

UNIVERSIDADE FEDERAL DA PARAÍBA
CENTRO DE CIÊNCIAS APLICADAS A EDUCAÇÃO
DEPARTAMENTO DE CIÊNCIAS EXATAS
BACHARELADO EM SISTEMAS DE INFORMAÇÃO

**GOVERNANÇA NO DESENVOLVIMENTO ÁGIL DE
SOFTWARE**

MÁRCIA AMÁVEL CABRAL
Orientador: Prof. Dr. HERMANN HRDLICKA

RIO TINTO - PB
2015

MÁRCIA AMÁVEL CABRAL

GOVERNANÇA NO DESENVOLVIMENTO ÁGIL DE SOFTWARE

Monografia apresentada para obtenção do título de Bacharel à banca examinadora no Curso de Bacharelado em Sistemas de Informação do Centro de Ciências Aplicadas e Educação (CCAEE), Campus IV da Universidade Federal da Paraíba.
Orientador: Prof. Dr. HERMANN HRDLICKA.

RIO TINTO - PB
2015

C117g Cabral, Márcia Amável.
Governança no desenvolvimento ágil de software. /
Márcia Amável Cabral. – Rio Tinto: [s.n.], 2015.
84f. : il.

Orientador(a): Prof. Dr. Hermann Hrdlicka.
Monografia (Graduação) – UFPB/CCAEE.

1. Software - desenvolvimento. 2. Governança em TI. 3.
Sistemas de informação.

UFPB/BS-CCAEE

CDU: 004.4(043.2)

MÁRCIA AMÁVEL CABRAL

GOVERNANÇA NO DESENVOLVIMENTO ÁGIL DE SOFTWARE

Trabalho de Conclusão de Curso submetido ao Curso de Bacharelado em Sistemas de Informação da Universidade Federal da Paraíba, Campus IV, como parte dos requisitos necessários para obtenção do grau de BACHAREL EM SISTEMAS DE INFORMAÇÃO.

Assinatura do autor: _____

APROVADO POR:

Orientador: Prof. Dr. HERMANN HRDLICKA
Universidade Federal da Paraíba – Campus IV

Prof. Dr. ADRIANA ZENAIDE CLERICUZI
Universidade Federal da Paraíba – Campus IV

Prof. Dr. WAGNER EMANOEL COSTA
Universidade Federal da Paraíba – Campus IV

RIO TINTO - PB
2015

Aos meus pais: Marinalva Amável e Severino Cabral que se dedicaram uma vida inteira para que eu pudesse estudar.

AGRADECIMENTOS

Agradeço primeiramente a Deus por me dar forças nos momentos mais difíceis dessa jornada e por reger silenciosamente os meus passos.

Aos meus pais pelo incentivo e apoio em todos esses anos pelas condições proporcionadas para a minha formação e desenvolvimento pessoal.

Ao meu orientador Hermann, gostaria de agradecê-lo por ter acreditado na minha caminhada.

Agradeço a minha banca orientadora: Wagner Costa e Adriana Clericuzi, pelas dicas valiosas que me foram dadas, essências para a construção desse trabalho.

À minha amiga e irmã Gabriela Pimenta que foi como um anjo ao longo dessa jornada.

Ao meu namorado Emileo Matos pelo amor, parceria e amizade de todo o tempo, sempre estando ao meu lado e me incentivando a perseverar nos meus sonhos.

À minha família e aos inúmeros colegas e amigos que ganhei ao longo do curso, os quais foram verdadeiros anjos em minha vida, agradeço pela amizade, incentivo e todo o auxílio que me deram ao longo dessa caminhada, que de uma forma ou de outra, contribuíram para a conclusão deste trabalho.

RESUMO

A governança é a chave do sucesso para as empresas que querem se destacar no mercado ajudando-as a orientar o progresso de seus casos. No entanto, as estruturas que suportam a aplicação da governança exigem grandes investimentos. Com este problema, surge um novo conceito conhecido como Governança Ágil - uma derivação do conceito de governança, que é utilizado para permitir a realização e manutenção de objetivos estratégicos dos negócios da organização fornecendo valor continuamente, de uma forma menos “burocrática”. O trabalho apresentado tem por objetivos identificar os problemas, desafios e fatores críticos de sucesso para a implementação da Governança Ágil no Desenvolvimento de Software adotando uma abordagem exploratória.

Palavras chave: Governança. Governança Ágil. Modelos de Maturidade. Metodologias Ágeis. Engenharia de *Software*.

ABSTRACT

Governance is the key to success for companies that want to stand out in the market, helping them to guide the progress of their cases. However, frameworks that support the implementation of governance generally require large investments. With this problem, a new concept known as Agile Governance - a derivation of governance, is applied to enabling the achievement and maintenance of strategic objectives of the organization's business by continuously delivering value in a less “bureaucratic” way. The work presented aims to identify problems, challenges and critical success factors for the implementation of Agile Software Development Governance in adopting an exploratory approach.

Keywords: Governance. Agile Governance. Maturity Models. Agile Methodologies. Software Engineering.

LISTA DE FIGURAS

Figura 1 – Modelo de solução de Desenvolvimento Ágil de Software	26
Figura 2- Ciclo de Vida do <i>Scrum</i>	30
Figura 3- Ciclo de vida do XP	33
Figura 4 – Estrutura da representação por estágio	37
Figura 5 - Níveis de maturidade por estágios	38
Figura 6 – Estrutura da representação contínua	39
Figura 7 – Níveis na representação contínua	40
Figura 8 – Níveis de maturidade do MPS.Br	41
Figura 9 – Interação entre elementos envolvidos pelo COBIT	46
Figura 10 – Domínios do COBIT	47
Figura 11 – Abrangência dos conceitos de governança	63
Figura 12 – Modelo de Governança Ágil Integrada	67

LISTA DE TABELAS

Tabela 1- Comparação entre os tipos de desenvolvimento	24
Tabela 2- Manifesto Ágil.....	25
Tabela 3-Vantagens e Desvantagens do XP e <i>SCRUM</i>	34
Tabela 4-Níveis de Maturidade do MR-MPS.....	43
Tabela 5 – Diferenças entre as representações contínuas e por estágios.....	50
Tabela 6-Comparação entre os Níveis de Capacidade e os Níveis de Maturidade	51
Tabela 7- Análise comparativa dos níveis de maturidade entre o CMMI e o MPS.BR.....	53
Tabela 8- Correlação entre os modelos CMMI e MPS-BR.....	54
Tabela 9 – Principais diferenças entre os modelos CMMI e MPS.Br	55
Tabela 10-Vantagens e Desvantagens do CMMI e MPS.BR.....	56
Tabela 11- Comparação entre CMMI e COBIT	57
Tabela 12-Elementos-chave da Governança integrada em um ambiente ágil.....	68

LISTA DE QUADROS

Quadro 1- Relação entre fases de pesquisas, capítulos, conteúdos e contribuições da monografia.....	21
Quadro 2 – Conceito e descrições de Governança de TIC	64
Quadro 3 Visão da Governança Ágil.....	69
Quadro 4 – Missão da Governança Ágil	69
Quadro 5 – Os valores da Governança Ágil propostos	70
Quadro 6 – Princípios da Governança Ágil.....	71

LISTA DE SIGLAS

AI	Adquirir e Implementar
CMMI	Capability Maturity Model Integration
COBIT	Control objectives for Information and related Technology
DS	Entregar e Suportar
GASD	Governance for Agile Software Development
ITGI	Information Technology Governance Institute
ISACA	Information Systems Audit and Control Association
ISO	Internacional Standards Organization
ITIL	Information Technology Infrastructure Library
IBGC	Instituto Brasileiro de Governança Corporativa
MANGVE	Modelo Ágil no Apoio à Governança em TIC
MMC	Modelo de Maturidade e Capacidade
MPS.BR	Melhoria de Processo do Software Brasileiro
ME	Monitorar e Avaliar
OTIC	Órgão de Tecnologia da Informação e Comunicação
PO	Planejar e Organizar
SEI	Software Engineering Institute
TI	Tecnologia da Informação
TIC	Tecnologia da Informação e Comunicação
XP	Extreme Programming

SUMÁRIO

RESUMO	VII
ABSTRACT	VIII
LISTA DE FIGURAS	IX
LISTA DE TABELAS.....	X
LISTA DE SIGLAS.....	XII
SUMÁRIO	XIII
1 INTRODUÇÃO	15
1.1 JUSTIFICATIVA E RELEVÂNCIA	18
1.2 METODOLOGIA DA PESQUISA.....	19
1.3 ESTRUTURA DO TRABALHO E CONTRIBUIÇÕES	20
2 ENGENHARIA DE SOFTWARE E A BUSCA POR AGILIDADE	23
2.1 METODOLOGIAS ÁGEIS DE DESENVOLVIMENTO DE SOFTWARE	25
2.2 MÉTODO COM FOCO EM GESTÃO: EXEMPLIFICANDO COM O SCRUM.....	27
2.3 MÉTODO COM FOCO EM ENGENHARIA: EXEMPLIFICANDO COM O XP	30
2.4 VANTAGENS E DESVANTAGENS DAS METODOLOGIAS ÁGEIS: XP E SCRUM	33
2.5 MODELOS DE MATURIDADE E CAPACIDADE (MMC).....	35
2.5.1 <i>Capability Maturity Model Integration – CMMI</i>	35
2.5.2 <i>Representação Contínua</i>	38
2.5.3 <i>Melhoria de Processo do Software Brasileiro - MPS.BR</i>	40
2.5.4 <i>Control objectives for Information and related Technology – COBIT</i>	43
2.6 DISCUSSÃO	49
2.6.1 <i>Comparação entre as Representações Contínua e por Estágios</i>	49
2.6.2 <i>Quanto aos níveis de Capacidade e de Maturidade</i>	50
2.6.3 <i>Comparação entre CMMI e MPS.BR</i>	52
2.6.4 <i>Vantagens e Desvantagens entre CMMI e MPS.BR</i>	55
2.6.5 <i>Comparação entre CMMI e COBIT</i>	57
3 GOVERNANÇA.....	59
3.1 GOVERNANÇA CORPORATIVA	60
3.2 GOVERNANÇA EM TIC.....	61
3.3 GOVERNANÇA ÁGIL EM TIC.....	65
3.3.1 <i>Métodos ágeis e a necessidade de governança</i>	65
3.3.2 <i>O Manifesto da Governança Ágil</i>	68

3.3.3	<i>Exemplo de Modelo de apoio a Governança Ágil em TIC: MANGVE.....</i>	72
3.3.4	<i>Conclusão do capítulo</i>	73
4	CONSIDERAÇÕES FINAIS.....	81
5	REFERÊNCIAS BIBLIOGRÁFICAS	82

1 INTRODUÇÃO

Este trabalho apresenta uma investigação sobre conceitos de governança e sua relação com a governança corporativa, governança de TIC e governança ágil; também apresenta fatores críticos de sucesso e problemas comentados pelos principais autores nessa linha de pesquisas. Por focar em duas metodologias ágeis: Scrum e XP, foi realizado um levantamento sobre os princípios presentes nessas metodologias; também se apresenta alguns modelos de maturidade para melhor entender os aspectos importantes para o desenvolvimento de software.

Empresas de software têm se aproveitado, cada vez mais, das oportunidades criadas na competição global, dos avanços da economia digital e da sofisticação dos meios de comunicação (PRIKLADNICKI, 2009). Mas para isso, elas precisam ser ágeis em seu negócio. Segundo o Gartner Group, “agilidade de negócio” é estar apto a responder rapidamente e eficientemente a mudanças no mundo dos negócios e, transformar essas mudanças em vantagem competitiva é o principal motivo para sua adoção (SCOTT, 2000).

Para serem bem sucedidas nessa agilidade, essas empresas também necessitam controlar seu ambiente de produção que envolve o desenvolvimento de softwares e outros produtos correlatos, que são imprescindíveis para se destacarem nesse competitivo mercado.

Observa-se que cada vez mais as organizações estão adotando a abordagem ágil como uma tática de sobrevivência nesses tempos economicamente turbulentos (CUMMINS, 2008). Sendo assim foi levado a uma série de opiniões interessantes avaliando quais atitudes e atributos seus times precisam para ser bem sucedido. Neste contexto, a agilidade de negócio é importante para “mudar o sentido do ambiente e responder eficientemente e efetivamente a essa mudança” (LUFTMAN et al., 1993).

O desenvolvimento de um projeto de software é um desafio complexo, pois envolve vários tipos de situações imprevisíveis a que está exposto, que incluem fatores internos e externos, como por exemplo: mudanças de tecnologias e/ou alterações que acontecem com frequência (SCHWABER; SUTHERLAND, 2011).

O desenvolvimento de softwares envolve diferentes processos encadeados e necessários para se atingir o nível de qualidade que espera obter. São etapas desse macro processo, fundamentadas na abordagem sistêmica: i) análise do sistema-objeto e o levantamento dos requisitos necessários; ii) o projeto do software específico, que envolve interfaces diversas

com o usuário, especificações de produção e outros; iii) o próprio, que pode ser assistido por diferentes ferramentas e metodologias de desenvolvimento; iv) a prototipagem; e v) os testes que antecedem a entrega do produto ao cliente final (O'BRIEN, 2001).

Surgiram ao longo do tempo algumas metodologias de boas práticas de desenvolvimento de software e guias para auxiliar a governança da Tecnologia da Informação e Comunicação -TIC surgiram junto com evolução da engenharia de software. O ITIL (*Information Technology Infrastructure Library*) e o COBIT (*Control Objectives for Information and related Technology*) são exemplos na área de governança.

Já o *Scrum* e o **XP** são alguns dos modelos de metodologias ágeis, as quais focam na redução de tempo, custos e documentação excessiva.

Segundo o Gartner Group, “agilidade de negócio” é estar apto a responder rapidamente e eficientemente a mudanças no mundo dos negócios e, transformar essas mudanças em vantagem competitiva é o principal motivo para sua adoção (SCOTT, 2000).

As metodologias ágeis têm contribuído significativamente no processo para o desenvolvimento de software, graças às suas boas práticas e valores adaptados nos conceitos de governança em TIC e o conceito de maturidade. Como a literatura sugere, a maturidade de uma firma afeta suas práticas adaptáveis (MATHIASSEN; VAINIO, 2007), e de acordo com Filho (2009), a maturidade de uma organização, tratando-se de Engenharia de Software, mede a capacidade técnica, gerencial e a competência que essa organização tem para o desenvolvimento de software.

Para definir o termo “maturidade” foi encontrado um conceito desenvolvido por um autor especializado em gestão de projetos, que, para ele, maturidade é “[...] o desenvolvimento de sistemas e processos que são por natureza repetitivos e garantem uma alta probabilidade de que cada um deles seja um sucesso. Entretanto, sistemas e processos repetitivos não são, por si só, garantia de sucesso. Apenas aumentam a sua probabilidade” (KERZNER, 2006 p. 45).

O que o autor quer explicar é que há níveis diferentes das relações entre processos e a qualidade de seus resultados, necessariamente desenvolvidos em termos de melhoria contínua tornando aqueles mais previsíveis. Seu propósito principal é garantir essa evolução na qualidade de processos e resultados.

Outra definição apresentada por Siqueira (2015) mais abrangente explica:

Maturidade, como uma extensão em que o processo é explicitamente definido, gerenciado, medido, controlado e eficaz. O conceito básico sob o termo maturidade é de que as organizações maduras fazem as coisas de modo sistemático e de que as imaturas atingem seus resultados graças aos esforços heroicos de indivíduos, usando abordagens que eles criam mais ou menos espontaneamente.

É importante distinguir entre os diferentes modelos de maturidade apresentados na literatura, por exemplo, o COBIT, CMMI, e outros, pois há um diferencial importante entre eles: enquanto uns focam na maturidade de processos, outros se orientam por níveis de capacidade, embora ambos se fundamentem nas melhores práticas aplicadas e utilizem atributos que expõem algumas características importantes e significativas de cada processo, no qual será possível avaliar o nível de maturidade e de capacidade dos mesmos.

O benefício principal do uso de um modelo de maturidade é permitir que uma organização possua métodos e processos avaliados segundo parâmetros existentes fundamentados, como retro mencionado, nas boas práticas existentes em determinada indústria, e de forma acumulativa. Esses parâmetros distinguirão os diferentes níveis de maturidade (ou capacidade) presentes na organização avaliada (APMG International, 2015).

Modelos de Maturidade avaliam sistematicamente e de forma abrangente a evolução em estágios de uma organização, por exemplo, analisando a relação entre pessoas, ferramentas e procedimentos e onde os processos são os elementos de integração e princípio de mudança evolutiva. Normalmente, esses modelos apresentam diferentes níveis ou “graus de melhoria de processo para um predeterminado conjunto de processos no qual todos os resultados esperados do processo e dos atributos dos processos são atendidos” (SOFTEX, 2006).

Já os modelos de Capacidade, dado resultados esperados analisam a capacidade de um processo e se o mesmo atinge o nível de eficiência, ou a habilidade do processo atingir os objetivos de negócio atuais ou futuros (SOFTEX, 2006).

A governança ágil tem se expandido cada vez mais nos últimos tempos, e, diante disso, não existem modelos que avaliem organizações em nível de maturidade ou capacidade no contexto da governança ágil.

1.1. Objetivos

Este trabalho busca compreender como o conceito de governança está inserido em diferentes metodologias de desenvolvimento, em especial, aquelas caracterizadas como ágeis. A partir disso, pretende-se:

- Levantar e caracterizar duas metodologias ágeis de desenvolvimento mais adotadas pelas empresas de desenvolvimento de softwares, apresentando um quadro explicativo sobre as principais diferenças.
- Conceituar governança, governança corporativa e governança em tic.
- Apresentar o conceito de governança ágil e seus elementos
- Identificar problemas, fatores críticos de sucesso e desafios para implantação de governança no desenvolvimento ágil de Software.

1.1 JUSTIFICATIVA E RELEVÂNCIA

Esta pesquisa tem relevância para as organizações, pois tem o intuito de identificar atributos que caracterizam os pontos importantes da Governança Ágil e consequentemente auxiliando na sua implantação, assim como a investigação da Governança no desenvolvimento ágil de software; de acordo com Luna (2011), a governança aplicada de forma ágil orienta por onde começar, o que adaptar e o que priorizar, proporcionando uma redução de custo, prazo e dependências externas da organização.

Segundo Ambler (2009), a governança não é algo que é comumente associado a projetos de desenvolvimento de software ágil. Todavia, projetos ágeis, como todos os outros projetos, podem e devem ser governados.

Esta pesquisa tem o intuito de contribuir na investigação da governança em desenvolvimento ágil de software assim como apresentar as métricas pertinentes, as quais são fundamentais para a gestão e controle de processos, pois em um processo ágil, não se procura definir um processo altamente especificado, mas sim a utilização de métricas para orientar o processo de adaptação (GREEN, 2007).

Umas das principais motivações para essa pesquisa é a possibilidade de auxiliar e contribuir nas disciplinas de Gestão da Informação e Engenharia de Software do Curso de Sistemas de Informação, assim como auxiliar a criação de modelos para avaliar a maturidade ágil das organizações, visto que, melhorar a possibilidade de empresas terem sua agilidade avaliada proporciona que a organização identifique maneiras apropriadas para introduzir essa agilidade no seu contexto (QUMER; HENDERSON-SELLERS, 2008).

Os resultados obtidos representam capacidades que organizações necessitam que sejam ágeis, visto que essas capacidades são os recursos que a empresa precisa para responder positivamente e tirar proveito das mudanças (ZHANG; VALOTA, 2005).

1.2 METODOLOGIA DA PESQUISA

Esta pesquisa tem caráter exploratório, sistemático e crítico. Exploratório porque envolve levantamento bibliográfico e análise de exemplos que estimulam a compreensão. Sistemática, pois deve exigir uma disciplina constante no processo de pesquisa; e crítica, por meio de abordagem objetiva para diminuir subjetividades por parte do autor (OATES, 2006).

