

UM MODELO PARA SUPORTE AO GERENCIAMENTO DE QUALIDADE DE SOFTWARE COM BASE NA DINÂMICA DE SISTEMAS

Germano Fenner

germanofenner@gmail.com
Universidade Federal do Ceará -
UFC, Fortaleza, CE, Brasil.

Alberto Sampaio Lima

albertosampaio@ufc.br
Universidade Federal do Ceará -
UFC, Quixadá, CE, Brasil.

José Maria da Silva Monteiro Filho

jmmfilho@gmail.com
Universidade Federal do Ceará -
UFC, Fortaleza, CE, Brasil.

Lincoln Souza Rocha

lincoln@dc.ufc.br
Universidade Federal do Ceará -
UFC, Fortaleza, CE, Brasil.

José Antão Beltrão Moura

antao@dsc.ufcg.edu.br
Universidade Federal de Campina
Grande – UFCG, Campina Grande,
PB, Brasil.

José Neuman de Souza

neuman.souza@gmail.com
Universidade Federal do Ceará -
UFC, Fortaleza, CE, Brasil.

RESUMO

O desenvolvimento de software consiste em um processo complexo, com inúmeras abordagens, que sugerem propostas de gerenciamento em fases bem definidas, visando uma produção que garanta aos produtos a qualidade desejada por seus clientes. Com base nas várias opções de tecnologia existentes e na crescente demanda em diferentes áreas das sociedades, é essencial que as empresas adotem modelos que permitam avaliar o processo de desenvolvimento e produção de software. Este trabalho apresenta um modelo para avaliar a qualidade de software dos cenários de negócios das organizações, com base na dinâmica dos sistemas. O modelo permite simulações de cenários, avaliando o percentual da capacidade de desenvolvimento de software, contribuindo para a melhoria dos processos de gerenciamento. Durante a fase de validação da pesquisa, foi realizado um estudo de caso em três empresas, e os resultados obtidos foram promissores em termos de preferência, utilidade e eficácia.

Palavras-chave: Gerenciamento de serviços de TI; gerenciamento de qualidade de software; dinâmica de sistemas; gerenciamento de TI orientado aos negócios.

1. INTRODUÇÃO

A globalização e a massiva ampliação do acesso à internet, associados ao crescimento da atuação das áreas de Tecnologia da Informação (TI) nas empresas, têm gerado uma demanda contínua e crescente por um serviço específico: a produção de software. Cada vez mais, os negócios dependem dos serviços de TI que envolvem software. Dessa forma, existe uma preocupação em gerenciar e manter estes serviços de forma efetiva (OGC, 2007). Os gestores das organizações bem-sucedidas reconhecem que a TI é tão significativa para os negócios como qualquer outra parte da organização (Fernandes e Abreu, 2012).

Serviços de TI cujos resultados possam ser mensurados (Lahtela *et al.*, 2010) são requisitados nos aplicativos móveis, nas ferramentas de trabalho, bem como na integração dos diversos tipos de equipamentos que podem se conectar à internet. O software é o componente essencial de um serviço de TI. Sem um suporte de software efetivo, os processos empresariais (ABPMP, 2013) não acontecem. Quando um evento é disparado, seja na forma de um lembrete de compromisso de reunião, um alerta de controle ou simplesmente a obtenção de uma informação eletrônica, existe a utilização de códigos de programação, onde, dependendo do grau de criticidade, um mínimo erro pode gerar consequências devastadoras para a organização. Um problema em um software pode comprometer a imagem de uma organização e colocar em cheque a sua credibilidade. A predição de comportamentos de processos tem sido estudada e avaliada com a proposição de diversas técnicas (de Mello, 2010).

O cenário global atual tem indicado uma crescente utilização de sistemas computacionais (softwares) integrados ao cotidiano de todos os setores da sociedade, atingindo um grande e diversificado número de usuários. Busca-se uma produção de software com qualidade (Guerra e Colombo, 2009).

