onde a consistência do protótipo exigiu novamente alguns ajustes, devido ao surgimento de instâncias imprevistas.

#### **4.8 CONSIDERAÇÕES FINAIS**

O protótipo desenvolvido diferencia-se de outros modelos de gerenciamento de informações em alguns aspectos.

O desenvolvimento monolítico aliado a prototipação permitiu o desenvolvimento de um modelo diferenciado de software, onde a informação apresenta-se de forma clara (e otimizada) ao usuário, na tentativa de minimizar a resistência e maximizar a compreensão do usuário.

As informações gerenciadas referem-se somente às informações realmente necessárias, sendo utilizado um modelo mínimo de gerenciamento dos dados, sendo disponibilizada a possibilidade ao usuário definir a que nível de complexidade pretende trabalhar.

O próprio usuário poderá definir quanto a utilização de um modelo gerencial que somente se utilize de informações financeiras, ou de um modelo que baseado no gerenciamento financeiro, possibilite o cadastro dos eventos contábeis de forma otimizada, tomando-se por base para a escrituração contábil as próprias movimentações financeiras.

O programa é orientado tanto a usuários com pouca experiência como a usuários avançados, ou seja, permite que o usuário escolha do nível de complexidade, o qual pretenda trabalhar, sendo que os modelos mais simples exigem poucos conhecimentos e os modelos mais avançados permitem a elaboração de relatórios mais complexos, onde a informação é apresentada de forma mais elaborada e especializada.

Enfim, a aplicação das técnicas expostas nos itens 2.2 e 2.3, somadas a um modelo específico de gerência de informações (financeiras e contábeis) em pequenas empresas, possibilitou a criação de um protótipo que atendeu aos objetivos propostos.

## CAPÍTULO 5 – CONCLUSÕES E RECOMENDAÇÕES

### 5.1 CONCLUSÕES

Os conceitos estudados no presente trabalho possibilitaram a definição das principais informações necessárias ao gerenciamento de pequenas empresas, bem como definiram as técnicas a serem utilizadas no desenvolvimento de um protótipo do modelo.

A utilização de informações específicas voltadas ao âmbito da pequena empresa otimizam a manipulação dos dados, pois trabalham somente com as informações necessárias à sua gerência.

Com este propósito, foram propostos métodos diferenciados para a manipulação das informações, considerando-se os problemas enfrentados pelos pequenos empresários, tanto do ponto de vista tecnológico quanto de seu contexto empresarial.

Assim, elaborou-se um protótipo que teve por interesse validar os métodos propostos, ou seja, seria uma ferramenta especificamente criada para o auxílio a pequenos empresários, visto seu enfoque claro no gerenciamento simplificado da empresa.

O protótipo possibilita a gerência da informação contábil e financeira, evidenciando a importância da utilização da informação como uma ferramenta de apoio ao gerenciamento da pequena empresa.

O desenvolvimento do protótipo aliou a utilização das técnicas de desenvolvimento de software, expostas no capítulo 2, ao gerenciamento das informações financeiras e contábeis do contexto da pequena empresa, expostas no capítulo 3.

Algumas conclusões podem ser apresentadas quanto ao desenvolvimento do protótipo, considerando as informações manipuladas e o objetivo do trabalho.

Considerando-se a criação do protótipo, pode-se afirmar que a utilização dos métodos de desenvolvimento padronizados, que sigam as normas apresentadas no capítulo 2, trazem consigo as seguintes vantagens:

- facilitam o processo de criação do programa, pois definem os objetivos e as metodologias a serem utilizadas previamente, com maior flexibilidade;
- otimizam o desenvolvimento do programa, pois diminuem os custos de desenvolvimento, assim como o processo de validação e manutenção;
- restringem o desenvolvimento, centralizando e organizando o esforço;

- criam um modelo padronizado, orientado a futuras manutenções, baixando custos, diminuindo os prazos e aumentando a confiabilidade geral do programa;
- disponibilizam a documentação necessária ao processo natural de manutenção.

O protótipo desenvolvido também pode trazer vantagens, como:

- acesso otimizado a informações, enfocando somente o que será realmente utilizado, aumentando, assim, a satisfação do usuário e a aceitação do modelo proposto;
- centralização da atenção do usuário às informações e não no programa, exigindo pouca atenção à mecânica envolvida no manuseio do programa;
- desenvolvimento de um modelo monolítico compacto, apresentando somente as informações necessárias à resolução do problema proposto;
- utilização de uma metodologia diferenciada de enfoque à informação, possibilitando a definição do nível de gerenciamento (complexidade da informação) pelo usuário.

## 5.2 RECOMENDAÇÕES PARA TRABALHOS FUTUROS

Considerando-se o ambiente de trabalho de uma pequena empresa onde, por vezes, todo o serviço de escrituração é realizado por uma única pessoa, a utilização de mecanismos de auxílio na validação das entradas teria a capacidade de detectar discrepâncias no caso de entradas incoerentes.

Uma resolução envolveria a aplicação de conceitos de Inteligência Artificial, provavelmente Redes Neurais, conjuntamente com a teoria de Rough Set.

