

**UNIVERSIDADE REGIONAL DE BLUMENAU**  
**CENTRO DE CIÊNCIAS EXATAS E NATURAIS**  
**CURSO DE CIÊNCIAS DA COMPUTAÇÃO – BACHARELADO**

***SOFTWARE PARA AVALIAÇÃO DA QUALIDADE DE***  
**SISTEMAS DE AUTOMAÇÃO COMERCIAL BASEADO NAS**  
**NORMAS ISO/IEC 9126 E NBR ISO/IEC 12119.**

**MARCO ANTONIO DOS SANTOS**

**BLUMENAU**  
**2004**

**2004/1-28**

**MARCO ANTONIO DOS SANTOS**

***SOFTWARE PARA AVALIAÇÃO DA QUALIDADE DE  
SISTEMAS DE AUTOMAÇÃO COMERCIAL BASEADO NAS  
NORMAS ISO/IEC 9126 E NBR ISO/IEC 12119.***

Trabalho de Conclusão de Curso submetido à  
Universidade Regional de Blumenau para a  
obtenção dos créditos na disciplina Trabalho  
de Conclusão de Curso II do curso de Ciências  
da Computação — Bacharelado.

Dr. Carlos Eduardo Negrão Bizzotto

**BLUMENAU  
2004**

**2004/1-28**

**SOFTWARE PARA AVALIAÇÃO DA QUALIDADE DE  
SISTEMAS DE AUTOMAÇÃO COMERCIAL BASEADO NAS  
NORMAS ISO/IEC 9126 E NBR ISO/IEC 12119.**

Por

**MARCO ANTONIO DOS SANTOS**

Trabalho aprovado para obtenção dos créditos  
na disciplina de Trabalho de Conclusão de  
Curso II, pela banca examinadora formada  
por:

Presidente: \_\_\_\_\_  
Prof. Carlos Eduardo Negrão Bizzotto, Dr. – Orientador, FURB

Membro: \_\_\_\_\_  
Prof. Jomi Fred Hubner, Dr., FURB

Membro: \_\_\_\_\_  
Prof. Wilson Pedro Cali, MsC, FURB

Blumenau, 02 de junho de 2004

Dedico este trabalho a minha esposa Daisy que sempre me incentivou com suas palavras de otimismo, principalmente quando eu achava que nada mais daria certo. E ao meu filho Lucas que sempre tem um sorriso para nos animar e, hoje aos cinco meses, é a maior prova de que nunca é tarde para realizarmos nossos sonhos.

“Se A é o sucesso, então é igual a X mais Y mais Z. O trabalho é X; Y é o lazer; e Z é manter a boca fechada.”

Albert Einstein

## AGRADECIMENTOS

A Deus por tudo que sempre me deu.

À minha esposa pelo incentivo, apoio e, principalmente, sua compreensão durante todo o curso.

À minha mãe que com sua infinita sabedoria, soube se desdobrar em mil para ser mãe, pai, amiga, médica, professora, etc. tudo que sempre precisei para ser a pessoa que sou hoje.

À minha avó Luiza, *in memoriam*, que mesmo sendo analfabeta soube me ensinar muito mais que a maioria das pessoas que encontrei pela vida.

Aos meus amigos, que me ajudaram durante o curso, dentre estes a Larissa , que muito me ensinou quando fizemos a disciplina programação III (Delphi) juntos.

Ao amigo Fabiano Oss, por todo apoio que me deu na época que estudamos juntos e durante a elaboração deste trabalho com minha eterna “briga” com o *Delphi*. Sem sua ajuda nada disso teria sido possível.

Ao amigo Gerson L. de Souza pelo incentivo na elaboração desta ferramenta utilizando o intraweb.

Ao meu orientador, Dr. Carlos E. N. Bizzotto, pela atenção e auxílio dispensados na elaboração deste trabalho.

## RESUMO

Este trabalho de conclusão de curso consiste na implementação de um *software* para avaliar sistemas de automação comercial, baseado nas normas ISO/IEC 9126 e NBR ISO/IEC 12119. O *software* foi desenvolvido no ambiente Delphi 7, utilizando o *Intraweb*, que é uma biblioteca de componentes de desenvolvimento de HTMLs. O *software* funciona via *web* e é acessado via *browser*, auxiliando o usuário no processo de avaliação da qualidade de *software*.

Palavras chaves: Normas ISO/IEC 9126; NBR ISO/IEC 12119; Qualidade de *Software*; Automação comercial.

## **ABSTRACT**

This course conclusion work consists on a software implementation to analyse commercial automated systems based on ISO/IEC 9126 and NBR ISO/IEC 12119 standards. The software was development on Delphi 7 environment, using Intraweb, that is a components' library of HTML development. The software works by web and it can be accessed by browser, helping the user on the analysis process of the software quality.

Key words: ISO/IEC 9126 standards; NBR ISO/IEC 12119; software quality; commercial automation.



## LISTA DE ILUSTRAÇÕES

FIGURA 1 – Conjunto de características a serem verificadas num <i>software</i> – norma ISO/IEC 9126 / NBR 13596.....	23
FIGURA 2 – Modelo do processo de avaliação.....	27
FIGURA 3 – Estrutura da norma NBR ISO/IEC 12119.....	29
FIGURA 4 – Diagrama da funcionalidade.....	43
FIGURA 5 – Diagrama de confiabilidade.....	44
FIGURA 6 – Diagrama de usabilidade.....	45
FIGURA 7 – Diagrama de eficiência.....	46
FIGURA 8 – Diagrama de Manutenibilidade.....	46
FIGURA 9 – Diagrama de portabilidade.....	47
QUADRO 1 – Lista de trabalhos correlatos.....	49
FIGURA 10 – Diagrama de contexto.....	49
FIGURA 11 – Diagrama de fluxo de dados.....	51
FIGURA 12 – Diagrama entidade relacionamento – modelo lógico.....	52
FIGURA 13 – Diagrama entidade relacionamento – modelo físico.....	53
FIGURA 14 – Tela inicial do <i>software</i> .....	55
FIGURA 15 – Tela de cadastro de usuário.....	56
FIGURA 16 – Cadastro de produtora.....	56
FIGURA 17 – Cadastro do <i>software</i> .....	57
FIGURA 18 – Cadastro das características.....	58
FIGURA 19 – Cadastro das subcaracterísticas.....	59
FIGURA 20 – Cadastro das questões.....	59
FIGURA 21 – Tela de cadastro dos pesos das características.....	60
FIGURA 22 – Tela de cadastro dos pesos das subcaracterísticas.....	60
FIGURA 23 – Tela inicial da avaliação.....	61
FIGURA 24 – Tela das perguntas do <i>check list</i> ......	62
FIGURA 25 – Tela do relatório de pontuação do <i>software</i> .....	62
FIGURA 26 – Gráfico comparativo de desempenho.....	63

## **LISTA DE SIGLAS**

ABNT – Associação Brasileira de Normas Técnicas

CASE – *Computer Aided Systems Enginneering*

DER – Diagrama Entidade Relacionamento

IEC – *International Eletrotechnical Committe*

ISO – *International Organization for Standardization*

NBR – Norma Brasileira de Regulamentação

URL – *Uniform Resource Locator*

## SUMÁRIO

<b>1 INTRODUÇÃO</b> .....	<b>12</b>
1.1 OBJETIVOS DO TRABALHO .....	14
1.2 ESTRUTURA DO TRABALHO .....	14
<b>2 QUALIDADE</b> .....	<b>15</b>
2.1 CONCEITO DA QUALIDADE.....	15
2.2 QUALIDADE EM <i>SOFTWARE</i> .....	16
2.2.1 Visão do Usuário.....	18
2.2.2 Visão do Desenvolvedor .....	18
2.2.3 Visão do Gerente de Desenvolvimento.....	19
2.3 NORMAS PARA ÁREA DE <i>SOFTWARE</i> .....	19
<b>3 NORMAS ISO/IEC 9126 E NBR ISO/IEC 12119</b> .....	<b>22</b>
3.1 QUALIDADE DE PRODUTO DE <i>SOFTWARE</i> .....	22
3.2 NORMA ISO/IEC 9126 (NBR 13596) .....	23
3.3 AVALIAÇÃO DA QUALIDADE A PARTIR DA ISO/IEC 9126 .....	26
3.3.1 Usos da ISO/IEC 9126 (NBR 13956) .....	26
3.3.2 Métricas.....	26
3.3.3 O Processo de Avaliação.....	27
3.4 NORMA NBR ISO/IEC 12119 .....	28
3.4.1 Descrição do Produto .....	29
3.4.1.1 Requisitos Gerais sobre o Conteúdo da Descrição .....	30
3.4.1.2 Identificações e Indicações .....	30
3.4.1.3 Declarações sobre Funcionalidade .....	31
3.4.1.4 Declarações sobre Confiabilidade .....	32
3.4.1.5 Declarações sobre Usabilidade .....	32
3.4.1.6 Declarações sobre Eficiência .....	33
3.4.1.7 Declarações sobre Manutenibilidade .....	33
3.4.1.8 Declarações sobre Portabilidade.....	33
3.4.2 Documentação de Usuário .....	33
3.4.2.1 Completitude.....	33
3.4.2.2 Correção.....	34
3.4.2.3 Consistência .....	34
3.4.2.4 Inteligibilidade .....	34

3.4.2.5 Apresentação e Organização.....	34
3.4.3 Programas e Dados.....	34
3.4.3.1 Funcionalidade.....	34
3.4.3.2 Confiabilidade.....	35
3.4.3.3 Usabilidade .....	36
3.4.3.4 Eficiência .....	37
3.4.3.5 Manutenibilidade .....	37
3.4.3.6 Portabilidade .....	38
<b>4 AVALIAÇÃO DA QUALIDADE DE SISTEMAS DE AUTOMAÇÃO COMERCIAL</b> .....	<b>39</b>
4.1 AUTOMAÇÃO COMERCIAL.....	39
4.2 EQUIPAMENTOS PARA AUTOMAÇÃO COMERCIAL.....	40
4.3 PROCESSO DE AVALIAÇÃO.....	42
4.4 CARACTERÍSTICA DA AVALIAÇÃO .....	42
4.5 PROPOSTA DE DETALHAMENTO DA NORMA ISO 9126 PARA AUTOMAÇÃO COMERCIAL.....	42
<b>5 DESENVOLVIMENTO DO TRABALHO.....</b>	<b>48</b>
5.1 ANÁLISE DOS REQUISITOS.....	48
5.2 ESPECIFICAÇÃO .....	49
5.2.1 Diagrama de Contexto.....	49
5.2.2 Diagrama de Fluxo de Dados.....	50
5.2.3 Diagrama Entidade Relacionamento.....	52
5.3 IMPLEMENTAÇÃO .....	54
5.3.1 <i>Intraweb</i> .....	54
5.3.2 Descrição das Telas do <i>Software</i> .....	55
<b>6 CONCLUSÕES.....</b>	<b>64</b>
6.1 EXTENSÕES .....	64
<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>66</b>
APÊNDICE A – avaliação de três <i>softwares</i> , com base na ferramenta desenvolvida. ....	68
APÊNDICE B – Questionários referentes uso de <i>software</i> de automação comercial.....	101
APÊNDICE C – perguntas desenvolvidas para serem usadas no <i>software</i> proposto para a avaliação de sistemas de automação comercial.....	109
APÊNDICE D – dicionário de dados .....	116
APÊNDICE E – Levantamento Bibliográfico realizado para elaboração deste TCC.....	119

# 1 INTRODUÇÃO

Inthurn (2001) comenta que aumentar a produtividade significa produzir cada vez mais e/ou melhor com cada vez menos. Com a globalização da economia, as empresas precisaram aumentar a produtividade com qualidade. Isto as levou para a automação de seus processos produtivos, visando o melhor controle de cada parte do mesmo.

A informática tornou-se fundamental para este controle, uma vez que a velocidade com que se pode obter uma determinada informação certamente torna o processo produtivo muito mais eficiente. As empresas acabaram por levar a informática para diversos setores, como vendas, contas a receber, etc. Isto fez com que surgisse no mercado diversos sistemas para atender estas demandas.

Com esse aumento crescente da disponibilidade de produtos, em muitos casos o usuário final acaba não conseguindo chegar a uma conclusão de qual seria o sistema ideal para sua real necessidade. Porém não basta apenas o *software* servir para suas necessidades, além disso, ele precisa ter qualidade. Assim torna-se necessário a definição de padrões de qualidade em relação aos quais eles serão avaliados.

Diante destes problemas, as empresas que desenvolvem *software* devem procurar associar sua imagem a produtos com qualidade. Para isto, torna-se necessário a definição de padrões de qualidade, com os quais os produtos de *software* possam ser validados.

Para isso a *International Organization for Standardization* (ISO) e a *International Electrotechnical Commission* (IEC), publicaram em 1991 a norma que representa a padronização mundial para qualidade de produtos de *software* denominada ISO/IEC 9126, que propõem as características e subcaracterísticas da qualidade de produtos de *software*. Esta norma é uma das mais antigas da área de qualidade de *software* e possui sua tradução para o Brasil, publicada em agosto de 1996 como NBR 13596 (INTHURN, 2001).

Em conjunto com a norma ISO/IEC 9126, surge em 1994 a norma ISO/IEC 12119, que traduzida e validada pela ABNT em 1998, passou a chamar NBR ISO/IEC 12119, que tem como objetivo verificar e validar pacotes de *software*.

Apesar da importância destas normas para a melhoria da qualidade de *software*, as mesmas possuem um alto grau de generalidade, ou seja, não foram desenvolvidas para a avaliação de *softwares* de áreas específicas. Adicionalmente, as características e subcaracterísticas propostas pelas normas indicadas anteriormente possuem um alto grau de

subjetividade. Isto faz com que, na avaliação da qualidade de um *software*, os resultados obtidos por diferentes avaliadores (com experiências distintas) possam ser significativamente diferentes.

Nesse sentido, alguns centros de pesquisa (como o Centro Renato Archer, em Campinas) vêm desenvolvendo estudos para reduzir o grau de subjetividade das normas mencionadas. A grande dificuldade enfrentada pelos pesquisadores é detalhar cada uma das características e subcaracterísticas da qualidade em itens que possam ser expressos na forma de métricas, permitindo uma avaliação quantitativa do *software* em questão.

Outro aspecto que dificulta a utilização das normas mencionadas é o fato delas serem genéricas, ou seja, não direcionadas ao *software* de uma área específica. Assim, para que a avaliação quantitativa da qualidade seja realizada é essencial que se faça um estudo da área de negócio em questão (automação comercial, gestão da produção, etc.).

É dentro desse contexto que se enquadra o presente trabalho, cujo objetivo é o desenvolvimento de um *software*, via *web*, que auxilie o processo de avaliação da qualidade de *softwares* de automação comercial.

Para isso foi feito um estudo da área de automação comercial, de forma a identificar as necessidades e requisitos que um *software* deve atender. O resultado deste estudo (a ser realizado junto a profissionais da área) foi organizado na forma de um *check list*, o qual servirá de base para o desenvolvimento do *software* proposto.

Além do estudo da área de automação comercial, foi realizado um estudo sobre os itens da norma, tomando como base trabalhos anteriores, como o de Ferreira (2003), onde foi desenvolvida uma ferramenta para auxiliar na avaliação da qualidade de produtos de *software*, baseado na norma NBR ISO/IEC 9126 (1996) e NBR ISO/IEC 12119. O objetivo é obter um conjunto de métricas que permitam a avaliação quantitativa da qualidade de um dado *software* de automação comercial.

Para que o resultado final desta avaliação seja quantitativo, o *software* desenvolvido utiliza cálculos estatísticos. Nestes cálculos é prevista a possibilidade do usuário (avaliador) definir os pesos das diferentes características e a forma de quantificação.

## 1.1 OBJETIVOS DO TRABALHO

O objetivo deste trabalho de conclusão de curso foi especificar e implementar um *software*, via *web*, para auxiliar na avaliação da qualidade de produtos de *software* para automação comercial, utilizando como base as características e subcaracterísticas da qualidade de *software* propostas pelas normas ISO/IEC 9126 e NBR ISO/IEC 12119.

Os objetivos específicos do trabalho são:

- a) permitir que o usuário possa incluir novos itens no *check list* proposto pelo *software* desenvolvido;
- b) possibilitar o estabelecimento de pesos diferenciados para os diferentes itens do *check list*, onde o próprio usuário poderá definir o peso conforme sua necessidade.

## 1.2 ESTRUTURA DO TRABALHO

Este trabalho é composto de 6 capítulos abrangendo fundamentação teórica, implementação do *software* para avaliar sistemas de automação comercial e conclusão.

O capítulo 1 apresenta uma introdução ao trabalho, alguns conceitos, os objetivos e a organização do texto.

O capítulo 2 apresenta o conceito de qualidade de um modo geral, além da qualidade específica em *software*, segundo alguns autores.

O capítulo 3 aborda as normas ISO/IEC 9126 e NBR ISO/IEC 12119, e qualidade de produtos de *software* baseada nestas normas.

O capítulo 4 aborda a qualidade de sistemas de automação comercial, apresenta alguns equipamentos que auxiliam nesta automação e algumas características.

No capítulo 5 encontra-se o desenvolvimento do trabalho abordando a análise dos requisitos, a especificação e a implementação.

Para finalizar, no capítulo 6 encontram-se conclusões e sugestões para trabalhos futuros.

## 2 QUALIDADE

Apesar de não haver uma única definição para o quesito qualidade, pode-se perceber que quem define a qualidade em termos de necessidades e expectativas implícitas e explícitas é o cliente. As necessidades e expectativas formam os requisitos a partir dos quais os produtos e serviços são projetados, o processo de fabricação, marketing, entrega, instalação e assistência técnica também são projetados e realizados (FERNANDES, 1995). Neste sentido, um conceito de qualidade que traduz estes aspectos é o proposto por Falconi (1992), que define que qualidade de um produto ou serviço é aquele que atende perfeitamente, de forma confiável, de forma acessível, de forma segura e no tempo certo, as necessidades do cliente

### 2.1 CONCEITO DA QUALIDADE

O conceito da qualidade é bastante antigo, pode-se dizer que surgiu mesmo antes dos homens começarem a se organizar para comercializar produtos. A ponderação da qualidade de um produto é subjetiva e varia com o local e a época, além disso, o mesmo produto pode ser considerado com maior ou menor qualidade quando avaliado por pessoas diferentes (CÔRTEZ E CHIOSSI, 2001).

Segundo Weber & Rocha (1999), qualidade e produtividade são conceitos amplos. Representam uma filosofia de gestão, que visa conduzir as organizações a uma postura de melhoria de seus processos, por meios de seus dirigentes e empregados.

Neste processo de entendimento do conceito de qualidade, segundo Inthurn (2001), é importante perceber que sempre estarão envolvidos dois personagens principais: o produtor e o consumidor. Rocha (1990), advertiu que qualidade é um atributo associado a alguma coisa e, assim sendo, qualidade não pode ser definida universalmente, mas deve ser definida para o item em questão.

No Brasil, a necessidade de um maior controle perante o aspecto qualidade fez com que, em 1990, o Governo criasse o Programa Brasileiro da Qualidade e Produtividade (PBQP) com o objetivo de estabelecer um conjunto ordenado de ações indutoras da modernização industrial e tecnológica, contribuindo para a retomada do desenvolvimento econômico e social. O PBQP resultou do ordenamento e aglutinação de projetos de abrangência geral e setorial, sob orientação estratégica única, executados de forma descentralizada nos diferentes níveis pelos vários agentes econômicos, com uma vigorosa atuação empresarial voltada para a qualidade e produtividade (PBQP, 1990 **apud** WEBER & ROCHA, 1999).



A partir desse momento, as empresas passaram a valorizar não só o produto final e sim todas as suas fases de desenvolvimento, além de se preocupar com sua entrega e a satisfação do cliente em adquirir este produto. As empresas também passaram a se preocupar com seu capital intelectual, investindo no aperfeiçoamento de seus colaboradores independentemente do cargo que ocupa, e sim de sua importância nas fases de desenvolvimento do seu produto.

A melhoria do produto faz com que este tenha uma maior aceitação, uma vez que todos buscam qualidade. Isto é um círculo vicioso, pois quem produz com qualidade tende a vender mais, o aumento das vendas é uma consequência para o aumento da receita, com aumento da receita pode-se investir na melhoria dos produtos e funcionários, funcionários qualificados tendem a desenvolver produtos bem elaborados e produtos bem elaborados tem uma melhor aceitação.

## 2.2 QUALIDADE EM *SOFTWARE*

Segundo Inthurn (2001), o desenvolvimento de *software* com qualidade é um assunto amplo, complexo e ainda muito discutido. São vários os fatores que precisam ser considerados para obtermos um resultado satisfatório. No entanto, é sempre bom lembrar que o principal indicador de qualidade no desenvolvimento de qualquer produto, incluindo o *software*, é a satisfação do cliente.

Podemos então, segundo Rocha (1990), definir qualidade de *software* como um conjunto de propriedades a serem satisfeitas em determinado grau, de modo que o *software* satisfaça as necessidades de seus usuários.

Neste contexto, pessoas com diferentes interesses sobre um produto têm diferentes visões sobre o conceito de qualidade. Clientes usualmente consideram que o *software* tem qualidade se possui características que atendam suas necessidades. Desenvolvedores usualmente vêem a qualidade através das medidas de suas propriedades que são comparadas com indicadores de qualidade preestabelecidos (CÔRTEZ E CHIOSSI, 2001).

Um cliente estará satisfeito quando um *software* adquirido por ele atender suas necessidades sem que para isso seja necessário um conhecimento abrangente na área de *software*, afinal ele precisa de conhecimento abrangente na sua área de atuação, e não em *software*. Porém, nem sempre é fácil escolher um *software* que satisfaça todas as suas necessidades apenas lendo suas características.

Desta forma, estas novas exigências fizeram com que as empresas adotassem novas filosofias para “sobreviver” no mercado mais competitivo que surgiu. Tendo o Programa Brasileiro da Qualidade e Produtividade papel importante neste processo.

Segundo Weber & Rocha (1999), o Programa Brasileiro da Qualidade e Produtividade teve uma primeira fase que foi de 1990 até 1995, e foi nesta fase, mais precisamente em 1º de junho de 1993, no âmbito do PBQP, que foi criado o Subcomitê Setorial da Qualidade e Produtividade em *Software* (PBQP/SSQP-SW). Sendo o termo de referência do PBQP/SSQP-SW composto de:

- a) diagnóstico do setor de *software* em relação à qualidade e produtividade;
- b) análise das tendências nacionais e internacionais da qualidade e produtividade em *software*;
- c) objetivo, estratégias e ações para soluções dos problemas que influenciam na obtenção de padrões internacionais de qualidade e produtividade em *software*.

Mas, para que este termo de referência seja colocado em prática não é necessário somente desejar e, sim que se minimizem o número de defeitos, que se crie mecanismos para garantir prazos e custos, garantir o uso do produto no mercado e melhorar qualidade de novas versões, além de melhorar a qualidade de novos produtos com base na experiência adquirida com os desenvolvimentos anteriores.

Portanto, segundo Rocha (1990), não é suficiente desejar que o *software* a ser produzido atinja um nível adequado de qualidade. Para que este nível seja alcançado, é necessário buscá-lo desde o início do desenvolvimento, uma vez que níveis satisfatórios de qualidade só são assegurados se o grupo de desenvolvedores atuar conscientemente e deliberadamente para atingi-los. Fernandes (1995), vai além quando menciona que esses requisitos podem evoluir com o tempo. Por isso é que este é um processo dinâmico e evolutivo.

Em face disto, a informática que nos apresenta um novo mundo, quebrando todas as barreiras de comunicação e modificando a própria estrutura da indústria e do consumo (INTHURN, 2001), torna-se uma grande aliada para o processo de globalização. Por isso, seria impossível para as empresas que trabalham com o desenvolvimento de *software* ficarem fora deste processo. E sendo a qualidade uma das principais exigências para este novo mercado globalizado; quem possuir produto com qualidade e baixo custo, estará na frente daqueles que não acreditarem nesta “revolução”.

Segundo Ferreira (2003), havendo liberdade de escolha, os clientes das organizações públicas e privadas fazem uma série de exigências. Tudo aquilo que estão acostumados a obter torna-se, para eles, qualidade obrigatória. Portanto, para conquistar clientes, diante da competição atual, é preciso encantá-los e seduzi-los. Qualidade intrínseca, preço acessível, flexibilidade, baixo custo de manutenção, valor de revenda, prazo de entrega, rapidez, variedade de opções, cordialidade no atendimento, boas condições de pagamento, imagem no mercado, segurança pessoal e ambiental são algumas entre as muitas possibilidades exploradas como vantagens competitivas.

Neste contexto, é de extrema importância conhecer as diferentes visões de um *software* para conquistar estes clientes. E através da norma ISO/IEC 9126, tem-se o conhecimento da visão do gerente de desenvolvimento, do desenvolvedor e a do usuário; sendo estas úteis para a tomada de decisões no processo de desenvolvimento de um *software*.

Nos itens a seguir serão descritas estas visões segundo a ISO/IEC 9126.

### 2.2.1 Visão do Usuário

O usuário comum, normalmente, não associa a qualidade de um *software* a fatores relacionados ao seu desenvolvimento. Para ele pouco importa se foi demorado seu desenvolvimento, se houve "retrabalho" por parte dos desenvolvedores, se sofreu muita manutenção até chegar ao seu poder. Para ele o importante é que este produto atenda suas necessidades.

Segundo a norma ISO/IEC 9126, as questões do usuário podem incluir:

- a) as funções requeridas estão disponíveis no *software*?
- b) Quão confiável é o *software*?
- c) Quão eficiente é o *software*?
- d) O *software* é fácil de usar?
- e) Quão fácil é transferir o *software* para outro ambiente?

### 2.2.2 Visão do Desenvolvedor

Para um desenvolvedor são características de qualidade em *software*: a sua conclusão no prazo estabelecido e que após os testes não se detecte nenhuma falha.

### 2.2.3 Visão do Gerente de Desenvolvimento

A visão do gerente de desenvolvimento, apresentada na ISO/IEC 9126, demonstra que o mesmo tem mais interesse em uma medida da qualidade geral do que em uma característica específica, do *software*, uma vez que sua visão objetiva os negócios da empresa.

Côrtes e Chiossi (2001), comentam que o gerente de desenvolvimento procura equilibrar as melhorias da qualidade do produto com outros critérios, tais como respeito aos cronogramas ou previsões de custo.

## 2.3 NORMAS PARA ÁREA DE *SOFTWARE*

Segundo Côrtes e Chiossi (2001), a engenharia de *software* e, em particular, a qualidade de *software*, estão em constante evolução. Para compreender este processo é importante entender como se procedeu a evolução das normas, na área de *software*. Mas primeiramente é necessário compreender que várias entidades de normatização, de mais de noventa países, fazem parte do grupo ISO, dentre as quais estão a *International Electrotechnical Commission* (IEC) e a Associação Brasileira de Normas Técnicas (ABNT).

Num segundo momento, tem que se ter claro que nas últimas décadas, surgiram vários modelos de normas voltadas para *software* como o *Capability Maturity Model* (CMM), o TickIT (derivado da ISO 9000-3), o Trillium, o Bootstrap e outros (CÔRTESE CHIOSSI, 2001). Portanto, tinham sobre a ISO 9001 a vantagem de terem sido concebidos especialmente para o setor de *software*.