Pontos críticos relacionados ao software incluem capacidade de processamento, exatidão e precisão, entre outros. Deve-se considerar que existe uma grande variedade de segmentos de software que abrangem aplicações comerciais, simulações militares, produtos educacionais, entre outros tipos. Ao se analisar a grande quantidade de itens e considerações que envolve a complexidade da confecção de um software, é possível fazer os seguintes questionamentos:

- Quais aspectos do negócio devem ser considerados na gestão da qualidade de software?
- Como tratar a qualidade de software no gerenciamento de serviços?

- Qual a influência/contribuição dos modelos de qualidade na visão do negócio?
- Qual a necessidade do avaliador/gestor em relação à efetividade das ferramentas de suporte à gestão?

Existe uma necessidade de se analisar diferentes cenários de negócio que envolvam a qualidade de software para que os gestores possam tomar decisões efetivas sobre o que deve ser testado para o atendimento das necessidades dos interessados. A presente pesquisa apresenta um novo modelo para simular e estimar quantitativamente a avaliação da qualidade de um software, dando suporte ao gerenciamento e tomada de decisão. A contribuição deste trabalho consiste na possibilidade de se realizar simulações para a avaliação de qualidade a partir de informações de cenários de negócios reais. As informações geradas são úteis para a tomada de decisão de gestores de qualidade e de serviços.

As contribuições geradas por esta pesquisa são de nível prático (desenho e implementação do modelo) e empírico (validação do modelo). Pode-se elencar ainda as seguintes contribuições específicas:

- Proposta de um modelo orientado ao negócio e inovador que possibilita a realização de simulações para suporte à tomada de decisão na gestão da qualidade de software;
- Alinhamento entre objetivos de negócio e informações geradas pelo modelo nas simulações.

Este artigo é constituído de seis seções. A primeira apresenta a introdução, enquanto a segunda seção apresenta a fundamentação teórica que norteou a pesquisa e mostra alguns trabalhos relacionados. Na terceira seção, o modelo proposto é detalhado. A metodologia é apresentada na quarta seção. O estudo de caso realizado e a análise dos resultados são discutidos na quinta seção. Na sexta seção são apresentadas as conclusões da pesquisa e trabalhos futuros.

2. FUNDAMENTAÇÃO TEÓRICA E TRABALHOS RELACIONADOS

Em 1977 foram dados os primeiros passos em direção à qualidade de software. O modelo de McCall avaliava o software em três níveis: o primeiro nível considerava a visão externa do software, como vista pelos usuários; o segundo nível avaliava a visão interna do software, como vista pelo desenvolvedor; e o terceiro nível considerava as métricas utilizadas para fornecer uma escala e métodos para medidas. Já o modelo da Hewlett-Packard, baseado no modelo de McCall, considerava mais dois atributos de qualidade, a funcionalidade, um conjunto de características e capaci-

dades do programa, generalidades das funções, no caso, a segurança do sistema global e a usabilidade, que levava em consideração os fatores humanos, estética global, consistência, documentação.

Esta pesquisa foi orientada pelos princípios de *Business-driven IT management (BDIM)* (Bartolini e Stefanelli, 2011), envolvendo a aplicação de um conjunto de modelos, práticas, técnicas e ferramentas, a fim de mapear e avaliar quantitativamente interdependências entre o desempenho dos negócios e soluções de TI. Teoria e prática em *BDIM* são discutidas a partir de uma perspectiva de tomada de decisão em Bartolini e Stefanelli (2011) e em Lima *et al.* (2018). Pesquisas *BDIM* também são apresentadas em Sousa *et al.* (2018) e Nóbrega *et al.* (2014). Os guias *COBIT - Control Objectives for Information and related Technology* (ISACA, 2014) e *ITIL - IT Infrastructure Library* (OGC, 2007) orientam sobre diretrizes para a classificação e priorização de investimentos que devem ser feitos em serviços. O guia *Project Management Body of Knowledge - PMBOK* (PMI, 2012) orienta sobre o gerenciamento de projetos que envolvem os serviços de TI.

