

MÓDULO DE RELACIONAMENTO ENTRE MODELOS DE REFERÊNCIA PARA MELHORIA DO PROCESSO DE SOFTWARE

Trabalho de Graduação

**Aluno:** Alexandre Barza – [ab@cin.ufpe.br](mailto:ab@cin.ufpe.br)

**Orientador:** PH. D. Alexandre Marcos Lins Vasconcelos – [amlv@cin.ufpe.br](mailto:amlv@cin.ufpe.br)

**Co-orientadora:** Tayanna C. Sotero – [tcs4@cin.ufpe.br](mailto:tcs4@cin.ufpe.br)

Recife, 20 de junho de 2008

**Resumo**

Um bom processo no desenvolvimento de software evita a presença de defeitos no produto. A implantação de um programa de qualidade começa pela definição e implantação de um processo de software. Diversos modelos de qualidade de software vêm sendo indicados ao longo dos últimos anos. Um modelo pode ser definido como uma representação abstrata de um item ou processo, a partir de um ponto de vista específico. A finalidade do modelo é expressar a essência de algum aspecto de um item ou processo, sem prover detalhes desnecessários **[3]**.

Os modelos de qualidade têm sido fortemente seguidos por organizações em todo o mundo, tais como: ISO 12207 **[17]**; ISO 15504 **[16]**; CMMI **[6]**; MPS-BR **[21]** e entre outros.

Alguns processos desses modelos e normas possuem semelhanças entre eles, podendo se relacionar e facilitar o seu avanço da qualidade. A grande parte de trabalhos que concerne ao mapeamento entre modelos de qualidade de software são definidos em documentos, desprovido de qualquer mecanismo que automatiza esse relacionamento. Diante dessa situação, foi construído um software que faz essa automatização, baseado no modelo e normas para a avaliação de processo, pois, além disso, este trabalho de graduação pode ter um papel importante, como um dos requisitos do módulo ProEvaluator do ambiente ImPPros.

**Abstract**

A good case in software development avoids the presence of defects in the product. The deployment of a quality program begins with the definition and implementation of a software process. Various models of quality of software have been shown in recent years. A model can be defined as an abstract representation of an item or process from a specific point of view. The purpose of the model is to express the essence of some aspect of an item or process, without providing details unnecessary. [Vasconcelos].

The models of quality have been strongly followed by organizations throughout the world, such as ISO 12207 **[17]**; ISO 15504 **[16]**; CMMI **[6]**; MPS-BR **[21]** and others.

Some of these processes and standards are similarities between them, are related and can ease their way of quality. A large part of work that comes to mapping between models of quality of software are defined in documents, devoid of any mechanism that automates this relationship. Given this situation, built a software that makes this automation, based on the model and standards for the evaluation process, therefore, also, this work for graduation can play an important role as one of the requirements of the module ProEvaluator environmental Imppros **[1]**.

**Agradecimentos**

A construção desse trabalho foi o resultado do acompanhamento da minha Co-orientadora, Tayanna Sotero, que contribuiu bastante com a evolução do projeto. Agradeço a ela; ao meu orientador, Alexandre Vasconcelos por ter me apoiado no desenvolvimento do trabalho; aos amigos que participaram indiretamente e aos meus familiares que sempre me deram suporte para poder continuar no ramo educacional.

Índice

[Capitulo 1 Introdução 6](#_Toc202758979)

[1.1 Contexto 6](#_Toc202758980)

[1.2 Motivação 7](#_Toc202758981)

[1.3 Objetivo 7](#_Toc202758982)

[1.4 Metodologia de Trabalho 8](#_Toc202758983)

[1.5 Estrutura do Trabalho 9](#_Toc202758984)

[Capítulo 2 Visão Geral de modelos e Normas de Qualidade de Software 10](#_Toc202758985)

[2.1 ISO/IEC 15504 11](#_Toc202758986)

[2.2 CMMI – Capability Maturity Model Integration 13](#_Toc202758987)

[2.3 MPS.BR - Programa de Melhoria do Processo de Software Brasileiro 16](#_Toc202758988)

[Capítulo 3 Ferramenta ProEvaluator 20](#_Toc202758989)

[3.1 ImPPros 20](#_Toc202758990)

[3.2 ProEvaluator 20](#_Toc202758991)

[Capítulo 4 Automação do Mapeamento 21](#_Toc202758992)

[4.1 Concepção do Modelo Genérico 22](#_Toc202758993)

[4.2 Cadastro de modelos 28](#_Toc202758994)

[4.3 Relacionamento dos itens dos modelos 38](#_Toc202758995)

[4.4 Arquitetura do Sistema 45](#_Toc202758996)

[4.5 Tecnologias 46](#_Toc202758997)

[Capítulo 5 Conclusões 47](#_Toc202758998)

[5.1 Considerações Finais 47](#_Toc202758999)

[5.2 Contribuições 48](#_Toc202759000)

[5.3 Trabalhos Futuros 48](#_Toc202759001)

[Referências Bibliográficas 49](#_Toc202759002)

# Capitulo 1 Introdução

Este capítulo apresenta o contexto no qual este trabalho foi elaborado e no qual o mesmo deve ser inserido. Em seguida, serão abordadas as razões que o motivaram. Com o fundamento no contexto e na motivação expostos, o objetivo é definido. Depois, a metodologia de trabalho e a estrutura de capítulos desta seção são descritas nesta ordem.

## Contexto

No grande mercado competitivo, As organizações de software vêm investindo cada vez mais na qualidade de seus produtos e serviços de software. A qualidade de software, dentre tantas definições, se baseia no conjunto de características a serem satisfeitas em um determinado grau de modo que o software satisfaça às necessidades de seus usuários. Para atingir determinado grau de qualidade, é necessário um controle de qualidade, ou seja, um conjunto planejado e sistemático de todas as ações necessárias para fornecer uma confiança adequada de que o item ou produto está de acordo com os requisitos técnicos estabelecidos. A atividade da garantia da qualidade abrange tanto a qualidade de processo quanto a qualidade do produto. Ambas as qualidades implicam no uso de um ambiente de Engenharia de software e de boa qualidade e adequado ao projeto. O interesse no processo de software está baseado em duas premissas:

1. A qualidade de um produto de software é fortemente dependente da qualidade do processo pelo qual ele é construído e mantido.
2. O processo de software pode ser definido, gerenciado, medido e melhorado.

Um bom processo evita a presença de defeitos no produto. A implantação de um programa de qualidade começa pela definição e implantação de um processo de software. Diversos modelos de qualidade de software vêm sendo propostos ao longo dos últimos anos. Um modelo pode ser definido como uma representação abstrata de um item ou processo, a partir de um ponto de vista específico. O objetivo do modelo é expressar a essência de algum aspecto de um item ou processo, sem prover detalhes desnecessários **[3]**.

Os modelos de qualidade têm sido fortemente adotados por organizações em todo o mundo, tais como: ISO 12207 **[17]**; ISO 15504 **[16]**; CMMI **[6]**; MPS-BR **[21]** e entre outros.

## Motivação

Diante de diversos modelos e normas de qualidade, muitos deles são utilizados em empresas de software, onde algumas dessas empresas utilizam mais de um modelo para atingir certo grau de melhoria do software. Alguns processos desses modelos e normas possuem semelhanças entre eles, podendo se relacionar e facilitar o seu avanço da qualidade. A grande parte de trabalhos que concerne ao mapeamento entre modelos de qualidade de software são definidos em documentos, desprovido de qualquer mecanismo que automatiza esse relacionamento. Diante dessa situação, foi construído um software que faz essa automatização, baseado no modelo e normas para a avaliação de processo, pois, além disso, este trabalho de graduação pode ter um papel importante, como um dos requisitos do módulo ProEvaluator do ambiente ImPPros **[1]**. Portanto, dentre essas razões, esse foi o motivo da elaboração de um projeto de graduação que faz referência ao relacionamento entre modelos.

## Objetivo

Este trabalho de graduação tem o objetivo de elaborar o relacionamento dos itens de cada modelo e norma de software, de modo que, quando estes itens forem cadastrados, a ferramenta automaticamente os relaciona. A ferramenta a ser utilizada pode servir como um dos módulos do ambiente ImPPros que foca a parte de avaliação do processo, denominada ProEvaluator.

## 1.4 Metodologia de Trabalho

Este trabalho de graduação está dividido em duas partes: A parte escrita e a parte de implementação do módulo de relacionamento entre os modelos de referências. A metodologia utilizada neste manuscrito se baseia na pesquisa qualitativa, ou seja, na leitura minuciosa de trabalhos, artigos e tutoriais relacionados ao modelo e normas de qualidade de software. As pesquisas permitiram uma comparação entre diversos modelos e normas que foram extraídos e adaptados neste trabalho, traçando seus aspectos mais relevantes.