De acordo com Sampieri, Collado e Lucio (1991) o processo de pesquisa envolve as seguintes atividades, que foram adotadas neste trabalho:

1. **Conceber a ideia do objeto de investigação:** Fase em que as ideias inicialmente são vagas e por esse motivo necessita de uma análise cuidadosa para que a pesquisa seja transformada em material organizado. Esta pesquisa englobou estudos de outros trabalhos semelhantes com o próprio tema “Governança Ágil em TIC” e Governança no Desenvolvimento Ágil de Software.
2. **Identificar o problema da pesquisa:** Após a pesquisa mais aprofundada sobre o tema analisado, a etapa seguinte é a reunião de ideias dessa pesquisa, a formulação do problema que é determinado pelo conhecimento que o pesquisador tem do assunto. Essa atividade se resumiu na pergunta da pesquisa e na justificativa para este trabalho.
3. **Elaborar a base teórica:** Depois da identificação da pergunta que serviu de base para esta pesquisa, o passo seguinte é o aprofundamento sobre o tema proposto e a busca de embasamento teórico para a pesquisa.
4. **Definir a estratégia e tipo de pesquisa:** Logo após a revisão dos textos para a pesquisa, o trabalho continuou com a pesquisa exploratória, que de acordo com Yin (2003) tem como finalidade desenvolver, esclarecer e modificar conceitos e ideias, com vistas à formulação de novas teorias, modelos e hipóteses pesquisáveis em estudos posteriores.
5. **Estabelecer as hipóteses da pesquisa:** Após a definição do problema base da pesquisa e a fundamentação teórica, devem-se determinar as hipóteses do problema de pesquisa.
6. **Definir o desenho de pesquisa e os métodos a serem utilizados:** O desenho de pesquisa é o processo prático, o qual será respondido quais metodologias serão usadas.

7. Determinação dos participantes: É necessária, após a seleção de participantes o método de pesquisa adotada, os quais tem o papel de tornar o processo de validação mais eficiente.

8. Formulário de Coleta de dados: Segundo Marconi e Lakatos (2003), é um instrumento de coleta de dados formado por uma série de perguntas ordenadas, as quais serão destrinchadas.

Análise de dados: Após os dados coletados, é feita uma análise dos dados que consiste na apresentação de contribuições para esta pesquisa. Considerou-se esta etapa como a análise do material pesquisado em fontes secundárias, pois não houve etapa de levantamento de dados primários.

10. Apresentação dos resultados: É a reunião dos resultados da investigação sob a forma de relatório e, no caso deste trabalho, a apresentação para a comunidade acadêmica, e avaliação por banca avaliadora de TCC.

Neste trabalho, as fases de estabelecimento de hipóteses, a determinação dos participantes e o desenvolvimento de formulário de pesquisa foram ignorados em função do caráter exploratório da pesquisa com o objetivo de compreender o assunto tratado.

1.3 ESTRUTURA DO TRABALHO E CONTRIBUIÇÕES

Este trabalho apresenta-se organizado conforme o Quadro 1 abaixo, que relaciona a diferentes fases da pesquisa com os capítulos, seus conteúdos e as diferentes contribuições que apresentam.

Quadro 1- Relação entre fases de pesquisas, capítulos, conteúdos e contribuições da monografia

Fase	Concepção	Pesquisa Teórica	Análise Teórica e de Resultados	Fechamento
Capítulo	1 INTRODUÇÃO	2 ENGENHARIA DE SOFTWARE	3 GOVERNANÇA	4 CONSIDERAÇÕES FINAIS
Conteúdo	Contextualização; Objetivos; Metodologia; Estrutura do trabalho	Metodologias ágeis ; Metodo com foco em gestão:SCRUM; Método com foco em engenharia; Modelos de maturidade	Governança Corporativa; Governança em TIC; Governança Ágil em TIC; Métodos Ágeis Manifesto da Governança Ágil	Reflexões finais Limitações do trabalho. Recomendações para futuros trabalhos.
Contribuição	Métodos, técnicas e critérios de pesquisa. Apresentação do estado atual do tema.	Apresentação das Vantagens e desvantagens das metodologias ágeis;	Apresentação de convergências ou divergências teóricas entre autores.	Consolidação da proposta de pesquisa

Fonte: Elaborado pela autora

Este trabalho está dividido em cinco Capítulos:

- No Capítulo 2 são apresentados conceitos relacionados às metodologias ágeis; e conceitos relacionados a modelos de maturidade, assim como uma discussão incluindo comparativos desses modelos e as vantagens e desvantagens da adoção dos mesmos e uma breve apresentação da metodologia desta pesquisa, destacando o planejamento do estudo e o método utilizado;
- No capítulo 3 são apresentados conceitos correlacionados à Governança, Governança Corporativa, Governança em TIC, Governança Ágil em TIC, Métodos Ágeis, Manifesto da Governança Ágil e um modelo de apoio à Governança ágil em TIC;
- O Capítulo 4 apresenta os resultados e discussões dos estudos conduzidos.

- O Capítulo 5 apresenta as considerações finais, incluindo as principais contribuições da pesquisa, limitações e direcionamento de estudos futuros, assim como as referências bibliográficas consultadas para a realização deste trabalho.

2 ENGENHARIA DE SOFTWARE E A BUSCA POR AGILIDADE

Com avanço da tecnologia, possibilitando a sociedade moderna o acesso a computadores cada vez mais modernos e eficientes com preços bem acessíveis, o mercado do software não poderia ficar para trás, é fundamental que também acompanhe o crescimento. Com isto, foi encadeada a "crise do software" a qual foi originada pela necessidade do crescimento de softwares com alta qualidade e complexidade.

O termo 'crise de software' vem sendo utilizado na indústria de software desde 1968, em uma época em que a Engenharia de Software praticamente não existia. O termo tem origem das dificuldades encontradas no desenvolvimento de software, dada à alta demanda e complexidade do processo (SOMMERVILLE, 2007).

Pensando em suprir cada vez mais as necessidades dos clientes (ou organizações), as empresas de desenvolvimento de software tem se preocupado cada vez mais em tornar mais ágeis seus processos produtivos e prevenir o atraso em seus projetos, evitando dessa forma estresses desnecessários na equipe envolvida, ou nos clientes, em função de imperfeição no projeto. Em complemento, a ausência de uma visão compartilhada ou comum é um dos principais fatores de falhas de projeto de software (QUMER2007).

Na busca por resultados mais eficazes na construção desses projetos, as empresas de TIC passam a adotar metodologias de desenvolvimento de software mais flexíveis e sujeitas a mudanças, as quais são chamadas de ágeis em contraposição às metodologias iterativas que são mais comumente usadas, as mesmas melhoram a interação entre o cliente e o sistema durante o desenvolvimento do projeto.

Os métodos ágeis são vistos, por gerentes e programadores, como forma de reduzir custos impostos por abordagens tradicionais de desenvolvimento de software. Por sua vez e para alguns, os métodos ágeis geralmente são percebidos como inadequados em situações de demandas complexas (QUMER, 2007).

As metodologias ágeis objetivam melhorar as metodologias iterativas, dando mais ênfase na participação dos envolvidos no projeto, com metodologias e princípios que as diferenciam em grande parte das metodologias tradicionais, ao permitir uma maior proximidade do cliente com o processo, dando foco a iterações reduzidas e testes automatizados.

Com a necessidade de desenvolvimento mais flexível, ágil e com menor custo em relação a métodos tradicionais mais comumente usados, o uso das metodologias ágeis vem crescendo gradativamente a cada ano, o que será demonstrado nos subcapítulos abaixo.

Na Tabela 1 é apresentado um resumo comparativo entre o desenvolvimento por planejamento (tradicional) e a abordagem ágil.

Tabela 1 Comparação entre os tipos de desenvolvimento

Metodologia Iterativa	Abordagem Ágil
Desenvolver com variadas habilidades.	Desenvolvedor ágil, educado, disposto e colaborador.
Níveis variáveis da capacidade de cliente.	Clientes mais representativos e autorizados.
Confiança em conhecimento documentado explícito.	Confiança em conhecimento interpessoal.
Requisitos conhecidos e estáveis.	Requisitos emergentes, mudança rápida.
Projetado para requisitos atuais e previsíveis.	Projetado para requisitos atuais.
Retrabalho e reestruturações de códigos são caros.	Retrabalho e reestruturações de códigos são baratos.
Equipe e produtos maiores.	Equipes e produtos menores.
Premia a garantia da qualidade obtida.	Premia o valor rápido obtido.

Fonte: Adaptado de Luna (2011)

Nota-se que as diferenças dizem respeito a mudanças no comportamento do desenvolvedor, a proximidade com o cliente, com projetos voltados para a atualidade do contexto, custos de retrabalho, o tamanho da equipe e a alteração de paradigma: enquanto a metodologia iterativa foca na qualidade, a ágil se concentra na busca de valor.

Como mencionamos, as metodologias ágeis tem como propósito o aprimoramento das metodologias iterativas, com a intenção de descentralizar a atenção dada ao processo em si e dar mais destaque a cooperação dos participantes do projeto.

Uma de suas características é a sua autonomia em relação à diminuição dos processos de iteração e testes automatizados, que sempre são enfatizados durante o processo. Além de permitir uma maior interação do cliente ao longo do processo, se diferenciando das definições das metodologias tradicionais.

A demanda crescente de um desenvolvimento maleável e de baixo custo se comparado aos métodos tradicionais utilizados no mercado de desenvolvimento, é um dos motivos da grande admissão e crescimento dos métodos ágeis.

Para Henderson-Sellers (2002) poucas organizações possuem capacidade psicológica ou técnica para adotar as abordagens ágeis de desenvolvimento de forma imediata, com êxito e em curto prazo; de modo geral são necessários alguns anos para que possam ser apreendidas.

Além do mais, para o autor, ser completamente ágil em aspectos de desenvolvimento pode não ser totalmente favorável: é preciso manter alguns elementos conhecidos e confiáveis das abordagens tradicionais dentro um projeto ágil geral.

Uma maneira que pode ser utilizada para se fazer isso seria através do método situacional de engenharia; porém, o engenheiro e o gerente de software podem sentir insegurança na identificação da melhor solução ágil, e em sua adoção gradual, buscando responder às perguntas: que bits escolher e dentre estes quais os mais adequados para cada situação específica; e como estimular o entusiasmo da equipe envolvida, dentre outras dúvidas possíveis (Syed-Abdullah et al., 2007), garantindo que o método adotado possa amadurecer e incentivar o crescimento das habilidades da equipe de desenvolvimento e não se deixar levar pelo desejo de sempre “resistir a mudança”(HENDERSON E SELLERS E SEROUR, 2005).

2.1 Metodologias Ágeis de Desenvolvimento de Software

A metodologia ágil nasceu a partir de reflexões de um grupo de programadores experientes e consultores em desenvolvimento de *software*, que se reuniu em Utah (Estados Unidos) para discutir uma série de mitos e práticas adotadas em abordagens tradicionais de Engenharia de *Software* e Gerência de Projetos.

Esse grupo, composto por 17 profissionais de diversas áreas de atuação, percebeu que as abordagens antigas utilizadas nos livros de Engenharia de *Software* eram complicadas quando utilizadas na prática. Embora cada participante tivesse práticas e teorias distintas sobre como fazer um projeto de *software* ter sucesso, todos concordavam que um conjunto de princípios dominava os projetos que davam certo. Compartilhando esses conhecimentos, o grupo criou um manifesto para encorajar melhores meios de desenvolver *softwares*, conhecido como Manifesto Ágil, que absorveu as principais ideias para expressar princípios que guiam as metodologias ágeis, descritos na Tabela 2 (BECK et al., 2001):

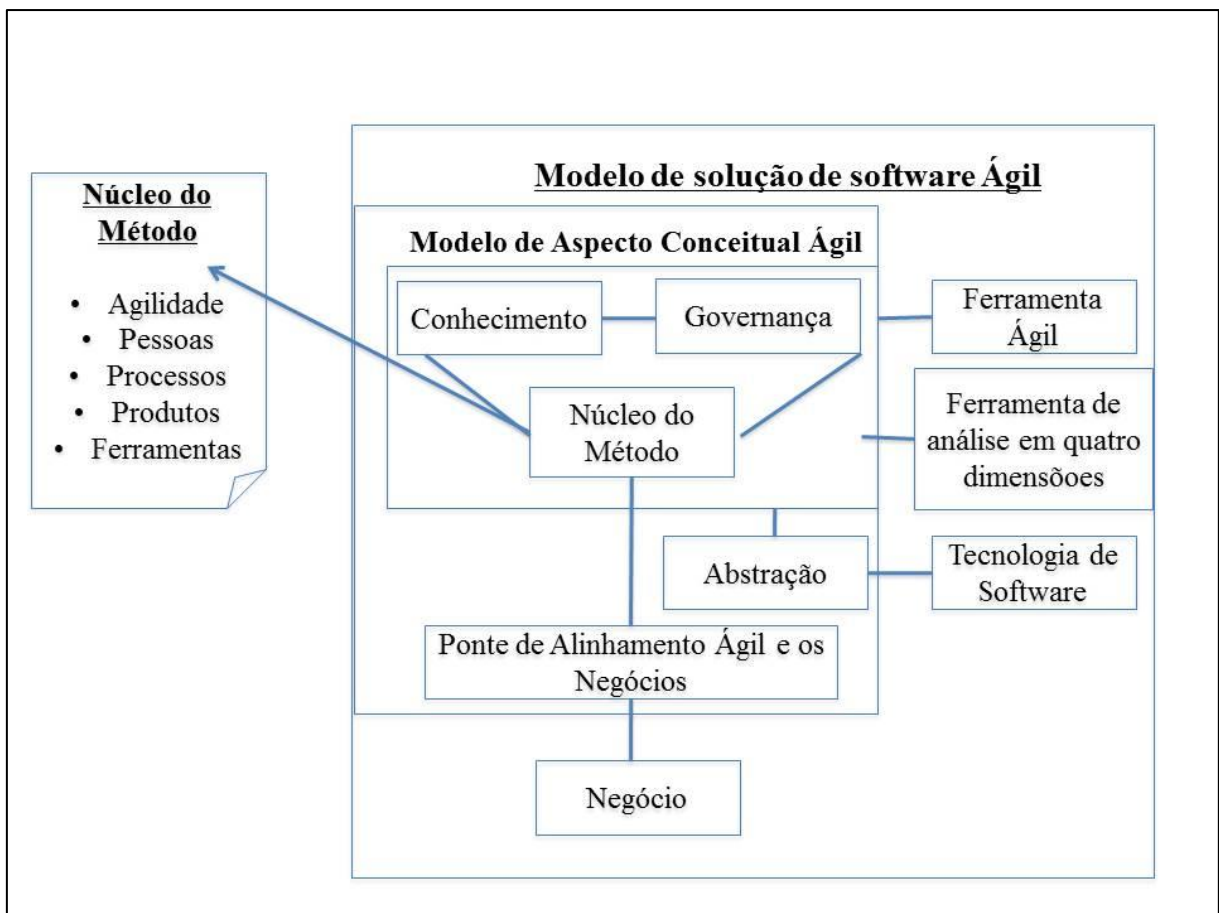
Tabela 2- Manifesto Ágil

Valorizamos	Mais que
Indivíduos e interações	Processos e ferramentas
Software funcional	Documentação extensa
Relacionamento com o cliente	Negociação de contrato
Responder a mudanças	Seguir um planejamento

Fonte: Adaptado de Kent (2001)

Atualmente existem diversos processos de desenvolvimento ágil de software tais como: Extreme Programming (**XP**), *Scrum*, Feature Driven Development, DSDM, Lean, Adaptive Software Development, Crystal, etc. Embora cada uma tenha as suas particularidades, todas elas compartilham as premissas do Manifesto Ágil. Um modelo comum a eles é apresentado por Qumer e Henderson-Sellers (2007) está adaptado na Figura1, onde serão destacados os elementos i) núcleo do método; ii) conhecimento, e iii) governança.

Figura 1 – Modelo de solução de Desenvolvimento Ágil de Software



Fonte: Adaptação de Qumer e Henderson-Seller (2007)

- Núcleo do método - O núcleo é composto por cinco elementos: agilidade, pessoas, processos, produtos e ferramentas. O elemento inovador é agilidade (velocidade) na entrega do produto, aqui visto como estratégia de obtenção de vantagem competitiva (BATEMAN, T.S. & SNELL, S.A.,pg. 36).
- Conhecimento - Segundo Qumer e Henderson-Sellers (2007), o desenvolvimento do

software absorve a transição de uma economia industrial para a economia do conhecimento. Para os autores,. o desenvolvimento ágil de software é um processo de conhecimento intensivo, onde o conhecimento é criado e compartilhado quando diferentes aspectos de uma metodologia (conceitos, produtos, ferramentas, processos, pessoas, etc.) interagem entre si. Ao se sentirem parte desse esforço, as pessoas cooperam, criam e compartilham o conhecimento.

- Governança - Governança ou, mais especificamente, a governança de TIC é fundamental porque proporciona um mecanismo ou o alinhamento estratégico do negócio visando adquirir o valor máximo de mercado entregue pelo consumo de recursos de TIC (QUMER e HENDERSON-SELLERS, 2007). Entende-se por valor “medida variável de importância que se atribui a um objeto ou serviço necessário aos desígnios humanos e que, embora condicione o seu preço monetário, frequentemente, não lhe é idêntico” (Dicionário Houaiss Online, 2007). Outra forma de definir valor é traduzi-lo como função entre benefícios obtidos pelos esforços para obtê-los.

Assim, o método ágil é função de agilidade, abstração, pessoas, processos, produtos, ferramentas, conhecimento, e governança, podem ser expressos pela seguinte fórmula (QUMER e HENDERSON-SELLERS, 2007):

$$\text{Metodologia Ágil} = f(\text{agilidade, abstração, pessoas, processos, produtos, ferramentas, conhecimento, e governança}).$$

A seguir, serão apresentados dois modelos ágeis de desenvolvimento: o *Scrum* e o XP.

2.2 Método com Foco em Gestão: Exemplificando com o *Scrum*

O método ágil *Scrum*, é constituído por uma aplicação da abordagem empírica da teoria de controle do processo industrial para o desenvolvimento de software, resultando em uma abordagem flexível, adaptativa e produtiva. Segundo Schwaber e Sutherland(2011), O *Scrum* trabalha de uma maneira que permita que a equipe trabalhe de uma forma flexível em um ambiente em constante mudança.

As regras do *Scrum* integram os enventos, papéis e artefatos que administram as relações e interações entre eles.

Existem 5 papéis no *Scrum* que desempenham diferentes tarefas e propósitos durante o processo e suas práticas: *Scrum Master*, *Product Owner*, *Scrum Team*, *Cliente* e *Gerência*. Estes papéis são definidos abaixo (SCHWABER; SUTHERLAND, 2011):

- **Scrum Master:** este papel é responsável por gerenciar o time e garantir o uso da metodologia, mas não é necessariamente um gerente, serve ao time e faz tudo para que o mesmo seja bem sucedido.
- **Product Owner:** O Product Owner representa a voz do cliente e é responsável por garantir que a equipe agregue valor ao negócio. o Profissional desempenhando este papel representa os stakeholders e define o que o produto deve ser, deve conhecer as necessidades do cliente e é responsável por garantir o retorno de investimento;
- **Scrum Team:** equipe de desenvolvimento multidisciplinar e auto gerenciável, é uma equipe responsável por desenvolver o produto com qualidade que agrade ao cliente de acordo com a visão do Product Owner;
- **Customers:** são os clientes participantes no projeto, os quais só estão diretamente envolvidos no processo durante as revisões Sprint;
- **Gerência:** São os profissionais que configuram o ambiente para o desenvolvimento de produtos. Este papel é responsável pelas decisões, padrões e convenções a serem seguidas pelo projeto.

