



Universidade de Brasília - UnB  
Faculdade UnB Gama - FGA  
Engenharia de Software

# Qualidade na Modelagem de Processos de Software

Autora: Ramaiane Medeiros da Silva  
Orientadora: Profa. Fabiana Freitas Mendes

Brasília, DF  
2014



Ramaiane Medeiros da Silva

## **Qualidade na Modelagem de Processos de Software**

Monografia submetida ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

Universidade de Brasília - UnB

Faculdade UnB Gama - FGA

Orientador: Profa. Fabiana Freitas Mendes

Brasília, DF

2014

---

Ramaiane Medeiros da Silva  
Qualidade na Modelagem de Processos de Software/ Ramaiane Medeiros da  
Silva. – Brasília, DF, 2014-  
125 p. : il. (algumas color.) ; 30 cm.

Orientador: Profa. Fabiana Freitas Mendes

Trabalho de Conclusão de Curso – Universidade de Brasília - UnB  
Faculdade UnB Gama - FGA , 2014.

1. modelagem de processo de software. 2. qualidade de modelagem. I. Profa.  
Fabiana Freitas Mendes. II. Universidade de Brasília. III. Faculdade UnB Gama.  
IV. Qualidade na Modelagem de Processos de Software

CDU 02:141:005.6

---

Ramaiane Medeiros da Silva

## **Qualidade na Modelagem de Processos de Software**

Monografia submetida ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

Trabalho aprovado. Brasília, DF, 25 de junho de 2014:

---

**Profa. Fabiana Freitas Mendes**  
Orientadora

---

**Profa. Elaine Venson**  
Convidado

---

**Prof. Dr. Luiz Carlos Miyadaira  
Ribeiro Júnior**  
Convidado

Brasília, DF  
2014

# Agradecimentos

Agradeço primeiramente a Deus que tem me sustentado e me guiado, agradeço a Ele que tem me dado sabedoria e me dado conhecimento necessário durante todos esses anos, e principalmente, agradeço por Seu amor incondicional.

Em segundo, agradeço a minha família que foi meu suporte, que sempre me ajudou e fez de tudo para me ajudar, e por todo amor, carinho, proteção e preocupação. Agradeço também aos meus amigos que sempre estiveram do meu lado e me deram todo apoio durante esses anos de curso. Agradeço também, aos meus amigos que fiz durante o curso que sempre me ajudaram a crescer durante o curso.

Agradeço aos meus professores pelo conhecimento compartilhado, pelos ensinamentos, e por colaborar para o meu crescimento acadêmico e profissional. Agradeço a minha orientadora, professora Fabiana Freitas Mendes que colaborou no desenvolvimento deste trabalho e soube conduzir da melhor forma possível o desenvolvimento deste trabalho.

# Resumo

A melhoria dos processos tem se tornado importante para o sucesso de empresas de desenvolvimento de software, pois elas dependem da qualidade e da eficiência do processo executado. A Melhoria de Processos de Software (MPS) envolve, muitas vezes, a modelagem de processos que pode considerar modelos e normas de qualidade de processo. O tema deste trabalho é a análise da qualidade da modelagem ou documentação de processos de software. Assim, objetiva-se definir um conjunto de características de qualidade de modelos ou documentos de processo que permita criar um instrumento de avaliação de qualidade. Para tanto, foi realizado um mapeamento sistemático a fim de encontrar características de qualidade de modelagem de processos, essas características foram posteriormente avaliadas/validadas através de um survey. O modelo final de características de qualidade é apresentado bem como diretrizes para seu uso.

**Palavras-chaves:** Processos de software. Modelagem de processo. Documentação de processo. Qualidade. Características de qualidade.

# Abstract

Process improvement has become important to the success of software development companies, because they depend on the quality and efficiency of the executed process. The Software Process Improvement (SPI) often involves the modeling of processes that can consider models and process quality standards. The theme of this work is the analysis of quality of the modeling or documentation software process. The aim of this work is to define a set of quality characteristics of models or process documents in order to create an instrument for quality evaluation. Therefore, a systematic mapping was conducted in order to find quality characteristics of modeling of processes, these characteristics were then evaluated/validated through a survey. The final model of quality characteristics is presented as well as guidelines for their use.

**Keywords:** Software Process. Process modeling. Quality. Process documentation. Quality. Quality characteristics.

# Lista de ilustrações

Figura 1 – Fases de pesquisa . . . . .	12
Figura 2 – Estrutura do modelo de qualidade. Adaptada da ISO/IEC 25010 (ISO, 2011) . . . . .	20
Figura 3 – Modelo de qualidade em uso. Adaptada da ISO/IEC 25010 (ISO, 2011)	21
Figura 4 – Modelo de qualidade do produto. Adaptada da ISO/IEC 25010 (ISO, 2011) . . . . .	21
Figura 5 – Grupos do Processo de Ciclo de vida. Adaptado:(ISO, 2008c) . . . . .	26
Figura 6 – Modelo IDEAL. Adaptado de(McFeeley, 1996) . . . . .	28
Figura 7 – Escopo do PSM. Fonte: PSM (McGarry et al., 2002) . . . . .	31
Figura 8 – Fases de pesquisa . . . . .	34
Figura 9 – Estágios principais de um survey (Schuman and Kalton, 1985) . . . . .	38
Figura 10 – Escala utilizada no questionário . . . . .	39
Figura 11 – Variação de confiabilidade pelo valor de alfa (adaptado de Freitas (2005))	42
Figura 12 – Trabalhos identificados por fonte de pesquisa . . . . .	44
Figura 13 – Seleção dos trabalhos identificados . . . . .	44
Figura 14 – Trabalhos selecionados por fonte de pesquisa . . . . .	45
Figura 15 – Trabalhos aceitos por fonte de pesquisa . . . . .	45
Figura 16 – Localização dos respondentes por UF . . . . .	49
Figura 17 – Tempo de atuação dos respondentes com MPS . . . . .	50
Figura 18 – Tempo de experiência com modelagem de processo . . . . .	50
Figura 19 – Curva da distribuição normal (Morettin and Bussab, 2009) . . . . .	52
Figura 20 – Item 4 - sintática . . . . .	53
Figura 21 – Item 22 - Tamanho . . . . .	54
Figura 22 – Item 23 - complexidade . . . . .	55
Figura 23 – Item 25 - escalabilidade . . . . .	55
Figura 24 – Processo de obtenção das características . . . . .	59
Figura 25 – Modelo das características de qualidade de processo de software . . . . .	61
Figura 26 – Modelo de medição de qualidade de produto de software. Adaptado (ISO, 2011) . . . . .	69
Figura 27 – Processo de medição de modelagem de processo de software . . . . .	70
Figura 28 – Processo de medição de modelagem de processo de software . . . . .	71
Figura 29 – Processo da fase diagnóstico. Adaptado do IDEAL (McFeeley, 1996). . . . .	76
Figura 30 – Processo da fase de aprendizado. Adaptado do IDEAL (McFeeley, 1996). . . . .	78



# Lista de tabelas

Tabela 1 – Cronograma do projeto . . . . .	14
Tabela 2 – Cronograma TCC 2 . . . . .	16
Tabela 3 – Tabela critérios de inclusão na fase de extração de artigos . . . . .	45
Tabela 4 – Tabela criterios de exclusão na fase de extração de artigos . . . . .	46
Tabela 5 – Indicadores do mapeamento sistemático . . . . .	47
Tabela 6 – Métricas propostas por Guceglioglu and Demirors . . . . .	48
Tabela 7 – Características por grau de importância . . . . .	56
Tabela 8 – Subcaracterísticas por grau de importância . . . . .	57
Tabela 9 – Modelo de especificação de métricas . . . . .	73
Tabela 10 – Especificação da métrica compreensibilidade funcional . . . . .	74
Tabela 11 – Tabela de palavras chaves . . . . .	87
Tabela 12 – Coleta de dados do trabalho . . . . .	90
Tabela 13 – Base de pesquisa - indicadores . . . . .	91
Tabela 14 – Características de qualidade do processo . . . . .	92
Tabela 15 – Características de qualidade do processo . . . . .	94

# 1 Introdução

O guia PMBOK define processo como sendo um conjunto de atividades inter-relacionadas realizadas para obter um conjunto específico de produtos, resultados ou serviços (PMI, 2008).

A utilização de processos tem se tornado um fator importante para o sucesso de empresas de desenvolvimento de software, pois um processo ajuda a controlar os custos, os prazos, os recursos e monitorar a produtividade da equipe.

Processos de software formam a base para o controle gerencial de projetos de software e estabelece o conteúdo no qual os métodos técnicos são aplicados, os produtos de trabalho (modelos, documentos, dados, relatórios, formulários, etc.) são produzidos, os marcos são estabelecidos, a qualidade é assegurada e as modificações são adequadamente geridas (Pressman, 2006, p. 17).

Dessa forma, processos formalmente estabelecidos permitem maior controle e segurança aos profissionais de uma organização, pois possibilitam que qualquer membro de uma organização, que possua treinamento adequado, possa conduzir o processo mantendo os níveis de serviços aceitáveis. Com a padronização das atividades e serviços durante o processo, o projeto se torna mais previsível permitindo que possíveis riscos sejam monitorados e mitigados.

O sucesso do desenvolvimento do software depende da qualidade e da eficiência do processo de desenvolvimento de software. Para obter um processo de qualidade muitas organizações têm usado modelos e normas de qualidade como CMMI (Capability Maturity Model – Integration)(CMM, 2006), MPS.BR (Melhoria de Processo de Software)(SOF, 2012), ISO 12207(International Organization for Standardization 12207)(ISO, 2008c), e modelo IDEAL (McFeeley, 1996).

Uma tarefa importante nesse contexto de definição e análise de processo é a modelagem do processo. A modelagem de um processo se concentra na definição, documentação e melhora dos processos de uma organização (García-Borgoñón, García-García, Alba, and Domínguez-Mayo, 2013). Por isso, para apoiar o processo de software, primeiro deve ser construído um modelo que especifique como cada atividade deve ser realizada, devem ser especificadas as funções e atribuições de tarefas envolvidas, os recursos consumidos, as ferramentas utilizadas, as entradas e saídas necessárias para as tarefas, o produto desenvolvido, e o mecanismo de comunicação entre as tarefas e funções (Zamli and Lee, 2001).

Segundo Olson (2010), um processo bem definido deve ser constituído por obje-

tivo, atividades, entradas, saídas, critérios de entrada e saídas, procedimentos, métodos e ferramentas. Além disso, um bom processo deve utilizar tanto palavras quanto figuras para representá-lo; deve ser bem escrito; deve ser fácil de navegar e deve ser pequeno.

O tema deste trabalho é a análise da qualidade da modelagem ou documentação de processos de software. Qualidade é definida como a totalidade de características de uma entidade que afetam a capacidade de satisfazer necessidades explícitas e implícitas, quando usada sob condições específicas (ISO, 2003). Em termos de qualidade de software, a qualidade do software está altamente relacionada à qualidade do processo de desenvolvimento de software (Rocha, Maldonado, and Weber, 2001), ou seja, para obter um bom produto é preciso ter-se um bom processo de software.

Este capítulo está organizado em quatro seções: introdução, justificativa, objetivo, metodologia e cronograma, sendo esta a primeira das seções. Nesta seção foram apresentados os principais conceitos relacionados a este trabalho, o contexto do trabalho e o tema que será explorado. A Seção seguinte, apresenta a justificativa e os problemas relacionados a este trabalho. Na Seção 1.2 são apresentados os objetivos: objetivo geral e objetivos específicos. Na Seção 1.3 será apresentada a metodologia do trabalho, o modo como o trabalho foi e será realizado e o cronograma executado para a realização deste trabalho.

## 1.1 Justificativa

Dada a importância do processo de software, organizações que trabalham com desenvolvimento de software estão cientes de que os processos são ativos muito importantes, assim como estão cientes da necessidade de implantar processos bem definidos, com o objetivo de melhorar a qualidade do desenvolvimento dos produtos de software (García-Borgoñón, García-García, Alba, and Domínguez-Mayo, 2013).

Tendo em vista a necessidade de implantar processos bem definidos, a comunidade de software tem mostrado um grande interesse em melhorar o processo de software, e em encontrar a melhor maneira de definir um processo em uma organização (Zamli and Lee, 2001). Existem diversas linguagens de modelagem e modelos de processo de software propostos para ajudar a modelar e definir um processo de software (Borgoñón-García, Barcelona, García-García, Alba, and Escalona, 2013). Entretanto, não se pode determinar se um processo é bem definido sem antes analisar sua qualidade. Uma das maneiras de analisar a qualidade do processo é analisar a qualidade da modelagem do processo de software.

Para se analisar a qualidade da modelagem é importante definir parâmetros e métricas, pois essas têm se mostrado bastante efetivas em controlar a complexidade e, dessa forma, assegurar a alta qualidade do software produzido (Muketha, Ghani, Selemat, and Atan, 2010). Alguns dos desafios relatados para melhoria de processo de software estão

diretamente relacionados à definição e uso de métricas, por exemplo: definir métricas para dar visibilidade do alcance dos objetivos a todos interessados; usar medidas simples, significantes e fáceis de serem obtidas; coletar apenas dados relevantes e que de fato serão usados (Mendes, 2010).

Ao longo dos anos as métricas têm se mostrado efetivas em controlar a complexidade e, portanto, em garantir a alta qualidade do produto. Se usadas corretamente, métricas podem permitir quantificar o sucesso ou fracasso, e podem ajudar na melhoria e tomada de decisões à respeito de um processo. Muitas métricas de processo têm sido propostas (Muketha, Ghani, Selemat, and Atan, 2010), mas ainda existem poucas pesquisas e poucos estudos que tratem de características de qualidade que podem, posteriormente transforma-se em métricas para avaliar a modelagem de um processo.

Constatada essas dificuldades, este trabalho propõe a seguinte questão: Quais são as características que definem a qualidade de modelagem ou documentação de processos de software?

Respondendo essa questão, este trabalho pode contribuir para a análise da qualidade da modelagem de um processo de software, identificando características de qualidade para tal análise. Os dados aqui identificados podem servir como referência trabalhos futuros que envolvem processos de software. Além disso, o trabalho aqui realizado pode ser utilizado como uma ferramenta de apoio a uma iniciativa de melhoria de processos de software, logo após a modelagem deles.

## 1.2 Objetivo

O objetivo geral deste trabalho é definir um instrumento que permita avaliar a qualidade da modelagem ou documentação de um processo de software.

Os objetivos específicos são:

- Identificar as características de qualidade na modelagem de processo
- Definir um modelo de características de qualidade de modelagem de processo
- Validar o modelo de características de qualidade de modelagem de processos de software
- Definir a forma de uso do modelo de características tanto de forma isolada quanto no contexto de MPS (melhoria de processo de software)

### 1.3 Metodologia

A metodologia de pesquisa utilizada nesse trabalho foi dividida em algumas fases de pesquisa sendo elas, fase decisória, fase construtiva e fase redacional, assim como mostra a Figura 1.

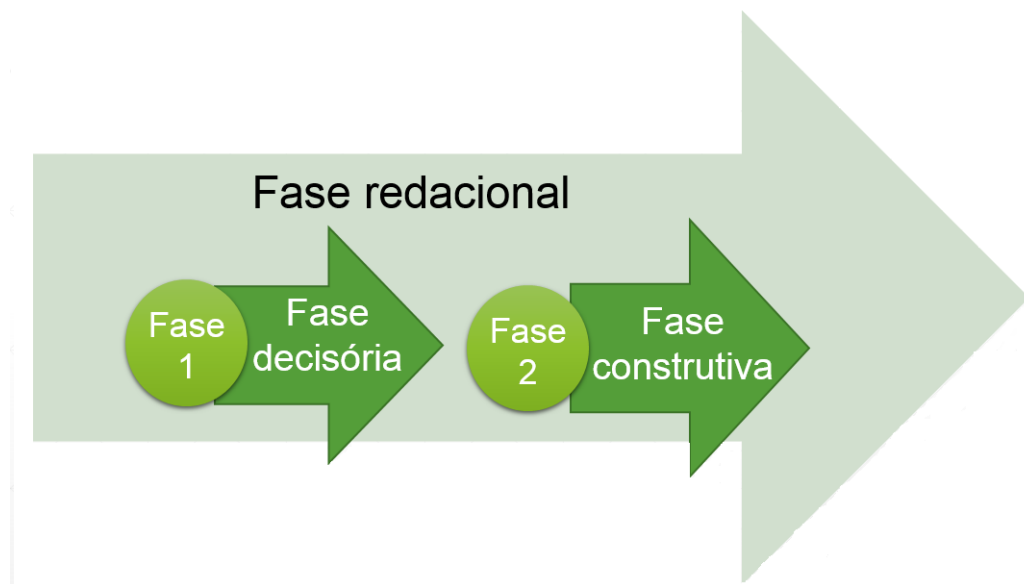


Figura 1 – Fases de pesquisa

A fase decisória é a fase inicial desta pesquisa, referente à escolha do tema, do problema de pesquisa. Nessa fase foi realizado um mapeamento sistemático para identificar na literatura as características de qualidade de processo. Já a fase construtiva, fase referente a execução da pesquisa, nessa fase o modelo de qualidade de processo construído foi validado. Na fase redacional, é a fase onde ocorre toda a análise dos dados e dos resultados da pesquisa e toda a documentação do trabalho. A forma como esta pesquisa foi conduzida é melhor detalhada no Capítulo 3.

Este trabalho foi dividido em duas partes, conforme requerido pelo curso de graduação no qual este trabalho está inserido.

Para o desenvolvimento da primeira parte do trabalho, foi executado um cronograma de atividades de trabalho, como é mostrado na Tabela 1. Este trabalho foi dividido em fases: fase decisória, construtiva e redacional. A fase decisória consistiu na leitura de artigos, livros, normas, materiais de apoio para a elaboração do trabalho, além da aplicação do estudo de caso que será definido. A fase redacional consistiu na definição e escrita do trabalho e entrega final do projeto, junto com a apresentação para uma banca de professores.

Foi definido também um cronograma para a segunda etapa deste trabalho, como pode ser visto na Tabela 1.3. Nessa etapa foram alcançados os outros objetivos específicos

deste trabalho que são: definir métricas e verificar a validade dos instrumentos obtidos na definição das métricas.

Tabela 1 – Cronograma do projeto

<b>Fase</b>	<b>Parte</b>	<b>Artigos</b>	<b>Data inicio</b>	<b>Data termino</b>	
Decisória	Familiarização com a área de Melhoria de Processo de Software (MPS)	Ler os capítulos de livros de Engenharia de Software (ES)	20/12/13	27/12/13	
		Ler artigos clássicos de ES MPS	27/12/13	10/01/14	
		Estudar a ISO/IEC 12207	10/01/14	17/01/14	
		Estudar o CMMI	17/01/14	31/01/14	
		Estudar o Guia Geral do MPS.BR	31/01/14	19/02/14	
		Estudar os problemas relacionados à MPS	20/02/14	12/03/14	
		Ler artigos específicos de definição de processos	20/02/14	12/03/14	
	Referencial Teórico		Ler com cuidado o artigo: Software Process Modeling Languages - A Systematic Literature Review	12/03/14	19/03/14
			Ler o artigo: Taxonomy of Process Modeling Languages	19/03/14	25/02/14
			Definir se será mapeamento ou revisão sistemática	26/03/14	16/04/14
			Definir o que ler sobre os temas GQM, PSM e Qualidade (PML, IDEAL, qualidade)	02/04/14	09/04/14
			Ler o material selecionado sobre GQM e PSM	09/04/14	23/04/14
	Mapeamento sistemático		Definir protocolo de mapeamento sistemático	02/04/14	16/04/14
			Definir string de busca pensando nos grupos de palavras-chave, montar várias opções de strings e testá-las	30/04/14	07/05/14

		Definir template de coleta de dados	30/04/14	14/05/14
		Definir indicadores da RS	30/04/14	14/05/14
		Finalizar Mapeamento Sistemático	14/05/14	21/05/14
Redacional	Introdução	Definir o tema e escrever a contextualização	03/03/14	12/03/14
		Escrever justificativa	12/03/14	19/03/14
		Definir e escrever o problema	12/03/14	19/03/14
		Definir e escrever os objetivos geral e específico	12/03/14	19/03/14
		Escrever a metodologia do trabalho	19/03/14	26/03/14
	Desenvolvimento	Definir estrutura de capítulos do TCC	26/03/14	02/04/14
		Definir estrutura do Capítulo 2	26/03/14	02/04/14
		Escrever sobre RS/MS	02/04/14	09/04/14
		Escrever sobre Qualidade	09/04/14	23/04/14
		Escrever o protocolo de MS	16/04/14	23/04/14
		Escrever sobre PSM/GQM	16/04/14	23/04/14
		Alterar o protocolo de acordo com a revisão realizada em reunião de orientação	07/05/14	14/05/14
		Escrever os resultados e discussões	21/05/14	28/05/17
	Conclusão	Escrever conclusão do trabalho	21/05/14	28/05/17
	Finalização	Entrega	Entregar TCC escrito	16/06/14
Apresentação a banca avaliadora			25/06/14	25/06/14



Tabela 2 – Cronograma TCC 2

<b>Fase</b>	<b>Parte</b>	<b>Tarefa</b>	<b>Data Inicio</b>	<b>Data fim</b>
Decisória	Survey	Estudar material relacionado ao tema	2-Jul	14-Jul
		Pesquisar outros trabalhos que já fizeram validação de modelos	19-Aug	29-Sep
		Pesquisar como validar questionários	5-Sep	22-Sep
		Pesquisar sobre forma de uso do conjunto de características de qualidade de processo (SQuaRE, IDEAL, QIP)	6-Oct	13-Oct
Construtiva	Survey	Definir instrumentos para execução do Survey (questionários por exemplo)	14-Jul	26-Aug
		Definir forma de validação do questionário	5-Sep	22-Sep
		Definir como extrair informações do questionário	5-Sep	22-Sep
		Escrever capítulo sobre o Survey	5-Sep	22-Sep
		Transferir o questionário para a plataforma	15-Sep	29-Sep
Redacional	Revisão TCC1	Fazer as alterações sugeridas pela banca	2-Jul	9-Jul
		Fazer alterações no modelo considerando o que foi definido no questionário	15-Sep	29-Sep
	Artigo	Propor um versão inicial do artigo (10 a 15 pag)	2-Jul	30-Jul
		Definir conferências alvo de publicação	9-Jul	14-Jul
		Revisão do artigo (pela profa)	30-Jul	26-Aug
		Tradução do artigo	26-Aug	3-Sep
		Revisão final e submissão	3-Sep	22-Sep
		Fazer alterações e submeter no ENASE	13-Oct	19-Nov
	Resultados	Pensar em como será o uso do modelo	13-Oct	20-Oct
		Fazer uma estrutura de como serão apresentados os resultados	6-Oct	20-Oct
	Revisão	Reorganizar texto do TCC	6-Oct	13-Oct

	Fazer alterações indicadas no pdf	13-Oct	20-Oct
--	-----------------------------------	--------	--------

## 1.4 Organização do trabalho

Este trabalho está organizado em seis capítulos. O Capítulo 1 apresenta a o tema deste trabalho e suas justificativa, o objetivo do trabalho e a metodologia de pesquisa. O Capítulo 2 apresenta os principais conceitos a serem utilizados nesse trabalho. O Capítulo 3 apresenta como este trabalho foi conduzido. O Capítulo 4 apresenta os resultados da condução deste trabalho. O Capítulo 5 apresenta o modelo elaborado a partir dos resultados deste trabalho. E por fim, o Capítulo 6 apresenta as considerações finais deste trabalho junto com a proposta de trabalhos futuros.

## 2 Fundamentação Teórica

Este capítulo tem o objetivo de apresentar os principais conceitos que serão tratados nesse trabalho, são eles, qualidade e medição. Este capítulo está dividido em duas seções: a Seção 2.1 apresenta os conceitos sobre qualidade, incluindo qualidade de processo, qualidade de modelagem, métodos de implantação de melhoria e normas de qualidade. Já, a Seção 2.2 apresenta os conceitos sobre medição e métodos de medição.

### 2.1 Qualidade

Segundo a ISO 9000 (2005a), qualidade é o grau no qual um conjunto de características inerentes satisfaz aos requisitos. Dessa forma é adequado concluir que a qualidade de software é definida em termos de adequação do produto aos propósitos para as quais ele foi desenvolvido.

A qualidade de software tem sido uma preocupação real em projetos e empresas, por isso esforços têm sido realizados para alcançar a qualidade tanto do produto de software quanto dos processos envolvidos durante o desenvolvimento e manutenção. A qualidade do produto é o que as empresas buscam, e a qualidade do processo é um dos meios para se conseguí-la. Dessa maneira, a qualidade de software possui dois aspectos a se avaliar, a qualidade do produto e a qualidade do processo.

A qualidade do produto tem como objetivo garantir que o produto tenha o efeito requerido em um contexto de uso particular (ISO, 2008a). Especificação e avaliação da qualidade do produto de software são fatores chave para garantir qualidade adequada e pode ser alcançada pela definição de características ou atributos relevantes de qualidade. Assim, a qualidade do produto de software pode ser avaliada medindo-se os atributos internos, os atributos externos ou os atributos de qualidade em uso.

Alguns padrões ou normas podem ser utilizados para garantir e verificar qualidade do produto de software. Uma delas é o conjunto de normas SQuaRE (ISO, 2005b), que trata da caracterização e medição de qualidade de produtos de software. Ela tem como objetivo melhorar e unificar os três principais processos pertinentes à qualidade de software, sendo eles a especificação de requisitos, a medição e avaliação de qualidade. As normas SQuaRE são a reformulação das normas ISO/IEC 9126 e ISO/IEC 14598 e possui cinco divisões: requisitos de qualidade, modelo de qualidade, gerenciamento de qualidade, medições e avaliação.

Já a qualidade do processo diz respeito à adoção de uma abordagem de processo para o desenvolvimento, implementação e melhoria da eficácia de um sistema de gestão

da qualidade para aumentar a satisfação do cliente pelo atendimento aos seus requisitos. Abordagem de processo trata da identificação e interações de processos, e da gestão dele para se atingir os resultados desejados (ISO, 2008a).

Em um sistema de gestão de qualidade a abordagem de processo enfatiza a importância do entendimento e atendimento dos requisitos, a necessidade de considerar os processos em termos de valor agregado, a obtenção de resultados de desempenho e eficácia de processo, e a melhoria contínua de processos baseada em medições objetivas (ISO, 2008a).

### 2.1.1 Qualidade de produto

A ISO/IEC 25010 é uma das normas SQuaRE e define um modelo de qualidade de produto detalhado. Através de um modelo hierárquico de características de qualidade, o qual descreve o que se espera de um produto de software. Nessa norma são definidos os conceitos de qualidade em uso e qualidade do produto.

A SQuaRE é dividida em três modelos de qualidade: modelo de qualidade em uso, modelo de qualidade do produto e o modelo de qualidade de dados presente na ISO/IEC 25012 (ISO, 2008b). Eles três juntos constituem uma estrutura para garantir que todas as características de qualidade sejam consideradas.

É definida na norma SQuaRE uma estrutura de modelo de qualidade, estrutura essa que foi usada nesse trabalho para organizar as características de qualidade de um processo.

De acordo com a ISO/IEC 25010 (2011), a qualidade do produto é categorizada em características e subcaracterísticas, como é mostrado na Figura 2.

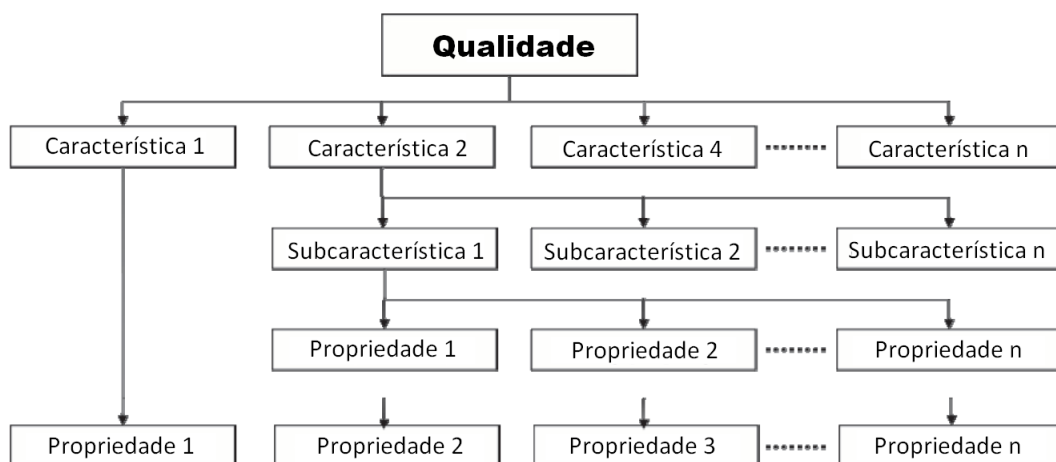


Figura 2 – Estrutura do modelo de qualidade. Adaptada da ISO/IEC 25010 (ISO, 2011)

O modelo de qualidade em uso define cinco características como pode-se observar

na Figura 3: efetividade, eficiência, satisfação, ausência de risco e cobertura de contexto. A qualidade em uso caracteriza o impacto que o produto tem sobre o usuário (ISO, 2011).

