

**UNIVERSIDADE FEDERAL DE SÃO CARLOS**  
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

**ELABORAÇÃO DE UM PROCESSO DE TESTE  
COM BASE EM UM ESTUDO DE CASO REAL**

**KAMILLA GOMES CAMARGO**

**ORIENTADORA: PROFA. DRA. SANDRA C. P. F. FABBRI**

São Carlos – SP  
Outubro/2012

**UNIVERSIDADE FEDERAL DE SÃO CARLOS**

CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

**ELABORAÇÃO DE UM PROCESSO DE TESTE  
COM BASE EM UM ESTUDO DE CASO REAL**

**KAMILLA GOMES CAMARGO**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de São Carlos, como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação, área de concentração: Engenharia de Software

Orientadora: Profa. Dra. Sandra C. P. F. Fabbri

São Carlos – SP

Outubro/2012

**Ficha catalográfica elaborada pelo DePT da  
Biblioteca Comunitária da UFSCar**

C172ep Camargo, Kamilla Gomes.  
Elaboração de um processo de teste com base em um estudo de caso real / Kamilla Gomes Camargo. -- São Carlos : UFSCar, 2013.  
118 f.

Dissertação (Mestrado) -- Universidade Federal de São Carlos, 2012.

1. Engenharia de software. 2. Software - testes. 3. Melhoria de processo. 4. TMMi. 5. Pequenas e médias empresas. I. Título.

CDD: 005.1 (20<sup>a</sup>)

# Universidade Federal de São Carlos

Centro de Ciências Exatas e de Tecnologia  
Programa de Pós-Graduação em Ciência da Computação

## “Elaboração de um Processo de Teste com Base em um Estudo de Caso Real”

Kamilla Gomes Camargo

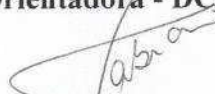
Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de São Carlos, como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação

Membros da Banca:



---

Profa. Dra. Sandra Camargo Pinto Ferraz Fabbri  
(Orientadora - DC/UFSCar)



---

Prof. Dr. Fabiano Cutigi Ferrari  
(DC/UFSCar)



---

Prof. Dr. Daniel Lucrédio  
(DC/UFSCar)



---

Prof. Dr. Auri Marcelo Rizzo Vincenzi  
(UFG)

São Carlos  
Dezembro/2012





## AGRADECIMENTOS

Ao meu co-orientador Prof. Dr. Fabiano Ferrari, por aceitar participar do projeto, pelo apoio constante ao longo deste trabalho, e pelo incentivo à busca constante pelo conhecimento, característica essencial para a pesquisa.

À minha, mais que orientadora, Prof<sup>a</sup>. Dr<sup>a</sup>. Sandra Fabbri pela oportunidade de ser sua aluna, pela excelente orientação, pela confiança, pela amizade e pelo exemplo de ética, característica essencial para um pesquisador.

À empresa que colaborou imensamente com este projeto, permitindo assim sua realização.

À Erika, pelo trabalho e pelas diversas ajudas ao longo da realização deste.

Ao Julián, que além de me apoiar e incentivar incondicionalmente, me suportou nos momentos de estresse, me ajudou diversas vezes e ainda teve coragem de se tornar meu marido! Por todo esse amor, só tenho a agradecer e retribuir...

À minha família, que mesmo de longe me deu o apoio necessário. À minha mãe Helena, que sempre me passa confiança nos momentos mais pertinentes. À Lud, minha irmã, que além de me dar uma princesa chamada Luana, me contagia com sua alegria mesmo nos momentos difíceis. Ao meu pai, que à sua maneira, também me apoiou nesta jornada.

À minha segunda família, a família “Lapesiana”, que tornou essa jornada muito mais divertida! Em especial à Elis e Juciara, pela amizade e ótimos dias de convivência. André e Fábio, pela amizade e por tornarem o ambiente muito mais alegre e descontraído, e a todos os amigos do LaPES pela amizade e apoio.

Aos que me acolheram em São Carlos nos momentos em que precisei Matheus e Bruno, em especial agradeço à Elis, que me recebeu em sua casa muitas e muitas vezes, e também à Aisla, André, Lelê e Pepe, que me aturaram durante uma semana.

À Cristina, secretária do PPGCC, cujo apoio foi essencial em diversos momentos.

À CAPES pelo apoio financeiro.





## RESUMO

**Contexto:** Pequenas empresas encontram algumas barreiras para implementar ou melhorar seus processos de teste devido à falta de profissionais especializados em teste, falta de recursos e também devido à dificuldade em adaptar modelos de maturidade em teste para o contexto específico da empresa. **Objetivo:** O objetivo deste trabalho é apresentar uma estratégia para que as empresas possam definir um processo de teste, ou melhorar o processo de teste existente. **Metodologia:** A estratégia para elaboração de um processo de teste para pequenas empresas foi extraída a partir da realização de um estudo prático em uma pequena empresa desenvolvedora de software. Nesse estudo estabeleceu-se como base o processo de teste existente na empresa e definiu-se um novo processo de teste, tendo como referência o TMMi e um *survey*, que coletou a opinião de profissionais que atuam na área de teste. **Resultados:** Os três resultados obtidos neste trabalho são: (i) O modelo de um processo reduzido, resultante do *survey*; (ii) um novo processo de teste que foi proposto para a empresa; e (iii) uma estratégia para elaboração de processo de teste extraída de um estudo prático, com base no TMMi e na opinião de profissionais de teste. **Conclusões:** A sequência de passos que foi realizada no estudo prático permitiu chegar a uma proposta de processo de teste que leva em consideração o TMMi, sendo portanto, uma proposta de processo com foco em qualidade. A generalização dos passos executados resultam em uma estratégia que pode ser aplicada por empresas para avaliar e melhorar seu processo de teste sempre que necessário.

**Palavras-chave:** teste de software, melhoria de processo, TMMi, definição de processo, empresas de pequeno porte



# ABSTRACT

**Background:** Small companies face barriers to either implement or enhance their software testing processes due to the lack of experts in testing, the lack of resources and difficulties to adapt testing-related maturity models to a particular context. **Aim:** The goal of this research is devising a strategy that can be applied by software companies to the definition of testing processes, or even to the enhancement of the current company's process. **Methodology:** The strategy for the definition of testing processes has been built upon the results of a case study of a small software development company. In such case study, we have characterized the company's current testing process and, based on it, we have devised a new process that relies both on the TMMi reference model and on the results of a survey that has gathered the opinion of professionals in software testing. **Results:** The three main contributions of this research are: (i) a reduced test process model as a result of the mentioned survey; (ii) a newly devised testing process for the case company; and (iii) a strategy for the definition of testing processes devised from the case study, which is based on the TMMi reference model and on the opinion of software testing professionals. **Conclusion:** The sequence of steps performed in the case study enabled us to establish a testing process that takes into consideration the TMMi model and other company's particularities, thus resulting in a process whose focus is on quality. Performing such steps enables the company to evaluate and enhance its testing process whenever required.

**Keywords:** software testing, process improvement, TMMi, process definition, small companies



## LISTA DE FIGURAS

1.1	Metodologia utilizada para desenvolvimento deste trabalho. . . . .	25
2.1	Níveis de maturidade e áreas de processo do TMMi [adaptada de TMMi Foundation (2010)]. . . . .	34
2.2	TMMi structure and components (TMMi Foundation, 2010). . . . .	35
2.3	Níveis de maturidade do MPT.Br (Adaptado de MPT.Br (2011)). . . . .	37
2.4	Parte do KITMap mostrando as etapas do processo genérico de teste (Adaptado de Höhn (2011).) . . . . .	39
2.5	Parte do KITMap mostrando a fase de projeto de casos de teste (HÖHN, 2011). . . . .	40
2.6	Avaliação por Objetivos na KITTool (HÖHN, 2011). . . . .	41
2.7	Avaliação por práticas na KITTool (HÖHN, 2011). . . . .	42
3.1	Exemplo da estrutura de questão utilizada no <i>survey</i> para o objetivo <i>Realizar avaliação de risco do produto</i> . . . . .	48
3.2	Exemplo da estrutura de questão utilizada no <i>survey</i> para o objetivo <i>Desenvolver um plano de teste</i> . . . . .	48
3.3	Parte do formulário do <i>survey</i> com algumas das questões de perfil. . . . .	49
3.4	Distribuição de frequência das escalas obtidas nas respostas ao questionário. . . . .	51
3.5	Perfis obtidos na amostra . . . . .	53
3.6	Diagrama de Venn para os quatro perfis descritos apresentando as intersecções dos resultados e práticas que foram identificadas como mandatórias pela maioria dos respondentes. Itens destacados em cinza indicam as práticas que compõem o processo reduzido apresentado na Seção 3.4 . . . . .	56

3.7	Práticas do TMMi relacionadas à fase de Planejamento do projeto de testes (adaptado de Höhn (2011)). . . . .	61
3.8	Práticas do TMMi relacionadas à fase de Projeto de Casos de Teste (adaptado de Höhn (2011)). . . . .	63
3.9	Práticas do TMMi relacionadas à fase de Configuração de Dados e de Ambiente de Teste (adaptado de Höhn (2011)). . . . .	64
3.10	Práticas do TMMi relacionadas à fase de Execução e Análise (adaptado de Höhn (2011)). . . . .	65
3.11	Práticas do TMMi relacionadas à fase de Monitoramento e controle (adaptado de Höhn (2011)). . . . .	67
4.1	Mapa mental do processo atual da empresa com as práticas do TMMi associadas. . . . .	74
4.2	Legenda apresentada pela KITTool para o diagnóstico realizado “por prática”. . . . .	78
4.3	Resultado apresentado pela ferramenta KITTool para o diagnóstico realizado “por prática” das áreas de processo “Política e Estratégia de Teste” e “Planejamento de Teste”. . . . .	79
4.4	Resultado apresentado pela ferramenta KITTool para o diagnóstico realizado “por prática” das áreas de processo “Monitoramento e Controle” e “Projeto e Execução de Teste”. . . . .	80
4.5	Resultado apresentado pela ferramenta KITTool para o diagnóstico realizado “por prática” das áreas de processo “Ambiente de Teste” e “Teste Não Funcional”. . . . .	81
4.6	Processo de teste da empresa remodelado, representando o Processo de Teste Real. . . . .	86
4.7	Processo Reduzido (PR), apresentado no Capítulo 3, na forma de mapa mental para melhor visualização. . . . .	89
4.8	Mapa mental do processo real da empresa. As folhas em azul coincidem com as práticas do processo reduzido, apresentado no Capítulo 3, que possuem relação com a respectiva atividade do processo real. . . . .	91

4.9	Processo atual da empresa com novas atividades e tarefas incluídas para que atenda ao conjunto mínimo de práticas do TMMi consideradas muito importantes conforme resultado do survey. . . . .	95
5.1	Fluxograma da estratégia para desenvolvimento de processo de teste. . . . .	102





## LISTA DE TABELAS

3.1	Escala de notas de acordo com a importância . . . . .	47
3.2	Tabela com quantidade de cada nota, da escala de 1 a 4, atribuída à prática <i>Identificar e priorizar condições de teste</i> . . . . .	52
3.3	Práticas que estão presentes no diagrama de Venn da Figura 3.6. . . . .	57
4.1	Resultados obtidos no diagnóstico realizado com a KITTool. . . . .	82
4.2	Práticas encontradas no diagnóstico com KITTool e na avaliação com mapa mental. . . . .	83
A.1	Dependências entre práticas do TMMi identificadas por Höhn (2011), sendo que N representa dependência <i>necessária</i> e A representa dependência de <i>alinhamento</i> . . . . .	118



# SUMÁRIO

<b>CAPÍTULO 1 – INTRODUÇÃO</b>	<b>21</b>
1.1 Contexto e Motivação . . . . .	21
1.2 Objetivo . . . . .	23
1.3 Metodologia . . . . .	23
1.4 Organização do trabalho . . . . .	24
<b>CAPÍTULO 2 – TESTE DE SOFTWARE</b>	<b>27</b>
2.1 Considerações Iniciais . . . . .	27
2.2 Processo de Teste de Software . . . . .	28
2.3 Modelos de Referência . . . . .	33
2.3.1 <i>Test Maturity Model integration</i> – TMMi . . . . .	33
2.3.2 Modelo de Melhoria do Processo de Teste Brasileiro – MPT.Br . . . . .	36
2.4 Base de Conhecimento em Teste . . . . .	38
2.5 Considerações Finais . . . . .	42
<b>CAPÍTULO 3 – SURVEY</b>	<b>45</b>
3.1 Considerações Iniciais . . . . .	45
3.2 Descrição do Survey . . . . .	46
3.2.1 Projeto do Survey . . . . .	46
3.2.2 Amostra Obtida . . . . .	50
3.2.3 Forma de Análise dos Dados . . . . .	50

3.3	Resultados . . . . .	52
3.3.1	Determinação dos Perfis com Base no Nível de Conhecimento dos Respondentes . . . . .	53
3.3.2	Caracterização da Importância das Práticas do TMMi . . . . .	55
3.4	Discussão . . . . .	58
3.4.1	Planejamento . . . . .	59
3.4.2	Projeto de Casos de Teste . . . . .	62
3.4.3	Configuração de Dados e de Ambiente de Teste . . . . .	63
3.4.4	Execução e Análise . . . . .	64
3.4.5	Monitoramento e Controle . . . . .	66
3.5	Riscos à Validade . . . . .	67
3.6	Considerações Finais . . . . .	68

**CAPÍTULO 4 – ESTUDO DE CASO** **71**

4.1	Considerações Iniciais . . . . .	71
4.2	Caracterização da Empresa . . . . .	72
4.3	Diagnóstico do processo de teste da empresa . . . . .	73
4.3.1	Avaliação por Mapa Mental . . . . .	73
4.3.2	Diagnóstico KITTool . . . . .	77
4.3.3	Comparação de resultados . . . . .	82
4.4	Processo Real . . . . .	84
4.5	Análise Crítica . . . . .	88
4.6	Considerações Finais . . . . .	96

**CAPÍTULO 5 – ESTRATÉGIA** **99**

5.1	Considerações Iniciais . . . . .	99
5.2	Definição da Estratégia . . . . .	100
5.3	Considerações Finais . . . . .	106

<b>CAPÍTULO 6 – CONCLUSÃO</b>	<b>107</b>
6.1 Contribuições . . . . .	108
6.2 Limitações . . . . .	108
6.3 Trabalhos Futuros . . . . .	109
<b>REFERÊNCIAS BIBLIOGRÁFICAS</b>	<b>111</b>
<b>GLOSSÁRIO</b>	<b>115</b>
<b>ANEXO A – TABELA DE DEPENDÊNCIAS</b>	<b>117</b>



# Capítulo 1

## INTRODUÇÃO

---

---

Este capítulo apresenta o contexto e a motivação que levaram à elaboração de uma proposta de estratégia para criação de um processo de teste para pequenas empresas.

### 1.1 Contexto e Motivação

Com uso muito difundido, o software desempenha um papel importante no cotidiano das pessoas e sua confiabilidade não pode ser ignorada (CAO; DONG; LIU, 2010). A maneira como esse software é desenvolvido influencia na sua confiabilidade e qualidade. As atividades de garantia de qualidade devem acompanhar todo o processo de desenvolvimento, o que contribui para a melhoria da qualidade do produto final e, conseqüentemente, o torna um produto mais confiável. Uma das principais atividades de qualidade é o teste de software que, quando bem executado, pode garantir uma menor quantidade de defeitos no produto final. No entanto, para que isso aconteça, é necessário que haja um processo bem definido capaz de acompanhar todo o desenvolvimento do software.

Colocar a eficiência e eficácia de seus processos em foco torna a empresa mais competitiva, pois ela pode oferecer produtos de software e serviços conforme padrões internacionais de qualidade (MPS.BR, 2011b). Contudo, mesmo sabendo da importância do teste de software, muitas empresas, principalmente as de pequeno porte, tem dificuldade em delinear um processo de teste e adequá-lo à sua realidade. Rodrigues, Pinheiro e Albuquerque (2010) afirmam que entre as principais dificuldades encontradas por pequenas empresas estão:

- a falta de profissionais especializados em teste de software;
- a falta de recursos;

- a falta de conhecimento para executar adequadamente as atividades de teste; e
- a dificuldade de adaptar modelos de maturidade em teste para o ambiente específico da organização.

Essas dificuldades demonstram o interesse de pequenas empresas em investir em testes de software; porém a limitação de recursos e a escassez de profissionais especializados se tornam barreiras para elas. Outro problema relacionado é a falta de conhecimento acerca de modelos de maturidade, em especial, do TMMi, que especifica um modelo de referência para processo de teste. Uma alternativa para empresas que não possuem recursos ou equipe suficientes para testar internamente seus produtos, é terceirizar o teste de software. Essa ideia tem sido discutida com base no conceito de fábrica de software. Tal conceito corresponde à padronização de processos em desenvolvimento de software, que foi proposta com o intuito de diminuir a variabilidade na produtividade da programação (CUSUMANO, 1991). Com o aumento da demanda pelos serviços de teste e a exigência por menores prazos de entrega, os conceitos de fábrica de software têm sido aplicados ao teste de software, culminando com o surgimento das fábricas de teste. Sanz et al. (2009) afirmam que esse tipo de serviço pode melhorar a qualidade, reduzir o tempo de entrega e os custos para testar o produto.

Independentemente de quem realiza o teste, se a própria empresa ou uma fábrica especializada, o processo pelo qual o teste de software é realizado deve ser bem definido e deve permitir revelar o maior número possível de defeitos. Assim como para os processos de desenvolvimento, existem modelos de referência, como o MPS.Br (MPS.BR, 2011a) (Guia de Implementação Parte 10: Fábrica de Teste) e o TMMi (TMMi Foundation, 2010), que são específicos para guiar a implantação de processos de teste. O MPS.Br é um modelo de referência para processos de desenvolvimento de software e possui um guia especial para fábricas de teste (MPS.BR, 2011a). No Brasil não há instituições implementadoras e avaliadoras para o TMMi, o que torna sua utilização cara e complicada por envolver instituições estrangeiras, o que levou à concepção do modelo brasileiro de melhoria de processo de teste, o MPT.Br (MPT.BR, 2011), que ainda não é muito difundido.

Ressalta-se que esses modelos indicam *o que* deve ser feito para que o processo seja criado e, conseqüentemente, uma melhor qualidade alcançada, mas não indicam *como* implementar o processo. Em geral, a equipe responsável pelo projeto de criação do processo lê o conteúdo do modelo e mapeia mentalmente as exigências para o processo desejado. Durante a elaboração do processo, essa equipe verifica manualmente se as práticas obrigatórias, exigidas pelo modelo, são contempladas (PURPER, 2000). Essa abstração de um modelo de referência para um processo real se torna uma tarefa difícil devido à dificuldade de compreensão e ao fato de esses



modelos não indicarem o que deve ser implementado prioritariamente, caso não haja tempo ou recursos para a implantação completa do modelo. Ressalta-se que esses modelos em geral indicam alguma priorização por meio de seus níveis, porém não fica claro dentro de cada nível o que pode ser mais importante em um primeiro momento.

Esses modelos não oferecem à empresa um meio de elaborar seu processo sozinha, pois são complexos e de difícil compreensão. Para pequenas empresas, o processo de teste desenvolvido deve contemplar todas as fases do teste de software, mas com o mínimo de atividades possível, de acordo com sua realidade e o modelo de desenvolvimento de software utilizado. Isso pode tornar o processo de teste mais simples de ser executado, não exigindo muitos recursos, nem equipe muito grande, ajudando a alcançar o objetivo de melhorar a qualidade dos produtos.

## **1.2 Objetivo**

O objetivo deste trabalho é apresentar uma estratégia para a elaboração de processos de teste, diante da falta de um modelo de referência de fácil compreensão, que mostre o que deve ser implementado prioritariamente, quando não se tem muitos recursos, e diante das barreiras que pequenas empresas encontram para implementar e institucionalizar o processo de teste. A estratégia a ser desenvolvida deve ter como base modelos de referência de teste de software, visando à sua utilização por empresas de pequeno e médio porte, que desejam implementar ou melhorar o processo já em uso.

Espera-se também que essa estratégia possa ser utilizada por fábricas de teste, mas salienta-se que essa estratégia não leva em consideração atividades relacionadas à venda do serviço de teste, nem atividades relacionadas à garantia de qualidade do serviço. O propósito deste trabalho também é encontrar um conjunto mínimo de atividades, indicando assim quais são as atividades prioritárias para uma versão inicial de um processo de teste, podendo exigir uma quantidade menor de recursos em sua implementação.

## **1.3 Metodologia**

O objetivo definido foi alcançado por meio da realização de um estudo prático em uma empresa de software. Essa empresa, no momento em que se iniciou este trabalho, já era certificada MPS.Br nível G e buscava a certificação nível F. Embora o MPS.Br não incluía a atividade de teste nestes níveis, essa empresa já buscava melhorar tal processo, visando uma futura certificação no nível D, em que há o teste de software. Atualmente a empresa já é certificada

nível F e a área de testes continua em evolução.

O trabalho foi iniciado com o estudo dos conceitos de processo de teste, pela busca por modelos de processo de teste existentes, e também com o estudo sobre o TMMi. Em seguida, iniciou-se a fase de contato com a empresa. O primeiro contato feito foi para a realização de um diagnóstico do processo de teste utilizado por ela. O objetivo do diagnóstico foi identificar se o processo que a empresa possuía modelado era realmente executado. Para esse diagnóstico foi utilizada uma ferramenta que avalia o grau de implementação das práticas do TMMi.

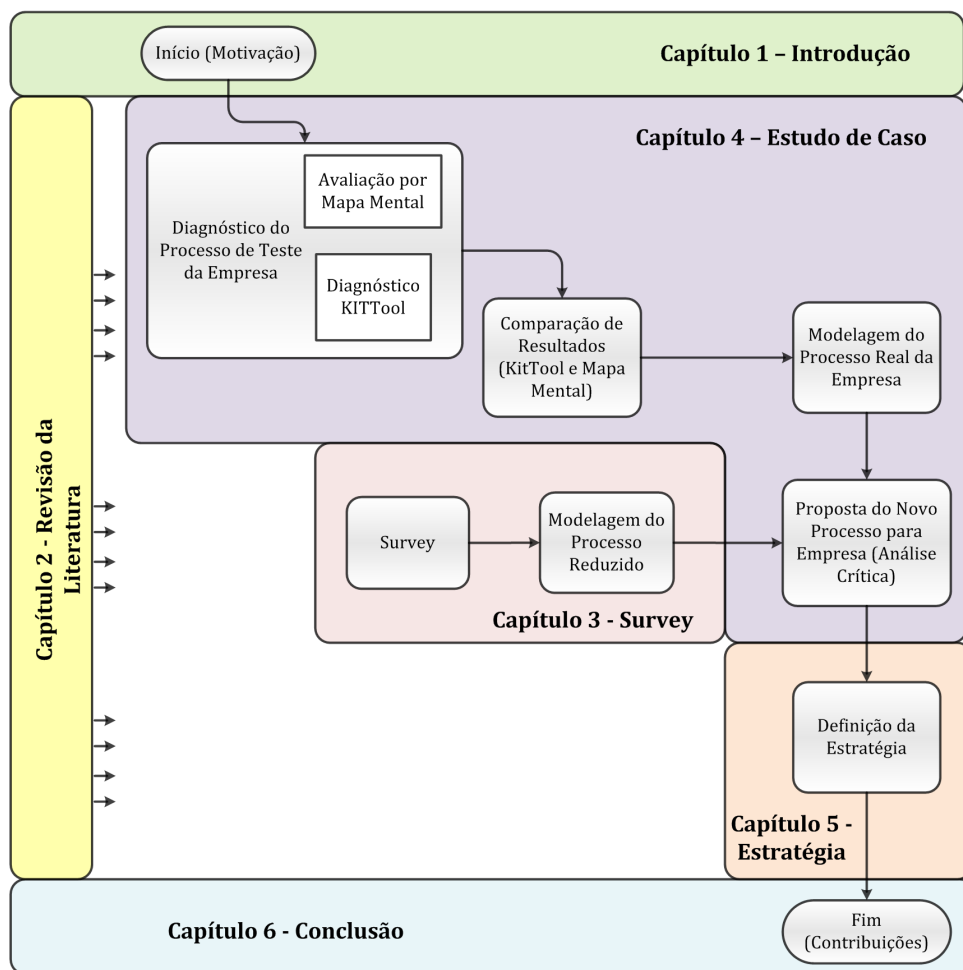
Após a conclusão do diagnóstico, o processo real foi modelado, pois foi constatado que havia diferenças entre o processo praticado e o processo modelado pela própria empresa. O novo modelo foi validado pela empresa.

Também foi realizado um *survey*, tendo como base o TMMi, com o objetivo de identificar se esse modelo de referência, que possui um número elevado de práticas a serem implementadas, poderia ser reduzido a um conjunto menor, de tal forma que ainda contemplasse todas as fases de um processo genérico de teste. Com base no resultado do *survey* e no modelo de processo realmente utilizado pela empresa, elaborou-se um novo processo para ela, que retratava um processo mínimo adaptado à sua realidade, e incluía atividades prioritárias.

Dessa experiência de definição de um processo de teste para uma pequena empresa real, foi então extraída a estratégia aqui apresentada. Essa estratégia foi modelada para servir de referência para que outras empresas possam definir ou melhorar seu próprio processo de teste. Para dar origem à estratégia foi necessário generalizar os passos realizados para a elaboração do processo de teste no estudo prático, tornando possível a utilização dessa estratégia por qualquer empresa. Empresas que desejam tornar-se fábrica de teste também podem utilizar os resultados deste trabalho, desde que incluam algumas atividades que não foram englobadas pela estratégia proposta, que estão relacionadas à venda e garantia de qualidade. A Figura 1.1 apresenta um diagrama representativo da metodologia descrita acima e permite também compreender a organização do trabalho que será apresentada na próxima seção.

## 1.4 Organização do trabalho

Este trabalho é composto por seis capítulos. No primeiro foram apresentados o contexto em que o trabalho se insere, a motivação que levou ao seu desenvolvimento, o objetivo pretendido e a metodologia utilizada para alcançá-lo. No Capítulo 2 são descritos os conceitos fundamentais para o entendimento do trabalho, sendo eles: teste de software, processo, modelos de referência e trabalhos relacionados. No Capítulo 3 é descrito o *survey* realizado e seus resultados. No



**Figura 1.1: Metodologia utilizada para desenvolvimento deste trabalho.**

Capítulo 4 é descrita a elaboração do processo para a empresa em questão. Em seguida, no Capítulo 5 é descrita a estratégia objetivada por este trabalho. Por fim, no Capítulo 6 são apresentadas as contribuições, limitações e trabalhos futuros.



# Capítulo 2

## TESTE DE SOFTWARE

---

---

Este capítulo apresenta uma visão geral sobre o que é um processo de teste de software, os conceitos relacionados a teste necessários para compreender todas as etapas do processo, os modelos de referência que podem guiar a melhoria desse processo e os trabalhos relacionados que permitiram a elaboração da proposta deste trabalho.

### 2.1 Considerações Iniciais

Muitas empresas de software vêm adotando processos de teste como forma de garantir maior qualidade e, conseqüentemente, agregar ao produto valores como confiabilidade. Algumas empresas optam por terceirizar as atividades de teste por não possuírem conhecimento e recursos humanos suficientemente preparados para executar o teste com eficiência e eficácia. As atividades de garantia de qualidade de software certificam que os processos e produtos de software, no ciclo de vida do projeto, estejam em conformidade com os requisitos especificados e referentes aos planos estabelecidos (ROCHA; MALDONADO; WEBER, 2001). Testar um software consiste em executá-lo na tentativa de revelar o maior número possível de defeitos (MYERS et al., 2004), de forma a aumentar a qualidade do produto em questão. A qualidade não é um atributo que pode ser imposto ao produto depois de finalizado (JINO; MALDONADO; DELAMARO, 2007), o que ressalta a importância de testar o software desde o início de sua construção. Defeitos podem ser introduzidos em qualquer fase do desenvolvimento de software, e o quanto antes eles forem identificados, menor será o custo de sua correção (MALDONADO; FABBRI, 2001).

No processo de desenvolvimento de um novo software, a falta de atividades de teste que acompanhem todo o processo pode custar caro. Mesmo em personalizações de software, o teste tem papel importante para garantir que o sistema funcione como esperado após as alterações e

inserções de novos módulos. Para o cliente, qualidade corresponde, em geral, a verificar que o software desempenha exatamente o que ele deveria desempenhar. As atividades de V&V envolvem análises estáticas e dinâmicas do produto em desenvolvimento. A análise estática não implica na execução propriamente dita do produto. As inspeções e revisões técnicas são exemplos dessa atividade. A análise dinâmica objetiva detectar defeitos e envolve a execução do produto, seja ele o código ou uma especificação executável. Um exemplo de análise dinâmica é a atividade de teste, a qual deve ser utilizada em conjunto com a análise estática, pois contribui para a eliminação de defeitos que persistem no software (MALDONADO; FABBRI, 2001). Na verificação é avaliado se os processos e métodos de desenvolvimento foram aplicados corretamente. O processo de verificação consiste em duas atividades: uma relacionada à sua implementação e a outra relacionada a atividades como verificação do contrato, do processo, do projeto, do código, da integração e da documentação (MALDONADO; FABBRI, 2001). A verificação permite avaliar se o produto desenvolvido está conforme o planejado. Já a validação lida diretamente com os requisitos do cliente para assegurar que o produto que foi solicitado é o mesmo que está sendo construído. Essa atividade verifica se uma função necessária, que é esperada pelo cliente, está presente no produto de software (TIAN, 2005). O processo de validação consiste em duas atividades principais, sendo uma delas relacionada à sua própria implementação, enquanto a outra consiste em tarefas como preparar os requisitos de teste, casos de teste e especificações de teste, executar os diversos tipos de testes, conduzir testes em seu ambiente alvo e validar se o software satisfaz seu uso específico (MALDONADO; FABBRI, 2001). Essas atividades descrevem um processo de teste. A próxima seção apresenta o que é um processo de teste em conjunto com as suas principais etapas.

Neste capítulo serão apresentadas a definição de processo de teste de software, e os conceitos relacionados, na Seção 2.2; na Seção 2.3 serão apresentados os modelos de referência em teste de software que estão relacionados à este trabalho. Na Seção 2.4 será apresentada a base de conhecimento em teste, desenvolvida por Höhn (2011), que foi utilizada como parte deste trabalho.