As ferramentas como: Product Backlog, Sprint Backlog, Burndown Chart e *Scrum Board* são de grande utilidade durante o ciclo de vida do modelo, conforme descrito abaixo:

- **Product Backlog:** uma lista que evolui durante o projeto com todos os requisitos que o Product Owner deseja, sem muitos detalhes técnicos, ordenada de forma que as funcionalidades mais valiosas fiquem no topo;
- **Sprint Backlog:** contém uma lista com as tarefas sobre os itens extraídos do Product Backlog. Tal lista será definida no *Sprint Planning Meeting* e deverá ser entregue ao Product Owner na data determinada independente se o trabalho planejado foi finalizado;
- **BurnDownChart:** gráfico que mostra o trabalho da equipe dia a dia, avaliando as condições e complicações da Sprint. Caso o gráfico demonstre que a equipe está adiantada, o Product Owner é consultado e novas funcionalidades são acrescentadas àquela Sprint;

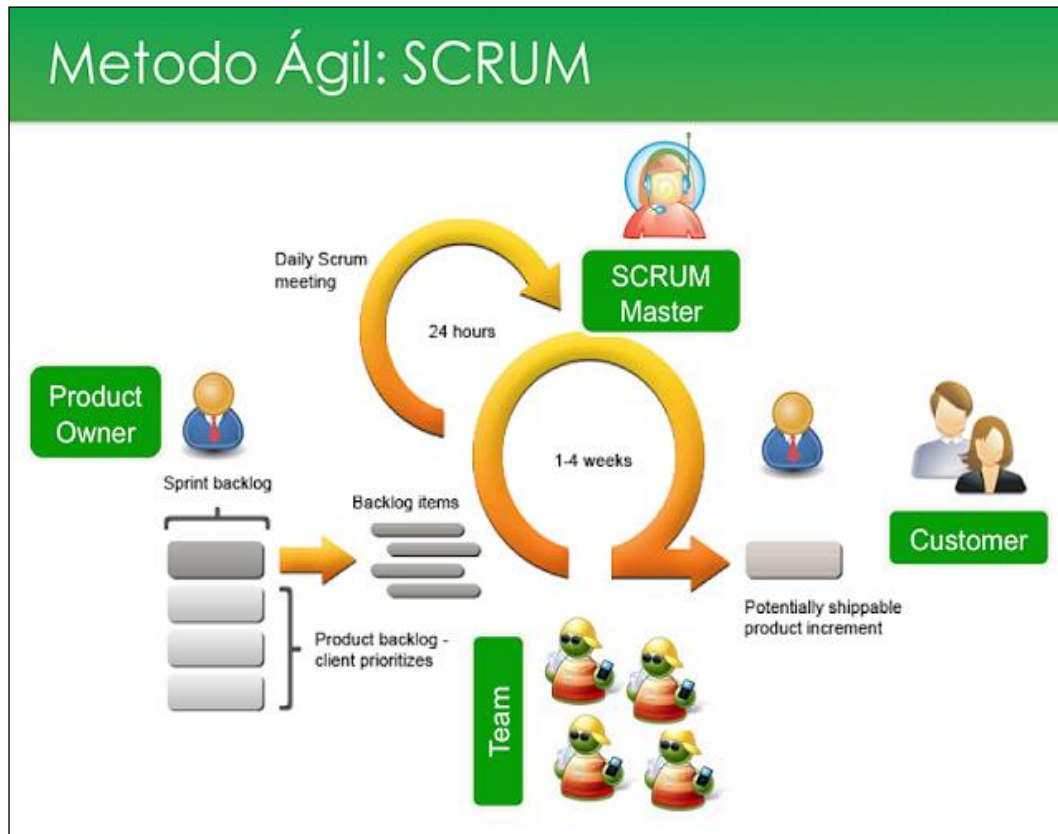
- **Scrum Board:** Quadro onde se pode visualizar o andamento de todas as tarefas que serão realizadas dentro de um Sprint e listadas, de acordo com as prioridades definidas pelo Product Owner.

Além das ferramentas, existem três tipos de cerimônias no *Scrum*, tais eventos são características importantes do ciclo de vida de cada Sprint. Segue abaixo uma breve descrição sobre as cerimônias:

- **Sprint Planning Meeting:** encontro para planejar o que será possível terminar nessa Sprint. A equipe responsável acessa o Product Backlog, avalia o tempo e complexidade das atividades que serão executadas no Sprint de acordo com suas prioridades e gera o Sprint Backlog;
- **Daily Scrum Meeting:** encontros diários, liderados pelo *Scrum Master*, com duração em média de 15 minutos, esta reunião gira em torno de 3 perguntas: O que fiz ontem? O que farei hoje? O que está impedindo de que alcance o objetivo?
- **Sprint Review Meeting:** encontro liderado pelo *Product Owner*, pode envolver todos os interessados no projeto e realizado ao fim de cada *Sprint*. Este encontro consiste em demonstrar ao *Product Owner* quais atividades definidas no *Product Backlog* foram realizadas, após tal demonstração o *Product Owner* e os interessados no projeto atualizam o *Product Backlog* redefinindo as prioridades, criando dessa forma o próximo Sprint. Na Segunda parte dessa reunião, o *Scrum Master* toma a liderança e começa uma reunião com a equipe para avaliar os pontos positivos e negativos do Sprint, colocando em foco o que poderia ser mudado para melhorar o próximo Sprint.

A Imagem abaixo exemplifica o ciclo de vida do *Scrum*, destacando os papéis e as cerimônias.

Figura 2- Ciclo de Vida do Scrum



Fonte: (ANDRES, 2012)

Explica-se as relações apresentadas na Figura 2 a partir das principais partes interessadas (*stakeholders*) no produto imaginado, respeitados os ciclos temporais de 24 horas e 1 a 4 semanas; o *sprint backlog* representa tudo o que deverá ser feito até o próximo Sprint (entrega ou versões) incluindo preparação, estimação e priorização; os *backlogs items* são os itens que serão desenvolvidos e entregues, para posterior incremento a partir da relação com o cliente. A equipe seleciona os itens do topo do *backlog* e estimam o tempo necessário para concluir as tarefas.

2.3 Método com Foco em Engenharia: exemplificando com o XP

XP, ou Programação extrema (do inglês *extreme Programming*), é uma das metodologias ágeis de desenvolvimento de software mais conhecidas e utilizadas para equipes pequenas e médias para desenvolvimento de software com requisitos vagos e em constante mudança. Possui este nome porque utiliza algumas boas práticas da engenharia de software de ma-

neira extrema para extrair os melhores resultados. Tal metodologia engloba vários valores, princípios e práticas que serão comentadas mais abaixo (BECK; ANDRES, 2004).

O principal objetivo do **XP** é assegurar que o cliente receba o máximo possível de desenvolvimento a cada dia de trabalho da equipe. Ele é formado baseado em valores e práticas que agem com harmonia e coesão para assegurar que o cliente sempre receba um bom retorno do investimento feito em implementar um Software na empresa. Nesse Contexto os quatro valores fundamentais que compõem o **XP** são, segundo Beck e Andres (2004):

- **Comunicação:** Muitos problemas são causados por falta de comunicação ou por uma comunicação mal feita. O objetivo é evitar muito esforço para o processo de troca de informações por meio de documentos oficiais que geralmente são mal interpretados;
- **Simplicidade:** Um projeto simples facilita a comunicação, a simplicidade deve garantir que o resultado do desenvolvimento apenas o suficiente para atender as necessidades do cliente, “enxugando” o sistema de qualquer funcionalidade não essencial;
- **Coragem:** devido ao **XP** utilizar várias metodologias que contrariam os processos tradicionais de desenvolvimento de software, é preciso coragem por parte da equipe de desenvolvimento para adotá-las.
- **Feedback:** conduzir o desenvolvimento de tal forma a fazer com que o cliente seja envolvido diariamente no processo de produção a fim de garantir que a equipe direcione o foco da produção para o que for considerado prioridade.

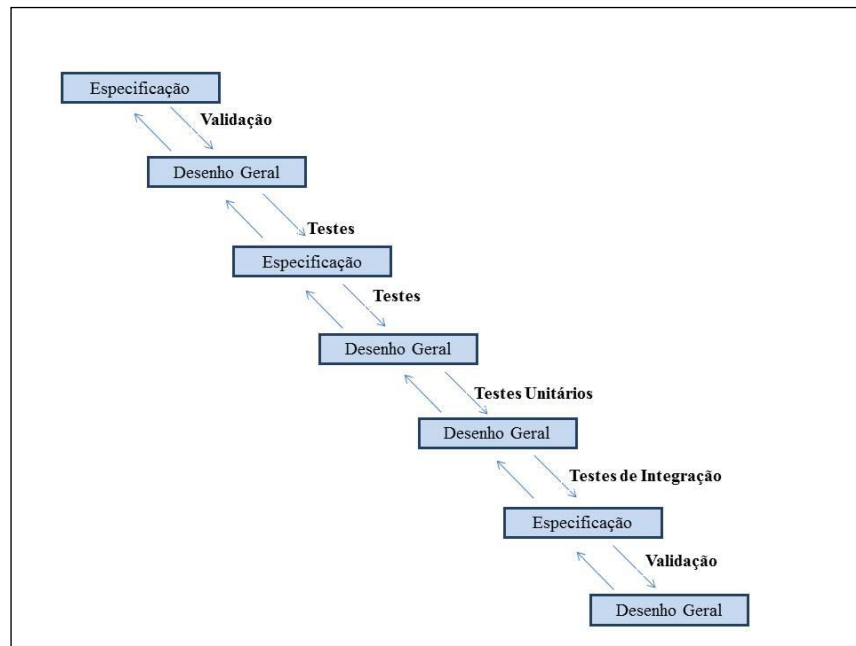
Abaixo segue a descrição de alguns pontos fortes do **XP** que auxiliam no processo de desenvolvimento (BECK; ANDRES, 2004):

- **Cliente Presente:** Os clientes presentes nos processos melhora a viabilidade e facilita a comunicação com a equipe de forma a permitir um *feedback* contínuo e rápido;
- **Jogo do Planejamento:** Reuniões com o cliente a cada nova versão com o objetivo de definir quais funcionalidades devem ser implementadas e as prioridades dessas funcionalidades;

- **Stand Up Meeting:** Reuniões com a equipe de desenvolvimento toda manhã com o objetivo de avaliar o trabalho executado no dia anterior e definir as prioridades do dia;
- **Refactoring:** Metodologia de desenvolvimento de *software* utilizada de forma a manter produto o mais simples possível, desta forma, possibilitando alterações no código com pouco impacto nas funcionalidades já implementadas;
- **Código Coletivo:** os códigos fonte do projeto devem ser de livre acesso a todos os desenvolvedores da equipe, permitindo alterações necessárias sem burocracia de autorizações;
- **Código Padronizado:** O código padronizado permite que o sistema seja o mais homogêneo possível deve ser um dos objetivos principais da equipe, estabelecendo padrões de codificação com o objetivo de facilitar a manutenção de código;
- **Metáfora:** Linguagem usada que é estabelecida entre a equipe e o cliente. Esta técnica facilita a comunicação nas reuniões de *feedback* com clientes que não tem conhecimento da área de *Software*;
- **Ritmo Sustentável:** Técnica em que a equipe trabalha apenas 8 horas por dia a fim de garantir o máximo de rendimento qualidade possível no processo de produção;
- **Design Simples:** Um sistema com design simples permite agilidade durante o desenvolvimento, visto que o *feedback* deve ser rápido ao cliente e a facilidade para desenvolvimento é maior;
- **Integração Contínua:** a equipe de desenvolvimento deve garantir a integração de códigos com o sistema sempre que for possível;
- **Releases Curtos:** disponibilizar novas funcionalidades rapidamente ao cliente é um dos princípios que deve ser aplicado no processo de produção para que o cliente possa utilizar o *software* diariamente sem maiores problemas;
- **Desenvolvimento Guiado pelos Testes:** A equipe de desenvolvimento deve escrever testes para cada funcionalidade antes mesmos da codificação fornecendo melhor entendimento das necessidades do cliente.

O Ciclo de vida do **XP** pode ser observado na Figura 3, exemplificando as fases mencionadas.

Figura 3- Ciclo de vida do **XP**



Fonte: Adaptado de (NAWROCKI; WALTER; WOJCIECHOWSKI, 2001)

Nota-se, na figura, que há um processo iterativo em que sempre ocorre a sincronização entre novas especificações e o desenho geral, com processos de validação, testes, testes unitários, testes de integração e validação final.

2.4 Vantagens e Desvantagens das Metodologias Ágeis: XP e Scrum

Esse tópico enfatiza vantagens e desvantagens das metodologias mais populares, levantando também algumas semelhanças e diferenças nas metodologias abordadas, e que são explicadas logo abaixo à Tabela 3.

Tabela 3-Vantagens e Desvantagens do **XP** e **SCRUM**

	XP	SCRUM
VANTAGENS	• Práticas técnicas fortes; • Cliente determina as prioridades. Desenvolvedores determinam as estimativas; • Várias oportunidades de feedback; • Técnica ágil mais adotada mundialmente.	• Complementa práticas existentes; • Times auto organizados e com feedback; • Participação e direção do cliente; • Prioridades baseadas no valor para o cliente; • Única abordagem que possui certificação.
DESVANTAGENS	• Requer cliente sempre presente; • Documentação se baseia apenas no código e na comunicação verbal; • Dificuldade para novas equipes adotarem as práticas num projeto já existente.	• Provê apenas suporte para gerenciamento de projetos; • Não especifica técnicas práticas; • Tempo necessário para o cliente definir as prioridades de cada requisito.

Fonte: (BECK; ANDRES, 2004)

Metodologias Ágeis são uma tentativa de refinar as metodologias iterativas, tirando o foco do processo em si e dando mais ênfase para a contribuição das pessoas.

O *Scrum* é em uma abordagem flexível, adaptativa e produtiva. Segundo Schwaber e Sutherland (2011), trabalha de uma maneira que permita que a equipe trabalhe de uma forma flexível em um ambiente em constante mudança.

O **XP**, por sua vez, tem o objetivo de assegurar que o cliente receba o máximo possível de desenvolvimento a cada dia de trabalho da equipe. Ele é formado baseado em valores e práticas que agem com harmonia e coesão para assegurar que o cliente sempre receba um bom retorno do investimento feito em implementar um Software na empresa (BECK; ANDRES, 2004).

Diante disso, conclui-se que as duas abordagens procuram resolver limitações dos métodos tradicionais de desenvolvimento, que diz respeito a redução de custos com mudanças de requisitos ao longo do desenvolvimento e a redução tempo de entrega do produto final ao cliente.

Embora o *Scrum* seja mais abrangente; ambas metodologias dependem de uma avaliação do contexto para serem utilizadas. O próximo tópico trata de modelos de

maturidade e capacidade que são importantes para se conhecer o grau de maturidade no desenvolvimento de software que empresas apresentam.

2.5 Modelos de Maturidade e Capacidade (MMC)

Esta pesquisa aborda os modelos de Maturidade e Capacidade, considerando que esses modelos utilizam atributos que caracterizam importantes para determinar a maturidade de organizações.

Tratando-se de Engenharia de Software, a maturidade de uma organização mede a capacidade técnica, gerencial e a competência que essa organização tem para o desenvolvimento de software (FILHO; PÁDUA, 2009). É associado a representações por estágios e é aplicada para a melhoria dos processos das organizações.

O modelo de capacidade de uma organização está relacionado com a previsibilidade do processo e seus resultados. Assim, a capacidade de um processo descreve os resultados esperados que possam ser alcançados, considerando um conjunto de processos e suas capacidades, enquanto os modelos de maturidade se propõem a identificar um estágio, ou nível, em que a organização ou a equipe se encontra (PRIKLADNICKI, 2009).

Os modelos apresentados a seguir: o CMMI (*Maturity Model Integration*) e o MPS.BR (Melhoria de Processo do *Software* Brasileiro) são alguns dos modelos (ou frameworks) que têm sido propostos ao longo do tempo para a evolução das organizações e, em particular, a Gestão dos Sistemas de Informação.

2.5.1 Capability Maturity Model Integration – CMMI

O CMMI (*Capability Maturity Model Integration* – Modelo Integrado de Maturidade e de Capacidade) é um modelo de maturidade para melhoria de processo, destinado ao desenvolvimento de produtos e serviços, e composto pelas melhores práticas associadas a atividades de desenvolvimento e de manutenção que cobrem o ciclo de vida do produto desde a concepção até a entrega e manutenção. (Mellon, 2006)

Segundo a SEI (2010) o CMMI é um guia de boas práticas que tem o objetivo de melhorar o desempenho e a capacidade de cumprir os objetivos dos negócios e o melhoramento dos processos das organizações. Com sua base com foco na otimização de processos o CMMI

identifica os pontos positivos e negativos da organização ou empresa e procura modificar os processos para melhorar os pontos fortes e reduzir as fraquezas.

O objetivo do CMMI para Desenvolvimento é auxiliar as organizações na melhoria de seus processos de desenvolvimento e manutenção de produtos e serviços. O CMMI para Desenvolvimento é um conjunto de melhores práticas gerado a partir do *Framework* do CMMI1, o qual apoia a suíte de produtos CMMI, permitindo a geração de diversos modelos, treinamentos e métodos de avaliação para áreas de interesse específicas (Mellon, 2006).

As áreas de processos incluem práticas que abrangem conceitos de gerenciamento de projetos, gerenciamento de processos, infraestrutura e suporte e são agrupadas em quatro categorias (SEI, 2010):

- Gerência de Processos: A qual contém atividades relacionadas para definir, planejar, implantar, monitorar, controlar, medir e melhorar processos;
- Gerência de Projeto: Se refere a atividades de planejar, monitorar e controlar o projeto;
- Engenharia: Refere-se às atividades de desenvolvimento e manutenção de várias disciplinas de engenharia;
- Suporte: Fornece suporte ao desenvolvimento e à manutenção de produtos.

Os modelos se utilizam de gradações para diferenciar o amadurecimento da engenharia de software nas empresas conhecidos por níveis de maturidade. Estes são formados por práticas, tanto genéricas como específicas relacionadas a um conjunto de processos, que melhoram o desempenho geral da organização no tocante à TIC. Cada nível é uma camada que representa a base para as atividades de melhoria contínua de processo. Um nível de capacidade para uma área de processo é alcançado quando todos os objetivos (metas) genéricos são satisfeitos até o nível presente (SEI, 2010).

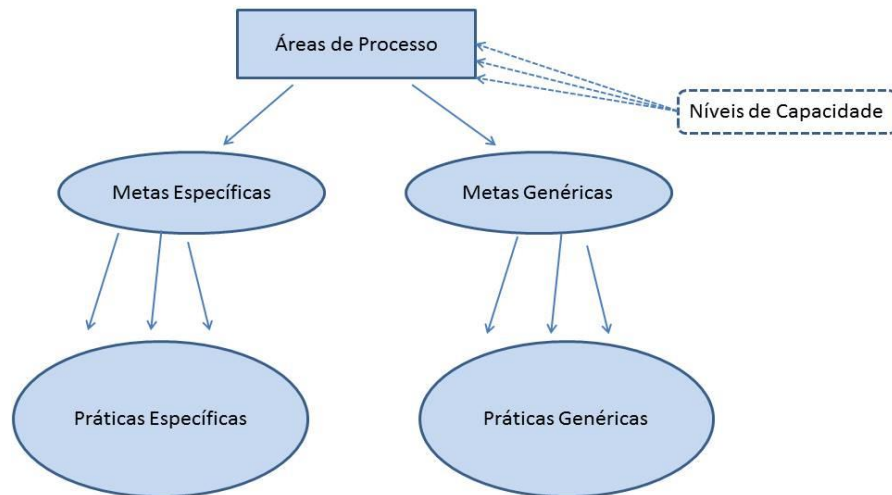
Segundo Franciscani (2012), o CMMI possui dois tipos de representações distintas: por estágios e contínua, que serão apresentados a seguir.

