UNIVERSIDADE REGIONAL DE BLUMENAU

CENTRO DE CIÊNCIAS EXATAS E NATURAIS

CURSO DE CIÊNCIAS DA COMPUTAÇÃO

(Bacharelado)

SOFTWARE DE APOIO AO PROCESSO DE AUDITORIA SEGUNDO NORMAS DE QUALIDADE

TRABALHO DE CONCLUSÃO DE CURSO SUBMETIDO À UNIVERSIDADE REGIONAL DE BLUMENAU PARA A OBTENÇÃO DOS CRÉDITOS NA DISCIPLINA COM NOME EQUIVALENTE NO CURSO DE CIÊNCIAS DA COMPUTAÇÃO — BACHARELADO

FABIO ALEXANDRE JUNCKES

BLUMENAU, DEZEMBRO/1999

1999/02-12

SOFTWARE DE APOIO AO PROCESSO DE AUDITORIA SEGUNDO NORMAS DE QUALIDADE

FABIO ALEXANDRE JUNCKES

ESTE TRABALHO DE CONCLUSÃO DE CURSO, FOI JULGADO ADEQUADO PARA OBTENÇÃO DOS CRÉDITOS NA DISCIPLINA DE TRABALHO DE CONCLUSÃO DE CURSO OBRIGATÓRIA PARA OBTENÇÃO DO TÍTULO DE:

BACHAREL EM CIÊNCIAS DA COMPUTAÇÃO

_____

Prof. Everaldo Artur Grahl — Orientador na FURB

_____

Prof. José Roque Voltolini da Silva — Coordenador do TCC

BANCA EXAMINADORA

_____

Prof. Everaldo Artur Grahl

_____

Prof. Evaristo Baptista

_____

Prof. Wilson Pedro Carli

Dedicatória

Dedico este trabalho a minha família pelo apoio e incentivo que recebi durante toda a minha vida.

Agradecimentos

Agradeço aos meus pais por terem me ajudado ao longo de minha trajetória.

Ao professor Everaldo Artur Grahl, pelo incentivo, orientação e atenção dispensada durante todo o desenvolvimento do trabalho.

Aos professores do curso que me repassaram da melhor maneira possível todos seus conhecimentos e orientações.

Volto a agradecer aos meus pais pelo incentivo e apoio que sempre obtive e com certeza sempre o terei.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE QUADROS

RESUMO

A auditoria de sistemas é o ramo da auditoria que revisa e avalia os controles internos informatizados. Com base no estudo das normas de qualidade ISO/IEC 12207, ISO 9000-3, ISO/IEC 15504 e CMM (Modelo de Capacidade e Maturidade), em particular o processo de auditoria, pretende-se especificar e implementar um software, no ambiente Delphi 3.0, que dê suporte aos auditores de sistemas quando da execução de alguma auditoria.

ABSTRACT

The audit system is the part of audit that reviews and evaluates the internal controls. With support of the studies of norms qualitys ISO/IEC 12207, ISO 9000-3, ISO/IEC 15504 and CMM (Capability Maturity Model), in special the audit process, intends to specify and develop a software, using the Delphi 3.0, that gives support to the system auditors when they are doing an audit.

1 INTRODUÇÃO

1.1 ORIGEM

A comunidade de informática vem criando normas e modelos de qualidade como ISO/IEC 12207, ISO 9000-3, CMM e ISO/IEC 15504 para regular e orientar a atividade de produção de software. A norma ISO/IEC 12207 (Processos de Ciclo de Vida de Software) tem por objetivo principal estabelecer uma estrutura comum para os processos de ciclo de vida de software. A norma ISO 9000-3 aborda basicamente situações em que um software específico é desenvolvido como parte de um contrato, de acordo com as especificações do comprador. O modelo CMM (Modelo de Capacidade e Maturidade de Software), é uma estrutura que descreve os elementos de um processo eficiente de software e um caminho revolucionário que aumenta a maturidade dos processos nas organizações de software. Já a norma ISO/IEC 15504, também conhecida como SPICE, constitui-se de um padrão para avaliação do processo de software, visando determinar a capacitação de uma organização.

A estrutura descrita na norma [ABN97], utiliza-se de uma terminologia bem definida e é composta de processos, atividades e tarefas a serem aplicadas em operações que envolvam, de alguma forma, o software, seja através de aquisição, fornecimento, desenvolvimento, operação ou manutenção.

Entre esses processos encontra-se o processo de auditoria. Segundo [ABN97] o processo de **auditoria** define as atividades para determinar adequação aos requisitos, planos e contrato, quando apropriado.

Em relação a [NBR93] a **auditoria** tem por função verificar se as atividades da qualidade estão em conformidade com a forma planejada e para determinar a eficácia do sistema da qualidade.

No modelo CMM, segundo [ANA96], a **auditoria** define atividades para verificar o quanto os produtos de software atendem os procedimentos e padrões aplicáveis.

Já em [IAH99], referente ao SPICE, a **auditoria** tem por finalidade assegurar que os produtos e processos empregados estão conforme os requisitos acordados.

Pretende-se a partir dos estudos das normas, em especial o processo de auditoria, implementar um software de apoio a este processo.

1.2 OBJETIVO

O objetivo deste trabalho é especificar e implementar um software que dê suporte ao processo de Auditoria de Sistemas segundo recomendações das normas e modelos de qualidade ISO/IEC 12207, ISO 9000-3, CMM e SPICE.

1.3 ORGANIZAÇÃO DO TEXTO

O trabalho é composto por seis capítulos, que serão descritos a seguir.

No primeiro capítulo é apresentado uma breve introdução sobre o trabalho que será exposto.

No segundo capítulo são apresentados os conceitos referentes a Auditoria de Sistemas.

No terceiro capítulo são apresentados os conceitos e estruturas das normas de qualidade que serão abordadas neste trabalho.

No quarto capítulo é apresentada a especificação do protótipo, onde constam o diagrama de contexto, modelo de entidade e relacionamento (MER) e o diagrama hierárquico funcional.

O quinto capítulo apresenta a implementação do protótipo, onde são apresentadas algumas telas do protótipo.

No sexto capítulo é apresentada a conclusão do trabalho.

2 AUDITORIA DE SISTEMAS

Devido a grandes mudanças ocorridas no últimos anos, em função da globalização da economia, e o avanço da área da ciência da computação ter ocorrido de forma bastante acentuada, a concorrência entre as empresas tem sido cada vez mais acirrada, uma vez que as empresas estão competindo não só com os concorrentes locais ou regionais, mas também com empresas estrangeiras.

Em função do crescimento da organização as operações tornam-se mais complexas e as decisões distanciam-se do centro de comando. As empresas têm procurado cada vez mais a qualidade de seus produtos. Isto vem ocorrendo através da utilização da informática por todos os setores da empresa. Esta difusão da informática, apesar de ter contribuído para o aumento da produtividade, tem causado, em alguns casos, diversos problemas relacionados a segurança, confiabilidade e interface com o usuário [LOO96].

Uma forma encontrada para tentar corrigir esses problemas foi o uso da Auditoria de Sistemas, onde o principal objetivo é garantir que um ambiente computadorizado (estrutura física, hardware, software e pessoas) esteja de acordo com padrões existentes, tanto internos como externos.

A questão que surge neste ponto é: como surgiu a Auditoria de Sistemas? Segundo [LOO96], a Auditoria de Sistemas surgiu com uma conseqüência natural da evolução da informática.

Inicialmente, os computadores eram operados basicamente por cientistas e utilizados para resolver problemas específicos. Com o passar dos anos, o número de profissionais que trabalhavam com informática e o aperfeiçoamento dos computadores aumentaram consideravelmente.

O surgimento do microcomputador provocou uma revolução na informática, resultando numa descentralização do CPD (Centro de Processamento de Dados) em direção ao usuário. Com isto, a preocupação com a qualidade dos sistemas passou a ser maior, provocando o

surgimento de novas técnicas, manuais e/ou computadorizados, para que os sistemas pudessem atender completamente as necessidades dos usuários.

Um fator complicador desta situação foi o surgimento das redes de computadores. A possibilidade de problemas com relação a segurança, confiabilidade, eficiência e eficácia tornaram-se extremamente grandes. Nesta realidade, a Auditoria de Sistemas tem um papel preponderante, como ferramenta auxiliar para a melhoria da qualidade dos sistemas informatizados.

2.1 DEFINIÇÕES

A principal preocupação dentro da Auditoria de Sistemas tem sido a segurança dos ambientes computadorizados. Isto pode ser verificado na definição feita pela Secretaria Especial de Informática - SEI, através de sua Comissão n^{o} 21, de Proteção de Dados, em seu relatório publicados em 1986 [FON91]:

A Auditoria de Sistemas é o ramo da auditoria, que revisa e avalia os controles internos informatizados visando:

a) Proteger os ativos da organização;
b) Manter a integridade dos dados;
c) Atingir eficaz e eficientemente os objetivos da organização

Com base na definição anterior, as responsabilidades do Auditor de Sistemas, segundo a comissão Especial n^{o} 21 da SEI [FON91], são:

a) Revisar, em forma contínua os controles estabelecidos pelos respectivos proprietários;
b) Recomendar os controles necessários para mitigar os riscos identificados;
c) Dar assistência aos proprietários para definir um curso de ação, se acontecer uma violação de segurança;
d) Fazer com que exista uma separação com relação a implantação e manutenção de software de controle de acesso;
e) Revisar os recursos de elaboração dos relatórios de software de controle de acesso para garantir a adequação dos mesmos.

Uma outra definição de Auditoria de Sistemas é a apresentada por [GIL98]: "...é um instrumento da direção da entidade, dos acionistas, do ambiente externo à organização, do povo para, indenpendemente, opinar, isto é, validar e avaliar a qualidade em termos de segurança, eficiência dos trabalhos desenvolvidos com a tecnologia dos computadores."

Apesar de ser mais abrangente do que a definição apresentada pela SEI, esta definição é bastante influenciada pelos objetivos e características da auditoria contábil, cuja preocupação básica é com os resultados.

Mais abrangente do que a definição apresentada por [GIL98], é o enfoque dado por [ARI94], "A função Auditoria de Sistemas tem por finalidade promover a adequação, avaliação e recomendações para o aprimoramento dos controles internos nos sistemas de informação da empresa, bem como na utilização dos recursos humanos, materiais e tecnológicos envolvidos no processamento dos mesmos."

Para entender melhor as definições apresentadas anteriormente, torna-se indispensável apresentar a definição de controle interno. Segundo [FON91] são todos os procedimentos internos instituídos pela empresa com o objetivo de evitar a ocorrência de falhas, involuntárias ou dolosas, e ao conjunto dá-se o nome de sistemas de controle interno.

Este enfoque tradicional, tem evitado que a Auditoria de Sistemas seja utilizada como uma ferramenta auxiliar na melhoria da qualidade de ambientes informatizados. Para isto, será necessário enfocar outros aspectos além de segurança, tratando a Auditoria de Sistemas de forma mais abrangente.

Este enfoque mais abrangente implica em definir a Auditoria de Sistemas como sendo um processo de avaliação independente da conformidade da infra-estrutura física, software (processo de desenvolvimento e produto) e seres humanos de um ambiente informatizado, com relação a padrões e/ou especificações pré-definidas, interna ou externamente.

Esta forma de visualizar a Auditoria de Sistemas, permite a efetiva melhora da qualidade dos ambientes informatizados, uma vez que direciona a atenção para dois aspectos fundamentais para a qualidade:

a) **processo**: a tendência é seguir as diretrizes das normas de qualidade, adaptando-as às características locais;
b) **seres humanos**: as pessoas são as responsáveis pela execução das atividades em uma empresa, tendo, portanto, grande impacto sobre a qualidade de produtos e/ou serviços.

Observando-se este último enfoque, percebe-se que a Auditoria de Sistemas aproxima-se muito da auditoria da qualidade. No entanto, enquanto a auditoria da qualidade centraliza a atenção sobre todos os itens que direta ou indiretamente influenciam a qualidade dos produtos e/ou serviços, a Auditoria de Sistemas focaliza sua atenção somente para os ambientes informatizados [LOO96].

2.2 IMPORTÂNCIA E VANTAGENS DA AUDITORIA DE SISTEMAS

A Auditoria de Sistemas, sendo abordada de uma maneira mais abrangente e direcionada a atual área de informática, é uma importante ferramenta para proporcionar a melhoria da qualidade dos ambientes informatizados.

Segundo [FON91], a importância da Auditoria de Sistemas fica evidente quando se observam os seguintes fatos:

a) as tarefas manuais estão sendo rapidamente transferidas para o computador;
b) a descentralização do processamento indica uma tendência cada vez maior da automatização integral dos processos;
c) os sistemas informatizados têm crescido em tamanho e complexidade;
d) o aumento do número de sistemas instalados indica a importância de suas informações para subsidiar as decisões administrativas;
e) a vulnerabilidade das organizações, devido à alta dependência da informática;
f) os desenvolvedores dos sistemas têm sido responsáveis pela implementação das auditorias nos sistemas da área de informática, devido à pouca experiência dos auditores tradicionais;
g) pouca importância dispensada à documentação dos sistemas, por parte dos analistas e programadores;

h) manutenção nos sistemas são inseguras, lentas e onerosas.

A Auditoria de Sistemas assume hoje um posicionamento de assessoria de alto nível, auxiliando a administração na execução eficiente de suas responsabilidades, para que a empresa possa melhorar a eficiência dos seus sistemas, diminuindo custos, riscos e aumentando o retorno dos investimentos. Desta forma, a Auditoria de Sistemas proporciona as seguintes vantagens:

a) sistemas mais seguros, eficientes e confiáveis;
b) facilidade para a implantação de normas de qualidade;
c) menor probabilidade de erros no processamento e/ou digitação;
d) ambientes mais interativos com os usuários;
e) obtenção de um histórico do desempenho dos sistemas informatizados na empresa.

Para a obtenção destas vantagens é necessário que o auditor de sistemas siga uma abordagem mais abrangente, preocupando-se inclusive com os seres humanos, responsáveis pelo desenvolvimento, manutenção e operação dos sistemas informatizados. Esta abordagem deve ser utilizada, independente da forma de Auditoria de Sistemas realizada [LOO96].

2.3 FORMAS DE AUDITORIA DE SISTEMAS

Segundo [LOO96] a Auditoria de Sistemas pode ser realizada de três formas diferentes:

a) Auditoria interna: auditoria realizada por um funcionário da própria empresa;
b) Auditoria externa: ocorre quando se contrata um auditor que não tem vínculos empregatícios com a empresa;
c) Associação da auditoria interna e externa: atuação conjunta de um funcionário da empresa e um auditor externo.