Os métodos agiriam em *background* (invisível ao usuário), e seriam apresentados e utilizados de acordo com alguma heurística baseada em alguma função derivada, provavelmente de métodos oriundos da inteligência artificial.

Assim, tem-se por sugestão quanto a trabalhos futuros, o desenvolvimento de um mecanismo que atue na monitoração dos dados no instante de sua entrada no sistema.

A contribuição do trabalho consistiria no desenvolvimento de uma metodologia de validação de entradas em um ambiente n-dimensional (n-códigos) inter-relacionado, baseado em um pré-processamento da informação.

A validação do modelo poderia ocorrer com base em um protótipo que utilizasse informações reais (de empresas), o que permitiria a definição de um grau de confiabilidade do algoritmo, bem como validaria a proposta do trabalho como um todo.

## REFERÊNCIAS BIBLIOGRÁFICAS

- ANTONIONI, José. Qualidade em Software ISO-9000. São Paulo: Makron Books, 1995.
- BASIL, Douglas. Organização e Controle da pequena empresa. Rio de Janeiro: Cultural, 1982.
- BERALDI, Lairce. Impacto da tecnologia de informações na gestão de pequenas empresas. São Paulo, 2000. São Paulo, 2000. Dissertação de Mestrado - Programa de Pós-Graduação, USP, 2000.
- BELLOQUIM, Átila. O Modelo SEI/CMM e o Processo de Desenvolvimento de Software. São Paulo, 1999. Programa de Pós-Graduação, USP, 1999.
- BORINELLI, Márcio Luís. A identificação do ciclo de vida das pequenas empresas através das demonstrações contábeis. Florianópolis, 1998. Dissertação - Programa de Pós Graduação em Engenharia da Produção, UFSC, 1998.
- CYBIS, Walter A. Ergonomia de Interfaces Homem-Computador. Florianópolis, Laboratório LabIUtil, UFSC, 2000.
- CARMO, Vadson B. Sistema de Informações Gerenciais para programa de qualidade total em pequenas empresas da região de campinas. Campinas, 1999. Programa de Pós Graduação, Unicamp, 1999.
- DART, Susan; CHRISTIE, Alan. A Case Study in Software Maintenance. Carnegie Mellon University 1993. Disponível na World Wide Web: <http://www.cs.cmu.edu/publications/thechnicalreports.html>. Acesso em 11/2000.
- DATE, C. Introdução a Sistemas de Banco de Dados. 4 Ed., Rio de Janeiro: Campus, 1991.
- DANTAS, Antônio. Análise de investimento e projetos aplicados à pequena empresa. Brasília, 1996. Programa de Pós graduação, Universidade de Brasília, 1996.
- ELMASRI, Rames; NAVATHE, Shankart. Fundamentals of Database Management System. New York: Addison-Wesley, 1994.
- FISH, Michael. Understanding the Process of Information Technology Implementation. Center for Research on Information Systems. Carnegie Mellon University, 1996. Disponível na World Wide Web: <http://www.cs.cmu.edu/publications/thechnicalreports.html>. Acesso em 11/2000.
- FREY, Irineu A. O uso de informações contábeis na pequena empresa. Cascavel, Programa de Pós Graduação, Unioesc, 1998.
- GARLAN, David; SHAW, Mary. An Introduction to Software Architecture. Carnegie Mellon University, 1994. Disponível na World Wide Web: <http://www.cs.cmu.edu/publications/thechnicalreports.html>. Acesso em 10/2000.

- HAMACKER, Silvio. Sistemas de Informações Gerenciais. Rio de Janeiro, Dissertação de Mestrado – Programa de Pós Graduação, PUC do Rio de Janeiro, 1998.
- HECKEL, Paul. Software Amigável. Rio de Janeiro: Campus, 1993.
- HEUSER, Carlos A. Projeto de Banco de Dados. Porto Alegre: Sagra Luzzato, 1998.
- IUDÍCIBUS, Sérgio. Teoria da Contabilidade. 5. Ed., São Paulo: Atlas, 1997.
- , Sérgio. Manual de Contabilidade para não Contadores. São Paulo, Atlas, 1990.
- KORTH, Henry F. Sistemas de Banco de Dados. 2. Ed., São Paulo: Makron Books, 1995.
- LAITINEN, Mauri; FAYAD, Mohamed. Software Engineerign in Small. IEEE Software Eletronic Magazine, New York, v.8, n.16, October 2000.
- LINDVAL, Mikael; RUS, Ioana. Process Diversity in Software Development. IEEE Software Eletronic Magazine, New York, v.8, n.16, October 2000.
- MACCONELL, Steve. The Best Influences in Software Engineering. IEEE Software Eletronic Magazine, New York, v.11, n.9, February, 2000.
- MARION, José C. Contabilidade Empresarial. 4. Ed., São Paulo: Atlas, 1989.
- MARTIN, James. Engenharia da Informação. Rio de Janeiro: Campus, 1991.
- NEVES, S., VICECONTI, P. Contabilidade Básica. São Paulo: Frase, 1997.
- MELTON, Ralph; GARLAN, David. Architectural Unification Carnegie Mellow University, 1997. Disponível na World Wide Web: <http://www.cs.cmu.edu/publications/thechnicalreports.html>. Acesso em 10/2000.
- NIELSON, J., GENTNER, D. The Anti-Mac Interface. Communications of ACM. Nova York, v.39, p.70-82, 1996.
- PRESSMAN, Roger Engenharia de Software. São Paulo: Mac Grow Hill, 1995.
- RESNIK, Paul A bíblia da pequena empresa. São Paulo: Mc Graw Hill, 1990.
- REZENDE, Denis A. Engenharia de Software e Sistemas de Informação. Rio de Janeiro: Brasport, 1999.
- ROSS, S., WESTERFIELD, R. Administração Financeira. São Paulo: Atlas, 1997.
- ROSS, S., WESTERFIELD, R., JORDAN, B. Princípios da Administração Financeira. São Paulo: Atlas, 1998.
- RUSS, Melissa A Software Development Process for Small Project. IEEE Software Eletronic Magazine, New York, n.11, May 2000.

