

**MAQSAAS - MÉTODO PARA AVALIAÇÃO DA
QUALIDADE EM PRODUTOS SAAS.**

NEMÉSIO FREITAS DUARTE FILHO

**MAQSAAS - MÉTODO PARA AVALIAÇÃO DA
QUALIDADE EM PRODUTOS SAAS.**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais. Departamento de Ciência da Computação como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

ORIENTADOR: CLARINDO ISAÍAS P. S. PÁDUA

CO-ORIENTADOR: ANDRÉ LUIZ ZAMBALDE

Belo Horizonte

03 de junho de 2011

© 2011, Nemésio Freitas Duarte Filho.
Todos os direitos reservados.

D812m Duarte Filho, Nemésio Freitas
MAQSaaS - Método para avaliação da qualidade em
produtos SaaS. / Nemésio Freitas Duarte Filho. —
Belo Horizonte, 2011
xxiv, 151 f. : il. ; 29cm

Dissertação (mestrado) — Universidade Federal de
Minas Gerais. Departamento de Ciência da
Computação

Orientador: Clarindo Isaías P. S. Pádua
Co-orientador: André Luiz Zambalde

1. Computação - Teses. 2. Engenharia de Software -
Teses. I. Título.

CDU 519.6*32 (043)



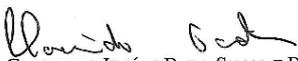
UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO

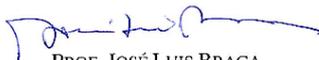
MAQSaaS - Método para avaliação da qualidade em produtos SaaS

NEMÉSIO FREITAS DUARTE FILHO

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:


PROF. CLARINDO ISAÍAS P. DA SILVA E PÁDUA - Orientador
Departamento de Ciência da Computação - UFMG


PROF. PAULO HENRIQUE DE SOUZA BERMEJO - Co-orientador
Departamento de Ciência da Computação - UFLA


PROF. JOSÉ LUIS BRAGA
Departamento de Informática - UFV


PROF. RODOLFO SÉRGIO FERREIRA DE RESENDE
Departamento de Ciência da Computação - UFMG

Belo Horizonte, 03 de junho de 2011.

Dedico esse trabalho a uma pessoa que foi de extrema importância na minha vida, Lucília Freitas Duarte, pessoa na qual sempre foi motivo de muito orgulho para mim. Mesmo estando longe permanentemente, sempre me espelharei em seus atos e seus ensinamentos, lembrando que a vida pode ser simples e ao mesmo tempo alegre.

Agradecimentos

Agradeço aos meus pais, Analice de Paula Mello Duarte e Nemésio Freitas Duarte, e toda minha família (irmão, primos, tios e avós) pelo apoio que me deram ao longo destes dois anos, me dando todas as oportunidades e incentivos para concluir este mestrado.

Agradeço também aos meus amigos que sempre estiveram ao meu lado me incentivando e ajudando nos momentos mais difíceis.

Aos professores:

- Clarindo Isaias Pereira da Silva (orientador) e André Luiz Zambalde (co-orientador) pela paciência e atenção que tiveram comigo ao longo da minha pesquisa, adicionando aspectos fundamentais para o início e finalização do trabalho;
- Paulo Henrique Bermejo pelos conselhos e incentivos dados ao longo do mestrado;
- e Marluce Rodrigues Pereira que desde a graduação me incentivou a seguir a área acadêmica.

Agradeço também ao pessoal do MINTER (Mestrado Interinstitucional UFLA-UFMG), grupo no qual pude passar por momentos únicos, de alegria, dificuldades e superação.

Agradecimentos finais aos amigos e colegas do Synergia-DCC/UFMG, que mesmo em poucos meses se mostraram companheiros e amigos.

Obrigado

*“Por que cometer erros antigos,
se há tantos erros novos a escolher?”*

(Denoth Speed)

Resumo

A utilização de software oferecido como serviço (SaaS) vem crescendo continuamente e tem atraído fornecedores de diferentes segmentos do mercado global de TI. Muitas organizações já estão substituindo seus produtos de software convencionais por produtos SaaS, porém sua utilização traz grandes problemas e desafios organizacionais, culturais e tecnológicos. Um dos problemas é a inexistência de métodos que auxiliem a avaliação da qualidade nestes produtos, atualmente os métodos existentes são muito genéricos, não contemplando especificações relevantes ao contexto dos produtos SaaS. O presente trabalho propõe e descreve um método para avaliar a qualidade em um produto SaaS, denominado MAQSaaS. O método possui um modelo de qualidade apropriado a este contexto, se baseando em normas de qualidade de software e modelos de gerenciamento de TI (ISO 20000, ITIL e COBIT). Seu processo de avaliação contempla algumas etapas da norma ISO 14598, possuindo modificações e ajustes em virtude do contexto SaaS. Os critérios e requisitos contidos no modelo de qualidade foram mapeados em atributos mensuráveis, possibilitando uma avaliação de forma clara e objetiva através da técnica de lista de verificação. Os resultados experimentais (validação e avaliação) mostram que o método pode agregar eficiência em avaliações práticas, conseguindo suprir ineficiências encontradas em métodos de avaliações atuais.

Palavras-chave: Qualidade de Produto de Software, Software oferecido como serviço (SaaS), Gestão de serviços de TI.

Abstract

The market for software products offered as a service (SaaS) is growing steadily and has attracted suppliers from different segments of the global IT market. Many organizations are replacing their conventional software products for SaaS products. However, the use of the SaaS products brings a range of challenges, both in the organizational, cultural and technological areas. A difficulty that exists today is the lack of methods and models for assessing the quality of these products. Currently existing models are very generic, not considering specifications relevant to the context of SaaS products. This document presents a method to assess the quality of a software product offered as a service, called MAQPSaaS. The proposed method has a quality model appropriate to the SaaS context, based on standards and models of software quality (ISO 9126) and models for IT management (ITIL and COBIT). Its evaluation process includes some parts of ISO 14598, and has some modifications and adjustments for the context of SaaS. The requirements proposed in this approach model were mapped on measurable attributes, allowing assessment by means of clear checklists. The experimental results obtained show that the method offers suitable assessment practices for software products offered as services.

Keywords: Software Product Quality, Software as a service (SaaS), IT Service Management.

Lista de Figuras

2.1	Guia de avaliação definido pela norma ISO/IEC 14598-5 [1997].	15
2.2	Arquitetura da Norma ISO/IEC 25000 [2005].	17
2.3	Visão detalhada do método MEDE-Pros [1996].	21
2.4	Modelo "Cauda Longa" (<i>The Long Tail</i>), Anderson [2004].	25
2.5	Modelo de maturidade para produtos SaaS, Chong & Carraro [2008].	26
2.6	Exemplo de arquitetura para produtos SaaS, Chong et al. [2009].	27
2.7	Principais atores presentes nos SaaS, Chong et al. [2009].	27
2.8	Modelos de governança de TI adotados por empresas, itSMF [2007].	29
2.9	Relacionamento de processos na gestão da ISO 20000.	30
2.10	Arquitetura da ITIL v3 [2008].	32
2.11	As principais áreas da governança de TI, ITGI [2007].	36
2.12	Arquitetura do <i>framework</i> COBIT 4.1, ISACA [2008].	37
3.1	Tipos de pesquisa segundo Jung [2004].	44
3.2	Primeira fase do processo metodológico.	46
3.3	Segunda fase do processo metodológico.	47
3.4	Terceira fase do processo metodológico.	49
3.5	Quarta fase do processo metodológico.	51
4.1	Relevância dos requisitos de qualidade relacionado ao paradigma SaaS.	59
4.2	Conjunto de critérios e requisitos de qualidade específicos ao SaaS.	60
4.3	Notação da prioridade dos requisitos.	62
5.1	Visão geral do MAQSaaS.	72
5.2	Etapas do processo de avaliação do MAQSaaS.	75
6.1	Tela principal do produto "Salesforce CRM Sales".	84
6.2	Critérios para julgamento da qualidade de um produto de software, Martinez [1999].	90

6.3	Perfil acadêmico dos Usuários Entrevistados.	92
6.4	Qualidade alcançada pelo produto "SalesForce CRM Sales".	94
6.5	Nota atribuída em relação a cada critério de qualidade.	95
6.6	Etapas para a realização do comparativo entre os métodos MAQSaaS e MEDE-Pros.	101
6.7	Dados obtidos pelo comparativo dos critérios de qualidade (MAQSaaS vs MEDE-Pros).	102
6.8	Resultado comparativo dos critérios de qualidade (MAQSaaS vs MEDE-Pros).102	
D.1	Qualidade final alcançada pelo produto "SalesForce CRM Sales".	143
D.2	Qualidade em relação a cada critério do produto "SalesForce CRM Sales".	144
E.1	Dados obtidos através da avaliação do critério funcionalidade.	149
E.2	Dados obtidos através da avaliação do critério usabilidade.	150
E.3	Dados obtidos através da avaliação do critério segurança.	150
E.4	Dados obtidos através da avaliação do critério desempenho.	150
E.5	Dados obtidos através da avaliação do critério apoio(suporte).	150
E.6	Dados obtidos através da avaliação do critério nível de serviço.	151
E.7	Dados obtidos através da avaliação do critério portabilidade.	151

Lista de Tabelas

2.1	Detalhes do requisito funcionalidade - ISO/IEC 9126 [1991].	11
2.2	Detalhes do requisito confiabilidade - ISO/IEC 9126 [1991].	11
2.3	Detalhes do requisito usabilidade - ISO/IEC 9126 [1991].	12
2.4	Detalhes do requisito eficiência - ISO/IEC 9126 [1991].	12
2.5	Detalhes do requisito manutenibilidade - ISO/IEC 9126 [1991].	13
2.6	Detalhes do requisito portabilidade - ISO/IEC 9126 [1991].	13
2.7	Divisão da norma ISO/IEC 14598 [1997].	14
4.1	Requisitos de funcionalidade - contexto SaaS.	55
4.2	Requisitos de confiabilidade - contexto SaaS.	55
4.3	Requisitos de usabilidade - contexto SaaS.	55
4.4	Requisitos de eficiência - contexto SaaS.	56
4.5	Requisitos de manutenibilidade - contexto SaaS.	56
4.6	Requisitos de portabilidade - contexto SaaS.	56
4.7	Requisitos de nível de serviço - contexto SaaS.	57
4.8	Requisitos de segurança - contexto SaaS.	57
4.9	Requisitos de desempenho - contexto SaaS.	57
4.10	Complementação das características de qualidade.	60
6.1	Objetivos e fundamentações das questões do <i>checklist</i>	88
6.2	Aspectos positivos e negativos em relação aos critérios avaliados.	98
C.1	Dados da aplicação da avaliação	125
C.2	Informações para preenchimento do <i>checklist</i>	126

Sumário

Agradecimentos	ix
Resumo	xiii
Abstract	xv
Lista de Figuras	xvii
Lista de Tabelas	xix
1 Introdução	1
1.1 Contextualização e Motivação	1
1.2 Problemática e Justificativa	3
1.3 Objetivos do Trabalho	3
1.4 Limites do Trabalho	4
1.5 Organização do Documento	4
2 Referencial Teórico	7
2.1 Engenharia e Qualidade do Produto de Software	7
2.2 Métodos, modelos e normas de avaliação da qualidade	9
2.2.1 ISO/IEC 9126	10
2.2.2 ISO/IEC 14598	14
2.2.3 ISO/IEC 25000 (SQuaRE)	16
2.2.4 Modelo GQM	18
2.2.5 Modelo Rocha	19
2.2.6 Método Mede-Pros	20
2.3 Software oferecido como serviço (<i>Software as Service - SaaS</i>)	21
2.3.1 Conceitos Gerais	22
2.3.2 Princípios e Arquitetura SaaS	24

2.4	Gestão de Serviços de TI	28
2.4.1	ISO/IEC 20000	29
2.4.2	ITIL v3	31
2.4.3	COBIT 4.1	35
2.5	Trabalhos Relacionados	38
3	Metodologia	43
3.1	Tipo de Pesquisa	43
3.2	Procedimento Metodológico	45
3.2.1	Concepção e Preparação da Pesquisa	46
3.2.2	Análise e Resultados Preliminares	47
3.2.3	Validação e Melhorias no Método	49
3.2.4	Verificações e Resultados Finais	50
4	Modelo de Qualidade para o MAQSaaS	53
4.1	Proposta dos critérios e requisitos de qualidade.	54
4.2	Relevância e impacto dos critérios de qualidade	67
4.3	Considerações Finais	69
5	Apresentação do MAQSaaS	71
5.1	Visão geral do MAQSaaS	71
5.2	Atores e papéis do processo de avaliação	73
5.3	Detalhamento das etapas do processo de avaliação	74
5.4	Considerações Finais	81
6	Validação e Avaliação do MAQSaaS	83
6.1	Estudo de caso - Produto Salesforce CRM	83
6.1.1	Apresentação do produto SaaS	83
6.1.2	Definir Objetivos e Contextualização da Avaliação	85
6.1.3	Definir e Configurar o Modelo de Qualidade	86
6.1.4	Identificar Métricas Mensuráveis	86
6.1.5	Definir Pontuação e Critérios para Julgamento	89
6.1.6	Projetar a Avaliação	91
6.1.7	Execução da Avaliação	93
6.1.8	Considerações e resultados das avaliações	94
6.2	Comparativo com o método MEDE-Pros	100
7	Conclusões	105

7.1	Discussões sobre o MAQSaaS	105
7.2	Contribuições do Trabalho	106
7.3	Trabalhos Futuros	107
7.4	Publicações Resultantes	108
	Referências Bibliográficas	109
	Apêndice A Convite para participação da entrevista	117
	Apêndice B Questionário Requisitos de Qualidade (SaaS)	119
	Apêndice C Lista de Verificação (<i>Checklist</i>)	125
	Apêndice D Relatório Final da Avaliação	143
	Apêndice E Dados obtidos durante as avaliações	149

Capítulo 1

Introdução

1.1 Contextualização e Motivação

Em meados dos anos 70, várias empresas começaram a ofertar um grande número de produtos de software. Muitas destas apresentavam dificuldades no desenvolvimento desses produtos e com um aumento da demanda, juntamente com o aumento da complexidade e dos problemas a serem resolvidos e da inexistência de técnicas estabelecidas para o desenvolvimento de softwares, que funcionassem adequadamente ou pudessem ser validados, as empresas enfrentaram a chamada "crise do software".

Essa crise causou custos e prejuízos em diversas organizações e só pôde ser parcialmente contornada com o surgimento da Engenharia de software, área de conhecimento da ciência da computação voltada para a especificação, desenvolvimento e manutenção de produtos de software. A Engenharia de Software aplica conhecimento voltado às tecnologias e práticas de ciência da computação, gerência de projetos e outras disciplinas, objetivando a garantia de uma maior produtividade e qualidade de seus produtos.

Dentro da Engenharia de software, encontramos diversas sub-áreas de conhecimento, como por exemplo, a área que trata da "qualidade de software". Essa objetiva garantir a conformidade com os requisitos do sistema, ou seja, que especificações explícitas e implícitas sejam atendidas por um produto de software, seja este produto oferecido sob a forma de pacote ou serviço (*Software as Service* - SaaS). O objetivo principal da qualidade de software é garantir que o produto final satisfaça a todas as expectativas do cliente, possibilitando uma maior confiabilidade, redução de erros e maior aceitação por parte das organizações, Kan [2002].

Atualmente as organizações estão cada vez mais utilizando produtos de software como ferramentas críticas aos seus negócios, fazendo com que boa parte do sucesso do negócio esteja ligado diretamente com a qualidade desses sistemas. Desta maneira

os requisitos de qualidade acabam se tornando critérios chaves para a aquisição desses produtos, refletindo diretamente na missão e nos objetivos de negócio das organizações, Ozkaya et al. [2008].

Neste contexto, a preocupação com a qualidade é um fator preponderante, uma vez que pode ser um diferencial para diversas empresas. Porém, a mensuração desta qualidade não é uma tarefa trivial, a garantia da qualidade esperada pelo usuário é uma atividade difícil de ser realizada e, muitas vezes, dependente da experiência da organização para definir as medidas e métricas de medição.

A qualidade de um produto software pode ser avaliada de diversas maneiras, a mais usual é utilizando um conjunto de requisitos funcionais (funcionalidade, segurança, entre outros), e de requisitos não funcionais (confiabilidade, eficiência, manutenção, portabilidade, entre outros), Lew et al. [2008]. Os requisitos funcionais e não funcionais são alguns exemplos de critérios usados para auxiliar a medição da qualidade em sistemas computacionais.

A dificuldade em se estabelecer e utilizar critérios de qualidade de software está ligada à complicação que temos atualmente em associar grandezas mensuráveis de produtos físicos a softwares comercializados, existindo muitas diferenças nas características de produtos manufaturados e produtos de software, Colombo & Guerra [2009]. Este cenário está proporcionando diversos trabalhos na área de "qualidade de software", com contribuições no sentido de estabelecer diferentes métodos de avaliação que auxiliem as organizações a mensurar e medir a tal qualidade.

Mesmo existindo diversos métodos de avaliação na literatura, muitos destes são ineficientes ou muito genéricos, o que nos impede de utilizá-los em produtos mais específicos, Barney & Wohlin [2009]. Um exemplo é a falta de métodos, modelos e métricas que proporcionem uma eficiente avaliação da qualidade em produtos de software oferecidos como serviço (*Software as a Service* - SaaS).

Esse novo paradigma, o SaaS, é baseado na entrega do software onde as empresas clientes pagam, não pela sua propriedade, mas pelo seu uso, ficando a cargo das empresas fornecedoras a manutenção, evolução e suporte técnico. Devido à concorrência do mercado, muitas empresas já estão aderindo a esse novo modelo, e estão tendo muitas dificuldades principalmente em assegurar a qualidade em seus produtos finais, considerando que ainda é um paradigma novo e incipiente.

Poucos trabalhos abordam definições e impactos de requisitos de qualidade relacionados a produtos SaaS, juntamente com um processo bem definido para avaliações práticas. Considerando tais dificuldades em se medir e garantir a qualidade em produtos SaaS e não conhecendo nenhum método ou metodologia fundamentado para tal questão, o objetivo do trabalho é propor um método eficiente para a avaliação da

qualidade em produtos SaaS, denominado MAQSaaS.

1.2 Problemática e Justificativa

O problema abordado neste trabalho é a falta de um método prático que avalie a qualidade em um produto SaaS. Um método de avaliação específico a este modelo é importante, pois muitas empresas estão aderindo a esse novo padrão de software, tendo dificuldades principalmente em assegurar a qualidade em seus produtos finais. Existe uma tendência de que muitas empresas desenvolvedoras de softwares irão migrar seus produtos finais para produtos SaaS, tendo como fator principal na concorrência a busca pela qualidade.

A mensuração e a garantia da qualidade de produtos SaaS em comparação aos softwares convencionais, se torna uma tarefa árdua e propícia a erros, considerando que não se conhece métodos, modelos e processos eficientes para tal questão. O método proposto contribuirá tanto para estudos, quanto para aperfeiçoamentos na área de qualidade voltado para este novo paradigma de software, o SaaS.

1.3 Objetivos do Trabalho

Este trabalho tem como objetivo geral propor um método para avaliar a qualidade de um produto SaaS. Para alcançar esta meta, alguns objetivos específicos serão considerados ao longo do trabalho, como por exemplo:

1. Estudar métodos e modelos de qualidade de produto de software, juntamente com métodos e modelos de gerenciamento de serviços de TI;
2. Integrar critérios e princípios de métodos de avaliação da qualidade de software, com critérios de métodos de gerenciamento de serviços de TI;
3. Levantar requisitos de qualidade específicos ao contexto SaaS, dividindo-os em grupos de características e sub-características, de maneira a mostrar seus impactos tanto positivos e negativos a este novo paradigma de software;
4. Propor e detalhar um modelo de qualidade com características específicas de um produto SaaS;
5. Pesquisa com especialistas da área, com o objetivo de priorizar e complementar os critérios e requisitos do modelo de qualidade;

6. Mapear requisitos de qualidade para atributos mensuráveis, possibilitando sua aplicação e avaliação através de lista de verificação;
7. Definir um processo de avaliação para o MAQSaaS;
8. Delimitar a abrangência e aplicabilidade do método proposto;
9. Avaliar e validar o método proposto através de um estudo de caso real e prático.

1.4 Limites do Trabalho

Muitos autores se preocupam somente em enumerar os objetivos do trabalho, esquecendo de esclarecer os seus limites. Esta seção visa delimitar o escopo que está sendo tratado no presente trabalho.

O método proposto tem uma aplicação genérica para produtos SaaS, ou seja, poderá ser aplicado a qualquer categoria desses sistemas, sendo ele educacional, científico, comercial, entre outros. Tal aplicação será verificada e fundamentada através de um estudo de caso apresentado ao final do trabalho. Deve ficar claro que a sua utilização em softwares convencionais pode acarretar resultados equivocados, pois o modelo de qualidade utilizado no MAQSaaS foi constituído contendo características específicas aos sistemas SaaS.

A medição da qualidade em um produto de software pode ser classificada como medição interna, externa ou em uso. O foco da medição em relação ao MAQSaaS será feita somente na qualidade externa e interna, deixando em segundo plano a qualidade em uso. O escopo foi delimitado desta maneira, pois as maiores diferenças dos softwares convencionais e dos SaaS se encontram na qualidade externa e interna de ambos. A qualidade em uso, por sua vez, não apresenta muitas diferenças em ambos os paradigmas, podendo ser tratada por métodos convencionais presentes na literatura.

O processo de desenvolvimento de um sistema computacional pode afetar diretamente a qualidade final de um produto de software. Muitos métodos de avaliação se preocupam com a qualidade do processo de desenvolvimento, porém, no contexto do atual trabalho a qualidade do processo de desenvolvimento de sistemas SaaS não será incorporada, devido ao escopo e prazo estipulado para o presente trabalho.

1.5 Organização do Documento

A dissertação encontra-se subdividida em 7 (sete) capítulos, descrito a seguir:

O capítulo 2 apresenta o referencial teórico e os trabalhos relacionados no qual este foi embasado, proporcionando ao leitor uma visão geral sobre os temas abordados no trabalho. O capítulo 3 aborda a metodologia empregada, evidenciando o tipo de pesquisa, definindo os aspectos teóricos e procedimentos metodológicos. O capítulo 4 traz o levantamento e o impacto dos requisitos de qualidade propostos para o método MAQSaaS. O capítulo 5 mostra a visão geral do método MAQSaaS, juntamente com o seu processo de avaliação, contendo atores, etapas e especificações. O capítulo 6 apresenta o estudo de caso utilizado neste trabalho, com o objetivo de validar e avaliar o método proposto. E por fim, o capítulo 7 traz a conclusão, juntamente com possíveis trabalhos futuros.

Capítulo 2

Referencial Teórico

Este capítulo apresenta os conceitos existentes na literatura sobre engenharia e qualidade do produto de software; os principais métodos, modelos e normas de avaliação da qualidade; uma contextualização sobre software oferecido como serviço (*Software as a Service - SaaS*); uma visão geral sobre gestão de serviços de TI; e por final a descrição dos principais trabalhos correlatos ao presente trabalho.

2.1 Engenharia e Qualidade do Produto de Software

Segundo Sommerville [2003], a Engenharia de Software é uma disciplina que se preocupa com os aspectos de produção dos softwares, desde os estágios iniciais de especificação do sistema até a sua manutenção.

A Engenharia de Software, semelhante a diversas outras disciplinas da Engenharia, tem como principal objetivo a busca pela melhora da qualidade de seus produtos, propondo modelos, métodos e técnicas que ajudam a garantir uma maior qualidade em seus produtos finais. Para Basili & Weiss [1984], qualidade em Engenharia de Software pode ser considerado como um conceito multidimensional, pois entre suas diversas características podemos incluir também o cliente/usuário do sistema, além de diversas outras características computacionais que afetam diretamente a competitividade desses sistemas.

Para o mercado, a qualidade tem se tornado uma peça fundamental para o sucesso das organizações, sendo um diferencial competitivo servindo para obtenção de selos e certificados internacionais de qualidade. Uma boa qualidade agrega valor nos produtos finais, favorecendo sua compra e venda dentro do mercado. Essa idéia não se restringe

somente aos produtos manufaturados, mas também aos produtos de software, Santos [1999].

Em relação ao software, a qualidade de seus produtos pode ser definida como um conjunto de características que devem ser alcançadas, para que o produto possa atender as necessidades implícitas e explícitas dos clientes, Rocha [1983]. Para tratar as necessidades dos clientes e aspectos de qualidade do produto de software, existem normas internacionais que auxiliam e identificam aspectos relevantes sobre a mensuração desta qualidade. Como exemplos, podemos citar as normas ISO/IEC 9126 [1991] e a ISO/IEC 14598 [1997], que respectivamente tratam da qualidade destes produtos trazendo diversos requisitos, servindo como um modelo de qualidade; e abordando uma metodologia própria de avaliação.

Uma das grandes dificuldades em se medir a qualidade em sistemas computacionais, se encontra na diferença que existe entre produtos de software e produtos manufaturados. De acordo com Capovilla [1989], são listadas algumas características inerentes a essência do software e diferenças relacionadas a um produto manufaturado.

- Complexidade: normalmente, um produto de software possui muitas linhas de código, muitas regras de negócios e diversos desenvolvedores envolvidos com diferentes idéias;
- Invisibilidade: o software é invisível para o usuário, o que se vê são as conseqüências da execução do sistema. Os próprios desenvolvedores necessitam utilizar modelos para representar o sistema em questão;
- Conformidade: o software é a interface entre diversas entidades do meio no qual será utilizado: equipamentos, outros produtos de software, usuários e cultura organizacional;
- Produção: para software não existe produção em série, pois dependendo do cliente o sistema necessita de aplicações específicas;
- Durabilidade: os softwares são duráveis, as falhas presentes nos sistemas computacionais acontecem devido a erros de implementação;
- Validade: o software não é sensível a problemas ambientais e nem sofre qualquer tipo de defeito devido ao efeito acumulativo de seu uso;
- Custo do software: cópias do software podem ser reproduzidas em segundos e distribuídas a vários clientes, com o custo unitário do projeto e do desenvolvimento.

Para Colombo & Guerra [2009] e Guerra et al. [2007] quando analisamos a qualidade de um produto físico, é mais fácil determinar padrões e métricas de qualidade do que em um sistema computacional, pois quando se trata de um software, existe uma dificuldade maior em se saber o que medir e o que comparar para chegar a uma definição do que é qualidade. Muito está sendo feito para a melhora da qualidade do produto de software, pois com seu grau de importância tanto na sociedade quanto para as organizações, sua garantia gera reduções no custo de manutenção e no desenvolvimento destes sistemas.

Partindo dessas dificuldades, de definição da qualidade em sistemas computacionais, muitas organizações aderem a normas, modelos e métodos, que auxiliam, apóiam e facilitam a avaliação da qualidade, como por exemplo, a norma ISO/IEC 9126, a ISO/IEC 14598, a ISO/IEC 25000, o modelo GQM, o Modelo Rocha, o Modelo MEDE-Pros, entre outros. As normas e os modelos citados acima serão abordados com detalhes na próxima seção, pois cada um apresenta aspectos positivos que irão auxiliar de alguma forma o desenvolvimento do método MAQSaaS.

Para que um método de avaliação de qualidade possa ser aplicado de maneira eficiente, é necessário utilizar um modelo da qualidade pretendida (com requisitos/sub-requisitos de qualidade), possibilitando determinar quais características devem ser medidas, Boehm [1981]. Em outras palavras, podemos afirmar que a maioria dos métodos de avaliação devem se apoiar em um conjunto de requisitos de qualidade bem definido, de maneira a proporcionar resultados eficientes em sua avaliação, Gatti & Werneck [2004].

Os produtos em domínios de aplicações específicas, como o SaaS, e as diferentes tecnologias utilizadas em seu desenvolvimento, requerem características próprias que irão determinar a sua qualidade, Herssens et al. [2008]. Cada uma das características pode ser detalhadas em vários níveis de sub-características, obtendo-se um conjunto de requisitos que descrevem a qualidade de um produto de software específico.

2.2 Métodos, modelos e normas de avaliação da qualidade

Esta sessão aborda as principais normas e modelos de qualidade para produtos de software. Muitos destes modelos têm eficiência comprovada em trabalhos científicos e em aplicações práticas.

As avaliações de qualidade em um sistema computacional não são tarefas triviais de se realizar, por tanto, é interessante fundamentar o estudo do trabalho com os

diversos métodos e normas de avaliação, principalmente para agregar pontos positivos e até mesmo verificar pontos limitantes de suas aplicações. O método MAQSaaS será proposto a partir da compilação da informação presente nos modelos, normas e métodos que são estudados nesta seção.

Para a revisão teórica do assunto "Métodos, modelos e normas de avaliação de qualidade", esta seção foi estruturada da seguinte forma:

1. Norma ISO/IEC 9126;
2. ISO/IEC 14598;
3. ISO/IEC 25000 (SQuaRE);
4. Paradigma GQM;
5. Modelo Rocha;
6. Método Mede-Pros.

2.2.1 ISO/IEC 9126

A norma ISO/IEC 9126 [1991] é uma das normas mais antigas na área de qualidade de software. Ela fornece seis características divididas em sub-características, tendo o objetivo de estabelecer um modelo de qualidade com os seguintes componentes:

1. Processo de desenvolvimento: cuja qualidade afeta a qualidade do produto de software gerado; e é influenciado pela natureza do produto desenvolvido;
2. Produto: compreendendo os requisitos de qualidade do produto de software. Estes requisitos de qualidade podem ser divididos entre atributos internos e externos;
3. Qualidade em uso: consiste na verificação da qualidade do software em cada contexto específico de uso pelo usuário. Esta é, também, a qualidade percebida pelo usuário.

As características da norma ISO/IEC 9126 podem ser utilizadas em qualquer categoria de software computacional, suas principais características juntamente com suas sub-características, podem ser vistas abaixo:

- **Funcionalidade:** requisito que mostra a existência de um grupo de funções e suas propriedades específicas.

Sub-característica	Definição
Adequação	mede o quanto o conjunto de funcionalidades é adequado às necessidades do usuário;
Acurácia	representa a capacidade do software de fornecer resultados precisos ou com a precisão dentro do que foi acordado;
Interoperabilidade	trata da maneira como o software interage com outro(s) sistema(s) especificados;
Segurança	mede a capacidade do sistema de proteger as informações do usuário e fornecê-las apenas às pessoas autorizadas.

Tabela 2.1: Detalhes do requisito funcionalidade - ISO/IEC 9126 [1991].

- **Confiabilidade:** requisito que mostra a capacidade com que o software tem em manter seu nível de desempenho sob diversas condições durante sua execução em um período de tempo.

Sub-característica	Definição
Maturidade	entendida como sendo a capacidade do software em evitar falhas decorrentes de defeitos no software;
Tolerância a Falhas	representando a capacidade do software em manter o funcionamento adequado mesmo quando ocorrem defeitos nele ou nas suas interfaces externas;
Recuperabilidade	foca na capacidade de um software em se recuperar após uma falha, restabelecendo seus níveis de desempenho e recuperando os seus dados.

Tabela 2.2: Detalhes do requisito confiabilidade - ISO/IEC 9126 [1991].

- **Usabilidade:** requisito que mostra a adequação do software ao contexto operacional onde será utilizado.

Sub-característica	Definição
Inteligibilidade	representa a facilidade com que o usuário pode compreender as suas funcionalidades e avaliar se o mesmo pode ser usado para satisfazer as suas necessidades específicas;
Apreensibilidade	identifica a facilidade de aprendizado do sistema para os seus potenciais usuários;
Operacionalidade	é como o produto facilita a sua operação por parte do usuário, incluindo a maneira como ele tolera erros de operação;
Atratividade	envolve características que possam atrair um potencial usuário para o sistema, o que pode incluir desde a adequação das informações prestadas para o usuário até os requintes visuais utilizados na sua interface gráfica.

Tabela 2.3: Detalhes do requisito usabilidade - ISO/IEC 9126 [1991].

- **Eficiência:** requisito que mostra o relacionamento entre o nível de desempenho do software e a quantidade de recursos usados.

Sub-característica	Definição
Relação ao Tempo	avalia se o tempo de resposta (ou de processamento) estão dentro das especificações;
Utilização de Recursos	mede tanto os recursos consumidos quanto a capacidade do sistema em utilizar os recursos disponíveis.

Tabela 2.4: Detalhes do requisito eficiência - ISO/IEC 9126 [1991].

- **Manutenibilidade:** requisito que mostra o trabalho necessário para fazer modificações e personalizações no software.

Sub-característica	Definição
Analisabilidade	identifica a facilidade em se diagnosticar eventuais problemas e identificar as causas das deficiências ou falhas;
Modificabilidade	caracteriza a facilidade com que o comportamento do software pode ser modificado;
Estabilidade	avalia a capacidade do software de evitar efeitos colaterais decorrentes de modificações introduzidas;
Testabilidade	representa a capacidade de se testar o sistema modificado, tanto quanto as novas funcionalidades quanto as não afetadas diretamente pela modificação;

Tabela 2.5: Detalhes do requisito manutenibilidade - ISO/IEC 9126 [1991].

- **Portabilidade:** requisito que mostra a capacidade do software de ser transferido de um ambiente para outro.

Sub-característica	Definição
Adaptabilidade	representa a capacidade do software de se adaptar a diferentes ambientes sem a necessidade de ações adicionais;
Capacidade Instalação	identifica a facilidade com que pode se instalar o sistema em um novo ambiente;
Coexistência	mede o quão facilmente um software convive com outros instalados no mesmo ambiente;
Capacidade Substituição	representa a capacidade que o sistema tem de substituir outro sistema especificado, em um contexto de uso e ambiente específicos.

Tabela 2.6: Detalhes do requisito portabilidade - ISO/IEC 9126 [1991].

A norma ISO 9126 será de grande importância para o presente trabalho, sendo considerada como um modelo de qualidade amplamente utilizado em diversos outros métodos e em diferentes categorias de software. A norma é usada neste trabalho, como um referencial inicial sobre "o que medir" visando à qualidade interna e externa de um software convencional, possibilitando comparações com características de um produto SaaS.

2.2.2 ISO/IEC 14598

A ISO/IEC 14598 [1997] é um guia para avaliação da qualidade em produtos de softwares. Sua principal finalidade é a definição de um processo de avaliação baseado na utilização da norma ISO 9126. O processo de avaliação proposto pode ser utilizado tanto para avaliar produtos já existentes ou em desenvolvimento, independentemente do domínio ao qual pertençam.

A norma ISO 14598 é constituída basicamente em seis documentos distintos, relacionados entre si, ISO/IEC 14598 [1997].

Norma	Nome	Finalidade
14598-1	Visão Geral	Ensina a utilizar os outros documentos do grupo;
14598-2	Planejamento e Gerenciamento	Sobre como fazer uma avaliação, de forma geral;
14598-3	Guia para Desenvolvedores	Como avaliar sob o ponto de vista de quem desenvolve;
14598-4	Guia para Aquisição	Como avaliar sob o ponto de vista de quem vai adquirir;
14598-5	Guia para Avaliação	Como avaliar sob o ponto de vista de quem certifica;
14598-6	Módulos de Avaliação	Detalhes sobre como avaliar cada característica.

Tabela 2.7: Divisão da norma ISO/IEC 14598 [1997].

Um dos aspectos mais importante da norma são as etapas de avaliação definidas na norma ISO/IEC 14598-5 [1997], exemplificado na Figura 2.1. A figura mostra as

principais atividades que devem ser executadas em uma avaliação.