Quanto à metodologia utilizada na confecção de software, foram utilizadas algumas práticas do “eXtreme Programming”, tais como:

1. Refatoração: é um processo que permite a melhoria contínua da programação, dividindo o código em módulos mais coesos e de maior reaproveitamento, evitando duplicação de código.
2. Integração contínua: responsável em integrar novas funcionalidades de forma contínua ao sistema, permitindo saber o status real da programação.

Além dessas práticas, outros métodos de boa programação, como dividir o sistema em camadas, possibilitou na mudança de camada sem afetar as outras; permitiu que a mesma versão de uma camada trabalhasse com diferentes versões de outra, como por exemplo, vários mecanismos de persistência suportados pela mesma aplicação. Isso facilitou bastante a implementação do software.

## Estrutura do Trabalho

O processo de relacionamento entre normas e modelos em um ambiente computacional está organizado em capítulos que abordam assuntos desde um overview sobre as normas e modelos de Qualidade de Software, como também a parte de implementação do módulo ProEvaluator. O trabalho contém cinco capítulos, distribuídos da seguinte forma:

* Capítulo 1: Apresenta em que contexto o projeto está inserido; a motivação que levou a sua elaboração; o objetivo do trabalho e a metodologia utilizada para o desenvolvimento do projeto.
* Capítulo 2: Aborda um estudo sobre os principais padrões e normas de qualidade de software. Uma breve explanação das características e processos do CMMI **[6]**, MPS.BR **[21]** e ISO/IEC 15504 **[16]** *.*
* Capítulo 3: Relaciona o módulo ProEvaluator com o ambiente de trabalho, ImPPros. Neste capítulo, será explicado sucintamente o ImPPros e a evolução do ProEvaluator.
* Capítulo 4: Explica o processo de automação do mapeamento, apresentando a arquitetura, as tecnologias e o modelos utilizados para a implementação do módulo ProEvaluator.
* Capítulo 5: apresenta a conclusão final de toda a pesquisa originada por este trabalho, incluindo as contribuições e as perspectivas de trabalhos futuros em relação ao tema pesquisado.

# Capítulo 2 Visão Geral de modelos e Normas de Qualidade de Software

Um Modelo de Referência de Processos de Software apresenta o que é reconhecido como boas práticas no desenvolvimento de produtos de software [IEEE, 2004]. Estas práticas podem estar somente relacionadas a atividades técnicas de engenharia de software, ou ainda referir também a atividades de gerência, engenharia de sistemas e gerência de recursos humanos.

O foco principal destes modelos é definir um conjunto de melhores práticas de Engenharia de Software que devem ser aplicadas de forma sistemática em projetos de desenvolvimento, com objetivo de se atingir um padrão estabelecido previamente de qualidade de software. Não define como o processo deve ser implementado, mas estabelece suas características estruturais e semânticas em termos de objetivos e do grau de qualidade com que o trabalho deva ser realizado.

Um modelo de processo de software pode ser definido através de níveis de maturidade e/ou capacidade. O conceito de Maturidade do Processo de Software surgiu através dos esforços do Software Engineering Institute (SEI) para atender a uma demanda da Força Aérea Americana, que necessitava de um método para avaliar a capacidade em desenvolver software das organizações que lhe prestava serviços terceirizados. Como entradas do levantamento inicial foram utilizados alguns trabalhos da IBM no qual pretendiam aplicar princípios da Gerência da Qualidade Total (TQM) para projetos de desenvolvimento de software.

Já o conceito de capacidade é o intervalo de resultados esperados que podem ser alcançados com o uso de um processo e a maturidade é a amplitude na qual um processo específico é definido, gerenciado, medido, controlado e executado, segundo Paulk **[27]**.

Nesta seção serão apresentados os modelos de referência de processos de software CMMI **[6]** e MR-MPS **[21]**, e ainda a norma internacional para avaliação de processos de software, a ISO/IEC 15504 **[16]**.

## 2.1 ISO/IEC 15504

A norma ISO/IEC 15504 foi oficialmente publicada em 2003 pela organização ISO. Esta norma surgiu a partir do projeto SPICE - Software Process Improvement Capacility dEtermination) que tinha como produzir inicialmente um Relatório Técnico que fosse mais geral e abrangente que os modelos existentes e mais específico que a norma ISO 9001 **[30]**. Uma versão do SPICE foi aprovada em 1998 como Relatório Técnico **[15]** e, apenas em 2003, a Norma ISO/IEC 15504 **[16]** foi publicada.

Outro importante objetivo da ISO/IEC 15504 é desenvolver uma norma internacional para avaliação dos processos de software **[28]**. ISO/IEC 15504 pode ser utilizada para a melhoria de processos e para a determinação da capacidade de processos de uma organização. Quando o objetivo da organização for a melhoria de processos, pode-se avaliá-los, gerando um perfil dos processos a ser utilizado na elaboração de um plano de melhorias. A análise dos resultados identifica os pontos fortes e fracos e os riscos inerentes aos processos. Já quando o objetivo da empresa for avaliar fornecedores para contratação, esta pode obter seus perfis de capacidade.

Seu modelo de referência define a dimensão de processo, que corresponde à definição de um conjunto de processos considerados universais e fundamentais para a boa prática da engenharia de software e a dimensão de capacidade, que corresponde à definição de um modelo de medição com base na identificação de um conjunto de atributos que permite determinar a capacidade de um processo para atingir seus propósitos, gerando os produtos de trabalho e os resultados estabelecidos **[24]**.

A norma ISO/IEC 15504 define 6 níveis de capacidade de processos seqüenciais e cumulativos (). Os 48 (quarenta e oito) processos são agrupados em 3 categorias (apoio, fundamentais e organizacionais) e a norma ainda estabelece as relações dos mesmos com fornecedores, clientes, equipe de desenvolvimento e equipe de suporte. Esta relação poderá ser visualizada na :

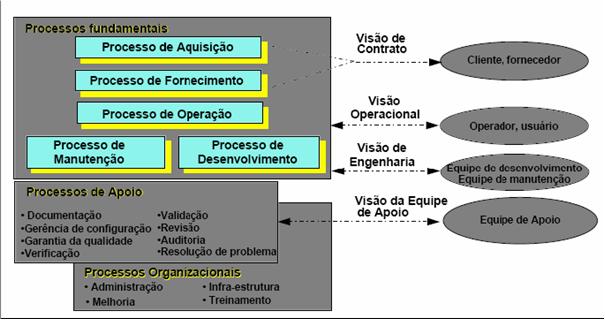


Figura 0.1 - Categorias de processos da ISO/IEC 15504 [30]

Um detalhamento destes processos poderá ser melhor visualizado na figura :

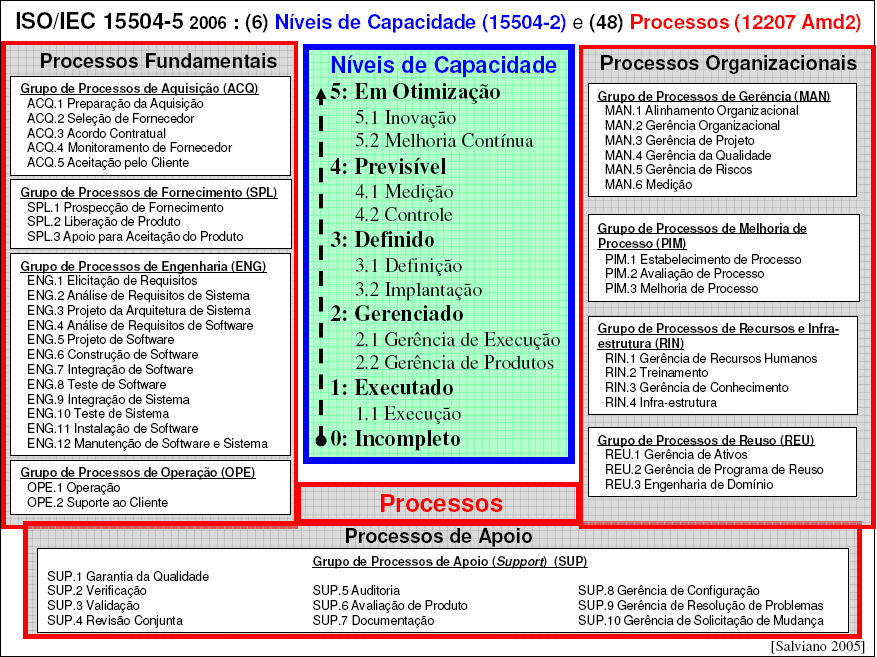


Figura 0.2 - Processos de ISO/IEC 15504 [30]

## 2.2 CMMI – Capability Maturity Model Integration

O surgimento do CMMI – Capability Maturity Model Integration **[6]** aconteceu devido ao aparecimento de vários modelos na década de 90. Dentre estes se destacavam:

* Software Acquisition CMM (AS-CMM) – utilizado para avaliar a maturidade de uma organização em seus processos de seleção, compra e instalação de software desenvolvido por terceiros.
* Systems Enginnering CMM (SE-CMM) – com foco na avaliação da maturidade da organização em seus processos de engenharia de sistemas, incluindo o hardware, o software e quaisquer outros elementos que participam do produto completo.
* Integrated Product Development CMM (IPD-CMM) – com abrangência maior que o SE-CMM, inclui também outros processos necessários à produção e suporte ao produto, tais como suporte ao usuário, processos de fabricação, etc.
* People CMM (P-CMM) – utilizado para avaliar a maturidade da organização em seus processos de administração de recursos humanos no que se refere a software: recrutamento e seleção de desenvolvedores, treinamento e desenvolvimento, remuneração, etc.