## 2.2 Processo de Teste de Software

Para desenvolver um produto de software é necessário seguir passos (ou atividades) bem definidos que possam ser repetidos. O conjunto dessas atividades, quando é devidamente executado e produz o resultado esperado, é chamado processo (HUMPHREY, 1989). Somente quando as atividades estão organizadas na forma de processos com a descrição das entradas, das atividades em si e dos produtos resultantes, elas podem ser monitoradas e melhoradas

(HASS, 2008). A busca por melhorias em seu processo vem se tornando prática comum em empresas cujo foco é a qualidade e a competitividade. Modelos de maturidade de processo de desenvolvimento de software como o Capability Maturity Model Integration (CMMI) (SEI, 2006) e Melhoria de Processo de Software Brasileiro (MPS.Br) (MPS.BR, 2011b) podem ser adotados para guiar a implementação de processos e também suas melhorias. Esses modelos de maturidade sugerem a elaboração de um processo padrão o qual descreve as práticas que devem ser adotadas em todos os projetos levando em conta as características da organização. Esse processo padrão será utilizado como base para a criação de processos especializados e instanciados. O processo padrão deve ser especializado de acordo com o tipo de software e padrão de desenvolvimento utilizado, acrescentando ou modificando as práticas do processo padrão. O processo especializado é instanciado para cada projeto específico, adaptando-o de acordo com o ciclo de vida, o método de desenvolvimento utilizado, as ferramentas adotadas no projeto, os recursos humanos e os requisitos de qualidade do produto.

O processo de teste apoia o desenvolvimento com o intuito de que o produto final esteja de acordo com o especificado. O sucesso do desenvolvimento de software e da atividade de teste dependem diretamente de um bom planejamento e da qualidade dos processos utilizados. Um processo teste, assim como de software, possui uma série de fases, procedimentos e passos (BURNSTEIN, 2003). Na atividade de teste não existem muitas informações acerca das atividades que devem fazer parte de todo processo de teste, independentemente do processo de desenvolvimento adotado pela empresa. Apesar disso, estão surgindo modelos de referência específicos para processos de teste. Exemplos são o *Test Maturity Model Integration* (TMMi Foundation, 2010) e o Melhoria de Processo de Teste Brasileiro (MPT.BR, 2011), que serão tratados na Seção 2.3. Esses modelos sugerem a elaboração de um processo padrão, mas nesse caso, específico para teste de software. O processo padrão de teste também deve ser especializado de acordo com o ciclo e vida e método de desenvolvimento. Sempre que um produto de software é desenvolvido, a atividade de teste deve acompanhar esse processo desde o início com o objetivo de encontrar defeitos, e conseqüentemente, aumentar a qualidade do produto final. Para que o teste de software possa ser executado sempre da mesma maneira, é preciso que os passos requeridos estejam bem definidos de forma que a atividade se torne facilmente controlável e possa ser repetida sempre que necessário, o que caracteriza um processo. Portanto, para que o teste de software possa ser realizado com qualidade é necessário que um processo esteja definido. Davis (2000) afirma que quando o processo de teste se inicia, algumas questões devem ser respondidas com o objetivo de organizar o trabalho, sendo elas:

- O que testar?

- Como testar?
- Quando testar?
- O que aconteceu depois que o produto foi testado?

Essas questões correspondem às atividades de planejamento, projeto de casos de teste, execução, medição e análise. Essas atividades se assemelham a um processo de teste fundamental e não necessariamente ocorrem em uma única sequência. Algumas delas devem ser executadas mais de uma vez, pois ganham maturidade com o andamento do processo. Crespo et al. (2010) afirmam que um modelo de processo genérico de teste de software é abrangente, envolve todas as situações de teste e possui as seguintes fases: *Planejamento do Teste, Projeto do Teste, Execução, Acompanhamento e Finalização do Teste*. Afirmam ainda que esse processo genérico pode ser instanciado e implementado em uma organização de acordo com suas necessidades, recursos, padrões e tipo de software que desenvolve. Graham et al. (2008) e Hass (2008) identificam fases semelhantes para um processo de teste fundamental. A seguir são descritas as fases de um processo genérico, ou fundamental.

**Planejamento:** Segundo a TMMi Foundation (2010), o proposta do planejamento de teste é definir uma abordagem de teste com base nos riscos identificados, definir uma estratégia de teste e estabelecer e manter um plano bem fundamentado para a realização e gerenciamento das atividades de teste. Na fase de planejamento também se identificam os recursos necessários, incluindo treinamentos, produtos a serem entregues, cronograma das atividades e o ambiente necessário para a realização dos testes (HASS, 2008). Outras atividades podem fazer parte do plano de teste de acordo com a realidade da empresa que o implementa. Com o andamento do projeto, o planejamento de teste deve ser revisado e modificado para que se ajuste à realidade do projeto. No planejamento também devem ser definidos o objetivo de se testar o software, qual estratégia deverá ser utilizada para alcançar esse objetivo e quais os resultados esperados (DAVIS, 2000), além de itens como definição de equipe, cronograma e definição dos critérios de parada (IEEE, 2008). Na fase de planejamento também devem ser obtidos e analisados os requisitos de ambiente de teste, com o objetivo reproduzir o ambiente em que o software será utilizado, para que os testes possam ser realizados adequadamente. Com o planejamento realizado, a etapa seguinte é o projeto (dos casos) de teste.

**Projeto de teste:** A proposta do projeto de teste é a criação dos casos de teste. Para isso é necessário refinar a abordagem definida no plano de teste, observar os riscos do produto, identificar cenários e condições de teste de acordo com o que foi definido que será testado. A seleção



dos dados de entrada para os testes é feita com base nas estratégias, técnicas e critérios que foram definidas no plano de teste. Os critérios de teste fornecem um direcionamento para o término da atividade de teste, pois os objetivos de cobertura e confiabilidade são analisados ao fim da execução. Projetar o ambiente e identificar ferramentas de apoio também fazem parte desta etapa. O apoio de ferramentas pode auxiliar tanto na criação de cenários de teste (OCTAVIANO et al., 2012) quanto no gerenciamento dos casos de teste e dos defeitos observados. O ganho de produtividade com a automatização do processo pode ser crucial no sucesso do cumprimento de prazos.

**Execução:** Nessa fase os casos de teste são executados no ambiente que foi planejado, com os dados e cenários que foram definidos no planejamento, e o resultado da execução é registrado. A atividade de execução dos testes é altamente iterativa, pois caso uma falha seja detectada, e o defeito causador seja corrigido, deve-se executar novamente o caso de teste a fim de verificar se a falha não é mais produzida e se não surgiram novas falhas decorrentes da correção do defeito. Sempre que um defeito é encontrado ele deve ser registrado detalhadamente. O gerenciamento de defeitos pode ser facilitado com o a utilização de ferramentas automatizadas.

**Acompanhamento:** Hass (2008) alerta para a importância do monitoramento, controle e replanejamento do projeto de teste para a realização de correções necessárias no plano de teste quando este já não reflete mais a realidade do projeto de construção do software. Esse é o objetivo da fase de acompanhamento: fornecer visibilidade do andamento do projeto de teste, constituindo então uma atividade complementar ao controle de projeto. Esse controle consiste no desenvolvimento e aplicação de um conjunto de ações corretivas quando o monitoramento mostra que houve um desvio do que foi planejado. A realização dessas atividades requer o uso de ferramentas, técnicas, formulários e métricas (BURNSTEIN, 2003). O uso de métricas permite responder questões sobre o andamento e a qualidade do processo e do produto resultante de sua aplicação. Analisar os resultados obtidos pode revelar a cobertura oferecida pelos casos de teste utilizados e a necessidade de se evoluir o conjunto de testes para avaliar outras partes do software. Um bom controle do projeto garante que os objetivos estabelecidos no plano de teste serão alcançados.

**Finalização do Teste:** Nesta fase são realizadas atividades para encerramento do teste no momento de entrega do software. São também analisadas e consolidadas informações sobre as lições aprendidas, os dados obtidos a partir dos resultados da execução do teste, desempenho da equipe, informações sobre o processo executado, e métricas, que já definidas no processo, podem indicar a utilização de recursos e tempo gasto em relação ao planejado (CRESPO et al.,

2010). O objetivo dessa fase também é liberar recursos físicos desfazendo ambientes de testes que não são mais necessários (HASS, 2008). Nessa fase também é possível identificar possíveis melhorias no processo.

As fases apresentadas caracterizam, de maneira geral, o que todo processo de teste deve realizar. Cada fase pode ser representada por subprocessos essenciais, que assumem formas diferentes de acordo com a realidade e contexto da empresa que especializa e instancia o processo de teste. Elaborar um processo de teste requer conhecimento da empresa que o faz, para que, quando instanciado em um projeto, se possa alcançar os objetivos esperados com relação à qualidade do produto.

O aumento da competitividade pela qualidade tem feito muitas empresas buscarem serviços especializados nessa área, ou mesmo criar uma equipe especialista dentro da própria organização. Uma forma alternativa de garantir a qualidade do produto construído é contratar uma empresa especializada para a realização dos testes. A terceirização (*outsourcing*) é vista como uma maneira de aumentar a eficiência e reduzir os custos no teste de software (TAIPALE; SMOLANDER; KALVIAINEN, 2006 apud KARHU; TAIPALE; SMOLANDER, 2007). Segundo a literatura, a confiança no produto será maior se o teste for totalmente realizado por uma agência independente de teste (KANER; FALK; NGUYEN, 1999). Problemas e desvantagens de uma agência independente de teste são os mesmos encontrados em uma equipe interna de teste da empresa (KARHU; TAIPALE; SMOLANDER, 2007). Essa terceirização pode ocorrer no modelo de fábrica, no qual uma empresa contrata o serviço de uma fábrica que possui equipe e infraestrutura especializada. A utilização dos conceitos de fábrica para construção de software surgiu da necessidade de se produzir produtos de software mais complexos, em menor tempo e com maior qualidade (CUSUMANO, 1991). Desde então o conceito de fábrica de software vem sendo utilizado para designar empresas que oferecem serviços exclusivamente em desenvolvimento de software. Esses mesmos conceitos de fábrica podem ser aplicados para empresas que fazem exclusivamente teste de software, denominando-se fábricas de teste. Uma fábrica de teste possui recursos dedicados e aumenta a qualidade e confiabilidade no produto final da empresa que a contrata (SANZ et al., 2009).

Como pode ser notado, é consenso entre alguns autores o que constitui um processo fundamental de teste, ou seja, quais atividades devem fazer parte de qualquer processo de teste. Entretanto, não se conhece um modelo de processo de teste que possa ser aplicado por empresas, facilmente. Existem modelos de referência específicos para teste de software que podem ser utilizados para guiar a implementação do processo de teste. Na seção a seguir são apresentados alguns modelos de referência relacionados ao teste de software e a este trabalho.

## 2.3 Modelos de Referência

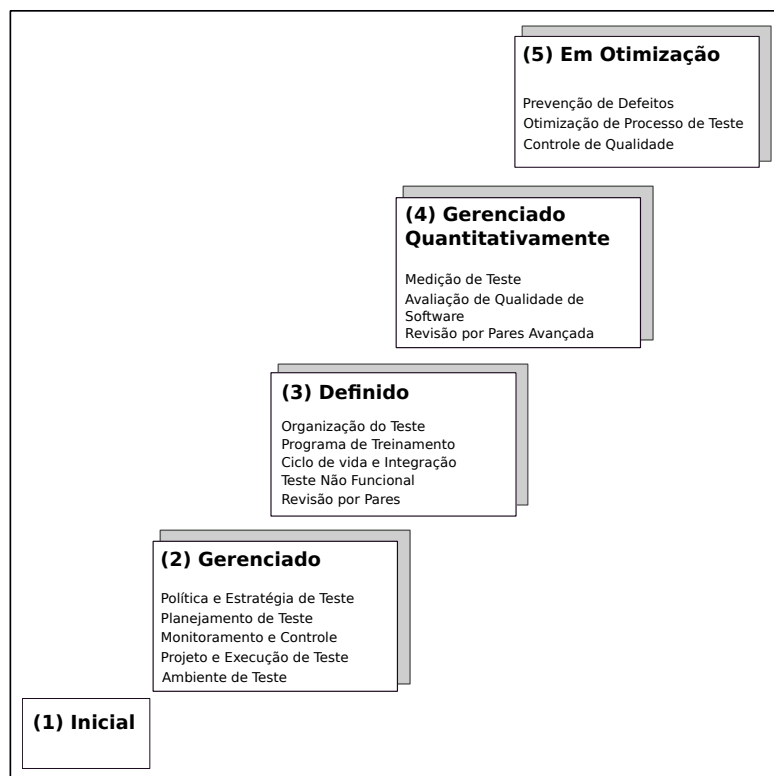
A definição de processos em uma empresa pode ser feita com base em modelos de referência. Esses modelos indicam as melhores práticas que podem ser utilizadas na elaboração do processo. Modelos de referência fornecem diretrizes do que deve ser implementado, porém a maneira como será implementado fica a cargo da organização. Além de fornecer informação para elaboração de processos, um modelo de referência também permite que a empresa obtenha maturidade para analisar o processo que já está em uso em busca de melhorias. Existem diversos “guias” que podem apoiar a criação de processo de teste em organizações. Entre eles podem-se citar modelos de referência específicos para teste de software como os *Test Maturity Model integration* (TMMi) (TMMi Foundation, 2010) e Melhoria do Processo de Teste Brasileiro (MPT.Br) (MPT.BR, 2011), e certificações como a *International Software Testing Qualifications Board* (ISTQB) (ISTQB, 2010). Além desses modelos, atualmente está em desenvolvimento uma nova norma que define um modelo padrão para o processo de teste nas organizações, a ISO/IEC 29119 (KASURINEN, 2010). A seguir são apresentados em mais detalhes os modelos de referência TMMi e MPT.Br, pois estão diretamente relacionados à este trabalho.

### 2.3.1 *Test Maturity Model integration* – TMMi

O modelo de maturidade *Capability Maturity Model* (CMM) oferece um meio de avaliar o processo de desenvolvimento de software fornecendo diretrizes de melhores práticas que podem ser adotadas (SEI, 2006). O CMMI (SEI, 2006) é uma evolução do CMM e objetiva ajudar organizações a melhorar o desenvolvimento e a manutenção de processos de produtos e serviços, e integra os modelos de software, sistemas de engenharia e a integração de desenvolvimento de produto.

Complementar ao CMMI, o TMMi (TMMi Foundation, 2010) foca o esforço de melhoria no processo de teste. O TMMi foi desenvolvido pela *TMMI Foundation* como um guia e um *framework* para a melhoria de processos de teste. O TMMi é semelhante ao CMMI em sua estrutura, pois possui níveis de maturidade para avaliação e melhoria de processo, como mostra a Figura 2.1. Metas, áreas de processo e práticas são identificadas, para que, ao aplicar os critérios de melhoria, o ganho de produtividade da equipe de teste e qualidade de produto seja maior.

No TMMi, o processo evolui de um estado inicial (nível 1), caótico, a um estado em que o processo é gerenciado, controlado e otimizado (nível 5). Na Figura 2.1 são mostrados os níveis



**Figura 2.1:** Níveis de maturidade e áreas de processo do TMMi [adaptada de TMMi Foundation (2010)].

de maturidade do TMMi com suas respectivas áreas de processo. Como pode ser observado, o Nível 1 (Inicial) não possui áreas de processo associadas. Nesse nível o teste é caótico e não há um processo definido; o objetivo se resume a mostrar que o software funciona sem defeitos muito graves. No Nível 2 (Gerenciado) o teste começa a ser gerenciado e o objetivo principal é verificar se o produto satisfaz os requisitos especificados. Um plano de teste é desenvolvido para guiar as atividades de teste, mesmo que ele comece tardiamente no ciclo de vida do software. A análise de risco de produto é utilizada para determinar a abordagem que será utilizada, além disso o teste é monitorado e controlado para garantir que o plano está sendo seguido. No Nível 3 (Definido) o teste está totalmente integrado ao ciclo de vida de desenvolvimento, existe um processo padrão definido e a organização entende a importância das revisões por pares no controle de qualidade. Nesse nível a melhoria de processo já está institucionalizada na organização. No Nível 4 (Gerenciado Quantitativamente), as revisões por pares estão totalmente integradas ao processo de teste dinâmico como parte da estratégia e também são feitas medições que permitem avaliar o desempenho e custo do processo de teste. Após implantar esse nível a organização já possui um processo teste completamente definido e mensurável, além de possuir toda a infraestrutura necessária para executar esse processo. Por fim, no Nível 5 (Otimizado) a organização deverá ser capaz de melhorar continuamente seu próprio processo, além de trabalhar na prevenção de defeitos e controle de qualidade de seus

produtos.

Com exceção do Nível 1, cada um dos níveis do TMMi é composto por áreas de processo específicas que permitem identificar o que deve ser implementado para atingir aquele nível. Na Figura 2.2 é mostrada a estrutura do TMMi. As **áreas de processo** identificam os pontos que devem ser abordados para alcançar o nível de maturidade ao qual elas pertencem e cada uma identifica um conjunto de atividades relacionadas entre si. Um nível só é alcançado quando todas as áreas de processo são satisfeitas. Uma área de processo possui objetivos específicos que auxiliam na compreensão do que deve ser feito para implementá-la. Cada **objetivo específico** referencia uma determinada característica que deve estar presente para que aquela área de processo seja satisfeita. Nas avaliações de certificação do TMMi esses objetivos específicos são utilizados para auxiliar na verificação se a respectiva área de processo é satisfeita. O objetivo específico é dividido em **práticas específicas** que descrevem quais atividades são importantes e podem ser realizadas para atingir esse objetivo. A realização das atividades de uma prática específica resulta no objetivo específico alcançado para determinada área de processo. Já os **objetivos genéricos** estão relacionados a mais de uma área de processo e descrevem características que podem ser utilizadas para institucionalizar o processo de teste. Objetivos genéricos possuem **práticas genéricas**, que são assim chamadas pois aparecem em mais de uma área de processo, e apenas descrevem uma atividade que é considerada importante para que se alcance o objetivo genérico associado. Na Figura 2.2 pode-se visualizar a hierarquia da estrutura descrita.

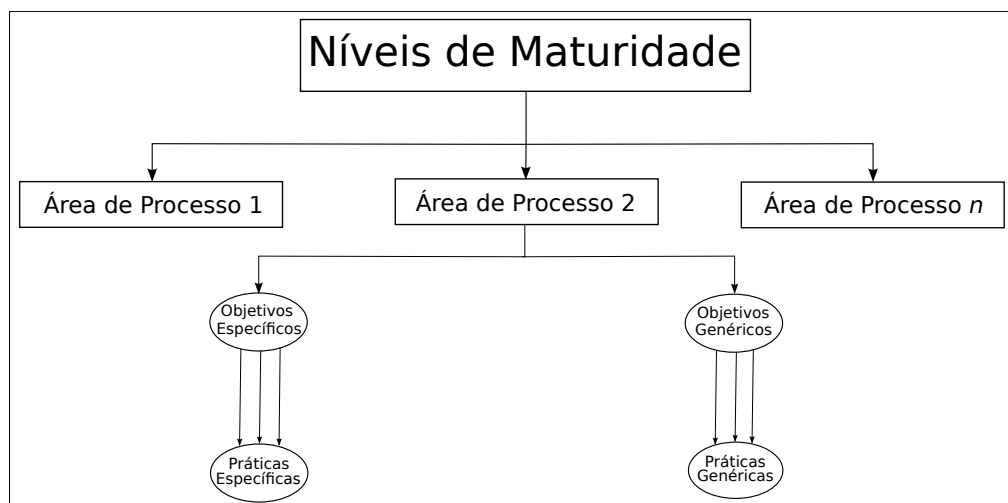


Figura 2.2: TMMi structure and components (TMMi Foundation, 2010).

Como pode ser visto, o TMMi possui uma estrutura que indica claramente o que deve ser feito para que determinado nível possa ser atingido, porém não há indicações, para implementações parciais, de quais áreas de processo são mais importantes ou que poderiam ser implantadas primeiro. A implementação sempre deve começar a partir do nível mais baixo

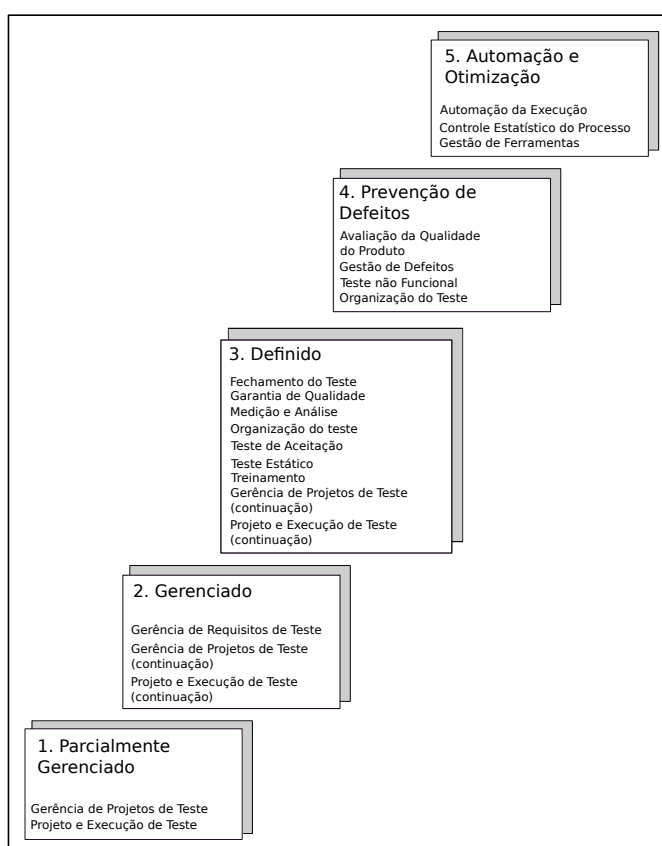
que a empresa se encontra e vai evoluindo conforme seu processo ganha maturidade. Observa-se, entretanto, que mesmo dentro de um nível uma organização (por exemplo, uma empresa de pequeno porte) poderia optar por fazer uma implementação parcial, sem objetivar a certificação. Esse modelo é pouco utilizado no Brasil, pois como não há instituição implementadora é necessário o envolvimento de instituições estrangeiras, o que torna o processo de certificação ainda mais caro. Sabe-se que existe um modelo de maturidade para teste de software criado no Brasil, o qual também oferece certificação para empresas, porém ainda é pouco utilizado. Trata-se do Modelo de Melhoria do Processo de Teste Brasileiro (MPT.Br), que é apresentado na próxima seção.

### **2.3.2 Modelo de Melhoria do Processo de Teste Brasileiro – MPT.Br**

A melhoria de processo de teste de software é interessante para organizações que possuem uma equipe de teste bem definida e que, de fato, testam todos seus produtos. Muitas empresas possuem equipe de teste que nem sempre são suficientes para a demanda interna existente. No Brasil, o modelo de referência *Melhoria do Processo de Software Brasileiro* (MPS.Br) (MPS.BR, 2011b) foi criado com o objetivo de oferecer um modelo de processo de desenvolvimento de software a pequenas e médias empresas. Compatível com modelos internacionais, o MPS.Br atende à necessidade de implantar os princípios de engenharia de software de acordo com o contexto da empresa (MPS.BR, 2011b). O MPS.Br oferece uma maneira mais barata, comparada com padrões internacionais, como o CMMI, para que a pequena ou média empresa adquira maturidade em seus processos de desenvolvimento de software, tornando-se, dessa forma, mais competitiva. O MPS.Br foca o desenvolvimento de software em geral, e inclui a área de teste, assim como o CMMI (SEI, 2006). O modelo de referência do MPS.Br está organizado em guias de implementação que indicam o que deve ser implementado, além de um guia geral que descreve o modelo. Como é crescente o número de empresas que fornecem serviços exclusivamente em teste de software, o MPS.Br oferece uma guia de implementação específica para organizações do tipo fábrica de teste (MPS.BR, 2011a). Além disso, foi criado um modelo exclusivamente para processo de teste, o Melhoria do Processo de Teste Brasileiro (MPT.Br) (MPT.BR, 2011).

O MPT.Br foi elaborado e é mantido pelo Softex Recife e RioSoft. Para a construção do MPT.Br, tomou-se como base diversos modelos de maturidade em teste de software, entre eles o TMMi. O objetivo da construção do MPT.Br foi criar uma certificação alternativa brasileira e que não envolvesse instituições estrangeiras, como é o caso do TMMi. Consequentemente, o custo seria reduzido e mais empresas brasileiras de diversos portes poderiam implantar e

melhorar seu processo de teste. O MPT.Br objetiva a melhoria do processo de teste por meio da realização das melhores práticas ao longo de todo ciclo de vida de teste do produto (MPT.BR, 2011) e pode ser visto como complementar ao MPS.Br. Entretanto, trata-se de um modelo ainda pouco conhecido, sendo que as primeiras empresas foram certificadas no ano de 2010 e até o momento o nível máximo de certificação é no nível 2<sup>1</sup>.



**Figura 2.3: Níveis de maturidade do MPT.Br (Adaptado de MPT.Br (2011)).**

O MPT possui uma estrutura semelhante ao TMMi e também está organizado em cinco níveis de maturidade, como mostrado na Figura 2.3. No primeiro nível o teste é parcialmente gerenciado e, diferentemente do TMMi, possui áreas de processo associadas para garantir que a organização realiza o mínimo necessário para que o teste ocorra de forma planejada e controlada. No Nível 2 (Gerenciado) o processo passa a ser monitorado e controlado. No Nível 3 (Definido) existe um processo definido que é institucionalizado na organização e integrado ao ciclo de vida do desenvolvimento. Teste estático e de aceitação também são formalizados, e nesse nível é inserido um programa de medição na organização. No Nível 4 (Prevenção a Defeitos) o enfoque está em prever defeitos por meio da gestão de defeitos, realização de testes não funcionais e avaliações de qualidade do produto. No nível mais alto de maturidade, o Nível 5 (Automação e Otimização), o processo é acompanhado continuamente e controlado estatisticamente para que

<sup>1</sup>Fonte: <http://www.mpt.org.br/certificacao/>

as melhorias possam ser identificadas e, além disso, há um processo sistemático para seleção e adoção de ferramentas que apoiam o teste. Na Figura 2.3 pode-se observar as áreas de processo que estão associadas a cada nível e que apoiam as atividades mencionadas.

Cada nível do MPT é composto por áreas de processo que agrupam as práticas que devem ser implementadas para atingir determinado objetivo. Para que a organização possa obter a certificação em determinado nível de maturidade, ela deve cumprir todas as áreas de processo daquele nível e, se houver, atender todos os níveis anteriores (MPT.BR, 2011). Assim como no TMMi, as áreas de processo também possuem práticas específicas que detalham melhor a área de processo e mostra quais os produtos típicos de trabalho. Existem algumas semelhanças entre o MPT.Br e o TMMi. Entretanto, como pode ser visto nas Figuras 2.1 e 2.3, o primeiro nível nos dois modelos são diferentes, pois no TMMi não há áreas de processo relacionadas (pois o processo ainda é considerado caótico), enquanto que no MPT.Br o processo já começa a ser organizado. Outra grande diferença entre os dois modelos é que no MPT.Br a automação está no último nível de maturidade, diferentemente do TMMi, em que a automação não está exclusivamente em um nível. O MPT.Br não trata da elaboração de uma política exclusiva de testes, como o TMMi, que possui áreas de processo e práticas específicas para institucionalização do processo.

Ambos os modelos tratam todas as etapas de um processo genérico de teste, porém modelos de referência são complexos e exigem grande esforço para interpretação. Modelos de maturidade em geral apresentam somente o que deve ser feito para que o processo possa ser criado, mas não como fazê-lo e nem qual a prioridade das áreas de processo. Para pequenas empresas, a utilização do TMMi, por exemplo, pode ser inviável, mesmo que apenas como fonte de consulta para construção de seu processo sem visar a certificação. Por essa razão, a proposta deste trabalho é fornecer uma estratégia de criação de processo de teste que permita às pequenas empresas maior facilidade na elaboração de seu processo de teste. Para este trabalho, optou-se pela utilização do TMMi por ser um modelo amplamente conhecido; ainda, este trabalho está relacionado a um trabalho de doutorado (HÖHN, 2011) que utilizou o TMMi como ponto de partida para construir uma base de conhecimento em teste, como é apresentado na seção a seguir.

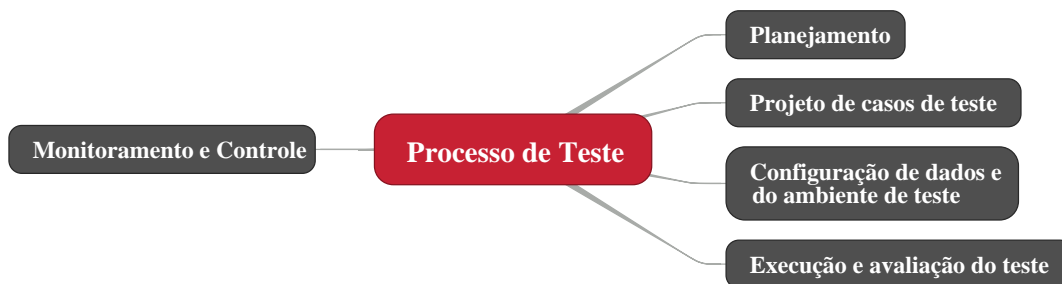
## 2.4 Base de Conhecimento em Teste

Devido à falta de informação centralizada e organizada a respeito de teste de software, Höhn (2011) desenvolveu uma base de conhecimento em teste de software na qual a informação é organizada e integrada de forma a facilitar a compreensão e aquisição de conhecimento. Para que



essa base de conhecimento pudesse ser utilizada pela comunidade no diagnóstico do processo de teste em uso, Höhn (2011) desenvolveu uma ferramenta que permite visualizar os resultados desse diagnóstico e também fornece diretrizes para a melhoria do processo de teste em uso com base no TMMi.

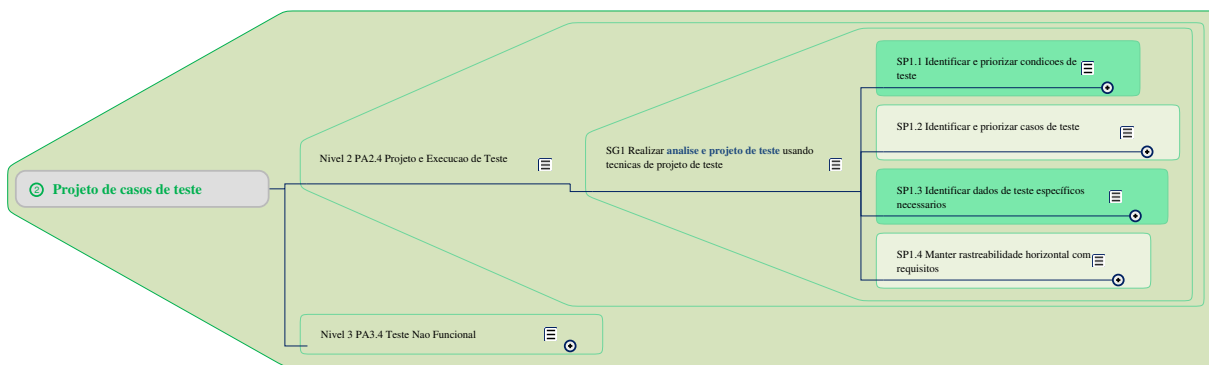
A primeira etapa de seu trabalho foi o desenvolvimento do KITMap (*Knowledge and Improvement Test Map*), que consiste em um mapa mental que contém as áreas de processo do TMMi distribuídas entre as fases de um processo de teste genérico. Um mapa mental é uma representação gráfica entre conceitos e ideias relacionados a um assunto central (BUZAN; BUZAN, 1996). O KITMap foi desenvolvido com o intuito de facilitar a compreensão do TMMi e o compartilhamento da informação. Nesse mapa mental, o nó raiz é o nome do tema central, ou seja, o processo de teste. Os nós no segundo nível são as etapas de um processo genérico de teste: Planejamento, Projeto de Casos de Teste, Configuração de Dados e Ambiente, Execução de Análise, e Monitoramento e Controle. No terceiro nível do KITMap encontram-se as áreas de processo do TMMi, no quarto nível estão os objetivos específicos relacionados e no quinto nível as práticas específicas de cada área de processo que está relacionada com a respectiva fase do processo genérico de teste. Na Figura 2.4 são mostrados os dois primeiros níveis do KITMap. Do lado direito estão os níveis que são diretamente relacionados com a execução do processo de teste e do lado esquerdo está o nível que é relacionado à parte de apoio ao processo, ou seja, monitoramento e controle, que acompanha todas as fases do processo desde o planejamento até a execução e avaliação.