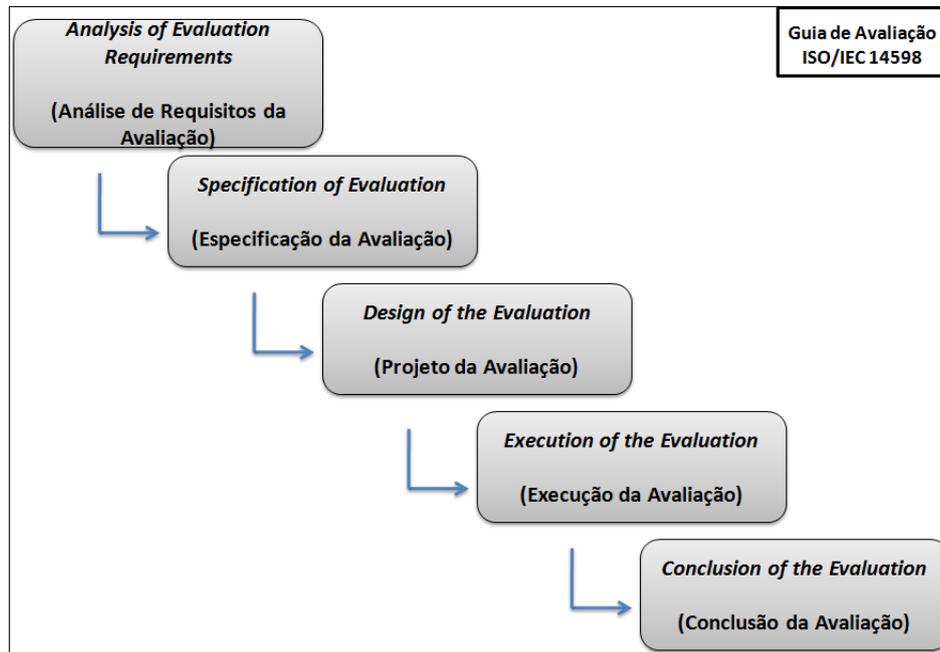


Figura 2.1. Guia de avaliação definido pela norma ISO/IEC 14598-5 [1997].

- *Analysis of Evaluation Requirements* (Análise de Requisitos da Avaliação): atividade em que são descritos os objetivos da avaliação, definindo-se os requisitos a serem avaliados. Vários pontos de vista podem ser considerados, dependendo dos diferentes usuários do produto;
- *Specification of the Evaluation* (Especificação da Avaliação): tem por objetivo a definição do escopo da avaliação e as medições a serem executadas no produto. São definidas também as restrições, os métodos a serem utilizados e as responsabilidades de todos os envolvidos no processo de avaliação;
- *Design of the Evaluation* (Projeto da Avaliação): são documentados os procedimentos a serem usados pelo avaliador para executar as medições especificadas na fase anterior. O avaliador deve produzir um plano que descreva os recursos necessários para realizar a avaliação especificada, a distribuição desses recursos nas várias ações a serem executadas bem como os prazos, a equipe de avaliação, os riscos associados e todas as atividades envolvidas;
- *Execution of the Evaluation* (Execução da Avaliação): são obtidos resultados da execução de ações para medir e verificar o produto de software de acordo com os

requisitos, com a especificação e com o projeto da avaliação. Ao final dessa etapa obtém-se o rascunho do relatório e dos registros da avaliação;

- *Conclusion of the Evaluation* (Conclusão da Avaliação): deve-se revisar o relatório da avaliação e disponibilizar os dados resultantes.

De modo geral, esta norma complementa a ISO/IEC 9126 e permite uma avaliação padronizada das características de qualidade de um software. É importante notar que, ao contrário da ISO 9126, a ISO 14598 vai a detalhes mínimos, incluindo modelos para relatórios de avaliação, técnicas para medição das características, documentos necessários para avaliação e suas fases, Azuma [2001].

O estudo da norma ISO 14598 é relevante ao presente trabalho principalmente no aspecto de definição de um processo de avaliação que seja repetível, reproduzível, imparcial e objetivo. Um método de avaliação além de possuir um modelo de qualidade bem definido, também deve ter um processo de avaliação que seja prático e não burocrático, facilitando sua utilização por partes de empresas e organizações, Colombo & Guerra [2009] e Guerra et al. [2007].

2.2.3 ISO/IEC 25000 (SQuaRE)

A norma ISO/IEC 25000 [2005] pode ser considerada como uma das normas mais relevantes no sentido de medir e avaliar a qualidade em produtos de software. Esta norma é uma evolução das normas ISO/IEC 9126 e da ISO/IEC 14598, onde também é conhecida como SQuaRE (*Software Quality Requirements and Evaluation*).

Para suprir uma falta de clareza e uma difícil compreensão pelos usuários em relação à norma ISO/IEC 9126 e a ISO/IEC 14598, em 1998, o grupo de trabalho WG6 do Subcomitê de Sistemas e Software (SC7) da ISO/IEC, responsável pela elaboração de normas internacionais que tratam da especificação, medição e avaliação da qualidade de produtos de software, propôs a criação de um guia reformulado que objetivava melhorar a compreensão dos usuários.

De maneira geral a norma SQuaRE vem a ser uma norma mais organizada, unificada, com abrangência de dois processos principais: a especificação de requisitos e, a avaliação da qualidade de software. Ela pode ser utilizada tanto para auxiliar desenvolvedores, quanto gerentes de produtos de software durante os processos de especificação de requisitos e avaliação da qualidade.

Ao final de maio de 2002, a norma juntamente com a sua numeração completa, foi concluída e aprovada podendo assim ser aplicada. A Figura 2.2 nos mostra a

arquitetura exemplificada da norma ISO/IEC 25000. Vemos que a série ISO/IEC 25000 é composta pelos seguintes documentos em sua arquitetura e divisão:

<u>Quality Requirements Division</u> 2503n	<u>Quality Model Division</u> 2501n	<u>Quality Evaluation Division</u> 2504n
	<u>Quality Management Division</u> 2500n	
	<u>Quality Metrics Division</u> 2502n	

Figura 2.2. Arquitetura da Norma ISO/IEC 25000 [2005].

- *Quality Management* (Gerenciamento de Qualidade): os documentos desta divisão da norma são voltados a todos os possíveis usuários, como gerentes, programadores, entre outros. São definidos os termos utilizados em todos os demais documentos e são feitas algumas recomendações;
- *Quality Model* (Modelo de Qualidade): corresponde à antiga ISO/IEC 9126, onde são definidos os conceitos de qualidade e permitem orientar diferentes perspectivas de avaliações;
- *Quality Metrics* (Métricas de Qualidade): define o que é medição e descreve diversos aspectos relacionados à realização desta atividade;
- *Quality Requirements* (Requisitos de Qualidade): corresponde aos princípios da norma 9126 no que se refere a estabelecer objetivos de qualidade para um produto, ou seja, não basta utilizar apenas medidas para avaliar a qualidade de um produto, mas alguns valores alvos como a especificação de requisitos do software;
- *Quality Evaluation* (Avaliação da Qualidade): os documentos desta divisão são voltados diretamente para as avaliações. São definidos os procedimentos, metodologia e demais recomendações sobre as melhores práticas em relação às avaliações práticas, objetivando a media da qualidade pretendida.

Seu estudo é de extrema importância, pois mostra de certa forma a evolução das normas de qualidade visando uma maior objetividade, melhorando e facilitando sua aplicação por parte das organizações.

Sua fundamentação no presente trabalho, também auxilia na incorporação de características contidas nas divisões da norma SQuaRe. Essas divisões, visualizadas na Figura 2.2, são compostas de normas, harmonicamente integradas, que detalham os tópicos relacionados à especificação e avaliação da qualidade de produtos de software. As idéias presentes nas divisões ISO/IEC 2501n - Divisão Modelo de Qualidade; ISO/IEC 2502n - Divisão Medição da Qualidade; e ISO/IEC 2504n - Divisão Avaliação da Qualidade, serão amplamente utilizadas no método MAQSaaS, pois, contemplam características importantes para um modelo de qualidade, medição de atributos de qualidade e utilização de um processo ao longo das avaliações.

2.2.4 Modelo GQM

De acordo com Basili & Musa [1991], o modelo GQM (*Goal, Question, Metric*), é uma linha de pesquisa que vem a acrescentar benefícios no gerenciamento de qualidade em grandes organizações. De acordo com os autores acima, GQM propõe um planejamento estratégico com interação de produtos e processos, para que se obtenha alta aceitação do cliente e baixa disfuncionalidade organizacional através de mercados em diferentes países.

Para Basili & Musa [1991], a utilização do GQM possibilita uma maior organização e planejamento do trabalho durante as medições de indicadores. Para os autores Basili & Weiss [1984], "Especificar objetivos é por si próprio um tópico importante, pois, sem objetivos, há riscos de coleta de dados sem importância e relevância ao contexto desejado".

O GQM como uma estrutura para o desenvolvimento de planos de métricas, ele segue alguns passos como: definição, planejamento, construção, análise e retorno dos resultados. O GQM também considera três etapas no que se diz respeito a questões de melhoramento:

- **Construir um conjunto de objetivos:** a idéia central desta etapa é determinar na organização o que se quer melhorar, definindo alvos e valores que se querem alcançar;
- **Construir um conjunto de questões:** as questões quantificam os objetivos, sendo relacionadas a produtos ou processos, fornecendo um *feedback* da perspectiva de qualidade;

- **Construir um conjunto de métricas:** essas métricas fornecem as informações necessárias para responder cada questão. As métricas podem ser objetivas e subjetivas, isto é, apresentam um valor que indique se o produto é de alta ou baixa qualidade. Uma vez definidos os objetivos, derivadas as questões, e desenvolvidas as métricas, são criadas matrizes para relacionar alvos/questões/métricas.

Seu estudo é importante no presente trabalho, pois mostra como técnicas auxiliares podem ajudar a melhorar a especificação de métricas para a avaliação da qualidade. Sendo o SaaS, um paradigma específico de software, é importante avaliar corretamente as métricas utilizadas, pois serão dessas métricas que obteremos informações relevantes à qualidade final do produto avaliado.

2.2.5 Modelo Rocha

O modelo criado por Rocha [1983], tem como objetivo a avaliação da qualidade de produtos de softwares. Seu modelo, mesmo sendo antigo, ainda é estudado e aplicado em diferentes métodos de avaliação da qualidade em produtos de software, tal modelo define a qualidade a partir de certos conceitos, estudados por Belchior [1997], descritos abaixo:

- Objetivo da qualidade: propriedades que devem estar presentes no produto;
- Fatores de qualidade: definição da qualidade em diferentes usuários que estejam em contato com o produto;
- Critérios: são considerados atributos que possam ser medidos e mensurados;
- Processo de avaliação: determinam o processo e a metodologia utilizados para medir critérios de qualidade do produto;
- Medidas: é o resultado final da avaliação da qualidade do produto, de acordo com os critérios seguidos.

Belchior [1997], aplicando conhecimentos do modelo Rocha em seu trabalho, afirma que os objetivos de qualidade em uma avaliação, são atingidos mediante fatores e subfatores de qualidade. Os fatores de qualidade não são tão triviais de serem mensuráveis, pois em muitas vezes são representados de forma abstratos. Já os subfatores por representarem um detalhamento maior aos fatores de qualidade, são mais utilizados nas avaliações, permitindo resultados mais significativos ao contexto das avaliações. Clunie [1997], também mostra em seu trabalho que o modelo Rocha pode

ser utilizado para definir qualidade em produtos de software através da utilização de fatores e sub-fatores.

O Modelo Rocha também pode ser utilizado para definir vários domínios de aplicação de qualidade. Juntamente com seu processo de avaliação, o modelo permite definir vários domínios de qualidade, como por exemplo, para um software científico, um software financeiro, software educacional, sistemas especialistas, entre outros.

Os estudos efetuados sobre o modelo Rocha mostraram que mesmo sendo um modelo mais antigo de avaliação, ele é completo, apresentando desde um modelo de qualidade (constituído por diversos fatores e subfatores) até um processo bem definido para suas avaliações. Muitos métodos e modelos atuais se baseiam em diversas de suas características, principalmente no seu domínio de aplicação, considerando que é um modelo genérico, possibilitando sua aplicação em diversas categorias de software.

Um dos objetivos do presente trabalho é garantir que o método MAQSaaS seja genérico para sistemas SaaS. Para tal objetivo o modelo Rocha será fundamental, pois ajudará a identificar características que permitam que o método proposto possa ser utilizado em diversas categorias de software, mantendo sua efetividade e garantindo resultados satisfatório nas avaliações de qualidade.

2.2.6 Método Mede-Pros

O MEDE-Pros [1996], pode ser considerado um método de avaliação da qualidade de software, permitindo efetuar a avaliação da qualidade baseado na ISO/IEC 9126. Basicamente esse método é composto por uma lista de verificação, manual do avaliador e um modelo de relatório de avaliação.

De acordo com Colombo & Guerra [2002], em seu livro, cada parte do modelo pode ser explicado da seguinte forma:

- Lista de Verificação: descreve questões de acordo com o objetivo principal da avaliação, sendo que cada questão contempla características que devem estar presentes no produto;
- Manual do Avaliador: serve para orientar o processo de avaliação, esclarecendo o que deve ser verificado em cada uma das etapas do processo;
- Modelo do Relatório: mostra os pontos positivos e negativos do produto para cada um dos componentes do software, apresentando sugestões para sua melhoria;

A Figura 2.3, nos mostra uma visão detalhada da estrutura desse método. Podemos ver as sub-características de qualidade da norma ISO/IEC 9126, que é o desdobramento de cada uma das características, mais os requisitos do conceito completitude.

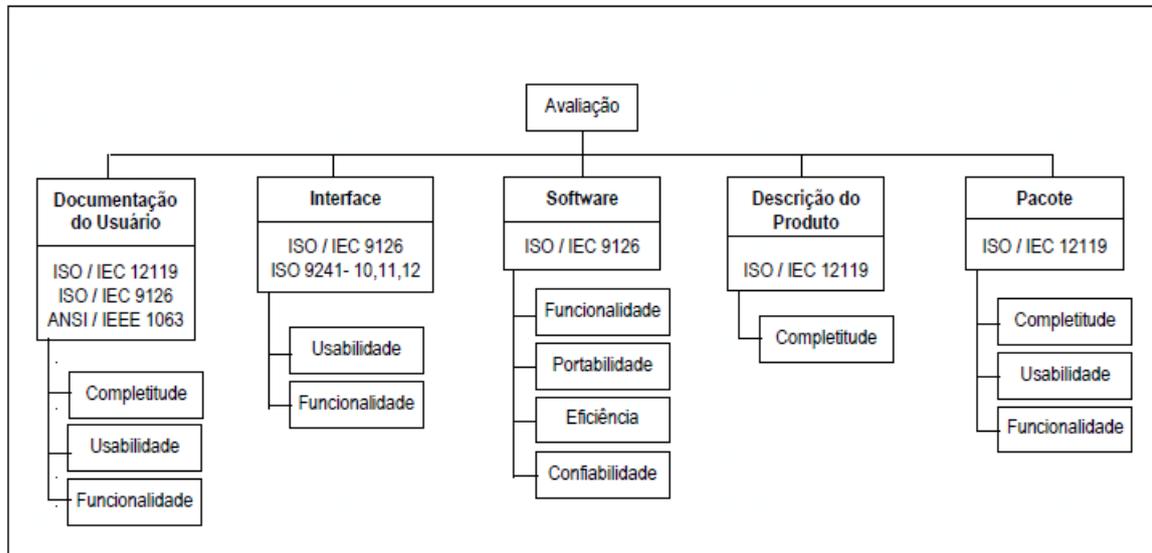


Figura 2.3. Visão detalhada do método MEDE-Pros [1996].

As avaliações utilizando esse método são cobradas, precisando do auxílio de uma pessoa especialista na área, um conjunto de dados para efetuar a avaliação, e não é diretamente focada para um cliente que irá adquirir um software, mas sim para a empresa de software que queira avaliar a qualidade do seu produto.

O método MEDE-Pros agrega conhecimentos relevantes ao presente trabalho, pois além de ser um método completo mais atual, sendo utilizado em diversas organizações, ele apresenta algumas etapas adicionais como a utilização de técnicas auxiliares que vão de alguma forma complementar e auxiliar as avaliações, como por exemplo, a utilização de listas de verificação, manuais para a avaliação, modelo de relatório final, entre outros. Todas estas técnicas e artefatos têm por objetivo deixar a avaliação mais clara e eficiente.

2.3 Software oferecido como serviço (*Software as Service - SaaS*)

Esta seção tem por objetivo mostrar ao leitor os principais conceitos, princípios e arquitetura deste novo modelo de software, o *Software as a Service* (SaaS). Por ser uma proposta nova, sendo foco de diversas pesquisas, muitos conceitos e princípios se tornam

conflitantes entre as visões de diversos autores e pesquisadores. No presente trabalho seguiremos a linha de pensamento dos autores mais conceituados nesta área até o momento, Ma [2007]; Goth [2008]; Laplante et al. [2008]; Herbert [2008]; Choudhary [2007]; Anderson [2004] e Chong & Carraro [2008].

Para a revisão teórica do assunto "Software oferecido como serviço", esta seção foi estruturada da seguinte forma:

1. Apresentação de Conceitos Gerais;
2. Definições de Princípios e Arquitetura SaaS.

2.3.1 Conceitos Gerais

A Internet tem evoluído muito nos últimos anos e, impulsionada pela disseminação do conceito da WEB 2.0, têm surgido uma série de novas oportunidades que utilizam meios como as redes de relacionamento, aplicações ricas em interatividade e, também, a disponibilização de software como serviço (*Software as a Service*), ampliando os horizontes dos negócios, Turner et al. [2003].

O SaaS é um modelo onde as empresas e usuários deixam de comprar licenças e passam a ser assinantes, adquirindo o direito de uso do software. Esses softwares são acessados de forma remota, pela internet, sendo desnecessária a instalação do software na máquina local do usuário. Normalmente é vista como uma prestação de serviço, daí o nome de software como serviço.

Esse novo paradigma está transformando o modo pelo qual as organizações que possuem departamentos de tecnologia da informação (TI) relacionam-se entre si e, até mesmo, o que pensam sobre o seu papel como provedores de serviços para o restante da empresa. Muitos dizem que o SaaS veio para revolucionar e acabar com a maioria dos problemas encontrado nas empresas de software, porém um estudo feito por Ma [2007], mostra que muitos não sabem ao certo como definir essa nova tecnologia e muito menos sabem como implementá-la.

Ma [2007] define SaaS (ou software como serviço) como "uma arquitetura que começa a aparecer como um novo modelo de desenvolvimento de software para as empresas que querem agregar valores aos seus serviços". Em resumo SaaS nada mais é que um software implementado como serviço, hospedado e acessado pela internet.

A idéia central do *software as a service* é fornecer um sistema na forma de serviço, com isso temos que seu modelo de entrega separa a governança do software e do usuário, pois o dono do serviço é quem esta hospedando o software e permite que clientes e usuários utilizem esse serviço em suas necessidades, Laplante et al. [2008].

É inevitável que esse novo modelo afete as grandes organizações de software, pois de acordo com Goth [2008], os SaaS irão mudar não somente como se implementa e se desenvolve um software, mas também como as pessoas irão vender, comprar e utilizá-los.

O software como serviço apresenta inúmeros benefícios em comparação com os softwares convencionais, porém por ser uma idéia nova e em constante desenvolvimento, apresenta alguns problemas e precauções dos quais deve-se ter cuidado, como por exemplo, dificuldades em se garantir e mensurar sua qualidade. A falta de garantia da qualidade nesses sistemas, pode proporcionar falhas de acesso, problemas relacionados com disponibilidade, falta de portabilidade, confiabilidade, integridade dos dados, entre outros.

De acordo com Herbert [2008], podemos citar alguns dos principais benefícios dos SaaS:

- **A redução de custo:** pois não é mais necessária a compra de hardware, uma vez que o software está na internet. Também é dispensada a aquisição de licença. Além disso, a contratação do serviço pode ser abatida no cálculo do imposto de renda sobre o lucro líquido, pois é contabilizada como despesa, enquanto a aquisição é considerada um ativo imobilizado;
- **Agilidade:** como o software está pronto para o uso no servidor do fornecedor, o processo de implantação normalmente é mais rápido. Pelo mesmo motivo, o suporte técnico é facilitado, não sendo necessário, por exemplo, deslocamento de equipe técnica;
- **Acessibilidade:** como o software está na internet, ele pode ser acessado pelos assinantes de qualquer lugar do mundo e a qualquer momento, permitindo mais integração entre unidades de uma mesma empresa;
- **Flexibilidade:** diferentemente do licenciamento, a quantidade de assinantes de um software como serviço pode aumentar e diminuir de acordo com a necessidade do contratante. Isso permite flexibilidade e adequação do custo da empresa a sua realidade;
- **Continuidade:** no software como serviço, a evolução (atualização) dos sistemas não precisa ser mais adquirida. A tendência é que os novos recursos e atualizações de versões sejam incorporados automaticamente e simultaneamente aos produtos.

Os SaaS podem ser classificados basicamente em duas categorias, sendo as mais usuais no mercado, "Serviços de linha de negócio" e "Serviços orientados a clientes".

Essas categorias foram detalhadas em estudos realizados por Choudhary [2007], segue abaixo a descrição de ambas:

1. Serviços de linha de negócios: oferecidos a empresas e organizações de pequeno, médio e grande porte. Os serviços de linha de negócios geralmente são soluções de negócios grandes e personalizadas direcionadas para facilitar processos de negócios como finanças, gerenciamento da cadeia de suprimentos e relações com o cliente;
2. Serviços orientados a clientes: oferecidos ao público em geral. Os serviços orientados a clientes às vezes são comercializados como assinatura, mas em geral são fornecidos sem custos financiados por anúncios.

De maneira geral o SaaS é um novo paradigma de software que tenta proporcionar maiores benefícios as organizações contratantes e fornecedores de software. Porém, por ser um paradigma ainda imaturo, muitos cuidados devem ser tomados, principalmente com a qualidade, pois a qualidade desses sistemas pode, impactar diretamente na missão e nos objetivos de negócios das organizações contratantes.

2.3.2 Princípios e Arquitetura SaaS

O *software as a service* traz também um conjunto de novos conceitos e uma nova arquitetura em relação aos softwares convencionais. Um desses conceitos é o termo *tenant*, que pode ser utilizado para referenciar o inquilino (cliente/organização) que acessa o software pela web.

Outro conceito muito utilizado é o *multi-tenant*, ou multi-inquilino (diversos clientes), no qual se refere ao uso da mesma aplicação/software por vários clientes e empresas ao mesmo tempo. Estes dois conceitos acabam proporcionando um cenário denominado de "Cauda Longa", onde é possível a disponibilização dos mesmos recursos para um conjunto muito maior de clientes, Figura 2.4, Anderson [2004].

Para explicar esse cenário de "Cauda Longa" proposto por Anderson [2004], vemos na Figura 2.4 um gráfico que demonstra que ao abaixar o custo de adoção do software, um maior número de clientes tendem a adotar o software como solução. Um dos principais princípios do SaaS é o fato de se dedicar a soluções de baixo custo, tendo um alto aproveitamento de recursos por um número muito grande de clientes, para atingir um público não suportado hoje em dia, devido aos custos proibitivos de entrada.

Em virtude de se aumentar a demanda de diversos clientes sobre um software como serviço, cria-se outro conceito conhecido como "micro-pagamento", no cenário

de "Cauda Longa" um número muito grande de clientes pode pagar pela utilização do software como serviço pagando por demanda, ou seja, pagar pela utilização somente quando houver a necessidade por parte do cliente.

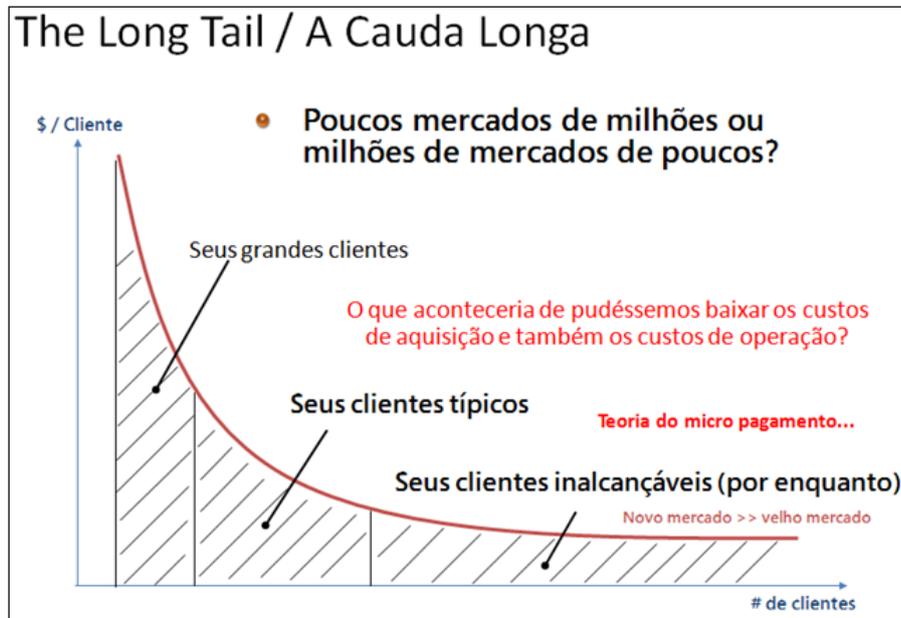


Figura 2.4. Modelo "Cauda Longa" (*The Long Tail*), Anderson [2004].

Isso faz com que a aquisição se torne mais fácil, pois como mostra a Figura 2.4, a busca é por "milhões de mercados de poucos", ao invés dos atuais "poucos mercados de milhões".

Para que esse conjunto de novos conceitos seja eficiente e traga benefícios aos softwares como serviço, é necessário a utilização de diversas tecnologias e infra-estrutura para atender este modelo. Chong & Carraro [2008] apresentam em seu trabalho um modelo de maturidade para o SaaS, Figura 2.5. Segue abaixo uma pequena explicação de cada quadrante do modelo de maturidade proposto por Chong & Carraro [2008].

- Quadrante 1: o software possui uma instância dedicada para cada *tenant*. Isso garante um completo atendimento das demandas do cliente, mas com elevado custo devido ausência de compartilhamento de recursos e customização elevada;
- Quadrante 2: o software apresenta uma instância dedicada para cada *tenant*, porém, já é possível observar que a solução é a mesma, com nenhuma personalização presente. Isso garante um custo menor de manutenção, já que a mesma solução atende a diversos clientes;

- Quadrante 3: o software já pode ser considerada como *multi-tenant* e apresenta total compartilhamento de recursos, havendo uma única instância para todos os clientes;
- Quadrante 4: o software permite um atendimento diferenciado para inquilinos que exigem elevada demanda de recursos, havendo uma carga balanceada na infra-estrutura do provedor.

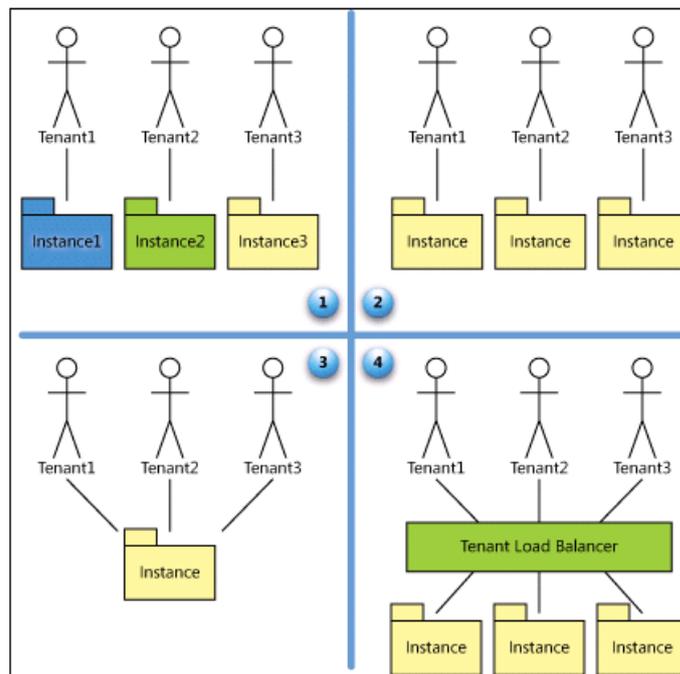


Figura 2.5. Modelo de maturidade para produtos SaaS, Chong & Carraro [2008].

Em outro trabalho realizado por Chong et al. [2009], é mostrado a importância que a arquitetura SaaS tem para que todos esses conceitos sejam realizados de forma eficiente, a partir do modelo de maturidade uma arquitetura SaaS de alto nível foi elaborada, Figura 2.6.

Em uma visão geral, podemos perceber uma arquitetura bem diferente onde temos diversas tecnologias se interagindo para formarem um novo conceito de software. É errado dizer que o SaaS é uma tecnologia nova, o mais correto é falar que o software como serviço é um conjunto de tecnologias aplicadas a um determinado fim.

Na Figura 2.6, pode-se perceber o inter-relacionamento que esses conceitos têm em relação aos sistemas SaaS, proporcionando um conjunto de tecnologias, com o objetivo de constituir um novo paradigma de software.

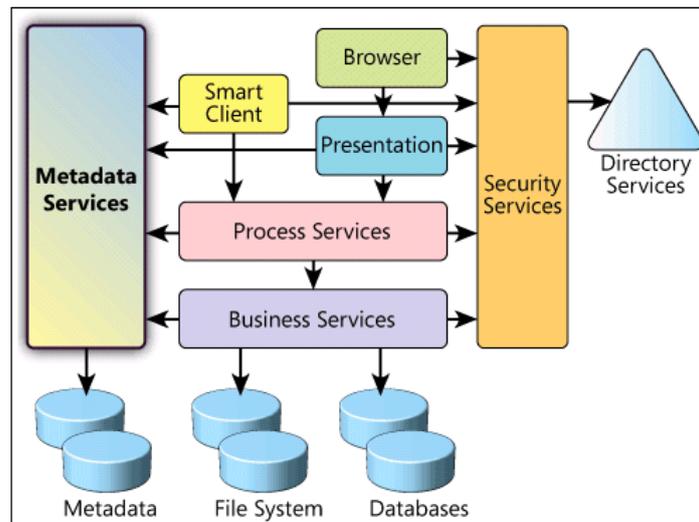


Figura 2.6. Exemplo de arquitetura para produtos SaaS, Chong et al. [2009].

Chong et al. [2009] também mostra os principais atores que estão relacionados aos SaaS, podemos ver na Figura 2.7 que cada ator possui perspectivas diferentes sobre os benefícios do SaaS.

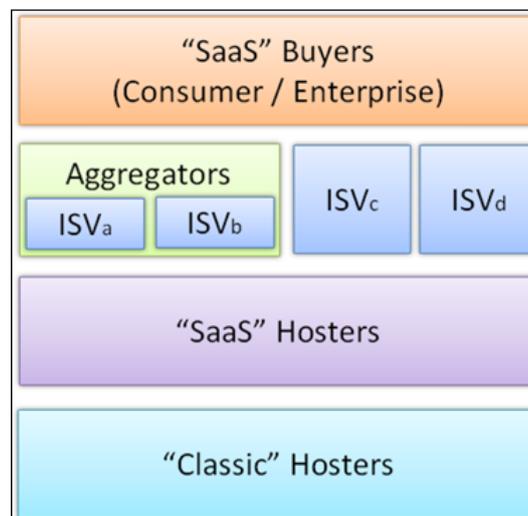


Figura 2.7. Principais atores presentes nos SaaS, Chong et al. [2009].

- Consumidores: maior controle, permitindo o teste da solução antes da compra, pagamento pelo uso, tarifação por demanda, menor risco de implantação;
- Agregadores de soluções: possibilidade de criação de novos mercados, oferecendo aplicações compostas e plataformas de integração;

- ISV's: possibilidade de novos modelos de negócio, fundamentados na "Cauda Longa", o que deve exigir uma reengenharia das soluções atuais;
- *Hosters* de SaaS: oferecer serviços compartilhados, como *billing*, SLA, monitoração, provisionamento, etc;
- *Hosters* clássicos: possibilidade de evolução para serviços com maior valor agregador, rumo ao modelo de *hoster* SaaS.

Nesta seção, observamos de modo geral, os conceitos, princípios e a arquitetura empregada nos sistemas SaaS. Muitas empresas não detêm corretamente esses conhecimentos, apresentando dificuldades em saber "o que é" e "para que serve" este novo paradigma. Esta falta de conhecimento irá impactar diretamente a definição de métricas e atributos de qualidade, pois tais métricas só podem ser levantadas de maneira eficiente se os conhecimentos aos princípios e arquitetura forem claros dentro do contexto avaliado.

2.4 Gestão de Serviços de TI

A abordagem da gestão de serviços de TI é fundamentada no presente trabalho para mostrar detalhes relevantes encontrados dentro dos SaaS. Muitos destes detalhes estão ligados a qualidade dos serviços de TI, que irão de maneira direta ou indiretamente impactar a qualidade dos sistemas SaaS.

De acordo com Miranda [2006], nos últimos anos a Gestão de Serviços de TI tornou-se uma área fundamental para as organizações, estas estão se tornando cada vez mais dependentes da informatização e dos processos de negócio. Essa dependência tem crescido tanto, que qualquer falha ou problemas nos serviços de TI, fazem as organizações perderem suas vantagens competitivas. Por isso, garantir a qualidade destes serviços tornou-se uma das atividades mais importantes das organizações, Lavado [2009].

A gestão de serviços de TI tem como filosofia definir, executar e manter os níveis requeridos de serviços de TI, de acordo com as prioridades do negócio e a um custo aceitável, Gomes et al. [2005]. De acordo com Weill [2009], atualmente, toda organização moderna que utiliza a TI precisa de ferramentas computacionais para o controle de suas atividades e serviços, garantindo-lhes uma maior qualidade e aceitação no mercado.

Muitas organizações de TI utilizam em prol de seus benefícios *frameworks* como a ITIL e guias de boas práticas como o COBIT, tendo como objetivo principal um

maior retorno dos investimentos dos departamentos de TI, juntamente com uma maior qualidade em seus serviços. O foco deste trabalho, dentro do contexto de gestão de serviços de TI, será na norma ISO/IEC 20000, da ITIL v3 (versão 3) e do COBIT 4.1, pelo motivo de que todos eles apresentam uma grande aceitação nas organizações, de acordo com uma pesquisa realizada pelo *IT Service Management Fórum Brasil*, itSMF [2007], Figura 2.8.

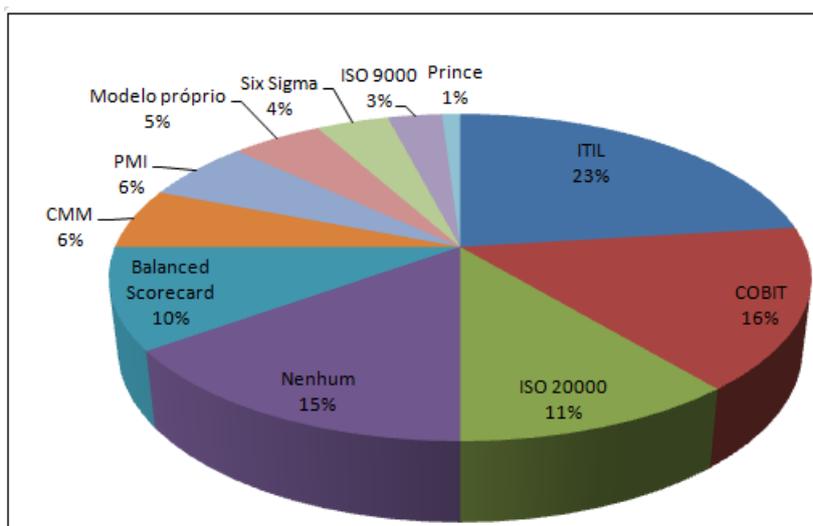


Figura 2.8. Modelos de governança de TI adotados por empresas, itSMF [2007].