Conforme o quadro 1 representada, estas formas de auditoria apresentam vantagens e desvantagens:

Quadro 1. Formas de auditoria

FORMAS DE AUDITORIA	VANTAGENS	DESVANTAGENS
INTERNA	• maior conhecimento da estrutura de poder da empresa; • maior conhecimento dos processos da empresa; • maior conhecimento da cultura da empresa; • atuação constante.	• maior tendência a sofrer pressão da estrutura formal da empresa; • maior dificuldade de uma visão externa e independente.
EXTERNA	• menos tendência a sofrer pressão da estrutura formal da empresa; • maior facilidade para coordenar problemas; • maior independência.	• desconhecimento da estrutura de poder da empresa; • desconhecimento dos processos da empresa; • desconhecimento da cultura da empresa; • atuação não freqüente.
ASSOCIAÇÃO DA AUDITORIA INTERNA E EXTERNA	• associa as vantagens da auditoria interna e externa.	• custo alto; • possibilidades de divergências entre auditores internos e externos.

Fonte: [LOO96]

Qualquer uma das formas de auditoria permite que se atinjam os objetivos desejados. A escolha depende basicamente das características e cultura da empresa. Portanto, o fator determinante do sucesso da auditoria de sistema não é a forma utilizada, mas sim a disposição da empresa em implementar as mudanças sugeridas pelo auditor de sistemas.

Com a extensão e complexidade dos projetos de auditoria, juntamente com a necessidade de realização de auditoria, surgiram novas formas de auditoria, a saber:

a) Auto-Auditoria: aplicação dos instrumentos de auditoria pelos executivos e profissionais das diversas áreas da empresa, em revisão/avaliação/emissão de opinião de suas próprias atividades em cada área;

b) Auditoria Indireta via Informações Empresarias: o próprio auditor acessa os bancos de dados locais ou corporativos com o auditado;

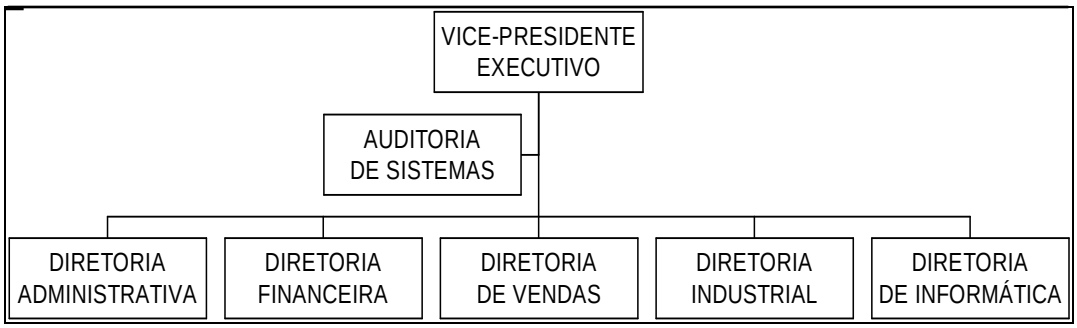
c) Auditoria à Distância: envio de questionários do auditor ao auditado, com o recebimento das respectivas respostas.

2.4 POSIÇÃO HIERÁRQUICA DA AUDITORIA DE SISTEMAS

Para que auditoria tenha um bom desempenho em sua função, caso a empresa opte por uma auditoria interna, a independência no posicionamento hierárquico é a principal chave.

Segundo [GIL98], a Auditoria de Sistemas torna-se mais efetiva quando atua a nível de assessoria, respondendo diretamente ao principal executivo da empresa, conforme figura a seguir:

Figura 1. Posição hierárquica da Auditoria de Sistemas (tradicional)



Fonte: [GIL98]

Nesta linha hierárquica, percebe-se que a Auditoria de Sistemas deve estar diretamente ligada à alta administração. No entanto, esta estrutura se depara com duas tendências atuais:

a) a terceirização das atividades das empresas;
b) a transformação do CPD em um órgão de assessoria.

A terceirização consiste no processo de transferir para terceiros algumas das atividades executadas pela empresa, para que esta se preocupe mais com o desenvolvimento de seus produtos e na prestação de serviços. Os serviços de limpeza e de vigilância foram as primeiras atividades a serem terceirizadas. Atualmente as atividades mais freqüentes para a terceirização têm sido: limpeza, transporte, vigilância, recepção, contabilidade, recursos humanos e serviços de informática, como desenvolvimento e suporte.

Em decorrência da terceirização dos serviços de informática, a tendência do CPD é se transformar em um órgão de Assessoria de Informática, cumprindo algumas responsabilidades, tais como [LOO96]:

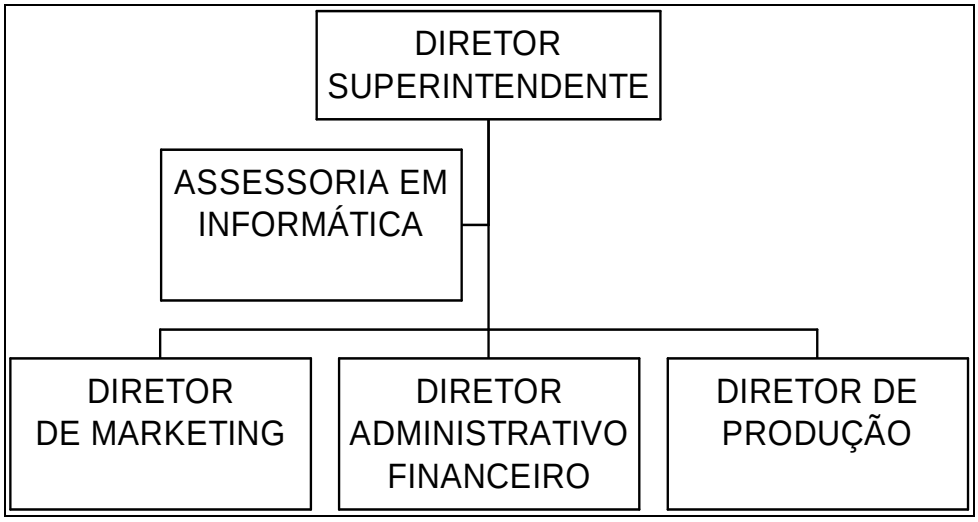
a) elaboração do Plano Diretor de Informática da empresa (PDI);

b) orientar a aquisição de softwares, hardwares e suprimentos;
c) realização da Auditoria de Sistemas e motivação dos usuários finais para a realização da auto-auditoria;
d) assessoria nos assuntos pertinentes à informática;
e) acompanhamento das tendências tecnológicas na área da informática.

Em função destas tendências, o tipo mais indicado de Auditoria de Sistemas é a auditoria externa combinada com uma auditoria interna. Esta auditoria interna poderá ser realizada pelo próprio usuário final, orientado e coordenado pela Assessoria em Informática.

Com base no texto exposto anteriormente chega-se ao organograma, conforme a figura 2:

Figura 2. Posição hierárquica da Auditoria de Sistemas (tendência)



Fonte: [LOO96]

A filosofia que embasa esta proposta, é que a motivação do usuário, quanto à necessidade e importância da Auditoria de Sistemas, é fundamental para que esta alcance seus objetivos. A auditoria externa será então utilizada periodicamente, de forma a se ter uma visão mais independente e abrangente do ambiente computadorizado.

2.5 AUDITOR DE SISTEMAS

O auditor é a pessoa designada pela alta administração da empresa, para realizar a Auditoria de Sistemas. O sucesso desta auditoria depende basicamente da competência e da capacitação do auditor de sistemas. Em primeiro lugar, o auditor de sistemas deve possuir um conjunto de conhecimentos que possibilitem o bom desempenho da Auditoria de Sistemas. Estes conhecimentos poder ser resumidos em três áreas básicas:

a) auditoria;

b) informática;

c) administração.

Os conhecimentos necessários para cada uma das áreas básicas são apresentados no quadro 2:

Quadro 2. Tipos de conhecimentos necessários ao Auditor de Sistemas

Áreas de conhecimento	Tipos de conhecimento
Auditoria	– uma metodologia de auditoria; – procedimentos a serem observados durante a realização da auditoria; – técnicas disponíveis; – formas de apresentação de resultados.
Informática	– metodologia de desenvolvimento de sistemas; – revisões e testes de sistemas; – normas de qualidade para software; – segurança; – hardware e software existentes; – tendências no desenvolvimento de hardware e software; – noções de redes de computadores.
Administração	– técnicas de O & M; – técnicas de motivação de seres humanos; – técnicas de avaliação de desempenho; – normas de qualidade.

Fonte: [LOO96]

Além dos conhecimentos técnicos, o auditor de sistemas deve possuir um “espírito de auditor”. Este “espiríto”, segundo [GIL98], envolve as seguintes características:

a) discrição;

b) objetividade;

c) facilidade na absorção de novos conhecimentos;
d) raciocínio lógico;
e) senso crítico para avaliar a situação do ambiente informatizado;
f) senso de organização;
g) capacidade de planejamento, de forma a planejar todo o processo de auditoria;
h) ética;
i) facilidade de relacionamento, de forma a manter um fácil relacionamento com profissionais de todos os níveis técnicos e de responsabilidades.

Independente das atribuições do cargo, existem responsabilidades básicas comuns a todos os auditores de sistemas, referentes a aspectos morais, de conduta, de relacionamento interpessoal e de ética profissional.

Segundo [FON91], é dever de todo auditor de sistemas:

a) zelar pela integridade moral, sua e do grupo;
b) desempenhar suas atividades com seriedade, honestidade, lealdade e profissionalismo;
c) respeitar, em qualquer situação, os colegas e superiores;
d) respeitar a hierarquia constituída;
e) promover e manter bom relacionamento com todos os membros do grupo e demais órgãos da empresa;
f) não se aproveitar de quaisquer informações conhecidas em função de seu trabalho;
g) não omitir qualquer informação que possa contribuir para o julgamento do setor, do sistema ou de seus funcionários;
h) não quebrar a confidencialidade das informações recebidas ou detectadas;
i) buscar o constante aprimoramento profissional, seu, do grupo e da empresa;
j) utilizar de forma racional os recursos que lhe foram confiados, visando minimizar seus custos.

Conforme [FON91], uma das tarefas mais difíceis do auditor de sistemas é a de promover a apuração de irregularidades. Embora não seja uma missão própria da Auditoria, na ausência de um departamento de inspeção, o auditor de sistemas é a pessoa mais indicada

para a execução desses trabalhos. Ele entende de processamento de dados, de controles internos, de segurança e de levantamento de responsabilidades.

A apuração de irregularidades objetiva:

a) promover o estancamento das falhas, a reversão dos desvios e dos seus efeitos;
b) a eliminação de fatores que tenham facilitado o evento, ou contribuído para a sua ocorrência;
c) a identificação das causas e dos envolvidos, e a avaliação do grau de responsabilidade de cada um destes.

Além de possuir um perfil técnico e psicológico adequado, torna-se essencial que o auditor de sistemas utilize um conjunto de métodos e técnicas que facilitem a execução da Auditoria de Sistemas.

2.6 METODOLOGIA DE AUDITORIA DE SISTEMAS

Uma metodologia eficiente adotada na execução de uma Auditoria de Sistemas, possibilita que os objetivos da respectiva auditoria sejam alcançados em sua totalidade.

Esta metodologia deve ser abrangente, de modo que possa ser utilizada nas três grandes áreas nas quais se subdivide a Auditoria de Sistemas, a saber: Auditoria de Sistemas em operação, Auditoria de Sistemas em desenvolvimento e auditoria em CPD [LOO96]:

a) Auditoria de Sistemas em Operação - consiste na avaliação e revisão dos processos e os resultados do sistema de informação, em operação pelos usuários sob os seguintes parâmetros: segurança física e lógica, fidelidade da informação em relação ao dado, confidencialidade, obediência à legislação em vigor, eficiência, eficácia e obediência às diretrizes administrativas. Os processos compreendem as rotinas e os módulos de programas que desenvolvem as atividades operacionais e de controle de processamento das informações. Os resultados compreendem aos dados e às informações contidas em documentos, formulários, relatórios e telas, ou armazenados nos arquivos em discos. Esse tipo de auditoria poderá ser realizada através das etapas de planejamento, levantamento do sistema a ser auditado,

inventário e eleição dos pontos de controle, avaliação dos pontos de controle, conclusão e acompanhamento da auditoria.

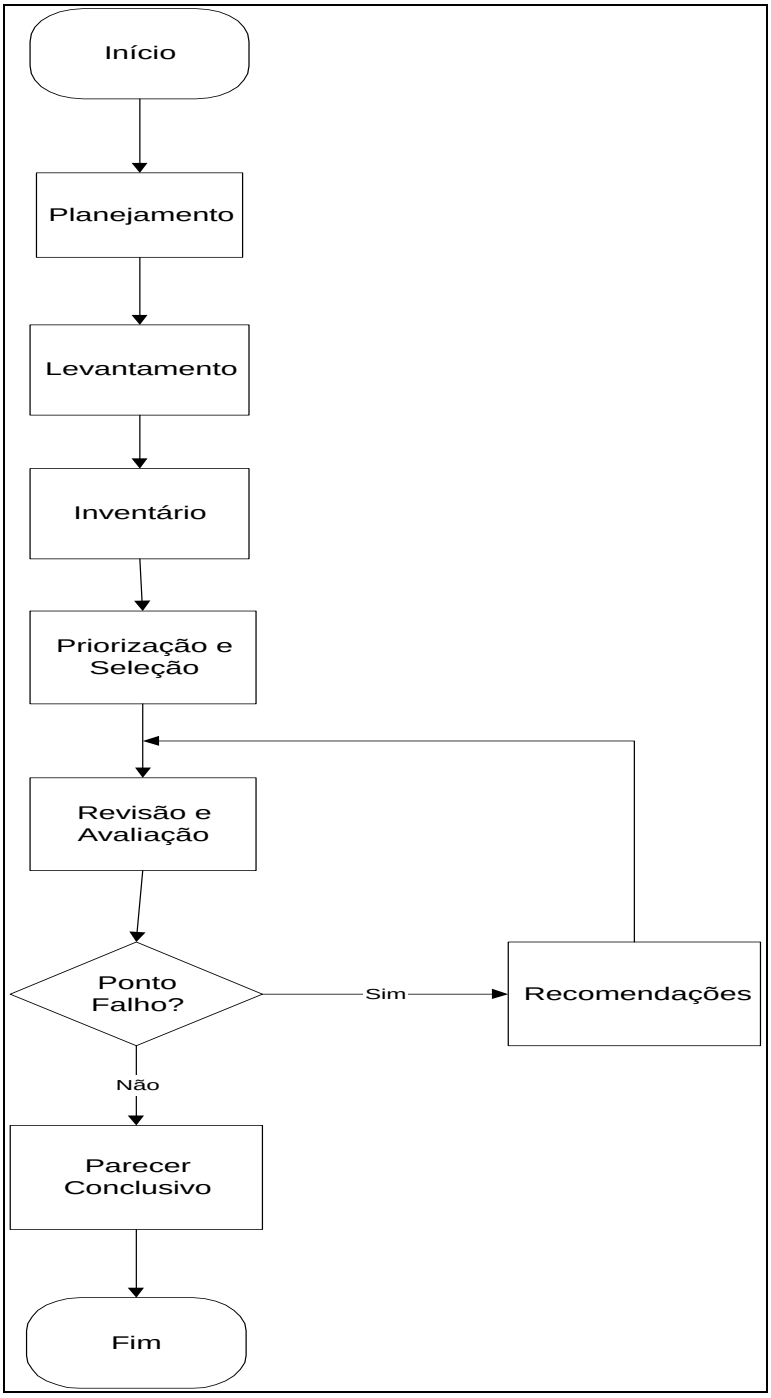
b) **Aditoria de Sistemas em Desenvolvimento -** consiste na avaliação e revisão dos sistemas de informação que estão sendo desenvolvidos, desde o levantamento e estudo de viabilidade do sistema até o teste de implantação do mesmo, sob os seguintes parâmetros: segurança física e lógica, confidencialidade, obediência à legislação em vigor, eficácia, eficiência e obediência às diretrizes administrativas. Esta auditoria poderá ser realizada através das etapas de planejamento, levantamento e caracterização do ambiente de desenvolvimento do sistema, inventário e eleição dos pontos de controle, avaliação dos pontos de controle e acompanhamento e conclusão de auditoria na implantação do sistema.

c) **Auditoria em CPD -** consiste na avaliação da estrutura física, administrativa e de recursos humanos do CPD da empresa. Além de avaliar também os sistemas em operação e/ou desenvolvimento, os equipamentos utilizados, entre outros, sob os seguintes parâmetros: eficácia, eficiência e segurança. Esta auditoria poderá ser realizada através das etapas de planejamento do projeto de auditoria do ambiente do CPD, levantamento e caracterização do ambiente, inventário e eleição dos pontos de controle, avaliação dos pontos de controle eleitos e conclusão e acompanhamento das recomendações.