Dentre os muitos enfoques dados ao assunto, na tabela 1 são apresentadas as principais normas nacionais e internacionais existentes na área de *software*.

Tabela 1 – normas nacionais e internacionais na área de *software*

Norma	Comentários
ISO 9126	Características da qualidade de produtos de <i>software</i> .
NBR 13596	Versão brasileira da ISO 9126.
ISO 14598	Guias para avaliação de produtos de <i>software</i> , baseados na utilização prática da norma ISO 9126.
ISO 12119	Características de qualidade de pacotes de <i>software</i> ( <i>software</i> de prateleira , vendido como um produto embalado).
IEEE P 1061	<i>Standard for software Quality Metrics Methodology</i> . Normas para métricas de qualidade de <i>software</i> (Produto de <i>software</i> ).
ISO 12207	<i>Software Life Cycle Process</i> . Normas para a qualidade do processo de desenvolvimento de <i>software</i> .
NBR ISO 9001	Sistemas de qualidade – Modelo para garantia de qualidade em projeto, desenvolvimento, instalação e assistência técnica.
NBR ISO 9002	Modelo para a garantia da qualidade em produção, instalação e serviços associados.
NBR ISO 9003	Gestão de qualidade e garantia da qualidade. Aplicação da norma ISO 9000 para o processo de desenvolvimento de <i>software</i> .
NBR ISO 10011	Auditoria de sistemas de qualidade (processo)

FONTE: Barreto (1997)

Em face da deficiência que a ISO 9001 apresentava, a ISO decidiu pela criação de normas com padrão internacional para *software*. E, procurou torná-las, de certa maneira, compatíveis com os modelos existentes (*upward compatible*).

Para isso, a ISO e a IEC realizaram um trabalho em conjunto na elaboração de um modelo que permitisse avaliar a qualidade dos produtos de *software*. Entretanto, apesar da grande virtude da série de normas ISO/IEC 9126, existe uma grande dificuldade na sua aplicação prática devido às características e particularidades dos produtos de *software* (TSUKUMO et al, 1996). Então, com base nas experiências de avaliação da qualidade dos produtos de *software* já publicadas, percebe-se a necessidade de um maior detalhamento ou

adequação da série de normas ISO/IEC 9126 para cada grupo de produto de *software* que se pretende avaliar (SANDRI, 1998). Assim sendo, a NBR ISO/IEC 12119, que trata dos requisitos da qualidade e testes de pacotes de *software*; descreve com profundidade cada uma das características e subcaracterísticas mencionadas na norma ISO/IEC 9126, passa a complementá-la (BARRETO, 1997).

Tendo em vista que este trabalho visa auxiliar na avaliação da qualidade de produtos de *software* para automação comercial e, não no seu desenvolvimento, as normas relativas ao processo de *software* (com, por exemplo, a ISO 9000-3), não serão abordadas, ficando restritas as normas ISO/IEC 9126 E NBR ISO/IEC 12119.

### 3 NORMAS ISO/IEC 9126 E NBR ISO/IEC 12119

A norma ISO/IEC 9126 trata da tecnologia de informação, com base na avaliação de produtos de *software*, das características de qualidade e diretrizes para o seu uso, enquanto a NBR ISO/IEC 12119 trata da tecnologia da informação de pacotes de *software*, testes e requisitos de qualidade. Neste capítulo será abordada a qualidade em produtos de *software* bem como as respectivas normas.

#### 3.1 QUALIDADE DE PRODUTO DE *SOFTWARE*

Há algumas décadas, a preocupação com a qualidade estava centrada diretamente no produto final. Esta abordagem parecia apropriada àquela época, uma vez que estava mais próxima da percepção do usuário. Dentro deste espírito realizava-se o chamado “controle da qualidade”, verificando-se a adequação do produto nos estágios finais no processo de produção (CÔRTEZ E CHIOSSI, 2001).

Na área de *software* foi semelhante, apesar de ser mais moderno. Tinha-se em mente que com a concentração dos esforços em sua depuração, liberando-o das possíveis falhas, o *software* poderia ser entregue. Uma vez garantido seu funcionamento, o mesmo era considerado de boa qualidade.

Com o passar do tempo dois fatores causaram mudanças nesta abordagem. Em primeiro lugar houve o fortalecimento do conceito de foco no cliente e/ou no mercado, que passou a ter um peso maior que o fabricante na definição das características desejáveis em um produto de *software*. E, em segundo lugar, com a evolução da tecnologia, um grande número de recursos e características tornou-se disponível, como por exemplo interface gráfica e suporte pós venda que se tornaram algo obrigatório (CÔRTEZ E CHIOSSI, 2001).

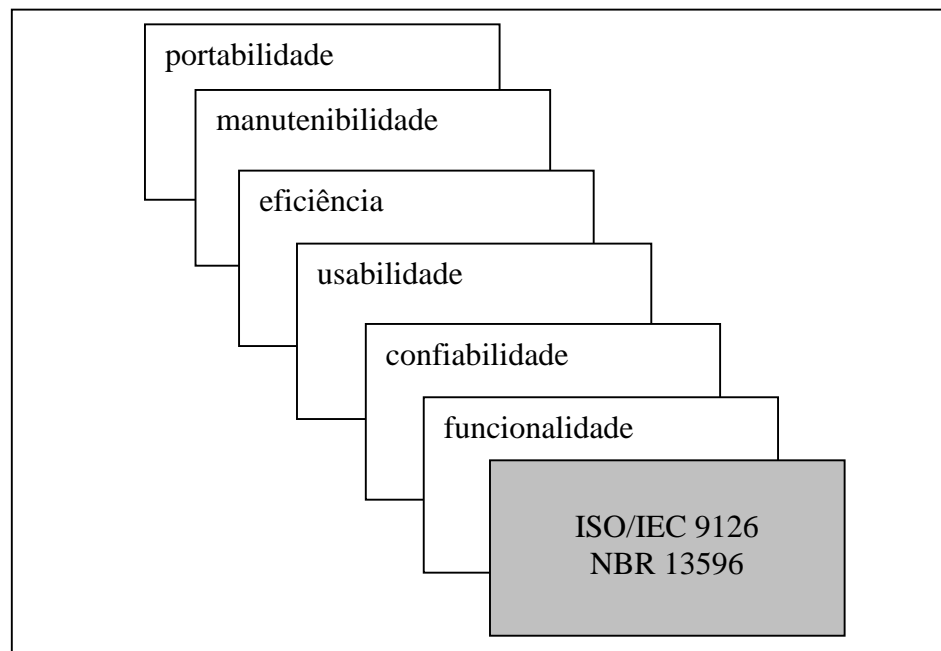
A evolução das tecnologias torna o que antes era um fator diferencial, em mais um item agregado ao produto. Podemos tomar um exemplo de Côrtes e Chiossi (2001), no qual mencionam que o controle remoto dos aparelhos de televisão antigamente era fator de competitividade e hoje em dia é algo obrigatório, não se pode imaginar um aparelho de televisão sem um controle remoto. O mesmo pode ser observado para produtos de *software*, que nem sempre apresentavam uma interface gráfica amigável com o usuário. Atualmente, não se pode imaginar um *software* sem interface gráfica amigável ou suporte pós-venda.

Em se tratando de qualidade de produtos, de modo geral, pode-se usar padrões de comparação como comprimento, largura, peso, cor, textura, etc. Porém quando se fala em qualidade de *software* não se pode medir dessa forma, pois não se lida com produtos mensuráveis fisicamente, mas com produtos que requerem verificação de forma diferente do habitual.

Atualmente existem diversas propostas de sistemas de qualidade, sendo a ISO a mais utilizada e aceita no mundo todo. Esta organização publicou uma norma que representa a atual padronização mundial para a qualidade de produtos de *software*. Esta norma denominada ISO/IEC 9126, foi publicada em 1991 e possui sua tradução para o Brasil, publicada em agosto de 1996, como NBR 13596.

### 3.2 NORMA ISO/IEC 9126 (NBR 13596)

A norma ISO/IEC 9126 (NBR 13596), lista um conjunto de características (figura 1) e subcaracterísticas que devem ser verificadas em um *software* para que este seja considerado de qualidade abrangendo seis grandes grupos:



Fonte: Inthurn (2001)

FIGURA 1 – Conjunto de características a serem verificadas num *software* – norma ISO/IEC 9126 / NBR 13596



- a) Funcionalidade – refere-se a existência de um conjunto de funções que satisfazem necessidades explícitas ou implícitas e suas propriedades específicas. Propõem como subcaracterísticas:
- Adequação – avalia a presença e a validade de um conjunto de funções para tarefas específicas.
  - Precisão – atributos do sistema que proporcionam a suficiente exatidão nos cálculos e resultados, de modo a satisfazer a utilização pretendida pelo usuário.
  - Interoperabilidade – atributos do sistema que suportam a habilidade de interagir com sistemas específicos, conectar-se com outros sistemas.
  - Conformidade – avalia a conformidade do sistema com relação a padrões funcionais ou regulamentações legais.
  - Segurança – refere-se a habilidade do sistema em evitar acesso não autorizado, deliberado ou acidentalmente, a programa e dados.
- b) Conformidade – é uma série de atributos que suportam a capacidade do sistema de manter seu nível de execução sem erros. Propõem como subcaracterísticas:
- Maturidade – atributos do sistema que evidenciam a frequência de falhas por defeitos no sistema.
  - Tolerância a falhas – atributos do sistema que tem com objetivo manter um nível específico de desempenho pré-definido em casos de falhas no sistema ou de violação ou de suas interfaces especificadas.
  - Recuperabilidade – atributos do sistema que tem por objetivo avaliar a capacidade de restabelecer seu nível de desempenho e recuperar os dados diretamente afetados em casos de falhas, no prazo e esforço necessários para isto.
- c) Usabilidade – é o esforço necessário para se utilizar, operar, aprender, bem como para o julgamento individual desse uso, por um conjunto de usuários explícitos ou implícitos. Propõem como subcaracterísticas:
- Compreensibilidade – atributos do sistema que suportam o esforço do usuário para reconhecer os conceitos lógicos na aplicabilidade.
  - Aprendizagem – atributos do sistema que suportam o esforço do usuário em aprender a aplicação.

- Operabilidade – atributos do sistema que suportam o esforço do usuário para operação e controle de operação.
- d) Eficiência – refere-se ao relacionamento entre o nível de desempenho do sistema e a quantidade de recursos usados, sob condições estabelecidas. Suas subcaracterísticas são:
- Comportamento em relação ao tempo – refere-se ao tempo de processamento e resposta e a taxa de tempo no desempenho de sua função.
  - Comportamento em relação aos recursos – refere-se a quantidade de recursos utilizados e a duração deste uso no desempenho de sua função.
- e) Manutenibilidade – é uma série de atributos que suportam um esforço necessário para fazer modificações específicas. São subcaracterísticas:
- Analisabilidade – atributos do sistema que suportam o esforço necessário para diagnosticar as deficiências ou causas de falhas, ou identificação de partes a serem modificadas.
  - Modificabilidade – atributos do sistema que suportam o esforço necessário para modificação, remoção de defeitos ou modificação de ambientes.
  - Estabilidade – refere-se ao risco de defeitos inesperados devido a modificações.
  - Capacidade de testes – atributos do sistema que suportam o esforço necessário para validar o sistema modificado.
- f) Portabilidade – série de atributos que suportam a habilidade do sistema ser transferido de um ambiente para outro qualquer. E, ser operado e manipulado de maneira fácil e adequada em configurações de equipamentos diferentes da original. Subcaracterísticas:
- Adaptabilidade – tecnologias avaliadas devem ser adaptáveis a mudanças de ambientes, tais como, mudanças sociais, mudanças nas ferramentas e mudanças no gerenciamento.
  - Capacidade de instalação – atributos do sistema que suportam o esforço necessário para instalar o sistema em um ambiente específico.
  - Conformidade – atributos que fazem manter os padrões ou convenções relativas à portabilidade.

- Capacidade de substituição – refere-se a oportunidade e ao esforço de usar o sistema no lugar de outro sistema no ambiente deste sistema.

### 3.3 AVALIAÇÃO DA QUALIDADE A PARTIR DA ISO/IEC 9126

A norma ISO/IEC 9126 (NBR 13956), é aplicada na definição dos requisitos da qualidade do *software* e na avaliação dos produtos de *software*.

#### 3.3.1 Usos da ISO/IEC 9126 (NBR 13956)

Os requisitos de qualidade de produtos de *software* podem ser agrupados em características e subcaracterísticas, da seguinte forma (ABNT, 1996):

- a) definição dos requisitos da qualidade de um produto de *software*;
- b) avaliação das especificações do *software*, durante o desenvolvimento, para verificar se os requisitos da qualidade estão sendo atendidos;
- c) descrição das características e atributos do *software* implementado, por exemplo, nos manuais do usuário;
- d) avaliação do *software* desenvolvido antes da entrega;
- e) avaliação do *software* desenvolvido antes da aceitação.

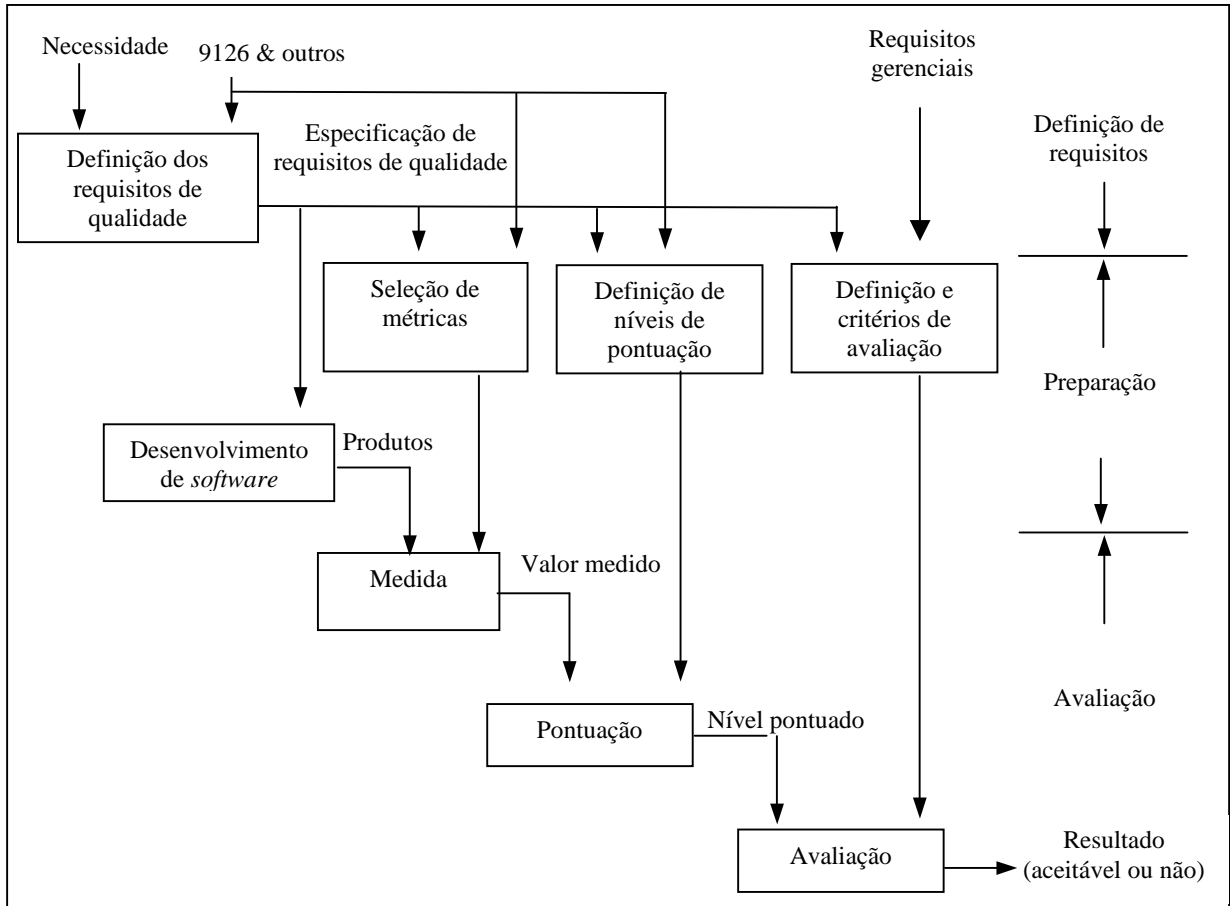
#### 3.3.2 Métricas

Atualmente, segundo a norma, existem apenas algumas métricas geralmente aceitas para as seis características da qualidade que são: funcionalidade, confiabilidade, utilizabilidade, eficiência, manutenibilidade e portabilidade. Portanto, no nível da utilização, a identificação das métricas a serem consideradas depende diretamente dos aspectos que se quer considerar (HEBERT & PRICE, 1995).

Ao estabelecer estas métricas, assim como o critério numérico de aceitação final, deve-se levar em consideração a classe de aplicação do produto de *software*. Por exemplo, a confiabilidade é muito importante em produtos de missão crítica (*software* para controle de transação financeira, por exemplo), a eficiência é importante em aplicações de tempo real e a usabilidade é importante para aplicações interativas, visando o usuário final não especializado.

### 3.3.3 O Processo de Avaliação

A figura 2 apresenta os passos principais na avaliação da qualidade de produtos de *software*.



Fonte: NBR 13596

FIGURA 2 – Modelo do processo de avaliação

O processo de avaliação desenvolve-se em três estágios:

- a) Definição dos requisitos da qualidade – é feita a definição das características e subcaracterísticas em função da área de aplicação do produto de *software*. E, esta definição deve ser feita antes do início do desenvolvimento do mesmo. Produtos de maior porte devem ser subdivididos em módulos e cada um destes deve ter seus próprios conjuntos de características e subcaracterísticas;
- b) Preparação da avaliação
  - Seleção das métricas da qualidade - escolha dos critérios para associar quantificações numéricas para cada um dos atributos. Essas métricas podem variar ao longo do ciclo de desenvolvimento, sem deixar de lado a perspectiva de avaliação do usuário;

- Definição dos níveis de pontuação – os resultados quantificados são mapeados, em uma escala com regiões sugeridas pela norma, três para pontuação *satisfatória* (excelente, bom e razoável) e uma para a pontuação *insatisfatória*;
  - Definição dos critérios de avaliação – definição dos critérios para fazer o mapeamento das características para valores numéricos.
- c) Avaliação Propriamente
- Medida – aplicação das métricas definidas ao produto de *software*;
  - Pontuação – determinação dos valores de pontuação;
  - Avaliação – determinação do resultado final, em aceitação ou não da qualidade do produto.

#### 3.4 NORMA NBR ISO/IEC 12119

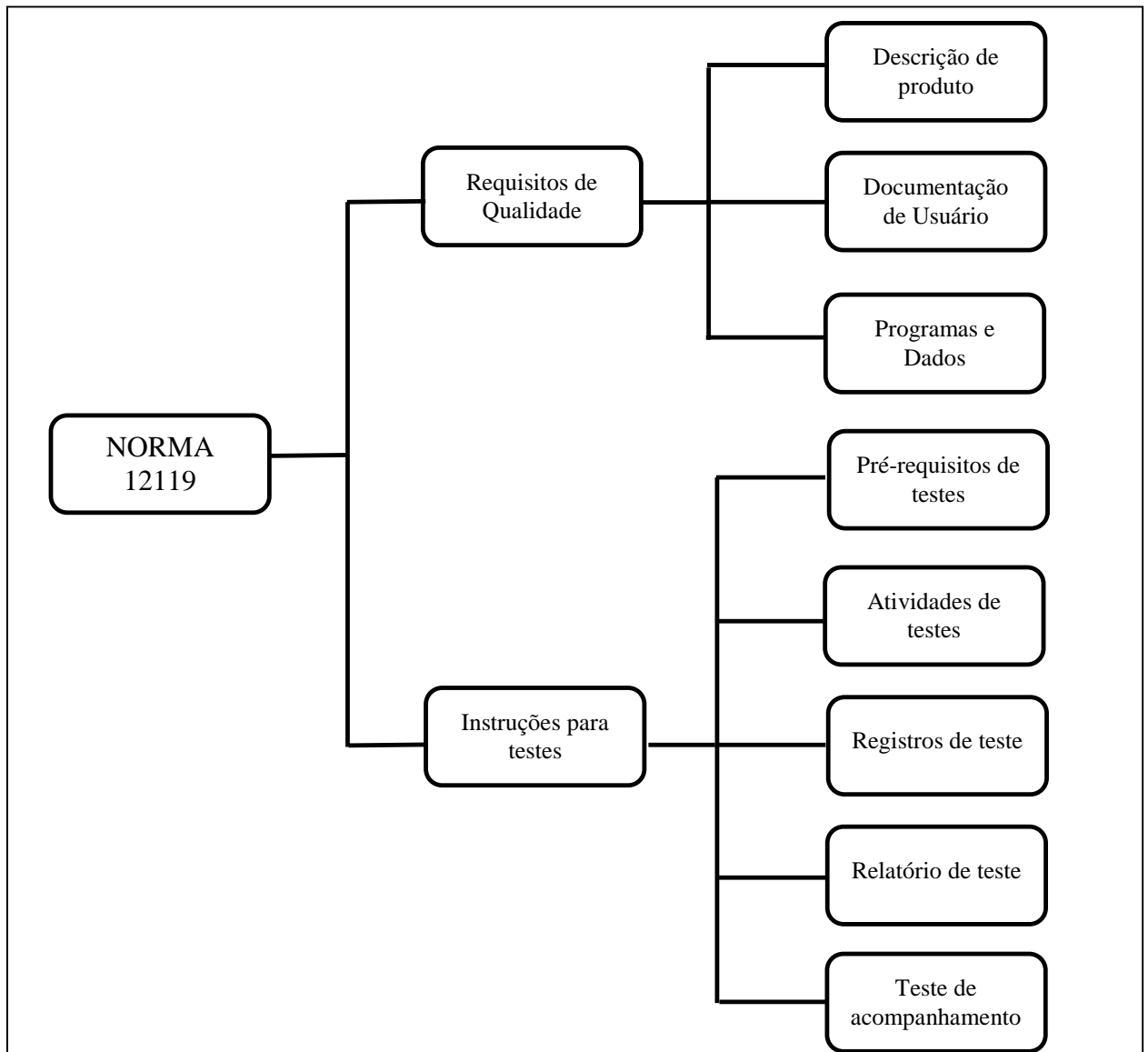
Como foi visto, a norma ISO/IEC 9126 define características e subcaracterísticas de qualidade de *software* de modo geral, porém somente com seu uso há uma grande dificuldade na sua aplicação prática, devido às características e particularidades dos produtos de *software*. Percebeu-se então a necessidade de um maior detalhamento ou adequação desta norma para que se pudessem avaliar grupos de produtos de *software*. A norma NBR ISO/IEC 12119, aplicável a pacotes de *software*, vem justamente definir com profundidade, cada uma dessas características e subcaracterísticas da norma ISO/IEC 9126.

A ABNT (1998), define como pacotes de *software* processadores de texto, planilhas eletrônicas, bancos de dados, *software* gráfico, programas para funções técnicas ou científicas e programas utilitários.

Segundo Ferreira (2003), a norma NBR ISO/IEC 12119 pode ser utilizada pelo produtor de *software* para realizar uma comparação com produtos semelhantes no mercado e auxiliar no desenvolvimento, através de um processo de evolução contínua da avaliação do produto. Esta norma pode ainda ser utilizada por potenciais usuários que desejam saber se um determinado pacote de *software* irá satisfazer suas necessidades; bem como pode ser utilizada por entidades de certificação, laboratórios de teste, auditores, dentre outros.

Conforme a ABNT (1998), a NBR ISO/IEC 12119 estabelece os requisitos de qualidade para pacotes de *software*, estabelece ainda instruções de como testar um pacote de *software* com relação aos requisitos estabelecidos, em particular para testes por terceiros. Na

figura 3 está estruturada a norma NBR ISO/IEC 12119, onde é possível ter um melhor entendimento das divisões e subdivisões desta norma.



Fonte: Sandri (1998)

FIGURA 3 – Estrutura da norma NBR ISO/IEC 12119

Um pacote de *software* estará em conformidade com a norma NBR ISO/IEC 12119 se atender todos os requisitos definidos a seguir, as recomendações indicadas pelo uso da forma verbal “convêm que” são opcionais (ABNT, 1998).

### 3.4.1 Descrição do Produto

Cada pacote de *software* deve ter uma descrição de produto.

A descrição de produto define o produto e é uma parte do conjunto de documentação deste. Ela fornece informação sobre a documentação de usuário, programas e, se existirem,

sobre os dados. A descrição de produto deve estar disponível para pessoas interessadas no produto.

#### 3.4.1.1 Requisitos Gerais sobre o Conteúdo da Descrição

Convém que a descrição do produto seja suficientemente inteligível, completa e possua boa organização e apresentação, a fim de auxiliar os compradores em potencial na avaliação da adequação do produto às suas necessidades, antes de comprá-lo. Deve ser livre de inconsistências internas. Convém que cada termo tenha um único significado.

#### 3.4.1.2 Identificações e Indicações

- a) Identificação da descrição do produto – a descrição do produto deve possuir uma única identificação de documento. A “descrição de produto” pode ser chamada de formas diferentes, por exemplo: “Descrição Funcional”, “Informação de Produto” e “Folha de Produto”;
- b) Identificação do produto - a descrição de produto deve identificar o produto. A identificação do produto deve ter no mínimo o nome do produto e uma versão ou data. Se existirem duas ou mais variantes mencionadas na descrição de produto, então cada variante deve ter no mínimo o nome do produto, o nome da variante e uma versão ou data;
- c) Fornecedor - a descrição de produto deve conter o nome e o endereço de no mínimo um fornecedor;  
Obs. O nome e endereço não precisam ser impressos; o carimbo é suficiente.
- d) Tarefa - a descrição de produto deve identificar as tarefas que podem ser executadas utilizando-se o produto;
- e) Conformidade e documentos de requisitos - A descrição de produto pode fazer referência aos documentos de requisitos com os quais o produto está em conformidade. Neste caso as edições relevantes devem ser identificadas;
- f) Requisitos de *hardware* e *software* - os requisitos para colocar o produto em uso devem ser especificados, incluindo nomes de fabricantes e identificação do tipo de todos os componentes;
- g) Interface com outros produtos - se a descrição de produtos faz referências a interfaces com outros produtos, as interfaces ou produtos devem ser identificados;
- h) Itens a serem entregues - todo o componente físico do produto fornecido deve ser identificado, em particular todos os documentos impressos e todos os meios de armazenamento de dados;

- i) Instalação - deve ser declarado se a instalação do produto pode ou não ser conduzida pelo usuário;
- j) Suporte - deve ser declarado se o suporte para a operação do produto é oferecido ou não;
- k) Manutenção - deve ser declarado se a manutenção é oferecida ou não. Se a manutenção for oferecida, deve ser declarado especificamente o que é incluído.

#### 3.4.1.3 Declarações sobre Funcionalidade

- a) Visão geral das funções - A descrição de produto deve fornecer uma visão geral das funções disponíveis para o usuário do produto, os dados necessários e as facilidades oferecidas. Deve ser declarada de forma nítida para cada função mencionada (especialmente para uma opção ou variante) se ela é parte:

- do produto;
- de uma extensão do produto integralmente apresentada na descrição do produto;
- de uma extensão do produto referida na descrição do produto;
- de um suplemento sem garantia.