2.5.1.1 Representação por estágio

A representação por estágios oferece uma forma sistemática e estruturada para abordar a melhoria de processo, baseada em modelo e enfocando um estágio por vez. As áreas de processo são organizadas em níveis de maturidade, o que reduz a necessidade de escolhas associadas à melhoria de processo (SEI, 2010). A representação por estágios prescreve uma ordem

de implementação das áreas de processo de acordo com níveis de maturidade, definindo um caminho de melhoria para a organização, do nível “inicial” ao nível “em otimização”.

Figura 4 – Estrutura da representação por estágio



Fonte: CMMI(2006)

A representação por estágios fornece um caminho pré-definido para melhoria por meio de implementação sequencial, onde cada nível é base para o próximo, e é dividido em cinco níveis, como mostra a Figura 5. Os cinco níveis são assim definidos:

- Nível 1 (Inicial): Os processos desse nível geralmente são caóticos e ad hoc. O gerenciamento de processos é quase inexistente. É considerado que todos os processos já estão no nível 1;
- Nível 2 (Gerenciado): Nos processos desse nível há uma disciplina que contribui para que as práticas existentes não sejam abandonadas em períodos críticos;
- Nível 3 (Definido): Os processos já são bem entendidos e caracterizados. É utilizado processos em um padrão característico para que seja estabelecida igualdade no contexto da organização;
- Nível 4 (Gerenciado Quantitativamente): O desempenho dos processos é previsível quantitativamente e é controlado utilizando técnicas estatísticas e outras técnicas quantitativas. A organização objetiva para melhorar a qualidade e desempenho de processos;

- **Nível 5 (Otimização):** Utilização de melhorias inovadoras e incrementais de processo e de tecnologia focando no desempenho do processo. A organização visa tratar modificações do processo a fim de atender aos objetivos quantitativos de melhoria estabelecidos.

Figura 5 - Níveis de maturidade por estágios

Inicial	Gerenciado	Definido	Gerenciado Quantitativamente	Otimizado
• O gerenciamento de processos é quase inexistente.	• Gestão de configuração de software; • Assegurar qualidade; • Gestão de subcontratação de software; • acompanhamento e supervisão de projetos de software; • Planejamento de projetos de software; • Gerenciamento de requisitos.	• Avaliações por pares; • Coodenação interna de grupo; • Engenharia de produto de software; • Gestão integrada de software; • Programa de formação; • Definição do processo de organização; • Foco na organização de processo.	• Gestão de qualidade de software; • Gestão de processo.	• Gerenciamento de mudanças processo; • Gerenciamento de mudanças e tecnologia; • Prevenção de erros.

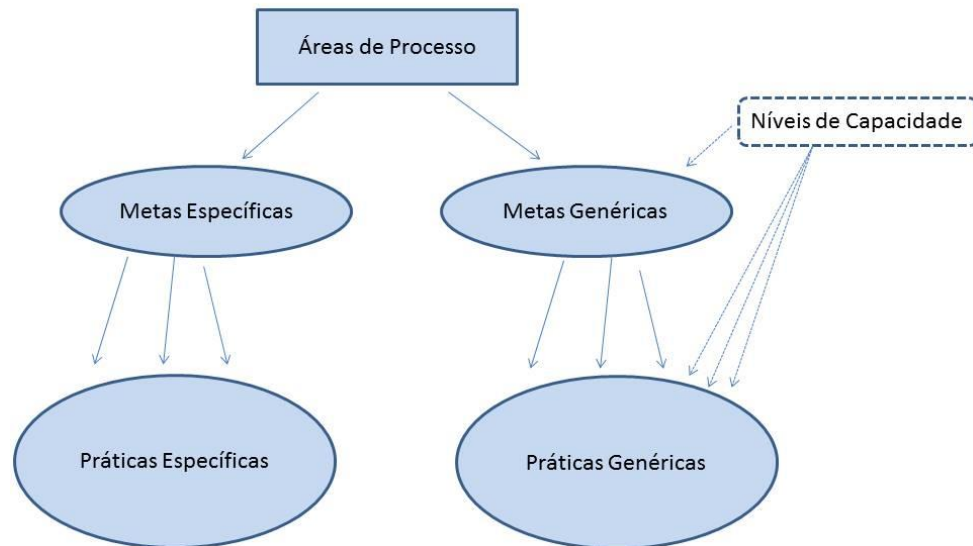
Fonte: Adaptado de (Amid, 2012)

Outra representação é apresentada a seguir.

2.5.2 Representação Contínua

A representação contínua oferece uma enorme flexibilidade na utilização de um modelo CMMI para melhoria de processo. As organizações focam em melhorias de desempenho de um ponto com algum problema associado a um processo isolado, ou podem dedicar-se a áreas que possam estar ligadas aos objetivos estratégicos das organizações. A representação contínua também permite que uma organização melhore diferentes processos ao longo do tempo (SEI, 2010). As organizações se limitam um pouco em suas escolhas devido às dependências entre as áreas de processo. Mas se os processos das organizações que precisam ser melhorados são conhecidos e se as dependências entre as áreas de processo descritas no CMMI são bem compreendidas, a representação contínua é uma boa escolha para essa organização.

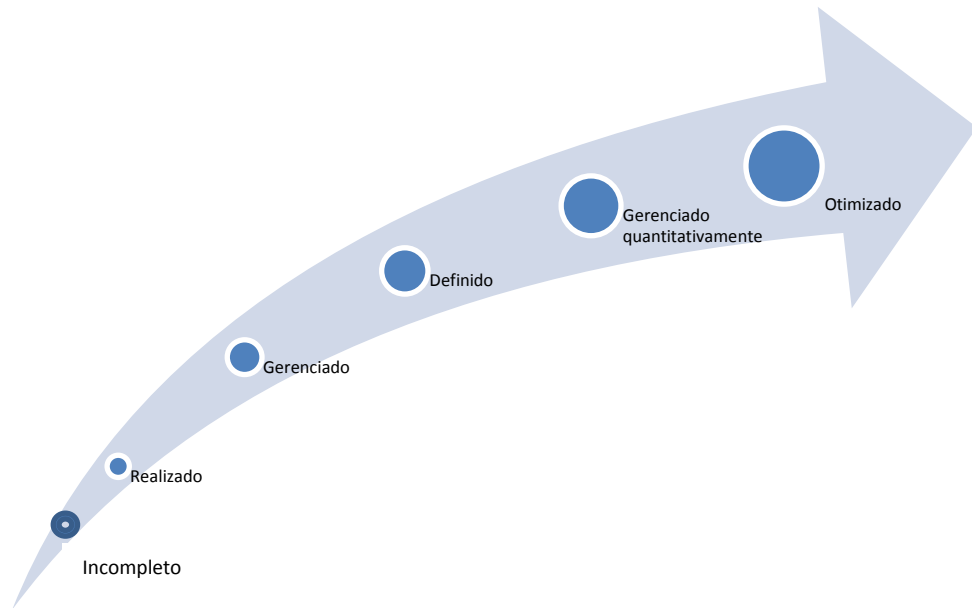
Figura 6 – Estrutura da representação contínua



Fonte: CMMI(2006)

A representação Contínua reduz os riscos fornecendo maior foco nas áreas de processo escolhidas de acordo com os objetivos de negócio. É dividido em seis níveis, como mostra o gráfico abaixo:

Figura 7 – Níveis na representação contínua



Os três primeiros níveis da Figura 7, de baixo para cima, definem a capacidade de um processo, enquanto os dois últimos níveis definem a maturidade, visto que não se faz necessária a utilização de apenas um dos modelos (capacidade ou maturidade) e sim a junção dos dois modelos para a gestão de um desenvolvimento de software com qualidade.

Essas representações – contínua e por estágio, foram criadas para oferecer resultados que são essenciais e equivalentes para a melhoria de processos ou para avaliação, apresentando conteúdo comum ao modelo CMMI. Portanto, uma organização não precisa escolher uma representação em detrimento da outra.

2.5.3 Melhoria de Processo do Software Brasileiro - MPS.BR

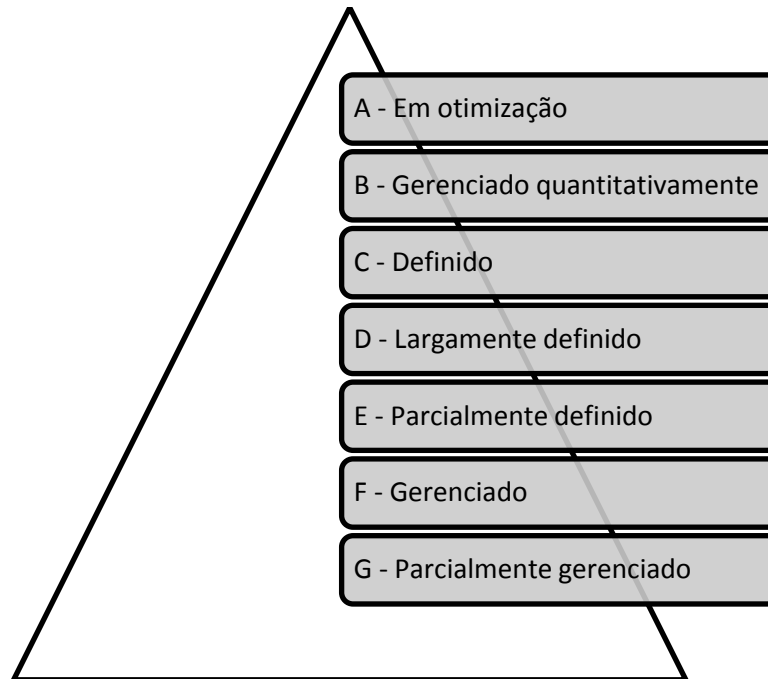
O modelo de maturidade MPS.BR foi um marco da engenharia de software no Brasil e tem como objetivo principal auxiliar na avaliação de processos visando melhoria no processo de software, possibilitando que as organizações possam ter mais qualidade e a produtividade (SOFTEX, 2012).

O modelo MPS tem sua base composta pelas Normas Internacionais ISO/IEC 12207:20081 e ISO/IEC 15504-22, possibilitando a adaptação dependendo da necessidade da comunidade. O MPS é compatível com o modelo CMMI e seu guia de implementação contém 11 documentos que contém orientações para as organizações programarem os níveis de maturidade contidos mesmo (SOFTEX, 2012).

De acordo com a (SOFTEX, 2012), o MPS divide seus níveis de maturidade, os quais são a combinação entre os processos e a capacitação de processos, em sete níveis que vão do

G (pior) ao A (melhor), demonstrado na Figura 7; cada nível inclui os processos do nível anterior. Os níveis estão descritos abaixo :

Figura 8 – Níveis de maturidade do MPS.Br



Explica-se cada nível abaixo:

- Nível G (Parcialmente Gerenciado): É responsável pela parte inicial da melhoria dos processos de software na organização. É composto por dois processos: Gerência de Projetos e Gerência de Requisitos.
- Nível F (Gerenciado): Este nível de maturidade é formado pelos processos do nível de maturidade anterior (G) juntamente com os processos de Aquisição, Garantia da Qualidade, Gerência de Configuração, Gerência de Portfólio de Projetos e Medição.
- O Nível E (Parcialmente Definido): Este é formado pelos processos dos níveis de maturidade anteriores (G e F), juntamente com os processos de Avaliação e Melhoria do Processo Organizacional, Definição do Processo Organizacional, Gerência de Recursos Humanos e Gerência de Reutilização.

- Nível D (Largamente Definido): é formado pelos processos dos níveis de maturidade anteriores (G, F e E), juntamente com os processos de Desenvolvimento de Requisitos, Integração do Produto, Projeto e Construção do Produto, Validação, e Verificação.
- Nível C (Definido) é formado pelos processos dos níveis de maturidade anteriores (G, F, E e D), juntamente com os processos de Desenvolvimento para Reutilização, Gerência de Decisões e Gerência de Riscos.
- Nível B (Gerenciado Quantitativamente): é formado pelos processos dos níveis de maturidade anteriores (G, F, E, D e C) e não possui processos específicos. Gerência de Projetos sofrerá sua segunda evolução nesse nível de maturidade implementando novos resultados para que os objetivos do gerenciamento quantitativo possam ser atendidos.
- Nível A (Em Otimização): é formado pelos processos dos níveis de maturidade anteriores (G, F, E, D, C e B) e da mesma forma que o nível de maturidade B, não possui processos específicos. Este nível tem o objetivo de analisar causas de problemas e encontrar sua resolução.

No MPS.BR, para cada nível de maturidade, existe número de capacidades (atributo de processos) a serem analisadas para cada processo, que são os resultados obtidos dos mesmos. São eles:

- AP 1.1 - O processo é executado;
- AP 2.1 - O processo é gerenciado;
- AP 2.2 - Os produtos de trabalho do processo são gerenciados;
- AP 3.1 - O processo é definido;
- AP 3.2 - O processo está implementado;
- AP 4.1 - O processo é medido;
- AP 4.2 - O processo é controlado;
- AP 5.1 - O processo é objeto de inovações;
- AP 5.2 - O processo é otimizado continuamente.

As capacidades descritas definem o nível de maturidade em que os processos de uma organização se encontram. A Tabela 4 mostra os atributos e processos divididos em níveis de Maturidade.

Tabela 4-Níveis de Maturidade do MR-MPS.

Nível	Processos	Atributos de Processo
A – Em Otimização		AP 1.1, AP 2.1, AP 2.2, AP 3.1, AP 3.2, AP 4.1, AP 4.2, AP 5.1 e AP 5.2
B – Gerenciado Quantitativamente	Gerência de Projetos – GPR (evolução)	AP 1.1, AP 2.1, AP 2.2, AP 3.1, AP 3.2, AP 4.1 e AP 4.2
C – Definido	Gerência de Riscos – GRI Desenvolvimento para Reutilização – DRU <i>Gerência de Decisões - GDE</i>	AP 1.1, AP 2.1, AP 2.2, AP 3.1 e AP 3.2
D – Largamente Definido	Verificação – VER Validação – VAL Projeto e Construção do Produto – PCP Integração do Produto – ITP <i>Desenvolvimento de Requisitos – DRE</i>	AP 1.1, AP 2.1, AP 2.2, AP 3.1 e AP 3.2
E – Parcialmente Definido	Gerência de Projetos – GPR (evolução) Gerência de Reutilização – GRU Gerência de Recursos Humanos – GRH Definição de Processo Organizacional – DFP <i>Avaliação e Melhoria do Processo Organizacional – AMP</i>	AP 1.1, AP 2.1, AP 2.2, AP 3.1 e AP 3.2
F - Gerenciado	Medição – MED Garantia de Qualidade – GQA Gerência de Portfólio de Projetos – GPP Gerência de Configuração – GCO <i>Aquisição - AQU</i>	AP 1.1, AP 2.1 e AP 2.2
G – Parcialmente Gerenciado	Gerência de Requisitos – GRE <i>Gerência de Projetos – GPR</i>	AP 1.1 e AP 2.1

Fonte: Adaptado de SOFTEX (2012).

O MPS.BR propõe uma quantidade maior de estágios em relação ao CMMI, o que deve permitir uma implantação mais gradual e adequada, facilitando as tarefas em empresas de pequeno porte (FONSECA, 2008).

2.5.4 Control objectives for Information and related Technology – COBIT

Outros padrões de engenharia de software se valem de repositórios de boas práticas de desenvolvimento, com destaque para o COBIT e o ITIL, este último não faz parte deste estudo.

O COBIT é um modelo para a implantação e auditoria de controles de TI, desenvolvido pelo ISACA (*Information Systems Audit & Control Association*) e mantido pelo ITGI. Provê melhores práticas para contribuir em entregas adequadas da TI, a partir de uma perspectiva de negócio, e tem foco na execução e controle (SALLÉ, 2004; FERNANDES; ABREU, 2008).

O COBIT é um conjunto de boas práticas de processos de gerenciamento de TI, a qual aborda de aspectos técnicos e processos às pessoas. Tem o objetivo de controlar a TIC para garantir que os seus recursos estão alinhados com os objetivos da organização através de um framework de controle. É baseado na ideia de que a TIC deve entregar as informações que a organização precisa e tem como objetivo a garantia da entrega de serviços (COBIT, 2007).

O COBIT foi criado para atender a necessidades de um framework de controle de TIC compreensivo para o negócio, gerência de TI, auditores e para eliminar disparidades de controle e guias de avaliação (TIEXAMES, 2009). Seu principal tema é a orientação para os negócios, o qual foi desenvolvido para assessorar provedores de serviços, usuários e auditores, bem como fornecer um guia completo para os executivos e para os gerentes de negócios.

O seu princípio fundamental é relacionar expectativas e responsabilidade de gestores de TI, buscando dessa forma atender às expectativas do negócio. É um padrão que deve ser adaptado para cada organização. Outros princípios envolvem:

- Responsabilidade corporativa: Se trata do pensamento e da ação pela perenidade da organização, com responsabilidade social e ambiental;
- Prestação de contas: Está relacionado à obrigação de prestar contas;
- Equidade: Trata-se do tratamento justo e igualitário;
- Transparência: Está relacionado ao desejo de informar.

De acordo com ISACA - Information Systems Audit and Control Association, a missão do COBIT é pesquisar, desenvolver, publicar e promover um conjunto de objetivos de controle para tecnologia que seja embasado, atual, internacional e aceito em geral para o uso do dia-adia de gerentes de negócio e auditores (ISACA, 2009). Portanto o COBIT trabalha principalmente dentro do seguinte conjunto de atividades:

- Alinhamento da TIC com o negócio da empresa;
- Definição do papel da TIC (TI Estratégica ou TIC Operacional);

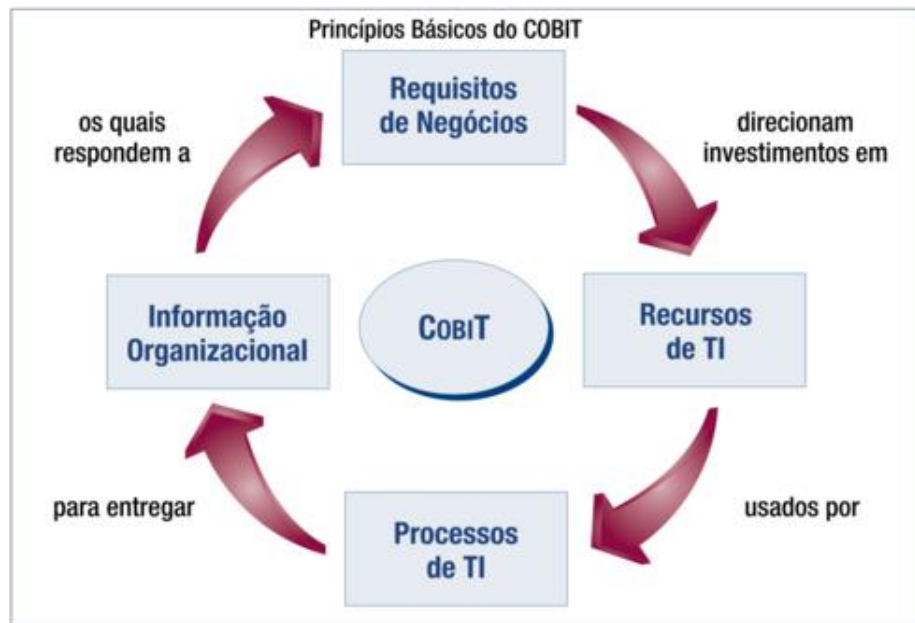
- Auxilia na organização das atividades da TIC a partir da adoção de um modelo de gestão;
- Ajuda a identificar quais recursos de TIC devem ser alavancados com maior efetividade;
- Define os objetivos e controles gerenciais a serem observados;
- Estabelece claramente papéis e responsabilidades.