- SANDMAN, Thomas. Decision Forest Enhancement Evaluation. MIS Department, School of Business, CSUS Sacramento, CA, 1998. Disponível na World Wide Web: <http://www.cs.csus.edu/publications/reports.html>. Acesso em 09/2000.
- SISTI, Frank; JOSEPH S. Software Risk Evaluation Method. Software Engineerign Institute, Carnegie Mellon University, 1994. Disponível na World Wide Web: <http://www.cs.cmu.edu/publications/technicalreports.html>. Aceso em 09/2000.
- VASUDEVAN, Satish. Initiation of Information Technology Innovations. Syracuse University, Syracuse, NY, 1996. Disponível na World Wide Web: <http://www.cs.syu.edu/publications/index.html>. Acesso em 10/2000.
- WHITTAKER, James. What is Software Testing? And Why it is so Hard ? IEEE Software, New York, v.11, n.9, February, 2000.
- ZDANOWICZ, José. Casos de Administração Financeira e Orçamento. Porto Alegre: Luzzato, 1986.

## BIBLIOGRAFIA

- DEMARCO, Tom. Análise Estruturada e Especificação de Sistemas. Rio de Janeiro: Campus, 1989.
- MAFFEO, Bruno. Engenharia de Software e Especificação de Sistemas. Rio de Janeiro: Campus, 1992.
- STAIR, Ralph. Princípios de Sistemas de Informação. Rio de Janeiro: LCT, 1996.
- REPORTS OF ISO. Ergonomic requeriments for office work with visual display terminals. Parts 1 to 17, Draft International Standart, ISO, 1993.
- VILELA, Dirley L A contabilidade gerencial e o processo de comunicação: Um Estudo Sobre a Utilização de Ferramentas de Contabilidade Gerencial pelas Empresas de Pequeno Porte e pelos Escritórios Prestadores de Serviços Contábeis. Cascavel, Programa de Pós-Graduação, Unioesc, 2000.
- WALTON, Richard. Tecnologia de Informação. São Paulo: Atlas, 1993.

## ANEXO 1 – PRINCIPAIS TELAS DO PROGRAMA

Serão apresentadas abaixo as principais telas do programa, sendo ainda comentadas suas principais características funcionais, de acordo com o descrito no capítulo 4.

Na figura 14, tem-se a tela onde é inserida a senha do operador, sendo necessário somente a digitação da senha para que o sistema defina o operador e os módulos os quais este tenha acesso.