Obs. Nem toda função disponível para o usuário necessita ser mencionada, assim como nem todos detalhes de como uma função é chamada necessitam ser descritos.

- b) Valores limite - se o uso do produto é limitado por valores-limite específicos, estes devem ser fornecidos. Exemplos:
  - valores máximos ou mínimos;
  - comprimento de chaves;
  - número máximo de registro em um arquivo;
  - número máximo de critérios de busca;
  - tamanho mínimo de amostra.
- c) Segurança de acesso - convém que a descrição de produto inclua informações a respeito de maneiras, se fornecidas, para evitar o acesso não autorizado (acidental ou intencional) a programas e dados.



#### 3.4.1.4 Declarações sobre Confiabilidade

Convém que propriedades adicionais do produto sejam descritas para assegurar a capacidade funcional do produto. São exemplos:

- verificação se a entrada é aceitável;
- proteção contra seqüências danosas decorrentes de erro de usuário;
- recuperação de erro.

#### 3.4.1.5 Declarações sobre Usabilidade

- a) Interface com o usuário - o tipo de interface com o usuário deve ser especificado, por exemplo: linha de comando, menu, janelas, teclas de função e função de auxílio.
- b) conhecimento requerido
  - conhecimento de uma área técnica;
  - conhecimento de um sistema operacional;
  - conhecimento que possa ser requerido via treinamento especial;
  - conhecimento de outro idioma diferente daquele em que foi escrita a descrição de produto.

Devem ser declarados todos os idiomas utilizados na documentação de usuário e na interface com o usuário (incluindo mensagem de erro e dados visíveis), tanto os do próprio pacote de *software* como os de todos os produtos mencionados na descrição de produto.

- c) adaptação às necessidades do usuário - se o produto pode ser adaptado pelo usuário, então as ferramentas para adaptação e as condições para seu uso devem ser identificadas. São exemplos:
  - mudança de parâmetros;
  - mudança de algoritmos para computação;
  - atribuição de teclas de função.
- d) proteção contra infrações e direitos autorais - se a proteção técnica contra infrações a direitos autorais pode dificultar a usabilidade, então esta proteção deve ser declarada. São exemplos:
  - proteção técnica contra cópias;
  - datas programadas de expiração de uso;
  - lembretes interativos para pagamento por cópia.

- e) Eficiência de uso e satisfação de usuário - a descrição de produto deve incluir dados sobre a eficiência de uso e satisfação de usuário.

#### 3.4.1.6 Declarações sobre Eficiência

A descrição de produto pode incluir dados sobre o comportamento do produto em relação ao tempo, tais como tempo de resposta e taxa de *throughput* para uma dada função sob condições estabelecidas (por exemplo, a configuração do sistema).

#### 3.4.1.7 Declarações sobre Manutenibilidade

A descrição de produto pode conter declarações sobre manutenibilidade.

#### 3.4.1.8 Declarações sobre Portabilidade

A descrição do produto pode conter declarações sobre portabilidade.

### 3.4.2 Documentação de Usuário

A documentação de usuário envolve os seguintes itens: completitude, correção, consistência, inteligibilidade e a apresentação e organização, conforme apresentado a seguir.

#### 3.4.2.1 Completitude

A documentação de usuário deve conter as informações necessárias para o uso do produto. Todas as funções estabelecidas na descrição de produto e todas as funções do programa que os usuários tenham acesso devem ser completamente descritas na documentação de usuário.

Todo valor limite citado na descrição do produto deve ser repetido na documentação de usuário.

Se a instalação puder ser executada pelo usuário, a documentação de usuário deverá incluir um manual de instalação contendo todas as informações necessárias. Convém que o manual de instalação estabeleça o espaço mínimo e máximo para instalação do produto.

Se algum tipo de manutenção puder ser executada pelo usuário, a documentação de usuário deverá incluir um manual de manutenção de programa contendo todas as informações necessárias para essa manutenção.

#### 3.4.2.2 Correção

Todas as informações na documentação de usuário devem estar corretas. Além disso, convém que sua apresentação não contenha ambigüidade nem erros.

#### 3.4.2.3 Consistência

Os documentos da documentação de usuário não devem apresentar contradições internas entre si e com a descrição de produto. Convém que cada termo tenha um significado único em toda documentação.

#### 3.4.2.4 Inteligibilidade

Convém que a documentação de usuário seja inteligível pela classe de usuários que normalmente executa a tarefa a ser atendida pelo produto, utilizando, por exemplo, uma seleção apropriada de termos, exibições gráficas explicações detalhadas e citando fontes úteis de informação.

#### 3.4.2.5 Apresentação e Organização

Convém que a documentação de usuário possua boa apresentação e organização, de tal modo que quaisquer relacionamentos sejam facilmente identificados.

Convém que todo documento tenha índice analítico e remissivo.

Se um documento não estiver na forma impressa, convém que o procedimento para impressão seja indicado.

### 3.4.3 Programas e Dados

Programas e dados envolvem funcionalidade, confiabilidade, usabilidade, eficiência, manutenibilidade e portabilidade.

#### 3.4.3.1 Funcionalidade

- a) instalação - se a instalação puder ser realizada pelo usuário, deve ser possível instalar os programas com sucesso, segundo as informações contidas no manual de

instalação. Os requisitos de *hardware* e *software* apresentados na descrição de produto devem ser suficientes para a instalação dos programas.

Após a instalação deve ser possível identificar se os programas funcionam, por exemplo, usando guias de teste fornecidos ou através de auto teste com as mensagens correspondentes;

- b) presença de funções - todas as funções mencionadas na documentação de usuário devem ser executáveis na forma nela descrita, com os correspondentes recursos, propriedades e dados, e dentro dos valores-limite fornecidos;
- c) correção - os programas e dados devem corresponder a todas as declarações contidas na descrição de produto e na documentação de usuário. As funções devem ser executadas de maneira correta para a realização de uma tarefa. Em particular, programas e dados devem estar de acordo com todos os requisitos definidos em qualquer documento de requisitos citados na descrição de produto;
- d) consistência - os programas e dados não devem conter contradições internas, contradições com a descrição do produto e com a documentação de usuário;

Convém que cada termo tenha significado único em toda a documentação.

Convém que o controle da operação do programa pelo usuário e o comportamento do programa (por exemplo: mensagens, formatos de tela de entrada e relatórios impressos) sejam estruturados de maneira uniforme.

#### 3.4.3.2 Confiabilidade

O sistema, compreendendo *hardware* e *software*, bem como os programas que pertencem ao produto, não devem entrar em um estado no qual o usuário não consiga controlá-lo, nem deve corromper ou perder dados.

Este requisito deve ser cumprido ainda que:

- a capacidade seja explorada até os limites especificados;
- tentativas sejam feitas para explorar a capacidade além dos limites especificados;
- uma entrada incorreta seja feita pelo usuário ou por outros programas listados na descrição de produto;
- instruções explícitas na documentação de usuário sejam violadas.

Estão excluídas somente possibilidades de interrupção do *hardware* e do sistema operacional que não podem ser controladas por nenhum programa (por exemplo, a tecla ou combinações de teclas para reinicializar o sistema operacional).

Os programas devem reconhecer as violações de sintaxe estabelecida para entrada de dados. No caso de um programa reconhecer uma entrada como errônea ou indefinida ele não deve processá-la como uma entrada permitida.

#### 3.4.3.3 Usabilidade

a) inteligibilidade - convém que as mensagens e resultados dos programas sejam inteligíveis, por exemplo:

- por uma seleção adequada de termos;
- por representações gráficas;
- pelo fornecimento de informações básicas que facilitem o entendimento;
- pelas explicações dadas por uma função de auxílio (*Help*).

As mensagens de erro devem fornecer informações detalhas, explicando a sua causa ou forma de correção (por exemplo, por uma referência a um item na documentação de usuário);

b) apresentação e organização - cada meio de armazenamento de dados deve apresentar a identificação do produto e, se existir mais de um meio, eles devem ser distinguidos por um número ou texto.

Deve ser sempre possível ao usuário, quando estiver trabalhando com os programas, descobrir qual função está sendo executada.

Convém que os programas forneçam ao usuário informações claramente visíveis e fáceis de serem lidas. Convém que o usuário seja guiado por informações codificadas e agrupadas adequadamente. Onde for necessário, convém que os programas alertem o usuário.

Convém que as mensagens dos programas sejam projetadas de forma que o usuário possa diferenciá-las facilmente pelo tipo, por exemplo:

- confirmação;
- solicitações;
- advertências;

- mensagens de erro.

Convém que os formatos de telas de entrada, de relatórios e de outras entradas e saídas sejam projetados para serem claros e com boa apresentação e organização. Algumas alternativas poderiam ser:

- campos alfanuméricos sejam alinhados pela esquerda;
- campos numéricos sejam alinhados pela direita;
- em tabelas, pontos decimais e vírgulas sejam colocados na mesma linha vertical;
- limites dos campos sejam reconhecidos;
- campos obrigatórios sejam reconhecidos como tal;
- na detecção de falhas de entrada, as mesmas sejam imediatamente realçadas na tela;
- quando ocorrer uma mudança no conteúdo da tela, o usuário seja alertado por um sinal auditivo ou visual.

- c) operacionalidade - a execução de funções que tem conseqüências graves deve ser reversível, ou os programas devem dar uma clara advertência sobre as conseqüências e requisitar a confirmação antes da execução do comando. Em particular, o processo de apagar dados ou sobrepô-los, bem como de interromper um processamento demorado, que tenha conseqüências graves.

Se um texto de documentação é exibido em um diálogo, convém que o usuário seja capaz de refazer acesso aos subitens do texto de uma maneira direta, por exemplo: pela seleção de uma tabela de conteúdo exibida na tela, ou por uma função de busca baseada em palavras chaves.

#### 3.4.3.4 Eficiência

Não há exigência. Entretanto, o produto deve estar em conformidade com as declarações de eficiência citadas em sua descrição.

#### 3.4.3.5 Manutenibilidade

Não há exigência. Entretanto, o produto deve estar em conformidade com as declarações de manutenibilidade citadas em sua descrição.

#### 3.4.3.6 Portabilidade

Não há exigência. Entretanto, o produto deve estar em conformidade com as declarações de portabilidade citadas em sua descrição.

## 4 AVALIAÇÃO DA QUALIDADE DE SISTEMAS DE AUTOMAÇÃO COMERCIAL

Este capítulo aborda a automação comercial e suas características, como por exemplo, alguns equipamentos que auxiliam na mesma, além de apresentar algumas características da avaliação propriamente dita.

### 4.1 AUTOMAÇÃO COMERCIAL

Segundo Romano (2001) automação comercial é um processo que visa a mecanização, desburocratização e a busca pela excelência de um ponto de venda, que abrange desde a entrada da mercadoria até a saída da mesma.

A informática é utilizada atualmente em lojas de todos os portes. Isto se deve ao fato dos equipamentos (*hardwares*) e programas (*softwares*) estarem mais baratos e, conseqüentemente, mais acessíveis. Os lojistas estão podendo aplicar novas tecnologias para resolver velhos problemas (CARTILHA, 2001).

Basta olhar a nossa volta para percebermos que as lojas estão se informatizando; supermercados, lojas de departamento, de calçados, de confecções, entre outros, estão apresentando excelentes resultados com a utilização da informática. Porém, é importante que se tenha em mente: automação não é simplesmente tomar uma decisão, comprar os equipamentos e esperar os resultados de braços cruzados. A organização interna da empresa é fator decisivo para que a automação traga os benefícios esperados.

Automatizar é um processo que, uma vez iniciado, podemos dizer que não termina nunca. Por isso, é importante saber exatamente o que se pretende antes mesmo de começar. Outro detalhe que não pode ser esquecido é o treinamento do pessoal. De nada adianta a empresa adquirir os equipamentos mais sofisticados se seus funcionários não souberem tirar proveito das informações e agilidade oferecidas por estes.

Um sistema de informática agiliza todo o processo, elimina os processos repetitivos e isola o usuário das tarefas mecânicas. Entretanto, um sistema de informática não elimina nenhuma das dificuldades burocráticas e nem melhora a qualidade da informação; sendo necessário que alguém “alimente” o sistema, portanto, se entrar informação errada, sai informação errada.



Um bom exemplo de que a informática não elimina problemas específicos quanto a saída de informação indevida é o fato de que escrever uma carta para um cliente hoje em dia é algo muito simples, pois temos o editor de textos que facilita muito a redação desta, possuindo até corretores ortográficos. Editor de textos é o nome de uma família de *software* usado para editar cartas e documentos no computador, e é a categoria de aplicativos mais vendida e difundida no mundo (CARTILHA, 2001). Mas, ninguém se torna um grande escritor se adquirir um excelente editor de textos. O *software* (no caso do editor de texto) apenas facilita a edição de textos, agilizando o processo de impressão, correção e armazenamento. Um editor de texto não ensina ninguém a escrever melhor.

Assim como com o editor de textos, usado no exemplo acima, todos os outros *softwares* também precisam de pessoal treinado para poder tirar proveito daquilo que eles tem a oferecer. Pois, uma pessoa que não está apta a operar um sistema automatizado, se falta conhecimento sobre o mesmo; e poderá passar uma interpretação de que o sistema implementado não corresponde, quando na verdade, o sistema pode estar funcionando perfeitamente e o único problema é sua operação, que está sendo realizada de forma incorreta.

## 4.2 EQUIPAMENTOS PARA AUTOMAÇÃO COMERCIAL

Atualmente, alguns dos instrumentos disponíveis para automação comercial em nível de loja são:

➤ PDV (Ponto de Venda) – utilizados para automatizar as funções dos caixas. Trata-se de um microcomputador (especial ou não), compatível com a arquitetura PC, com periféricos voltados à área de automação comercial, tais como: impressora fiscal, leitor óptico de barras, impressora de cheques, gavetas, teclados com ou sem display e com ou sem *slot*, *display* para clientes; nem sempre todos reunidos. Enfim, uma máquina que substitui uma caixa registradora aliada a todas as vantagens de um PC.

Pelo fato de estar interligado ao sistema interno da loja, dependendo do *software*, os PDV's poderão baixar o estoque, atualizar os mapas de caixas e gerar dados gerenciais;

➤ impressoras de cheques – como o próprio nome diz, é uma máquina destinada à impressão de cheques. Podem ser conectadas ou não ao PDV, e nos modelos de PDV's mais modernos são parte integrante da impressora de cupom fiscal;

➤ leitores ópticos / *scanners* – são equipamentos destinados à leitura de código de barras. Existem leitores de mesa ou manuais e podem ser do tipo CCD (*charge coupled device* - dispositivo de acoplamento por carga, que é projetado para detectar alterações repentinas de tensão) ou *laser* que utiliza a emissão de um raio *laser* de baixa densidade e há captação da reflexão deste raio sobre o código.

Existem diversos fabricantes de leitores ópticos com um número significativo de modelos; de acordo com a aplicação, deverá ser escolhido o tipo de leitor a ser utilizado;

➤ teclados especiais com *displays* e *slots* – são teclados com um número diferenciado de teclas que possuem ou não *display* de cristal líquido;

➤ impressoras de código de barras – são impressoras especiais, destinadas à emissão de etiquetas de código de barras. Dependendo do fabricante e do modelo, as impressoras podem imprimir texto, logotipos, figuras, linhas e vários padrões de código de barras;

➤ impressoras fiscais – são impressoras com uma unidade de processamento próprio. Gerencia e monitora todas as operações fiscais (sujeitas a impostos); possuem um módulo fiscal que têm como função o gerenciamento das informações sobre os totalizadores, alíquotas, situação do módulo fiscal, contadores de operações, de reduções, enfim, é um gerenciador de todos os processos sofridos pela impressora (fiscal e não fiscal);

➤ coletor de dados – trata-se de microcomputadores portáteis destinados à coleta de dados sendo especialmente projetados para uso industrial, comercial e de campo. Essas máquinas podem operar com diversos sistemas operacionais, tais como o *DOS* e o *CP/M*, entre outros;

➤ terminais de consultas de preços – são microcomputadores que possuem alguns periféricos como leitores de código de barras, telas de *touch-screen*, podem ter ou não recursos de multimídia e são destinados a consultas de preços e de estoque;

➤ balanças eletrônicas – as balanças eletrônicas além de executarem suas funções básicas, permitem a interligação com o microcomputador podendo com isso passar informações ao sistema, permitindo assim um controle integrado de estoque.

### 4.3 PROCESSO DE AVALIAÇÃO

A garantia da qualidade do sistema depende de como ele foi planejado e implantado (GUESSER, 1995).

O objetivo de se avaliar um sistema é principalmente saber se ele atende os requisitos básicos de qualidade, e para isto temos as normas ISO/IEC 9126 E A NBR ISO/IEC 12119, porém, em alguns casos o produto pode ter qualidade, mas não atender as necessidades daqueles que estão interessados em sua aquisição e nem sempre é fácil saber isso simplesmente lendo suas características e, para um leigo no assunto, fica muito mais difícil encontrar algo que realmente vai atender suas necessidades.

### 4.4 CARACTERÍSTICA DA AVALIAÇÃO

A avaliação de um determinado sistema de automação deve seguir os requerimentos especificados abaixo (GUESSER, 1995):

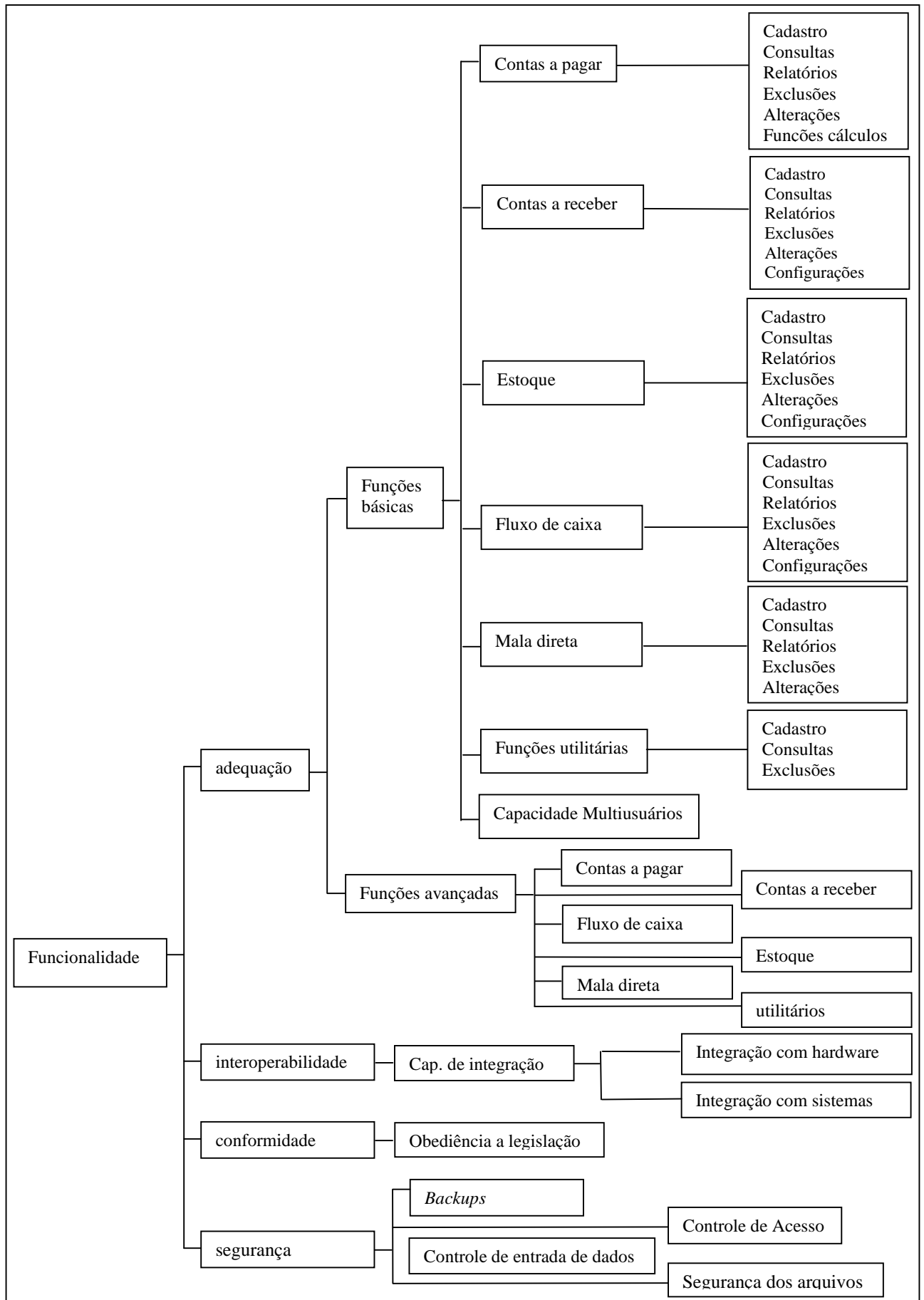
- a) repetibilidade – repetir a avaliação de um mesmo produto com os mesmos requisitos de testes e estes devem dar os mesmos resultados;
- b) reproducibilidade – repetir a avaliação de um mesmo produto com os mesmos requisitos de avaliação em diferentes testes de laboratório e estes devem dar os mesmos resultados;
- c) imparcialidade – a avaliação é desonesta se apresentar um resultado particular;

Precisamos sempre lembrar que a avaliação de um mesmo produto poderá dar diferentes resultados, se forem conduzidas com diferentes especificações.

Sempre que se avaliar produtos diferentes, somente poderá ser comparado o resultado das partes idênticas destes produtos.

### 4.5 PROPOSTA DE DETALHAMENTO DA NORMA ISO 9126 PARA AUTOMAÇÃO COMERCIAL

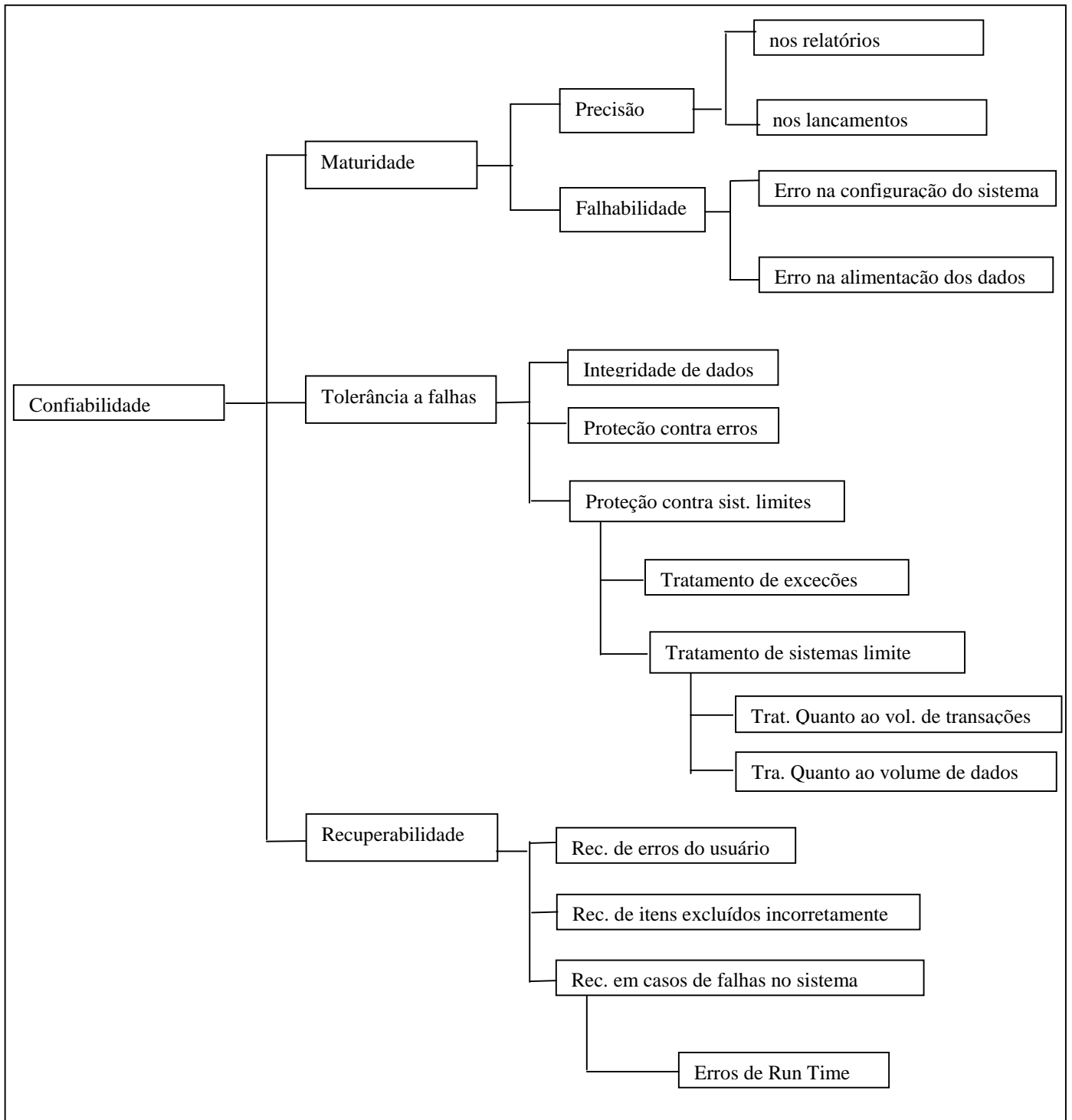
Guessser (1995) em seu trabalho propõem alterações na norma ISO 9126, como por exemplo no item precisão que na norma é uma característica de funcionalidade, porém o *software* pode possuir todos itens necessários para o perfeito funcionamento e mesmo assim não ser preciso, logo, foi sugerido que o mesmo aplicar-se-ia melhor em confiabilidade, pois para ser confiável deveria ser preciso. Na figura 4 pode ser visto o detalhamento da característica funcionalidade com suas respectivas subcaracterísticas.