O estudo de gestão de serviços de TI é relevante ao presente trabalho, pois devemos lembrar que o SaaS além de possuir características de um software convencional também apresenta muitas características ligadas a um serviço de TI, podendo ser considerado um software e um serviço ao mesmo tempo. A seguir, daremos uma abordagem detalhada das principais normas e modelos que auxiliam a gestão e a qualidade de serviços de TI.

Para a revisão teórica do assunto "Gestão de serviços de TI", esta seção foi estruturada da seguinte forma:

1. ISO/IEC 20000;
2. ITIL v3;
3. COBIT 4.1.

2.4.1 ISO/IEC 20000

Segundo o Instituto-Online [2006], a ISO/IEC 20000 foi a primeira norma formal para gestão de serviços de TI. Foi desenvolvida pelo *British Standards Institution* (BSI) em

novembro de 2000, é baseada no modelo de processos do ITIL e fornece especificações claras para implementação de processos de gestão de TI.

A Norma ISO 20000, procura por meio de sua implementação, proporcionar um efetivo sistema de gerenciamento, incluindo políticas e estrutura para permitir a gerência e implementação de todos os serviços de TI.

Uma de suas principais características é a de estar alinhada às recomendações previstas pelo *IT Infrastructure Library* - ITIL (Biblioteca de Infra-estrutura de TI). Os processos existentes na Norma ISO 20000 são baseados nas melhores práticas da ITIL. A Figura 2.9 mostra a correlação que deve existir entre os processos, segundo a ISO 20000.

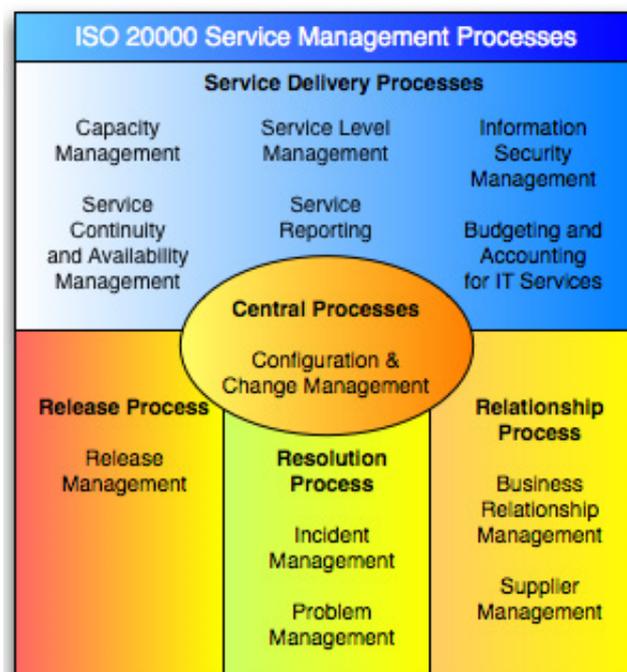


Figura 2.9. Relacionamento de processos na gestão da ISO 20000.

A norma está publicada em duas partes:

- Parte 1 - Especificação: promove a adoção de uma abordagem de processo integrada para efetivamente entregar serviços dirigidos ao encontro das necessidades do negócio e requisitos do cliente, ISO/IEC 20000-1 [2005];
- Parte 2 - Código de prática: fornece orientação e recomendações baseadas no consenso de indústria para promover melhoras no planejamento de serviços e/ou buscar auditoria pela norma ISO 20000, e pelos seus auditores ISO/IEC 20000-2 [2005].

O escopo da norma ISO/IEC 20000 integra a abordagem de processo baseados no sistema de gestão da qualidade ISO 9001, pela inclusão do ciclo PDCA (Planejar, Construir, Executar e Monitorar), e exigência da melhora contínua. A primeira parte da norma contém dez seções:

1. Escopo;
2. Termos e definições;
3. Requisitos de um sistema de gestão;
4. Planejamento e implementação de gestão de serviço;
5. Planejamento e implementação de serviços novos ou modificados;
6. Processos de entrega de serviço;
7. Processos de relacionamento;
8. Processos de resolução;
9. Processos de controle;
10. Processos de entrega.

Segundo Borges [2006], a norma ISO 20000 é importante, pois com ela se criam os requisitos certificáveis para a gestão de serviços de TI. Ela também define uma abordagem de processo integrada para a entrega de serviços gerenciada, oferecendo valor agregado à sua implementação e qualidade. Assim, como a reconhecida certificação ISO 9000, a ISO 20000 traz para a empresa mais um selo de qualidade (agora na gestão de TI), tornando-se um diferencial para o serviço de TI prestado.

2.4.2 ITIL v3

A ITIL v3 [2008] (*Information Technology Infrastructure Library*), pode ser considerada como um modelo, um conjunto de publicações ou um conjunto de boas práticas que contribuem para benefícios na área de gerenciamento de serviços de TI.

De acordo com Lawes [2003], a ITIL pode ser livremente traduzida por bibliotecas de infra-estruturas de TI. Esse trabalho de reunião das melhores práticas no gerenciamento de serviços de TI começou na década de 80 pela *Office of Government Commerce*, OGC [2008]. É composto basicamente por cinco livros principais, na Figura 2.10 podemos ver sua arquitetura.



Figura 2.10. Arquitetura da ITIL v3 [2008].

De acordo com Macfarlane & Rudd [2005], a ITIL v3 apresenta em sua arquitetura vários domínios que são na maioria deles inter-relacionados e são baseados em processos de boas práticas, impactando diretamente o ciclo de vida de seus serviços. Abaixo segue uma descrição de seus livros, juntamente com seus processos.

Service Strategy: a principal característica deste livro é o foco nas estratégias, políticas e restrições sobre os serviços da TI, tendo como principal preocupação a parte financeira. Neste livro os requisitos de negócio são identificados e os resultados esperados são acordados em um SLP (*Service Level Package*). Seus processos se dividem em:

- Gerenciamento Financeiro: responsável pela estrutura de custos, opção pela cobrança e quantificação do valor dos serviços;
- Gerenciamento do portfólio de serviços: responsável por governar investimentos na gerência de serviços e adicionar valores ao negócio;
- Gerenciamento de Demanda: responsável pela produção dos serviços e do ciclo de consumo dos serviços.

Service Design: traz um guia para desenho e evolução contínua dos serviços e seus elementos associados. Propõe também desenhos de serviços de TI adequados e inovadores, incluindo arquiteturas, processos, políticas e documentação, para atender

aos requisitos de negócio estabelecidos no presente e no futuro. Seus processos se dividem em:

- Gerenciamento de catálogo de serviços: responsável por tratar as informações atualizadas, tanto dos negócios como dos serviços disponíveis;
- Gerenciamento do Nível de serviço: responsável por manter e melhorar a qualidade dos serviços. Utiliza atividades como planejamento, coordenação, elaboração, acordo de metas de desenho, monitoramento e divulgação;
- Gerenciamento da Capacidade: gerencia a capacidade da infra-estrutura, fazendo com que consiga absorver as novas demandas do mercado;
- Gerenciamento da Disponibilidade: responsável por atender os níveis de disponibilidade e confiabilidade do negócio, proporcionando minimização de risco de interrupção;
- Gerenciamento da Continuidade de serviço: responsável por garantir continuidade e não interrupção de todos os recursos e serviços de TI, fazendo com que possam ser recuperado em tempo hábil. Faz a utilização de planos de recuperação, análise e gerencia de riscos;
- Gerenciamento da Segurança da informação: responsável pelo tratamento da confidencialidade, integridade, disponibilidade dos serviços e recursos;
- Gerenciamento de Fornecedor: responsável pelo gerenciamento entre fornecedor e contratante.

Service Transition: proporciona um guia para implantação dos serviços em produção, a partir da conclusão do estágio de *Service Design*. Tem foco em todos os aspectos dos serviços (incluindo suporte em caso de falhas), não apenas no uso normal. Para sua compreensão é necessário criar valores para o negócio, partes interessadas, tais como fornecedores, clientes e outras áreas afetadas. Seus processos se dividem em:

- Gerenciamento de Mudança: responsável pelo tratamento das mudanças minimizando impactos na qualidade do serviço;
- Gerenciamento da Configuração e de ativos: responsável por registrar ativos e realizar a sua proteção;

- Gerenciamento de Liberação e implantação: responsável por implantar componentes finais no ambiente de produção, gerenciar itens aprovados e a políticas de liberação;
- Validação e testes: responsável pela qualidade da liberação, validação e liberação do que está pronto para uso;
- Avaliação: realiza meios para o desempenho das mudanças;
- Gerenciamento do conhecimento: responsável pelas informações sobre a entrega dos serviços.

Services Operation: aborda características das operações de rotina juntamente com o suporte de serviços. É o único estágio em que os serviços efetivamente entregam valores ao cliente, onde a responsabilidade da equipe de operação é garantir essa entrega. Seus processos se dividem em:

- Gerenciamento de Evento: responsável por monitorar todos os eventos na infraestrutura de TI;
- Gerenciamento de Incidentes: responsável por restaurar a operação normal do serviço;
- Gerenciamento de Problema: responsável por minimizar os impactos de incidentes e problemas para o negocio;
- Cumprimento de requisição: realiza requisições de usuários a partir de solicitações de serviços;
- Gerenciamento de acesso: responsável pelo controle de acesso dos usuários ao utilizar serviços.

Continual Service Improvement: é o livro que traz a maior inovação da ITIL v3, esse livro trás aspectos de como garantir o bom ciclo de vida de um serviço, utilizando o ciclo PDCA (Planejar, Construir, Executar e Monitorar). Apresenta um guia para avaliação e melhoria de maturidade de gerenciamento de serviços; qualidade e alinhamento dos serviços; e desempenho dos processos de TI. Apresenta somente dois processos:

- Relatório de serviço e Medição de serviço: ambos focado na melhoria contínua dos serviços de TI

De acordo com Macfarlane & Rudd [2005], os livros mais importantes da ITIL v3 para o gerenciamento de serviço de TI são: *Service Design* e *Service Transition*, pois se concentra diretamente nas estratégias dos serviços, possibilitando uma melhor entrega e suporte dos serviços, juntamente com um melhor planejamento e melhoria na prestação de seus serviços a longo prazo.

Para Bon et al. [2006], a ITIL adota uma estratégia orientada a processos, com o intuito de prestar benefícios tanto para grandes como para pequenas empresas. Um ponto forte da ITIL é o relacionamento existente entre as atividades de um conjunto de processos estreitamente integrados, considerados importantes para muitas organizações, principalmente no tocante à qualidade de seus serviços.

O estudo da ITIL nos proporciona um arcabouço coeso de informações sobre gerenciamento de serviços de TI. Dentro deste gerenciamento encontramos diversas características sobre garantia da qualidade, que se enquadram diretamente nas características dos SaaS, como por exemplo, disponibilidade, segurança, níveis de serviços, suporte, entre outros.

As áreas da ITIL que mais contribuíram para este trabalho foram os livros *Service Design*, *Service Transition* e *Continual Service Improvement*, revelando aspectos importantes sobre o ciclo de vida destes serviços.

2.4.3 COBIT 4.1

De acordo com a ISACA [2008], o COBIT (*Control Objectives for Information and Related Technology*), é um *framework* que traz um conjunto de boas práticas, compostos por domínios e processos apresentando uma estrutura manejável e lógica para a gerência dos serviços da TI.

Para a ITGI [2007] (*Information technology governance institute*), o *framework* COBIT 4.1 foi criado tendo como base normas internacionais, tentando estabelecer métodos documentados e estrutura de guias para o gerenciamento e auditoria da área de TI, o que permite uma efetividade maior da governança de TI dentro das organizações.

Em estudos realizados por Vieira [2004], o COBIT pode ser utilizado para a realização de um melhor alinhamento estratégico da TI com o foco de negócio da organização.

O COBIT se baseia nos conceitos fundamentais da governança de TI, esses conceitos estão divididos em cinco áreas principais, segue abaixo um pequeno detalhamento sobre cada uma (Figura 2.11).

- *Strategic Alignment* (Alinhamento Estratégico): é focada na garantia da ligação entre o negócio e os planos de TI, alinhando as operações da empresa com as de

TI juntamente com definição, manutenção e validação do valor proposto para a TI;

- *Value Delivery* (Entrega de Valor): agregar valor com os benefícios das entregas de serviços de TI de acordo com a estratégias de negócios da organização, otimizando custos e riscos;
- *Resource Management* (Gerenciamento de Recursos): é caracterizada por otimizar investimentos nas operações críticas da TI juntamente com um gerenciamento adequado dos recursos, (aplicações, informações, infra-estrutura e pessoas);
- *Risk Management* (Gerenciamento de Riscos): objetivo de divulgação dos principais riscos para a alta direção, entendimento da conformidade de requisitos, transparência sobre os riscos significativos para a empresa, e incorporação da responsabilidade de gerenciamento dos riscos;
- *Performance Measurement* (Monitoramento de Performance): tem como objetivo monitorar e acompanhar a implementação da estratégia, finalização/andamento dos projetos e desempenho de processos de entrega dos serviços utilizando indicadores de desempenho.

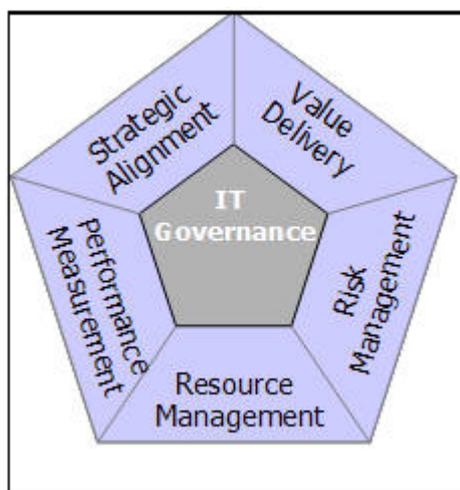


Figura 2.11. As principais áreas da governança de TI, ITGI [2007].

De acordo com a ISACA [2008], a arquitetura do COBIT pode ser visualizada como um conjunto de processos agrupados em domínios, em conjunto de recursos, considerando-se alguns critérios de informações a serem atendidos para os requisitos de negócio da organização, Figura 2.12.

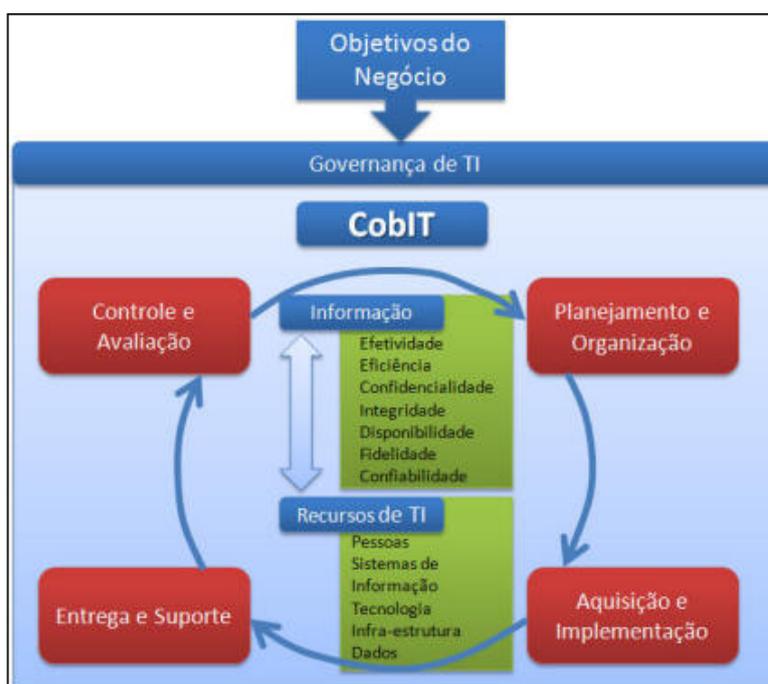


Figura 2.12. Arquitetura do *framework* COBIT 4.1, ISACA [2008].

Os processos de TI do COBIT estão divididos dentro de quatro domínios, ITGI [2007], são eles:

- PO - *Plan and Organise* (Planejamento e Organização);
- AI - *Acquire and Implement* (Aquisição e Implementação);
- DS - *Deliver and Support* (Entrega e Suporte);
- ME - *Monitor and Evaluate* (Monitoração e Avaliação).

Em cada domínio existem processos que levam a metas de negócio. Essas metas são definidas de forma que haja correlação entre elas. As metas por atividade permitem o acompanhamento eficaz do desempenho do processo.

Para Vieira [2004], o COBIT também pode ser visto de uma forma lógica e estruturada, abrangendo praticamente todas as atividades de TI, possibilitando uma medição de desempenho de seus processos através de parâmetros e controles, relacionando, dentre outros fatores, os riscos de negócios e as necessidades de controle de ferramentas.

Fica clara a gama de benefícios que ferramentas como o COBIT podem trazer para as organizações, diversas empresas adotando-o em seu dia-dia como forma de

controle dos investimentos em TI, principalmente pela relação da TI às estratégias da organização, bem como uma maior garantia em seus serviços oferecidos pela TI.

Os estudos realizados sobre o COBIT, mais precisamente no domínio *DS - Deliver and Support* (Entrega e Suporte) proporcionaram ao atual trabalho informações complementares às características e requisitos de qualidade encontrados dentro da ITIL. Ambas as ferramentas tiveram um papel importante, principalmente na agregação de conhecimentos de gestão de serviços de TI dentro do contexto SaaS, proporcionando um modelo de qualidade que seja eficiente e coeso a este novo paradigma de software.

2.5 Trabalhos Relacionados

Pesquisas referentes à garantia da qualidade em sistemas SaaS são relativamente recentes na comunidade científica. No entanto, o crescente aumento no número de publicações em periódicos e conferências especializadas na área qualidade demonstra o claro interesse da comunidade por iniciativas no sentido de estabelecer critérios que auxiliem na mensuração da qualidade neste novo paradigma. Esta seção apresenta a revisão dos trabalhos correlatos, mais relevantes ao tema do trabalho, exemplificando e expondo o que atualmente está sendo pesquisado nesta área.

Iniciamos analisando o trabalho de Dubey & Wagle [2006], onde foram descritos os principais problemas e preocupações que as organizações estão tendo com os sistemas SaaS. Muitos destes problemas podem afetar diretamente a qualidade dos produtos SaaS, além de impactar outros fatores, como por exemplo, aspectos funcionais, conformidades, comerciais, entre outros, são eles:

- **Acesso:** pois no contexto SaaS, os sistemas funcionam online no servidor do fornecedor, devido a este aspecto uma das exigências para sua adoção é que o acesso à internet seja rápido, estável e redundante;
- **Aplicação:** os sistemas de missão crítica para a empresa, como os responsáveis pelas vendas (frente de loja e estoque), não são recomendados no modelo SaaS. Mesmo com acesso rápido, estável e redundante, a empresa não está imune à indisponibilidade, o que pode causar perda temporária de operação;
- **Personalização:** ao optar por um SaaS, a possibilidade de personalização é quase inexistente. Afinal um dos aspectos que o torna de baixo custo é ser padrão para todos os assinantes;
- **Integração:** por funcionar no servidor do fornecedor e ser padronizado, o SaaS não se integra automaticamente com os demais sistemas instalados no ambiente

de trabalho do contratante. A troca de informações pode acontecer, contudo, de forma não trivial, através da importação e da exportação de dados.

Mesmo analisando as principais preocupações deste novo paradigma, Dubey & Wagle [2006] não apresentam métricas nem critérios que auxiliam e contribuam para uma avaliação prática destes problemas. Porém, as análises feitas em seu trabalho, juntamente com a definição dos principais problemas foram de grande contribuição ao presente trabalho, principalmente para a elaboração de um modelo de qualidade específico aos SaaS.

Outro trabalho ligado a qualidade de sistemas SaaS, foi o de O'Brien et al. [2005], onde realizaram uma análise entre a arquitetura orientada a serviços (SOA), e atributos de qualidade de software, tendo em vista que a arquitetura de um software é um fator importante entre os objetivos de negócio de uma organização e um sistema computacional. Um dos objetivos do trabalho foi examinar como uma arquitetura SOA (arquitetura utilizada nos sistemas SaaS) suporta atributos de qualidade, como disponibilidade, segurança, confiabilidade, portabilidade e interface, analisando questões e problemas relacionados a eles.

Devido ao SOA ser ainda uma arquitetura incipiente, muitas das questões entre os fatores de qualidade não foram exaustivamente pesquisados, e muitas das normas de qualidade referentes a este cenário estão imaturas. Neste contexto, O'Brien et al. [2005] também analisa que quando uma organização busca desenvolver sistemas SaaS, utilizando abordagens da arquitetura SOA, deve-se fazer modificações cautelosas em seus requisitos de qualidade, pois poderão de certa forma impactar a organização tanto em um aspecto positivo como negativo. O'Brien et al. [2005] mesmo analisando diversos atributos de qualidade para a arquitetura SOA, não abordam a idéia de construção de um método ou modelo que possibilitaria a aplicação prática desses atributos de qualidade, mas identifica diversas características que são ligadas tanto em um produto de software como em um serviços de TI, integrando perfeitamente ao paradigma SaaS.

Seguindo a mesma lógica de raciocínio dos trabalhos anteriores, Havelka et al. [1998], abordaram em seu trabalho duas importantes áreas da qualidade em produtos SaaS, visando uma análise sistemática dos impactos que determinados requisitos teriam sobre estes sistemas, principalmente em requisitos funcionais.

Para realização do trabalho de Havelka et al. [1998], um estudo empírico foi proposto envolvendo alguns usuários da área de TI, com objetivo de identificar fatores críticos que poderiam impactar tanto de modo positivo como negativo os sistemas SaaS. Os fatores identificados por esses usuários foram posteriormente integrados em um instrumento de questionário, para levantar evidências do verdadeiro impacto dos

aspectos levantados. O estudo levantou diversos fatores impactantes, porém, sem muita garantia de relevância, pois a pesquisa foi feita somente na opinião e conhecimento dos usuários, não tendo um software concreto como referência a estes impactos.

A principal contribuição retirada do trabalho de Havelka et al. [1998] é o fato que, os impactos dos requisitos de qualidade em sistemas SaaS podem afetar tanto positivamente como negativamente estas aplicações. Por isso antes de se iniciar a fase de levantamento de requisitos para construção de sistemas SaaS, deve-se avaliar com cuidado os impactos contidos neste novo paradigma de software.

Na mesma lógica dos trabalhos anteriores, mas entrando em aspectos ligados a proposta de um método que auxilie a mensuração da qualidade em sistemas SaaS, destacamos o trabalho de Jureta et al. [2009]. Nesta pesquisa, tendo em vista que os produtos SaaS apresentam perspectivas diferentes em relação a qualidade de um software convencional, como por exemplo, características de portabilidade, suporte, personalização, e não encontrando na literatura uma metodologia que contemplasse aspectos ligado ao paradigma Saas, Jureta et al. [2009] analisaram as ISO's 9126 e 20000, propondo um novo método de qualidade chamado QVDP.

Este método tem ênfase somente na avaliação da qualidade voltado para características de serviços de TI, pecando em alguns aspectos de qualidade de produtos de software. Foram criados também sub-modelos com a intenção de utilizá-los como ponto de referência para a evolução da qualidade em sistemas SaaS (possuindo diversos atributos de qualidade), porém o modelo não possui um processo próprio de avaliação, dificultando sua utilização por parte das empresas, principalmente nas atividades, etapas e artefatos que possam ser utilizados.

Seguindo a mesma linha de pesquisa, Cancian [2009] considerando que a dificuldade em garantir a qualidade em sistemas SaaS poderia dificultar sua contratação por parte das organizações, avaliou e estudou o processo de desenvolvimento desses sistemas. O processo de desenvolvimento de software é um dos fatores que pode impactar diretamente a qualidade final destes produtos. Cancian [2009] propôs um guia de referência para avaliação do processo de desenvolvimento de sistemas SaaS, baseando-se em diversos requisitos de qualidade, realizando posteriormente uma síntese e análise dos requisitos, mapeado-os em relação a normas e modelos de referência já existentes no mercado.

O estudo desse trabalho é relevante ao presente trabalho, pois muitas das características de qualidade do processo de desenvolvimento de soluções SaaS podem ser reaproveitadas para a qualidade do produto final, vinculadas a qualidade externa e interna. O trabalho de Cancian [2009] não possui uma aplicação prática em um estudo de caso real, mas apresenta um guia completo e fundamentado em referência a

avaliação do processo de desenvolvimento de sistemas SaaS.

Em outro trabalho realizado por Cancian et al. [2009], observando a necessidade de um entendimento claro em um conjunto de compromissos firmados entre um fornecedor e um cliente de soluções SaaS (SLA - *Service Level Agreement*), foi proposto uma estrutura genérica de SLA, definindo itens de qualidade que mais se adequam ao SaaS.

Além da estrutura genérica, Cancian et al. [2009] também identificou requisitos de qualidade relevantes a serem observados sob o modelo SaaS, principalmente nos níveis de serviço (SLA's). Os SLA's são importantes para a garantia da qualidade, pois trazem uma maior segurança na contratação de um serviço, sendo um instrumento cada vez mais presente e importante dentro das negociações. O presente trabalho se baseou em muitos requisitos de qualidade propostos para os SLA's, contemplando e adicionado informações de normas e modelo, como a ITIL e o COBIT.

No trabalho de Rosenberg et al. [2008], foi verificado que o desenvolvimento de produtos SaaS é uma tarefa difícil e propensa a erros, tendo dificuldade até mesmo em se garantir uma boa qualidade em seu produto final. Para isso foi proposto no trabalho uma composição automatizada de serviços denominada CaaS (*Composition as a Service*), onde facilita o desenvolvimento rápido em pequenas aplicações orientadas a serviços, invocando diversos serviços garantindo uma maior eficiência durante o desenvolvimento e uma maior qualidade interna do software, gerando uma padronização em sua arquitetura e em seu código fonte.

A contribuição desse trabalho se reflete no fato de que, sua composição automatizada CaaS (*Composition as a Service*) traz diversos aspectos de qualidade interna, que devem ser analisados e estudados dentro de sistemas SaaS.

Em outro contexto, ligado a qualidade de serviços de TI, Silva et al. [2006] elaboraram uma proposta para analisar a administração dos serviços corporativos de Tecnologia da Informação (TI) em empresas onde esta função é estratégica e vital para a competitividade dos negócios, bem como entender o desafio de se melhorar a percepção dos gestores de negócios, que são os clientes internos das áreas de TI, quanto à qualidade dos serviços prestados.

A proposta foi feita através de uma pesquisa exploratória com os clientes de TI, representados pelos gestores de negócios, a partir da técnica de incidentes críticos, visando a obtenção das dimensões consideradas mais importantes no processo de avaliação da qualidade destes serviços. Os resultados obtidos mostraram a importância de uma correta avaliação das necessidades específicas de um cliente de serviço, servindo de base para uma condução mais segura nas decisões estratégicas de encaminhamento dos negócios. Como contribuição, o trabalho traz uma análise construtiva sobre alguns aspectos de fatores de qualidade em serviços de TI, como por exemplo, níveis de

serviço, disponibilidades, continuidades dos serviços e auditabilidade.

Miguel & Salomi [2004] em suas pesquisas, apresentam uma revisão na literatura sobre os principais modelos e conceitos propostos referentes à mensuração da qualidade em serviços externos. A revisão teve um cunho analítico, temático, histórico e bibliográfico. Após fazerem apresentações dos modelos mais eficientes e utilizados, uma comparação entre eles foi realizada mostrando suas principais características. De acordo com os resultados do trabalho, foi concluído que não existe um consenso na literatura quanto ao modelo mais apropriado para a mensuração da qualidade em serviços de TI. Além disso, existem questões ainda não respondidas, como por exemplo, a relação de causalidade entre qualidade dos serviços de TI e a satisfação de seus clientes.

Capítulo 3

Metodologia

Sempre que se inicia uma pesquisa científica, o primeiro passo deve ser a delimitação do método científico e do tipo de pesquisa, pois de acordo com Vergara [2004] "o método é um caminho, uma forma, uma lógica de pensamento", ou seja, é o método da pesquisa que norteia o pesquisador na realização de seu trabalho. Este capítulo descreve e classifica o tipo de pesquisa utilizada, além de apresentar o detalhamento de seus processos metodológicos, que vão desde a concepção e revisão bibliográfica, até procedimentos relacionados à proposição, execução e validação das contribuições do método MAQSaaS.

Espera-se que através deste capítulo seja possível a outros pesquisadores da área compreender o método de pesquisa utilizado, bem como reproduzir em trabalhos futuros os procedimentos aqui adotados.

3.1 Tipo de Pesquisa

Na literatura, de acordo com Jung [2004] e Marconi & Lakatos [2003] a pesquisa científica pode ser classificada em 4 (quatro) dimensões:

- quanto à sua natureza;
- quanto aos seus objetivos;
- quanto aos procedimentos;
- e quanto ao local de realização da mesma.

Segundo Jung [2004], exemplificado pela Figura 3.1, quanto à natureza, o trabalho pode ser classificado como uma pesquisa básica/fundamental ou aplicada/tecnológica,

quanto a seus objetivos a pesquisa pode ser exploratória, descritiva ou explicativa, já em relação aos procedimentos a pesquisa pode ser experimental, operacional ou um estudo de caso.



Figura 3.1. Tipos de pesquisa segundo Jung [2004].

Outros autores, Wohlin et al. [2000], propõem uma classificação que abrange quatro métodos relevantes de pesquisa, principalmente no detalhamento do método empírico, por esse ser bastante apropriado para pesquisas em engenharia de software. Os quatro métodos são:

- **Método científico:** a realidade é observada, e um modelo é construído para representá-la;
- **Método de engenharia:** alguma evolução das soluções atuais é proposta após essas terem sido estudadas;
- **Método empírico:** um modelo é proposto e evoluído através de estudos empíricos (pesquisa qualitativa e quantitativa);
- **Método analítico:** uma teoria formal é proposta e então comparada com as observações empíricas.

Após as fundamentações dos autores, esta pesquisa classifica-se como qualitativa, de natureza tecnológica, com objetivos de caráter exploratórios/descritivos, utilizando procedimentos de estudos de caso e baseando seus materiais e métodos em pesquisas bibliográficas e documental.

De acordo com os fundamentos encontrados no trabalho de Wohlin et al. [2000], a atual pesquisa pode ser classificada como qualitativa, pois baseia-se no julgamento humano, buscando entender um fenômeno específico da qualidade de software, trabalhando com descrições, comparações e interpretações.

Fundamentado em Jung [2004] e Silva & Menezes [2000], o trabalho é considerado de natureza tecnológica, pois objetiva combinar conhecimentos para geração de novos produtos, processos e métodos. Assim o trabalho irá propor um novo método para avaliar a qualidade em um produto SaaS, utilizando, agregando e combinando técnicas de avaliação de produto de software com ferramentas que auxiliam o gerenciamento de serviços da TI.

Quanto aos objetivos, é de caráter exploratório(Gil [1991]) e descritivo(Jung [2004]), pois visa à descoberta de novas práticas, registrando e analisando suas características em modelos de avaliação da qualidade de produtos de softwares para um novo contexto, o SaaS, objetivando a criação de um novo método de avaliação.

O trabalho utiliza procedimentos de estudo de caso (Jung [2004]), permitindo estudar o método proposto dentro de um contexto local e real, possibilitando definir limites mais claramente, facilitando sua validação e avaliação.

As abordagens teóricas são fundamentadas em revisões bibliográficas e pesquisas documentais, visando à análise de livros de diversos autores, periódicos, revistas e demais publicações a respeito do tema proposto. Proporcionando uma gama de informações e fenômenos, principalmente quando os dados estão dispersos, tendo o cuidado com a origem e a relevância das informações.

3.2 Procedimento Metodológico

O processo metodológico seguido no trabalho foi definido em 4 (quatro) fases:

1. Concepção e Preparação da Pesquisa;
2. Análise e Resultados Preliminares;
3. Validação e Melhorias no Método;
4. Verificações e Resultados Finais.

A descrição e as especificações de cada fase, juntamente com os resultados esperados em relação às respectivas etapas, são listadas a seguir nas próximas subseções, de modo a denotar um aspecto construtivo e seqüencial da pesquisa, evidenciando a evolução dos trabalhos executados.

3.2.1 Concepção e Preparação da Pesquisa

A primeira fase diz respeito à concepção e aspectos iniciais para a realização da pesquisa em questão, a Figura 3.2 exemplifica cada uma de suas etapas, sendo descritas logo a seguir.

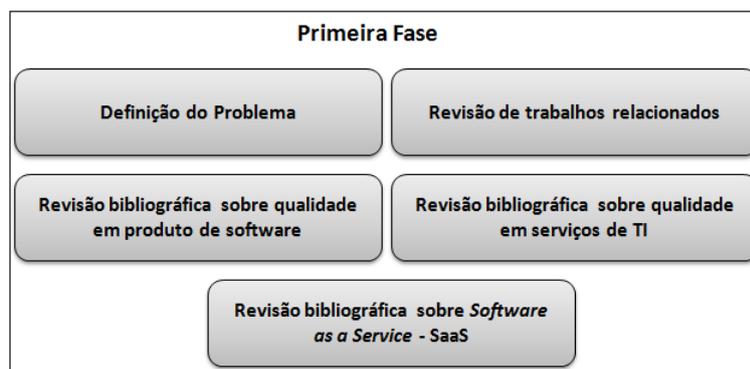


Figura 3.2. Primeira fase do processo metodológico.

Definição do problema: uma das etapas mais importante de um trabalho científico é a definição da problemática, ou seja, definir claramente e objetivamente qual o problema a ser estudado e planejado. A definição do problema deve levar em conta as necessidades no meio científico, verificando se os esforços são realmente válidos e relevantes.

No presente trabalho (seção 1.2) a questão abordada é a falta de um método prático e eficiente que avalie e auxilie a qualidade em um produto SaaS, juntamente com um modelo de qualidade específico ao paradigma.

Revisão de trabalhos relacionados: esta etapa se refere a revisões de trabalhos atuais que sejam relacionados com o tema proposto neste trabalho. Estas revisões mostram o que outros pesquisadores já tiveram como resultados, e os problemas que ainda persistem, permitindo criar objetivos mais específicos ao trabalho.

Os trabalhos pesquisados e relacionados ao tema "Qualidade de software" e "SaaS" (seção 2.5), mostram iniciativas de pesquisadores dentro da área de qualidade de software voltado para novas características proporcionadas pelo paradigma SaaS, apresentando benefícios (resultados positivos) e dificuldades (problemas em aberto) presentes no tema do trabalho.

Revisão bibliográfica sobre qualidade em produto de software: esta etapa (seções 2.1 e 2.2) se refere à coleta e estudos de materiais bibliográficos, revisões

literárias de métodos e modelos utilizados para auxiliar a avaliação de qualidade sobre produtos de softwares.

A revisão tem por objetivo dar uma visão geral aos leitores sobre aspectos relacionados à mensuração da qualidade em sistemas computacionais, juntamente com as dificuldades encontradas e alternativas propostas por diversos autores.

Revisão bibliográfica sobre Software oferecido como serviços (SaaS): nesta etapa (seção 2.3) é feito um levantamento bibliográfico de definições de o que é o SaaS, juntamente com as características presentes neste novo paradigma de software, de maneira a mostrar aos leitores limitações e dificuldades em sua utilização e implantação.

Revisão bibliográfica sobre qualidade em serviços de TI: esta etapa (seção 2.4) se refere à coleta e estudos de materiais bibliográficos e revisões literárias sobre gerenciamento de serviços de TI, tendo como base a utilização de *frameworks* e guias de boas práticas, englobando experiências tanto nacionais como internacionais.