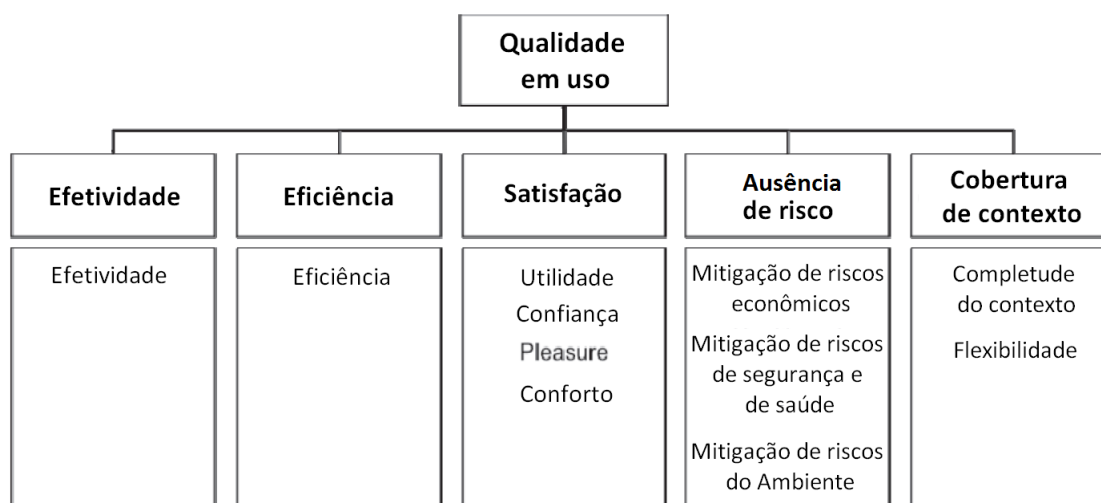


Figura 3 – Modelo de qualidade em uso. Adaptada da ISO/IEC 25010 (ISO, 2011)

Já o modelo de qualidade do produto caracteriza as propriedades do produto em oito características: adequabilidade funcional, eficiência do desempenho, compatibilidade, usabilidade, confiabilidade, segurança, manutenibilidade e portabilidade. A Figura 4 mostra cada característica e subcaracterísticas associadas.

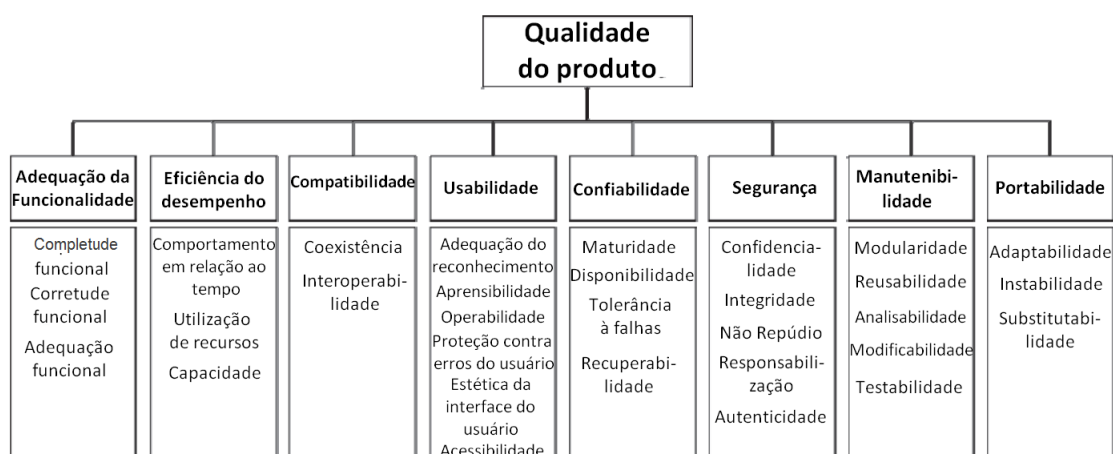


Figura 4 – Modelo de qualidade do produto. Adaptada da ISO/IEC 25010 (ISO, 2011)

Cada uma das características e subcaracterísticas de qualidade em uso e qualidade do produto são definidas na norma. Como exemplo dessa definição é apresentado a seguir a descrição de compatibilidade, uma das características da qualidade de produto.

“**Compatibilidade:** Grau em que um produto, sistema ou componente pode trocar informações com outros produtos, sistemas ou componentes, e/ou realizar as suas funções necessárias, ao compartilhar o mesmo ambiente de hardware ou software” (ISO, 2011).

### 2.1.2 Qualidade de Processo

Uma questão fundamental no gerenciamento de qualidade é que a qualidade do processo de desenvolvimento afeta a qualidade do produto desenvolvido. Qualidade de software é um conceito complexo que não é diretamente comparável com a qualidade na produção. Na produção, a noção de qualidade tem sido que o produto desenvolvido deve estar de acordo com a especificação (Sommerville, 2007, p. 644).

Seguindo a ideia de um sistema de produção comum onde a qualidade do produto está relacionada à qualidade do processo de produção, pode-se tomar como exemplo um sistema de produção automotiva em que seu processo consiste na configuração, criação e operação das máquinas. Se as máquinas forem operadas corretamente, conseqüentemente, isso resultará em um produto de qualidade, já que em um sistema de produção o processo é fácil de padronizar e controlar. Isso é, diferente de um software em que o processo de desenvolvimento é mais complexo e abstrato que um processo mecânico, e a influência de habilidades individuais e experiência do indivíduo é significativa para a qualidade do processo (Sommerville, 2007, p. 644).

Em um processo de desenvolvimento de software, a relação entre a qualidade do processo e a qualidade do produto é mais complexa, pois é difícil de medir a características da qualidade de software. Além disso, é difícil dizer como uma mudança no processo vai afetar o produto de software pelo fato dele ser mais elaborado e complexo.

Para gerenciar a qualidade de software, é preciso definir o padrão de processo, monitorar o desenvolvimento do processo para garantir que o padrão esteja sendo seguido e relatar o processo do software para o gerente do projeto e para o cliente. A gestão de qualidade abrange também um processo de garantia de qualidade de software, tarefas específicas de garantia de qualidade e controle de qualidade, prática de engenharia de software efetiva, controle de todos os produtos de trabalho de software e das modificações feitas neles, um procedimento para garantir a satisfação de normas de desenvolvimento de software e mecanismos de medição e relatório (Pressman, 2006, p. 577).

O processo de garantia da qualidade fornece a garantia de que os processos e produtos de software, no ciclo de vida do projeto, estejam em conformidade com os requisitos

especificados e sejam aderentes aos planos estabelecidos (ISO, 2008c).

Alguns padrões ou normas podem ser utilizados para garantir e verificar qualidade do processo de desenvolvimento de software. Existem diversos padrões de qualidade como ISO 9000 (ISO, 2005a), ISO 12207 (ISO, 2008c).

Uma melhor avaliação de software só pode ser obtida se incluir todas as áreas de qualidade de software, tanto qualidade do produto quanto qualidade do processo, pois concentrar-se apenas no processo é limitante. Para um melhor software, não se deve apenas concentrar em qualidade de processo, mas sim em uma ligação mais estreita entre a qualidade do processo e a qualidade do produto e no compromisso ativo dos envolvidos no projeto. A fim de estabelecer, manter e fortalecer essa ligação, é necessário medir o produto em relação à especificação do produto, desenvolvimento e manutenção do produto, às visões subjetivas dos envolvidos no projeto, e em relação a qualidade do processo (Tyrrell, 2000).

### 2.1.3 Qualidade de modelagem de processo

Modelagem de processo é a representação de abstrações definidas de acordo com objetivos e interesses dos envolvidos nos processos (Storolli, Zanolla, Guidini, and Borsoi, 2009).

O modelo de processo deve possuir quatro elementos básicos chamados de atividade, produtos, papéis e ferramentas (Derniame, Kaba, and Wastell, 1999). Atividades são os passos do processo, os quais detalham o modo de execução do processo. Produtos são artefatos gerados durante ou ao fim do processo. Papéis descrevem as responsabilidades de quem executa a atividade. Ferramentas são usadas na produção do software.

Para representar o modelo de processo são usadas linguagens de modelagem de processos. Os modelos obtidos com essas linguagens obedecem a critérios sintáticos e semânticos dos conceitos notacionais. Esses conceitos compõem o vocabulário básico para representar a arquitetura de processo (Storolli, Zanolla, Guidini, and Borsoi, 2009).

Nos últimos doze anos, muitas linguagens de modelagem de processo foram desenvolvidas, como BPM, UML, Modal, eSPEM, DPEL, PBOOL, entre outras (Zamli and Lee, 2001). Zamli (2001) apresenta características básicas de linguagem de modelagem de processo, propondo uma taxonomia para linguagem de modelagem de software.

Segundo Zamli (Zamli and Lee, 2001), as características de uma linguagem de modelagem são modelagem, aprovação, avaliação, evolução e dimensão humana.

- Modelagem: uma linguagem de modelagem deve ser capaz de representar os quatro elementos básicos de um processo, ou seja, deve representar atividades, artefatos,



papéis e ferramentas. Além disso, deve ser possível representar a comunicação entre atividades e papéis.

- **Aprovação:** para apoiar diretamente as atividades dos engenheiros de software, um modelo de processo precisa ser aprovado. Existem dois tipos de aprovação: aprovação em um ambiente distribuído e aprovação para modelos de processo de software incompletamente especificados.
- **Avaliação:** já que uma modelagem está sendo usada para guiar práticas da engenharia de software, é necessário que seja provido um meio para medição de questões relevante à linguagem de modelagem de processo, para que assim ocorra a avaliação da modelagem.
- **Evolução:** deve ser possível fornecer um mecanismo de modelo de processo de software para retomar de onde a execução parou e recuperar estados anteriores de produtos ou artefatos, atividades do processo e recursos.
- **Dimensão humana:** A engenharia de software envolve a coordenação de pessoas fazendo trabalho colaborativo. Assim, a linguagem de modelagem deve expressar da melhor maneira possível as questões do trabalho colaborativo. A dimensão humana é categorizada como apoio visual da notação, consciência do usuário e visualização do processo.

#### 2.1.4 Melhoria de processos de software

Qualidade de software é um importante fator de sucesso para a indústria. As mudanças que ocorrem nos ambientes de negócios têm impulsionado empresas a modificarem suas estruturas organizacionais e seus processos, com o intuito de melhorar seus processos e produtos conforme os padrões internacionais de qualidade. Durante a última década, a indústria de software tem se preocupado mais e mais com a melhoria de processo de software (Dyba, 2005).

Melhoria de processo de software é a disciplina de caracterizar, definir, medir e melhorar os processos de gestão e engenharia de software, levando a uma bem sucedida gestão de engenharia de software, maior a qualidade do produto, maior inovação de produtos, ciclos mais rápidos e menores custos de desenvolvimento, ao mesmo tempo (Rico, 2000).

Existem diversas pesquisas e estudos de casos que tratam a respeito da melhoria de processo de software (Dyba, 2005). Com a busca constante por melhoria de processos, surgiram vários modelos e normas de qualidade que têm o objetivo de descrever as melhores práticas a se aplicar em processos de software. Os modelos de qualidade mais utilizados são CMMI e MPS. BR. Todos eles basearam-se na ISO/IEC 12207 e 15504 para

definir, respectivamente, o escopo de processos a serem incluídos no modelo, e a forma de avaliar a qualidade dos processos.

#### 2.1.4.1 ISO 12207

A ISO 12207 tem como objetivo fornecer um conjunto definido de processos para facilitar a comunicação entre compradores, fornecedores e outras partes interessadas no ciclo de vida de um produto de software. Ela descreve 43 processos que são executados durante todo o ciclo de vida de um software.

Os processos são distribuídos em duas categorias: contexto do sistema, que contém 25 processos, e Específicos de Software, que contém 18 conforme ilustrado na Figura 5.

A categoria de contexto de sistema é dividida em subgrupos: processos de acordo, de projeto, técnicos e organizacionais. Os **processos de acordo** ou processos contratuais são os processos necessários para estabelecer um acordo entre duas organizações. Com relação aos **processos de projeto**, esses descrevem os processos que se preocupam com o planejamento, avaliação e controle do projeto. Já os **processos técnicos** são usados para definir os requisitos para o sistema, para transformar os requisitos em um produto efetivo, para permitir a reprodução consistente do produto onde for necessário, para usar o produto e prover os serviços requeridos. Os **processos organizacionais** são usados para estabelecer uma organização que possui uma estrutura constituída por processos de ciclo de vida e pessoal associados, e são usados também para melhorar continuamente a estrutura e os processos definidos..

A outra categoria é a de processos Específicos de software que é dividida em três subgrupos: processos de implementação de software, processos de apoio à implementação de software e processos de reuso.

Os **processos de implementação de software** são aqueles usados para produzir um elemento específico do sistema implementado no software, processos ligados diretamente a produção do software como análise de requisitos, construção, teste de software, entre outros. Já os **processos de apoio a implementação de software** possibilitam a execução de processos específicos de software por meio de um conjunto de atividades necessárias para produzir um software com a qualidade esperada. E, por fim, **os processos de reuso** que são processos que apoiam a habilidade da organização em reusar itens de software através dos limites do projeto, a fim de evitar a criação o de um mesmo código em várias partes do software.

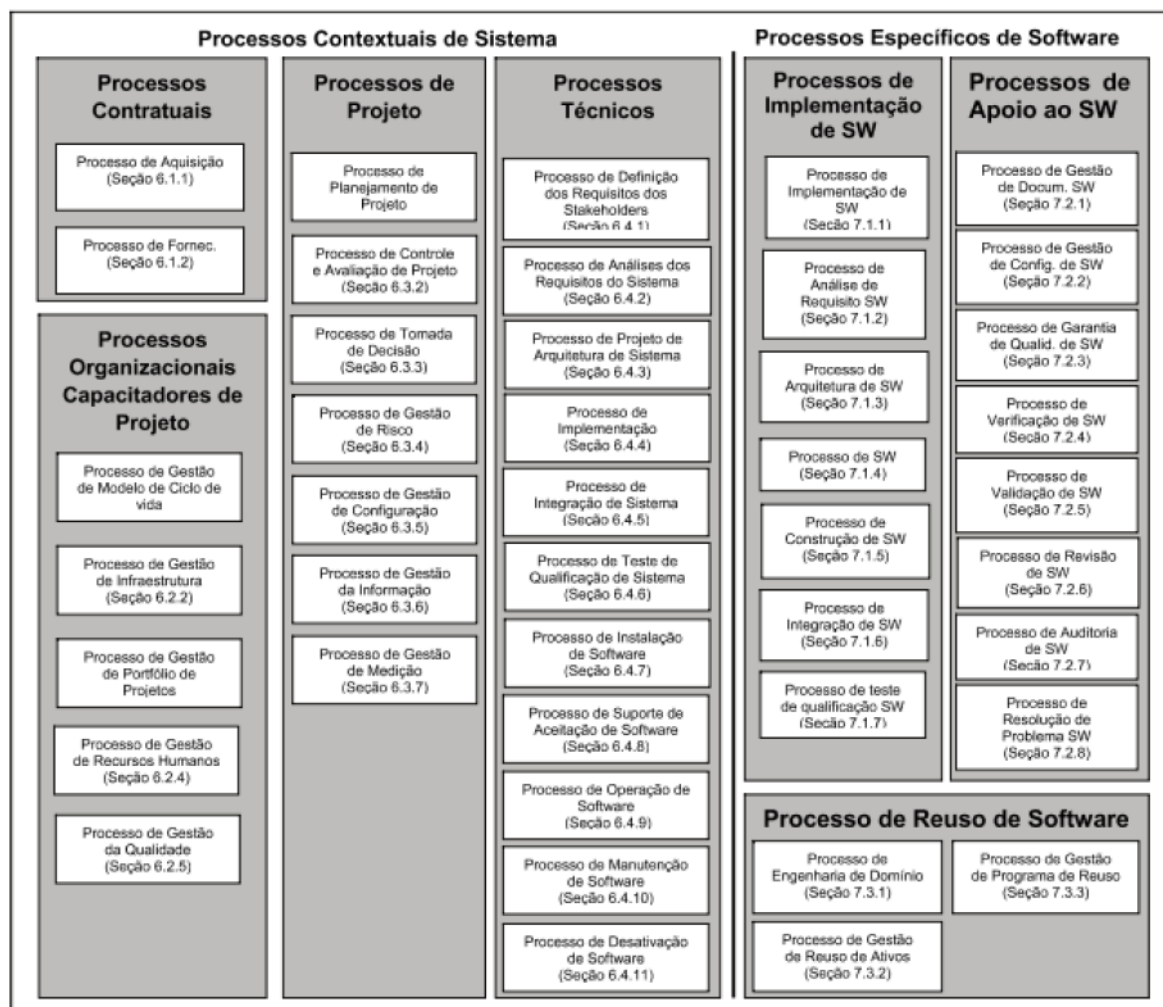


Figura 5 – Grupos do Processo de Ciclo de vida. Adaptado:(ISO, 2008c)

#### 2.1.4.2 ISO/IEC 15504

A norma ISO/IEC 15504 (ISO, 1998), fornece um modelo para avaliação de processos de software. Essa norma tem o objetivo de determinar a capacidade dos processos de uma empresa, determinar o estado do processo de uma empresa e orientar uma empresa para melhorias no seu processo.

Essa norma é dividida em dez partes: a parte 1 que apresenta os conceitos e o vocabulário; a parte 2 que apresenta a estrutura do processo de avaliação; a parte 3, a qual apresenta as recomendações para realizar uma avaliação; a parte 4 que apresenta as recomendações para melhoria de processos e determinação de capacidade; a parte 5 apresenta um exemplo de aplicação para software; a parte 6 apresenta uma aplicação para sistemas; a parte 7 apresenta a avaliação de maturidade organizacional; a parte 8 apresenta um exemplo de aplicação para gestão de serviços de TI; a parte 9 apresenta os perfis de processos alvo; e por fim, a parte 10 apresenta a extensão de segurança.

A avaliação do processo determina a capacidade dos processos de uma organização.

A parte 2 da norma define essas capacidades, são definidos seis níveis de capacidade de processos:

- Nível 0** - Processo Incompleto: Nesse nível existe uma falha na satisfação do propósito do processo. Existem poucos artefatos de trabalho ou resultados de processo.
- Nível 1** - Processo executado: O propósito do processo é geralmente alcançado, pode não ser rigorosamente planejado e monitorado. Nesse nível existem artefatos de trabalho para o que processo evidencie a satisfação do propósito do processo.
- Nível 2** - Processo gerenciado: o processo produz artefatos de acordo com os procedimentos especificados. Nesse nível o processo é planejado e monitorado. A diferença para o nível 2 é que a execução do processo entrega artefatos que cumprem com os requisitos especificados, dentro de um cronograma de tempo.
- Nível 3** - Processo estabelecido: o processo é executado e gerenciado usando um processo definido. A implantação desse processo utiliza versões adaptadas de um processo padrão aprovado. O processo estabelecido utiliza um processo padrão que é capaz de alcançar os resultados definidos.
- Nível 4** - Processo previsível: O processo definido é executado na prática, dentro de limites de controle definidos. São coletadas e analisadas medições do desempenho do processo. Dessa forma, a qualidade do produto é reconhecida de forma quantitativa. O processo passa a ser executado dentro de limites definidos a fim de atingir seus resultados definidos.
- Nível 5** - Processo em otimização: Nesse nível a execução do processo é continuamente melhorada e o processo atinge repetibilidade no cumprimento de suas metas de negócios definidas. A principal diferença para os outros níveis é que os processos definidos e padronizados se adaptam para atender com eficácia os objetivos de negócio, atuais e futuros, de maneira dinâmica.

A forma de avaliação da norma ISO/IEC 15504 difere da forma de avaliação apresentada neste trabalho, pois essa norma avalia o processo de software em nível de execução do processo. Já este trabalho avalia o processo de software em nível de documentação, ou seja, avalia a documentação ou a modelagem do processo, antes da execução dele.

### 2.1.5 Método de implantação de MPS

Um método de implantação de melhoria de processo de software descreve um processo para a condução de uma iniciativa de MPS, e tem como objetivo orientar o desenvolvimento a longo prazo, integrar um plano para iniciar e gerenciar um programa

de melhoria de processo de software (McFeeley, 1996). O método de implantação mais utilizado atualmente é o modelo IDEAL.

Modelo IDEAL é um modelo que visa a melhoria de um processo de software, e serve como guia para iniciar, planejar e implementar ações de melhoria. A palavra IDEAL é uma sigla em inglês para Iniciar (*initiating*), Diagnosticar (*diagnosing*), Estabelecer (*establishing*), Agir (*acting*) e Aprender (*learning*). Essas siglas consistem nas fases do modelo de melhoria. A Figura 6 ilustra o ciclo de melhorias e as fases citadas.

A **Fase Iniciação** consiste em estabelecer um alicerce para um programa de melhoria, criando um plano de melhoria, estabelecendo uma infraestrutura inicial, regras e definindo responsabilidade. Na **Fase Diagnóstico** é determinado onde a organização está e onde ela deseja estar. Já na **Fase Estabelecimento** é realizado o planejamento dos detalhes de como a organização alcançará o objetivo definido na fase anterior, estabelecendo também suas prioridades. Na **Fase Ação** ocorre a execução do trabalho de acordo com o planejado. E, por fim, na **Fase de Aprendizado** acontece o aprendizado sobre as experiências e a definição de melhorias para novos ciclos.

Um modelo de qualidade de processos pode ser utilizado para guiar toda a execução de um método de implantação de MPS.



Figura 6 – Modelo IDEAL. Adaptado de (McFeeley, 1996)

## 2.2 Medição

A medição é fundamental para a compreensão e controle de processos e produtos de desenvolvimento de software, oferecendo um mecanismo para avaliação objetiva (Pressman, 2006).

A medição de software tem o objetivo de melhorar de forma contínua tanto de um processo quanto de um produto ou serviço de software. Pode ser usada, também, para auxiliar na estimativa, no controle da qualidade, na avaliação de produtividade e no controle do projeto, e avaliar a qualidade dos produtos de trabalho.

Segundo Park, Goethert e Florac (1996) existem várias razões para se medir um processo ou produto de software, sendo elas: obter entendimento de processos, produtos, recursos e ambiente; estabelecer referências para comparação com futuras avaliações; avaliar a fim de determinar o estado em relação aos planos definidos; aperfeiçoar, pela identificação de bloqueios, causas fundamentais, ineficiências e outras oportunidades; e melhorar a qualidade do produto e o desempenho do processo.

A medição pode ser realizada ao longo de um projeto de software, durante as diversas fases do ciclo de vida de um software. Por exemplo, podem ser medidas as características de software para verificar se requisitos estão completos; se o projeto definido tem boa qualidade; se o processo definido pode garantir que o produto será entregue no prazo, se o código é consistente, de fácil entendimento e está pronto para ser testado. Na fase de manutenção, a medição pode ser usada para avaliar se o produto pode ser melhorado. Dessa maneira, é visto que a medição pode ser usada para garantir a qualidade de todo o processo e projeto de software durante seu desenvolvimento.

Segundo Roche (1994), um processo de medição pode ser caracterizado por cinco atividades:

1. Formulação: derivação de medidas e métricas de software adequadas para a representação do software que está sendo considerado.
2. Coleta: mecanismo usado na obtenção dos dados necessários para derivar as métricas formuladas.
3. Análise: cálculo de métricas por meio de ferramentas matemáticas.
4. Interpretação: Avaliação das métricas de forma a se obter profundidade nas características do sistema e respaldo para a tomada de decisão.
5. Realimentação: recomendações derivadas da interpretação das métricas de produto, transmitidas à equipe de software.

Ao se medir um processo, definem-se métricas com o objetivo de fornecer um conjunto de indicadores visando o aperfeiçoamento do processo de software a longo do prazo.

Existem diversas abordagens para medição de processo de software, neste trabalho serão apresentadas duas abordagens: o *Practical Software Measurement*(PSM) e o *Goal/Question/Metric*(GQM).

### 2.2.1 Practical Software Measurement (PSM)

PSM é uma abordagem para medição de processos de software que tem o objetivo de estabelecer um processo de implantação de medição de projetos de software. O PSM tem também o objetivo de prover bases de informações e comunicações para tomada de decisões e estabelecer uma base para melhorar o gerenciamento organizacional e executivo, resultando na melhoria da maturidade dos processos de software.

O PSM se concentra na medição de nível de projeto e mostra como a medição pode ser adaptada para satisfazer às necessidades de cada projeto. Além disso, o PSM fornece orientação baseada na experiência em como definir e implementar um processo de medição (McGarry et al., 2002).

O PSM é estruturado em dois modelos complementares: o modelo de informação e o modelo de processo. O modelo de informação aborda o desenvolvimento de uma estrutura de informações de medição do projeto, provê a estrutura de organização das medidas a serem executadas, os resultados de operações sobre estas medidas, até o entendimento das necessidades de informação. Já o modelo de processo descreve as atividades e tarefas relacionadas à medição a serem executadas (Card, 2003).

O PSM se enquadra na categoria de Abordagens Dirigidas pelas Informações (*Information Driven Measurement Approach*), que é constituída de coleta, análise e elaboração de relatórios de dados de medição que se relaciona diretamente com a necessidade de informação das tomadas de decisões de um projeto (McGarry, Card, Jones, Layman, Clar, Dean, and hall, 2002). O PSM também provê informações aos profissionais que tomam decisões nos projetos de software. Dessa maneira, percebe-se que o PSM possui uma forte ligação com a gestão de projetos, sendo uma abordagem baseada na medição de software para gerenciamento de projetos

De acordo com McGarry (2002), o PSM define sete categorias de informação de software que ajudam na identificação e priorização das informações específicas que um projeto precisa:

1. Programação e progresso: aborda a realização de etapas do projeto e a conclusão das unidades de trabalho individuais.

2. Recursos e custo: refere-se ao equilíbrio entre o trabalho a ser executado e os recursos de pessoal designado para o projeto.
3. Tamanho do produto e estabilidade: aborda a estabilidade da funcionalidade ou a capacidade necessária do software.
4. Qualidade do produto: trata da capacidade do produto de software entregue para apoiar as necessidades do usuário sem falhas.
5. Desempenho do processo: refere-se à capacidade do fornecedor em relação às necessidades do projeto.
6. Eficácia tecnológica: aborda a viabilidade da abordagem técnica proposta.
7. Satisfação do cliente: aborda o grau em que os produtos e serviços prestados pelo projeto atendem às expectativas do cliente.

O modelo de processo do PSM é composto por quatro atividades-chave: (1) estabelecer e sustentar comprometimento; (2) planejar a medição; (3) executar a medição; e (4) avaliar a medição. A Figura 7 apresenta cada uma dessas atividades.

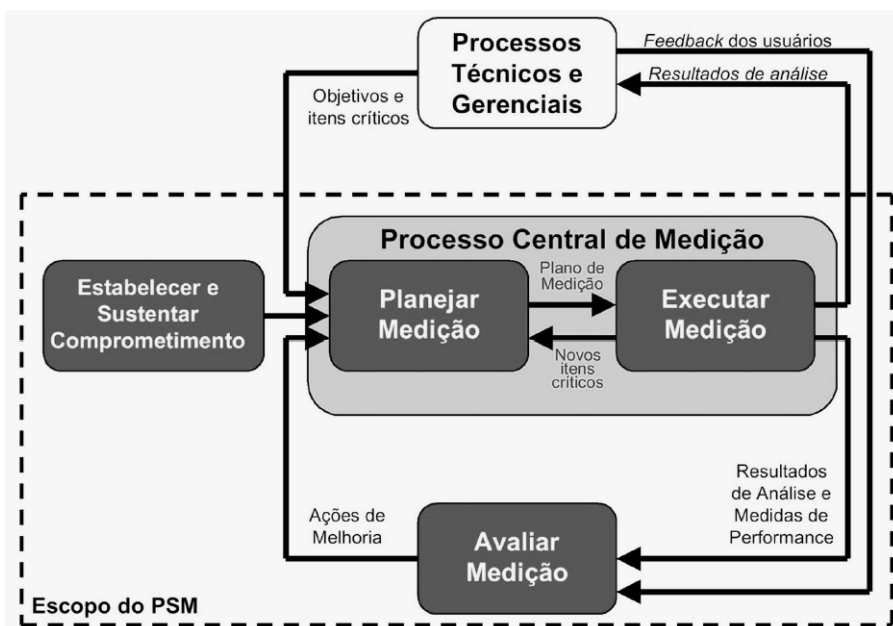


Figura 7 – Escopo do PSM. Fonte: PSM (McGarry et al., 2002)

Como apresentado na Figura 4, pode-se identificar as principais atividades do modelo PSM:

- **Estabelecer e sustentar comprometimento:** possui tarefas como obter comprometimento organizacional, definir responsabilidades, prover recursos, e rever o progresso do programa de mensuração.



- **Planejar medição:** tem o objetivo de definir as métricas, integrando as partes administrativas do projeto com as partes técnicas, sempre buscando o mais baixo custo. Envolve três tarefas: identificar e priorizar as necessidades de informação, selecionar e especificar medidas e integrar medições nos processos dos projetos.
- **Executar medição:** tem o objetivo de executar as métricas planejadas. Consiste nas tarefas de coletar processos de dados, analisar informações, e fazer recomendações.
- **Avaliar medição:** tem o objetivo de avaliar os resultados de análises realizadas na execução e avaliar as medidas de desempenho, propondo, no fim da avaliação, ações de melhoria. Consiste nas tarefas de avaliar medidas e indicadores, avaliar o processo de medida, identificar melhorias, e atualizar base de conhecimento.