Qualidade de software

A demanda por soluções de software tem crescido bastante nos últimos anos. Cada vez mais, existe um crescimento da complexidade das soluções ofertadas e uma maior exigência em termos de qualidade do produto. De acordo com Bartolini e Stefanelli (2011), a qualidade reflete o grau em que um conjunto de características inerentes a um objeto satisfaz seus requisitos. Quando maior for este grau, maior será a satisfação do cliente. O conceito de qualidade inicialmente foi aplicado na indústria. Com o passar o tempo, outras áreas adotaram a terminologia, tais como transportes (ISO 14.001 e 28.000), alimentos (ISO 22.000), segurança e saúde (ISO 45.001), entre outras.

Na área de TI, esforços em nível internacional para garantir a qualidade têm resultado na criação de normas como a ISO 20.000 para serviços de TI, a série da ISO 27.000, que trata da segurança da informação, além de diversas orientações como OGC (2007) e ISACA (2014), que abordam aspectos relacionados à Governança Corporativa de TI e ABNT (2015), que trata dos aspectos relacionados aos serviços de infraestrutura de TI. De acordo com Guerra e Colombo (2009), qualidade de software constitui uma área cuja demanda está crescendo significativamente, pois os usuários exigem cada vez mais eficiência e eficácia, dentre outras características de qualidade importantes para um produto tão especial como o software. Paralelamente à demanda do mercado, existe um movimento nacional e internacional no sentido de estabelecer normas na área de Engenharia de Software, como é o caso da ISO – Organização Internacional

de Normalização, no Subcomitê de Engenharia de Software, e da ABNT – Associação Brasileira de Normas Técnicas.

No Brasil, a partir de 1993, iniciativas do governo, universidades e indústria têm surgido na área de qualidade de produto e processos de software. Desde a sua criação, a Associação Brasileira de Normas Técnicas (ABNT) vem ampliando ações relacionadas à qualidade de software. Um exemplo disso foi a criação do subcomitê de software (ABNT, 2016), responsável por elaborar normas com o objetivo de gerar normas brasileiras relacionadas à qualidade de software.

Se por um lado a indústria vem continuamente aprimorando seus produtos e alinhando critérios com os padrões mais rigorosos em uso no mundo, a área de governança de TI necessita acompanhar essa evolução para oferecer suporte e condições para um efetivo alinhamento entre a TI e o negócio.

Ao se considerar as diversas normas ISO relacionadas ao assunto, sob a perspectiva de software, é possível se identificar que o assunto qualidade é bastante extenso. A norma ISO/IEC 9126 se refere a um conjunto de atributos que tem impacto na capacidade do software de manter o seu nível de desempenho dentro de condições estabelecidas por um dado período. Essa norma se divide em quatro partes:

1. ISO 9126-1:2001 – Ênfase nas diretrizes de uso e características de qualidade de produto de software;
2. ISO 9126-2:2003 – Refere-se às métricas externas para determinar a taxa de implementação das funções definidas na especificação de requisitos e o controle de ocorrência de falhas;
3. ISO 9126-3:2003 – Refere-se às métricas internas, verifica se as funções são adequadas ao que foi especificado, determina o número de falhas previstas, estima o tempo de resposta da aplicação e verifica se as alterações são registradas adequadamente;
4. ISO 9126-4:2004 – Refere-se às métricas de qualidade em uso baseadas na eficácia, produtividade, segurança e satisfação do usuário.

A norma ISO/IEC 9126 foi substituída no ano de 2011 pela norma ISO/IEC 25010 (ISO, 2016), que define as características de qualidade que todos os softwares devem ter. O intuito dessa substituição foi trazer ainda mais benefícios, tais como compatibilidade e segurança. Além do conjunto de normas da ISO, a indústria de software conta com metodologias para ajudar no processo de garantia de qualidade.