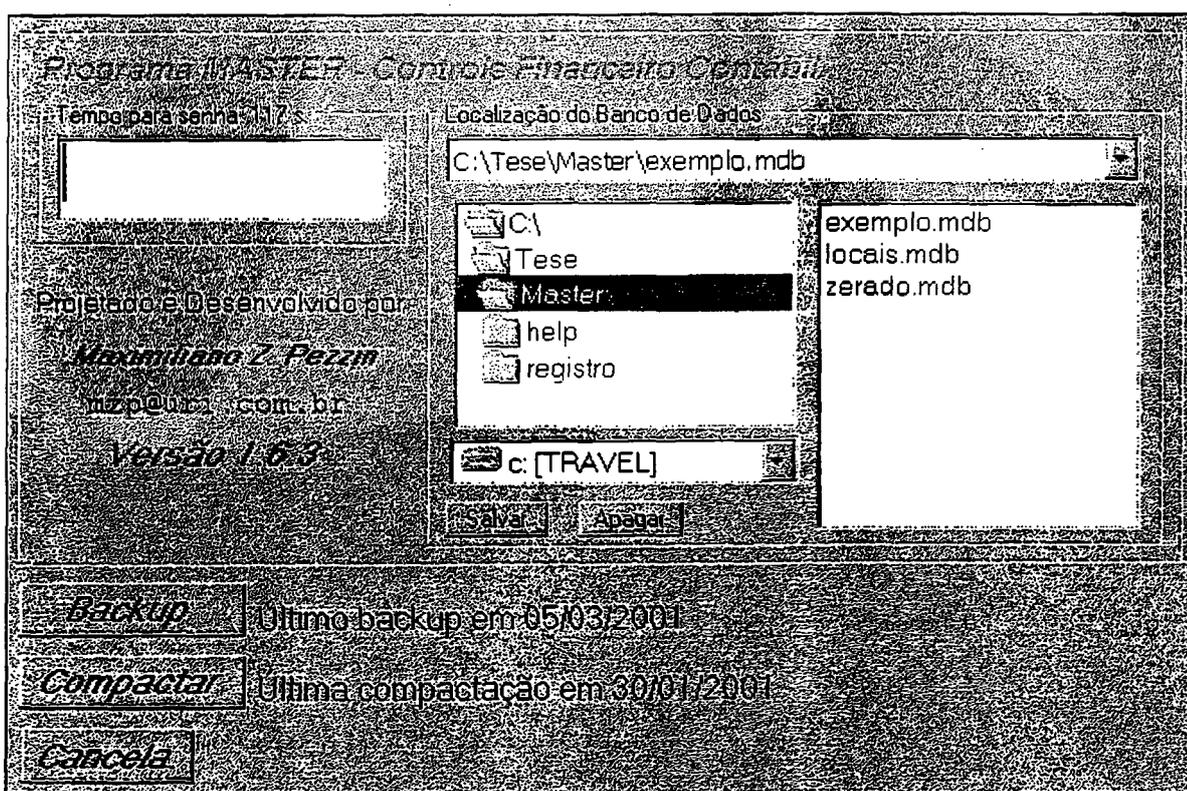


Figura 14 – Tela de senhas e entrada

Na figura 14 é possível ainda definir o banco de dados a ser utilizado, assim, o usuário poderá definir outro banco de dados, que não seja o padrão.

Outras duas funções da tela de senha é a possibilidade de realização de backup, onde o arquivo do banco de dados é compactado logicamente (eliminação de registros vazios) e fisicamente (compactação de baixo nível utilizando um software comercial de compactação de arquivos).

Na figura 15 tem-se o menu principal, sendo que os itens de menus serão habilitados ou não de acordo com a configuração do programa e de acordo com o usuário.



Figura 15 – Tela de Menu

Na figura 16 é apresentado o módulo de Cadastros de Clientes e Fornecedores com um nível de complexidade máxima, sendo mostrado na parte superior uma tabela de cadastros, na tabela inferior a esquerda as informações financeiras do cliente / fornecedor selecionado e a direita são mostradas as movimentações de produtos do cliente em questão.