Diferenças de arquitetura, conteúdo e abordagens além do alto custo para treinamentos, avaliações e projetos de melhorias para as organizações se adaptarem a tantos modelos resultaram na criação do CMMI com o objetivo principal de integrar as práticas, de forma que, organizações que almejem melhorar seus processos nas diferentes disciplinas, tenham a disposição um único modelo consistente.

Sendo assim, o CMMI integra os diversos CMMs numa estrutura única, todos com a mesma terminologia, processos de avaliação e estrutura. O projeto também se preocupou em tornar o CMM compatível com a norma ISO/IEC 15504, de modo que avaliações em um modelo sejam reconhecidas como equivalentes aos do outro **[5]**.

Outro importante objetivo do CMMI é fornecer direcionamentos para melhorar os processos da organização e sua capacidade de gerenciar o desenvolvimento, aquisição e manutenção de produtos e serviços. O CMMI disponibiliza abordagens comprovadas em uma estrutura que auxilia as organizações a avaliar a sua maturidade organizacional ou a capacitação da área de processo, estabelecer prioridades de melhoria e implementá-las **[6]**.

O modelo CMMI oferece 2 (duas) abordagens de avaliação: por estágios (como o antigo SW-CMM) e contínua (baseada na ISO/IEC 15504). Estas abordagens podem ser visualizadas na :

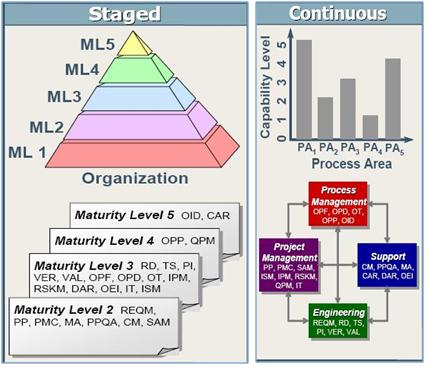


Figura 0.3 - Abordagens do modelo CMMI - Em estágios e Contínua

A abordagem em estágios oferece uma seqüência comprovada de melhorias, começando com práticas básicas de gerenciamento e progredindo por um caminho pré-definido e comprovado de níveis sucessivos, cada um servindo como base para o próximo, além de oferecer uma classificação única que resume os resultados de avaliações e permite comparação entre organizações.

Já a abordagem contínua, a organização escolhe a seqüência de melhorias que melhor atende aos seus objetivos de negócio e reduz suas áreas de risco. Esta abordagem possibilita comparações dentro e entre organizações em termos de áreas de processo **[6]**.

Os principais elementos do modelo são: área de processo, objetivos específicos, objetivos genéricos, práticas específicas, práticas genéricas, sub-práticas e produtos típicos de trabalho. Uma área de processo é agrupamento de práticas relacionadas que, quando implementadas coletivamente satisfazem um conjunto de objetivos considerados importantes para realizar uma melhoria significativa nesta área **[10]**.

Os objetivos específicos são obrigatórios e descrevem uma única característica que deve estar presente para que uma área de processo esteja satisfeita. Objetivos genéricos, também obrigatórios, são objetivos que estão distribuídos em várias áreas de processo e descrevem características que devem estar presentes para institucionalizar os processos que implementam determinada área de processo **[10]**.

As práticas genéricas e específicas, não são obrigatórias, mas desejáveis, e descrevem uma atividade considerada importante para atingir determinado objetivo. As sub-práticas fornecem uma descrição detalhada para interpretar e implementar uma prática. Produtos típicos de trabalho são apenas informativos, e fornecem uma lista de possíveis resultados de uma prática **[10]**.

## 2.3 MPS.BR - Programa de Melhoria do Processo de Software Brasileiro

O MPS.BR é um programa para melhoria do processo de software brasileiro que surgiu em Dezembro de 2003 com a finalidade de melhorar a qualidade do software que é produzido pelas empresas brasileiras, principalmente pequenas e médias, a um custo mais acessível.

Este programa é coordenado pela Associação para Promoção da Excelência do Software Brasileiro (SOFTEX), contando com apoio do Ministério da Ciência e Tecnologia (MCT), da Financiadora de Estudos e Projetos (FINEP) e do Banco Interamericano de Desenvolvimento (BID) e ainda conta com o apoio da indústria representada COPPE/UFRJ (Programa de Engenharia de Sistemas e Computação da Universidade Federal do Rio de Janeiro), CESAR (Centro de Estudos e Sistemas Avançados de Recife), CenPRA (Centro de Pesquisas Renato Archer) e organizações integrantes do Sistema SOFTEX do Rio de Janeiro e Campinas.

O MPS.BR é compatível com os padrões de qualidade aceitos internacionalmente e tem como pressuposto o aproveitamento de toda competência existente nos padrões e modelos de melhoria de processo já disponíveis. Ele é baseado nas abordagens internacionais para definição, avaliação e melhoria de processos de software. A base técnica do MPS.BR é composta pelas seguintes normas: ISO/IEC 12207 **[17]** e suas emendas 1 e 2, ISO/IEC 15504 – SPICE e o CMMI-SE/SW **[5]**.

O programa é constituído de 3 (três) componentes: a) Modelo de Referência (MR-MPS) **[21]**, b) Método de Avaliação (MA-MPS) **[22]** e o Modelo de Negócio, conforme pode ser visualizado na :

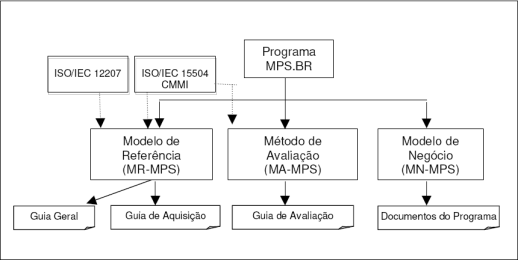


Figura 0.4 - Componentes do programa MPS.BR [21]

O MR-MPS segue a abordagem em estágios baseada no modelo CMMI, contudo existem mais níveis, de forma que as organizações podem gradualmente serem avaliadas, diminuindo assim o custo da avaliação. Os níveis previstos pelo MPS.BR podem ser visualizados na :

Figura 0.5 - Níveis do MR-MPS

Os processos são descritos em termos de propósito, resultados e informações adicionais. O propósito descreve o objetivo geral da execução do processo. Os resultados esperados estabelecem as metas a serem atingidas com a efetiva implementação do processo. As informações adicionais são referências que podem ajudar na definição do processo pela organização. Elas fornecem descrições de atividades, tarefas e melhores práticas que podem apoiar a definição e implementação do processo nas organizações. Os processos foram divididos em 3 classes, conforme o seu objetivo principal no ciclo de vida de software ().

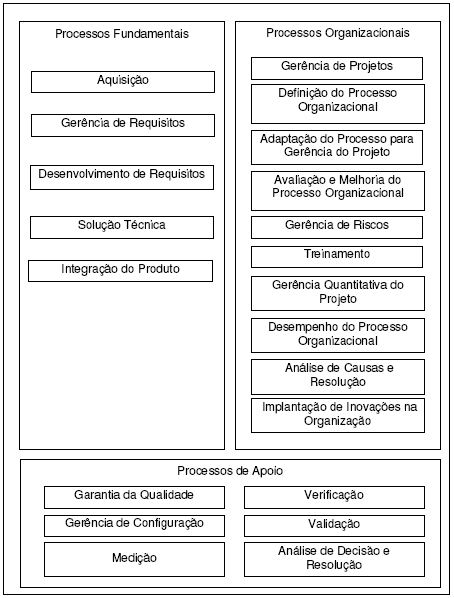


Figura .6 - Processos do MPS.BR [21]

.

# Capítulo 3 Ferramenta ProEvaluator

Esta seção será apresentada, de uma forma sucinta, o ambiente ImPPros, os componentes que estão integrados e a ferramenta ProEvaluator.

## 3.1 ImPPros

O ImPProS é um ambiente de implementação de processo de software com o foco na definição, simulação, execução e avaliação do processo a partir do uso de ferramentas de suporte ao processo de software e de apoio ao desenvolvimento de software. Ele deve prover em sua estrutura a integração dos serviços (funcionalidades) de ferramentas, a partir da disponibilização por parte do seu fornecedor, de um conector de comunicação entre o ImPProS e a ferramenta a ser integrada. Esta integração dá-se a partir da categoria de controle da integração das ferramentas **[32]**, uma vez que o ImPProS deve ser capaz de notificar eventos para outras ferramentas, ativar outras ferramentas e compartilhar funções de outra ferramenta, ou seja, exercer influência sobre outras **[1]**. Abaixo segue a sua estrutura.



