

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/307512360>

Proposição e Aplicação de uma Metodologia baseada no AHP e na ISO/IEC 25000 para apoiar a Avaliação da Qualidade de Softwares de Gestão de Projetos

Article · May 2017

CITATIONS

5

READS

585

2 authors, including:



Francisco Rodrigues Lima Junior

Federal University of Technology - Paraná/Brazil (UTFPR)

116 PUBLICATIONS 1,170 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Decision models for problems related to environmental management [View project](#)



Decision models for supplier selection and evaluation [View project](#)

Proposição e aplicação de uma metodologia baseada no AHP e na ISO/IEC 25000 para apoiar a avaliação da qualidade de *softwares* de gestão de projetos

Proposal and application of an approach based on AHP and ISO/IEC 25000 to support the evaluation of the quality of project management software systems

Matheus Henrique Bartolomeu Marques Morais¹ - Eng. de Prod. e Qualidade - Centro Univ. da Fund. Educacional Guaxupé
Francisco Rodrigues Lima Junior² - Univ. de São Paulo - Escola de Engenharia de São Carlos - Engenharia de Produção

RESUMO

No atual cenário competitivo, as organizações tendem a buscar o aprimoramento dos fatores que influenciam na condução e no sucesso de seus projetos. O uso de um *software* de gestão de projetos adequado aos requisitos da equipe e do gerente de projetos contribui para o alcance de resultados satisfatórios. Nesse contexto, este estudo propõe e aplica uma metodologia para apoiar a avaliação da qualidade de *softwares* de gestão de projetos baseada no método AHP (*Analytic Hierarchy Process*) e nos critérios e subcritérios da norma ISO/IEC 25000:2014. O método AHP é utilizado para definir os pesos dos critérios e dos subcritérios adotados. A metodologia proposta foi aplicada em um caso ilustrativo para selecionar o *software* mais adequado aos requisitos de três tomadores de decisão. Foram avaliados os *softwares* Basecamp, MS Project, Service Desk e Primavera. O maior desempenho final foi alcançado pelos *softwares* Primavera e MS Project, que atenderam a 29 dos 30 requisitos avaliados. Entretanto, devido a algumas particularidades desses *softwares*, o Primavera foi selecionado.

Palavras-chaves: *Software* de gestão de projetos. ISO/IEC 25000. Método AHP.

ABSTRACT

In the current competitive scenario, organizations are seeking to improve factors that influence the administration and the success of their projects. The adoption of a project management software suitable to organizational requirements contributes to ensure satisfactory results. In this context, this study proposes a methodology to support the quality evaluation of project management software, which is based on the AHP method (*Analytic Hierarchy Process*) as well as on the criteria and sub-criteria of ISO/IEC 25000. The AHP method is used to define the relevance of the criteria and sub-criteria chosen. The proposed methodology was applied to an illustrative case to select the most appropriate software. The alternatives Basecamp, MS Project, Service Desk and Primavera were evaluated. Primavera and MS Project achieved the higher global performance in the process evaluation, since they fulfill 29 of 30 requirements. However, due to some particularities of the evaluated products, Primavera was selected as being the most suitable.

Keywords: Project management software. ISO/IEC 25000. AHP method.

1. INTRODUÇÃO

Diante da necessidade de responder as pressões competitivas do mercado atual, empresas de diferentes setores vêm investindo na realização de projetos com o propósito de produzir resultados exclusivos e criar diferenciais competitivos (HAZIR, 2015; LINZALONE; SCHIUMA, 2015). Embora existam diferentes definições para a palavra “projeto”, uma das mais difundidas é apresentada no PMBOK (*Project Management Body of Knowledge*) que declara que um projeto é um esforço temporário empreendido para criar um produto, serviço ou resultado único (PMI, 2013). Hazir (2015) afirma que estruturas organizacionais baseadas em projetos são capazes de prover suporte ao alcance de objetivos com foco em responsabilidade, melhor controle e coordenação.

Segundo o PMBOK (PMI, 2013), o gerenciamento (ou a gestão) de projetos é a aplicação de conhecimentos, habilidades, ferramentas e técnicas para apoiar as atividades do projeto a fim de alcançar os resultados esperados. Um tipo de ferramenta capaz de apoiar as atividades inerentes à gestão de projetos consiste nos *softwares* PMIS (*Project Management Information System*). *Softwares* PMIS são capazes de auxiliar os gestores nas atividades de planejamento, organização, controle e comunicação (CANIËLS; BAKENS, 2012), possibilitando o gerenciamento integrado e o monitoramento contínuo do progresso dos projetos, a criação, a atualização e a atribuição de tarefas em tempo real, além de melhorias na comunicação entre a equipe e no gerenciamento dos recursos (BRAGLIA; FROSOLINI, 2014). Dessa forma, o uso correto desses *softwares* possibilita avançar a eficiência e reduzir os custos do projeto, podendo assim resultar em um aumento no retorno sobre o investimento (BRAGLIA; FROSOLINI, 2014).

A aceitação e o uso correto de um *software* PMIS dependem, principalmente, de sua capacidade de atender às necessidades do gestor, da equipe e dos *stakeholders* do projeto (BRAGLIA; FROSOLINI, 2014). Tais necessidades podem ser traduzidas em requisitos relacionados à gestão do escopo, da qualidade, dos prazos e dos riscos do projeto e, também, em requisitos relacionados à usabilidade, eficiência, segurança e portabilidade do *software* (LIMA JUNIOR; FONDAZZI, 2010; BRAGLIA; FROSOLINI, 2014). Nesse sentido, a avaliação da qualidade de *softwares* PMIS se mostra necessária para verificar o nível de adequação de uma ou mais ferramentas computacionais aos requisitos implícitos e explícitos dos membros e *stakeholders* do projeto (TRENDOWICZ; KOPCZYNSKA, 2014).

Por envolver a avaliação de uma ou mais alternativas considerando um conjunto de métricas de desempenho quantificáveis, a avaliação da qualidade de *software* vem sendo abordada por diversos estudos como um problema de tomada de decisão multicritério (LIMA JUNIOR et al., 2012; YANG, 2012; TRENDOWICZ; KOPCZYNSKA, 2014). A partir de buscas realizadas em diversas bases de dados, verificou-se a inexistência de metodologias de tomada de decisão multicritério voltadas à avaliação da qualidade de *softwares* PMIS. Em um estudo de revisão sistemática da literatura, Trendowicz e Kopczynska (2014) analisaram 18 metodologias multicritério de apoio à avaliação da qualidade de *softwares* e também não encontraram nenhum estudo voltado para a avaliação de *softwares* PMIS. Diante desse contexto, este estudo propõe e aplica uma metodologia para apoiar a avaliação da qualidade de *softwares* PMIS baseada no método AHP (*Analytic Hierarchy Process*) e nas características e subcaracterísticas da qualidade sugeridas pela norma ISO/IEC 25000:2014.

