

Consórcio Doutoral

Proposta de uma Estratégia para o Uso de Modelos de Maturidade em Equipes Ágeis de desenvolvimento de software: diretrizes para o desenvolvimento dos aspectos humanos

Nome do proponente:

Email:

Telefones de contato:

Programa de Pós-Graduação

Orientador:

Área Temática: Gestão de Projetos

Proposta de uma Estratégia para o Uso de Modelos de Maturidade em Equipes Ágeis de desenvolvimento de software: diretrizes para o desenvolvimento dos fatores humanos

Resumo: Os estudos de equipes ágeis maduras sugerem que é necessário ter foco em fatores humanos e sociais para ter sucesso, sendo impossível excluir tais fatores durante o desenvolvimento de software. Em paralelo, o aumento da exigência por qualidade em produtos de software fez com que os desenvolvedores seguissem modelos, uma vez que as metodologias tradicionais para gerenciamento de projetos já não atendem plenamente às necessidades gerenciais. Dentro desse contexto, surgiram padrões, frameworks, normas e regulamentos para a implementação de modelos de maturidade para gestão de projetos e, mais especificamente, para gestão das pessoas em empresas de desenvolvimento de software. Porém, esses modelos estão alinhados aos modelos tradicionais de desenvolvimento de software. Diante do exposto, o presente trabalho propõe, utilizando os arcabouços existentes e o pensamento de agilidade, uma estratégia para o uso de modelos de maturidade para o desenvolvimento de fatores humanos em equipes ágeis de desenvolvimento de software. Por ser uma pesquisa descritiva, visa a construção da solução proposta utilizando metodologia de Design Science Research, fundamentando a condução da pesquisa. Os resultados permitirão uma melhor compreensão em torno do fenômeno explorado, além da definição de uma estratégia para o uso dos modelos de maturidade no desenvolvimento dos fatores humanos em equipes ágeis de desenvolvimento de software.

Palavras-chaves: Metodologias Ágeis, Modelos de Maturidade, Fatores Humanos, Equipes de desenvolvimento de software.

Proposed Strategy for the Use of Maturity Models in Agile Software Development Teams: Guidelines for the Development of Human Aspects

Abstract: Studies of mature agile teams suggest that it is necessary to focus on human and social factors to be successful, making it impossible to exclude such elements during software development. In parallel, the increased demand for quality in software products has led developers to follow models, since traditional methodologies for project management no longer adequately meet managerial needs. Within this context, standards, frameworks, norms, and regulations emerged for the implementation of maturity models for project management and, more specifically, for people management in software development companies. However, these models are in line with traditional software development models. Given the above, the present work proposes, using existing frameworks and agility thinking, a strategy for the use of maturity models for the development of human factors in agile software development teams. As it is descriptive research, it aims to build the proposed solution using Design Science Research methodology, supporting the conduct of the study. The results will allow a better understanding of the phenomenon explored, as well as the definition of a strategy for the use of maturity models in the development of human factors in agile software development teams.

Keywords: Agile Methodologies, Maturity Models, Human Factors, Software Development Teams.

1. Introdução

As inúmeras mudanças culturais, tecnológicas, políticas e econômicas que ocorrem no mundo atual impõem desafios aos projetos em um contexto geral, levando as organizações a buscarem novas formas de atuação e a revisarem suas estratégias. Essas mudanças e tendências trazem novas abordagens, mais flexíveis e dinâmicas, que devem ser utilizadas para garantir a sobrevivência das organizações (D'Aveni *et al.*, 2010); (Sirmon *et al.*, 2010). Como tal, o cenário competitivo e o macro ambiente estão cada vez mais dinâmicos, gerando a necessidade de inovações que, por sua vez, impactam na estruturação e no desenvolvimento de projetos como um todo (Semuel *et al.*, 2017).

Para enfrentar essa situação de constante mudança, as organizações planejam suas estratégias de acordo com o objetivo que desejam alcançar. São estratégias que orientam a execução de ações em relação ao mercado e a seus concorrentes. O cumprimento dessas estratégias e, conseqüentemente, o sucesso da organização, dependem do resultado dos seus projetos e do alcance de seus objetivos individuais (Ralph, 2006).

De acordo com o *Project Management Institute* (PMI), um projeto é um conjunto de atividades temporárias, realizadas em grupo, destinadas a produzir um produto, serviço ou resultado únicos (PMI, 2017). E para que o desenvolvimento da gestão por projetos seja eficaz é imprescindível a aplicação de conhecimentos, habilidades, ferramentas e técnicas (D'Almeida *et al.*, 2010). Assim, as empresas estão buscando novos métodos e processos de gerenciamento de projetos que sejam mais eficientes, especialmente os voltados para o desenvolvimento de software (Vargas, 2016).

Com o crescimento no uso dos produtos de software, também houve um aumento pela exigência na qualidade desses itens, fazendo com que os desenvolvedores seguissem alguns padrões a fim de aumentar a produtividade (Rovai, 2013). Nesta busca por um padrão de qualidade nos produtos criados, as empresas fornecedoras de software passaram a se adequar e certificar seus processos dentro de alguns padrões tidos como referência. O intuito é que a qualidade de um software esteja diretamente vinculada à qualidade do processo usado para desenvolvê-lo.

Um desses padrões é modelo de maturidade *Capability Maturity Model Integration* (CMMI). Tal modelo para o desenvolvimento de software foi elaborado pelo *Software Engineering Institute* (SEI), da Universidade *Carnegie Mellon*, e é uma evolução do modelo *Capability Maturity Model* (CMM). Como objetivo, procura estabelecer um modelo para o processo de melhoria corporativo, integrando diferentes modelos e disciplinas. É considerado um padrão internacional e está cada vez mais sendo adotado nas empresas de software (Chrissis *et al.*, 2011). Este modelo de maturidade passou por várias atualizações desde a sua concepção: versão 1.0 em 1986, versão 1.2 em 2006, versão 1.3 em 2010 e, por fim, a versão 2.0 em 2018. De acordo com o *CMMI Institute*, de 2019 até julho de 2020, 93 empresas foram avaliadas pelo modelo CMMI ao redor do mundo. Destas, 23 estão nos Estados Unidos, 23 na Índia e 7 no Brasil (<https://sas.cmmiinstitute.com/pars/pars.aspx>, acessado em maio de 2020).

Um outro modelo de maturidade é o Melhoria do Processo de Software Brasileiro (MPS-BR), um programa da Associação para Promoção da Excelência do Software Brasileiro (SOFTEX) com apoio do Ministério da Ciência, Tecnologia, Inovações e Comunicações (MCTICS), e tem como objetivo melhorar a capacidade de desenvolvimento de software e serviços na indústria de Tecnologia da Informação e Comunicação (TIC) (Softex, 2012). É considerado, simultaneamente, um movimento para a melhoria e um modelo de qualidade de processo. É adequado ao perfil de empresas

com diferentes tamanhos e características, públicas e privadas, embora com especial atenção às micro e pequenas empresas (PME) de desenvolvimento de software no Brasil. É baseado no CMMI, nas normas ISO/IEC 12207 e ISO/IEC 33000 e na realidade do mercado brasileiro (Softex, 2011).

Ao mesmo tempo em que há uma busca maior pela qualidade dos produtos, há uma tendência de que a indústria de software busque prazos mais curtos para obter uma vantagem inicial e atender as necessidades atuais do cliente. No entanto, tais necessidades evoluem constantemente, levando a uma alta volatilidade dos requisitos, o que exige que as empresas de software sejam altamente flexíveis (Hajjdiab & Taleb, 2011). Portanto, mais e mais empresas de software começaram a adotar métodos ágeis ao invés dos métodos tradicionais de desenvolvimento de software (Petersen & Wohlin, 2010).