## 3.2 ProEvaluator

O ProEvaluator é uma ferramenta de suporte ao ImPProS com o objetivo de prover a avaliação dos processos instanciados em execução, ou seja, um suporte que possibilita avaliar o processo de software instanciado e já institucionalizado na organização, durante a sua execução de acordo com as características definidas a ele **[1]**.

Com a possibilidade de prover ao usuário do ImPProS uma avaliação dos processos instanciados na sua estrutura durante a sua execução em uma organização, a Figura abaixo permite se ter um entendimento do fluxo de execução desta função: inicialmente o usuário necessita ativar a execução do ProEvaluator a partir do ImPProS; posteriormente o ImPProS gera um arquivo XML contendo informações relevantes (o identificador do projeto de implementação do processo do ImPProS) para efetuar a parametrização e a configuração dos dados do processo instanciado no ProEvaluator de acordo com o projeto de implementação aberto no ImPProS; logo em seguida o ImPProS ativa a execução do ProEvaluator; o qual lê e captura as informações mantidas no arquivo XML (dados.xml); em seguida o usuário efetua o logon no ProEvaluator; por sua vez este carrega os seus serviços que possibilitam a avaliação do processo instanciado e em execução; e o usuário efetua a seleção do processo a ser avaliado de acordo com a consulta na base de processos do ImPProS **[1]**.

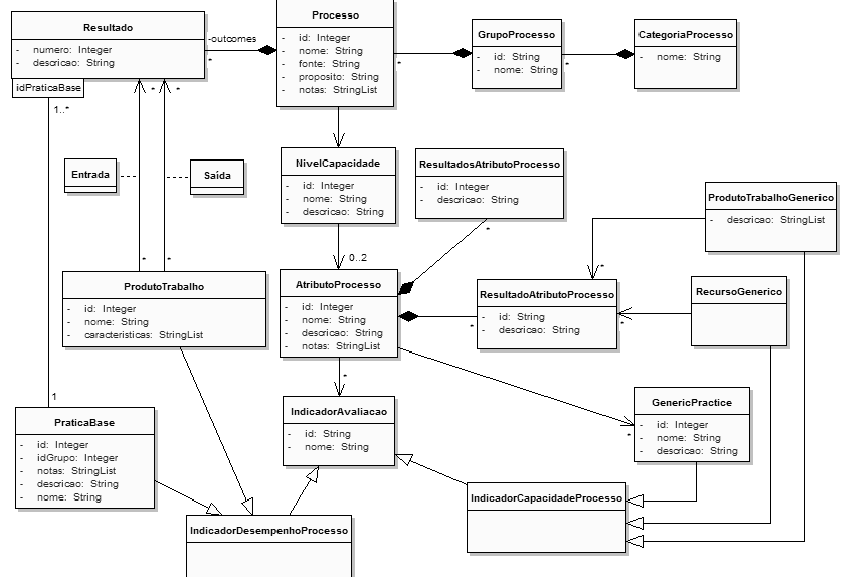


# Capítulo 4 Automação do Mapeamento

Este Capítulo aborda a concepção do modelo Genérico, ou seja, de que forma este modelo foi criado e a sua necessidade para o funcionamento do sistema de automatização. Também serão apresentados o cadastramento de modelos, o relacionamento entre eles e as tecnologias utilizadas.

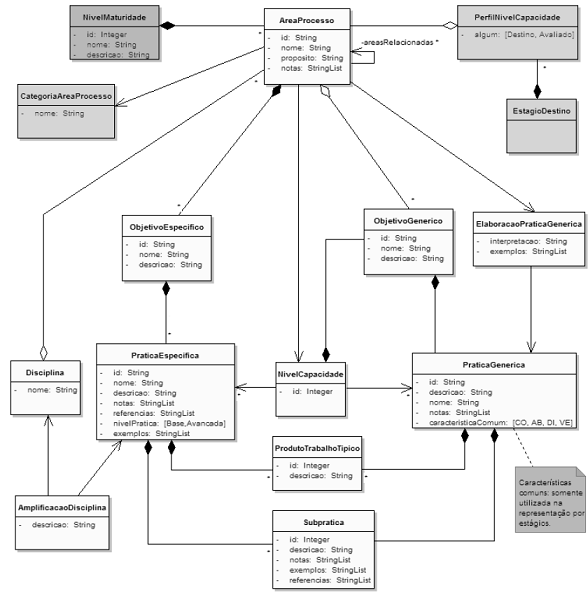
## 4.1 Concepção do Modelo Genérico

O relacionamento entre os itens do modelo de referência foi originado a partir de uma pesquisa qualitativa das diversas literaturas de normas e modelos de qualidade, trabalhos de graduação, relatório técnico e outros artigos relacionados. Muitos trabalhos que concerne ao mapeamento de modelos ou normas foram especificados em documentos, desprovido de qualquer programa que armazene os dados desse mapeamento. Com o objetivo de esse projeto automatizar o mapeamento, criou-se um software que faz o relacionamento entre os modelos ou normas. Primeiramente, pensou-se em instanciar um modelo genérico, onde se possa cadastrar qualquer modelo existente e possíveis modelos que podem surgir no mercado. Entretanto, esse modelo genérico estaria relacionado à melhoria e avaliação de processo de software, pois faz parte do contexto do trabalho de graduação. No mercado, existem vários modelos e normas para a avaliação de processos, tais como: ISO 9000, Bootstrap, CMM/CMMI, a ISO/IEC 15504, o MPS-BR e entre outros. A partir de então, foram selecionados três desses modelos/normas, a ISO/IEC 15504, o MPS-BR e o CMMI, visando a sua integração, originando um modelo genérico. Esse modelo genérico pode ser utilizado para qualquer modelo que seja baseado na norma ou modelo citado acima. Um estudo foi feito, observando as particularidades e similaridades de cada item do modelo/norma. Abaixo segue a modelagem da estrutura do modelo de avaliação do processo ISO/IEC 15504, CMMI e MPS-BR.



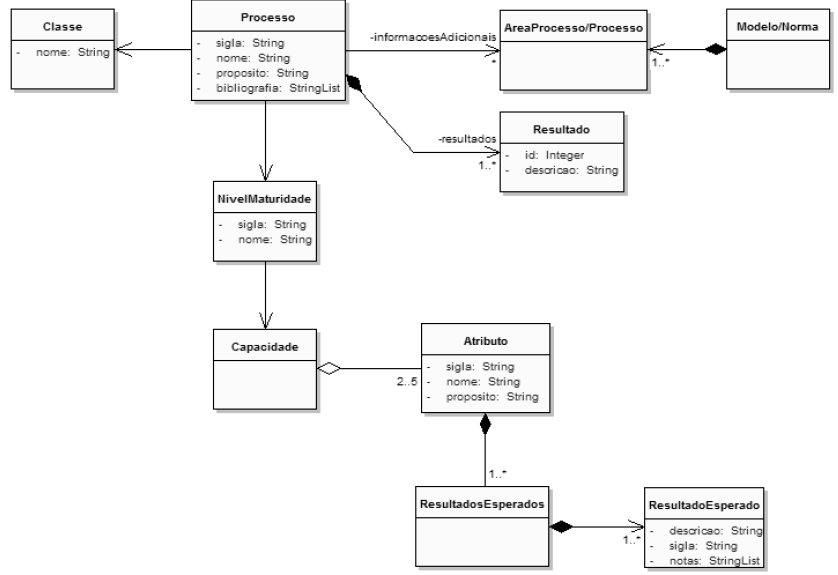
**Modelo de avaliação do processo ISO/IEC 1550.**

O modelo de avaliação do processo ISO/IEC 15504 está estruturado na expansão do modelo(s) de referência de processos, acrescentando a definição e uso de indicadores. Um indicador de avaliação de processo é um atributo ou característica objetiva de uma prática ou produto de trabalho que suporta o julgamento do desempenho ou capacidade de um processo implementado. A seguir, encontra-se o modelo de avaliação do processo CMMI-SE/SW.