**Figura 2.4:** Parte do KITMap mostrando as etapas do processo genérico de teste (Adaptado de Höhn (2011).)

As áreas de processo do TMMi foram distribuídas de acordo com as suas relações com cada fase do processo. Na Figura 2.5 é mostrado, para uma fase do processo genérico, a distribuição das áreas de processo do TMMi feita por Höhn (2011). Nessa figura é possível visualizar o objetivo específico relacionado à área de processo e suas várias práticas específicas. Também é possível observar que áreas de processo de diferentes níveis estão associadas à fase de *Projeto de Casos de Teste*, como é o caso das áreas *Projeto e Execução de Teste*, que pertence ao

Nível 2, e *Teste Não Funcional*, que pertence ao Nível 3. Todos os outros níveis do KITMap, com as respectivas áreas de processo e práticas específicas, poderão ser vistos no Capítulo 3 na Seção 3.4. Essa distribuição, em mapa mental, pode permitir uma melhor visualização do modelo TMMi como um todo e, conseqüentemente, pode facilitar a sua compreensão, pois indica as áreas de processo e sua relação com um processo real. As fases do processo genérico de Höhn (2011) são ligeiramente diferentes do processo genérico citado na Seção 2.2 justamente para facilitar esse mapeamento do TMMi em um processo real. Com o objetivo de ampliar o uso dessa base de conhecimento, Höhn (2011) desenvolveu também uma ferramenta que utiliza essa informação modelada no mapa mental para diagnosticar o processo em uso e visualizar quais práticas que sugere o TMMi, que faltam ser implementadas.



**Figura 2.5:** Parte do KITMap mostrando a fase de projeto de casos de teste (HÖHN, 2011).

A KITTool (*Knowledge and Improvement on Test - Tool*) permite que o usuário faça um diagnóstico do processo de teste que utiliza, a partir das práticas específicas ou objetivos específicos do TMMi. Além disso, a ferramenta indica o caminho para a definição de um novo processo com base no resultado do diagnóstico. Sabe-se que algumas práticas sugeridas pelo TMMi têm relação direta com práticas de outras fases do processo ou mesmo objetivos. Por exemplo, para que se possa realizar o planejamento do teste é necessário que antes se faça a análise de risco do produto, como mostrado na Seção 2.2. Essas dependências entre as práticas do TMMi foram mapeadas por Höhn (2011), pois o resultado apresentado pela KITTool considera essas dependências, já que isso influencia na ordem em que poderão ser implementadas as melhorias. Höhn (2011) afirma que existem dois tipos de dependências: a *necessária*, que ocorre quando uma prática precisa do resultado da prática que deve ser implementada antes, e a de *alinhamento*, que ocorre quando os produtos gerados pelas práticas dependentes precisam ser coerentes entre si. Um exemplo é a estratégia de teste, que deve estar alinhada à política de teste. Nesse caso, existe uma dependência de alinhamento entre a estratégia de teste e a política de teste. A tabela de dependências mapeadas por Höhn (2011) encontra-se no Anexo A.

A ferramenta KITTool utiliza essas dependências para indicar quando uma prática deve

ser implementada primeiro que outras no momento de se implementarem as melhorias em um processo de teste. A indicação do que deve ser implementado baseia-se no que foi diagnosticado com o auxílio dessa ferramenta. A avaliação com base nos *objetivos* é realizada por meio da atribuição de notas para cada objetivo de acordo com o quanto o processo satisfaz o objetivo. As notas devem ser atribuídas em 3 itens: abordagem, implantação e resultados, e devem seguir a escala fornecida pela ferramenta. A Figura 2.6 apresenta a tela da ferramenta correspondente a essa avaliação. Na parte destacada em vermelho da figura pode-se visualizar a escala de notas.

Área de processo: Nível 2 PA2.1 Política e Estratégia de Teste			
Objetivos	Abordagem	Implementação	Resultados
SG1 Estabelecer política de teste	2	0	0
SG2 Estabelecer uma estratégia de teste	2	0	0
SG3 Estabelecer indicadores de desempenho de teste	0	0	0

Tabela de referência				
Pontuação	Conceito	A - Approach (Abordagem)	D - Deployment (Implantação)	R - Results (Resultados)
0	Improdutivo (Poor)	Nenhum reconhecimento da necessidade pela gerência. Nenhuma habilidade organizacional. Nenhum compromisso organizacional. A prática não é evidente.	Nenhuma parte da organização utiliza essa prática. Nenhuma parte da organização mostra interesse.	Ineficiente.
2	Fracó (Weak)	A gerência começa a reconhecer a necessidade. Processos e padrões para a prática começam a ser criados. Algumas partes da organização são capazes de implementar a prática e são treinados.	Uso fragmentado. Uso inconsistente. Implantada em algumas partes da organização. Monitoramento/verificação limitados do uso.	Resultados imprecisos. Resultados inconsistentes. Algumas evidências de efetividade em algumas partes da organização.
4	Razoável	Ampla, mas não total comprometimento da gerência. Um roteiro para a implementação prática definido, ex. um plano de treinamento. Vários processos e padrões para a prática em vigor.	Uso menos fragmentado. Alguma consistência no uso. Implantada em algumas partes importantes da organização. Monitoramento/verificação do uso em diversas partes da organização.	Resultados consistentes e positivos em várias partes da organização. Resultados inconsistentes para outras partes da organização.
6	Qualificado marginalmente	Algum comprometimento da gerência; algumas gerências tornam-se proativas. Implementação da prática bem encaminhada por toda parte da organização, ex. muitos engenheiros treinados. Processos e normas em vigor.	Implantada em algumas partes da organização. Na maior parte das vezes, uso consistente em muitas partes da organização. Monitoramento/verificação do uso em muitas partes da organização.	Resultados positivos mensuráveis na maior parte da organização. Resultados positivos consistentemente ao longo do tempo em muitas partes da organização.
8	Qualificado	Total comprometimento da gerência. Maioria da gerência é proativa. A prática está estabelecida como parte integrante do processo e todos os envolvidos estão treinados. Há itens de apoio, ex. processos e padrões incentivam e facilitam o uso da prática.	Implantada em quase todas as partes da organização. Uso consistente em quase todas as partes da organização. Monitoramento/verificação do uso em quase todas as partes da organização.	Resultados positivos mensuráveis em quase todas as partes da organização. Resultados positivos consistentemente ao longo do tempo em quase todas as partes da organização.
10	Excelente	Gerência fornece uma liderança dedicada e comprometida. Excelência organizacional na prática reconhecida até mesmo fora da organização.	Implantação difundida e consistente em todas as partes da organização. Uso consistente ao longo do tempo em toda a parte da organização. Monitoramento/verificação por todas as partes da organização.	Requisitos excedem. Resultados consistentemente "world-class".

Figura 2.6: Avaliação por Objetivos na KITTool (HÖHN, 2011).

Uma outra maneira de avaliar o processo de teste que está em uso, utilizando a KITTool, é realizar a avaliação por *práticas*. Na Figura 2.7 é mostrada a tela de avaliação por meio das práticas. Nesse modo o usuário seleciona a área de processo que será avaliada e a ferramenta apresenta os objetivos específicos com suas respectivas práticas. Para cada uma dessas práticas o usuário deve atribuir uma nota, que é um valor percentual atribuído de acordo com o quanto a prática já foi implementada no processo. Essa avaliação também fornece a tabela de referência para a atribuição das notas, como pode ser visto na parte inferior da Figura 2.7.

Em resumo, o trabalho desenvolvido por Höhn (2011) facilita a compreensão do TMMi e aquisição de conhecimento com relação ao teste de software, e também auxilia empresas na compreensão de seu próprio processo de teste, permitindo que esta tenha claro quais práticas são realmente implementadas. No contexto deste trabalho, o trabalho de Höhn (2011) foi usado para melhor compreensão do TMMi e para realização de um *survey* para identificar quais práticas

Objetivo: SG1 Estabelecer política de teste

Práticas	Nota (%)
SP 1.1 Definir objetivos de teste	0
SP 1.2 Definir a política de teste	0
SP 1.3 Distribuir política de teste aos interessados	0

Fechar Cancelar ? Voltar Avançar

Tabela de referência

Porcentagem	Enfoque e Aplicação no Processo
0%	Sem metodologia/procedimento ou atividades que atendam a prática solicitada.
25%	a – Não existe uma metodologia ou procedimento definido.
	b – Algumas práticas são implementadas, mas não em todos os projetos. (<30%)
	c – Não são atendidos todos os requisitos da prática solicitada (<70%).
50%	a – Existe uma metodologia ou procedimento definido, mas não documentado.
	b – Algumas práticas são implementadas, mas somente em alguns projetos definidos como críticos ou estratégicos para a empresa. (30% a 80%)
	c – Não são atendidos todos os requisitos da prática solicitada (<70%).
75%	a – Existe uma metodologia ou procedimento definido e documentado.
	b – Algumas práticas são implementadas na maioria dos projetos. (> 80%)
	c – São atendidos, em sua maioria, os requisitos da prática solicitada (70% a 99%).
100%	a – Existe uma metodologia ou procedimento definido e documentado.
	b – As práticas são implementadas em todos os projetos.
	c – São atendidos todos os requisitos da prática solicitada.

Figura 2.7: Avaliação por práticas na KITTool (HÖHN, 2011).

sugeridas pelo TMMi podem ser priorizadas numa primeira implementação de um processo de teste em uma pequena empresa, como será visto no Capítulo 3. O trabalho apresentado nessa seção também foi utilizado na realização do diagnóstico do processo de teste utilizado pela empresa, na qual foi realizado um estudo de caso, como será apresentado no Capítulo 4.

## 2.5 Considerações Finais

Neste capítulo apresentaram-se os principais conceitos relacionados à atividade de teste e aplicação em no processo de desenvolvimento do produto de software. As atividades de V&V, quando utilizadas de forma bem planejada e sistemática, agregam valor ao produto, resultando em maior satisfação do cliente e aumento da confiança de que sua solicitação foi atendida. Uma das principais atividades de V&V é a de teste de software, que foi sintetizada neste capítulo. O uso de ferramentas automatizadas para apoiar as atividades de teste garante que o processo possa ser executado de forma mais eficiente, tornando-o menos propenso a erros.

O processo de teste foi abordado neste capítulo de forma a apresentar os principais conceitos relacionados. Um processo bem definido pode ser o fator determinante do sucesso de um

produto de software. Qualquer produto que se deseje desenvolver requer um bom planejamento para que se possa gerenciar bem tanto recursos financeiros quanto tempo. A etapa de planejamento possui grande importância para o teste, pois é nesse momento que se definem itens como estratégias, recursos, tarefas, objetivos a serem alcançados, cronograma, e quais partes do software serão testadas. Ressalta-se que definir as partes do software a serem testadas é de grande importância para a atividade de definição dos casos de teste, já que ajuda a dimensionar o esforço da atividade de teste. Outro item importante é a estratégia de teste a ser utilizada, pois permite uma seleção de critérios que limitam os requisitos de teste e, conseqüentemente, limitam a quantidade de dados necessários para os casos de teste. Com o planejamento realizado, o projeto de casos de teste concluído e os dados de teste identificados, é possível iniciar a execução dos testes em busca de defeitos. À medida que falhas são observadas, tais falhas devem ser devidamente relatadas para que se possa tomar atitudes corretivas. Todas as fases do processo de teste são monitoradas e controladas para que o planejamento seja cumprido e que à medida que sejam necessários novos replanejamentos, esses possam ser feitos.

Para que a organização possa implantar seu processo de teste, existem alguns modelos de maturidade que indicam o que deve ser feito para que um processo de teste de qualidade seja criado. Um problema desses modelos de maturidade é a ausência de informação a respeito de como implementar cada uma das práticas sugeridas e, ainda, quais dessas práticas são prioritárias. Rodrigues, Pinheiro e Albuquerque (2010) afirmam que no cenário de pequenas empresas, poucas realizam atividades relacionadas ao teste de software, pois ainda encontram dificuldades na implantação ou execução do processo de teste. Os autores ainda afirmam que entre as principais dificuldades estão: falta de tempo; falta de orçamento; falta de especialistas; falta de ferramentas de suporte; pouca importância dada à atividade de teste; e pouco conhecimento sobre os benefícios que a realização de testes trazem.

Diante disso pode-se perceber que modelos de referência não são o guia ideal para essas empresas, de pequeno porte, implantarem processo de teste. O trabalho apresentado nesta dissertação visa a fornecer uma estratégia de elaboração de processo de teste, mas limitando-se ao contexto de pequenas empresas brasileiras. Para isso, foi necessário identificar o que é importante no TMMi segundo alguns profissionais da comunidade de testes brasileira. Com esse objetivo foi realizado um *survey* que será apresentado no próximo capítulo.



# Capítulo 3

## SURVEY

---

---

Este capítulo apresenta o survey realizado para identificar quais práticas do TMMi, que estão relacionadas a um processo genérico de teste, podem ser mais importantes em uma implementação de processo de teste em empresas de software, que desejam executar a atividade de teste de forma planejada e estruturada.

### 3.1 Considerações Iniciais

O teste de software tem ganhado cada vez mais espaço dentro das organizações, porém, ainda é visto como um processo caro, que demanda mais tempo que o planejado. Muitas organizações compreendem que a melhoria de processo pode solucionar esses problemas; no entanto, na prática pode ser difícil saber quais medidas podem ser tomadas para melhorar e controlar o processo de teste e em que ordem essas medidas devem ser implementadas (ANDERSIN, 2004). Modelos de referência e *frameworks*, como o TMMi, apontam o que pode ser feito para que o processo de teste possa ser melhorado, mas não indicam como fazê-lo. Apesar de ser organizado em níveis como o CMMI, e ser implantado de maneira incremental começando pelo nível mais baixo, o TMMi não indica quais áreas são prioritárias e podem ser implantadas primeiro, ou mesmo quais áreas podem deixar de ser implantadas caso o objetivo não seja obter a certificação.

Este capítulo relata um estudo que foi realizado com o propósito de identificar quais são as práticas mais importantes do TMMi e que devem ser realizadas prioritariamente na execução de um processo de teste, segundo o ponto de vista de pessoas que trabalham há pelo menos três anos com teste de software em empresas que possuem um processo formal implantado. Acredita-se que existam atividades básicas relacionadas a cada fase de um processo de teste que não devem ser deixadas de lado, mesmo que o orçamento, tempo ou equipe sejam escassos.

Ressalta-se que um processo genérico de teste deve incluir as fases de planejamento, projeto de casos de teste, execução e análise, e acompanhamento dos testes (CRESPO et al., 2010; HASS, 2008).

Para a realização deste trabalho optou-se pelo método de investigação do tipo *survey*. Um *survey* é uma estratégia de investigação que permite a coleta de informações quantitativas e qualitativas por meio de questionários respondidos por uma amostra representativa da população que se deseja investigar. Os resultados do *survey* devem ser analisados para obter conclusões descritivas e explanatórias (WOHLIN, 2000). O questionário foi elaborado com base no Kit-Map, desenvolvido por Höhn (2011), apresentado no capítulo anterior.

Neste capítulo apresentam-se o *survey* realizado e os resultados observados. Na Seção 3.2 é apresentado o planejamento feito para a realização do *survey*, a forma de divulgação e os métodos de avaliação dos dados obtidos. Na Seção 3.3 são apresentados os resultados observados e o perfil dos respondentes. Em seguida, na Seção 3.4 discutem-se os resultados observados e as dependência entre as práticas presentes nos resultados. Dando sequência, na Seção 3.5 são levantadas as possíveis ameaças à validade do *survey*; na Seção 3.6 apresentam-se as conclusões obtidas a partir desses resultados e as considerações finais deste capítulo.

## 3.2 Descrição do Survey

Nesta seção serão apresentados a forma como o *survey* foi planejado, tornando viável sua aplicação e de forma a facilitar a posterior análise dos dados. Ainda nesta seção também serão apresentados uma breve descrição da amostra obtida e a forma como essa amostra foi analisada para se chegar ao resultado apresentado na Seção 3.3.

### 3.2.1 Projeto do Survey

O questionário foi projetado com o apoio da ferramenta *Lime Survey* (TEAM, 2011) a qual permite organizar as questões em grupos e visualizá-las em páginas Web distintas. Para a confecção do questionário foi feita a tradução do TMMi e utilizado o agrupamento das práticas feito por Höhn (2011), apresentado no Capítulo 2 (Seção 2.4). O questionário foi dividido em seis grupos de questões. O primeiro grupo, relacionado ao perfil dos participantes, teve como objetivo identificar a experiência e o conhecimento, dos participantes, em modelos de referência, sendo eles o CMMI, o MPS.Br e o TMMi. Cada um dos cinco grupos restantes representam uma das fases do processo genérico de teste definido por Höhn (2011). São eles:



- A. Planejamento
- B. Projeto de Casos de Teste
- C. Configuração de Dados e de ambiente
- D. Execução e Análise
- E. Monitoramento e Controle

Cada página do questionário apresentava um único grupo de questões. Na primeira página optou-se por incluir, juntamente com o primeiro grupo, algumas instruções para o preenchimento do questionário e uma tabela contendo a escala que deveria ser utilizada na resposta às questões. A escala é apresentada na Tabela 3.1. Essa escala não forneceu valor neutro com o objetivo de forçar o respondente a tomar um posicionamento entre as alternativas opostas de resposta. Todas as questões relacionadas ao TMMi eram obrigatórias, de forma que o participante não poderia se abster de nenhuma resposta relacionada ao processo teste, diferentemente de algumas questões de perfil, que não eram obrigatórias. Os participantes não foram informados a respeito da construção do questionário ter sido feita com base no TMMi. Também foi solicitado ao participante que respondesse o questionário de acordo com sua opinião pessoal independentemente de influências de seu ambiente de trabalho.

**Tabela 3.1: Escala de notas de acordo com a importância**

<b>1- Dispensável</b>	Atividade dispensável que não precisa ser realizada.
<b>2- Opcional</b>	Atividade que não necessariamente precisa ser realizada.
<b>3- Desejável</b>	Atividade que deve ser realizada sempre que possível.
<b>4- Mandatória</b>	Atividade essencial que deve sempre ser realizada.

Para a construção do questionário foi feita uma tradução adaptada, para o jargão técnico em Português, dos objetivos e práticas específicas presentes no TMMi. Cada pergunta do questionário era um objetivo específico, que no TMMi possui práticas específicas relacionadas. Essas práticas eram as opções de cada pergunta, nas quais o respondente poderia atribuir uma nota, de 1 a 4 conforme a escala da Tabela 3.1, de acordo com sua opinião pessoal em relação à importância daquela opção (prática) para a satisfação da pergunta descrita (objetivo específico). No TMMi, o objetivo específico é alcançado com a realização do conjunto de suas práticas relacionadas. Na Figura 3.1 pode-se visualizar um exemplo de questão para o objetivo *Realizar avaliação de risco do produto*.

Nessa questão observa-se que há três respostas, que correspondem às práticas daquele objetivo, e para cada uma dessas respostas o participante deve atribuir uma nota de 1 a 4 conforme

\* 12 Para realizar uma análise de risco, qual a importância das atividades abaixo?

	1	2	3	4
Definir categorias e riscos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Identificar riscos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Analisar riscos	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

**?** Realizar uma análise de risco de produto significa: (i) definir uma lista de riscos ao produto; (ii) classificar por categorias; (iii) definir prioridades de gerenciamento de acordo com o impacto de um risco no sucesso do projeto e a probabilidade de sua ocorrência.

**Figura 3.1: Exemplo da estrutura de questão utilizada no survey para o objetivo *Realizar avaliação de risco do produto*.**

sua importância para a realização da avaliação de risco do produto, nesse caso. A Figura 3.2 mostra outro exemplo de questão do formulário do survey. Essa questão apresenta o objetivo *Desenvolver um plano de teste*, que possui cinco práticas associadas. As notas da escala de importância são atribuídas à cada uma dessas práticas.

\* 15 Para desenvolver um plano de teste, qual a importância das atividades abaixo?

	1	2	3	4
Estabelecer o cronograma de teste	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Planejar equipe de teste	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Planejar envolvimento dos interessados	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Identificar riscos do projeto de teste	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Estabelecer plano de teste	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

**?** Desenvolver o plano de teste requer: (i) estabelecer o cronograma para as atividades propriamente dito; (ii) estabelecer a equipe de acordo com as atividades definidas e o envolvimento de interessados (stakeholders); (iii) identificar necessidades de treinamento; (iv) identificar a probabilidade e o impacto de fatores que possam comprometer o andamento do projeto de testes, e estabelecer medidas de contingência; (v) criar o documento que contém o plano de teste.

**Figura 3.2: Exemplo da estrutura de questão utilizada no survey para o objetivo *Desenvolver um plano de teste*.**

O primeiro grupo de questões do formulário teve como objetivo identificar o perfil do participante levando em consideração seu ambiente de trabalho. A Figura 3.3 mostra parte do formulário de perfil, apresentando as questões relacionadas a experiência e maturidade do processo de teste na empresa. Considerou-se que a experiência e o grau de maturidade em que se encontra a empresa na qual o participante está inserido podem afetar seu conhecimento sobre teste. Por essa razão as seguintes perguntas foram feitas, além de informações de contato:

- Faixa etária - Para classificação por idade dos participantes;
- Grau de instrução - Pergunta não obrigatória, apenas para classificação;
- Tempo de experiência em teste de software (empresa, academia e docência) - Sabe-se que o conhecimento tácito é diferente do conhecimento explícito, por isso a diferenciação entre pessoas com experiência em empresas de software na área de teste e pessoas com experiência somente em pesquisa ou docência;

- Processo de teste na empresa - Para os respondentes com experiência em empresa de software foi perguntado se havia processo de teste implantado;
- Certificação em Modelos de Maturidade - Para os respondentes com experiência em empresa de software foi perguntado se a empresa possui alguma certificação em modelos de maturidade, e para os que responderam sim foi perguntado qual a certificação (MPS.Br, CMMI e TMMi) e o nível. Considerou-se que se a empresa possui processo de teste e/ou alguma certificação, como CMMI ou MPS.Br, afeta a maturidade de quem responde às perguntas;
- Conhecimento em relação ao modelos de maturidade TMMi e MPS.Br - O conhecimento de modelos de referência, especialmente o TMMi que é completamente voltado para o teste de software, confere um maior grau de maturidade em relação ao processo de teste.

<b>* 4 Possui quanto tempo de experiência em Teste de Software? (em anos)</b>				
	Não possuo	Até 1 ano	De 1 a 3 anos	Mais de 3 anos
Experiência na academia (pesquisa)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Experiência em empresas de software	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Docência	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

**5 A empresa que você trabalha/trabalhou possui um processo de testes implantado?**

Sim     Não     Sem resposta

**6 A empresa na qual você trabalhou possui certificação em algum modelo de maturidade de processo?**

Sim     Não     Sem resposta

**\* 7 Qual certificação?**  
Escolha uma das seguintes respostas:

MPS.BR  
 CMMI  
 TMMI  
 Outra. Qual?

**\* 9 Qual nível?**  
Escolha uma das seguintes respostas:

Nível 1  
 Nível 2  
 Nível 3  
 Nível 4  
 Nível 5

Figura 3.3: Parte do formulário do survey com algumas das questões de perfil.

O questionário foi disponibilizado por meio de um *link*<sup>1</sup> e divulgado por *e-mail* entre alguns especialistas da academia e da indústria de software, e em uma comunidade virtual que reúne profissionais de teste, e possui mais de três mil participantes cadastrados<sup>2</sup>. O questionário também foi distribuído entre algumas empresas de software da cidade de Ribeirão Preto, no estado de São Paulo, pertencentes ao PISO (Pólo Industrial de Software)<sup>3</sup>.

### 3.2.2 Amostra Obtida

O *survey* ficou disponível por 45 dias, período no qual houve 113 acessos e desses 39 responderam todas as perguntas do questionário. Após esse período o questionário foi desativado, dessa forma impedindo que novas respostas pudessem ser enviadas. Foram consideradas para análise somente as respostas dos participantes que responderam a todas as questões obrigatórias do questionário. Como a ferramenta utilizada oferecia a opção “salvar e continuar depois”, muitas respostas ficaram incompletas.

A amostra obtida é pequena, em relação ao número de convidados a responderem o questionário. Entretanto, esse número ainda permitiu que os dados fossem analisados por meio de métodos estatísticos, mas com um nível de confiança menor que o tradicionalmente utilizado (isso é discutido mais adiante nesta seção). A seguir discute-se a forma de análise dos dados.

### 3.2.3 Forma de Análise dos Dados

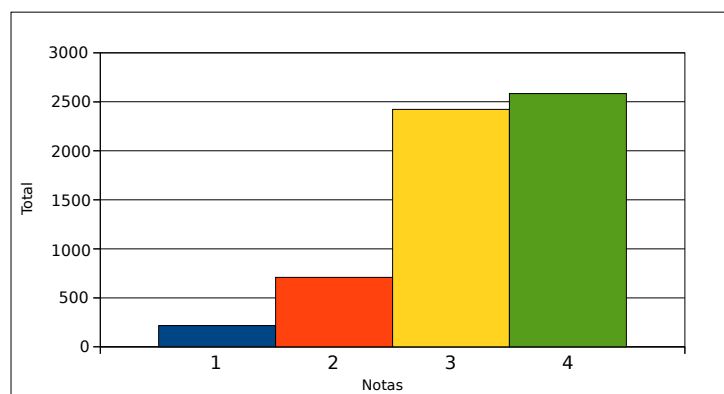
Uma análise inicial foi feita e notou-se que grande parte das respostas concentravam-se nas escalas 3 e 4, como pode ser observado no gráfico da Figura 3.4, que mostra o total de respostas para cada nota da escala. Essa análise inicial também permitiu a identificação de duas respostas discrepantes. A primeira possuía todas as práticas assinaladas com valor 4 (quatro). Na segunda, o participante informou que a escala foi utilizada de forma invertida, ou seja, atribuiu-se o valor 1 (um) para práticas consideradas mandatórias. Ambas as respostas foram consideradas *outliers* para os objetivos deste estudo, e foram removidas do conjunto final. Assim sendo, restaram 37 respostas para serem analisadas.

Os dados obtidos foram inicialmente analisados por meio de um teste não-paramétrico, por terem nível de mensuração ordinal e uma distribuição não simétrica, pois a maioria das respostas estavam concentradas em 3 ou 4, conforme mostrado na Figura 3.4. Neste *survey*, as variáveis independentes são: experiência em empresa com processo de teste, conhecimento e

<sup>1</sup><http://amon.dc.ufscar.br/limesurvey/index.php?sid=47762&lang=pt-BR>

<sup>2</sup><http://br.dir.groups.yahoo.com/group/DFTestes> - acessado em 09/04/2012

<sup>3</sup><http://www.piso.org.br/> - acessado em 09/04/2012.



**Figura 3.4: Distribuição de frequência das escalas obtidas nas respostas ao questionário.**

utilização do MPS, e conhecimento do TMMi. A variável dependente é a escala de importância atribuída a cada questão. O teste escolhido foi o teste de sinal (do inglês, *Sign Test*) para uma amostra (WHITLEY; BALL, 2002). O teste do sinal avalia se a mediana, calculada para uma determinada questão, é maior que um valor fixo. Neste estudo foi escolhido o valor fixo de 3,5, pois é o valor próximo da mediana. Assim, com base nesse valor, foi possível identificar quais das questões tiveram respostas efetivamente mandatórias, o que aconteceria caso a mediana dessas questões fosse maior que o valor fixo, significando que mais de 50% dos entrevistados atribuíram nota 4 àquela questão. Foi utilizado o nível de confiança (*p-value*) de 0,15 devido ao tamanho da amostra. Apesar de não ser um nível de confiança comum, observa-se que para amostras pequenas e em estudos exploratórios alguns pesquisadores (MILLER, 2004; BASILI; REITER, 1981) utilizam níveis de confiança menos rigorosos do que os tradicionais níveis de 0,01 ou 0,05.

Observando os resultados dessa análise percebeu-se que, para alguns casos, não foi possível encontrar significância estatística mesmo quando a maioria dos participantes responderam 3 e 4 para uma dada questão. Por exemplo, a prática *Identificar e priorizar dados de teste* foi classificada como mandatória pela maioria (19 respostas nota 4), mas na análise estatística essa prática não está presente nos resultados por não apresentar significância estatística. A escala utilizada e o tamanho da amostra podem ter influenciado a sensibilidade do teste estatístico, fazendo com que mesmo algumas práticas que obtiveram grande diferença na quantidade de respostas 3 e 4 não estivessem presentes no resultado final. Apesar de a estatística não paramétrica não ser exigente nem no tipo de distribuição, nem nos tamanhos das amostras, o teste estatístico utilizado mostrou-se pouco sensível a algumas das diferenças observadas na amostra. Por essa razão, algumas práticas foram excluídas. Por exemplo, a prática *Identificar e priorizar condições de teste* não apresentou diferença estatística significativa. Na Tabela 3.2 apresenta-se a quantidade de cada nota que essa prática recebeu e, como pode ser visto, o número de respostas nota

3 é maior que o número de respostas nota 4 ( $19 > 14$ ), e mesmo assim não houve diferença estatística significativa, portanto essa prática não esteve presente nos resultados.

**Tabela 3.2:** Tabela com quantidade de cada nota, da escala de 1 a 4, atribuída à prática *Identificar e priorizar condições de teste*

Nota	Quantidade
4	14
3	19
2	2
1	2

Após a análise dos resultados da análise estatística foi planejada uma segunda pesquisa para verificação de alguns pontos que mereciam maior atenção. Foram elaboradas três perguntas do tipo “sim ou não”, com base nas dependências não resolvidas observadas nos resultados. Essas dependências, identificadas por Höhn (2011), indicam que para a implementação de algumas práticas é necessário antes que outras tenham sido implementadas. A Tabela de Dependências, presente no Anexo A, lista as práticas que possuem dependências. Essa pesquisa foi enviada para todos os respondentes da primeira pesquisa, e ficou disponível por duas semanas.

A análise das frequências das respostas permite perceber que há algumas tendências na amostra que estatisticamente não foi possível esclarecer. Nada se pode afirmar sobre a redução de práticas do TMMi observando apenas o resultado estatístico, pois algumas das práticas que tiveram maior quantidade de respostas como “mandatória” não estão presentes. Diante disso, optou-se por realizar uma análise descritiva da amostra, baseada nas frequências das escalas. O objetivo foi identificar quais práticas possuíam maior frequência de respostas como “mandatória” em comparação com as outras respostas da escala (desejável, opcional ou descartável). Para isso, foi calculado o total de respostas para cada nota da escala.