Em paralelo a esta abordagem, deve-se considerar que o desenvolvimento de software é uma atividade intelectual que depende essencialmente de pessoas, que geralmente formam equipes e trabalham juntas. Dessa forma, é impossível excluir os fatores humanos durante o desenvolvimento de software, porque o software é desenvolvido por pessoas e para elas (Miranda, 2011). Mas estudos recentes (Stark & Crocker, 2003); (Hidding & Nicholas, 2009); (Tripp et al., 2016); (Capretz et al., 2017); (Moser *et al.*, 2019) apontam que problemas identificados no desenvolvimento de software podem estar relacionados a aspectos humanos. Então, além da busca por padrões para o desenvolvimento de software, surgiram padrões para o desenvolvimento das pessoas nas empresas. Dessa forma, modelos de maturidade voltados para o desenvolvimento dos recursos humanos foram sendo adotados pelas organizações (Humphrey, 2005).

Dois desses modelos são o *People Capability Maturity model* (P-CMM) (Curtis *et al.*, 2009) e o Modelo de Referência MPS para Gestão de Pessoas (MPS-RH) (Softex, 2014), os quais são derivações dos modelos CMM e MPS-BR, mencionados anteriormente. São voltados para melhorar a capacidade das organizações de atrair, treinar, motivar, organizar e reter seus recursos humanos. O modelo P-CMM foi escolhido para esta pesquisa uma vez que emprega a estrutura de maturidade de processo do CMM como base para um modelo de melhores práticas para gerenciar e desenvolver a força de trabalho de uma organização. A versão 2 do P-CMM foi projetada para ser consistente com o CMMI (Chrissis *et al.*, 2007). Já o MPS-RH foi selecionado para este estudo por ser considerado o modelo brasileiro no que diz respeito ao desenvolvimento de pessoas dentro das organizações.

É interessante destacar que, para aumentar a produtividade, as metodologias ágeis têm sido um consenso entre as empresas de desenvolvimento (Boehm, 2006) e (Poppendieck & Poppendieck, 2011), enquanto os modelos de maturidade, com suas certificações, documentações e formalismos, são a forma mais óbvia de atestar a qualidade de um processo (SEI, 2010) e CMMI Institute (cmmiinstitute.com/cmmi/dev, acessado em maio de 2020). No entanto, as duas coisas podem ser vistas, erroneamente, como conflitantes, uma vez que os modelos de maturidade não são focados na implantação de uma gestão ágil de projetos. Em contrapartida, existe uma motivação para uso de métodos que promovam agilidade ao desenvolvimento (Coram & Bohner, 2005); (Packlick, 2007); (Dyba & Dingsoyr, 2009); (Asnawi et al., 2012); (Choudhary & Rakesh, 2016); (Aldahmash et al., 2017); (Dingsøyr et al., 2018). E, ao mesmo tempo, percebe-se um movimento crescente das empresas na busca por certificações e aderência aos níveis de maturidade (Staples & Niazi, 2008). Neste sentido, iniciativas que resultem em

maturidade de processos baseados em princípios ágeis têm sido alvo comum entre as empresas de software.

Dessa forma, a complexidade dos modelos tradicionais mais utilizados atualmente abre uma oportunidade para que os processos de implantação de tais padrões sejam adequados às necessidades específicas de cada organização. Nessa linha, existem trabalhos na literatura que comparam e sugerem a adoção do modelo CMMI e métodos ágeis, inclusive a sua integração (Potter & Sakry, 2009); (Lina & Dan, 2012); (Silva *et al.*, 2019); (Henriques & Tanner, 2017), numa tentativa de melhorar a qualidade do produto em ambientes ágeis de desenvolvimento de software. Mas no que tange aos modelos de maturidade para a gestão de pessoas em ambientes ágeis, percebe-se que seu uso ainda é escasso.

Dados oficiais apresentados pelo CMMI Institute mostram que, de 2017 até julho de 2020, apenas 27 empresas foram avaliadas dentro do modelo de maturidade P-CMM, em grande parte, situadas na Índia (<https://sas.cmmiinstitute.com/pars/pars.aspx>, acessado em maio de 2020). Apesar de a última versão do modelo P-CMM ter sido publicada em 2009, é a que está em uso atualmente. Estes dados foram apresentados a especialistas, os quais validaram as informações e confirmaram que o modelo está em pleno uso. Em relação ao uso do MPS-RH, apenas 2 empresas foram avaliadas até 2017 (<https://softex.br/mpsbr/avaliacoes/?page=2f571=6&filter=tipo-avaliacao=2f571=MPS-SW>, acessado em maio de 2020). Após este ano, não há dados oficiais disponíveis relacionados ao uso do modelo.

Dessa forma, é encontrada uma lacuna no cenário do uso de um modelo de maturidade voltado para o desenvolvimento dos aspectos humanos em equipes ágeis, o que abre espaço para estudos nessa área. Diante do exposto, o presente trabalho propõe, utilizando os arcabouços existentes e o pensamento de agilidade, uma estratégia para o uso dos modelos de maturidade para o desenvolvimento de fatores humanos em equipes ágeis de desenvolvimento de software, com base nos modelos de maturidade P-CMM e o MPS-RH.

1.1 Problema de Pesquisa

Os aspectos humanos se tornaram fator crítico para o sucesso dos projetos de software (Dingsøyr *et al.*, 2012) e são considerados de grande relevância para a construção de um produto (Moe & Dingsøyr, 2008). Com base nesse contexto, surgiram os modelos de maturidade MPS-RH e P-CMM, os quais têm como objetivos atrair, treinar, organizar, motivar e reter talentos dentro da organização. Apesar de esses modelos serem muito bem estruturados, o formalismo, por algumas vezes excessivo, tem tornado a adoção e melhoria contínua de seus processos uma tarefa complexa.

Em 2007, Giehl analisou a aplicação do P-CMM em empresas brasileiras de desenvolvimento de software, sob a ótica de líderes de grupos de processos de engenharia de software. Os resultados do trabalho mostram que existe complexidade para a implementação do modelo. Vale ressaltar que este estudo foi realizado antes da versão 2 do modelo ser apresentada, em 2009. Porém, estudos realizados após esta data, corroboram com o teor da complexidade do modelo.

Em 2011, Miranda recomendou a revisão dos indicadores propostos no P-CMM tendo em vista o número elevado de variáveis definidas no modelo original. A proposta é reduzir o modelo para uma versão compacta e precisa, buscando chegar em um sistema de mensuração mais simples e consistente.

O trabalho realizado por Zhang (2015) estudou a questão do gerenciamento de desempenho em pequenas e médias empresas privadas na China, e como o P-CMM poderia ser utilizado para trazer melhorias para a área. De acordo com o autor, devido ao gerenciamento de recursos humanos ainda estar em sua fase inicial nessas empresas, o uso do modelo de maturidade P-CMM tem suas limitações, uma vez que aborda a questão do gerenciamento de desempenho apenas nos seus níveis mais altos. Em suma, de acordo com o autor, o P-CMM não pode ser completamente utilizado para pequenas e médias empresas, necessitando de uma adequação.