O *Capability Maturity Model Integration (CMMI)* é um modelo internacional de boas práticas específicas para o

desenvolvimento de produtos de software (da Silva, 2016). Conforme apresentado na Figura 1, o modelo CMMI está organizado em cinco níveis: 1 - Inicial, 2 - Gerenciado, 3 - Definido, 4 - Gerenciado e 5 - Otimizado. O CMMI é baseado nas melhores práticas para desenvolvimento e manutenção de produtos. Empresas que adotam essa prática possuem maiores chances de produzir softwares mais estáveis, confiáveis e seguros, pois o modelo assegura que a empresa segue padrões rígidos de processo de desenvolvimento de software.

A partir da existência do CMMI em nível internacional, foi criado no Brasil o modelo denominado Melhoria do Processo de Software Brasileiro – MPS.BR (PROMOVE, 2016). Trata-se de um programa brasileiro proposto pela organização SOFT-EX, com o propósito de desenvolver e disseminar as melhores práticas em desenvolvimento de software.

Conforme mostra a Figura 2, o modelo MPS.BR é organizado em sete níveis de maturidade. Ao contrário do CMMI, o MPS.BR utiliza letras, que vão de G até A, para determinar o nível de maturidade da organização. Esses níveis servem como um roteiro para melhoria contínua.

Com o passar do tempo, a produção de software deixou de ser uma atividade baseada apenas na intuição ou na experiência dos desenvolvedores, tendo sido objeto de inúmeros estudos. Busca-se oferecer modelos que possibilitem o gerenciamento da produção e assegurem a qualidade desejada por seus consumidores. Apesar da crescente e reconhecida complexidade envolvida no desenvolvimento de software, algumas empresas parecem não ter consciência de que precisam criar, ou mesmo aperfeiçoar, os seus processos internos (Mecenas e de Oliveira, 2005).

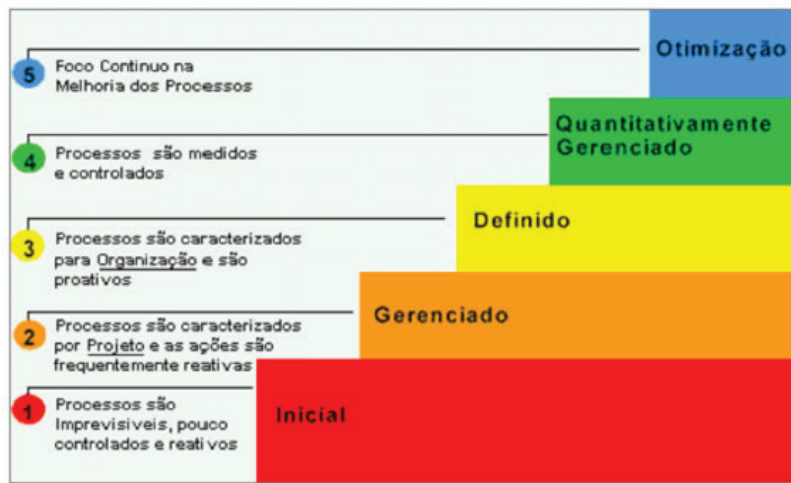


Figura 1. Estágios do Modelo CMMI

Fonte: da Silva (2016)