Nesta etapa foi dado um enfoque na ITIL v3, COBIT 4.1 e a ISO 20000, proporcionando um conjunto de características e aspectos de gerenciamento de serviços de TI. Esse estudo permitiu identificar o que existe de diferente e o que se pode aproveitar em relação à qualidade de um produto de software e um serviço de TI.

3.2.2 Análise e Resultados Preliminares

Nesta fase os autores efetuaram análises e estudos em relação a características e requisitos de qualidade voltados ao paradigma SaaS, possibilitando a construção de um modelo de qualidade específico. A Figura 3.3 descreve cada uma de suas etapas.

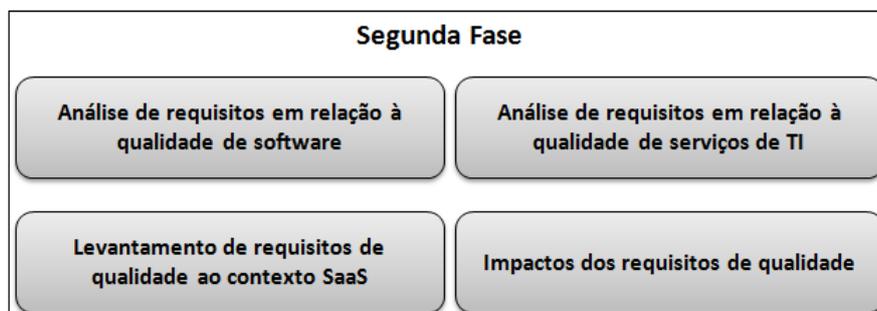


Figura 3.3. Segunda fase do processo metodológico.

Análise de requisitos em relação à qualidade de software: a análise abrangeu estudos de normas e modelos de qualidade, proporcionando uma análise de critérios de qualidade esperados em um produto de software. Esta análise possibilitou identificar quais características possuem aplicabilidade e escopo dentro dos sistemas SaaS.

Análise de requisitos em relação à qualidade de serviços de TI: foram utilizados *frameworks* e guias de boas práticas de gerenciamento de serviço de TI, com o objetivo de levantar e analisar critérios de qualidade que possam ser encontrados também em sistemas SaaS, lembrando que esses sistemas apresentam características semelhantes a de um software convencional e um serviço de TI.

Levantamento de requisitos de qualidade ao contexto SaaS: o objetivo desta etapa (seção 4.1) é levantar e integrar em um só conjunto critérios e requisitos de qualidade referentes a um produto SaaS, possibilitando sua utilização em um método de avaliação.

A integração foi feita utilizando comparações e definições de aplicabilidade entre as diversas características levantadas, com base na norma ISO 9126, ITIL v3, COBIT 4.1 e da norma ISO 20000, de modo a identificar características de qualidade relevantes ao contexto SaaS.

Após a identificação e integração dos critérios, houve uma complementação e priorização destas características, através de uma pesquisa *online* com pessoas relacionadas ao paradigma SaaS (empresários, especialistas e pesquisadores), proporcionando uma maior complementação e confiabilidade nas características levantadas.

A pesquisa foi enviada através de contatos de fóruns, blogs e listas de discussão, todas tendo amplo domínio do tema *Software as a Service*. A carta convite para participação da pesquisa pode ser visualizada no Apêndice A, onde, no total foram selecionadas 85 pessoas para a sua participação.

Das 85 pessoas selecionadas, após o envio da carta convite, 63 responderam e deram o feedback correto sobre a pesquisa, juntamente com o preenchimento do formulário. O formulário utilizado nas pesquisas de complementação e priorização dos requisitos de qualidade pode ser visualizado no Apêndice B, sua construção e fundamentação serão exemplificadas no Capítulo 4, seção 4.1.

Impacto dos requisitos de qualidade: após o levantamento e a definição do conjunto de requisitos específicos ao modelo SaaS, podendo ser utilizado em avaliações práticas, é importante deixar claro os impactos tanto positivos e negativos que estes

têm sobre os objetivos de negócio das organizações (seção 4.2).

3.2.3 Validação e Melhorias no Método

A terceira fase metodológica do trabalho faz referência ao início do estudo de caso, onde tem por objetivo validar o método proposto, juntamente com a amplitude do modelo de qualidade. Ainda nesta fase, um processo de avaliação foi definido, tendo também uma etapa de melhoria contínua para o MAQSaaS. A Figura 3.4 descreve cada uma de suas etapas.

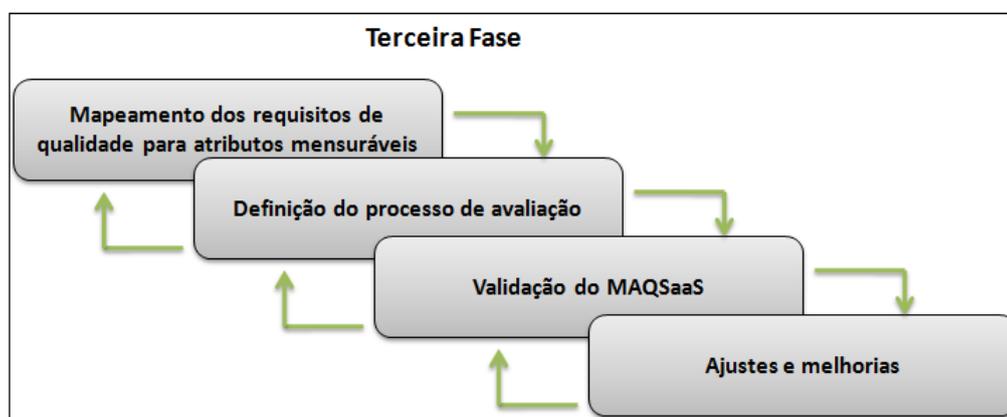


Figura 3.4. Terceira fase do processo metodológico.

Mapeamento dos requisitos de qualidade para atributos mensuráveis: após a definição das características de qualidade a ser utilizado pelo MAQSaaS, deve-se mapear todas estas para atributos mensuráveis. O mapeamento destas características para atributos mensuráveis não é uma tarefa trivial, necessitando o desdobramento dessas características em atributos que possam ser medidos e pontuados, juntamente com uma fundamentação e visão de autores da área de qualidade em sistemas SaaS. Os atributos devem ser os mais objetivos possíveis, facilitando sua utilização em guias e métodos de avaliação.

Este mapeamento possibilitou a confecção de uma lista de verificação (*checklist*), possibilitando uma avaliação prática dos requisitos de qualidade. Os detalhes da construção e fundamentação da lista de verificação podem ser encontrados nas seções 5.3 (Capítulo 5) e 6.1.4 (Capítulo 6).

Definição do processo de avaliação: esta etapa (seções 5.2 e 5.3) se refere à criação de um processo de avaliação que seja repetível, reproduzível, imparcial e

objetivo.

O processo de avaliação do MAQSaaS foi baseado na norma ISO/IEC 14598, porém algumas modificações e ajustes foram feitos visando uma maior adequação aos produtos SaaS, juntamente com os papéis desempenhados pelos atores participante do processo. Inevitavelmente a ITIL v3 e o COBIT 4.1 tiveram um papel importante nesses ajustes, pois foram neles que os autores buscaram características próprias e relevantes ao contexto de gerenciamento de serviços de TI. O processo de avaliação pode ser visto com mais detalhes no Capítulo 5 deste trabalho.

Validação do MAQSaaS: esta etapa (seção 6.1) faz referência a validação do método proposto relacionado a um estudo de caso. O estudo de caso é importante, pois mostra dentro de um ambiente real os objetivos alcançados, limitações e possíveis melhorias.

Para a realização do estudo de caso, avaliações efetuadas sobre o produto "SalesForce CRM" foram realizadas tendo como base diretrizes e métricas apresentadas dentro do método MAQSaaS. O estudo de caso também tem por objetivo avaliar a amplitude do modelo de qualidade (critérios e requisitos) proposto neste trabalho, juntamente com o processo de avaliação modelado no MAQSaaS.

Ajustes e melhorias: durante a etapa de validação (Estudo de Caso), alguns problemas e limitações foram encontrados durante as primeiras avaliações práticas. As avaliações preliminares se mostraram pouco eficientes em relação ao tempo demandado, tendo pouca objetividade em relação a algumas questões da lista de verificação.

Esta etapa teve a finalidade de ajustar e aperfeiçoar alguns pontos do método de avaliação, principalmente no processo de avaliação e no mapeamento dos atributos mensuráveis, possibilitando uma maior repetitividade, reprodução, imparcialidade e objetividade ao método.

Todos os erros identificados foram corrigidos ao longo do Estudo de Caso, tendo a necessidade de se aumentar o número de avaliações, garantindo assim uma maior confiança e aplicabilidade para o método.

3.2.4 Verificações e Resultados Finais

Por fim, a última fase proporcionou resultados finais através de um comparativo, juntamente com verificações dos resultados obtidos na comunidade científica, especificamente na área de "Qualidade de Software". A Figura 3.5 descreve cada uma de suas etapas.

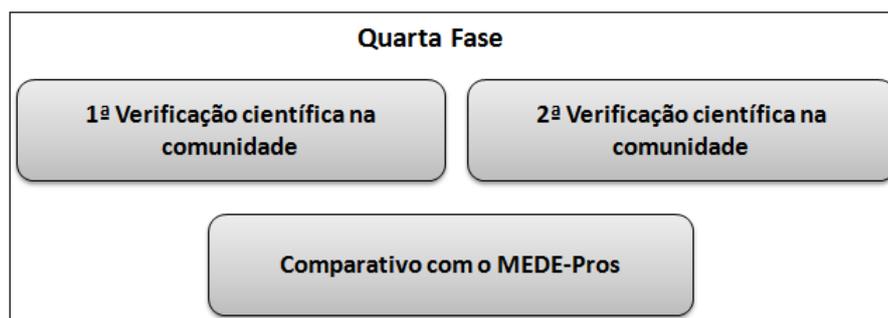


Figura 3.5. Quarta fase do processo metodológico.

1ª Verificação científica na comunidade: a primeira verificação científica em relação à relevância dos primeiros resultados foi efetuada através da submissão de um artigo, publicado e aceito no 5th CISTI (*Conferencia Ibérica de Sistemas y Tecnologías de Información*), intitulado "Atributos de qualidade para produtos SaaS". Os comentários dos revisores proporcionaram pontos positivos e críticas construtivas ao trabalho, principalmente na parte da construção do modelo de qualidade.

2ª Verificação científica na comunidade: a segunda verificação científica foi feita através da submissão de um artigo, publicado e aceito no III WTDSI (Workshop de Teses e Dissertações em Sistemas de Informação) ocorrido no VI SBSI 2010 (Simpósio Brasileiro de Sistemas de Informação), intitulado "MAQSaaS - Método para Avaliação da Qualidade em Produtos de Softwares Oferecidos como Serviços". Os revisores identificaram pontos importantes sobre o processo de avaliação proposto no presente trabalho, constituindo idéias para a sua melhoria.

Comparativo com o MEDE-Pros: após a etapa de validação do método MAQSaaS, foi efetuado um comparativo com um método de avaliação convencional (seção 6.2), no caso o MEDE-Pros, para complementar a sua validação. Este comparativo mostra diversos pontos que não são abordados ou avaliados pelos métodos convencionais, juntamente com limitações encontradas.

Capítulo 4

Modelo de Qualidade para o MAQSaaS

A identificação de critérios e requisitos de qualidade em um sistema computacional é uma tarefa importante para a missão/objetivos de negócios de uma organização. Esta tarefa contribui para a agregação de valor ao produto de software, principalmente na definição e avaliação de sua qualidade, Herssens et al. [2008]. Porém, a tarefa em questão não é algo trivial de ser realizar, em outras palavras, podemos dizer que muitas organizações não sabem ao certo "o que medir" para conseguir a qualidade pretendida em um sistema computacional, Gatti & Werneck [2004].

Nesse contexto, de identificação de requisitos de qualidade, observamos dificuldades maiores dentro do paradigma SaaS, pois, como é analisado em Goth [2008], o SaaS além de apresentar características de um produto de software, também apresenta características de um serviço de TI, ou seja, possui aspectos de qualidade que devem ser analisados tanto como um produto como um serviço.

Diante desta peculiaridade, não encontramos na literatura nem em normas internacionais, modelos de qualidade que contemplem adequadamente as características de qualidade em um sistema SaaS. A falta desses modelos prejudica a utilização de métodos de avaliação voltados para o modelo SaaS, pois como é abordado no trabalho de Colombo & Guerra [2009]; Guerra et al. [2007]; Herssens et al. [2008], a definição da qualidade pretendida através de modelos de qualidade é um passo fundamental para o sucesso dos métodos de avaliações.

Como contribuição deste trabalho, neste capítulo será proposto um conjunto de critérios e requisitos de qualidade, específico ao contexto SaaS, contemplando de maneira abrangente muitas de suas características, possibilitando ser utilizado como apoio ao método MAQSaaS. Será mostrado nas próximas seções como foi desenvolvida

essa proposta, de maneira a detalhar o escopo dos critérios levantados, destacando suas sub-características e algumas implicações/impactos que este conjunto têm sobre este novo paradigma de software, o SaaS.

4.1 Proposta dos critérios e requisitos de qualidade.

A definição deste conjunto, abrangendo as características de um produto SaaS, foi fundamentada em diversos trabalhos, modelos, normas e *frameworks*, que destacam aspectos importantes relacionados tanto a produtos de softwares como a serviços de TI. A norma ISO/IEC 9126 [1991] e os guias de boas práticas da ITIL v3 [2008] e COBIT (ISACA [2008]), ambos apresentados e analisados no capítulo 2, foram os de maior relevância e apoio para a identificação de tais requisitos.

A norma ISO 9126 proporcionou ao conjunto, características bem definidas em relação a um produto de software, fundamentando aspectos de funcionalidade, confiabilidade, usabilidade, eficiência, manutenibilidade e portabilidade. Por sua vez, a ITIL v3 e o COBIT 4.1 constituíram elementos importantes para a identificação de características voltadas para a melhora do gerenciamento e qualidade dos serviços de TI, de modo a oferecer uma maior qualidade, principalmente no tocante a questões legais, desempenho e de segurança a um produto SaaS.

Do modelo da ITIL v3, os requisitos mais relevantes contendo um maior impacto ao contexto foram os baseados nos livros *Service Design*, *Service Transition* e *Continual Service Improvement*. Ambos os livros se baseiam principalmente no gerenciamento da estratégia de serviços, incluindo melhorias no ciclo de vida, de modo a garantir maior qualidade na entrega e no suporte dos serviços de TI, estando sempre preparados para possíveis verificações de conformidades com normas internacionais.

O COBIT 4.1 com seu processo DS - *Deliver and Support*, proporcionou um complemento aos requisitos levantados nos processos da ITIL, facilitando seu entendimento e aplicações. Seu domínio foca em aspectos de entrega de tecnologia da informação, cobrindo a execução de aplicações dentro dos sistemas de TI e seus resultados. Esse processo também inclui questões ligadas à segurança e treinamento de seus usuários, de modo a garantir maior qualidade e satisfação aos serviços oferecidos.

Em complementação aos estudos efetuados, nos trabalhos de O'Brien et al. [2005]; Cancian [2009]; Dubey & Wagle [2006]; Havelka et al. [1998], foram observados e analisados outros aspectos de qualidade que impactam diretamente os sistemas SaaS, principalmente na qualidade externa e interna desses sistemas. Os estudos complementares

efetuados em outros trabalhos, com a visão e ponto de vista de outros autores são relevantes, pois trazem importantes contribuições e complementação aos critérios de qualidade estudados na ISO 9126, ITIL e COBIT, dando uma maior fundamentação na definição de cada característica de qualidade.

Analisando e incorporando todas as informações dos estudos efetuados, a listagem dos principais critérios e requisitos relacionados ao contexto SaaS, juntamente com as referências dos trabalhos que permitiram tal contribuição podem ser verificados abaixo:

Funcionalidade	Referências
Adequação	ISO/IEC 9126 [1991]; Cancian [2009].
Acurácia	ISO/IEC 9126 [1991].
Interoperabilidade	ISO/IEC 9126 [1991]; Dubey & Wagle [2006].

Tabela 4.1: Requisitos de funcionalidade - contexto SaaS.

Confiabilidade	Referências
Maturidade	ISO/IEC 9126 [1991]; Cancian [2009].
Tolerância a falhas	ISO/IEC 9126 [1991].
Recuperabilidade	ISO/IEC 9126 [1991].

Tabela 4.2: Requisitos de confiabilidade - contexto SaaS.

Usabilidade	Referências
Inteligibilidade	ISO/IEC 9126 [1991].
Apreensibilidade	ISO/IEC 9126 [1991].
Operacionalidade	ISO/IEC 9126 [1991]; Cancian [2009]; Dubey & Wagle [2006]; Havelka et al. [1998].
Atratividade	ISO/IEC 9126 [1991].

Tabela 4.3: Requisitos de usabilidade - contexto SaaS.

Eficiência	Referências
Comportamento em relação ao tempo	ISO/IEC 9126 [1991].
Utilização de recursos	ISO/IEC 9126 [1991]; ITIL v3 [2008]; COBIT (ISACA [2008]).
Robustez	ISO/IEC 9126 [1991].

Tabela 4.4: Requisitos de eficiência - contexto SaaS.

Manutenibilidade	Referências
Analisabilidade	ISO/IEC 9126 [1991].
Personalização	ISO/IEC 9126 [1991]; ITIL v3 [2008]; COBIT (ISACA [2008]).
Estabilidade	ISO/IEC 9126 [1991].
Testabilidade	ISO/IEC 9126 [1991]; ITIL v3 [2008].

Tabela 4.5: Requisitos de manutenibilidade - contexto SaaS.

Portabilidade	Referências
Adaptabilidade	ISO/IEC 9126 [1991]; Dubey & Wagle [2006]; Havelka et al. [1998].
Coexistência	ISO/IEC 9126 [1991]; Cancian [2009].
Instalação	ISO/IEC 9126 [1991];
Capacidade para substituir	ISO/IEC 9126 [1991];

Tabela 4.6: Requisitos de portabilidade - contexto SaaS.

Nível de Serviço	Referências
SLA	ITIL v3 [2008]; COBIT (ISACA [2008]); Cancian [2009].
Continua na próxima página	

Tabela 4.7 – continua na próxima página

Nível de Serviço	Referências
Plano de Continuidade	ITIL v3 [2008]; COBIT (ISACA [2008]); Dubey & Wagle [2006].
Gerenciamento dos dados	ITIL v3 [2008]; COBIT (ISACA [2008]); Cancian [2009]; Dubey & Wagle [2006]; Havelka et al. [1998].
Auditabilidade	COBIT (ISACA [2008]).

Tabela 4.7: Requisitos de nível de serviço - contexto SaaS.

Segurança	Referências
Confiabilidade	ITIL v3 [2008]; COBIT (ISACA [2008]); O'Brien et al. [2005]; Cancian [2009]; Dubey & Wagle [2006].
Integridade	ITIL v3 [2008]; COBIT (ISACA [2008]); Cancian [2009]; Dubey & Wagle [2006].
Disponibilidade	ITIL v3 [2008]; COBIT (ISACA [2008]); Dubey & Wagle [2006].

Tabela 4.8: Requisitos de segurança - contexto SaaS.

Desempenho	Referências
Escalabilidade	ISO/IEC 9126 [1991]; O'Brien et al. [2005].
Expansibilidade	O'Brien et al. [2005]; Dubey & Wagle [2006].
Utilização de recursos	ISO/IEC 9126 [1991]; O'Brien et al. [2005]; Cancian [2009].
Comportamento em relação ao tempo	ISO/IEC 9126 [1991]; Dubey & Wagle [2006].

Tabela 4.9: Requisitos de desempenho - contexto SaaS.

Ao final, o levantamento dos requisitos deste estudo não teve a intenção de mostrar relatórios com diversas características de qualidade, mas mostrar apenas aquelas que tenham uma aplicabilidade e maior relevância para a qualidade dos SaaS. Para dar apoio e maior fundamentação aos requisitos levantados, devido a falta de aplicação prática e real deste conjunto, houve a necessidade de se priorizar e complementar as características levantadas. Este apoio foi feito através de entrevistas entre especialistas da área de sistemas SaaS, tendo como participantes analistas de TI, consultores e gerentes de produtos SaaS.

A pesquisa *online*, aplicada para priorizar e complementar o conjunto de características foi efetuada envolvendo 63 especialistas da área de TI, ligados diretamente a produtos SaaS. Para a aplicação da pesquisa, foi elaborado um *checklist*, onde o especialista poderia analisar e priorizar cada requisito levantado durante o estudo.

O *checklist* possui uma lista de critérios de qualidade decompostos em requisitos, onde cada requisito possui uma definição formal de sua aplicação juntamente com um campo de enumeração, onde o especialista pode definir sua relevância dentro do contexto SaaS. A relevância dos requisitos de qualidade irá indicar quais características são realmente importantes baseando-se no conhecimento e experiência prática dos especialistas. A relevância pôde ser definida baseada em alguns critérios de julgamento:

- **Muito relevante:** características importantes ao sistema, sendo de extrema importância a garantia de sua qualidade, interferindo diretamente na qualidade externa e interna dos produtos SaaS, principalmente em sistemas críticos das organizações;
- **Relevante:** características importantes principalmente na qualidade externa do sistema, mais voltada a aspectos perceptíveis ao usuário, impactando a qualidade final do produto;
- **Pouco relevante:** características desejáveis, mas não essencial para o sucesso e operação da aplicação;
- **Não aplicável:** características que não se aplicam ao contexto SaaS. Existem aspectos de qualidade que são aplicáveis e mensuráveis em softwares convencionais, porém não no contexto SaaS.

Além da indicação de relevância, ao final da pesquisa os especialistas participantes complementaram de forma adicional características de qualidade que porventura não foram listadas na pesquisa. O modelo do *checklist* completo, utilizado e aplicado nas pesquisas, pode ser visualizado no Apêndice B. Todos os detalhes metodológicos da

aplicação da pesquisa, juntamente com o planejamento, escolha das pessoas participantes, meios de aplicação, foram definidos no capítulo 3, seção 3.2.

De acordo com os dados numéricos obtidos pela pesquisa, é possível verificar a relevância de cada requisito dentro do contexto estudado, podendo assim inferir conclusões sobre a sua aplicação e escopo: os dados são listados abaixo, Figura 4.1.

Requisitos	Relevância			
	1 - Não Aplicável	2 - Pouco Relevante	3 - Relevante	4 - Muito Relevante
Adequação	--	6,7 %	83,6 %	9,7 %
Acurácia	--	3%	78,4 %	18,6 %
Interoperabilidade	--	2,6 %	65,6 %	31,8 %
Maturidade	17,5%	71,9 %	5%	5,6 %
Tolerância a Falhas	32,1%	56,8%	7,8%	3,3%
Recuperabilidade	22,8%	67,4%	6,7%	3,1%
Inteligibilidade	--	4,8 %	85,9 %	9,3 %
Apreensibilidade	--	3,4 %	82,6 %	14%
Operacionalidade	--	11,1 %	78,1 %	10,8 %
Atratividade	--	6%	74,3 %	19,7 %
Comportamento em relação ao tempo	15,1 %	76,6 %	5,5 %	2,8 %
Utilização de recursos	22,4%	50,1%	17,5%	10,0%
Robustez	40,7%	33,5%	15,8%	10,0%
Analisabilidade	10,2 %	11,1 %	77,3 %	1,4 %
Personalização	6,7 %	44,7 %	40,8 %	7,8 %
Estabilidade	10,0%	13,4%	42,6%	34%
Testabilidade	--	9%	12,1 %	78,9 %
Adaptabilidade	--	10%	81,6 %	8,4 %
Instalação	97,8 %	2,2 %	--	--
Coexistência	--	7,2 %	77,6 %	15,2 %
Capacidade para substituir	50,6 %	9,1 %	36,6 %	3,7 %
SLA's	--	4,5 %	2,4 %	93,1 %
Plano de continuidade	--	4,6 %	13,3 %	82,1 %
Gerenciamento de dados	--	2,3 %	10,1 %	87,6 %
Auditabilidade	1%	3,2 %	4,5 %	91,3 %
Confiabilidade	--	1,8 %	2,6 %	95,6 %
Integridade	--	0,9 %	5%	94,1 %
Disponibilidade	--	2,2 %	1,1 %	96,7%
Escalabilidade	--	4%	6,8 %	89,2 %
Expansibilidade	2%	1,1 %	5,2 %	93,7 %
Capacidade de infraestrutura	37,9%	26,5 %	15,6 %	20%
Complementação				
Acesso	--	9,5%	20%	70,5 %
Incidentes/Problemas	--	5,3 %	14,2 %	80,5 %

Figura 4.1. Relevância dos requisitos de qualidade relacionado ao paradigma SaaS.

Os resultados obtidos na pesquisa ajudam a dar uma confiança maior ao conjunto proposto, principalmente pela combinação do conhecimento prático dos especialistas com as das fundamentações expostas por diversos autores da área de qualidade.

Analisando os dados obtidos na Figura 4.1, conclui-se que a maioria das características levantadas foram consideradas relevantes, possuindo aplicabilidade direta em relação aos SaaS, já outras, possuindo uma pontuação elevada no critério "Não Aplicável" foram desconsideradas, considerando-se que não são aplicáveis ao contexto desses sistemas.

A pesquisa também permitiu a adição de duas novas características que foram indicadas e priorizadas pelos especialistas, são elas: Acesso e Incidente/Problemas. Antes de incorporar tais requisitos ao conjunto proposto, é importante validá-los através de outros autores e normas, de forma a garantir verificabilidade de aplicação e definição de seu escopo ao paradigma SaaS. A tabela 4.10 mostra a lista dos novos requisitos juntamente com as fundamentações utilizadas:

Requisitos Complementares	Fundamentação
Acesso	ISO/IEC 9126 [1991]; Turner et al. [2003]; Ma [2007].
Incidentes/Problemas	ITIL v3 [2008]; COBIT (ISACA [2008]); Aulbach et al. [2008].

Tabela 4.10: Complementação das características de qualidade.

Ao final de todo o processo, o mapa do conjunto contendo critérios e requisitos de qualidade específicos ao SaaS, pode ser visualizado abaixo pela Figura 4.2.

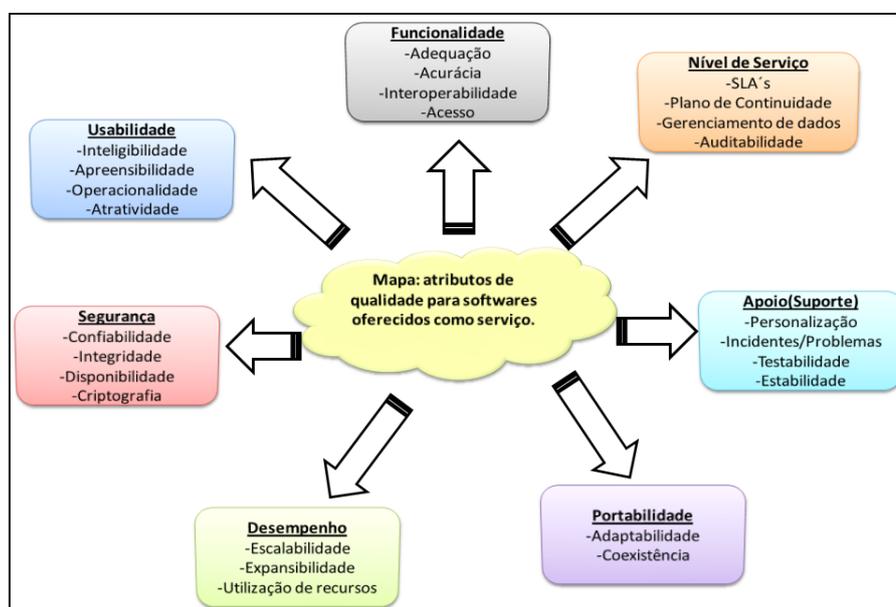


Figura 4.2. Conjunto de critérios e requisitos de qualidade específicos ao SaaS.

Para que as características definidas na Figura 4.2 possam apoiar as avaliações práticas em um método de avaliação, é necessário atribuir um detalhamento de seus

requisitos, de maneira a identificar seu escopo e aplicação, facilitando assim sua utilização, Boehm [1981]; Gatti & Werneck [2004]. Muitas dessas características herdam a aplicabilidade de um produto de software, porém outras necessitam de uma adaptação dentro do escopo de serviços de TI, para que sejam adequadamente aplicáveis a produtos SaaS.

Para facilitar o entendimento do conjunto proposto, melhorando sua estrutura e facilitando sua manutenção, foi utilizada e definida uma estrutura hierárquica semelhante à especificação de avaliação de produto de software, Bertoa & Vallecillo [2002], que se baseiam em critérios, requisitos (diretrizes) e atributos de verificação.

A estrutura hierárquica definida neste trabalho se assemelha à estrutura proposta no trabalho de Hoffman et al. [2004], de modo a facilitar a verificação de requisitos de qualidade que devem ser observados na construção de um produto SaaS. Para o atual trabalho foram utilizados três níveis de hierarquia, tendo cada um, uma finalidade específica. Os três níveis dessa estrutura são apresentados a seguir:

- **Nível 1 - Critérios para a avaliação:** No primeiro nível da hierarquia foram definidos critérios a serem observados durante a avaliação da qualidade em um sistema SaaS, são eles:

- Funcionalidade;
- Usabilidade;
- Segurança;
- Desempenho;
- Portabilidade;
- Apoio(Suporte);
- Nível de Serviço.

- **Nível 2 - Requisitos (Diretrizes):** O segundo nível é formado por requisitos de qualidade. Esses requisitos representam diretrizes específicas para cada critério definido no primeiro nível, por exemplo, o critério "Funcionalidade" pode ser decomposto em requisitos de forma a detalhar e facilitar as avaliações de qualidade. Os Requisitos do segundo nível da hierarquia devem ser priorizados e ter uma definição clara dentro do contexto SaaS, pois a partir deles serão definidos os atributos mensuráveis (terceiro nível), definindo como medir os critérios de qualidade abordados no primeiro nível da hierarquia. Outra importância deste nível, se baseia no fato de que os requisitos levantados representam diretrizes relevantes para o desenvolvimento de produtos SaaS, ou seja, definindo características e funcionalidades importantes para o desenvolvimento desses sistemas.

- **Nível 3-Atributos de Verificação:** O terceiro nível é formado pelos atributos de verificação, ou seja, atributos mensuráveis que possam ser medidos e pontuados. Tais atributos identificam e mostram como avaliar os requisitos de qualidade em relação ao segundo nível da hierarquia. Os atributos de verificação (nível 3) não serão abordados e detalhados neste capítulo, sua descrição se dará adiante nos capítulos 5 e 6, onde será mostrado a lista de verificação (*checklist*) aplicada para avaliar os requisitos de qualidade.

Após apresentar os níveis da hierarquia que serão utilizados neste modelo de qualidade, será feito um detalhamento principalmente nos elementos do primeiro e segundo nível. Para facilitar a visualização da relevância dos requisitos, as prioridades obtidas ao longo da pesquisa com os especialistas (Tabela 4.1) foram simbolizadas pelos símbolos (+), (++) e (+++), de modo a expressar a importância e o impacto que os mesmos têm sobre os sistemas SaaS. A Figura 4.3 mostra a representação gráfica dos símbolos utilizados para priorizar os requisitos.

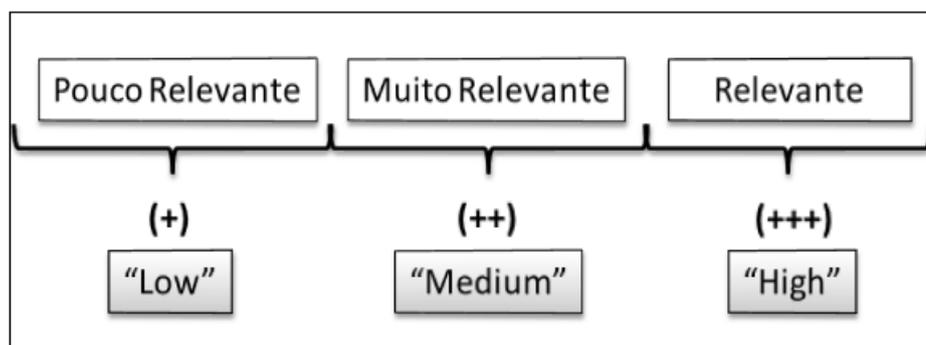


Figura 4.3. Notação da prioridade dos requisitos.

Abaixo será feita a análise e o detalhamento de cada critério de qualidade relacionado a um produto SaaS, juntamente com o detalhamento dos seus requisitos e prioridade referente a hierarquia definida.

Funcionalidade: todo sistema de software, e SaaS não é exceção, precisa prover funções que atendam às necessidades explícitas e implícitas acordadas com o cliente. Para os produtos SaaS podemos destacar os seguintes requisitos:

- Adequação (++) : capacidade que o produto SaaS tem de prover um conjunto apropriado de funções para tarefas e objetivos específicos do usuário;
- Acurácia (++) : quando utilizado em missões críticas da organização, o software precisa ter a capacidade de prover, com grau de precisão necessário, resultados

ou efeitos corretos ou conforme acordados com o cliente. Falhas na acurácia de um software podem ter efeitos diretamente ao negócio da organização;

- **Interoperabilidade (++)**: grandes organizações não utilizam um software ou um sistema apenas, a interoperabilidade é importante principalmente para permitir a interação com um ou mais sistemas específicos da organização;
- **Acesso (+++)**: um pouco diferente dos softwares convencionais, o SaaS permite o acesso de um ou mais clientes ao mesmo sistema, porém cada um com sua seção de trabalho. O SaaS deve prover um acesso eficiente e seguro para que não ocorra de maneira acidental ou maliciosa o acesso de pessoas não autorizadas aos dados dos usuários.

Usabilidade: a interface de um software para o usuário é um dos pontos fortes para aceitação por parte dos clientes. Os requisitos em destaque são listados a seguir:

- **Inteligibilidade (++)**: possibilita ao usuário compreender se o software é apropriado e como ele pode ser usado para tarefas e condições de uso específicos;
- **Apreensibilidade (++)**: o software deve possibilitar ao usuário apreender sua aplicação sozinho ao longo do tempo, evitando que o usuário sempre recorra a manuais ou serviços de suporte;
- **Operacionalidade (++)**: o software deve possibilitar ao usuário operá-lo e controlá-lo de maneira fácil e eficiente. Possibilitando uma interação amigável com o usuário;
- **Atratividade (++)**: envolve características que possibilitam uma satisfação maior por parte dos clientes do sistema, incluindo desde a adequação das informações prestadas para o usuário, até os requintes visuais utilizados em sua interface gráfica.