Em relação ao MPS-RH, apesar de ser um modelo de maturidade proposto pela SOFTEX em 2014, ainda é pouco utilizado pelas empresas. Publicações sobre o tema ainda são incipientes, o que o torna um modelo de maturidade pouco conhecido (https://softex.br/mpsbr/avaliacoes/?page_2f571=6&filter_tipo-avaliacao_2f571=MPS-SW, acessado em maio de 2020).

Na área dos aspectos humanos também fica evidente a necessidade de padronização e validação dos métodos empregados, existindo uma carência de estudos mais aprofundados, conforme Conte *et al.*, (2019).

Diante do contexto citado, o problema de pesquisa que tem motivado o desenvolvimento deste trabalho está, principalmente, relacionado aos desafios da área de maturidade como apoio à gestão das pessoas nas equipes de desenvolvimento de software. Esta pesquisa pretende explorar a lacuna supracitada como uma oportunidade para expandir e explorar tal paradigma respondendo o questionamento: Como mensurar e priorizar o desenvolvimento dos aspectos humanos em equipes ágeis de desenvolvimento de software com base nos atuais arcabouços existentes na área?

1.2 Motivação e Justificativa

Capretz *et al.* (2017) apontam que as falhas em projetos de software podem ser atribuídas a problemas humanos. No entendimento de Highsmith e Cockburn (2002), o fator humano no desenvolvimento de software influencia a formação de times, focando-se na interação das pessoas e nos indivíduos. No caso específico dos métodos ágeis, a literatura mostra que tais abordagens representam um apoio estratégico para que as organizações consigam alcançar maior desempenho no desenvolvimento de seus sistemas de software com resultados positivos e sucesso.

Estudiosos adeptos de metodologias ágeis apontam para os fatores humanos como geradores de interferência na aplicabilidade de tais métodos. Esses pesquisadores entendem que problemas nos recursos humanos são capazes de interferir negativamente no cumprimento dos objetivos da empresa, apresentando dificuldades em atender às normas e aos procedimentos que visam corrigir falhas, atingir o mais alto padrão de qualidade e conduzir ao sucesso organizacional (Moe & Dingsøyr, 2008; Pirzadeh, 2010).

A popularidade dos métodos ágeis está crescendo e, junto com esse crescimento, a adoção de modelos de maturidade também ganha relevância. Estudos sobre o uso do gerenciamento ágil de projetos em conjunto com os modelos de maturidade apontam que a área ainda precisa se desenvolver, pois existe uma carência relativa às atividades de desenvolvimento de software e as técnicas que podem ser utilizadas para atender as questões que não são diretamente abordadas pelas técnicas ágeis (Chagas *et al.*, 2015).

As equipes de desenvolvimento de software ágil são sistemas sociotécnicos adaptativos complexos (Whitworth & Biddle, 2007), contando com membros multifacetados da equipe, equipados com uma ampla gama de habilidades técnicas,

sociais e de liderança (Nerur *et al.*, 2010). O lado técnico dos métodos ágeis ajuda as equipes a se tornarem mais produtivas, aumentando a qualidade do produto final. No entanto, o lado humano das equipes ainda é um desafio. Formar essas equipes exige tempo e recursos (Moe *et al.*, 2010) e requer mais habilidades sociais (Tan & Teo, 2007) e experiência (Boehm & Turner, 2005). Os estudos de Melo (2013) apontam que existe um caminho para pesquisas adicionais sobre as habilidades e o treinamento necessários para equipes ágeis.

Em termos de contribuição teórica, este trabalho trata novos conhecimentos em diversas direções. Em termos gerais, para o próprio campo do uso de modelos de maturidade em equipes ágeis, principalmente em relação ao ambiente acadêmico brasileiro, no qual os estudos sobre o tema ainda estão em fase de evolução e o volume da produção literária ainda é pequeno.

As organizações de software são as principais partes interessadas desta pesquisa. Estas estão constantemente buscando equipes que estejam mais satisfeitas, motivadas e engajadas no processo de desenvolvimento de software.

Espera-se que os resultados deste trabalho possam contribuir para a compreensão de novas práticas de desenvolvimento humano nas organizações de software, como forma de incrementar os pontos fortes dos colaboradores a partir de uma perspectiva mais positiva nas relações de trabalho.

1.3 Objetivos Geral e Específicos

O Objetivo Geral deste trabalho é propor uma estratégia para o uso de modelos de maturidade no desenvolvimento dos fatores humanos em equipes ágeis de desenvolvimento de software, através da utilização, em conjunto, dos principais arcabouços existentes na área.

Como Objetivos Específicos, tem-se:

- a. Analisar problemas críticos associados a gestão de pessoas em uma equipe de desenvolvimento ágil;
- b. Analisar e recomendar as práticas para gerenciar e desenvolver a força de trabalho em uma equipe de desenvolvimento ágil;
- c. Consolidar as iniciativas para aferir a qualidade e maturidade das pessoas no desenvolvimento de software;
- d. Aferir e ajudar as equipes de desenvolvimento ágil a caracterizar a maturidade de suas práticas de força de trabalho;
- e. Estabelecer um programa de desenvolvimento contínuo da força de trabalho;
- f. Integrar o desenvolvimento da força de trabalho à melhoria de processos.

2. Trabalhos Correlatos

Dentre alguns autores a se mencionar, pode-se destacar a publicação de Fontana *et al.*, (2014), que propuseram uma nova definição para a maturidade do desenvolvimento de software ágil. As descobertas mostram que os profissionais não veem maturidade no desenvolvimento ágil de software como definição de processo ou recursos de gerenciamento quantitativo. Em vez disso, maturidade ágil significa promover capacidades mais subjetivas, como colaboração, comunicação, compromisso, cuidado, compartilhamento e auto-organização.

Também pode-se destacar a publicação de Henriques & Tanner, (2017), uma revisão sistemática da literatura sobre o uso de modelos de maturidade em ambientes que utilizam

métodos ágeis para o desenvolvimento de software. A análise de Henriques & Tanner (2017) resultou em dois grandes temas emergentes, como Agile/CMMI e Agile Maturity. O primeiro (Agile/CMMI) preocupa-se, principalmente, com a coexistência de métodos ágeis em um ambiente em que o CMMI está presente; enquanto o último (Agile Maturity), preocupa-se, principalmente, com a melhoria da implementação ágil sem se preocupar com outras estruturas de melhoria de processo (com foco na definição de caminhos de melhoria baseados em ágil), levando a uma maior agilidade.

Outro ponto abordado no estudo de Henriques & Tanner (2017), foi o uso simultâneo dos métodos ágeis e do CMMI. Eles concluem que o ágil e o CMMI podem coexistir com sucesso quando o ágil é introduzido em ambientes já altamente maduros ou quando o objetivo principal é focado apenas na entrega. As conclusões indicam que, se o objetivo for atingir níveis mais altos de maturidade do CMMI, o ágil não pode ser usado sem ser suplementado com outras práticas não-ágeis.

O estudo de Alcântara *et al.*, (2018) propôs um modelo para servir como guia de gerenciamento de projetos em projetos ágeis, independentemente das características da organização e da equipe de desenvolvimento. A gestão de pessoas foi modelada de forma interativa, tendo em vista a presença desta característica em projetos de desenvolvimento ágil, para que a gestão ocorra concomitante ao processo de desenvolvimento. Vale ressaltar que o modelo é genérico e que foi desenvolvido com base nos conceitos da literatura em relação à gestão de pessoas e com as boas práticas observadas nos estudos analisados. Além disso, este modelo não foi validado em cenários reais na indústria de software.