Fonte: Guesser (1995)

FIGURA 4 – Diagrama da funcionalidade

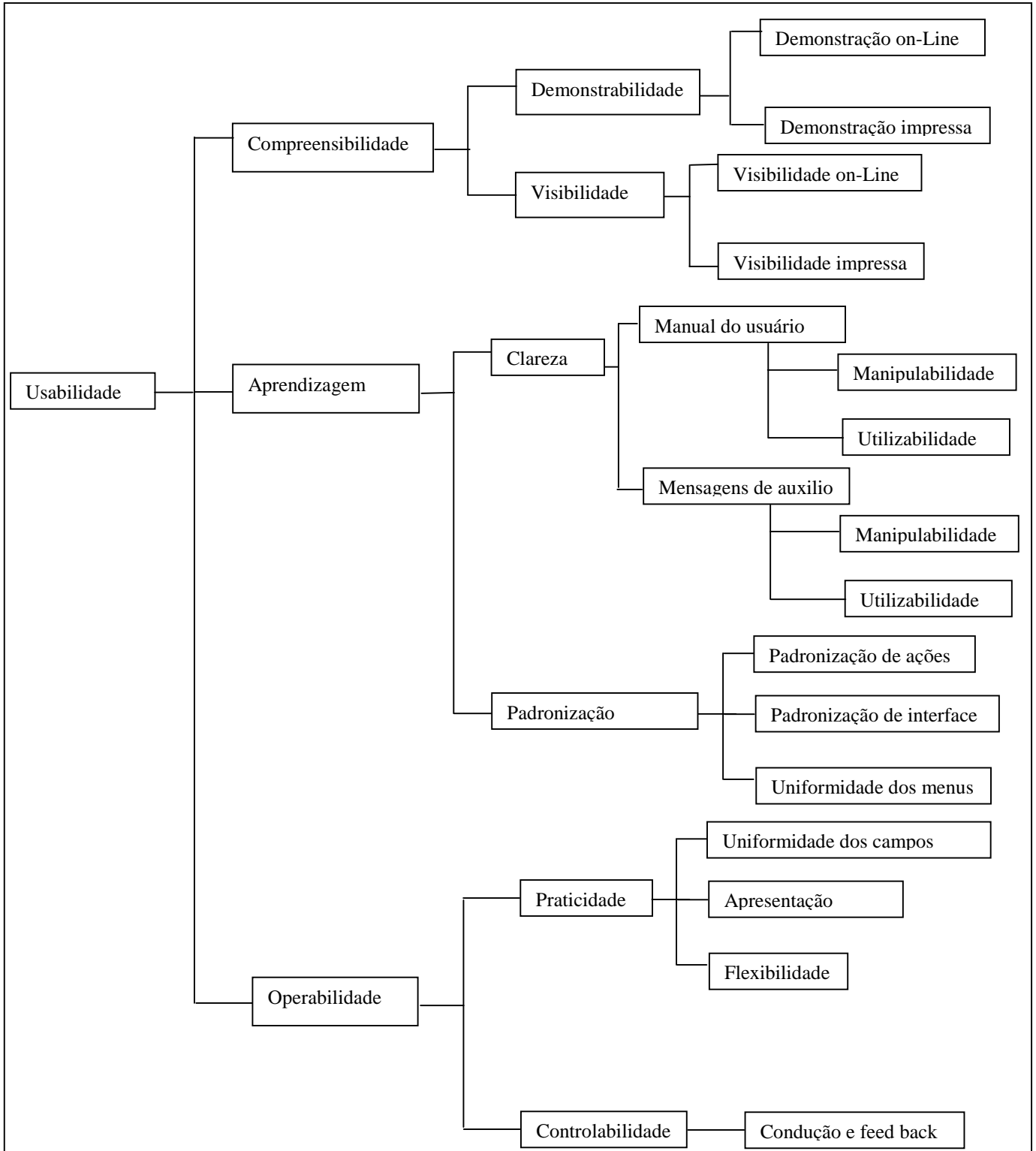
Na figura 5 temos o diagrama confiabilidade onde pode ser visto o item precisão conforme proposto por Guessser (1995).



Fonte: Adaptado de Guessser (1995)

FIGURA 5 – Diagrama de confiabilidade

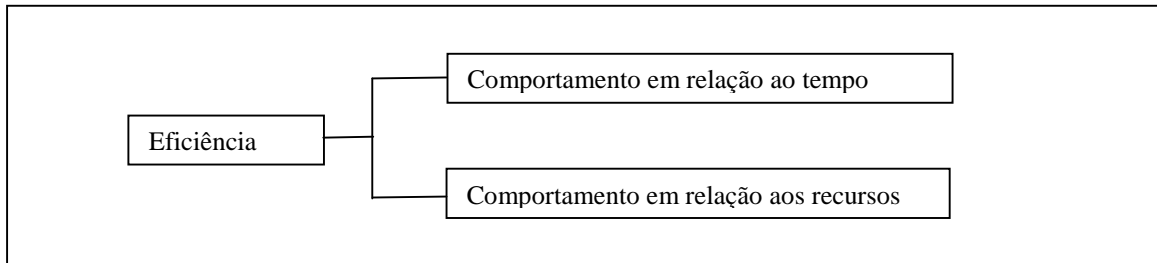
Na figura 6 pode ser visto o diagrama de usabilidade, com suas respectivas subdivisões proposta em Guessser (1995).



Fonte: Guessser (1995)

FIGURA 6 – Diagrama de usabilidade

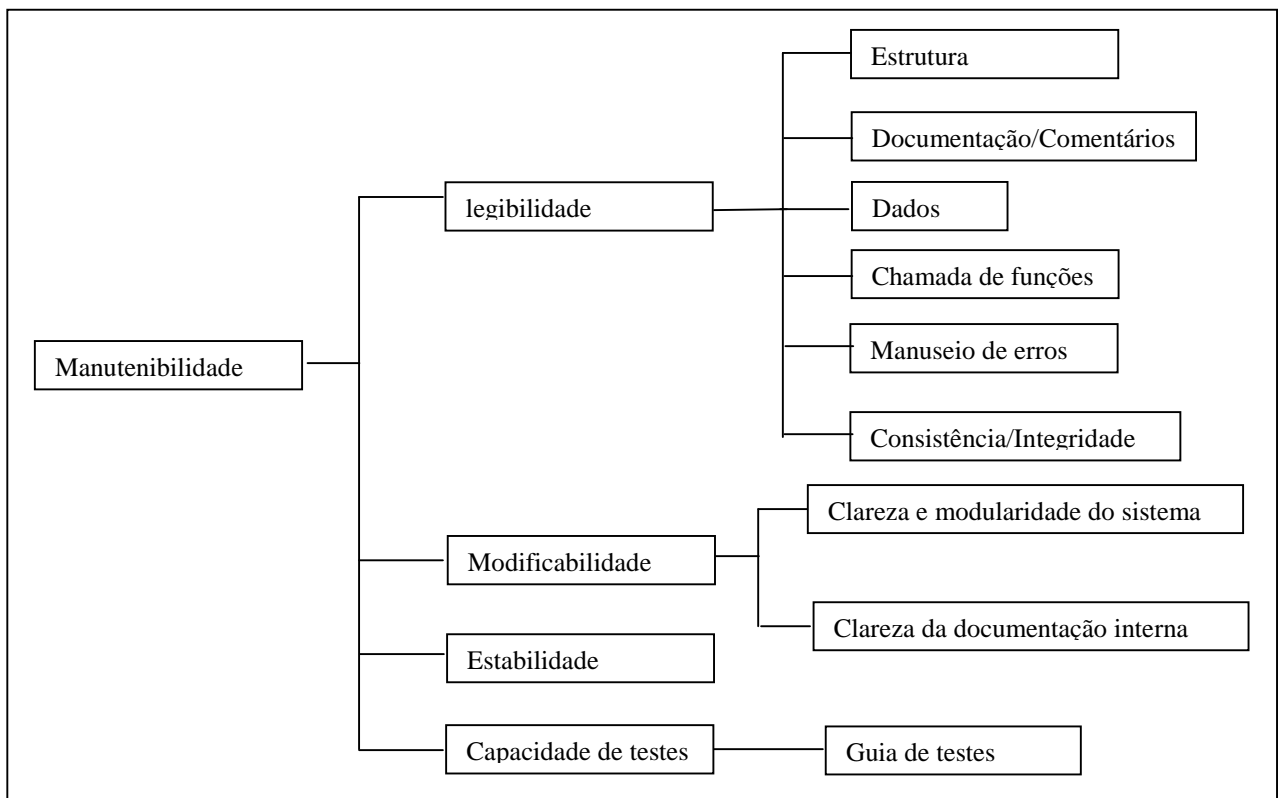
Na figura 7 temos o diagrama de eficiência. A eficiência se refere ao relacionamento entre o nível de desempenho do sistema e a quantidade de recursos usados, sob condições estabelecidas



Fonte: Guessser (1995)

FIGURA 7 – Diagrama de eficiência

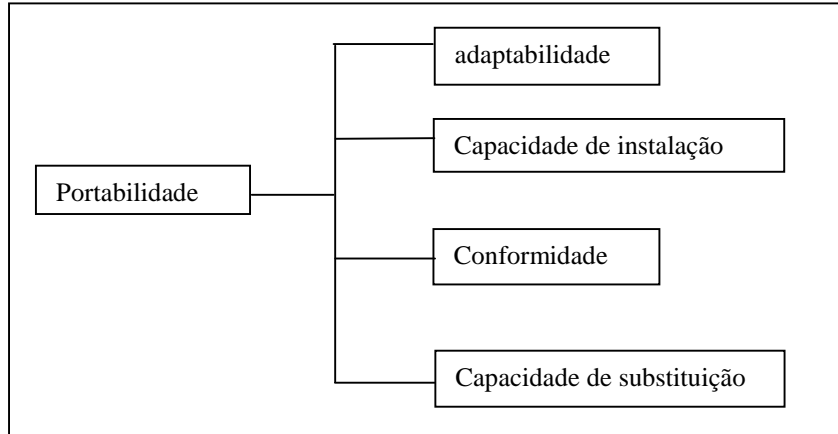
Na figura 8 pode ser visto o diagrama de manutenibilidade que é uma série de atributos necessários para que se faça modificações específicas.



Fonte: Guessser (1995)

FIGURA 8 – Diagrama de Manutenibilidade

O diagrama de portabilidade pode ser visto na figura 9. A portabilidade é uma série de atributos que o sistema possui para que seja possível sua transferência de um ambiente para um outro qualquer.



Fonte: Guessier (1995)

FIGURA 9 – Diagrama de portabilidade



## 5 DESENVOLVIMENTO DO TRABALHO

De acordo com os objetivos propostos para este trabalho, desenvolveu-se um *software* que auxilia na atividade de avaliação da qualidade de produtos de *software*, via *web*, para sistemas de automação comercial, utilizando como base as características e subcaracterísticas da qualidade de *software* propostas pelas normas ISO/IEC 9126 e NBR ISO/IEC 12119. Para melhor entendimento do funcionamento do *software* desenvolvido neste trabalho, e sua avaliação, foram realizados testes com três *softwares*, versão de avaliação e os mesmos se encontram no apêndice A.

### 5.1 ANÁLISE DOS REQUISITOS

A seguir serão apresentados os principais requisitos do trabalho desenvolvido, bem como as funcionalidades propostas e suas características.

A ferramenta CASE utilizada foi a *Power Designer 9*, que é voltada para análise, projeto e correção de desenvolvimento de sistemas. Pode ser classificada como uma ferramenta de documentação de sistemas e engenharia de informação, atuando em análise, projeto, modelagem e documentação de dados existentes.

A avaliação, desenvolvida neste trabalho, pode ser tanto para fins de liberação de um produto no mercado quanto para uma comparação entre produtos de *software*, além de auxiliar no momento de uma aquisição. Portanto o desenvolvimento deste *software*, desde o início, objetivou o auxílio no processo de avaliação de produtos de *software*.

Os requisitos para as perguntas do questionário foram obtidos através de entrevistas, que podem ser vistas no apêndice B, levando em consideração as características das normas estudadas e trabalhos correlatos, que podem ser visto no quadro 1.

Título	Objetivos do trabalho
Protótipo de sistema para auxiliar a avaliação da qualidade de sistemas de automação comercial. (Guesser, 1995).	Implementação de uma ferramenta para auxiliar a avaliação da qualidade de sistemas de automação comercial.
Ferramenta para avaliação da qualidade de produtos de <i>software</i> baseado nas normas ISO/IEC 9126 e NBR ISO/IEC 12119. (Ferreira, 2003).	Especificar e implementar uma ferramenta para auxiliar na avaliação da qualidade de produtos de <i>software</i> .

<p><i>Software</i> de apoio a avaliação da qualidade de pacotes baseado na norma ISO-IEC 12119. (Sandri, 1998)</p>	<p>Especificação e implementação de uma ferramenta de apoio a avaliação da qualidade de pacotes de <i>software</i> baseado na norma NBR ISO/IEC 12119.</p>
<p>Proposta de avaliação da qualidade de produtos de <i>software</i> utilizando a norma ISO/IEC 9126. (Storch, 2000)</p>	<p>Propor um roteiro de avaliação da qualidade de produtos de <i>software</i> baseado na norma ISO/IEC 9126.</p>

QUADRO 1 – Lista de trabalhos correlatos

## 5.2 ESPECIFICAÇÃO

Aqui estão descritas as atividades desempenhadas na fase de especificação do trabalho desenvolvido, apresentando os modelos, diagramas e estruturas de dados que constituem na especificação do trabalho. Para a especificação do *software*, utilizou-se a técnica de análise estruturada e a ferramenta CASE Power Designer 9.

### 5.2.1 Diagrama de Contexto

O diagrama de contexto identifica uma visão macro do sistema, onde está declarado o objetivo principal do problema. Na figura 10 pode ser visto o diagrama de contexto, representado com a notação de Yourdon.

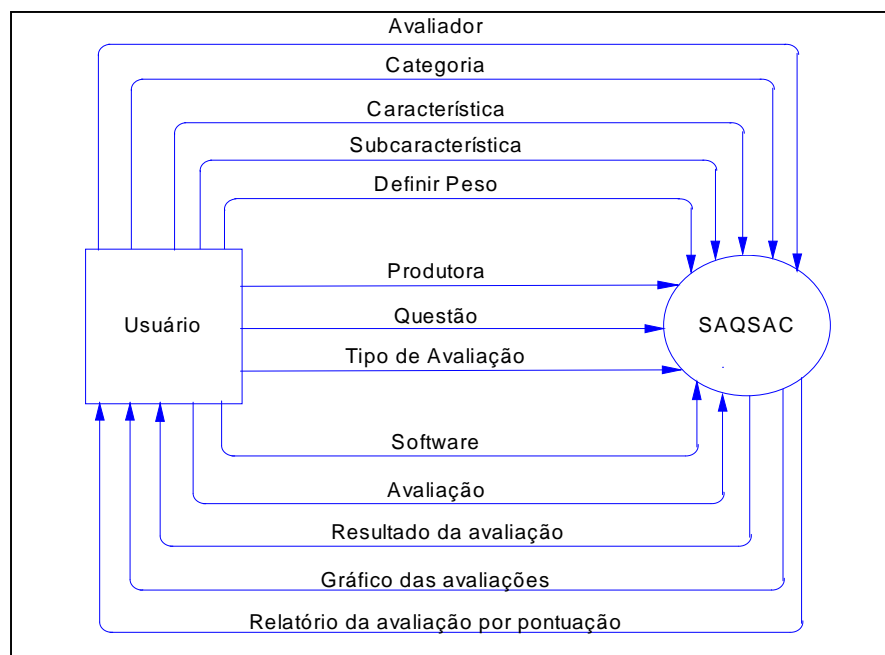


FIGURA 10 – Diagrama de contexto

O sistema de avaliação da qualidade de sistemas de automação comercial possui uma entidade externa denominada usuário/avaliador, este entrará com as informações necessárias para que haja a avaliação, como os cadastros e/ou demais informações como pesos das características e subcaracterísticas.

Vale aqui ressaltar que apesar do próprio usuário poder definir os pesos e perguntas das características e subcaracterísticas a serem avaliadas os mesmos já estarão pré-cadastrados, no intuito de facilitar a utilização do *software*.

### 5.2.2 Diagrama de Fluxo de Dados

O diagrama de fluxo de dados apresentados na figura 11, descreve graficamente o sentido do fluxo das informações no sistema. Identifica as entidades externas com as quais ele se comunica. Através dele pode-se ter uma visão geral do comportamento do sistema.

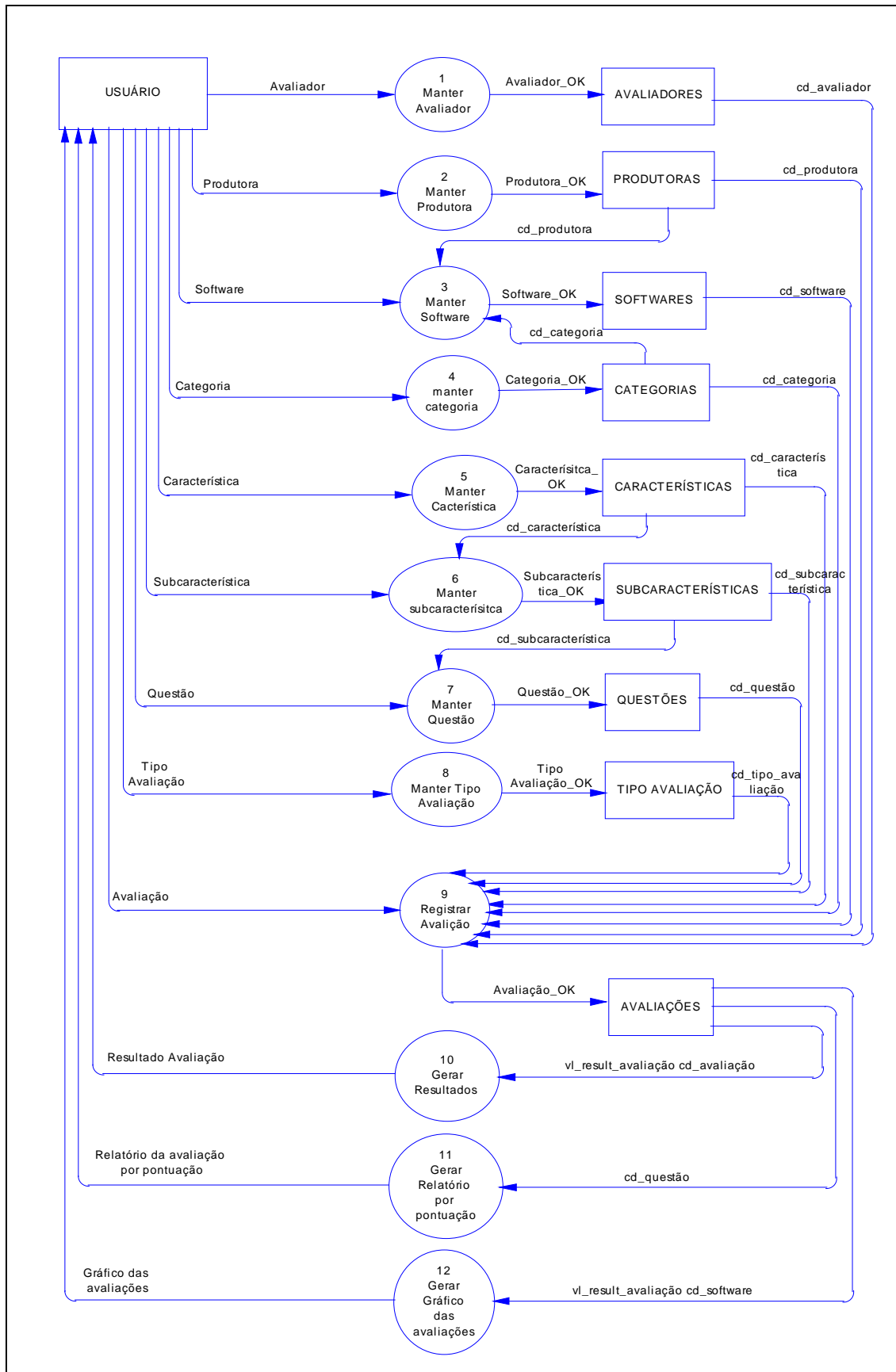


FIGURA 11 – Diagrama de fluxo de dados

### 5.2.3 Diagrama Entidade Relacionamento

O diagrama entidade relacionamento, apresentado na figura 12 e 13, enfatiza os principais objetivos e as entidades do sistema. Através dele temos os detalhes das associações existentes entre as entidades do sistema.

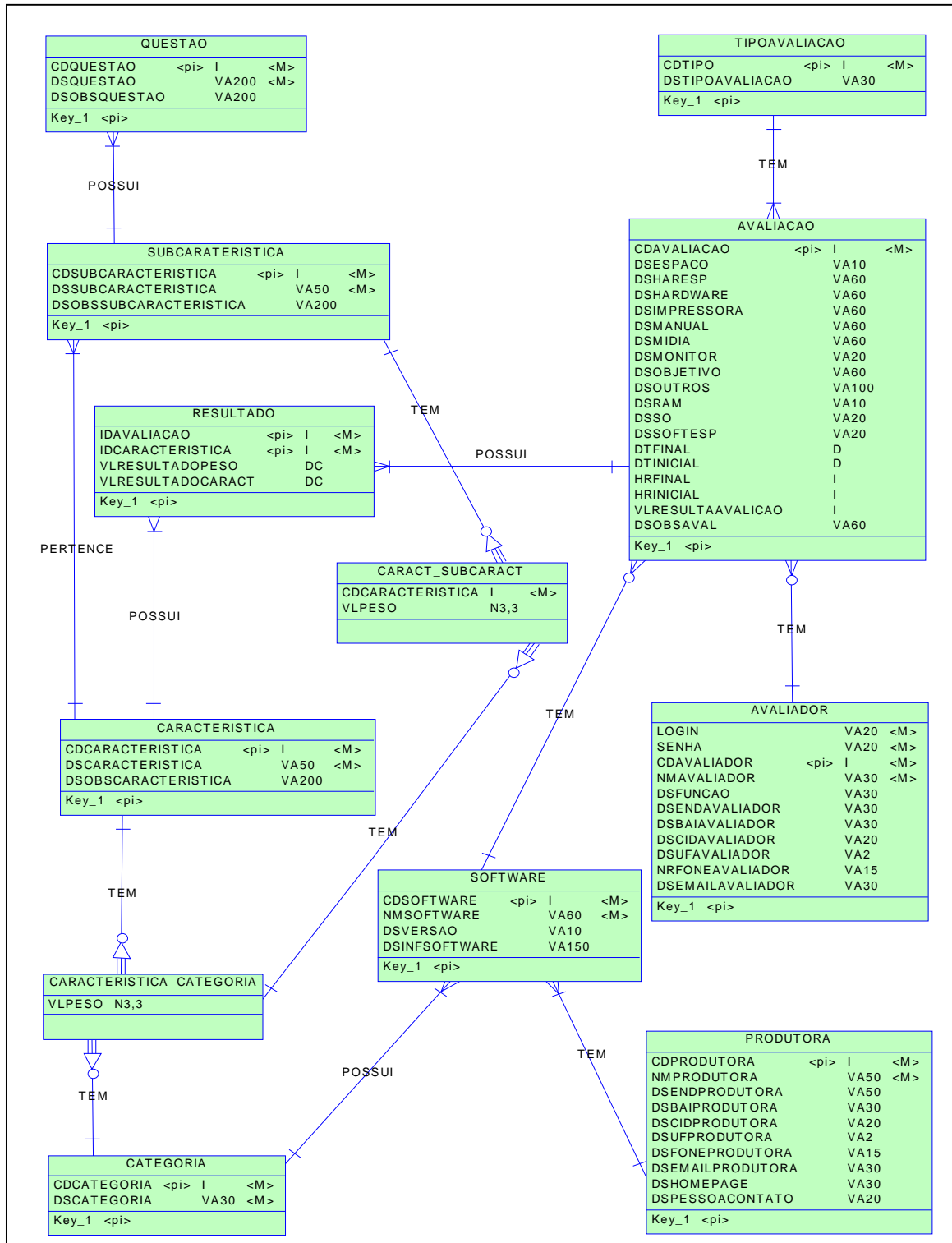


FIGURA 12 – Diagrama entidade relacionamento – modelo lógico

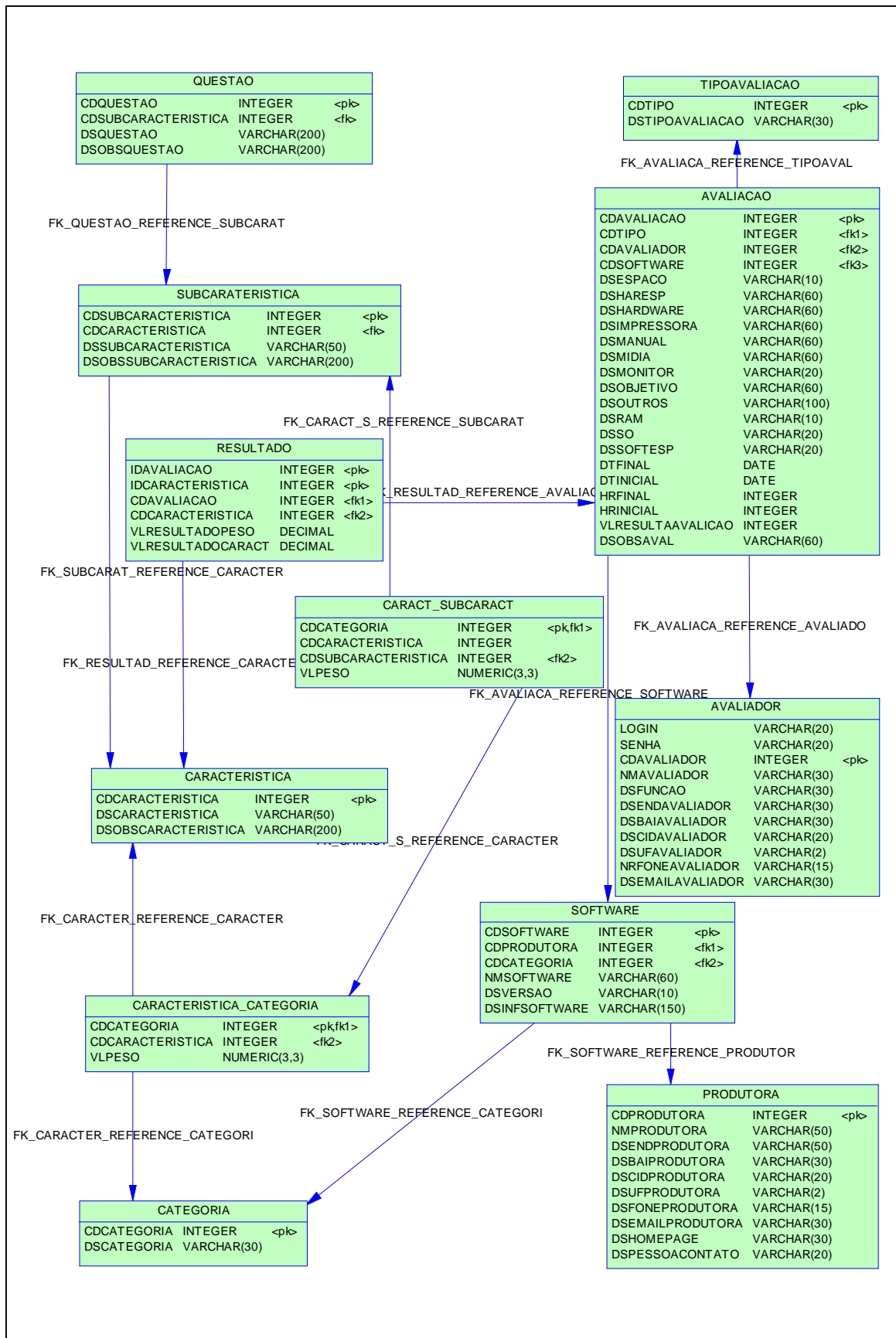


FIGURA 13 – Diagrama entidade relacionamento – modelo físico

### 5.3 IMPLEMENTAÇÃO

Para a implementação foi utilizado o ambiente de desenvolvimento *Delphi 7*, com os recursos do *Intraweb* e para a base de dados, cujo dicionário de dados pode ser visto no apêndice C, foi utilizado o banco de dados *Interbase 6*.