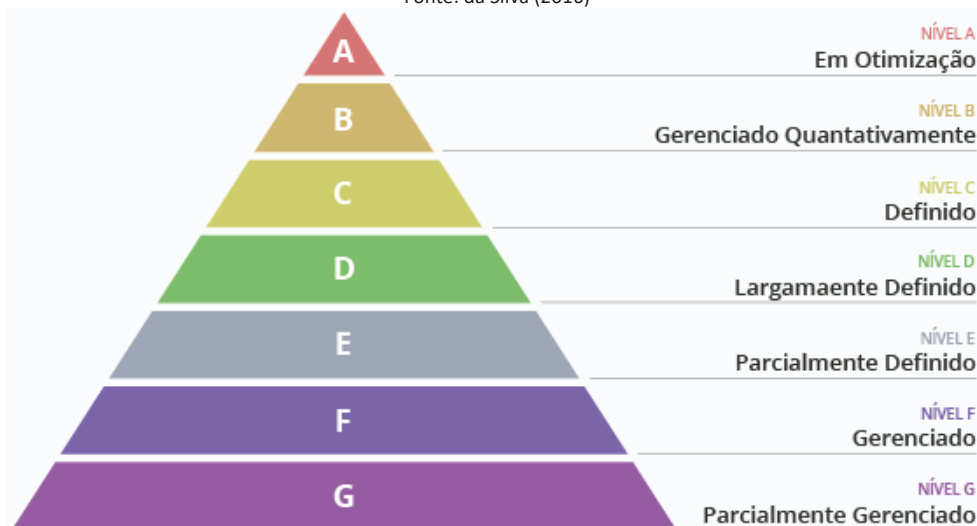


Figura 2. Estágios do MPS.BR

Fonte: PROMOVE (2016)

Dinâmica de sistemas

A teoria da dinâmica de sistemas (DS), aspecto da teoria de sistemas, foi criada para se modelar e analisar o comportamento complexo de sistemas (Sterman, 2000). O método tem como base o reconhecimento de que a estrutura de qualquer sistema (muitas vezes circulares e fechados), algumas vezes com atrasos entre os relacionamentos de seus componentes, é tão importante na determinação de seu comportamento quanto os próprios componentes individuais. A técnica permite modelar e analisar o comportamento em cenários complexos de sistemas sociais. A DS teve aplicação mais específica no contexto industrial em suas primeiras aplicações. No entanto, a técnica tem sido utilizada para o desenvolvimento de modelos dos mais diversos tipos de sistemas: social, econômico e ambiental, onde uma visão holística possui papel relevante e os enlaces de feedback são fundamentais para que se compreenda os complexos inter-relacionamentos (Rodrigues, 1996).

A DS é baseada em uma estrutura direta de fluxos e estoques, projetada para modelar sistemas com numerosas variáveis e com realimentações atrasadas entre as variáveis. Um modelo para um processo de demandas recebidas por um setor de TI é apresentado na Figura 3. A DS prevê o uso de dois tipos de modelos: diagramas de influência ou de enlaces-causais (modelos qualitativos) e modelos baseados em computador ou diagramas de estoque-fluxo (quantitativos). O primeiro é considerado o cerne da DS. Muitos estudos que adotaram a metodologia DS criaram apenas diagramas de influência. Esses diagramas são suficientes para a identificação dos diversos laços de feedback (reforço e balanceamento) e dos fatores que influenciam o fluxo do

sistema modelado. No entanto, apenas a partir da simulação em computadores (modelos quantitativos) pode-se ter uma visão dos reais efeitos dinâmicos desses laços de feedback e uma maior compreensão do sistema em análise. Dessa forma, as possibilidades oportunizadas por meio das possíveis simulações subsidiam a exploração de experimentação com o modelo (Sterman, 2000; Rodrigues, 1996).

É importante ressaltar que os modelos de DS apresentam uma visão dinâmica do desempenho dos sistemas de negócio (Sterman, 2000). Os modelos são do tipo diagramas “atividade na seta” ao invés de diagramas “atividade na caixa” como outras notações, sendo especialmente úteis no desenvolvimento de modelos dinâmicos de ciclo de vida que focam no desempenho geral de sistemas e no impacto de mudar variáveis-chave que afetam o desempenho geral. Os modelos buscam a visão do negócio (Morecroft, 2008) e são mais utilizados na modelagem e uma organização completa ou de uma linha de negócio, e pouco usados para modelos de fluxo de trabalho de baixo nível.

Trabalhos relacionados

Um modelo que usa dinâmica de sistemas não é estático, pois apresenta o movimento e reflete como a mudança em variáveis afeta um processo (Bezerra *et al.*, 2014).