Como pode ser observado até o momento, os trabalhos correlatos apresentados mencionam o uso dos modelos CMMI em ambiente ágil de forma simultânea. Também apresentam aplicações do modelo P-CMM em ambientes ágeis. Mas nenhum estudo menciona uma proposta para o uso dos modelos de maturidade, como o P-CMM e o MPS-RH em equipes ágeis de desenvolvimento de software, com diretrizes mais assertivas (e não genéricas) para o desenvolvimento dos fatores humanos.

3. Metodologia

De acordo com Gil (2010), a pesquisa é desenvolvida mediante o concurso dos conhecimentos disponíveis e a utilização cuidadosa de métodos e técnicas de investigação científica. As pesquisas podem ter diversas classificações e estão divididas pela forma como o problema é abordado, quanto ao objetivo para o qual a pesquisa é realizada e em relação aos procedimentos utilizados para a coleta e análise dos dados.

Diante do exposto, a pesquisa em questão, que visa a proposta de uma estratégia para o uso de modelos de maturidade no desenvolvimento dos fatores humanos em equipes ágeis de desenvolvimento de software, tem sua abordagem metodológica inserida na Tabela 1.

Tabela 1: Classificação Metodológica

Característica	Classificação da Pesquisa
Objetivo	Exploratória e Descritiva
Procedimentos Técnicos	Pesquisa bibliográfica, Pesquisa de campo/Levantamento (survey + entrevistas semiestruturadas), Design Science Research (DSR).
Abordagem	Quanti-qualitativa

A execução da pesquisa está sendo realizada em atividades divididas em duas fases distintas. A primeira fase, um estudo preliminar, consistiu de uma fase exploratória e teve por objetivo a construção de uma base teórica consistente para suportar a etapa seguinte. Nela, foi realizada uma revisão da literatura (ad hoc) exploratória (etapa 1), um survey (etapa 2) e entrevistas semiestruturadas (etapa 3 – em andamento), a fim de realizar um levantamento de informações relevantes para a estruturação do problema da pesquisa.

A fase dois, ainda não iniciada, tem uma característica mais descritiva e visa a real construção da solução proposta, utilizando metodologia de *Design Science Research*. Tal método operacionaliza a pesquisa realizada sob o paradigma da *Design Science*, fundamentando a condução da pesquisa quando o objetivo a ser alcançado é o desenvolvimento de um artefato (Çağdaş, V., & Stubkjær, 2011).

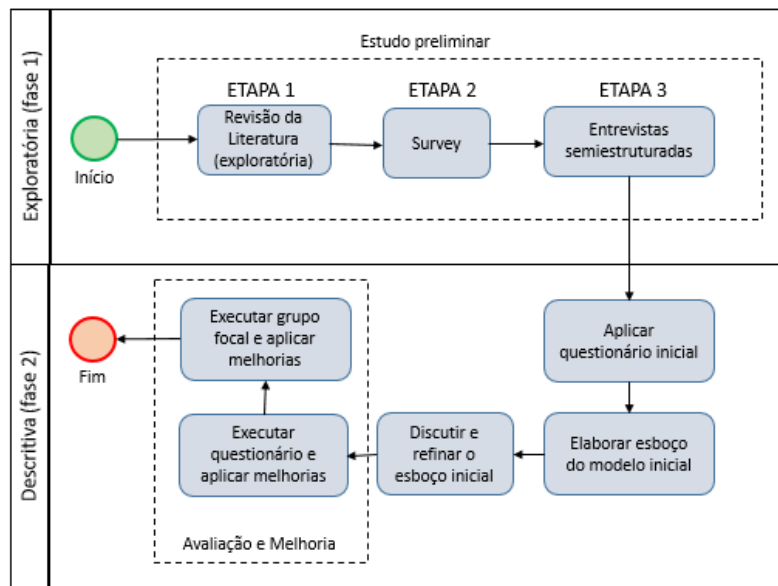


Figura 1 – Macro etapas metodológicas da pesquisa (Elaborado pela autora)

A etapa “Fim” da Figura 1 representa a apresentação completa, em uma primeira versão, do objeto deste trabalho. Além da visão metodológica apresentada na Figura 1, e para um melhor entendimento da construção do objeto proposto, segue a Figura 2, com um viés mais prático da base e etapas que devem ser seguidas.

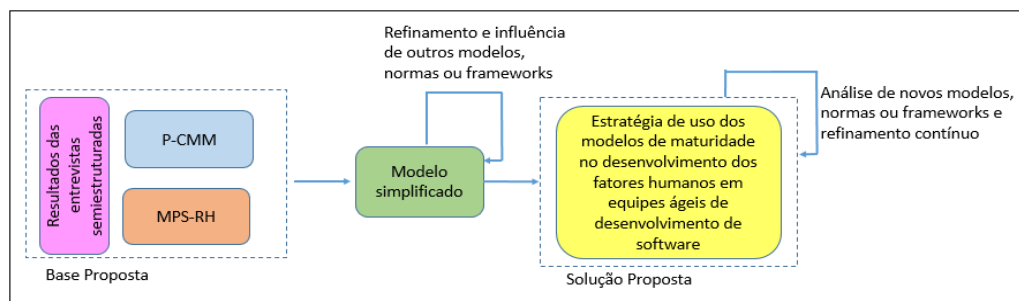


Figura 2 – Macro etapas práticas da pesquisa

Serão analisados os modelos de maturidade P-CMM e MPS-RH bem como os resultados das entrevistas semiestruturadas realizadas na fase 1 (estudo preliminar). A partir dessa base proposta será elaborado um modelo simplificado para a estratégia

bases ACM, IEEE e Compendex. Tanto para a busca manual quanto para a automática foi considerado o intervalo de publicações entre 2001 e 2019, partindo do princípio que o *People Capability Maturity Model* (P-CMM) teve seu marco inicial em 2001 (Curtis *et al.*, 2001).

Após a aplicação dos critérios de inclusão e de exclusão, a busca automática retornou 37 artigos. Após a retirada dos duplicados, a soma com os artigos da busca manual e exclusão pela leitura, permaneceram 41 artigos. Para o gerenciamento dos arquivos foi utilizado o software gratuito Mendeley (www.mendeley.com).

3.1.2 Survey (Etapa 2 da Fase Preliminar)

O survey buscou o entendimento de questões relacionadas às metodologias de desenvolvimento e modelos de maturidade em equipes ágeis de desenvolvimento de software.

Como instrumento de pesquisa deste estudo preliminar, foi utilizado um questionário estruturado, composto de 23 questões, onde se buscou o perfil do respondente, perfil da empresa onde o respondente atua bem como a experiência/conhecimento do respondente em desenvolvimento de software, métodos ágeis e modelos de maturidade. Foi direcionado para profissionais da área de tecnologia da informação, mais especificamente, envolvidos em projetos de desenvolvimento de software.

O início do questionário trouxe orientações iniciais com o objetivo de esclarecer ao respondente sobre o propósito da pesquisa bem como para qual público o questionário estava direcionado. Das 23 questões do questionário, 22 eram fechadas, de múltipla escolha, com campos em aberto para as justificativas das respostas. O questionário trouxe uma única questão aberta e foi adaptado de Silva *et al.*, (2017).

Para a questão aberta do survey foi feita uma análise qualitativa. O objetivo da análise qualitativa é consolidar, reduzir e interpretar dados obtidos de várias fontes, e dar sentido a eles (Merriam & Tisdell, 2015). Envolve rotular e codificar todos os dados, a fim de identificar semelhanças e diferenças para descrever o fenômeno em estudo.