2.7 METODOLOGIA TRADICIONAL

Existem algumas propostas de metodologias apresentadas por diversos autores. Neste trabalho será apresentada a metodologia sugerida por [ARI94], conforme figura 3.

Figura 3 - Etapas da Auditoria de Sistemas



Fonte: [ARI94]

2.7.1 PLANEJAMENTO

Esta etapa consiste em definir as necessidades de recursos humanos, tecnológicos, materiais e financeiros para desenvolvimento do projeto. Tais recursos devem ser dimensionados em função do enfoque, abrangência e delimitação do sistema a ser auditado em relação ao prazo estabelecido pela alta administração.

2.7.2 LEVANTAMENTO

Uma vez selecionado o sistema a ser auditado, inicia-se o processo de levantamento de toda a infra-estrutura da área de informática. O levantamento deve ser suficiente e abrangente para o entendimento pleno e global das características do sistemas.

2.7.3 INVENTÁRIO

A caracterização do sistema em questão, permite a identificação de diversos pontos de controle que merecem ser validados, esse processo denomina-se inventário de pontos de controle.

Os pontos de controle podem ser definidos e identificados pelos documentos de entrada, relatórios de saída, telas, arquivos magnéticos, rotinas e/ou programas de computador e demais elementos que compõem o sistema de informação.

2.7.4 PRIORIZAÇÃO E SELEÇÃO

Esta etapa consiste da priorização e seleção da execução da auditoria dos pontos de controle inventariados na etapa anterior, por parte do grupo de coordenação. A seleção dos pontos de controle para auditoria pode ser efetuada em função do grau de risco existente no ponto, em relação ao sistema como um todo.

2.7.5 REVISÃO E AVALIAÇÃO

Esta etapa consiste em executar testes de validação dos pontos de controle, segundo especificações de controle interno determinados para auditoria do respectivo sistema de informação. Esses testes implicam em aplicar técnicas de auditoria que evidenciam falhas ou fraquezas de controle interno.

2.7.6 PARECER CONCLUSIVO

Após o término dos testes de validação dos pontos de controle, quaisquer que sejam os resultados, deverão ser objetos de relatórios de auditoria, que conterão o diagnóstico ou a situação atual em que se encontram os pontos de controle, apontando fraquezas de controle interno, se houver, segundo especificações determinadas no respectivo sistema de informação.

A detecção de falhas de controle interno implica na necessidade de fazer recomendações com alternativas de solução, que minimizem ou até eliminem as fraquezas existentes.

Qualquer que seja o resultado da auditoria, é importante que os sistemas sejam reavaliados periodicamente, de modo a assegurar a sua proteção e manutenção de forma permanente.

3 NORMAS E MODELOS DE QUALIDADE

Com o progresso da tecnologia da informação, a quantidade de software vem crescendo e tornando essencial a qualidade de produtos de software. Um meio encontrado para tentar atingir a qualidade do software foi a criação de normas de qualidades.

Neste capítulo serão apresentados os conceitos e estruturas de algumas normas e modelos de qualidade de software.

3.1 NORMA ISO/IEC12207- PROCESSOS DE CICLO DE VIDA DE SOFTWARE

Segundo [HUG97] a norma NBR ISO/IEC 12207 - Processos de Ciclo de Vida de Software tem como principal objetivo o estabelecimento de uma estrutura comum para os processos de cicio de vida de software, para ser utilizada como referência. Além disso, a norma considera que o desenvolvimento e a manutenção de software devem ser conduzidos da mesma forma que a disciplina de engenharia.

A estrutura descrita na norma é composta de processos, atividades e tarefas a serem aplicados em operações que envolvam, de alguma forma, o software, seja através de aquisição, fornecimento, desenvolvimento, operação ou manutenção. Esta estrutura permite estabelecer ligações claras com o ambiente de engenharia de sistemas, ou seja, aquele que inclui práticas de software, hardware, pessoal e negócios.

Segundo [ABN97], os processos de cicio de vida são agrupados em três classes, que representam a sua natureza, descritos a seguir.

3.1.1 PROCESSOS FUNDAMENTAIS

Atendem ao início e à execução do desenvolvimento, operação ou manutenção dos produtos de software durante o ciclo de vida de software. São eles:

a) **Processo de Aquisição -** Define as atividades do adquirente, organização que adquire um sistema, produto de software ou serviço de software. Inicia-se com a definição da necessidade de adquirir um sistema, um produto de software ou um serviço de software. O Processo continua com a preparação e emissão de pedido de proposta, seleção de fornecedor e gerência do processo de aquisição através da aceitação do sistema, produto de software ou serviço de software;

b) **Processo de Fornecimento -** Define as atividades e tarefas do fornecedor, organização que provê o sistema, produto de software ou serviço de software adquirente. O processo pode ser iniciado tanto por uma decisão de preparar uma proposta para responder a um pedido de proposta de um adquirente quanto pela assinatura e celebração de um contrato com o adquirente para fornecer o sistema, produto de software ou serviço de software. O processo continua com a determinação dos procedimentos e recursos necessários para gerenciar e garantir o projeto, incluindo o desenvolvimento e a execução dos planos de projeto até a entrega do sistema, produto de software ou serviço de software para o adquirente;

c) **Processo de Desenvolvimento -** Define as atividades e tarefas do desenvolvedor, organização que define e desenvolve um produto de software. O processo contém as atividades para análise de requisitos, projeto, codificação, integração, testes, instalação e aceitação relacionada aos produtos de software;

d) **Processo de Operação -** Define as atividades do operador, organização que provê serviço de operação de um sistema computacional no seu ambiente de funcionamento para seus usuários. O processo cobre a operação do produto de software e o suporte operacional aos usuários;

e) **Processo de Manutenção -** Define as atividades do mantenedor, organização que provê o serviço de manutenção do produto de software, isto é, gerenciando as modificações no produto de software para mantê-lo atualizado e em perfeita operação. Este processo é ativado quando o produto de software é submetido a modificações no código e na documentação associada devido a um problema, ou à necessidade de melhoria ou adaptação. O objetivo é modificar um produto de software existente, preservando a sua integridade.

3.1.2 PROCESSOS DE APOIO

Auxiliam um outro processo e contribuem para o sucesso e qualidade do projeto de software. Um processo de apoio é empregado e executado, quando necessário, por outro processo. São eles:

a) Processo de Documentação - Define as atividades para registro da informação produzida por um processo de ciclo de vida. O processo de documentação contém o conjunto de atividades que planeja, projeta, desenvolve, produz, edita, distribui e mantém aqueles documentos necessários a todos os interessados, tais como gerentes, engenheiros e usuários do sistema ou produto de software;

b) Processo de Gerência de Configuração - É um processo de aplicação de procedimentos administrativos e técnicos, por todo o ciclo de vida de software, destinado a :

- identificar e definir os itens de software em um sistema além de estabelecer suas linhas básicas *(baseline)*;
- controlar as modificações e liberações dos itens;
- registrar e apresentar a situação dos itens e dos pedidos de modificação;
- garantir a completeza, a consistência e a correção dos itens;
- controlar o armazenamento, a manipulação e a distribuição dos itens.

c) Processo de Garantia da Qualidade - Define as atividades para garantir objetivamente que os produtos e processos de software estão em conformidade com seus requisitos especificados e aderem aos seus planos estabelecidos;

d) Processo de Verificação - É um processo para determinar se os produtos de software de uma atividade atendem completamente os requisitos ou condições impostas a eles nas atividades anteriores;

e) Processo de Validação - Define as atividades para validação dos produtos de software do projeto de software. É um processo para determinar se os requisitos e o produto final (sistema ou software) atendem ao uso especifico pretendido;

f) Processo de Revisão Conjunta - É o processo que avalia a situação e produtos de uma atividade de um projeto, se apropriado. As revisões conjuntas são feitas tanto

nos níveis de gerenciamento do projeto como nos níveis técnicos e são executadas durante a vigência do contrato;

g) Processo de Auditoria - Define as atividades para determinar a conformidade com requisitos, planos e contratos;

h) Processo de Resolução de Problemas - Processo para analisar e resolver os problemas (incluindo não conformidades), de qualquer natureza ou fonte, que são descobertos durante a execução do desenvolvimento, operação, manutenção ou outros processos. O objetivo é prover os meios em tempo adequado e de forma responsável e documentada para garantir que todos os problemas encontrados sejam analisados e resolvidos e tendências sejam identificadas.

3.1.3 PROCESSOS ORGANIZACIONAIS

São empregados por uma organização para estabelecer e implementar uma estrutura constituída de processos de ciclo de vida e pessoal associados, melhorando continuamente a estrutura e os processos. Eles são tipicamente empregados fora do domínio de projetos e contratos específicos. Entretanto, ensinamentos destes projetos e contratos contribuem para a melhoria da organização. São eles:

a) Processo de Gerência - Define as atividades e tarefas genéricas que podem ser empregadas por quaisquer das partes que têm que gerenciar seu(s) respectivo(s) processo(s). O gerente é responsável pelo gerenciamento de produto, gerenciamento de projeto e gerenciamento de tarefa do(s) processo(s) aplicável(eis), tais como aquisição, fornecimento, desenvolvimento, operação, manutenção ou processos de apoio;

b) Processo de Infra-estrutura - Define as atividades para estabelecer e manter a infra-estrutura necessária para qualquer outro processo. A infra-estrutura pode incluir hardware, software, ferramentas, técnicas, padrões e recursos para o desenvolvimento, operação ou manutenção;

c) Processo de Melhoria - Define as atividades básicas que uma organização (isto é, adquirente, fornecedor, desenvolvedor, operador, mantenedor, ou o gerente de outro processo) executa para estabelecer, avaliar, medir, controlar e melhorar um processo de cicio de vida de software;

d) Processo de Treinamento - Define as atividades para prover e manter pessoal treinado. A aquisição, o fornecimento, o desenvolvimento, a operação ou a manutenção de produtos de software é extremamente dependente de pessoal com conhecimento e qualificação. Portanto é essencial que o treinamento de pessoal seja planejado e implementado com antecedência para que o pessoal treinado esteja disponível quando o produto de software for adquirido, fornecido, desenvolvido, operado ou mantido.

A Norma também descreve o Processo de Adaptação que contém as atividades básicas para adaptar a Norma a uma organização ou projeto específico.

3.2 PROCESSO DE AUDITORIA SEGUNDO A NORMA ISO/IEC 12207

Conforme o item 6.7 da [ABN97], o Processo de Auditoria é definido como sendo um processo para determinar adequação aos requisitos, planos e contrato, quando apropriado. Este processo pode ser empregado por quaisquer das duas partes, onde uma parte (parte auditora) faz a auditoria nos produtos de software ou nas atividades da outra parte (parte auditada).

Este processo consiste nas seguintes atividades:

a) Implementação do processo;
b) Auditoria.

3.2.1 IMPLEMENTAÇÃO DO PROCESSO.

Esta atividade consiste nas seguintes tarefas:

a) As auditorias devem ser promovidas em marcos pré-determinados, conforme especificado no(s) plano(s) do projeto;
b) O pessoal da auditoria não deve ter nenhuma responsabilidade direta pelos produtos de software e atividades que eles auditam;
c) Todos os recursos requeridos para conduzir a auditoria devem ser acordados pelas partes. Esses recursos incluem pessoal de apoio, local, instalações, hardware, software e ferramentas;

d) As partes deveriam concordar com os seguintes itens em cada auditoria: agenda; produtos de software (e resultados de uma atividade) a serem revisados; escopo e procedimentos da auditoria; e critérios de início e término da auditoria;
e) Problemas detectados durante as auditorias devem ser registrados e incluídos no Processo de Resolução de Problema, quando requerido;
f) Após a conclusão de uma auditoria, os resultados da auditoria devem ser documentados e entregues à parte auditada. A parte auditada deve apresentar à parte auditora quaisquer problemas encontrados na auditoria e o planejamento das resoluções dos problemas relatados;
g) As partes devem concordar com o resultado da auditoria e quaisquer responsabilidades pelo item de ação e critérios de encerramento.

3.2.2 AUDITORIA

As auditorias devem ser conduzidas para assegurar que:

a) Produtos de software codificados (tais como item de software) reflitam a documentação do projeto;
b) A revisão de aceitação e requisitos de teste prescritos pela documentação estejam adequados para aceitação dos produtos de software;
c) Dados de teste estejam aderentes à especificação;
d) Os produtos de software sejam testados com sucesso e atendam às suas especificações;
e) Os relatórios de teste estejam corretos e discrepâncias entre o resultado real e o esperado foram resolvidos;
f) A documentação do usuário esteja aderente aos padrões, conforme o especificado;
g) As atividades sejam conduzidas de acordo com os requisitos, planos e contrato aplicáveis; e
h) Os custos e cronogramas adiram aos planos estabelecidos.

3.3 NORMA ISO 9000-3

Segundo [ANT95] e [NBR93], a NBR ISO 9000-3 define diretrizes a aplicação da ISO 9001 ao desenvolvimento, fornecimento e manutenção de software.