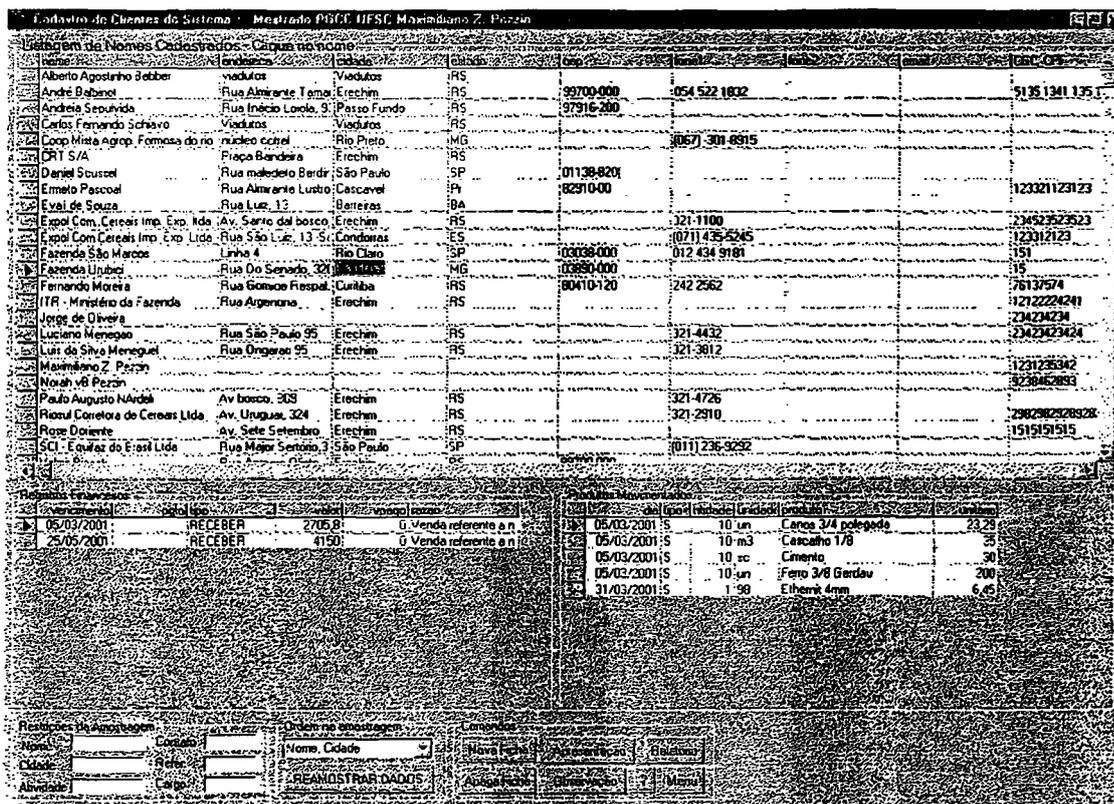


Figura 16 – Cadastros de Clientes/ Fornecedores em complexidade máxima

Na figura 17, é apresentada a tela de cadastro em seu nível de complexidade mínima, não sendo apresentadas informações financeiras ou de produtos do cliente ou fornecedor selecionado.

Cadastro de Clientes do Sistema - Mestrado PGCC UFSC Maximiliano Z. Pezzin

Listagem de Nomes Cadastrados - Clique no nome

nome	endereco	cidade	estado	cep	fone
Alberto Agostinho Beber	viadutos	Viadutos	RS		
André Balbinot	Rua Almirante Tamandaré	Erechim	RS	99700-000	054
Andreia Sepulveda	Rua Inácio Loiola, 9	Passo Fundo	RS	97916-200	
Carlos Fernando Schiavo	Viadutos	Viadutos	RS		
Coop.Mista Agrop. Formosa do rio	núcleo cotrel	Rio Preto	MG		067
CRT S/A	Praça Bandeira	Erechim	RS		
Daniel Scussel	Rua maledeto Berdir	São Paulo	SP	01138-820	
Ermeto Pascoal	Rua Almirante Luro	Cascavel	Pr	82910-00	
Evaí de Souza	Rua Luiz, 13	Barreiras	BA		
Expol Com. Cereais Imp. Exp. Ltda	Av. Santo dal bosco	Erechim	RS		321
Expol Com. Cereais imp. Exp. Ltda	Rua São Luiz, 13 -S	Condorras	ES		071
Fazenda São Marcos	Linha 4	Rio Claro	SP	03038-000	012
Fazenda Urubici	Rua Do Senado, 32	Cardeias	MG	03890-000	
Fernando Moreira	Rua Gomide Respal	Curitiba	RS	80410-120	242
ITR - Ministério da Fazenda	Rua Argentina	Erechim	RS		
Jorge de Oliveira					
Luciano Menegati	Rua São Paulo 95	Erechim	RS		321
Luis da Silva Meneguel	Rua Ongarati 95	Erechim	RS		321
Maximiliano Z. Pezzin					
Norah vB Pezzin					
Paulo Augusto Nardeli	Av bosco, 809	Erechim	RS		321
Riosul Corretora de Cereais Ltda	Av. Uruguai, 324	Erechim	RS		321
Rose Doriente	Av. Sete Setembro	Erechim	RS		
SCI - Equilaz do Brasil Ltda	Rua Major Sertório,3	São Paulo	SP		011

Restrições de Amostragem: Nome, Cidade, Atividade; Contato, Bate, Cargo; Ordem de amostragem: Nome, Cidade; Comandos: Nova Ficha, Apresentação, Relatório, Apaga Ficha, Observação, Menu; REAMOSTRAR DADOS

Figura 17 – Cadastros de Clientes/ Fornecedores em complexidade mínima

Algumas características são comuns nas figuras 16 e 17, independentemente do nível de complexidade, tais como as restrições de amostragem e os comando de criação e remoção de fichas.

Na figura 18 e 19 é apresentado o módulo financeiro de seus níveis de mínima e máxima complexidade, sendo que no nível de máxima complexidade são apresentados os lançamentos contábeis.

É evidenciada ainda a questão da racionalização do módulo, sendo utilizado praticamente todo o espaço da tela, permitindo que um máximo de informações sejam visualizados e manipulados.

Na porção direita da tela (figura 19) são apresentadas as movimentações financeiras, sendo que na figura 19 tem-se, a esquerda, as movimentações contábeis relacionadas à movimentação financeira selecionada a direita.

**Lançamentos financeiros e contábeis em 27/06/2001 - Mestrado PGCC UFSC Maximiliano Z. Pezzin**

Cadastro dos códigos Contábeis - Partidas Dobradas

Acreditar em: 0,00

Debitar em: 0,00

Total dos créditos R\$ 80,00 de R\$ 80,00 - Pagar

09/02/2001: Contas a Receber 80: Paga

Total dos débitos R\$ 80,00 de R\$ 80,00 - Receber

19/02/2001: Contas a Pagar 80: Rece

**Listagem de Códigos Contábeis**

Classe	Nome
1.1.1	Caixas
1.1.2	Bancos
1.1.3	Contas a Receber
1.1.4	Aplicações Financeiras
1.1.5	Empréstimos a Receber
1.1.6	Despesas Exercício Segurte
1.1.7	Estoques
1.2.1	Empréstimos a Receber
1.3.1	Investimentos e Ações
1.3.2	Imobilizado
1.3.3	OEM e Cursos
1.3.4	Diferido
2.1.1	Fornecedores
2.1.2	Impostos