Dessa forma, a análise da questão aberta começou com a codificação aberta das respostas. Com a leitura da primeira resposta, os códigos iniciais foram definidos. A cada leitura de novas respostas, os códigos decorrentes de cada resposta foram constantemente comparados aos códigos das outras respostas.

A partir das constantes comparações dos códigos, as categorias foram formadas. Dessa forma, os 40 códigos iniciais foram sintetizados em 6 categorias: Viabilidade da Solução; Responsabilidade da Equipe; Responsabilidade da Gestão; Organização e Planejamento; Harmonia entre Metodologia e Modelo e Ambiente/Empresa. À medida que o processo de análise de dados progrediu, as relações entre categorias foram construídas, o que possibilitou a construção da narrativa.

Por fim, as categorias principais foram escolhidas de acordo com sua relevância dentro do contexto estudado. A partir da definição das categorias centrais, foram definidas as proposições, o que possibilitou a interpretação das respostas, apresentadas na seção 4.

3.1.3 Entrevistas (Etapa 3 da Fase Preliminar)

A entrevista, terceira etapa de coleta de dados da fase preliminar, ainda em andamento, busca uma maior investigação em relação aos problemas enfrentados pelas empresas no âmbito da gestão das pessoas nas equipes ágeis de desenvolvimento de software.

O roteiro de entrevistas seguiu as recomendações propostas por Merriam & Tisdell (2015), onde as perguntas estão no centro das entrevistas e para coletar dados significativos, o pesquisador deve fazer boas perguntas. As questões mais abertas são as melhores de serem utilizadas. Além disso, ter menos perguntas, porém mais abertas, faz com que o pesquisador possa ser flexível no uso do roteiro, permitindo que o pesquisador ouça o que seu entrevistado tem para compartilhar.

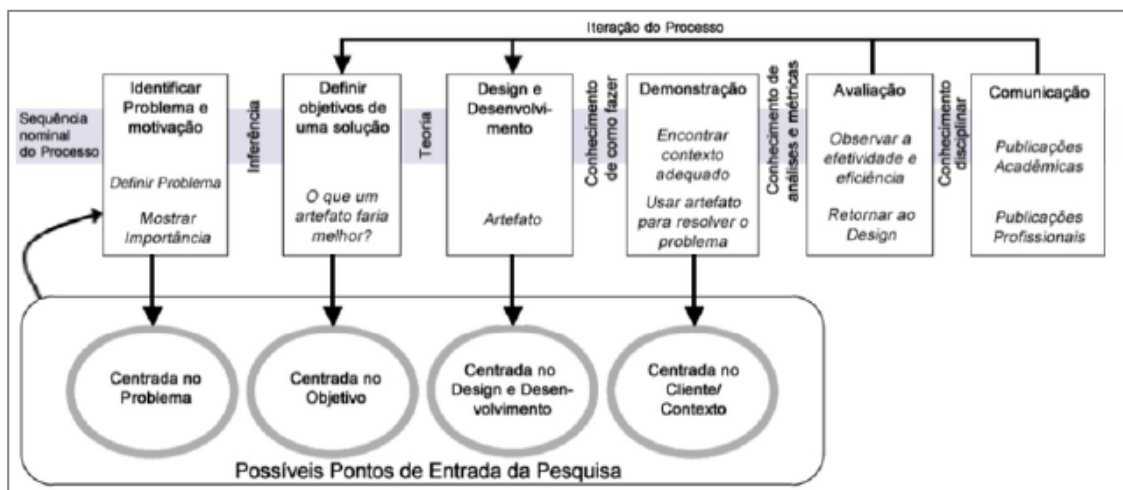
3.1.4 Design Science Research (DSR)

A DSR faz parte da fase 2 desta pesquisa, ainda não iniciada. Possui uma característica mais descritiva e visa a real construção da solução proposta.

A DSR tem como objetivo estudar, pesquisar e investigar o artificial e seu comportamento, tanto do ponto de vista acadêmico quanto da organização, constituindo-se como um processo rigoroso de projetar e avaliar artefatos para resolver problemas reais e diminuir a lacuna entre a teoria e a prática (Dresch *et al.*, 2015). As pesquisas que utilizam a DSR devem prezar por dois fatores fundamentais para o sucesso: o rigor científico e a relevância para as organizações. Ou seja, a pesquisa deve ser aplicada e trazer melhorias ao ambiente. Para isso deve utilizar técnicas e fundamentos consolidados. Então, a DSR é orientada à solução de problemas específicos, não necessariamente buscando a solução ótima, mas sim, a solução satisfatória para a situação, sendo passível de generalização dentro de uma mesma classe de problemas (Dresch *et al.*, 2015).

Segundo Peffers *et al.*, (2007), a DSR, como um processo metodológico, deve possuir seis etapas, conforme Figura 4. Embora as atividades sejam inseridas de forma sequencial na representação de Peffers *et al.*, (2007), não é imposta pelos autores uma ordem exata para o início da pesquisa. Sendo este um de seus grandes diferenciais, possibilitando que o início da pesquisa possa ocorrer em quatro pontos distintos, considerando o foco que se pretenda dar à investigação (Peffers *et al.*, 2007).

Figura 4: Modelo do Processo da *Design Science Research* (Peffers *et al.*, 2007)



A presente pesquisa é centrada no problema, visto que a ideia central deste trabalho emergiu de problemas já existentes. Dessa forma, o processo metodológico seguirá a sequência proposta por Peffers *et al.*, (2007).

3.1.5 Grupo Focal

O Grupo Focal faz parte da fase 2 desta pesquisa, ainda não iniciada.

Os grupos focais são grupos de discussão que debatem sobre uma temática previamente definida, ao receberem gatilhos que estimulem as discussões. Essa técnica distingue-se por suas características próprias, principalmente pelo processo de interação grupal, que é uma resultante da procura de dados. Facilita a formação de ideias novas e originais, gerando possibilidades contextualizadas pelo próprio grupo de estudo. Oportuniza a interpretação de crenças, valores, conceitos, conflitos, confrontos e pontos de vista. E ainda possibilita entender o estreitamento em relação ao tema, no cotidiano (Ressel *et al.*, 2008).

Morgan (1996) afirma que os grupos focais podem ser utilizados para levantamento de dados iniciais, como para avaliar o assunto proposto pelo pesquisador podendo, em ambos os casos, ser associado a outras técnicas, por exemplo, entrevistas individuais, surveys ou experimentos. Combinação de técnicas que também é citada por Carlini-Cotrim (1996) e Dresch *et al.*, (2015) para avaliação de artefatos, inclusive na DSR.

Para execução do grupo focal será utilizada a estrutura de etapas proposta por Kontio *et al.*, (2004). São elas: a. Definir o problema de pesquisa; b. Planejar o evento do grupo de focal; c. Selecionar os participantes; d. Conduzir a sessão do grupo focal e e. Analisar os dados e gerar relatório.

4. Resultados Preliminares

O objetivo geral deste trabalho é a elaboração de uma estratégia para o uso de modelos de maturidade no desenvolvimento dos fatores humanos em equipes ágeis de desenvolvimento de software. Para tal, foi necessário um estudo preliminar, realizado em duas etapas (survey e entrevistas semiestruturadas), a fim de suportar e justificar a necessidade da construção do artefato proposto. Os resultados deste estudo preliminar estão apresentados nesta seção.