Dessa forma, foram identificadas quais práticas foram classificadas como mandatórias pela maioria dos respondentes e assim chegou-se a um conjunto de práticas semelhante ao encontrado na análise estatística, porém, com algumas práticas a mais. Esses resultados serão apresentados na seção a seguir, sendo que o resultado encontrado na análise estatística poderá ser observado no diagrama de *Venn* da Figura 3.6 (Seção 3.3.2).

### 3.3 Resultados

Os resultados serão apresentados em duas partes, para facilitar a compreensão. Primeiramente serão apresentados os dados de caracterização dos participantes da amostra obtida, ressaltando-se os perfis gerados a partir dessa caracterização. Em seguida serão apresentados

os resultados relativos à obrigatoriedade das práticas do TMMi para cada um dos perfis mencionados anteriormente.

### 3.3.1 Determinação dos Perfis com Base no Nível de Conhecimento dos Respondentes

Como apresentado anteriormente, a análise inicial das respostas permitiu a identificação de duas respostas discrepantes que foram excluídas da amostra de 39 respondentes do *survey*. Após essa identificação, foram caracterizados os níveis de conhecimento do conjunto final, de 37 respondentes. A Figura 3.5 sumariza os resultados oriundos da caracterização dos níveis de conhecimento, que serão descritos a seguir.

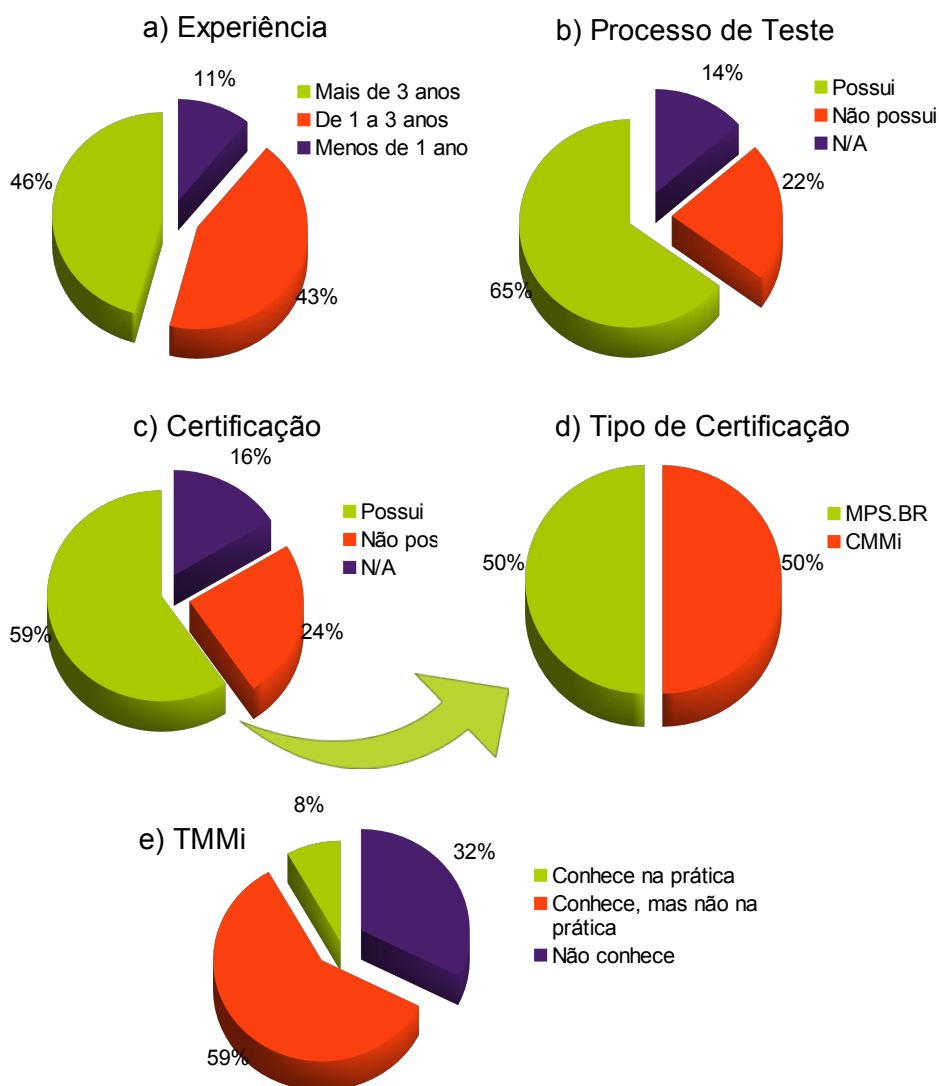


Figura 3.5: Perfis obtidos na amostra

- **Experiência** - O primeiro gráfico, intitulado **Experiência**, mostra que dos 37 profissionais que responderam o questionário, 46% (ou seja, 17 deles) possuem mais de três anos de experiência com teste em empresas de software, enquanto 43% (16 respondentes) possuem de um a três anos de experiência e apenas 11% (4 respondentes) possui menos de um ano de experiência.
- **Processo de Software** - O gráfico intitulado **Processo de Software** mostra que 65% dos respondentes (isto é, 24 de um total de 37) trabalham com um processo de teste implantado, enquanto 22% (8 respondentes) atuam em uma organização que não possui processo de teste formalmente implantado. Observa-se ainda que 14% (ou seja, 5 respondentes) não responderam essa questão, já que esta não era uma questão obrigatória por se tratar de informação que pode ser considerada sigilosa por algumas empresas.
- **Certificação** - O gráfico **Certificação** revela que 22 respondentes trabalham em uma organização que possui um certificado de maturidade de processo de software, o que representa 59% do total de respondentes. Outros 24% (9 respondentes) não atuam em organizações certificadas 16% (6 respondentes) não responderam a essa questão.
- **Tipo de Certificação** - Dos 22 respondentes que afirmaram trabalhar em uma organização com certificação de maturidade para processo de software, metade deles atua em organizações que possuem certificação CMMi em algum nível, enquanto a outra metade atua em organizações certificadas pelo modelo MPS.Br. Essas informações estão sumarizadas no gráfico **Tipo de Certificação** da Figura 3.5.
- **TMMi** - Por fim, o gráfico **TMMi** mostra que apenas 8% dos respondentes (isto é, apenas 3 deles) já utilizaram o TMMi na prática, enquanto 59% (22 respondentes) afirmam conhecer o modelo de maturidade de processo de teste e 32% (12 respondentes) desconhecem o modelo.

Com base nos resultados da caracterização do nível de conhecimento dos respondentes do questionário, conclui-se que, em geral, eles possuem um nível adequado para os objetivos traçados para o estudo reportado neste capítulo. Dentre os 37 respondentes, 89% possuem algum nível de experiência com teste de software, 65% trabalham em uma organização que possui um processo de teste implantado, 59% trabalham em uma organização certificada com CMMi ou MPS.Br e 67% conhecem o TMMi, com alguns já tendo utilizado-o na prática. O nível das empresas certificadas em CMMi são 2, 3 e 5. Já o nível das empresas certificadas em MPS.Br varia entre E, F e G. Para esses níveis do MPS.Br, a área de processo que envolve teste de software não está presente (MPS.BR, 2011b), por isso não foi levado em consideração na análise



estatística. No caso do CMMi o tamanho da amostra de respondentes com essa certificação é muito pequeno e por essa razão não houve resultado com diferença estatística significativa. A seguir, apresentam-se os perfis que foram considerados para análise dos resultados.

Conforme poderá ser observado na discussão apresentada na próxima seção, as respostas foram agrupadas conforme um conjunto pré-definido de três perfis e o Grupo-Todo. Esses perfis são:

- *Perfil-Especialista*– Constituído por profissionais com mais de três anos de experiência com teste em empresas de software e que já trabalharam com algum processo de teste formalmente implantado;
- *Perfil-MPS.Br*– Constituído por profissionais que conhecem e utilizam o MPS.Br;
- *Perfil-TMMi*– Formado por profissionais que conhecem o TMMi;
- Grupo-Todo – Todas as 37 respostas obtidas.

Ressalta-se que esses grupos foram selecionados pois, em geral, espera-se que o grau de conhecimento tácito, acerca de processos e de teste de software, seja muito maior que o grau de conhecimento dos respondentes que não se encaixam nesses perfis.

### 3.3.2 Caracterização da Importância das Práticas do TMMi

A descrição dos resultados e as respectivas análises apresentadas a seguir são baseadas no Grupo-Todo e nos três perfis definidos na seção anterior, sendo eles *Perfil-Especialista*, *Perfil-MPS.Br* e *Perfil-TMMi*. Foi observado em cada perfil quais as práticas tiveram maioria de respostas mandatórias, ou seja, nota 4. A Figura 3.6 mostra um diagrama de *Venn* desses perfis com as práticas classificadas como mandatórias pela maioria em cada perfil. Nesse diagrama, cada losango representa uma prática, o número em seu interior identifica qual é a prática, de acordo com a Tabela 3.3 e losangos de cor azul mostram quais práticas estão presentes também no resultado da análise estatística, a qual teve seu resultado complementado com a análise descritiva devido ao fato de que a análise não permitiu afirmar nada sobre a redução do TMMi, conforme descrito na seção anterior. Losangos destacados em cinza indicam quais práticas constituem o resultado apresentado na Seção 3.4. Na Tabela 3.3 é possível identificar quais práticas estão no diagrama. As elipses permitem identificar quais práticas foram selecionadas por quais perfis. Observe que além dos três perfis caracterizados anteriormente, uma quarta elipse representa o Grupo-Total que contém todas as respostas consideradas na análise. A seguir comentam-se os resultados da Figura 3.6.

- **Grupo-Todo** - Na análise de todo o conjunto de respostas, 31 práticas receberam nota 4 da maioria dos participantes. A Figura 3.6 mostra quais práticas pertencem à análise do conjunto todo. Algumas dessas práticas também estão presentes nos resultados de outros perfis, como pode ser visto nas intersecções do diagrama. O resultado apresentado na próxima seção é constituído das respostas desse conjunto mais duas práticas que foram incluídas após a realização de uma segunda pesquisa apresentada na próxima seção.
- **Perfil-Especialista** - A maior parte dos especialistas atribuiu a 49 práticas nota 4, sendo que destas 27 possuem intersecção com outros perfis, como mostrado na parte central do diagrama na Figura 3.6, e 10 práticas foram selecionadas somente por este perfil.
- **Perfil-MPS.Br** - Para os que conhecem e utilizam o MPS.Br, 33 práticas estão presentes nos resultados, sendo que 3 práticas foram classificadas como mandatórias somente por este perfil. O restante possui intersecção com outros perfis, como mostrado na Figura 3.6.
- **Perfil-TMMi** - Para quem conhece o TMMi, 42 práticas são mandatórias, sendo que a prática *Monitorar Compromissos do Teste* foi selecionada exclusivamente por esse perfil. O restante das práticas está em intersecções com os outros perfis.

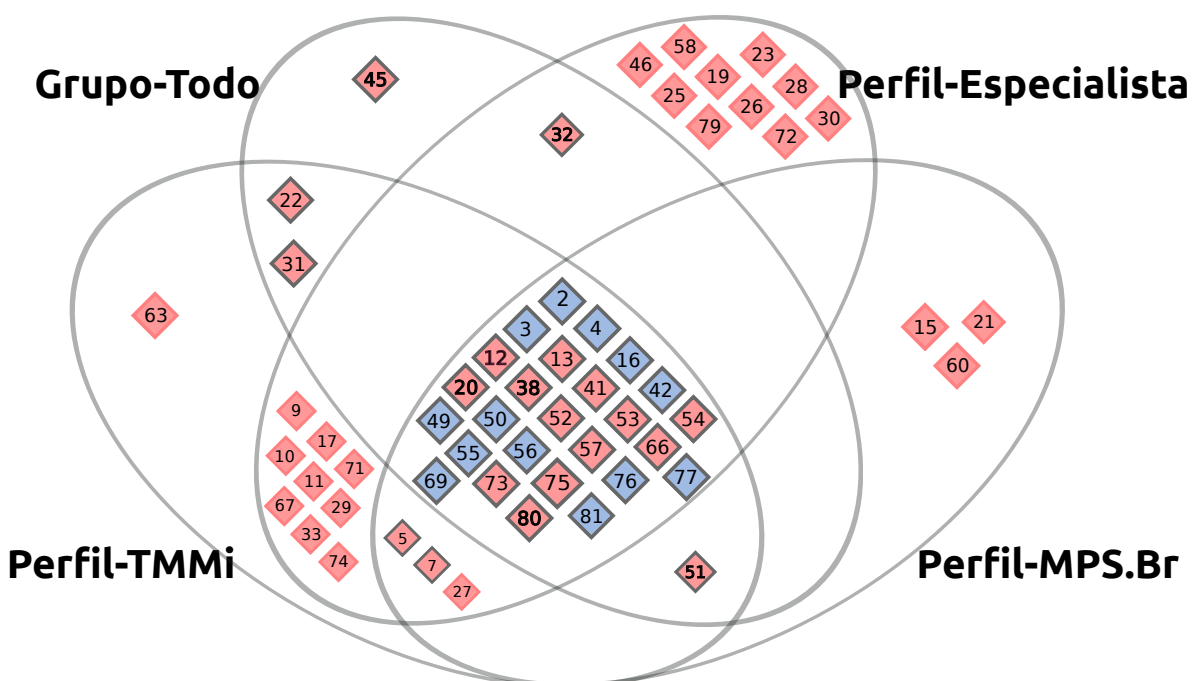


Figura 3.6: Diagrama de Venn para os quatro perfis descritos apresentando as intersecções dos resultados e práticas que foram identificadas como mandatórias pela maioria dos respondentes. Itens destacados em cinza indicam as práticas que compõem o processo reduzido apresentado na Seção 3.4

Nº	Prática	Nº	Prática
2	Identificar riscos do produto	42	Implementar o ambiente de teste
3	Analisar riscos do produto	45	Realizar o teste <i>intake</i> (pré-teste) do ambiente
4	Identificar elementos e características a serem testados	46	Desenvolver e priorizar os procedimentos de teste não funcional
5	Definir abordagem de teste	49	Executar casos de teste
7	Definir critérios de parada	50	Relatar incidentes de teste
9	Estabelecer uma lista de tarefas de alto nível	51	Escrever <i>log</i> de teste
10	Definir ciclo de vida de teste	52	Decidir sobre incidentes com o grupo de controle de configuração
11	Determinar estimativas de esforço e custo de teste	53	Executar ações apropriadas para corrigir os incidentes de teste
12	Estabelecer o cronograma de teste	54	Acompanhar o status dos incidentes de teste
13	Planejar a equipe de teste	55	Executar casos de teste não funcional
15	Identificar riscos ao projeto de teste	56	Relatar incidentes de teste não funcional
16	Estabelecer o plano de teste	57	Escrever <i>log</i> de teste não funcional
17	Revisar o plano de teste	58	Conduzir revisões por pares
19	Obter comprometimento com o plano de teste	60	Analisar dados de revisão por pares
20	Obter necessidades do ambiente de teste	63	Monitorar compromissos de teste
21	Desenvolver os requisitos do ambiente de teste	66	Conduzir revisões do progresso do teste
22	Analisar requisitos do ambiente de teste	67	Conduzir revisões em marcos do progresso de teste
23	Identificar riscos não funcionais do produto	69	Monitorar Defeitos
25	Identificar características não funcionais a serem testadas	71	Monitorar critérios de parada
26	Definir abordagem de teste não funcional	72	Monitorar critérios de suspensão e recomeço
28	Identificar produtos de trabalho que devem ser revisados	73	Conduzir revisões de qualidade do produto
29	Definir critérios para revisão por pares	74	Conduzir revisões de marcos da qualidade do produto
30	Identificar e priorizar condições de teste	75	Analisar problemas
31	Identificar e priorizar casos de teste	76	Tomar ação corretiva
32	Identificar dados de teste específicos necessários	77	Gerenciar ação corretiva
33	Manter rastreabilidade horizontal com requisitos	79	Realizar o gerenciamento dos dados de teste
38	Desenvolver e priorizar os procedimentos de teste	80	Coordenar a disponibilidade e o uso dos ambientes de teste
41	Desenvolver cronograma de execução de teste	81	Relatar e gerenciar incidentes do ambiente de teste

**Total de Práticas = 56**

**Tabela 3.3: Práticas que estão presentes no diagrama de Venn da Figura 3.6.**

### 3.4 Discussão

Após a análise dos resultados foi realizada uma segunda pesquisa para verificação de alguns pontos que mereciam maior atenção. Foram elaboradas três perguntas do tipo “sim ou não”, com base nas dependências não resolvidas observadas nos resultados. Essas dependências, identificadas por Höhn (2011), indicam que para a implementação de algumas práticas é necessário antes que outras tenham sido implementadas. A Tabela de Dependências, presente no Anexo A, lista as práticas e suas respectivas dependências. Essa pesquisa foi enviada para os 37 respondentes da primeira pesquisa, ficou disponível por duas semanas e foram obtidas 14 respostas. As dependências não solucionadas com a análise dos resultados da pesquisa anterior se referem a três práticas: *Analisar riscos do produto*, *Definir a abordagem de teste* e *Definir critérios de parada*.

Foi perguntado se a análise de risco de produto deve ser feita como parte da atividade de teste. O resultado encontrado foi: 12 respondentes concordam que a análise de riscos é realizada como parte da atividade de teste. Portanto, essa observação reforça o fato de a comunidade de testes concordar que a análise de risco de produto é importante para o sucesso do projeto de teste, pois é por meio dessa análise que se podem definir os casos de teste e priorizá-los. A prática *Analisar riscos de produto* já estava presente nos resultados, mas percebeu-se a necessidade de esclarecer se houve a compreensão por parte dos respondentes que essa análise é realizada como parte das atividades de teste. Portanto essa prática foi mantida no resultado final. Também foi perguntado aos participantes se uma vez que os riscos de produto tenham sido analisados e elementos e características a serem testados tenham sido identificados, se o entendimento é que a abordagem de teste estava plenamente definida, já que a prática *Definir a abordagem de teste* não aparece nos resultados. O resultado observado é que 10 dos 14 respondentes não consideram que a abordagem de teste esteja plenamente definida apenas com os riscos de produto analisados e características a serem testadas identificadas. Com isso, decidiu-se por incluir essa prática no resultado. Dessa forma, pode-se garantir que essa fase ficará mais completa e obedecerá às dependências entre as práticas do TMMi identificadas por Höhn (2011). A prática *Definir critério de parada* representa a definição de um item importante para a realização da atividade de teste. Como essa prática não apareceu no resultado geral, foi perguntado, nessa segunda pesquisa, se é possível executar um processo de teste sem que haja um critério de parada explicitamente definido. O resultado observado foi que a maior parte dos respondentes, em torno de 65% (9 participantes), concorda que não é possível executar um processo de teste sem um critério de parada definido. Por essa razão, essa prática foi incluída no resultado final.

O mapa mental feito por Höhn (2011) foi adaptado para permitir uma melhor visualização dos resultados em relação às práticas do TMMi, tendo sido separado de acordo com a respectiva fase do processo. A seguir são discutidos os resultados por fase do processo genérico de teste, utilizado por Höhn (2011) (vide Figura 2.4). O resultado de cada fase do processo genérico é ilustrado com um mapa mental, que possui as práticas classificadas como mandatórias pela maioria em destaque, incluindo, na fase Planejamento, as respostas da segunda pesquisa.

### 3.4.1 Planejamento

Planejar o teste de software é uma das fases mais importantes do processo de teste, pois é a fase na qual se define como será executado todo o processo para um projeto específico e que permite aos gestores acompanhar e mensurar o resultado da execução do processo. Nessa fase deve-se definir um plano de teste que inclui definições sobre cronograma, equipe, identificação do que será testado, do que não será testado e a abordagem de teste que será utilizada (IEEE, 2008). No TMMi estão relacionadas a essa fase o planejamento de testes não funcionais, definição do ambiente de teste e revisão por pares, além do planejamento de teste em si. Existem 29 práticas do TMMi, distribuídas entre 9 objetivos específicos, que estão relacionadas a essa fase. Os objetivos específicos são:

- *Realizar avaliação de risco do produto;*
- *Estabelecer abordagem de teste;*
- *Estabelecer estimativas de teste;*
- *Desenvolver um plano de teste;*
- *Obter comprometimento com o plano de teste;*
- *Desenvolver requisitos de ambiente de teste;*
- *Realizar avaliação de risco não funcional;*
- *Estabelecer abordagem para teste não funcional; e*
- *Definição de abordagem de revisão por pares.*

Como foi dito no Capítulo 2, o TMMi se divide em objetivos que são alcançados por meio da execução das práticas indicadas em cada um deles. Portanto, para atingir esses nove objetivos pretendidos para a fase de planejamento, deve-se cumprir as práticas apresentadas na Figura 3.7.

Oito das práticas em destaque nessa figura foram identificadas como mandatórias pelo Grupo-*Todo* e duas práticas foram incluídas após a realização da segunda pesquisa. Conforme definido no trabalho de Höhn (2011), e discutido no Capítulo 2 (Seção 2.4), o TMMi possui algumas dependências internas, ou seja, a realização de uma prática pode requerer a realização prévia de outras. Nessa fase do processo genérico de teste existem algumas dependências que serão discutidas ao longo desta seção.

As práticas *Identificar riscos* e *Analisar riscos* destacadas na Figura 3.7 estão relacionadas à avaliação de risco de produto. Isso mostra que é reconhecida a importância da avaliação de risco de produto, já que ela fornece dados importantes para a atividade de teste, que serão utilizados como entrada para a definição de abordagem, plano e criação de casos de teste. A TMMi Foundation (2010) divide a avaliação de risco em três práticas: *Definir categorias de riscos*, *Identificar riscos* e *Analisar riscos*. Essas práticas quando executadas geram como resultado uma lista com os riscos identificados e categorizados. A prática *Analisar Riscos* consiste na distribuição dos riscos identificados nas categorias definidas. Todos os perfis concordam que as práticas de identificação e análise de riscos são importantes para o processo de teste e devem, obrigatoriamente, fazer parte do processo.

As práticas *Identificar elementos e características a serem testadas*, *Estabelecer o cronograma de teste* e *Planejar a equipe de teste* estão diretamente relacionadas com a prática *Estabelecer o Plano de Teste*, pois alguns dos itens que devem compor um plano de teste, segundo a norma 829 do IEEE (IEEE, 2008), são: o que será testado e o que não será testado; a abordagem a ser utilizada; o cronograma; as classes e condições de teste identificadas; o critério de parada; entre outros. Quase todos os elementos básicos que constituem um plano de teste estão presentes nos resultados o que mostra coerência nas práticas que foram marcadas como mandatórias pela maioria dos participantes. A prática *Definir critério de parada* não aparece no resultado da estatística e nem possui como maioria de respostas nota 4, mas de acordo com a segunda pesquisa, essa prática é um item essencial para o planejamento de teste. Na fase de Planejamento existem ainda práticas que estão relacionadas à definição do ambiente que será utilizado durante a execução do teste. Com relação ao objetivo de desenvolver requisitos do ambiente de teste, duas práticas possuem maior frequência de notas 4. São elas: *Obter necessidades do ambiente de teste* e *Analisar requisitos do ambiente de teste*. Essas duas práticas também estão relacionadas entre si, já que primeiro é necessário levantar os requisitos e necessidades específicas de ambiente do software que está sendo desenvolvido, para então analisar os requisitos levantados e sua viabilidade de implantação.

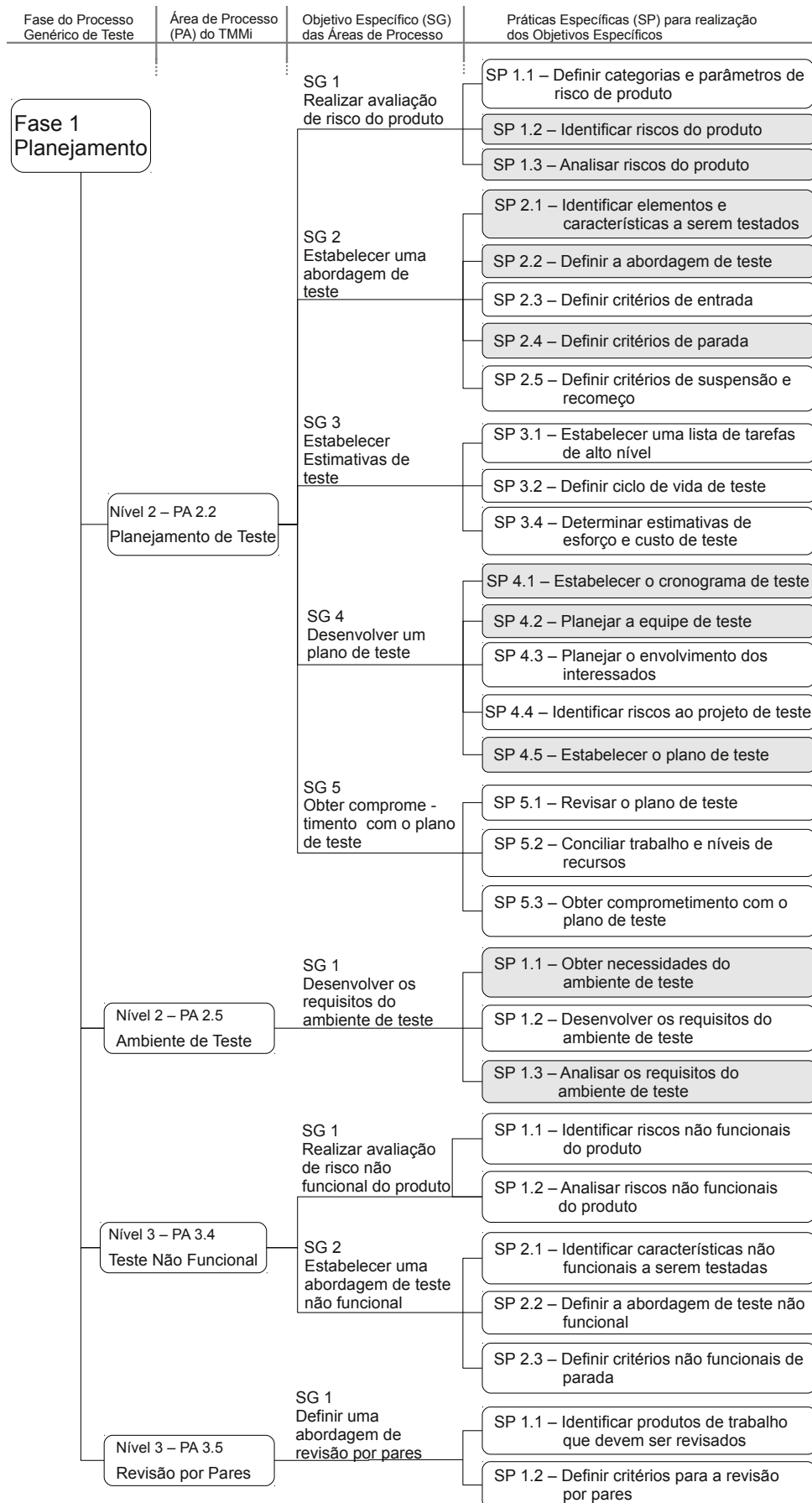


Figura 3.7: Práticas do TMMi relacionadas à fase de Planejamento do projeto de testes (adaptado de Höhn (2011)).

Ressalta-se que o questionário do *survey* não forneceu informação sobre as dependências internas identificadas por Höhn (2011) que ocorrem no TMMi (Anexo A). Para alguns casos, os resultados não necessariamente indicavam como mandatória as práticas com suas dependências. Esse é o caso da *Definição de abordagem de Teste*, que para ser realizada, precisa que antes sejam executadas as práticas *Analisar Riscos do produto* e *Identificar elementos e características a serem testados*. Nesse caso, as duas práticas que são necessárias estão presentes nos resultados, porém a definição de abordagem de teste, que é um item do plano de teste presente na norma IEEE-829 (IEEE, 2008), não aparece como mandatória quando é observado o grupo todo. Entretanto, analisando os perfis definidos (*Perfil-MPS.Br*, *Perfil-Especialista* e *Perfil-TMMi*), essa prática é classificada como mandatória para todos eles. A prática *Definir a abordagem de teste* foi incluída no resultado, pois de acordo com a segunda pesquisa, essa prática também é importante para o planejamento de teste.

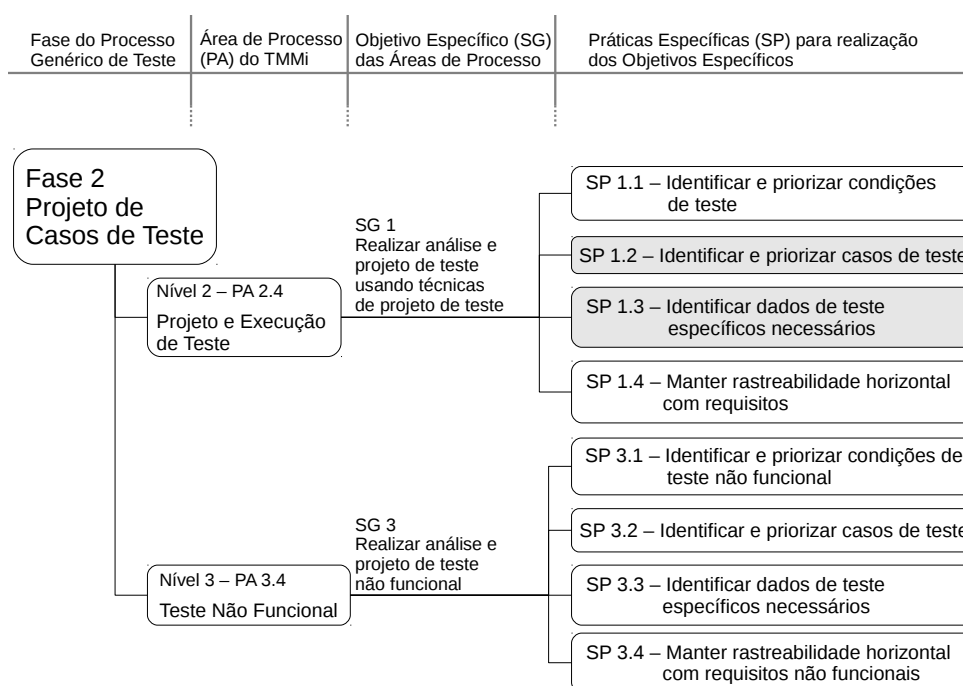
Nem todos os objetivos dessa fase são cumpridos, ou seja, nos resultados da análise do grupo todo não foram observadas práticas mandatórias relacionadas com testes não funcionais e revisões por pares. O *Perfil-Especialista* identificou como mandatórias quase todas as práticas relacionadas com esses dois objetivos, excetuando-se a prática *Analisar Riscos não Funcionais do Produto*. A prática *Definir critérios não funcionais de parada* foi classificada como mandatória pelos três perfis. Como não há nenhuma outra relacionada à definição de testes não funcionais, essa prática não foi incluída no resultado final. Mesmo sem cumprir todos os objetivos determinados pelo TMMi, pode-se observar que o planejamento inclui as principais atividades que deve compor um bom planejamento, que seja capaz de tornar o processo claro, gerenciável e mensurável. A fase seguinte do processo genérico de teste, discutida a seguir, trata do projeto de casos de teste.