### 2.2.2 Goal/Question/Metric (GQM)

O GQM é utilizado para definir e avaliar um conjunto de objetivos operacionais usando medição. Representa uma abordagem sistemática para definir e integrar a modelos de processos, produtos e perspectivas de qualidade baseados em necessidades específicas do projeto e organizações através de um programa de medição (Basili, 1992).

O GQM segue uma estrutura em que se inicia definindo um objetivo de medição. Por meio do objetivo são definidas questões de medição, e para cada questão são definidas as métricas, a fim de respondê-las. Essa estrutura é dividida em três níveis: (1) o nível conceitual, no qual há a definição dos objetivos para um objeto ou produto, processo ou recurso; (2) o nível operacional, no qual há a definição de um conjunto de questões usadas para caracterizar um modo de alcançar um determinado objetivo; (3) o nível quantitativo, onde há a definição de métricas, ou seja, um conjunto de dados é associado a cada questão a fim de respondê-la de maneira quantitativa (Basili, Caldiera, and Rombach, 1994).

O processo do GQM é realizado em três fases: (1) a fase de planejamento, (2) fase de execução e (3) fase de análise e armazenamento de dados. A **fase de planejamento** tem o objetivo de coletar toda a informação necessária para o programa de métricas, preparar e motivar os membros da organização e definir objetivos, cronogramas e responsabilidades, ferramentas e métodos.

Já a **fase de execução** tem o propósito de executar o processo, construir o produto, coletar e validar os dados, e analisar em tempo real para fornecer um feedback para uma ação de correção no projeto.

Por fim, na **fase de análise e armazenamento de dados**, é realizada uma análise dos dados e as informações são recolhidas para avaliar as práticas ocorrentes, determinar os problemas e registrar as descobertas. Além disso, são feitas recomendações

para futuros projetos, é armazenada a experiência adquirida, e são refinados os modelos e outras formas de conhecimento estruturado (Basili, 1992).

A abordagem GQM possui um foco na definição de questões, e é usado quando for necessário fazer a medição de um projeto de software como um todo (Ghani, Muketha, and Wen, 2008).

Tendo em vista essas duas abordagens de medição, o PSM e o GQM, pode-se concluir que o PSM foi desenvolvido para ajudar nos problemas existentes em projetos de software, tanto nos aspectos técnicos quanto gerenciais. O PSM é uma abordagem mais abrangente que foca, por exemplo, em todo um processo ou projeto. O PSM define suas métricas a partir de propriedades ou características de informação. Já o GQM é uma abordagem mais específica, pois define questões para um objetivo específico e a partir das questões são definidas as métricas. Assim, tendo em vista o tema deste trabalho (a análise da qualidade da modelagem de processo de software), e o modo como a pesquisa será conduzida, (identificando características e propriedades de qualidade de modelagem de processo de software e definindo métricas a partir dessas propriedades), percebe-se que a abordagem que melhor se adapta a este trabalho é o PSM.

### 3 Metodologia

Este capítulo tem como objetivo descrever a forma como esta pesquisa foi conduzida. Para tanto, inicialmente é importante entender como esta pesquisa pode ser classificada.

Segundo Silva (2001), do ponto de vista da natureza da pesquisa, esta é uma pesquisa aplicada com o objetivo de gerar conhecimento para a solução de um problema específico. Com relação aos objetivos do trabalho, a pesquisa possui um fim descritivo, já que visa descrever as características que determinam a qualidade do processo. Em relação aos procedimentos técnicos ou meios de investigação utilizados, foi realizado um mapeamento sistemático para identificar as características de qualidade de processo e aplicado um questionário visando identificar as características mais relevantes e validar o modelo de qualidade de processo construído.

Para responder as questões de pesquisa e alcançar os objetivos apresentados no Capítulo 1, este trabalho foi dividido em algumas fases de pesquisa sendo elas: fase decisória, fase construtiva e fase redacional (Silva, 2001), como é mostrado na Figura 8.

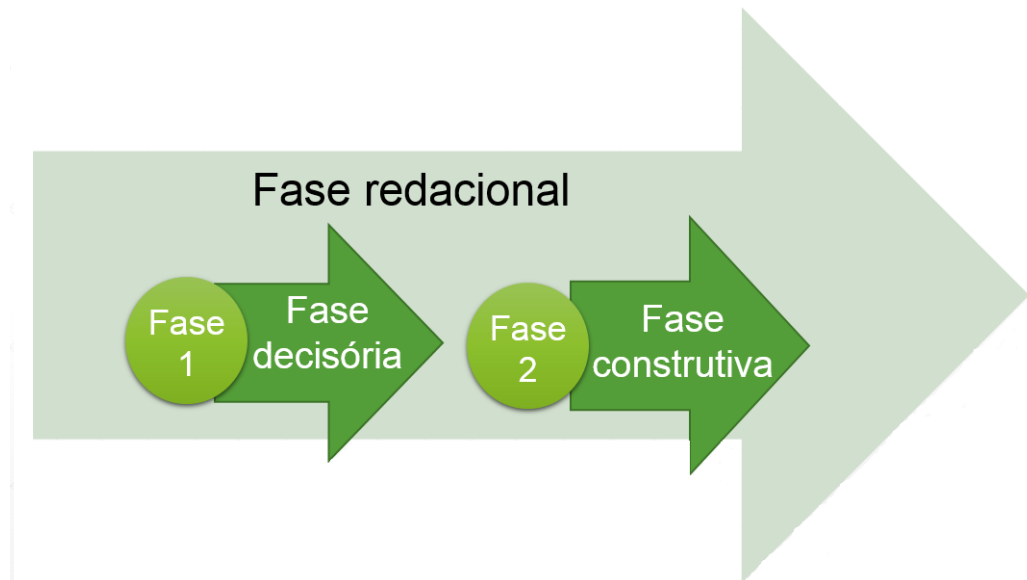


Figura 8 – Fases de pesquisa

Na primeira fase da pesquisa, fase decisória, foi realizado um mapeamento sistemático, uma metodologia de pesquisa para auxiliar os estudos a respeito do tema proposto, qualidade de modelagem de processo de software. Nessa fase, foram identificadas informações disponíveis na literatura a respeito de definição de processo, de modelagem de processos de software e sobre métricas de processo. A partir desse mapeamento sistemá-

tico, foi possível identificar as características de qualidade de uma modelagem de processo de software e, além disso, foram identificadas métricas para verificar a qualidade de uma modelagem de processo. Nesta primeira fase, foi possível alcançar o primeiro dos três objetivos específicos de pesquisa propostos nesse trabalho que é identificar características de uma boa modelagem de processo

Na segunda fase da pesquisa, fase construtiva, as características identificadas foram validadas por especialistas. Para tanto, foram selecionadas apenas as características mais relevantes, levando em consideração a visão de pessoas com experiência em modelagem de processos. Para isso, foi aplicado um questionário para obter a opinião dos especialistas com dois focos: validação semântica e validação sintática. A validação semântica foi feita por meio de um questionário e objetivou verificar a clareza das características. Já a avaliação sintática foi feita aplicando-se o mesmo questionário, considerando-se as modificações aderidas da validação semântica. A validação sintática verificou a importância das características do modelo. A partir dessas características foi validado o modelo de qualidade de modelagem de processo de software, cujas características foram identificadas com o mapeamento sistemático.

A fase redacional ocorreu durante todo o trabalho, e documentou os dados obtidos, a análise realizada e elaborou este relatório.

Este capítulo está organizado em duas seções: A Seção 3.1 aborda os conceitos relacionados a mapeamento sistemático e a apresenta como ele foi conduzido nesta pesquisa. Já a Seção 3.2, apresenta os conceitos e a aplicação do survey desenvolvido para validar o modelo de características de qualidade elaborados a partir do mapeamentos sistemático.

## 3.1 Mapeamento Sistemático

Um estudo secundário é obtido através da análise de diversos estudos primários com o objetivo de caracterizar um determinado tema de pesquisa. Dessa forma, estudos primários funcionam como fonte de informação de estudos secundários (Mafra and Travassos, 2006).

Um estudo secundário tem como objetivo identificar, avaliar e interpretar todos os resultados relevantes a um determinado tópico de pesquisa, fenômeno de interesse ou questão de pesquisa (Kitchenham, 2004). Revisões sistemáticas são tipos de estudos secundários, pois se utilizam de literatura já existente sobre estudos primários anteriores para selecionar as evidências disponíveis sobre o tema de pesquisa.

Uma Revisão Sistemática (RS) de literatura é um meio de identificar, avaliar e interpretar todas as pesquisas disponíveis relevantes para uma questão de pesquisa específica, ou área temática, ou fenômeno de interesse (Kitchenham and Charters, 2007). Ela

deve ser desenvolvida de maneira formal e sistemática, isso significa que o processo de condução da pesquisa de uma RS segue uma sequência bem definida e estrita de passos metodológicos, de acordo com um protocolo desenvolvido (Biolchine, Mian, Natali, and Travassos, 2005).

A RS tem por objetivo construir uma síntese da pesquisa existente que seja: justa, não seja tendenciosa; rigorosa, de acordo com o procedimento definido; aberta, assegurando que o procedimento de revisão esteja visível para outros pesquisadores; e objetiva, assegurando que o procedimento de revisão seja reproduzível para outros pesquisadores (Mafra and Travassos, 2006).

O Mapeamento Sistemático (MS) é também considerado como um estudo secundário, visto que é um método projetado para fornecer uma ampla visão de uma determinada área de pesquisa de estudos primários, que permite identificar, quantificar e analisar os resultados, estabelecendo evidências sobre um determinado tema (Kitchenham and Charters, 2007; Arksey and O'Malley, 2005). A metodologia utilizada para fazer um estudo de MS é apoiada na mesma visão da metodologia de uma RS, pois é conduzida de maneira auditável, rigorosa e transparente (Arksey and O'Malley, 2005).

O MS é recomendado quando existe uma lacuna relevante de estudos primários e/ou o tema de pesquisa seja muito amplo (Kitchenham and Charters, 2007).

Além disso, o MS é útil para conhecer a extensão, alcance e natureza das áreas de pesquisa, mapeando as áreas de estudo onde se tem conhecimento dos materiais que estão disponíveis; identificar a relevância da necessidade de se fazer ou não uma RS; descrever com mais detalhes os resultados de uma pesquisa em uma determinada área de estudo; proporcionando, assim, uma síntese dos resultados do MS (Arksey and O'Malley, 2005).

O MS analisa um tema mais amplo em relação à RS e seleciona os principais trabalhos de pesquisa nesse domínio. Dessa maneira, o MS pode ser de grande importância para pesquisadores de Engenharia de Software, fornecendo uma visão geral da literatura em áreas temáticas específicas (Kitchenham et al., 2010).

As principais diferenças entre RS e MS de acordo com Kitchenham (2011) são:

- Diferenças quanto às questões de pesquisa: no MS as questões de pesquisa são mais genéricas, está relacionada às tendências de pesquisa, e podem ser classificadas como pesquisas de natureza exploratória e descritiva. Já a RS possui questões de pesquisa mais específicas, focadas no ponto específico da área de estudo empírico.
- Diferença quanto aos objetivos: MS possui seu principal foco relativo à classificação, realização de análise temática da literatura. Já o foco da RS é identificar as melhores práticas e efetividade da área de estudo, tecnologias, métodos ou ferramentas.

- Diferença quanto ao escopo: o escopo do MS é mais amplo que o da RS que é um escopo focado no assunto da pesquisa.
- A divulgação dos resultados do MS pode ser mais limitada do que os resultados de uma RS

Identificadas essas diferenças, conclui-se que um MS é mais simples e menor que uma RS, e objetiva classificar informações de uma área. Como o resultado do estudo secundário conduzido nesta pesquisa tem como objetivo identificar uma lista de características de qualidade, a estratégia de MS parece ser a mais adequada.

### 3.1.1 Processo de Mapeamento Sistemático

MS é definido em cinco etapas: 1) definição de questões de pesquisa; 2) condução da pesquisa para estudos primários; 3) triagem de documentos; 4) *keywording* usando resumos, classificação do esquema; e 5) extração de dados e mapeamento do estudo (Petersen, 2008).

**Definição de questões de pesquisa** é a fase em que as questões que conduzem todos os passos do MS são definidas. Já a **condução da pesquisa para estudos primários** trata da identificação de trabalhos através de *strings* de busca em bases de dados científicas, ou através de buscas manuais em conferências ou em jornais. A **triagem de documentos** diz respeito à análise de estudos selecionados e aplicação dos critérios de inclusão e exclusão a fim de selecionar trabalhos, que, de fato, respondem as questões de pesquisa. A fase ***keywording*** tem como objetivo agrupar um conjunto de palavras-chaves e formar categorias do mapa gerado no MS, para, assim, desenvolver um entendimento de alto nível sobre a natureza da pesquisa. Por fim, na fase de **extração de dados e mapeamento do estudo** é realizada a extração e documentação do processo de extração de dados, calculando a frequência de publicações em cada categoria determinada.

### 3.1.2 Aplicação do Mapeamento Sistemático

Para a pesquisa desenvolvida neste trabalho, foi realizado um mapeamento sistemático com o objetivo de identificar as características de qualidade de modelagem de processo. A condução do MS foi realizado de acordo com o protocolo de MS definido no Apêndice A.

## 3.2 Survey

Neste trabalho foi construído um *survey* para levantamento de dados com o objetivo de avaliar quão adequado para especialistas em MPS é o modelo de características

de qualidade elaborado a partir do MS. De acordo com Fink and Kosecoff (1985), *survey* pode ser definido como um método para coletar informação de pessoas acerca de suas ideias, sentimentos, planos, crenças, bem como origem social, educacional e financeira. Esse tipo de validação foi escolhido baseado em outros trabalhos que também realizaram esse tipo de validação Simon (2004); Brito (2014).

### 3.2.1 Construção do questionário

O instrumento utilizado no *survey* foi um questionário, para medir a opinião, interesses, aspectos de personalidade e informação biográfica do respondente (Yaremko, Herari, Harrison and Lynn, 1986). O levantamento de dados por amostragem, ou *survey*, assegura melhor representatividade e permite generalização para uma população mais ampla. A pesquisa *survey* é classificada como uma pesquisa descritiva, pois visa verificar se a percepção dos fatos está ou não de acordo com a realidade Freitas et al. (2000). No caso deste trabalho, a pesquisa *survey* visa verificar se a percepção dos respondentes está de acordo com o que foi definido através do MS.

Segundo Hartmut Gunther (2003), para elaborar um questionário deve-se primeiro definir o objetivo de pesquisa em termos de conceitos a serem pesquisados e da população-alvo. O objetivo da pesquisa irá determinar os conceitos que serão investigados e a população-alvo. Os conceitos definidos derivarão os itens que são as perguntas do questionário, assim como ilustrado na Figura 9

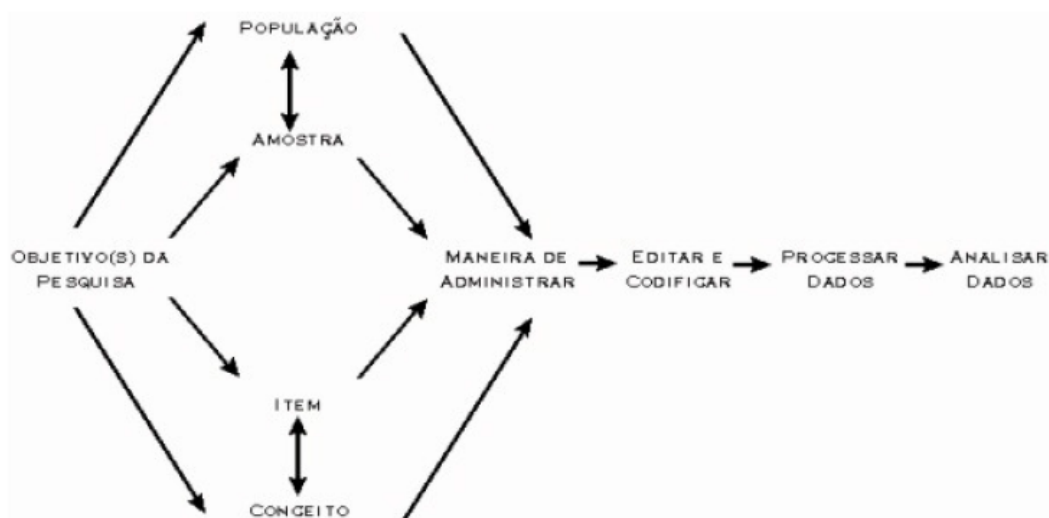


Figura 9 – Estágios principais de um survey (Schuman and Kalton, 1985)

O objetivo da pesquisa deste *survey* foi validar o modelo de características de qualidade de modelagem de processo de software elaborado através de um MS.

Com base no objetivo de pesquisa foi possível determinar a população-alvo da pesquisa e os conceitos a serem investigados pelo questionário. Os conceitos investigados

neste trabalho foram a quantidade de características de qualidade apresentadas, a importância das características. A população-alvo são pessoas que tenham experiências com modelagem de processo de software ou tenha atuado em MPS.

Para determinar a amostra do instrumento de pesquisa foi levado em consideração que a população-alvo é um grupo restrito de pessoas que trabalharam direta ou indiretamente com modelagem de processo, pois os conceitos investigados no *survey* são conceitos técnicos que apenas especialistas conseguiriam analisar. Dessa maneira, foi definido utilizar uma amostra não probabilística por conveniência, onde os participantes são escolhidos por estarem disponíveis (Freitas et al., 2000). De acordo com Everitt (1975), o número mínimo da amostra deveria ser pelo menos 10 respondentes para cada item avaliado.

Como os conceitos investigados é a importância das características, foi utilizada uma escala *Likert*, a qual é utilizada especialmente em levantamento de atitudes, opiniões e avaliações. Segundo Mattar (1996), este tipo de escala se destaca por sua simplicidade de emprego em medições ordinais e também por associar valores nominais (conceitos) a valores numéricos. Na escala *Likert* geralmente avalia-se um conceito em uma escala de cinco alternativas: aplica-se totalmente, aplica-se, nem sim nem não, não se aplica e definitivamente não se aplica (Gunther, 2003).

O questionário foi construído na ferramenta disponibilizada pela Google, com o intuito de facilitar a aplicação e a coleta de dados. O itens que compuseram o questionário foram as próprias características de qualidade de processo. Para cada característica o respondente é convidado a definir sua importância de acordo com a escala apresentada na Figura 10.

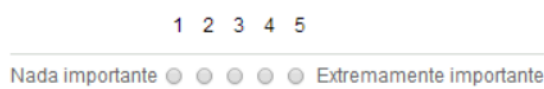


Figura 10 – Escala utilizada no questionário

Dessa maneira, foi elaborada uma primeira versão do questionário apresentado no Apêndice C. A partir desse questionário foi realizada uma validação por um grupo de professores, quatro no total, da universidade de Brasília.

### 3.2.2 Validação semântica do questionário

Depois de construído o questionário, procedeu-se com a sua validação semântica. Essa validação pôde verificar a clareza dos itens do questionário e a estrutura do instrumento.

A validação consistiu na apresentação do questionário para quatro professores da Universidade de Brasília - Faculdade Gama, três com experiência em modelagem de



processo de software e um especialista em condução e construção de questionários para survey.

A seguir seguem advindas da validação semântica:

- Recomendação que o número de questões do questionário fosse reduzido, pois com uma grande quantidade de questões é possível que os respondentes não respondam todas as questões e a grande quantidade de conceitos poderiam gerar confusão e dificuldade no momento de responder.
- Inclusão, ao final do questionário, itens destinados a caracterização de amostra.
- Inclusão da definição das subbucaracterísticas no texto descritivo das subcaracterísticas, pois dessa forma diminuiria o número de questões.
- As características sintática e adequação ao formato foram consideradas semelhantes.
- As características clareza e apreensibilidade foram consideradas semelhantes.
- As características especificidade e apoio foram consideradas semelhantes.
- A características confiabilidade e sua subcaracterística recuperabilidade foram consideradas confusas.
- Recomendação de incluir o foco da característica 'complexidade'.
- Recomendação de incluir o foco da característica de 'segurança', como segurança em resultados precisos ou proteger os dados de transação.
- A característica projetabilidade, ajustabilidade, aplicabilidade e escalabilidade foram consideradas semelhantes.
- A característica controlabilidade foi considerada relacionada à adaptabilidade.
- A característica funcionalidade foi considerada confusa, pois a definição está muito abstrata.
- A definição da característica texto alternativo foi considerada confusa.
- A definição da característica flexibilidade foi considerada confusa.
- A característica fidelidade foi considerada confusa, podendo não ser considerada uma característica do processo, mas sim da organização.
- A característica repetibilidade foi considerada confusa, podendo não ser considerada uma característica do processo, mas sim da organização.

- Recomendação de incluir os significados de correto e agradável na característica acurácia, pois essa foi dada como confusa por não se saber o que é considerado correto e agradável.
- A característica adequação ao meio de comunicação foi considerada confusa pois pode ser uma decisão externa ao processo.

As sugestões foram avaliadas e o questionário foi evoluído em uma segunda versão. Essa nova versão pode ser vista no Apêndice D.

### 3.2.3 Coleta de dados

Após os ajustes no questionário, este foi transferido para uma plataforma online, gratuita do Google Drive, onde foi garantido o sigilo e a privacidade do respondente.

A divulgação do questionário foi feita por meio do link: [https://docs.google.com/forms/d/114yxfg-e\\_a3B1Lb6jVLWnjfbrq7HVHE81oVPXq33EdM/viewform?usp=send\\_form](https://docs.google.com/forms/d/114yxfg-e_a3B1Lb6jVLWnjfbrq7HVHE81oVPXq33EdM/viewform?usp=send_form).

Através da lista de e-mails de avaliadores e implementadores do MPS.BR, lista de email de empresas avaliadas pelo MPS.BR, e para pessoas especialistas e que trabalhem com melhoria de processo ou com modelagem de processo. O instrumento foi disponibilizado no dia 30 de setembro de 2014 até o dia 15 de outubro de 2014, totalizando 15 dias de disponibilidade.

No prazo de três semanas, 20 pessoas responderam ao questionário, por se obter uma pequena população é recomendado que em trabalhos futuros aplique esse *survey* para uma população maior. As respostas preenchidas pelos respondentes foram registradas pela ferramenta que disponibilizou o instrumento na internet. Os dados foram coletados a partir da planilha eletrônica disponibilizada pela ferramenta da Google. Os dados coletados passaram por procedimentos estatísticos, a fim de validar a pesquisa e extrair informações dos dados coletados.

### 3.2.4 Avaliação estatística do questionário

Segundo Almeida (2010), um questionário devidamente elaborado deve levar em consideração dois aspectos muito importantes: sua validade e sua confiabilidade. A validade pode ser considerada como o grau no qual as pontuações de um teste estão relacionados a algum critério externo do mesmo teste, onde é realizada uma análise fatorial e é medida sua confiabilidade.

O principal método para se medir a confiabilidade de um instrumento de pesquisa é o método alfa de Cronbach, o qual mede a confiabilidade do instrumento em função do

número de itens e a proporção da variância total e da covariância entre os itens, dado que todos os itens do questionário usem a mesma escala de medição. A seguir é apresentada a equação utilizada para medir a confiabilidade, onde  $K$  é o número de itens do questionário,  $S_i$  é a variância do item e  $S_t$  é a variância total do questionário.

$$\alpha = \left(\frac{K}{K-1}\right)\left(1 - \frac{\sum_{i=1}^k S_i^2}{S_t^2}\right)$$

O grau de confiabilidade varia de acordo com o valor de alfa, esse grau pode ser visto na Figura 11.

<b>Confiabilidade</b>	<b>Valor de Alfa</b>
Muito baixa	$\alpha \leq 0,30$
Baixa	$0,30 < \alpha \leq 0,60$
Moderada	$0,60 < \alpha \leq 0,75$
Alta	$0,75 < \alpha \leq 0,90$
Muito alta	$\alpha > 0,90$

Figura 11 – Variação de confiabilidade pelo valor de alfa (adaptado de Freitas (2005))

O cálculo de confiabilidade do Alfa Cronbach do instrumento de pesquisa foi feito na ferramenta IBM SPSS Statistics IBM. O cálculo de confiabilidade apontou como resultado um valor de 0.854. Utilizando o grau de confiabilidade apresentado por Freitas (2005), o grau de confiabilidade foi alto. Dessa forma, o questionário foi considerado que possui uma boa consistência interna e uma alta confiabilidade.

Não foi realizado uma análise fatorial do instrumento de pesquisa, pois segundo Hair (1998) para uma análise fatorial o tamanho da amostra deve ser quatro a cinco vezes o número de questões, e o número de respondentes não alcançou esse tamanho de amostra. E também foi realizado o teste de adequação de amostragem de Kaiser-Meyer-Olkin (KMO teste), o qual mostra se os dados podem ser analisados pelo método de análise fatorial (Simon, 2004). Segundo o teste de KMO realizado na ferramenta SPSS Statistics, não é aconselhável utilizar o método de análise fatorial nos dados do questionário.

## 4 Resultados

Este capítulo tem como objetivo apresentar os resultados da condução deste trabalho. Esse capítulo é dividido em duas seções: a Seção 4.1 apresenta os resultados obtido no mapeamento sistemático. Já a Seção 4.2 apresenta os resultados da validação do modelo de características de qualidade.

### 4.1 Resultados do Mapeamento Sistemático

Inicialmente, por meio da escolha do tema e da definição dos objetivos deste trabalho, foi explicitada a questão de pesquisa que daria início a condução da pesquisa e a coleta de dados.

Para auxílio na condução da pesquisa e coleta de dados foi utilizada a ferramenta StArt, (*State of the Art through Systematic Review*), desenvolvida pelo Laboratório de Pesquisa em Engenharia de Software (LAPES) na Universidade Federal de São Carlos (UFSCar) (DC/UFSCar, 2005). Essa ferramenta auxilia na execução do MS, ajuda na sumarização de dados provendo gráficos interativos que dá ao pesquisador uma maior visibilidade do estado da pesquisa de dados durante o MS. A ferramenta pode ser obtida gratuitamente em endereço: <http://lapes.dc.ufscar.br>.

Para a coleta de dados foi construída uma *string* de busca, a qual foi formatada de acordo com as palavras-chaves da questão de pesquisa. Foram identificados, por meio da *string* de busca, 256 trabalhos, sendo 89 provenientes da biblioteca digital IEEEExplore, 6 da *ACM Digital Library*, 81 da *Science Direct* e 80 da *Web of Science*, como pode-se observar na Figura 12.

Após a identificação dos trabalhos foi realizado um refinamento, em que apenas 36 trabalhos foram selecionados para análise. Essa seleção teve como objetivo excluir os documentos que não possuíam relação com qualidade de processo ou modelagem de processo a partir do título e do resumo. A Figura 13 apresenta os trabalhos selecionados, excluídos e duplicados durante a seleção dos trabalhos. Já a Figura 14 apresenta os trabalhos selecionados classificados por fonte de pesquisa.

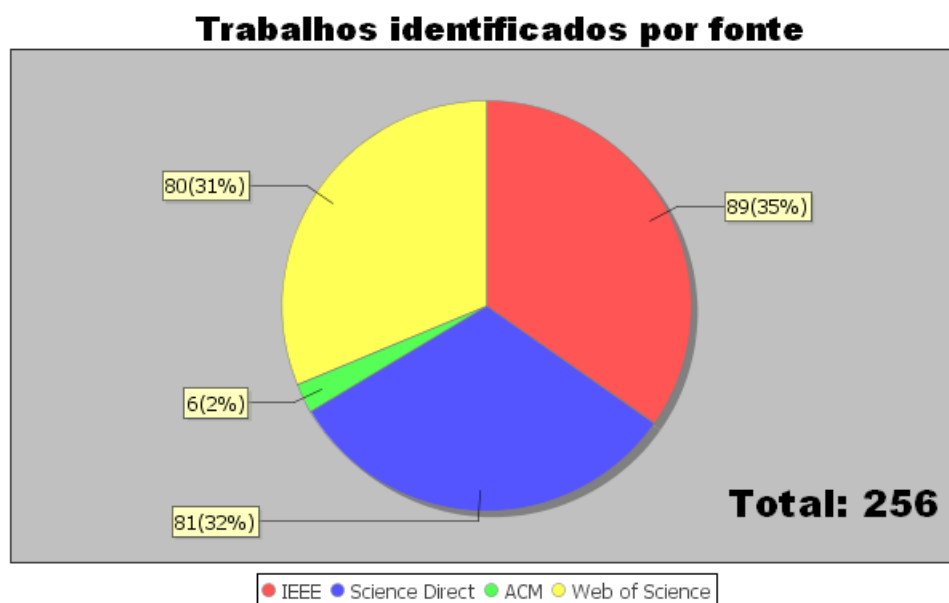


Figura 12 – Trabalhos identificados por fonte de pesquisa



Figura 13 – Seleção dos trabalhos identificados

Desses 36 trabalhos selecionados, apenas 6 foram extraídos e aproveitados com base na leitura da introdução e conclusão dos trabalhos selecionados. Foram coletados os dados necessários para responder a questão de pesquisa estabelecida no início do MS. Pode-se observar os trabalhos extraídos e as fontes de pesquisa dos trabalhos na Figura 15.