2. REFERENCIAL TEÓRICO

2.1. Gestão de projetos e *softwares* PMIS

A gestão de projetos engloba um conjunto de atividades com o propósito de identificar e atender efetivamente aos requisitos dos clientes e *stakeholders* do projeto. Para isso, é necessário gerenciar as limitações e os objetivos conflitantes relacionados ao escopo, à qualidade, ao cronograma, aos orçamentos, aos recursos e aos riscos do projeto (PMI, 2013). Uma norma que pode ser usada por organizações públicas, privadas ou comunitárias para apoiar qualquer tipo de projeto, independentemente do tamanho e complexidade, é a ISO 21500:2012. Essa norma fornece uma descrição de alto nível dos conceitos e processos que são considerados para formar boas práticas em gerenciamento de projetos (ISO 21500, 2012).

Rolstadås et al. (2014) ressaltam que os desafios da equipe do projeto crescem conforme aumenta o número de *stakeholders* internos e externos, passando a envolver mais culturas distintas e podendo criar barreiras de comunicação. Em equipes de projeto, o compartilhamento de conhecimento é fundamental porque cria relacionamentos colaborativos entre os membros da equipe, melhora a comunicação e pode conduzir a benefícios como redução de custos, aumento da satisfação e diminuição da rotatividade dos funcionários (NAVIMIPOUR;

CHARBAND, 2016). Por incluir atividades relacionadas à gestão do conhecimento, a gestão de projetos conduz ao aprendizado organizacional contínuo, tornando mais fácil o acúmulo do conhecimento gerado pelas experiências vivenciadas pela equipe do projeto (MARCELINO-SÁDABA; PÉREZ-EZCURDIA; GONZÁLEZ-JAEN, 2015). Além de ser essencial para o compartilhamento de conhecimentos, o suporte de *softwares* é indispensável em várias outras atividades da gestão de projetos (HAZIR, 2015).

De acordo com Braglia e Frosolini (2014), o uso de *softwares* de apoio à gestão de projetos (PMIS) permite aumentar a eficiência, tornando o ciclo de vida do projeto mais visível, uma vez que todos os usuários são capazes de controlar tarefas específicas e podem ter uma melhor compreensão sobre como o projeto está acontecendo. Dependendo da complexidade e do porte do projeto, muitos requisitos podem ser requeridos do *software*. Alguns requisitos básicos se referem à capacidade de gerenciar recursos humanos, físicos e financeiros, utilizar indicadores de desempenho, gerar relatórios, dar suporte ao funcionamento multiusuário, permitir o compartilhamento de arquivos e a troca de informações entre a equipe e operacionalizar técnicas básicas de gestão de projetos, como gráfico de Gantt e rede PERT (LIMA JUNIOR; FONDAZZI, 2010). O processo de escolha de um *software* PMIS necessita considerar os requisitos de seus usuários e não deve se orientar apenas pelos custos (BRAGLIA, M; FROSOLINI, 2014).

Braglia e Frosolini (2014) acrescentam que muitos gerentes de projeto usam *softwares* PMIS, mas a maioria deles ainda desconhece os potenciais benefícios que poderiam ser alcançados. Para esses autores, isso se deve principalmente à falta de um entendimento do que esse tipo de *software* realmente é e de como ele pode influenciar os projetos, desde a sua concepção até a sua conclusão. O uso de metodologias de avaliação da qualidade de *software* possibilita verificar o alinhamento entre as necessidades existentes e os recursos oferecidos pelos *softwares* em uso. Além disso, tais metodologias ajudam a orientar o processo de escolha de um novo *software* PMIS.

2.2. Avaliação da qualidade de *softwares*

Para Pressman (2011), a qualidade de um *software* se refere à conformidade com os requisitos funcionais e de desempenho explicitamente declarados, a padrões de desenvolvimento claramente documentados e às características implícitas que são esperadas de todo *software* profissionalmente desenvolvido.

Os modelos de qualidade permitem avaliar *softwares* de acordo com diferentes aspectos e comumente representam as características desejáveis a um *software* em uma estrutura hierárquica (YANG, 2012). Algumas normas foram criadas para padronizar internacionalmente as características de implementação e avaliação da qualidade de *software*. A série de normas ISO/IEC 25000:2014 (*Software Quality Requirements and Evaluation*, SQuaRE), dividida em 5 partes, foi desenvolvida para fornecer suporte à definição de requisitos e ao processo de avaliação da qualidade de *softwares*. O modelo de qualidade apresentado na ISO/IEC 25010:2014 aborda duas dimensões da qualidade de *software*. A primeira delas é a qualidade em uso, que especifica características de qualidade relacionadas à interação humana com o sistema. A qualidade em uso se refere à capacidade de atender a requisitos para atingir metas específicas com produtividade, efetividade, segurança e satisfação do usuário, em um contexto de uso especificado (ISO/IEC25000, 2014).

A segunda dimensão proposta pela ISO/IEC 25010 é a qualidade do produto, que define um conjunto de oito características da qualidade relacionadas a atributos internos e externos do software. A adequação funcional é a capacidade do *software* de prover funções que atendam às necessidades implícitas e explícitas quando usado em condições especificadas. A eficiência de desempenho se refere ao desempenho em relação à quantidade de recursos utilizados. A confiança é a capacidade de manter um nível de desempenho especificado em um determinado período de tempo. A usabilidade é a capacidade de ser usado para atingir metas específicas com efetividade, produtividade e satisfação do usuário. A manutenibilidade é a capacidade de ser modificado visando à melhoria, correção ou adaptação a mudanças no ambiente ou nos requisitos. A portabilidade é a capacidade de ser transferido de um *hardware*, sistema operacional ou ambiente de uso para outro. A compatibilidade é a capacidade de trocar informações com outros *softwares* e desempenhar as funções que lhe forem requeridas enquanto compartilha recursos de *hardware* e *software*. A segurança diz respeito à capacidade do *software* de proteger informações para que pessoas ou outros *softwares* tenham o nível de acesso apropriado aos seus níveis de permissão (ISO/IEC25000, 2014). Conforme descreve o Quadro 1, o modelo de qualidade apresentado na ISO/IEC 25010 desdobra as características da qualidade em um conjunto de subcaracterísticas relacionadas.

Quadro 1 – Características e subcaracterísticas da qualidade de *software*.

CARAC-TERÍSTICA	SUBCARAC-TERÍSTICA	DEFINIÇÃO DA SUBCARACTERÍSTICA
Adequação funcional	Integridade funcional	Capacidade de atender às tarefas e aos objetivos específicos do usuário a que foi destinado.
	Correção funcional	Capacidade de fornecer resultados corretos e com precisão.
	Adequação funcional	Capacidade de realizar tarefas e certos objetivos de maneira fácil.
Eficiência de desempenho	Comportamento em relação ao tempo	Capacidade de fornecer tempos de resposta e processamento apropriados quando o <i>software</i> executa suas funções.
	Utilização de recursos	Grau com que os tipos e as quantidades de recursos usados atende aos requisitos.
	Capacidade	Grau com que a capacidade máxima de parâmetros do <i>software</i> atende aos requisitos.
Usabilidade	Reconhecibilidade	Grau com que o usuário é capaz de reconhecer se o produto é adequado às suas necessidades.
	Apreensibilidade	Capacidade que o <i>software</i> possui de ser usado para alcançar objetivos específicos de aprendizagem, de forma eficiente, eficaz e sem riscos, garantindo que o usuário se sinta satisfeito no contexto em questão.
	Operacionalidade	Capacidade do <i>software</i> de permitir ao usuário operá-lo e controlá-lo de forma fácil.
	Proteção de erro	Capacidade de proteger os usuários de cometer falhas.
	Estética da interface	Capacidade de possuir uma interface que seja satisfatória ao usuário.
	Acessibilidade	Capacidade de ser usado por usuários com diferentes características e habilidades para alcançar objetivos especificados.
Compatibilidade	Coexistência	Capacidade de compartilhar recursos com outro <i>software</i> sem causar impactos sobre qualquer outro produto.
	Interoperabilidade	Capacidade de interagir com um ou mais sistemas especificados.
Confiança	Maturidade	Capacidade de atender às necessidades de confiabilidade quando operado em condições normais.
	Disponibilidade	Capacidade de ser operacional e acessível quando requerido para uso.
	Tolerância a falhas	Capacidade de garantir um nível de desempenho especificado em caso de ocorrência de falhas de <i>software</i> ou <i>hardware</i> .
	Capacidade de recuperação	Capacidade de restabelecer seu nível de desempenho especificado e recuperar os dados diretamente afetados no caso de uma falha.