### 3.4.2 Projeto de Casos de Teste

Projeto de casos de teste é a fase seguinte ao planejamento e tem como entrada o plano de teste, que possui algumas definições essenciais para essa etapa como: análise de risco, o que será testado e abordagem de teste. Na Figura 3.8 são mostradas as práticas do TMMi relacionadas com essa fase. Nela estão destacadas as duas práticas que tiveram maior frequência de respostas 4 na análise do Grupo-Todo. Como a norma IEEE-829 inclui no planejamento a definição de alguns itens relacionados aos casos de teste, como definição de classes e condições de teste (IEEE, 2008), acredita-se que alguns consideram que o plano de teste já contempla a fase de projeto de casos de teste, como é o caso do *Perfil-MPS.Br* que não teve nenhuma prática selecionada como mandatória nessa fase. Já pra os especialistas (*Perfil-Especialista*) as práticas



*Identificar e priorizar condições de teste, Identificar dados de teste específicos necessários e Manter rastreabilidade horizontal com requisitos* devem obrigatoriamente constituir essa fase de projeto de casos de teste. Para o *Perfil-TMMi*, são as práticas *Identificar e priorizar casos de teste* e *Manter rastreabilidade horizontal com requisitos* que devem obrigatoriamente estar presentes para a definição dos casos de teste. Essas diferenças entre os três perfis mostra que não há um consenso claro a respeito dessa fase. Isso pode mostrar que nem sempre casos de teste são documentados separadamente do plano de teste, quando elaborados.

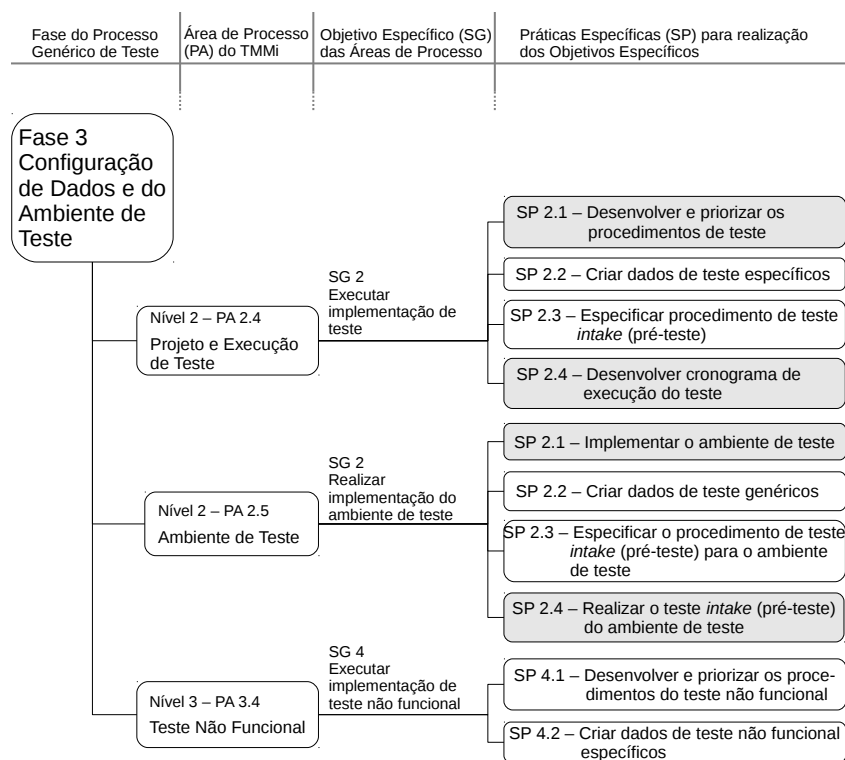


**Figura 3.8: Práticas do TMMi relacionadas à fase de Projeto de Casos de Teste (adaptado de Höhn (2011)).**

### 3.4.3 Configuração de Dados e de Ambiente de Teste

Como foi mostrado anteriormente na fase de planejamento, as necessidades de ambiente são identificadas e descritas para que possam ser implementadas. A fase de configuração de dados e de ambiente de teste possui como objetivo a implementação do teste e do ambiente de teste. Esses objetivos e suas respectivas práticas podem ser vistos na Figura 3.9, que mostra as práticas do TMMi relacionadas a essa fase. Nessa figura estão destacadas as práticas que, segundo a maioria dos respondentes do *survey*, devem obrigatoriamente fazer parte da configuração de dados e de ambiente de teste. Segundo o TMMi, a prática *Desenvolver e priorizar procedimentos de teste* consiste em determinar a ordem de execução dos casos de teste de acordo com os riscos de produto que foram identificados. Esse resultado é coerente com as práticas presentes no planejamento, que estão relacionadas à análise de risco, e ainda com as práticas presentes na

fase anterior de criação de casos de teste. Outra prática relacionada presente nesses resultados é *Desenvolver cronograma de execução do teste*, a qual está diretamente relacionada à prática anterior, já que o cronograma depende da ordem de execução dos casos de teste. A prática *Implementar o ambiente* requer que se coloque em prática os requisitos de ambiente que foram analisados na fase de planejamento, enquanto *Realizar o teste intake (pré-teste) do ambiente de teste* significa certificar-se de que o ambiente está funcional. As práticas classificadas como mandatórias (destacadas na Figura 3.9) pela maioria nessa fase cumprem seu objetivo principal de implementar o ambiente necessário para a execução dos testes.

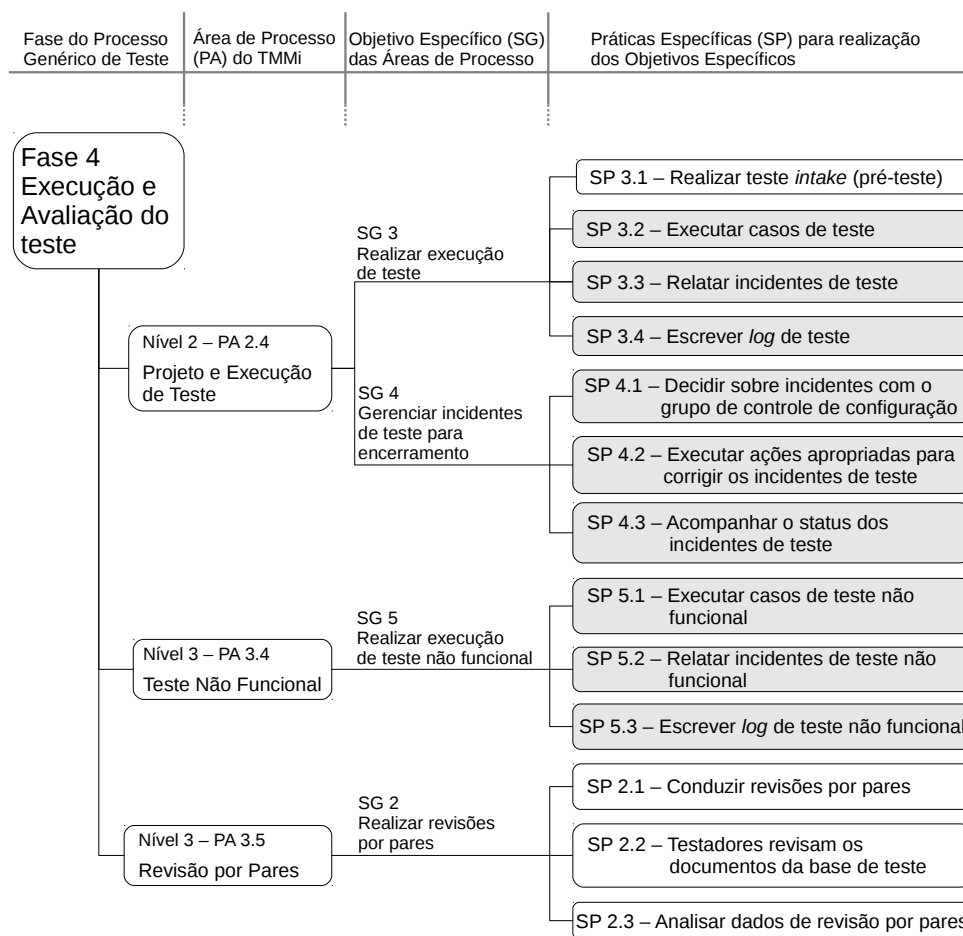


**Figura 3.9: Práticas do TMMi relacionadas à fase de Configuração de Dados e de Ambiente de Teste (adaptado de Höhn (2011)).**

### 3.4.4 Execução e Análise

Após a criação dos dados de teste, e do ambiente ter sido configurado para ser utilizado pela equipe de teste, o próximo passo é realizar a execução dos testes planejados. A equipe que foi alocada executa os casos de teste e, como resultado, apresenta uma lista dos defeitos encontrados e registrados em relatório. A análise da execução visa assegurar que o objetivo do teste foi alcançado e comunicar os resultados aos interessados (HASS, 2008). Nessa fase do processo genérico de Teste Höhn (2011) alocou 13 práticas do TMMi que estão relacionadas aos objetivos de execução, gerenciamento de incidentes, execução do teste não funcional e re-

visão por pares, como pode ser visto na Figura 3.10. Para essa fase, apenas quatro práticas não foram marcadas como mandatórias, o que mostra a grande importância dessa fase para o projeto de teste, já que nele se encontram as atividades relacionadas à execução e gerenciamento dos incidentes. As práticas relacionadas aos objetivos *Realizar execução de teste* e *Gerenciar incidentes de teste para encerramento* dão continuidade ao processo que vem sendo definido desde a fase de planejamento. Na fase de execução e análise, inicia-se a execução dos casos de teste de acordo com a prioridade definida, no ambiente que foi implementado, seguindo o planejamento estabelecido. Conforme a execução é realizada, os incidentes devem ser registrados. O TMMi aconselha que se registre um *log*, que funciona como um diário da execução. Esse *log* de execução servirá como base para reuniões de lições aprendidas e como base histórica para futuras melhorias. Com os incidentes registrados, devem ser tomadas as medidas adequadas para solucioná-los, como a definição de prioridade de correção e da equipe para correção, e o estabelecimento do prazo. Essas atividades devem ser acompanhadas de perto, para que se possa garantir que o incidente foi de fato corrigido de acordo com o planejado.



**Figura 3.10:** Práticas do TMMi relacionadas à fase de Execução e Análise (adaptado de Höhn (2011)).

Nos resultados do *survey* também estão presentes práticas relacionadas à execução de testes

não funcionais. Entretanto, nas fases de planejamento e projeto de casos de teste não estiveram presentes práticas relacionadas à definição desse tipo de teste. Apesar da aparente incoerência, pode-se considerar que as definições de testes e testes não funcionais não são vistas como atividades distintas que precisem ser definidas separadamente. Essa separação é característica da estrutura do TMMi, porém desnecessária, já que na prática essas atividades são realizadas de forma conjunta. As práticas para testes não funcionais são semelhantes às práticas para testes que não os testes não funcionais, como pode ser visto na Figura 3.10. Portanto, essas práticas relacionadas à execução, pertencentes ao objetivos *Realizar execução de teste* e *Realizar execução de teste não funcional*, não devem ser vistas como individuais, mas sim complementares. Com isso, a fase de execução se encerra. A fase apresentada a seguir é de acompanhamento de todo o processo e visa a garantir a qualidade do processo em execução.

### 3.4.5 Monitoramento e Controle

A execução das quatro fases apresentadas nas seções anteriores terá gerado uma quantidade razoável de informação. Essa informação precisa ser organizada e consolidada para que seja possível verificar rapidamente a situação do teste e também do software em teste. Isso é contemplado na fase de Monitoramento e Controle (CRESPO et al., 2010). Na Figura 3.11 são mostradas as práticas do TMMi que estão relacionadas a essa fase proposta por Höhn (2011), com os resultados do *survey* destacados. Todas as práticas destacadas apareceram nos resultados das análises de todos os perfis e também na análise do grupo todo; ou seja, é consenso que essas práticas, obrigatoriamente, fazem parte da fase de monitoramento e controle e ajudam no acompanhamento de todo o processo. *Conduzir revisões do progresso do teste* significa acompanhar o progresso da atividade de teste, verificando seu andamento e permite *Monitorar defeitos* com o objetivo de coletar métricas dos problemas encontrados. Para manter os interessados informados deve-se *Conduzir revisões de qualidade do produto* e deve-se *Analisar problemas encontrados* e *Tomar ação corretiva* para solucionar cada um desses problemas. *Gerenciar a ação corretiva* é importante também, pois tem o intuito de verificar se a correção é de fato realizada.

O ambiente de teste também deve ser acompanhado, tanto sua disponibilidade e uso, quanto possíveis incidentes que possam vir a ocorrer. Assim, pode-se gerar um histórico de uso de ambiente para um melhor gerenciamento em projetos futuros. Como pode ser visto, essas atividades apresentam uma sequência de monitoramento, o que mostra coerência nos resultados observados. Na seção a seguir são apresentadas algumas das possíveis limitações deste *survey*.

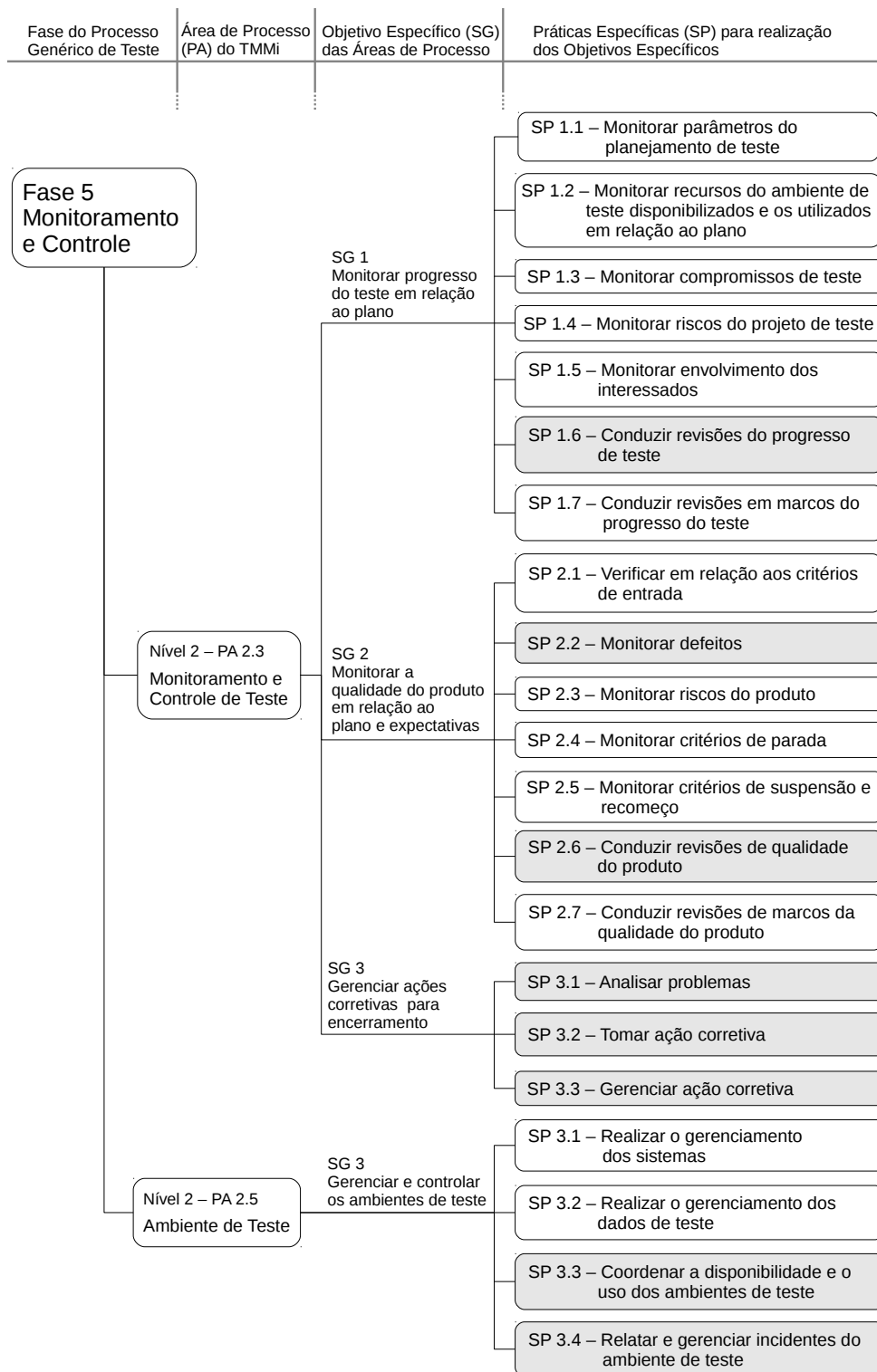


Figura 3.11: Práticas do TMMi relacionadas à fase de Monitoramento e controle (adaptado de Höhn (2011)).

### 3.5 Riscos à Validade

A utilização do TMMi como estrutura fundamental para construção das questões, sem modificar a descrição das práticas, apenas traduzindo-as, pode ter representado uma limitação ao

estudo. A ajuda fornecida foi baseada na própria explicação provida no documento do TMMi, porém muitas vezes o texto da ajuda pode ter sido insuficiente para permitir a correta compreensão da questão associada. O TMMi possui uma estrutura muito característica e semelhante ao CMMI, que é extremamente subdividida com a intenção de facilitar sua implementação. Entretanto, muitas vezes essa estrutura pode confundir o leitor, o qual não compreende a separação de algumas atividades, como é o caso dos testes funcionais e não funcionais. Na pesquisa relatada neste capítulo, por exemplo, ficou claro que, na fase de execução e análise, as práticas de execução de testes funcionais e não-funcionais não são compreendidas como atividades distintas. A forma como o *survey* foi planejado também pode representar uma ameaça à validade dos resultados obtidos. A escala utilizada possui apenas quatro valores, o que limitou bastante a análise estatística, já que as respostas ficaram concentradas em 3 e 4. A utilização de uma escala com mais valores, por exemplo de 1 a 10, poderia ter gerado uma melhor distribuição das respostas. Um modelo de interpretação também poderia ter fornecido melhores resultados. O uso do GQM (BASILI; CALDIERA; ROMBACH, 1994), por exemplo, poderia ter auxiliado na melhor elaboração das perguntas e da estrutura do *survey*, com base em um modelo de interpretação que deve ser elaborado antes mesmo da definição das questões.

A amostra também representa uma limitação do trabalho. Apesar de incluir somente profissionais de teste de software, seu tamanho foi reduzido, diante de uma população que é sabidamente grande. Talvez os meios de divulgação e o tempo em que ficou disponível – apenas 30 dias – podem ter limitado o dimensionamento da amostra. Mas, mesmo com essas limitações considera-se que o *survey* contribuiu bastante para corroborar várias opiniões que a equipe possuía, ou que foram se formando ao longo deste trabalho. Na seção a seguir são traçadas as conclusões da pesquisa realizada.

### 3.6 Considerações Finais

Este capítulo descreveu uma pesquisa (no formato de um *survey* que foi realizada em duas etapas, cujo intuito foi observar se é possível reduzir o conjunto de práticas do TMMi que está relacionado a um processo genérico de teste. A pesquisa foi realizada com profissionais que trabalham com teste de software. Com base na resposta desses participantes e nas análises mostradas nas seções anteriores pode-se observar que o conjunto de 81 práticas do TMMi, que Höhn (2011) atribuiu ao processo genérico de teste, pode ser reduzido a 33 (trinta e três) práticas, ainda assim mantendo a completude e coerência necessárias para a execução de um processo de teste. Isso representa uma redução em torno de 60% do TMMi. As outras práticas do TMMi não são descartáveis, porém, quando se tem como objetivo implementar um processo

reduzido, ou ainda, quando a empresa não tem o *know-how* necessário para implementar sozinha seu próprio processo, ela pode utilizar esse conjunto de práticas para fazê-lo. Dessa forma, os resultados aqui relatados representam uma maneira simplificada de criação ou melhoria de processos de teste, elaborada com base em um modelo de referência reconhecido e maduro.

As práticas mostradas em destaque nas Figuras 3.7, 3.8, 3.9, 3.10 e 3.11 também podem indicar a prioridade de implementação para a empresa que esteja utilizando o TMMi para implementar seu processo. Esse modelo não indica o que pode ser implementado primeiro, nem as possíveis dependências entre as áreas de processo. Com o resultado aqui apresentado, tem-se um conjunto de atividades que podem ser implantadas antes de se implantar o restante das práticas exigidas pelo framework, sem prejuízo para o processo, já que ele poderá ser inteiramente executado com esse conjunto base. A empresa pode optar por continuar a implantar o restante das práticas exigidas, com o objetivo de se certificar no modelo.

A divisão em práticas do TMMi pode ser uma tentativa de facilitar a implementação das atividades; porém, muitas vezes, torna o modelo complexo e de difícil compreensão. O processo a ser construído, com base em algum modelo de referência, deve estar de acordo com a realidade da empresa que o adotará. Nem todas as atividades sugeridas no TMMi são factíveis para todos os tamanhos de empresas e equipes. Com base nisso, considera-se que seja importante ter como base quais práticas, caso não sejam executadas, possam comprometer mais a qualidade do processo e conseqüentemente do produto em teste. Diante disso, os resultados da pesquisa ajudam a identificar o que deve ser feito essencialmente num processo de teste, segundo os especialistas da área. Acredita-se que o resultado da pesquisa servirá de base para pequenas e médias empresas que desejem implementar um novo processo de teste ou, ainda, melhorar o processo que já estiver em uso. Desta forma, a pesquisa serviu como base para a criação do processo que será apresentado no Capítulo 4. O processo foi criado no contexto de uma pequena empresa, que já possuía um processo de teste, porém que não atendia às necessidades da empresa. Maiores detalhes sobre a empresa, o contexto em que se insere e a elaboração do processo serão apresentados no Capítulo 4.





# Capítulo 4

## ESTUDO DE CASO

---

---

Este capítulo apresenta o estudo realizado em parceria com uma pequena empresa de software para a elaboração de um processo de teste adaptado ao seu contexto, a partir do processo genérico proposto no capítulo anterior. O processo elaborado foi comparado com o processo existente na empresa, com o objetivo de identificar se o processo elaborado respeita as principais características da empresa.

### 4.1 Considerações Iniciais

Pequenas empresas que não executam atividades de teste tem, em geral, dificuldade em implementar um processo de teste e executá-lo efetivamente. As principais razões são a falta de recursos e especialistas no assunto, tempo e orçamento escassos, e falta de conhecimento sobre a relevância e benefícios que podem ser obtidos quando um processo de teste é executado adequadamente (RODRIGUES; PINHEIRO; ALBUQUERQUE, 2010). Apesar disso, a preocupação com a qualidade de software é crescente, e oferecer software de qualidade pode ser essencial para que o produto se destaque diante de um mercado tão concorrido. Assim, como foi dito anteriormente, esse contexto motivou a realização deste trabalho devido ao interesse de uma empresa desenvolvedora de software em melhorar seu processo de teste e oferecer o teste de software como serviço.

A motivação da empresa é tornar o teste de software um novo serviço, em forma de fábrica, semelhante ao modelo que já possui de desenvolvimento de software. O objetivo da empresa é, além de melhorar o processo de teste existente para testar o que é desenvolvido internamente, é oferecer também o serviço de teste de software no mesmo modelo da fábrica de software. O modelo fábrica de teste seria utilizado internamente também, tendo como cliente a própria fábrica de software. Outra motivação para a melhoria do processo de teste existente é obter a

certificação nível D do MPS.Br. Ressalta-se que a empresa em questão possui um processo de teste que atualmente faz parte do processo da fábrica de software. O processo de teste em uso serviu como ponto de partida para esta pesquisa.

As seções deste capítulo apresentam o estudo na ordem em que as atividades ocorreram. Na Seção 4.2 será apresentada as características da empresa e na Seção 4.3 será apresentado o diagnóstico do processo de teste dessa empresa, que buscou identificar as atividades de teste realizadas pela empresa bem como as práticas do TMMi que são implementadas por ela. Na Seção 4.4 será apresentado o processo real da empresa, que foi remodelado. A Seção 4.5 apresenta os pontos positivos e de melhorias identificados após uma análise crítica do processo real. Por fim, na Seção 4.6 são apresentadas as considerações finais referentes ao estudo realizado.

## 4.2 Caracterização da Empresa

A empresa na qual foi realizado o estudo é de pequeno porte e está no mercado de desenvolvimento de software há 14 anos. Localizada na cidade de Ribeirão Preto, a empresa faz parte de uma associação das empresas de software, chamada PISO<sup>1</sup> (Pólo das Indústria de Software de Ribeirão Preto). A pedido da empresa seu nome não será divulgado. São consideradas empresas de pequeno porte aquelas que possuem faturamento anual de até R\$ 3,6 milhões<sup>2</sup>. Essa empresa é uma fábrica de software desenvolvedora de sistemas de informação para diversos fins, sendo em geral sistemas web. Possui uma equipe com 40 colaboradores, dos quais 19 são desenvolvedores que podem estar na empresa ou alocados em clientes. Atualmente a equipe conta com um analista de teste e um gestor de projeto; porém, no início deste trabalho a equipe contava com dois analistas de teste, um analista de qualidade e um gestor de projeto.

O analista de teste, quando alocado para o projeto, é responsável por planejar e executar os testes. A fábrica de software recebe novos projetos quando o gerente de desenvolvimento recebe a proposta de projeto com o aceite do cliente e então atribui o projeto ao gestor de projetos. O gestor elabora a documentação necessária a partir dos dados recebidos, cria a estrutura do projeto na ferramenta de armazenamento e solicita ao analista de requisitos que realize a elicitação dos mesmos. Com o documento de requisitos aprovado pelo cliente, o gestor de projetos atribui aos desenvolvedores os requisitos que devem ser desenvolvidos. A partir desse momento inicia-se o processo de desenvolvimento e teste, no qual os requisitos são implementados e testados (teste unitário). Após a conclusão do desenvolvimento, o software é liberado para o cliente.

---

<sup>1</sup><http://www.piso.org.br/>

<sup>2</sup>Fonte: <http://www.sebrae.com.br/customizado/estudos-e-pesquisas/temas-estrategicos/perfil-dos-pequenos-negocios/empresas-de-pequeno-porte>

Para a realização deste estudo, primeiramente foi necessário analisar o processo de teste em uso na empresa, buscando identificar quais atividades de teste são realizadas e como o teste acompanha o processo de desenvolvimento. Inicialmente foram feitas algumas reuniões com a empresa para apresentar a proposta do trabalho a ser realizada. Acompanhou-se também o desenvolvimento de um processo de verificação e validação que já estava em andamento devido à certificação MPS.Br. Porém, identificou-se a necessidade de realizar um diagnóstico do processo atual para possibilitar um melhor entendimento do processo de teste utilizado pela empresa. Foram observados o processo de teste corrente e o processo de verificação e validação que estava em desenvolvimento por causa da certificação. Os resultados desse diagnóstico são apresentados na próxima seção. Embora a empresa não permita divulgar os diagramas de seu processo de teste que foram elaborados por ela, esse processo será apresentado remodelado na Seção 4.4.

### **4.3 Diagnóstico do processo de teste da empresa**

Para conhecer melhor o processo de teste que a empresa possui foi realizado um diagnóstico, tendo como base de comparação o TMMi (TMMi Foundation, 2010), com o intuito de identificar melhor quais práticas do TMMi estão contempladas no processo atual da empresa e obter um direcionamento de quais práticas ainda deveriam ser implementadas. Para esse diagnóstico utilizou-se a ferramenta KITTool, desenvolvida por Höhn (2011) e apresentada no Capítulo 2. No entanto, como essa ferramenta não foi explorada no contexto real de uma empresa, realizou-se uma avaliação por mapa mental do processo atual da empresa esperando-se que o resultado fosse o mesmo que o encontrado com o diagnóstico da KITTool. Nas próximas seções são apresentados a avaliação por mapa mental e o diagnóstico realizados, juntamente com respectivos resultados.

#### **4.3.1 Avaliação por Mapa Mental**

Utilizando o mapa mental criado por Höhn (2011) foi feita uma avaliação do processo da empresa com relação à aplicação das práticas do TMMi. Essa avaliação foi feita observando-se o processo de teste em uso na empresa, com base no conhecimento prévio que se tinha desse processo e das atividades nele realizadas. Na Figura 4.1 é apresentado o mapa mental, que retrata esse processo de teste em uso, o qual foi inspirado no mapa mental proposto por Höhn (2011). O nó raiz é o assunto que trata o mapa mental, ou seja, o processo de teste atual da empresa; Os dois nós seguintes são os subprocessos do processo de teste da empresa:

“Gerar ou Revisar Artefatos de Teste” e “Executar Teste”, sendo que este último é também um subprocesso do processo de desenvolvimento de software da empresa. O terceiro nível do mapa contém as atividades dos subprocessos em questão. Assim como no KITMap, as folhas do mapa mental são as práticas do TMMi que, nesse caso, estão relacionadas ou às atividades dos subprocessos. Atividades que fazem parte do conhecimento tácito foram retratadas como “atividades não descritas” no mapa mental.