As diretrizes da ISO 9000-3 cobrem questões como: entendimento comum para as partes, contratante e contratada, de requisitos funcionais; uso de metodologias consistentes para o desenvolvimento de software e gerenciamento do projeto como um todo, da concepção até a instalação do software no cliente.

A ISO 9000-3 encontra-se dividida em três partes principais:

a) Estrutura do sistema de qualidade;
b) Atividades do ciclo de vida do software;
c) Atividades de suporte.

3.3.1 ESTRUTURA DO SISTEMA DE QUALIDADE

A aplicação da ISO 9000-3 deve propiciar aos fornecedores de software uma política de qualidade formal, documentada, divulgada e compreendida por todos os funcionários. Esses funcionários precisam ter como parte do seu trabalho a responsabilidade e a autoridade suficientes para implementar essas políticas. A empresa deve possuir pessoas e recursos para verificar, de forma independente, o emprego correto das suas políticas de qualidade. Ou seja, o executor não pode ser o seu próprio auditor.

As diretrizes da ISO 9000-3 definem também atribuições para o cliente. É proposto que o cliente indique um representante para negociar com o fornecedor de software as questões contratuais, incluindo definição de requisitos, definição de critérios de aceitação e acordos de conclusão.

São quatro os pontos cobertos pelo capítulo da norma que trata da Estrutura do Sistema de Qualidade, que estão descritos em [NBR93]:

a) Estabelecimento de responsabilidades gerenciais;
b) Definição e documentação do próprio sistema de qualidade;

c) Procedimentos para auditoria interna do sistema de qualidade;

d) Procedimentos para ações corretivas.

3.3.2 ATIVIDADES DO CICLO DE VIDA DO SOFTWARE

Faz parte do projeto de desenvolvimento de um software a definição do ciclo de vida em fases e, dentro destas fases, as atividades de cada uma delas. A norma define que o desenvolvimento de software deve ser feito segundo um determinado modelo de ciclo de vida, o qual pode ser estabelecido pela própria empresa.

Uma estrutura usual das fases deste ciclo de vida, segundo [ANT95], é a seguinte:

a) Fase 1 - Definição de requisitos;

b) Fase 2 - Projeto;

c) Fase 3 - Implementação;

d) Fase 4 - Teste;

e) Fase 5 - Liberação para produção;

f) Fase 6 - Liberação para embarque.

Independentemente do modelo de ciclo de vida definido pela empresa, a [NBR93] prevê que as atividades do ciclo podem ser agrupadas em nove grandes categorias, conforme segue:

a) **Análise Crítica de Contrato -** Cobre itens padrões que devem constar nos contratos relativos a compra e venda de software, tais como: abrangência do trabalho, contingências e proteção de informações proprietárias;

b) **Especificação de Requisitos do Comprador -** Trata da especificação de requisitos funcionais que devem ser preparados em conjunto pelo comprador e fornecedor. Deve incluir aspectos de performance, confiabilidade, segurança e privacidade;

c) **Planejamento do Desenvolvimento -** Enfatiza a necessidade e define um plano de desenvolvimento do software. O plano deve incluir a definição do projeto, organização dos recursos, fases, cronograma, planos de teste, formas de controle de entradas e saídas para cada fase do ciclo de vida e um método de monitorar e verificar o progresso;

d) **Planejamento da Qualidade -** Aborda a elaboração de um plano de qualidade específico para o projeto em pauta que englobe também itens não cobertos pelo sistema da qualidade geral da empresa, como requisitos ou atividade especiais previstas no contrato de um determinado cliente. Deve tratar dos objetivos de qualidade do produto de software, critérios de saída de cada fase e entrada na seguinte, planejamento detalhado de atividades de verificação e validação, bem como responsabilidades específicas para atividades de qualidade;

e) **Projeto e Implementação -** Preconiza um projeto de software disciplinado, onde o comprador e fornecedor concordam, previamente, sobre o conjunto de informações do projeto que serão fornecidas ao comprador. O projeto deve levar em consideração as futuras atividades de manutenção e aderir à regras e convenções de programação;

f) **Teste e Certificação -** Notifica a necessidade de teste e homologação do software em vários níveis. Um plano de testes deve ser sugerido cobrindo alguns fatores como: ambiente, documentação, cases e teste de dados. A validação do sistema completo e testes de campo devem também ser abordados pelo plano de testes;

g) **Aceitação -** Cobre os termos acordados previamente e condições impostas pelo comprador para aceitação do produto. São abordados nesta categoria, questões como: teste de aceitação, procedimentos para avaliação, ambientes e recursos de software e hardware;

h) **Reprodução, Expedição e Instalação -** Trata do registro de considerações relativas ao número de cópias, tipo de meio físico utilizado, direitos autorais e licenças, critérios de envio e obrigações do fornecedor e do comprador ligados à instalação;

i) **Manutenção**

Identifica a manutenção como uma questão da qualidade, onde esse serviço é incluso no contrato de compra. Atividades de manutenção normalmente envolvem mudanças no software, solução de problemas, correção de defeitos, modificação de interfaces, melhorias de desempenho e expansões funcionais. A Norma propõe a existência de um plano de manutenção, documentação e critérios de liberação em função da incorporação de alterações no software.

3.3.3 ATIVIDADES DE SUPORTE DO SISTEMA DE QUALIDADE

As atividades de suporte não se encontram atreladas a uma determinada fase do ciclo de vida do software. Elas permeiam todas as fases.

Essas atividades compreendem nove itens, os quais devem ser desenvolvidos e implementados pelo fornecedor do software [ANT95] [NBR93]:

a) **Sistema de gestão de configuração -** Deve fornecer um mecanismo para controle e rastreabilidade do software de modo que seja possível identificar de forma inequívoca cada versão, controlar a atualização simultânea do software por mais de uma pessoa, identificar e seguir todas as alterações resultantes de uma solicitação de alteração e assim por diante;

b) **Controle de documentos -** Estabelece que o fornecedor deve manter procedimentos para controlar toda a documentação relacionada à qualidade, exigida pelas normas. Esses procedimentos devem determinar quais documentos devem ser controlados, as instruções para sua alteração e, ainda, para retirada da central de documentos, entre outras;

c) **Registros da qualidade -** Indica que o fornecedor deve manter formas para identificar, coletar, indexar, arquivar, armazenar, manter e dispor dos registros da qualidade, de forma que sejam prontamente recuperáveis;

d) **Medição -** Trata-se das métricas e das técnicas de medição estabelecidas para realizar medições nos produtos e nos processos desde o desenvolvimento até a expedição;

e) **Regras, práticas e convenções -** Cada fornecedor de software deve definir suas regras, práticas e convenções desde que tornem efetivo o sistema de qualidade estipulado na ISO 9000-3;

f) **Ferramentas e técnicas -** Impõe como responsabilidade do fornecedor a utilização de ferramentas, recursos e técnicas que garantam a efetividade das diretrizes das Normas;

g) **Aquisição -** Cabe ao fornecedor de software certificar-se de que todos os produtos e serviços necessários para compor o produto final estão de acordo com os requisitos especificados, como forma de garantir a sua qualidade;

h) **Produto para ser incluído no software -** Trata basicamente dos cuidados relativos à qualidade para emprego de partes de softwares do próprio comprador que serão integradas para compor o software contratado;

i) **Treinamento -** Apresenta a responsabilidade do fornecedor relativa a identificação das necessidades de treinamento interno, visando à qualificação do próprio pessoal que executa tarefas que possam influenciar na qualidade.

3.4 PROCESSO DE AUDITORIA SEGUNDO A NORMA ISO 9000-3

Segundo [NBR93], o fornecedor deve implantar um sistema abrangente de auditorias internas da qualidade, planejadas e documentadas, para verificar se as atividades da qualidade estão em conformidade com a forma planejada e para determinar a eficácia do sistema da qualidade.

As auditorias devem ser programadas com base na situação atual e importância da atividade. As auditorias e as ações de acompanhamento devem ser executadas conforme os procedimentos documentados.

Os resultados das auditorias devem ser documentados e levados ao conhecimento do pessoal que tenha responsabilidade pela área auditada. O pessoal responsável pela administração da área deve tomar, em tempo hábil, ações corretivas referentes às deficiências encontradas pela auditoria.

3.5 MODELO CMM/SEI

Em meados de 1986 surge a tentativa de classificar os processos de desenvolvimento em níveis. Trata-se do Modelo de Capacidade e Maturidade do Software (CMM), desenvolvido pelo SEI (*Software Engineering Institute*) da Carnegie Mellon University.

O CMM tem como proposta implantar um processo de melhoria na qualidade do software, tendo como foco não apenas no produto, mas principalmente no processo. O

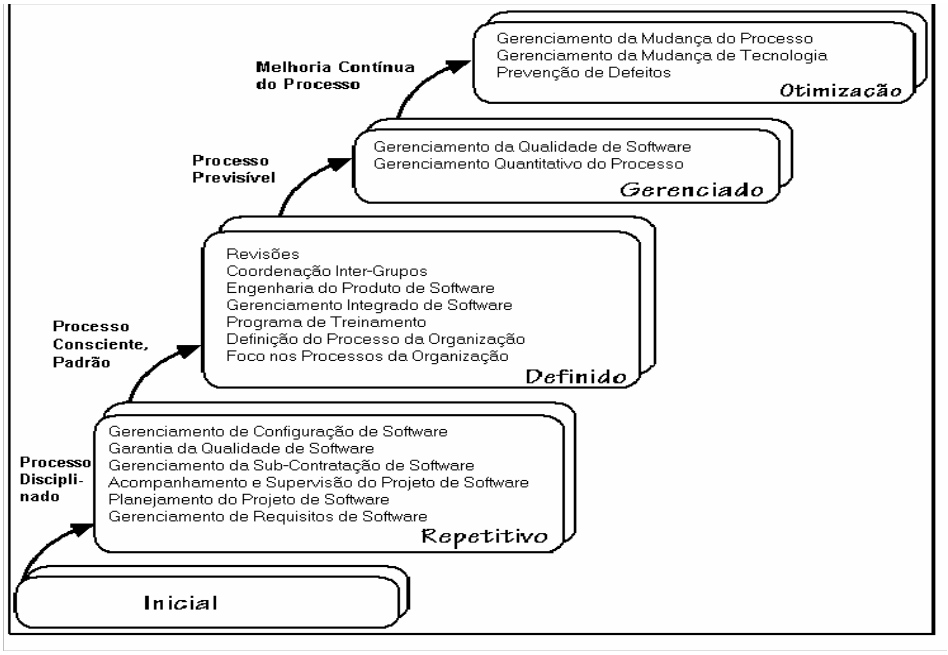
conceito básico que permeia esse modelo é a gerência dos processos através de práticas de gerenciamento, como planejamento, acompanhamento e controle de mudanças.

3.5.1 ESTRUTUTA DO MODELO CMM

Para entender melhor a estrutura do modelo CMM, é importante entender as palavras Maturidade e Capacidade. A capacidade do processo de software descreve a faixa de resultados que podem ser alcançados pela realização de um processo de software. A capacidade do processo de software de uma organização fornece meios para prever resultados a serem esperados num próximo projeto de software. A maturidade do processo de software é a extensão na qual um processo específico é explicitamente definido, gerenciado, mensurado, controlado e efetivo. Maturidade implica num potencial para crescimento em termos de capacidade e indica tanto a qualidade de um processo de software da organização quanto a consistência de sua aplicação [ANA96] [GRA97].

O modelo CMM permite avaliar um centro de desenvolvimento de software e classificá-lo em um dos cinco níveis crescentes de maturidade: inicial, repetitivo, definido, gerenciado e otimização. O nível de maturidade indica a capacitação do processo de software da organização, e a melhoria desse processo é baseada numa evolução gradual em cinco níves, em vez de uma transformação única. Os níveis de maturidade e suas áreas-chave do processo constam na figura 4:

Figura 4: Níveis de maturidade e suas áreas-chave de processo.



Fonte: [ANA96]

3.5.2 ÁREAS-CHAVE DE PROCESSO

Segundo [VAL99], cada nível de maturidade é composto de um certo número de áreas-chaves de processo (KPAs - Key Process Areas). Estas descrevem os objetivos que devem ser atingidos, assim como as questões a serem endereçadas para se alcançarem esses objetivos e atingir aquele nível. Uma organização está num determinado nível quando os objetivos das KPAs definidas para aquele nível e níveis anteriores forem atingidas.

A seguir serão descritas as áreas-chave em seus níveis e suas metas.

3.5.2.1 NÍVEL 1 - INICIAL

O processo de software é desestruturado e eventualmente caótico. Poucos processos são definidos e o sucesso depende de esforços individuais;

3.5.2.2 NÍVEL 2 - REPETITIVO

Processos básicos de gerenciamento de projeto são estabelecidos para planejar e acompanhar custo, cronograma e funcionalidade. Os processos são realizados de forma a repetir o sucesso de projetos anteriores em projetos similares. As áreas-chaves do processos são:

a) Gerenciamento de Requisitos de Software: O propósito é estabelecer um entendimento comum, entre o cliente e o projeto do software, dos seus requisitos que serão citados pelo projeto;

b) Planejamento do Projeto de Software: O propósito é estabelecer planos razoáveis para a execução de atividades da engenharia de software e para o gerenciamento do projeto de software;

c) Acompanhamento e Supervisão do Projeto de Software: O propósito é prover uma visão adequada do progresso real, de forma que o gerenciamento possa realizar ações corretivas quando a execução do projeto de software divergir significativamente dos planos de software;

d)Gerenciamento da Sub-contratação de Software: O propósito é selecionar sub-contratados de software qualificados e gerenciá-los eficientemente;

e) Garantia da Qualidade de Software: O propósito é prover um gerenciamento com uma visão apropriada dos processos utilizados pelo projeto de software e dos produtos construídos;

f) Gerenciamento de Configuração de software: O propósito é estabelecer e manter a integridade dos produtos do projeto de software ao longo do ciclo de vida do projeto de software.

3.5.2.3 NÍVEL 3 - DEFINIDO

O processo de software envolvendo as atividades de gerenciamento e engenharia está documentado, padronizado e integrado em um determinado processo de software da organização. Todos os projetos usam uma versão adaptada desse processo padrão para

desenvolver e manter produtos de software. As áreas-chaves do processo utilizadas nesse nível são:

a) Foco no processo da Organização: O propósito é estabelecer a responsabilidade organizacional para as atividades do processo de software que melhore a capacidade geral do processo de software da organização;

b) Definição do Processo da Organização: O propósito é desenvolver e manter um conjunto utilizável de recursos de processo de software que melhore o desempenho do processo através dos projetos e fornecer um histórico que traga benefícios à organização;

c) Programa de Treinamento: O propósito é desenvolver as habilidades e conhecimento dos indivíduos para que eles possam executar suas funções efetivamente e eficientemente;

d) Gerenciamento Integrado de software: O propósito é integrar as atividades de engenharia e gerenciamento de software em um processo de software coerente e definido, que é adaptado a partir do processo padronizado de software da organização e seus respectivos recursos;

e) Engenharia do Produto de Software: O propósito é executar consistentemente um processo bem definido de engenharia que integre todas as atividades de engenharia de software para produzir efetiva e eficientemente produtos de software corretos e consistentes;

f) Coordenação Inter-grupos: O propósito é estabelecer um meio para o grupo de engenharia de software participar ativamente com outros grupos de engenharia para que assim o projeto esteja mais capacitado a satisfazer as necessidades dos clientes de forma efetiva e eficiente;

g) Revisões: O objetivo é remover defeitos dos produtos intermediários de software antecipada e eficientemente. Uma importante conseqüência é desenvolver um melhor entendimento dos produtos intermediários de software e dos defeitos que podem ser prevenidos.