CARAC-TERÍSTICA	SUBCARAC-TERÍSTICA	DEFINIÇÃO DA SUBCARACTERÍSTICA
Segurança	Confidencialidade	Capacidade de permitir acesso de dados somente a usuários autorizados.
	Integridade	Capacidade de bloquear acesso e modificações de usuários não autorizados.
	Não repúdio	Capacidade de comprovar ações, eventos, alterações e envio de informações para que não possam ser repudiados futuramente.
	Responsabilidade	Capacidade de rastrear as ações de entidades específicas.
	Autenticidade	Capacidade de comprovar a identidade de um sujeito ou recurso caso seja requerido.
Manutenibi-lidade	Modularidade	Capacidade de alterar elementos do <i>software</i> com impacto mínimo.
	Reutilização	Capacidade que os componentes do <i>software</i> possuem de serem utilizados por outros sistemas existentes ou em construção.
	Analisabilidade	Capacidade de avaliar o impacto de uma mudança em uma ou mais partes de um sistema, diagnosticar partes do sistema em que podem haver falhas e identificar componentes a serem modificados.
	Modificabilidade	Capacidade de permitir que uma modificação seja implementada sem causar defeitos no produto existente.
	Testabilidade	Capacidade de estabelecer critérios de teste para um sistema que foi modificado e de determinar se esses critérios foram cumpridos de forma eficaz e eficiente.
Portabilidade	Adaptabilidade	Capacidade de ser adaptado para ambientes de operação especificados, sem a necessidade de aplicação de outras ações ou meios além daqueles fornecidos para essa finalidade pelo <i>software</i> considerado.
	Instalabilidade	Capacidade de ser corretamente instalado e / ou desinstalado em um ambiente especificado.
	Substituibilidade	Capacidade de substituir outro <i>software</i> no mesmo ambiente e para o mesmo fim.

Fonte: ISO/IEC25000 (2014).

Lima Junior et al. (2012) destacam que a forma como algumas das subcaracterísticas da qualidade são definidas pela ISO/IEC não permite sua medição direta, devido à subjetividade de interpretação e medição destas subcaracterísticas. Por isso, é necessário desdobrar as subcaracterísticas em requisitos objetivos que sirvam como atributos de medição. O Quadro 2 exemplifica o desdobramento de uma subcaracterística em atributos.

Quadro 2 – Desdobramento de uma subcaracterística da qualidade em atributos.

SUBCARACTERÍSTICA	ATRIBUTOS
Adequação funcional	O <i>software</i> provê agendamento de tarefas?
	Possui gráfico de Gantt?
	Possui redes de atividades?
	Permite o gerenciamento da estrutura organizacional?

Fonte: Lima Junior et al. (2012).

Embora a ISO/IEC 25010 sugira um conjunto de 31 subcaracterísticas da qualidade, a realização da avaliação de um ou mais *softwares* usando todas essas subcaracterísticas pode se tornar inviável devido ao tempo consumido. Dessa forma, essas características e subcaracterísticas devem servir de referência para a criação de modelos de avaliação adequados ao tempo, aos recursos humanos e às informações disponíveis (JUNG, 2007).

Na literatura podem ser encontrados diversos modelos de avaliação da qualidade para tipos de *software* específicos. Silva et al. (2015) desenvolveram um modelo para avaliar a qualidade de um sistema acadêmico com base nos critérios facilidade de uso, comportamento em relação a falhas, eficiência e satisfação das necessidades. Rocha (2012) propôs um modelo para suportar a avaliação da qualidade de um *website* que considera a qualidade do conteúdo, a qualidade do serviço e a qualidade técnica. Lima Junior e Fondazzi (2010) propuseram um modelo de apoio à avaliação da qualidade de *softwares* PMIS com base na norma ISO/IEC 9126. Uma limitação desse modelo é a impossibilidade de avaliar os níveis de importância relativa (ou pesos) das características e subcaracterísticas da qualidade. Como os requisitos dos usuários variam de acordo com o contexto de aplicação do *software*, é desejável atribuir pesos diferentes para as características da qualidade (YANG, 2012). Esse processo de ponderação pode ser apoiado por métodos de tomada de decisão multicritério (TRENDOWICZ; KOPCZYNSKA, 2014).

2.3. Uso de métodos multicritério no apoio à avaliação da qualidade de *software*

Um processo de tomada de decisão pode ser entendido como um estudo que envolve a identificação e a seleção de alternativas considerando o nível em que estas satisfazem necessidades, requisitos, objetivos e expectativas de pessoas responsáveis pela decisão, denominadas tomadores de decisão (LIMA JUNIOR; OSIRO; CARPINETTI, 2014; TRENDOWICZ; KOPCZYNSKA, 2014). Segundo Trendowicz e Kopczynska (2014), a avaliação da qualidade de *software* pode

ser entendida como um caso específico de problema de tomada de decisão multicritério. O Quadro 3 apresenta algumas analogias entre elementos básicos da tomada de decisão multicritério e da avaliação da qualidade de *software*.

Quadro 3 – Analogias entre elementos da decisão multicritério e da avaliação da qualidade de *software*.

TOMADA DE DECISÃO MULTICRITÉRIO	AValiação DA QUALIDADE DE SOFTWARES
Objetivo do problema: descreve o objetivo da decisão (por exemplo, ranquear ou categorizar alternativas).	Propósito da avaliação: verificar o nível de qualidade do produto (atendimento aos requisitos), se o produto requer investimentos, modificações ou se deve ser abandonado.
Alternativas: um conjunto de possíveis soluções para o problema de decisão.	Produtos que estão sendo avaliados: um escopo delimitado de ou mais <i>softwares</i> ou um produto avaliado holisticamente.
Crítérios e subcritérios: fatores considerados para avaliar ou comparar alternativas.	Características e subcaracterísticas da qualidade: requisitos ou expectativas em relação ao <i>software</i> .
Atributos: características das alternativas que são importantes para avaliar seu desempenho em um determinado subcritério.	Propriedade mensurável do <i>software</i> : propriedade do <i>software</i> cujo valor pode ser determinado. Por exemplo, tempo de resposta.

Fonte: Adaptado de Trendowicz e Kopczynska (2014).