Segurança: é um dos critérios mais importantes para os SaaS, em virtude de sua aplicabilidade e utilização. Devido ao fato que seu acesso é feito via web (acesso remoto), existem algumas questões que devem ser garantidas para uma maior segurança. Os requisitos levantados dentro do critério segurança podem ser vistos a seguir:

- **Confiabilidade (+++)**: permite que o acesso à informação/serviço só possa ser concedido a usuários autorizados, evitando que o sistema ou informações sejam usados por partes que não estejam vinculados a organização contratante do sistema;

- **Integridade (+++):** No SaaS as informações trafegam pela web tanto do lado do usuário como do lado do servidor. É importante que essas informações estejam íntegras, isto é, não sejam corrompidas ao longo do caminho;
- **Disponibilidade (+++):** capacidade que o SaaS apresenta em manter o seu funcionamento *online* no servidor do fornecedor. Uma das exigências para sua adoção é o acesso a Internet, que precisa ser rápido, estável e redundante. Esses cuidados minimizam a possibilidade de descontinuidade do serviço das aplicações;

Desempenho: pode ter significados diferentes em contextos diferentes, dentro do escopo SaaS: eles estão relacionados ao tempo de resposta (quanto tempo leva para processar um pedido), taxa de transferência (quantos pedidos globais podem ser processados por unidade de tempo), expansibilidade (capacidade de adicionar novos serviços ao sistema) e escalabilidade (capacidade de prover o acesso a diversos usuários ao mesmo tempo), Choudhary [2007]; Chong & Carraro [2008]. Os requisitos que contribuem para um bom desempenho são mostrados a seguir:

- **Escalabilidade (+++):** capacidade com que o software suporta um número específico de usuários na mesma seção, observando que o SaaS possibilita sua utilização por diversos usuários ao mesmo tempo, ou seja, implementa a capacidade do multi-inquilino (*multi-tenant*);
- **Expansibilidade (+++):** dentro do paradigma SaaS, a expansibilidade pode significar a facilidade com que novos serviços podem ser adicionados ou estendidos dentro de um sistema sem afetar os outros serviços e funcionalidades. Essa característica é importante, principalmente pelo motivo de que atualmente o ambiente de negócio de um software vive em constantes mudanças e evolução;
- **Utilização de recursos (+):** capacidade do sistema em utilizar tipos e quantidades apropriados de recursos, principalmente pela largura de banda consumida na aplicação. O tipo e o volume de dados a serem transmitidos por um SaaS dependem da largura de banda da Internet, podendo acarretar atrasos nas transmissões dos dados, prejudicando o negócio da organização contratante;

Portabilidade: apresenta semelhanças com a portabilidade dos produtos de softwares convencionais, porém apresenta algumas diferenças no escopo de sua aplicação. O SaaS não precisa necessariamente estar instalado na máquina local do usuário, sendo acessado através de navegadores web. A portabilidade entre os navegadores e a utilização de diferentes portas de comunicação capacitam esses sistemas para serem

transferidos e executados em diferentes ambientes. A seguir são listados seus principais requisitos:

- Adaptabilidade (++): capacidade do software de ser adaptado para diferentes ambientes especificados, sem necessidade de aplicação de outras ações ou meios além daqueles fornecidos para a finalidade do software;
- Coexistência (++): capacidade do software oferecido como serviço de coexistir com outros sistemas independentes em um ambiente comum, independente da porta de comunicação utilizada, compartilhado diversos recursos computacionais disponíveis;

Apoio(Suporte): pode ser considerado um diferencial de qualidade para as grandes organizações. Um dos motivos que tornam o SaaS mais barato é a sua forma de entrega, um aplicativo que é comercializado de acordo com um modelo de assinatura, ou seja, um serviço. Todo serviço deve ter um suporte eficiente aos seus clientes, de maneira a possibilitar questionamentos, esclarecimento de dúvidas e notificações de eventuais problemas ao sistema. O suporte ao cliente nos SaaS deve ser capaz de proporcionar diversas formas de relacionamento (*online*, telefone, entre outros) e muitas vezes em tempo contínuo (24 horas por dia, dependendo da localização das filiais). Abaixo são listados os principais requisitos do critério suporte:

- Personalização (+): caracteriza a facilidade com que o software pode ser modificado. O SaaS apresenta uma dificuldade em sua personalização, pois um dos aspectos que o torna de baixo custo é ser padrão para todos os assinantes. Esta personalização é muito ligada às funcionalidades do sistema, porém por outro lado os SaaS permitem customizações pré-definidas em sua interface, através de mudanças de temas, disposição dos objetos, entre outros, permitindo uma adequação maior aos usuários pelo menos na parte visual do sistema;
- Incidentes/Problemas (+++): capacidade com que a empresa fornecedora tem em identificar e diagnosticar eventuais problemas, identificando suas causas, podendo ser incidentes, problemas ou falhas;
- Testabilidade (+++): Representa a capacidade de se poder testar o sistema modificado, tanto as novas funcionalidades do sistema como as não afetadas diretamente pela modificação;
- Estabilidade (++): avalia a capacidade do software de evitar efeitos colaterais decorrentes de modificações introduzidas no sistema.

Nível de serviço: é um critério essencial aos SaaS: representa uma característica extremamente importante dentre os critérios de qualidade levantados até o momento. A transferência de dados críticos em processos de negócio de uma organização pode gerar riscos de perda de informações ou de exposição de informações sigilosas. É preciso estabelecer níveis de serviços possibilitando maior segurança na utilização das aplicações. Os requisitos atribuídos a este critério estão mostrados abaixo:

- SLA's (+++): acordos de Níveis de Serviços permitem manter um bom relacionamento entre cliente e o fornecedor, melhorando a especificação e o entendimento dos requisitos do sistema. Permite também um balanceamento eficiente sobre a demanda do cliente e o custo da aquisição do software, juntamente com a melhora do desempenho e disponibilidade oferecidos pelo provedor;
- Plano de Continuidade (+++): estratégias e procedimentos permitindo planos de recuperação gradual, recuperação intermediária e recuperação imediata em sistemas SaaS, ITIL v3 [2008]. Os planos de recuperação devem ser testados regularmente para garantir maior confiabilidade;
- Gerenciamento dos dados (+++): estratégias e procedimentos para migração de dados e serviços de relatórios, pois em qualquer momento, a empresa contratante do sistema SaaS pode migrar de um aplicativo para outro, necessitando a transferência dos dados de uma aplicação para a outra. A emissão de relatórios precisos dos dados armazenados é muito importante, considerando-se que a organização contratante não tem controle direto sobre seus dados;
- Auditabilidade (+++): fator de qualidade que representa o grau em que um sistema/software mantém registros suficientes para suportar auditorias financeiras ou jurídicas. A necessidade de se cumprir normas e regulamentos são cada vez maiores nas organizações. Devido à grande flexibilidade oferecida nos SaaS, as auditorias acabam se tornando processos difíceis, devido ao fato que muitos serviços utilizados em seus softwares são externos e esses não possuem uma documentação ou auditoria efetuada.

Para complementar o conjunto de características de qualidade proposto neste capítulo, a seguir, serão mostrados os impactos tanto negativos como positivos, que os critérios de qualidade levantados têm sobre os produtos SaaS.

4.2 Relevância e impacto dos critérios de qualidade

A escolha de se utilizar uma abordagem SaaS no desenvolvimento de sistemas depende de vários fatores, como por exemplo, pessoal qualificado, oportunidades externas, investimentos, interesse de *stakeholders*, entre outros. Porém, independente do tipo de fator que estimule a adoção do paradigma SaaS, algumas questões devem ser respondidas e mitigadas, como por exemplo:

- Que efeito a escolha de um sistema SaaS terá sobre o alinhamento estratégico da organização?
- Quais critérios de qualidade serão impactados positivamente e negativamente pelo uso do SaaS?
- Quais modificações devem ser feitas nos requisitos de qualidade durante o desenvolvimento de um sistema SaaS?

O que se nota é que o impacto e as mudanças neste novo paradigma devem ser avaliados e estudados cuidadosamente pelas organizações adotantes, Laplante et al. [2008] e Chong & Carraro [2008]. Os sistemas a serem desenvolvidos, precisam atender ao mínimo de critérios de qualidade, de modo a atingir os objetivos de negócios de uma organização. A seguir é apresentado um resumo dos principais impactos que as características propostas para o método MAQSaaS têm sobre os sistemas SaaS.

Funcionalidade: através da utilização de arquiteturas e tecnologias, um sistema SaaS proporciona uma alta interoperabilidade entre sistemas, juntamente com uma eficiente adequação de suas funcionalidades, permitindo que os serviços e aplicações construídas em diferentes linguagens possam ser implantadas em diferentes plataformas para interagir entre si;

Usabilidade: a boa qualidade em usabilidade de um sistema pode diminuir os serviços dentro do suporte da organização, permitindo uma boa interação humana com o sistema, evitando problemas de desempenho com os serviços. Cabe aos prestadores de sistemas SaaS incorporarem usabilidade em seus sistemas, semelhante a usabilidade em softwares convencionais;

Segurança: a necessidade de criptografia, integridade e confiança dentro de uma abordagem SaaS requer uma atenção detalhada dentro de sua arquitetura. Muitos padrões da arquitetura SOA estão sendo desenvolvidos para dar suporte à segurança, mas a maioria ainda estão imaturos, Ma [2007]. Se estas questões não forem tratadas adequadamente dentro do SaaS, a segurança poderá ser afetada negativamente. Entretanto, temos que ter em mente que os dados que estão circulando pelo sistema são

críticos ao negócio da organização contratante, ou seja, o problema com a segurança é um fator crucial para qualidade nesses sistemas;

Desempenho: uma abordagem SaaS pode ter um impacto negativo sobre o desempenho de uma aplicação, devido a atrasos na rede e a sobrecarga de usuários em um mesmo sistema. A organização contratante, deve projetar e avaliar a quantidade de usuários que irão utilizar o sistema; e a organização fornecedora deve projetar e avaliar cuidadosamente os seus serviços/sistemas certificando-se de que os requisitos de desempenho necessários sejam cumpridos. Existem várias maneiras de lidar com a escalabilidade em sistemas SaaS, permitindo um número maior de usuários na mesma seção de um sistema, Ma [2007]; Choudhary [2007]. No entanto, essas soluções exigem uma análise detalhada dos prestadores de serviços para se certificar de que os requisitos de qualidade não serão afetados negativamente;

Apoio(Suporte): a capacidade de suporte no SaaS pode proporcionar um efeito positivo ou negativo para a empresa contratante e fornecedora. A testabilidade pode ser afetada negativamente quando se usa um conjunto muito grande de serviços terceirizados. Muitos desses serviços que constituem o sistema podem ser prestados por organizações externas, onde o acesso ao código fonte não está disponível, dificultando a identificação de eventuais *bugs* ou problemas internos. Sendo assim, uma boa interface/contrato de serviço deve ser concebida com cuidado, pois as mudanças do sistema podem impactar negativamente o usuário, dificultando identificar os problemas ou incidentes gerados por tal modificação;

Nível de Serviço: os contratantes de sistemas SaaS podem negociar um SLA com seu fornecedor, proporcionando acordos de níveis de disponibilidade e sanções por descumprimento do acordo. A auditabilidade pode ser afetada negativamente se as capacidades necessárias para auditoria não forem incorporadas ao sistema. Um prestador de soluções SaaS também deve possuir planos de contingências de suas aplicações e possuir um eficiente gerenciamento de dados, evitando a indisponibilização do sistema e problemas relacionados com migrações ou *backups* de dados das organizações contratantes;

Portabilidade: o uso de uma abordagem SaaS, deve ter um impacto positivo sobre a adaptabilidade e portabilidade quando as adaptações forem geridas de forma adequada, observando-se que a preocupação maior associada à portabilidade é sobre os navegadores web, não a sistemas operacionais ou à distribuição. Este requisito deve ser gerido em coordenação com outros requisitos de qualidade, incluindo a estabilidade, desempenho e disponibilidade, possibilitando um sistema adaptável e portátil.

4.3 Considerações Finais

Em resumo, podemos identificar relações entre os objetivos de negócios de uma organização desenvolvedora de sistemas SaaS, com os requisitos de qualidade de um sistema convencional. Por exemplo, se um dos objetivos de negócio da empresa é ter rapidez e se tornar líder de mercado, isso pode significar que a adaptabilidade, a interoperabilidade, a escalabilidade e a expansibilidade sejam um diferencial. A rapidez vai exigir que os sistemas tenham capacidades de ser adaptados e reutilizados em novas aplicações, combinando maneiras novas e dimensionadas para atender a demanda crescente de usuários em uma única aplicação.

Escolher um sistema e um fornecedor de soluções SaaS é uma decisão importante para a organização. Essa decisão se torna ainda mais importante se a organização em questão não está requisitando somente um sistema, mas vários sistemas de diversos fornecedores com tecnologias diferentes. Nesse contexto a interoperabilidade e considerações de segurança se tornam atributos essenciais para o sucesso da organização.

Desempenho, confiabilidade e portabilidade podem ser critérios fundamentais para a organização que está desenvolvendo produtos SaaS, em torno de seus sistemas legados. Quaisquer que sejam os objetivos de negócios é importante compreender os critérios e requisitos de qualidade para se destacar no mercado competitivo.

Capítulo 5

Apresentação do MAQSaaS

O presente capítulo apresenta uma visão geral da estrutura do MAQSaaS. Serão abordados e discutidos neste capítulo os principais atores e as etapas do processo de avaliação do método proposto.

5.1 Visão geral do MAQSaaS

O MAQSaaS foi proposto e modelado seguindo a lógica de outros métodos de avaliação, exemplificados nos trabalhos de Colombo & Guerra [2009]; Guerra et al. [2007]; Rocha [1983]; MEDE-Pros [1996]. Todos os trabalhos referenciados seguem uma estrutura genérica, se baseando de modo geral em um modelo de qualidade específico, juntamente com um processo de avaliação bem estruturado.

A Figura 5.1 mostra a visão geral do MAQSaaS. Pode ser observado que o método proposto se baseia em um modelo de qualidade específico, proporcionando a definição da qualidade pretendida dentro de um contexto de aplicação, neste caso o SaaS.

Para se basear em um modelo de qualidade, o método deve seguir um processo de avaliação estruturado, possibilitando a aplicação prática das características de qualidade em um sistema SaaS. Por sua vez, o processo de avaliação como mostrado na Figura 5.1, utiliza determinados artefatos que serão produzidos e consumidos nas atividades do processo.

A utilização e incorporação de atores ao método, dentro de seu escopo de aplicação são importantes para estruturar e melhor definir o processo de avaliação. Serão os atores que desempenharão papéis em relação às suas atividades pré-definidas ao longo do processo. Os atores e seus respectivos papéis, juntamente com a descrição das etapas presentes no MAQSaaS serão mostrados abaixo, nas seções 5.2 e 5.3.

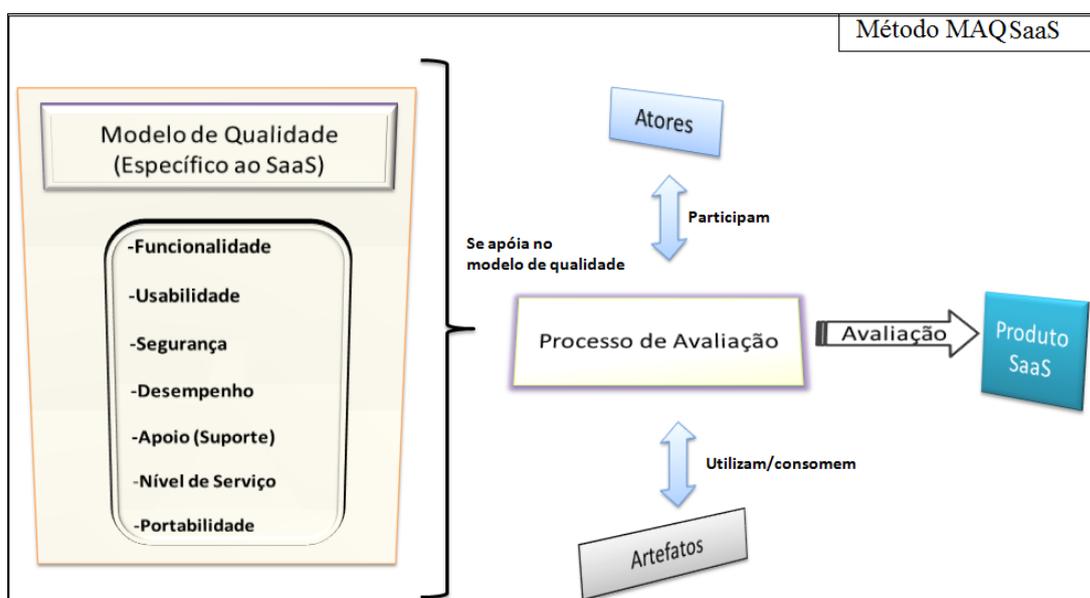


Figura 5.1. Visão geral do MAQSaaS.

De acordo com Boegh et al. [1999], um processo de avaliação pode ser considerado como um dos aspectos mais relevantes e importantes em um método de avaliação. A definição de um processo é necessária para que o método se torne estruturado e aplicável na prática. Diante deste motivo, será apresentado neste capítulo um detalhamento do processo definido no MAQSaaS, tendo como preocupação a sua aplicabilidade durante as avaliações práticas.

De modo a contextualizar e fundamentar, um processo de avaliação é compreendido pela execução de um conjunto de atividades, que visam determinados resultados de acordo com o contexto de aplicação, Colombo & Guerra [2009]; Boegh et al. [1999]. Tais atividades são baseadas em informações fornecidas pelos participantes do processo, de modo a formar um sistema encadeado de atividades.

Para que a subjetividade da avaliação seja mínima, garantindo uma maior eficiência, é aconselhável que o processo a ser seguido possa ser **Repetível**, **Reproduzível**, **Imparcial** e **Objetivo**, Lawlis et al. [2001]. Essas características podem ser definidas da seguinte forma:

- **Repetível:** um processo repetível permite que as avaliações possam ser repetidas várias vezes, produzindo resultados que possam ser aceitos como idênticos ou semelhantes nas demais avaliações;
- **Reproduzível:** o processo deve conter especificações, passos e etapas bem definidas, permitindo ser realizada por diferentes avaliadores e mesmo assim pos-

suir resultados aceitos ao contexto da avaliação;

- **Imparcial:** o processo deve permitir que as avaliações não sejam influenciadas em decorrência de outros resultados e fatores internos ou externos;
- **Objetivo:** o processo deve garantir às avaliações resultados claros, não possuindo ambigüidades e impossibilitando diferentes interpretações dos dados finais.

O processo de avaliação e suas características foram fundamentados se baseando nas melhores práticas da ISO/IEC 14598 [1997]. Algumas modificações foram adicionadas com o objetivo de adequar o processo ao contexto do paradigma SaaS. Nas próximas seções serão mostrados os principais participantes e um detalhamento de cada etapa do processo de avaliação.

5.2 Atores e papéis do processo de avaliação

Além da ISO/IEC 14598 [1997], outras pesquisas semelhantes ao presente trabalho como Colombo & Guerra [2009]; Guerra et al. [2007]; Rocha [1983]; MEDE-Pros [1996] ajudaram a definir e fundamentar os principais atores do processo de avaliação.

Para fins de definição, de acordo com Sommerville [2003] na Engenharia de Software atores podem ser considerados humanos ou entidade máquina que interagem com sistemas ou processos de modo a executar um trabalho ou uma tarefa.

No contexto deste trabalho, tipicamente, um ator representa um papel que um ser humano desempenha ao interagir com o método de avaliação. Basicamente são quatro os atores que participam neste processo: o **Requisitante**, o **Avaliador**, o **Usuário Entrevistado** e o **Especialista**.

O papel básico do **Requisitante** é de solicitar a avaliação de um determinado produto SaaS. Este ator pode ser, por exemplo, uma organização que planeje desenvolver um novo sistema SaaS ou melhorar um produto existente; ou uma entidade que pretende adquirir ou reutilizar um produto SaaS. Ao final da avaliação o requisitante terá em mãos uma visão geral da qualidade do produto avaliado, sob o ponto de vista do paradigma SaaS.

O **Avaliador** tem o papel de gerenciar o processo de avaliação, servindo como guia e contato principal do entrevistado e o processo de avaliação. Ele domina todas as diretrizes da aplicação prática do processo, podendo realizar ajustes no processo, visando uma maior eficiência nas avaliações. É relevante que este ator possua conhecimentos ao produto avaliado e ao domínio de aplicação específico ao qual o software se relaciona, neste caso o SaaS.

O **Usuário Entrevistado** por sua vez, tem por finalidade expressar opiniões positivas e negativas sobre o produto avaliado. As opiniões devem ser fundamentadas dentro do contexto da avaliação, seguindo as informações e critérios definidos pelo avaliador. As opiniões e fatos identificados pelos entrevistados que não se adaptem ao contexto da avaliação e não reproduzam fielmente a realidade da avaliação, devem ser analisadas cuidadosamente, podendo ser descartadas pelo avaliador. Pode ser interessante considerar especializações deste ator, no sentido que podem existir diversos perfis de Usuários Entrevistados, não somente o usuário final, mas também um gerente funcional, um administrador, um analista, entre outros. No entanto, por simplificação, mas sem perda de generalidade, consideraremos somente o ator Usuário Entrevistado.

Outro papel de extrema importância dentro do processo de avaliação é o do **Especialista**, que atua diretamente nas fases de definição e planejamento das avaliações. O Especialista é um indivíduo que possui conhecimento do negócio onde o software encontra-se inserido e experiências na realização de atividades ligadas ao domínio da aplicação, conhecendo normas e os padrões que atuam neste domínio. A criação e revisão dos artefatos utilizados durante o processo de avaliação de qualidade, como por exemplo, lista de verificação (*checklist*), relatórios, entre outros, são em grande parte responsabilidade do especialista.

De modo geral estes papéis podem ser executados por diferentes pessoas ou, em alguns casos, os papéis dos atores Especialista e Avaliador podem ser representados pela mesma pessoa, dependendo da complexidade e tempo demandado nas avaliações. No atual trabalho, o próprio autor desempenhou os papéis dos atores Avaliador e Especialista durante as avaliações.

5.3 Detalhamento das etapas do processo de avaliação

Nesta seção são abordados os conceitos e características básicas de cada uma das etapas do processo de avaliação do MAQSaaS, dando um visão geral dos objetivos de cada etapa.

A descrição prática de cada etapa do processo será mostrada no Capítulo 6 (seis), no qual traz um estudo de caso real, mostrando na prática a aplicação e um detalhamento aprofundado de todas as atividades do processo.

As etapas definidas para o processo de avaliação do MAQSaaS podem ser visualizadas de forma resumida na Figura 5.2.

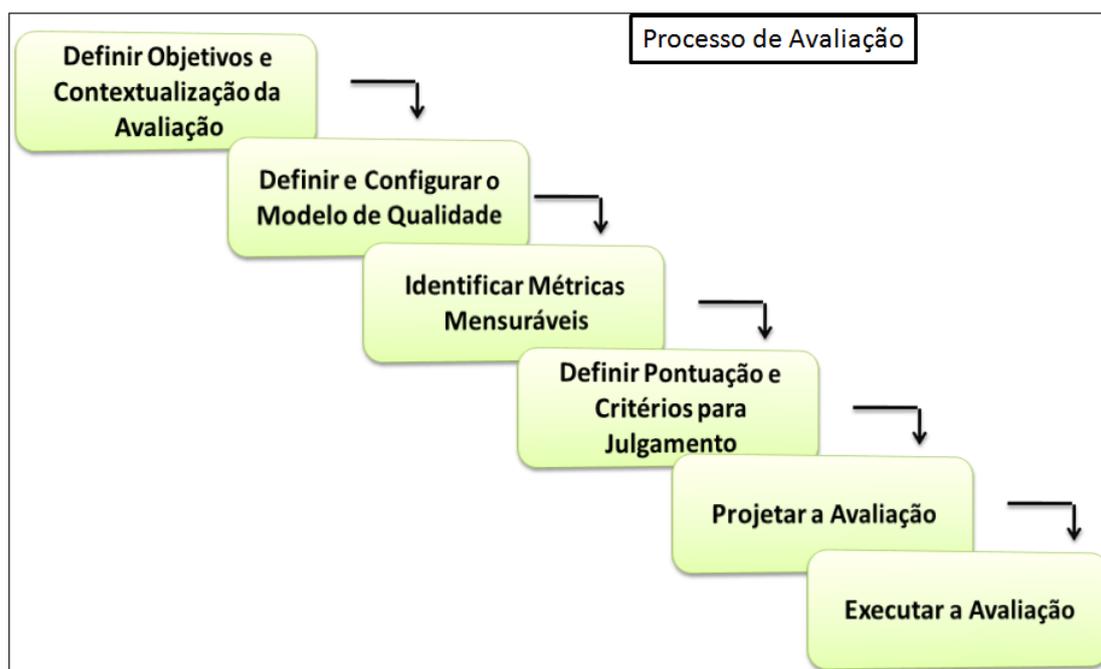


Figura 5.2. Etapas do processo de avaliação do MAQSaaS.

As etapas descritas a seguir são específicas do MAQSaaS, sendo que, foram baseadas nas etapas e nos conhecimentos prescritos do processo de avaliação da norma ISO/IEC 14598 [1997], porém apresentam uma simplificação e uma definição própria voltados para sistemas SaaS.

1-Definir Objetivos e Contextualização da Avaliação: o objetivo desta etapa é deixar claro o intuito da avaliação que será realizada. O papel do Requisitante, com a assessoria do Avaliador, é muito importante nessa fase, pois é ele quem irá especificar o software a ser avaliado juntamente com a versão e outros requisitos necessários.

Nesta etapa são apresentadas questões para a identificação do produto a ser avaliado, como por exemplo, documentos que o produto oferece, requisitos de hardware e software necessários para a execução do produto. Para isso é sugerido que a equipe de avaliação, principalmente o Requisitante, identifiquem o software e seus requisitos mais importantes, como por exemplo, em qual categoria o sistema SaaS se enquadra, quais são os principais usuários do sistema, nível de conhecimento exigido para sua utilização, necessidade de recursos computacionais, entre outros. Em alguns casos o Requisitante não possui o conhecimento adequado para o levantamento de todas estas informações, por isso, a assessoria do Avaliador nesta etapa se torna essencial.

Uma boa avaliação tem início principalmente nesta etapa, onde é relevante deixar claro os objetivos, evitando assim ambigüidades e não-conformidades durante

as avaliações. Um exemplo de como deve ser feita a definição dos objetivos juntamente com questionamentos a serem realizados podem ser verificados na subseção 6.1.2 do Capítulo 6.

2-Definir e Configurar o Modelo de Qualidade: nesta fase deve-se definir qual o modelo de qualidade será utilizado no método de avaliação, pois os métodos devem conter ou se basear em um modelo de qualidade bem definido (conjunto de características que definem o que medir em um sistema computacional), Lew et al. [2008].

É importante que este modelo seja estruturado em grupos de características e sub-características de qualidade, pois tal estruturação permite e facilita o mapeamento de características para atributos mensuráveis, possibilitando assim sua pontuação e mensuração ao final das avaliações. É essencial que este modelo seja específico para o paradigma SaaS, contemplando características relevantes a um produto de software oferecido como serviço.

O modelo de qualidade utilizado no MAQSaaS é referenciado no Capítulo 4, modelo no qual contempla inúmeras características específicas aos SaaS, levando em conta 7 (sete) critérios de avaliação, são eles:

- Funcionalidade;
- Usabilidade;
- Segurança;
- Desempenho;
- Portabilidade;
- Apoio(Suporte);
- Nível de Serviço.

Após a definição do modelo de qualidade, nesta etapa deve ser realizada a configuração do mesmo, especificando quais critérios de qualidade serão utilizados. O modelo deve ser personalizado a partir dos objetivos e de toda a contextualização da avaliação definida na etapa anterior. Por exemplo, dependendo do produto a ser avaliado, o critério "Funcionalidade" e "Segurança", ou qualquer outros, pode(m) ser mais relevantes.

O modelo de qualidade utilizado foi fundamentado em revisões bibliográficas, complementado e priorizado através de pesquisas e contribuições de especialistas na área de produtos SaaS. Para dar uma maior confiabilidade durante as aplicações práticas, foi elaborado um artigo com o detalhamento do modelo junto a suas características,

sendo aprovado e aceito no CISTI'2010 (*5ª Conferência Ibérica de Sistemas e Tecnologias de Informação*), Duarte Filho et al. [2010], de modo a obter críticas positivas e opiniões da comunidade científica, especificamente da área de qualidade de software.

Esta etapa de definição e configuração do modelo de qualidade pode ser considerada uma das mais importantes de todo o processo, pois é nela que será definido "o que" será medido em um produto SaaS.

3-Identificar Métricas Mensuráveis: esta etapa se baseia no mapeamento das características do modelo de qualidade em atributos que possam ser medidos e pontuados. Os atributos levantados devem ser completos e abrangentes, proporcionando um significado completo do requisito a ser avaliado, evitando ambigüidades. Ao final da avaliação, cada atributo avaliado nos dará a soma total de sua qualidade através de escalas de pontuação, definidas na etapa 4, "Definir pontuação e critérios de julgamento".

Existem várias técnicas que auxiliam e apóiam este tipo de mapeamento. A técnica utilizada no MAQSaaS é a utilização de "lista de verificação", que semelhante a um *checklist*, proporciona ao avaliador um conjunto de questões simples e objetivas, no qual através de uma escala se atribui um valor ou uma pontuação para cada resposta das questões, Host & Runeson [2007]. O motivo da escolha de lista de verificação se deve ao fato de ser uma técnica de fácil utilização e ter uma eficiência comprovada em outros trabalhos semelhantes, Colombo & Guerra [2009]; Host & Runeson [2007]; Herrmann & Paech [2008]; Goh et al. [2006]; Wagner et al. [2008].

É importante que cada requisito do modelo de qualidade possa ser desdobrado em diversos atributos mensuráveis, formando questões para a aplicação da lista de verificação. Porém, o número de itens ou questões a serem levantados e especificados no *checklist* depende da complexidade e do grau de subjetividade da avaliação, assim, o número de itens varia de acordo com cada requisito avaliado ou mesmo do objetivo da avaliação.

O papel do Especialista é importante nesta etapa, pois é ele quem irá identificar os atributos mensuráveis relacionados com os critérios e aos objetivos da avaliação. A construção da lista de verificação deve ser feita de maneira cuidadosa, onde todas as questões devem ser relevantes ao contexto SaaS e ao mesmo tempo serem fundamentadas garantindo-as uma maior confiabilidade nas avaliações.

Na subseção 6.1.4 será mostrado um exemplo de como foi feito o levantamento dos atributos mensuráveis, juntamente com a construção do *checklist* de verificação.

4-Definir Pontuação e Critérios para Julgamento: após estabelecer os atri-

butos mensuráveis, nesta etapa são estabelecidos os níveis de pontuação e critérios para o julgamento dos dados resultantes das avaliações. Muitas vezes as pontuações obtidas nos resultados da lista de verificação são do tipo "nominais", representando somente uma nomenclatura, e para que seja possível sua pontuação, é necessário fazer um mapeamento dessas respostas para valores numéricos, sendo de extrema importância a utilização de escalas com intervalos bem definidos de pontuação.

Muitos autores afirmam que não existe um critério de pontuação aceito de maneira geral para avaliação da qualidade em sistemas computacionais, e as que existem não são totalmente maduras e fundamentadas, portanto as métricas definidas para a pontuação exigem um bom senso e experiência do Avaliador e do Especialista durante as avaliações, principalmente no contexto das aplicações a serem avaliadas.

Outro ponto importante é que, quando se deseja fazer uma avaliação mais aprofundada de uma característica de qualidade, como, por exemplo, específica a aspectos de segurança, desempenho, suporte, entre outros, convém identificar pesos e medidas diferentes para as outras características, de maneira a atribuir um grau de relevância maior aos requisitos específico da avaliação.

O Avaliador deve considerar que as questões contidas na lista de verificação são proposições lógicas sobre um atributo a ser avaliado. Cada proposição deverá ter uma pontuação bem definida permitindo se enquadrar corretamente aos critérios de julgamento.

No MAQSaaS à pontuação dada para cada questão tem o valor máximo definido como "1"(maior pontuação) e o menor de "0"(menor pontuação), no qual sua granularidade é dada por décimos de valores. Conforme prescrito pela lista de verificação, são definidos dois grupos de questões, as questões objetivas onde a resposta nominal é sempre binária (apresentando duas opções de respostas), e as questões abrangentes, possibilitando uma opinião mais clara do Usuário Entrevistado (contendo mais de duas opções de respostas), Colombo & Guerra [2009]. Os dois tipos de questões serão detalhados abaixo.

Para as questões binárias, são assumidas as seguintes respostas.

- "S"(Sim) para proposições verdadeiras - representando valor 1;
- "N"(Não) para proposições falsas - representando valor 0;

Para as questões mais abrangentes, são assumidas as seguintes respostas.

- "Eficiente" para proposições que cumprem de maneira rápida os objetivos - representando valor 1;

- "Regular" para proposições que cumprem os objetivos, mas que poderiam apresentar melhorias - representando valor 0,75;
- "Deficiente" para proposições que cumprem os objetivos, mas apresentam alguns erros em sua execução - representando valor 0,50;
- "Ineficiente" para proposições que não que não cumprem de maneira eficiente os objetivos pretendidos - representando valor 0,25;

Para complementar e facilitar a avaliação final, todas as questões podem receber respostas "NA"(Não se aplica) ou "AP"(Avaliação prejudicada). Todas as questões que receberem estas qualificações deverão ser justificadas facilitando assim a elaboração do relatório final da avaliação.

- "NA"(Não se Aplica) para proposições que fazem referência a um aspecto que não se enquadra no produto avaliado. O avaliador deve verificar se a proposição se aplica ou não ao produto de software. Exemplo: aspectos ligados a "instalação" de um software não são aplicáveis aos SaaS, devido ao seu modo de acesso via WEB;

- "AP"(Avaliação Prejudicada) para proposições em que o entrevistado não está em condições de avaliar. Como por exemplo, falta de recursos, falta de informações, ou mesmo falta de conhecimento específico. A atribuição dessa alternativa é pouco desejável, pois significa que os recursos de que se dispõe são insuficientes. Exemplo: se a avaliação estiver sendo executada com um equipamento cuja resolução gráfica é diferente da exigida, então o avaliador deverá optar pela alternativa "AP"(Avaliação Prejudicada).

Veremos na subseção 6.1.5 como que podemos através desses critérios definidos obter um julgamento do nível de qualidade a ser atingido pelo software avaliado.

5-Projetar a Avaliação: para esta etapa deve ser definido o plano de avaliação a ser seguido durante as avaliações práticas. O plano em destaque deve listar as diretrizes, os procedimentos, ou seja, mostrar de modo geral como devem ser realizadas as avaliações. Esta fase deve ser minuciosa e ao mesmo tempo objetiva. É nesta fase que será informado qual técnica de avaliação será utilizada, devendo conter informações claras ao Avaliador, como instruções, recursos necessários, artefatos, juntamente com um cronograma pré-definido.

No plano de avaliação alguns pontos devem ser destacados de modo a facilitar a projeção e a documentação das avaliações, abaixo segue uma lista não taxativa desses elementos:

- Identificar o conhecimento básico que os Usuários Entrevistados devem possuir para permitir a sua participação nas avaliações;
- Listar todo o tipo de material necessário e utilizado para a aplicação das avaliações, como por exemplo, computador *desktop*, acesso a internet, lista de verificação impressa, documentação do software avaliado, entre outros;
- O intervalo de tempo que foi estipulado para a realização das avaliações, evitando que o Avaliador perca o controle do tempo, fazendo com que as avaliações se tornem longas e exaustivas;
- Localização física para a aplicação das avaliações, exemplo: prédios, laboratórios, instituições de ensino, cidade, entre outros;
- O número de avaliações a serem realizadas, justificando o número de repetições que foram executadas;
- Modo no qual o resultado final da avaliação será apresentado, através de gráficos, relatórios ou até mesmo apresentação oral para o Requisitante da avaliação.

É relevante que o Avaliador seja treinado no método de avaliação antes de sua aplicação prática. Outro aspecto relevante, é que o mesmo conheça o modelo de qualidade utilizado (neste caso contemplando características de qualidade para produtos SaaS), tendo como características importantes ser uma pessoa disciplinada e organizada, garantindo maior organização no andamento das avaliações.

Nesta etapa o Avaliador deverá ter um computador exclusivo para executar a avaliação do produto, com os requisitos de hardware e software necessários para tal execução. Além desse recurso, ele deverá ter um cronograma para fazer a avaliação; de preferência, de maneira contínua e conclusiva, do início ao fim, para não perder o foco das avaliações.