#### 5.3.1 *Intraweb*

*Intraweb* é uma biblioteca de componentes de geração de HTMLs. Esta biblioteca é desenvolvida pela empresa *Atozed Computer Software ltd.*

Segundo Guarasemini (2003?) o *Intraweb* está disponível para várias linguagens de programação, entre elas:

- a) Borland Delphi;
- b) Borland Kylix;
- c) Microsoft Visual net;
- d) Borland J Builder;

O desenvolvimento assemelha-se com o de uma aplicação normal Delphi. Com o *Intraweb* pode-se desenvolver soluções para internet, intranet etc. e as aplicações desenvolvidas podem ser acessadas de *browser* que aceitam HTML4, que são: Mozilla, Netscape 6 ou superior e Internet Explorer 4.0 ou superior.

Os trabalhos dos aplicativos *Intraweb* são semelhantes aos de uma aplicação normal, sendo que a interface não mais será uma janela do Delphi e sim o *browser*. A seção desta aplicação será inicializada pelo usuário utilizando a *URL (Uniform Resource Locator)* de onde se encontra a aplicação. Conforme Araújo Junior (2003) as informações dos usuários são rastreadas pela instância da aplicação em uso, não havendo a possibilidade de perda de informação. Cada usuário possui sua própria seção, não existindo a possibilidade de um usuário interferir na seção do outro.

Para avaliar o *software* foi desenvolvido um *check list*. O desenvolvimento deste se deu através de uma compilação de dados de trabalhos de Guesser (1995), Grupo de estudo de qualidade em *software* (1998), Scarton e Nau (2002) e das normas ISSO/IEC 9126 e NBR ISO/IEC 12119.

Além da compilação de dados acima mencionada, foram realizadas entrevistas (ver apêndice C) com quatro empresas em diferentes estágios de automação, desde a que ainda está na fase de projeto de automação, até a que já se encontra totalmente automatizada, sendo elas: Madeblu industria e comércio de madeiras ltda; Bio bichos clínica veterinária ltda; Querubim calçados ltda e Hering store, identificadas no questionário apenas por Empresa A, B, C e D, não possuindo as letras identificadoras, apresentadas junto ao nome da empresa, qualquer referência aos nomes das mesmas.

Para o desenvolvimento deste trabalho também foi realizada uma revisão bibliográfica (Apêndice E), no intuito de levantar as diferentes abordagens para o termo “qualidade” e conceitos básicos de automação comercial . Além de dar suporte ao presente trabalho, esta revisão é de grande valia pois, permite a localização de diferentes trabalhos correlatos ao assunto num único local .

### 5.3.2 Descrição das Telas do *Software*

A seguir são apresentadas algumas das telas do sistema, com suas respectivas funcionalidades.

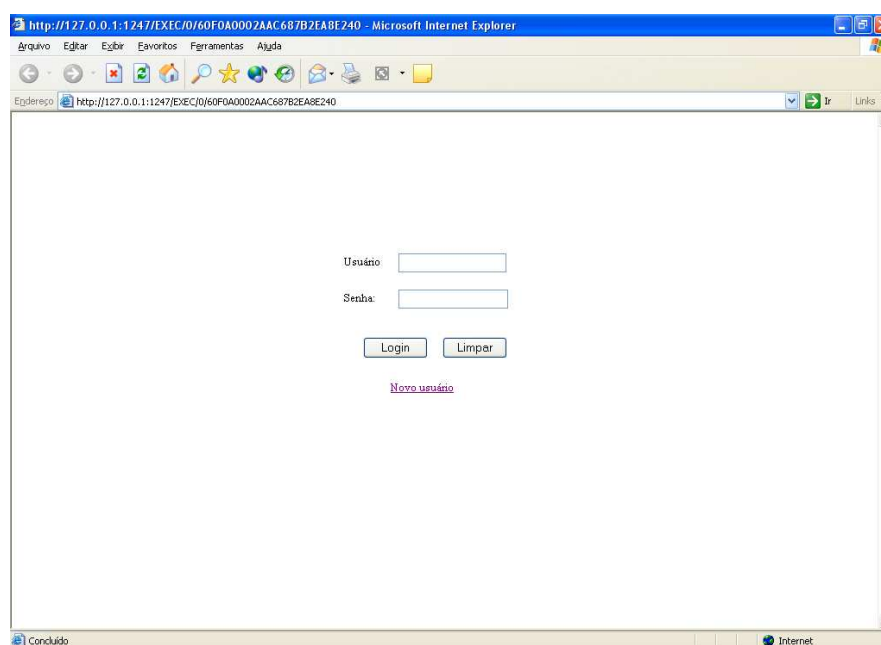


FIGURA 14 – Tela inicial do *software*

A figura 14 apresenta a tela inicial, onde o usuário entra com seu *username* e senha para ter acesso ao sistema. Caso ainda não seja cadastrado, o mesmo deve clicar sobre o *link* novo usuário para ter acesso a tela de cadastro deste.



http://127.0.0.1:3007/EXEC/2/08F1A0008D70CF9A509FE240 - Microsoft Internet Explorer

Endereço: http://127.0.0.1:3007/EXEC/2/08F1A0008D70CF9A509FE240

### Cadastro de Usuário

LOGIN:

SENHA:  CONFIRMA:

NOME:

FUNÇÃO:

ENDEREÇO:

BAIRRO:

CIDADE:

UF:

FONE:

E-MAIL:

Internet

FIGURA 15 – Tela de cadastro de usuário

Na figura 15 está representada a tela de cadastro de novos usuários. Aqui novos usuários devem escolher um *login* e senha, que posteriormente deverão informar para ter acesso ao sistema. Além de informar dados pessoais como endereço, telefone, e-mail, etc.. Após esse cadastro pessoal, o usuário terá acesso ao processo de avaliação do *software*. Para tanto, deve iniciar o processo cadastrando as produtoras do *software*, cujo intuito é avaliar (Figura 16).

http://127.0.0.1:3007/EXEC/7/CC9AA2005C0BA0CD509FE240 - Microsoft Internet Explorer

Endereço: http://127.0.0.1:3007/EXEC/7/CC9AA2005C0BA0CD509FE240

### Cadastro de Produtora

Código:

Produtora:

Endereço:

Bairro:

Cidade:  UF:

Fone:

E-mail:

Home page:

Pessoa contato:

**Cadastros**

- [Categoria](#)
- [Produtora](#)
- [Tipo da avaliação](#)
- [Software](#)
- [Característica](#)
- [Subcaracterística](#)
- [Questões](#)
- [Definir pesos](#)
- [Home](#)

**Avaliador:**  
Marco Antonio

Internet

FIGURA 16 – Cadastro de produtora

A figura 16 mostra a tela de cadastro das produtoras de *software*. Cabe aqui mencionar que o processo de avaliação de um *software* passa preliminarmente por seu cadastro. Entretanto, o cadastro do mesmo só poderá ser feito após o cadastrado da produtora que o desenvolveu. Este cadastro, consta de dados referentes à produtora como: nome, endereço, *home page*, etc.

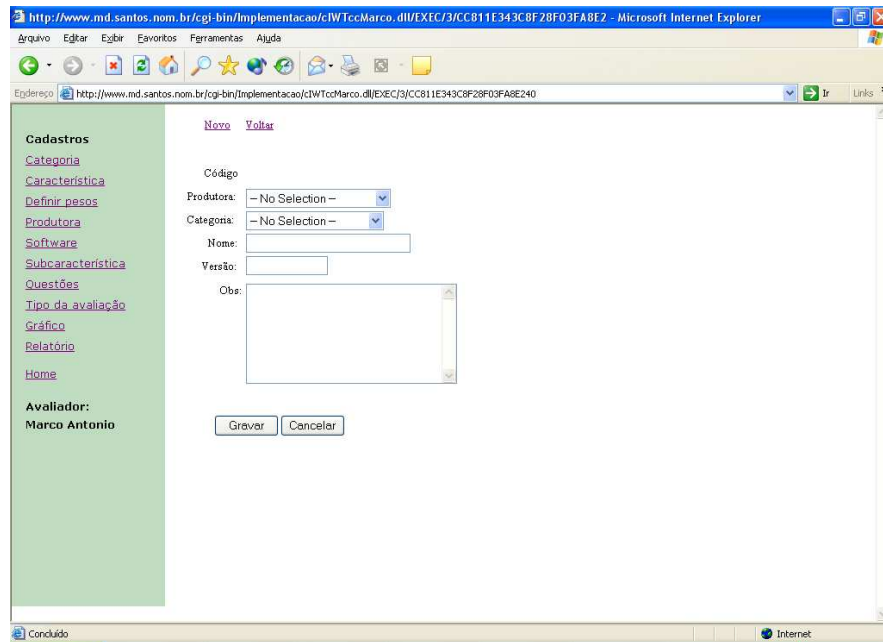


FIGURA 17 – Cadastro do *software*

Após cadastrada a produtora, procede-se ao cadastro do *software* que será avaliado. Na figura 17 pode-se observar a tela de cadastro do mesmo. Vale aqui salientar que o *software* a ser avaliado está vinculado a uma produtora previamente cadastrada, portanto, deve-se selecionar qual a produtora do mesmo, pois, internamente o programa associa o código da produtora ao *software*.

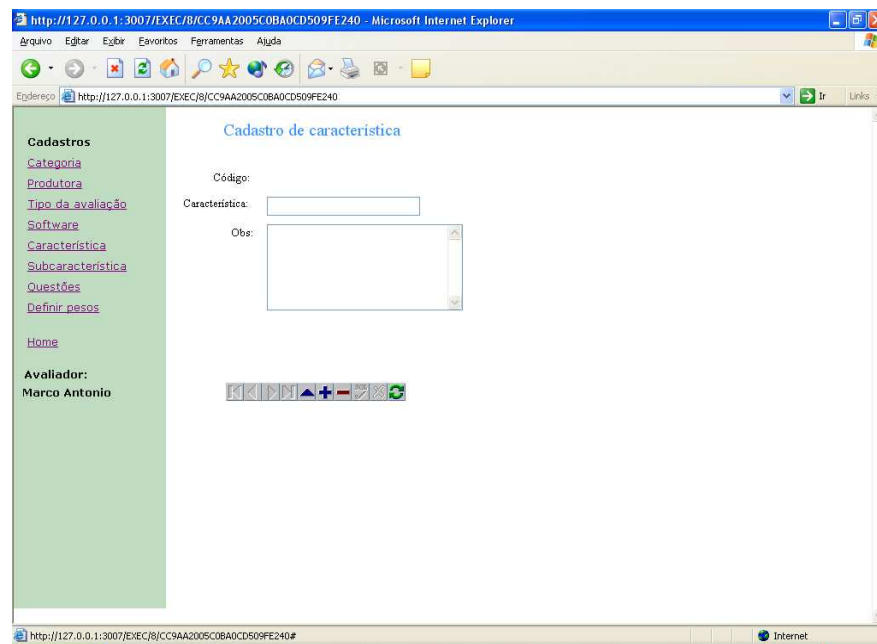


FIGURA 18 – Cadastro das características

Uma vez cadastrado o *software*, passa-se então ao processo de cadastro das características de qualidade, segundo a norma ISO/IEC 9126. Na figura 18 temos a tela para cadastro dessas características. O campo destinado a observação deve ser preenchido com a descrição que define a característica (encontrado na ISO/IEC 9126), para o melhor entendimento da mesma.

O segundo passo é o cadastro das subcaracterísticas, cujo processo é similar ao feito no cadastro da característica, ou seja, o campo observação deve ser preenchido com a descrição do conjunto de atributos que a define (encontrado na ISO/IEC 9126), a tela para cadastrar a mesma pode ser vista na figura 19.

FIGURA 19 – Cadastro das subcaracterísticas

Após o cadastro da característica e subcaracterística, procede-se ao cadastro das questões. Primeiramente cadastram-se as questões da subcaracterísticas (Figura 20). Ao fazer este cadastramento, automaticamente o programa vincula estas questões a característica, pois, internamente o programa as associa.

FIGURA 20 – Cadastro das questões

Procedido o cadastramento das questões, o usuário passa então a definir os pesos das características (figura 21). O usuário deve então selecionar que característica deseja avaliar e

definir os pesos para a mesma; fazendo isto, automaticamente aparecerá a tela referente as subcaracterísticas da característica escolhida. Conforme pode ser visto na figura 22.

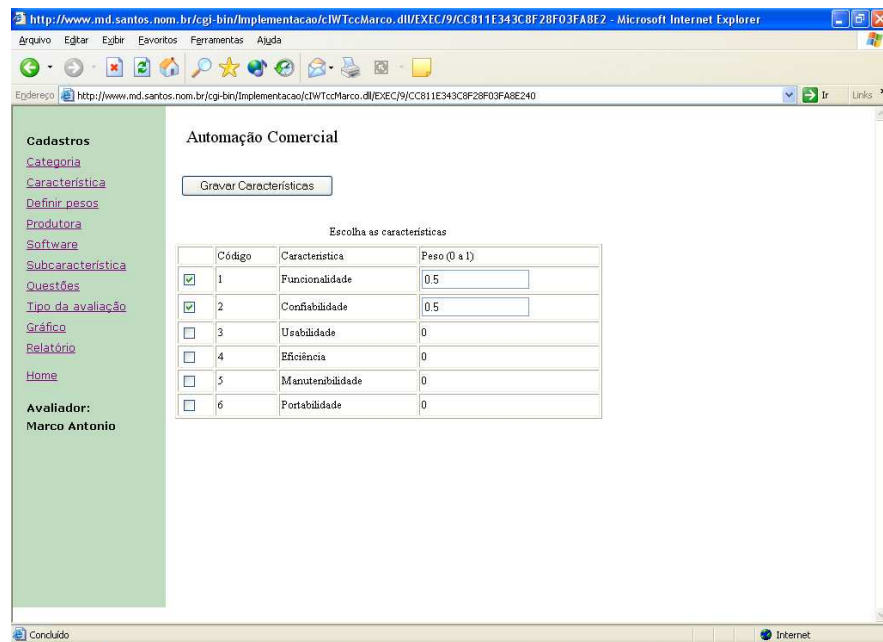


FIGURA 21 – Tela de cadastro dos pesos das características

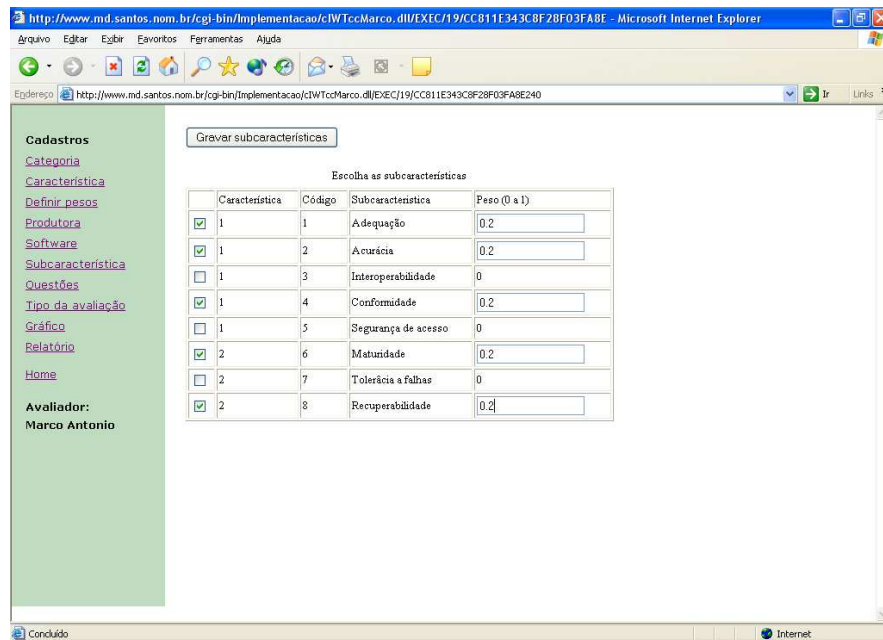


FIGURA 22 – Tela de cadastro dos pesos das subcaracterísticas

Feito o processo de cadastramento das características e subcaracterísticas a serem avaliadas no *software*, passa-se então, efetivamente, ao processo de avaliação do mesmo.

FIGURA 23 – Tela inicial da avaliação

A figura 23 apresenta a tela de avaliação, onde o usuário devidamente cadastrado poderá entrar com as informações necessárias para iniciar uma avaliação. Nesta etapa, o usuário pode optar por fazer uma avaliação completa ou parcial (tipo de avaliação); faz a seleção do *software*, previamente cadastrado, além de poder informar particularidades do mesmo como: espaço mínimo em disco exigido para sua performance, em qual sistema operacional ele roda, etc.

Na seqüência (Figura 24), o usuário responde uma série de perguntas, estando as mesmas ligadas diretamente as subcaraterísticas, as quais foram selecionadas e seus pesos definidos anteriormente

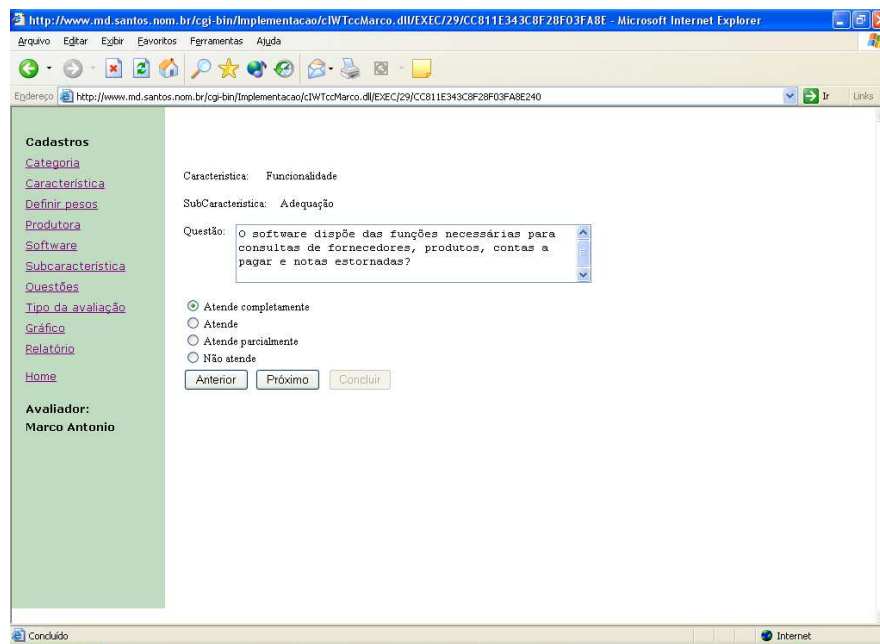


FIGURA 24 – Tela das perguntas do *check list*.

Após responder a todas as perguntas, ou seja, sempre que uma avaliação for concluída, é possível gerar um relatório para saber a pontuação obtida em determinada avaliação dos *softwares* já avaliados. Para obtermos o mesmo, selecionamos o *software*, na seqüência selecionamos a faixa em que se encontra a resposta, por exemplo: resultados abaixo de “atende completamente, atende, atende parcialmente, não atende”, e pedimos para gerar o relatório. Teremos então as perguntas com sua respectiva pontuação. Este relatório é importante quando uma pessoa deseja saber onde o *software* teve o pior desempenho.

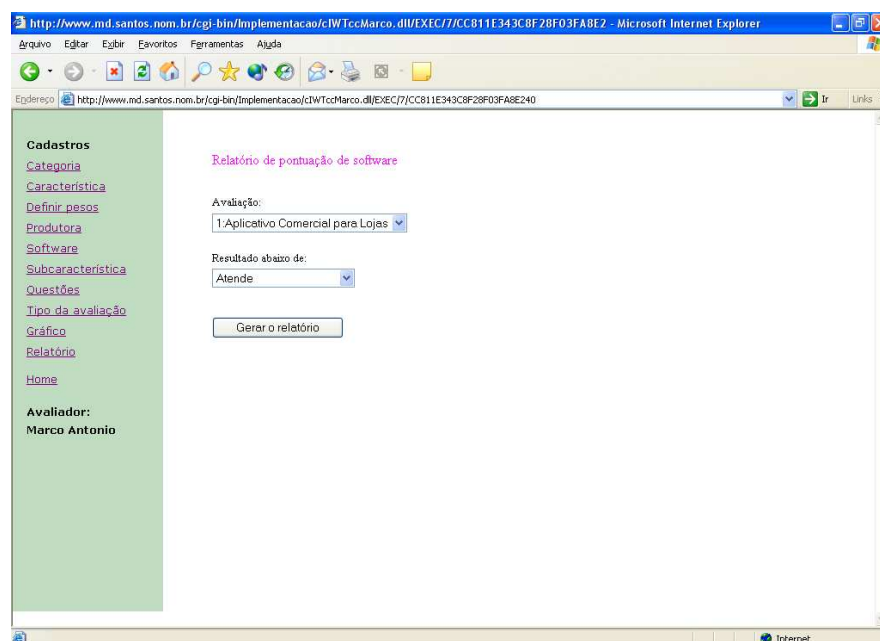


FIGURA 25 – Tela do relatório de pontuação do *software*

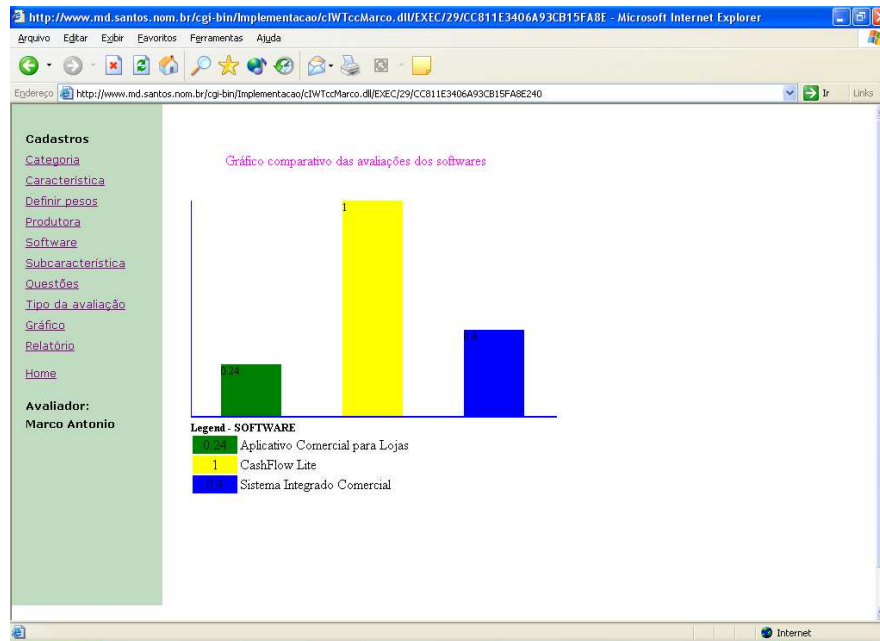


FIGURA 26 – Gráfico comparativo de desempenho

Além de gerar um relatório, é possível obter-se um gráfico (Figura 26) que apresenta o desempenho dos *softwares* avaliados. Este gráfico é dinâmico, e basta apenas clicar no *link* correspondente para obter o mesmo. Caso um determinado *software* seja avaliado mais de uma vez, este aparecerá no gráfico com o valor médio das avaliações.



## 6 CONCLUSÕES

Durante o desenvolvimento deste trabalho foram apresentados os principais conceitos e características das normas ISO/IEC 9126, avaliação de produto de *software*, e NBR ISO/IEC 12119, pacotes de *softwares*. Pode-se dizer que o objetivo principal foi atingido, uma vez que se tratava da avaliação da qualidade de sistemas de automação comercial baseado nestas normas.

Com uma interface de fácil entendimento, o *software* disponibiliza ao usuário diversas telas, onde já se encontram cadastrados itens para avaliar a qualidade de sistemas de automação comercial baseados nas normas objeto de estudo deste trabalho, e é possível incluir itens que satisfaçam seus interesses quanto a esta qualidade, podendo atender necessidades específicas para uma pessoa interessada em adquirir um determinado *software*. Esta mesma pessoa pode deixar cadastradas informações onde um desenvolvedor poderá consultar o que precisa atender para que seu *software* seja aceito pelo cliente, tendo em vista que muitas vezes os termos técnicos dificultam o entendimento entre comprador e desenvolvedor.

Uma característica deste *software* é que o usuário pode optar por avaliar os itens que satisfaçam suas necessidades, como por exemplo, apenas algumas características ou algumas subcaracterísticas das características que escolher.

Embora existam trabalhos desenvolvidos, onde foram estudadas as normas vistas no presente trabalho, juntas ou separadamente, ver quadro 1, em nenhum deles houve o direcionamento específico para área de automação comercial abordando as duas normas. Desta forma o *software* desenvolvido neste trabalho é de grande valia para aqueles que necessitam avaliar *software* específico desta área.

### 6.1 EXTENSÕES

Como extensões deste trabalho, sugerem-se:

- a) Estudo aprofundado da área de automação comercial, onde se possa envolver o desenvolvedor e o usuário para minimizar a dificuldade de diálogo entre ambos, devido a termos técnicos que “povoam” o vocabulário das pessoas da área de desenvolvimento de *software*;

- b) Desenvolvimento de um *check list* que crie um elo mais consistente entre ambos;
- c) A efetiva parametrização da fórmula, pelo usuário, onde o mesmo teria a sua disposição uma fórmula prefixada, e caso optasse por outra forma de cálculo ele poderia com um simples arrastar/soltar montar outra fórmula com o *software* fazendo as devidas consistências.

## REFERÊNCIAS BIBLIOGRÁFICAS

ARAÚJO JUNIOR, José R. Ferreira de, **Intraweb - Uma ferramenta revolucionária para desenvolvimento RAD para WEB**. Ribeirão Preto, 2003. disponível em: <<http://www.activedelphi.com.br/modules.php>>. Acesso em 11 de maio de 2004.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 13596**: tecnologia de informação – avaliação de produto de *software* – características de qualidade e diretrizes para seu uso. Rio de Janeiro, 1996.

\_\_\_\_\_. **NBR ISO/IEC 12119**: tecnologia de informação – pacotes de *software* – teste e requisitos de qualidade. Rio de Janeiro, 1998.

BARRETO JUNIOR, J. **Qualidade de pacotes de *software* – ISO 12119**. Santa Maria, [1997?]. Disponível em: <<http://www.inf.ufsm.br/~oliveira/elc311/qualidadeSW.html#Pacotes>>. Acesso em: 28 abr. 2004.

CARTILHA de automação comercial. Piracicaba.Net – 2001. Disponível em <<http://www.abscard.com.br/cartilha.htm>>. Acesso em 28 abr. 2004.

CÔRTEZ, Mário L. CHIOSSI, Thelma C. dos Santos. **Modelos de qualidade de *software***. Campinas: Unicamp, 2001.

FALCONI, Vicente C. **TQC – Controle de qualidade total (no estilo japonês)**. Belo Horizonte: Fundação Christiano Ottoni, 1992.

FERNANDES, Aginaldo A. **Gerência de *software* através de métricas**: garantindo a qualidade do projeto, processo e produto. São Paulo: Atlas, 1995.

FERREIRA, Fabiano S. **Ferramenta para avaliação da qualidade de produtos de *software* baseado nas normas ISO/IEC 9126 e NBR ISO/IEC 12119**. 2003. 103 p. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

GUESSER, Enezilda. **Protótipo de sistema para auxiliar a avaliação da qualidade de sistemas de automação comercial**. 1995. 116 p. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

GUARASEMINI, Rafael E. **Intraweb**. [s.l.], [2003?]. disponível em: <<http://www.delphibr.com.br/artigos/iw1.php>>. acesso em 20 de maio 2004.