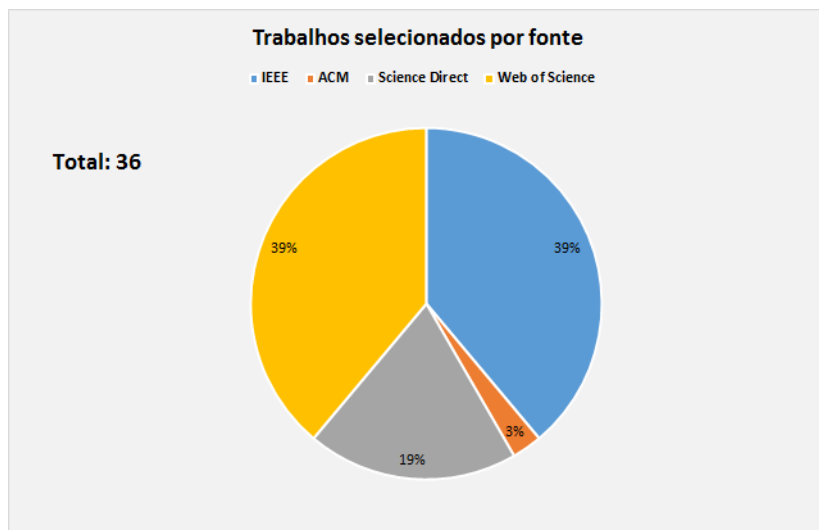


Figura 14 – Trabalhos selecionados por fonte de pesquisa

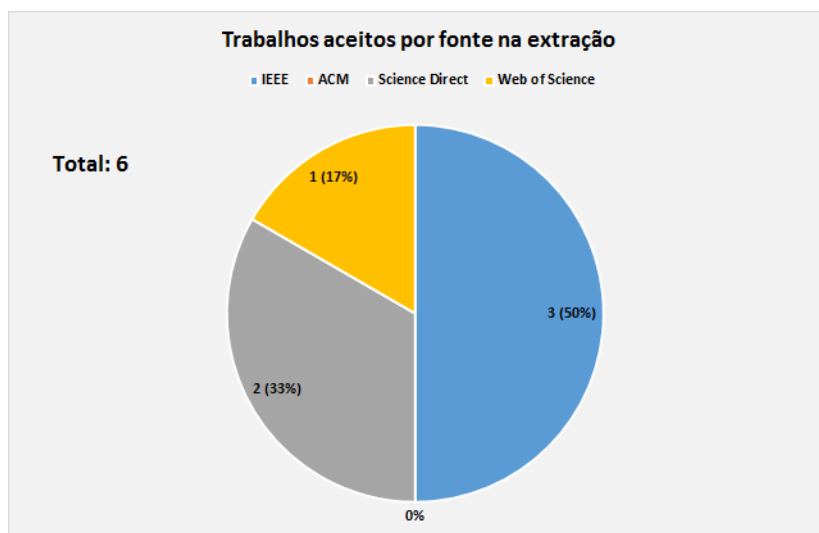


Figura 15 – Trabalhos aceitos por fonte de pesquisa

Os trabalhos foram aceitos ou excluídos de acordo com os critérios de inclusão e exclusão definidos no protocolo de MS (Apêndice A). Os trabalhos aproveitados foram aqueles que atenderam com todos os critérios de inclusão definidos no protocolo de MS no Apêndice A, como é apresentado na Tabela 3.

Tabela 3 – Tabela critérios de inclusão na fase de extração de artigos

<b>Crítério de inclusão</b>	<b>Artigos aceitos</b>
O trabalho deve ser em inglês	5
O trabalho deve estar disponível na web	5
O trabalho deve possuir menção à características de qualidade de processo ou possuir menção à qualidade ou taxonomias ou propriedades sobre modelagem de um processo	6

Pode-se observar que dos seis artigos aceitos o critério de inclusão com maior frequência foi: “O trabalho deve possuir menção à características de qualidade de processo ou possuir menção à qualidade da modelagem ou taxonomias ou atributos ou propriedades sobre a modelagem de um processo ”

Dos 29 trabalhos rejeitados, o critério em comum na maioria dos trabalhos: "O trabalho não faz menção a característica de qualidade de modelagem ou qualidade de processo", assim como é apresentado na Tabela 4.

Tabela 4 – Tabela criterios de exclusão na fase de extração de artigos

<b>Critério de inclusão</b>	<b>Artigos rejeitados</b>
Os estudos que não relataram resultados empíricos ou literatura que estava disponível apenas na forma de resumos ou apresentações PowerPoint	1
O estudo está fora do domínio de computação ou engenharia de software	0
O trabalho não faz menção a modelagem ou a qualidade de modelagem	16
O trabalho não faz menção às características de qualidade de modelagem ou qualidade de processo	25

Após a execução do MS foi realizado a sumarização dos dados e construídos indicadores do mapeamento realizado. A Tabela 5 apresenta os valores dos indicadores. Na tabela é apresentado o nome do indicador, e o valor obtido do indicador.

Tabela 5 – Indicadores do mapeamento sistemático

Nome do Indicador	Valor
Número de trabalhos identificados	256
Número de trabalhos identificados por fonte: IEEE	89
Número de trabalhos identificados por fonte: ACM	6
Número de trabalhos identificados por fonte: Science Direct	81
Número de trabalhos identificados por fonte: WOK	80
Número de trabalhos excluídos na seleção	203
Número de trabalhos duplicados na seleção	17
Número de trabalhos selecionados	36
Número de trabalhos selecionados por fonte: IEEE	14
Número de trabalhos selecionados por fonte: ACM	1
Número de trabalhos selecionados por fonte: Science Direct	7
Número de trabalhos selecionados por fonte: WOK	14
Número de trabalho excluídos na extração	29
Número de trabalhos aceitos na extração	6
Número de trabalhos duplicados na extração	1
Número de trabalho aceitos por fonte: IEEE	3
Número de trabalho aceitos fonte: ACM	0
Número de trabalho aceitospor fonte: Science Direct	2
Número de trabalho aceitos por fonte: WOK	1
Número de trabalhos aceitos por critério de inclusão 1	5
Número de trabalhos aceitos por critério de inclusão 2	5
Número de trabalhos aceitos por critério de inclusão 3	6
Número de trabalho excluídos por critério de exclusão 1	25
Número de trabalho excluídos por critério de exclusão 2	16
Número de trabalho excluídos por critério de exclusão 3	1
Número de características de qualidade de processo identificadas	105

Como pode-se observar na Tabela 5, foram identificados 105 características de qualidade de processo. Essas características foram obtidas através da coleta de dados realizada no MS, e foram agrupadas e categorizadas como será apresentado na Seção 5.1. As características identificadas podem ser usadas para a definição de métricas para avaliar a qualidade da modelagem de um processo.

Os trabalhos selecionados a partir do MS foram:

1. Kitchenham, B. and Pickard, L. and Linkman, S. and Jone, P., **A framework for evaluating a software bidding model**, 2005.
2. Kroeger, T. A. and Davidson, N. J. and Cook, S. C., **Understanding the characteristics of quality for software engineering processes: A Grounded Theory investigation**, 2013.
3. Borgoñón, L. G. and García, J. A. G. and Barcelona, M. A. and Escalona, M.J., **Software Process Accessibility in practice: a case study**, 2013.



4. Lonchamp, J. and Benali, K. and Godart, C and Derniame, J.C, **Modeling and Enacting Software Processes : an Analysis**, 1990.
5. Kroeger, T. A. and Davidson, **A Perspective-based Model of Quality for Software Engineering Processes**, 2009.
6. Guceglioglu, A. S. and Demirors, O., **The Application of a New Process Quality Measurement Model for Software Process Improvement Initiatives**, 2011.

As características identificadas foram selecionadas a partir destes trabalhos. Entretanto, a referência de número (4) não foi utilizada na extração de dados, pois, após uma análise do artigo, percebeu-se que as características que eram apresentadas não estavam relacionadas a qualidade da modelagem de processo de software.

#### 4.1.1 Métricas de Qualidade

Além de identificar características de qualidade de processo, foi possível identificar algumas métricas definidas pelos autores dos trabalhos selecionado no MS. As métricas encontradas foram definidas por Guceglioglu e Demirors (2011) e são apresentadas na Tabela 6.

Tabela 6 – Métricas propostas por Guceglioglu and Demirors

<b>Qualidade de processo</b>		
<b>Características</b>	<b>Subcaracterísticas</b>	<b>Métricas</b>
Evolubilidade	Analisabilidade	Complexidade
		Acoplamento
Confiabilidade	Tolerância a falhas	Prevenção à falhas
	Recuperabilidade	Restauração
		Eficácia da Restauração
Funcionalidade	Adequabilidade	Adequação Funcional
		Compleitude Funcional
	Funcionalidade Base de TI	Uso da TI
		Densidade de TI
	Acurácia	Acurácia Computacional
	Interoperabilidade	Permutabilidade de dados
Segurança	Auditabilidade de Acesso	
Usabilidade	Compreensibilidade	Compreensibilidade Funcional
	Apreensibilidade	Existência em documentos
	Operabilidade	Verificação de Validade de Entradas
		Anulabilidade
Atratividade	<i>Interação Atrativa</i>	

As métricas foram detalhadas no trabalho na qual ela foi mencionada.

## 4.2 Resultados da validação do modelo de características

Essa seção apresenta os resultados da validação do modelo de características de qualidade de modelagem de processo de software. São apresentados: a caracterização da amostra; a análise dos resultados do questionário; e a classificação das características de qualidade de processo por grau de importância.

### 4.2.1 Caracterização da População Respondente

Com o intuito de caracterizar a população respondente do questionário, foram elaboradas algumas questões relacionadas a: sexo, localização, tempo de atuação em mps, tempo de experiência com modelagem de processos, modelos de qualidade trabalhados, ferramentas ou linguagens de modelagem de processo utilizado, área de atuação com mps ou com modelagem de processo

Em relação ao gênero, dos 20 respondentes 8 são mulheres e 12 são homens. Já em relação a localização, de todos os respondentes 30% foram do estado de São Paulo seguido de 25% do Distrito Federal. A Figura 16 detalha a distribuição geográfica dos respondentes.

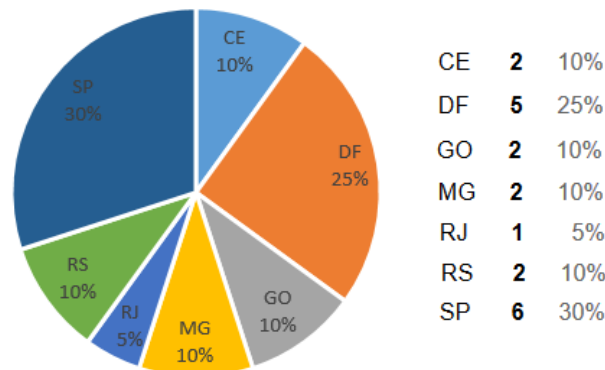


Figura 16 – Localização dos respondentes por UF

Conforme pode ser visto na Figura 17, em relação ao tempo de atuação com MPS, 30% dos respondentes tem mais de 6 anos de experiência com MPS, seguido por 20% que tem até um ano de experiência.

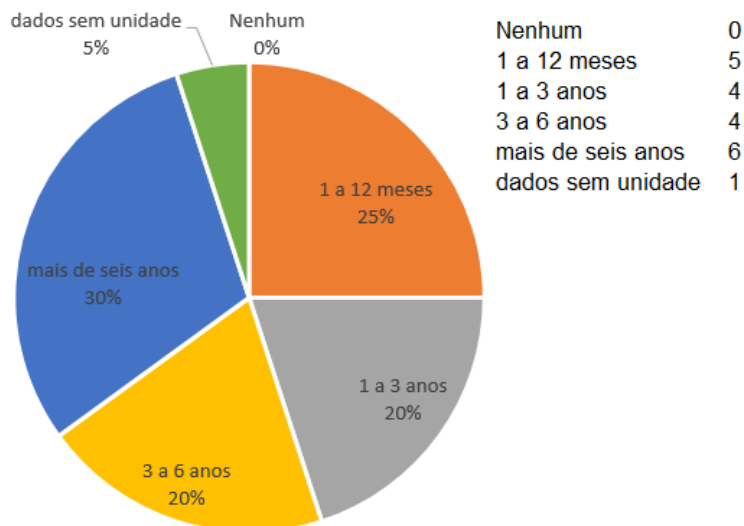


Figura 17 – Tempo de atuação dos respondentes com MPS

Já sobre o tempo de experiência com modelagem de processo 35% dos respondentes tinham entre 3 e 6 anos de experiência, conforme pode ser visto na Figura 18.

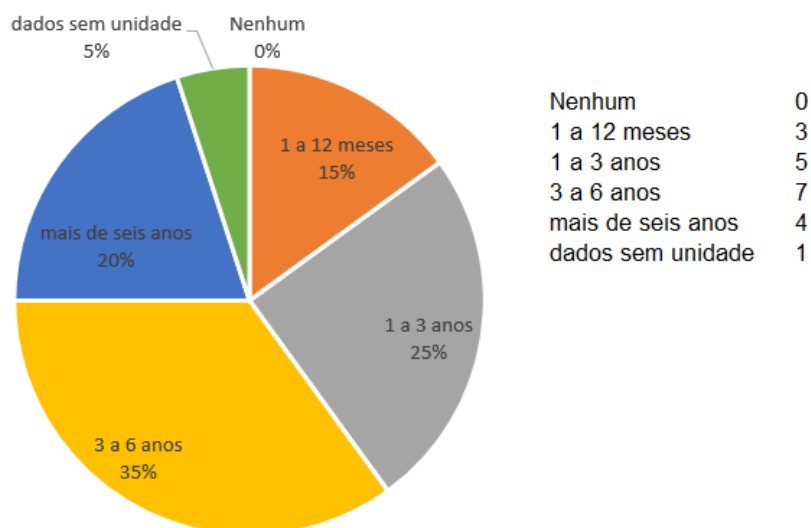


Figura 18 – Tempo de experiência com modelagem de processo

Em relação ao modelo de qualidade de processo, o mais utilizado foi o MPS.BR com 60% dos respondentes, seguido pelo CMMI com 10%. As ferramentas ou linguagens de modelagem de processo de software mais utilizadas pelos respondentes foi o Bizagi, com 45%, seguido pelo diagrama de atividades UML, com 30%, e outros 10%.

Em relação a área de atuação de MPS ou modelagem de processos a maioria dos respondentes atuaram na iniciativa privada, com 60% dos respondentes; 25% são do setor público; e 10% da academia.

Os respondentes puderam também deixar suas considerações, sendo elas:

- Corrigir a questão sobre adequabilidade, na qual está escrito “Adequabilidade pode ser classificada em”, o correto seria “é classificada em”. Como se está propondo um modelo o termo “pode” não deveria ser utilizado, já que é uma definição e não uma possibilidade.
- Em relação a questão sobre atratividade, a capacidade do projeto pode ser esteticamente agradável e visualmente envolvente, mas essas impressões não são muito importantes, pois o processo pode ser agradável, mas se não for útil não servirá para ser aplicado na organização.
- Adicionar a característica consistência junto com sua subcaracterística em usabilidade. Adicioná-las, talvez, na subcaracterística apreensibilidade, haja vista que o conceito de apreensibilidade está relacionado à utilizar elementos para facilitar o aprendizado do usuário, já representação está relacionada à uma questão linguística e de elementos padrões que devem estar disponíveis a todos os interessados.

#### 4.2.2 Análise dos resultados do questionário

Analisando as respostas das características, de um modo geral, as características avaliadas tiveram um grau de importância tendendo para a direita, que são mediantemente, muito e extremamente importante. Isso demonstra que as características estão em concordância com a literatura levantada.

Para analisar os dados, foram verificadas aquelas características que não se pode obter uma opinião clara dos respondentes. Para verificar isso foi observado gráficos, apresentados no apêndice E, que foram gerados a partir dos resultados do questionário. Foi observado um histograma das respostas do questionário. O histograma pode ser usado para dar uma visão geral da densidade de distribuição da amostras de uma variável. Um histograma é constituído por barras com alturas que representam a frequência (ou a frequência relativa) de um valor ou de um intervalo de valores, assim o histograma é uma representação gráfica de uma frequência (Wohlin et al., 2012).

Foi observado a curva da função de densidade da distribuição normal para cada item, essa curva apresenta o comportamento das respostas de um item (Morettin and Bussab, 2009), assim como mostra a Figura 19.

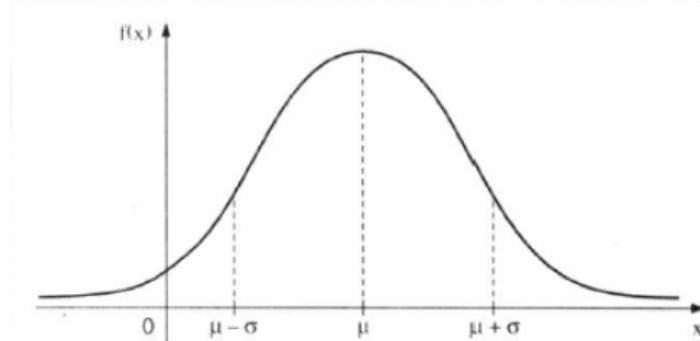


Figura 19 – Curva da distribuição normal (Morettin and Bussab, 2009)

Para verificar a distribuição normal de cada item foi utilizado o cálculo de distribuição normal mostrado na fórmula abaixo, esse cálculo foi realizado utilizando a ferramenta IBM SPSS Statistics:

$$f(X) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

Os itens que possuem uma curvatura mais para a esquerda, mais próximo do eixo y, indicando um grau de resposta como "nada importante" ou "pouco importante", não devem ser considerado para análise pois apresenta um baixo grau de importância para a comunidade. Já itens em que sua curvatura de densidade de distribuição normal tende para a direita, mais afastado do eixo y, indica que as características tiveram seu conceito considerados como "muito importante" ou "extremamente importante" e não houve nenhuma inconsistência.

Os itens selecionados para um análise mais detalhada foram os itens 4, 22, 23 e 25. Esses itens foram escolhidos pois apresentaram um conceito tendendo para a opção "mediamente importante" equivalente com "muito importante", ou aqueles itens em que o ápice da curvatura do histograma na escala "mediamente importante". Também foram selecionados aqueles itens que possuíam a média da distribuição normal mais próximo de três e seu desvio padrão mais próximo de um. Esse itens não apresentam com clareza a opinião do respondente. Com o intuito de entender o comportamento desses itens foram analisados as extremidades do histograma, ou seja, foram analisados as respostas dadas como "nada importante", "pouco importante", "muito importante" e "extremamente importante".

A análise de cada característica selecionada foi baseada na caracterização da população respondente, onde foi levado em consideração: a quantidade de respondentes para os

intervalos da escala; a quantidade de respondentes dividida pelo contexto de onde ocorreu a experiência como MPS ou modelagem de processo.

#### 4.2.2.1 Item 4 - Sintática

O item 4, selecionado para análise corresponde a característica sintática. A curvatura normal das respostas é mostrada na Figura 20.

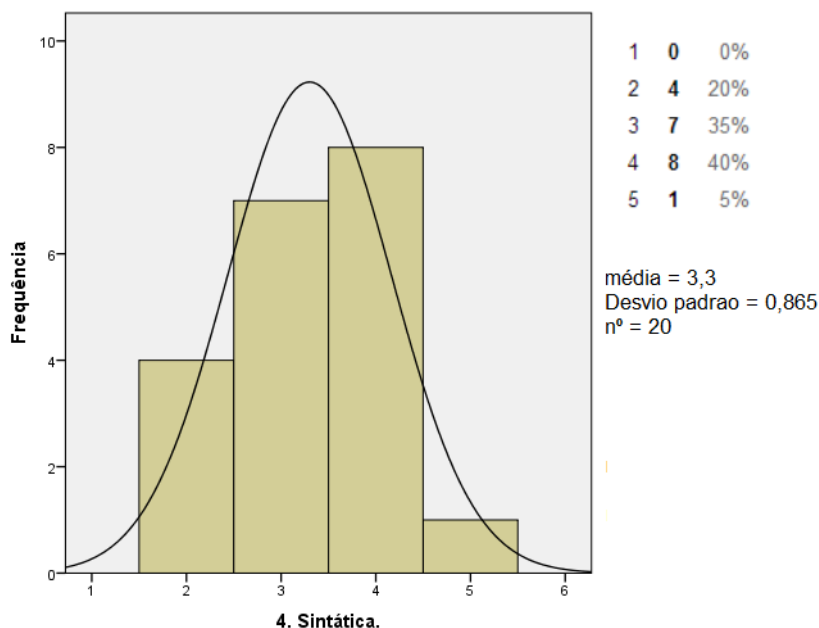


Figura 20 – Item 4 - sintática

A média da curvatura desse item foi de 3,3, sendo que 35% respondentes consideraram ela “mediamente importante”, 40% consideraram ela como “muito importante”, e outros 20% a consideraram “pouco importante”.

Levando em consideração o número de respondentes que responderam como “nada importante” e “pouco importante”, totalizaram-se quatro respondentes. Já em relação aos que responderam como “muito importante” e “extremamente importante”, totalizaram-se nove respondentes. Levando em consideração que o número de respostas como “muito importante” e “extremamente importante” foi maior, pode-se concluir que o item possui o maior número de respostas à direita da escala, expressando a importância da característica. Sendo assim, a característica sintática foi mantida no modelo de características de qualidade de processo.

#### 4.2.2.2 Item 22 - Tamanho

O item 22 se refere a característica tamanho. A curvatura das respostas deste item é apresentado na Figura 21, sendo que a média da curvatura de distribuição normal foi de 3,65.

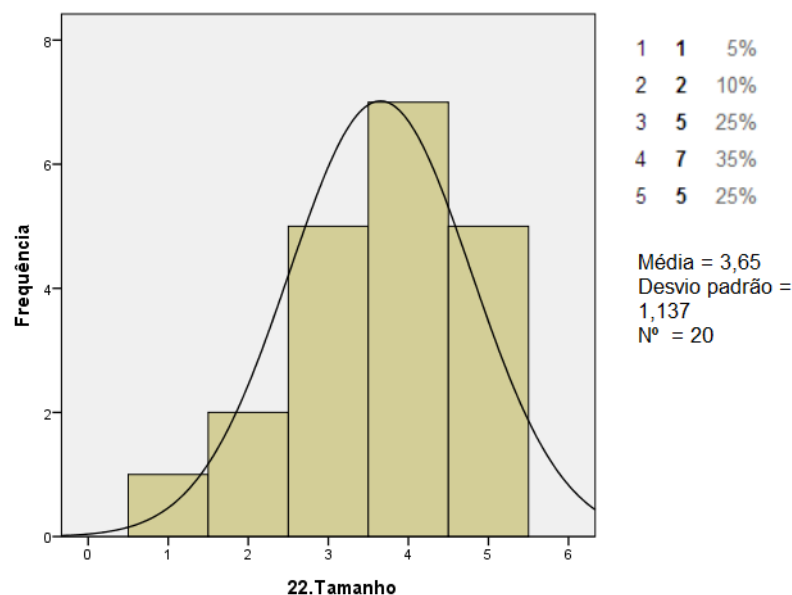


Figura 21 – Item 22 - Tamanho

35% dos respondentes classificaram o item como “muito importante”, seguido de 25% dos que responderam como “extremamente importante” e também como “mediamente importante”. Analisando as extremidades da escala no gráfico, percebe-se que, em relação a “nada importante” e “pouco importante” tiveram três respostas. Já em relação a “muito importante” e “extremamente importante” tiveram doze respostas, correspondendo a 60% das respostas. Sendo assim, pode-se considerar que a característica é importante e será mantida no modelo de características.

#### 4.2.2.3 Item 23 - Complexidade

O item 23 selecionado para análise corresponde a característica de "complexidade", que é uma subcaracterística de arquitetura. A curvatura da sua resposta, a qual teve média de 3,8, é mostrada na Figura 22.

A curvatura dessa característica indica que sete (35%) dos respondentes consideraram a característica "mediamente importante", e outros seis (30%) consideraram a característica "extremamente importante". Além disso um (5%) considerou essa característica "pouco importante".

Os que responderam como "nada importante" e "pouco importante", totalizaram um respondente, sendo este atuou na iniciativa privada.

Levando em consideração que o número de respostas para "muito importante" e "extremamente importante", onze respostas, é maior que o número de respostas "nada importante" e "pouco importante", uma resposta, apesar de possuir um grande número de respostas como "mediamente importante". Sendo assim, essa característica possui o maior

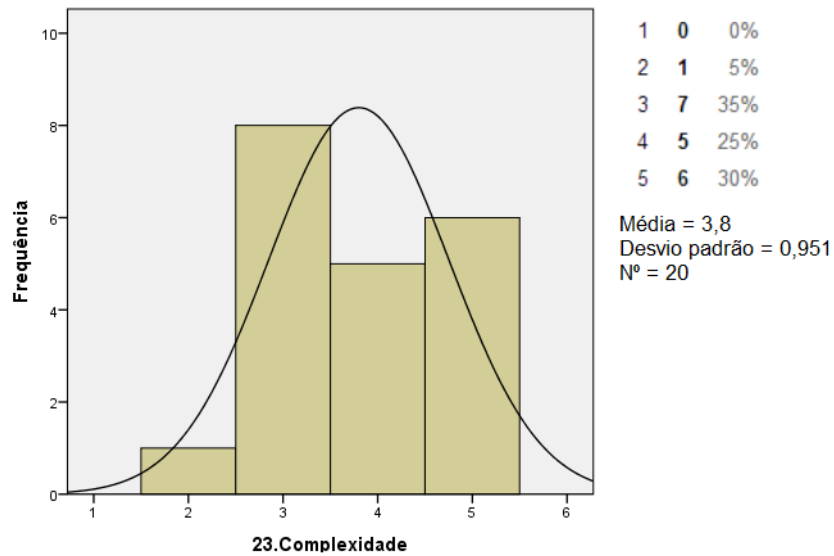


Figura 22 – Item 23 - complexidade

número de respostas na escala à direita, expressando a importância da característica, mantendo essa característica no modelo.

#### 4.2.2.4 Item 25 - Escalabilidade

O item 25 é referente a subcaracterística de escalabilidade. A média da curvatura de distribuição normal foi de 3,55, como pode ser visto na Figura 23.

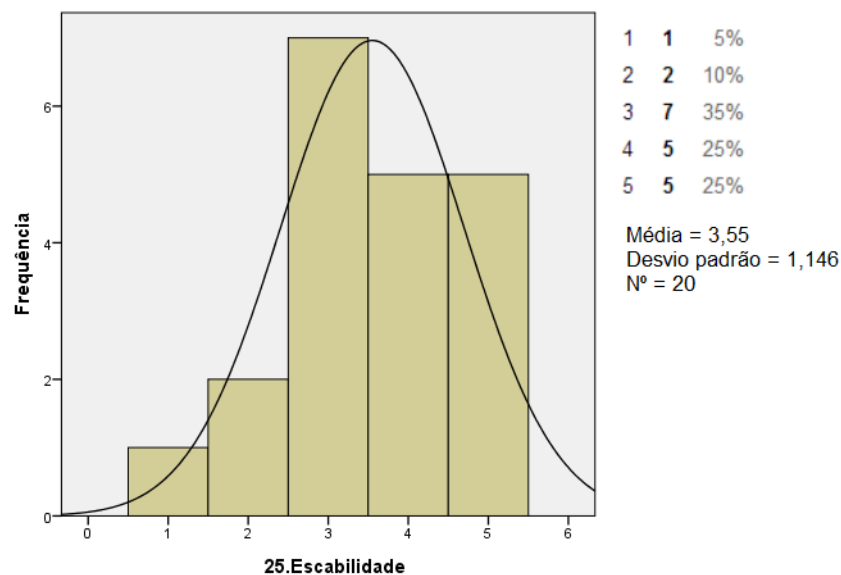


Figura 23 – Item 25 - escalabilidade

A curvatura dessa característica indica que sete (35%) dos respondentes consideraram a característica "mediamente importante", e outros cinco (25%) consideraram a



característica "extremamente importante". Além disso um (5%) considerou essa característica como "nada importante".

Levando em consideração que o número de respostas para "muito importante" e "extremamente importante", dez respostas, é maior que o número de respostas "nada importante" e "pouco importante", três respostas, apesar de possuir um grande número de respostas como "mediamente importante". Sendo assim, essa característica possui o maior número de respostas na escala à direita, mostrando a importância da característica.

### 4.2.3 Importância das características

Para propor as características que tem o maior grau de importância do ponto de vista do respondente foram listadas todas as características ordenadas por grau de importância. O ordenamento da importância foi feito da seguinte maneira: para cada item somou-se a quantidade de respostas absolutas referentes aos valores das escalas 4 e 5, no caso de empate nos valores aquelas que tiveram o maior número de respostas referente a Escala 5 tem preferência. Sendo que esses valores indicam que o item é importante.

Tabela 7 – Características por grau de importância

Nº	Característica	Resultado	
		Absoluto	Percentual
1º	Funcionalidade	19	95%
2º	Usabilidade	18	90%
3º	Evolubilidade	15	75%
4º	Arquitetura	14	70%
5º	Representação	13	65%
6º	Sintática	9	45%

As características apresentaram um grau de importância alto, com exceção da característica sintática que apresentou um grau de importância médio de 45%. Entretanto, considerando que todas as subcaracterísticas associadas a ela obtiveram índice maior que 60% (veja Tabela 8), esta característica foi mantida. Dessa maneira, todas as características foram consideradas importantes para se avaliar um processo de software e foram mantidas no modelo de características de processo de software.

A importância das subcaracterísticas foram agrupadas de acordo com suas características. Assim como é apresentado na Tabela 8. E da mesma forma que as características, todas as subcaracterísticas foram consideradas importantes para o modelo, e assim, foram mantidas no modelo de características de processo.