**Listagem das Contas**

TIPO	Competência	Vencimento	VALOR	NOME
RECEBIDA	09/02/2001	05/07/2000	1100	Flávio Baldissera
RECEBIDA	14/02/2001	14/07/2000	500	Coop.Mista Agrop. Formose
RECEBIDA	14/02/2001	14/07/2000	1200	Coop.Mista Agrop. Formose
RECEBIDA	14/02/2001	21/07/2000	200	Antônio Pereira
PAGA	14/02/2001	21/07/2000	90	Maximiliano Z. Pezzin
RECEBIDA	14/02/2001	14/08/2000	90	Coop.Mista Agrop. Formose
RECEBIDA	14/02/2001	21/08/2000	200	Antônio Pereira
PAGA	14/02/2001	21/08/2000	90	Maximiliano Z. Pezzin
RECEBIDA	14/02/2001	14/09/2000	90	Coop.Mista Agrop. Formose
RECEBIDA	14/02/2001	21/09/2000	200	Antônio Pereira
PAGA	14/02/2001	21/09/2000	90	Maximiliano Z. Pezzin
RECEBIDA	14/02/2001	14/10/2000	90	Coop.Mista Agrop. Formose
RECEBIDA	14/02/2001	21/10/2000	200	Antônio Pereira
PAGA	14/02/2001	21/10/2000	90	Maximiliano Z. Pezzin
RECEBIDA	14/02/2001	21/11/2000	200	Antônio Pereira
PAGA	14/02/2001	21/11/2000	90	Maximiliano Z. Pezzin
RECEBIDA	14/02/2001	21/12/2000	200	Antônio Pereira
PAGA	14/02/2001	21/12/2000	90	Maximiliano Z. Pezzin
PAGA	19/02/2001	03/01/2001	183	Ermelo Pascoal
PAGA	19/02/2001	03/01/2001	183	Ermelo Pascoal
RECEBIDA	19/02/2001	10/01/2001	80	Antônio Pereira
RECEBIDA	19/02/2001	10/01/2001	80	Antônio Pereira
PAGA	14/02/2001	10/01/2001	420	Caminhão Rodopen
PAGA	14/02/2001	10/01/2001	420	Caminhão Rodopen
RECEBIDA	14/02/2001	10/01/2001	800	Domenico Brambali
RECEBIDA	14/02/2001	10/01/2001	800	Domenico Brambali
RECEBER	19/03/2001	12/01/2001	10	Fazenda São Marcos
RECEBER	14/02/2001	13/01/2001	460	Nielsen Kombrlach
RECEBIDA	14/02/2001	13/01/2001	460	Nielsen Kombrlach
PAGAR	19/03/2001	15/01/2001	25	Flávio Baldissera
PAGA	30/01/2001	17/01/2001	284,62	IMPORTADORA BIO NATI
PAGA	30/01/2001	17/01/2001	478,92	IMPORTADORA BIO NATI
RECEBER	14/02/2001	21/01/2001	200	Antônio Pereira
PAGA	14/02/2001	02/02/2001	100	Anderson Barbosa

Figura 18 – Módulo financeiro em complexidade mínima

**Lançamentos financeiros e contábeis em 27/06/2001 - Mestrado PGCC UFSC Maximiliano Z. Pezzin**

Cadastro de Novas Contas - REC

Nome: Flávio Baldissera

Razão: [ ] Listagem

Unidade: Fazenda

Comandos: [Apaga] [Amostragem] [Novo] [Relatório] [Copia Contas] [Menu]

Qual Contas: 68 regs - R\$ 18.080,55 - Pagar; 26 regs - R\$ 7.763,54 - Pagas; 95 regs - R\$ 42.917,98 - Receber; 26 regs - R\$ 8.757,40 - Recebidas

Resumo: Competência: 27/06/2000 - 27/06/2002; Vencimento: [ ]; Pagamento: [ ]; Anotação: [Seleção]; Unidade: [ ]; Direção: [Mostrar]; Grupo: [ ]; Vencimento nome: [ ]; Conta: [ ]