O trabalho de Oliveira (2010) sugere um modelo para quantificar o valor de um serviço de TI, enquanto o trabalho de Lima *et al.* (2012) buscou quantificar o percentual de qualidade de um serviço de TI, considerando que o mesmo influencia o seu valor. Em Bezerra *et al.* (2014) é proposto

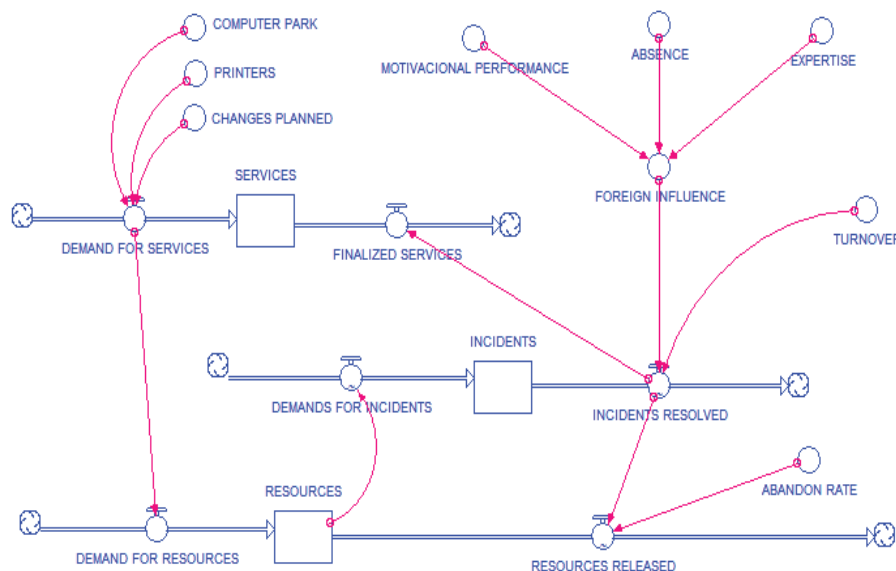


Figura 3. Modelagem com DS para um setor de TI

Fonte: O(s) próprio(s) autor(es)

um modelo de tomada de decisão sobre terceirização de TI. Em Wan *et al.* (2011) foi apresentado um modelo que avalia as capacidades dinâmicas e gestão de benefícios para avaliar gerenciamento de recursos de TI em empresas públicas que precisam de terceirização. Em Fernandes (2003) foi apresentada uma proposta de incorporação da metodologia de dinâmica de sistemas com as etapas de planejamento estratégico, utilizando o modelo do BSC. A utilização da técnica de dinâmica de sistemas em problemas relacionados a processos de software foi proposta em Macachy (2007) e Abdel-Hamid (1989).

O modelo proposto

A qualidade do produto de software pode ser avaliada de diversas maneiras. As avaliações irão variar de acordo com o porte da empresa, criticidade do negócio e tipo de segmento da organização. Existem diferentes visões da qualidade de software e de suas métricas. Essas métricas poderão variar de acordo com o estágio do ciclo de vida do software.

O modelo proposto considera as atividades interativas a partir da visão do guia ITIL OGC (2007). Essas atividades podem ser executadas em um ciclo de melhoria contínua, semelhante ao ciclo PDCA (Plan-Do-Check-Act) (Lima *et al.*, 2012). O modelo utiliza como referência a ISO/IEC 25010 (ISO, 2016). A qualidade do produto categoriza as propriedades de qualidade do produto em oito características (adequação funcional, confiabilidade, eficiência de desempenho, usabilidade, segurança, compatibilidade, manutenção e portabilidade). Cada característica é composta por um conjunto de subcaracterísticas relacionados (ISO, 2016). A Figura 4 mostra as características e subcaracterísticas de avaliação da ISO/IEC 25010.

O modelo proposto considera cada uma das subcaracterísticas indicadas na Figura 4. A modelagem do cenário com a utilização dos princípios de dinâmica de sistemas, por meio da ferramenta de software especializada (Isee Systems, 2016), utilizada nessa pesquisa é apresentada na Figura 5.