Tabela 8 – Subcaracterísticas por grau de importância

Característica			Resultado	
			Absoluto	Percentual
Funcionalidade	1º	Adequabilidade	16	80%
	2º	Interoperabilidade	15	75%
	3º	Segurança	13	65%
Usabilidade	1º	Compreensibilidade	20	100%
	2º	Adaptabilidade	19	95%
	3º	Apreensibilidade	18	90%
	4º	Acessibilidade	17	85%
	5º	Execução	15	75%
	6º	Atratividade	14	70%
	7º	Operabilidade	14	70%
	8º	Semântica	12	60%
Evolubilidade	1º	Modificabilidade	17	85%
	2º	Apoio	15	75%
	3º	Analisabilidade	14	70%
Arquitetura	1º	Modularidade	13	65%
	2º	Tamanho	12	60%
	3º	Complexidade	11	55%
	4º	Escalabilidade	10	50%
Representação	1º	Consistência	15	75%
Sintática	1º	Corretude	14	70%
	2º	Adequação ao formato	12	60%

## 5 Características de qualidade de processos

Este capítulo tem o objetivo de apresentar o modelo de características de qualidade de processo elaborado neste trabalho e apresentar o uso desse modelo. Este capítulo está dividido em duas seções: a Seção 5.1 apresenta o modelo de características de qualidade de processo, e a Seção 5.2 apresenta as formas de uso do modelo de características de qualidade.

### 5.1 Modelo de características de qualidade de processo

Para classificar as características encontradas, foi levado em consideração a estrutura de modelo de qualidade definida na ISO/IEC 25010 (ISO, 2011). Essa norma categoriza a qualidade em características e subcaracterísticas de qualidade para uma melhor análise do produto.

As características e subcaracterísticas encontradas durante o MS estão listadas no Apêndice B. Esse resultado está relacionada à questão de pesquisa do MS: “Quais são as características de qualidade de uma modelagem de processo de software?”. Dessa forma, foi possível identificar atributos que caracterizam a qualidade de um processo de software e validar essas características através de um *survey*. A Figura 24 apresenta o processo de como ocorreu a seleção das características finais do modelo apresentado neste trabalho.

Após a condução do MS foram identificadas 105 características e realizado um refinamento dessas características. Em um primeiro momento, das 105 características foram selecionadas apenas aquelas que eram relacionadas a qualidade na modelagem de processo. Após essa seleção, características que possuíam o significado semelhantes a outras foram agrupadas, e características que eram relacionadas a outras foram transformadas em subcaracterísticas. Apesar desse refinamento, a maioria das características foram aproveitadas da maneira em que foram identificadas nos trabalhos selecionados no mapeamento sistemático. Após esse refinamento foram aproveitadas 63 características.

Dessas 63 características foram selecionadas 48 características a partir de uma validação semântica e sintática do modelo. Essas validações se deram por meio da aplicação de um instrumento de pesquisa *survey*. Na validação semântica foram selecionadas aquelas características que apresentavam uma definição clara, e também, foram reagrupadas aquelas características que estavam relacionadas a alguma outra característica. Já na validação sintática foram selecionadas aquelas características que mostram ter maior importância do ponto de vista do respondente. De acordo com o ponto de vista do respondente, o modelo de características de qualidade processo apresentado é válido e todas

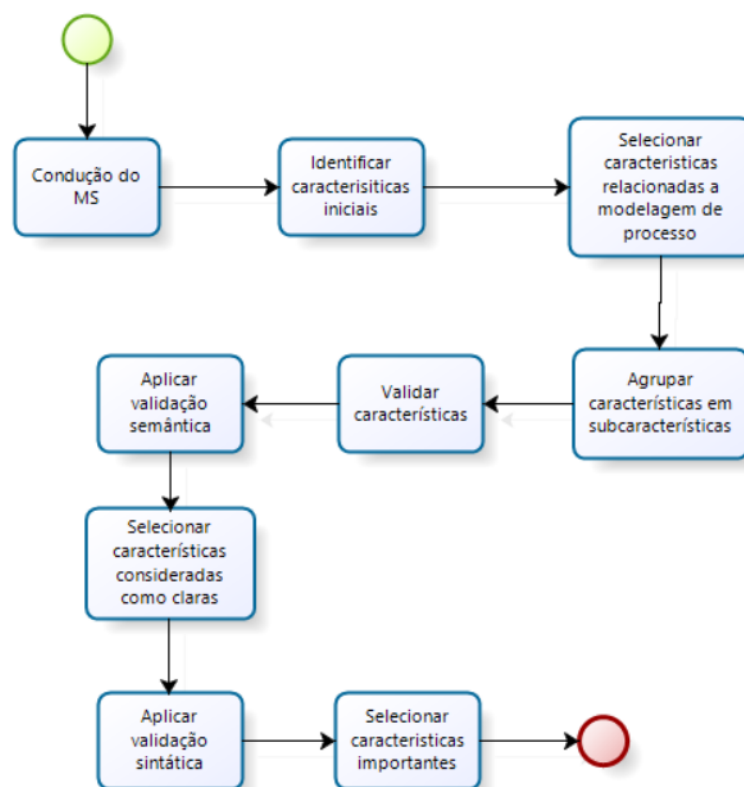


Figura 24 – Processo de obtenção das características

as características foram consideradas importantes.

Baseado na Norma SQuaRe (ISO, 2011), foi elaborado um modelo representativo das características de qualidade, apresentado na Figura 25. Cada item presente na figura é melhor detalhada a seguir.

### 5.1.1 Usabilidade

Usabilidade é definida como a facilidade de interpretar e executar uma modelagem de processo, sob condições específicas, para alcançar os resultados desejados (Kroeger and Davidson, 2009; Guceglioglu and Demirors, 2011). A usabilidade pode ser classificada como: acessibilidade, compreensibilidade, apreensibilidade, adaptabilidade, operabilidade, atratividade, semântica e execução

#### 5.1.1.1 Acessibilidade

Acessibilidade é definida como a facilidade de encontrar informações sobre um processo de engenharia de software. (Kroeger and Davidson, 2009). De acordo com Garcia-Borgoñón (2013), acessibilidade também pode ser definida como a capacidade com a qual usuários com deficiência podem: perceber e compreender os processos de software; executar os processos definidos e; projetar e desenvolver novos processos. Neste trabalho serão consideradas as três definições:

### 1. **Texto Alternativo**

Trata da capacidade de transformar qualquer conteúdo não textual em outras formas, de acordo com a necessidade do usuário, tais como letras grandes, braile, fala, símbolos ou linguagens mais simples (Borgoñón et al., 2013).

### 2. **Navegabilidade**

Navegabilidade é a capacidade do usuário navegar, localizar conteúdos e determinar onde eles estão em um processo de software (Borgoñón et al., 2013).

### 3. **Acessibilidade da sequência do fluxo de trabalho**

Trata da capacidade da sequência de leitura do fluxo de trabalho ser apresentada de forma correta, pois a sequência em que o conteúdo é apresentado afeta o seu significado (W3C, 2014).

#### 5.1.1.2 **Compreensibilidade**

Compreensibilidade é definida como a facilidade com a qual o usuário de um processo é capaz de compreender se um processo de software é relevante e como ele pode ser utilizado para alcançar os resultados desejados (Kroeger and Davidson, 2009).

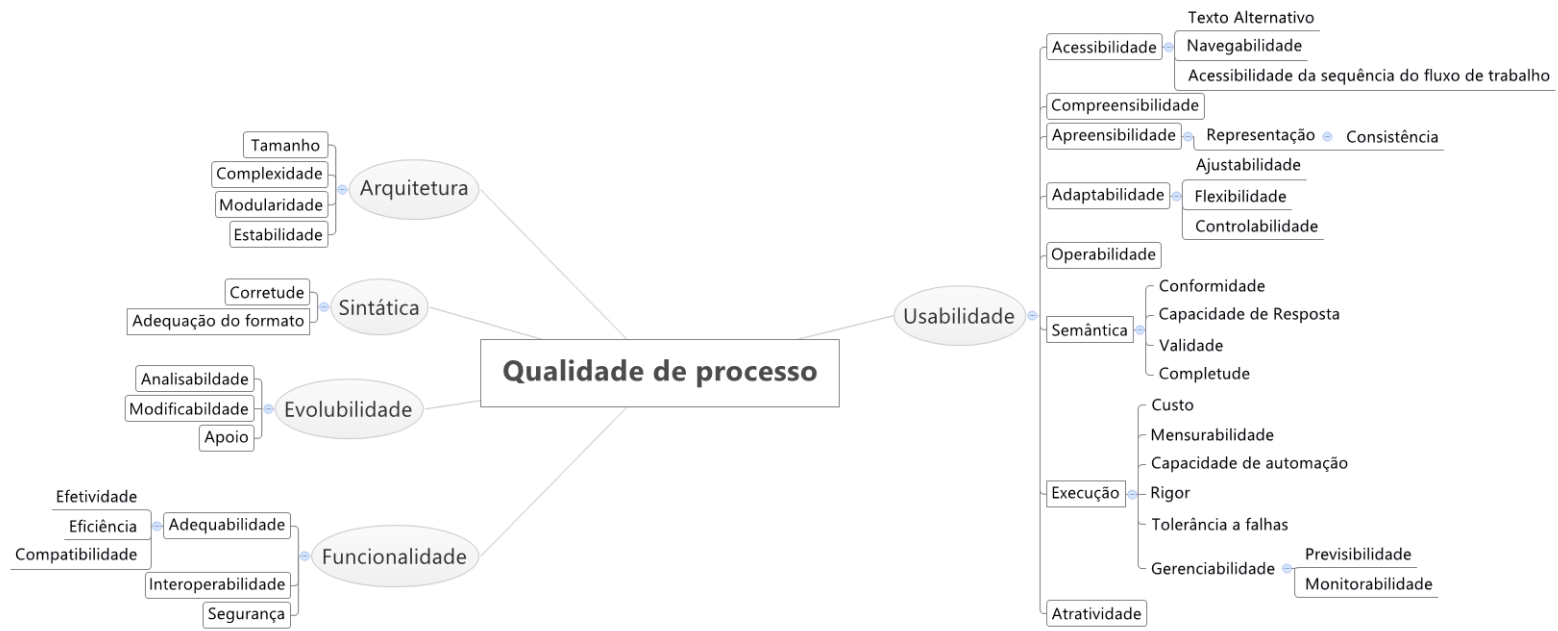


Figura 25 – Modelo das características de qualidade de processo de software

### 5.1.1.3 Apreensibilidade

Apreensibilidade é definida como a capacidade apresentar todos os seus elementos (atividades, fluxogramas, papéis, etc) de forma a facilitar o entendimento do usuário, para que esse tenha a capacidade de executar uma atividade do processo (Kroeger and Davidson, 2009). Apreensibilidade é diferente de compreensibilidade, pois um processo pode ser fácil de entender mas difícil de aprender. Essa situação pode surgir, por exemplo, se o processo envolver atividades que, embora, sejam conceitualmente simples, requerem muita prática para se aprender a executá-la. Apreensibilidade pode ser classificada como Representação.

**Representação** É definida como a capacidade de um processo ser documentado e comunicado à todas as partes interessadas (Kroeger et al., 2013). Representação pode ser classificado como consistência.

**Consistência** Consistência é definida como a medida em que um processo de software está descrito de forma que seja coerente, em termos de representação, como o próprio processo e com os outros processos aos quais ele está relacionado (Kroeger et al., 2013).

### 5.1.1.4 Adaptabilidade

Adaptabilidade é a capacidade do processo se adaptar em diferentes situações sem perder suas informações ou sua estrutura, facilitando utilização do processo pelo usuário (Kroeger and Davidson, 2009; Borgoñón et al., 2013). Adaptabilidade é extremamente importante, por exemplo, quando uma organização pretende criar um conjunto padrão de processos para uso em múltiplos projetos. A adaptabilidade pode ser classificada em três subclassificações:

#### 1. Ajustabilidade

Trata da facilidade de um processo padrão ser adaptado para formar processos específicos em determinado contexto, de diferentes tamanhos (Kroeger and Davidson, 2009).

#### 2. Flexibilidade

Trata da possibilidade de modificação, em tempo de execução, de algum aspecto do processo como o objetivo de melhorar o desempenho dele. O processo deve ser flexível, mas uma alta taxa de flexibilidade pode também inutilizar a documentação do processo provida. (Kroeger and Davidson, 2009).

#### 3. Controlabilidade

Controlabilidade é definida como a facilidade com que gestor de processos é capaz de aplicar ações preventivas e corretivas para um processo e continuar mantendo um nível de desempenho especificado (Kroeger et al., 2013)

#### 5.1.1.5 Operabilidade

Operabilidade é a capacidade do processo ser utilizado e controlado (Guceglioglu and Demirors, 2005).

#### 5.1.1.6 Atratividade

Atratividade é a capacidade do processo de envolver os usuários com os componentes do processo, por exemplo, templates ou representação gráfica do processo (Guceglioglu and Demirors, 2005).

#### 5.1.1.7 Semântica

Semântica é definida como a capacidade de relacionar o modelo com o domínio, considerando as relações entre as instruções e seus significados, ou seja, relacionam o escopo do modelo com a intenção de suas atividade (Kroeger et al., 2013; Kitchenham et al., 2005). Semântica pode ser classificada em quatro subcaracterísticas: conformidade, capacidade de resposta, especificidade, validade e completude.

1. **Conformidade** É definida como a capacidade com a qual o processo de software está de acordo com padrões e modelos de processos reconhecidos na indústria (Kroeger et al., 2013).
2. **Capacidade de Resposta** É definida como a velocidade de resposta com que um processo de software é capaz de incorporar um *feedback* nos envolvidos com o processo (Kroeger et al., 2013)
3. **Validade** É relacionada a adequação do processo ao domínio e às características de time para o qual foi documenta (Kitchenham et al., 2005).
4. **Completude** Capacidade do processo de software ou modelo conter todas as declarações sobre o domínio, sendo todas elas corretas e relevantes (Kitchenham, Pickard, Linkman, and Jone, 2005).

#### 5.1.1.8 Execução

Execução é definida como o uso do processo de software dentro de um ambiente específico (Kroeger et al., 2013). Execução pode ser classificada em seis subcaracterísticas: custo, mensurabilidade, capacidade de automação, rigor, tolerância a falhas e gerenciabilidade.



### 1. Custo

O custo é definido como a quantidade de esforço, tempo e dinheiro necessários para executar um processo de software dentro de um contexto específico (Kroeger et al., 2013).

### 2. Mensurabilidade

Mensurabilidade é definida como a capacidade da execução de um processo de software oferecer medidas relacionadas com seu desempenho (Kroeger et al., 2013).

### 3. Capacidade de automação

Definida como a capacidade de um processo de software ser executado usando recursos computacionais nas atividades do processo (Kroeger et al., 2013).

### 4. Rigor

Definido como a medida que um processo de software requer para que suas atividades sejam realizadas de maneira controlada e disciplinada (Kroeger et al., 2013).

### 5. Gerenciabilidade

Gerenciabilidade é definida como a facilidade de gerenciar as mudanças em um processo, ou seja, a facilidade com que uma mudança em um processo pode ser prevista, monitorada e controlada para alcançar os resultados desejados, em um contexto específico. A gerenciabilidade está preocupada com ter ou não um processo que possa ser planejado e suas atividades executadas de acordo com o planejado. (Kroeger, Davidson, and Cook, 2013; Sorumgard and Sindre, 1995). Gerenciabilidade pode ser classificada como:

- **Previsibilidade** É definida como a facilidade com que um gestor de processo é capaz de determinar o tempo, recursos, riscos e limitações associadas à execução do processo. Essa característica é importante em qualquer situação em que compromissos precisam ser feitos antes do início dos trabalhos. Qualquer desenvolvimento de um processo precisa ser planejado, esses planos são usados como base para alocação de recursos como tempo e pessoas. (Kroeger and Davidson, 2009).

- **Monitorabilidade**

Monitorabilidade é definida como a facilidade com que um gerente de processo é capaz de determinar o status e os resultados de um processo executado (Kroeger et al., 2013). Se um processo tem um nível baixo de monitorabilidade, os problemas que ocorrem ao longo da execução desse processo podem não aparecer até que o processo esteja completo, assim, poderia ser tarde demais para corrigir estes problemas.

## 6. Tolerância a falhas

Tolerância a falhas é definida como o grau em que um processo, uma vez que uma intrusão ocorreu, ou continua a produzir resultados precisos ou reconhece a imprecisão de seus resultados e inicia uma ação corretiva (Feiler and Humphrey, 1992).

### 5.1.2 Funcionalidade

Funcionalidade é a capacidade do processo em descrever tudo o que é realizado ou que deveria ser realizado pela equipe ou organização, ou seja as funções esperadas do processo. (Guceglioglu and Demirors, 2011). Essas funções são: adequação, segurança e interação com outros processos.

#### 5.1.2.1 Adequabilidade

Adequabilidade é definida como a capacidade do processo para produzir os resultados desejados de forma apropriada em um contexto específico, ou seja, a capacidade das atividades do processo de serem completas e adequadas para a execução de tarefas (Kroeger et al., 2013; Guceglioglu and Demirors, 2005). A adequabilidade é classificada em três subclassificações:

##### 1. Efetividade

É definida como a capacidade de um processo de software transformar um conjunto de entradas em um conjunto de saídas desejadas. Dessa maneira, entende-se que um processo eficaz deve ajudar a produzir um produto certo (Kroeger et al., 2013).

##### 2. Eficiência

É a capacidade de um processo de software conseguir resultados com o menor tempo e esforço possível (Kroeger, Davidson, and Cook, 2013; Sorumgard and Sindre, 1995).

##### 3. Compatibilidade

É definida como a capacidade do processo de software de interagir com um ou mais processos ou tecnologias especificadas (Kroeger et al., 2013).

#### 5.1.2.2 Interoperabilidade

Interoperabilidade é a capacidade de um processo interagir com outros processos, ou seja, a interoperabilidade indica o quão fácil é acoplar o processo em outros processos (Sorumgard and Sindre, 1995; Guceglioglu and Demirors, 2005).

### 5.1.2.3 Segurança

Segurança é definida como a capacidade do processo de software proteger as informações e dados, de forma que pessoas não-autorizadas não podem lê-los ou modificá-los, e que não seja negado acesso às pessoas autorizadas (Guceglioglu and Demirors, 2005).

## 5.1.3 Evolubilidade

Evolubilidade é definida como a facilidade com que um processo de software está disponível para ser avaliado e modificado para um propósito de manutenção e melhoria de processo (Kroeger et al., 2013). Evolubilidade pode ser classificado em três subcaracterísticas: analisabilidade, modificabilidade e apoio.

### 5.1.3.1 Analisabilidade

É definida como a facilidade de se diagnosticar deficiências e potenciais melhorias no processo (Kroeger et al., 2013; Guceglioglu and Demirors, 2005).

### 5.1.3.2 Modificabilidade

Modificabilidade é definida como a facilidade com que um processo de software permite que uma modificação seja implementada (Kroeger et al., 2013).

### 5.1.3.3 Apoio

É definida como a facilidade com que um processo de software apoia a realização das atividades necessárias para sua utilização, por exemplo: antes de um processo ser implantado, é importante que os recursos, conhecimento e tecnologias necessárias para que um processo seja realizado com sucesso estejam disponíveis (Kroeger et al., 2013).

## 5.1.4 Sintática

Sintática é definida como a relação do modelo com a linguagem de modelagem. Descreve as relações entre as construções de linguagem, sem considerar o seu significado (Kitchenham et al., 2005). Sintática pode ser classificada como corretude e adequação do formato.

### 5.1.4.1 Corretude

Corretude é definida como a capacidade do modelo de processo utilizar adequadamente a linguagem de modelagem na qual ele foi descrito (Kitchenham et al., 2005; Sorumgard and Sindre, 1995).

#### 5.1.4.2 Adequação do formato

Formato é definido como a notação usada para descrever um processo de software (Kroeger et al., 2013).

### 5.1.5 Arquitetura

É definida como os elementos de um processo de software que dizem respeito à sua organização. Influencia significativamente na forma como o processo é percebido (Kroeger et al., 2013). Arquitetura pode ser classificado em quatro subcaracterísticas: tamanho, complexidade, modularidade e escalabilidade.

#### 5.1.5.1 Tamanho

Definido como a magnitude de um processo de software em termos de número de atividades, papéis e artefatos. O tamanho de um processo não inclui apenas o modelo do processo de alto nível, mas também qualquer apoio a documentação (Kroeger et al., 2013).

#### 5.1.5.2 Complexidade

Definida em termos da quantidade de relações estabelecidas entre os itens necessários para descrever o processo, itens como, como a complexidade de um processo de engenharia de software, em termos do seu número de peças constituintes e fluxo de controle e interfaces (Kroeger et al., 2013).

#### 5.1.5.3 Modularidade

Modularidade é definida como a capacidade de um processo de software ser auto-contido, com interfaces externas bem definidas (Kroeger et al., 2013).

#### 5.1.5.4 Estabilidade

A estabilidade é definida como o período de tempo em que um processo de software tem sido utilizado num ambiente de informação, sem modificações significativas na semântica do processo (Kroeger et al., 2013).

As características obtidas por meio do mapeamento sistemático facilitam uma visualização e avaliação do nível de qualidade de uma modelagem de processo.

## 5.2 Uso do modelo de qualidade de processo

O modelo de características qualidade de processo de software pode ser utilizado de diversas formas, como:

- Auxílio na definição de processos;
- Guia para definição de métricas de avaliação de qualidade de documentação e de processo;
- Uso das características de qualidade do modelo para a verificação da documentação do processo no início da avaliação, para que se possa verificar quais características já estão presentes no processo da organização;
- Durante as fases de um ciclo de melhoria contínua;
- Avaliação de qualidade de processo assim como o modelo SQuaRE é usado para qualidade do produto;
- Auxiliar no contexto do nível 3 do CMMI (CMM, 2006) ou do nível D do MPS.BR (SOF, 2012), os quais tratam da definição de processos organizacionais;
- Avaliações tipo B do SCAMPI. O SCAMPI possui três tipos de avaliações: A, B e C. As avaliações tipo B, são avaliações não oficiais, intermediárias, cujo objetivo é verificar oportunidades de melhoria para ser submetida a uma avaliação oficial.

Deve-se ressaltar que a importância das características de qualidade irá depender do nível de metas e objetivos propostas para o projeto específico.

Dentre as diversas formas de uso foi escolhido duas formas de uso para se detalhar neste trabalho sendo elas, o modo de uso do modelo no contexto das normas SQuaRE e do IDEAL.

### 5.2.1 Uso no contexto da norma SQuaRE

Assim como na SQuaRE é possível usar o modelo de qualidade para a medição da qualidade do processo. As características de qualidade presentes no modelo podem ser, assim como no modelo apresentado no ISO/IEC 25010, utilizadas para se criar métricas para aferir qualidade. A SQuaRE apresenta uma forma para se utilizar as características de qualidade de produto, como é mostrado na Figura 26.

Como mostra a Figura 26, para realizar a medição é preciso utilizar um método de medição. Um método de medição é uma sequência lógica de operações usadas para quantificar as propriedades no que diz respeito a uma escala especificada. O resultado da aplicação de um método de medição é chamado um elemento de medidas de qualidade. As características de qualidade e subcaracterísticas podem ser quantificadas através da aplicação de funções de medição. A função de medição é um algoritmo usado para combinar elementos de medir a qualidade. O resultado da aplicação de uma função de medição é chamado de uma medida de qualidade de software. Desta forma, as medidas de qualidade

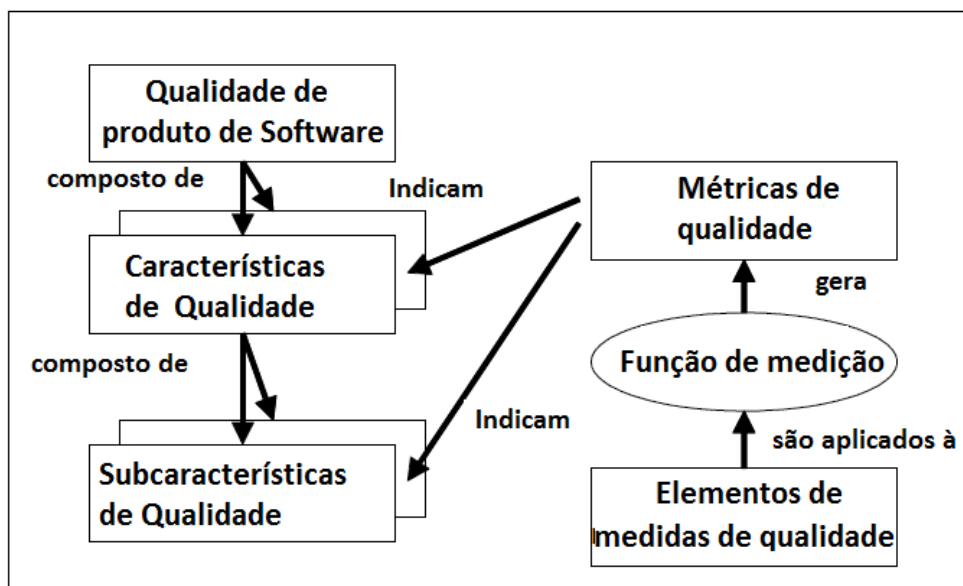


Figura 26 – Modelo de medição de qualidade de produto de software. Adaptado (ISO, 2011)

de software tornam-se quantificações das características de qualidade e subcaracterísticas (ISO, 2011).

Assim como na SQuaRE é possível aplicar esse modelo de medição de qualidade do produto para qualidade de modelagem do processo. Nesse modelo, ao invés de utilizar características de qualidade do produto, usa-se características de qualidade do processo. A medição de qualidade do software é utilizada para realizar a avaliação da qualidade do processo. Para realizar essa medição foi definido um processo de medição de modelagem de processo de software. Esse processo é composto pelas seguintes fases: definição de requisitos, definição de métricas, coleta de dados, análise de dados e relato dos resultados. Assim como é apresentado na Figura 27.

#### 5.2.1.1 Definição de requisitos

Na fase de definição de requisitos serão definidas, das características e subcaracterísticas apresentadas no modelo (Figura 25), quais delas são relevantes para o contexto em que o processo está inserido. Essa definição deve levar e conta as características da organização, da equipe e do processo que foi modelado. Esses requisitos podem incluir várias áreas do processo, ou especificamente, da modelagem. Por exemplo: requisitos para artefatos, processos, projetos, organização do processo de desenvolvimento. O modelo de qualidade de processo definido nesse trabalho provê cinco características:

1. **Usabilidade:** A facilidade de interpretar e executar uma modelagem do processo, sob condições específicas, para alcançar os resultados desejados.

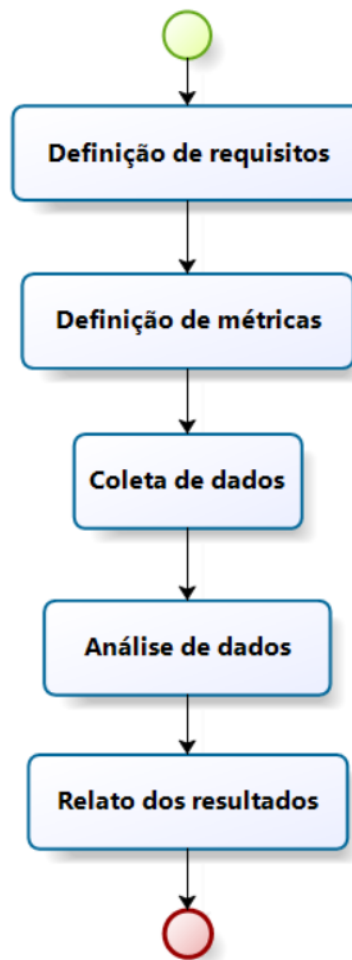


Figura 27 – Processo de medição de modelagem de processo de software

2. **Funcionalidade:** A capacidade do processo em descrever tudo o que é realizado ou que deveria ser realizado pela equipe ou organização, ou seja as funções esperadas do processo.
3. **Evolubilidade:** A facilidade com que um processo de software está disponível para ser avaliado e modificado para um propósito de manutenção e melhoria de processo.
4. **Sintática:** A relação do modelo com a linguagem de modelagem. Descreve as relações entre as construções de linguagem, sem considerar o seu significado.
5. **Arquitetura:** Os elementos de um processo de software que dizem respeito à sua organização

Essas características, das quais também foram definidas subcaracterísticas, abrange os aspectos de qualidade gerais para qualquer processo de software, e assim, podem ser usadas como uma lista de verificação para garantir uma avaliação completa da qualidade do processo.

Essa fase é constituída basicamente por três atividades, sendo elas, verificação do processo, especificação de requisitos e validação dos requisitos:

1. **Verificação do processo:** Essa atividade tem o objetivo definir os estado atual do processo. Para isso, é necessário verificar quais as características do modelo de características de qualidade estão presentes no processo. Para realizar esta atividade deve-se usar o modelo como uma lista de verificação e verificar quais características estão no modelo, e quais estão sendo cumpridas totalmente ou parcialmente. Por meio dessa lista de verificação é possível definir os requisitos para a avaliação do processo.
2. **Especificação de requisitos:** Essa atividade consiste em definir quais as características de qualidade que o processo deve conter de acordo com o modelo de qualidade de processo. A partir da lista de verificação definida é possível definir quais características devem ser avaliadas e quais devem estar presentes no processo.
3. **Validação dos requisitos:** Essa atividade consiste em rever e aprovar os requisitos definidos.