**Listagem das Contas**

TIPO	Competência	Vencimento	VALOR	NOME
RECEBIDA	09/02/2001	05/07/2000	1100	Flávio Baldissera
RECEBIDA	14/02/2001	14/07/2000	500	Coop.Mista Agrop. Formose
RECEBIDA	14/02/2001	14/07/2000	1200	Coop.Mista Agrop. Formose
RECEBIDA	14/02/2001	21/07/2000	200	Antônio Pereira
PAGA	14/02/2001	21/07/2000	90	Maximiliano Z. Pezzin
RECEBIDA	14/02/2001	14/08/2000	90	Coop.Mista Agrop. Formose
RECEBIDA	14/02/2001	21/08/2000	200	Antônio Pereira
PAGA	14/02/2001	21/08/2000	90	Maximiliano Z. Pezzin
RECEBIDA	14/02/2001	14/09/2000	90	Coop.Mista Agrop. Formose
RECEBIDA	14/02/2001	21/09/2000	200	Antônio Pereira
PAGA	14/02/2001	21/09/2000	90	Maximiliano Z. Pezzin
RECEBIDA	14/02/2001	14/10/2000	90	Coop.Mista Agrop. Formose
RECEBIDA	14/02/2001	21/10/2000	200	Antônio Pereira
PAGA	14/02/2001	21/10/2000	90	Maximiliano Z. Pezzin
RECEBIDA	14/02/2001	21/11/2000	200	Antônio Pereira
PAGA	14/02/2001	21/11/2000	90	Maximiliano Z. Pezzin
RECEBIDA	14/02/2001	21/12/2000	200	Antônio Pereira
PAGA	14/02/2001	21/12/2000	90	Maximiliano Z. Pezzin
PAGA	19/02/2001	03/01/2001	183	Ermelo Pascoal
PAGA	19/02/2001	03/01/2001	183	Ermelo Pascoal
RECEBIDA	19/02/2001	10/01/2001	80	Antônio Pereira
RECEBIDA	19/02/2001	10/01/2001	80	Antônio Pereira
PAGA	14/02/2001	10/01/2001	420	Caminhão Rodopen
PAGA	14/02/2001	10/01/2001	420	Caminhão Rodopen
RECEBIDA	14/02/2001	10/01/2001	800	Domenico Brambali
RECEBIDA	14/02/2001	10/01/2001	800	Domenico Brambali
RECEBER	19/03/2001	12/01/2001	10	Fazenda São Marcos
RECEBER	14/02/2001	13/01/2001	460	Nielsen Kombrlach
RECEBER	14/02/2001	13/01/2001	460	Nielsen Kombrlach
PAGAR	19/03/2001	15/01/2001	25	Flávio Baldissera
PAGA	30/01/2001	17/01/2001	284,62	IMPORTADORA BIO NATI
PAGA	30/01/2001	17/01/2001	478,92	IMPORTADORA BIO NATI
RECEBER	14/02/2001	21/01/2001	200	Antônio Pereira
PAGA	14/02/2001	02/02/2001	100	Anderson Barbosa

Figura 19 – Módulo financeiro em complexidade máxima

Na figura 20 e 21 tem-se o módulo de controle de caixa, onde a diferenciação entre os níveis de complexidade mínima, apresentada na figura 20 e a complexidade máxima, apresentada na figura 21, refere-se à especialização da informação.

No nível de menor complexidade tem-se um controle onde são desconsiderados os fechamentos de saldos diários e a realização do livro caixa.

O funcionamento do caixa nos dois modos é, para o usuário, praticamente igual.