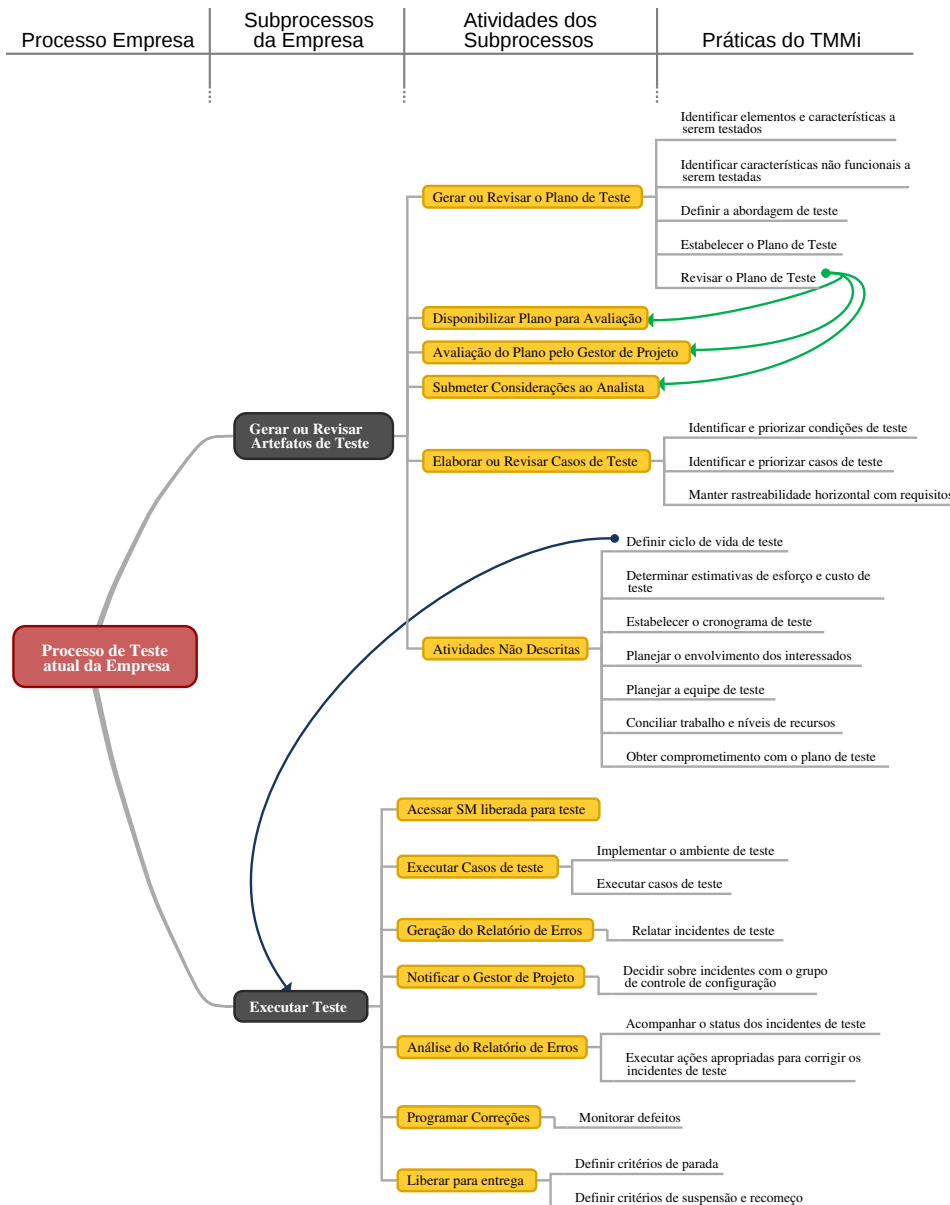


Figura 4.1: Mapa mental do processo atual da empresa com as práticas do TMMi associadas.

No subprocesso “Gerar ou Revisar Artefatos de Teste” existem cinco atividades formalmente descritas que estão relacionadas ao Planejamento e ao Projeto de Casos de Teste. A primeira delas, “Gerar ou Revisar o Plano de Teste”, está relacionada a cinco práticas do TMMi. O plano de teste da empresa busca identificar tanto características funcionais quanto não fun-

cionais a serem testadas, além de identificar também a abordagem de teste que será utilizada. Por essa razão as práticas *Identificar elementos e características a serem testados*, *Identificar características não funcionais a serem testadas* e *Definir a abordagem de Teste* estão associadas a essa atividade. *Estabelecer o Plano de Teste* também está relacionada a essa atividade, pois é por meio dela que o plano é elaborado e mantido. Observe que na Figura 4.1 existe uma ligação entre a prática *Revisar o Plano de Teste* e as atividades de “Disponibilizar Plano para Avaliação”, “Avaliação do Plano pelo Gestor de Projeto” e “Submeter Considerações ao analista”, representada pelas linhas verdes. Essa ligação ocorre porque o gestor deve analisar o plano de teste para que então o analista de teste o revise. Nesse mesmo subprocesso, a atividade “Elaborar e Revisar Casos de Teste” está relacionada às práticas *Identificar e priorizar condições de teste*, *Identificar e priorizar casos de teste* e *Manter rastreabilidade horizontal com requisitos*, pois o analista de teste identifica condições de teste a partir dos requisitos e elabora casos de teste com base nos requisitos, condições de teste e no documento de projeto fornecido pelo gestor. Os casos de teste são gerenciados por meio de uma ferramenta integrada com o sistema de gestão de requisitos da empresa, mantendo portanto rastreabilidade horizontal com requisitos. Vale lembrar que as práticas presentes na Figura 4.1 foram consideradas mesmo quando eram implementadas parcialmente.

Ainda no subprocesso “Gerar ou Revisar Artefatos de Teste” foi observado que seis práticas do TMMi são colocadas em prática, mas não estão descritas formalmente. Essas práticas estão relacionadas às atividades do gestor de projeto e são assinaladas na Figura 4.1 como “Atividades não descritas”. O gestor planeja a equipe, define cronograma, estima esforço e custo, realiza a reunião inicial com a equipe e obtém o comprometimento da mesma com o projeto de teste. Por essa razão identificou-se que as práticas *Determinar estimativas de esforço e custo de teste*, *Estabelecer o cronograma de teste* e *Planejar o envolvimento dos interessados* são implementadas. Além disso, o gestor ainda acompanha o projeto e renegocia prazos e orçamento, caso identifique a necessidade; por isso, *Conciliar trabalho e níveis de recursos* também está associada diretamente ao subprocesso. *Definir ciclo de vida de teste* também está contemplada uma vez que é feito um planejamento, casos de teste são projetados e executados. Por essa razão existe uma associação entre a prática e o subprocesso “Executar Teste” representada pela linha azul na Figura 4.1.

No subprocesso “Executar Teste”, a atividade “Executar casos de teste” está claramente associada à prática que diz respeito à execução dos casos de teste e também está associada à prática de *Implementar o ambiente de teste*, pois apesar de não estar evidente no processo, essa atividade é realizada e é necessária para que os casos de teste possam ser executados. À medida que se executam os casos de teste os defeitos são encontrados e são inseridos em um

relatório de defeitos. Portanto, a prática *Relatar incidentes de teste* foi associada à atividade “Executar casos de teste”. “Notificar o Gestor do Projeto” é tarefa realizada pelo analista de teste após o término dos testes programados. Nesse momento, o gestor de projeto decide a atitude corretiva a ser tomada com relação aos defeitos encontrados. Apesar dessa decisão não ser realizada necessariamente com o gestor de configuração, considera-se que a prática *Decidir sobre incidentes com o grupo de controle de configuração* é parcialmente realizada, já que o gestor de projeto assume esse papel.

A atividade “Análise do Relatório de Defeitos” é realizada pelo gestor de projeto, que acompanha o status do defeito e toma a atitude corretiva necessária. Por isso, as práticas *Acompanhar status dos incidentes de teste* e *Executar ações apropriadas para corrigir os incidente teste* estão relacionadas a esta atividade. Após programar as correções, o gestor passa a monitorar os defeitos verificando o status do defeito, à medida que o analista realiza testes de regressão nos itens corrigidos. A atividade “Liberar para Entrega” está associada às práticas *Definir critérios de parada* e *Definir critérios de suspensão e recomeço*, pois são nos marcos de liberação do software que o gestor de projeto avalia se os testes devem parar, se novas atitudes corretivas devem ser tomadas, ou se a atividade de teste deve ser suspensa. Essas definições são feitas, mas não são documentadas.

A avaliação descrita nesta seção permitiu observar que a empresa realiza atividades, especialmente de gerenciamento do projeto de teste, que não estão descritas no processo. Após a realização dessa avaliação, identificou-se a necessidade de remodelar o processo da empresa, de forma a refletir todas as atividades que são executadas nesse processo de teste. Observou-se ainda que o gestor de projeto acumula a função de gerenciar o andamento do projeto de desenvolvimento do software e da atividade de teste. O analista de teste, apesar de realizar atividades de planejamento, auxiliando o gestor de projeto, não acompanha o andamento do projeto e ele é o responsável pela execução do teste. Para dar continuidade à avaliação do processo da empresa foi realizado um diagnóstico utilizando a ferramenta KITTool (HÖHN, 2011), apresentada no Capítulo 2, Seção 2.4. Esse diagnóstico será apresentado na próxima seção. Como o diagnóstico do processo com a ferramenta já estava planejado, optou-se por realizar a remodelagem do processo de teste após a obtenção dos resultados do diagnóstico, pois assim o processo da empresa poderia ser melhor conhecido. Serão incluídas referências às etapas da estratégia apresentada no Capítulo 5 ao final das respectivas seções.

### 4.3.2 Diagnóstico KITTool

Após a avaliação relatada na seção anterior, foi feito o diagnóstico com a ferramenta KIT-Tool (HÖHN, 2011) para, novamente, identificar quais práticas e objetivos do TMMi a empresa cumpre. O diagnóstico foi realizado em conjunto com o analista de teste e com o analista de qualidade, responsável pela certificação MPS.Br e, conseqüentemente, pelo estabelecimento dos processos da empresa. Antes de dar início ao diagnóstico foi feita uma breve apresentação da ferramenta e do TMMi; também foi explicado que o diagnóstico seria feito por meio de perguntas sobre o que é realizado no processo de teste da empresa. Para cada pergunta do formulário da KITTool era fornecida uma breve explicação sobre a prática em questão e sobre o que se espera como produto dessa prática. A ferramenta oferece as opções de realização do diagnóstico tanto por práticas quanto por objetivos do TMMi, como mostrado no Capítulo 2, Seção 2.4. Optou-se pela realização do diagnóstico por práticas devido a essa ser a opção mais detalhada<sup>3</sup>.

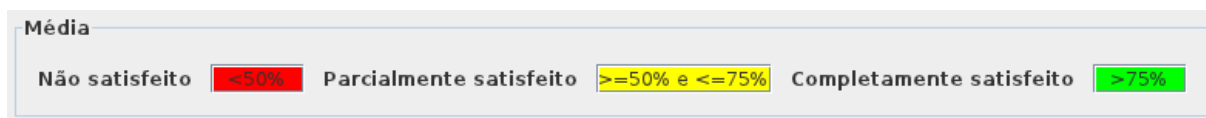
No diagnóstico com a KITTool o usuário indica, para cada prática do TMMi, a porcentagem de aplicação da prática no processo da empresa, com base na tabela de referência que é apresentada na tela do usuário, conforme a Figura 2.7 do Capítulo 2. Essas práticas indicam:

- 0% indica que não há metodologia, atividades ou procedimentos que atendam à prática solicitada;
- 25% significa que não existe metodologia ou procedimento desenvolvido, menos de 70% dos requisitos solicitados da prática são atendidos e as práticas que são implementadas não são aplicadas para todos os projetos;
- 50% significa que existe um procedimento definido, porém não documentado, as práticas que são implementadas são aplicadas somente em projetos estratégicos para empresa, mas nem todos os requisitos das práticas que são solicitados estão implementados;
- 75% significa que já existe um procedimento ou metodologia que além de definido está documentado, as práticas que são implementadas são aplicadas para a maioria dos projetos, e são atendidos a maioria dos requisitos solicitados para implementação das práticas;
- 100% significa que, além de possuir um procedimento definido e documentado, a empresa implementa as práticas para todos os projetos e atende a todos os requisitos solicitados para esta prática.

---

<sup>3</sup>Conforme apresentado na Capítulo 2, Seção 2.4, há também a alternativa de realizar o diagnóstico por objetivos.

Na Figura 4.2 é ilustrado o significado das cores e valores apresentados nos resultados. Caso a nota ou média esteja abaixo de 50%, conclui-se que a prática (ou objetivo) não é satisfeita, ou seja, ainda há práticas a serem implementadas ou nem todos os seus requisitos são atendidos. Se o resultado estiver entre 50% e 75% significa que a prática é parcialmente satisfeita, ou seja, existem práticas e requisitos que já estão definidos e documentados. Caso o resultado seja maior que 75% significa que a prática, ou objetivo, é completamente satisfeito, ou seja, as práticas são definidas e documentadas. Essa mesma regra vale para as áreas de processo.



**Figura 4.2:** Legenda apresentada pela KITTool para o diagnóstico realizado “por prática”.

O diagnóstico da empresa é mostrado nas Figuras 4.3, 4.4 e 4.5, lembrando que as notas das práticas são atribuídas pelo usuário e as notas dos objetivos foram atribuídas pela KITTool por meio do cálculo da média das notas atribuídas às práticas. Para efeito de síntese do processo de teste existente na empresa as práticas que possuem nota superior a 0% serão comentadas tendo-se como referência os requisitos exigidos pelo TMMi (TMMi Foundation, 2010). Note, nas Figuras 4.3 e 4.4, que apesar de existirem algumas práticas que são bem conduzidas pela empresa com notas de 50% e 100%, como é o caso da prática SP4.5 da Figura 4.3, por exemplo, pode-se observar que nenhum dos objetivos e áreas de processo são satisfeitos.

Na Figura 4.3 pode-se observar que na área de processo *Política e Estratégia de Teste* a prática SP 2.2 recebeu nota 25%, pois entendeu-se que há uma estratégia de teste que é adotada, já que alguns níveis de teste são identificados (testes unitários e de sistema), os tipos de teste a serem realizados também são identificados e os testes são executados em um ambiente definido para tal. A prática SP 3.1, dessa mesma área de processo, também recebeu nota de 25%, pois a empresa já definiu algumas métricas para teste, embora não sejam utilizadas em todos os projetos e também não estejam documentadas. Para a área de processo *Planejamento de Teste* (também mostrada na Figura 4.3) a prática SP 2.1 recebeu nota 25%, pois o plano de teste identifica itens e características que serão testadas, porém essa definição não é feita com base em uma análise de risco do produto. A empresa possui um ciclo de vida de teste, por isso foi atribuído 25% à prática SP 3.2, mas ela não é totalmente definida e não são identificados marcos de projeto para re-planejamento quando necessário. Estimativas de esforço e custo são feitas pelo gestor de projeto levando em consideração o custo de esforço da equipe de teste. As estimativas não são feitas em relação aos atributos das tarefas de teste e não leva em consideração medidas como a complexidade do item a ser testado ou número de casos de teste, por exemplo. Por essa razão considera-se que a prática SP 3.3 é satisfeita em partes.



O objetivo *Desenvolver um plano de Teste* obteve média de 25%, pois a empresa possui um plano de teste que é definido e documentado, porém, os riscos ao projeto de teste não são identificados. As práticas que constituem esse objetivo em sua maioria não são satisfeitas e apenas a prática SP 4.5 está parcialmente satisfeita. A empresa possui um plano de testes estabelecido, que inclui a definição do cronograma e de equipe, mas nem todos os projetos da empresa são testados. A empresa também identifica o envolvimento dos interessados, a participação da equipe e cliente no projeto de testes é identificada, porém não é documentada. Na Figura 4.3 observa-se que o objetivo *Obter comprometimento com o plano de teste* possui duas práticas parcialmente satisfeitas, que são SP 5.1 e SP 5.3. O plano de testes é revisado pelo gestor de projeto e necessita de sua aprovação prévia para que possa ser utilizado e o comprometimento com o plano é obtido por meio de reunião com a equipe. Nesse mesmo objetivo a prática SP 5.2 tem nota 25% porque o gestor de projeto re-planeja a atividade de testes de acordo com os recursos disponíveis, porém isso não é refletido no plano de testes.

Nível 2 PA2.1 Política e Estratégia de Teste	7%
SG2 Estabelecer uma estratégia de teste	8%
SP2.1 Executar uma avaliação de risco do produto genérico	0%
SP2.2 Definir estratégia de teste	25%
SP2.3 Distribuir a estratégia de teste aos interessados	0%
SG3 Estabelecer indicadores de desempenho de teste	13%
SP3.1 Definir indicadores de desempenho de teste	25%
SP3.2 Divulgar indicadores de desempenho de teste	0%
Nível 2 PA2.2 Planejamento de Teste	18%
SG2 Estabelecer uma abordagem de teste	5%
SP2.1 Identificar elementos e características a serem testados	25%
SP2.2 Definir a abordagem de teste	0%
SP2.3 Definir critérios de entrada	0%
SP2.4 Definir critérios de parada	0%
SP2.5 Definir critérios de suspensão e recomeço	0%
SG3 Estabelecer estimativas de teste	17%
SP3.1 Estabelecer uma wbs de alto nível	0%
SP3.2 Definir ciclo de vida de teste	25%
SP3.3 Determinar estimativas de esforço e custo de teste	25%
SG4 Desenvolver um plano de teste	25%
SP4.1 Estabelecer o cronograma de teste	25%
SP4.2 Planejar a equipe de teste	25%
SP4.3 Planejar envolvimento dos interessados	25%
SP4.4 Identificar riscos ao projeto de teste	0%
SP4.5 Estabelecer o plano de teste	50%
SG5 Obter comprometimento com o plano de teste	42%
SP5.1 Revisar o plano de teste	50%
SP5.2 Conciliar trabalho e níveis de recursos	25%
SP5.3 Obter comprometimento com o plano de teste	50%

**Figura 4.3: Resultado apresentado pela ferramenta KITTool para o diagnóstico realizado “por prática” das áreas de processo “Política e Estratégia de Teste” e “Planejamento de Teste”.**

A área de processo *Monitoramento e Controle*, apresentada na Figura 4.4, não é satisfeita, sendo que das 17 práticas relacionadas, quatro possuem nota 25%, e apenas uma é parcialmente satisfeita, com nota 50%. Para o objetivo *Monitorar progresso do teste em relação ao plano* apenas as práticas SP 1.2 e SP 1.3 são realizadas em partes pelo gestor de projeto. A prática SP 2.2 é parcialmente satisfeita porque a empresa acompanha os defeitos encontrados, sendo que esses dados são reunidos em um relatório gerencial, porém não são utilizados para ava-

Nível 2 PA2.3 Monitoramento e controle de teste	10%
SG1 Monitorar progresso do teste em relação ao plano	7%
SP1.1 Monitorar parâmetros do planejamento de teste	0%
SP1.2 Monitorar recursos do ambiente de teste disponibilizados e os utilizados em relação ao plano	25%
SP1.3 Monitorar compromissos de teste	25%
SP1.4 Monitorar riscos do projeto de teste	0%
SP1.5 Monitorar envolvimento dos interessados	0%
SP1.6 Conduzir revisões do progresso do teste	0%
SP1.7 Conduzir revisões em marcos do progresso de teste	0%
SG2 Monitorar a qualidade do produto em relação ao plano e expectativas	7%
SP2.1 Verificar em relação aos critérios de entrada	0%
SP2.2 Monitorar defeitos	50%
SP2.3 Monitorar riscos do produto	0%
SP2.4 Monitorar critérios de parada	0%
SP2.5 Monitorar critérios de suspensão e recomeço	0%
SP2.6 Conduzir revisões de qualidade do produto	0%
SP2.7 Conduzir revisões de marcos da qualidade do produto	0%
SG3 Gerenciar ações corretivas para encerramento	17%
SP3.1 Analisar problemas	25%
SP3.2 Tomar ação corretiva	25%
SP3.3 Gerenciar ação corretiva	0%
Nível 2 PA2.4 Projeto e Execução de Teste	16%
SG1 Realizar análise e projeto de teste usando técnicas de projeto de teste	31%
SP1.1 Identificar e priorizar condições de teste	0%
SP1.2 Identificar e priorizar casos de teste	25%
SP1.3 Identificar dados de teste específicos necessários	0%
SP1.4 Manter rastreabilidade horizontal com requisitos	100%
SG3 Realizar execução de teste	25%
SP3.1 Realizar teste intake (pre-teste)	0%
SP3.2 Executar casos de teste	100%
SP3.3 Relatar incidentes de teste	0%
SP3.4 Escrever log de teste	0%
SG4 Gerenciar incidentes de teste para encerramento	8%
SP4.1 Decidir sobre incidentes com o grupo de controle de configuração	0%
SP4.2 Executar ações apropriadas para corrigir os incidentes de teste	25%
SP4.3 Acompanhar o status dos incidentes de teste	0%

**Figura 4.4:** Resultado apresentado pela ferramenta KITTool para o diagnóstico realizado “por prática” das áreas de processo “Monitoramento e Controle” e “Projeto e Execução de Teste”.

liar o planejamento realizado. Os dados sobre defeitos também não são utilizados como base histórica para a elaboração de outros planos de teste. O objetivo *Gerenciar ações corretivas para encerramento* não é satisfeito, mas duas de suas práticas possuem itens que são realizados pelo gestor de projetos, são elas: SP 3.1 (*Analisar Problemas*) e SP 3.2 (*Tomar Ação Corretiva*). O gestor acompanha a atividade de teste e, caso identifique problemas, toma as atitudes necessárias para solucioná-los, porém essas atividades não são documentadas e o plano de teste não é atualizado. A área de processo *Projeto e Execução de Teste* também é apresentada na Figura 4.4. Nessa PA duas práticas que foram marcadas como totalmente satisfeitas: SP 1.4 (*Manter rastreabilidade horizontal com requisitos*) e SP 3.2 (*Executar casos de teste*). Essa nota foi atribuída, pois a empresa possui um sistema de cadastro dos requisitos, que permite a liberação para desenvolvimento ou para teste, o que permite a integração com o sistema utilizado para elaboração dos casos de teste, mantendo assim a rastreabilidade com os requisitos. Além disso, a empresa afirma que os casos de teste planejados são documentados, executados e os resultados também são documentados. As outras práticas que tiveram nota 25% foram a SP 1.2 e SP 4.2. A SP 1.2 é realizada parcialmente, pois a empresa identifica casos de teste mas não os prioriza ou não o faz com base em análise de risco de produto. A SP 4.2, por sua vez,

se refere a atitudes corretivas para os defeitos encontrados após a execução dos casos de teste. A empresa corrige os defeitos encontrados, mas o produto pode ser liberado mesmo contendo defeitos ou sem realizar teste de regressão no item que foi corrigido.

Na Figura 4.5 são mostrados os resultados para as áreas de processo *Ambiente de Teste* e *Teste Não Funcional*. Para tais áreas de processo, cinco práticas possuem algum item que é realizado pela empresa. Com relação ao ambiente, a empresa implementa o ambiente de testes (SP 2.1), conforme a descrição do ambiente de produção (SP 1.1), presente no Plano de Projeto Interno, e quando ocorrem problemas relacionados a esse ambiente, o analista de teste relata para quem o implementou, que por sua vez busca uma solução (SP 3.4). Entretanto, essas atividades não são documentadas e requisitos do ambiente de teste não são desenvolvidos. Com relação a testes não funcionais (área de processo *Teste Não Funcional*), a empresa identifica algumas características não funcionais a serem testadas (SP 2.1) e relata os eventuais defeitos encontrados (SP 5.2), mas não planeja e não define uma abordagem de teste não funcional.

Nível 2 PA2.5 Ambiente de Teste	7%
SG1 Desenvolver os requisitos do ambiente de teste	8%
SP1.1 Obter (eliciar) necessidades do ambiente de teste	25%
SP1.2 Desenvolver os requisitos do ambiente de teste	0%
SP1.3 Analisar os requisitos do ambiente de teste	0%
SG2 Realizar implementacao do ambiente de teste	6%
SP2.1 Implementar o ambiente de teste	25%
SP2.2 Criar dados de teste genericos	0%
SP2.3 Especificar o procedimento de teste intake (pre-teste)para o ambiente de teste	0%
SP2.4 Realizar o teste intake (pre-teste) do ambiente de teste	0%
SG3 Gerenciar e controlar os ambientes de teste	6%
SP3.1 Realizar o gerenciamento dos sistemas	0%
SP3.2 Realizar gerenciamento dos dados de teste	0%
SP3.3 Coordenar a disponibilidade e o uso dos ambientes de teste	0%
SP3.4 Relatar e gerenciar incidentes do ambiente de teste	25%
Nível 3 PA3.4 Teste Nao Funcional	3%
SG2 Estabelecer uma abordagem de teste nao funcional	8%
SP2.1 Identificar caracteristicas nao funcionais a serem testadas	25%
SP2.2 Definir a abordagem de teste nao funcional	0%
SP2.3 Definir critérios não funcionais de parada	0%
SG5 Realizar execucao de teste nao funcional	8%
SP5.1 Executar casos de teste nao funcional	0%
SP5.2 Relatar incidentes de teste	25%
SP5.3 Escrever log de teste	0%

**Figura 4.5: Resultado apresentado pela ferramenta KITTool para o diagnóstico realizado “por prática” das áreas de processo “Ambiente de Teste” e “Teste Não Funcional”.**

Com esse diagnóstico foi possível perceber que existe um esforço por parte da empresa em definir e manter um processo de teste. Entretanto, esse processo ainda é imaturo e não realiza todas as atividades mínimas necessárias para que satisfaça os objetivos presentes no TMMi, mesmo que parcialmente segundo a definição de Höhn (2011), como mostrado na Figura 4.2. Outro ponto que pode ser observado refere-se ao fato de a empresa não documentar todas as atividades que são realizadas. A Tabela 4.1 apresenta em resumo o resultado do diagnóstico com a KITTool, mostrando as práticas com suas respectivas notas e justificativas. Com relação à ferramenta KITTool (HÖHN, 2011), pôde-se observar que apesar dela servir como um auxílio

para a compreensão e a avaliação de um processo de teste, é necessário profundo conhecimento do TMMi para sua utilização, uma vez que o título das práticas muitas vezes não é intuitivo. A escala de notas que deve ser atribuída na avaliação por prática exige um profundo conhecimento do processo sob avaliação, para que os resultados reflitam a realidade. Na seção seguinte é apresentada a comparação dos resultados da KITTool e da avaliação por mapa mental. (Essa é a etapa “Realizar Diagnóstico” da estratégia apresentada no Capítulo 5).

**Tabela 4.1: Resultados obtidos no diagnóstico realizado com a KITTool.**

<b>Prática</b>	<b>Nota</b>	<b>Justificativa</b>
PA2.1SP2.2	25%	Empresa adota alguma estratégia de teste
PA2.1SP3.1	25%	Existem métricas de teste definidas
PA2.2SP2.1	25%	Plano de teste não é baseado em análise de risco do produto
PA2.2SP3.2	25%	Não existem marcos de re-planejamento no ciclo de vida de teste
PA2.2SP3.3	25%	Não utiliza base histórica para definição de estimativas
PA2.2SP4.1	25%	Gestor define um cronograma e não o analista de teste
PA2.2SP4.2	25%	Atividade não documentada
PA2.2SP4.3	25%	Atividade não documentada
PA2.2SP4.5	50%	Nem todos os projetos tem a atividade de teste planejada
PA2.2SP5.1	50%	Revisão é realizada, mas não documentada
PA2.2SP5.2	25%	Replanejamento não é refletido no plano de testes
PA2.2SP5.3	50%	Atividade realizada informalmente
PA2.3SP1.2	25%	Gestor monitora informalmente se o ambiente é utilizado
PA2.3SP1.3	25%	Gestor acompanha atividades informalmente
PA2.3SP2.2	50%	Plano não é avaliado com base na lista de defeitos
PA2.3SP3.1	25%	Gestor acompanha a atividade mas não documenta
PA2.3SP3.2	25%	Replanejamentos são realizados, mas não documentados
PA2.4SP1.2	25%	Identifica casos de teste mas não prioriza
PA2.4SP1.4	100%	Possui software que permite manter a rastreabilidade
PA2.4SP3.2	100%	Executa os casos de teste planejados
PA2.4SP4.2	25%	Gestor encaminha ao desenvolvimento itens a serem corrigidos
PA2.5SP1.1	25%	Dados necessários para teste não são identificados
PA2.5SP2.1	25%	Um ambiente de teste é implementado mas não é testado
PA2.5SP3.4	25%	Problemas no ambiente são relatados, mas nem sempre corrigidos
PA3.4SP2.1	25%	Nem sempre testes não funcionais são realizados
PA3.4SP5.2	25%	Nem sempre testes não funcionais são realizados

### 4.3.3 Comparação de resultados

Nesta seção são comparados os resultados da avaliação por mapa mental e os resultados do diagnóstico com a KITTool (Tabela 4.2), que embora receba uma nota para cada prática sofre uma distorção do que está parcialmente contemplado. Essa distorção ocorre porque a tabela de referência utilizada para atribuição das notas no diagnóstico não permite identificar práticas que

**Tabela 4.2: Práticas encontradas no diagnóstico com KITTool e na avaliação com mapa mental.**

<b>Práticas presentes em ambos resultados</b>	<b>Somente Avaliação</b>	<b>Somente Diagnóstico</b>
Acompanhar incidentes de teste	X	
Analisar problemas		X
Conciliar trabalho e níveis de recursos	X	X
Decidir sobre incidentes com o grupo de controle de configuração	X	
Definir a abordagem de teste	X	
Definir ciclo de vida de teste	X	X
Definir critério de parada	X	
Definir critério de suspensão e recomeço	X	
Definir estratégia de teste		X
Definir indicadores de desempenho de teste		X
Determinar estimativas de esforço e custo de teste	X	X
Estabelecer o cronograma de teste	X	X
Estabelecer o plano de teste	X	X
Executar ações apropriadas para corrigir os incidentes de teste	X	X
Executar casos de teste	X	X
Identificar e priorizar casos de teste	X	X
Identificar e priorizar condições de teste	X	
Identificar elementos e características a serem testados	X	X
Identificar elementos e características não funcionais a serem testadas	X	X
Implementar o ambiente de teste	X	X
Manter rastreabilidade horizontal com requisitos	X	X
Monitorar compromissos de teste		X
Monitorar defeitos	X	X
Obter comprometimento com o plano de teste	X	X
Obter necessidades do ambiente de teste		X
Planejar a equipe de teste	X	X
Planejar o envolvimento dos interessados	X	X
Relatar e gerenciar incidentes do ambiente de teste		X
Relatar incidentes de teste	X	
Relatar incidentes de teste não funcional		X
Revisar o plano de teste	X	X
Tomar ação corretiva		X
Monitorar recursos do ambiente de teste disponibilizados e os utilizados em relação ao plano		X

tem implementação intermediária ao que é sugerido na tabela de referência. Um outro ponto a ser observado é que a atribuição de diferentes notas para o uso da prática, apesar de ser uma vantagem, torna o diagnóstico mais difícil se o conhecimento a respeito dos requisitos de cada prática for baixo. Por essa razão reitera-se que é necessário amplo conhecimento do TMMi, já que o título de cada prática nem sempre deixa claro o que realmente é pedido.

Comparando os resultados apresentados pela ferramenta e os resultados observados com a avaliação por mapa mental, observa-se que são semelhantes, pois há uma sobreposição de 17 práticas identificadas em ambas as avaliações, como pode ser visto na Tabela 4.2. Na avaliação por mapa mental foram identificadas 24 práticas que são realizadas, enquanto na avaliação com a KITTool foram identificadas 26 práticas. As práticas encontradas são listadas na Tabela 4.2, que também indica em qual resultado a prática está presente. Na tabela fica claro que as duas avaliações tiveram resultados próximos, pois a maioria das práticas foram observadas em ambos resultados. Isso mostra que a ferramenta é eficaz no diagnóstico, com a vantagem de detectar os diferentes níveis de utilização de cada prática.

As divergências entre os resultados pode se dever ao fato de que não havia o conhecimento necessário do TMMi e do processo da empresa para a realização de ambas as avaliações. Outro ponto que pode ter interferido no resultado é a dificuldade no entendimento do funcionamento da ferramenta. Esses resultados auxiliaram na construção do diagrama do processo real da empresa, pois permitiram identificar todas as atividades que são de fato realizadas. Na Seção 4.4 é apresentado o processo real da empresa remodelado a partir do conhecimento obtido por meio da realização do diagnóstico do processo.