4.1 Survey

O survey foi realizado com dois objetivos: 1. fazer um levantamento de informações que são relevantes para a estruturação do problema da pesquisa; 2. realizar uma sondagem para verificar o nível de conhecimento dos respondentes em relação a métodos ágeis e modelos de maturidade, dando sustentação ao objetivo geral deste estudo. Por uma questão de limitação de espaço deste documento, os resultados serão sintetizados e abordados a seguir.

O perfil da amostra é de profissionais do sexo masculino, entre 26 e 35 anos e com maior qualificação. O perfil das organizações em que os respondentes atuam são de grande porte (45,1%), de iniciativa privada (64,7%), nacionais (78,4%), localizadas no Nordeste (NE) (96,1%), atuando na área de Consultoria/Serviços (23,5)% e Educação (11,8%) e com uma equipe de tecnologia com menos de 20 pessoas.

Considerando a experiência/conhecimento do respondente em desenvolvimento de software, métodos ágeis e modelos de maturidade, chegou-se a alguns resultados:

1. Mais da metade dos respondentes (56,9%) têm mais de 5 (cinco) anos de experiência em desenvolvimento de software;
2. Quase a metade dos respondentes (49,1%) atuam na área de desenvolvimento, seja desenvolvendo ou gerenciando projetos de software;
3. Mais da metade dos respondentes (51%) têm mais de 3 anos experiência em métodos ágeis;

4. 35,3\% dos respondentes têm conhecimento avançado em SCRUM;
5. Os três principais frameworks utilizados na organização são Scrum (78,4\%), Kanban (49\%), e as metodologias tradicionais baseadas no PMBok (31,4\%). CMMI é pouco utilizado, contabilizando 5,9\%. Pode-se destacar o uso dos frameworks ágeis nas organizações pesquisadas;
6. É importante destacar que uma parcela relativamente alta dos respondentes (35,3\%) não sabem o que é um modelo de maturidade para o desenvolvimento de software, a considerar o perfil dos respondentes;
7. 49\% dos respondentes afirmam que a organização em que trabalha não utiliza nenhum modelo de maturidade. Apenas 21,6 responderam de forma positiva;
8. Considerando os modelos de maturidade para a gestão de pessoas, a grande maioria dos respondentes (76,5\%) não sabem o que é. Considerando que parte da amostra tem cargo gerencial (41,2\%), esta fato chama a atenção;
9. Mais da metade dos respondentes afirmam (51\%) que as organizações onde trabalham não usam um modelo de maturidade para o desenvolvimento de pessoas;
10. A grande maioria dos respondentes (88,2\%) desconhece que haja algum modelo de maturidade voltado para o desenvolvimento ágil de software;
11. 94,1\% dos respondentes não conhecem um modelo de maturidade de desenvolvimento de software que envolva a gestão de pessoas;
12. Apenas 21,6\% dos respondentes afirma que a organização na qual trabalham fazem uso simultâneo de modelos de maturidade e metodologias ágeis. Na questão aberta fica claro, inclusive, que existe um senso comum de incompatibilidade entre os frameworks;
13. Além disso, considerando as respostas abertas (nas questões onde era possível a justificativa das respostas), pode-se perceber que os respondentes fazem confusão entre modelos de maturidade e metodologias de desenvolvimento de software. Este fato também chama a atenção uma vez que a amostra, em sua maioria, tem profissionais mais bem capacitados tecnicamente.

Na opinião dos respondentes, para que os modelos de maturidade e as metodologias ágeis possam ser utilizadas em conjunto, alguns fatores devem ser considerados. Primeiramente, a organização e o planejamento são fundamentais para que o processo funcione. É preciso que tudo esteja muito bem mapeado para que não haja falhas na implementação. Além disso, a equipe deve estar bem preparada, com responsabilidades definidas. O papel da gestão também é fundamental, devendo conduzir o processo de forma transparente e com certo rigor. Por fim, a organização deve ter estrutura para suportar a integração dos frameworks, mantendo um ambiente propício para o uso de modelos de maturidade em ambiente ágil.

4.2 Entrevistas

As entrevistas buscam uma maior investigação em relação aos problemas enfrentados pelas empresas no âmbito da gestão das pessoas nas equipes ágeis de desenvolvimento de software.

Os dados referentes às entrevistas ainda não estão disponíveis, uma vez que ainda estão sendo realizadas.

5. Considerações Finais

Com os resultados do estudo até o presente momento, dentro do universo da amostra pesquisada, chegou-se a um perfil do uso (ou não uso) dos modelos de maturidade dentro de equipes de desenvolvimento ágil de software.

Constata-se a utilização dos métodos ágeis de desenvolvimento de software pela maioria das organizações. Em relação aos modelos de maturidade, apesar de conhecidos (como é o caso do CMMI e do MPS-BR), não são muito utilizados.

Apesar de reduzido, percebe-se o uso simultâneo dos modelos de maturidade e metodologias ágeis no desenvolvimento de software por algumas empresas. Para que possam ser utilizadas em conjunto, fatores como organização e planejamento, responsabilidades da gestão, responsabilidades da equipe, um bom ambiente e uma boa estrutura da empresa devem ser levados em consideração.

Os modelos de maturidade voltados para a gestão das pessoas nas empresas de software ainda são pouco conhecidos, como é o caso do P-CMM e do MPS-RH. Talvez esse seja o motivo para que tais padrões não sejam utilizados. Mas os dados colhidos pelas poucas entrevistas realizadas até momento (ainda não inseridas neste trabalho em função de estarem em andamento), apontam para dois fatores: a complexidade das normas e a falta de apoio do alto escalão.

Por fim, acredita-se que a implementação de uma estratégia para o uso de modelos de maturidade no desenvolvimento dos fatores humanos em equipes ágeis de desenvolvimento de software, apesar de não ser por si só suficiente para atingir um nível aceitável de gestão de pessoas, é um grande passo para uma melhoria significativa na área.

Referências Bibliográficas

- Aldahmash, A., Gravell, A. M., & Howard, Y. (2017, September). A review on the critical success factors of agile software development. In European conference on software process improvement (pp. 504-512). Springer, Cham.
- Aysolmaz, B., & Demirörs, O. (2011, June). A detailed software process improvement methodology: BG-SPI. In European Conference on Software Process Improvement (pp. 97-108). Springer, Berlin, Heidelberg.
- Asnawi, A. L., Gravell, A. M., & Wills, G. B. (2012, February). Factor analysis: Investigating important aspects for agile adoption in Malaysia. In *2012 Agile India* (pp. 60-63). IEEE.
- Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M. & Kern, J. (2001). Manifesto for agile software development.
- Boehm, B. (2002). Get ready for agile methods, with care. *Computer*, 35(1), 64-69.
- Boehm, B., & Turner, R. (2003). *Balancing agility and discipline: A guide for the perplexed*. Addison-Wesley Professional.
- Boehm, B., & Turner, R. (2005). Management challenges to implementing agile processes in traditional development organizations. *IEEE software*, 22(5), 30-39
- Boehm, B. (2006, May). A view of 20th and 21st century software engineering. In *Proceedings of the 28th international conference on Software engineering* (pp. 12-29).