HERBERT, J.S. & PRICE, A.M. de A. **Métodos de avaliação da qualidade de *software***. JAI – XVI Jornada de Atualização em Informática. Canela: Ago1995.

INTHURN, Cândida. **Qualidade & teste de *software***. Florianópolis: Visual Books, 2001.

PBQP, 1990 **apud** WEBER, K. C.; ROCHA, Ana R. C. da. **Qualidade e produtividade em *software***. 3. ed. São Paulo: Makron Books, 1999.

ROCHA, Ana R. C. **Análise e projeto estruturado de sistemas**. Rio de Janeiro: Campus, 1990.

ROMANO, Regiane Relva. **Conceitos de automação comercial**. Disponível em: <[http://www.pegar.com.br/pegar\\_tutor\\_gerais.asp?link\\_id=6070](http://www.pegar.com.br/pegar_tutor_gerais.asp?link_id=6070)>. Acesso em 28 de abr. 2004

SANDRI, Vivian. **Software de apoio a avaliação da qualidade de pacotes baseados na norma ISO/IEC 12119**. 1998. 60 f. Estágio supervisionado submetido a Universidade Regional de Blumenau para obtenção dos créditos de disciplina com nome equivalente do curso de Ciências da Computação – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

STORCH, Mirian M. **Proposta de avaliação da qualidade de produtos de *software* utilizando a norma ISO/IEC 9126**. 2000. 82 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

TSUKUMO, Alfredo N. et al. **Avaliação incremental de qualidade de produto de *software* baseada na ISO/IEC 9126 (NBR 13596)**. In: X Simpósio Brasileiro de Engenharia de *Software* – Workshop de Qualidade e Produtividade de *Software*, 1996, São Carlos: Anais... São Carlos: [s.n.], 1996. p. 1-5.

WEBER, K. C.; ROCHA, Ana R. C. da. **Qualidade e produtividade em *software***. 3. ed. São Paulo: Makron Books, 1999.

## APÊNDICE A – avaliação de três *softwares*, com base na ferramenta desenvolvida.

Obs.: As avaliações foram feitas em nível de estudo de caso, utilizando o cálculo através da média ponderada, podendo também, como uma segunda opção, ser feita utilizando a média simples.

### Avaliação do APLICATIVO COMERCIAL PARA LOJAS v2.00 da Empresa Horner Informática Ltda. Mogi das Cruzes/SP

#### FUNCIONALIDADE

Refere-se a existência de um conjunto de funções que satisfazem necessidades explícitas ou implícitas e suas propriedades específicas.

**Adequação:** atributos do *software* que evidenciam a presença de um conjunto de funções e sua apropriação para as tarefas especificadas.

##### ➤ Contas a pagar

O *software* dispõe das funções necessárias para consultas de fornecedores, produtos, contas a pagar e notas estornadas?

Atende completamente [ ] Atende [ ] Atende parcialmente [x] Não atende [ ]

o *software* permite a exclusão de fornecedores desativados, faturas devolvidas e recebimentos estornados?

Atende completamente [ ] Atende [ ] Atende parcialmente [ ] Não atende [x]

O *software* permite que valores como transportes e outras taxas sejam agregados para formar preço de venda?

Atende completamente [ ] Atende [ ] Atende parcialmente [ ] Não atende [x]

##### ➤ Contas a receber

O *software* permite incluir duplicatas manualmente e seus respectivos desdobramentos?

Atende completamente [ ] Atende [ ] Atende parcialmente [x] Não atende [ ]

O *software* permite consultar duplicatas por vencimento, em atraso até a data da consulta ou todas as duplicatas em aberto de um determinado cliente?

Atende completamente [ ] Atende [ ] Atende parcialmente [x] Não atende [ ]

O *software* permite consultar pedidos em aberto e/ou entregues, produtos com maior ou menor giro e tabela de preço durante uma operação de venda?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

➤ **Estoque**

O *software* permite cadastrar quantidade mínima e etiquetas de código de barras?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

O *software* permite consultar documentos de devolução, de transferência e de venda direta por produto?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

O *software* permite listar produtos com a data de última venda e ou compra?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

➤ **Mala direta**

O *software* permite selecionar clientes por data de última compra, aniversário?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

O *software* permite editar correspondência utilizando os campos dos arquivos?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

**Interoperabilidade:** atributos do *software* que evidenciam sua capacidade de interagir com sistemas especificados.

➤ **Capacidade de integração**

O *software* permite que seja integrado a outro sistema sem que haja necessidade de adaptações?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

O *software* permite a integração com hardware, como por exemplo, scanner, canetas óticas, impressoras de cheques e coletores de dados?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

**Conformidade:** atributos do *software* que fazem com que ele esteja de acordo com as normas, convenções ou regulamentações previstas em leis e descrições similares, relacionadas à aplicação.

➤ **Obediência à legislação**

O *software* efetua os cálculos de ICMS, ISS e outros impostos com precisão?

Atende completamente  Atende  Atende parcialmente  Não atende

O *software* destaca corretamente os impostos quando prepara os relatórios de livro caixa?

Atende completamente  Atende  Atende parcialmente  Não atende

**Segurança de acesso:** atributos do *software* que evidenciam sua capacidade de evitar o acesso não autorizado, acidental ou deliberado, a programas e dados.

➤ **Backups**

O *software* manda mensagem lembrando da necessidade de *backup* e faz a compactação deste?

Atende completamente  Atende  Atende parcialmente  Não atende

➤ **Controle de acesso**

O *software* é protegido por senha, e há diferenciação de senhas para acessar os diferentes módulos?

Atende completamente  Atende  Atende parcialmente  Não atende

## CONFIABILIDADE

O sistema, compreendendo *hardware* e *software*, bem como os programas que pertencem ao produto, não devem entrar em um estado no qual o usuário não consiga controlá-lo, nem deve corromper ou perder dados.

**Maturidade:** atributos do *software* que evidenciam a frequência de falhas por defeitos no *software*.

➤ **Precisão**

O *software* efetua calculo de impostos, baixa e entrada no estoque e calcula os preços dos produtos corretamente?

Atende completamente  Atende  Atende parcialmente  Não atende

➤ **integridade de dados**

O *software* possui controle para exclusão de registros, informando as conseqüências sobre os diversos módulos?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

**Tolerância a falhas:** atributos do *software* que evidenciam sua capacidade em manter um nível de desempenho especificados nos casos de falhas no *software* ou de violação nas interfaces especificadas.

➤ **Proteção contra erros**

O *software* possui consistência de campos controlando dados que fogem da média, e evita cadastro de dados já existentes?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

➤ **Proteção contra situações limites**

O *software* volta ao estado anterior após uma parada anormal, e informa ao usuário a situação dos dados?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

O *software* tem controle de situação limite, como por exemplo, número de usuários suportados simultaneamente por ele?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

**Recuperabilidade:** atributos do *software* que evidenciam sua capacidade de restabelecer seu nível de desempenho e recuperar os dados diretamente afetados, em caso de falha, e no tempo e esforço necessário para tal.

➤ **Recuperação de erros do usuário**

O sistema permite desfazer uma operação para recuperar dados por erro do usuário?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

➤ **Recuperação em caso de falhas no sistema**

O *software* não perde informações em casos de “*run time*” e é possível continuar utilizando o mesmo?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ x]



## USABILIDADE

Esforço necessário para se utilizar, operar, aprender, bem como para o julgamento individual desse uso, por um conjunto de usuários explícitos ou implícitos.

**Inteligibilidade:** atributos do *software* que evidenciam o esforço do usuário para reconhecer o conceito lógico e sua aplicabilidade.

O *software* possui tutorial *on-line*, empresa demonstração e tutorial impresso?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

**Apreensibilidade:** atributos do *software* que evidenciam o esforço do usuário para compreender sua aplicação (por exemplo: controle de operação, entradas, saídas).

O *help* apresenta uma explicação para todos os itens relacionados com a operação do sistema e os termos utilizados possuem o mesmo significado durante todo o *help*?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

**Operacionalidade:** atributos do *software* que evidenciam o esforço do usuário para sua operação e controle da sua operação.

O *software* informa ao usuário as opções disponíveis e como deve proceder para executar a tarefa em questão?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

## EFICIÊNCIA

A descrição de produto pode incluir dados sobre o comportamento do produto em relação ao tempo, como por exemplo tempo de resposta.

**Comportamento em relação ao tempo:** atributos do *software* que evidenciam seu tempo de resposta, tempo de processamento e velocidade na execução de suas funções.

O *software* faz reorganização da base de dados constantemente?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

O *software* tem seu desempenho comprometido quando trabalha com grandes volumes de dados?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

**Comportamento em relação aos recursos:** atributos do *software* que evidenciam a quantidade de recursos usados e a duração de seu uso na execução de suas funções.

O *software* exige pouco recurso de máquina, e faz pouco acesso a disco?

**Atende completamente** [ ] **Atende** [x] **Atende parcialmente** [ ] **Não atende** [ ]

## MANUTENIBILIDADE

Série de atributos que suportam um esforço necessário para fazer modificações específicas.

**Analisabilidade:** atributos do *software* que evidenciam o esforço necessário para diagnosticar deficiências ou causas de falhas, ou para identificar partes a serem modificadas.

O *software* utiliza um padrão para nomes identificadores e estes são significativos e concisos?

**Atende completamente** [ ] **Atende** [x] **Atende parcialmente** [ ] **Não atende** [ ]

**Modificabilidade:** atributos do *software* que evidenciam o esforço necessário para modificá-lo, remover seus defeitos inesperados, ocasionados por modificações.

O *software* permite incluir ou excluir campos e aumentar o número de casas decimais?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

**Estabilidade:** atributos do *software* que evidenciam o risco de defeitos inesperados, ocasionados por modificações.

O *software* possui estabilidade (não apresentando necessidade de manutenções freqüentes)?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

**Testabilidade:** atributos do *software* que evidenciam o esforço necessário para validar o *software* modificado.

O *software* possui uma base de demonstração para testes e quando este é modificado são adicionados casos de testes para validar as modificações?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

## PORTABILIDADE

Série de atributos que suportam a habilidade do sistema ser transferido de um ambiente para outro qualquer. E, ser operado e manipulado de maneira fácil e adequada em configurações de equipamentos diferentes da original.

**Adaptabilidade:** atributos do *software* que evidenciam sua capacidade de ser adaptado a ambientes deferentes especificados, sem a necessidade de aplicação de outras ações ou meios além daqueles fornecidos para essa finalidade pelo *software* considerado.

O *software* pode ser facilmente modificado para atender as necessidades dos usuários, como por exemplo, alterações em legislação, ampliação da empresa ou abertura de filial?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

**Capacidade de ser instalado:** atributos do *software* que evidenciam o esforço necessário para sua instalação em um ambiente especificado.

O *software* possui comandos intuitivos durante a instalação e faz modificações de *config* e *autoexec* automaticamente?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

**Conformidade:** atributos do *software* que o tornam consonante com padrões ou convenções relacionados à portabilidade.

O *software* é adequado ao fim que foi destinado?

**Atende completamente** [x] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

**Capacidade de substituir:** atributos do *software* que evidenciam sua capacidade e esforço necessário para substituir um outro *software*, no ambiente estabelecido para este outro *software*.

O *software* é capaz de substituir um outro e possui a capacidade para ser instalado e adaptabilidade idêntica à deste?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

**Avaliação do APLICATIVO COMERCIAL PARA LOJAS v2.00 da Empresa  
Horner Informática Ltda. Mogi das Cruzes/SP**

Para calcular a qualidade do *software* desta avaliação, devemos primeiramente calcular a média ponderada de cada subcaracterística, terminado o cálculo das subcaracterísticas de uma determinada característica, calcula-se a média ponderada da característica. Ao final calculamos a qualidade do *software* com base nos cálculos das características a saber:

Para calcularmos a média ponderada das subcaracterísticas vamos utilizar a seguinte fórmula:

$$VS = ((PPg_1 * Ps) + (PPg_2 * Ps) + \dots + (PPg_n * Ps)) / \sum Ps$$

Onde **VS** é Valor da Subcaracterística; **PPg** é a Pontuação da Pergunta em questão (cada pergunta desta característica) e **Ps** é o Peso da subcaracterística.

Característica: **FUNCIONALIDADE** (peso 0,3)

Subcaracterística: **Adequação** (peso 0,3)

$$VS = ((1 * 0,3) + (0 * 0,3) + (0 * 0,3) + (1 * 0,3) + (1 * 0,3) + (1 * 0,3) + (0 * 0,3) + (0 * 0,3) + (0 * 0,3) + (1 * 0,3) + (1 * 0,3)) / (0,3 + 0,3 + 0,3 + 0,3 + 0,3 + 0,3 + 0,3 + 0,3 + 0,3 + 0,3 + 0,3)$$

$$VS = 0,55$$

Subcaracterística: **Interoperabilidade** (peso 0,2)

$$VS = ((0 * 0,2) + (0 * 0,2)) / (0,2 + 0,2)$$

$$VS = 0,0$$

Subcaracterística: **Conformidade** (peso 0,2)

$$VS = ((3 * 0,2) + (2 * 0,2)) / (0,2 + 0,2)$$

$$VS = 2,5$$

Subcaracterística: **Segurança de acesso** (peso 0,3)

$$VS = ((0 * 0,3) + (1 * 0,3)) / (0,3 + 0,3)$$

$$VS = 0,5$$

Após obtermos os valores das subcaracterísticas, devemos calcular o valor da característica, com a seguinte fórmula:

$$VC = ((VS_1 * PsC) + (VS_2 * PsC) + (VS_3 * PsC) + (VS_4 * PsC)) / \sum PsC$$

Onde **VC** é o Valor da Característica; **VS** é o Valor da Subcaracterística e **PsC** é o valor do Peso da Característica.

$$VC = ((0,55 * 0,3) + (0 * 0,3) + (2,5 * 0,3) + (0,5 * 0,3)) / (0,3 + 0,3 + 0,3 + 0,3)$$

$$VC = 0,89$$

Característica: **CONFIABILIDADE** (peso 0,1)

Subcaracterística: **Maturidade** (peso 0,4)

$$VS = ((1 * 0,4) + (0 * 0,4)) / (0,4 + 0,4)$$

$$VS = 0,5$$

Subcaracterística: **Tolerância a falhas** (peso 0,2)

$$VS = ((0 * 0,2) + (0 * 0,2) + (0 * 0,2)) / (0,2 + 0,2 + 0,2)$$

$$VS = 0$$

Subcaracterística: **Recuperabilidade** (peso 0,4)

$$VS = ((0 * 0,4) + (0 * 0,4)) / (0,4 + 0,4)$$

$$VS = 0$$

Calculando o valor da característica **confiabilidade**:

$$VC = ((VS_1 * PsC) + (VS_2 * PsC) + (VS_3 * PsC)) / \Sigma PsC$$

$$VC = ((0,5 * 0,1) + (0 * 0,1) + (0 * 0,1)) / (0,1 + 0,1 + 0,1)$$

$$VC = 0,17$$

Característica: **USABILIDADE** peso 0,3

Subcaracterística: **Inteligibilidade** (peso 0,3)

$$VS = (0 * 0,3) / (0,3)$$

$$VS = 0$$

Subcaracterística: **Apreensibilidade** (peso 0,4)

$$VS = (0 * 0,4) / (0,4)$$

$$VS = 0$$

Subcaracterística: **Operacionalidade** (peso 0,3)

$$VS = (0 * 0,3) / (0,3)$$

$$VS = 0$$

Calculando o valor da característica **usabilidade**:

$$VC = ((VS_1 * PsC) + (VS_2 * PsC) + (VS_3 * PsC)) / \Sigma PsC$$

$$VC = ((0 * 0,3) + (0 * 0,3) + (0 * 0,3)) / (0,3 + 0,3 + 0,3)$$

$$VC = 0$$

Característica: **EFICIÊNCIA** (peso 0,1)

Subcaracterística: **Comportamento em relação ao tempo** (peso 0,6)

$$VS = ((1 * 0,6) + (0 * 0,6)) / (0,6 + 0,6)$$

$$VS = 0,5$$

Subcaracterística: **Comportamento em relação aos recursos** (peso 0,4)

$$VS = (2 * 0,4) / (0,4)$$

$$VS = 2$$

Calculando o valor da característica eficiência:

$$VC = ((VS_1 * PsC) + (VS_2 * PsC)) / \Sigma PsC$$

$$VC = ((0,5 * 0,1) + (2 * 0,1)) / (0,1 + 0,1)$$

$$VC = 1,25$$

Característica: **MANUTENIBILIDADE** (peso 0,1)

Subcaracterística: **Analisabilidade** (peso 0,3)

$$VS = (2 * 0,3) / (0,3)$$

$$VS = 2$$

Subcaracterística: **Modificabilidade** (peso 0,2)

$$VS = (0 * 0,2) / (0,2)$$

$$VS = 0$$

Subcaracterística: **Estabilidade** (peso 0,3)

$$VS = (1 * 0,3) / (0,3)$$

$$VS = 1$$

Subcaracterística: **Testabilidade** (peso 0,2)

$$VS = (0 * 0,2) / (0,2)$$

$$VS = 0$$

Calculando o valor da característica **manutenibilidade**:

$$VC = ((VS_1 * PsC) + (VS_2 * PsC) + (VS_3 * PsC) + (VS_4 * PsC)) / \Sigma PsC$$

$$VC = ((2 * 0,1) + (0 * 0,1) + (1 * 0,1) + (0 * 0,1)) / (0,1 + 0,1 + 0,1 + 0,1)$$

$$VC = 0,75$$

Característica: **PORTABILIDADE** (peso 0,1)

Subcaracterística: **Adaptabilidade** (peso 0,3)

$$VS = (0 * 0,3) / (0,3)$$

$$VS = 0$$

Subcaracterística: Capacidade de ser instalado (peso 0,2)

$$VS = (1 * 0,2) / (0,2)$$

$$VS = 1$$

Subcaracterística: Conformidade (peso 0,3)

$$VS = (3 * 0,2) / (0,2)$$

$$VS = 3$$

Subcaracterística: Capacidade de substituir (peso 0,2)

$$VS = (0 * 0,2) / (0,2)$$

$$VS = 0$$

Calculando o valor da característica portabilidade:

$$VC = ((VS_1 * PsC) + (VS_2 * PsC) + (VS_3 * PsC) + (VS_4 * PsC)) / \Sigma PsC$$

$$VC = ((0 * 0,1) + (1 * 0,1) + (3 * 0,1) + (0 * 0,1)) / (0,1 + 0,1 + 0,1 + 0,1)$$

$$VC = 1$$

Tendo calculado os valores de cada característica, devemos calcular o valor final de qualidade do *software*.

**PSo** = Pontuação do *Software*

**VPC** = Valor Parcial das Características

**PMP** = Pontuação Máxima Possível

$$PSo = (VC1 + VC2 + VC3 + VC4 + VC5 + VC6) / PMP$$

$$PSo = (0,89 + 0,17 + 0 + 1,25 + 0,75 + 1) / 18$$

$$PSo = 0,23 \text{ ou } 23\%$$

O *software* atingiu 23% de qualidade, sendo considerado um *software* **insuficiente**, segundo a ISO/IEC 9126.

<b>0</b>	<b>a</b>	<b>25%</b>	<b>=</b>	<b>Insuficiente</b>
<b>25,1</b>	<b>a</b>	<b>50%</b>	<b>=</b>	<b>Regular</b>
<b>50,1</b>	<b>a</b>	<b>75%</b>	<b>=</b>	<b>Bom</b>
<b>75,1</b>	<b>a</b>	<b>100%</b>	<b>=</b>	<b>Excelente</b>

**Avaliação do *software* CASHFLOW LITE v6.02 da Empresa DBI Micro-Sistemas  
- São Paulo/SP**

**FUNCIONALIDADE**

Refere-se a existência de um conjunto de funções que satisfazem necessidades explícitas ou implícitas e suas propriedades específicas.

**Adequação:** atributos do *software* que evidenciam a presença de um conjunto de funções e sua apropriação para as tarefas especificadas.

➤ **Contas a pagar**

O *software* dispõe das funções necessárias para consultas de fornecedores, produtos, contas a pagar e notas estornadas?

**Atende completamente** [ ] **Atende** [x] **Atende parcialmente** [ ] **Não atende** [ ]

O *software* permite a exclusão de fornecedores desativados, faturas devolvidas e recebimentos estornados?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

O *software* permite que valores como transportes e outras taxas sejam agregados para formar preço de venda?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

➤ **Contas a receber**

O *software* permite incluir duplicatas manualmente e seus respectivos desdobramentos?

**Atende completamente** [x] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

O *software* permite consultar duplicatas por vencimento, em atraso até a data da consulta ou todas as duplicatas em aberto de um determinado cliente?

**Atende completamente** [x] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

O *software* permite consultar pedidos em aberto e/ou entregues, produtos com maior ou menor giro e tabela de preço durante uma operação de venda?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

➤ **Estoque**

O *software* permite cadastrar quantidade mínima e etiquetas de código de barras?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]



O *software* permite consultar documentos de devolução, de transferência e de venda direta por produto?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

O *software* permite listar produtos com a data de última venda e ou compra?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

➤ **Mala direta**

O *software* permite selecionar clientes por data de última compra ou aniversário?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

O *software* permite editar correspondência utilizando os campos dos arquivos?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

**Interoperabilidade:** atributos do *software* que evidenciam sua capacidade de interagir com sistemas especificados.

➤ **Capacidade de integração**

O *software* permite que seja integrado a outro sistema sem que haja necessidade de adaptações?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

O *software* permite a integração com *hardware*, como por exemplo, scanner, canetas óticas, impressoras de cheques e coletores de dados?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

**Conformidade:** atributos do *software* que fazem com que ele esteja de acordo com as normas, convenções ou regulamentações previstas em leis e descrições similares, relacionadas à aplicação.

➤ **Obediência à legislação**

O *software* efetua os cálculos de ICMS, ISS e outros impostos com precisão?

**Atende completamente** [x] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

O *software* destaca corretamente os impostos quando prepara os relatórios de livro caixa?

**Atende completamente** [ ] **Atende** [x] **Atende parcialmente** [ ] **Não atende** [ ]

**Segurança de acesso:** atributos do *software* que evidenciam sua capacidade de evitar o acesso não autorizado, acidental ou deliberado, a programas e dados.

➤ **Backups**

O *software* manda mensagem lembrando da necessidade de *backup* e faz a compactação deste?

Atende completamente [ ] Atende [x] Atende parcialmente [ ] Não atende [ ]

➤ **Controle de acesso**

O *software* é protegido por senha, e há diferenciação de senhas para acessar os diferentes módulos?

Atende completamente [x] Atende [ ] Atende parcialmente [ ] Não atende [ ]

## CONFIABILIDADE

O sistema, compreendendo *hardware* e *software*, bem como os programas que pertencem ao produto, não devem entrar em um estado no qual o usuário não consiga controlá-lo, nem deve corromper ou perder dados.

**Maturidade:** atributos do *software* que evidenciam a frequência de falhas por defeitos no *software*.

➤ **Precisão**

O *software* efetua cálculo de impostos, baixa e entrada no estoque e calcula os preços dos produtos corretamente?

Atende completamente [x] Atende [ ] Atende parcialmente [ ] Não atende [ ]

➤ **integridade de dados**

O *software* possui controle para exclusão de registros, informando as conseqüências sobre os diversos módulos?

Atende completamente [ ] Atende [x] Atende parcialmente [ ] Não atende [ ]

**Tolerância a falhas:** atributos do *software* que evidenciam sua capacidade em manter um nível de desempenho especificados nos casos de falhas no *software* ou de violação nas interfaces especificadas.

➤ **Proteção contra erros**

O *software* possui consistência de campos controlando dados que fogem da média, e evita cadastro de dados já existentes?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

➤ **Proteção contra situações limites**

O *software* volta ao estado anterior após uma parada anormal, e informa ao usuário a situação dos dados?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

O *software* tem controle de situação limite, como por exemplo, número de usuários suportados simultaneamente por ele?

**Atende completamente** [x] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

**Recuperabilidade:** atributos do *software* que evidenciam sua capacidade de restabelecer seu nível de desempenho e recuperar os dados diretamente afetados, em caso de falha, e no tempo e esforço necessário para tal.

➤ **Recuperação de erros do usuário**

O sistema permite desfazer uma operação para recuperar dados por erro do usuário?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

➤ **Recuperação em caso de falhas no sistema**

O *software* não perde informações em casos de “*run time*” e é possível continuar utilizando o mesmo?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

## USABILIDADE

Esforço necessário para se utilizar, operar, aprender, bem como para o julgamento individual desse uso, por um conjunto de usuários explícitos ou implícitos.

**Inteligibilidade:** atributos do *software* que evidenciam o esforço do usuário para reconhecer o conceito lógico e sua aplicabilidade.

O *software* possui tutorial *on-line*, empresa demonstração e tutorial impresso?

**Atende completamente** [ ] **Atende** [x] **Atende parcialmente** [ ] **Não atende** [ ]

**Apreensibilidade:** atributos do *software* que evidenciam o esforço do usuário para compreender sua aplicação (por exemplo: controle de operação, entradas, saídas).

O *help* apresenta uma explicação para todos os itens relacionados com a operação do sistema e os termos utilizados possuem o mesmo significado durante todo o *help*?

**Atende completamente**  **Atende**  **Atende parcialmente**  **Não atende**

**Operacionalidade:** atributos do *software* que evidenciam o esforço do usuário para sua operação e controle da sua operação.

O *software* informa ao usuário as opções disponíveis e como deve proceder para executar a tarefa em questão?

**Atende completamente**  **Atende**  **Atende parcialmente**  **Não atende**

## EFICIÊNCIA

A descrição de produto pode incluir dados sobre o comportamento do produto em relação ao tempo, como por exemplo tempo de resposta.

**Comportamento em relação ao tempo:** atributos do *software* que evidenciam seu tempo de resposta, tempo de processamento e velocidade na execução de suas funções.

O *software* faz reorganização da base de dados constantemente?

**Atende completamente**  **Atende**  **Atende parcialmente**  **Não atende**

O *software* tem seu desempenho comprometido quando trabalha com grandes volumes de dados?

**Atende completamente**  **Atende**  **Atende parcialmente**  **Não atende**

**Comportamento em relação aos recursos:** atributos do *software* que evidenciam a quantidade de recursos usados e a duração de seu uso na execução de suas funções.

O *software* exige pouco recurso de máquina, e faz pouco acesso a disco?

**Atende completamente**  **Atende**  **Atende parcialmente**  **Não atende**

## MANUTENIBILIDADE

Série de *atributos* que suportam um esforço necessário para fazer modificações específicas.

**Analisabilidade:** atributos do *software* que evidenciam o esforço necessário para diagnosticar deficiências ou causas de falhas, ou para identificar partes a serem modificadas.

O *software* utiliza um padrão para nomes identificadores e estes são significativos e concisos?

**Atende completamente**  **Atende**  **Atende parcialmente**  **Não atende**

**Modificabilidade:** atributos do *software* que evidenciam o esforço necessário para modificá-lo, remover seus defeitos inesperados, ocasionados por modificações.

O *software* permite incluir ou excluir campos e aumentar o número de casas decimais?

**Atende completamente**  **Atende**  **Atende parcialmente**  **Não atende**

**Estabilidade:** atributos do *software* que evidenciam o risco de defeitos inesperados, ocasionados por modificações.

O *software* possui estabilidade (não apresentando necessidade de manutenções freqüentes)?