**Modelo de avaliação do processo CMMI-SE/SW.**

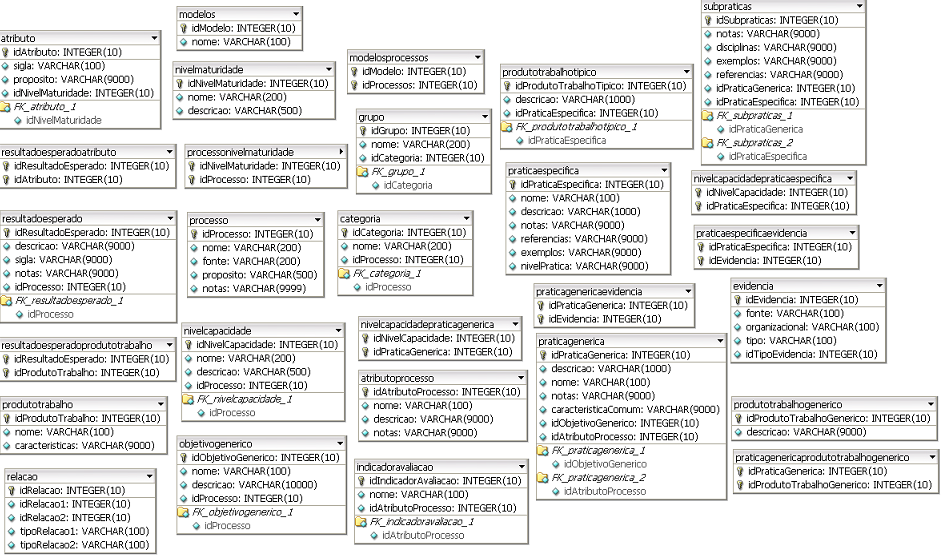
A modelagem acima representa o modelo conceitual, abrangendo os objetos do modelo com seus atributos e tipos. A estrutura cobre ambas as representações do CMMI, por estágios e contínua. Abaixo, encontra-se o último modelo de avaliação, o MPS-BR.



**Modelo de avaliação do processo MPS-BR.**

O diagrama de classes do modelo acima mostra os objetos com seus atributos e tipos. De acordo com os três modelos de avaliações dos processos, foi elaborado um modelo genérico que contemplasse a grande parte dos objetos, eliminando aqueles que não possuem atributos ou que são semelhantes, como por exemplo, áreas de processos e processo. Neste Caso, foi selecionado, processo, pois ambos possuem características similares. A seguir, será apresentado o “Modelo Relacional do Modelo Genérico” originado pela fusão dos três modelos.

**Modelo Relacional do Modelo Genérico**



Nesse Modelo Genérico, encontram-se as tabelas necessárias para a efetivação dos cadastros de modelos, citados acima e outros possíveis modelos, baseados na ISO/IEC, 15504, CMMI e MPS-BR. Abaixo será descrita a definição de cada “item” do modelo, caracterizado pelo nome da tabela.

1. **Atributo**: Uma característica mensurável.
2. **Atributo** **Processo**: Uma característica mensurável da capacidade do processo aplicável a qualquer processo.
3. **Categoria:** Modos de como os processos são agrupados.
4. **Evidência:** Dados que demonstram a existência ou veracidade de alguma coisa
5. **Grupo:** É a organização dos processos, dentro de uma categoria.
6. **Indicador Avaliação:** Apóia a determinação do grau de extensão em que um atributo de processo é satisfeito
7. **Modelo:** Compreende as definiçõesde processos no ciclo de vida descrito em termos de propósitos e resultados, junto com uma arquitetura que descrevem as relações entre os processos.
8. **Nível Capacidade:** Caracteriza a melhoria relacionada a uma área de processo.
9. **Nível Maturidade:** Grau de melhoriade processo para um predeterminadoconjunto de processos no qual todos os resultados esperados e dos atributos de processos são atendidos.
10. **Objetivo Genérico:** Descreve que a institucionalização precisa ser atingida para satisfazer uma área de processo.
11. **Prática Específica:** Descreve as atividades esperadas para alcançar um objetivo específico de uma área de processo.
12. **Prática Genérica:** São atividades de um tipo genérico e fornece um guia na implementação das características do atributo.
13. **Processo:** Um conjunto de atividades inter-relacionadas ou interativas que transforma insumos (entradas) em produtos (saídas)
14. **Produto Trabalho:** Um artefato associado à execução de um processo
15. **Produto Trabalho Genérico:** É um conjunto de características que se espera que seja evidente em produtos de trabalho de tipos genéricos como resultado do atendimento de um produto
16. **Produto trabalho Típico:** São possíveis saídas (exemplos) de uma prática específica
17. **Resultado Atributo Processo:** Um resultado ao alcance do propósito do atributo processo.
18. **Resultado Esperado:** Umresultadoobservável do sucesso do alcance do propósito do processo.
19. **Subpráticas:** Podem ser consideradas explicações informativas das práticas

A partir desse Modelo Genérico, foi possível fazer o cadastramento dos diversos modelos baseados na restrição citada. Essa concepção do modelo genérico foi necessária para que o relacionamento entre os modelos fossem realizadas. Depois que os modelos são cadastrados, o mapeamento entre eles podem ser efetivados.

## 4.2 Cadastro de modelos

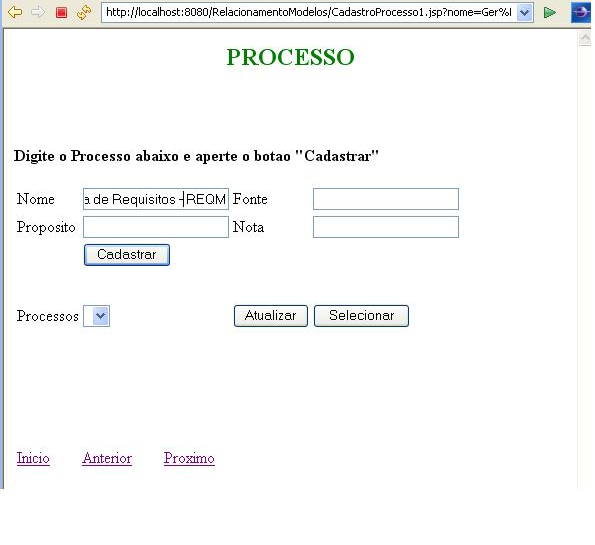
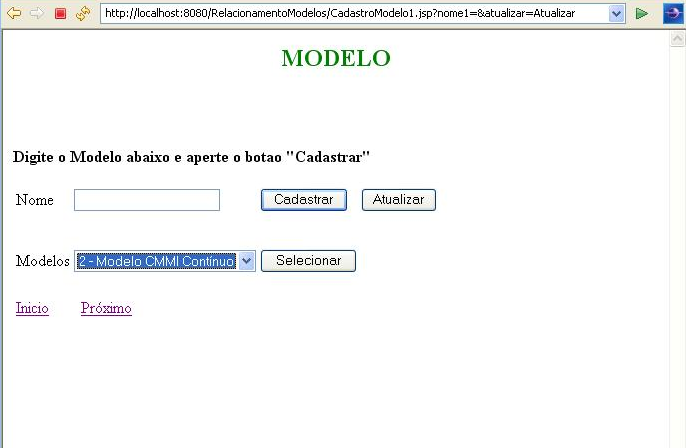
De acordo com o modelo genérico, existem 19 itens que compreende características do modelo de avaliação 15504, CMMI e MPS-BR. Então, para o cadastramento de um desses modelos, haverá itens que não pertencem a um determinado modelo, então o usuário cadastra o item como nulo. De forma análoga, o usuário pode cadastrar outros modelos que não sejam propriamente aquelas que foram citadas acima, mas que possuem características similares. Abaixo segue o Cadastramento de dois processos.

1. 1 – Cadastro do processo “Gerência de Requisitos - REQM” da Categoria “Engenharia” do Modelo “CMMI Contínuo”. Este processo pode visualizado na página 15 da figura 0.3. Abaixo segue as etapas do cadastro do processo

** Figura**

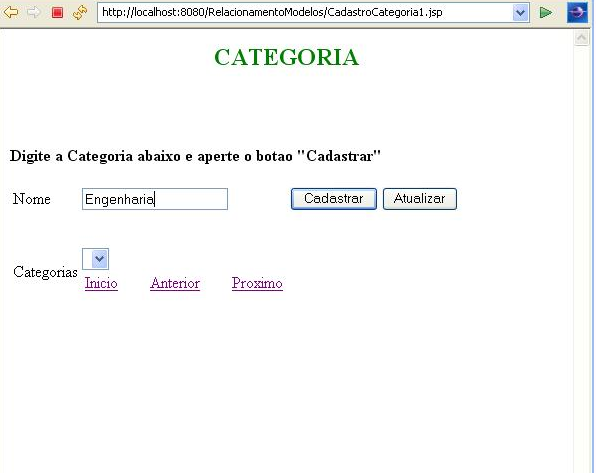
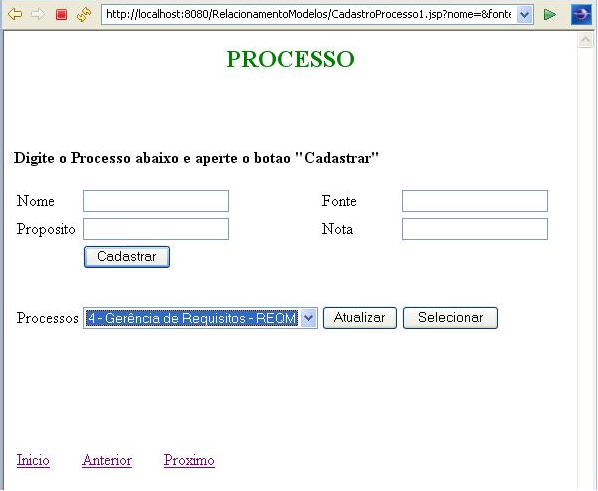
**Cadastro Modelo CMMI Contínuo 1/7**