### 5.2.1.2 Definição de métricas

Na fase de definição de métricas objetiva-se definir as métricas e os métodos de medição para avaliar a modelagem de um processo de software.

Para definir as métricas, baseado no modelo de informação do PSM, deve-se definir os atributos, o método de medição, a medida básica, a função de medição, a medida derivada e o indicador. Como é apresentado na Figura 28.

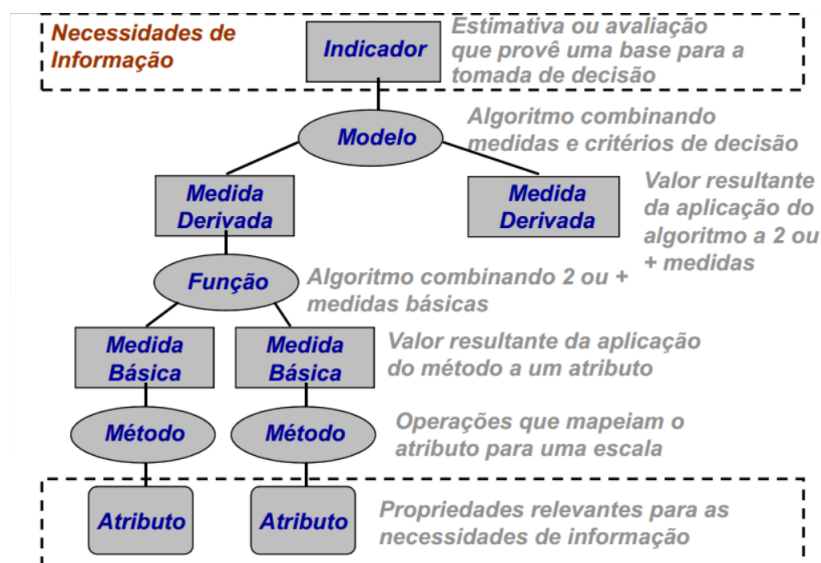


Figura 28 – Processo de medição de modelagem de processo de software



Cada um dos itens presentes na Figura 28 são detalhados a seguir:

- Atributo: propriedade ou uma característica de uma entidade de software que pode ser distinguida quantitativamente ou qualitativamente ISO (2007).
- Método de medição: sequência lógica de operações usadas para quantificar um atributo no que diz respeito a uma escala específica ISO (2007).
- Medida básica: medida de um único atributo definida por um método de medição específico ISO (2007);
- Medida derivada: medida ou quantidade definida como uma função de duas ou mais medidas base, ou medidas derivadas ISO (2007);
- Indicador: é a base quantitativa para a análise das medições e para as tomadas de decisão;

Com base na SQuaRE 25020 (2007), a definição de uma métrica deve incluir: o nome da métrica, a característica e a subcaracterística correspondentes, a descrição da métrica, o propósito da métrica, o tipo de métrica, o foco de medição, a função de medição, o método de medição. A especificação de uma métrica é mostrada na Tabela 9.

Tabela 9 – Modelo de especificação de métricas

<b>Título</b>	<b>Conteúdo</b>
Nome	Nome da métrica
Descrição	Expressa a questão a ser respondida ao aplicar a métrica
Propósito	razão de usar a métrica
Característica de qualidade	Característica de qualidade do modelo usado
Subcaracterística de qualidade	SubCaracterística de qualidade do modelo usado
Função de medição	Fórmula Medição e significados dos elementos de dados usados
Tipo de medida	Tipo de medida usada, podem ser: tamanho, tempo, contagem.
Tipo de escala	o tipo de valores utilizados, podem ser: escala nominal, ordinal, intervalo, racional.
Tipo de coleta	a forma como se dará a coleta de dados da métrica, pode ser manual, por meio de ferramentas online, questionário, entrevistas.
Coleta	o que será coletado durante a aplicação da métrica
Foco de medição	Foco da medição (por exemplo, o foco pode estar em "saídas", "decisões" ou "interações com outros processos")
Entradas	Nome dos artefatos que devem ser levados em consideração durante a medição

A Tabela 10 apresenta um exemplo de detalhamento de métrica para a característica Compreensibilidade Funcional, a qual foi adaptada de (Guceglioglu and Demirors, 2011).

Tabela 10 – Especificação da métrica compreensibilidade funcional

<b>Título</b>	<b>Conteúdo</b>
Nome	Compreensibilidade Funcional
Descrição	Quão fácil é compreender as atividades do processo?
Propósito	examinar a facilidade ou dificuldade encontrada pela equipe em compreender as atividades do processo
Característica de qualidade	Usabilidade
Subcaracterística de qualidade	Compreensibilidade
Função de medição	$X = A/B$ ; A= número de atividades em que o pessoal não encontrou dificuldades para compreender as tarefas a serem executadas, B= número de atividades
Tipo de medida	Quantidade de atividades
Tipo de escala	Nominal
Tipo de coleta	Questionário.
Coleta	Coletar o número de atividades. Coletar o número de atividades em que não se houve dificuldades para se compreender. A coleta será realizada analisando a quantidade de atividades compreendidas que o usuário relatou
Foco de medição	Atividade do processo
Entradas	

### 5.2.1.3 Coleta de dados

Na fase de coleta de dados, será executada a avaliação do processo com os envolvidos no processo e serão coletados os dados dessa avaliação. A coleta será realizada tal qual está especificada na definição da métrica.

### 5.2.1.4 Análise de dados

Nessa fase serão analisadas as métricas coletadas durante a execução da avaliação. Nessa fase pode ser feita uma comparação dos dados coletados com as metas da organização ou com as metas do processo. Pode ser aplicada uma lista de verificação dos requisitos de qualidade definidos no início do processo de medição. É importante realizar uma revisão da avaliação, revisar os resultados da avaliação.

### 5.2.1.5 Relato dos resultados

Ao coletar os dados e realizada a análise é importante elaborar um relatório como o resultados da avaliação da modelagem do processo. Esse relatório deve conter o objetivo da avaliação, os participantes da avaliação, o escopo da avaliação, o método utilizado para realizar a avaliação, o contexto do processo utilizado na avaliação, o procedimento e como

ocorreu a avaliação, as métricas utilizadas, os resultados obtidos, e as oportunidades de melhoria encontradas no processo.

## 5.2.2 Uso no contexto de implantação de MPS

O modelo de características de qualidade de processo pode ser utilizado ao decorrer das fases do IDEAL (iniciação, diagnóstico, estabelecimento, ação e aprendizado), conforme ilustrado na Figura 6 do Capítulo 2. O método de medição de processo de software e o uso do modelo de características de qualidade do processo pode ser utilizado, principalmente, em duas fases do IDEAL: na fase de diagnóstico e na fase de aprendizado.

### 5.2.2.1 Fase diagnóstico

A fase de diagnóstico é composta por duas atividades principais: caracterizar o estado atual e o desejado, e desenvolver recomendações. Durante a fase de diagnóstico no IDEAL, e com o objetivo de caracterizar o estado atual da organização é possível analisar o cenário atual do processo, e analisar a modelagem de processo de acordo com as características de qualidade definidas. Mas para realizar essa análise é necessário medir a modelagem, e assim, determinar a qualidade da modelagem do processo. Depois dessa análise pode-se identificar as oportunidades de melhoria. E por fim, definir os objetivos para alcançar essa melhoria, e assim, definir as recomendações para melhoria de processo.

Pode-se incluir as atividades do método de medição de modelagem de processo na fase de diagnóstico, adaptando assim o IDEAL ao modelo de qualidade de processo. A Figura 29 apresenta o processo para a fase de Diagnóstico e em qual atividade cada etapa do método de medição pode ser incluído.

Como pode ser visto na figura 29, cada atividade do método de medição pode ser realizado dentro das atividades da fase de diagnóstico do IDEAL. Sendo elas:

1. Determinar qual baseline é necessária: De acordo com o IDEAL (1996), essa atividade tem como objetivo determinar quantas baselines serão executadas e determinar quais os tipos de baselines. Incluindo a atividade, definição de requisitos, do processo de medição definido neste trabalho, pode-se definir quais os requisitos necessários para a partir disso definir as métricas. Dessa maneira o objetivo dessa atividade passa a ser: determinar quantas baselines serão executadas, determinar quais os tipos de baselines e definir os requisitos.
2. Planejar baselines: Essa atividade tem por objetivo garantir que todos os aspectos das atividades das baselines são levados em consideração e documentar as atividades necessárias para cada baseline. Além disso, nessa atividade pode-se definir as métricas para a medição da modelagem do processo.

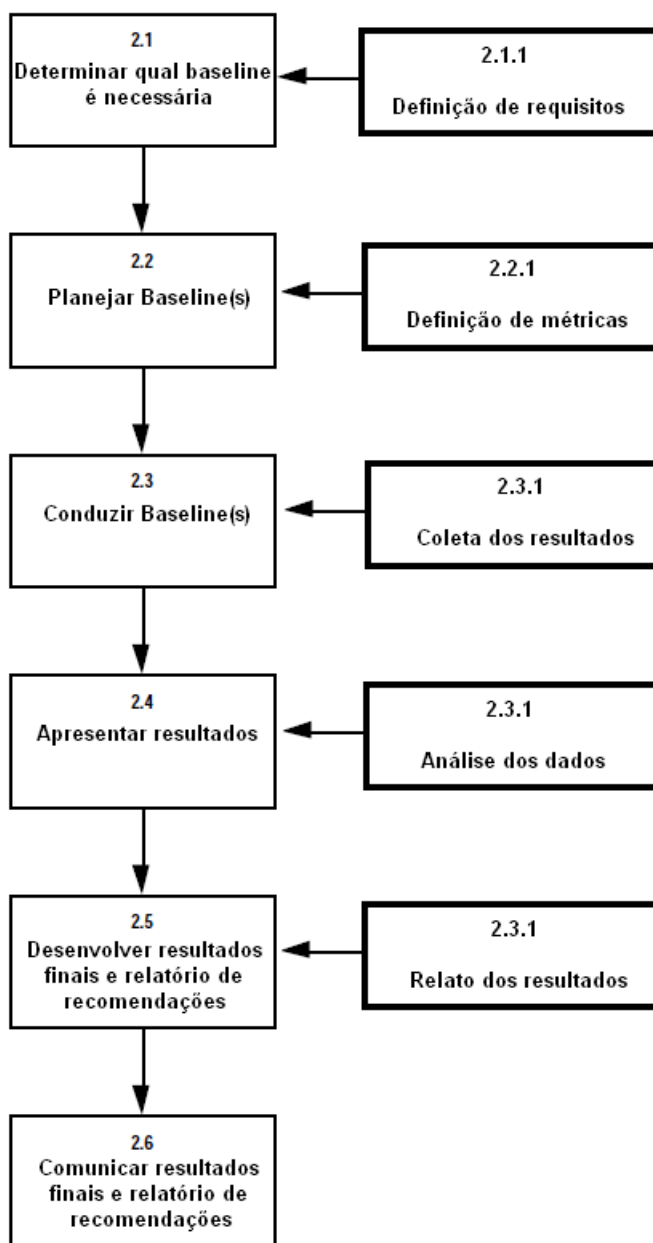


Figura 29 – Processo da fase diagnóstica. Adaptado do IDEAL (McFeeley, 1996).

3. Conduzir baselines: nessa etapa será conduzida a baseline com o objetivo de reunir todas as informações necessárias para a melhoria do processo. Já no processo de medição, durante essa atividade é realizada a medição da modelagem do processo.
4. Apresentar resultados: Nessa atividade são coletados os resultados da condução tanto da baseline quanto da medição da modelagem. Os resultados são analisados e um feedback inicial é apresentado.
5. Desenvolver resultados finais e relatório de recomendações: Durante essa atividade é desenvolvido um relatório apresentando os resultados finais e as recomendações desenvolvidas para cada resultado encontrado. É realizado também um relatório da

medição da modelagem do processo.

6. Comunicar resultados finais e relatórios de recomendações: Nessa atividade serão comunicados à organização os resultados e as recomendações feitas pela equipe de melhoria de processo.

#### 5.2.2.2 Fase aprendizado

Definido os objetivos, é realizado um planejamento durante a fase de estabelecimento. Após essa etapa, ocorre a execução do trabalho planejado, a fase de ação. E por fim, após a fase de execução, ocorre a fase de aprendizado, onde toda a experiência do IDEAL é revisada para determinar o que foi alcançado.

Nessa fase pode-se verificar se as mudanças implantadas trouxeram solução para o problema diagnosticado no início desse ciclo, para isso deve-se aplicar uma lista de verificação do modelo de características de qualidade e averiguar quais características estão presentes e se estão cumprindo as especificações definidas no início do processo de melhoria.

Essa atividade pode ser realizada durante a execução de uma das atividades da fase de aprendizado, a atividade de analisar lições aprendidas. Ela tem o objetivo de analisar as práticas e os processos de melhoria que foram executados para que se possa fazer com que o próximo ciclo do IDEAL trabalhe melhor, considerar a adição de novos processos para um melhor resultado no próximo ciclo do IDEAL, e considerar a exclusão de práticas e processos que não fizeram bem para o processo.

Como pode ser visto na Figura 30, pode-se incluir a atividade de aplicar a lista de verificação para auxiliar na obtenção dos objetivos dessa atividade.

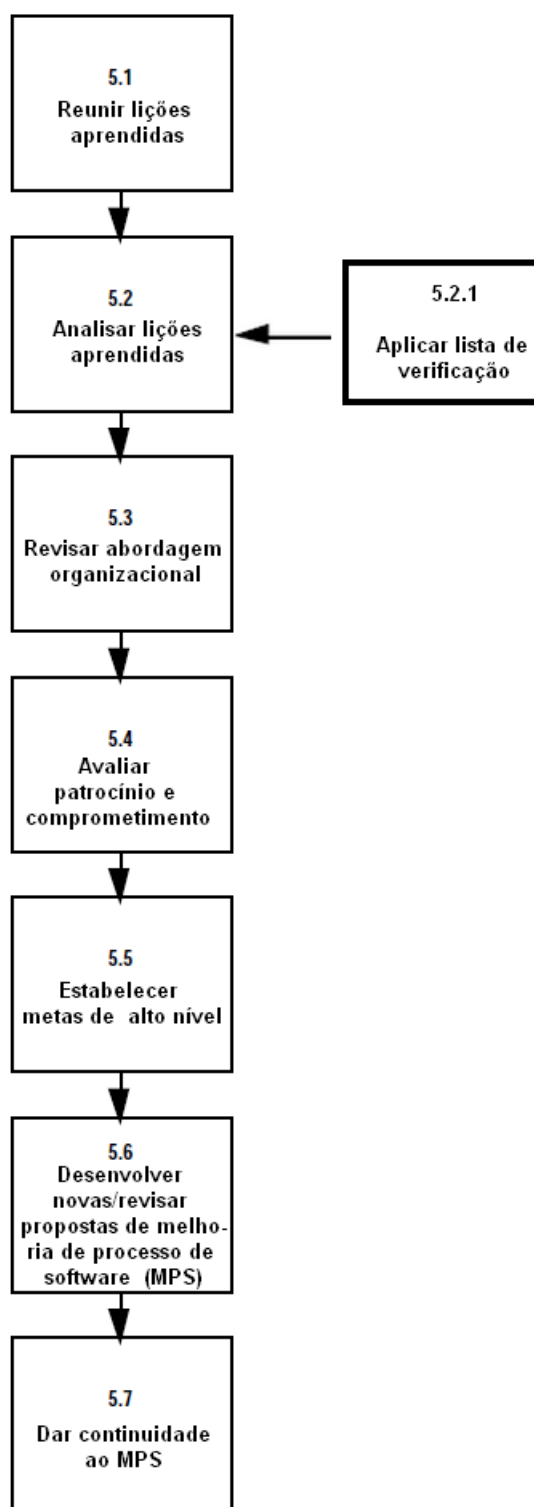


Figura 30 – Processo da fase de aprendizado. Adaptado do IDEAL (McFeeley, 1996).

## 6 Considerações Finais

Este trabalho teve como objetivo definir um instrumento para avaliar a qualidade da modelagem de um processo de software. Para alcançar esse objetivo foi realizado um mapeamento sistemático. Após a avaliação inicial de 256 artigos, foram selecionados seis estudos para compor o estudo secundário. Foi definida uma questão de pesquisa para encontrar características de qualidade de processo de software e algumas métricas já definidas.

Dos seis trabalhos selecionados foi possível extrair um grande número de dados, sendo identificadas 105 características de qualidade de modelagem de processos. Após um refinamento e validação dos dados foram selecionadas 48 características e, por meio destas, foi elaborado um modelo de qualidade de processo baseando-se na forma como a ISO/IEC 25010 define seu modelo de qualidade.

Foi realizada uma validação semântica e sintática do modelo, onde foi coletada a opinião de especialistas em melhoria de processo de software. Como resultado dessa validação, as 48 características tiveram seus conceitos validados. A partir da análise do resultado da validação foi possível identificar mudanças tanto na forma como as características e subcaracterísticas foram definidas, quanto na agregação de características identificadas como semelhantes.

Ressalta-se que, a partir da análise dos dados, o maior foco de respondentes foi do estado de São Paulo. Além disso, 60% dos respondentes já utilizaram o MPS.BR.

Dados os resultados deste trabalho conclui-se que os objetivos específicos propostos foram cumpridos. Esses objetivos foram: identificar as características de qualidade na modelagem de processo; definir um modelo de característica de qualidade de modelagem de processo; validar o modelo de qualidade de modelagem de processos de software; definir a forma de uso do modelo de características tanto de forma isolada quanto no contexto de MPS.

Foi definido neste trabalho formas de uso do modelo de característica de qualidade de processo, assim, esse modelo pode ser usado para definição de processos, para medição da modelagem de processo de software e para avaliar a qualidade da modelagem ou documentação de processo de software. Dessa forma, é possível garantir a qualidade do processo de software, e assim, ajudar a obter um produto de software com qualidade.

É importante ressaltar o modelo de características de qualidade de processo e o método de avaliação definido neste trabalho difere das diversas normas de avaliação de processo de software como, a SQuaRE e a ISO/IEC 15504, pois essas normas avaliam o



processo em execução, já o modelo de características de qualidade avalia a modelagem ou a documentação do processo antes da execução dele.

Considera-se, também, que como resultado do TCC1 foi elaborado um artigo que foi aceito em um congresso internacional, o SIMPDA (*Symposium on Data-Driven Process Discovery and Analysis*). E pretende-se escrever e submeter outro trabalho, resultado do TCC2, em outro congresso.

A partir das evidências identificadas, foi possível concluir que existem poucos estudos que tratam da qualidade de processo de software, pois em um mapeamento sistemático só foram encontrados seis trabalhos. Dessa maneira, este trabalho se propõe a preencher uma lacuna nos estudos relacionados a qualidade de modelagem de processo de software de modo a auxiliar na construção de um instrumento que permita avaliar a qualidade da modelagem de processo de software.

## 6.1 Trabalhos Futuros

Devido ao tema do trabalho ser restrito, houve dificuldades em obter um número grande de respondentes. Dado o número de respondentes não foi possível realizar uma análise fatorial do questionário. Dessa forma, para uma melhor validação do instrumento de pesquisa sugere-se como trabalho futuro, em vez de realizar um questionário, aplicar o survey em uma população maior e realizar um estudo de caso para verificar o uso do método em um ambiente real. Assim, será possível identificar benefícios no uso do modelo de características de qualidade em modelagem de processos.

Além disso, é importante definir exemplos de métricas para cada um dos itens do modelo a fim de facilitar sua aplicação em um ambiente real. Incentiva-se também a definição de métricas para todas as características do modelo definido, e a aplicação dessas métricas em um estudo de caso.

Ressalta-se que é aconselhável detalhar as formas de uso quer foram apresentadas no Capítulo 5 mas não foram desenvolvidas nesse trabalho. As formas de uso que devem ser detalhadas são: Auxiliar no contexto do nível 3 do CMMI (CMM, 2006) ou do nível D do MPS.BR (SOF, 2012); e, avaliações tipo B do SCAMPI.

# Referências

- CMMI Para Desenvolvimento*, 2006. 9, 68, 80
- D. Almeida, M. A. R. Santos, and A. F. B Costa. AplicaÇÃo do coeficiente alfa de cronbach nos resultados de um questionÁrio para avaliaÇÃo de desempenho da saÚde pÚblica. *XXX Encontro Nacional de Engenharia da ProduÇÃo*, 2010. 41
- H. Arksey and L. O'Malley. Scoping studies: Towards a methodological framewrok. *International Journal of Social Research Methodology: Theory and Practice*, 2005. 36
- V. R. Basili. Software modeling and measurement: The goal/question/metric paradigm. *Institute for Avanced Computer Studies*, 1992. 32, 33
- Victor R. Basili, G. Caldiera, and H. D. Rombach. The goal question metric approach. *Institute for Avanced Computer Studies*, 1994. 32
- J. Biolchine, Paula G Mian, Ana C. C. Natali, and G. H. Travassos. Systematic review in software engineering. 2005. 36
- L. G. Borgoñón, J. A. G. García, M. A. Barcelona, and M.J. Escalona. Software process accessibility in practice: a case study. *5th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-exclusion*, 2013. 59, 60, 62
- L. Borgoñón-García, M.A. Barcelona, J.A. García-García, M. Alba, and M.J. Escalona. Software process modeling languages: A systematic literature review. *Jornadas de Ingeniería del Software y bases de Datos (JISBD)*, 2013. 10
- V. M. Brito. Proposta para um conjunto de competências para um product owner. 2014. 38
- D. N. Card. Practical software measurement. *International Conference on Software Engineering (ICSE)*, 2003. 30
- Departamento de Computação da Universidade Federal de São Carlos DC/UFSCar. Start, state of the art through systematic review, 2005. URL <http://lapes.dc.ufscar.br/>. 43, 89
- J. C. Derniame, B. A. Kaba, and D. Wastell. Software process:principles, methodology and technology. 1999. 23
- T. Dyba. An empirical investigation of the key factors for success in software process improvement. 2005. 24

- B. S. Everitt. Multivariate analysis: The need for data, and other problems. *British Journal of Psychiatry*, 1975. 39
- P. Feiler and W. Humphrey. Software process development and enactment: concepts and definitions. 1992. 65
- A. Fink and J. Kosecoff. How to conduct surveys: A step by step guide. 1985. 38
- A. L. P. Freitas and S. G. Rodrigues. A avaliação da confiabilidade de questionários: uma análise utilizando o coeficiente alfa de cronbach. 2005. 7, 42
- H. Freitas, M. Oliveira, A. Z. Saccol, and J. Moscarola. O metodo de pesquis survey. *Revista de Administração*, 2000. 38, 39
- L. García-Borgoñón, J.A. García-García, M. Alba, and F. J. Domínguez-Mayo. Gestión de procesos en organizaciones de desarrollo de software: Un enfoque basado en modelos. *Jornadas de Ingeniería del Software y bases de Datos (JISBD)*, 2013. 9, 10
- A. A. A. Ghani, K. T. W. M. Muketha, and W. P. Wen. Complexity metrics for measuring the understandability and maintainability of business process models using goal-question-metric (gqm). *IJCSNS International Journal of Computer Science and Network Security*, 2008. 33
- A. S. Guceglioglu and O. Demirors. A process based model for measuring process quality attributes. *Lecture notes in computer science*, 2005. 63, 65, 66
- A. S. Guceglioglu and O. Demirors. The application of a new process quality measurement model for software process improvement initiatives. *11th International Conference On Quality Software*, 2011. 48, 59, 65, 73
- H. Gunther. Como elaborar um questionário, planejamento de pesquisas para ciências sociais. 2003. 39
- J. F. Hair, R. E. Anderson, R. L. Tatham, and W. C. Black. *Multivariate data analysis*, volume 5. New Jersey: Prentice Hall, 1998. 42
- IBM. Ibm spss statistics. URL <http://www-01.ibm.com/software/analytics/spss/products/statistics/>. 42
- ISO9126-1 Engenharia de software - Qualidade de produto parte 1: Modelo de qualidade*. ISO/ABNT NBR - Associação Brasileira de Normas tecnicas, Rio de Janeiro, Brasil, 2003. 10
- ISO9000 Sistemas de gestão de qualidade - Fundamentos e vocabularios*. ISO/ABNT NBR - Associação Brasileira de Normas tecnicas, Rio de Janeiro, Brasil, 2 edition, 2005a. 19, 23

- ISO9001 Sistemas de gestão de qualidade - Requisitos.* ISO/ABNT NBR - Associação Brasileira de Normas técnicas, Rio de Janeiro, Brasil, 2 edition, 2008a. [19](#), [20](#)
- ISO25000 Software Engineering - Software product Quality Requirements and Evaluation (SQuaRE) - Guide to SQuaRE.* ISO/IEC - International Organization for Standardization and International Electrotechnical Commission, Switzerland, 1 edition, 2005b. [19](#)
- ISO25020 Software engineering — Software product Quality Requirements and Evaluation (SQuaRE) — Measurement reference model and guide.* ISO/IEC - International Organization for Standardization and International Electrotechnical Commission, Switzerland, 1 edition, 2007. [72](#)
- ISO25012 System and Software Engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) — Data quality model.* ISO/IEC - International Organization for Standardization and International Electrotechnical Commission, Switzerland, 1 edition, 2008b. [20](#)
- ISO25010 System and Software Engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models.* ISO/IEC - International Organization for Standardization and International Electrotechnical Commission, Switzerland, 1 edition, 2011. [7](#), [20](#), [21](#), [22](#), [58](#), [59](#), [69](#)
- Internationa Standard ISO/IEC-15504 Systems and software engineering.* ISO/IEC - International Organization for Standardization, International Electrotechnical Commission and Institute of Electrical and Electronics Engineers, USA, 1 edition, 1998. [26](#)
- Internationa Standard ISO/IEC-12207 Systems and software engineering.* ISO/IEC - International Organization for Standardization, International Electrotechnical Commission and Institute of Electrical and Electronics Engineers, USA, 2 edition, 2008c. [7](#), [9](#), [23](#), [26](#)
- B. Kitchenham. Procedures for performing systematic reviews. *Joint Technical Report Software Engineering Group*, 2004. [35](#)
- B. Kitchenham and P. Brereton. A systematic review of systematic review process research in software engineering. 2013. [88](#)
- B Kitchenham and S. Charters. Guidelines for performing systematic literature reviews in software engineering. *EBSE Technical Report*, 2007. [35](#), [36](#)
- B. Kitchenham, L. Pickard, S. Linkman, and P. Jone. A framework for evaluating a software bidding model. *Information and Software Technology*, 2005. [63](#), [66](#)

- B. A. Kitchenham, D. Budgen, and O. P. Brereton. Using mapping studies as the basis for further research – a participant-observer case study. 2010. 36
- T. A. Kroeger and Davidson. A perspective-based model of quality for software engineering processes. 2009. 59, 60, 62, 64
- T. A. Kroeger, N. J. Davidson, and S. C. Cook. Understanding the characteristics of quality for software engineering processes: A grounded theory investigation. 2013. 62, 63, 64, 65, 66, 67
- S. N. Mafra and G. H. Travassos. Estudos primários e secundários apoiando a busca por evidência em engenharia de software. 2006. 35, 36
- F. N. MATTAR. *Pesquisa de Marketing*. São Paulo, 1996. 39
- B. McFeeley. *Ideal: A user's guide fo software process improvement*. 1996. 7, 9, 28, 75, 76, 78
- J. McGarry, D. Card, C. Jones, B. Layman, E. Clar, J. Dean, and F. hall. *Practical Software Measurement: Objective Information for Decision Makers*. 2002. 7, 30, 31
- F. F. Mendes. Melhoria de processos de tecnologia da informação multi-modelo. 2010. 11
- P. A. Morettin and W. O. Bussab. *Estatística Básica*, volume 5. 2009. 7, 52
- G. M. Muketha, A. A. A. Ghani, M. H. Selemat, and R. Atan. A survey of business process complexity metrics. *Information Technology Journal*, 2010. 10, 11
- Timothy G. Olson. Defining short and usable processes. *The Journal of Defense Software Engineering*, 2010. 9
- R. E. Park, Goethert W. B., and Florac W. A. Goal driven software measurement – a guidebook. 1996. 29
- Guia PMBOK Um guia do conhecimento em gerenciamento de projetos*. PMI, Project Management Institute, Inc, Pennsylvania, EUA, 2008. 9
- R. S. Pressman. *Engenharia de Software*, volume 6. McGraw-Hill, 2006. 9, 22, 29
- D. F. Rico. Software process improvement - impacting the bottom line by using powerful “solutions”. 2000. 24
- A. R. C. Rocha, José C. Maldonado, and Kival C. Weber. Qualidade de software teórica e pratica. 2001. 10
- J. M Roche. Software metrics and measurement principles. *Software Engineering Notes, ACM*, 1994. 29

- E. M. Silva, E. L. e Menezes. Metodologia de pesquisa e elaboração de dissertação. volume 3, Universidade Federal de Santa Catarina, Florianópolis, Brasil, 2001. 34
- F. O. Simon. Habilidades e competências em engenharia: Criação e validação de um instrumento. 2004. 38, 42
- MPS.BR *Melhoria de Processo do Software Brasileiro*. SOFTEX - Associação para Promoção da Excelência do Software Brasileiro, Brasil, 2012. 9, 68, 80
- I. Sommerville. *Software Engineering*, volume 8. Pearson Education, 2007. 22
- S. Sorumgard and G. Sindre. *Aspects of process quality, in: Proceedings of the 4th Software Quality Conference*. University of Abertay Dundee, 1995. 64, 65, 66
- A. L. Storolli, G. I. Zanolla, J. E. Guidini, and B. T. Borsoi. Modelagem de processo de software. 2009. 23
- S. Tyrrell. The many dimensions of the software process. *Crossroads - Special issue on Windows programming*, 2000. 23
- W3C. Web content accessibility guidelines quick reference, 2014. URL <http://www.w3.org/WAI/WCAG20/quickref>. 60
- C. Wohlin, P. Runeson, M. Host, M. C. Ohlsson, B. Regnel, and A. Wesslen. *Experimentation in Software Engineering*. Springer, 2012. 51
- K. Z. Zamli and P. A. Lee. Taxonomy of process modeling languages. England, 2001. 9, 10, 23

# Apêndices

# APÊNDICE A – Protocolo de Mapeamento Sistemático

## A.1 Tema

O objetivo deste mapeamento sistemático é identificar as características de qualidade de uma modelagem de processos.