3.5.2.4 NÍVEL 4 - GERENCIADO

Métricas detalhadas do processo de software e da qualidade do produto são coletadas. Ambos os produtos e os processos de software são quantitativamente compreendidos e controlados. As áreas-chaves do processo utilizadas são descritas abaixo:

a) **Gerenciamento Quantitativo do Processo:** O propósito é controlar quantitativamente o desempenho do processo do projeto de software;
b) **Gerenciamento da Qualidade de software:** O propósito do Gerenciamento da Qualidade de Software é desenvolver um entendimento quantitativo da qualidade dos produtos de software do projeto e atingir metas específicas de qualidade.

3.5.2.5 NÍVEL 5 - OTIMIZAÇÃO

O processo deve ser melhorado através de um feedback quantitativo do processo e da incorporação controlada de novas tecnologias. As áreas-chaves do processo desse nível são:

a) **Prevenção de Defeitos:** O propósito é identificar a causa dos defeitos e prevenir que voltem a ocorrer;
b) **Gerenciamento da Mudança de Tecnologia:** O propósito é identificar novas tecnologias (por exemplo, ferramentas, métodos e processos) e realizar sua transição para a organização de maneira ordenada;
c) **Gerenciamento da Mudança do Processo:** O propósito é melhorar continuamente os processos de software utilizados na organização com o objetivo de melhorar a qualidade do software, aumentar a produtividade e diminuir o tempo para desenvolvimento do produto.

3.6 PROCESSO DE AUDITORIA SEGUNDO O MODELO CMM

Segundo [ANA96], o propósito da Auditoria de Software, que esta localizado no nível 2 - Garantia da Qualidade de Software, é prover um gerenciamento com uma visão apropriada dos processos utilizados pelo projeto de software e dos produtos construídos. Esta

área envolve a revisão e auditoria dos produtos de software, atividades para verificar o quanto eles atendem os procedimentos e padrões aplicáveis, e o fornecimento dos resultados destas revisões e auditorias ao gerente do projeto e outros gerentes envolvidos.

Metas:

a) atividades da garantia da qualidade de software são planejadas;

b) as atividades da garantia da qualidade de software verificam objetivamente se os produtos de software e atividades seguem padrões, procedimentos e requisitos aplicáveis;

c) os grupos e indivíduos envolvidos são informados quanto as atividades e resultados da garantia da qualidade de software;

d) discordâncias que não possam ser resolvidas dentro do projeto de software são enviadas à Gerência superior.

3.7 NORMA ISO/IEC 15504

Segundo [IAH99], a ISO/IEC 15504 é uma norma em elaboração conjunta pela ISO (*International Organization for Standardization)* e pelo IEC (*International Electrotechnical Comission*) e atualmente é conhecida como projeto SPICE (*Software Process Improvement and Capability dEtermination*). Ela constitui-se de um padrão para a avaliação do processo de software, visando determinar a capacitação de uma organização. A norma visa ainda orientar a organização para uma melhoria contínua do processo. Ela cobre todos os aspectos da Qualidade do Processo de Software.

Um grupo de estudos da ABNT (Associação Brasileira de Normas Técnicas) está participando do processo de desenvolvimento, além de trabalhar na tradução das versões preliminares da norma para o português.

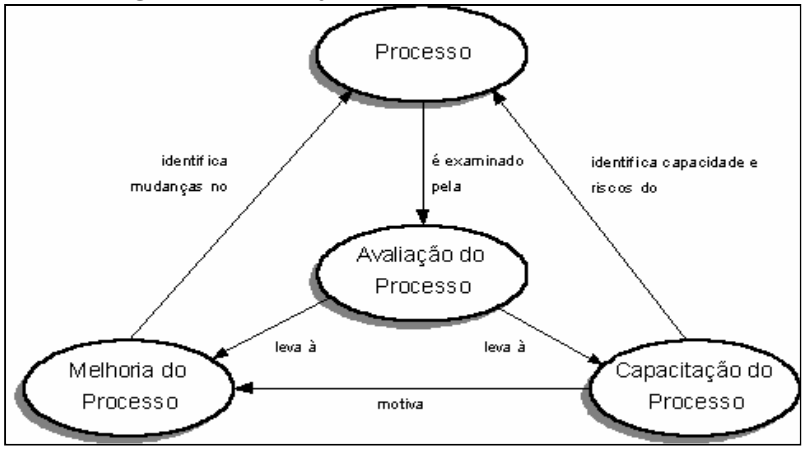
O projeto SPICE baseia-se nas melhores características de vários modelos de avaliação de processos existentes hoje, tal como: SW-CMM, Trillium, Software Technology Diagnostic (STD), Bootstrap e ISO 9001/9000-3. O resultado deste projeto será transformado na norma ISO/IEC 15504 (Tecnologia de Informação – Avaliação de Processos de Software) até o ano 2001.

3.7.1 PROPÓSITO DO SPICE

Segundo [EMA98], dentro da visão do SPICE (Figura 5), a avaliação de processos de software tem como propósito:

a) entender o estado dos processos de uma organização para a melhoria destes processos;
b) determinar a adequação dos processos de uma organização para um requisito particular ou uma classe de requisitos;
c) determinar a adequação dos processos de uma outra organização para um determinado contrato ou para uma classe de contratos.

Figura 5 – Avaliação de Processo de Software – SPICE.



Dentro do contexto da melhoria de processos, a avaliação significa a caracterização das práticas correntes de uma organização, unidade organizacional ou projeto em termos da capacidade dos processos selecionados. A análise dos resultados é feita em relação às necessidades de negócio da organização, identificando os aspectos positivos e negativos, e os riscos associados aos processos. Isto leva a determinar se os processos estão atingindo efetivamente seus objetivos e identificar causas da baixa qualidade, alto custo ou tempo excessivo, indicando a priorização na melhoria dos processos.

A determinação da capacidade dos processos de uma organização é feita através da comparação das capacidades de suas práticas contra um modelo para gerenciamento de processos, onde engloba atividades que se acredita serem fundamentais para uma boa engenharia de software. Essas atividades são estruturadas de modo a proporcionar um modelo

lógico do processo de software, identificando práticas que permitem o gerenciamento e melhoria de qualquer parte do processo ou do processo como um todo.

A complexidade do software moderno tem dificultado as companhias que desenvolvem, ou adquirem o software a identificar riscos, para controle de custos, melhoria da eficiência e da qualidade. Gerentes necessitam entender a capacidade de uma organização em desenvolver sistemas de software. Este entendimento, que o SPICE fornece, deve ser profundo o suficiente para habilitar a identificação dos riscos e promover melhorias na maneira que a organização produz software.

3.7.2 ESTRUTURA DO MODELO

O SPICE inclui um modelo de referência, que serve de base para o processo de avaliação. Este modelo é um conjunto padronizado de processos fundamentais, que orientam para uma boa engenharia de software. Este modelo é dividido em cinco grandes categorias de processo: Cliente-Fornecedor, Engenharia, Suporte, Gerência e Organização. Cada uma destas categorias é detalhada em processos mais específicos. Tudo isso é descrito em detalhes pela norma ISO/IEC 15504.

Além dos processos, o SPICE define também os 6 níveis de capacitação de cada processo, que podem ser incompleto, executado, gerenciado, estabelecido, previsível e otimização.

3.7.2.1 CLIENTE-FORNECEDOR

A categoria CLIENTE-FORNECEDOR consiste de processos que impactam diretamente com o cliente, tais como desenvolvimento de suporte, transição do software para o cliente e fornecimento de assistência/consultoria com relação à operação e uso do produto de software e/ou serviço. Os processos desta categoria são demonstrados a seguir:

a) **Processo de Aquisição:** obtenção do produto de software e/ou serviço que satisfaça as necessidades expressas pelo cliente. O processo inicia com a identificação das necessidades do consumidor e finaliza com a aceitação do produto e/ou serviço solicitado pelo cliente;

b) **Processo de Fornecimento:** fornece o software ao cliente conforme os requisitos concordados;
c) **Processo de Elicitação de Requisitos:** localiza problemas que venham interferir na vida do produto de software e/ou serviço, também para estabelecer uma linha que servirá para identificar a linha base do produto de software e/ou serviço;
d) **Processo de Operação:** define o processo de operação e entendimento do produto de software e fornecimento de suporte ao cliente.

3.7.2.2 ENGENHARIA

Esta categoria consiste de processos que especificam, implementam ou mantém o produto de software com relação ao sistema e documentação ao cliente. Seus processos estão descritos abaixo:

a) **Processo de Desenvolvimento:** transforma os requisitos de software anteriormente acordado em um produto de software funcional;
b) **Processo de Manutenção de Software e Sistema:** modificações, migrações e exclusões de componentes do sistema, são realizadas neste processo.

3.7.2.3 SUPORTE

Esta categoria consiste de processos que podem ser empregados por qualquer um dos outros processos. Estes processos são demonstrados abaixo:

a) **Processo de Documentação:** desenvolve e mantém documentos que registram informações produzidas por processos ou atividades;
b) **Processo de Gerenciamento de Configurações:** estabelece e mantém a integridade de todos os produtos ou projetos;
c) **Processo de Garantia da Qualidade:** assegura que os produtos e atividades de um processo ou projeto está conforme todos os padrões aplicáveis;
d) **Processo de Verificação:** confirma que cada produto de software e/ou serviço de um processo ou projeto estão de acordo com os requisitos;
e) **Processo de Validação:** confirma que os requisitos especificados estão de acordo com o software que foi trabalhado;

f) **Processo de Revisão Conjunta:** mantém um entendimento comum com o cliente do progresso do projeto contra os objetivos do contrato e que deve ser continuado para assegurar e desenvolvimento de um produto que satisfaça o cliente;
g) **Processo de Auditoria:** confirma independentemente que os produtos e processos empregados estão conforme os requisitos acordados;
h) **Processo de Resolução de Problemas:** assegura que todos os problemas descobertos são analisados e resolvidos;
i) **Processo de Medição:** coleta e analisa dados relativos aos produtos desenvolvidos e processos implementados dentro da unidade organizacional e demonstra objetivamente a qualidade dos produtos;
j) **Processo de Reuso:** promove e facilita o reuso de software.

3.7.2.4 GERÊNCIA

A categoria Gerência são processos que contém práticas de natureza genérica que podem ser usadas por quem gerencia projetos ou processos dentro de um ciclo de vida de software. Estes processos são apresentados a seguir:

a) **Gerenciar o projeto:** define os processos necessários para estabelecer, coordenar e gerenciar um projeto e seus recursos necessários para a produção do produto;
b) **Gerenciar a qualidade:** gerencia a qualidade dos serviços e dos produtos do projeto e assegura que os mesmos satisfaçam as necessidades do cliente;
c) **Gerenciar riscos:** identifica continuamente e alivia os riscos do projeto, desde de seu início e durante o ciclo de vida do mesmo;
d) **Gerenciar subcontratantes:** seleciona os subcontratados qualificados e gerencia a eficiência dos mesmos.

3.7.2.5 ORGANIZAÇÃO

Os processos descritos a seguir estabelecem os objetivos de negócios da organização:

a) **Construir o negócio:** suprir as pessoas na organização e projetos com uma visão e cultura que capacitem os mesmos para uma função eficiente;

b) **Definir o processo:** construir e reutilizar bibliotecas de definições de processos (incluindo padrões, procedimentos e modelos);
c) **Melhorar o processo:** melhoria contínua, efetiva e eficaz dos processos usados pela organização que estão em linha com as necessidades do negócio da mesma;
d) **Prover recursos de treinamento:** suprir a organização e os projetos com pessoas capazes e eficientes, fornecendo a elas treinamento;
e) **Prover infra-estrutura organizacional:** sustentar um ambiente estável e confiante, integrando métodos de desenvolvimento com ferramentas que auxiliem nos processos da organização.

3.7.2.6 NÍVEIS DE CAPACITAÇÃO

O SPICE, entretanto, não se limita a listar categorias e processos. Seu principal objetivo, na realidade, é avaliar a capacitação da organização em cada processo e permitir a sua melhoria. O modelo de referência do SPICE inclui seis níveis de capacitação. Cada um dos processos mencionados anteriormente deve ser classificado nestes níveis. Os níveis são descritos a seguir:

a) **Nível 0 - Incompleto:** há uma falha geral em realizar o objetivo do processo. Não existem produtos de trabalho nem saídas do processo facilmente identificáveis;
b) **Nível 1 - Realizado:** o objetivo do processo em geral é atingido, embora não necessariamente de forma planejada e controlada. Há um consenso na organização de que as ações devem ser realizadas e quando são necessárias. Existem produtos de trabalho para o processo e eles são utilizados para atestar o atendimento dos objetivos;
c) **Nível 2 - Gerenciado:** o processo produz os produtos de trabalho com qualidade aceitável e dentro do prazo. Isto é feito de forma planejada e controlada. Os produtos de trabalho estão de acordo com padrões e requisitos;
d) **Nível 3 - Estabelecido:** o processo é realizado e gerenciado usando um processo definido, baseado em princípios de Engenharia de Software. As pessoas que implementam o processo usam processos aprovados, que são versões adaptadas do processo padrão documentado;

e) **Nível 4 - Predizível:** o processo é realizado de forma consistente, dentro dos limites de controle, para atingir os objetivos. Medidas da realização do processo são coletadas e analisadas. Isto leva a um entendimento quantitativo da capacitação do processo a uma habilidade de predizer a realização.

f) **Nível 5 - Otimização:** a realização do processo é otimizada para atender às necessidades atuais e futuras do negócio. O processo atinge seus objetivos de negócio e consegue ser repetido. São estabelecidos objetivos quantitativos de eficácia e eficiência para o processo, segundo os objetivos da organização. A monitoração constante do processo segundo estes objetivos é conseguida obtendo *feedback* quantitativo e o melhoramento é conseguido pela análise dos resultados. A otimização do processo envolve o uso piloto de idéias e tecnologias inovadoras, além da mudança de processos ineficientes para atingir os objetivos definidos.