**Figura Cadastro Modelo CMMI Contínuo 2/7**

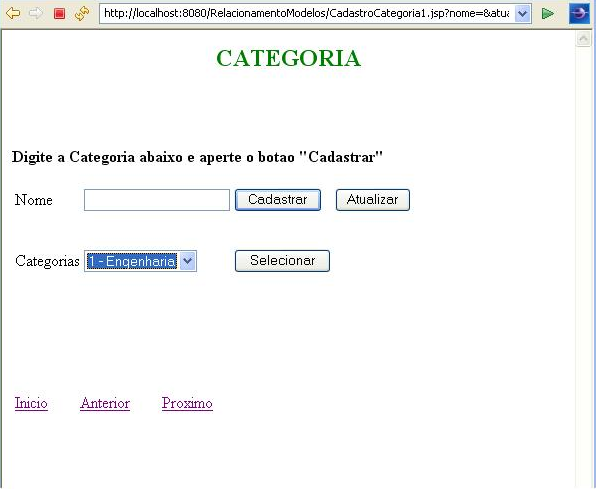


**Figura Cadastro Modelo CMMI Contínuo 3/7**

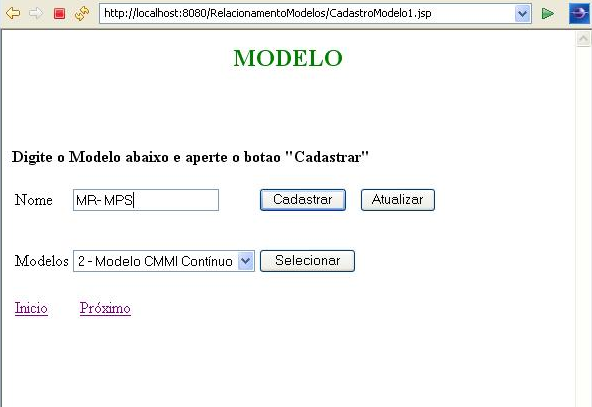
**Figura Cadastro Modelo CMMI Contínuo 4/7**

**Figura Cadastro Modelo CMMI Contínuo 5/7**

**Figura Cadastro Modelo CMMI Contínuo 6/7**

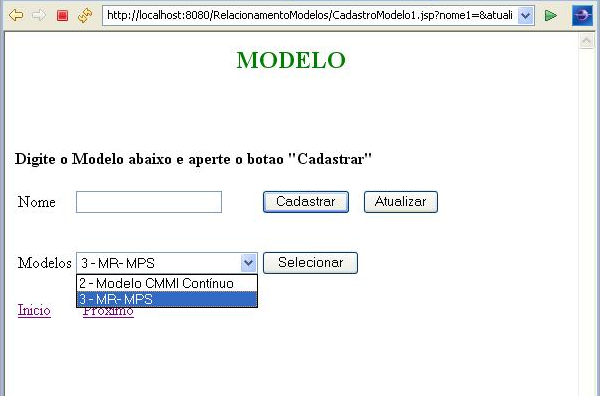
**Figura Cadastro Modelo CMMI Contínuo 7/7**

1. – Cadastro do processo “ Gerência de Requisitos - GRE” da Categoria “Processos Fundamentais” do Modelo de Referência “MR-MPS”. Este processo pode visualizado na página 19 da figura 0.6. Abaixo segue as etapas do cadastro do processo.

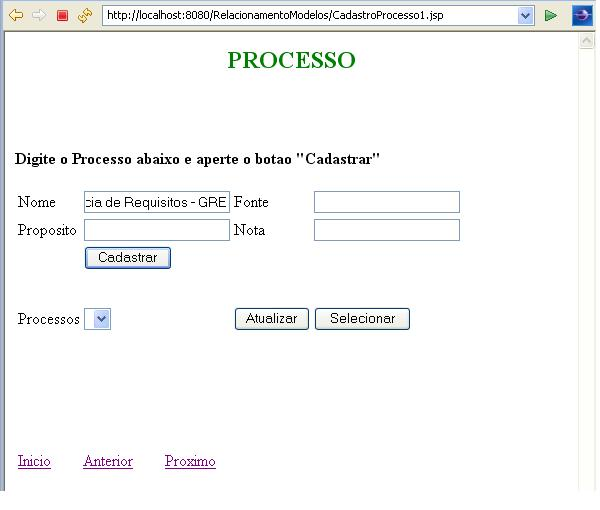


**Figura Cadastro Modelo MR-MPS 1/6**

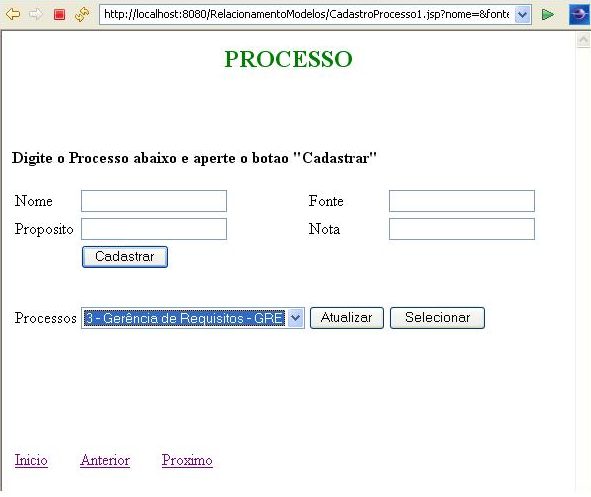
## 



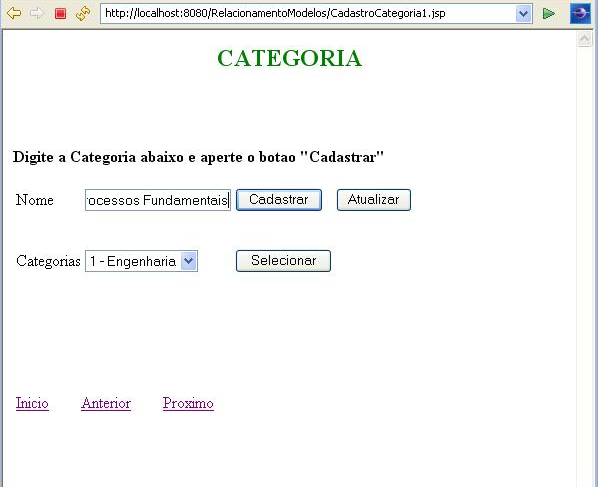
**Figura Cadastro Modelo MR-MPS 2/6**



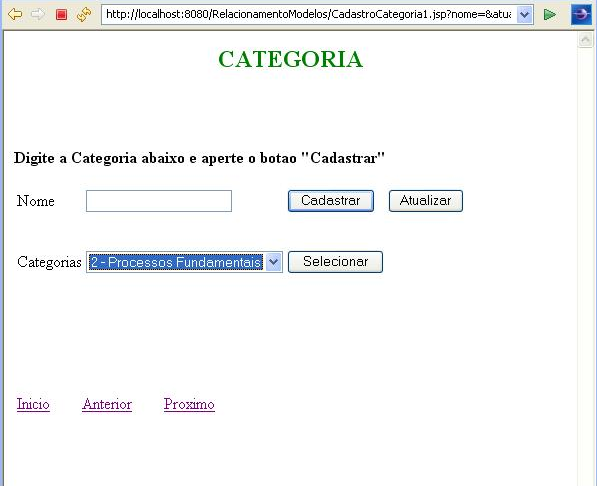
**Figura Cadastro Modelo MR-MPS 3/6**



**Figura Cadastro Modelo MR-MPS 4/6**



**Figura Cadastro Modelo MR-MPS 5/6**



**Figura Cadastro Modelo MR-MPS 6/6**

No Sistema foi feito o cadastramento de dois processos. Na seção seguinte, existe um mapeamento entre estes dois processos. O usuário pode relacionar esses processos utilizando o software construído.

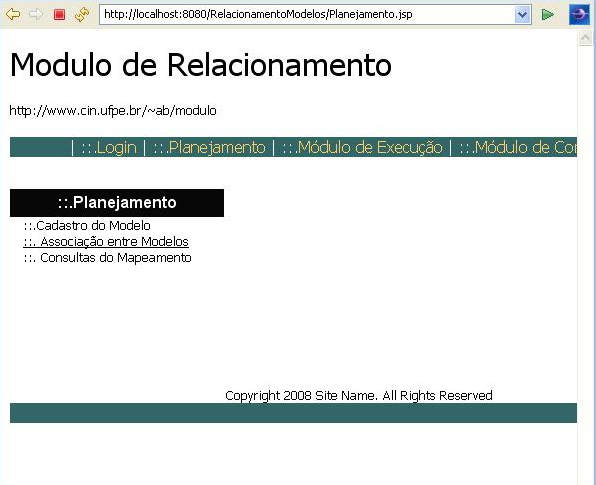
## 4.3 Relacionamento dos itens dos modelos

O relacionamento entre os modelos podem utilizados entre diversos itens do Modelo Genérico. O sistema permite que o usuário possa mapear, por exemplo, o processo “A” do modelo “X” com o processo “B” do modelo “Y”. O relacionamento está ligado a dois itens dos modelos. Será demonstrado o relacionamento de processos baseado do mapeamento entre eles. Na figura abaixo, há o mapeamento, em destaque, do processo GRE- Gerência de Requisitos da Categoria Processos Fundamentais do Modelo de Referência MR-MPS com o processo REQM- Gerência de Requisitos da Categoria Engenharia do Modelo CMMI Contínuo.