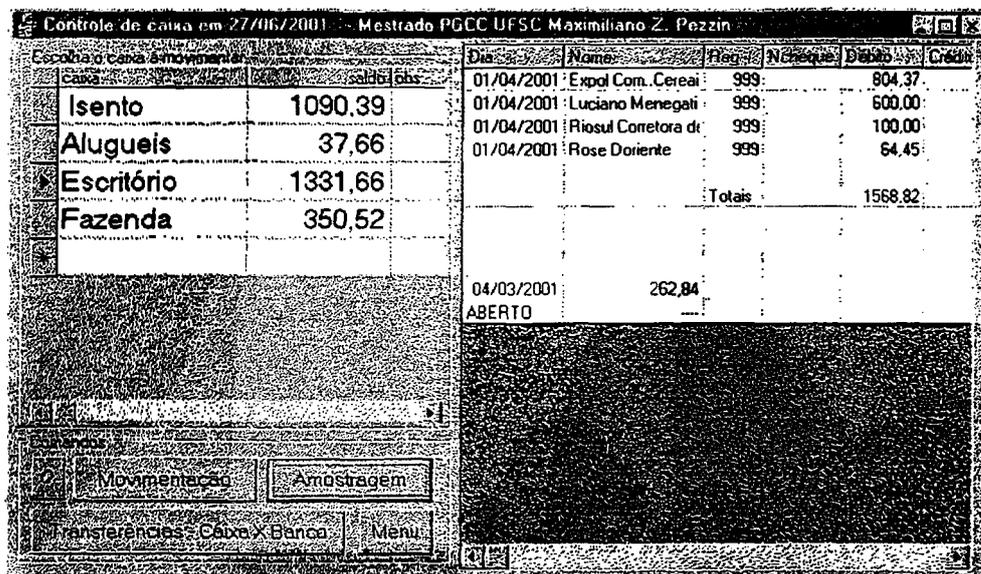


Figura 20 – Tela de controle de caixa em complexidade mínima

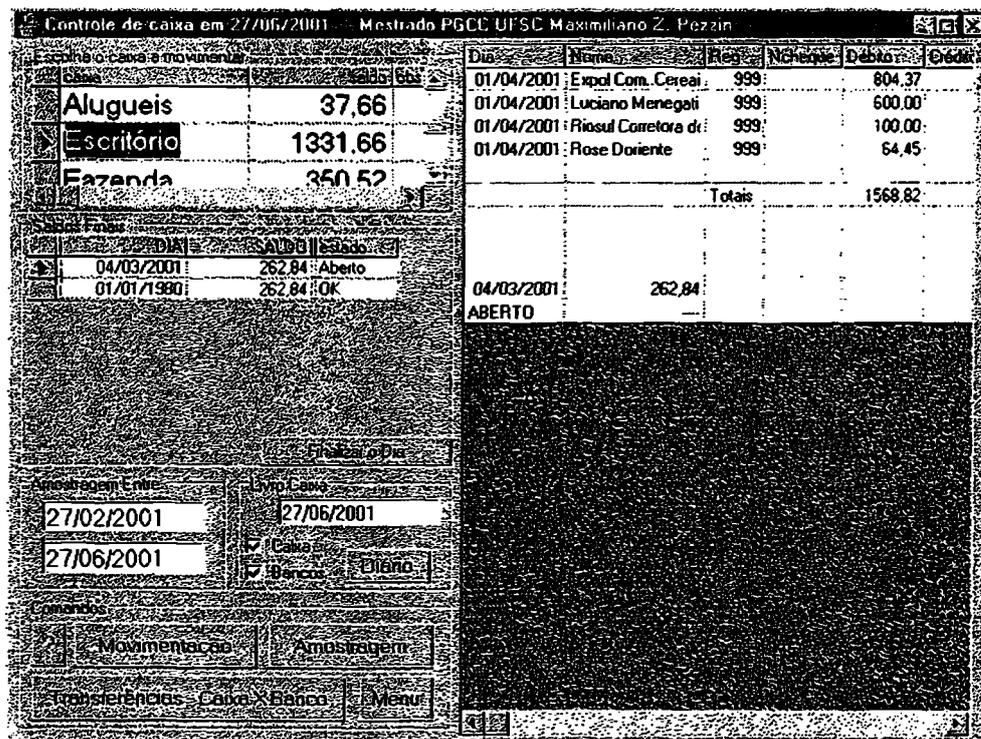


Figura 21 – Tela de controle de caixa em complexidade máxima

Na figura 22 é apresentado um exemplo de relatório totalizado, sendo mostrado uma totalização em três níveis, sendo apresentados os valores totais e os percentuais.

Relatórios sem restrições Total previsto R\$ 13021,55 Total pago R\$ 0,00 Tempo 8 s									
Enviar para Excel		Enviar para Word		Arquivos de Texto		Impressão		Menu	
Relatório de contas	A PAGAR	Entre	/04/2000 e	28/08/2001	Ordem	Grupo	C		
*****									
Caminhões	4452,00	0,00	34,19 %						
	Extras	4452,00	100,00 %						
		Caminhão R	4452,00	0,00	100,00 %				
*****									
Despesas Judiciais	2323,00	0,00	17,84 %						
	Acordos	1123,00	48,34 %						
		Ermeto Pasc	1098,00	0,00	97,77 %				
		Flávio Bakis	25,00	0,00	2,23 %				
	Execuções	1200,00	51,66 %						
		Flávio Bakis	1200,00	0,00	100,00 %				
*****									
Despesas Automóveis	1007,55	0,00	7,74 %						
	Extras	661,15	65,62 %						
		Flávio Bakis	250,00	0,00	37,81 %				
		IMPORTAD	411,15	0,00	62,19 %				
	Multas	346,40	34,38 %						
		Nelson Pulc	346,40	0,00	100,00 %				
*****									
Despesas Operacionais	119,00	0,00	0,91 %						
	Informatica	119,00	100,00 %						
		Bitshop Infor	119,00	0,00	100,00 %				
*****									
Salários	5120,00	0,00	39,32 %						
	Empreitadas	800,00	15,63 %						
		Raquel Giore	800,00	0,00	100,00 %				

Opções	Comandos	Restrições	Relatório Anual	Níveis
<input type="checkbox"/> Pagos	Relatório	Vencimento	Mes: 3	1
<input type="checkbox"/> Recebidas	Menu	Sintático	Ano: 2001	2: Relatórios restrix
<input checked="" type="checkbox"/> A Pagar	Totalizado	entre: 01/04/2000 e 28/08/2001	Relatório	3: Relatório de salá
<input type="checkbox"/> A Receber	Fluxo	Para: Ordenação	Plane Contas	21: Relatório em 04/1
	Tabela Cruzada	Avançadas		

Figura 22 – Tela de Relatório Totalizado

Na figura 23 tem-se um exemplo de relatório contábil, sendo apresentado o Balanço Patrimonial, sendo os dados já processados e apresentado o BP finalizado.

Na figura 23, os dados já foram processados, mas há a possibilidade do usuário alterar o demonstrativo contábil ou mesmo criar outros relatórios de acordo com suas necessidades. Os campos não processados, no final do relatório serão processados assim que o relatório for exportado para o Excel, que realizará os cálculos finais.

A codificação para a criação do demonstrativo contábil é apresentada e explicada em detalhes no arquivo de ajuda e no relatório de procedimentos, constantes no CD que compõem este trabalho.