Trendowicz e Kopczynska (2014) analisaram 18 aplicações de métodos de decisão multicritério (*Multicriteria Decision Making*, MCDM) em problemas de avaliação da qualidade de *software*. Esses autores constataram que uma ampla diversidade de métodos MCDM vem sendo usada, incluindo os métodos AHP, ANP (*Analytic Network Process*), SMART (*Simple Multiattribute Rating Theory*), *Fuzzy AHP*, *Fuzzy ANP*, entre outros. De acordo com Trendowicz e Kopczynska (2014), o método AHP é o mais utilizado no apoio a este tipo de problema. O Quadro 4 descreve algumas aplicações de métodos MCDM que envolvem a avaliação da qualidade de *softwares*. Nessas aplicações, tais métodos são usados para apoiar a escolha de um novo produto de *software*. Entre as aplicações analisadas por este estudo e por Trendowicz e Kopczynska (2014), não foram encontradas metodologias focadas no apoio à avaliação de *softwares* PMIS.

Quadro 4 – Algumas aplicações de metodologias multicritério na avaliação da qualidade de *software*.

AUTORES	TIPO DE APLICAÇÃO	DESCRIÇÃO
Wang e Yang (2007)	<i>Software</i> ERP (<i>Enterprise Resource Planning</i>)	Seleção de ERP com base em fatores econômicos, fatores relacionados à qualidade do produto, gestão, riscos, estratégia e recursos. <i>Fuzzy ANP</i> é usado para calcular o desempenho global do <i>software</i> .
Olson (2007)	<i>Software</i> ERP	Seleção de <i>software</i> ERP usando o método SMART.
Lima Junior et al. (2012)	<i>Software</i> de tomada de decisão	Os critérios e os subcritérios são baseados na ISO/IEC 9126 e o método AHP é usado para ponderá-los.
Yang (2012)	<i>Software</i> de gestão financeira	Os critérios são baseados na ISO/IEC 25000:2004. Os valores de desempenho das alternativas em cada critério são agregados usando o método .

Fonte: Proposto pelo autor.

A escolha de um método MCDM adequado necessita considerar o alinhamento entre os requisitos do problema modelado e as características do método MCDM (LIMA JUNIOR; OSIRO; CARPINETTI, 2014). Alguns requisitos para a escolha de um método MCDM apropriado à avaliação da qualidade de *software* consistem na capacidade de lidar com informações incertas e com julgamentos subjetivos de especialistas, capacidade de fornecer suporte à tomada de decisão em grupo e de prover um processo sistemático de ponderação de critérios (TRENDOWICZ; KOPCZYNSKA, 2014). Com base nos estudos propostos por Lima Junior et al. (2012) e Trendowicz e Kopczynska (2014), este estudo adotou o método AHP devido à sua capacidade de lidar com incerteza e subjetividade, de apoiar a tomada de decisão em grupo e de prover um procedimento comparativo sistemático de ponderação.

2.4. O método AHP

O método AHP foi desenvolvido por Thomas Saaty no início da década de 1970. Para aplicá-lo, é necessário estruturar uma hierarquia de três níveis, em que o primeiro nível representa o objetivo do problema, o segundo define os critérios de decisão e o terceiro identifica as alternativas. De acordo com a

necessidade do problema, haverá mais níveis na hierarquia se forem incluídos subcritérios (SAATY, 1980; SAATY, 2008). Na estrutura hierárquica do AHP, os elementos de um mesmo nível devem ser comparados par a par por meio dos julgamentos de especialistas do domínio de problema abordado (LIMA JUNIOR et al., 2012). A escala mostrada no Quadro 5 é comumente usada para avaliar o peso dos critérios e dos subcritérios, bem como o desempenho das alternativas. A fim de prover um melhor entendimento do método AHP, é apresentada a seguir uma síntese dos passos deste método com base em Saaty (1980), Saaty (2008) e Salomon (2010).

Quadro 5 – Escala usada para comparação dos elementos do problema no método AHP.

1/9	1/7	1/5	1/3	1	3	5	7	9
Extrema- mente	Bastante	Muito	Pouco	Igual impor- tância	Pouco	Muito	Bastante	Extrema- mente
...menos importante					...mais importante			

Fonte: Saaty (1980).

Primeiramente, é necessário estruturar a hierarquia de decisão a partir do objetivo do problema. Em seguida, deve-se construir uma matriz de comparação pareada para determinar o peso dos critérios. Se os critérios estiverem decompostos em subcritérios, estes devem ser avaliados seguindo os mesmos procedimentos. Esta comparação deve ser feita somente entre subcritérios de um mesmo critério. Na sequência, constroem-se matrizes de comparação para avaliar as alternativas. A quantidade de matrizes de comparação de alternativas equivale ao número de critérios adotados. Deve-se verificar a consistência das comparações realizadas pelos especialistas a partir do cálculo do índice de consistência dos julgamentos (*consistency ratio* – CR), conforme a Equação 1, onde λ é o autovalor máximo e n é a ordem da matriz de comparações. Considera-se que os julgamentos são coerentes se não ultrapassarem o limite de 0,1. Julgamentos em que $CR > 0,1$ devem ser revistos (SAATY, 1980; SALOMON, 2010).

$$CR = \frac{\lambda - n}{(n-1) * RI} \quad (1)$$

O índice de consistência dos julgamentos também considera um erro de consistência aleatória (*random consistency index* – RI), cujos valores são tabelados e determinados de acordo com a ordem da matriz de comparações. Após

assegurar a consistência dos julgamentos, usando a matriz de comparação entre os critérios, deve-se obter o vetor de pesos (autovetor W) por meio do cálculo do autovalor máximo, de acordo com a Equação 2. Nessa expressão, D é uma matriz quadrada normalizada e λ é o autovalor máximo. Para obter o vetor de pontuações das alternativas em relação a cada critério, devem ser realizados procedimentos análogos aos descritos no item anterior. Em seguida, conforme mostra a Equação 3, os valores do autovetor de pesos W (w_1, w_j, \dots, w_m) e os valores dos vetores de desempenho de cada alternativa i em relação a cada critério X_j ($x_{1j}, x_{ij}, \dots, x_{nj}$) são usados para calcular o desempenho global de cada alternativa (D_i). Finalmente, é necessário ordenar o desempenho final das alternativas em ordem decrescente. A normalização do ranking pode fornecer uma melhor visualização sobre o quão preferível é cada alternativa (SAATY, 1980; SALOMON, 2010).

$$D * W = \lambda * W \quad (2)$$

$$D_i = \sum_{j=1}^m (w_j * x_{ij}) \quad (3)$$

Embora o AHP ajude a entender melhor e a avaliar as soluções de problemas de tomada de decisão multicritério, ele não elimina a necessidade do apoio de especialistas que forneçam informações baseadas na experiência, intuição e/ou em dados físicos. O fato de esse método ser capaz de lidar com avaliação de fatores qualitativos e subjetivos com base na opinião de especialistas contribui para que o mesmo seja de grande valor no apoio à avaliação do nível de importância das características e subcaracterísticas da qualidade de *software* pelos clientes e usuários do sistema (LIMA JUNIOR et al., 2012).