Este modelo de governança estabelece sete critérios de informação baseados em requisitos de qualidade, guarda e segurança, suficientes para atender aos objetivos de negócios, nos quais as informações precisam estar adequadas a critérios de controle. O COBIT denomina esses critérios de qualidade como sendo necessidades de informação da empresa. (COBIT, 2007). Esses critérios são:

- **Efetividade:** lida com a informação relevante e pertinente para o processo de negócio bem como a mesma sendo entregue em tempo, de maneira correta, consistente e utilizável.
- **Eficiência:** relaciona-se com a entrega da informação através do melhor (mais produtivo e econômico) uso dos recursos.
- **Confidencialidade:** está relacionada com a proteção de informações confidenciais para evitar a divulgação indevida.
- **Integridade:** relaciona-se com a fidedignidade e totalidade da informação bem como sua validade de acordo os valores de negócios e expectativas.
- **Disponibilidade:** relaciona-se com a disponibilidade da informação quando exigida pelo processo de negócio hoje e no futuro. Também está ligada à salvaguarda dos recursos necessários e capacidades associadas.
- **Conformidade:** lida com a aderência a leis, regulamentos e obrigações contratuais aos quais os processos de negócios estão sujeitos, isto é, critérios de negócios impostos externamente e políticas internas.
- **Confiabilidade:** relaciona-se com a entrega da informação apropriada para os executivos para administrar a entidade e exercer suas responsabilidades fiduciárias e de governança.

A interação entre a informação organizacional e os recursos de TIC realizada dentro do modelo COBIT se mostra na Figura 9:

Figura 9 – Interação entre elementos envolvidos pelo COBIT



Fonte: (COBIT, 2007)

Por ser orientado para processos e definir algumas atividades em um modelo de processos genérico, o COBIT utiliza quatro domínios (FERNANDES; ABREU, 2008), como mostra a Figura 10; e são descritos a seguir:

Figura 10 – Domínios do COBIT



Fonte: (COBIT, 2007)

Explica-se a figura em cada uma das fases:

a. Planejar e Organizar (PO)

- a. Provê direção para entrega de soluções (AI) e entrega de serviços (DS).
- b. é responsável por identificar a melhor maneira que a TIC pode contribuir para que a empresa possa atingir seus objetivos;
- c. Para que a visão estratégica seja eficaz é necessário que haja planejamento, comunicado e gerenciado por diferentes perspectivas.

O domínio de planejamento e organização ajuda a entender algumas questões de gerenciamento, como por exemplo, se todos na organização entendem os objetivos da TI, se os riscos são entendidos e gerenciados e se a qualidade dos sistemas é adequada às necessidades de negócio.

b. Adquirir e Implementar (AI)

- a. Provê as soluções e as transfere para tornarem-se serviços;
- b. Inclui alterações e manutenções nos sistemas, com o propósito de assegurar que as soluções continuem atendendo aos objetivos de negócios.
- c. Este domínio trata algumas questões de gerenciamento, que são:
 - Os novos projetos serão entregues no tempo e orçamento previstos?

- As alterações ocorrerão sem afetar as operações de negócios atuais?
- Os novos projetos fornecerão soluções que atendam às necessidades de negócios?

As soluções de TIC precisam ser identificadas, desenvolvidas ou adquiridas, implementadas e integradas.

c. Entregar e Suportar (DS)

- a. Recebe as soluções e tornam elas passíveis de uso pelos usuários finais;
- b. Algumas questões que são tratadas por este domínio são:
 - Os serviços de TIC estão sendo entregues de acordo com as prioridades de negócios?
 - Os custo de TIC estão otimizados?
 - Os aspectos de confidencialidade, integridade e disponibilidade estão sendo contemplados para garantir a segurança da informação?

Este domínio é responsável pela entrega de serviços solicitados, gerenciamento da segurança e continuidade, serviços de suporte, gerenciamento de dados e recursos operacionais.

d. Monitorar e Avaliar (ME)

- a. Monitora todos os processos para garantir que a direção definida seja seguida;
- b. Aborda o gerenciamento de desempenho, o monitoramento do controle interno, a aderência regulatória e a governança;
- c. Inclui algumas questões de gerenciamento:
 - O desempenho de TIC é mensurado para detectar problemas antes que seja muito tarde?
 - O gerenciamento assegura que os controles internos sejam efetivos e eficientes?
 - O desempenho da TIC pode ser associado aos objetivos de negócio?
 - Existem controles adequados para garantir confidencialidade, integridade e disponibilidade das informações?

Segundo o COBIT, a maioria das organizações definiram as responsabilidades de TIC, como sendo: planejar, construir, processar e monitorar, embora muitas delas tenham os mesmos processos-chave, terão a mesma estrutura de processos. Esse modelo fornece uma completa lista de processos que podem ser utilizados para verificar a totalidade das atividades e responsabilidades, mas nem todos precisam ser aplicados e podem ser combinados conforme as necessidades de cada empresa (COBIT, 2007).

Segundo Qumer e Henderson-Sellers (2007), os pontos fortes do COBIT em governança se resumem a:

- Gestão de desempenho;
- Fatores críticos de sucesso;
- Modelos de maturidade da capacidade.

Sua responsabilidade é prover a informação que a organização necessita para alcançar seus objetivos, gerenciar e controlar os recursos de TIC usando um conjunto estruturado de processos para prover os serviços que disponibilizam as informações necessárias para a organização.

O tópico abaixo apresenta uma discussão entre as Representações Contínua e por Estágio.

2.6 Discussão

Este tópico apresenta algumas comparações a partir do referencial teórico levantado para este trabalho. Serão desenvolvidas algumas considerações com respeito às formas de representação, níveis de capacidade, relações entre modelos, e vantagens e desvantagens entre modelos, entre outros.

2.6.1 Comparação entre as Representações Contínua e por Estágios

A Tabela 5 apresenta os pontos fortes entre as representações Contínua e por Estágios e pode auxiliar na escolha da representação mais adequada para a organização.

Tabela 5 – Diferenças entre as representações contínuas e por estágios

Representação Contínua	Representação por Estágios
Permite livre escolha da sequência de melhorias, de forma a melhor satisfazer aos objetivos estratégicos e mitigar as áreas de risco da organização.	Permite que as organizações tenham um caminho de melhoria predefinido e testado.
Permite visibilidade crescente da capacidade alcançada em cada área de processo.	Foca em um conjunto de processos que fornece à organização uma capacidade específica caracterizada por cada nível de maturidade.
Permite que melhorias em diferentes processos sejam realizadas em diferentes níveis.	Resume os resultados de melhoria de processo em uma forma simples: um único número que representa o nível de maturidade.
Reflete uma abordagem mais recente que ainda não dispõe de dados para demonstrar seu retorno do investimento.	Baseia-se em uma história relativamente longa de utilização, com estudos de casos e dados que demonstram o retorno do investimento.

Fonte: (Mellon, 2006)

Em resumo, a representação contínua tem como foco uma determinada área de processo para melhorias e definição do nível de capacidade desejado para aquela área de processo. Por sua vez, a representação por estágios visa à maturidade global da organização, sem focar se os processos individuais são executados ou incompletos (SEI, 2010).

As organizações podem encontrar utilidade em ambas as representações. Dificilmente uma organização implementa alguma dessas representações exatamente conforme prescritas. As organizações bem sucedidas em melhoria de processo frequentemente definem um plano de melhoria que foca suas necessidades específicas e então utilizam os princípios tanto da representação contínua como da representação por estágios.

A próxima discussão trata das associações entre os níveis de capacidade e maturidade, e seus significados.

2.6.2 Quanto aos níveis de Capacidade e de Maturidade

Maturidade diz respeito a processos que são explicitamente definidos, gerenciados, medidos, controlados e eficazes. Maturidade também pode ser entendida através de como as organizações maduras realizam as tarefas, independentes de como elas conseguem atingir os resultados.

Organizações com maturidade atingem objetivos de qualidade, prazos e custos de forma consistente e eficiente, em contraposição à organizações imaturas, as quais criam objetivos com muita frequência, e perdem seus objetivos por largas margens de erros.

Como mencionado; o modelo de capacidade de uma organização está relacionado com a previsibilidade do processo e seus resultados; assim, a capacidade de um processo descreve os resultados esperados que possam ser alcançados, considerando um conjunto de processos e suas capacidades (PRIKLADNICKI, 2009).

As empresas de desenvolvimento de *software* utilizam os Modelos de Maturidade de Processos como referenciais para:

- Avaliar a capacidade de processos na realização de seus objetivos;
- Localizar oportunidades de melhoria de produtividade e qualidade e de redução de custos;
- Planejar e monitorar as ações de melhoria contínua dos processos empresariais

A **Tabela 6** mostra um comparativo entre os níveis de Maturidade e Capacidade nas representações Contínua e por Estágios, com relação ao CMMI, como exemplo.

Tabela 6-Comparação entre os Níveis de Capacidade e os Níveis de Maturidade

Nível	Representação Contínua Níveis de Capacidade	Representação por Estágios Níveis de Maturidade
Nível 0	Incompleto	
Nível 1	Executado	Inicial
Nível 2	Gerenciado	Gerenciado
Nível 3	Definido	Definido
Nível 4		Gerenciado Quantitativamente
Nível 5		Otimizado

Fonte: Adaptado de CMMI-SEI(2010)

Nessa tabela, o nível 0 representa a quase ausência de identificação de atividades; no nível 1, os processos ainda são imprevisíveis, pouco controlados e reativos; o nível 2, os processos são caracterizados por “Projeto” e ações frequentemente reativas; o nível 3, os proces-

sos são caracterizados por “Organização” e são proativos, já o nível 4, os processos são medidos e controlados e o nível 5 há um foco contínuo na melhoria dos processos.

O que também notamos pela Tabela 6 é que há associação com significados diferentes entre as duas representações, por exemplo, o Nível 3, o nível de capacidade está definido, assim como o nível de maturidade; o Nível 2, os níveis de maturidade e capacidade estão como Gerenciado, porém, na representação por estágios de maturidade, nota-se que há ainda estágios a serem percorridos até que a organização possa estar classificada como em estágio otimizado.

Quanto à rigidez ou não da aplicação das representações, segundo Fonseca (2008), as organizações também podem variar entre as duas representações dependendo da sua necessidade momentânea. Normalmente recorre-se a uma empresa especializada para implementar o CMMI devido ao seu custo e complexidade de implementação; além disso, nem todos os detalhes metodológicos podem ser implantados, mas sim adaptados a cada diferente tipo de organização.

O próximo tópico abordará uma breve comparação entre os modelos de maturidade, os quais focam no desenvolvimento de software tradicional. São eles o CMMI e o MPS.BR.

2.6.3 Comparação entre CMMI e MPS.BR

Entre os modelos de maturidade estudados, tanto o CMMI, quanto o MPS.BR tem o mesmo objetivo, que é avaliar processos, e se utilizam de modelos da capacidade como forma de avaliação para esses processos. Mas, apesar dos dois modelos de desenvolvimento compartilhar objetivos semelhantes, eles possuem focos diferentes.

Enquanto o CMMI tem um foco global mais voltado para as empresas de maior porte, o MPS.BR é um modelo criado em função de empresas e organizações de menos porte, (OLIVEIRA, 2008). As pequenas e médias empresas optam por adotar o modelo o MPS.BR buscando alcançar padronização e qualidade nos seus processos com um menor custo e tempo. Depois que essa padronização for alcançada a empresa pode focar em se certificar como CMMI (FRANCISCANI, 2012).

Segundo Franciscani (2012), ambos os modelos demonstram diversos pontos comuns em suas fases de implementação, bem como aos aspectos organizacionais envolvidos em sua utilização.

Os dois modelos apresentam diferenças em relação aos seus níveis de maturidade. Por mais que ambos tenham a mesma finalidade, que é a de demonstrar a evolução de cada empresa de acordo com os níveis de segurança e qualidade demonstrados por elas. Seus níveis de maturidade são apresentados em diferentes escalas quanto a seu processo de identificação e de mensuração.

O CMMI é determinado por cinco níveis de maturidade, os quais determinam a posição que a empresa ocupa na escala do modelo, enquanto o MPS.BR é composto por sete níveis de maturidade, as empresas precisam de um tempo maior para atingir o topo, o que assegura a empresa confiança e qualidade total nos processos desenvolvidos (FRANCISCANI, 2012).

A tabela abaixo mostra uma comparação entre os níveis do CMMI e os níveis do MPS.BR.

Tabela 7- Análise comparativa dos níveis de maturidade entre o CMMI e o MPS.BR

CMMI	MPS.BR
5	A
4	B
3	C
	D
	E
2	G
	F
1	

Fonte: Adaptado de (OLIVEIRA, 2008)

Há uma compatibilidade entre o MPS-BR e o CMMI. Esta compatibilidade é total do ponto de vista do MPS-BR para o CMMI, isto é, todos os requisitos das áreas de processo do CMMI estão presentes no MPS-BR (Franciscani, 2012). Entretanto não existe esta equivalência total do ponto de vista do CMMI para o MPS-BR, pelas seguintes razões:

- Nível F do MPS-BR: existe o processo Gerência de Portfólio de Projetos que não existe no CMMI;
- Nível E do MPS-BR: o processo Gerência de Recursos Humanos inclui os requisitos da área de processo Treinamento Organizacional, mas tem requisitos relacionados à Aquisição de Pessoal e Gerência de Conhecimento que não estão presentes na área de Treinamento Organizacional do CMMI;
- Nível D do MPS-BR: o processo Gerência de Recursos Humanos inclui os requisitos da área de processo Treinamento Organizacional, mas tem requisitos relacionados à Aquisição de Pessoal e Gerência de Conhecimento que não estão presentes na área de Treinamento Organizacional do CMMI;

- Nível E do MPS-BR: existe o processo Gerência de Reutilização que não existe no CMMI;
- Nível C do MPS-BR: existe o processo Desenvolvimento para Reutilização que não existe no CMMI.

A Tabela 8 representa a correlação entre CMMI e MPS.BR, respectivamente, nela estão caracterizadas as principais semelhanças contidas nos dois modelos:

Tabela 8- Correlação entre os modelos CMMI e MPS-BR

CMMI		MPS.BR	
5	Análise Casual e Resolução – CAR Inovação e Melhoria Organizacional - OID	A	Análise de Causas de Problemas e Resolução
4	Desempenho do Proc. Org. – OPP Gerência Quantitativa de Projeto - QPM	B	Gerência Quantitativa do Projeto
3	Foco no Processo da Organização – OPF Definição do Proc. da Organização – OPD Treinamento Organizacional – OT Gerência Integrada de Projeto – IPM Gerência de Risco – RSKM Desenvolvimento de Requisitos – RD Solução Técnica – TS Integração de Produto – PI Verificação – VER Validação – VAL Análise de Decisão e Resolução - DAR	C	Análise de Decisão e Resolução Gerência de Riscos Desenvolvimento de Reutilização
		D	Desenvolvimento de Riscos Integração do Produto Projeto e Construção do Produto Verificação Validação
		E	Gerência de Recursos Humanos Avaliação e Melhoria do Proc. Org. Definição do Proc. Organizacional Gerência de Reutilização
2	Gerência de Requisitos – REQM Planejamento de Projeto – PP Acompanhamento e Contr. de Proj. – PMC Ger. de Acordo com Fornecedores – SAM Gar. de Qual. de Proc. e Produto – PPQA Gerência de Configuração – CM Medição e Análise - MA	F	Medição Gerência de Configuração Aquisição Garantia da Qualidade
		G	Gerência de Requisitos Gerência de Projetos

Fonte: (Franciscani, 2012)

Segundo Oliveira (2008) as principais diferenças entre os dois modelos são apresentados na Tabela 9:

Tabela 9 – Principais diferenças entre os modelos CMMI e MPS.Br

CMMI	MPS.BR
O Modelo de Qualidade CMMI é reconhecido internacionalmente.	O MPS.BR é mais conhecido nacionalmente e na América Latina.
O modelo CMMI envolve um grande custo na Avaliação e Certificação do Modelo.	No MPS.BR o custo da certificação é mais acessível.
No CMMI é necessário investir tempo, geralmente para se chegar aos níveis de maturidade mais altos.	No MPS.BR as avaliações são bienais.
O CMMI tem foco global voltado para empresas de maior porte.	MPS.BR é um modelo criado em função das médias e pequenas empresas.
O CMMI possui cinco níveis de maturidade por estágio e seis na Contínua.	MPS.BR possui sete níveis de maturidade, onde a implantação é mais gradual.
O CMMI torna as empresas competitivas internacionalmente.	O MPS.BR não torna as empresas competitivas internacionalmente.
O CMMI não utiliza contrato conjunto de empresas	No MPS.BR pode acontecer contrato cooperado em grupo de empresas que queiram a Certificação
Implementação mais complexa.	Implementação mais simples.
Desenvolvido pelo Software Engineering Institute – SEI em 1992	Desenvolvido por algumas instituições Brasileiras em 2003.

Fonte: (Franciscani, 2012)

O próximo tópico apresenta as vantagens e desvantagens entre os modelos.

2.6.4 Vantagens e Desvantagens entre CMMI e MPS.BR

Esse subcapítulo tem como objetivo apresentar as vantagens e desvantagens da adoção de cada processo dos dois modelos de qualidade adotados no Brasil, o CMMI e o MPS.BR, os quais tem se destacado de formas distintas.

As principais vantagens e desvantagens comparativas entre os dois modelos são apresentadas, de forma resumida, na Tabela 7.

Tabela 10-Vantagens e Desvantagens do CMMI e MPS.BR

	CMMI	MPS.BR
Vantagens	• Desenvolvimento de software com qualidade, garantindo o cumprimento dos prazos e atendendo as necessidades do cliente, deixando-os mais satisfeito com o produto entregue pela empresa; • Eliminação de inconsistências e redução de duplicidade; • Utilização de terminologia comum e estilo consistente; • Consistências com a norma ISO/SEC 15504	• Modelo de processo mais rápido de ser adquirido, adequado a realidade brasileira; • Mais acessível do que os modelos de projeto como CMMI; • Possui sete níveis de maturidade, onde a implantação é mais gradual e adequada a pequenas e médias empresas; • Possui compatibilidade com CMMI, facilitando a obtenção do certificado; • Avaliação bienal das empresas; • Integração universidade-empresa.
Desvantagens	• O modelo CMMi é proprietário e envolve um grande custo para a realização das avaliações do modelo para obter a certificação; • É necessário investir tempo, geralmente para se chegar aos níveis de maturidade mais altos leva em média 4 a 8 anos.	• Apesar do foco do MPS BR ser um meio das médias e pequenas empresas alcançarem a qualidade nos processos e nos produtos desenvolvidos, servindo como uma alternativa para o CMMI, a certificação não é competitiva o suficiente para tornar a empresa competitiva internacionalmente.

Fonte: Adaptado de Oliveira (2008) e (Franciscani, 2012)

Como mostra a tabela acima, uma das maiores vantagens de se utilizar o CMMI é o desenvolvimento de software com qualidade, o CMMI garante o cumprimento dos prazos e atende as necessidades do cliente, deixando-os mais satisfeito com o produto entregue pela empresa, enquanto o MPS.BR é um modelo de processo mais rápido de ser adquirido, mais adequado a realidade brasileira e mais acessível do que os modelos de projeto como CMMI. Como dito anteriormente, o CMMI tem um foco global voltado para grandes empresas e envolve um custo alto para as avaliações enquanto o MPS.BR é um modelo criado com foco em pequenas e médias empresas, com o intuito de as mesmas alcançarem a qualidade nos produtos desenvolvidos, porém a certificação não é tão competitiva, a ponto de torna-la competitiva internacionalmente.