3.8 PROCESSO DE AUDITORIA SEGUNDO A NORMA ISO/IEC 15504

Segundo [ISO97], o objetivo do processo de auditoria é confirmar que os produtos e processo empregados estão em conformidade com os requisitos estabelecidos. Os resultados da auditoria devem assegurar que:

a) uma estratégia de auditoria deve ser desenvolvida e implementada;
b) auditorias devem ser mantidas em tempos pré-determinados;
c) a condução da auditoria de software executada por uma parte independente deve ser programada;
d) problemas detectados durante a auditoria devem ser identificados, resolvidos e comunicados a parte auditada.

3.9 COMPARATIVO DOS PROCESSOS DE AUDITORIA SEGUNDO AS NORMAS DE QUALIDADE

Após o estudo dos processos de auditoria nas normas ISO/IEC 12207, ISO 9000-3, SPICE e CMM, chegou-se a um comparativo, conforme quadro 3. Esse quadro apresenta o que cada norma diz a respeito do processo de auditoria. No quadro é apresentado a coluna **N.**,

que indica o número da linha, e as demais colunas indicam os nomes das normas de qualidade. O critério utilizado para especificar essa tabela foi separar cada item dos processo de auditoria das normas de qualidade e colocá-los lado a lado para fazer um comparativo para verificar se há pontos em comum nessas normas.

Quadro 3 - Comparativo dos processos de auditoria

N.	ISO/IEC 12207	ISO 9000-3	SPICE	CMM
1	Ao término da auditoria os resultados devem ser documentados e entregues a parte auditada.	Resultados devem ser documentados e levados para o pessoal responsável pela área auditada.		Pessoal envolvido na auditoria é informado dos resultados.
2	Problemas detectados durante a auditoria devem ser registrados e incluídos no Processo de Resolução de Problemas.	Pessoal responsável pela área auditada deve tomar ações corretivas referente deficiências encontradas na auditoria.	Problemas detectados durante a auditoria devem ser identificados, resolvidos e comunicados a parte auditada.	Discordâncias que não são resolvidas dentro do projeto são enviadas à gerência superior.
3	Auditorias devem ser promovidas em marcos pré-determinados, conforme especificado no plano do projeto.	Auditorias devem ser promovidas com base na situação atual e importância da atividade.	Auditorias devem ser mantidas em tempos pré-determinados.	Atividades de auditoria são planejadas.
4		Auditorias devem ser executadas conforme os procedimentos documentados.	Uma estratégia de auditoria deve ser desenvolvida e implementada.	

Quadro 4 - Continuação Comparativo dos processos de auditoria

5				Verificar se os produtos de software seguem padrões, procedimentos e requisitos aplicáveis.
6	Auditores não podem ter responsabilidades diretas sobre os itens que irão auditar.		A condução da auditoria de software executada por uma parte independente deve ser programada.	
7	Recursos requeridos para executar a auditoria devem ser acordados pelas partes.			
8	As partes devem concordar com os itens da auditoria.			
9	A parte auditada deve apresentar à parte auditora quaisquer problemas encontrados na auditoria.			
10	As partes devem concordar com o resultado da auditoria.			

Como na primeira linha da tabela os itens das normas refletem a mesma idéia, deve-se documentar e entregar a conclusão para parte auditada, chegou-se ao seguinte passo: **documentar** e **entregar conclusão.** Na segunda linha os processo dizem que os problemas encontrados devem ser registrados para posterior correção, então criou-se o seguinte passo: **erros encontrados na auditoria** e **resolução de problemas.** Na terceira linha as normas já aconselham que as auditorias devem ser marcadas com antecedência e com base na situação atual da atividade a ser auditada, por isso estabeleceu-se os seguinte passos: **auditoria**

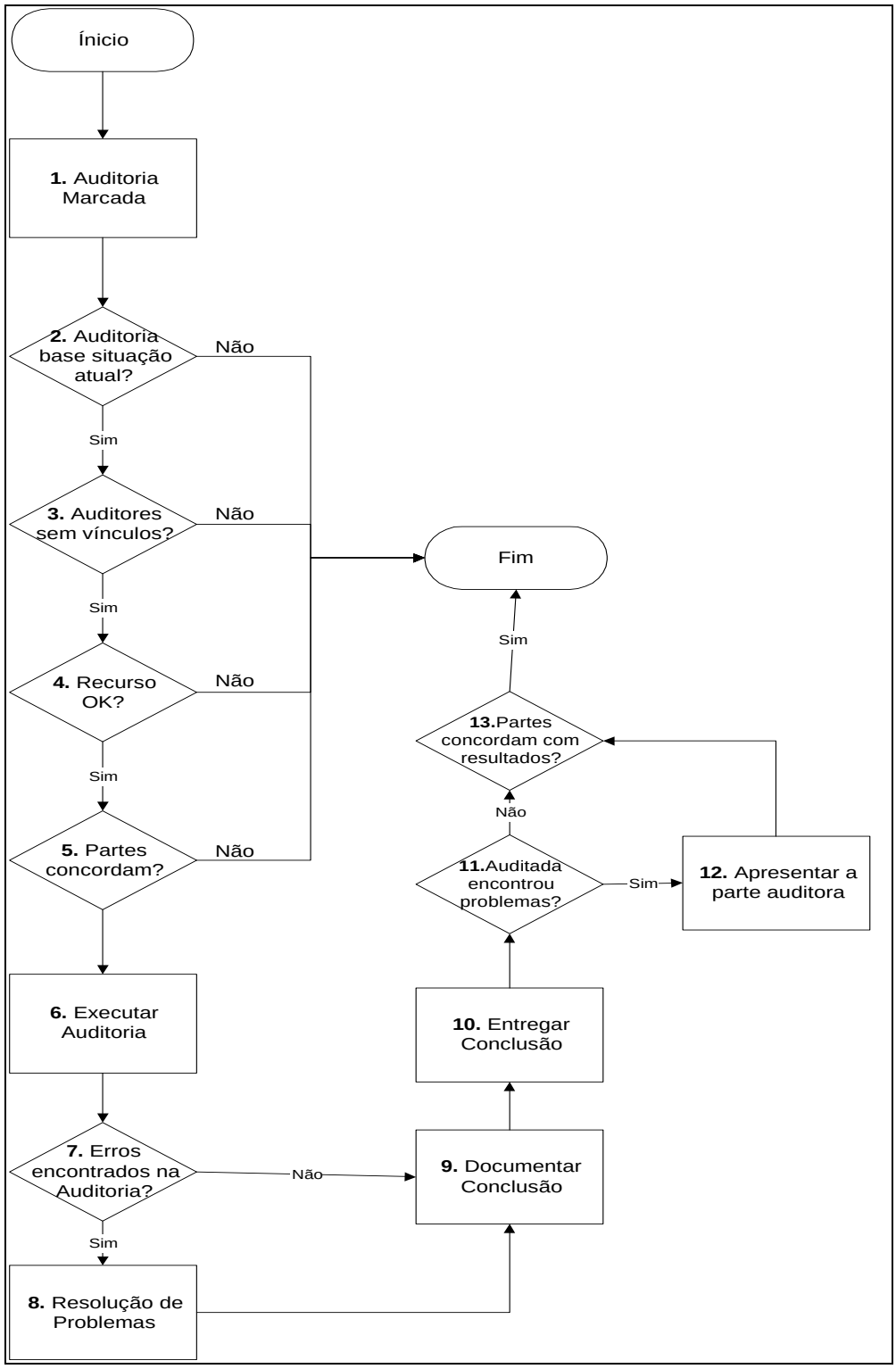
marcada e **auditoria com base na situação atual.** A quarta e quinta linha já se referem mais ao ato da auditoria em si, por isso criou-se o passo **executar auditoria.** A sexta linha já se refere ao auditor, onde ele não pode ter nenhum vínculo ou responsabilidade com área a ser auditada, por isso estabeleceu-se o passo **auditores sem vínculo.** A sétima linha refere-se ao fato de que os recursos requeridos para executar a auditoria devem ser acordados pelas partes, por isso criou-se o passo **recursos OK.** Estabeleceu-se o passo **partes concordam,** pois a oitava linha se refere ao fato de que as partes envolvidas devem concordar com os itens da auditoria. Na nona linha, a norma diz que a parte auditada deve apresentar à parte auditoria qualquer problema encontrado na auditoria, por isso criou-se o passo **auditoria encontrou problemas** e **apresentar a parte auditora.** E na última linha a norma aconselha que as partes envolvidas na auditoria devem concordar com o resultado da auditoria, por isso criou-se o passo **partes concordam com resultados.**

3.10 ROTEIRO DE AUDITORIA PROPOSTO

Após análise dos processos de auditoria, apresentados no quadro 3, das normas ISO/IEC 12207, ISO 9000-3, CMM e SPICE chegou-se ao conjunto de passos a serem concluídos para se executar uma auditoria. Esses passos foram estabelecidos somente através da teoria das normas, eles não foram testados e gerados a partir de experiências em empresas. Esses conjuntos de passos são demonstrados na figura 6 e posteriormente detalhados.

Para especificar esse roteiro usou-se a norma ISO/IEC 12207 como base, pois ela apresenta o processo de auditoria mais detalhado do que as demais.

Figura 6.- Roteiro de Auditoria



Para se entender melhor esses passos demonstrados no fluxograma da figura 6, tem-se um detalhamento a seguir:

1) **Auditoria marcada:** para iniciar uma auditoria esta por sua vez tem que ser planejada, marcada com antecedência, ela deve ser executa em marcos pré-determinados;
2) **Auditoria base situação atual:** a auditoria deve ser executada com base na situação atual e a importância da atividade que vai ser auditada;
3) **Auditores sem vínculo:** os auditores, para executar um auditoria, não deve ter nenhuma responsabilidade direta sobre os itens que serão auditados;
4) **Recursos OK:** todos os recursos requeridos para conduzir a auditoria devem ser acordados pelas partes. Esses recursos envolvem pessoal de apoio, local, instalações, hardware, software e ferramentas.
5) **Partes concordam:** as partes devem concordar com os seguintes itens: agenda, produtos de software a serem auditados, escopo e procedimentos da auditoria, critérios de inicio e término da auditoria;
6) **Executar auditoria:** esse passo consiste na execução da auditoria em si, o auditor irá executar a auditoria nas atividades planejadas conforme os procedimentos documentados;
7) **Erros encontrados na auditoria:** verificar se foram encontrados falhas nas atividades que foram auditadas durante a auditoria;
8) **Resolução de problemas:** neste item deverão ser relatados todos os problemas encontrados durante a auditoria;
9) **Documentar conclusão:** neste item deverão ser documentadas todas as conclusões obtidas na auditoria;
10) **Entregar conclusão:** a parte auditora deverá entregar a parte auditada as conclusões obtidas na auditoria;
11) **Auditada encontrou problemas:** verificar se a parte auditada encontrou problemas nas conclusões da parte auditora;
12) **Apresentar a parte auditora:** a parte auditada deve apresentar a parte auditora quaisquer problemas encontrados na auditoria e o planejamento das resoluções dos problemas relatados;

13) **Partes concordam com resultados:** as partes devem concordar com o resultado da auditoria e quaisquer responsabilidades pelo item de ação e critérios de encerramento.

4 ESPECIFICAÇÃO DO PROTÓTIPO

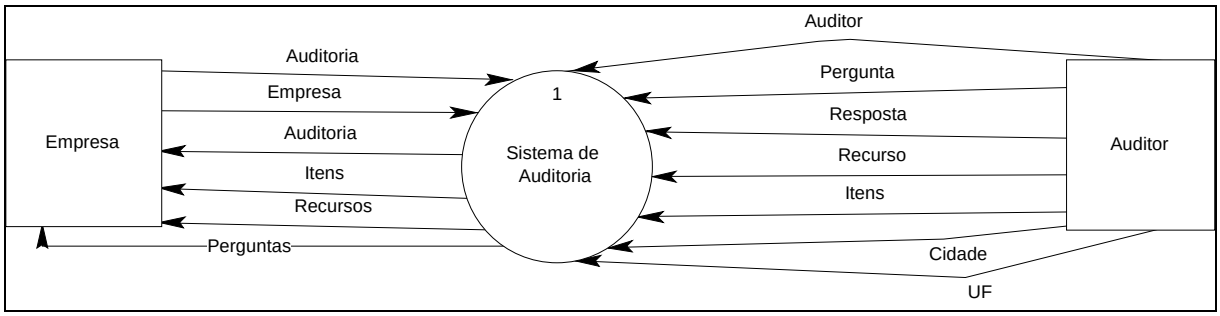
Com base no estudo das normas e modelos de qualidade, principalmente na norma ISO/IEC 12207 que se mostrou mais completa e na experiência do autor foi especificado e implementado o protótipo.

Para a especificação do protótipo utilizou-se a técnica de análise estruturada. A seguir são demonstrados o diagrama de contexto, modelo de entidade e relacionamento (MER) lógico e físico, e diagrama hierárquico funcional. As ferramentas utilizadas para a especificação e desenvolvimento do software foram a ferramenta CASE PowerDesigner 6.1 e ambiente Delphi 3.0.

4.1 DIAGRAMA DE CONTEXTO

Na figura 7 é apresentado o diagrama de contexto, gerado a partir da ferramenta CASE PowerDesigner 6.1. Aqui pode-se ter uma visão macro do sistema como um todo. O sistema é composto de duas entidades externas: auditor e empresa e treze fluxos de dados: auditoria, empresa, auditor, pergunta, resposta, recurso, itens, cidade, uf e os relatórios: auditoria, itens, recursos, perguntas. A notação utilizada é a de Yourdon.

Figura 7 - Diagrama de Contexto



4.2 MODELO DE ENTIDADE E RELACIONAMENTO

O Modelo Entidade Relacionamento (MER) enfatiza os principais objetos ou entidades do sistema. Na figura 8 é apresentado o MER Lógico e na figura 9 é apresentado o MER Físico, ambos gerados a partir da ferramenta CASE PowerDesigner 6.1. A notação utilizada é a de James Martin, também conhecida com “pé de galinha”.