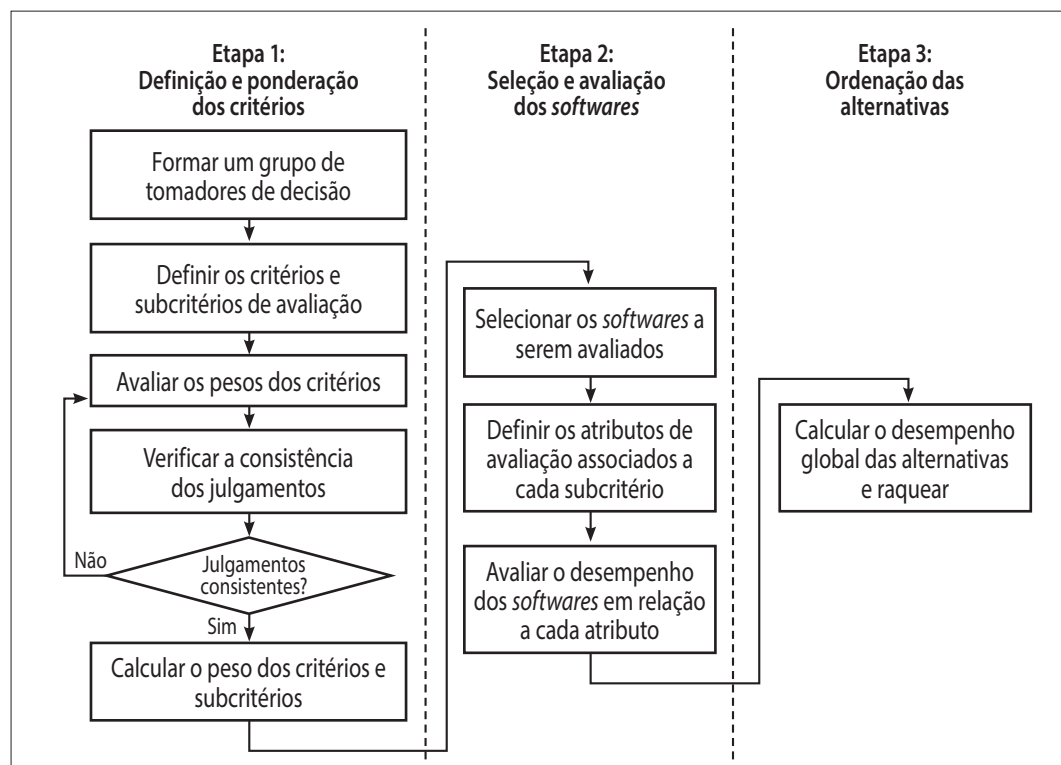
3. MÉTODO

Esta pesquisa pode ser caracterizada como modelagem e simulação computacional (PIDD, 2004), pois envolve o uso de modelos quantitativos, baseados em um conjunto de variáveis que representam um domínio específico, existindo um relacionamento causal e quantitativo entre essas variáveis (LIMA JUNIOR; OSIRO; CARPINETTI, 2014). Quanto à organização da pesquisa, esta foi dividida nas etapas de pesquisa de campo, modelagem e simulação. A pesquisa de campo envolveu a coleta de dados junto a três especialistas de uma empresa. Esses especialistas escolheram alguns critérios e subcritérios de avaliação de *softwares* PMIS conforme seus requisitos e avaliaram a importância desses critérios e subcritérios.

Na etapa de modelagem e simulação, um modelo computacional baseado no AHP foi construído usando o *software Expert Choice*. Os julgamentos coletados foram inseridos nesse modelo para calcular os valores dos pesos dos critérios e dos subcritérios. Um dos autores deste estudo atuou junto aos funcionários da empresa na aplicação da metodologia proposta, apoiando a seleção dos critérios, subcritérios e atributos, assim como a avaliação dos pesos e do desempenho dos *softwares*. Os valores resultantes foram usados para calcular o desempenho final dos *softwares* por meio do MS Excel. Seguindo as definições apresentadas por Pidd (2004), as simulações realizadas são caracterizadas como: discretas, pois as variáveis dependentes variam discretamente em pontos específicos do tempo simulado; estáticas, uma vez que o fator tempo não influencia na simulação e; determinísticas, já que os valores das variáveis não apresentam aleatoriedade.

A metodologia proposta por este estudo para apoiar a avaliação da qualidade de *softwares* de gestão de projetos (PMIS) é baseada na norma ISO/IEC 25000:2014 e no AHP. Como ilustrado na Figura 1, a metodologia proposta é composta por três etapas.

Figura 1 – Metodologia proposta para apoiar a avaliação da qualidade de *softwares* de gestão de projetos.



Fonte: Proposto pelo autor.

Na etapa 1, é necessário compor um grupo de funcionários de áreas relacionadas à gestão de projetos, para atuar como tomadores de decisão e avaliadores do(s) *software(s)*. Trendowicz e Kopczynska (2014) sugerem que sejam incluídos nesse grupo *stakeholders* da área de qualidade do projeto. Os tomadores de decisão devem escolher as características (critérios) e as respectivas subcaracterísticas (subcritérios) da qualidade sugeridas pela ISO/IEC 25000:2014 que melhor correspondem às suas necessidades. Em seguida, deve-se coletar os julgamentos dos tomadores de decisão em relação aos pesos dos critérios e dos subcritérios de forma comparativa, usando a escala mostrada no Quadro 3. Também é necessário avaliar o índice de inconsistência de cada uma das matrizes comparativas seguindo os procedimentos propostos por Saaty (1980). Os julgamentos que apresentarem índice de inconsistência maior que 0,1 devem ser revistos. Caso todas as matrizes estejam consistentes, devem-se agregar os julgamentos fornecidos pelos diferentes tomadores de decisão. Os valores dos pesos de cada critério e subcritério devem ser calculados usando o método AHP (SAATY, 2008) por meio de *softwares* específicos ou de planilhas eletrônicas.

Na etapa 2, definem-se os *softwares* a serem avaliados de acordo com o objetivo do processo de avaliação (compra de novo produto ou avaliação de *softwares* em uso). Após isso, os tomadores de decisão devem determinar um ou mais atributos de avaliação para cada critério. Tais atributos devem ser desdobrados a partir dos requisitos funcionais (o que o *software* deve fazer) e não funcionais (como deve fazer) dos usuários, membros da equipe e *stakeholders* do projeto. Os *softwares* devem ser avaliados de forma objetiva pelos tomadores de decisão ou por um analista de suporte. Durante o processo de avaliação, manuais, arquivos de ajuda e outros materiais de apoio devem ser consultados.

Finalmente, na etapa 3, o objetivo é o cálculo do desempenho final de cada *software*. Para isso, para cada *software* avaliado (i), primeiramente, deve-se calcular o percentual de atributos atendidos em cada subcritério (x_{ij}). Conforme representa a Equação 3, o desempenho final de uma alternativa (D_i) é dado pela média ponderada entre o peso dos subcritérios e a pontuação de cada alternativa em relação a cada subcritério. Após o cálculo do desempenho final, ordenam-se as alternativas de forma decrescente de acordo com a pontuação alcançada.