Muitas empresas acreditam que a qualidade gerada pelo CMMI possui um preço muito alto a se pagar e que a mesma não agrega muito valor à organização. Muitas tratam o CMMI como um processo e não como um modelo e consideram desnecessárias algumas de suas práticas e relatam que muito trabalho poderia ser evitado principalmente em pequenos projetos (FRANCISCANI, 2012).

2.6.5 Comparação entre CMMI e COBIT

A Tabela 11 mostra a análise dos modelos em questão.

A respeito das representações dos modelos de maturidade apresentados na tabela 11, levando em conta que o grau de competência de uma organização é medido com base apenas em um único valor na representação por estágios, concluímos que a mesma permite uma melhor comparação entre empresas concorrentes. Observamos também que os dois modelos analisados tem esse tipo de representação em comum. Adicionalmente, o modelo CMMI, além da representação estagiada, também tem a forma de representação contínua.

Tabela 11- Comparação entre CMMI e COBIT

Crítérios de Avaliação	CMMI	COBIT
Representações	Contínua e por Estágio	Apenas por Estágio
Quantidade de Níveis	5 níveis de maturidade e 4 níveis de capacidade	6 Níveis de Capacidade de processos
Número de Processos	24	9 Atributos de processos
Foco Principal	Serviços	COBIT
Ano de Publicação	2010	2012
Autoria	SEI(<i>Software Engineering Institute</i>)	ITGI(<i>IT Governance e Institute</i>)
País de Origem	Estados Unidos	Estados Unidos

Fonte: Adaptado de Silva (et al., 2013)

As quantidades de níveis são similares nos modelos abordados. Em destaque nesse quesito, o CMMI apresenta o conceito de níveis de capacidade além dos níveis de maturidade, e o modelo COBIT apresenta 6 níveis de maturidade, tornando assim a avaliação mais detalhada.

Em complemento, os dois modelos apresentados possuem suas próprias características e tem como objetivo avaliar uma área específica da área de gestão de serviços, auxiliando diferentes tipos de organizações em sua escolha pela avaliação da maturidade de seus processos.

Lembrando que são estruturas (*frameworks*) e, portanto possuem rigores metodológicos próprios que exigem trabalho intensivo, sua adoção é muito criticada por seus usuários em função dessa complexidade presente.

O próximo capítulo abordará conceitos de Governança, Governança Corporativa e Governança em TIC e como esses conceitos se inserem nas metodologias ágeis e nos modelos de maturidade de *software*.

3 GOVERNANÇA

O objetivo deste capítulo é abordar alguns conceitos sobre governança, dando ênfase a Governança corporativa e Governança em TIC, assim como metodologias ágeis de desenvolvimento de software e dar uma concepção sobre modelos de maturidade e modelos de capacidade, enfocando a diferença entre as duas.

O termo “Governança” teve origem a partir de reflexões conduzidas pelo Banco Mundial, na tentativa de construir Estados mais eficientes, e foi ampliada a análise outrora limitada a aspectos econômicos, de modo a incluir dimensões sociais e políticas da gestão pública (GONÇALVES, 1995).

Segundo Rosenau (2000, p. 15): “Governança não é o mesmo que governo”. Ainda segundo ele, “governo sugere atividades sustentadas por uma autoridade formal, pelo poder de polícia que garante a implementação das políticas devidamente instituídas, enquanto governança refere-se a atividades apoiadas em objetivos comuns, que podem ou não derivar de responsabilidades legais e formalmente prescritas e não dependem, necessariamente, do poder de polícia para que sejam aceitas e vençam resistências”. Vale notar ainda que a governança é um conceito suficientemente amplo para conter dentro de si a dimensão governamental.

Para Rosenau, “governança é um fenômeno mais amplo que governo; abrange as instituições governamentais, mas implica também mecanismos informais, de caráter não governamental, que fazem com que as pessoas e as organizações dentro da sua área de atuação tenham uma conduta determinada, satisfaçam suas necessidades e respondam às suas demandas” (Ibid., p. 15-16).

Já para (Slaughter, 1997): “Governança sem governo é governança sem poder, e governo sem poder raramente funciona”. O conceito de governança sem um governo leva especialmente ao estudo da política mundial, na medida em que nesse domínio é conspícua a ausência de uma autoridade central, embora seja também óbvio que um mínimo de ordem e de entendimentos rotinizados está normalmente presente na conduta da vida mundial”. (ROSENAU, 2000, p. 19).

Segundo Finkelstein, o estudo “sobre governança global está preocupado não apenas com decisões, mas também com suas consequências, como por exemplo, efeitos distributivos,

programas e projetos, eficácia, consentimento, e implementação doméstica” (FINKELSTEIN, 1991, p. 369).

Esses autores conduzem a discussão para o campo político, que não é o escopo deste trabalho, mas sim a aplicação do conceito em organizações, de modo a realizarem a missão a que foram destinadas. Nesse sentido, governança diz respeito aos meios e processos que são utilizados para produzir resultados eficazes.

Em geral, a Governança está ligada apenas à adequação a normas e padrões do mercado aberto de ações, relatórios financeiros e regulamentos jurídicos; mas a abrangência do conceito não se esgotou se ampliou em outras áreas. Por isso, compreender a Governança jurídica é: “nada mais simples e nada mais complicado” (POUND apud STEINBERG et al., 2003, p. 28).

3.1 Governança Corporativa

Para que governança de TIC possa ser entendida, primeiramente deve-se passar pelo conceito da governança corporativa, fazendo com que a mesma seja uma questão mais ampla em governança nas empresas (WEILL; ROSS, 2006).

Para o Instituto Brasileiro de Governança Corporativa – IBGC (2008, p. 1), Governança corporativa é o sistema pelo qual as sociedades são dirigidas e monitoradas, envolvendo os relacionamentos entre acionistas/cotistas, conselho de administração, diretoria, auditoria independente e conselho fiscal. As boas práticas de governança corporativa têm a finalidade de aumentar o valor da sociedade, facilitar seu acesso ao capital e contribuir para a sua perenidade. A empresa que opta pelas boas práticas de governança corporativa adota como linhas mestras transparência, prestação de contas (*accountability*) e equidade.

A Governança Corporativa está crescendo no Brasil e no mundo. Como em qualquer processo evolutivo, é irregular e inconstante (STEINBERG et al., 2003). Apesar de o termo Governança Corporativa ser relativamente recente, problemas com a governança das instituições são apontadas desde 1720, na Inglaterra e na Companhia das Índias Orientais (WINDSOR, 2009).

A Governança corporativa faz referência aos processos e técnicas utilizados para manter uma empresa ou organização. Também corresponde às relações entre a gestão, conselho de administração, acionistas e outras partes interessadas. A boa Governança Corporativa colabo-

ra para o desenvolvimento econômico sustentável, aprimorando o desempenho das empresas e aumentando o acesso a fontes externas de capital (IFC, 2010).

Segundo o IBGC (2009), os princípios básicos da Governança Corporativa são:

- **Transparência:** A transparência tem o objetivo de criar um clima de confiança nas relações internas e externas. Visa a comunicação interna e externa objetiva, clara, espontânea e oportuna. Vai além da legislação específica e expande-se para assuntos e fatores que possam ser do interesse dos públicos da organização, como valores e ações estratégicas.

Uma empresa deve disponibilizar as partes interessadas todas as informações que sejam solicitadas;

- **Equidade:** Busca erradicar atitudes ou práticas discriminatórias, consideradas como inaceitáveis. É correspondente ao tratamento justo e igualitário dos grupos minoritários, abrangendo pequenos acionistas, colaboradores, clientes, fornecedores e credores (IBGC, 2009). Deve existir tratamento justo para todos os sócios e demais partes interessadas;
- **Prestação de Contas (*accountability*):** Tem o intuito de atribuir a responsabilidade integral pelos atos praticados no decorrer de seus mandatos. Determina que todos os responsáveis pela Governança devessem prestar contas a quem os impostou ou lhes atribuiu as responsabilidades (IBGC, 2009). Assim, sócios, administradores, conselheiros fiscais e auditores devem prestar conta de sua atuação, assumindo integralmente as consequências de seus atos e omissões;
- **Responsabilidade Corporativa:** Deve haver zelo pela continuidade e sustentabilidade da organização, acoplando uma visão ao longo prazo e de sustentabilidade. É caracterizada por ser uma visão de longo prazo da estratégia organizacional, contemplando todos os seus relacionamentos. Visa a criação de riqueza, a geração de oportunidade de emprego, o estímulo ao desenvolvimento científico e a melhoria da qualidade de vida. Também inclui as questões ambientais e a defesa do meio ambiente (IBGC, 2009).

3.2 Governança em TIC

O termo Tecnologia da Informação (TI) começou a ser mais utilizado na década de 1980 em resposta a diversas mudanças tecnológicas e a maior utilização dos computadores deu início a um processo de propagação da informática nas organizações (LUNA, 2011).

Também segundo o Luna (2011), com o surgimento de novas ferramentas e tecnologias adotadas globalmente, e, a integração entre computação e comunicação, o termo TIC se transformou, adotando a sua denominação mais recente: Tecnologia da Informação e Comunicação (TIC).

Na visão de Gespol (2009, p. 38), Gestão de Tecnologia da Informação e Comunicação é base para todas as demais áreas gerenciais, pois representa o suporte necessário para que se possa desenvolver melhorias estratégicas no sistema de gestão e compartilhar informações para as tomadas de decisão.

Já para o Órgão de Tecnologia da Informação e Comunicação(OTIC), a governança de TIC é a liderança dos processos nas estruturas organizacionais com o fim de sustentar os objetivos e ampliar as estratégias na geração de valor da organização.

A governança de TIC é definida por Fernandes e Abreu (2006) como o compartilhamento das decisões de TIC com outros dirigentes da organização, além da implantação de regras, a organização e os processos necessários para dirigir o uso da tecnologia da informação pelos usuários, departamentos, divisões, negócios da organização, fornecedores e clientes, e também determinar de que forma a TIC deve fornecer os serviços para a empresa. Seu principal objetivo é o alinhamento da TIC aos requisitos do negócio, que tem como base a continuidade do negócio e no atendimento às suas estratégias e aos marcos de regulação externa.

Um conceito atual e legitimado que aponta para um alguns princípios éticos, os quais determinam atividades organizacionais com o intuito de gerar práticas transparentes, justas, sustentáveis, eficazes, eficientes e orientadas por objetivos comuns é a chamada “governança para aumento de valor”.

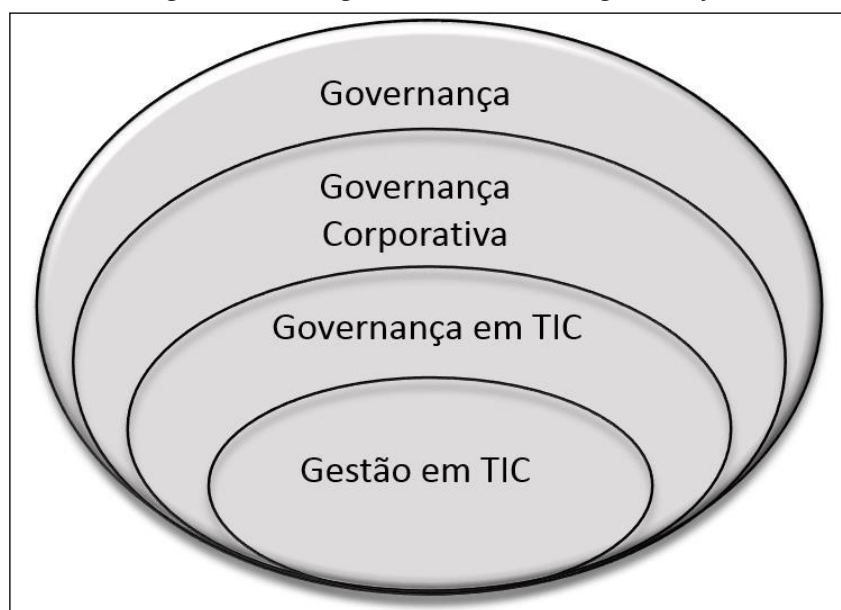
Os princípios de governança de TI, segundo Fernandes e Abreu (2008, p.46, apud WEILL E ROSS, 2004 e BROADBENT & KITZIS, 2005) “[...] tratam de papel da TIC para a empresa, informações e dados, padrões de arquitetura e serviços de TI, comunicações e ativos de TI”. Para o OTIC-PMESP, os princípios de:

- Direção e liderança - forneceu diretrizes para as mudanças através dos programas, projetos, iniciativas ou ações e assegurou que os objetivos sejam alcançados e minimizados os riscos e incidentes;
- Papéis e responsabilidades - definidos pelo controle interno do OTIC para gestão do portfólio de programas e projetos e portfólio de serviços;
- Prestação de contas - através da monitoração e avaliação, relatórios, controles, medições e auditorias, nos níveis estratégico, tático e operacional;

- Processos e atividades - processos que são realizados por meio de atividades que permeiam a estrutura hierárquico-funcional.

A figura 10 apresenta a abrangência dos conceitos de governança encontrados na literatura do assunto:

Figura 11 – Abrangência dos conceitos de governança



Fonte: Adaptada de Luna (2011)

Explica-se a figura como a gestão de TIC é direcionada pela governança em TIC, que, por sua vez, é orientada pela governança corporativa, esta condicionada aos princípios de governança.

A Governança de TIC apresenta dois tipos de componentes em seu fluxo (partindo do alinhamento estratégico até o resultado final); os componentes típicos da Governança de TIC e os de gestão e operacional. Segundo Fernandes e Abreu (2012), o primeiro componente (típico da Governança de TIC) é composto por:

- Riscos e observância das regras e leis;
- Gestão de mudança organizacional;
- Alinhamento estratégico;
- Entrega de valor;
- Gestão do desempenho;
- Comunicação;
- Gerenciamento de recursos

O Segundo componente da Governança de TIC (típico da gestão e operações) é composto por:

- Estratégia do negócio;
- Estratégia de TI;
- Plano de TIC (negócios);
- Plano de TIC (internos);
- Mecanismos de decisão;
- Portfólio de TIC (orçamento e investimentos);
- Clientes/usuários;
- Operações de serviços;
- Fornecedores;
- Resultados da TI.

O entrelaçamento dos dois componentes se transforma em orientações e limitações decisórias e de comportamento funcional, próprios de qualquer modelo de governança.

Por sua vez, Qumer e Henderson-Sellers (2007) apresentam os conceitos e descrição de governança de TIC para que possam ser úteis na definição de governança ágil que os autores desenvolveram. Tais conceitos, descrições e definições estão apresentados no Quadro 2 – Conceito e descrições de Governança de TIC

Quadro 2 – Conceito e descrições de Governança de TIC

Conceitos e Descrições de Governança de TIC	
Conceitos	Descrição
Estrutura de tomada de decisão	A liderança e tomada de decisão em governança em TIC é facilitada se tiverem uma estrutura organizacional adequada. Governança de TIC não é estrutura, mas precisa de uma para ser efetiva.
Estrutura de responsabilidade e prestação de contas	A Governança de TIC envolve e é responsabilidade da alta gestão. A prestação de contas, como princípio, é fundamental para monitorar e avaliar o desempenho.
Estrutura deve suportar processos. Maximizar o valor do negócio através de TIC Alinhamento entre TIC e negócios.	Processos relacionados com governança de TIC devem estar presentes em estrutura organizacional que os suportem. Os recursos de TIC devem ser usados para maximizar o valor do negócio. Objetivos, tanto de negócio quanto de TIC, devem estar alinhados e sincronizados para suportar o alinhamento Negócio-TIC.

(Continuação do Quadro 2)

Conceitos e Descrições de Governança de TIC	
Conceitos	Descrição
Governança em TIC como parte da governança corporativa	A governança de TIC é mais focada em TIC enquanto governança corporativa é focada nos negócios da organização. Mas a governança de TIC faz parte da governança corporativa;
A gestão de riscos	A gestão de riscos de TIC relaciona os riscos na tecnologia de informação, mas não os riscos relacionados com os negócios da organização;
Gestão de desempenhos	A utilização eficaz dos recursos de TIC para obter o desempenho ideal

Fonte: Adaptado de Qumer e Henderson-Seller (2007)

O subcapítulo abaixo apresentará os conceitos envolvidos pela Governança Ágil, sua definição, objetivos principais e os resultados esperados com a Governança Ágil.

3.3 Governança Ágil em TIC

Um dos primeiros trabalhos, senão o seminal, que apresentava princípios e práticas de engenharia concorrente foi desenvolvido por Takeushi e Nonaka, no artigo para a Harvard business Review, em 1986, e se intitulava “*The new product development Game*”. A partir dele, outros autores foram absorvendo seus conceitos e divulgou essa nova engenharia que necessitava, àquela época, de certas orientações típicas em governança.

3.3.1 Métodos ágeis e a necessidade de governança

Para inserir efetivamente práticas de governança em ambientes ágeis, são necessários liderança disseminada em toda a organização (e não apenas para o nível executivo), e esforços de colaboração. Mecanismos de avaliação de cooperação e colaboração devem incentivar e apoiar o aperfeiçoamento das equipes ágeis (auto avaliação, auto-organização e competência). Governança de TIC em organizações emergentes (tais como ambientes ágeis) requer práticas de governança que fluam através da organização, com a finalidade de garantir avanços (PA-TEL, 2002).

Em outras palavras, a governança deve ser projetada e comunicada em toda a organização de forma colaborativa (Colaboração multinível entre as partes interessadas: cliente, o conselho executivo e de gestão, gestor e equipes ágeis). De acordo com Meyer (2004), uma orga-

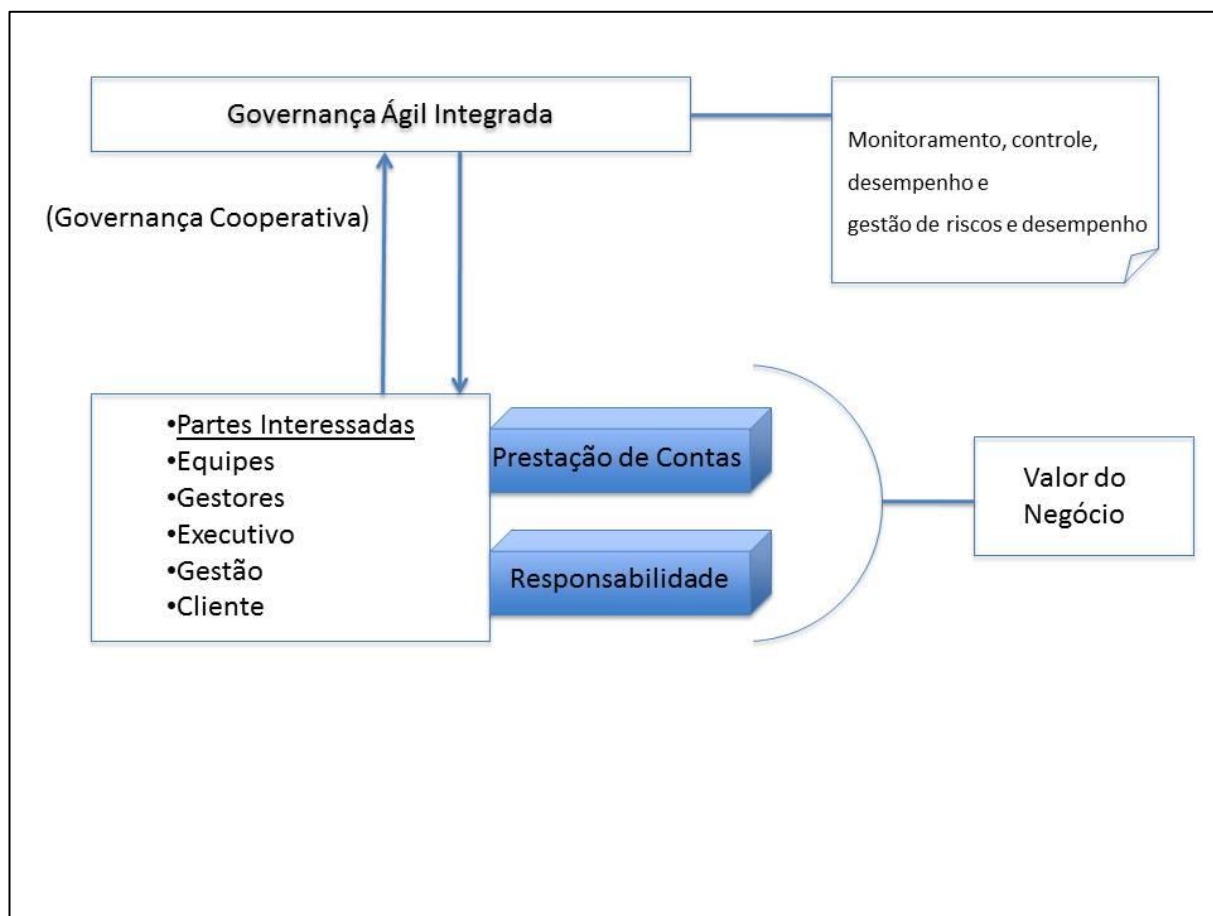
nização deve se concentrar na comunicação, ensino, convicção, refino e medição do sucesso de governança de TIC. Os fatores econômicos e de evolução ressaltam o incentivo de práticas das práticas retro mencionadas (auto avaliação, auto aperfeiçoamento, autodisciplina e auto responsabilidade) minimizar o custo de governança.