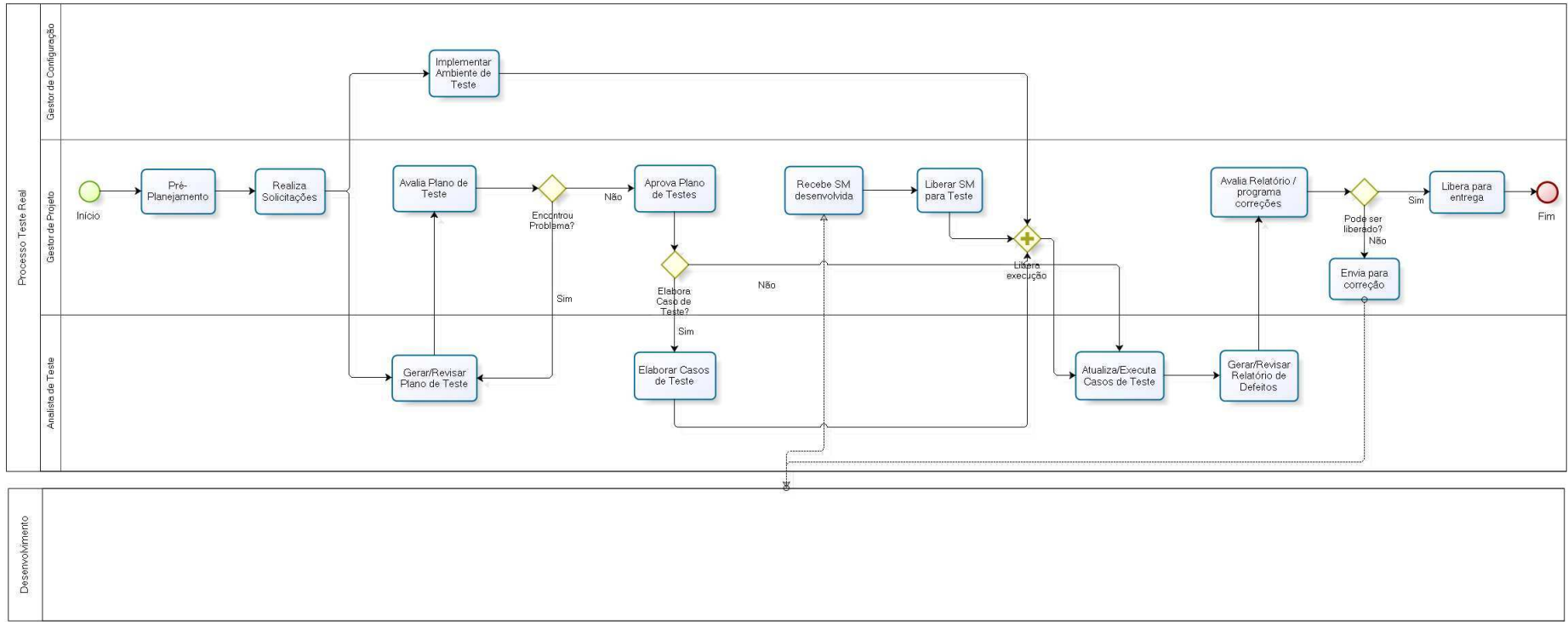
## 4.4 Processo Real

Durante a avaliação do processo da empresa, relatada neste capítulo, com base no mapa mental do TMMi proposto por Höhn (2011), identificou-se a necessidade de remodelar esse processo, pois algumas atividades que notadamente eram realizadas não estavam representadas. Para a realização dessa etapa do trabalho, estabeleceu-se como ponto de partida os diagramas do processo de teste, desenhados pela própria empresa e os resultados do diagnóstico, gerando-se o desenho do processo real.

O processo real, apresentado na Figura 4.6, foi validado pelo analista de teste (essa atividade é a etapa “Validar Processo Real” da estratégia apresentada no Capítulo 5). De maneira geral, a empresa executa o processo que estava modelado, porém que não apresentava explicitamente todas as atividades realizadas e não contemplava as atividades realizadas pelo gestor de projeto. Na Figura 4.6 pode-se observar que há três perfis identificados que executam as atividades de teste: *Gestor de Projeto*, que define alguns itens de planejamento, acompanha a atividade de teste e supervisiona o analista de teste; *Gestor de Configuração*, que é responsável por implementar o ambiente de teste; e *Analista de Teste*, responsável pelo planejamento e execução dos testes. O processo de desenvolvimento, relativo à fábrica de software, é denotado

---

na Figura 4.6 como um “*pool*” vazio. Observe que a comunicação entre o processo de teste e o processo de desenvolvimento se dá apenas em dois pontos: entre o processo de desenvolvimento e a atividade “Recebe SM Desenvolvida” e entre a atividade “Envia para correção” e o processo de desenvolvimento. A comunicação ocorre por intermédio do gestor de projeto em ambos os casos.



**Figura 4.6: Processo de teste da empresa remodelado, representando o Processo de Teste Real.**



No processo real da empresa, a atividade “Pré-Planejamento” representa as tarefas que o gestor de projeto realiza para a definição da equipe que executará os papéis relacionados ao teste e o cronograma das atividades de teste, que determina quando as atividades serão iniciadas e quando deverão ser encerradas. O gestor utiliza um critério para determinar o fim das atividades de teste em um projeto, porém esse critério não é definido formalmente. Após essas definições, o gerente de projeto solicita ao gestor de configuração a implementação do ambiente de teste. Os requisitos de ambiente de teste são identificados e fornecidos ao gestor de configuração, porém o modelo de plano de testes indica que as definições de ambiente encontram-se no “Plano de Projeto Interno”, documento que é elaborado pelo gestor de projeto conforme as especificações fornecidas pelo cliente, e guia a execução do projeto de software. Como o gestor de configuração também executa atividades no processo de desenvolvimento, ele já está ciente de qual ambiente deve ser implementado para o projeto em questão.

Juntamente com a solicitação de implementação do ambiente, o gestor de projeto solicita ao analista de teste a elaboração do Plano de Teste. O plano de teste deve seguir um modelo padrão da empresa, o qual indica que nesse documento serão definidos os itens de teste, quais características serão testadas e quais não, qual a abordagem de teste que será utilizada, além de fornecer os critérios de classificação de defeitos e gravidade. O Plano de Teste faz menção ao ambiente e ao cronograma de testes, indicando onde tais itens estão definidos. O plano é revisado pelo gestor de projeto, que pode aprová-lo ou solicitar correções ao analista de teste. Com o Plano de Teste aprovado, o analista de teste inicia a elaboração dos casos de teste com base nos documentos Plano de Testes, Plano de Projeto Interno e Especificação de Requisitos, sendo este último o documento que contém os requisitos detalhados do projeto.

Paralelamente ao processo de teste, o processo de desenvolvimento é executado. Cada requisito do sistema é tratado como uma Solicitação de Mudança (sigla SM na Figura 4.6) que é enviada ao programador, que após concluir o desenvolvimento e testes unitários, sinaliza para o gestor que a SM foi desenvolvida e está pronta para ser testada. Se o ambiente de teste estiver implementado, os casos de teste desenvolvidos e existirem SM's disponíveis para teste, o gestor de projeto libera essas SM's para serem testadas. O analista de teste atualiza os casos de teste, caso seja necessário, e então os executa. As falhas observadas nos testes são inseridas na área do projeto na ferramenta utilizada para o gerenciamento dos documentos do projeto. Após a execução de todos os casos de teste, gera-se um relatório que é enviado para ser analisado pelo gestor de projeto. O gestor avalia as falhas reportadas e programa as correções. Conforme as correções são feitas, a SM volta para o gestor, que então analisa a correção e libera novamente para teste. Caso o gestor perceba que há itens testados que podem ser liberados, segundo algum critério de parada definido previamente, ele então libera o sistema, ou partes dele, para entrega.

Nesse momento o sistema passa a seguir um outro processo da empresa, o de “Liberação e Entrega”. O gestor de projeto afirma que a definição do critério de parada depende do projeto e em alguns casos o sistema pode ser liberado sem as correções terem sido realizadas, mas o cliente é informado por meio de um “Release Notes”.

Um dos pontos observados no processo fornecido pela empresa é que os casos de teste elaborados não são a entrada da atividade de execução dos casos de teste. Somente quando isso foi questionado o analista de teste informou que isso se deve ao fato de que nem sempre os casos de teste são elaborados, ou documentados. Como a elaboração de casos de teste ocorre somente para alguns projetos a empresa optou por não colocar como entrada na atividade de execução. Sendo assim, foi inserido uma decisão antes da atividade “Elaborar casos de Teste” (Figura 4.6), mostrando que há um fluxo alternativo nos casos em que essa atividade não é executada. O fluxo alternativo leva à atividade “Executar Casos de Teste”. Após a liberação da SM pelo gestor, o analista executa os testes.

Com as análises descritas nesta seção foi possível delinear o processo que a empresa realmente utiliza para testar os produtos que desenvolve. A elaboração desse processo deu-se a partir de entrevistas com a equipe de teste, com o gestor e com o analista de qualidade responsável pela elaboração dos processos na empresa para a certificação MPR.Br. Após sua elaboração, o diagrama apresentado na Figura 4.6 foi validado pelo analista de teste, confirmando que trata-se de fato do processo executado pela empresa. (Essa é a etapa “Modelar Processo Real” da estratégia apresentada no Capítulo 5).

## 4.5 Análise Crítica

Nesta seção apresentam-se a análise crítica do estudo realizado nesta empresa, que está dividido em duas partes: comparação dos processos modelados e proposta de um processo mínimo para a empresa. Com base nas etapas realizadas anteriormente coletaram-se informações que permitiram modelar dois processos: o Processo Reduzido (PR) e o Processo Real da Empresa (PRE). Os mapas mentais dos resultados do Capítulo 3, apresentados nas Figuras 3.7, 3.8, 3.9, 3.10 e 3.11 foram reunidos em um único mapa mental para melhor visualização de quais práticas do TMMi fazem parte em cada fase do processo genérico de teste. Esse mapa mental representa o processo reduzido, gerado a partir do resultado do *survey*, e é apresentado na Figura 4.7. Vale lembrar que nas folhas desse mapa mental, em azul, estão as práticas do TMMi que segundo a comunidade de teste são essenciais em um processo de teste.

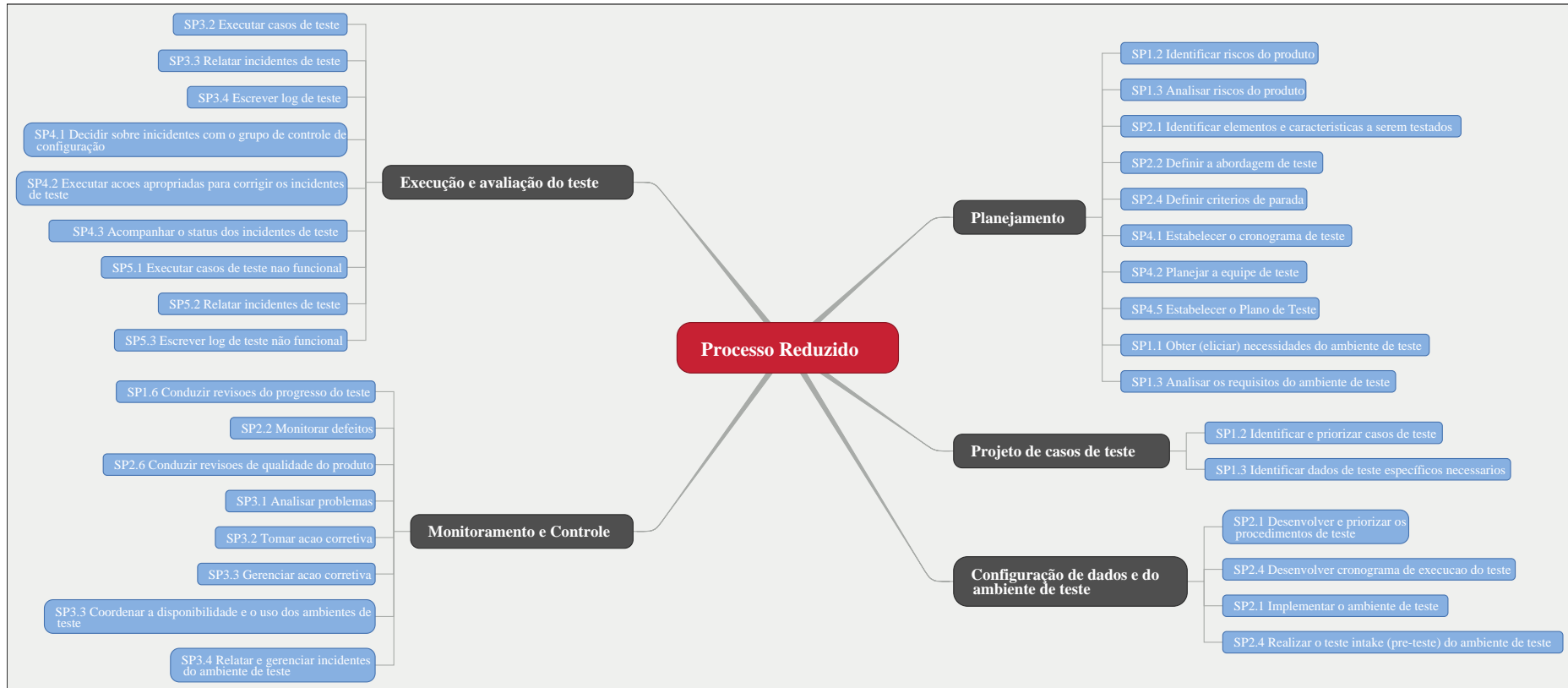


Figura 4.7: Processo Reduzido (PR), apresentado no Capítulo 3, na forma de mapa mental para melhor visualização.

Para a comparação foram observadas as práticas presentes no processo proposto da Figura 4.7 e as atividades presentes no processo de teste em uso pela empresa. O resultado dessa comparação pode ser observado no mapa mental da Figura 4.8. Esse mapa mental foi construído tendo como base o processo real da empresa e o processo reduzido. Do lado esquerdo da figura está o processo real da empresa e do lado direito o processo reduzido. No centro da figura estão as práticas do TMMi, presentes no processo proposto que tem ligação o processo real da empresa. Algumas atividades do processo real não foram incluídas neste mapa, pois são atividades que estão relacionadas à institucionalização do processo, que não foi tratada neste trabalho.

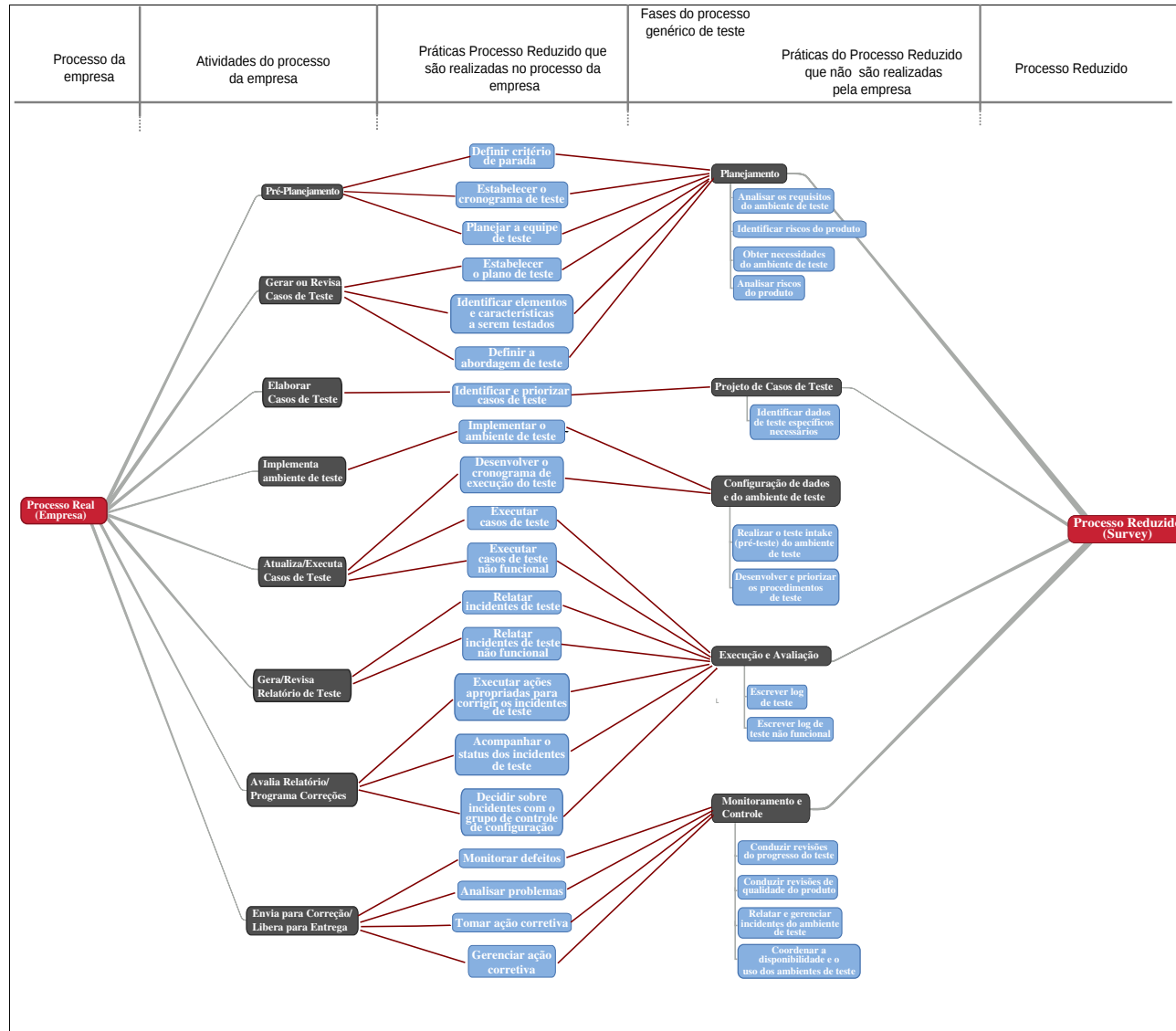


Figura 4.8: Mapa mental do processo real da empresa. As folhas em azul coincidem com as práticas do processo reduzido, apresentado no Capítulo 3, que possuem relação com a respectiva atividade do processo real.

Pode-se observar na Figura 4.8 que o processo utilizado pela empresa contempla todas as fases de um processo genérico de teste. Entretanto, nem todas as práticas presentes no processo proposto da Figura 4.7 possuem correspondência com o processo da empresa. Das 33 práticas consideradas essenciais, conforme resultado do survey conduzido, pode-se afirmar que 19, ou 23% das práticas, são implementadas atualmente, mesmo que parcialmente. A associação entre as atividades do processo da empresa e as práticas do TMMi foi realizada por meio da análise do que é solicitado para a prática no TMMi e o que a empresa realmente executa em seu processo.

A fase de Planejamento é parcialmente satisfeita pelo processo da empresa, pois apenas seis práticas são executadas pela empresa. As práticas realizadas pelo gestor de projeto, que fazem parte da fase de planejamento são *Definir critérios de parada*, *Estabelecer o cronograma de teste* e *Planejar a equipe de teste*. Essas três práticas foram associadas à atividade “Pré-Planejamento”, pois é nesta atividade que o gestor define esses itens para dar início à atividade de teste, conforme apresentado na Seção 4.4. Dentre as atividades executadas pelo analista de teste, identificou-se que três práticas da fase de planejamento estão relacionadas, são elas: *Estabelecer plano de teste*, *Identificar elementos e características a serem testados* e *Definir a abordagem de teste*. Conforme dito na Seção 4.4, o analista de teste é o responsável por elaborar e manter o Plano de Teste, documento no qual são definidos a abordagem e identificados os elementos a serem testados. No processo da empresa não são encontradas atividades relacionadas à análise de risco do produto, identificadas pelas práticas *Identificar riscos do produto* e *Analisar riscos do produto*. Também não foram identificadas atividades relacionadas ao desenvolvimento de requisitos do ambiente de teste, identificado pelas práticas *Obter requisitos do ambiente de teste* e *Analisar requisitos do ambiente de teste*.

Na fase de Projeto de Casos de Teste, a prática *Identificar e priorizar casos de teste* está relacionada ao processo da empresa porque existe a atividade descrita para elaboração de casos de teste. Pode-se afirmar que essa prática é implementada parcialmente porque a empresa não prioriza os casos de teste, já que não é feita análise de risco de produto e também porque essa atividade não é realizada para todos os projetos da empresa. A prática *Identificar dados específicos necessários* não pode ser associada ao processo da empresa, pois esta prática não foi identificada dentre as atividades realizadas pela empresa. Na fase de Configuração de Dados e do Ambiente de Teste foram identificadas duas práticas que são executadas pela empresa: *Implementar ambiente de teste* e *Desenvolver cronograma de execução de teste*. O ambiente de teste é implementado pelo gestor de configuração, após a solicitação do gestor de projetos. O cronograma de execução de teste, por sua vez, é uma atividade do analista de teste, que programa a execução dos casos de teste após receber as solicitações de Mudança do gestor de projeto. *Realizar teste intake (pré-teste) do ambiente de teste* e *Desenvolver e priorizar proce-*

*dimentos de teste* são práticas do processo proposto que não são implementadas pela empresa.

Algumas práticas associadas a fase de execução estão distribuídas entre as atividades realizadas pelo analista de teste e as realizadas pelo gestor de projeto. O analista de teste é o responsável por executar os casos de teste, inclusive não funcionais, e relatar os defeitos encontrados classificando-os por gravidade. Essas atividades estão relacionadas às práticas *Executar casos de teste*, *Executar casos de teste não funcional*, *Relatar incidentes de teste* e *Relatar incidentes de teste não funcional*. Após a execução, o analista fornece o relatório de falhas ao gestor de projeto, que o analisa para decidir sobre as ações a serem tomadas para cada falha, programa as correções e acompanha o status da correção à medida que a tarefa é realizada e os testes são reexecutados. Essas atividades estão relacionadas às práticas *Decidir sobre incidentes com o grupo de controle de configuração*, *Executar ações apropriadas para corrigir defeitos* e *Acompanhar status do incidente de teste*. Durante a análise do relatório, o gestor pode aceitar ou não um item que foi listado como falha, pode modificar a gravidade, mas essas ações não são realizadas em conjunto com o grupo de controle de configuração. Por essa razão, a prática *Decidir sobre incidentes com o grupo de controle de configuração* é implementada parcialmente. As práticas relacionadas à escrita de *log* de teste e *log* de teste não funcional não são implementadas pela empresa.

Por fim, a fase de Monitoramento e Controle, que acompanha toda a execução do processo teste possui algumas práticas que são implementadas parcialmente, e realizadas pelo gestor de projeto. No momento da liberação do sistema, o gestor de projeto realiza o acompanhamento da atividade de teste, observando se o sistema pode ser liberado, se as correções programadas foram executadas, se há falhas reportadas ainda não investigadas, e observa se o tempo e orçamento existente são suficientes para continuar a atividade. Essas tarefas podem ser associadas às práticas *Monitorar defeitos*, *Analisar Problemas*, *Tomar ação corretiva* e *Gerenciar ação corretiva*. Para as práticas de revisões de progresso e de qualidade não foram identificadas atividades relacionadas, no processo que a empresa utiliza, assim como as práticas relacionadas ao monitoramento do ambiente de teste.

Observando a Tabela 4.2, que contém o resultado do diagnóstico do processo da empresa, e a Figura 4.7, que contém as práticas consideradas essenciais, foram identificadas as seguintes práticas em comum:

- *Identificar elementos e características a serem testados*
- *Estabelecer o cronograma de teste*
- *Planejar a equipe de teste*

- *Estabelecer o plano de teste*
- *Identificar e priorizar casos de teste*
- *Implementar o ambiente de teste*
- *Executar casos de teste*
- *Executar ações apropriadas para corrigir os incidentes de teste*
- *Monitorar defeitos*

Como o processo utilizado para comparação foi o processo real modelado após o diagnóstico do processo atual, nota-se que essas práticas já estão associadas com o mapa mental da Figura 4.8, ou seja, o diagnóstico reforça o fato de que a empresa de fato já implementa algumas das práticas mais importantes do TMMi. A empresa iniciou o desenvolvimento de um novo processo de teste para a futura fábrica de teste. No entanto esse processo não foi concluído, mas a empresa apresentou um diagrama modelado por ela, que continha algumas atividades. Dessa forma pode-se identificar o desejo da empresa em realizar atividades relacionadas a análise de risco de produto. Por essa razão foi incluída a prática *Realizar análise de risco de produto* e a priorização dos casos de teste com base nessa análise, na sugestão de processo para a empresa, apresentado na Figura 4.9. Para que as outras práticas também sejam satisfeitas é possível acrescentar algumas tarefas às atividades já existentes (destacadas em verde na Figura 4.9), gerando desta forma um processo mínimo, que esteja de acordo com a realidade da empresa e com baixo impacto na cultura da empresa, já que não serão necessárias mudanças na equipe e na hierarquia atuais. Considera-se que este processo contém as atividades e tarefas necessárias para se satisfaça ao conjunto mínimo de práticas do TMMi. Nota-se que neste mapa mental estão descritas atividades que satisfazem às práticas do TMMi que estão presentes no processo reduzido. (Essa etapa do trabalho equivale à etapa “Propor Processo” da estratégia apresentada no Capítulo 5.





Figura 4.9: Processo atual da empresa com novas atividades e tarefas incluídas para que atenda ao conjunto mínimo de práticas do TMMi consideradas muito importantes conforme resultado do survey.

O processo atual da empresa não é completo, porém já implementa boa parte das práticas necessárias para tal. Por essa razão acredita-se que o impacto da mudança do processo seja baixo, pois, as principais mudanças consistem em documentar artefatos que já são produzidos, além de elaborar e incluir novos itens como parte da documentação existente. As tarefas e documentos a serem incluídos visam a facilitar o trabalho de quem executa e melhorar a qualidade do processo de teste.

## 4.6 Considerações Finais

Neste capítulo foi apresentado o trabalho realizado em parceria com uma pequena empresa desenvolvedora de software em busca de melhorias em seu processo de teste. À época da escrita dessa dissertação a empresa possuía um processo de teste, e que foi o ponto de partida para a realização deste estudo. Realizou-se uma avaliação do processo de teste da empresa comparando-o com as práticas do TMMi, buscando-se identificar quais dessas práticas estão contempladas nesse processo. Para essa avaliação foi utilizado o mapa mental definido por Höhn (2011). Em seguida realizou-se um diagnóstico mais profundo, identificando quais práticas e o nível de implementação das práticas do TMMi no processo real da empresa. Para este diagnóstico foi utilizada a ferramenta KITTool, também desenvolvida por Höhn (2011). Como a ferramenta não havia sido utilizada, até então, por uma empresa real, a primeira avaliação foi realizada com o intuito de verificar o resultado fornecido pela KITTool. A realização dessas etapas permitiu um conhecimento mais aprofundado do processo da empresa, de seu contexto e das atividades que são realizadas para que os testes sejam executados e seus produtos entregues com melhor qualidade. Como foram identificadas atividades de apoio que são realizadas como parte do processo de teste, tornou-se evidente a necessidade de remodelar o processo da empresa de forma a refletir o que ocorre na prática.

Com esse conjunto de informações, foi possível realizar uma análise crítica do processo da empresa utilizando como base as práticas mais importantes do TMMi, segundo a comunidade de profissionais de teste de software. Como resultado obteve-se uma sugestão de processo para a empresa que incluiu algumas práticas no processo que a empresa utiliza atualmente. Com isso, nota-se que seriam necessárias somente algumas adaptações do processo atual, para que se cumpra o conjunto mínimo de práticas do TMMi. Esses resultados foram apresentados aos responsáveis da empresa, que demonstraram interesse em implantar o processo proposto já que, segundo eles, é um processo viável por não impactar fortemente a cultura existente e também por não implicar no aumento de recursos destinados à área de testes na empresa.

Como a empresa objetiva oferecer o teste de software como um serviço, além de testar o que é desenvolvido por ela, o processo aqui sugerido pode ser adaptado para tal. Atividades relacionadas ao gerenciamento da venda do serviço e à garantia de qualidade desse serviço devem ser consideradas no momento de implementar a fábrica de teste, pois tanto a empresa quanto o cliente devem possuir meios de medir a qualidade do serviço da fábrica de teste, e para isso são necessárias atividades específicas que não fizeram parte do escopo deste trabalho. Como a empresa já possui o *know-how* de fábrica de software, acredita-se que esse mesmo conhecimento possa ser aplicado ao contexto de teste e que o processo aqui proposto possa ser facilmente adaptado para o modelo de fábrica de teste.

Como o objetivo desta pesquisa foi o desenvolvimento de uma estratégia para definição de um processo inicial de teste para pequenas empresas de software, o Capítulo 5 apresenta uma síntese do trabalho apresentado neste capítulo, na forma de uma estratégia que possa ser reutilizada por outras empresas. Assim, espera-se que a estratégia aqui proposta sirva de roteiro para a definição do processo inicial de teste de outras empresas, em especial as de pequeno porte, que possuem poucos recursos, equipe reduzida e pouca maturidade em teste de software.



# Capítulo 5

## ESTRATÉGIA

---

---

Este capítulo apresenta a estratégia utilizada na elaboração de um processo de teste para empresas desenvolvedoras de software. Espera-se que a generalização dessa estratégia permita sua aplicação para outras empresas, especialmente as de pequeno porte, que desejem implantar ou melhorar o processo de teste existente.

### 5.1 Considerações Iniciais

Kasurinen et al. (2011) afirmam que projetos de software têm quatro objetivos principais: (i) produzir as funcionalidades exigidas; (ii) atingir nível de qualidade aceitável; (iii) respeitar o prazo estipulado; e (iv) não exceder o orçamento. Para pequenas empresas, atingir esses objetivos pode ser essencial para o sucesso do negócio. No entanto, nem sempre é possível cumprir esses quatro objetivos devido ao orçamento reduzido e prazos muito exigentes, o que obriga a empresa a dirigir o esforço no cumprimento dos prazos, implementando as funcionalidades que foram solicitadas. Mas esse cenário vem mudando com o aumento da concorrência no mercado de software. Pequenas empresas têm buscado meios de mostrar que são capazes de produzir software de qualidade por meio da obtenção de certificações em modelos de maturidade e uma maior preocupação com atividades de garantia de qualidade. Dentre as atividades de garantia de qualidade, o teste de software tem ganhado importância, pois sabe-se que o custo de um defeito em um produto entregue pode ir além do ponto de vista financeiro e afetar a imagem da empresa.

Apesar do esforço em melhorar a qualidade de seus produtos, Rodrigues, Pinheiro e Albuquerque (2010) afirmam que as pequenas empresas brasileiras encontram barreiras para implementar um processo de teste e, segundo a revisão bibliográfica deste artigo, os fatores que foram mais mencionados são: falta de profissionais especializados em teste de software, grande vari-

idade de técnicas relacionadas a teste, falta de conhecimento para executar adequadamente as atividades de teste, e dificuldade para adaptar modelos de maturidade para o ambiente específico da organização. Este último fator deixa em evidência um dos pontos citados no Capítulo 3 com relação ao TMMi (TMMi Foundation, 2010), que além de ser de difícil compreensão, não deixa claro quais práticas são prioritárias caso o objetivo de sua utilização seja uma implementação parcial para a elaboração de um processo de teste mais enxuto em uma pequena empresa.