**Atende completamente**  **Atende**  **Atende parcialmente**  **Não atende**

**Testabilidade:** atributos do *software* que evidenciam o esforço necessário para validar o *software* modificado.

O *software* possui uma base de demonstração para testes e quando este é modificado são adicionados casos de testes para validar as modificações?

**Atende completamente**  **Atende**  **Atende parcialmente**  **Não atende**

## PORTABILIDADE

Série de atributos que suportam a habilidade do sistema ser transferido de um ambiente para outro qualquer. E, ser operado e manipulado de maneira fácil e adequada em configurações de equipamentos diferentes da original.

**Adaptabilidade:** atributos do *software* que evidenciam sua capacidade de ser adaptado a ambientes deferentes especificados, sem a necessidades de aplicação de outras ações ou meios além daqueles fornecidos para essa finalidade pelo *software* considerado.

O *software* pode ser facilmente modificado para atender as necessidades dos usuários, como por exemplo, alterações em legislação, ampliação da empresa ou abertura de filial?

**Atende completamente** [ ] **Atende** [x] **Atende parcialmente** [ ] **Não atende** [ ]

**Capacidade de ser instalado:** atributos do *software* que evidenciam o esforço necessário para sua instalação em um ambiente especificado.

O *software* possui comandos intuitivos durante a instalação e faz modificações de *config* e *autoexec* automaticamente?

**Atende completamente** [x] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

**Conformidade:** atributos do *software* que o tornam consonante com padrões ou convenções relacionados à portabilidade.

O *software* é adequado ao fim que foi destinado?

**Atende completamente** [x] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

**Capacidade de substituir:** atributos do *software* que evidenciam sua capacidade e esforço necessário para substituir um outro *software*, no ambiente estabelecido para este outro *software*.

O *software* é capaz de substituir um outro? Possui a capacidade para ser instalado e adaptabilidade idênticas à deste?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

**Avaliação do *software* CASHFLOW LITE v6.02 da Empresa DBI Micro-Sistemas  
- São Paulo/SP**

Para calcularmos a qualidade do *software* desta avaliação, devemos primeiramente calcular a média ponderada de cada subcaracterística, terminado o calculo das subcaracterísticas de uma determinada característica, calculamos a média ponderada da característica. Ao final calculamos a qualidade do *software* com base nos cálculos das características a saber:

Para calcularmos a média ponderada das subcaracterísticas vamos utilizar a seguinte fórmula:

$$VS = ((PPg_1 * Ps) + (PPg_2 * Ps) + \dots + (PPg_n * Ps)) / \Sigma Ps$$

Onde **VS** é Valor da Subcaracterística; **PPg** é a Pontuação da Pergunta em questão (cada pergunta desta característica) e **Ps** é o Peso da subcaracterística.

Característica: **FUNCIONALIDADE** (peso 0,3)

Subcaracterística: **Adequação** (peso 0,3)

$$VS = ((2 * 0,3) + (1 * 0,3) + (0 * 0,3) + (3 * 0,3) + (3 * 0,3) + (1 * 0,3) + (0 * 0,3) + (1 * 0,3) + (1 * 0,3) + (1 * 0,3) + (0 * 0,3)) / (0,3 + 0,3 + 0,3 + 0,3 + 0,3 + 0,3 + 0,3 + 0,3 + 0,3 + 0,3 + 0,3)$$

$$VS = 0,91$$

Subcaracterística: **Interoperabilidade** (peso 0,2)

$$VS = ((0 * 0,2) + (1 * 0,2)) / (0,2 + 0,2)$$

$$VS = 0,5$$

Subcaracterística: **Conformidade** (peso 0,2)

$$VS = ((3 * 0,2) + (2 * 0,2)) / (0,2 + 0,2)$$

$$VS = 2,5$$

Subcaracterística: **Segurança de acesso** (peso 0,3)

$$VS = ((2 * 0,3) + (3 * 0,3)) / (0,3 + 0,3)$$

$$VS = 2,5$$

Após obtermos os valores das subcaracterísticas, devemos calcular o valor da característica **funcionalidade**, com a seguinte fórmula:

$$VC = ((VS_1 * PsC) + (VS_2 * PsC) + (VS_3 * PsC) + (VS_4 * PsC)) / \Sigma PsC$$

Onde **VC** é o Valor da Característica; **VS** é o Valor da Subcaracterística e **PsC** é o valor do Peso da Característica.

$$VC = ((0,91 * 0,3) + (0,5 * 0,3) + (2,5 * 0,3) + (2,5 * 0,3)) / (0,3 + 0,3 + 0,3 + 0,3)$$

$$VC = 1,6$$

Característica: **CONFIABILIDADE** (peso 0,1)

Subcaracterística: **Maturidade** (peso 0,4)

$$VS = ((3 * 0,4) + (2 * 0,4)) / (0,4 + 0,4)$$

$$VS = 2,5$$

Subcaracterística: **Tolerância a falhas** (peso 0,2)

$$VS = ((1 * 0,2) + (0 * 0,2) + (3 * 0,2)) / (0,2 + 0,2 + 0,2)$$

$$VS = 1,33$$

Subcaracterística: **Recuperabilidade** (peso 0,4)

$$VS = ((1 * 0,4) + (0 * 0,4)) / (0,4 + 0,4)$$

$$VS = 0,5$$

Calculando o valor da característica **confiabilidade**:

$$VC = ((VS_1 * PsC) + (VS_2 * PsC) + (VS_3 * PsC)) / \Sigma PsC$$

$$VC = ((2,5 * 0,1) + (1,33 * 0,1) + (0,5 * 0,1)) / (0,1 + 0,1 + 0,1)$$

$$VC = 1,44$$

Característica: **USABILIDADE** peso 0,3

Subcaracterística: **Inteligibilidade** (peso 0,3)

$$VS = (2 * 0,3) / (0,3)$$

$$VS = 2$$

Subcaracterística: **Apreensibilidade** (peso 0,4)

$$VS = (3 * 0,4) / (0,4)$$

$$VS = 3$$

Subcaracterística: **Operacionalidade** (peso 0,3)

$$VS = (3 * 0,3) / (0,3)$$

$$VS = 3$$

Calculando o valor da característica **usabilidade**:

$$VC = ((VS_1 * PsC) + (VS_2 * PsC) + (VS_3 * PsC)) / \Sigma PsC$$



$$VC = ((2 * 0,3) + (3 * 0,3) + (3 * 0,3)) / (0,3 + 0,3 + 0,3)$$

$$VC = 2,67$$

Característica: **EFICIÊNCIA** (peso 0,1)

Subcaracterística: **Comportamento em relação ao tempo** (peso 0,6)

$$VS = ((2 * 0,6) + (1 * 0,6)) / (0,6 + 0,6)$$

$$VS = 1,5$$

Subcaracterística: **Comportamento em relação aos recursos** (peso 0,4)

$$VS = (3 * 0,4) / (0,4)$$

$$VS = 3$$

Calculando o valor da característica **eficiência**:

$$VC = ((VS_1 * PsC) + (VS_2 * PsC)) / \Sigma PsC$$

$$VC = ((1,5 * 0,1) + (3 * 0,1)) / (0,1 + 0,1)$$

$$VC = 2,25$$

Característica: **MANUTENIBILIDADE** (peso 0,1)

Subcaracterística: **Analisabilidade** (peso 0,3)

$$VS = (3 * 0,3) / (0,3)$$

$$VS = 3$$

Subcaracterística: **Modificabilidade** (peso 0,2)

$$VS = (0 * 0,2) / (0,2)$$

$$VS = 0$$

Subcaracterística: **Estabilidade** (peso 0,3)

$$VS = (2 * 0,3) / (0,3)$$

$$VS = 2$$

Subcaracterística: **Testabilidade** (peso 0,2)

$$VS = (1 * 0,2) / (0,2)$$

$$VS = 1$$

Calculando o valor da característica **manutenibilidade**:

$$VC = ((VS_1 * PsC) + (VS_2 * PsC) + (VS_3 * PsC) + (VS_4 * PsC)) / \Sigma PsC$$

$$VC = ((3 * 0,1) + (0 * 0,1) + (2 * 0,1) + (1 * 0,1)) / (0,1 + 0,1 + 0,1 + 0,1)$$

$$VC = 1,5$$

Característica: **PORTABILIDADE** (peso 0,1)

Subcaracterística: **Adaptabilidade** (peso 0,3)

$$VS = (2 * 0,3) / (0,3)$$

$$VS = 2$$

Subcaracterística: **Capacidade de ser instalado** (peso 0,2)

$$VS = (3 * 0,2) / (0,2)$$

$$VS = 3$$

Subcaracterística: **Conformidade** (peso 0,3)

$$VS = (3 * 0,2) / (0,2)$$

$$VS = 3$$

Subcaracterística: **Capacidade de substituir** (peso 0,2)

$$VS = (0 * 0,2) / (0,2)$$

$$VS = 0$$

Calculando o valor da característica **portabilidade**:

$$VC = ((VS_1 * PsC) + (VS_2 * PsC) + (VS_3 * PsC) + (VS_4 * PsC)) / \Sigma PsC$$

$$VC = ((2 * 0,1) + (3 * 0,1) + (3 * 0,1) + (0 * 0,1)) / (0,1 + 0,1 + 0,1 + 0,1)$$

$$VC = 2$$

Tendo calculado os valores parciais de cada característica, devemos calcular o valor final de qualidade do *software*.

**PSo** = Pontuação do *Software*

**VPC** = Valor Parcial das Características

**PMP** = Pontuação Máxima Possível

$$PSo = (VC_1 + VC_2 + VC_3 + VC_4 + VC_5 + VC_6) / PMP$$

$$PSo = (1,6 + 1,44 + 2,67 + 2,25 + 1,5 + 2) / 18$$

$$PSo = 0,64 \text{ ou } 64\%$$

O *software* atingiu 64% de qualidade, sendo considerado um *software* bom, segundo a ISO/IEC 9126.

<b>0</b>	<b>a</b>	<b>25%</b>	<b>=</b>	<b>Insuficiente</b>
<b>25,1</b>	<b>a</b>	<b>50%</b>	<b>=</b>	<b>Regular</b>
<b>50,1</b>	<b>a</b>	<b>75%</b>	<b>=</b>	<b>Bom</b>
<b>75,1</b>	<b>a</b>	<b>100%</b>	<b>=</b>	<b>Excelente</b>

**Avaliação do *software* SISTEMA INTEGRADO COMERCIAL versão 4.76 da Empresa Pic Informática Ltda. São José do Rio Preto/SP.**

**FUNCIONALIDADE**

Refere-se a existência de um conjunto de funções que satisfazem necessidades explícitas ou implícitas e suas propriedades específicas.

**Adequação:** atributos do *software* que evidenciam a presença de um conjunto de funções e sua apropriação para as tarefas especificadas.

➤ **Contas a pagar**

O *software* dispõe das funções necessárias para consultas de fornecedores, produtos, contas a pagar e notas estornadas?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

O *software* permite a exclusão de fornecedores desativados, faturas devolvidas e recebimentos estornados?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

O *software* permite que valores como transportes e outras taxas sejam agregados para formar preço de venda?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

➤ **Contas a receber**

O *software* permite incluir duplicatas manualmente e seus respectivos desdobramentos?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

O *software* permite consultar duplicatas por vencimento, em atraso até a data da consulta ou todas as duplicatas em aberto de um determinado cliente?

**Atende completamente** [ ] **Atende** [x] **Atende parcialmente** [ ] **Não atende** [ ]

O *software* permite consultar pedidos em aberto e/ou entregues, produtos com maior ou menor giro e tabela de preço durante uma operação de venda?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

➤ **Estoque**

O *software* permite cadastrar quantidade mínima e etiquetas de código de barras?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

O *software* permite consultar documentos de devolução, de transferência e de venda direta por produto?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

O *software* permite listar produtos com a data de última venda e ou compra?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

➤ **Mala direta**

O *software* permite selecionar clientes por data de última compra ou aniversário?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

O *software* permite editar correspondência utilizando os campos dos arquivos?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

**Interoperabilidade:** atributos do *software* que evidenciam sua capacidade de interagir com sistemas especificados.

➤ **Capacidade de integração**

O *software* permite que seja integrado a outro sistema sem que haja necessidade de adaptações?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

O *software* permite a integração com hardware, como por exemplo, scanner, canetas óticas, impressoras de cheques e coletores de dados?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

**Conformidade:** atributos do *software* que fazem com que ele esteja de acordo com as normas, convenções ou regulamentações previstas em leis e descrições similares, relacionadas à aplicação.

➤ **Obediência à legislação**

O *software* efetua os cálculos de ICMS, ISS e outros impostos com precisão?

**Atende completamente** [x] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

O *software* destaca corretamente os impostos quando prepara os relatórios de livro caixa?

**Atende completamente** [ ] **Atende** [x] **Atende parcialmente** [ ] **Não atende** [ ]

**Segurança de acesso:** atributos do *software* que evidenciam sua capacidade de evitar o acesso não autorizado, acidental ou deliberado, a programas e dados.

➤ **Backups**

O *software* manda mensagem lembrando da necessidade de backup e faz a compactação deste?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

➤ **Controle de acesso**

O *software* é protegido por senha, e há diferenciação de senhas para acessar os diferentes módulos?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

## CONFIABILIDADE

O sistema, compreendendo hardware e *software*, bem como os programas que pertencem ao produto, não devem entrar em um estado no qual o usuário não consiga controlá-lo, nem deve corromper ou perder dados.

**Maturidade:** atributos do *software* que evidenciam a frequência de falhas por defeitos no *software*.

➤ **Precisão**

O *software* efetua calculo de impostos, baixa e entrada no estoque e calcula os preços dos produtos corretamente?

**Atende completamente** [ ] **Atende** [x] **Atende parcialmente** [ ] **Não atende** [ ]

➤ **integridade de dados**

O *software* possui controle para exclusão de registros, informando as conseqüências sobre os diversos módulos?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

**Tolerância a falhas:** atributos do *software* que evidenciam sua capacidade em manter um nível de desempenho especificados nos casos de falhas no *software* ou de violação nas interfaces especificadas.

➤ **Proteção contra erros**

O *software* possui consistência de campos controlando dados que fogem da média, e evita cadastro de dados já existentes?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

➤ **Proteção contra situações limites**

O *software* volta ao estado anterior após uma parada anormal, e informa ao usuário a situação dos dados?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

O *software* tem controle de situação limite, como por exemplo, número de usuários suportados simultaneamente por ele?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

**Recuperabilidade:** atributos do *software* que evidenciam sua capacidade de restabelecer seu nível de desempenho e recuperar os dados diretamente afetados, em caso de falha, e no tempo e esforço necessário para tal.

➤ **Recuperação de erros do usuário**

O sistema permite desfazer uma operação para recuperar dados por erro do usuário?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

➤ **Recuperação em caso de falhas no sistema**

O *software* não perde informações em casos de “*run time*” e é possível continuar utilizando o mesmo?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

## USABILIDADE

Esforço necessário para se utilizar, operar, aprender, bem como para o julgamento individual desse uso, por um conjunto de usuários explícitos ou implícitos.

**Inteligibilidade:** atributos do *software* que evidenciam o esforço do usuário para reconhecer o conceito lógico e sua aplicabilidade.

O *software* possui tutorial *on-line*, empresa demonstração e tutorial impresso?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

**Apreensibilidade:** atributos do *software* que evidenciam o esforço do usuário para compreender sua aplicação (por exemplo: controle de operação, entradas, saídas).

O *help* apresenta uma explicação para todos os itens relacionados com a operação do sistema e os termos utilizados possuem o mesmo significado durante todo o *help*?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

**Operacionalidade:** atributos do *software* que evidenciam o esforço do usuário para sua operação e controle da sua operação.

O *software* informa ao usuário as opções disponíveis e como deve proceder para executar a tarefa em questão?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

## EFICIÊNCIA

A descrição de produto pode incluir dados sobre o comportamento do produto em relação ao tempo, como por exemplo tempo de resposta.

**Comportamento em relação ao tempo:** atributos do *software* que evidenciam seu tempo de resposta, tempo de processamento e velocidade na execução de suas funções.

O *software* faz reorganização da base de dados constantemente?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

O *software* tem seu desempenho comprometido quando trabalha com grandes volumes de dados?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [x] **Não atende** [ ]

**Comportamento em relação aos recursos:** atributos do *software* que evidenciam a quantidade de recursos usados e a duração de seu uso na execução de suas funções.

O *software* exige pouco recurso de máquina, e faz pouco acesso a disco?

**Atende completamente** [ ] **Atende** [x] **Atende parcialmente** [ ] **Não atende** [ ]

## MANUTENIBILIDADE

Série de atributos que suportam um esforço necessário para fazer modificações específicas.

**Analisabilidade:** atributos do *software* que evidenciam o esforço necessário para diagnosticar deficiências ou causas de falhas, ou para identificar partes a serem modificadas.

O *software* utiliza um padrão para nomes identificadores e estes são significativos e concisos?

Atende completamente  Atende  Atende parcialmente  Não atende

**Modificabilidade:** atributos do *software* que evidenciam o esforço necessário para modificá-lo, remover seus defeitos inesperados, ocasionados por modificações.

O *software* permite incluir ou excluir campos e aumentar o número de casas decimais?

Atende completamente  Atende  Atende parcialmente  Não atende

**Estabilidade:** atributos do *software* que evidenciam o risco de defeitos inesperados, ocasionados por modificações.

O *software* possui estabilidade (não apresentando necessidade de manutenções freqüentes)?

Atende completamente  Atende  Atende parcialmente  Não atende

**Testabilidade:** atributos do *software* que evidenciam o esforço necessário para validar o *software* modificado.

O *software* possui uma base de demonstração para testes e quando este é modificado são adicionados casos de testes para validar as modificações?

Atende completamente  Atende  Atende parcialmente  Não atende

## PORTABILIDADE

Série de atributos que suportam a habilidade do sistema ser transferido de um ambiente para outro qualquer. E, ser operado e manipulado de maneira fácil e adequada em configurações de equipamentos diferentes da original.



**Adaptabilidade:** atributos do *software* que evidenciam sua capacidade de ser adaptado a ambientes deferentes especificados, sem a necessidade de aplicação de outras ações ou meios além daqueles fornecidos para essa finalidade pelo *software* considerado.

O *software* pode ser facilmente modificado para atender as necessidades dos usuários, como por exemplo, alterações em legislação, ampliação da empresa ou abertura de filial?

**Atende completamente** [ ] **Atende** [x] **Atende parcialmente** [ ] **Não atende** [ ]

**Capacidade de ser instalado:** atributos do *software* que evidenciam o esforço necessário para sua instalação em um ambiente especificado.

O *software* possui comandos intuitivos durante a instalação e faz modificações de *config* e *autoexec* automaticamente?

**Atende completamente** [x] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

**Conformidade:** atributos do *software* que o tornam consonante com padrões ou convenções relacionados à portabilidade.

O *software* é adequado ao fim que foi destinado?

**Atende completamente** [x] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

**Capacidade de substituir:** atributos do *software* que evidenciam sua capacidade e esforço necessário para substituir um outro *software*, no ambiente estabelecido para este outro *software*.

O *software* é capaz de substituir um outro e possui a capacidade para ser instalado e adaptabilidades idênticas à deste?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [x]

**Avaliação do *software* SISTEMA INTEGRADO COMERCIAL versão 4.76 da Empresa Pic Informática Ltda. São José do Rio Preto/SP.**

Para calcularmos a qualidade do *software* desta avaliação, devemos primeiramente calcular a média ponderada de cada subcaracterística, terminado o cálculo das subcaracterísticas de uma determinada característica, calculamos a média ponderada da característica. Ao final calculamos a qualidade do *software* com base nos cálculos das características a saber:

Para calcularmos a média ponderada das subcaracterísticas vamos utilizar a seguinte fórmula:

$$VS = ((PPg_1 * Ps) + (PPg_2 * Ps) + \dots + (PPgn * Ps)) / \Sigma Ps$$

Onde **VS** é Valor da Subcaracterística; **PPg** é a Pontuação da Pergunta em questão (cada pergunta desta característica) e **Ps** é o Peso da subcaracterística.

Característica: **FUNCIONALIDADE** (peso 0,3)

Subcaracterística: **Adequação** (peso 0,3)

$$VS = ((1 * 0,3) + (0 * 0,3) + (0 * 0,3) + (1 * 0,3) + (2 * 0,3) + (1 * 0,3) + (1 * 0,3) + (1 * 0,3) + (1 * 0,3) + (1 * 0,3) + (0 * 0,3)) / (0,3 + 0,3 + 0,3 + 0,3 + 0,3 + 0,3 + 0,3 + 0,3 + 0,3 + 0,3 + 0,3)$$

$$VS = 0,82$$

Subcaracterística: **Interoperabilidade** (peso 0,2)

$$VS = ((0 * 0,2) + (0 * 0,2)) / (0,2 + 0,2)$$

$$VS = 0$$

Subcaracterística: **Conformidade** (peso 0,2)

$$VS = ((3 * 0,2) + (2 * 0,2)) / (0,2 + 0,2)$$

$$VS = 2,5$$

Subcaracterística: **Segurança de acesso** (peso 0,3)

$$VS = ((0 * 0,3) + (1 * 0,3)) / (0,3 + 0,3)$$

$$VS = 0,5$$

Após obtermos os valores das subcaracterísticas, devemos calcular o valor da característica, com a seguinte fórmula:

$$VC = ((VS_1 * PsC) + (VS_2 * PsC) + (VS_3 * PsC) + (VS_4 * PsC)) / \Sigma PsC$$

Onde **VC** é o Valor da Característica; **VS** é o Valor da Subcaracterística e **PsC** é o valor do Peso da Característica.

$$VC = ((0,82 * 0,3) + (0 * 0,3) + (2,5 * 0,3) + (0,5 * 0,3)) / (0,3 + 0,3 + 0,3 + 0,3)$$

$$VC = 0,96$$

Característica: **CONFIABILIDADE** (peso 0,1)

Subcaracterística: **Maturidade** (peso 0,4)

$$VS = ((2 * 0,4) + (0 * 0,4)) / (0,4 + 0,4)$$

$$VS = 1$$

Subcaracterística: **Tolerância a falhas** (peso 0,2)

$$VS = ((0 * 0,2) + (0 * 0,2) + (0 * 0,2)) / (0,2 + 0,2 + 0,2)$$

$$VS = 0$$

Subcaracterística: **Recuperabilidade** (peso 0,4)

$$VS = ((0 * 0,4) + (0 * 0,4)) / (0,4 + 0,4)$$

$$VS = 0$$

Calculando o valor da característica **confiabilidade**:

$$VC = ((VS_1 * PsC) + (VS_2 * PsC) + (VS_3 * PsC)) / \Sigma PsC$$

$$VC = ((1 * 0,1) + (0 * 0,1) + (0 * 0,1)) / (0,1 + 0,1 + 0,1)$$

$$VC = 0,33$$

Característica: **USABILIDADE** (peso 0,3)

Subcaracterística: **Inteligibilidade** (peso 0,3)

$$VS = (0 * 0,3) / (0,3)$$

$$VS = 0$$

Subcaracterística: **Apreensibilidade** (peso 0,4)

$$VS = (0 * 0,4) / (0,4)$$

$$VS = 0$$

Subcaracterística: **Operacionalidade** (peso 0,3)

$$VS = (1 * 0,3) / (0,3)$$

$$VS = 1$$

Calculando o valor da característica usabilidade:

$$VC = ((VS_1 * PsC) + (VS_2 * PsC) + (VS_3 * PsC)) / \Sigma PsC$$

$$VC = ((0 * 0,3) + (0 * 0,3) + (1 * 0,3)) / (0,3 + 0,3 + 0,3)$$

$$VC = 0,33$$

Subcaracterística: **EFICIÊNCIA** (peso 0,1)

Subcaracterística: **Comportamento em relação ao tempo** (peso 0,6)

$$VS = ((1 * 0,6) + (1 * 0,6)) / (0,6 + 0,6)$$

$$VS = 1$$

Subcaracterística: **Comportamento em relação aos recursos** (peso 0,4)

$$VS = (2 * 0,4) / (0,4)$$

$$VS = 2$$

Calculando o valor da característica **eficiência**:

$$VC = ((VS_1 * PsC) + (VS_2 * PsC) + (VS_3 * PsC)) / \Sigma PsC$$

$$VC = ((1 * 0,1) + (2 * 0,1)) / (0,1 + 0,1)$$

$$VC = 1,5$$

Característica: **MANUTENIBILIDADE** (peso 0,1)

Subcaracterística: **Analisabilidade** (peso 0,3)

$$VS = (3 * 0,3) / (0,3)$$

$$VS = 3$$

Subcaracterística: **Modificabilidade** (peso 0,2)

$$VS = (1 * 0,2) / (0,2)$$

$$VS = 1$$

Subcaracterística: **Estabilidade** (peso 0,3)

$$VS = (2 * 0,3) / (0,3)$$

$$VS = 2$$

Subcaracterística: **Testabilidade** (peso 0,2)

$$VS = (1 * 0,2) / (0,2)$$

$$VS = 1$$

Calculando o valor da característica **manutenibilidade**:

$$VC = ((VS_1 * PsC) + (VS_2 * PsC) + (VS_3 * PsC) + (VS_4 * PsC)) / \Sigma PsC$$

$$VC = ((3 * 0,1) + (1 * 0,1) + (2 * 0,1) + (1 * 0,1)) / (0,1 + 0,1 + 0,1 + 0,1)$$

$$VC = 1,75$$

Característica: **PORTABILIDADE** (peso 0,1)

Subcaracterística: **Adaptabilidade** (peso 0,3)

$$VS = (2 * 0,3) / (0,3)$$

$$VS = 2$$

Subcaracterística: **Capacidade de ser instalado** (peso 0,2)

$$VS = (3 * 0,2) / (0,2)$$

$$VS = 3$$

Subcaracterística: **Conformidade** (peso 0,3)

$$VS = (3 * 0,2) / (0,2)$$

$$VS = 3$$

Subcaracterística: **Capacidade de substituir** (peso 0,2)

$$VS = (0 * 0,2) / (0,2)$$

$$VS = 0$$

Calculando o valor da característica **portabilidade**:

$$VC = ((VS_1 * PsC) + (VS_2 * PsC) + (VS_3 * PsC) + (VS_4 * PsC)) / \Sigma PsC$$

$$VC = ((2 * 0,1) + (3 * 0,1) + (3 * 0,1) + (0 * 0,1)) / (0,1 + 0,1 + 0,1 + 0,1)$$

$$VC = 2$$

Tendo calculado os valores parciais de cada característica, devemos calcular o valor final de qualidade do *software*.

**PSo** = Pontuação do *Software*

**VPC** = Valor Parcial das Características

**PMP** = Pontuação Máxima Possível

$$PSo = (VC_1 + VC_2 + VC_3 + VC_4 + VC_5 + VC_6) / PMP$$

$$PSo = (0,96 + 0,33 + 0,33 + 1,5 + 1,5 + 2) / 18$$

$$PSo = 0,37 \text{ ou } 37\%$$

O *software* atingiu 37% de qualidade, sendo considerado um *software regular*, segundo a ISO/IEC 9126.