4. DISCUSSÃO E RESULTADOS

Uma empresa necessita adquirir um software PMIS. Para isso, necessita avaliar o quanto as opções disponíveis no mercado atendem aos seus requisitos. Três funcionários (um desenvolvedor de sistemas e dois gerentes de projetos) formaram um comitê para avaliar algumas das alternativas disponíveis e decidir qual delas atende melhor aos seus requisitos. A partir da análise de uma lista dos critérios e subcritérios da norma ISO/IEC 25000:2014, os tomadores de decisão selecionaram o conjunto mostrado no Quadro 6.

Quadro 6 – Critérios e subcritérios escolhidos pelos tomadores de decisão.

CRITÉRIOS E SUBCRITÉRIOS RELACIONADOS
Adequação funcional (C_1): adequação funcional (SC_1)
Eficiência de desempenho (C_2): utilização de recursos (SC_2)
Compatibilidade (C_3): Interoperabilidade (SC_3)
Usabilidade (C_4): apreensibilidade (SC_4), operacionalidade (SC_5), estética da interface (SC_6), acessibilidade (SC_7)
Confiança (C_5): capacidade de recuperação (SC_8)
Segurança (C_6): confidencialidade (SC_9), integridade (SC_{10})
Portabilidade (C_7): instalabilidade (SC_{11})

Fonte: Proposto pelo autor.

Para definir o peso dos critérios e subcritérios, os julgamentos dos tomadores de decisão foram coletados usando a escala de valores do Quadro 5. A Tabela 1 apresenta somente os julgamentos relativos aos pesos dos critérios. Os julgamentos referentes aos subcritérios foram omitidos devido à limitação de espaço do artigo. Usando a Equação 1, o índice de inconsistência (IC) foi calculado para todas as matrizes de julgamentos. Todos os valores obtidos foram menores que 0,1, o que atesta a consistência dos julgamentos coletados. Os julgamentos dos tomadores de decisão foram combinados usando média geométrica. Os pesos calculados para C_1 , C_2 , C_3 , C_4 , C_5 , C_6 e C_7 usando a Equação 2 foram, respectivamente, 0,126; 0,044; 0,036; 0,373; 0,161; 0,230; 0,030. Portanto, os critérios com maior peso são usabilidade e segurança. Os pesos calculados para os subcritérios são mostrados na Tabela 2.

Tabela 1 – Valores dos julgamentos dos tomadores de decisão em relação aos pesos dos critérios.

Tomador de decisão 1	CR = 0,06	C ₁	C ₂	C ₃	C ₄	C ₅	C ₆	C ₇
	C ₁	-	9	9	1/5	4	1/2	9
	C ₂	-	-	1	1/8	1/3	1/5	1
	C ₃	-	-	-	1/8	1/3	1/5	1
	C ₄	-	-	-	-	4	3	9
	C ₅	-	-	-	-	-	1/4	4
	C ₆	-	-	-	-	-	-	9
	C ₇	-	-	-	-	-	-	-
Tomador de decisão 2	CR = 0,05	C ₁	C ₂	C ₃	C ₄	C ₅	C ₆	C ₇
	C ₁	-	7	4	1/5	1/3	1/3	4
	C ₂	-	-	1/2	1/7	1/6	1/6	2
	C ₃	-	-	-	1/6	1/4	1/4	3
	C ₄	-	-	-	-	3	3	9
	C ₅	-	-	-	-	-	1	7
	C ₆	-	-	-	-	-	-	7
	C ₇	-	-	-	-	-	-	-
Tomador de decisão 3	CR = 0,04	C ₁	C ₂	C ₃	C ₄	C ₅	C ₆	C ₇
	C ₁	-	1	5	1/6	1/4	1/4	3
	C ₂	-	-	5	1/6	1/4	1/4	3
	C ₃	-	-	-	1/7	1/7	1/7	1/2
	C ₄	-	-	-	-	1	1	1
	C ₅	-	-	-	-	-	1	7
	C ₆	-	-	-	-	-	-	7
	C ₇	-	-	-	-	-	-	-
Julgamentos combinados	CR = 0,03	C ₁	C ₂	C ₃	C ₄	C ₅	C ₆	C ₇
	C ₁	-	3,9791	5,6462	1/5,3133	1/1,4422	1/2,8845	4,7622
	C ₂	-	-	1,3572	1/6,9520	1/4,1602	1/4,9324	1,8171
	C ₃	-	-	-	1/6,9520	1/4,3795	1/5,1925	1,1447
	C ₄	-	-	-	-	2,2889	2,0801	8,2768
	C ₅	-	-	-	-	-	1/1,5874	5,8088
	C ₆	-	-	-	-	-	-	7,6117
	C ₇	-	-	-	-	-	-	-

Fonte: Proposto pelo autor.

Após a ponderação dos critérios e subcritérios, os tomadores de decisão escolheram 4 alternativas de *software* PMIS: o *Microsoft Project Server* (MS) versão 2013, a versão gratuita do *Basecamp* (B), o *Primavera P6* (P) e o módulo de gestão de projetos do *software Service Desk* (SD). Baseando-se em Lima Junior e

Fondazzi (2010) e Kostalova, Tetreova e Svedik (2015), os atributos para avaliar esses 4 *softwares* foram definidos pelos autores deste estudo em conjunto com os tomadores de decisão, assim como mostra o Quadro 7.

Quadro 7 – Atributos de *softwares* de gestão de projetos selecionados a partir dos requisitos dos tomadores de decisão.