Para Weill e Ross (2004), equipes ágeis e competentes tomam decisões ou influenciam nas decisões que parecem bem aceitas em organizações pequenas ou médias e em desenvolvimentos, mas, a fim de intensificar um processo ágil de aplicação no desenvolvimento de projetos grandes complexos, uma gestão integrada ou modelo é indispensável.

Qumer e Henderson-Seller (2007) desenvolveram um modelo de responsabilidade, prestação de contas e governança do valor do negócio no contexto de desenvolvimento ágil, que introduz controle suficiente, disciplina e racionalidade para incrementar métodos de desenvolvimento ágil de software em grandes e complexos projetos (Fig. 2). Apesar de governança soar burocrática, é necessário integrar práticas de governança aos modelos com abordagens de desenvolvimento ágil de software, pois há retornos interessantes. Segundo pesquisas, a governança como mecanismo para uma melhor administração de TIC proporciona um retorno superior a 20%, se comparada a outras administrações sem esse mecanismo (WEILL e ROSS, 2004).

Com base em Qumer e Henderson-Seller (2007, p.3) a Figura 12 apresenta os elementos da governança ágil integrada e adequada às metodologias ágeis.

Figura 12 – Modelo de Governança Ágil Integrada



Fonte: Adaptado de: Qumer e Henderson-Seller (2007)

Governança integrada ágil se vale de cooperação e comunicação com as partes interessadas, e utiliza mecanismos de monitoramento, controle, gestão de riscos e desempenho.

Quanto ao que se entende por Governança Integrada de TIC, adotou-se a definição (QUMER E HENDERSIN-SELLER(2007):

Uma governança ágil integrada que envolve leveza, colaboração, comunicação orientada, econômica e envolve quadro de responsabilização eficaz, controles, processos, estruturas para maximizar o valor ágil de negócios, pelo alinhamento estratégico de metas negócio-ágil, desempenho e gestão de riscos.

A definição adotada pelos autores mencionados destaca as cinco principais perspectivas de governança para processos ágeis: leveza, colaborativa, comunicação orientada, econômico e evolução. O fator leveza, entendido como menor trabalho intensivo, salienta a governança, mas sem sobrecarregar desnecessariamente ambientes de desenvolvimento ágil de software. Pelo contrário, vai apoiar e ajudar a fortalecer as raízes de uma mentalidade ágil.

Os elementos de destaque da governança ágil estão apresentados na Tabela 12, onde se percebe as perspectivas ágeis mencionadas, as ferramentas de governança compostas por modelos, processos e estruturas, e as principais metas e operações necessárias.

Tabela 12-Elementos-chave da Governança integrada em um ambiente ágil

Elementos integrados de Governança ágil		
Perspectivas ágeis	Ferramentas de governança	Metas e operações
Leveza	Modelos Processos Estruturas	1. Controle
Colaboração		2. Prestação de Contas
Orientada para a comunicação		3. Maximizar valor do negócio
Econômica		4. Alinhamento estratégico de metas do negócio e metodologia ágil.
Em evolução constante		5. Gestão de riscos e desempenho

Fonte: Adaptado de: Qumer e Henderson-Seller (2007)

Em complemento, Luna (2011, p.77) ensina:

Governança ágil é o conjunto de comportamentos, conhecimentos, habilidades, recursos, métodos, técnicas, ferramentas que viabilizará o alcance e a manutenção dos objetivos estratégicos do negócio da organização através da contínua entrega de valor com utilidade e garantia, sendo propriedade conjunta de todos os setores da organização e encarregada de direcionar todos os envolvidos na obtenção e encarregada de direcionar todos os envolvidos na obtenção de diferenciais estratégicos, através dos princípios e do Manifesto da Governança Ágil.

Percebe-se que a governança para o desenvolvimento ágil de software (no original, *Governance for Agile Software Development* -GASD) foca na aplicação de governança no contexto do desenvolvimento de software e os problemas que tal aplicação possa acarretar.

A definição de Luna (2011) faz referência ao Manifesto da Governança Ágil, que será tratado a seguir.

3.3.2 O Manifesto da Governança Ágil

Com o intuito de ser criada uma mobilização em torno de valores e princípios para a governança foi escrito o manifesto para os desafios da Governança Ágil, sob uma abordagem

comportamental intitulado de Agile Governance Manifesto – Manifesto da Governança Ágil. O manifesto propõe a visão, a missão e os valores apresentados nos quadros a seguir (LUNA, 2011), onde seus principais elementos serão discutidos:

Quadro 3 Visão da Governança Ágil

Visão
Desejamos descobrir maneiras melhores de gerenciar as iniciativas de governança em nossas organizações, fazendo-o nós mesmos e ajudando outros a fazê-lo. Através desse trabalho, passamos a valorizar: 1. Pessoas e as relações entre elas mais que processos e ferramentas. 2. Valor útil entregue ao negócio através dos resultados alcançados mais que documentação abrangente. 3. Colaboração com o cliente e alinhamento com o negócio mais que negociação de contratos. 4. Responder às mudanças com agilidade e eficácia mais que seguir um plano. 5. Efetividade e eficácia das iniciativas mais que eficiência no processo. 6. Qualidade como atitude das pessoas mais que como requisito de um processo. Ou seja, mesmo havendo valor nos itens à direita, valorizamos mais os itens da esquerda.

Fonte: Extraído de Luna (2011)

Explica-se: o foco da Visão está nas pessoas e suas relações com ferramentas e processos, colaboração com o cliente e alinhamento do negócio e na agilidade das respostas às mudanças.

Por sua vez, entende-se como missão o propósito básico e escopo das operações de uma organização (BATEMAN & SNELL, 2006).

O Manifesto Ágil estabelece as seguintes atribuições à missão:

Quadro 4 – Missão da Governança Ágil

Missão
O time reconhece como sua principal atribuição fornecer “valor útil ao negócio”: • De forma incremental: através de uma abordagem iterativa, onde os problemas são divididos em partes menores e resolvidos. • Potencialmente entregável: onde, ao final de cada iteração, cada entrega tem que possuir valor reconhecidamente útil para o negócio.

Fonte: Extraído de Luna (2011)

Na missão estão enfatizadas a referência incremental e iterativa do processo e a entrega necessária a cada final de ciclo na iteração.

Por valores entendemos os princípios e crenças que servem como parâmetros para os critérios que definem comportamentos, atitudes e decisões que afetam as pessoas, e devem

permeiar as relações internas e externas das organizações. A Governança Ágil estabeleceu os seguintes, conforme Quadro 5 – Os valores da Governança Ágil propostos:

Quadro 5 – Os valores da Governança Ágil propostos

Valores
Valores: os indivíduos trabalham em equipe, acreditando nos seguintes valores: • Confiança mútua: a união é obtida através de credibilidade atribuída a cada membro do time. O sentimento de equipe reforça o vínculo e potencial do time. • Cooperação: a colaboração é o aspecto essencial para o sucesso, e cada membro deve dar o melhor de si para o alcance dos resultados do time. Colaboração gera colaboração e este ciclo virtuoso se retroalimenta. • Compromisso: o empenho de cada membro com as metas definidas pelo time deve ser visto como se a “melhor parte” de si mesmo estivesse sendo “doada” para o alcance de um bem comum. Esta abordagem gera um vínculo que reforça os demais valores mencionados. • Transparência: as iniciativas devem ser difundidas entre o time, as boas atitudes e práticas devem ser incentivadas e o andamento das ações deve ser amplamente divulgado. • Honestidade: a atitude de apontar o erro e propôr melhorias. A atitude de não fazer “corpo mole” nem se omitir perante os problemas. A atitude de encarar os problemas de frente e não ter medo de pensar diferente é um valor essencial a cada membro do time. • Auto-organização: o comportamento do time deve ser caracterizado por uma superposição de ordem, auto-organização e uma inteligência coletiva, que é maior que a soma das partes. O time deve exibir habilidades para se adaptar a ambientes complexos e dinâmicos, onde as mudanças são únicas e constantes. • Comunicação: a comunicação é um fator crítico de sucesso em projetos de qualquer natureza. Os canais de comunicação devem ser claros para todos os membros do time. O diálogo face a face deve ser incentivado e priorizado, sempre que possível. A informação deve ser patrimônio coletivo do time. As relações interpessoais entre os membros do time, assim como entre o time, a equipe de negócio e os clientes, devem ser incentivadas. • Simplicidade: as arquiteturas das soluções devem ser tão simples quanto possível. O time deve procurar resolver as questões emergentes através da racionalização entre os resultados desejados e os recursos disponíveis no momento. Perder tempo elaborando uma solução mais completa para um problema que pode ser resolvido com uma solução mais simples barata e que não é ágil. • Feedback: é essencial encorajar o time a realizar periodicamente auto avaliações, bem como análises sobre lições aprendidas, assim como uma reflexão do que pode e deve ser melhorado. Mesmos os acertos devem ser encarados nestas avaliações, contudo sempre sob o ponto de vista apreciativo. • Coragem: é preciso ter coragem para reconhecer que estava “errado”, “parar”, “jogar tudo fora” e “refazer” quando necessário. Coragem para ajudar o companheiro a completar seu trabalho, mesmo já tendo feito o seu. Coragem para mudar de ideia, para inovar. Coragem para reconhecer erros e assumir riscos. E principalmente coragem para aplicar todos os valores e princípios deste manifesto.

Fonte: Extraído de Luna (2011)

Observa-se que estão presentes termos relacionados a princípios e comportamentos esperados, como coragem, simplicidade, compromisso, e confiança mútua, dentre outros. O

Manifesto Ágil também enuncia doze princípios para servirem de referência aos processos da Governança Ágil, assim como mostra o Quadro 6:

Quadro 6 – Princípios da Governança Ágil

Princípios
1. A prioridade é o alcance dos objetivos do negócio e a satisfação dos clientes mediante o rápido e contínuo fornecimento de iniciativas que agreguem valor ao negócio.
2. As mudanças fazem parte de uma natureza do negócio. Responder de forma ágil e eficaz é um compromisso da equipe e uma habilidade que deve ser desenvolvida e cultivada. Gerar vantagem competitiva para nossos clientes e organizações é nossa missão.
3. Fazer entregas frequentes e contínuas de valor útil ao negócio através de resultados que se concretizem a partir de poucas semanas a poucos meses, sempre procurando o menor intervalo entre as entregas, é um compromisso da equipe.
4. As pessoas de negócio (executivos) e as pessoas técnicas devem trabalhar juntas frequentemente ao longo de todo o projeto. As pessoas de negócio devem priorizar as entregas, partindo dos princípios da utilidade e do impacto para o negócio.
5. Incentivar e cultivar a motivação, a cooperação, a capacitação contínua e o trabalho colaborativo são compromissos da equipe. A confiança mútua e o respaldo de alta direção da organização são fatores imprescindíveis para o sucesso das iniciativas.
6. O diálogo face a face e as relações interpessoais são as mais eficientes e eficazes formas de comunicação, quer seja entre os membros da equipe, quer seja entre a equipe e o cliente. A informação é um patrimônio coletivo da equipe.
7. Valor útil entregue é a principal medida de progresso. Fazer “aquilo” que gere o maior impacto positivo no negócio, no menor intervalo de tempo, com os recursos disponíveis no momento são critérios para priorizar as iniciativas.
8. Os processos ágeis promovem um desenvolvimento sustentável. O time, o cliente e a organização devem ser capazes de manter um ritmo de trabalho constante e com qualidade por tempo indeterminado.
9. A qualidade técnica e o desenho (design) das iniciativas devem continuamente refinados e aperfeiçoados. Em função de limitações nos recursos disponíveis, retrabalhos podem ser necessários. Contudo, arquitetura baseada em componentes e desenhos simples melhoram a capacidade e a velocidade em responder de forma ágil às mudanças.
10. A simplicidade é imprescindível, mas devem estar adequadas à utilidade e garantia produzida para o negócio. Soluções complexas que requerem arquiteturas rebuscadas e recursos em demasia são candidatas ao fracasso. É preciso estar atento ao que realmente é essencial ao negócio, evitando perda de tempo com features que estão pouco ou nunca usadas.
11. A inteligência coletiva e colaborativa da equipe, através de sua pro atividade, auto-organização e aproximação com cliente e pessoas de negócio da organização, gerará as condições ideais para a proposição e construção das melhores arquiteturas, requisitos e desenhos de soluções para o negócio.
12. É um compromisso de a equipe refletir sobre como se tornar mais eficaz e gerar mais valor ao negócio da organização, em intervalos regulares e frequentes, ajustando seu comportamento sempre que necessário para alcançar os objetivos estratégicos do negócio.

Fonte: Extraído de Luna (2011)

Com o avanço tecnológico se espera que o Manifesto Ágil seja a referência comportamental para aplicar o conhecimento das metodologias que englobam o contexto de governança ágil.

No próximo tópico será apresentado um modelo desenvolvido no Brasil, especificamente na Universidade Federal de Pernambuco (UFPE).

3.3.3 Exemplo de Modelo de apoio a Governança Ágil em TIC: MANGVE

O MAnGve (Modelo Ágil no apoio à Governança em TIC) teve origem de um trabalho de mestrado em Ciência da Computação da UFPE e, segundo Luna (2011) é definido como:

Um framework ágil para implantação e melhoria dos processos de governança e serviços de TIC em organizações de qualquer natureza e tamanho que provê uma abordagem de ação prática, adaptativa, orientada a pessoas, de maneira flexível e iterativa, buscando continuamente a simplicidade.

Tal modelo utiliza a expressão “Do Caos à Lama” para caracterizar a evolução da organização com os seus processos. Tal expressão significa passar por uma situação de renovação da cultura organizacional, com o foco em gerenciamento de serviços, processos e indicadores claros, para garantir a continuidade dos processos de negócio da organização (LUNA, 2011).

O MAnGve visa uma reflexão sobre a necessidade de adaptação à realidade das organizações, assim como o comportamento da equipe comprometida. Tem como base os princípios do Manifesto Ágil e o Manifesto da Governança Ágil, ambos visam que todos os envolvidos trabalhem com alguns valores, os quais serão descritos abaixo:

- **Confiança mútua:** onde cada membro do time deve ter credibilidade atribuída. Ter uma equipe confiável reforça o vínculo e melhora o potencial do time;
- **Cooperação:** cada membro deve ser prestativo no processo de desenvolvimento, dar o melhor de si para o alcance de bons resultados deve ser uma filosofia pessoal da equipe;
- **Compromisso:** metas definidas pelo time devem ser definidas para cada membro como se a o mesmo estivesse trabalhando com o máximo de potencial para o alcance de um bem comum;

- **Transparência:** todas as iniciativas devem ser difundidas, boas atitudes e práticas devem ser incentivadas e o andamento das ações deve ser amplamente divulgado entre a equipe;
- **Honestidade:** ter uma boa atitude a respeito de críticas, erros e melhorias são essenciais para evitar problemas e complicações;
- **Auto-organização:** todos os membros do time devem ser organizados e devem ter uma inteligência coletiva no projeto;
- **Comunicação:** a comunicação é um fator em qualquer projeto de qualquer natureza. Devem existir canais de comunicação claros para todos os membros do time;
- **Simplicidade:** as soluções e os produtos devem ser o mais simples possível. A equipe deve saber distribuir de maneira eficiente os recursos disponíveis para alcançar resultados satisfatórios;
- **Feedback:** a equipe deve realizar periodicamente auto avaliações, visando utilizar lições aprendidas, assim como uma reflexão do que pode e deve ser melhorado para otimizar os resultados da produção;
- **Coragem:** é necessário ter a coragem de reconhecer algo que está sendo feito errado ou algo que precisa ser refeito quando necessário.

3.3.4 Conclusão do capítulo

Neste capítulo, vimos que a preocupação da governança em um ambiente ágil é identificar e implementar iniciativas para dar poder através da competência às equipes, para exercerem o direito de tomar decisões, melhorá-las por novas informações percebidas, mas sendo responsáveis pelas decisões tomadas, a fim de alcançar o valor desejado para o negócio (QUMER e HENDERSON-SELLER, 2007).

Inicialmente a governança ágil foi prevista para aplicação em pequenas e médias empresas desenvolvedoras de softwares, e a pergunta é: será que seus preceitos podem ser utilizados em grandes projetos?

A discussão é longa, pois embora haja implicações de modelos de desenvolvimento ágil de software, métodos, estruturas e teorias, assim como o interesse que eles causaram na indústria, há também questionamentos sobre a capacidade destes métodos serem usados em projetos de grande dimensão e complexidade.

Mas, para Qumer e Henderson-Seller (2007), pode-se afirmar que a principal razão por trás de certa falta de capacidade possa ser a falta de práticas de governança aplicadas ao desenvolvimento ágil de ambientes de software, uma vez que o volume de negócios e desempe-

nho de uma empresa de software são diretamente influenciados por práticas de governança de TIC.

Porém, nem tudo são rosas quando se pensa em governança ágil. Há problemas e desafios relacionados aos fatores críticos em sua implantação. Segundo Silva(2013), em sua monografia intitulada “Investigando Atributos para Governança no Desenvolvimento Ágil de Software” apresenta os seguintes pontos de discussão, fundamentados em extensa bibliografia.