Diante desse cenário, realizou-se um trabalho de elaboração de um processo de teste para uma pequena empresa, que foi apresentado no Capítulo 4. A experiência na elaboração desse processo permitiu extrair uma estratégia que pode ser utilizada por outras empresas que desejem definir um novo processo de teste ou apenas implementar melhorias no processo que já possui. A definição dessa estratégia é apresentada detalhadamente na Seção 5.2, a qual sintetiza os passos relacionados ao estudo de caso apresentado no Capítulo 4. Na Seção 5.3 são apresentadas as considerações finais deste capítulo.

## **5.2 Definição da Estratégia**

O trabalho apresentado no Capítulo 4 consistiu na definição de melhorias no processo de teste de uma empresa de pequeno porte. A empresa já possui um processo em utilização, mas seu objetivo é criar uma fábrica de teste para, além de oferecer um novo serviço, testar os produtos que são desenvolvidos internamente. Este trabalho focou na melhoria do processo de teste, deixando para a empresa, que é uma fábrica de software, o papel de adaptar o processo para o modelo de fábrica conforme sua necessidade. O método de elaboração do processo, em conjunto com a empresa, foi generalizado para que outras empresas também possam utilizá-lo como forma de definir ou melhorar seu processo de teste.

O enfoque dado neste trabalho foi para pequenas empresas porque, em geral, empresas desse porte encontram maior dificuldade na elaboração de um processo de teste, principalmente pela falta de tempo e pela baixa importância dada a esse tipo de atividade (RODRIGUES; BESSA; PINHEIRO, 2010). Geralmente o tempo dedicado aos testes é sacrificado quando o projeto atrasa, mas o prazo precisa ser atendido. Assim, com o tempo curto o teste é realizado somente no fim do projeto. Para pequenas empresas essa situação é comum, mas um processo de teste bem definido e institucionalizado pode reduzir o impacto dessa compressão do tempo e melhorar a qualidade dos produtos oferecidos. Em suma, a estratégia aqui apresentada objetiva auxiliar essas empresas na definição de um processo de teste de maneira simplificada, indicando quais práticas do TMMi são prioritárias em uma definição de um processo de teste. Espera-se

que o processo elaborado, a partir da utilização dessa estratégia, contemple todas as fases de um processo genérico de teste e que seu custo de implantação não seja elevado, pois o processo é definido de acordo com os recursos disponíveis na empresa.

Um processo consiste na realização de uma série de atividades para cumprir um propósito, produzindo uma saída tangível a partir de uma determinada entrada dada (HASS, 2008). O objetivo de um processo de teste é a realização de atividades que servem para diminuir a quantidade de defeitos presentes e, conseqüentemente, aumentar a qualidade do produto desenvolvido. Modelos de maturidade específicos para teste de software têm surgido, dada a importância dessa atividade. Tais modelos visam a apoiar a elaboração e melhoria de processo de teste. Entretanto, modelos de maturidade são de difícil compreensão e não identificam quais as atividades são prioritárias para que possa ser elaborado um processo enxuto, que exija um número reduzido de executores e de atividades a serem realizadas. Como forma de identificar quais são as atividades mais importantes de um processo de teste, segundo o modelo de maturidade TMMi (TMMi Foundation, 2010), foi feita uma pesquisa com a comunidade de profissionais de teste de software e seu resultado foi apresentado no Capítulo 3. O resultado obtido foi um modelo de processo reduzido, que possui apenas as práticas consideradas essenciais para composição de uma primeira versão do processo (Figura 4.7). Esse modelo serviu como um guia para obtenção de uma proposta de processo, apresentada no Capítulo 4, para uma empresa de pequeno porte. As etapas da estratégia de elaboração de um processo de teste para empresas de software de pequeno porte são descritas a seguir. A Figura 5.1 ilustra a ordem de execução dessas etapas.

**Realizar o Diagnóstico:** A primeira etapa da estratégia consiste em realizar o diagnóstico da situação atual do teste de software na empresa, visando a identificar quais atividades de teste já são realizadas. O objetivo dessa etapa é conhecer melhor a empresa por meio de seus processos de desenvolvimento e teste, caso existam. Os passos a serem realizados são:

1. Identificar as atividades de teste que são realizadas;
2. Conhecer como são realizadas essas atividades de teste identificadas;
3. Identificar quais os papéis dos envolvidos e qual o tamanho da equipe de teste;
4. Realizar diagnóstico com KITTool (HÖHN, 2011), para mapear quais práticas do TMMi são implementadas pela empresa e o quanto de cada prática está implementado, conforme o exemplo apresentado no Capítulo 4, na Seção 4.3.

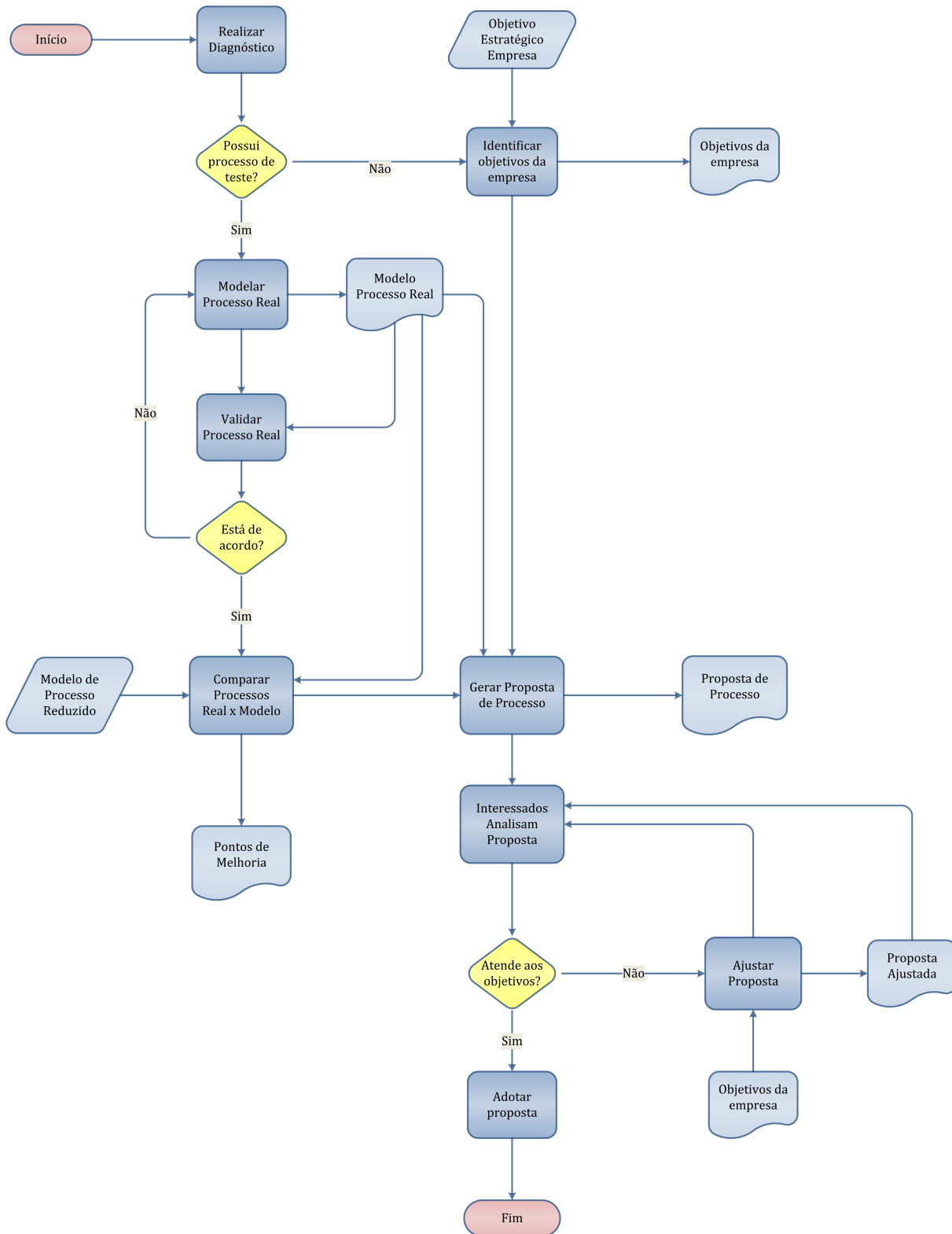


Figura 5.1: Fluxograma da estratégia para desenvolvimento de processo de teste.



A utilização da KITTool facilita o diagnóstico do processo na seguinte forma: além de permitir detectar os diferentes graus de enfoque dado a cada prática que é implementada, o resultado gerado pela ferramenta indica quais áreas de processo estão mais completas. Esse resultado é útil quando comparado com o resultado da utilização dessa estratégia, pois pode guiar a empresa indicando as práticas que podem ser implementadas primeiro para cumprir determinada área de processo. Caso a empresa possua processo de teste a próxima etapa é a modelagem do processo real. Se a empresa não realizar atividades de teste a próxima etapa deve ser a identificação dos objetivos da empresa.

**Identificar Objetivos da Empresa:** Com a realização do diagnóstico será possível identificar se a empresa possui ou não processo de teste definido. Caso a empresa não possua, ou não realize atividades de teste, devem ser realizados os seguintes passos:

1. Identificar o tamanho da equipe que a empresa poderá dispor para a atividade de teste;
2. Identificar o objetivo da empresa com relação à atividade de teste

Essas informações serão relevantes no momento da elaboração da proposta de processo, em que se identificam os possíveis papéis para a execução das atividades sugeridas e as atividades podem ser propostas alinhadas ao objetivo da empresa. Para a realização desta etapa é necessário que a empresa forneça informações sobre seus objetivos estratégicos, desde que estejam relacionados à atividade de teste ou desenvolvimento.

**Desenhar o Modelo do Processo Real:** A realização do diagnóstico permite conhecer a empresa e a forma como o teste de software é tratado em seu processo de desenvolvimento. Como discutido no Capítulo 4 (Seção 4.4), foi nessa etapa que se notou que nem todas as atividades realizadas como parte do processo de teste estavam documentadas. Por essa razão, foi realizada a modelagem do processo real desta empresa, para que todas as atividades relacionadas ao teste de software pudessem estar representadas.

1. Desenhar diagrama do processo real com base nas atividades identificadas na etapa anterior;
2. Incluir os papéis responsáveis por cada atividade no diagrama;

A realização desta etapa gera um documento, como mostrado na Figura 5.1, que é o “Modelo de Processo Real”. Esse modelo contém o diagrama do modelo de processo real que deve ser validado pela empresa.

**Validar Processo Real:** Com as atividades identificadas e o processo real devidamente modelado, é importante que esse processo seja validado pelo responsável pela equipe de teste da empresa para confirmar que as atividades ali representadas são de fato realizadas, e que os papéis foram identificados corretamente e estão de acordo com suas funções dentro do processo.

1. Enviar diagrama do processo real para validação;
2. Empresa valida o diagrama;
3. Caso haja inconsistências o diagrama deve ser remodelado.

No estudo de caso realizado, apresentado no Capítulo 4, a empresa foi solicitada a verificar se todas as atividades e papéis presentes no diagrama representavam de fato como a atividade de teste é realizada, incluindo as tarefas que não são executadas formalmente ou que apenas não são documentadas.

**Comparar Processos:** Deve ser conduzida uma comparação entre o processo real validado e o “Modelo de Processo Reduzido”, proposto no Capítulo 3 e mostrado na Figura 4.7. O objetivo dessa comparação é detectar quais atividades do modelo de processo reduzido já estão contempladas no processo que a empresa utiliza, e quais atividades não estão, identificando os pontos de melhoria.

1. Verificar quais práticas do “Modelo de Processo Reduzido” estão representadas no Processo Real;
2. Verificar as práticas que estão presentes no “Modelo de Processo Reduzido” e que não estão no Processo Real;
3. Listar as práticas que estão implementadas, total ou parcialmente, e as que ainda não estão implementadas;
4. Gerar documento de pontos de melhoria constando as atividades que devem ser incluídas ou retiradas.

Essa comparação foi realizada no estudo de caso apresentado no capítulo anterior (Seção 4.5). O mapa mental da Figura 4.8 mostra o resultado da comparação feita com a empresa, com a qual se realizou o estudo de caso, indicando quais práticas já são implementadas e a que fase pertencem.

**Gerar Proposta de Processo:** O objetivo dessa etapa é a elaboração do novo processo da empresa, para que esta possa avaliar.

1. Elaborar processo de acordo com os pontos de melhoria identificados na fase anterior;
2. Identificar papéis necessários para a execução das atividades do processo;
3. Enviar proposta para análise da empresa.

No estudo de caso apresentado no capítulo anterior, foram sugeridas melhorias no processo de teste da empresa. A Figura 4.7 mostra o processo proposto à empresa após a realização das etapas acima.

**Interessados Analisam Proposta:** A proposta de processo deve ser apresentada aos gestores, equipe de teste e outros interessados na implantação do processo de teste. A empresa deve observar os seguintes itens:

1. Viabilidade de implementação do processo proposto;
2. Se está de acordo com os interesses da empresa;
3. Se a empresa poderá disponibilizar os recursos necessários para sua implantação;
4. Os papéis propostos estão de acordo com a capacidade da empresa.

**Ajustar Proposta:** Caso a proposta não atenda às expectativas, ela deve ser ajustada, sempre levando em consideração o contexto e os objetivos da empresa até que seja aprovada, como mostrado na Figura 5.1.

1. Ajustar proposta de processo de acordo com objetivos da empresa;
2. Gerar documento contendo a proposta ajustada.

**Adotar Proposta:** Após a aprovação do processo que foi proposto com base na utilização da estratégia apresentada na Figura 5.1, a empresa deve tomar as ações necessárias para que o novo processo possa ser implantado. Sugere-se que após a implantação do processo seja realizado um novo diagnóstico utilizando a ferramenta KITTool (HÖHN, 2011), para que se possa

acompanhar a evolução do processo de teste da empresa. Esse diagnóstico também fornece uma visão de quais práticas do TMMi ainda podem ser implementadas.

Essa estratégia pode ser aplicada com facilidade, visto que ela oferece diretrizes de quais práticas devem ser implementadas para satisfazer cada fase de um processo de teste. Além disso, a estratégia leva em consideração o contexto da empresa utilizando como ponto de partida o conhecimento e a prática existentes na empresa. O modelo de processo de teste utilizado na etapa de comparação de processos indica quais práticas do TMMi são prioritárias para implantação, portanto indica para a empresa o que implantar primeiro para que se tenha um processo definido que pode ser controlado, avaliado e melhorado. Espera-se que a estratégia apresentada possa ser aplicada pela própria equipe da empresa que deseja implantar o processo de teste.

### **5.3 Considerações Finais**

Neste capítulo foi apresentada a estratégia de elaboração de processo de teste tendo como base um modelo de processo sintetizado que indica quais práticas do TMMi são prioritárias. Essa estratégia permite que empresas desenvolvedoras de software de pequeno porte realizem atividades de teste de maneira formal, pois toda a atividade passa a ser planejada e documentada. Essa estratégia também permite que o processo definido possa ser melhorado, acompanhando o crescimento da empresa, tendo em vista que a ferramenta de diagnóstico fornece indicativos de que outras práticas podem ser implementadas além das práticas do modelo de processo sintetizado. Esse modelo de processo sintetizado sugere o que pode ser implementado primeiro para que se tenha um processo mínimo que compreenda o planejamento, projeto de casos de teste, configuração de ambiente, execução e acompanhamento.

A estratégia aqui apresentada foi elaborada a partir da definição de um processo de teste de uma pequena empresa feita com base no indicativo de quais práticas do TMMi são essenciais para um processo de teste. Como a empresa possuía um processo em uso, ele foi levado em consideração e o processo proposto consistiu em uma melhoria do processo atual para que algumas práticas, consideradas essenciais, fossem acrescentadas. Acredita-se que a estratégia descrita neste capítulo possa ser utilizada por outras empresas, também de pequeno porte. Para isso, os passos realizados foram generalizados de forma que possam ser reproduzidos e apresentem o mesmo resultado, um novo processo de teste definido, ou a identificação de melhorias no processo existente. O próximo capítulo apresenta as conclusões do trabalho realizado, as contribuições obtidas, as respectivas limitações e possíveis trabalhos futuros.

# Capítulo 6

## CONCLUSÃO

---

---

Neste trabalho foi apresentada uma estratégia que fornece um roteiro para definição ou melhoria de processo de teste, para empresas de software. A construção dessa estratégia levou em consideração a opinião de profissionais que trabalham com teste de software, e muitos deles em empresas com alguma certificação em modelos de maturidade de processo, o que confere um maior grau de maturidade à esses profissionais. O TMMi foi utilizado como base para a construção dessa estratégia, pois é um modelo de referência específico para o teste de software e permite que o processo elaborado, com base na estratégia, tenha maior qualidade. O trabalho aqui apresentado foi realizado em parceria com uma pequena empresa e teve como foco a melhoria do seu processo de teste, pois o objetivo dessa empresa é oferecer o teste de software como um serviço no modelo de fábrica. O enfoque deste trabalho foi somente em práticas relacionadas ao teste de software e não envolveu a elaboração de processos essenciais para uma fábrica, como a garantia de qualidade do serviço oferecido e o gerenciamento da venda desse serviço.

Um modelo de processo reduzido foi construído com base em um *survey* que buscou identificar, junto aos profissionais que trabalham com teste, quais são as práticas prioritárias do TMMi. A quantidade de respostas à essa pesquisa permitiu analisar estatisticamente os dados e como forma de conferir os resultados foi feita também uma análise dos números absolutos. O resultado dessa pesquisa foi um conjunto de práticas que contempla todas as fases de um processo genérico de teste e representa um processo mínimo que deve ser implementado para que a atividade de teste possa ser executada de maneira organizada e estruturada. O processo resultante possui 40% das práticas do TMMi. Esse processo mínimo resultante do *survey* foi utilizado durante a elaboração do novo processo proposto para a empresa, com a qual se realizou este trabalho. A estratégia para definição ou melhoria de processo de teste foi extraída da elaboração desse novo processo. A seguir são apresentadas as contribuições e limitações deste trabalho,

assim como algumas possibilidades de trabalhos futuros.

## 6.1 Contribuições

As principais contribuições deste trabalho são:

- O estabelecimento de uma estratégia de elaboração de processo simples, que permite a definição de um processo de teste que leva em conta as atividades já realizadas na empresa.
- O incentivo à utilização do TMMi – um modelo de referência que tem se tornado cada vez mais conhecido pela comunidade brasileira devido ao crescente interesse pela atividade de teste – como fonte de informação para construção do processo.
- A definição de um modelo de processo enxuto, construído com base nas práticas do TMMi a partir da opinião de profissionais de teste de software.
- Aplicação da estratégia proposta em empresas reais.

## 6.2 Limitações

Algumas das limitações deste trabalho são:

- A ferramenta KITTool não está disponível à comunidade, o que limita as possibilidades de realizar diagnósticos de processos existentes de forma similar ao que foi feito neste trabalho.
- O processo sugerido ainda não foi validado pela empresa.
- O *survey*, que deu origem ao modelo de processo reduzido, não obteve um número amplo de participantes e pode não retratar um consenso da comunidade de profissionais de teste;
- A estratégia foi extraída de um estudo de caso em uma empresa específica, e não foi reaplicada em outras empresas, de diferentes portes.

## 6.3 Trabalhos Futuros

Algumas atividades podem ser realizadas como forma de contribuir para a melhoria do trabalho desenvolvido. Propõe-se que as seguintes atividades sejam realizadas para dar continuidade ao trabalho:

- Validar o processo sugerido, resultante do estudo, junto à empresa;
- Realizar um novo diagnóstico com a ferramenta KITTool após o novo processo ter sido implantado na empresa, com o objetivo de avaliar a evolução da maturidade do processo de teste.
- Expandir o modelo de processo reduzido, identificando melhor os requisitos de cada prática, os produtos de trabalho esperados e as entradas e saídas do processo.
- Realizar novos estudos de casos a fim de validar a estratégia e observar possíveis falhas ainda não detectadas.





## REFERÊNCIAS BIBLIOGRÁFICAS

- ANDERSIN, J. TPI- A model for Test Process Improvement. In: *Seminar on Quality Models for Software Engineering*. [S.l.: s.n.], 2004.
- BASILI; CALDIERA, G.; ROMBACH, D. Goal/question/metric Paradigm. *Encyclopedia of Software Engineering*, v. 1, p. 528–532, 1994.
- BASILI, V.; REITER, R. A Controlled Experiment Quantitatively Comparing Software Development Approaches. *IEEE Transactions on Software Engineering*, SE-7, n. 3, p. 299–320, maio 1981. ISSN 0098-5589.
- BURNSTEIN, I. *Practical software testing: a process-oriented approach*. [S.l.]: Springer, 2003. ISBN 9780387951317.
- BUZAN, T.; BUZAN, B. *The Mind Map Book*. Reprint. [S.l.]: New American Library, 1996. ISBN 0452273226.
- CAO, P.; DONG, Z.; LIU, K. An optimal release policy for software testing process. In: *Proceedings of the 29th Chinese Control Conference, CCC'10*. [S.l.: s.n.], 2010. p. 6037–6042.
- CRESPO, A. N. et al. *Modelo de Processo Genérico de Teste de Software*. *Portal do software público brasileiro 5CQualiBr*. 2010. Online. <http://www.softwarepublico.gov.br/5cqualibr/xowiki/Teste-item13> - acessado em 27/03/2012.
- CUSUMANO, M. A. Factory Concepts and Practices in Software Development. *IEEE Ann. Hist. Comput.*, v. 13, n. 1, p. 3–32, jan. 1991. ISSN 1058-6180.
- DAVIS, G. Managing the test process [software testing]. In: *International Conference on Software Methods and Tools, 2000. SMT 2000. Proceedings*. [S.l.]: IEEE, 2000. p. 119–126. ISBN 0-7695-0903-7.
- GRAHAM, D. et al. *Foundations of Software Testing: ISTQB Certification*. Revised. [S.l.]: Thomson Learning, 2008. ISBN 1844809897.
- HASS, A. M. J. Testing Processes. In: *IEEE International Conference on Software Testing Verification and Validation Workshop, 2008. ICSTW '08*. [S.l.]: IEEE, 2008. p. 321–327. ISBN 978-0-7695-3388-9.
- HÖHN, E. N. *KITest: Um arcabouço de conhecimento e melhoria de processo de teste*. Tese (Doutorado) — Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, SP - Brazil, jun. 2011.

- HUMPHREY, W. S. *Managing the software process*. [S.l.]: Addison-Wesley, 1989. ISBN 9780201180954.
- IEEE. IEEE standard for software and system test documentation. *IEEE Std 829-2008*, p. 1–118, 2008.
- ISTQB, I. S. T. Q. B. *Standard glossary of terms used in Software Testing*. 2010. 1–51 p.
- JINO, M.; MALDONADO, J. C.; DELAMARO, M. E. *Introdução ao Teste de Software*. [S.l.]: Campus, 2007. ISBN 9788535226348.
- KANER, C.; FALK, J. L.; NGUYEN, H. Q. *Testing computer software*. [S.l.]: Wiley, 1999. ISBN 9780471358466.
- KARHU, K.; TAIPALE, O.; SMOLANDER, K. *Outsourcing and Knowledge Management in Software Testing*. 2007.
- KASURINEN, J. Elaborating Software Test Processes and Strategies. In: *2010 Third International Conference on Software Testing, Verification and Validation (ICST)*. [S.l.]: IEEE, 2010. p. 355–358. ISBN 978-1-4244-6435-7.
- KASURINEN, J. et al. Exploring perceived quality in software organizations. In: *Research Challenges in Information Science (RCIS), 2011 Fifth International Conference on*. [S.l.: s.n.], 2011. p. 1–12. ISSN 2151-1349.
- MALDONADO, J. C.; FABBRI, S. C. P. F. Verificação e validação de software. In: ROCHA, A.; MALDONADO, J.; WEBER, K. C. (Ed.). *Qualidade de Software: Teoria e Prática*. 1. ed. São Paulo: Prentice Hall, 2001. v. 1, p. 66–73.
- MILLER, J. Statistical significance testing—a panacea for software technology experiments? *Journal of Systems and Software*, v. 73, n. 2, p. 183–192, 2004.
- MPS.BR. *Melhoria de Processo do Software Brasileiro - Guia de Implementação - Parte 10: Implementação do MR-MPS em organização do tipo Fábrica de Teste*. [S.l.], 2011.
- MPS.BR. *Melhoria de Processo do Software Brasileiro - Guia Geral*. [S.l.], 2011.
- MPT.BR. *Melhoria do Processo de Teste Brasileiro*. [S.l.], 2011.
- MYERS, G. J. et al. *The art of software testing*. [S.l.]: John Wiley and Sons, 2004. ISBN 9780471469124.
- OCTAVIANO, F. R. et al. Test scenarios generation based on use cases. In: *Anais do CIBSE12 - Congresso Ibero-Americano em Engenharia de Software*. Buenos Aires: [s.n.], 2012. ISBN 978-987-1636-46-7.
- PURPER, C. B. Transcribing Process Model Standards into Meta-Processes. In: *Proceedings of the 7th European Workshop on Software Process Technology*. London, UK: Springer-Verlag, 2000. (EWSPT '00), p. 55–68. ISBN 3-540-67140-4.
- ROCHA, A. R. C. d.; MALDONADO, J. C.; WEBER, K. C. *Qualidade de software: teoria e prática*. [S.l.]: Prentice Hall, 2001. ISBN 9788587918543.

- RODRIGUES, A.; BESSA, A.; PINHEIRO, P. R. Barriers to implement test process in small-sized companies. In: LYTRAS, M. D. et al. (Ed.). *Organizational, Business, and Technological Aspects of the Knowledge Society*. [S.l.]: Springer Berlin Heidelberg, 2010, (Communications in Computer and Information Science, v. 112). p. 233–242. ISBN 978-3-642-16324-1.
- RODRIGUES, A.; PINHEIRO, P. R.; ALBUQUERQUE, A. The Definiton of a Testing Process to Small-Sized Companies: The Brazilian Scenario. In: *Quality of Information and Communications Technology (QUATIC), 2010 Seventh International Conference on the*. [S.l.]: IEEE, 2010. p. 298–303. ISBN 978-1-4244-8539-0.
- SANZ, A. et al. A proposal of a process model to create a test factory. In: *ICSE Workshop on Software Quality, 2009. WOSQ '09*. [S.l.]: IEEE, 2009. p. 65–70. ISBN 978-1-4244-3723-8.
- SEI. *Capability Maturity Model Integration Version 1.2. (CMMI-SE/SW, V1.2 – Continuous Representation)*. [S.l.], 2006.
- TAIPALE, O.; SMOLANDER, K.; KALVIAINEN, H. Cost reduction and quality improvement in software testing. In: . [S.l.: s.n.], 2006. p. 63–74.
- TEAM, T. L. project. Software, *LimeSurvey*. abr. 2011. [Http://www.limesurvey.org/](http://www.limesurvey.org/). Disponível em: <<http://www.limesurvey.org/>>.
- TIAN, J. *Software quality engineering: testing, quality assurance, and quantifiable improvement*. [S.l.]: John Wiley and Sons, 2005. ISBN 9780471713456.
- TMMi Foundation. *Test Maturity Model integration (TMMi )(Version 3.1)*. [S.l.], 2010. 1-181 p.
- WHITLEY, E.; BALL, J. Statistics review 6: Nonparametric methods. *Critical Care*, v. 6, n. 6, p. 509, set. 2002. ISSN 1364-8535.
- WOHLIN, C. *Experimentation in software engineering: an introduction*. [S.l.]: Springer, 2000. ISBN 9780792386827.



# GLOSSÁRIO

---

---

**CMMI** – *Capability Maturity Model Integration*

**KITMap** – *Knowledge and Improvement Map*

**KITTool** – *Knowledge and Improvement on Test Tool*

**MPS.Br** – *Melhoria do Processo de Software Brasileiro*

**MPT.Br** – *Melhoria do Processo de Teste de Software Brasileiro*

**PA** – *Process Area*

**PISO** – *Polo das Indústrias de Software*

**PRE** – *Processo Real da Empresa*

**PR** – *Processo Reduzido*

**SG** – *Specific Goal*

**SP** – *Specific Pratic*

**TMMi** – *Test Maturity Model integration*



# **Anexo A**

## **TABELA DE DEPENDÊNCIAS**

---

---

Este anexo apresenta a tabela de dependências entre práticas do TMMi identificadas por Höhn (2011).

Prática	Dep	Prática
Analisar riscos do produto	N	Definir a abordagem de teste
	N	Identificar e priorizar casos de teste
	N	Identificar e priorizar condicoes de teste
	N	Identificar elementos e características a serem testados
Definir a abordagem de teste	N	Definir criterios de entrada
	N	Definir criterios de parada
	N	Definir criterios de suspensao e recomeco
	N	Estabelecer uma wbs de alto nivel
Definir a abordagem de teste	N	Identificar e priorizar condicoes de teste
Definir categorias e parâmetros de risco do produto	N	Analisar riscos do produto
	N	Identificar riscos do produto não funcionais
Definir ciclo de vida de teste	N	Determinar estimativas de esforco e custo de teste
	N	Estabelecer o cronograma de teste
	N	Monitorar riscos do projeto de teste
	N	Planejar envolvimento dos interessados
Definir criterios de entrada	N	Verificar em relacao aos criterios de entrada
	A	Definir criterios de parada
Definir criterios de parada	N	Monitorar criterios de parada
Definir estrategia de teste	A	Definir a abordagem de teste
	A	Definir a abordagem de teste nao funcional
Desenvolver os requisitos do ambiente de teste	N	Conduzir revisoes em marcos do progresso
Determinar estimativas de esforco e custo de teste	N	Estabelecer o cronograma de teste
	N	Planejar a equipe de teste
Estabelecer o cronograma de teste	N	Monitorar riscos do projeto de teste
	N	Planejar a equipe de teste
Estabelecer o plano de teste	N	Conduzir revisoes em marcos do progresso
	N	Estabelecer o cronograma de teste
	N	Monitorar recursos do ambiente de teste
	N	Monitorar riscos do produto
	N	Planejar a equipe de teste
Estabelecer uma wbs de alto nivel	N	Definir ciclo de vida de teste
	N	Determinar estimativas de esforco e custo de teste
	N	Estabelecer o cronograma de teste
	N	Monitorar riscos do projeto de teste
	N	Planejar a equipe de teste
	N	Planejar envolvimento dos interessados
	A	Definir ciclo de vida de teste
Identificar características não funcionais a serem testadas	N	Definir a abordagem de teste nao funcional
Identificar elementos e características a serem testados	N	Definir a abordagem de teste
Identificar riscos ao projeto de teste	N	Conduzir revisoes em marcos do progresso
	N	Monitorar riscos do projeto de teste
Obter comprometimento com o plano de teste	N	Conduzir revisoes em marcos do progresso
	N	Monitorar compromissos de teste

**Tabela A.1: Dependências entre práticas do TMMi identificadas por Höhn (2011), sendo que N representa dependência *necessária* e A representa dependência de *alinhamento*.**