Figura 8 - MER Lógico

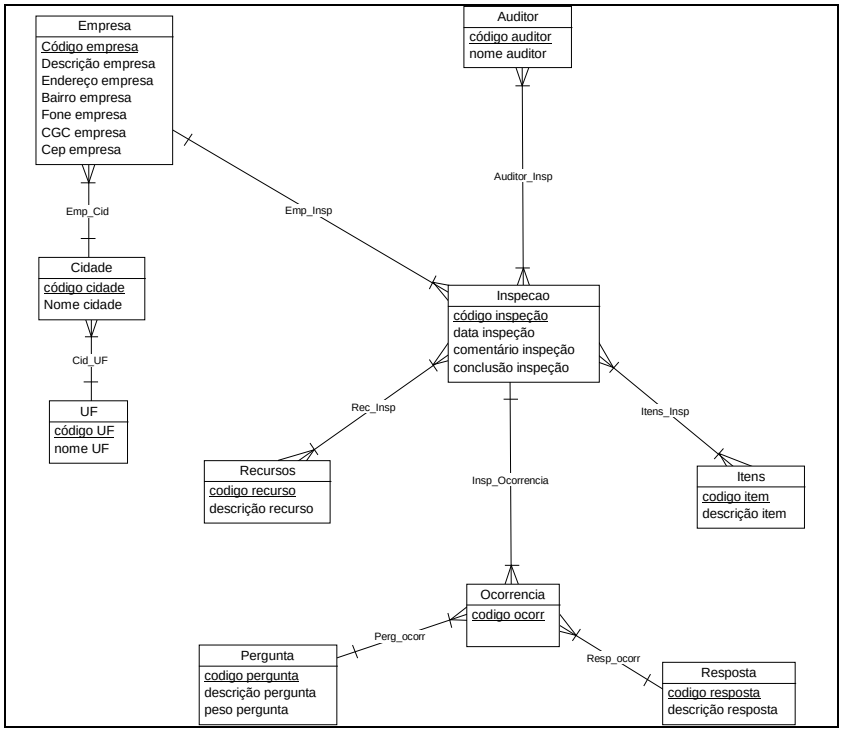
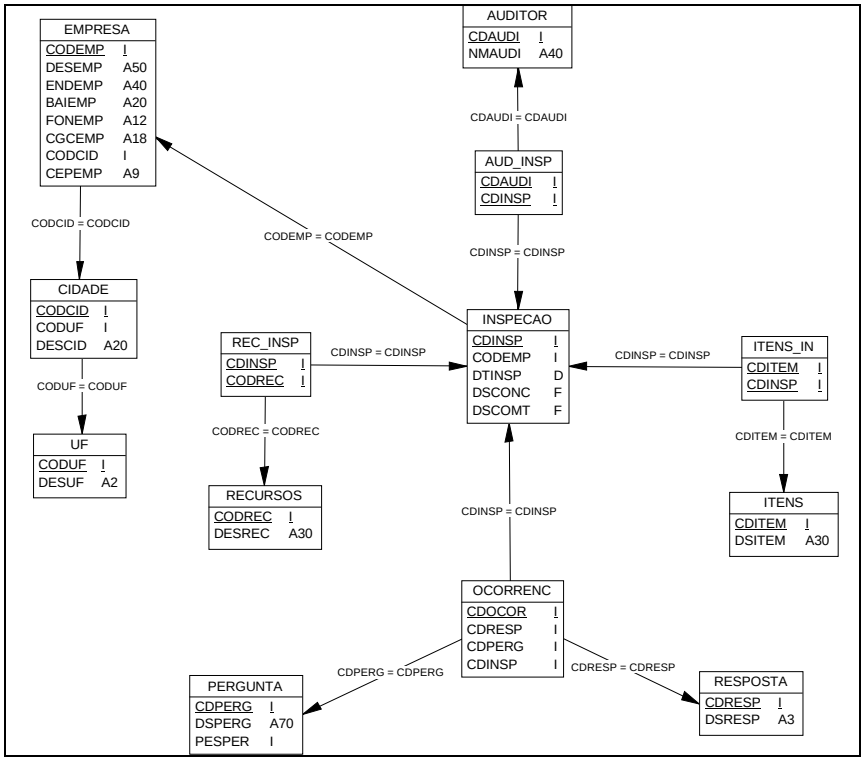


Figura 9 - MER Físico



4.3 DICIONÁRIO DE DADOS

O dicionário de dados consiste em uma descrição de todas as tabelas do sistema com seus respectivos atributos. As tabelas que mostram a descrição de cada tabela utilizada no protótipo foram geradas a partir da ferramenta CASE PowerDesigner 6.1. Na especificação foram usados os seguintes tipos de dados:

a) I: Integer;

b) A: Alfanunérico, seguido do tamanho do campo;

c) D: Data;

d) F: Arquivo texto.

As tabelas foram geradas em PARADOX. Nas tabelas geradas encontram-se as colunas **Name**, que mostra a descrição dos campos da tabela, **Code**, que apresenta os atributos de cada tabela, **Type**, que mostra os tipos de dados que serão armazenados, **P**, que indica se o atributo é chave primária e **M**, que indica se o atributo é obrigatório.

A seguir serão apresentadas as tabelas utilizadas no protótipo.

Figura 10 - Descrição das tabelas do protótipo

Erro! Indicador não definido.Aud_Insp

Column List

Name	Code	Type	P	M
código auditor	CDAUDI	I	Yes	Yes
código inspeção	CDINSP	I	Yes	Yes

Erro! Indicador não definido.Auditor

Column List

Name	Code	Type	P	M
código auditor	CDAUDI	I	Yes	Yes
nome auditor	NMAUDI	A40	No	No

Erro! Indicador não definido.Cidade

Column List

Name	Code	Type	P	M
código cidade	CODCID	I	Yes	Yes
código UF	CODUF	I	No	Yes
nome cidade	DESCID	A20	No	No

Erro! Indicador não definido.Empresa

Column List

Name	Code	Type	P	M
Código empresa	CODEMP	I	Yes	Yes
Descrição empresa	DESEMP	A50	No	No
Endereço empresa	ENDEMP	A40	No	No
Bairro empresa	BAIEMP	A20	No	No
Fone empresa	FONEMP	A12	No	No
CGC empresa	CGCEMP	A18	No	No
código cidade	CODCID	I	No	Yes
Cep empresa	CEPEMP	A9	No	No

Figura 11 - Continuação Descrição das tabelas do protótipo

Erro! Indicador não definido.Inspecao

Column List

Name	Code	Type	P	M
código inspeção	CDINSP	I	Yes	Yes
Código empresa	CODEMP	I	No	Yes
data inspeção	DTINSP	D	No	No
conclusão inspeção	DSCONC	F	No	No
comentário inspeção	DSCOMT	F	No	No

Erro! Indicador não definido.Itens

Column List

Name	Code	Type	P	M
codigo item	CDITEM	I	Yes	Yes
descrição item	DSITEM	A30	No	No

Erro! Indicador não definido.Itens_Insp

Column List

Name	Code	Type	P	M
codigo item	CDITEM	I	Yes	Yes
código inspeção	CDINSP	I	Yes	Yes

Erro! Indicador não definido.Ocorrencia

Column List

Name	Code	Type	P	M
codigo ocorr	CDOCOR	I	Yes	Yes
codigo resposta	CDRESP	I	No	Yes
codigo pergunta	CDPERG	I	No	Yes
código inspeção	CDINSP	I	No	Yes

Erro! Indicador não definido.Pergunta

Column List

Name	Code	Type	P	M
codigo pergunta	CDPERG	I	Yes	Yes
descrição pergunta	DSPERG	A70	No	No
peso pergunta	PESPER	I	No	No

Figura 12 - Continuação Descrição das tabelas do protótipo

Erro! Indicador não definido.Rec_Insp

Column List

Name	Code	Type	P	M
código inspeção	CDINSP	I	Yes	Yes
codigo recurso	CODREC	I	Yes	Yes

Erro! Indicador não definido.Recursos

Column List

Name	Code	Type	P	M
codigo recurso	CODREC	I	Yes	Yes
descrição recurso	DESREC	A30	No	No

Erro! Indicador não definido.Resposta

Column List

Name	Code	Type	P	M
codigo resposta	CDRESP	I	Yes	Yes
descrição resposta	DSRESP	A3	No	No

Erro! Indicador não definido.UF

Column List

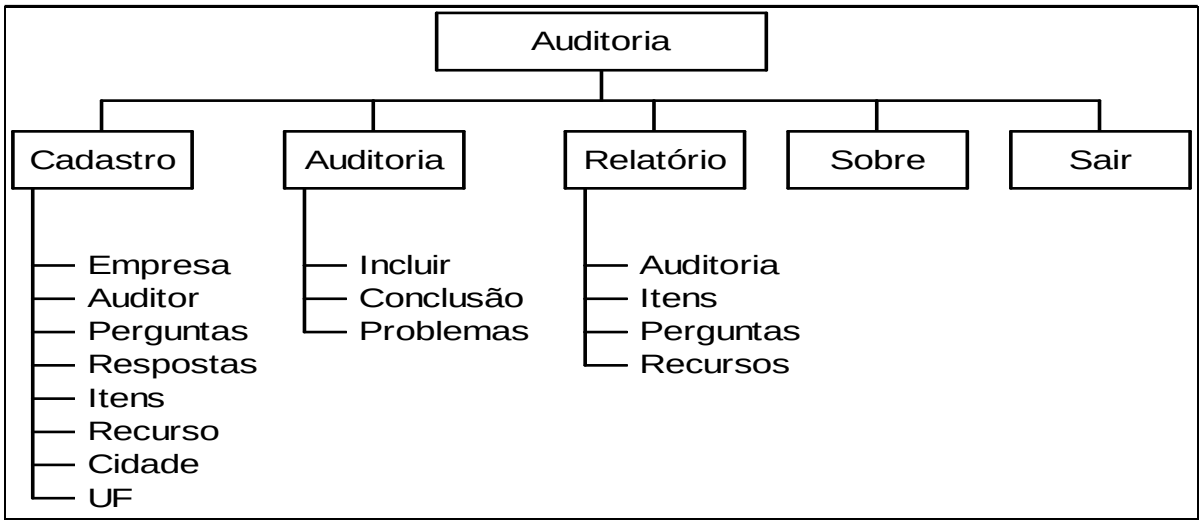
Name	Code	Type	P	M
código UF	CODUF	I	Yes	Yes
nome UF	DESUF	A2	No	No

4.4 DIAGRAMA HIERÁRQUICO FUNCIONAL

O diagrama hierárquico funcional do protótipo é apresentado na figura 13. Ele consiste das funções principais disponíveis pelo protótipo. Na função **“Cadastro”** estão os cadastramentos do sistema: Empresa, Auditor, Perguntas, Respostas, Recursos, Itens, Cidade e UF. Na opção **“Auditoria”** encontram-se os itens “Incluir”, onde ocorre todo o processo de auditoria da empresa, “Conclusão” onde é documentada a conclusão da auditoria e “Problemas” onde são documentados os problemas encontrados durante a auditoria. Na opção **“Relatório”** o usuário poderá imprimir a conclusão obtida da “Auditoria”, a relação de

“Itens” e “Recursos” utilizadas em uma auditoria, e a relação de “Perguntas” cadastradas no sistema. Na opção “Sobre” o tem-se as informações sobre o protótipo e na opção “Sair” o usuário poderá sair do protótipo.

Figura 13 – Diagrama Hierárquico Funcional



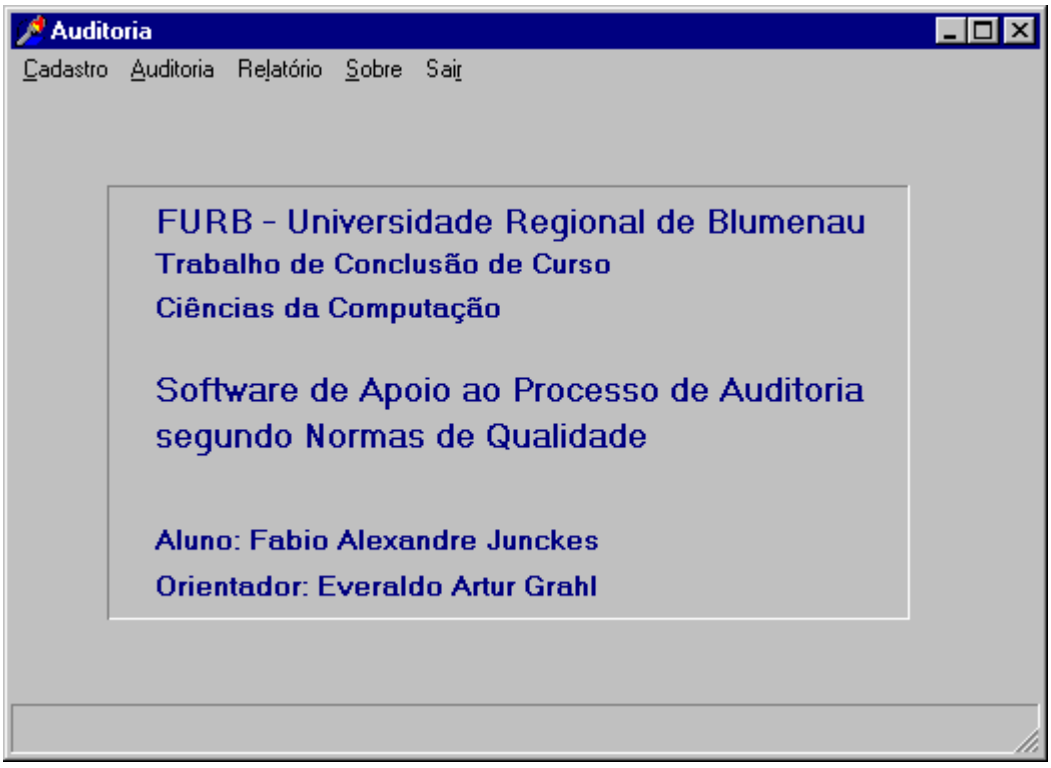
5 IMPLEMENTAÇÃO DO PROTÓTIPO

A seguir serão apresentadas as principais telas do protótipo.

5.1 TELA PRINCIPAL DO PROTÓTIPO

Ao executar o aplicativo, será apresentado a tela principal do protótipo que dará acesso aos demais recursos do mesmo, tais como cadastro de empresa, auditor, auditoria, relatório conforme figura 14.

Figura 14 - Tela principal do protótipo



Para se iniciar a auditoria em si, primeiramente todos os cadastros devem ser preenchidos. Os cadastros existentes serão demonstrados a seguir conforme figuras 15, 16, e 17.

5.2 TELA CADASTRO DE EMPRESAS

Para o cadastramento de empresas o usuário deverá clicar no botão Incluir e depois preencher os campos conforme figura 15.

Figura 15 - Tela de Cadastro de Empresas

5.3 TELA CADASTRO DE AUDITORES

Para cadastrar um auditor o usuário deverá clicar no botão Incluir e depois preencher os campos solicitados conforme figura 16.

Figura 16 - Tela Cadastro de Auditores

5.4 TELA DE CADASTROS

Para o cadastramento de perguntas, respostas, recursos requeridos para executar a auditoria, itens a serem acordados pelas partes envolvidas na auditoria, cidade e estado o usuário deverá clicar na página desejada e depois clicar no botão Incluir para então preencher os campos informados conforme figura 17. Na opção de cadastro de perguntas encontra-se o campo “Peso”, esse campo serve para que o auditor possa dar pesos as perguntas. Mas nesse protótipo não foi possível implementar esse cálculo de pesos nos relatórios.