6-Executar a Avaliação: nesta última etapa do processo de avaliação, o Avaliador deve utilizar tudo o que foi apresentado nas etapas anteriores, garantindo avaliações eficientes de acordo com os propósitos selecionados. Esta etapa compõe-se das seguintes tarefas: Obtenção de Medidas; Comparação de Critérios; e Julgamento dos Resultados.

Obtenção de Medidas: no processo de avaliação proposto, a etapa de obtenção de medidas faz referência a aplicação de entrevistas estruturadas, que contenham questões alusivas às características e sub-características de produtos SaaS. As respostas obtidas

devem ser sintetizadas, proporcionando os primeiros dados da fase de obtenção das medidas.

Comparação de Critérios: a etapa de comparação de critérios é necessária para verificar se os resultados obtidos ao longo da avaliação estão de acordo com os critérios de avaliação estabelecidos pelo Avaliador e o Especialista. A fase de comparação dos resultados obtidos com os critérios estabelecidos é significativa, principalmente para dar mais rapidez e segurança ao julgamento final dos resultados.

Julgamento dos Resultados: ao final, com os dados em mãos, é possível realizar o julgamento dos resultados. Julgar significa revelar o nível de satisfação do produto, de acordo com os níveis de pontuação e critérios para julgamento, definidos nas etapas anteriores.

Esta etapa poderá ser melhor exemplificada nas subseções 6.1.7 e 6.1.8 onde será abordado os resultados do estudo de caso realizado, utilizando o MAQSaaS como método de avaliação.

5.4 Considerações Finais

O capítulo apresentou de modo geral todas as etapas que constituem um conjunto de atividades encadeadas, permitindo verificar a qualidade em um produto SaaS. Tais atividades incorporadas dentro do MAQSaaS mostram que todos os aspectos relatados são relevantes e importantes, devendo ser tratados e detalhados, de modo a garantir uma efetividade maior durante as avaliações.

Algumas destas etapas podem sofrer pequenas modificações em virtude de avaliações mais específicas. Tais modificações ficam a cargo do Avaliador e do Especialista, pessoas no qual, detêm conhecimento sobre o processo e sua aplicação. No próximo capítulo será apresentado uma aplicação prática de todas estas etapas do processo, juntamente com um detalhamento da técnica de lista de verificação e do modelo de qualidade empregado dentro do MAQSaaS.

Capítulo 6

Validação e Avaliação do MAQSaaS

Este capítulo tem por objetivo avaliar de forma fundamentada o MAQSaaS juntamente com seu modelo de qualidade. Para tais tarefas foi realizado um estudo de caso com a aplicação do método ao produto SaaS "Salesforce CRM", possibilitando obter dados e resultados dentro de um ambiente real, de maneira a definir seus limites mais claramente. Para complementar a validação do trabalho, um comparativo foi realizado entre o MAQSaaS e um método convencional de avaliação de produto de software, com o objetivo de verificar limites e demais contribuições.

6.1 Estudo de caso - Produto Salesforce CRM

6.1.1 Apresentação do produto SaaS

A terminologia CRM (*Customer Relationship Management*), traduzindo para o português "Gestão de Relacionamento com o Cliente", pode ser definida como um sistema que tem por objetivo principal definir uma classe de ferramentas que automatizam as funções de contato com os clientes. Essas ferramentas compreendem sistemas informatizados e, fundamentalmente, uma mudança de atitude corporativa, que objetiva ajudar as companhias a criar e manter um bom relacionamento com seus clientes, Peppard [2000].

O Salesforce CRM pode ser considerado como um produto SaaS, pois, além de possuir diversos aspectos de um SaaS, sua comercialização e distribuição é feita sob a forma de serviços, sendo que o acesso ao sistema é feito integralmente via web.

O produto Salesforce CRM é composto por 4 (quatro) módulos de sistemas, sendo o "Salesforce CRM Sales", o "Salesforce CRM Marketing", o "Salesforce CRM Service" e o "Salesforce CRM Partners". Essas ferramentas auxiliam e proporcionam

uma maior automatização da força de vendas da organização, uma assistência e suporte a clientes, automatização de campanhas de marketing, aplicações à medida do seu negócio, entre outros benefícios.

O estudo de caso é baseado somente no módulo "Salesforce CRM Sales". Os outros módulos apresentam características semelhantes entre si, porém, a justificativa de se escolher o "Salesforce CRM Sales" foi o fato deste ser um módulo mais amplo e completo, apresentando um maior número de funcionalidades, possibilitando avaliações minuciosas e precisas no produto avaliado. De modo geral o objetivo deste módulo é permitir que as empresas gerenciem pessoas e processos de maneira mais eficaz, possibilitando aos representantes concluir negócios de forma eficiente com os clientes.

A Figura 6.1 mostra a tela principal do "Salesforce CRM Sales", onde se pode realizar o gerenciamento de contas dos clientes, visualizando informações de vendas, mensagens, advertências, gráficos de vendas e possibilidade de criação e emissão de diversos tipos de relatórios.

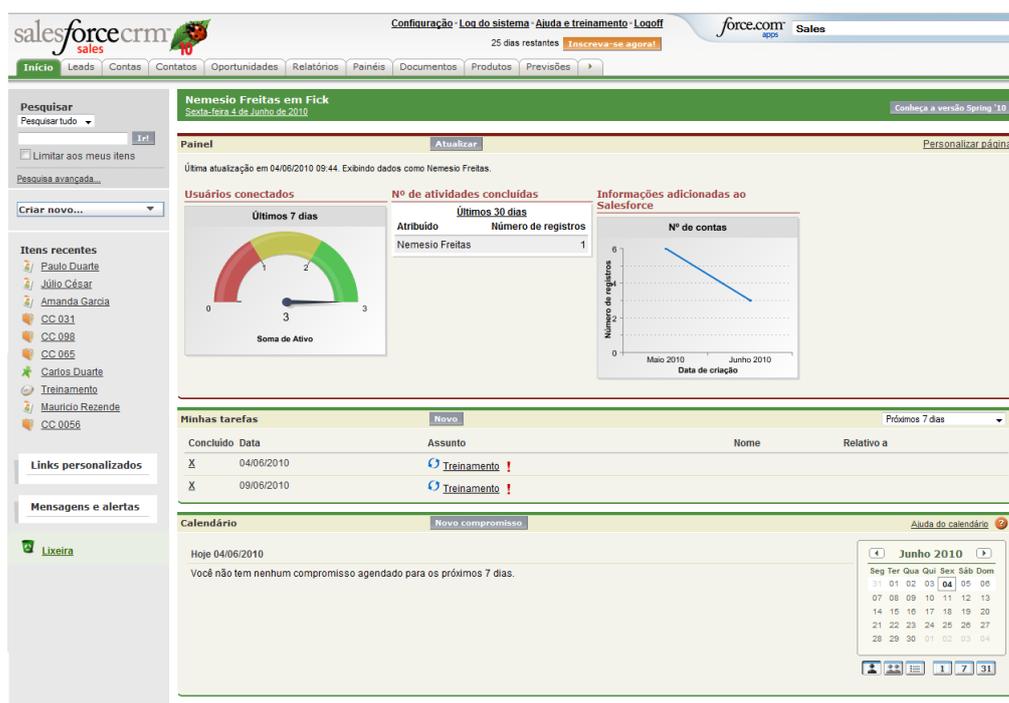


Figura 6.1. Tela principal do produto "Salesforce CRM Sales".

Nas próximas seções, será feito um detalhamento de cada etapa do processo de avaliação, relacionando-as ao estudo de caso, dando uma visão prática e sistemática de cada etapa do processo.

6.1.2 Definir Objetivos e Contextualização da Avaliação

No contexto deste trabalho, o objetivo da aplicação do método ao estudo de caso, é o de avaliar o MAQSaaS em relação ao seu modelo de qualidade, seu processo de avaliação e as técnicas utilizadas. A expectativa do estudo também é avaliar o produto "Salesforce CRM Sales", verificando suas conformidades com requisitos de qualidade.

Para definir claramente os objetivos e o propósito da avaliação, algumas questões foram formuladas e respondidas pelo Requisitante da avaliação:

- Qual é o domínio de aplicação do produto SaaS avaliado?

É um sistema CRM, comercializado sobre a forma de serviço. Tem o objetivo geral de facilitar e auxiliar a gestão de relacionamento com os clientes dentro das organizações.

- Quais são os objetivos específicos buscados nesta avaliação? (ex: aquisição de novos produtos, melhoria de produtos já existentes, conformidades com normas e padrões, entre outros)

O objetivo principal buscado nesta avaliação é avaliar o produto "Salesforce CRM Sales", de modo a assegurar a sua qualidade, indicando pontos de melhorias para as características definidas no modelo do Capítulo 4.

- Quais critérios de qualidade serão avaliados?

Funcionalidade, Usabilidade, Segurança, Desempenho, Apoio (Suporte), Nível de Serviço e Portabilidade, todas com embasamento no paradigma SaaS.

- Quais são as principais funções do produto avaliado?

Suas principais funcionalidades são as de integrar pessoas, processos e tecnologia da organização, tendo como principal aplicação: gerenciamento de venda, marketing, serviços e parceiros.

- Quem são os principais usuários do produto?

São gerentes de vendas, diretores, executivos e usuários vinculados aos negócios da organização.

- Qual o nível de conhecimento exigido dos usuários?

O nível exigido é relativo ao conhecimento para se utilizar e manusear ferramentas de escritório, como por exemplo, ferramentas similares a da *Microsoft Office*. Como a avaliação segue um roteiro com pequenas atividades atribuídas ao Usuário Entrevistado, esse conhecimento não se torna um problema em questão.

- Qual a velocidade de banda larga necessária para a execução do sistema? (Lembrando que o desempenho de um produto SaaS está ligado diretamente a capacidade da banda larga disponível, Turner et al. [2003]; Ma [2007])

A taxa ideal mínima de banda larga deve ser entre 200 a 300 kbps.

As questões e respostas listadas nesta etapa são exemplificativas ao contexto do estudo de caso realizado, não sendo taxativas a outras avaliações. Dependendo do nível da avaliação, outras questões podem ser elaboradas relacionando-as com os objetivos e propósitos da avaliação.

6.1.3 Definir e Configurar o Modelo de Qualidade

O modelo de qualidade adotado nas avaliações práticas foi o modelo proposto no Capítulo 4, podendo ser visualizado de forma resumida pela Figura 4.2, na seção 4.1.

Tal modelo foi escolhido e utilizado, pois além de contemplar um conjunto de critérios ligados à avaliação da qualidade de um sistema SaaS, ele também apresenta uma divisão de Requisitos (diretrizes) e atributos de avaliação, facilitando a sua utilização em avaliações práticas.

Os atributos mensuráveis utilizados para efetuar as avaliações no produto SaaS serão obtidos desse modelo de qualidade. Porém, deve-se frisar que nem todas as características contidas no modelo precisam ser utilizadas. É relevante utilizar somente aquelas referentes aos objetivos e escopo da avaliação. No trabalho em questão todos os critérios e requisitos foram utilizados como forma de avaliação, pois um dos objetivos do estudo de caso é avaliar a amplitude deste modelo de qualidade.

6.1.4 Identificar Métricas Mensuráveis

Esta etapa diz respeito ao desdobramento dos requisitos de qualidade em atributos que possam ser medidos e pontuados. Para a realização desta tarefa, técnicas e metodologias auxiliares podem ser utilizadas para apoiarem e facilitar a identificação de métricas mensuráveis. A técnica utilizada na presente avaliação foi a utilização de lista de verificação, de modo a permitir um eficiente mapeamento dos requisitos do modelo de qualidade em atributos e itens mensuráveis.

O desdobramento em itens e questões tem por objetivo a criação de uma escala de pontuação, facilitando o uso pelos Avaliadores, pois os itens serão usados como guias de identificação do que deve ser avaliado. Devem ser, portanto, mais completos e abrangentes possíveis, expressando corretamente o significado do atributo, do ponto de vista da avaliação.

Os atributos mensuráveis contidos no modelo de qualidade foram levantados através do conhecimento do Especialista, criando questões simples e objetivas que passaram a ser avaliados e pontuados na forma de um *checklist*.

Ao total foram propostas 70 questões a serem avaliadas, sendo questões decompostas em 7 (sete) grupos distintos, referentes aos critérios definidos no modelo de qualidade, sendo eles: Funcionalidade, Usabilidade, Segurança, Desempenho, Apoio (Suporte), Nível de serviço e Portabilidade.

A lista de verificação proposta para as avaliações também incluem outras informações e instruções, de maneira a possibilitar sua utilização pelo Avaliador na forma de um *template* de aplicação. Essas informações podem ser relacionadas e adicionadas à lista de verificação, indicando, por exemplo, o tempo médio decorrido nas avaliações, objetividade da avaliação, guia para preenchimento das alternativas do *checklist*, definição do local e da data, entre outras informações que facilitem sua aplicação e controle.

O *template* completo usado no estudo de caso, contendo as questões da avaliação (lista de verificação) e suas informações, pode ser visualizado no Apêndice C. A seguir, na Tabela 6.1, são mostrados os objetivos esperados pelo conjunto de questões formuladas no Apêndice C, tendo como apoio fundamentações de autores e trabalhos científicos.

Critério de qualidade	Referência	Objetivo das questões
Funcionalidade	ISO/IEC 9126 [1991]; Cancian [2009]; Dubey & Wagle [2006].	Tem por objetivo avaliar aspectos ligados a funcionalidade, como por exemplo, coesão e precisão na realização das tarefas. De modo geral poderá ser verificado se o sistema funciona corretamente de acordo com as necessidades dos usuários.
Usabilidade	ISO/IEC 9126 [1991]; Cancian [2009]; Dubey & Wagle [2006]; Havelka et al. [1998].	As questões foram montadas com o objetivo de avaliar se os usuários empregam com facilidade as opções do sistema; e qual seu nível de satisfação visando à ergonomia, interface, entre outras características de usabilidade.
Continua na próxima página...		

Critério de qualidade	Referência	Objetivo das questões
Segurança	ITIL v3 [2008]; COBIT (ISACA [2008]); O'Brien et al. [2005]; Cancian [2009]; Dubey & Wagle [2006]; Turner et al. [2003]; Ma [2007].	Sendo um dos aspectos mais críticos para os sistemas SaaS, as questões elaboradas tiveram preocupação em avaliar itens referentes ao acesso dos usuários, mecanismos contra invasão, principalmente em senhas, dados e transações que são efetuadas via web.
Desempenho	ISO/IEC 9126 [1991]; O'Brien et al. [2005]; Cancian [2009]; Dubey & Wagle [2006]; Choudhary [2007].	As questões alusivas ao desempenho se preocupam com a estabilidade do sistema durante o acesso de vários usuários, em diferentes organizações, juntamente com o tipo de dados que é transmitido.
Apoio (Suporte)	ISO/IEC 9126 [1991]; ITIL v3 [2008]; COBIT (ISACA [2008]); Aulbach et al. [2008].	Neste critério as questões foram elaboradas objetivando avaliar como o produto se comporta após ocorrerem modificações em suas funções; juntamente ao tratamento de incidentes, problemas e correções de defeitos.
Nível de serviço	ITIL v3 [2008]; COBIT (ISACA [2008]); Cancian [2009]; Dubey & Wagle [2006]; Havelka et al. [1998].	Tem por objetivo avaliar o processo gerencial de planejamento, controle e definição dos acordos de níveis de serviço.
Portabilidade	ISO/IEC 9126 [1991]; Dubey & Wagle [2006]; Havelka et al. [1998]; Cancian [2009].	As questões tem por objetivo verificar como o sistema se comporta em diferentes sistemas operacionais, navegadores; e até mesmo em execução com outros sistemas SaaS ao mesmo tempo.

Tabela 6.1: Objetivos e fundamentações das questões do *checklist*.

6.1.5 Definir Pontuação e Critérios para Julgamento

Para definir a pontuação das questões da lista de verificação, deve-se primeiramente especificar qual o domínio de respostas nominais que podem ser aplicadas às questões do *checklist* e posteriormente identificar seus valores numéricos (seção 5.3).

No estudo de caso, a lista de verificação apresenta para cada critério de qualidade 10 atributos na forma de questões, totalizando uma lista de verificação com 70 itens. Tal número foi definido, pois permite construir uma lista de verificação completa e ao mesmo tempo não exaustiva, o que prejudicaria o tempo de realização das avaliações. A pontuação dada para cada questão foi definida da seguinte maneira:

- Valor máximo: "1"(maior pontuação);
- Valor mínimo: "0"(menor pontuação).

Cada critério de qualidade avaliado possuía necessariamente 10 itens a serem analisados e pontuados, portanto, o intervalo de valor em relação à nota obtida poderia ser de 0 (zero) a 10 (dez). Para cada critério de qualidade foram realizadas 15 avaliações com pessoas distintas, ou seja, diferentes instancias de avaliações foram efetuadas, sendo obtidas notas diferentes em relação à qualidade final do critério avaliado.

Ao final de todas as 15 avaliações, obtendo 15 valores distintos em relação à qualidade de cada critério avaliado, foi realizada a média aritmética desses valores, possibilitando uma melhor comparação dos resultados.

Este processo, de obtenção da média da qualidade, foi efetuado para os 7 (sete) critérios avaliados, permitindo uma separação da nota específica de cada critério frente a nota final do produto avaliado, pois é interessante para o Requisitante da avaliação identificar as notas separadamente, verificando quais critérios de qualidade estão abaixo do nível de qualidade exigidos, possibilitando ajustes e melhorias.

As notas obtidas em relação aos critérios de qualidade passaram por um processo de normalização, todas as 7 (sete) notas obtidas foram normalizadas para 100 (cem), tal procedimento foi realizado visando uma maior facilidade para realizar as comparações com os critérios de julgamento propostos.

A nota da qualidade final do produto avaliado, tendo como base os 7 (sete) critérios avaliados, foi calculada através da média aritmética das médias de qualidade em relação a cada critério, sendo eles: Funcionalidade, Usabilidade, Segurança, Desempenho, Apoio (Suporte), Nível de Serviço e Portabilidade.

Note que a formulação matemática utilizada neste processo é bem simples, utilizada basicamente para achar a média entre os valores, pois neste caso estamos interessados em achar o valor da qualidade final do produto. Se o objetivo desta avaliação

fosse avaliar critérios mais específicos, como por exemplo, segurança, desempenho, entre outros, deveriam ser atribuídos pesos diferentes para tais critérios, utilizando uma descrição estatística mais apropriada e elaborada.

As notas obtidas em referência a média da qualidade de cada critério, juntamente com a nota obtida pela qualidade final do produto avaliado serão exemplificados e detalhadas na subseção 6.1.8.

Ao final, tendo os dados calculados, devem-se definir os critérios para o julgamento da nota final. Para este julgamento os autores se basearam em um levantamento feito em aproximadamente 300 produtos de software, proposto por Martinez [1999]. Os critérios presentes no trabalho de Martinez [1999], se baseiam basicamente em três níveis: Superior, Médio e Inferior, podendo ser visualizados pela Figura 6.2.

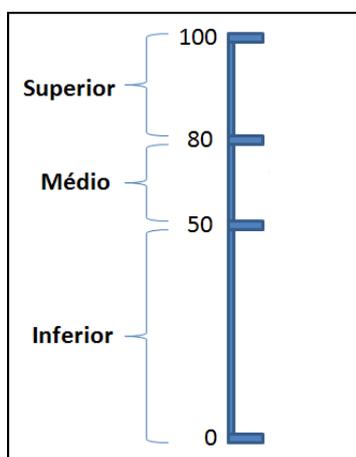


Figura 6.2. Critérios para julgamento da qualidade de um produto de software, Martinez [1999].

Esses critérios são aceitos e confiáveis devido a sua ampla utilização na comunidade de qualidade de software, mostrando eficiências nos resultados dos trabalhos que os utilizaram. Tais critérios foram modificados e incorporados ao MAQSaaS, tendo as seguintes definições e perspectivas de conclusão:

- Para que o produto de software obtenha o nível **Superior**, ele deve estar de acordo com pelo menos 80% dos requisitos de qualidade especificados. Isso significa que a nota pode variar de 100 a 80, numa escala de 0 a 100;
- Para que o produto de software obtenha o nível **Médio**, ele deve estar de acordo com pelo menos 50% dos requisitos de qualidade especificados. Isso significa que a nota pode variar de 80 a 50;

- Para que o produto de software obtenha o nível **Inferior**, a nota deve variar de 50 a 0.

De acordo com a escala visualizada na Figura 6.2, se o produto SaaS analisado estiver no intervalo "Superior", pode-se constatar que o produto de software apresenta um bom nível de qualidade, se adequando aos critérios de qualidade estabelecidos. Por outro lado se ele estiver no intervalo "Médio", pode-se concluir que o próprio precisa de alguns ajustes, pecando em alguns critérios de qualidade.

Por fim, se a qualidade do software alcançar a pontuação do nível "Inferior", significa que uma análise aprofundada deve ser realizada, pois o sistema não está em conformidade com os critérios mínimos de qualidade estabelecidos.

6.1.6 Projetar a Avaliação

Nesta etapa é definido o plano de avaliação que foi seguido durante o estudo de caso.

Em resumo, como já foi destacado nas seções anteriores, as avaliações foram realizadas sobre o produto SaaS "Salesforce CRM Sales", tendo a disposição um versão *Trial* do sistema, versão no qual possuía todas as funcionalidade da versão paga, porém, com uma única restrição, poderia ser acessado somente durante 2 meses, o bastante para realizar e efetuar as avaliações.

Os Usuários Entrevistados durante as avaliações foram alunos com conhecimentos avançados em informática, possuindo de certa forma familiaridade com o produto SaaS avaliado. A maioria dos alunos participantes cursavam os cursos de Ciência da Computação e Sistemas de Informação.

Para a realização das avaliações foi utilizado um computador *desktop* com acesso a internet, possibilitando o acesso ao sistema. Para cada avaliação o Usuário Entrevistado teve a sua disposição uma lista de verificação impressa, juntamente com lápis ou caneta para preenchê-la com seus dados e opiniões. O tempo de cada avaliação foi estimado e executado entre 40 a 50 minutos, tentando não ultrapassar este tempo, visando garantir uma avaliação rápida e prática.

As avaliações foram efetuadas em aproximadamente três semanas, sendo realizadas nas dependências da UFLA (Universidade Federal de Lavras) e UFMG (Universidade Federal de Minas Gerais). Todas as avaliações foram agendadas considerando os horários disponíveis dos Usuários Entrevistados.

As respostas recebidas durante as avaliações foram obtidas mediante a opinião dos Usuários Entrevistados e as questões mais técnicas tiveram o auxílio da documentação do produto, evidenciando aspectos mais técnicos e específicos da avaliação.

Todas as informações e dados levantados dos Usuários Entrevistados foram mantidos de posse somente dos autores, garantindo assim sigilo por parte dos entrevistados. Tais informações levantadas não têm valor comercial, somente agregam valores para a pesquisa e ao desenvolvimento do presente trabalho.

No total foram realizadas 15 avaliações com 15 pessoas distintas. Este número de avaliações teve como objetivo a melhora do método proposto, juntamente com a convergência dos dados obtidos. Isso porque as primeiras avaliações mostraram erros e detalhes a serem melhorados, tanto por parte do modelo de qualidade como por parte do processo de avaliação.

A Figura 6.3 mostra alguns dados da aplicação das avaliações, como por exemplo, as titulações, cursos e instituições de cada Usuário Entrevistado, juntamente com o tempo decorrido de cada avaliação.

Usuário Entrevistado	Titulação	Curso	Instituição	Tempo Decorrido (min)
#1	Mestrado	Ciência da Computação	UFMG - Universidade Federal de Minas Gerais	72
#2	Mestrado	Ciência da Computação	UFMG - Universidade Federal de Minas Gerais	80
#3	Graduação	Sistemas de Informação	UFMG - Universidade Federal de Minas Gerais	65
#4	Graduação	Sistemas de Informação	UFMG - Universidade Federal de Minas Gerais	59
#5	Graduação	Ciência da Computação	UFMG - Universidade Federal de Minas Gerais	43
#6	Mestrado	Ciência da Computação	UFMG - Universidade Federal de Minas Gerais	52
#7	Mestrado	Ciência da Computação	UFMG - Universidade Federal de Minas Gerais	46
#8	Mestrado	Ciência da Computação	UFMG - Universidade Federal de Minas Gerais	43
#9	Graduação	Ciência da Computação	UFMG - Universidade Federal de Minas Gerais	55
#10	Graduação	Sistemas de Informação	UFLA - Universidade Federal de Lavras	51
#11	Graduação	Sistemas de Informação	UFLA - Universidade Federal de Lavras	54
#12	Graduação	Sistemas de Informação	UFLA - Universidade Federal de Lavras	42
#13	Graduação	Ciência da Computação	UFLA - Universidade Federal de Lavras	57
#14	Graduação	Ciência da Computação	UFLA - Universidade Federal de Lavras	45
#15	Graduação	Ciência da Computação	UFLA - Universidade Federal de Lavras	42

Figura 6.3. Perfil acadêmico dos Usuários Entrevistados.

Assim que os problemas foram sanados foi possível a obtenção de dados convergentes nas demais avaliações, ou seja, os dados obtidos estavam sendo consistentes, evidenciando assim uma maior repetitividade, reprodução, imparcialidade e objetividade nas avaliações.

Os resultados finais associados a cada critério de qualidade do produto avaliado foram definidos e descritos em um relatório. O relatório final da avaliação aborda de modo geral os resultados ao longo das avaliações juntamente com aspectos positivos e

negativos observados durante a sua realização. O detalhamento destes resultados será feito na subseção 6.1.8 deste capítulo.

6.1.7 Execução da Avaliação

Basicamente nesta etapa do estudo de caso o Avaliador executou três tarefas:

- coleta das medidas;
- comparação das medidas com critérios pré-definidos;
- e por fim, o julgamento dos dados obtidos ao longo das avaliações.

De acordo com a atividade de obtenção de medidas, todas foram coletadas através das avaliações feitas pelos Usuários Entrevistados, deixando clara a impessoalidade do Avaliador que estava presente somente para tirar dúvidas e executar de forma repetível a avaliação.

Ao término das avaliações, iniciando a atividade de comparação de critérios, os dados numéricos foram tratados (normalizados, plotados em gráficos ou em tabelas) e comparados com os critérios da subseção 6.1.5, para que fosse possível um julgamento final. Para efeito de amostragem do trabalho científico, todas as notas obtidas em relação a cada questão da lista de verificação podem ser visualizadas no Apêndice E, juntamente com as suas descrições.

Tendo os dados numéricos em mãos, a atividade de julgamento dos dados foi efetuada juntamente com o desenvolvimento de um relatório final de avaliação. O objetivo de elaborar um relatório final é de indicar os pontos positivos e negativos relacionando-os com os critérios de qualidade julgados.

O relatório além de identificar aspectos relevantes sobre a qualidade do produto SaaS, também identifica aspectos problemáticos ocorridos durante a execução das avaliações, como por exemplo, falta de informações, indisponibilidade do produto, erros gerados ao longo da execução, entre outros.

Esta etapa finaliza o processo de avaliação do MAQSaaS. Cada avaliação pode requerer pequenas modificações no método de acordo com as suas necessidades. Os resultados finais das avaliações, identificando a qualidade obtida pelo sistema avaliado, juntamente com algumas discussões sobre as notas obtidas e a elaboração do relatório final, serão detalhados e comentados na próxima subseção.

6.1.8 Considerações e resultados das avaliações

Ao final das 15 avaliações, tendo os dados tratados e o julgamento dos dados terem sido realizados, algumas conclusões e considerações podem ser apresentadas. Nesta subseção será discutida a nota que o sistema SaaS obteve ao final das avaliações, juntamente com as notas obtidas em relação a cada critério avaliado.

A Figura 6.4 mostra a relação da nota final obtida pelo produto "Salesforce CRM Sales" ao longo das avaliações. O Produto teve uma nota média de 76,6, que, de acordo com os critérios definidos na subseção 6.1.5, é tido como nível "Médio" para a qualidade de um produto SaaS.

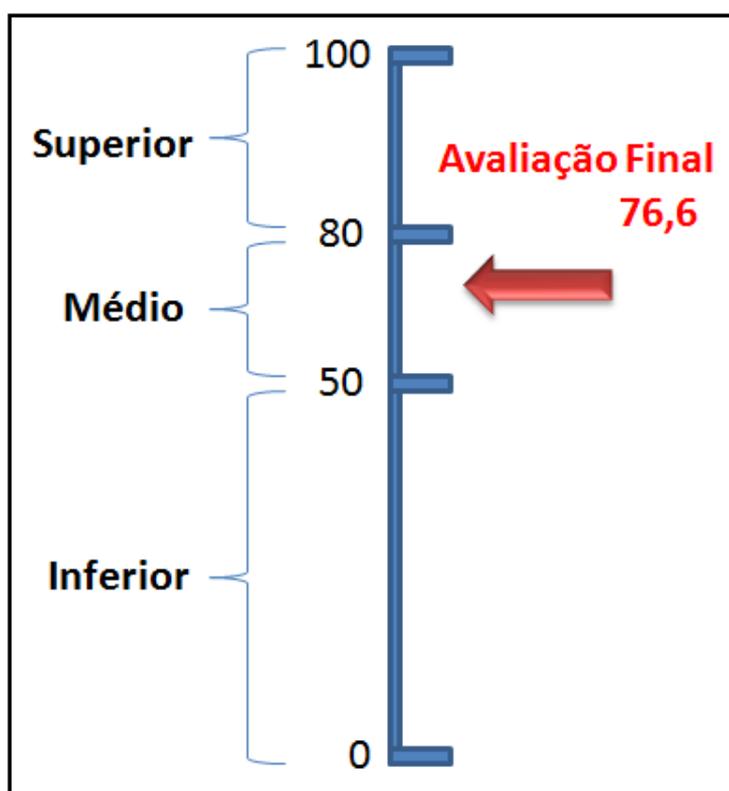


Figura 6.4. Qualidade alcançada pelo produto "SalesForce CRM Sales".

Analisando a nota obtida pode-se concluir que, de modo geral o produto analisado necessita de alguns ajustes e melhorias, pois o produto não está de acordo com todos os requisitos de qualidade exigidos nas avaliações.

Mesmo apresentando um nível "Médio" de qualidade, nem todos os requisitos do produto avaliado tiveram notas insatisfatórias dentro do contexto SaaS. A análise e o resultado da avaliação de cada critério avaliado podem ser visualizados abaixo, Figura 6.5 :

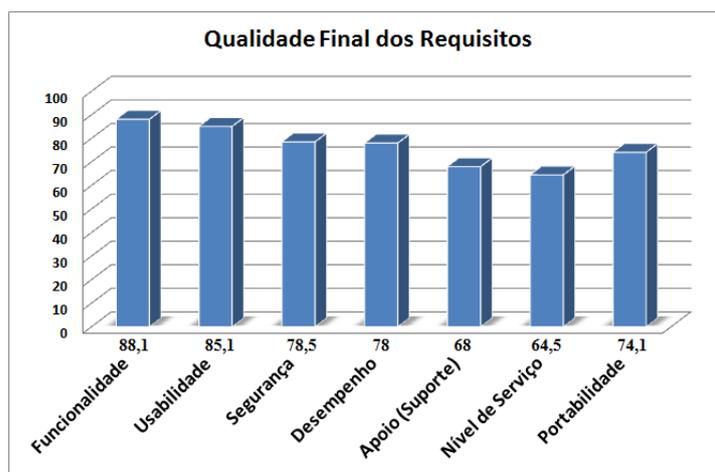


Figura 6.5. Nota atribuída em relação a cada critério de qualidade.

De acordo com a Figura 6.5 pode-se concluir que, os critérios **Funcionalidade** e **Usabilidade** tiveram as maiores pontuação na avaliação efetuada, contemplando níveis satisfatórios de qualidade. Nota-se também, que estes critérios são muito similares a de um produto de software convencional, não contemplando características intrínsecas a de um sistema SaaS.

Por outro lado, analisando os critérios **Apoio(Suporte)** e **Nível de serviço**, constata-se que ambos tiveram as piores notas dentro dos critérios avaliados, observando que são características voltadas diretamente a um produto SaaS. Este resultado tem um impacto negativo para a qualidade do produto, pois ambos os critérios, representam uma relevância e impacto grande ao contexto SaaS, necessitando estudos e melhorias em relação aos requisitos que os constituem, como por exemplo, aspectos ligados a SLA´s, gerenciamento de dados, personalização, estabilidade, entre outros.

Em adição a análise realizada, para cada um dos critérios avaliados foram descritos abaixo na Tabela 6.2 aspectos positivos e negativos de seus requisitos durante as avaliações. Estes aspectos têm por objetivo agregar informações adicionais ao Requisite, possibilitando-lhe futuramente realizar ajustes e mudanças em relação aos aspectos negativos ocorridos durante as avaliações, garantindo assim uma maior qualidade para o produto avaliado.

Critério	Pontos Positivos	Pontos negativos
Funcionalidade	A maioria das funções presentes no sistema satisfizeram de maneira correta e eficiente as necessidades dos usuários, funcionando de modo correto em relação ao contexto aplicado.	Durante as avaliações no sistema CRM, perceberam-se algumas limitações na exportação de dados para formatos específicos de documentos, principalmente no reconhecimento de tipos de documentos utilizados durante a inserção de fotos dos funcionários.
Usabilidade	A primeira vista o sistema chegou a "assustar" devido à variedade de opções disponíveis em sua tela. Porém analisando os quesitos da avaliação, o sistema mostrou estar bem organizado, juntamente com as abas de serviços e funções. De maneira geral o produto analisado apresentou uma interface "amigável" e prática aos usuários.	Um ponto que prejudicou a avaliação foi a ausência de sistemas de demonstrações e tutoriais de uso, dificultando o contato de usuários inexperientes.
Segurança	O sistema apresentou diversos mecanismos técnicos que garantiram uma maior segurança, agregando confiabilidade e integridade às informações dos usuários, principalmente na utilização do multi-inquilinismo.	O aspecto mais relevante durante a avaliação deste critério foi a ausência de um mecanismo que verificasse quantas vezes o usuário tentou efetuar seu login utilizando uma senha errada. Similar a um mecanismo de caixa eletrônico, se o usuário tentou várias vezes este mesmo procedimento, o sistema deveria efetuar o bloqueio da conta (mecanismo eficiente contra ataques de <i>cracker's</i>), liberando-a mediante a notificação do administrador.
Continua na próxima página...		

Critério	Pontos Positivos	Pontos negativos
Desempenho	O sistema se mostrou eficiente utilizando o multi-inquilinismo (utilização do sistema por usuários diferentes ao mesmo tempo). Sua utilização também pode ser efetuada por organizações diferentes, garantindo sigilozidade dos dados.	Apresentou problemas de inclusão de novos serviços, não apresentando uma interface de contrato bem definido, dificultando a inclusão de novos serviços por parte dos gestores da aplicação.
Apoio (Suporte)	Apresentou um processo eficiente para a resolução de incidentes e problemas, possibilitando aos usuários verificarem seus status ao longo do processo. Todo o processo de erros é protocolado facilitando o acompanhamento dos gestores e usuários. O sistema também possibilita pequenas modificações e customizações feitas pelo próprio administrador. São algumas customizações limitadas, mas que ao mesmo tempo ajudam a deixar o sistema mais adequado ao perfil dos usuários, como por exemplo, troca de idiomas; cores dos label's e botões; e redimensionamento de grid's e paletas.	O atendimento ao suporte técnico não está disponível online; outro problema que se agrava é que não foi verificado um suporte técnico disponível no idioma nativo que a aplicação estava sendo utilizada, apenas os idiomas inglês e espanhol estavam disponíveis.

Continua na próxima página...