- Capretz, L. F., Ahmed, F., & da Silva, F. Q. B. (2017). Soft sides of software. arXiv preprint arXiv:1711.07876.
- Carlini-Cotrim, B. (1996). Potencialidades da técnica qualitativa grupo focal em investigações sobre abuso de substâncias. *Revista de Saúde Pública*, 30, 285-293.
- Çağdaş, V., & Stubkjær, E. (2011). Design research for cadastral systems. *Computers, Environment and Urban Systems*, 35(1), 77-87
- Chagas, A., Santos, M., Santana, C., & Vasconcelos, A. (2015, August). The impact of human factors on agile projects. In *2015 Agile Conference* (pp. 87-91). IEEE.
- Choudhary, B., & Rakesh, S. K. (2016, February). An approach using agile method for software development. In *2016 International Conference on Innovation and Challenges in Cyber Security (ICICCS-INBUSH)* (pp. 155-158). IEEE.
- Chrissis, M. B., Konrad, M., & Shrum, S. (2007). *CMMI (Capability Maturity Model Integrated) for Development 2nd Edition-Guidelines for Process Integration and Product Improvement*. SEI Series in Software Engineering.
- Chrissis, M. B., Konrad, M., & Shrum, S. (2011). *CMMI for development: guidelines for process integration and product improvement*. Pearson Education.
- CMMI Institute. <https://sas.cmmiinstitute.com/pars/pars.aspx>. Acesso em maio de 2020.
- CMMI Institute. <https://cmmiinstitute.com/cmmi/dev>. Acesso em maio de 2020.
- Conte, T., Prikladnicki, R., & Dutra, A. (2019). A Field Research on the Practices of High Performance Software Engineering Teams. In *Proceedings of the 21st International Conference on Enterprise Information Systems, 2019*, Grécia.
- Coram, M., & Bohner, S. (2005, April). The impact of agile methods on software project management. In *12th IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS'05)* (pp. 363-370). IEEE
- Curtis, B., Hefley, W. E., & Miller, S. A. (2001). *People capability maturity model (P-CMM)*. Pittsburgh, PA.
- Curtis, B., Hefley, B., & Miller, S. (2009). *People capability maturity model (P-CMM) version 2.0* (No. CMU/SEI-2009-TR-003). CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST.
- da Silva, P. R., dos Santos, M. R., & Shiba, F. Y. (2019). DESENVOLVIMENTO DE SOFTWARES: CMMI E METODOLOGIAS ÁGEIS. *Revista Livre de Sustentabilidade e Empreendedorismo*, 4(3), 157-184.
- de Alcântara, P. T. R., Canedo, E. D., & da Costa, R. P. (2018, June). People Management in Software Agile Development. In *Proceedings of the XIV Brazilian Symposium on Information Systems* (pp. 1-10)
- D'Aveni, R. A., Dagnino, G. B., & Smith, K. G. (2010). The age of temporary advantage. *Strategic management journal*, 31(13), 1371-1385.
- D'Almeida, S., Silva, V. F., & André, V. M. (2010). Pearson Education do Brasil.
- Deemer, P., & Benefield, G. *The Scrum Primer: An Introduction to Agile Project Management with Scrum*. goodagile, 2007.

- Dikert, K., Paasivaara, M., & Lassenius, C. (2016). Challenges and success factors for large-scale agile transformations: A systematic literature review. *Journal of Systems and Software*, 119, 87-108.
- Dirks, K. T. (1999). The effects of interpersonal trust on work group performance. *Journal of applied psychology*, 84(3), 445.
- de Miranda, D. L. A. (2011). Maturidade e desempenho em gestão de pessoas: uma análise em fundações de apoio à pesquisa a partir do People Capability Maturity Model (P-CMM).
- DeOrtentiis, P. S., Summers, J. K., Ammeter, A. P., Douglas, C., & Ferris, G. R. (2013). Cohesion and satisfaction as mediators of the team trust–team effectiveness relationship. *Career Development International*.
- Dingsøy, T., Nerur, S., Balijepally, V., & Moe, N. B. (2012). A decade of agile methodologies: Towards explaining agile software development
- Dingsøy, T., Moe, N. B., Fægri, T. E., & Seim, E. A. (2018). Exploring software development at the very large-scale: a revelatory case study and research agenda for agile method adaptation. *Empirical Software Engineering*, 23(1), 490-520.
- Dresch, A., Lacerda, D. P., & Júnior, J. A. V. A. (2015). Design science research: método de pesquisa para avanço da ciência e tecnologia. Bookman Editora.
- Dybå, T., & Dingsøy, T. (2008). Empirical studies of agile software development: A systematic review. *Information and software technology*, 50(9-10), 833-859.
- Dybå, T., & Dingsøy, T. (2009). What do we know about agile software development?. *IEEE software*, 26(5), 6-9.
- Faraj, S., & Sproull, L. (2000). Coordinating expertise in software development teams. *Management science*, 46(12), 1554-1568.
- Fontana, R. M., Fontana, I. M., da Rosa Garbuio, P. A., Reinehr, S., & Malucelli, A. (2014). Processes versus people: How should agile software development maturity be defined?. *Journal of Systems and Software*, 97, 140-155.
- Fontana, R. M., Meyer Jr, V., Reinehr, S., & Malucelli, A. (2015). Progressive Outcomes: A framework for maturing in agile software development. *Journal of Systems and Software*, 102, 88-108.
- Galín, D., & Avrahami, M. (2006). Are CMM program investments beneficial? Analyzing past studies. *IEEE software*, 23(6), 81-87.
- Gil, A. C. (2010). Como elaborar projetos de pesquisa. São paulo: Atlas, 2006. Gil, Antônio Carlos. Como elaborar projetos de pesquisa, 5.
- Giehl, R. B. T. P-CMM: Qualidade no Processo de Software e Gestão de Pessoas.
- Hidding, G. J., & Nicholas, J. (2009, January). Reducing IT project management failures: A research proposal. In 2009 42nd Hawaii International Conference on System Sciences (pp. 1-10). IEEE.
- Humphrey, W. S. (2005). Psp (sm): a self-improvement process for software engineers. Addison-Wesley Professional.
- Hajjdiab, H., & Taleb, A. S. (2011). Adopting agile software development: issues and challenges. *International Journal of Managing Value and Supply Chains (IJMVSC)*, 2(3), 1-10.