**Figura do Mapeamento entre os processos dos Modelos MR-MPS e CMMI Contínuo**

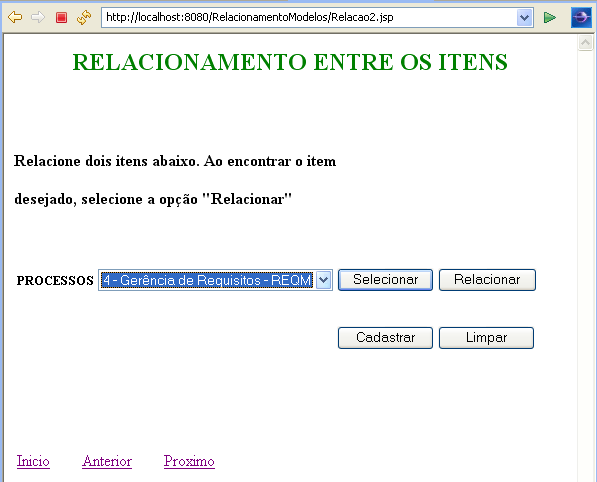
Na figura acima, o círculo que marca o mapeamento entre os processos indica o nível da sua similaridade. Quanto mais preenchido o círculo, maior o grau de semelhança. Agora esse mapeamento será cadastrado no sistema desenvolvido. Seguem as etapas:



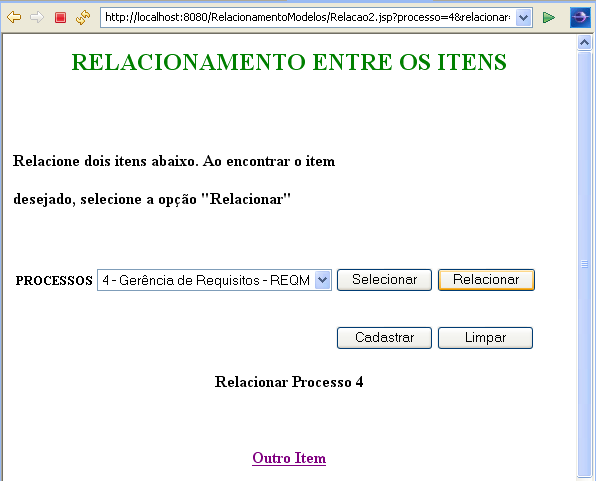
Relacionamento de Processos 1/10



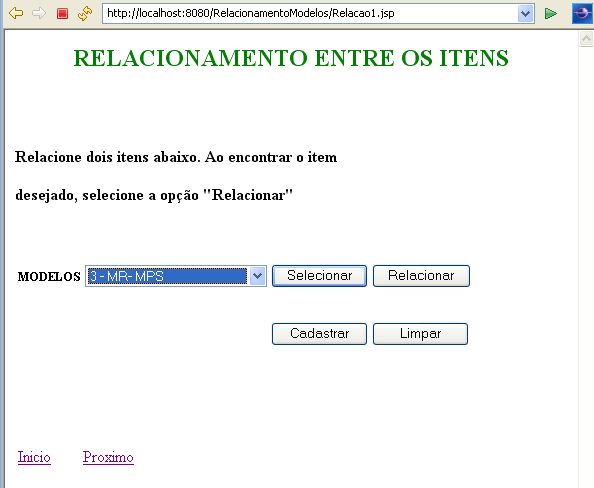
Relacionamento de Processos 2/10



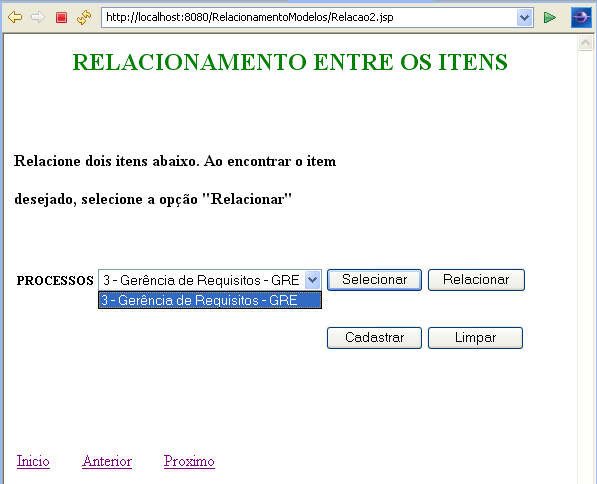
Relacionamento de Processos 3/10



Relacionamento de Processos 4/10



Relacionamento de Processos 5/10



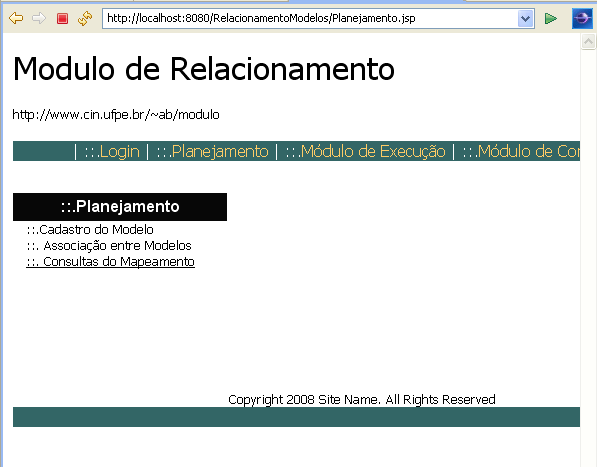
Relacionamento de Processos 7/10



Relacionamento de Processos 8/10



Relacionamento de Processos 9/10



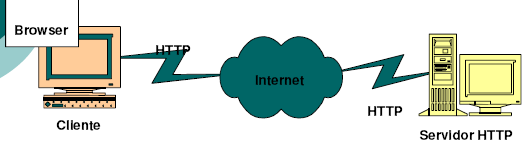
Relacionamento de Processos 10/10



Relacionamento de Processos 10/10

Na opção “Associação entre Modelos” foi possível fazer o relacionamento entre os itens dos Modelos e cadastrá-los. A consulta desse relacionamento pode ser visualizada na opção “Consultas do Mapeamento”.

## 4.4 Arquitetura do Sistema

O sistema de automação está voltado para o módulo web e estruturado em camadas, permitindo a modularidade; reusabilidade e extensabilidade; vários mecanismos de persistências suportados pela mesma aplicação; várias GUIs para a mesma aplicação e entre outros benefícios.

**Interação Cliente e Servidor**



**Acesso aos recursos web**

## 4.5 Tecnologias

No desenvolvimento do software, foram utilizadas ferramentas, linguagens de programação e entre outras tecnologias. Seguem abaixo os recursos tecnológicos.

* + 1. Java – Linguagem de programação
    2. JSP – Linguagem de programação Java para web
    3. HTML – Linguagem de marcação
    4. Eclipse wtp – Ferramenta para implementação Java para web
    5. WampSever – Programa que contem o servidor MySQL 5.0
    6. MySQL Query Browser – Programa do módulo cliente para o servidor MySQL.
    7. DBDesigner4 – Programa responsável pela modelagem do Banco de Dados.
    8. Tomcat 5.5 – Servidor Java para web
    9. JDK1.6.0 – Kit de desenvolvimento Java, que contem a máquina virtual Java.

# Capítulo 5 Conclusões

Neste capítulo será apresentada uma recapitulação dos principais tópicos abordados por este trabalho, assim como as suas contribuições e sugestões de possíveis trabalhos futuros.

## 5.1 Considerações Finais

Nas últimas décadas, várias abordagens associadas à melhoria de processo têm ganhado importância na comunidade de software. Os conceitos, métodos, e práticas englobam uma maneira de pensar, de agir e de entender os dados gerados pelos processos que, coletivamente, resultam em melhoria da qualidade, aumento da produtividade e competitividade dos produtos de software.

Muitas organizações têm investido na implantação de modelos de referência e normas com o objetivo maior de garantir a qualidade e se manter no mercado. Dentre os modelos e norma em destaque têm-se a ISO/IEC 15504, CMMI e MPS-BR. No capítulo 2 foi apresentada uma visão geral destes com o propósito conhecer suas características e possíveis semelhanças em alguns aspectos. No capítulo 3, foi abordado o ambiente ImPPros e a sua ferramenta ProEvaluator, pois possuem um papel importante para uma possível integração no módulo de relacionamento entre os modelos de referências. Finalmente o capítulo 4 que esclarece a concepção do modelo genérico e a necessidade da sua existência para o sistema, servindo como base para o cadastramento de modelos e relacionamento entre eles.