SUBCARACTERÍSTICA	ATRIBUTOS	B	MS	P	SD
Adequação funcional (SC ₁)	O <i>software</i> permite o cadastro de instituições (como clientes, fornecedores e <i>stakeholders</i>)?	Sim	Sim	Sim	Sim
	O <i>software</i> permite a criação de lista de tarefas?	Sim	Sim	Sim	Sim
	O <i>software</i> permite a priorização de tarefas?	Não	Sim	Sim	Sim
	O <i>software</i> realiza a codificação de tarefas automaticamente?	Não	Sim	Sim	Sim
	O <i>software</i> permite o sequenciamento das atividades por meio de relação de dependência?	Não	Sim	Sim	Sim
	O <i>software</i> permite a organização das tarefas em níveis diferentes?	Sim	Sim	Sim	Sim
	Dispõe de calendários onde possam ser visualizadas as datas importantes do projeto?	Sim	Sim	Sim	Não
	O <i>software</i> permite a criação de linha de tempo (<i>timeline</i>)?	Não	Sim	Sim	Não
	O <i>software</i> permite a criação de gráfico de Gantt?	Não	Sim	Sim	Sim
	O <i>software</i> possui indicadores de desempenho sobre os custos estimados e realizados?	Não	Sim	Sim	Sim
	O <i>software</i> possui indicadores de desempenho sobre os níveis de produtividade da equipe?	Não	Sim	Sim	Sim
	O <i>software</i> possibilita o gerenciamento da utilização de recursos para nivelar o esforço?	Não	Sim	Sim	Sim
	O <i>software</i> emite relatórios para análise de desempenho do projeto?	Não	Sim	Sim	Sim
	O <i>software</i> permite a troca de informações entre a equipe do projeto?	Sim	Sim	Sim	Não
	O <i>software</i> permite <i>upload</i> e <i>download</i> de documentos referentes às tarefas e projetos?	Sim	Sim	Sim	Sim
	É possível escolher o idioma usado nos menus do <i>software</i> ?	Sim	Sim	Sim	Sim
	O <i>software</i> possibilita o cálculo da utilização de recursos financeiros?	Não	Sim	Sim	Sim
	O <i>software</i> calcula datas de início e término com base nas datas das tarefas “filhas”?	Sim	Sim	Sim	Sim
Utilização de recursos (SC ₂)	O <i>software</i> utiliza menos memória RAM que a média dentre os <i>softwares</i> avaliados?	Sim	Não	Não	Sim
Interoperabilidade (SC ₃)	Possibilita a interação com outros sistemas, permitindo importar e exportar dados?	Sim	Sim	Sim	Sim
Apreensibilidade (SC ₄)	O <i>software</i> apresenta tutorial que auxilia o usuário?	Sim	Sim	Sim	Sim
Operacionalidade (SC ₅)	O <i>software</i> possui serviço de suporte técnico ao usuário?	Não	Sim	Sim	Sim
	O <i>software</i> permite trabalhar <i>off-line</i> ?	Não	Sim	Sim	Não
Estética da interface do usuário (SC ₆)	O <i>software</i> dispõe de atalhos constantemente visíveis para as funcionalidades essenciais?	Sim	Sim	Sim	Sim
	É possível customizar a exibição dos elementos da interface gráfica?	Não	Sim	Sim	Não
Acessibilidade (SC ₇)	O <i>software</i> exibe o nome dos ícones dos menus, facilitando a compreensão dos usuários?	Sim	Sim	Sim	Sim
Capacidade de recuperação (SC ₈)	O <i>software</i> permite a recuperação de dados em caso de falha?	Não	Sim	Sim	Sim
Confidencialidade (SC ₉)	O <i>software</i> possibilita o uso de senha de acesso para cada usuário?	Sim	Sim	Sim	Sim
Integridade (SC ₁₀)	O <i>software</i> permite a definição de níveis de acesso diferentes para os usuários?	Sim	Sim	Sim	Sim
Instalabilidade (SC ₁₁)	O <i>software</i> pode ser acessado por meio de dispositivos alternativos (<i>tablet</i> ou celular)?	Sim	Sim	Sim	Não

Fonte: Proposto pelo autor.

Com o suporte de um dos autores desta pesquisa, os tomadores de decisão avaliaram o desempenho dos 4 *softwares* em relação aos atributos escolhidos. Os *softwares* foram estudados durante, aproximadamente, um mês e diversas simulações de uso foram feitas. O Quadro 7 também apresenta o resultado da avaliação dos 4 produtos. O percentual de atributos atendidos por cada produto em cada critério e os pesos dos subcritérios foram usados para calcular o desempenho final (D_i) de cada *software* de acordo com a Equação 3. Os valores resultantes foram, posteriormente, normalizados. A Tabela 2 apresenta os pesos calculados para cada subcritério, os percentuais de atributos atendidos pelas alternativas para cada subcritério e o desempenho final de cada alternativa.

Tabela 2 – Peso dos subcritérios, percentual de atributos atendidos e desempenho final de cada *software*.

CRITÉRIOS	SUBCRITÉRIOS	PESO DOS SUBCRITÉRIOS	PERCENTUAL DE REQUISITOS ATENDIDOS			
			B	MS	P	SD
C ₁	SC ₁	0,1260	0,4444	1,0000	1,0000	0,8333
C ₂	SC ₂	0,0440	1,0000	0	0	1,0000
C ₃	SC ₃	0,0360	1,0000	1,0000	1,0000	1,0000
C ₄	SC ₄	0,1231	1,0000	1,0000	1,0000	1,0000
	SC ₅	0,0966	0	1,0000	1,0000	0,5000
	SC ₆	0,0224	0,5000	1,0000	1,0000	0,5000
	SC ₇	0,1313	1,0000	1,0000	1,0000	1,0000
C ₅	SC ₈	0,1610	0	1,0000	1,0000	1,0000
	SC ₉	0,0941	1,0000	1,0000	1,0000	1,0000
C ₆	SC ₁₀	0,1359	1,0000	1,0000	1,0000	1,0000
C ₇	SC ₁₁	0,0300	1,0000	1,0000	1,0000	0
Desempenho final (D_i)			0,6616	0,9564	0,9564	0,8899
Desempenho final normalizado			0,1910	0,2761	0,2761	0,2569
Classificação relativa			P = MS > SD > B			

Fonte: Proposto pelo autor.

O *MS Project* e o *Primavera* se mostraram os produtos mais adequados para apoiar a gestão de projetos na empresa em questão, uma vez que obtiveram o maior desempenho final normalizado, igual a 0,2761. Dentre os 30 requisitos avaliados, ambos os *softwares* atenderam a 29, o que é um ótimo desempenho. O

software Service Desk se mostrou uma boa alternativa, atendendo a 24 requisitos e apresentando desempenho final normalizado de 0,2569. Em contrapartida, o *Basecamp* se mostrou uma alternativa menos viável, já que atendeu apenas 16 requisitos e alcançou desempenho final normalizado de 0,191. Por não atenderem a alguns requisitos, todos os *softwares* obtiveram uma pontuação nula em ao menos um subcritério. O *MS Project* e o Primavera não atenderam às expectativas voltadas à utilização de recursos (SC₂). O *Service Desk* teve um baixo desempenho em relação aos subcritérios operacionalidade (SC₅), estética da interface (SC₆) e instalabilidade (SC₁₁). Já o *Basecamp* não atendeu a alguns dos requisitos relacionados à adequação funcional (SC₁), operacionalidade (SC₅), estética da interface (SC₆) e capacidade de recuperação (SC₈).

Como o desempenho avaliado do *MS Project* e do Primavera foi o mesmo, algumas pequenas diferenças foram consideradas pelos tomadores de decisão para chegar à escolha final. O *MS Project* tem boa compatibilidade com outros produtos da empresa Microsoft. Além disso, ele é de fácil utilização devido à interface intuitiva, exigindo menos treinamento para sua operacionalização. Em relação ao Primavera, o mesmo possui algumas funcionalidades que o favorecem em relação ao *MS Project*, tais como permitir que vários usuários trabalhem em um único projeto ao mesmo tempo e usar uma quantidade não limitada de linhas de base. Diante disso, optou-se pela escolha do Primavera.

5. CONCLUSÃO

Este estudo propôs uma nova metodologia para apoiar a avaliação da qualidade de *softwares* de gestão de projetos, baseada nas características e subcaracterísticas da norma ISO/IEC 25000:2014 e no AHP. Essa metodologia foi aplicada em um caso que envolveu a avaliação da qualidade de *softwares* PMIS com o propósito de selecionar um novo *software* que fosse adequado aos requisitos de seus usuários. Os *softwares* *Microsoft Project*, *Basecamp*, Primavera e *Service Desk* foram avaliados considerando 7 critérios e 11 subcritérios. O desempenho desses *softwares* nesses subcritérios foi quantificado por meio de 30 atributos. O maior desempenho final foi alcançado pelos *softwares* Primavera e *MS Project*, mas, devido a algumas diferenças entre eles, o Primavera foi selecionado.