<b>0</b>	<b>a</b>	<b>25%</b>	<b>=</b>	<b>Insuficiente</b>
<b>25,1</b>	<b>a</b>	<b>50%</b>	<b>=</b>	<b>Regular</b>
<b>50,1</b>	<b>a</b>	<b>75%</b>	<b>=</b>	<b>Bom</b>
<b>75,1</b>	<b>a</b>	<b>100%</b>	<b>=</b>	<b>Excelente</b>

## APÊNDICE B – Questionários referentes uso de *software* de automação comercial

Empresa A

Quanto ao uso do *software*, o que seria necessário para este ser considerado um *software* de qualidade?

- No que diz respeito a contas a pagar:
  - o mesmo deve permitir que possam ser cadastradas notas fiscais sem que haja atualização simultânea do estoque. Somente após a devida conferência esta atualização seja feita;
  - consultar produtos por nome ou código;
  - permitir que possam ser consultados itens de uma determinada nota de entrada;
  - ser possível consultar contas a pagar por período e fornecedor;
  - formar preço de vendas computando valores como transportes e outras taxas que por ventura tenham incidido sobre aquelas mercadorias.
  
- Ao contas a receber:
  - permitir que ao cadastrar um cliente seja possível ir direto a emissão do pedido, apenas com uma pergunta ao usuário;
  - permitir consultar duplicatas pendentes por período/data.
  
- Ao estoque:
  - permitir o cadastro de descontos por grupo de produto, além do tradicional “por produto”;
  - permitir cadastrar entrada de produto no estoque sem que haja pedido.
  
- Mala direta:
  - permitir listar clientes por data de aniversário para envio de correspondência;
  - listar clientes por data da última visita.
  
- Capacidade multiusuário:
  - o bloqueio existir em nível de arquivo;

- não existir limite de máquinas operando ao mesmo tempo.
- Funções avançadas:
  - ocultar funções que não sejam utilizadas por determinados usuários.
- Capacidade de integração:
  - fazer importações de dados de outros sistemas.
- Segurança:
  - lembrar da necessidade de efetuar *backup* diariamente ao final do expediente;
  - possuir acesso protegido por senhas;
  - consistir dados, na falta ou entrada errônea destes.
- Maturidade:
  - possuir controle de acesso para exclusão de registros.
- Usabilidade:
  - possuir tutorial *on-line*;
  - durante a operação do sistema, ser possível acessar o tutorial.
- Portabilidade:
  - ser facilmente modificável para atender um determinado usuário;
  - avisar da necessidade de uma cópia de segurança antes da instalação de uma nova versão.

Obs: os itens eficiência e manutenibilidade não foram respondidos por que o entrevistado não soube dar uma opinião sobre os mesmos. A empresa terceirizou todos os serviços que não estão ligados ao objetivo fim da mesma.

## Empresa B

Quanto ao uso do *software*, o que seria necessário para este ser considerado um *software* de qualidade?

- No que diz respeito a contas a pagar:
  - consultar uma nota ou seus itens;
  - permitir a consulta por fornecedor;
- No que diz respeito a contas a receber:
  - ser possível emitir relatório de pedidos pendentes.
- No que diz respeito ao estoque:
  - aceitar a entrada de produtos sem que haja pedido;
  - ser possível cadastrar descontos por grupo e por produto;
  - permitir cadastrar quantidade mínima;
  - consultar produto por grupo, subgrupo ou fornecedor.
- Mala direta:
  - consultar e emitir relatórios de clientes por data de fundação ou aniversário;
  - permitir selecionar clientes por faixa etária.
- Capacidade multiusuário:
  - não existir limites de estações operando ao mesmo tempo;
  - possuir prioridade para inclusões;
  - permitir submeter relatórios e outros processos em *background*.
- Funções avançadas:
  - permitir configurar cores do *software*;
  - permitir ocultar funções que não são utilizadas por determinado usuário;
  - o cadastro de clientes estar integrado com a mala direta.
- Capacidade de integração:
  - Integração com outros sistemas sem a necessidade de adaptações, ou se necessário, com o mínimo de tempo possível;



- Estar pronto para utilizar os equipamentos para automação como por exemplo: impressoras de cheques, leitores ópticos, terminais de consultas de preços, etc.
- Segurança:
  - ter rotinas de *backup* internamente e lembrar da necessidade deste diariamente;
  - possuir diferenciação de senhas com relação ao nível de acesso aos diferentes módulos.
- Maturidade:
  - nas alterações informar das conseqüências sobre os outros módulos do sistema;
  - consistir campos, com controle de dados que fogem da média, alertando o usuário através de uma mensagem.
- Usabilidade:
  - ter tutorial *on-line*;
  - durante sua operação ser possível acessar este tutorial;
  - possuir empresa demonstração;
- Eficiência:
  - exigir poucos recursos de máquina;
  - tempo de resposta adequado a complexidade das funções executadas;
- Portabilidade
  - Possuir capacidade de substituir outro *software*;
  - Avisar da necessidade de cópia de segurança antes da instalação de novas versões.

Obs: Todas as operações no que diz respeito a parte fiscal, financeira e de RH são realizados pela matriz da empresa.

## Empresa C

Quanto ao uso do *software*, o que seria necessário para este ser considerado um *software* de qualidade?

- No que diz respeito ao contas a pagar:
  - permitir o cadastro de notas com a opção de não atualizar o estoque automaticamente;
  - permitir que os produtos possam ser consultado por nome e/ou código;
  - permitir que sejam consultados itens de uma nota de compra;
  - permitir que sejam excluídas faturas devolvidas.
  
- No que diz respeito a contas a receber:
  - permitir listar duplicatas pendentes por período;
  - permitir listar pedidos pendentes;
  - listar contas a receber por período.
  
- No que diz respeito ao estoque:
  - ser possível cadastrar produtos por grupo e subgrupos;
  - permitir cadastrar quantidade mínima, para avisar da necessidade de reposição do estoque;
  - consultar produto por grupo, subgrupo.
  
- Mala direta:
  - consultar clientes por aniversário;
  - permitir selecionar clientes por faixa etária.
  - permitir selecionar clientes por última compra.
  
- Capacidade multiusuário:
  - bloqueio em nível de arquivos;
  - ter prioridade para inclusões;
  - permitir submeter relatórios e outros processos em *background*.
  
- Funções avançadas:
  - o cadastro de clientes e fornecedores integrado com a mala direta.

- Capacidade de integração:
  - Integração com outros sistemas sem a necessidade de adaptações.
- Segurança:
  - lembrar da necessidade de *backup* diariamente;
  - fazer consistência de dados, da falta ou entrada errônea destes.
- Maturidade:
  - possuir controle de acesso para exclusão de registro;
  - em casos de alterações, apresentar mensagem indicando os módulos que serão afetados e suas possíveis conseqüências.
- Usabilidade:
  - ter tutorial *on-line*;
  - durante sua operação ser possível acessar este tutorial;
  - manual de usuário com índice remissivo.
- Eficiência:
  - exigir poucos recursos de máquina;
- Portabilidade
  - possuir capacidade de substituir outro *software*.

Obs: O entrevistado não soube opinar sobre o item manutenibilidade. Embora não utilize todos os itens do questionário, o entrevistado respondeu o que achava que um *software* precisaria para o mesmo considerar com qualidade.

## Empresa D

Quanto ao uso do *software*, o que seria necessário para este ser considerado um *software* de qualidade?

- No que diz respeito a contas a pagar:
  - permitir cadastrar notas de compras para faturamento posterior;
  - permitir consultar contas a pagar por fornecedor e data num determinado período;
  - permitir listar crédito de impostos por período.
  
- No que diz respeito a contas a receber:
  - permitir consultar extrato do cliente em vídeo;
  - permite gerar e imprimir carnês e duplicatas a partir do próprio sistema;
  - permite que seja configurada a autenticação obrigatória para todos os documentos emitidos, por tipo deste.
  
- No que diz respeito ao estoque:
  - ser possível cadastrar margem de lucro e desconto máximo, por produto;
  - permitir cadastrar quantidade mínima, para avisar da necessidade de reposição do estoque;
  - permite listar inventário de produtos, quantidade física e valores financeiros.
  
- Mala direta:
  - consultar clientes por aniversário e faixa etária;
  - permitir selecionar cliente por tipo de pessoa, física ou jurídica;
  - permitir selecionar clientes por última compra.
  
- Capacidade multiusuário:
  - Possuir bloqueio em nível de arquivos;
  - possuir prioridade para inclusões;
  - não possuir limites de estações trabalhando.

- Funções avançadas:
  - permitir utilizar dados cadastrados em outras filiais;
  - permitir desativar opções que não são utilizadas por outros usuários.
  
- Capacidade de integração:
  - Integração com outros sistemas sem a necessidade de adaptações.
  
- Segurança:
  - emitir mensagem lembrando da necessidade de *backup* diariamente;
  - fazer consistência de dados, da falta ou entrada errônea destes.
  
- Maturidade:
  - controlar o acesso para exclusão de registro, por usuário;
  - informar, as possíveis conseqüências, nos diversos módulos quando for realizada alguma alteração.
  
- Usabilidade:
  - ter tutorial *on-line* e empresa demonstração;
  - permitir acessar o tutorial durante a operação do sistema;
  - texto do manual do usuário com o mínimo possível de termos técnicos, para seu melhor entendimento por este.
  
- Eficiência:
  - exigir poucos recursos de máquina;

Obs: O entrevistado não soube opinar sobre os itens manutenibilidade e portabilidade.

## **APÊNDICE C – perguntas desenvolvidas para serem usadas no *software* proposto para a avaliação de sistemas de automação comercial**

### **FUNCIONALIDADE**

Refere-se a existência de um conjunto de funções que satisfazem necessidades explícitas ou implícitas e suas propriedades específicas.

**Adequação:** atributos do *software* que evidenciam a presença de um conjunto de funções e sua apropriação para as tarefas especificadas.

#### ➤ **Contas a pagar**

1. O *software* dispõe das funções necessárias para consultas de fornecedores, produtos, contas a pagar e notas estornadas?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

2. o *software* permite a exclusão de fornecedores desativados, faturas devolvidas e recebimentos estornados?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

3. O *software* permite que valores como transportes e outras taxas sejam agregados para formar preço de venda?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

#### ➤ **Contas a receber**

1. O *software* permite incluir duplicatas manualmente e seus respectivos desdobramentos?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

2. O *software* permite consultar duplicatas por vencimento, em atraso até a data da consulta ou todas as duplicatas em aberto de um determinado cliente?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

3. O *software* permite consultar pedidos em aberto e/ou entregues, produtos com maior ou menor giro e tabela de preço durante uma operação de venda?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

➤ **Estoque**

1. o *software* permite cadastrar quantidade mínima e etiquetas de código de barras?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

2. O *software* permite consultar documentos de devolução, de transferência e de venda direta por produto?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

3. O *software* permite listar produtos com a data de última venda?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

➤ **Mala direta**

1. O *software* permite selecionar clientes por data de última compra ou aniversário?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

2. O *software* permite editar correspondência utilizando os campos dos arquivos?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

**Interoperabilidade:** atributos do *software* que evidenciam sua capacidade de interagir com sistemas especificados.

➤ **Capacidade de integração**

1. O *software* permite que seja integrado a outro sistema sem que haja necessidade de adaptações?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

2. O *software* permite a integração com hardware, como por exemplo, *scanners*, canetas óticas, impressoras de cheques e coletores de dados?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

**Conformidade:** atributos do *software* que fazem com que ele esteja de acordo com as normas, convenções ou regulamentações previstas em leis e descrições similares, relacionadas à aplicação.

➤ **Obediência à legislação**

1. O *software* efetua os cálculos de ICMS, ISS e outros impostos com precisão?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

**Segurança de acesso:** atributos do *software* que evidenciam sua capacidade de evitar o acesso não autorizado, acidental ou deliberado, a programas e dados.

➤ **Backups**

1. O *software* manda mensagem lembrando da necessidade de backup e faz a compactação deste?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

➤ **Controle de acesso**

1. O *software* é protegido por senha, e há diferenciação de senhas para acessar os diferentes módulos?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

## CONFIABILIDADE

O sistema, compreendendo hardware e *software*, bem como os programas que pertencem ao produto, não devem entrar em um estado no qual o usuário não consiga controlá-lo, nem deve corromper ou perder dados.

**Maturidade:** atributos do *software* que evidenciam a frequência de falhas por defeitos no *software*.

➤ **Precisão**

1. O *software* efetua cálculo de impostos, baixa e entrada no estoque e calcula os preços dos produtos corretamente?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

➤ **integridade de dados**

1. O *software* possui controle para exclusão de registros, informando as conseqüências sobre os diversos módulos?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

**Tolerância a falhas:** atributos do *software* que evidenciam sua capacidade em manter um nível de desempenho especificados nos casos de falhas no *software* ou de violação nas interfaces especificadas.



➤ **Proteção contra erros**

1. O *software* possui consistência de campos controlando dados que fogem da média, e evita cadastro de dados já existentes?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

➤ **Proteção contra situações limites**

1. O *software* volta ao estado anterior após uma parada anormal, e informa ao usuário a situação dos dados?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

2. O *software* tem controle de situação limite, como por exemplo, número de usuários suportados simultaneamente por ele?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

**Recuperabilidade:** atributos do *software* que evidenciam sua capacidade de restabelecer seu nível de desempenho e recuperar os dados diretamente afetados, em caso de falha, e no tempo e esforço necessário para tal.

➤ **Recuperação de erros do usuário**

1. O sistema permite desfazer uma operação para recuperar dados por erro do usuário?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

➤ **Recuperação em caso de falhas no sistema**

1. O *software* não perde informações em casos de “*run time*” e é possível continuar utilizando o mesmo?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

## USABILIDADE

Esforço necessário para se utilizar, operar, aprender, bem como para o julgamento individual desse uso, por um conjunto de usuários explícitos ou implícitos.

**Inteligibilidade:** atributos do *software* que evidenciam o esforço do usuário para reconhecer o conceito lógico e sua aplicabilidade.

1. O *software* possui tutorial *on-line*, empresa demonstração e tutorial impresso?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

**Apreensibilidade:** atributos do *software* que evidenciam o esforço do usuário para compreender sua aplicação (por exemplo: controle de operação, entradas, saídas).

1. O *help* apresenta uma explicação para todos os itens relacionados com a operação do sistema e os termos utilizados possuem o mesmo significado durante todo o *help*?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

**Operacionalidade:** atributos do *software* que evidenciam o esforço do usuário para sua operação e controle da sua operação.

1. O *software* informa ao usuário as opções disponíveis e como deve proceder para executar a tarefa em questão?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

## EFICIÊNCIA

A descrição de produto pode incluir dados sobre o comportamento do produto em relação ao tempo, como por exemplo tempo de resposta.

**Comportamento em relação ao tempo:** atributos do *software* que evidenciam seu tempo de resposta, tempo de processamento e velocidade na execução de suas funções.

1. O *software* faz reorganização da base de dados constantemente?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

2. O *software* tem seu desempenho comprometido quando trabalha com grandes volumes de dados?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

**Comportamento em relação aos recursos:** atributos do *software* que evidenciam a quantidade de recursos usados e a duração de seu uso na execução de suas funções.

1. O *software* exige pouco recurso de máquina, e faz pouco acesso a disco?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

## MANUTENIBILIDADE

Série de atributos que suportam um esforço necessário para fazer modificações específicas.

**Analisabilidade:** atributos do *software* que evidenciam o esforço necessário para diagnosticar deficiências ou causas de falhas, ou para identificar partes a serem modificadas.

1. O *software* utiliza um padrão para nomes identificadores e estes são significativos e concisos?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

**Modificabilidade:** atributos do *software* que evidenciam o esforço necessário para modificá-lo, remover seus defeitos inesperados, ocasionados por modificações.

1. O *software* permite incluir ou excluir campos e aumentar o número de casas decimais?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

**Estabilidade:** atributos do *software* que evidenciam o risco de defeitos inesperados, ocasionados por modificações.

1. O *software* possui estabilidade (não apresentando necessidade de manutenções frequentes)?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

**Testabilidade:** atributos do *software* que evidenciam o esforço necessário para validar o *software* modificado.

1. O *software* possui uma base de demonstração para testes e quando este é modificado são adicionados casos de testes para validar as modificações?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

## **PORTABILIDADE**

Série de atributos que suportam a habilidade do sistema ser transferido de um ambiente para outro qualquer. E, ser operado e manipulado de maneira fácil e adequada em configurações de equipamentos diferentes da original.

**Adaptabilidade:** atributos do *software* que evidenciam sua capacidade de ser adaptado a ambientes deferentes especificados, sem a necessidades de aplicação de outras ações ou meios além daqueles fornecidos para essa finalidade pelo *software* considerado.

1. O *software* pode ser facilmente modificado para atender as necessidades dos usuários, como por exemplo, alterações em legislação, ampliação da empresa ou abertura de filial?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

**Capacidade de ser instalado:** atributos do *software* que evidenciam o esforço necessário para sua instalação em um ambiente especificado.

1. O *software* possui comandos intuitivos durante a instalação e faz modificações de *config* e *autoexec* automaticamente?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

**Conformidade:** atributos do *software* que o tornam consonante com padrões ou convenções relacionados à portabilidade.

1. O *software* é adequado ao fim que foi destinado?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

**Capacidade de substituir:** atributos do *software* que evidenciam sua capacidade e esforço necessário para substituir um outro *software*, no ambiente estabelecido para este outro *software*.

1. O *software* é capaz de substituir um outro e possui a capacidade para ser instalado e adaptabilidade idênticas à deste?

**Atende completamente** [ ] **Atende** [ ] **Atende parcialmente** [ ] **Não atende** [ ]

## APÊNDICE D – dicionário de dados

AVALIAÇÃO				
Name	Code	Type	P	M
CÓDIGO DA AVALIAÇÃO	CDAVALIACAO	Integer	Yes	Yes
CODIGO DO TIPO	CDTIPO	Integer	No	Yes
C DO SOFTWARE	CDSOFTWARE	Integer	No	Yes
CODIGO DO AVALIADOR	CDAVALIADOR	Integer	No	Yes
DESCRIÇÃO DO ESPAÇO	DSESPACO	varchar(10)	No	No
DESCRIÇÃO DO HARDWARE ESPECIAL	DSHARDESP	varchar(60)	No	No
DESCRIÇÃO HARDWARE	DSHARDWARE	varchar(60)	No	No
DESCRIÇÃO DA IMPRESSORA	DSIMPRESSORA	varchar(60)	No	No
DESCRIÇÃO DOS MANUAIS	DSMANUAL	varchar(60)	No	No
DESCRIÇÃO DA MIDIA	DSMIDIA	varchar(60)	No	No
DESCRIÇÃO DO MONITOR	DSMONITOR	varchar(20)	No	No
DESCRIÇÃO DO OBJETIVO	DSOBJETIVO	varchar(60)	No	No
DESCRIÇÃO DE OUTROS NÃO MENCIONADOS	DSOUTROS	varchar(100)	No	No
DESCRIÇÃO DA MEMÓRIA RAM	DSRAM	varchar(10)	No	No
DESCRIÇÃO DO SISTEMA OPERACIONAL	DSSO	varchar(20)	No	No
DESCRIÇÃO DE SOFTWARE ESPECIAL	DSSOFTESP	varchar(20)	No	No
DATA FINAL	DTFINAL	date	No	No
DATA INICIAL	DTINICIAL	date	No	No
HORA FINAL	HRFINAL	time	No	No
HORA INICIAL	HRINICIAL	time	No	No
VALOR DO RESULTADO AVALIAÇÃO	VLRESULTAAVALIACAO	decimal(3,2)	No	No
VALOR DO RESULTADO DA MÉDIA SIMPLES	VLRESULTAMEDIASIMPLES	varchar(60)	No	No

AVALIADOR				
Name	Code	Type	P	M
LOGIN	LOGIN	Varchar(20)	No	Yes
SENHA	SENHA	Varchar(10)	No	Yes
CODIGO DO AVALIADOR	CDAVALIADOR	integer	Yes	Yes
NOME DO AVALIADOR	NMAVALIADOR	varchar(30)	No	No
DESCRIÇÃO DA FUNÇÃO	DSFUNCAO	varchar(30)	No	No
DESCRIÇÃO DO ENDEREÇO DO AVALIADOR	DSENDAAVALIADOR	varchar(30)	No	No
DESCRIÇÃO BAIRRO DO AVALIADOR	DSBAIAVALIADOR	varchar(15)	No	No
DESCRIÇÃO CIDADE DO AVALIADOR	DSCIDAVALIADOR	varchar(20)	No	No
DESCRIÇÃO DA UF DO AVALIADOR	DSUFAVALIADOR	varchar(2)	No	No
NÚMERO DO TELEFONE DO AVALIADOR	NRFONEAVALIADOR	varchar(15)	No	No
DESCRIÇÃO DO EMAIL DO AVALIADOR	DSEMAILAVALIADOR	varchar(30)	No	No

<b>CARACTERISTICA</b>				
<b>Name</b>	<b>Code</b>	<b>Type</b>	<b>P</b>	<b>M</b>
CODIGO DA CARACTERÍSTICA	CDCARACTERISTICA	integer	Yes	Yes
DESCRIÇÃO DA CARACTERÍSTICA	DSCARACTERISTICA	varchar(50)	No	No
DESCRIÇÃO OBSERVAÇÃO DA CARACTERÍSTICA	DSOBSCHARACTERISTICA	varchar(200)	No	No

<b>CATEGORIA</b>				
<b>Name</b>	<b>Code</b>	<b>Type</b>	<b>P</b>	<b>M</b>
CÓDIGO DA CATEGORIA	CDCATEGORIA	integer	Yes	Yes
DESCRIÇÃO DA CATEGORIA	DSCATEGORIA	varchar(30)	No	No

<b>PRODUTORA</b>				
<b>Name</b>	<b>Code</b>	<b>Type</b>	<b>P</b>	<b>M</b>
CÓDIGO DA PRODUTORA	CDPRODUTORA	integer	Yes	Yes
NOME DA PRODUTORA	NMPRODUTORA	varchar(50)	No	No
DESCRIÇÃO DO ENDERECO DA PRODUTORA	DSENDPRODUTORA	varchar(50)	No	No
DESCRIÇÃO DO BAIRRO DA PRODUTORA	DSBAIPRODUTORA	varchar(30)	No	No
DESCRIÇÃO DA CIDADE DA PRODUTORA	DSCIDPRODUTORA	varchar(20)	No	No
DESCRIÇÃO DA UF DA PRODUTORA	DSUFPRODUTORA	varchar(2)	No	No
DESCRIÇÃO FONE DA PRODUTORA	DSFONEPRODUTORA	varchar(15)	No	No
DESCRIÇÃO EMAIL DA PRODUTORA	DSEMAILPRODUTORA	varchar(30)	No	No
DESCRIÇÃO DA HOME PAGE	DSHOMEPAGE	varchar(30)	No	No
DESCRIÇÃO DA PESSOA DE CONTATO	DPESSOACONTATO	varchar(20)	No	No

<b>QUESTAO</b>				
<b>Name</b>	<b>Code</b>	<b>Type</b>	<b>P</b>	<b>M</b>
CÓDIGO DA QUESTÃO	CDQUESTAO	integer	Yes	Yes
IDENTIFICAÇÃO DA CARACTERÍSTICA	CDSUBCARACTERISTICA	Integer	No	Yes
DESCRIÇÃO DA QUESTÃO	DSQUESTAO	varchar(200)	No	No
DESCRIÇÃO DA OBSERVAÇÃO DA QUESTÃO	DSOBSQUESTAO	varchar(200)	No	No

<b>RESULTADO</b>				
<b>Name</b>	<b>Code</b>	<b>Type</b>	<b>P</b>	<b>M</b>
CODIGO DA AVALIAÇÃO	CDAVALIACAO	integer	No	Yes
CÓDIGO DA CATEGORIA	CDCATEGORIA	integer	No	Yes
CÓDIGO DA CARACTERISTICA	CDCARACTERISTICA	integer	No	Yes
CÓDIGO DA SUBCARACTERISTICA	CDSUBCARACTERISTICA	integer	No	Yes
CÓDIGO DA QUESTÃO	CDQUESTAO	Integer	Yes	Yes
VALOR RESULTADO	VLRESULTADO	decimal(3,2)	No	No

<b>SOFTWARE</b>				
<b>Name</b>	<b>Code</b>	<b>Type</b>	<b>P</b>	<b>M</b>
CÓDIGO DO SOFTWARE	CDSOFTWARE	integer	Yes	Yes
CÓDIGO DA CATEGORIA	CDATEGORIA	integer	No	Yes
CÓDIGO DA PRODUTORA	CDPRODUTORA	integer	No	Yes
NOME DO SOFTWARE	NMSOFTWARE	varchar(60)	No	No
DESCRIÇÃO DA VERSÃO	DSVERSAO	varchar(10)	No	No
DESCRIÇÃO DAS INFORMAÇÕES DO SOFTWARE	DSINF SOFTWARE	varchar(150)	No	No

<b>SUBCARACTERISTICA</b>				
<b>Name</b>	<b>Code</b>	<b>Type</b>	<b>P</b>	<b>M</b>
CODIGO DA SUBCARACTERÍSTICA	CDSUBCARACTERISTICA	integer	Yes	Yes
CÓDIGO DA CARACTERÍSTICA	CDCARACTERISTICA	integer	No	No
DESCRIÇÃO DA SUBCARACTERÍSTICA	DSSUBCARACTERISTICA	varchar(50)	No	No
DESCRIÇÃO OBSERVAÇÃO DA CARACTERÍSTICA	DSOBSCARACTERISTICA	varchar(400)	No	No

<b>TIPOAVALIACAO</b>				
<b>Name</b>	<b>Code</b>	<b>Type</b>	<b>P</b>	<b>M</b>
CODIGO TIPO AVALIACAO	CDTIPO	integer	Yes	Yes
DESCRIÇÃO DO TIPO DA AVALIAÇÃO	DSTIPOAVALIACAO	varchar(5)	No	No

## **APÊNDICE E – Levantamento Bibliográfico realizado para elaboração deste TCC**

### **Processo de desenvolvimento de sistemas com qualidade**

Autor: Danilo Scalet -DT

<http://www.pr.gov.br/batebyte/edicoes/1995/bb39/processo.htm>

### **Qualidade de *software***

UNIVERSIDADE DE PASSO FUNDO

<http://www.mct.gov.br/temas/info/Dsi/qualid99/99anexo2.htm>

### **Modelos de qualidade de desenvolvimento de *software***

Autor: Marco Aurélio Cordeiro - GPS

<http://www.pr.gov.br/batebyte/edicoes/2000/bb99/modelos.htm>

### **Métricas de *software***

Autor: Marco Aurélio Cordeiro - GPS

<http://www.pr.gov.br/batebyte/edicoes/2000/bb101/metricas.htm>

### **Grupo de qualidade do *software* - critérios de confiabilidade**

Autores: Adilson Fabris - GPS

Jair Fernandes - GAC

Roberto Mücke - GPS

Paulo Roberto Rosa - GPS

Viro Miguel Back

Wilson Mauri de Bonfim – GPS

<http://www.pr.gov.br/batebyte/edicoes/1995/bb39/critconf.htm>

### **Cartilha de automação comercial**

EAC - Engenharia Automação e Controle

[www.eacnet.com.br](http://www.eacnet.com.br)

### **Análise de requisitos para sistemas de informações, utilizando as ferramentas da qualidade e processos de *software***

Claudiomir Selner

<http://www.eps.ufsc.br/disserta99/selner/cap2.html>