## A.2 Problema

Não existem estudos suficientes que tratam de métricas para avaliar a modelagem de processo de um software, e conseqüentemente não são muito bem conhecidas as características de uma boa modelagem de processo.

### A.2.1 Questão

Será procurada resposta para a seguinte questão:

- Quais são as características de qualidade de uma modelagem de processo de software?

### A.2.2 Palavras-chave e sinônimos

Serão consideradas como palavras-chave da língua inglesa as seguintes palavras, como mostra a Tabela 11.

Tabela 11 – Tabela de palavras chaves

characteristics	modeling	software process	quality
taxonomy	defining		
attributes	description		
property	representation		

## A.3 Seleção de Fontes de Pesquisa

Para selecionar os trabalhos é importante definir as fontes de pesquisa utilizar, para isso foram definidos alguns critérios.



### A.3.1 Critérios para seleção de fonte

As fontes de pesquisa foram selecionadas de acordo com os seguintes critérios:

- Os trabalhos da fonte de pesquisa deverão estar disponíveis gratuitamente
- Deverá existir uma máquina de busca para a realização da pesquisa
- Os trabalhos da fonte de pesquisa deverão possuir disponibilidade para consulta na Internet
- As fontes de pesquisar deverão estar presentes utilizando as palavras-chave

### A.3.2 Idiomas

Serão selecionados trabalhos escritos em inglês.

### A.3.3 Strings de pesquisa

A string de busca criada para a pesquisa é a seguinte:

(characteristic OR taxonomy OR attribute OR property) AND (modeling OR defining OR description OR representation) AND ("software process") AND (quality))

### A.3.4 Lista inicial de Fontes de Pesquisa

De acordo com Kitchenham (2013), para selecionar as fontes de pesquisar é importante identificar um conjunto adequado de bibliotecas para realizar a pesquisa, duas fontes de pesquisa podem ser específicas da área de forma a garantir fontes com boa cobertura de revistas e conferências, e pelo menos duas fontes de indexação geral. Tomando esse princípio como base, foram selecionadas as seguintes bases:

- IEEE Computer Science Digital Library: <http://ieeexplore.ieee.org/Xplore/home.jsp>
- ACM Digital Library: <http://dl.acm.org/>
- Web of Knowledge: <http://apps.webofknowledge.com/>
- Science Direct: <http://www.sciencedirect.com/>

## A.4 Seleção de trabalhos

Nesta seção serão apresentados os critérios de seleção utilizados, bem como os procedimentos adotados para decidir qual trabalho entraria e qual não entraria neste estudo.

#### A.4.1 Critérios de Inclusão

Os trabalhos deverão ser selecionados seguindo os seguintes critérios:

1. O trabalho deve ser em inglês;
2. O trabalho deve estar disponível na web;
3. O trabalho deve possuir menção a características de qualidade de processo ou possuir menção à qualidade da modelagem ou taxonomias ou atributos ou propriedades sobre a modelagem de um processo

#### A.4.2 Critérios de Exclusão

Os trabalhos deverão ser excluídos seguindo os seguintes critérios

1. Os estudos que não relataram resultados empíricos ou literatura que estava disponível apenas na forma de resumos;
2. Os estudos que não relataram resultados empíricos ou literatura disponível em forma de apresentações PowerPoint;
3. O estudo fora do domínio de computação ou engenharia de software;
4. O trabalho não faz menção a modelagem ou a qualidade de modelagem;
5. O trabalho não faz menção às características de qualidade de modelagem ou qualidade de processo.

#### A.4.3 Procedimento para seleção dos trabalhos

As *strings* de busca devem ser utilizadas para a busca de trabalho nas fontes de pesquisa escolhidas. A condução do MS será apoiada por uma ferramenta chamada StArt (DC/UFSCar, 2005). Devem ser lidos os títulos, *abstracts* e palavras-chave, a fim de aplicar os critérios de inclusão e exclusão definidos na Seção A.4.1 e A.4.2 deste protocolo. Com bases nessas informações o trabalho será pré-selecionado. Durante a análise dos trabalhos será lida a introdução e conclusão do trabalho, assim será feita uma segunda pré-seleção. Finalizando, os trabalhos que passarem por todos os critérios de exclusão deverão ser lidos por completo. Durante a leitura completa do trabalho deverão ser extraídas as informações constantes no formulário de coleta de dados. Seção A.5.

## A.5 Coleta de Dados

Após a aplicação da string de busca os trabalhos selecionados serão catalogados em uma planilha de dados para análise posterior. Para cada trabalho serão extraídos dados presentes na tabela 12.

Tabela 12 – Coleta de dados do trabalho

<b>Coleta de dados do trabalho</b>	
<b>Título:</b>	
<b>Autor(es):</b>	
<b>Ano de publicação:</b>	
<b>Palavras-chave:</b>	
<b>Resumo:</b>	
<b>Fonte:</b>	<IEEm ACM, WOK e Science Direct>
<b>Assuntos Contemplados</b>	
<b>Lista das características de qualidade presentes no artigo</b>	
<Colocar aqui algum atributo ou característica que detalhe o que vem a ser qualidade em processos de software. Espera-se que seja algo semelhante as características e atributos de qualidade de produto de software na ISO 9126 >	

## A.6 Indicadores

Ao final do mapeamento sistemático será realizado um sumarização dos dados com o levantamento e análise dos indicadores, mostrados na Tabela 13.

Tabela 13 – Base de pesquisa - indicadores

<b>Nome do Indicador</b>
Número de trabalhos identificados
Número de trabalhos identificados por fonte: IEEE
Número de trabalhos identificados por fonte: ACM
Número de trabalhos identificados por fonte: Science Direct
Número de trabalhos identificados por fonte: WOK
Número de trabalhos excluídos na seleção
Número de trabalhos duplicados na seleção
Número de trabalhos selecionados
Número de trabalhos selecionados por fonte: IEEE
Número de trabalhos selecionados por fonte: ACM
Número de trabalhos selecionados por fonte: Science Direct
Número de trabalhos selecionados por fonte: WOK
Número de trabalho excluídos na extração
Número de trabalhos aceitos na extração
Número de trabalhos duplicados na extração
Número de trabalho aceitos por fonte: IEEE
Número de trabalho aceitos fonte: ACM
Número de trabalho aceitospor fonte: Sicience Direct
Número de trabalho aceitos por fonte: WOK
Número de trabalhos aceitos por critério de Inclusão 1
Número de trabalhos aceitos por critério de Inclusão 2
Número de trabalhos aceitos por critério de Inclusão 3
Número de trabalho excluidos por critério de exclusão 1
Número de trabalho excluidos por critério de exclusão 2
Número de trabalho excluídos por critério de exclusão 3
Número de características de qualidade de processo identificadas

## APÊNDICE B – Conjunto inicial de características de qualidade de processo

São apresentadas neste apêndice as características iniciais levantadas a partir do mapeamento sistemático. A referência das características utilizada na Tabela B que representa de onde foram inspiradas as características em questão, segue a numeração definida na seção 4.1.

Tabela 14 – Características de qualidade do processo

Características			Origem
Usabilidade	Acessibilidade	Texto Alternativo	[ 2 e 3]
		Navegabilidade	[ 2 e 3]
		Acessibilidade da sequencia do fluxo de trabalho	[ 2 e 3]
	Compreensibilidade		[1, 2 e 6]
	Apreensibilidade		[2 e 6]
	Adaptabilidade	Ajustabilidade	[2 e 3]
		Escalabilidade	[2 e 3]
		Flexibilidade	[2 e 3]
	Operabilidade		[6]
	Atratividade		[6]
Confiabilidade	Recuperabilidade		[6]
Funcionalidade	Adequabilidade	Efetividade	[2 e 6]
		Eficiência	[2 e 6]
		Aplicabilidade	[2 e 6]
		Compatibilidade	[2 e 6]
	Interoperabilidade		[6]
	Acurácia		[6]
	Segurança		[6]
Evolubidade	Analisabilidade		[2 e 6]
	Modificabilidade		[2]
	Projetabilidade		[2]
	Apoio		[2]
Semântica	Conformidade		[2]
	Capacidade de resposta		[2]
	Validade		[1]
	Especificidade		[2]
	Completude		[1]

Sintática	Corretude	[1]	
Arquitetura	Tamanho	[2]	
	Complexidade	[2]	
	Modularidade	[2]	
	Estabilidade	[2]	
Representação	Consistência	[2]	
	Clareza	[2]	
	Adequação ao formato	[2]	
	meio de comunicação	[2]	
Execução	Custo	[2]	
	Fidelidade	[2]	
	Repetibilidade	[2]	
	Mensurabilidade	[2]	
	Capacidade de automação	[2]	
	Tolerância a falhas	[2]	
	Rigor	[2]	
	Gerenciabilidade	Previsibilidade	[3 e 5]
		Monitorabilidade	[5]
		Controlabilidade	[5]

# APÊNDICE C – Conjunto final de características de qualidade de processo

Conjunto final de características de qualidade de processo. A referência das características utilizada na Tabela B que representa de onde foram inspiradas as características em questão, segue a numeração definida na seção 4.1.

Tabela 15 – Características de qualidade do processo

Características			Referencia	
Usabilidade	Acessibilidade	Texto Alternativo	[ 2 e 3]	
		Navegabilidade	[ 2 e 3]	
		Acessibilidade da sequencia do fluxo de trabalho	[ 2 e 3]	
	Compreensibilidade		[1, 2 e 6]	
	Apreensibilidade		[2 e 6]	
	Adaptabilidade	Ajustabilidade	[2 e 3]	
		Flexibilidade	[2 e 3]	
		Controlabilidade	[5]	
	Operabilidade		[6]	
	Atratividade		[6]	
	semântica	Conformidade		[2]
		Capacidade de resposta		[2]
		Validade		[1]
		Compleitude		[1]
	Execução	Custo		[2]
		Mensurabilidade		[2]
		Capacidade de automação		[2]
		Tolerância a falhas		[2]
		Rigor		[2]
		Gerenciabilidade	Previsibilidade	
Monitorabilidade			[5]	
Funcionalidade	Adequabilidade	Efetividade	[2 e 6]	
		Eficiencia	[2 e 6]	
		Compatibilidade	[2 e 6]	
	Interoperabilidade		[6]	
	Segurança		[6]	
Evolubilidade	Analisabilidade		[2 e 6]	
	Modificabilidade		[2]	
	Apoio		[2]	

Sintática	Corretude	[1]
	Adequação ao formato	[2]
Arquitetura	Tamanho	[2]
	Complexidade	[2]
	Modularidade	[2]
	Estabilidade	[2]
Representação	Consistência	[2]



# APÊNDICE D – Primeira versão do questionário

## **Avaliação de Qualidade de modelagem de processo de software**

Prezado Colaborador(a)

Você está sendo convidado(a) a responder um questionário que avaliará as características de qualidade de processo de software em termos de completude, coerência e clareza. Esse questionário faz parte de um trabalho de conclusão de curso em Engenharia de Software na Universidade de Brasília (UnB).

A Figura 1 traz uma representação gráfica do modelo que será avaliado. O objetivo do modelo é prover um conjunto de características que auxiliem na avaliação da qualidade de um processos de software, tanto em termos da execução quanto em termos da documentação. Esse modelo foi elaborado a partir de um mapeamento sistemático, onde foram identificadas características de qualidade de processo de software. Essas características foram analisadas e refinadas, e baseada no modelo de qualidade de produto da norma ISO 9126 foi elaborado o modelo apresentado na Figura 1.

O questionário está organizado em duas grandes partes. Inicialmente será proposta a avaliação do primeiro nível do modelo, ou seja, as características de qualidade. A segunda parte irá avaliar os demais níveis do modelo, ou seja, as subcaracterísticas de qualidade.

Desde já agradecemos a sua colaboração e colocamo-nos a disposição para qualquer esclarecimento.



### **Características de Qualidade**

Você deverá atribuir uma nota de 0 ou 1 para avaliar a clareza da explicação da característica, onde 0 deve ser atribuído quando não estiver clara e 1 quando estiver clara.

Confuso (0)

Claro (1)

2. **Confiabilidade.** Relacionada à capacidade do processo ser confiável, de modo a lidar com taxa de falhas. Algumas falhas que podem ser geradas por meio de erros de usuários, cometidos, por exemplo, no momento de completar ou atualizar um formulário, documentos ou registros, pela equipe ou gerentes. A única subcaracterística relacionada é recuperabilidade.

0. Confuso   1. Claro

2. **Usabilidade.** Relacionada à facilidade de interpretar e executar uma modelagem de processo, sob condições específicas, para alcançar os resultados desejados. Pode ser classificada como: acessibilidade, compreensibilidade, apreensibilidade, adaptabilidade, operabilidade, atratividade, semântica e execução.

0. Confuso   1. Claro

3. **Funcionalidade.** Relacionada à capacidade do processo de disponibilizar as funções necessárias para o processo, essas funções são: adequação, acurácia, segurança e interação com outros processos.

0. Confuso   1. Claro

4. **Evolubilidade.** Relacionada à facilidade com que um processo de software está disponível para ser avaliado e modificado com o propósito de manutenção e melhoria de processo. Evolubilidade pode ser classificada em quatro subcaracterísticas: analisabilidade, modificabilidade, projetabilidade e apoio.

0. Confuso   1. Claro

5. **Sintática.** É definida como a relação do modelo com a linguagem de modelagem. Descreve as relações entre as construções de linguagem, sem considerar o seu significado. Pode ser classificado como corretude.

0. Confuso   1. Claro

6. **Arquitetura.** Relacionada com os elementos de um processo de software que dizem respeito à sua organização. Influencia significativamente na forma como o processo é percebido. Arquitetura pode ser classificado em quatro subcaracterísticas: tamanho, complexidade, modularidade e estabilidade.

0. Confuso   1. Claro

7. **Representação.** Relacionada à capacidade de um processo ser documentado e comunicado a todas as partes interessadas. Representação pode ser classificado em quatro subcaracterísticas: clareza, consistência, formato e meio de comunicação.

0. Confuso   1. Claro

### Detalhamento das características de qualidade

Como pode ser visto na Figura 1, cada uma das características de qualidade propostas são detalhadas em termos de subcaracterísticas de qualidade e, em alguns casos, as subcaracterísticas de qualidade são detalhadas em subsubcaracterísticas de qualidade. Nesta parte do questionário será avaliada a clareza das definições propostas quanto a coerência, ou seja, se de fato as subcaracterísticas deveriam estar onde estão no modelo.

Você deverá atribuir uma nota de 0 ou 1 para avaliar a clareza da explicação da característica, onde 0 deve ser atribuído quando não estiver clara e 1 quando estiver clara.

Confuso (0)

Claro (1)

### Quanto aos aspectos específicos de confiabilidade

1. A única característica relacionada a confiabilidade é **recuperabilidade** que pode ser definida como a capacidade do processo continuar com os dados depois de ocorrer eventos anormais.

0. Confuso   1. Claro

### Quanto aos aspectos específicos de usabilidade

2. **Acessibilidade**, definida como a facilidade de encontrar informações sobre um processo de engenharia de software, acessibilidade esta dividida em três subcaracterísticas, texto alternativo, navegabilidade e acessibilidade da sequência do fluxo de trabalho.

0. Confuso   1. Claro

- 2.1. **Texto alternativo** é definido como a capacidade de transformar qualquer conteúdo não textual em outras formas de acordo com a necessidade do usuário, tais como letras grandes, braile, fala, símbolos ou linguagens mais simples

0. Confuso   1. Claro

- 2.2. **Navegabilidade** é definida como a capacidade do usuário navegar, localizar conteúdos e determinar onde eles estão em um processo de software

0. Confuso   1. Claro

- 2.3. **Acessibilidade da sequência do fluxo de trabalho** Trata da capacidade da sequência de leitura do fluxo de trabalho ser apresentada de forma correta, pois a sequência em que o conteúdo é apresentado afeta o seu significado

0. Confuso   1. Claro

3. **Compreensibilidade** é definida como a facilidade com a qual o usuário de um processo é capaz de compreender se um processo de engenharia de software é relevante e como ele pode ser utilizado para alcançar os resultados desejados.

0. Confuso   1. Claro

4. **Apreensibilidade** é definida como a facilidade com que um usuário do processo é capaz de aprender a executar uma atividade do processo.

0. Confuso   1. Claro

5. **Adaptabilidade** é a capacidade do processo se adaptar em diferentes situações sem perder suas informações ou sua estrutura, facilitando a utilização do processo pelo usuário, por exemplo quando uma organização busca criar um conjunto de padrões de processo para uso em múltiplos projetos. A adaptabilidade pode ser classificada em três subclassificações: Ajustabilidade, escalabilidade e flexibilidade.

0. Confuso   1. Claro

- 5.1. **Ajustabilidade** Trata da facilidade de um processo padrão ser adaptado para formar um processo mais específico.

0. Confuso   1. Claro

- 5.2. **Escalabilidade** é definida como facilidade de um processo poder ser adaptado para projetos de diferentes tamanhos.

0. Confuso   1. Claro

- 5.3. **Flexibilidade** Trata da facilidade que um profissional pode variar seu desempenho em certas atividades do processo para satisfazer uma necessidade específica, sem a necessidade de alteração do próprio processo.

0. Confuso   1. Claro

6. **Operabilidade** é a capacidade do processo ser utilizado e controlado.

0. Confuso   1. Claro

7. **Atratividade** é a capacidade do processo de envolver os usuários com seus projetos de modelos e/ou documentos, por exemplo: interface de usuário.

0. Confuso   1. Claro

8. **Execução**. É definida como o uso do processo de software dentro de um ambiente específico. Execução pode ser classificado em oito subcaracterísticas: fidelidade, custo, repetibilidade, mensurabilidade, capacidade de automação, rigor, tolerância a falhas e gerenciabilidade.

0. Confuso   1. Claro

- 8.1. **Fidelidade** é definida como a capacidade com a qual um processo de software é executado tal qual está documentado.

0. Confuso   1. Claro

**8.2. Custo** é definido como a quantidade de esforço, tempo e dinheiro necessários para executar um processo de software dentro de um contexto específico.

0. Confuso   1. Claro

**8.3. Repetibilidade** é definida como a capacidade de um processo de software ser executado várias vezes de uma forma similar.

0. Confuso   1. Claro

**8.4. Mensurabilidade** é definida como a capacidade da execução de um processo de software oferecer medidas relacionadas com seu desempenho.

0. Confuso   1. Claro

**8.5. Capacidade de automação** é definida como a capacidade de um processo de software ser executado usando recursos computacionais nas atividades do processo.

0. Confuso   1. Claro

**8.6. Rigor** é definido como a medida que um processo de software requer para que suas atividades sejam realizadas de maneira controlada e disciplinada.

0. Confuso   1. Claro

**8.7. Tolerância a falhas** é definida como o grau em que um processo, uma vez que uma intrusão ocorra, ou continua a produzir resultados precisos ou reconhece a imprecisão de seus resultados e inicia uma ação corretiva.

0. Confuso   1. Claro

**8.8. Gerenciabilidade** definida como a facilidade de gerenciar as mudanças em um processo, ou seja, a facilidade com que uma mudança em um processo pode ser prevista, monitorada e controlada para alcançar os resultados desejados, em um contexto específico. Gerenciabilidade pode ser classificada como: previsibilidade, monitorabilidade e controlabilidade.

0. Confuso   1. Claro

**8.8.1. Previsibilidade** é definida como a facilidade com que um gestor de processo é capaz de determinar o tempo, recursos, riscos e limitações associadas com a execução do processo.

0. Confuso   1. Claro

**8.8.2. Monitorabilidade** é definida como a facilidade com que um gerente de processo é capaz de determinar o status e os resultados de um processo executado

0. Confuso   1. Claro

**8.8.3. Controlabilidade** é definida como a facilidade com que gestor de processos é capaz de aplicar ações preventivas e corretivas para um processo e continuar mantendo um nível de desempenho especificado

0. Confuso   1. Claro

9. **Semântica** é definida como a capacidade de relacionar o modelo com o domínio, considerando as relações entre as instruções e seus significados, ou seja, relacionam o escopo do modelo com a intenção de suas atividades. Semântica pode ser classificada em cinco subcaracterísticas: conformidade, capacidade de resposta, especificidade, validade e completude.

0. Confuso   1. Claro

- 9.1. **Conformidade** é definida como a capacidade com a qual o processo de software está de acordo com padrões e modelos de processos reconhecidos na indústria.

0. Confuso   1. Claro

- 9.2. **Capacidade de resposta** é definida como a velocidade de resposta com que um processo de software é capaz de incorporar um feedback nos envolvidos com o processo.

0. Confuso   1. Claro

- 9.3. **Especificidade** é definida como a capacidade de um processo de software ser adequado para a sua execução em um contexto particular. Por exemplo: determinados processos podem ser criados de tal forma que se encontrem otimizados para utilização em uma situação específica.

0. Confuso   1. Claro

- 9.4. **Validade** está relacionada a adequação do processo ao domínio e às características do time para o qual foi documentado

0. Confuso   1. Claro

- 9.5. **Completude** é capacidade do processo de software ou modelo conter todas as declarações sobre o domínio, sendo todas elas corretas e relevantes.

0. Confuso   1. Claro

#### Quanto aos aspectos específicos de funcionalidade

10. **Adequabilidade** é definida como a capacidade do processo para produzir os resultados desejados de forma apropriada em um contexto específico, ou seja, a capacidade das atividades do processo de serem completas e adequadas para a execução de tarefas. Adequabilidade por ser classificada em: aplicabilidade, efetividade, eficiência e compatibilidade.

0. Confuso   1. Claro

- 10.1. **Aplicabilidade** É definida como a medida que um processo descreve as atividades que são necessárias para realizar determinado trabalho em um contexto específico

0. Confuso   1. Claro

**10.2. Efetividade** É definida como a capacidade de um processo de software transformar um conjunto de entradas em um conjunto de saídas desejadas, dessa maneira, entende-se que um processo eficaz deve ajudar a produzir um produto certo

0. Confuso   1. Claro

**10.3. Eficiência** É a capacidade de um processo de software conseguir resultados com o menor tempo e esforço possível.

0. Confuso   1. Claro

**10.4. Compatibilidade** É definida como a capacidade do processo de software de interagir com um ou mais processos ou tecnologias especificadas

0. Confuso   1. Claro

**11. Acurácia** é a capacidade do processo de alcançar resultados considerados corretos ou agradáveis.

0. Confuso   1. Claro

**12. Interoperabilidade** é a capacidade de um processo interagir com outros processos, ou seja, a interoperabilidade indica o quão fácil é acoplar o processo em outros processos.

0. Confuso   1. Claro

**13. Segurança** é definida como a capacidade do processo de software proteger as informações e dados, de forma que pessoas não-autorizadas não podem lê-los ou modificá-los, e que não seja negado acesso às pessoas autorizadas.

0. Confuso   1. Claro

#### Quanto aos aspectos específicos de Evolubilidade

**14. Analisabilidade** é definida como a facilidade de se diagnosticar deficiências e potenciais melhorias no processo

0. Confuso   1. Claro

**15. Modificabilidade** é definida como a facilidade com que um processo de software permite que uma modificação seja implementada.

0. Confuso   1. Claro

**16. Projetabilidade** é definida como a facilidade com que um processo de software permite ser utilizado em um determinado contexto.



0. Confuso   1. Claro

17. **Apoio (suportabilidade)** é definido como a facilidade com que um processo de software apoia determinado contexto, por exemplo: antes de um processo ser implantado, é importante que os recursos, conhecimento e tecnologias necessárias para que um processo seja realizado com sucesso estejam disponíveis.

0. Confuso   1. Claro

### Quanto aos aspectos específicos de representação

18. **Clareza** é definida como a capacidade de um processo de apresentar todos os seus elementos (atividades, fluxogramas, papéis, etc) de forma a facilitar o entendimento do usuário.

0. Confuso   1. Claro

19. **Consistência** é definida como a medida em que um processo de software está descrito de uma forma que seja coerente, em termos de representação, com o próprio processo e com outros processos aos quais ele está relacionado.

0. Confuso   1. Claro

20. **Adequação do formato** é definido como o a notação usada para descrever um processo de software.

0. Confuso   1. Claro

21. **Adequação do meio de comunicação** é definido o quão eficiente é o método utilizado para comunicar uma modelagem de processo de software através da equipe que utiliza o processo e demais envolvidos.

0. Confuso   1. Claro

### Quanto aos aspectos específicos de arquitetura

22. **Tamanho** é definido como a magnitude de um processo de software em termos de número de atividades, papéis e artefatos.

0. Confuso   1. Claro

23. **Complexidade** é definida em termos da quantidade de itens necessários para descrever o processo e suas relações.

0. Confuso   1. Claro

24. **Modularidade** é definida como a capacidade de um processo de software ser autocontido com interfaces externas bem definidas.

0. Confuso   1. Claro

25. **Estabilidade** é definida como o período de tempo em que um processo de software tem sido utilizado em um ambiente de informação, sem modificações significativas na semântica do processo.

0. Confuso   1. Claro

#### Quanto aos aspectos específicos de sintática

26. **Corretude** é definida como a capacidade do modelo de processo utilizar adequadamente linguagem de modelagem na qual ele foi descrito

0. Confuso   1. Claro

#### Quanto aos aspectos gerais do modelo

27. Com relação a todas as características de qualidade de process apresentadas existem características duplicadas? Caso sim, quais?

0. Sim   1. Não

28. Você acrescentaria alguma característica ou subcaracterística ao modelo proposto se sim qual? E onde ela seria colocada?

0. Sim   1. Não

#### Quanto a semelhança das características

29. Com relação as característica de **especificidade**, cuja definição é a capacidade de um processo de software ser adequado para a sua execução em um contexto particular; e **Apoio** que é definido como a facilidade com que um processo de software apoia determinado contexto. Na sua opinião essas características são semelhantes?

Sim   Não

30. Com relação as características de **Clareza** é definida como a capacidade de um processo de apresentar todos os seus elementos de forma a facilitar o entendimento do usuário, e **Apreensibilidade** é definida como a facilidade com que um usuário do processo é capaz de aprender a executar uma atividade do processo. Na sua opinião essas características são semelhantes?

Sim   Não

31. Com relação as característica de **sintática** que é definida como a relação do modelo com a linguagem de modelagem, e **Adequação do formato** é definido como o a notação usada para descrever um processo de software. Na sua opinião essas características são semelhantes?

Sim   Não

# APÊNDICE E – Segunda versão do questionário

## Avaliação de Qualidade de modelagem de processo de software

Prezado Colaborador(a)

Você está sendo convidado(a) a responder um questionário que avaliará as características de qualidade de processo de software em termos de completude, coerência e clareza. Esse questionário faz parte de um trabalho de conclusão de curso em Engenharia de Software na Universidade de Brasília (UnB).

A Figura abaixo traz uma representação gráfica do modelo que será avaliado. O objetivo do modelo é prover um conjunto de características que auxiliem na avaliação da qualidade de um processos de software, tanto em termos da execução quanto em termos da documentação. Esse modelo foi elaborado a partir de um mapeamento sistemático, onde foram identificadas características de qualidade de processo de software. Essas características foram analisadas e refinadas, e com base na estrutura, de características e subcaracterísticas, do modelo de qualidade de produto da norma ISO 9126 foi elaborado o modelo apresentado na Figura abaixo. O questionário está organizado em duas grandes partes. Inicialmente será proposta a avaliação do primeiro nível do modelo, ou seja, as características de qualidade. A segunda parte irá avaliar os demais níveis do modelo, ou seja, as subcaracterísticas de qualidade.

Desde já agradecemos a sua colaboração e colocamo-nos a disposição para qualquer esclarecimento.

\*Obrigatório

### Modelo de qualidade de processo de software



## Características de Qualidade

Você deverá atribuir uma nota de um a cinco para indicar o grau de importância da característica apresentada, conforme a escala a seguir:

- Nada importante (1)
- Pouco importante (2)
- Mediamente importante (3)
- Muito importante (4)
- Extremamente importante (5)

### 1. Usabilidade. \*

Relacionada à facilidade de interpretar e executar uma modelagem de processo, sob condições específicas, para alcançar os resultados desejados. Pode ser classificada como: acessibilidade, compreensibilidade, apreensibilidade, adaptabilidade, operabilidade, atratividade, semântica e execução

1 2 3 4 5

Nada importante      Extremamente importante

### 2. Funcionalidade. \*

Relacionada à capacidade do processo em descrever tudo o que é realizado ou que deveria ser realizado pela equipe ou organização, ou seja as funções esperadas do processo. Essas funções são: adequação, segurança e interação com outros processos

1 2 3 4 5

Nada importante      Extremamente importante

### 3. Evolubilidade. \*

Relacionada à facilidade com que um processo de software está disponível para ser avaliado e modificado com o propósito de manutenção e melhoria de processo. Evolubilidade pode ser classificada em três subcaracterísticas: analisabilidade, modificabilidade e apoio

1 2 3 4 5

Nada importante      Extremamente importante

### 4. Sintática. \*

É definida como a relação do modelo com a linguagem de modelagem. Descreve as relações entre as construções de linguagem, sem considerar o seu significado. Pode ser classificado como correteude e adequação do formato

1 2 3 4 5

Nada importante      Extremamente importante

**5. Arquitetura. \***

Relacionada com os elementos de um processo de software que dizem respeito à sua organização. Influencia significativamente na forma como o processo é percebido. Arquitetura pode ser classificado em quatro subcaracterísticas: tamanho,complexidade, modularidade e escalabilidade.