Para aplicar a abordagem proposta, as métricas devem ser imparciais, objetivas e alinhadas com as particularidades do cenário de uso. Como os resultados da avaliação dependem diretamente das métricas usadas, uma aplicação desta metodologia a fim de avaliar a adequação dos mesmos *softwares* em outro contexto de uso provavelmente obterá resultados distintos. Ressalta-se que o processo de avaliação da qualidade de *softwares* requer o estudo detalhado das ferramentas computacionais de interesse. Os profissionais requeridos para a avaliação dos *softwares* necessitam ter, inicialmente, apenas conhecimentos na

área de gestão de projetos e de informática básica. Contudo, necessitam investir um tempo considerável no estudo e no uso simulado dos produtos avaliados, com o apoio de arquivos de ajuda, suporte ao usuário, vídeos instrucionais, entre outros.

A avaliação de *softwares* apresentada neste estudo complementa o estudo de Lima Junior e Fondazzi (2010), ao apresentar uma avaliação de quatro ferramentas diferentes daquelas já avaliadas por estes autores, utilizando alguns critérios, subcritérios e atributos adicionais. Pesquisas futuras podem adotar a metodologia proposta para apoiar a aquisição de um novo *software* de gestão de projetos ou para avaliar o nível de adequação de *softwares* em uso visando à identificação dos aspectos que precisam ser melhorados. Estudos futuros também podem usar a metodologia proposta por este estudo para avaliar sites que oferecem ferramentas de apoio à gestão de projetos ou para avaliar *softwares* como *GP Web*, *Runrun*, *Zoho Projects*, *Pivoltracker*, *Artia* e *Smartsheet*.

REFERÊNCIAS

- BRAGLIA, M; FROSOLINI, M. An integrated approach to implement Project Management Information Systems within the Extended Enterprise. **International Journal of Project Management**, v. 32, n. 1, p. 18-29, 2014.
- CANIËLS, M. C. J.; BAKENS, R. J. J. M. The effects of Project Management Information Systems on decision making in a multi project environment. **International Journal of Project Management**, v. 30, p. 162–175, 2012.
- HAZIR, O. A review of analytical models, approaches and decision support tools in project monitoring and control. **International Journal of Project Management**, v. 33, n. 4, p. 808-815, 2015.
- INSTITUTE, Project Management. **A guide to the project management body of knowledge (PMBOK guide)**. 5 ed. United States: Project Management, 2013. 589 p.
- ISO 21500:2012. **Guidance on Project Management**. Genebra: ISO, 2012.
- ISO/IEC 25000:2014. **Software engineering - System and software Quality Requirements and Evaluation (SQuaRE)**. Genebra: ISO, 2014.
- JUNG, H. Validating the external quality sub characteristics of software products according to ISO/IEC 9126. **Computer Standards & Interfaces**, v. 29, p.653-661, 2007.

- KOSTALOVA, J.; TETREVOVA, L; SVEDIK, J. Support of Project Management Methods by Project Management Information System. **Procedia - Social and Behavioral Sciences**, p. 96-104, 2015.
- LIMA JUNIOR, F. R.; CARPINETTI, L. C. R.; OSIRO, L.; GANGA, G. M. Avaliação da qualidade de softwares de apoio à decisão. *In*: ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO, 32, 2012, Bento Gonçalves, RS. **Anais...** Bento Gonçalves, RS: ENEGEP, 2012.
- LIMA JUNIOR, F. R.; FONDAZZI, L. A. M. Avaliação da qualidade de software voltados a gestão de projetos. *In*: ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO, 30, 2010, São Carlos, SP. **Anais...** São Carlos, SP: ENEGEP, 2010.
- LIMA JUNIOR, F. R.; OSIRO, L.; CARPINETTI, L. C. R. A comparison between Fuzzy AHP and Fuzzy TOPSIS methods to supplier selection. **Applied Soft Computing**, v. 21, p. 194-209, 2014.
- LINZALONE, R.; SCHIUMA, G. A review of program and project evaluation models. **Measuring Business Excellence**, v. 19, n. 3, p. 90-99, 2015.
- MARCELINO-SÁDABA, S.; PÉREZ-EZCURDIA, A.; GONZÁLEZ-JAEN, L. F. Using Project Management as a way to sustainability: from a comprehensive review to a framework definition. **Journal of Cleaner Production**, v. 99, n. 15, p. 1-16, 2015.
- NAVIMIPOUR, N. J.; CHARBAND, Y. Knowledge sharing mechanisms and techniques in project teams: Literature review, classification, and current trends. **Computers in Human Behavior**, v. 62, p. 730-742, 2016.
- OLSON, D. L. Evaluation of ERP outsourcing. **Computers & Operations Research**, v. 34, n. 12, p. 3715-3724, 2007.
- PIDD, M. **Computer Simulation in management Science**. 5 ed. Lancaster, United Kingdom: Ed. John. Wiley & Sons, 2004, 328 p.
- PRESSMAN, R. S. **Engenharia de Software**. 7ª ed., São Paulo, SP: Makron Books, 2011.
- ROCHA, A. Framework for a global quality evaluation of a website. **Online Information Review**, v. 36, n. 3, p. 374-382, 2012.
- ROLSTADÅS, A.; TOMMELEIN, I.; SCHIEFLOE, P. M.; BALLARD, G. Understanding project success through analysis of project management approach. **International Journal of Managing Projects in Business**, v. 7, n. 4, p. 638-660, 2014.

SAATY, T. L. Decision Making with the Analytic Hierarchy Process. **International Journal of Services Sciences**, v. 1, n. 1, p. 83-98, 2008.

SAATY, T. L. **The Analytic Network Process**. 1 ed. McGraw Hill: New York, 1980.

SALOMON, V. A. P. Analytic Hierarchy Process. *In*: MARINS, F.A.S; PEREIRA, M. S.; BELDERRAIN, M. C. N.; URBINA, L. S. (Org.). **Métodos de tomada de decisão com múltiplos critérios**. São Paulo, Blucher, p. 21-40, 2010.

SILVA, S. V.; SALLES, A. R. A.; NETO, C. M. S.; CABRAL, C. P. C.; SILVA, J.; MARCELINO, J. V.; SILVA, J. R. P. P.; BERNARDES, L. H. R.; FRANCO, P. M.; SOUSA, R. M.; FERREIRA, R. K. R.; TEIXEIRA, S. S. F.; ROCHA, W. S. Avaliação da qualidade de um sistema acadêmico: estudo de caso no Q-acadêmico. *In*: CONGRESSO INTEGRADO DE TECNOLOGIA DE INFORMAÇÃO, 8, 2015, Campo dos Goytacazes. **Anais...** Campo dos Goytacazes, 2015.

TRENDOWICZ, A.; KOPCZYNSKA, S. Adapting Multi-Criteria Decision Analysis for Assessing the Quality of Software Products: Current Approaches and Future Perspectives. **Advances in Computers**, v. 93, p. 153-226, 2014.

WANG, J.-J.; YANG, D.-L. Using a hybrid multi-criteria decision aid method for information systems outsourcing. **Computers and Operations Research**, v. 34, n. 12, p. 3691-3700, 2007.

YANG, H. Measuring software product quality with ISO standards base on fuzzy logic technique. *In*: LUO, J. (Ed.). **Affective Computing and Intelligent Interaction**. New York: Springer-Verlag Berlin Heidelberg, 2012. p. 59-67.