Figura 17 - Tela de Cadastros

5.5 TELA INICIAL DE CADASTRO DE AUDITORIA

Para se iniciar o cadastro da auditoria, conforme figura 18, obrigatoriamente todos os cadastros já devem ter sido concluídos. O usuário deverá executar os seguintes passos:

a) clicar no botão Incluir;
b) selecionar a empresa a ser auditada;
c) informar a data da auditoria;
d) selecionar os auditores que executarão a auditoria;
e) selecionar o itens que deverão ser acordados pelas partes envolvidas na auditoria;
f) selecionar os recursos que serão necessários para executar a auditoria;
g) clicar nos itens que já foram concluídos.

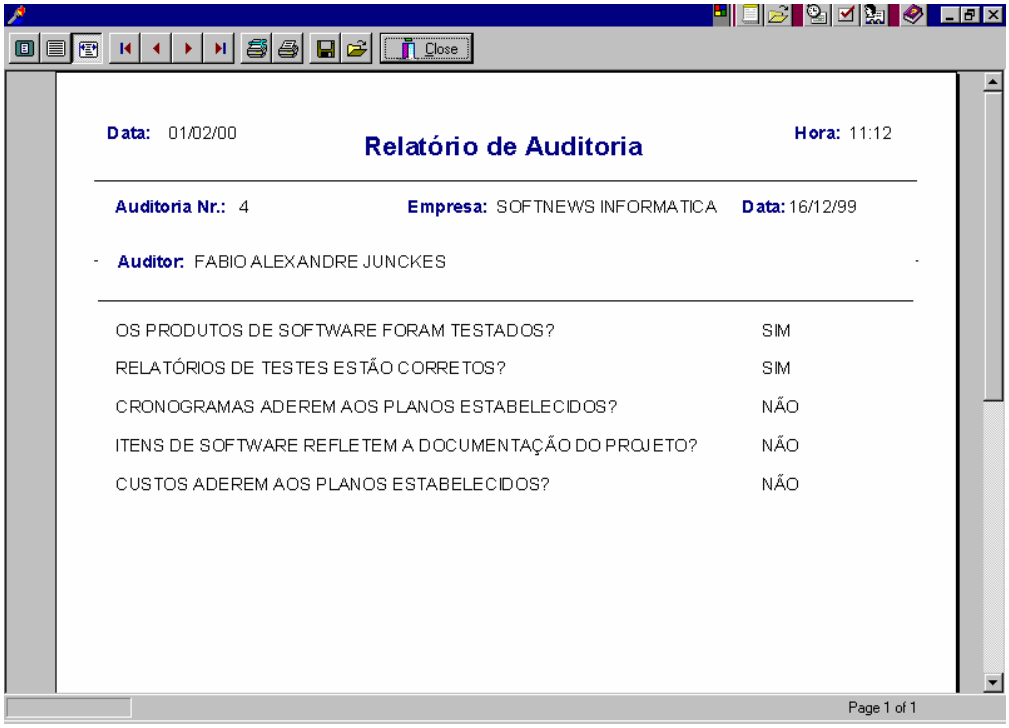
Após ter concluído os passos acima, o usuário deverá clicar no botão Próxima Página que então aparecerá uma tela para responder as perguntas. O usuário só vai poder passar para próxima tela se os campos do “Selecione os itens que já foram concluídos” estiverem todos selecionados. Essa tela atende aos passos 1, 2, 3, 4 e 5 do roteiro de auditoria proposto.

Figura 18 - Tela inicial do Cadastro de Auditoria

5.6 RELATÓRIO DE AUDITORIA

Conforme figura 19, neste relatório o auditor poderá ter uma relação de todas as perguntas e suas respectivas respostas referente a uma determinada auditoria.

Figura 19 - Relatório de Auditoria



5.7 RELATÓRIO DE ITENS

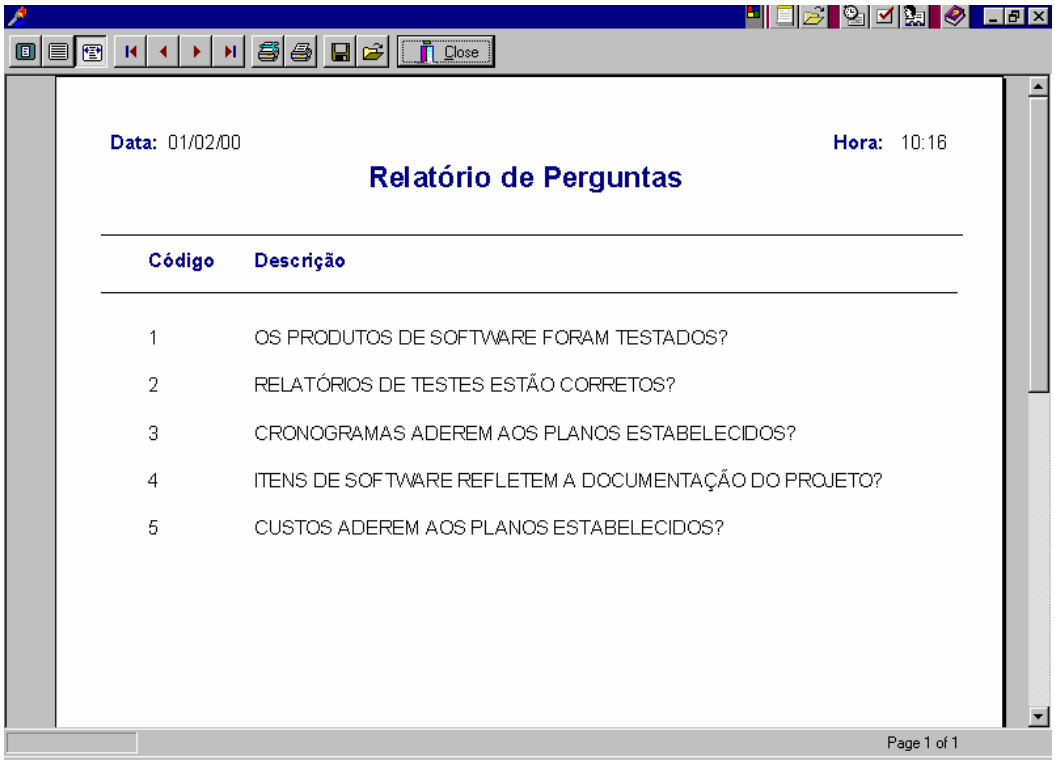
Conforme figura 20, neste relatório o auditor poderá ter uma relação de todos itens acordados em uma determinada auditoria

Figura 20 - Relatório de Itens

5.8 RELATÓRIO DE PERGUNTAS

Conforme figura 21, neste relatório o auditor poderá ter uma relação de todas as perguntas cadastradas no software.

Figura 21 - Relatório de Perguntas



5.9 RELATÓRIO DE RECURSOS

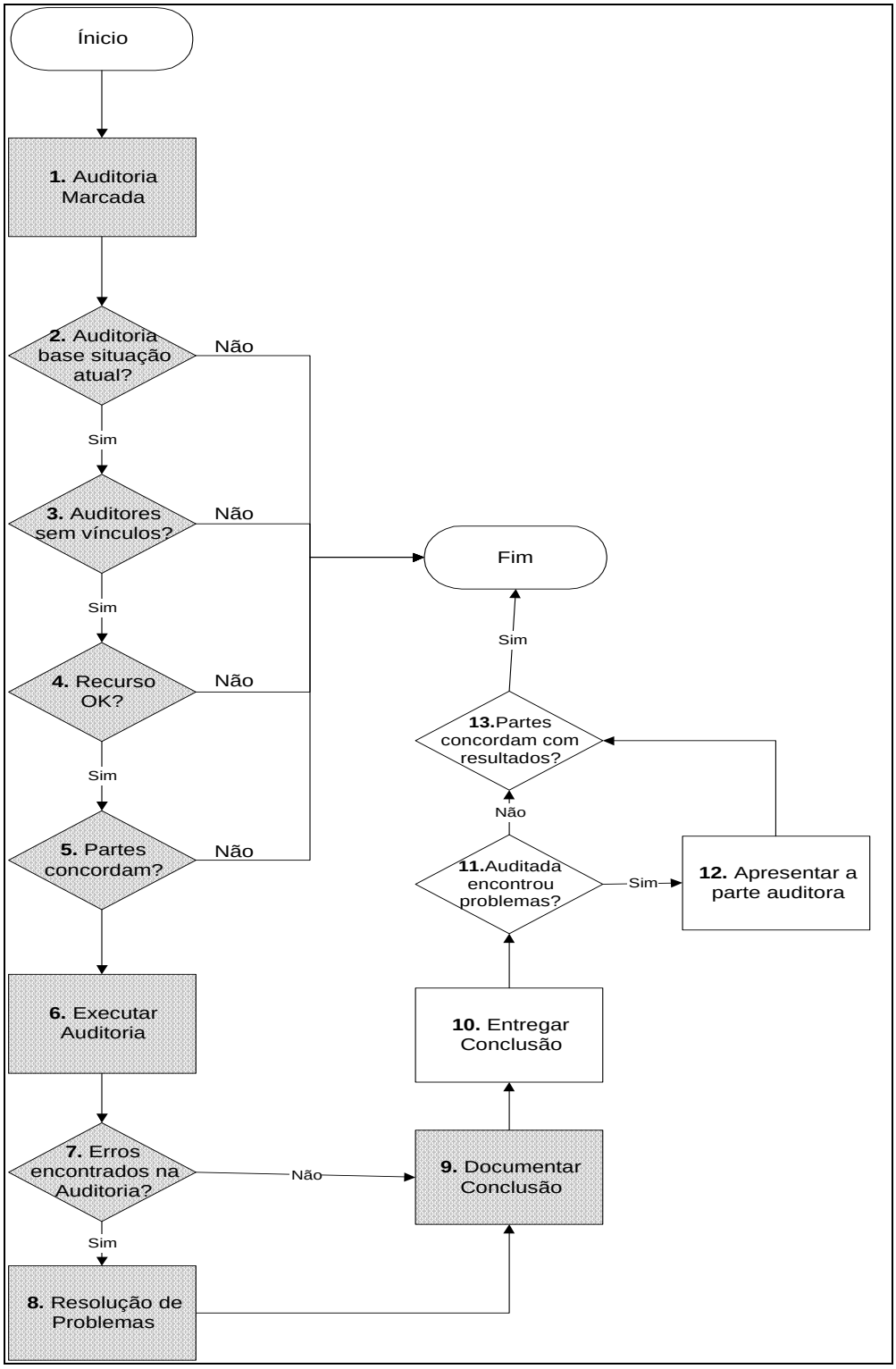
Conforme figura 22, neste relatório o auditor poderá ter uma relação de todos os recursos utilizados em uma determinada auditoria.

Figura 22 - Relatório de Recursos

5.10 CONSIDERAÇÕES FINAIS

Conforme figura 23, pode-se concluir que após a implementação do protótipo o software atingiu boa parte do roteiro proposto para auditoria apresentado no capítulo 3.10. Os passos do roteiro que foram suportados pelo protótipo estão em destaque na figura 23.

Figura 23 - Passos alcançados com o protótipo



6 CONCLUSÃO

Neste trabalho procurou-se analisar os processos de auditoria das normas de qualidade ISO/IEC 12207, ISO 9000-3, SPICE e o modelo CMM, e após essa análise especificar e implementar um software que auxiliasse os auditores na execução de auditoria segundo os passos especificados. Utilizou-se como base a norma ISO/IEC 12207, pois ela especifica com mais detalhes o processo de auditoria.

Este trabalho pode ser usado como fonte para o ensino de Auditoria de Sistemas visto que possui uma visão geral das normas e modelos de qualidade e suas recomendações para o processo de auditoria. Além disso o protótipo pode ser usado como guia para a realização de auditorias.

A principal dificuldade encontrada no decorrer do trabalho foi o fato de que as normas dizem o que fazer para se atingir as metas estabelecidas pelas mesmas, mas elas não dizem como fazer para se atingir essas metas.

Como sugestão para trabalhos futuros o software poderia ser testado em várias empresas que desejam realizar auditorias, visando uma maior validação do trabalho. Além disso poderiam ser implementados outros relatórios de ajuda ao auditor de sistemas.

GLOSSÁRIO

ABNT - Associação Brasileira de Normas Técnicas

CMM - Capability Maturity Model

IEC - International Electrotechnical Comission

ISO - International Organization for Standardization

SEI - Software Engineering Institute

SPICE - Software Process Improvement and Capability dEtermination

REFERÊNCIAS BIBLIOGRÁFICAS

[ANA96] ANACLETO, Ana Lúcia. **Mensuração do processo de software baseado no modelo CMM/SEI.** Blumenau : FURB, 1996. Trabalho de Conclusão de Curso.

[ABN97] ABNT - Associação Brasileira de Normas Técnicas. **NBR ISO/IEC 12207 - Processos do ciclo de vida de software,** NBR 12207. Rio de Janeiro, 1997.

[ANT95] ANTONIONI, José A.. **Qualidade em software - manual de aplicação da ISO-9000.** São Paulo : Makron Books, 1995.

[ARI94] ARIMA, Carlos Hideo. **Metodologia de auditoria de sistemas.** São Paulo : Érica, 1994.

[EMA98] EMAM, Khaled El. **SPICE: the theory and practice of software process improvement and capability determination.** Los Alamitos, Califórnia : IEE Computer Society, 1998.

[FON91] FONTES, Joaquim Rubens. **Manual de auditoria de sistemas.** Rio de Janeiro : Ciência Moderna, 1991.

[GIL98] GIL, Antônio de Loureiro. **Auditoria de computadores.** 3. Ed. São Paulo : Atlas, 1998.

[GRA97] GRAHL. Everaldo Artur. **Um comparativo entre o modelo CMM-SEI e a norma ISO/IEC 12207.** Anais: WQS97 - Workshop de Qualidade de Software. Fortaleza, 1997.

[HUG97] HUGO, Marcel. **Processos de ciclo de vida de software - Norma ISO/IEC 12207.** Anais: Seminco - Seminário Interno de Computação - Universidade Regional de Blumenau. Blumenau, 1997.

[IAH99] IAHN, Anisio. Avaliação de processos de software utilizando a norma ISO/IEC 15504. Blumenau : FURB, 1999. Trabalho de Conclusão de Curso.

[ISO97] ISO/IEC JTC1/SC7. **DTR 15504-2: Information technology - software process assessment. Part 2: A reference model for processes and process capability.** Proposta da NBR 15504. Canadá, 1997.

[LOO96] LOOS, Josemeire Geni. **Desenvolvimento de um software para auxiliar na execução de auditoria em CPD.** Blumenau : FURB, 1996. Trabalho de Conclusão de Curso.

[NBR93] NBR ISO 9000-3. **Normas de gestão da qualidade e garantia da qualidade - Diretrizes para aplicação da NBR 19001 ao desenvolvimento, fornecimento e manutenção de software,** NBR ISO 9000-3. Rio de Janeiro, 1993.

[VAL99] VALENTE, Fabio Feu Rosa. **A implantação do CMM de qualidade de software na Xerox.** Revista Developers. Rio de Janeiro, N. 35, P. 42-44, Jul. 1999