1 2 3 4 5

Nada importante      Extremamente importante

**6. Representação. \***

Relacionada à capacidade de um processo ser documentado e comunicado a todas as partes interessadas. Representação pode ser classificada como consistência,.

1 2 3 4 5

Nada importante      Extremamente importante

## Detalhamento das características de qualidade

Como pode ser visto na Figura abaixo, cada uma das características de qualidade propostas são detalhadas em termos de subcaracterísticas de qualidade e, em alguns casos, as subcaracterísticas de qualidade são detalhadas em subsubcaracterísticas de qualidade. Nesta parte do questionário será avaliada a clareza das definições propostas quanto a coerência, ou seja, se de fato as subcaracterísticas deveriam estar onde estão no modelo.

Você deverá atribuir uma nota de um a cinco para indicar o grau de importância da característica apresentada, conforme a escala a seguir:

- Nada importante (1)
- Pouco importante (2)
- Mediamente importante (3)
- Muito importante (4)
- Extremamente importante (5)

## Modelo de qualidade de processo de software



**10. Adaptabilidade é a capacidade do processo se adaptar em diferentes situações sem perder suas informações ou sua estrutura, facilitando a utilização do processo pelo usuário, por exemplo quando uma organização busca criar um conjunto de padrões de processo para uso em múltiplos projetos. A adaptabilidade pode ser classificada em três subclassificações:**

(1) Ajustabilidade trata da facilidade de um processo padrão ser adaptado para formar processos específicos em determinado contexto, de diferentes tamanhos. (2) Flexibilidade trata da possibilidade de modificação, em tempo de execução, de algum aspecto do processo com o objetivo de melhorar o desempenho dele. O processo deve ser flexível, mas uma alta taxa de flexibilidade pode inutilizar a documentação de processo provida. (3) Controlabilidade é definida como a facilidade com que gestor de processos é capaz de aplicar ações preventivas e corretivas para um processo em execução e continuar mantendo um nível de desempenho especificado

1 2 3 4 5

Nada importante      Extremamente importante

**11. Operabilidade é definida como a capacidade do processo ser utilizado e controlado.**

1 2 3 4 5

Nada importante      Extremamente importante

**12. Atratividade é definida como a capacidade do processo de envolver os usuários com seus projetos de modelos e/ou documentos, por exemplo: interface de usuário.**

1 2 3 4 5

Nada importante      Extremamente importante

**13. Execução é definida como o uso do processo de software dentro de um ambiente específico. Execução pode ser classificado em seis subcaracterísticas:**

(1) Custo é definido como a quantidade de esforço, tempo e dinheiro necessários para executar um processo de software dentro de um contexto específico. (1) Mensurabilidade é definida como a capacidade da execução de um processo de software oferecer medidas relacionadas com seu desempenho. (3) Capacidade de automação é definida como a capacidade de um processo de software ser executado usando recursos computacionais nas atividades do processo. (4) Rigor é definido como a medida que um processo de software requer para que suas atividades sejam realizadas de maneira controlada e disciplinada. (5) Tolerância a falhas é definida como o grau em que um processo, uma vez que uma intrusão ocorra, ou continua a produzir resultados precisos ou reconhece a imprecisão de seus resultados e inicia uma ação corretiva. (6) Gerenciabilidade definida como a facilidade de gerenciar as mudanças em um processo, ou seja, a facilidade com que uma mudança em um processo pode ser prevista, monitorada e controlada para alcançar os resultados desejados, em um contexto específico. Gerenciabilidade pode ser classificada como: Previsibilidade que é definida como a facilidade com que um gestor de processo é capaz de determinar o tempo, recursos, riscos e limitações associadas com a execução do processo. E, monitorabilidade é definida como a facilidade com que um gerente de processo é capaz de determinar o status e os resultados de um processo executado.

1 2 3 4 5

Nada importante      Extremamente importante

**14. Semântica é definida como a capacidade de relacionar o modelo com o domínio, considerando as relações entre as instruções e seus significados, ou seja, relacionam o escopo do modelo com a intenção de suas atividades. Semântica pode ser classificada em quatro subcaracterísticas**

(1) Conformidade é definida como a capacidade com a qual o processo de software está de acordo com padrões e modelos de processos reconhecidos na indústria. (2) Capacidade de resposta é definida como a velocidade de resposta com que um processo de software é capaz de incorporar um feedback nos envolvidos com o processo. (3) Validade está relacionada a adequação do processo ao domínio e às características do time para o qual foi documentado. (4) Completude é capacidade do processo de software ou modelo conter todas as declarações sobre o domínio, sendo todas elas corretas e relevantes.

1 2 3 4 5

Nada importante      Extremamente importante

## Quanto aos aspectos específicos de funcionalidade

**15. Adequabilidade é definida como a capacidade do processo para produzir os resultados desejados de forma apropriada em um contexto específico, ou seja, a capacidade das atividades do processo de serem completas e adequadas para a execução de tarefas. Adequabilidade por ser classificada em:**

(1) Efetividade é definida como a capacidade de um processo de software transformar um conjunto de entradas em um conjunto de saídas desejadas, dessa maneira, entende-se que um processo eficaz deve ajudar a produzir um produto certo. (2) Eficiência é a capacidade de um processo de software conseguir resultados com o menor tempo e esforço possível. (3) Compatibilidade é definida como a capacidade do processo de software de interagir com um ou mais processos ou tecnologias especificadas.

1 2 3 4 5

Nada importante      Extremamente importante

**16. Interoperabilidade é definida como a capacidade de um processo interagir com outros processos, ou seja, a interoperabilidade indica o quão fácil é acoplar o processo em outros processos.**

1 2 3 4 5

Nada importante      Extremamente importante

**17. Segurança é definida como a capacidade do processo de software proteger as informações e dados, de forma que pessoas não-autorizadas não podem lê-los ou modificá-los, e que não seja negado acesso às pessoas autorizadas.**

1 2 3 4 5

Nada importante      Extremamente importante

## Quanto aos aspectos específicos de Evolubilidade

18. Analisabilidade é definida como a facilidade de se diagnosticar deficiências e potenciais melhorias no processo

1 2 3 4 5

Nada importante      Extremamente importante

19. Modificabilidade é definida como a facilidade com que um processo de software permite que uma modificação seja implementada.

1 2 3 4 5

Nada importante      Extremamente importante

20. Apoio é definido como a facilidade com que um processo de software apoia a realização das atividades necessárias para sua utilização, por exemplo: antes de um processo ser implantado, é importante que os recursos, conhecimento e tecnologias necessários para que um processo seja realizado com sucesso estejam disponíveis

1 2 3 4 5

Nada importante      Extremamente importante

## Quanto aos aspectos específicos de representação

21. Consistência é definida como a medida em que um processo de software está descrito de uma forma que seja coerente, em termos de representação, com o próprio processo e com outros processos aos quais ele está relacionado.

1 2 3 4 5

Nada importante      Extremamente importante

## Quanto aos aspectos específicos de arquitetura

22. Tamanho é definido como a magnitude de um processo de software em termos de número de atividades, papéis e artefatos

1 2 3 4 5

Nada importante      Extremamente importante



23. Complexidade é definida em termos da quantidade de relações estabelecidas entre os itens necessários para descrever o processo, itens como, partes constituintes do processo, fluxo de controle e interface.

1 2 3 4 5

Nada importante      Extremamente importante

24. Modularidade é definida como a capacidade de um processo de software ser autocontido com interfaces externas bem definidas.

1 2 3 4 5

Nada importante      Extremamente importante

25. Escalabilidade é definida como o período de tempo em que um processo de software tem sido utilizado em um ambiente de informação, sem modificações significativas na semântica do processo

1 2 3 4 5

Nada importante      Extremamente importante

## Quanto aos aspectos específicos de sintática

26. Corretude é definida como a capacidade do modelo de processo utilizar adequadamente linguagem de modelagem na qual ele foi descrito

1 2 3 4 5

Nada importante      Extremamente importante

27. Adequação do formato é definido como o a notação usada para descrever um processo de software.

1 2 3 4 5

Nada importante      Extremamente importante

## Dados do entrevistado

Sexo \*

- Feminino  
 Masculino

**Qual sua localização ? \***

Informe sua UF

**Qual o tempo de atuação com MPS (melhoria de processo de software)?**

Informe em meses

**Qual a sua experiência com modelagem de processos ?**

Informe em meses

**Com quais modelos de qualidade de processo você tem trabalhado?** CMMI [MPS.BR](#) Outro: **Quais ferramentas ou linguagens de modelagem de processos você tem utilizado?** Bizagi EPF Composer Diagrama de Atividades da UML Outro: **Qual(is) a(s) aplicação(ões) dessa experiência com MPS ou modelagem de process?** Setor público Iniciativa privada Academia

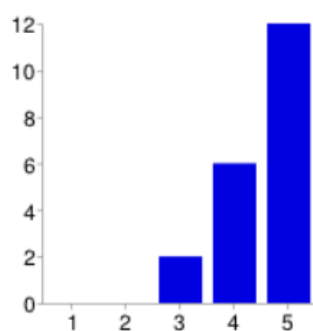
# APÊNDICE F – Resultados do Questionário

O relatório apresentado neste apêndice é exatamente semelhante ao que é disponibilizado pela ferramenta adotada para coletar os dados do questionário. Uma análise destes resultados é apresentado na Seção 4.2.

## Modelo de qualidade de processo de software

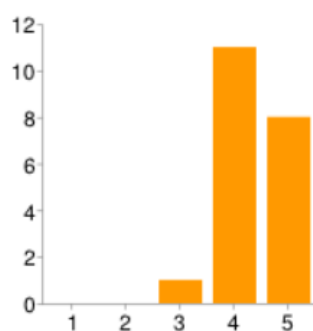
### Características de Qualidade

#### 1. Usabilidade.



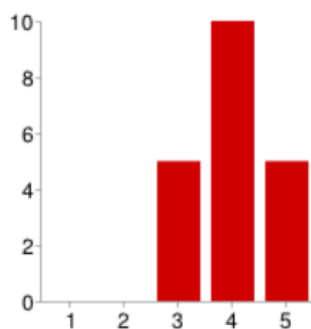
1	0	0%
2	0	0%
3	2	10%
4	6	30%
5	12	60%

#### 2. Funcionalidade.



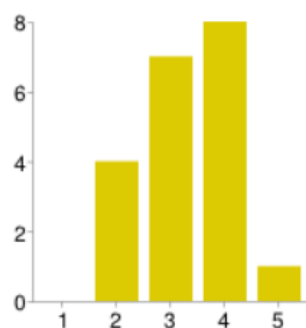
1	0	0%
2	0	0%
3	1	5%
4	11	55%
5	8	40%

**3. Evolubilidade.**



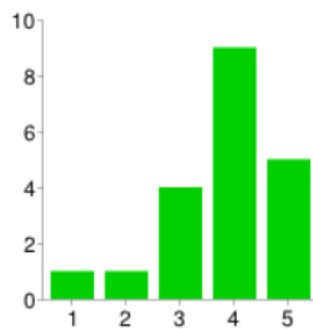
1	0	0%
2	0	0%
3	5	25%
4	10	50%
5	5	25%

**4. Sintática.**



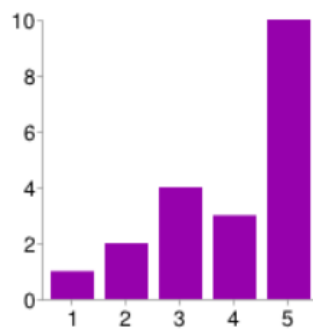
1	0	0%
2	4	20%
3	7	35%
4	8	40%
5	1	5%

**5. Arquitetura.**



1	1	5%
2	1	5%
3	4	20%
4	9	45%
5	5	25%

**6. Representação.**

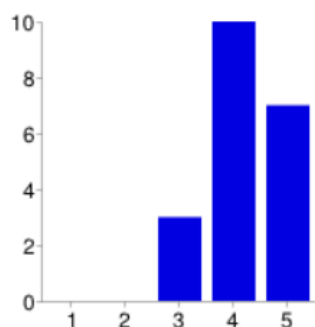


1	1	5%
2	2	10%
3	4	20%
4	3	15%
5	10	50%

## Detalhamento das características de qualidade

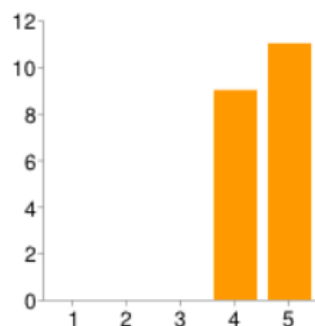
### Quanto aos aspectos específicos de usabilidade

**7. Acessibilidade é definida como a facilidade de encontrar informações sobre um processo de engenharia de software, acessibilidade esta dividida em três subcaracterísticas:**



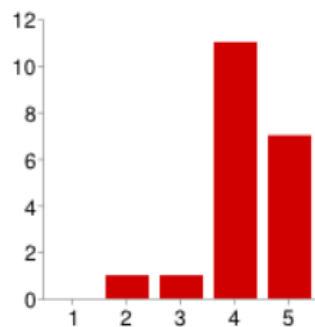
1	0	0%
2	0	0%
3	3	15%
4	10	50%
5	7	35%

**8. Compreensibilidade é definida como a facilidade com a qual o usuário de um processo é capaz de compreender se um processo de engenharia de software é relevante e como ele pode ser utilizado para alcançar os resultados desejados.**



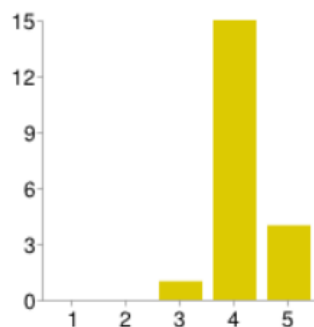
1	0	0%
2	0	0%
3	0	0%
4	9	45%
5	11	55%

**9. Apreensibilidade é definida como a capacidade de um processo de apresentar todos os seus elementos (atividades, fluxogramas, papéis, etc) de forma a facilitar o entendimento do usuário, para que esse tenha a capacidade de executar uma atividade do processo.**



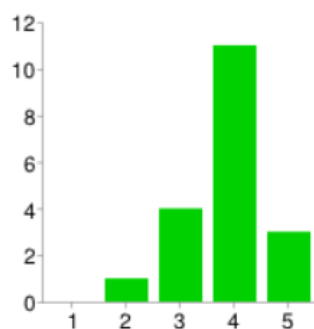
1	0	0%
2	1	5%
3	1	5%
4	11	55%
5	7	35%

**10. Adaptabilidade é a capacidade do processo se adaptar em diferentes situações sem perder suas informações ou sua estrutura, facilitando a utilização do processo pelo usuário, por exemplo quando uma organização busca criar um conjunto de padrões de processo para uso em múltiplos projetos. A adaptabilidade pode ser classificada em três subclassificações:**



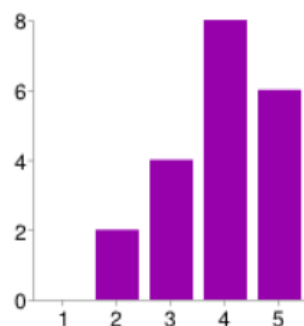
1	0	0%
2	0	0%
3	1	5%
4	15	75%
5	4	20%

**11. Operabilidade é definida como a capacidade do processo ser utilizado e controlado.**



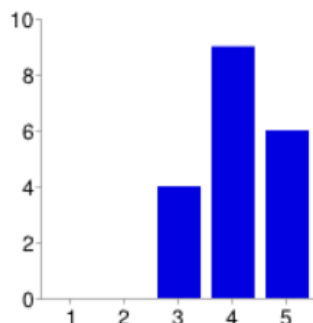
1	0	0%
2	1	5%
3	4	20%
4	11	55%
5	3	15%

**12. Atratividade é definida como a capacidade do processo de envolver os usuários com seus projetos de modelos e/ou documentos, por exemplo: interface de usuário.**



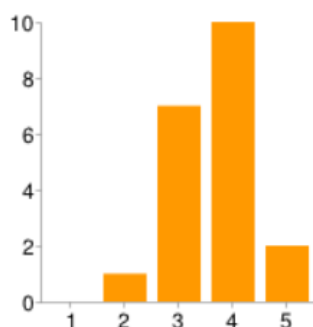
1	0	0%
2	2	10%
3	4	20%
4	8	40%
5	6	30%

**13. Execução é definida como o uso do processo de software dentro de um ambiente específico. Execução pode ser classificado em seis subcaracterísticas:**



1	0	0%
2	0	0%
3	4	20%
4	9	45%
5	6	30%

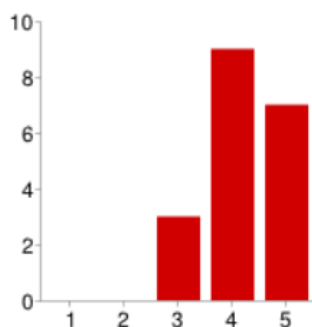
**14. Semântica é definida como a capacidade de relacionar o modelo com o domínio, considerando as relações entre as instruções e seus significados, ou seja, relacionam o escopo do modelo com a intenção de suas atividades. Semântica pode ser classificada em quatro subcaracterísticas**



1	0	0%
2	1	5%
3	7	35%
4	10	50%
5	2	10%

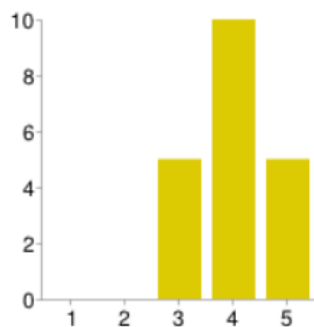
### Quanto aos aspectos específicos de funcionalidade

**15. Adequabilidade é definida como a capacidade do processo para produzir os resultados desejados de forma apropriada em um contexto específico, ou seja, a capacidade das atividades do processo de serem completas e adequadas para a execução de tarefas. Adequabilidade por ser classificada em:**



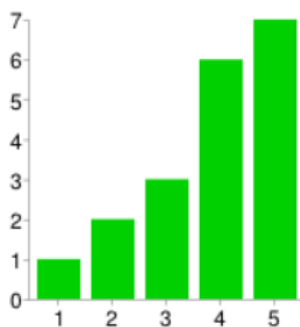
1	0	0%
2	0	0%
3	3	15%
4	9	45%
5	7	35%

**16. Interoperabilidade é definida como a capacidade de um processo interagir com outros processos,ou seja, a interoperabilidade indica o quão fácil é acoplar o processo em outros processos.**



1	0	0%
2	0	0%
3	5	25%
4	10	50%
5	5	25%

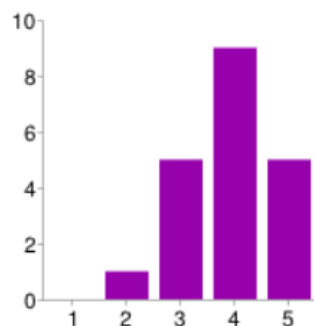
**17. Segurança é definida como a capacidade do processo de software proteger as informações e dados, de forma que pessoas não-autorizadas não podem lê-los ou modificá-los,e que não seja negado acesso às pessoas autorizadas.**



1	1	5%
2	2	10%
3	3	15%
4	6	30%
5	7	35%

### Quanto aos aspectos específicos de Evolubilidade

**18. Analisabilidade é definida como a facilidade de se diagnosticar deficiências e potenciais melhorias no processo**

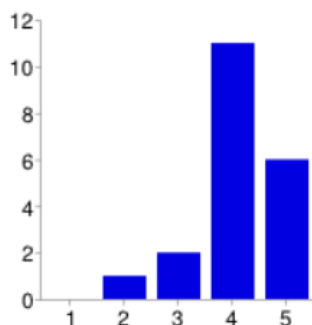


1	0	0%
2	1	5%
3	5	25%
4	9	45%
5	5	25%



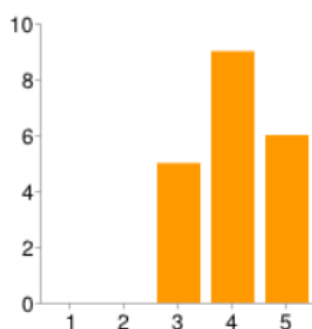
### Modelo de qualidade de processo de software

**19. Modificabilidade é definida como a facilidade com que um processo de software permite que uma modificação seja implementada.**



1	0	0%
2	1	5%
3	2	10%
4	11	55%
5	6	30%

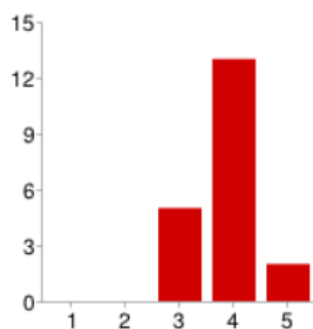
**20. Apoio é definido como a facilidade com que um processo de software apoia a realização das atividades necessárias para sua utilização, por exemplo: antes de um processo ser implantado, é importante que os recursos, conhecimento e tecnologias necessários para que um processo seja realizado com sucesso estejam disponíveis**



1	0	0%
2	0	0%
3	5	25%
4	9	45%
5	6	30%

### Quanto aos aspectos específicos de representação

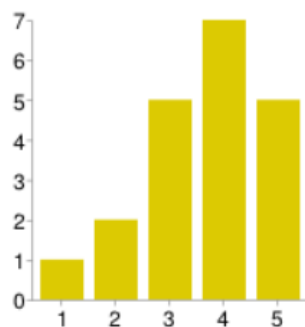
**21. Consistência é definida como a medida em que um processo de software está descrito de uma forma que seja coerente, em termos de representação, com o próprio processo e com outros processos aos quais ele está relacionado.**



1	0	0%
2	0	0%
3	5	25%
4	13	65%
5	2	10%

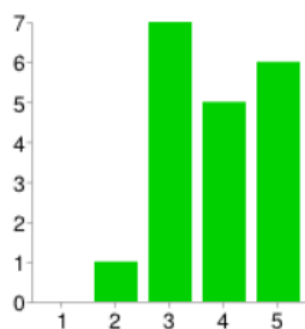
### Quanto aos aspectos específicos de arquitetura

**22. Tamanho é definido como a magnitude de um processo de software em termos de número de atividades, papéis e artefatos**



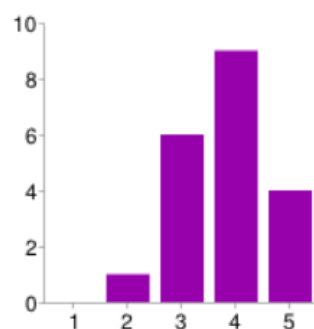
1	1	5%
2	2	10%
3	5	25%
4	7	35%
5	5	25%

**23. Complexidade é definida em termos da quantidade de relações estabelecidas entre os itens necessários para descrever o processo, itens como, partes constituintes do processo, fluxo de controle e interface.**



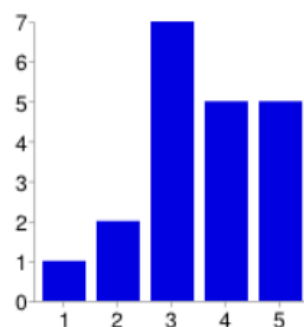
1	0	0%
2	1	5%
3	7	35%
4	5	25%
5	6	30%

**24. Modularidade é definida como a capacidade de um processo de software ser autocontido com interfaces externas bem definidas.**



1	0	0%
2	1	5%
3	6	30%
4	9	45%
5	4	20%

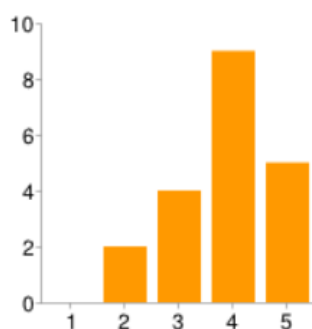
**25. Escalabilidade é definida como o período de tempo em que um processo de software tem sido utilizado em um ambiente de informação, sem modificações significativas na semântica do processo**



1	1	5%
2	2	10%
3	7	35%
4	5	25%
5	5	25%

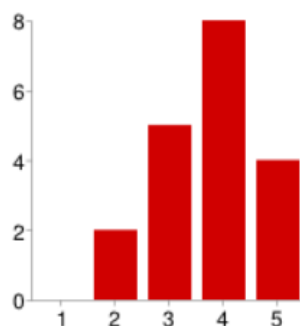
### Quanto aos aspectos específicos de sintática

**26. Corretude é definida como a capacidade do modelo de processo utilizar adequadamente linguagem de modelagem na qual ele foi descrito**



1	0	0%
2	2	10%
3	4	20%
4	9	45%
5	5	25%

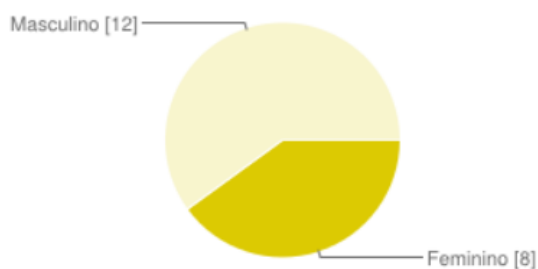
**27. Adequação do formato é definido como o a notação usada para descrever um processo de software.**



1	0	0%
2	2	10%
3	5	25%
4	8	40%
5	4	20%

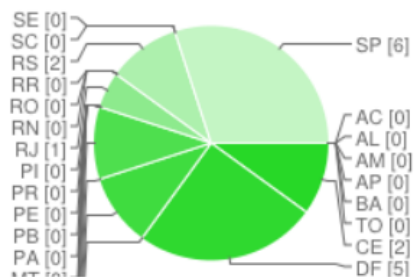
## Dados do entrevistado

### Sexo



Feminino	8	40%
Masculino	12	60%

### Qual sua localização ?



AC	0	0%
AL	0	0%
AM	0	0%
AP	0	0%
BA	0	0%
CE	2	10%
DF	5	25%
ES	0	0%
GO	2	10%
MA	0	0%
MG	2	10%
MS	0	0%
MT	0	0%
PA	0	0%
PB	0	0%
PE	0	0%
PI	0	0%
PR	0	0%
RJ	1	5%
RN	0	0%
RO	0	0%
RR	0	0%
RS	2	10%
SC	0	0%
SE	0	0%
SP	6	30%
TO	0	0%

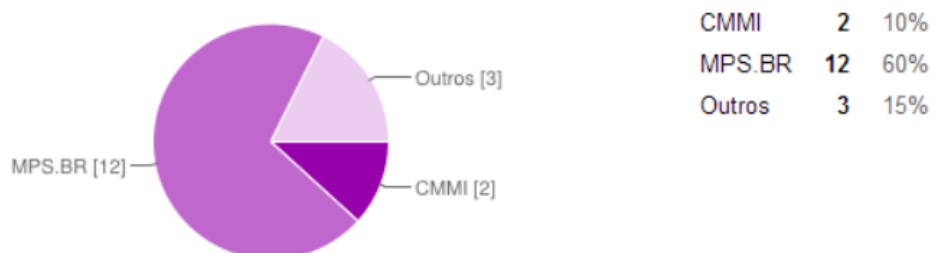
**Qual o tempo de atuação com MPS (melhoria de processo de software)?**



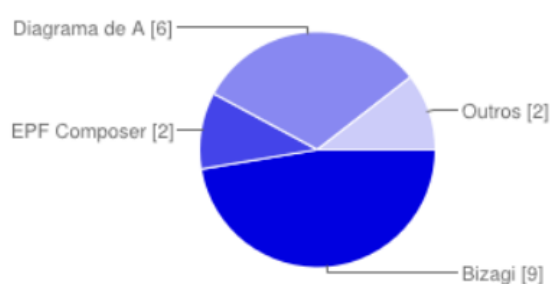
**Qual a sua experiência com modelagem de processos ?**



**Com quais modelos de qualidade de processo você tem trabalhado?**

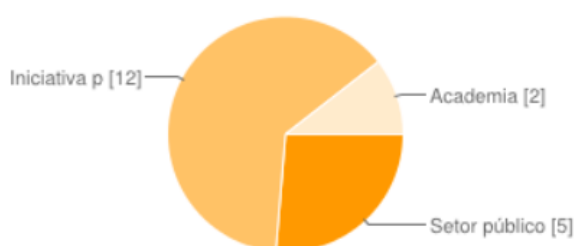


**Quais ferramentas ou linguagens de modelagem de processos você tem utilizado?**



Bizagi	9	45%
EPF Composer	2	10%
Diagrama de Atividades da UML	6	30%
Outros	2	10%

**Qual(is) a(s) aplicação(ões) dessa experiência com MPS ou modelagem de process?**



Setor público	5	25%
Iniciativa privada	12	60%
Academia	2	10%