Para Silva (2013), são fatores críticos:

- **Experiência da equipe com desenvolvimento ágil:** Práticas de desenvolvimento ágil visam à priorização do cliente, incorporando entregas rápidas e comunicação constante. Uma empresa que deseja aderir a governança ágil e já possui experiência com práticas ágeis, normalmente encontra maior facilidade neste processo (ABBAS; GRAVELL; WILLS, 2010);
- **Viabilizar a participação ativa do cliente no projeto:** No que diz respeito a projetos ágeis, a participação ativa do cliente é algo que é extremamente relevante (MALLADI; DOMINIC; KAMIL, 2011). Para a governança ágil esse fator também é muito importante visto que as tomadas de decisão na empresa podem ser algo lento e essa comunicação dará as partes interessadas uma grande visibilidade ao trabalho realizado pela equipe. Isto pode assegurar que o produto final está sendo construído de forma correta (QUMER, 2007; TUDOR; WALTER, 2006). Como as partes interessadas estão controlando o orçamento e cronograma, a equipe poderá ser dirigida mais efetivamente (ABBAS; GRAVELL; WILLS, 2010);
- **Considerar outros fatores impactantes na capacidade de escalar técnicas de desenvolvimento ágil de software:** A abordagem para a governança é apenas um dos vários fatores que impactam a capacidade de escalar técnicas de desenvolvimento ágil de softwares o que de certa forma traz algum impacto direto também a Governança Ágil (AMBLER, 2009);
- **Realizar a Transição do Método Tradicional para o Ágil:** De acordo com o artigo, diante de seu estudo de governança em empresas de desenvolvimento de software foram descobertos vários desafios. Dois deles estavam diretamente relacionados com a funcionalidade do modelo SDG: falta de *feedback* e falta de priorização de temas do negócio. Os demais estavam relacionados a própria transição da empresa que traba-

lhava de maneira tradicional para o efetivo trabalho com os princípios ágeis(LEHTO; TKK; RAUTIAINEN, 2009);

- **Alinhar os Processos ágeis com a Governança da Empresa:** Governança ágil, não só garante o alinhamento com a estratégia e alocação eficiente de recursos, como, adicionalmente, reduz o risco de sobreposição de projetos. Por motivos como estes, os processos ágeis da organização não podem funcionar fora do processo de governança(HOUNIOUS; CLARK, 2006);
- **Tornar claro as definições de Software Development Governance - SDG e boa Governança:** Organizações de desenvolvimento de software que desejam alcançar com sucesso seus objetivos estratégicos precisam de um entendimento explícito sobre governança e como abordá-la. Um desafio significativo para essa compreensão é que os gerentes de projetos de desenvolvimento de *software*, os desenvolvedores e as organizações parecem confusos sobre o que exatamente é SDG e o que constitui uma boa governança(CHULANI et al., 2008);
- **Focar a Software Development Governance - SDG na realização dos objetivos estratégicos:** Uma boa SDG tende a ser focada na realização dos objetivos estratégicos e apoia organizações a realizar seus objetivos, levando a benefícios de negócios e criação de valor. Além disto, uma boa SDG apoia também a mitigação de riscos e a melhoria na eficácia da equipe, através de boa comunicação eficaz e de medição e controle(CHULANI et al., 2008);
- **Definir Modelos de Negócio para auxílio à adoção da governança:** Modelos de negócio fornecem uma teoria de como estabelecer limites para a empresa, de como criar valor, e como organizar sua estrutura interna e governança (DOZ; KOSONEN, 2010). Dessa forma, como a governança ágil visa maximizar o valor de negócio, ter um modelo de negócio bem definido na empresa, no contexto de governança ágil, melhora o retorno sobre investimentos, visto que qualidade e satisfação do cliente fazem parte dos princípios da governança ágil (QUMER, 2007);
- **Definir Indicadores de desempenho e ações de Intervenção:** De acordo com o artigo em questão, providos com *Key Performance Indicators* (KPIs), os gerentes podem tomar uma ação se estas começam a indicar a perda de controle do projeto ou mesmo o objetivo de satisfação mostre-se em perigo(CHENG; JANSEN; REMMERS, 2009);

- **Disponibilizar indicadores a toda organização:** De acordo com o artigo, uma lição aprendida destacada no artigo é que *Key Performance Indicators* (KPIs) não devem ser apenas visíveis para a gestão, mas também ser disponibilizado para a organização como um todo (CHENG; JANSEN; REMMERS, 2009).

Para ser bem sucedida, segundo diferentes trabalhos, a governança ágil se depara com diferentes desafios (SILVA, 2013).

- **Escalar governança ágil para organizações de grande porte:** A agilidade tem sido observada para trabalhar bem em organizações de pequeno porte, ou para projetos menores, e frequentemente terminam por necessitar de uma melhor organização de governança, avaliação de investimento e controle (HOUNIOUS; CLARK, 2006). Com relação a organizações maiores, que possuem projetos grandes e complexos, implantar governança ágil se torna um desafio visto que as mudanças ditadas por esse tipo de governança terão um efeito muito maior nesses tipos de projetos (QUMER, 2007);
- **Implantar governança ágil em organizações que não tem experiência ágil:** Muitas empresas são novas na adoção de um modelo de governança ágil e essa implantação requer novas habilidades em tecnologia, arquitetura, desenvolvimento e projeto de infraestrutura (HAKI; FORTE, 2010). Além disto, é apontado que se faz necessário uma boa organização para se trabalhar com as equipes que não são "ágeis", manter os planos de entregas e instruir a gerência sobre os processos ágeis para atender suas expectativas sobre o orçamento e entrega (HOUNIOUS; CLARK, 2006);
- **Tornar a comunicação efetiva entre os *Product Owners* e o time:** Em estudos sobre projetos ágeis, foi observado a falta de comunicação entre os *product owners* e o time, incluindo a própria ausência de *feedback*, ocasionando demora nas entregas e transtorno para o cliente (LEHTO; TKK; RAUTIAINEN, 2009; LOEWENSTERN et al., 2010);
- **Conseguir uma Governança adequada para garantir resultados benéficos tanto para o cliente quanto para o fornecedor:** Contratos de terceirização começam com um único projeto e com o tempo essa quantidade tende a aumentar em sua grande maioria. Este crescimento de projetos requer uma governança adequada para garantir

resultados benéficos tanto para o cliente quanto para o fornecedor (CHENG; JANSEN; REMMERS, 2009);

- **Criar uma Abordagem inovadora para Governança:** Atualmente uma nova abordagem para governança é necessária para efetivamente suportar novos paradigmas de desenvolvimento (AMBLER, 2009);
- **Estruturar uma lista abrangente e totalmente completa de KPI's e Intervenções:** O campo de desenvolvimento ágil de software ligado a KPIs e intervenções ainda não foi tão explorado. Desta maneira mostra-se quase que impossível criar uma lista abrangente e totalmente completa de tais mecanismos (CHENG; JANSEN; REMMERS, 2009);
- **Utilizar o modelo SDG em cenários mais complexos:** Métodos ágeis, por exemplo, Scrum e XP, fornecem de maneira própria um modelo pré-definido de SDG. Nestes modelos originalmente são assumidas situações simples, onde uma única equipe está desenvolvendo um produto de cada vez. No entanto, segundos estes autores, quando uma grande organização quer ser ágil e usar processos ágeis, pouca orientação é fornecida para modificar o simples e básico modelo ágil SDG para atender a uma situação mais complexa, mas preservando os princípios ágeis. Alguma literatura emergiu nos últimos anos para tratar ágeis no contexto de grandes empresas, por exemplo, mas pesquisas empíricas são ainda escassas sobre o assunto (LEHTO; TKK; RAUTIAINEN, 2009);
- **Fazer com que a agilidade não deva ser definida em termos apenas de práticas:** No que diz respeito ao apoio de organizações de software na adoção de processos ágeis de desenvolvimento de software é importante primeiro definir o contexto através do uso do seu modelo com vários fatores. Somente a partir daí pode-se compreender qual dimensão (s) do projeto falharam de acordo com o contexto ideal descrito pelo "*agile sweet spot*". A idéia é não forçar a adoção de várias práticas ágeis por puro modismo (KRUCHTEN, 2011);
- **Fazer com que os princípios ágeis funcionem fora do "*agile sweet spot*":** Esse ponto prega um ambiente que fazendo o desenvolvimento em campos conhecidos e para sistemas não críticos de segurança, em um ambiente muito volátil, arquitetura do sistema definida e estável e os papéis para governança simples. Isso não quer dizer que

as práticas ágeis não funcionam fora deste contexto, mas que muitos são os desafios, necessitando de adaptação, e, em alguns casos, isto pode não ser adequado (KRUCHTEN, 2011);

- **Evitar conflito entre a estrutura do time e os princípios ágeis:** Como desafios focados em papéis e responsabilidades, foram observados em seus estudos: conflito entre a estrutura do time e os princípios ágeis e a falta de prioridades entre os temas do negócio (LEHTO; TKK; RAUTIAINEN, 2009; LOEWENSTERN et al., 2010).

Finalmente, os grandes problemas a serem neutralizados, segundo o autor, apresentam-se abaixo:

- **Tentar realizar Governança de Projetos Ágeis a partir de Mecanismos tradicionais:** Os métodos ágeis podem ser aplicados com sucesso variável fora da aplicabilidade sob a qual foram realmente criados. Por exemplo, é possível customizá-los para projetos maiores ou em equipes distribuídas. Os fatores contextuais que apresentam maiores riscos de insucesso em projetos ágeis de acordo com sua experiência são: Tamanho, grandes sistemas com uma falta de foco arquitetônico, desenvolvimento de *software* não impulsionado pela demanda dos clientes, falta de apoio das partes interessadas, uso da governança tradicional, *Novice team*, restrição muito alta em algum atributo de qualidade (sistema de segurança crítica, restrições de tempo real), (KRUCHTEN, 2011);
- **Carência de discussão sobre Governança em TI no contexto do Desenvolvimento Ágil de *Software*:** Governança ágil é uma área relativamente recente, especialmente sob o contexto de desenvolvimento ágil de *software*. Com pouca discussão sobre o assunto, a aceitação de boas práticas e técnicas da governança ágil, ditadas para orientar os processos de empresas e organizações, acaba sendo dificultada (QUMER, 2007);
- **Realizar Governança baseada nos princípios ágeis ainda é incipiente:** Segundo o autor, a governança não é algo que é comumente associada ainda a projetos de Desenvolvimento ágil de *software*. Projetos ágeis, como todos os outros projetos, podem e devem ser governados. Infelizmente a literatura ágil raramente menciona governança e quando isso acontece tem pouco a dizer sobre o tema, o que provavelmente demonstra um reflexo do estado atual de maturidade neste contexto (AMBLER, 2009);

- Escassez de Literatura e Pesquisa Empírica demonstrem uso efetivo dos princípios ágeis em situações complexas: Os métodos ágeis, por exemplo, *Scrum* e *XP*, fornecem de maneira própria um modelo pré-definido de *Software Developer Governance* (SDG). Nestas maneiras originalmente são assumidas situações simples, onde uma única equipe preocupa-se com o desenvolvimento de um produto de cada vez. No entanto, quando uma grande organização quer ser ágil e usar processos ágeis, pouca orientação é fornecida para modificar este simples cenário ágil no modelo de SDG para atender a uma situação mais complexa. Alguma literatura emergiu nos últimos anos para tratar ágeis no contexto de grandes empresas, por exemplo, mas pesquisas empíricas ainda são muito escassas nesta área (LEHTO; TKK; RAUTIAINEN, 2009);
- Dificuldade em obter uma visão unificada para *Governance in Distrusted Agile Software Development*: Conseguir uma visão unificada pra realizar Governança no Desenvolvimento Ágil de *Software* Distribuído pode ser um problema. Enquanto esta visão unificada pode ser um fator encorajador, esta mesma visão padrão pode atrapalhar projetos por causa da dependência entre as entidades remotas e a entidade matriz em situações que envolvem principalmente tomadas de decisão coletivas sobre questões críticas. Além disso, a ausência de colaboração contínua na compreensão dos processos de engenharia, os métodos e protocolos podem causar desconexão entre a entidade matriz e as entidades locais (BAVANI, 2011).

As principais conclusões depois do levantamento realizado é que, por ser metodologia relativamente nova, sem muitos modelos alternativos, a governança ágil:

- a. Grandes projetos possui embasamento nos modelos de maturidade apresentados neste trabalho.
- b. Adota princípios de governança tradicionais, com a inserção do elemento leveza, aqui considerado como maior facilidade ou menor esforço em sua realização.
- c. Está inserida nos princípios de governança, em especial, os de prestação de contas e responsabilidade.
- d. Um dos desafios para o sucesso de sua implantação diz respeito à comunicação; outro diz respeito é aplicar seus princípios aos grandes projetos ou organizações de grande porte.
- e. Alguns dos seus problemas se relacionam à aplicação de mecanismos de governança

tradicionais sem considerar os aspectos relacionados a desenvolvimento ágil; outro problema importante relacionado com o desafio anterior é a escassez de pesquisas relacionando o uso de princípios ágeis em situações complexas.

- f. Apesar de ser ainda inédita em certos círculos de produção de software, em especial,

O próximo capítulo trata das considerações finais.

4 CONSIDERAÇÕES FINAIS

Neste trabalho foi apresentada uma introdução sobre conceitos de governança, destacando sua importância em uma organização que visa se destacar no mercado do trabalho. Também foram apresentados alguns modelos de maturidade e de metodologia ágil, em especial, o Scrum e o XP.

A seguir, foram apresentados diversos conceitos relacionados à governança, finalizando com o conceito de governança ágil, com definições, escopo, razões e a apresentação de um modelo desenvolvido pela UFPE: O MANGVE. Ponto de destaque foi a apresentação da listagem de alguns problemas, desafios e fatores críticos para o sucesso na implantação da governança ágil.

Com isso, os objetivos deste estudo foram alcançados.

Porém, o trabalho apresenta limitações de ordem metodológica e de fundamentação teórica especificamente relacionada ao tema. Uma principal dificuldade foi relacionar princípios, valores e missão presentes no Manifesto Ágil com o conceito de governança, e os modelos de maturidade – o que faz alertar para possível insuficiência de interpretação. Outra dificuldade está relacionada com a carência de discussão sobre governança no contexto de desenvolvimento ágil, como bem lembrou (SILVA, 2013), parafraseando Qumer e Henderson-Seller (2007).

Uma vez que é possível aprofundar estudos sobre o tema, sugere-se a realização de estudos de caso múltiplos sobre “como aplicar” a governança ágil em grandes empresas.

5 Referências Bibliográficas

- Amid, A. (2012). *Um Quadro de Avaliação híbrida de CMM e COBIT para a melhoria do Desenvolvimento da Qualidade de Software*.
- APMG International. (2015). <http://www.apmg-international.com/br/consultoria/modelo-maturidade.aspx>. (APMG Group) Acesso em 26 de 02 de 2015, disponível em APMG International: <http://www.apmg-international.com/en/home.aspx>
- BATEMAN, T. S., & SNELL, S. A. (2006). *Administração - Novo cenário Competitivo*. S.Paulo: Atlas.
- BASILI, V. R.; CALDIERA, G.; ROMBACH, H. D. The goal question metric approach. v. 2, p. 1–10, 2006.
- BECK, K. et al. Agile Manifesto. Disponível em: <<http://www.agilemanifesto.org>>. Acesso em: 10 set. 2013.
- BECK, K.; ANDRES, C. *Extreme Programming Explained*. 2. ed. Boston: Addison-Wesley, 2004.
- COBIT. (2007). Governance Institute.
- Franciscani, J. d. (2012). *CMMI e MPS.BR- Um estudo comparativo*.
- Instituto Antônio Houaiss. (04 de 2007). Dicionário Eletrônico Houaiss de Língua Portuguesa. *HOUAISS*. S.Paulo, SP: Objetiva.
- COMISSÃO SOBRE GOVERNANÇA GLOBAL. Nossa Comunidade Global. Rio de Janeiro: [s.n.]. p. 2.
- CHENG, T.-H.; JANSEN, S.; REMMERS, M. Controlling and monitoring agile software development in three dutch product software companies. 2009 ICSE Workshop on Software Development Governance, p. 29–35, maio. 2009.
- DUBINSKY, Y.; KRUCHTEN, P. 2nd workshop on software development governance (SDG). 2009 31st International Conference on Software Engineering - Companion
- DICIO. Dicio - Dicionário de Português. Disponível em: <<http://www.dicio.com.br/>>. Acesso em: 22 nov. 2013.
- DOMINGUES, L.; GOMES, J. A. C. Management Model Proposal for Portuguese Public Administration Shared Services. 2011 IEEE World Congress on Services, p. 25–32, jul. 2011.
- DOZ, Y. L.; KOSONEN, M. Embedding Strategic Agility. *Long Range Planning*, v. 43, n. 2-3, p. 370–382, abr. 2010.
- FERNANDES, A.; ABREU, V. *Implantando a Governança de TI: da Estratégia à Gestão dos Processos e Serviços*. Rio de Janeiro: Brasport, 2006. p. 439

- FILHO, P.; PÁDUA, W. DE. Engenharia de Software: fundamentos, métodos e padrões. 3. ed. Rio de Janeiro: [s.n.].
- KERZNER, H. (2006). *Gestão de Projetos - As melhores práticas* (2a. ed.). Porto Alegre: Bookman.
- LUNA, A. Implantando Governança Ágil - MAnGve. 2011.
- LUNA, A. Do Caos à Lama: Avanço na eficácia do Gerenciamento de Serviços em TIC utilizando o MAnGveCONIP - Congresso de Informática e Inovação na Gestão Pública. Anais...2011Disponível em: <http://www.mangve.org/downloads/CONIP2011_DocaosàLama_palestra_AlexandreLuna_VersãoPublicadaemPDF.pdf>. Acesso em: 12 set. 2013
- LUNA, A. et al. Agile Governance: A Systematic Literature Review. 2013.
- Mellon, C. (2006). *Melhoria de processos visando melhores produtos*. Software Engineering Institute.
- MATHIASSEN, L.; VAINIO, A. M. Dynamic Capabilities in Small Software Firms: A Sense-and-Respond Approach. IEEE Transactions on Engineering Management, v. 54, n. 3, p. 522–538, ago. 2007.
- O'BRIEN, J. A. (2001). *Sistemas de informação e as decisões gerenciais na era da internet*. (C. K. Moreira, Trad.) S.Paulo: Saraiva.
- Oliveira, C. d. (2008). Comparando CMMI e MPS.BR: As vantagens e desvantagens do modelo de qualidade do Brasil.
- PRIKLADNICKI, R. Padrões de Evolução na Prática de Desenvolvimento Distribuído de
- Silva, V. M. (2013). *Investigando Atributos para Governança no Desenvolvimento Ágil de Software*. João Pessoa.
- SIQUEIRA, J. (2015). *O modelo de maturidade de processos*. Acesso em 26 de 02 de 2015, disponível em www.ibqn.com.br: <http://www.abcq.org.br/14/o-modelo-maturidade-processos.html>
- SOFTEX. (2006). *MPS.BR - Melhoria de processo de software brasileiro - Guia Geral* (Versão 1.1 ed.). Brasília: SOFTEX.
- Software em Ambientes de Internal Offshoring : Um Modelo de Capacidade Dados Internacionais de Catalogação na Publicação (CIP). 2009.
- SCHWABER, K.; SUTHERLAND, J. Guia do Scrum. Disponível em: <[https://www.scrum.org/Portals/0/Documents/Scrum Guides/Scrum Guide - Portuguese BR.pdf#zoom=100](https://www.scrum.org/Portals/0/Documents/Scrum%20Guides/Scrum%20Guide%20-%20Portuguese%20BR.pdf#zoom=100)>. Acesso em: 9 set. 2013.
- SEI - SOFTWARE ENGINEERING INSTITUTE. CMMI for Development (CMMI-DEV). Disponível em: <http://www.sei.cmu.edu/library/assets/whitepapers/CMMI-DEV_1-2_Portuguese.pdf>. Acesso em: 12 set. 2013.

SILVA, V. M. et al. Uma análise comparativa entre Modelos de Maturidade baseados em Gerenciamento de Serviços de TI. 2013.

SOMMERVILLE, I. Engenharia de Software. [s.l.] Addison-Wesley, 2007. v. 8

WEILL, P.; ROSS, J. Governança de TI: Tecnologia da Informação. São Paulo: M.Books, 2006.

YIN, R. K. Case Study Research: Design and Methods. 3. ed. London: Sage Publications, 2003.

ZHANG, Z.; VALOTA, O. Methodology to form agile strategies in manufacturing organisations. Proceedings. 2005 IEEE International Engineering Management Conference, 2005., v. 1, p. 410–414, 2005.