## 5.2 Contribuições

As principais contribuições deste trabalho de graduação são:

1. Automatização de relacionamento de itens entre os modelos, facilitando o trabalho de muitas organizações que utilizam mais de um modelo ou norma de qualidade de software.
2. Integração com a ferramenta ProEvaluator, este sistema pode servir como um módulo da ferramenta, permitindo mais funcionalidades.

## 5.3 Trabalhos Futuros

Além dos resultados obtidos com este trabalho de graduação, futuros trabalhos podem ser derivados através do tema proposto, e na melhoria do sistema de relacionamento entre modelos de referências, servindo como base para possíveis refinamentos e evolução do sistema. Os pontos importantes para a melhoria do sistema seriam:

1 – Tornar a interface gráfica mais intuitiva.

2 – Cadastrar os itens do modelo sem ter que seguir uma seqüência, ou seja, cadastrar apenas itens que pertencem ao modelo.

# Referências Bibliográficas

**[1]** ImPPros - Ambiente de Implementação Progressiva de Processo de Software. Disponível em: http://www.cin.ufpe.br/~imppros

**[2]**  Tayanna C. Sotero. Requisitos para uma Ferramenta de Avaliação de Processos. Artigo.

**[3]**  Vasconcelos, A.M. L., Rouiller, A. C., Machado, C. A. F. & Medeiros, T.M.M. Introdução à Engenharia de Software e à Qualidade de Software. Universidade Federal de Pernambuco. Disponível em: http://www.cin.ufpe.br/~if720

**[4]** Andrade, J. S., Albuquerque, A. B., Campos, B.F., et al., 2004, “Consequências e Características de um Processo de Desenvolvimento de Software de Qualidade e Aspectos que o influenciam: uma avaliação de especialistas”, III Simpósio Brasileiro de Qualidade de Software, Brasília, DF, Brasil.

**[5]** CMU/SEIa, (2002) – “STANDARD CMMI APPRAISAL METHOD FOR PROCESS IMPROVEMENT (SCAMPI)”, Version1.1: Method Implementation Guidance for Government Source and Contract Process Monitoring, CMU/SEI-2002-HB-002, Pittsburgh Software Engineering

**[6]** CMU/SEIb, (2002) “CAPABILITY MATURITY MODEL INTEGRATION (CMMI)”, Version 1.1 CMMI for Software Engineering (CMMI-SW, V1.1), Pittsburgh, Software Engineering Institute, Carnegie Mellon University. Disponível em: <http://www.sei.cmu.edu>

**[7]** CMU/SEIc, (2001) “APPRAISAL REQUIREMENTS FOR CMMI”, Version 1.1 (ARC, V1.1), (CMU/SEI-2001-TR-034). Pittsburgh, Software Engineering Institute, Carnegie Mellon University, 2001. Disponível em: <http://www.sei.cmu.edu>

**[8]** CMU/SEId, (2001) “Standard CMMI Appraisal Method for Process Improvement Version 1.1: Method Definition Document (CMU/SEI-2001-HB-001)”. Pittsburgh, Software Engineering Institute, Carnegie Mellon University, December 2001. Disponível em: <http://www.sei.cmu.edu>

**[9]** CMU/SEIe, (2001) “Standard CMMI Appraisal Method for Process Improvement (SCAMPI), Version 1.1: Method Implementation Guidance for Government Source Selection and Contract Process Monitoring” (CMU/SEI-2002-HB-002). Pittsburgh, Software Engineering Institute, Carnegie Mellon University, September 2002.

**[10]** Chrissis, M.B., Konrad,M., Shrum,S.,(2004), “CMMI Guidelines for Process Integration and Product Improvement”, SEI Series in Software Engineering, Ed. Addison Wesley

**[11]** Fabbrini F., Fantini E., Fusani M., Lami G. (2003) "Performing SPICE Assessments: Yet Another Tool". In Joint ESA - 3rd International SPICE Conference on Process Assessment and Improvement 17-21 March 2003 ESTEC, Noordwijk, The Netherlands.

**[12]** Florac, W.A., Park, R.E., Carleton, A.D., 1997, Practical Software Measurement: Measuring for Process Management and Improvement, CMU/SEI- 97-HB-003, Pittsburgh, Software Engineering Institute, Carnegie Mellon University.

**[13]** Galotta, C., Oliveira, K., Rocha, A.R. (2004), “Apoio a Interação entre Processos de Negócio e de Software através de Gerência do Conhecimento.” Simpósio Brasileiro de Qualidade de Software - 2004 - Brasília, DF, Brasil.

**[14]** IEEE, (2004) “SWEBOK – Guide to the Software Engineering Body of Knowledge” – A project of the IEEE Computer Society Professional Practices Committee.

**[15]** ISO/IEC TR 15504, (1998), Information technology – Software Process Assessment.

**[16]** ISO/IEC 15504-3, (2004) The International Organization for Standardization and the International Electrotechnical Commission. ISO/IEC 15504-3: Information Technology - Process Assessment - Part 3 - Guidance on Performing an Assessment, Geneve: ISO, 2004. [ISO/IEC 15504-4, 2004] - the International Organization for Standardization and the International Electrotechnical Commission. ISO/IEC 15504-4: Information Technology - Process Assessment – Part 4 - Guidance on use for Process Improvement and Process Capability Determination, Geneve: ISO, 2004.

**[17]** ISO/IEC 12207, (1995) The International Organization for Standardization and the International Electrotechnical Commission. ISO/IEC 12207 Information technology – Software life cycle processes, Geneve: ISO, 1995.

**[18]** Jones, C. (2000) “Software Assessments, Benchmarks, and Best Practices”. Estados Unidos: Addison-Wesley. 2000.

**[19]** Kan, S. H., 2003, “Metrics and Models in Software Quality Engineering”, Second Edition, Addison-Wesley.

**[20]** MCT/SEPIN – Secretaria de Política de Informática e Automação / Ministério da Ciência e Tecnologia, 2002, Qualidade e Produtividade no Setor de Software Brasileiro – 2001 / n.4., Brasília.

**[21]** MPS.BR, (2007a) ASSOCIAÇÃO PARA PROMOÇÃO DA EXCELÊNCIA DO SOFTWARE BRASILEIRO – SOFTEX. MPS.BR - Guia Geral, versão 1.2, 2007. Disponível em: [www.softex.br](http://www.softex.br).

**[22]** MPS.BR, (2007b) ASSOCIAÇÃO PARA PROMOÇÃO DA EXCELÊNCIA DO SOFTWARE BRASILEIRO – SOFTEX. MPS.BR – Guia de Avaliação, versão 1.1, 2007. Disponível em: [www.softex.br](http://www.softex.br)

**[23]** Moura J. (2007) “ProEvaluator: Uma Ferramenta para Avaliação de Processos de Software”, dissertação de mestrado defendida em Julho/2007, CIn-UFPE, Brasil.

**[24]** Moreau, B. et al., 2003, Software Quality Improvement in France Telecom Research Center, Software Process: Improvement and Practice, v.8:3, pp. 135 – 144.

**[25]** Muradas, F. M.(2006) “Processo de Avaliação MPS.BR: Definição e Ambiente de Apoio”, COPPE/UFRJ, 2006, Rio de Janeiro.

**[26]**  Oliveira, S., Vasconcelos, A., Rouiller, A. C.,(2005) “Uma Proposta de um Ambiente de Implementação de Processo de Software”, Artigo publicado na Revista InfoComp – Revista de Ciência da Computação da UFLA – vol. 4, n. 1, Lavras-MG, 2005.

**[27]** Paulk, M. C., Weber, C. V., Curtis, B., Chrossis, M. B. (eds), 1995, The Capability Maturity Model: Guidelines for Improving the Software Process, Addison-Wesley.

**[28]** Rout, P.T., (1995) “SPICE: A Framework for Software Process Assessment” – Software Process – Improvement and Practice”, 1995.

**[29]**  Santos, G., 2003, Representação da Distribuição do Conhecimento, Habilidades e Experiências através da Estrutura Organizacional, Tese de M. Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.

**[30]** Salviano, C., Wangenheim C., Pickler K., Thiry M., Zoucas A. (2001) “Aplicando Avaliações de Contextualização em Processos de Software Alinhados ao CMMI-SE/SW” – Simpros – VII Simpósio Internacional de Melhoria de Processos de Software, 2001.

**[31]** Walker, A.J., Lok, H.R., (1995), “SPICE Assessments using the SEAL assessment tool”, Software Engineering Applications Laboratory.

[32] Jorgensen, S. A. (1994) “An Object-Oriented Approach to Tool Integration in an Integrated CASE Environment”, M.S. Thesis Electrical and Computer Engineering, University of New Mexico;