Critério	Pontos Positivos	Pontos negativos
Nível de serviço	<p>O sistema está em conformidade com diversas questões de auditoria jurídicas e financeiras, principalmente tendo suporte a auditoria SAS 70, que está sendo amplamente utilizada em todas as organizações (fornecedoras e contratantes de soluções SasS).</p> <p>Outro aspecto positivo é o fato do sistema avisar aos usuários sempre que houver uma indisponibilização em virtude de manutenções. Este aviso geralmente é feito com antecedência e as manutenções acontecem geralmente nos finais de semana, possibilitando que a organização e os gestores se organizem em virtude desta indisponibilização.</p>	<p>O produto apresenta um SLA (acordo de nível de serviço), porém este SLA não é bem definido, possuindo ambigüidades e não deixando claras as implicações legais que a organização fornecedora deveria ter durante o descumprimento de cláusulas essenciais ao funcionamento do sistema.</p> <p>Outro fato relevante é a questão do gerenciamento dos dados. A empresa garante que os dados estão seguros e íntegros, mas em nenhum momento ela fornece funcionalidades para o gerenciamento desses dados na forma de relatórios; nem a possibilidade de <i>backups</i> e migração desses dados para outros sistemas.</p>
Portabilidade	<p>O produto analisado teve bom comportamento durante sua utilização em diferentes navegadores (<i>browser's</i>), mantendo suas características e funcionamento em todos os navegadores testados, sendo eles: Internet Explorer, Firefox, Chrome, Safari e Opera.</p>	<p>Um dos problemas encontrados foi em relação à portabilidade para dispositivos móveis.</p> <p>Alguns aparelhos móveis, compatíveis com o CRM avaliado de acordo com a documentação, foram analisados apresentando dificuldades na utilização. Não existia um manual e todo procedimento era baseado nas experiências próprias dos usuários.</p> <p>Os aparelhos que conseguiram se conectar ao sistema apresentaram uma qualidade inferior e performance abaixo do esperado.</p>

Tabela 6.2: Aspectos positivos e negativos em relação aos critérios avaliados.

Para finalizar as considerações sobre os resultados das avaliações, poucos pro-

blemas foram identificados durante a realização do estudo de caso. De modo geral o processo de avaliação se mostrou eficiente podendo ser aplicado de maneira correta juntamente com o modelo de qualidade proposto. Ao final, o método garantiu ser **Repetível, Reproduzível, Imparcial e Objetivo**, pois:

- As avaliações puderam ser repetidas com diferentes entrevistados; garantindo tempos similares de execução dentro de um intervalo de tempo definido pelo Avaliador. O único ponto em destaque, foi o fato no qual as primeiras avaliações gastaram mais tempo do que o estipulado para suas execuções. Isso se deve ao fato da inexperiência do Avaliador durante as primeiras avaliações; e erros encontrados dentro do próprio método utilizado.

A partir da terceira e da quarta avaliação, possibilitando maior experiência por parte do Avaliador em relação ao método proposto; e os erros tendo sido corrigidos, foi possível estipular melhor o tempo de execução, permitindo a realização das avaliações práticas em aproximadamente 50 minutos;

- O método contém especificações, diretrizes, artefatos (lista de verificação, relatórios de resultados), passos e etapas bem definidas, possibilitando ser aplicável por diferentes pessoas e organizações, através de um processo de avaliação que se apóia em um modelo de qualidade específico aos SaaS;
- O estudo de caso foi aplicado em pessoas que possuíam titulações e formações acadêmicas diferentes, cursando cursos em diferentes instituições de ensino. Isso mostrou que, mesmo sobre a influência de fatores internos e externos os dados obtidos ao longo das avaliações se mostraram convergentes, possuindo uma porcentagem muita pequena de discrepância em relação ao conjunto de dados obtidos;
- Ao final do estudo de caso o método obteve resultados claros, definindo aspectos positivos e negativos de qualidade em relação ao software avaliado. O relatório final indica aspectos de melhoria, de modo a evitar ambigüidades e diferentes interpretações nos dados finais.

O resumo das considerações da avaliação final e das notas obtidas pelos critérios de qualidade, juntamente com os pontos positivos e negativos, foram agrupados e relacionados com informações adicionais, proporcionando um *template* "Relatório final", visualizado no Apêndice D.

A próxima seção mostra um comparativo entre o MAQSaaS e o método MEDE-Pros, com a finalidade de mostrar as diferenças e limitações entre um método específico ao paradigma SaaS e outro específico a produtos de software convencionais.

6.2 Comparativo com o método MEDE-Pros

Esta seção foi desenvolvida com o propósito de responder alguns questionamentos que podem ser levantados e evidenciados ao final do estudo de caso. A seguir segue alguns exemplos dessas questões:

- Existe mesmo a necessidade do desenvolvimento de um método específico para avaliação de produtos SaaS?
- Os métodos convencionais não estão preparados e adequados para avaliar um produto SaaS?
- Quais as semelhanças e diferenças do MAQSaaS em comparação com métodos convencionais?

Para que tais questionamentos pudessem ser respondidos de forma fundamentada, resolveu-se realizar antes da conclusão do presente trabalho, uma análise comparativa entre o método proposto, MAQSaaS, e o método MEDE-Pros, método convencional de avaliação de produto de software, apresentado no Capítulo 2.

Para realização deste comparativo alguns passos foram seguidos, permitindo a comparação entre os dados e resultados de ambos os métodos. As etapas efetuadas para a aplicação do comparativo podem ser visualizadas na Figura 6.6, apresentando resumidamente 5 (cinco) etapas para a realização do comparativo. Logo abaixo, para cada etapa será feita uma breve descrição das atividades desempenhadas.

1-Etapa: A primeira atividade do comparativo foi a escolha de um método convencional que proporcionasse uma avaliação prática da qualidade em um produto de software. O MEDE-Pros foi escolhido, pois além de avaliar a qualidade de um software de acordo com a visão do usuário, ele possibilita também verificar conformidades com padrões de normas internacionais de qualidade, MEDE-Pros [1996].

Outro aspecto importante para sua adoção neste comparativo foi a facilidade da sua aplicação em avaliações práticas. Em diversos trabalhos, como por exemplo, Maintinguer et al. [2003]; Martinez [1999]; Beus-Dukic & Boegh [2003], é observado a eficiência e facilidade no qual este método é aplicado, sendo um método fácil e bem estruturado de ser utilizado;

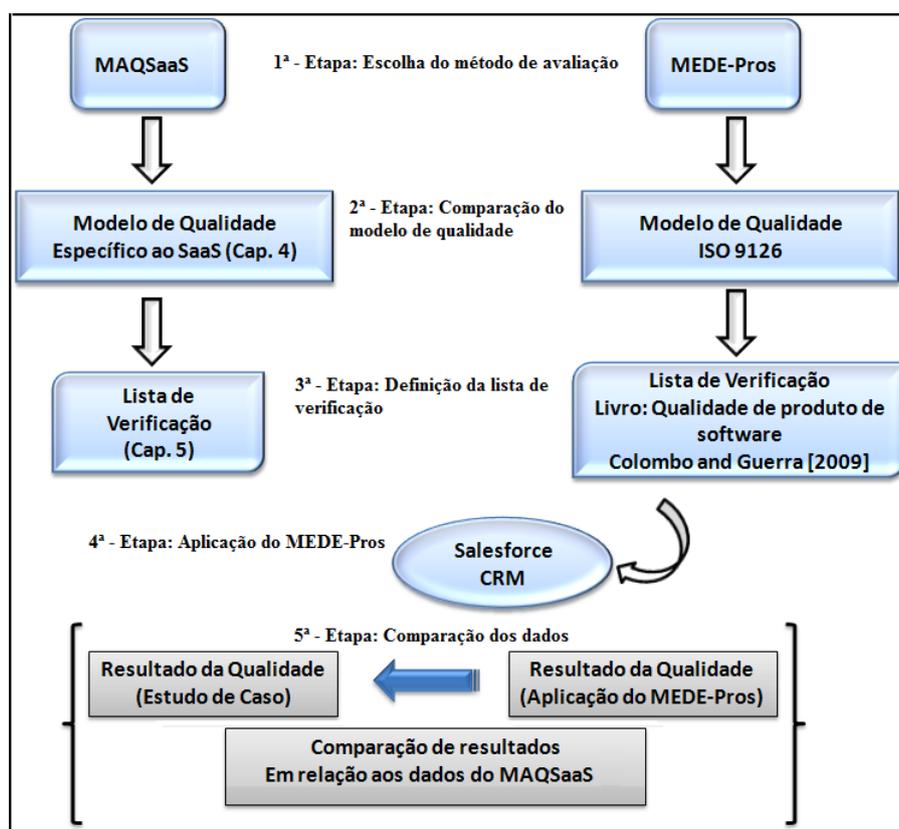


Figura 6.6. Etapas para a realização do comparativo entre os métodos MAQSaaS e MEDE-Pros.

2-Etapa: A segunda etapa, foi o estudo e o detalhamento do modelo de qualidade no qual o MEDE-Pros se baseia. O modelo de qualidade do MEDE-Pros, a ISO 9126, foi comparado com o modelo de qualidade seguido pelo MAQSaaS, identificando as principais diferenças e semelhanças entre ambos os modelos.

Além de algumas diferenças de critérios e diretrizes de qualidade, a maior diferença é o fato de que a ISO 9126 se baseia fortemente na qualidade em uso, interna e externa, enquanto que, o modelo de qualidade do MAQSaaS se baseia somente nos aspectos a qualidade interna e externa devido ao escopo do trabalho;

3-Etapa: Para a realização da terceira etapa, foi necessário utilizar uma instância de perguntas e verificações (*checklist*) presente no livro: "Qualidade de Produto de Software" proposto pelos autores Colombo & Guerra [2009], onde descrevem a realização de um estudo de caso aplicando na prática o método MEDE-Pros.

A lista de verificação dos autores Colombo & Guerra [2009] foi utilizada, pois além de estar de acordo com todas as características e diretrizes da norma ISO 9126, ela também está de acordo com os moldes do processo da norma ISO 14598, tendo diretrizes para avaliação e pontuação dos atributos de qualidade;

4-Etapa: A quarta etapa faz referência à aplicação do método MEDE-Pros sobre o produto "Salesforce CRM Sales", porém, para a realização da aplicação da lista de verificação ao produto SaaS, seguiu-se o processo de avaliação proposto pelo MEDE-Pros, no caso a ISO 14598, juntamente com as suas etapas e diretrizes.

Alguns aspectos foram desconsiderados em virtude do escopo e tempo do trabalho. Apenas 3 (três) avaliações foram realizadas, sendo os dados obtidos e tratados de acordo com o processo da norma ISO 14598;

5-Etapa: Nesta fase, os dados obtidos nas avaliações com o MEDE-Pros foram comparados com os dados de qualidade obtidos no estudo de caso do Capítulo 6, seção 6.1. Para a realização do comparativo, foi feito um mapeamento dos requisitos de qualidade da norma ISO 9126 (MEDE-Pros) para os requisitos do modelo de qualidade do MAQSaaS, de maneira a adequar e facilitar as comparações.

Ao final do comparativo, com os dados analisados e interpretados (Figura 6.7), um gráfico do tipo "radar" foi montado, Figura 6.8, sendo plotado os valores de cada critério de qualidade ao longo dos eixos, representados pelos dois métodos, podendo assim verificar a variação de valores das notas obtidas por ambos os métodos.

Notas dos critérios de qualidade (MAQSaaS x MEDE-Pros)							
	Funcionalidade	Usabilidade	Segurança	Desempenho	Apoio (Suporte)	Nível de Serviço	Portabilidade
MAQSaaS	88.1	85.1	78.5	78	68	64.5	74.1
MEDE-Pros	95	94	80	65	30	40	92

Figura 6.7. Dados obtidos pelo comparativo dos critérios de qualidade (MAQSaaS vs MEDE-Pros).

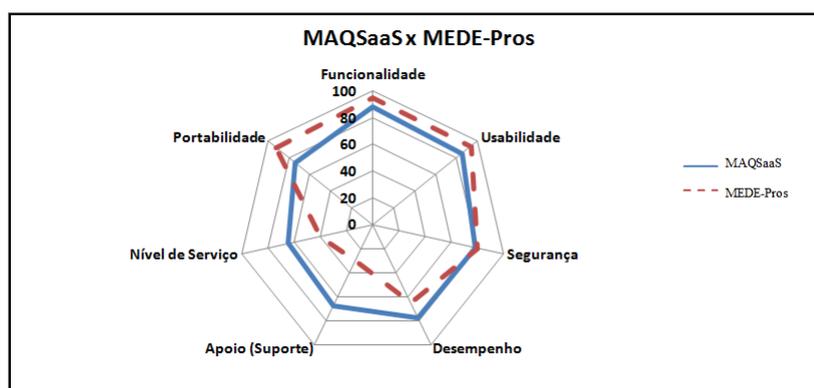


Figura 6.8. Resultado comparativo dos critérios de qualidade (MAQSaaS vs MEDE-Pros).

De maneira a responder aos questionamentos levantados no início desta seção, em relação às diferenças e limitações que os métodos convencionais podem possuir em

relação ao MAQSaaS, verificando a real necessidade de se propor um método específico ao paradigma SaaS, ressaltaremos abaixo algumas análises sobre os dados obtidos pelo comparativo.

De acordo com a Figura 6.8, o que se observa é a convergência e a divergência dos dados em relação aos critérios de qualidade. Tanto **Funcionalidade**, **Usabilidade** e **Segurança**, tiveram uma grande convergência em relação aos métodos. Isso se deve ao fato de que tanto em sistemas convencionais como nos SaaS esses critérios de qualidade são semelhantes, abaixo uma pequena exemplificação dessas semelhanças:

- **Funcionalidade**: se assemelha principalmente nos aspectos de interoperabilidade, acesso e acurácia, permitindo a garantia de funções que atendam as necessidades dos clientes, de forma correta e precisa, juntamente com a possibilidade de integração com outros sistemas legados da organização;
- **Usabilidade**: os requisitos apreensibilidade, operacionalidade e atratividade se fazem necessários em sistemas computacionais, sendo de extrema importância para uma maior satisfação dos usuários;
- **Segurança**: mesmo tendo um impacto maior sobre o paradigma SaaS, aspectos ligados a segurança são de grande relevância em sistemas convencionais, principalmente sistemas que trabalham com dados críticos aos negócios das organizações, necessitando garantir uma maior confiabilidade, integridade e disponibilidades desses dados.

O critério **Portabilidade** também teve uma convergência em ambos os métodos, porém, esta característica apresenta um contexto de aplicação bem diferente em ambos os casos. No SaaS ele está focado mais para navegadores (*browsers*) e portas de comunicação. Já nos sistemas convencionais a portabilidade está mais associada a sistemas operacionais e distribuições. Ou seja, os dados obtidos mesmo sendo convergentes, não podem ser considerados coincidentes, pois, os requisitos adaptabilidade e coexistência possuem aplicabilidade bem diferente em ambos os paradigmas.

O **Desempenho** teve uma pequena divergência nos dados, porém era algo já esperado, pois os sistemas SaaS devido ao seu modo de acesso, através de inquilinos e multi-inquilinos (*tenant* e *multi-tenant*) apresentam características distintas na escalabilidade de seus usuários e na expansibilidade de novos serviços ao sistema, Choudhary [2007]; Chong & Carraro [2008].

Os critérios **Nível de Serviço** e **Apoio** tiveram as maiores divergências de resultados. O motivo se deve ao fato de que seus requisitos de qualidade sofrem influências

diretamente de características voltadas do gerenciamento de serviços de TI, não contempladas pela norma ISO 9126, como por exemplo, personalizações; gerenciamento de incidentes e problemas; SLA's; planos de continuidade; gerenciamento de dados; e auditabilidade. Abaixo, uma pequena análise desses quesitos:

- O critério Nível de Serviço não é possível ser avaliado através de um método como o MEDE-Pros, pois para sistemas convencionais (diferentes dos SaaS) características como SLA's, gerenciamento de dados, planos de continuidade, entre outros, não são contemplados em seu modelo de qualidade;
- Já o Apoio teve uma boa relação entre as características de testabilidade e estabilidade, porém, quando se trata de incidentes e personalização, as características entre os SaaS e sistemas convencionais são bem diferentes, causando assim divergências entre as avaliações de ambos os métodos.

Ao final deste comparativo conclui-se que, a partir das comparações efetuadas, fica claro que um método convencional não está adequado totalmente para avaliações de produtos SaaS. Algumas características foram bem avaliadas pelo método MEDE-Pros, porém outros tiveram sua avaliação prejudicada, devido ao contexto que estavam sendo empregadas. Isso se deve ao fato, dentre outros fatores, da utilização do modelo de qualidade.

No MEDE-Pros, como observado anteriormente, o modelo utilizado para aplicação das avaliações foi a norma ISO 9126, que por apresentar somente características de um produto de software convencional, não possibilita o uso do MEDE-Pros para avaliar dentro do contexto apropriado as características de um sistema SaaS.

De modo geral esta seção tentou complementar a validação do MAQSaaS. O comparativo efetuado com o MEDE-Pros identificou semelhanças e diferenças entre ambos os métodos, técnica no qual foi utilizada com sucesso em outros trabalhos científicos, Kitchenham et al. [1995]; Boegh et al. [1999]; Matinlassi [2004], onde foram comparados métodos de avaliação identificando em qual cenário de aplicação o método é mais eficiente e eficaz, identificando também semelhanças e diferenças entre ambos os métodos.

Capítulo 7

Conclusões

Neste capítulo serão apresentadas as conclusões finais sobre o método proposto, bem como suas contribuições nas áreas de pesquisa em Engenharia e Qualidade de Software. As seções seguintes apresentam as últimas considerações relacionadas ao MAQSaaS, incluindo um resumo de suas práticas, aplicações, limitações e possíveis trabalhos futuros.

7.1 Discussões sobre o MAQSaaS

Este trabalho apresentou a proposta de desenvolvimento de um método para avaliação da qualidade em um produto de software oferecido como serviço (SaaS), denominado MAQSaaS. O método proposto se baseou em um modelo de qualidade específico ao paradigma SaaS, propondo um processo de avaliação próprio, simples e eficaz.

Para a construção do modelo específico ao paradigma SaaS, e que pudesse ser utilizado na prática pelo MAQSaaS, houve uma integração entre critérios e princípios de métodos de avaliação da qualidade do produto de software, com critérios de métodos de gerenciamento de serviços de TI.

Os critérios definidos possibilitaram o levantamento e o impacto de critérios de qualidade específicos ao contexto dos SaaS, dividindo-os em grupos de características e sub-características, de maneira a mostrar seus impactos tanto positivos e negativos a este novo paradigma de software. O modelo faz uma compilação das principais características presentes em um produto de software e de um serviço de TI, contemplando critérios e requisitos próprios de um SaaS.

As avaliações propostas pelo método em questão são realizadas através de uma lista de verificação (*checklist*), de modo a contemplar através de atributos mensuráveis todos as características propostas pelo conjunto de qualidade.

7.2 Contribuições do Trabalho

Após a discussão sobre o método proposto, MAQSaaS, serão listadas e comentadas as principais contribuições que o presente trabalho teve para a área de "Qualidade de Software" e SaaS (*Software as a Service*).

As principais contribuições são:

- A proposta de um método de avaliação de qualidade para produtos de software oferecidos como serviços, denominado MAQSaaS;
- Levantamento bibliográfico de diversos métodos, metodologias e normas sobre a garantia de qualidade tanto em produtos de software como em serviços de TI, destacando seus pontos positivos, limitações e dificuldades de aplicação;
- Proposta de um conjunto de características de qualidade específico ao paradigma SaaS. O conjunto foi dividido em grupos de critérios e requisitos, de maneira a facilitar sua aplicação prática. Todos os critérios foram definidos e fundamentados através da visão de diversos autores da área de qualidade de software;
- Definição dos impactos positivos e negativos dos critérios de qualidade, juntamente com uma priorização e complementação dessas características, através de uma pesquisa com especialistas da área de sistemas SaaS. Os resultados deram maior confiança e fundamentação para a aplicação em avaliações reais, proporcionando maior relevância entre critérios e requisitos de qualidade;
- Mapeamento dos requisitos de qualidade em atributos mensuráveis, possibilitando sua pontuação e avaliação. O mapeamento foi realizado pelo Especialista da avaliação, se fundamentando em normas e autores da área de qualidade;
- Listagem de atributos mensuráveis através de uma lista de verificação (*checklist*), permitindo expressar a opinião dos Usuários Entrevistados em relação à qualidade pretendida;
- Definição de um processo de avaliação, definindo seus atores e papéis, juntamente com as suas etapas. O processo se mostrou eficiente podendo ser utilizado como guia em outras avaliações;
- Geração de *template's* de documentos propostos pela MAQSaaS, que orientam e padronizam a coleta de dados em uma avaliação prática;

- Avaliação de um produto SaaS, mostrando a qualidade final do produto. Além da avaliação, são listados aspectos de melhoria, principalmente para as características intrínsecas de um sistema SaaS;
- Comparação entre os métodos MAQSaaS e MEDE-Pros, mostrando as diferenças entre um método convencional e um específico aos SaaS.

Diante das contribuições propostas no presente trabalho, na próxima seção serão abordados os possíveis trabalhos futuros, de modo a dar continuidade nas pesquisas e desenvolvimento na área de qualidade de software.

7.3 Trabalhos Futuros

Há uma gama de possibilidades para a condução de estudos e trabalhos futuros a partir do trabalho desenvolvido. Considerando que a área de qualidade em produtos SaaS ainda é pouco explorada, muitas possibilidades de estudos podem ser consideradas, como até mesmo melhorias no método de avaliação (MAQSaaS) e estudos aprofundados sobre o assunto, gerando e proporcionando contribuições. Ao final do trabalho os autores destacam quatro aspectos principais que possibilitariam o prolongamento nos estudos e resultados obtidos no presente trabalho.

O primeiro aspecto se refere à proposta de um método que avalie e auxilie a garantia da qualidade associado ao processo de desenvolvimento de um produto SaaS. O processo de desenvolvimento de um software pode ter impacto direto sobre a qualidade de seus produtos finais. Neste contexto, devido as características intrínsecas presentes no paradigma SaaS algumas particularidades ligadas ao processo de desenvolvimento devem ser analisadas e estudadas. No trabalho de Cancian [2009] foi proposto um guia de referência de características para o processo de desenvolvimento, porém sem apresentar qualquer processo sistematizado de avaliação. A idéia do trabalho futuro seria incorporar os resultados obtidos no guia de referência juntamente com o processo de avaliação prescrito no MAQSaaS.

O segundo aspecto se refere à medição do MAQSaaS quanto à qualidade em uso. Durante a definição do problema foi dada uma ênfase maior na qualidade interna e externa, pelo motivo de serem essenciais ao escopo do problema. Para deixar o método mais completo e abrangente, melhorias na garantia da qualidade em uso poderiam ser incorporadas em seu escopo.

O terceiro se refere a uma complementação e aprofundamento da lista de verificação (*checklist*). No atual trabalho, foi dada ênfase em atributos que proporcionassem

uma avaliação geral da qualidade em um produto SaaS; todos estes atributos foram avaliados com o mesmo peso e pontuação. Para a aplicação do MAQSaaS em outros tipos de avaliações, a lista de verificação pode ser modificada dando maior ênfase a outros requisitos, mais especificamente associado a algumas características de qualidade, como por exemplo, Segurança, Níveis de Serviço, Suporte, entre outros.

Finalizando, avaliações adicionais podem ser realizadas principalmente em tipos diferentes de um produto SaaS, considerando outras categorias de software na aplicação do método MAQSaaS. Diferenças e novas limitações podem ser encontradas, podendo ser tratadas e incorporadas ao método, desse modo, possibilitando uma avaliação mais abrangente.

Algumas das possibilidades de trabalhos futuros destacados pelos autores já se encontram em etapa de planejamento para elaboração de novos trabalhos. As possibilidades exemplificadas acima, não são taxativas podendo outros autores criarem e definirem outros aspectos para trabalhos futuros, tendo como referência o presente trabalho.

7.4 Publicações Resultantes

- DUARTE FILHO, N. F.; PADUA, C. I. P. S.; ZAMBALDE, A. L.; BERMEJO, P. H. S. Atributos de qualidade para produtos SaaS. In: *5ª Conferencia Ibérica de Sistemas y Tecnologías de Información*, 2010, 2010, Santiago de Compostela-Espanha. CISTI' 2010, 2010.
- DUARTE FILHO, N. F.; PADUA, C. I. P. S.; ZAMBALDE, A. L.; BERMEJO, P. H. S. MAQSaaS - Método para Avaliação da Qualidade em Produtos de Softwares Oferecidos como Serviços. In: *Simpósio Brasileiro de Sistemas de Informação*, 2010, Marabá - PA. III Workshop de Teses e Dissertações em Sistemas de Informação (WTDSI), 2010.

Referências Bibliográficas

- Anderson, C. (2004). The long tail: Why the future of business is selling less of more. New York: Hyperion, 2004, issue 12.10.
- Aulbach, S.; Grust, T.; Jacobs, D.; Kemper, A. & Rittinger, J. (2008). Multi-tenant databases for software as a service: schema-mapping techniques. Proceedings of the 2008 ACM SIGMOD international conference on Management of data, Vancouver, Canada.
- Azuma, M. (2001). Square: the next generation of the iso/iec 9126 and 14598 international standards series on software product quality. In ESCOM (European Software Control and Metrics conference).
- Barney, S. & Wohlin, C. (2009). Software product quality: Ensuring a common goal. Lecture Notes in Computer Science, Springer. v.5543, p.256-267.
- Basili, V. & Musa, J. (1991). The future engineering of software: A management perspective. IEEE Computer Society Press Los Alamitos, CA, USA.
- Basili, V. R. & Weiss, D. M. (1984). A methodology for collecting valid software engineering data. IEEE Transaction on Software Engineering. v.10, n.6, p. 728-738.
- Belchior, A. (1997). Um modelo fuzzy para avaliação de qualidade de software. Tese de Doutorado do programa de engenharia de sistemas e computação, COPPE-UFRJ/Brasil.
- Bertoa, M. F. & Vallecillo, A. (2002). Quality attributes for cots components. in Proceedings of the 6th ECOOP Workshop on Quantitative Approaches in Object-Oriented Software Engineering (QAOOSE 2002), Spain.
- Beus-Dukic, L. & Boegh, L. (2003). Cots software quality evaluation. Proceedings of the Second International Conference on COTS-Based Software Systems, p.72-80, February 10-13.

- Boegh, J.; Depanfilis, S.; Kitchenham, B. & Pasquini, A. (1999). A method for software quality planning, control, and evaluation. *IEEE Software*, vol. 16, no. 2, pp. 69-85.
- Boehm, B. W. (1981). *Software engineering economics*. Prentice-Hall.
- Bon, J.; Pieper, M. & Van, D. V. A. (2006). *Foundations of it service management, based on itil*. A publication of ITSMF-NL. Publisher: Van Haren Publishing. 2^a edition.
- Borges, E. (2006). Lançamento e implantação da iso 20000. Disponível em: <http://www.brunise.com.br/informativo.asp?infID=22>. São Paulo. Brunise.
- Cancian, M.; Rabelo, R. & Wangenheim, C. (2009). Uma proposta para elaboração de contrato de nível de serviço para software-as-a-service (saas). In: *Proceedings of the 8th International Information and Telecommunication Technologies Symposium (I2TS'2009)*, Florianópolis.
- Cancian, M. H. (2009). Uma proposta de guia de referência para provedores de software como um serviço. *Dissertação de Mestrado em Engenharia de Automação e Sistemas*, Universidade Federal de Santa Catarina.
- Capovilla, I. G. G. (1989). Elementos intrínsecos do software e sua influência na qualidade do processo de desenvolvimento. *Dissertação-Instituto de Matemática, Estatística e Computação Científica, UNICAMP - Universidade Estadual de Campinas, Campinas*.
- Chong, F. & Carraro, G. (2008). Architecture strategies for catching the long tail. Disponível em: <http://msdn.microsoft.com/en-us/library/aa479069.aspx>.
- Chong, F.; Carraro, G. & Wolter, R. (2009). Multi-tenant data architecture. Disponível em: <http://msdn.microsoft.com/en-us/library/aa479086.aspx>.
- Choudhary, V. (2007). Softwares as a services: Implications for investment in software development. *IEEE IT Professional*, 40th Annual Hawaii International Conference on System Sciences.
- Clunie, C. E. (1997). Avaliação da qualidade de especificações orientadas a objeto. *Tese de Doutorado, Departamento de Engenharia de Sistemas e Computação-COPPE, Universidade Federal do Rio de Janeiro*.
- Colombo, R. & Guerra, A. (2002). The evaluation method for software product. *ICSSEA-International Conference. Software and Systems Engineering and their Applications*. Paris-França.

- Colombo, R. M. T. & Guerra, A. C. (2009). Tecnologia da informação: Qualidade de produto de software. Brasília: MCT/SEPIN, p. 429.
- Duarte Filho, N. F.; Padua, C. I. P. S.; Zambalde, A. L. & Bermejo, P. H. S. (2010). Atributos de qualidade para produtos saas. In: 5ª Conferencia Ibérica de Sistemas y Tecnologías de Información, , Santiago de Compostela-Espanha. CISTI.
- Dubey, A. & Wagle, D. (2006). Delivering software as a service. Disponível em: http://www.mckinseyquarterly.com/Delivering_software_as_a_service_2006.
- Gatti, M. A. C. & Werneck, V. B. (2004). Requisitos de qualidade de aplicações web. Cadernos do IME, Disponível em: <http://www.ime.uerj.br/cadernos/cadinf/>.
- Gil, A. C. (1991). Como elaborar projetos de pesquisa. ed. São Paulo, Atlas.
- Goh, D. H.; Chua, A.; Khoo, D. A.; Khoo, E. B.; Mak, E. B. & Ng, M. (2006). A checklist for evaluating open source digital library software. Online Information Review, Vol. 30, No. 4, pp. 360–379.
- Gomes, S. B.; Falbo, R. A. & Menezes, C. S. (2005). Um modelo para acordo de nível de serviço em ti. SBQS-Simpósio brasileiro de qualidade de software. Porto Alegre. Anais. p. 377-391.
- Goth, G. (2008). Software as a service: The spark that will change softwares engineering ? IEEE Distributed Systems Online. v. 9, n. 7, p. 3.
- Guerra, A. C.; Colombo, R. T.; Aguayo, M. T. V. & Peres, R. D. (2007). Modelo de qualidade de componentes de software. Conferência Ibero - Americana, Vila Real, Portugal. Conferência IADIS Ibero-Americana.
- Havelka, D.; Sutton, S. G. & Arnold, V. (1998). A methodology for developing measurement criteria for assurance services: An application in information systems assurance. Auditing: A Journal of Practice and Theory. Supplement.
- Herbert, L. (2008). Competing in the fast-growing saas market. Forrester paper.
- Herrmann, A. & Paech, B. (2008). Moqare: misuse-oriented quality requirements engineering. Requirements Engineering, v.13 n.1, p.73-86.
- Herssens, C.; Jureta, I. J. & Faullkner, S. (2008). Capturing and using qos relationships to improve service selection. Proceedings of the International Conference on Advanced Information Systems Engineering (CAiSE08).

- Hoffman, M.; Kuhn, N.; Weber, M. & Bittner, M. (2004). Requirements for requirements management tools. 12th IEEE International Requirements Engineering Conference (RE'04) Kyoto, Japan.
- Host, M. & Runeson, P. (2007). Checklists for software engineering case study research. Proceedings of the First International Symposium on Empirical Software Engineering and Measurement, p.479-481.
- Instituto-OnLine (2006). Entendendo e implementando a norma abnt nbr iso/iec 17799:2005: aspectos teóricos e práticos para a implantação da norma abnt iso/iec 17799:2005. Revisão 1.0. p. 1-84. Instituto OnLine.
- ISACA (2008). Information systems audit and control association. Serving IT Governance Professionals. Disponível em: <http://www.isaca.org/>.
- ISO/IEC 14598, . (1997). International standard. information technology - software product evaluation - part 1: General overview.
- ISO/IEC 14598-5, . (1997). International standard. information technology - software product evaluation - part 5: Process for evaluators.
- ISO/IEC 20000-1, . (2005). Tecnologia da informação - gerenciamento de serviços - especificação. IT Governance Institute.
- ISO/IEC 20000-2, . (2005). Tecnologia da informação - gerenciamento de serviços - código de prática. IT Governance Institute.
- ISO/IEC 25000, . (2005). Software product quality requirements and evaluation (square). Software Engineering, Guide to SQuaRE.
- ISO/IEC 9126, . (1991). The international organization for standardization. Software engineering Product quality.
- ITGI (2007). Information technology governance institute. COBIT 4.1: Control objectives, Management guidelines, Maturity models. Rolling Meadows: ITGI.
- ITIL v3, . (2008). Information technology infrastructure library. Publications Service Management. Disponível em: <http://www.best-management-practice.com/Online-Bookshop/IT-Service-Management-ITIL/ITIL-Version-3/>.
- itSMF (2007). It service management fórum brasil. Disponível em: <http://www.itsmf.com.br>.

- Jung, C. F. (2004). Metodologia para pesquisa e desenvolvimento: aplicada a novas tecnologias, produtos e processos. Rio de Janeiro/RJ: Axcel Books do Brasil Editora.
- Jureta, I. J.; Herzsens, C. & Faulkner, S. (2009). A comprehensive quality model for service-oriented systems. *Software Quality Control*. v. 17, p.65-98.
- Kan, S. H. (2002). Metrics and models in software quality engineering. 1ª ed. Boston: Addison-Wesley, p. 344.
- Kitchenham, B.; Pickard, L. & Pfleeger, S. (1995). Case studies for method and tool evaluation. *IEEE Software*, vol. 11, no. 4, pp. 52- 62.
- Laplante, P. A.; Zhang, J. & Voas, J. (2008). What's in a name? distinguishing between saas and soa. *IT Professional*. v.10, n.3, p.46-50.
- Lavado, M. (2009). Certificação de sistemas de gestão de serviços de ti em portugal. ITSMF - The IT Service Management Fórum.
- Lawes, A. (2003). Itil & itsmf-next steps in the global success story. Proc. of the SwissICT and itsMF, Conference on IT Service Management. Zürich.
- Lawlis, P.; Mark, K.; Thomas, D. & Courtheyn, T. (2001.). A formal process for evaluating cots software products. *IEEE Computer*, 34 (5): 58-63,.
- Lew, P.; Zhang, L.; Wang, S. & Jiang, W. (2008). Guidelines to determine quality for web-based software applications. *Computational Intelligence for Modelling Control & Automation (CIMCA)*. p.709-714.
- Ma, D. (2007). The business model of 'software-as-a-service'. *IEEE International Conference on Services Computing. SCC*, p.701-702.
- Macfarlane, I. & Rudd, C. (2005). Gerenciamento de serviços de ti. São Paulo: The IT Service Management Fórum. New Millenium Editora e Serviços Gráficos Ltda.
- Maintinguer, S.; Colombo, R. M. T. & Guerra, A. C. (2003). Adaptação do mede-pros para o processo de pré-qualificação do pnafm. Programa Brasileiro da Qualidade e Produtividade em Software - PBQP. Campinas, Brazil.
- Marconi, M. A. & Lakatos, E. (2003). Fundamentos de metodologia científica. Editora Atlas, São Paulo/SP.
- Martinez, M. R. M. (1999). The software product evaluation data base -supporting mede-pros. In: ISESS - International Software Engineering Standards Symposium - Best Software Practices for the Internet age, 4, Curitiba. Anais da 4 ISESS.