- Henriques, V., & Tanner, M. (2017). A systematic literature review of agile and maturity model research. *Interdisciplinary Journal of Information, Knowledge, and Management*, 12, 053-073.
- Jakobsen, C. R., & Sutherland, J. (2009, August). Scrum and CMMI going from good to great. In *2009 Agile Conference* (pp. 333-337). IEEE.
- Kitchenham, B., & Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering.
- Kontio, J., Lehtola, L., & Bragge, J. (2004, August). Using the focus group method in software engineering: obtaining practitioner and user experiences. In *Proceedings. 2004 International Symposium on Empirical Software Engineering, 2004. ISESE'04.* (pp. 271-280). IEEE.
- Latussek, D., & Jemielniak, D. (2007). trust in software projects: A thrice told tale. On dynamic relationships between software engineers, IT project managers, and customers. *The International Journal of Technology, Knowledge and Society*, 3(1), 117-125.
- Leppänen, M. (2013). A comparative analysis of agile maturity models. In *Information Systems Development* (pp. 329-343). Springer, New York, NY.
- Lina, Z., & Dan, S. (2012, March). Research on combining scrum with CMMI in small and medium organizations. In *2012 International Conference on Computer Science and Electronics Engineering* (Vol. 1, pp. 554-557). IEEE.
- Lu, X., Xu, D., & Han, J. (2010, August). Research on the Staff Maturity of Software Companies in China-----Take Hangzhou City as a Sample. In *2010 International Conference on Management and Service Science* (pp. 1-4). IEEE.
- Melo, C. D. O., Santos, V., Katayama, E., Corbucci, H., Prikladnicki, R., Goldman, A., & Kon, F. (2013). The evolution of agile software development in Brazil. *Journal of the Brazilian Computer Society*, 19(4), 523-552.
- Merriam, S. B., & Tisdell, E. J. (2015). *Qualitative research: A guide to design and implementation*. John Wiley & Sons.
- Miranda, R. (2011). Uma Revisão Sistemática Sobre Equipes de Desenvolvimento de Software: Tipologia, Características e Critérios de Formação. Centro de Informática (CIn), Universidade Federal de Pernambuco.
- Moe, N. B., & Dingsøyr, T. (2008, June). Scrum and team effectiveness: Theory and practice. In *International conference on agile processes and extreme programming in software engineering* (pp. 11-20). Springer, Berlin, Heidelberg.
- Moe, N. B., Dingsøyr, T., & Dybå, T. (2010). A teamwork model for understanding an agile team: A case study of a Scrum project. *Information and Software Technology*, 52(5), 480-491.
- Morgan, D. L. (1996). *Focus groups as qualitative research* (Vol. 16). Sage publications.
- Moser, P., Araújo, J., Oliveira, E., Ferreira, H., & Pereira, C. (2019, July). Internalização de novos membros em equipes de desenvolvimento de software: benefícios e limitações. In *Anais do IV Workshop sobre Aspectos Sociais, Humanos e Econômicos de Software* (pp. 61-70). SBC.
- Nerur, S., Mahapatra, R., & Mangalaraj, G. (2005). Challenges of migrating to agile methodologies. *Communications of the ACM*, 48(5), 72-78.

- Nerur, S., Cannon, A., Balijepally, V., & Bond, P. (2010). Towards an understanding of the conceptual underpinnings of agile development methodologies. In *Agile software development* (pp. 15-29). Springer, Berlin, Heidelberg.
- Packlick, J. (2007, August). The agile maturity map a goal oriented approach to agile improvement. In *Agile 2007 (AGILE 2007)* (pp. 266-271). IEEE.
- Peffer, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of management information systems*, 24(3), 45-77.
- Petersen, K., & Wohlin, C. (2010). The effect of moving from a plan-driven to an incremental software development approach with agile practices. *Empirical Software Engineering*, 15(6), 654-693.
- Pirzadeh, L. (2010). Human factors in software development: a systematic literature review (Master's thesis).
- PMI, P., & PMI. (2017). Um guia do conhecimento em gerenciamento de projetos (Guia PMBOK). Project Management Institute Inc.
- Poppendieck, M., & Poppendieck, T. (2009). Implementando o desenvolvimento Lean de Software: do conceito ao dinheiro. Bookman Editora.
- Potter, N., & Sakry, M. (2009). Implementing SCRUM (agile) and CMMI together. *The Process Group-Post newsletter*, 16(2), 1-6.
- Quinn, R. E. (2003). *Competências gerenciais: princípios e aplicações*. Elsevier.
- Ralph, K. (2006). *Gestão de Projetos: uma abordagem global*. Tradução de Cid Knipel Moreira–São Paulo: Editora Saraiva.
- Ressel, L. B., Beck, C. L. C., Gualda, D. M. R., Hoffmann, I. C., Silva, R. M. D., & Sehnem, G. D. (2008). O uso do grupo focal em pesquisa qualitativa. *Texto & Contexto-Enfermagem*, 17(4), 779-786.
- Ribeiro, J. (2008). *comportamento organizacional*.
- Rovai, R. L. (2013). Metodologias inovadoras para gestão de projetos: modelo referencial para implantação da ITILV3 através da metodologia PRINCE2: estudo de caso. *Revista de Gestão e Projetos-GeP*, 4(2), 252-270.
- SEI (2010). Team, C. P. CMMI® for Development, Version 1.3, CMMI-DEV, V1. 3, Improving processes for developing better products and services November 2010. TECHNICAL REPORT, CMU/SEI-2010-TR-033, ESC-TR-2010-033. Software Engineering Process Management Program. <http://www.sei.cmu.edu>.
- Semuel, H., Siagian, H., & Octavia, S. (2017). The effect of leadership and innovation on differentiation strategy and company performance. *Procedia-Social and Behavioral Sciences*, 237, 1152-1159.
- Sharp, H., & Robinson, H. (2010). Three ‘C’s of agile practice: collaboration, coordination and communication. In *Agile software development* (pp. 61-85). Springer, Berlin, Heidelberg.
- Shrum, S., & Phillips, M. (2004). CMMI overview for executives. Software Engineering Institute.

- Silva, R. C. D. C. (2017). Um estudo sobre a influência de fatores humanos e culturais em projetos de desenvolvimento de software ágeis (Master's thesis, Universidade Federal de Pernambuco).
- Sirmon, D. G., Hitt, M. A., Arregle, J. L., & Campbell, J. T. (2010). The dynamic interplay of capability strengths and weaknesses: Investigating the bases of temporary competitive advantage. *Strategic Management Journal*, 31(13), 1386-1409.
- Soares, F. S. F., & Meira, S. R. L. (2013, October). An agile maturity model for software development organizations. In *Proceedings of the Eighth International Conference on Software Engineering Advances (ICSEA'13)*, IARIA, Venice, Italy (pp. 324-328).
- Softex, (2012). Guia Geral, M. P. S. Mps. br-melhoria de processo do software brasileiro.
- Softex, (2014). BR, M. Guia Geral MPS de Gestão de Pessoas.
- Sommerville, I. (2003). *Engenharia de Software*. 6^a. Edição. São Paulo: Addison Wesley.
- Staples, M., & Niazi, M. (2008). Systematic review of organizational motivations for adopting CMM-based SPI. *Information and software technology*, 50(7-8), 605-620.
- Stark, J. A., & Crocker, R. (2003). Trends in software process: the PSP and agile methods. *IEEE Software*, 20(3), 89-91.
- Sutherland, J., Jakobsen, C. R., & Johnson, K. (2008, January). Scrum and CMMI level 5: The magic potion for code warriors. In *Proceedings of the 41st Annual Hawaii International Conference on System Sciences (HICSS 2008)* (pp. 466-466). IEEE.
- Tripp, J. F., Riemenschneider, C., & Thatcher, J. B. (2016). Job satisfaction in agile development teams: Agile development as work redesign. *Journal of the Association for Information Systems*, 17(4), 1.
- Vargas, L. M. (2016). Gerenciamento Ágil de Projetos em Desenvolvimento de Software: Um Estudo Comparativo sobre a Aplicabilidade do Scrum em Conjunto com PMBOK e/ou PRINCE2. *Revista de Gestão e Projetos-GeP*, 7(3), 48-60.
- Versione (2018). 12th Annual State of Agile Report. Disponível em: <https://explore.versionone.com/state-of-agile/versionone-12th-annual-state-of-agile-report>>. Acesso em: ago. 2019.
- Whitworth, E., & Biddle, R. (2007, August). The social nature of agile teams. In *Agile 2007 (AGILE 2007)* (pp. 26-36). IEEE.
- Wohlin, C., & Aurum, A. (2015). Towards a decision-making structure for selecting a research design in empirical software engineering. *Empirical Software Engineering*, 20(6), 1427-1455
- Zhang, C. (2015, December). Design of Human Capability Maturity Analysis System Online P-CMM Model. In *2015 International Conference on Intelligent Transportation, Big Data and Smart City* (pp. 302-305). IEEE.