- Matinlassi, M. (2004). Comparison of software product line architecture design methods: Copa, fast, form, kobra and qada. Proc, of the 26th Int' Conf. Software Eng. Edinburgh, Scotland.
- MEDE-Pros (1996). Método de avaliação da qualidade de produto de software. Centro de Tecnologia da Informação Renato Archer-CTI/MCT. LAPS-Laboratório de Tecnologia de Avaliação de Qualidade de Produto de Software. LAQS-Laboratório de Avaliação de Qualidade de Software.
- Miguel, P. A. C. & Salomi, G. E. (2004). Uma revisão dos modelos para medição da qualidade em serviços. *Revista Produção*, v. 14, n. 1, p. 12-30.
- Miranda, L. (2006). Gestão de serviços de ti. *Insight SINFIC. Newsletter*, n.60, v.4.
- O'Brien, L.; Bass, L. & P., M. (2005). Quality attributes and service-oriented architectures. Technical report, Carnegie Mellon University.
- OGC (2008). Office of government commerce. OGC Home. Disponível em: <http://www.ogc.gov.uk/>.
- Ozkaya, I.; Bass, L.; Sangwan, R. & Nord, R. (2008). Making practical use of quality attribute information. *IEEE Software*. v.25, n.2, p.25-33.
- Peppard, J. (2000). Customer relationship management (crm) in financial services. *European Management Journal* 18, 312-327.
- Rocha, A. R. C. (1983). Um modelo para avaliação da qualidade de especificações. Tese de Doutorado, PUC-RJ, Rio de Janeiro.
- Rosenberg, F.; Leitner, P.; Michlmayr, A.; P., C. & Dustdar, S. (2008). Towards composition as a service-a quality of service driven approach. Distributed Systems Group, Technical University Vienna.
- Santos, A. (1999). Estabelecimento de processos da iso/iec tr qualidade total. Dissertação-Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, Campinas, SP.
- Silva, E. L. & Menezes, E. M. (2000). Metodologia da pesquisa e elaboração de dissertação. Florianópolis.
- Silva, E. M.; Yue, G. K.; Rotondaro, R. G. & Laurindo, F. J. B. (2006). Gestão da qualidade em serviços de ti: em busca de competitividade. *Revista Produção*, vol.16, n.2, pp. 329-340.

- Sommerville, I. (2003). Engenharia de software. 6^a ed. São Paulo, Addison Wesley.
- Turner, M.; Budgen, D. & Brereton, P. (2003). Turning software into a service. IEEE Computer Society. Keele University, Staffordshire.
- Vergara, S. C. (2004). Projetos e relatórios de pesquisa em administração. 5^a ed. São Paulo: Atlas.
- Vieira, E. (2004). Governança de ti. Info Corporate. v.2, n.3, p.50-61.
- Wagner, S.; Deissenboeck, F. & Winter, S. (2008). Managing quality requirements using activity-based quality models. In: WoSQ '08: Proceedings of the 6th international workshop on Software quality, New York.
- Weill, I. T. (2009). Governance: How top performers manage it decision rights for superior results. Boston: Harvard Business School Press.
- Wohlin, C.; Runeson, P.; Host, M.; Ohlsson, M. C.; Regneel, B. & Wesslén, A. (2000). Experimentation in software engineering: An introduction. Kluwer Academic Publishers. 1. Software engineering 2. Computer software-evaluation.

Apêndice A

Convite para participação da entrevista

Prezado (a),

Me chamo Nemésio Freitas Duarte Filho, sou aluno de mestrado da Universidade Federal de Minas Gerais (UFMG), e estou trabalhando em um projeto denominado "MAQSaaS - Método para Avaliação da Qualidade em Produtos de Softwares Oferecidos como Serviço", orientado e co-orientado respectivamente pelos professores Clarindo Isaías Pereira da Silva Pádua (UFMG) e Andre Luiz Zambalde (UFLA).

O trabalho em questão tem como objetivo principal propor um método para avaliar a qualidade em um produto de software oferecido como serviço, destacando seus principais requisitos de qualidade proporcionando uma eficiente avaliação de produtos SaaS (*Software as a Service*).

Venho através desta notificação, convidá-lo a participar de uma pesquisa qualitativa sobre requisitos de qualidade para SaaS. Esta pesquisa é extremamente rápida, tendo um tempo médio de 10 minutos, sendo muito relevante ao trabalho proposto, pois visa uma priorização e complementação dos requisitos de qualidade aplicados aos SaaS. O questionário da pesquisa se encontra em anexo no e-mail enviado, intitulado 'questionarioSaaS.doc'

Os requisitos apresentados no questionário em anexo foram levantados em estudos realizados em uma etapa anterior do trabalho. A sua ajuda será muito relevante, para a complementação do trabalho, vale lembrar que todas as informações pessoais serão mantidas em sigilo. No mais, agradeço sua atenção e paciência.

Atenciosamente.

Nemésio Freitas Duarte Filho

Apêndice B

Questionário Requisitos de Qualidade (SaaS)

Este questionário tem por objetivo priorizar e complementar os requisitos de qualidade levantados até o momento no presente trabalho. Cada característica foi decomposta em sub-características. Cada sub-característica deve ser enumerada de acordo com a legenda abaixo, e se por ventura achar necessário atribuir um novo requisito de forma a complementar a pesquisa:

[4-Muito relevante]

[3-Relevante]

[2-Pouco relevante]

[1-Não aplicável]

Segue abaixo uma pequena explicação e descrição de cada pontuação.

4-Muito relevante: Característica importante ao sistema, sendo de extrema importância a garantia de sua qualidade, inferindo diretamente na qualidade externa e interna dos produtos SaaS, principalmente em sistemas críticos das organizações;

3-Relevante: Característica importante principalmente para a qualidade externa do sistema, vinculada mais para o lado do usuário final, ela é importante na qualidade final do produto;

2-Pouco relevante: Característica no qual seja desejável sua qualidade, mas não essencial para o sucesso da aplicação;

1-Não aplicável: Característica no qual não se aplica ao contexto SaaS. Existem aspectos de qualidade que são aplicados e mensurados em softwares convencionais, porém no contexto SaaS algumas podem não ser aplicáveis e tão menos mensuráveis.

Priorização

Funcionalidade

(__) - **Adequação** [mede o quanto o conjunto de funcionalidades é adequado às necessidades do usuário]

(__) - **Acurácia** [representa a capacidade do software de fornecer resultados precisos ou com a precisão dentro do que foi acordado]

(__) - **Interoperabilidade** [trata da maneira como o software interage com outro(s) sistema(s) especificados]

Confiabilidade

(__) - **Maturidade** [entendida como sendo a capacidade do software em evitar falhas decorrentes de defeitos no software]

(__) - **Tolerância a falhas** [representando a capacidade do software em manter o funcionamento adequado mesmo quando ocorrem defeitos]

(__) - **Recuperabilidade** [foca na capacidade de um software em se recuperar após uma falha]

Usabilidade

(__) - **Inteligibilidade** [representa a facilidade com que o usuário pode compreender as suas funcionalidades]

(__) - **Apreensibilidade** [identifica a facilidade de aprendizado do sistema para os seus potenciais usuários]

(__) - **Operacionalidade** [modo como o produto facilita a sua operação por parte do usuário, incluindo a maneira como ele tolera erros de operação]

(__) - **Atratividade** [envolve características que possam atrair um potencial usuário para o sistema]

Eficiência

(__) - **Comportamento em relação ao tempo** [avalia se o tempo de resposta (ou de processamento) está dentro das especificações]

(__) - **Utilização de recursos** [mede tanto os recursos consumidos quanto a capacidade do sistema em utilizar os recursos disponíveis]

(__) - **Robustez** [serviços devem ser providos com alto grau de robustez. Robustez representa até que ponto um serviço pode trabalhar sobre a presença de dados inválidos ou incompletos]

Manutenibilidade

(__) - **Analisabilidade** [identifica a facilidade em se diagnosticar eventuais problemas e identificar as causas de deficiências ou falhas]

(__) - **Modificabilidade** [caracteriza a facilidade com que o software pode ser modificado]

(__) - **Estabilidade** [avalia a capacidade do software em evitar efeitos colaterais decorrentes de modificações introduzidas]

(__) - **Testabilidade** [representa a capacidade de se testar o sistema modificado, tanto em relação a novas funcionalidades quanto as não afetadas diretamente pela modificação]

Portabilidade

(__) - **Adaptabilidade** [representa a capacidade do software em se adaptar a diferentes ambientes sem a necessidade de ações adicionais]

(__) - **Instalação** [identifica a facilidade de se instalar o sistema em um novo ambiente]

(__) - **Coexistencia** [mede o quão facilmente um software convive com outros sistemas funcionando ao mesmo tempo]

(__) - **Capacidade para substituir** [representa a capacidade que o software tem de substituir outro sistema especificado]

Nível de Serviço

(__) - **SLA´s** [acordos de níveis de serviços permitem manter um bom relacionamento entre cliente e o fornecedor, melhorando a especificação e o entendimento dos requisitos do software]

(__) - **Plano de continuidade** [estratégias e procedimentos permitindo planos de recuperação gradual, recuperação intermediária e recuperação imediata para sistemas SaaS]

(__) - **Gerenciamento de dados** [estratégias e procedimentos para migração de dados; e serviços de relatórios dos dados]

(__) - **Auditabilidade** [é um fator de qualidade que representa o grau em que um sistema/software mantém registros suficientes para suportar auditorias financeiras ou jurídicas]

Segurança

(__) - **Confiabilidade** [permite que o acesso à informação/serviço só pode ser concedido a usuários autorizados]

(__) - **Integridade** [nos SaaS as informações trafegam pela web tanto do lado do usuário como do lado do servidor, é importante que essas informações sejam íntegras]

(__) - **Disponibilidade** [capacidade com que os SaaS mantenham seu funcionamento *online* no servidor do fornecedor]

Desempenho

(__) - **Escalabilidade** [capacidade com que o software suporta um número específico de usuários na mesma seção]

(__) - **Expansibilidade** [expansibilidade pode significar a facilidade com que novos serviços podem ser adicionados/expandidos dentro de um sistema sem afetar os outros serviços]

(__) - **Utilização de recursos** [capacidade do produto de software de usar tipos e quantidades apropriados de recursos, principalmente relacionados à largura de banda consumida pela aplicação]

(__) - **Capacidade de infra-estrutura** [assegura que a capacidade da infra-estrutura está adequada às demandas do negócio, conforme a necessidade e o tempo especificado]

Complementação

(__) - **Nome do Requisito** [Breve descrição]

(__) - _____

(__)- _____

(__)- _____

(__)- _____

(__)- _____

(__)- _____

(__)- _____

(__)- _____

(__)- _____

(__)- _____

Apêndice C

Lista de Verificação (*Checklist*)

Produto avaliado:	
Método utilizado:	
Nome do Entrevistado:	
N. Avaliação:	
Local:	Data:

Tabela C.1. Dados da aplicação da avaliação

Informações Gerais

- Siga atentamente as instruções do avaliador.
- O tempo estimado da avaliação é em torno de 40 a 50 minutos.
- Seja objetivo e imparcial em relação às respostas.
- As informações dos entrevistados não serão divulgadas, preservando seu senso crítico.

As questões podem ser respondidas de acordo com as seguintes opções:

Opção	Definição
Sim	Para sentenças verdadeiras.
Não	Para proposições falsas.
Eficiente	Para proposições que cumprem de maneira rápida os objetivos.
Continua na próxima página...	

Opção	Definição
Regular	Para proposições que cumprem os objetivos, mas que poderiam apresentar melhorias.
Deficiente	Para proposições que cumprem os objetivos, mas apresentam alguns erros em sua execução.
Ineficiente	Para proposições que não que não cumprem os objetivos pretendidos.
NA (Não se Aplica)	Para proposições que fazem referência a um aspecto que não se enquadra no produto em avaliação.
AP (Avaliação Prejudicada)	Para proposições que o entrevistado não está em condições de avaliar. Pode ser por falta de recursos, de informações, ou conhecimento específico.

Tabela C.2: Informações para preenchimento do *checklist*.

As questões que receberem respostas como "NA" e "AP" deverão ser justificadas, com ajuda do Avaliador, de modo a facilitar a elaboração do relatório final da avaliação.

Algumas questões necessitam de um entendimento aprofundando do produto, por isso é relevante seguir as instruções do Avaliador, seguindo um roteiro de execução de funções, navegação de janelas e destaque para trechos relevantes da documentação do produto.

Questionário

Funcionalidade

1- As funções presentes no produto satisfazem as necessidades específicas do usuário?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

2- O sistema apresenta algum sistema de busca, facilitando a busca de funções específicas pelo usuário do sistema?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

3- O sistema informa ao usuário o que um "botão", menu, ícone ou caixa de diálogos faz ao posicionar o curso do mouse sobre ele, trazendo balões explicativos?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

4- Os resultados obtidos são coesos e objetivos em relação às necessidades dos usuários?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

5- As opções oferecidas pelo software estão todas funcionando corretamente? Em que nível você as classificaria?

(__) - Eficiente

(__) - Regular

(__) - Deficiente

(__) - Ineficiente

(__) - NA

(__) - AP

6- O sistema permite a integração com outros tipos de softwares?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

7- O sistema possibilita a exportação de dados em formatos específicos de documentos? (ex. documentos texto, vídeos, fotos, entre outros.)

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

8- Qual a avaliação do nível de operação/reconhecimento de diferentes tipos de arquivos.

(__) - Eficiente

(__) - Regular

(__) - Deficiente

(__) - Ineficiente

(__) - NA

(__) - AP

9- O fornecedor da solução SaaS oferece um teste gratuito por período de tempo, para que o usuário/organização teste as principais funcionalidades do produto, antes de fechar um SLA (acordo de nível de serviço)?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

10- O sistema apresenta ao administrador funcionalidades de estatísticas de utilização do produto, como por exemplo, gerenciamento dos usuários, itens criados, notificações, entre outros?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

Usabilidade

11- Como pode ser avaliada a padronização do sistema? (fotos, letras, cores, menus, entre outros)

- (__) - Eficiente
- (__) - Regular
- (__) - Deficiente
- (__) - Ineficiente
- (__) - NA
- (__) - AP

12- O sistema possui alguma demonstração de suas funcionalidades?

- (__) - Sim
- (__) - Não
- (__) - NA
- (__) - AP

13- Os usuários com diferentes tipos de experiências têm iguais possibilidades de obter sucesso com as funcionalidades estabelecidas?

- (__) - Sim
- (__) - Não
- (__) - NA
- (__) - AP

14- As denominações de títulos, menus, label's estão de acordo com o que eles representam no contexto do sistema?

- (__) - Sim
- (__) - Não
- (__) - NA
- (__) - AP

15- Qual é o grau de aprendizado que os usuários têm em relação ao sistema?

- (__) - Eficiente
- (__) - Regular
- (__) - Deficiente
- (__) - Ineficiente
- (__) - NA
- (__) - AP

16- O sistema possibilita ajudas através de tutoriais e manuais?

- (__) - Sim

(__) - Não

(__) - NA

(__) - AP

17- O sistema apresenta uma separação eficaz entre as áreas selecionáveis de um painel do menu, de modo a evitar e minimizar ações acidentais?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

18- O sistema pode ser usado e controlado de forma fácil e eficiente?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

19- Qual a avaliação em relação à interface "amigável" do sistema?

(__) - Eficiente

(__) - Regular

(__) - Deficiente

(__) - Ineficiente

(__) - NA

(__) - AP

20- Como são avaliadas as mensagens de erros geradas pelo sistema?

(__) - Eficiente

(__) - Regular

(__) - Deficiente

(__) - Ineficiente

(__) - NA

(__) - AP

Segurança

21- O acesso as informações/serviços é concebido somente a usuários credenciados?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

22- As contas dos usuários apresentam privilégios bem definidos, com vários níveis de acesso? (administrador, usuários, convidados, entre outros)

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

23- Utiliza o protocolo SSL (*Secure Sockets Layer*) e tunelamento através de VPN's ?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

24- O sistema faz uma contagem e bloqueio em relação à quantidade de vezes que o usuário tentou se logar e não conseguiu? Possibilitando um travamento da seção com liberação mediante a um administrador.

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

25- O sistema esteve disponível (ininterruptamente) durante a avaliação?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

26- Havendo falhas ao longo de uma transação (ex. desconexão), as informações inseridas no sistema não são perdidas?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

27- O sistema contém proteção (desabilita) contra *scripts* java ou similares? Sendo uma das principais causas de *malwares* e atividades mau-intencionadas.

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

28- As trocas de informações/arquivos entre o servidor e o cliente são criptografadas?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

29- As senhas dos usuários apresentam algum tipo de criptografia (chaves públicas e privadas) ou certificados?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

30- O sistema utiliza mecanismos de *cracker*? Possibilitando identificar senhas "fáceis" dos usuários do sistema, indicando este fato ao próprio usuário ou ao administrador do sistema.

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

Desempenho

31- O sistema pode ser usado por múltiplos usuários ao mesmo tempo (multi-inquilinismo)?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

32- Como o sistema se comporta em relação a utilização do multi-inquilinismo?

(__) - Eficiente

(__) - Regular

(__) - Deficiente

(__) - Ineficiente

(__) - NA

(__) - AP

33- O mesmo sistema pode ser utilizado por organizações diferente?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

34- O sistema possibilita a adição de serviços extras, que não estejam vinculados diretamente ao fornecedor?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

35- O sistema é protegido contra a inclusão de novos serviços? (podendo afetar a estrutura ou a lógica do sistema)

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

36- Existe alguma padronização/contrato para a inclusão de novos serviços?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

37- Como o sistema se comporta em relação ao desempenho de tempo de resposta?
(quanto tempo demora a processar um pedido ou uma transação)

(__) - Eficiente

(__) - Regular

(__) - Deficiente

(__) - Ineficiente

(__) - NA

(__) - AP

38- O sistema apresenta mecanismos de medição de banda larga, verificando se a banda larga utilizada pelos usuários está de acordo com a mínima exigida pelo sistema?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

39- O tipo e o volume de dados a serem transferidos pelo sistema independem da largura de banda utilizada?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

40- Como o sistema se comporta em relação à largura de banda disponível para as trocas de informações?

(__) - Eficiente

(__) - Regular

(__) - Deficiente

(__) - Ineficiente

(__) - NA

(__) - AP

Apoio(Suporte)

41- O sistema pode ser modificado de acordo com as necessidades do cliente?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

42- O atendimento a resolução de incidentes está disponível 24h por dia?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

43- O atendimento a resolução de incidente pode ser efetuado on-line?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

44- O atendimento a resolução de incidente pode ser efetuado por telefone?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

45- O atendimento é gravado ou protocolado?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

46- O custo do suporte técnico está incluído na assinatura de uso do produto?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

47- O sistema indica claramente qual é a sua versão atual, juntamente com as últimas atualizações/modificações realizadas?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

48- Como é o processo de resolução de problemas/incidentes? (em relação ao tempo médio que a empresa contratante se prontifica em resolver um erro no sistema)

(__) - Eficiente

(__) - Regular

(__) - Deficiente

(__) - Ineficiente

(__) - NA

(__) - AP

49- O sistema apresenta mecanismo contra erros voluntários/involuntários dos usuários? Como por exemplo, entrar com dados inválidos em determinados campos.

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

50- Como o sistema se comporta durante a ocorrência de um erro grave?

(__) - Eficiente

(__) - Regular

(__) - Deficiente

(__) - Ineficiente

(__) - NA

(__) - AP

Nível de Serviço

51- O sistema apresenta um acordo de nível de serviço (SLA) com a empresa contratante?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

52- Como o SLA é definido em relação a implicações legais ao fornecedor? Caso haja descumprimento de normas estabelecidas.

(__) - Eficiente

(__) - Regular

(__) - Deficiente

(__) - Ineficiente

(__) - NA

(__) - AP

53- No SLA, é definido o custo da aquisição em relação aos serviços oferecidos e os requisitos especificados no sistema?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

54- O SLA deixa claro a responsabilidade por parte da empresa fornecedora sobre o sigilo dos dados de aplicações críticas da empresa contratante?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

55- O sistema possui algum plano de recuperação gradual, intermediário ou imediato?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

56- A aquisição de planos de recuperação já está incluso nas taxas financeiras cobradas pela organização fornecedora?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

57- O cliente da aplicação pode fazer *backup* 's dos dados que estão alocados nos servidores da empresa fornecedora?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

58- O sistema apresenta algum tipo de relatórios com os dados críticos da empresa?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

59- A empresa fornecedora possibilita a migração dos dados através de *script* 's? Sendo necessário quando o cliente desejar migrar para outro sistema.

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

60- O sistema apresenta um processo bem definido para aplicações de auditorias fiscais e jurídicas? (ex. Norma SAS 70)

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

Portabilidade

61- O sistema pode ser utilizado de maneira eficiente e correto em diferentes navegadores (*browser*)?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

62- Como o sistema se comporta sendo executado e acessado em *browser*'s diferentes? (ex. Internet Explorer, Mozilla Firefox, Google Chrome, entre outros)

(__) - Eficiente

(__) - Regular

(__) - Deficiente

(__) - Ineficiente

(__) - NA

(__) - AP

63- O sistema pode ser utilizado de maneira eficiente e correto em diferentes sistemas operacionais?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

64- Como o sistema funciona através de dispositivos móveis (ex. *BlackBerrys*, *iPhones*, *Smartphones*, entre outros)?

(__) - Eficiente

(__) - Regular

(__) - Deficiente

(__) - Ineficiente

(__) - NA

(__) - AP

65- O sistema permite uma navegação e utilização offline (através da sincronização de dados)? Retendo os dados do último acesso, para uma consulta rápida e prática, disponibilizando pelo menos funcionalidades básicas do sistema.

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

66- O sistema pode ser utilizado juntamente com outros sistemas em execução, utilizando o mesmo navegador?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

67- As trocas de informações são feitas por uma porta padrão de comunicação?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

68- O sistema possibilita o compartilhamento de recursos adicionais, como impressora, multimídia, memória, entre outros?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

69- O sistema pode ser utilizado com outro sistema SaaS sendo executado ao mesmo tempo?

(__) - Sim

(__) - Não

(__) - NA

(__) - AP

70- Como o sistema se comporta, sendo utilizado juntamente com outros sistema SaaS ?

(__) - Eficiente

(__) - Regular

(__) - Deficiente

(__) - Ineficiente

(__)-NA

(__)-AP

Apêndice D

Relatório Final da Avaliação

Este relatório tem por objetivo compilar todas as informações obtidas durante as avaliações efetuadas em relação ao produto SaaS "Salesforce CRM".

O relatório é constituído por alguns tópicos, abordando a avaliação geral da qualidade do produto avaliado, juntamente com as notas obtidas em cada critério de qualidade. Para cada característica avaliada é listado aspectos positivos e negativos encontrados durante as avaliações, com o objetivo de mostrar ao Requisitante os pontos necessários para a melhoria do produto.

Avaliação Geral

Ao final das avaliações, compilando e observando os dados obtidos, o produto avaliado teve uma nota geral igual a 76,6 se enquadrando no nível "Médio" de qualidade, Figura D.1.

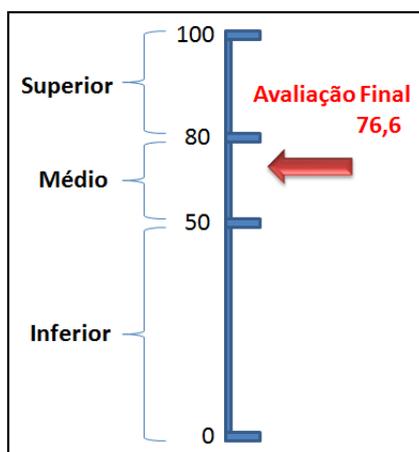


Figura D.1. Qualidade final alcançada pelo produto "SalesForce CRM Sales".

Analisando a nota obtida juntamente com a escala de pontuação, temos que, de modo geral o produto avaliado precisa de alguns ajustes. Ainda não se pode considerar que ele esteja de acordo com todos os requisitos exigidos pelas normas de qualidade. Apesar de apresentar uma nota de nível "Médio", o produto analisado mostrou um desempenho satisfatório na avaliação geral, principalmente aos diversos requisitos avaliados, ou seja, apresenta algumas características que estão de acordo com a qualidade pretendida.

Avaliação específica de cada critério

A Figura D.2 mostra a pontuação da qualidade obtida em cada critério avaliado.



Figura D.2. Qualidade em relação a cada critério do produto "SalesForce CRM Sales".

Os dados analisados na Figura D.2 permitem a conclusão de que ao final das avaliações os critérios Funcionalidade e Usabilidade tiveram as maiores pontuação na avaliação efetuada. Por outro lado, as características de Apoio (Suporte) e Nível de serviço tiveram as piores notas.

Este resultado tem um impacto negativo, pois ambos os critério, Apoio (Suporte) e Nível de Serviço representam características de maior relevância e impacto ao contexto SaaS. A seguir, para cada critério mostraremos os pontos positivos e negativos obtidos durante as avaliações.

Funcionalidade

Este critério foi avaliado visando à adequação, acurácia, interoperabilidade e acesso que o sistema proporciona aos usuários do sistema.

- Pontos Positivos:

A maioria das funcionalidades presentes no sistema satisfizeram de maneira correta e eficiente as necessidades dos usuários. Funcionaram de maneira correta dentro do contexto do produto.

- Pontos Negativos:

Durante as avaliações, o sistema CRM teve algumas limitações na exportação de dados para formatos específicos de documentos, tendo também uma limitação no reconhecimento de tipos diferentes de documentos, principalmente na inserção de fotos dos funcionários e exportação de relatórios.

Usabilidade

Este critério foi avaliado visando à inteligibilidade, apreensibilidade, operacionalidade e atratividade, garantindo uma usabilidade eficiente e amigável por parte dos usuários do sistema.

- Pontos Positivos:

A primeira vista o sistema chegou a "assustar" devido à variedade de opções disponíveis em sua tela. Porém analisando os quesitos de avaliação, o sistema se mostrou estar bem organizado em relação a abas de serviços e funções. De modo geral o produto analisado apresentou uma interface "amigável" e prática.

- Pontos Negativos:

Um ponto que prejudicou a avaliação foi a ausência de sistemas de demonstrações e tutoriais de uso, dificultando o contato de usuários inexperientes.

Segurança

Este critério foi avaliado levando em consideração aspectos mais técnico como a garantia da confiabilidade, integridade e disponibilidade dos dados, juntamente com a capacidade de aspectos criptográficos.

- Pontos Positivos:

O sistema apresenta diversos mecanismos funcionais de segurança, agregando confiabilidade e integridade para as informações.

- Pontos Negativos:

O aspecto relevante durante a avaliação deste critério foi o fato do sistema se encontrar indisponível durante uma das avaliações por motivos de manutenção. Mesmo considerando que a indisponibilização do sistema foi verificada apenas uma vez, o fato deste sistema ser considerado como ferramenta crítica a organização pode causar impactos diretos nos objetivos e missão da organização. Outro fato negativo foi a ausência de um mecanismo que verificasse quantas vezes o usuários tentou se logar com uma senha errada, viabilizando o bloqueio da conta.

Desempenho

Este critério foi avaliado levando em consideração aspectos como a escalabilidade, expansibilidade e utilização dos recursos por parte do sistema e usuários.

- Pontos Positivos:

O sistema se mostrou eficiente na utilização do multi-inquilinismo; e sua utilização pode ser feita por organizações diferentes.

- Pontos Negativos:

Apresentou problemas na inclusão de serviços adicionais ao sistema, não apresentando uma interface de contrato bem definida, dificultando por parte dos gestores a inclusão de novos serviços.

Apoio (Suporte)

Este critério foi avaliado visando à capacidade de personalização, incidentes/problemas, testabilidade e estabilidade do sistema.

- Pontos Positivos:

Apresentou um processo eficiente para resolução de incidentes e problemas, possibilitando os usuários verificarem seus status ao longo do processo. Todo processo de erros é protocolado facilitando o acompanhamento dos gestores e usuários. O sistema possibilita pequenas modificações e customizações feitas pelo próprio administrador. São algumas customizações limitadas, mas que ao mesmo tempo ajudam a deixar o sistema mais usual ao perfil dos usuários.

- Pontos Negativos:

O atendimento ao suporte técnico não estava disponível online, e outro problema que se agrava é que não foi verificado um suporte técnico disponível no idioma nativo que a aplicação estava sendo utilizada, apenas os idiomas inglês e espanhol estavam disponíveis.

Nível de serviço

Este critério foi avaliado tendo como base a abordagem de SLA's, planos de continuidade, gerenciamento de dados e auditabilidade.

- Pontos Positivos:

O sistema está acordado com diversas questões de auditoria jurídicas e financeiras, principalmente tendo suporte a auditoria SAS 70, que está sendo amplamente utilizada em todas as organizações fornecedoras e contratantes de soluções SasS. Outro aspecto positivo é o fato do sistema avisar aos usuários sempre que houver uma indisponibilização em virtude de manutenções. Este aviso geralmente é feito com antecedência e as manutenções acontecem geralmente nos finais de semana, possibilitando que a organização e os gestores se organizem em virtude desta indisponibilização.

- Pontos Negativos:

Os resultados mostraram diversos pontos negativos. Inicialmente o produto apresenta um SLA (acordo de nível de serviço), porém este SLA não é bem definido, possuindo ambigüidades e não deixando claras as implicações legais que a organização fornecedora deveria ter durante o descumprimento de cláusulas essenciais ao funcionamento do sistema. Outro fato relevante é a questão do gerenciamento dos dados; a empresa garante que os dados estão seguros e íntegros, mas em nenhum momento ela fornece funcionalidade para gerenciamento desses dados na forma de relatórios, nem a possibilidade de *backup's* e migração para outros sistemas.

Portabilidade

Este critério foi avaliado tendo como prioridades os aspectos de adaptabilidade e coexistência com outros sistemas computacionais.

- Pontos Positivos:

O produto analisado teve bom comportamento na utilização com diferentes navegadores (*browser*), mantendo suas características e funcionamento em todos os navegadores testados, proporcionando uma flexibilidade maior por parte dos usuários durante a escolha dos navegadores.

- Pontos Negativos:

Um dos problemas encontrados foi em relação à portabilidade de dispositivos móveis. Alguns aparelhos móveis, compatíveis com o CRM avaliado de acordo com a documentação, foram analisados apresentando dificuldades na utilização. Não existia um manual e todo procedimento era baseado nas experiências próprias dos usuários. Os aparelhos que conseguiram se conectar ao sistema apresentaram uma qualidade inferior e desempenho abaixo do esperado.

Observações durante as avaliações

De maneira geral poucos problemas ocorreram durante a execução das avaliações. O processo se mostrou eficiente podendo ser aplicado de maneira correta juntamente com o modelo de qualidade proposto.

O único fato mais equivocado foi o fato de que as primeiras avaliações gastaram mais tempo do que o estipulado para as avaliações. Isso se deve ao fato da inexperiência do avaliador durante as primeiras avaliações. A partir da terceira e quarta avaliação o avaliador tendo mais experiência com o método, pôde estipular melhor o tempo conseguindo executar as avaliações em até 50-60 minutos.

Apêndice E

Dados obtidos durante as avaliações

Este apêndice mostra detalhadamente os dados e pontuações obtidos ao longo das avaliações executadas.

A seguir, são exibidas 7 (sete) figuras que fazem referências aos sete critérios avaliados. Cada figura apresenta em sua primeira coluna a numeração das questões que foram aplicadas na lista de verificação. As demais colunas, enumeradas de 1 até 15, fazem referência ao número de avaliações efetuadas, seguida de sua pontuação atribuída pelos entrevistados. A última coluna mostra a média da pontuação da qualidade obtida pelo critério avaliado.

Na Figura E.1, temos os dados e a média da qualidade obtidos na avaliação em relação ao critério funcionalidade.

Questões	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Média
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
4	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
5	1	1	1	1	1	1	1	0.75	1	1	1	1	1	0.75	1	
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
7	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
8	0.75	0.75	0.75	1	1	1	0.75	0.75	1	1	1	0.75	0.75	0.75	0.75	
9	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
10	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
Soma =	8.75	8.75	8.75	9	9	9	8.75	8.5	9	9	9	8.75	8.75	8.5	8.75	8.816667

Figura E.1. Dados obtidos através da avaliação do critério funcionalidade.

Na Figura E.2, temos os dados e a média da qualidade obtidos na avaliação em relação ao critério usabilidade.

Logo abaixo, Figura E.3, podemos verificar os dados e a média da qualidade obtidos na avaliação em relação ao critério segurança.

Questões	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Média
11	1	1	1	1	1	1	1	1	1	1	1	1	1	0.75	0.75	
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
13	1	1	1	1	1	1	0	1	1	0	1	1	1	1	1	
14	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
15	0.75	0.75	1	1	1	1	0.75	0.75	1	0.75	0.75	1	1	1	1	
16	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
17	1	1	1	1	0	0	1	1	1	1	1	1	1	0	1	
18	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
19	1	1	1	1	1	1	1	1	0.75	1	1	1	1	1	1	
20	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
Soma =	8.75	8.75	9	9	8	8	7.75	8.75	8.75	7.75	8.75	9	9	7.75	8.75	8.516667

Figura E.2. Dados obtidos através da avaliação do critério usabilidade.

Questões	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Média
21	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
22	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
23	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
25	1	1	1	1	0	1	1	1	0	1	1	1	1	1	1	
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
27	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
28	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
29	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
30	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
Soma =	8	8	8	8	7	8	8	8	7	8	8	8	8	8	8	7.866667

Figura E.3. Dados obtidos através da avaliação do critério segurança.

Na Figura E.4, temos os dados e a média da qualidade obtidos na avaliação em relação ao critério desempenho.

Questões	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Média
31	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
32	1	1	1	1	1	1	1	0.75	0.75	0.75	1	1	1	1	1	
33	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
34	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
35	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
36	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
37	1	1	1	1	1	1	1	0.75	0.75	0.75	1	1	1	1	0.75	
38	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
39	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
40	1	1	1	1	0.75	0.75	1	1	1	1	0.75	1	1	0.75	0.75	
Soma =	8	8	8	8	7.75	7.75	8	7.5	7.5	7.5	7.75	8	8	7.75	7.5	7.8

Figura E.4. Dados obtidos através da avaliação do critério desempenho.

Na Figura E.5, temos os dados e a média da qualidade obtidos na avaliação em relação ao critério apoio(suporte).

Questões	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Média
41	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
42	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
43	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
44	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
45	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
46	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
47	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
48	0.75	1	1	0.75	1	1	1	1	0.75	0.75	1	1	1	1	1	
49	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
50	1	1	1	1	1	1	1	1	0.75	1	0.75	1	1	0.75	0.75	
Soma =	5.75	7	7	6.75	7	7	7	7	6.5	6.75	6.75	7	7	6.75	6.75	6.8

Figura E.5. Dados obtidos através da avaliação do critério apoio(suporte).

Na Figura E.6, temos os dados e a média da qualidade obtidos na avaliação em relação ao critério nível de serviço.

Questões	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Média
51	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
52	0.25	0.5	0.25	0.75	0.5	0.75	0.5	0.75	0.25	0.75	0.5	0.25	0.25	0.25	0.25	
53	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
54	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
55	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
56	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
57	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
58	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
59	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
60	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
Soma =	6.25	6.5	6.25	6.75	6.5	6.75	6.5	6.75	6.25	6.75	6.5	6.25	6.25	6.25	6.25	6.45

Figura E.6. Dados obtidos através da avaliação do critério nível de serviço.

Na Figura E.7, temos os dados e a média da qualidade obtidos na avaliação em relação ao critério portabilidade.

Questões	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Média
61	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
62	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
63	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
64	0.25	0.25	0.75	0.5	0.5	0.25	0.5	0.5	0.75	0.5	0.75	0.5	0.5	0.5	0.25	
65	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
66	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
67	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
68	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
69	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
70	1	1	1	1	0.75	1	1	1	1	1	1	0.75	1	0.75	0.75	
Soma =	7.25	7.25	7.75	7.5	7.25	7.25	7.5	7.5	7.75	7.5	7.75	7.25	7.5	7.25	7	7.416667

Figura E.7. Dados obtidos através da avaliação do critério portabilidade.