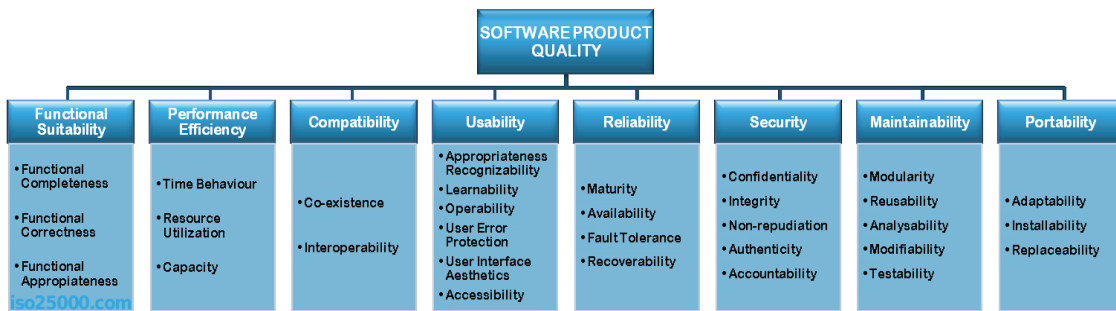


Figura 4. Características e subcaracterísticas da ISO 25.10

Fonte: ISO (2016)

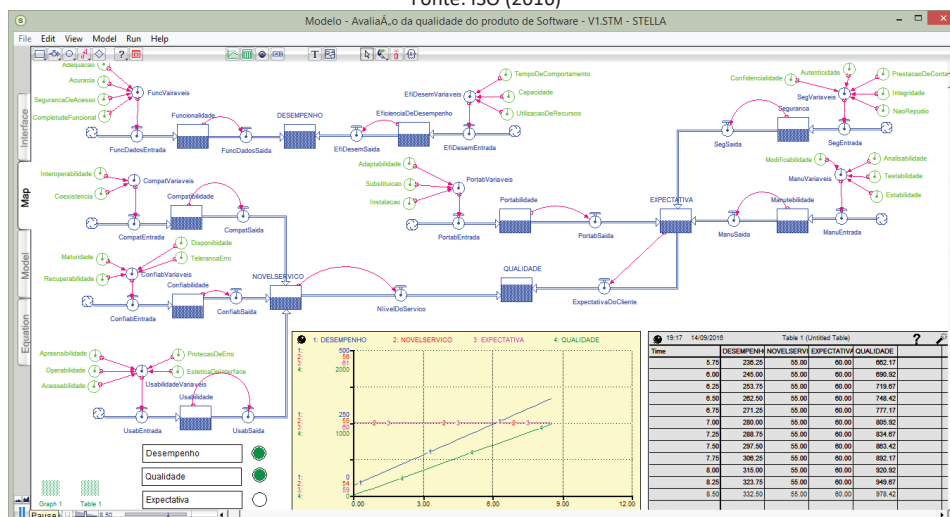


Figura 5. Modelagem das subcaracterísticas da ISO 25010

Fonte: O(s) próprio(s) autor(es)

O modelo proposto é genérico, podendo ser utilizado por qualquer empresa que necessite aplicar critérios de qualidade para avaliação do seu produto de software tendo como referência a norma ISO 25010. O modelo considera os aspectos mais genéricos de avaliação. Características muito específicas de cada empresa não foram consideradas nesta proposta.

Entrada de dados

O usuário (gestor) deve fornecer diversas informações para que o modelo possa realizar as simulações. A Figura 6 apresenta a tela inicial onde o avaliador/gestor fornece os dados, configurações necessárias para simulação.

Nesta etapa, o gestor deverá fornecer a maior quantidade de dados possível para que a simulação alcance um maior grau de precisão. As informações deverão ser configuradas nas seguintes características:

Funcionalidade

O modelo permite que o avaliador configure informações considerando a característica de funcionalidade do software (Figura 7). Essa funcionalidade aborda questões adequação, acurácia, segurança de acesso e completude funcional.

Eficiência de desempenho

Além das características de funcionalidade, o gestor deve fornecer informações à característica de eficiência de desempenho do software (Figura 8). Essa característica considera pontos como o tempo de comportamento, a utilização de recursos e a capacidade.

As entradas dos parâmetros de configuração apresentadas mostram o processo no modelo de dinâmica de sistemas, possuindo uma interface que facilita a entrada de dados, conforme apresentado na Figura 9.

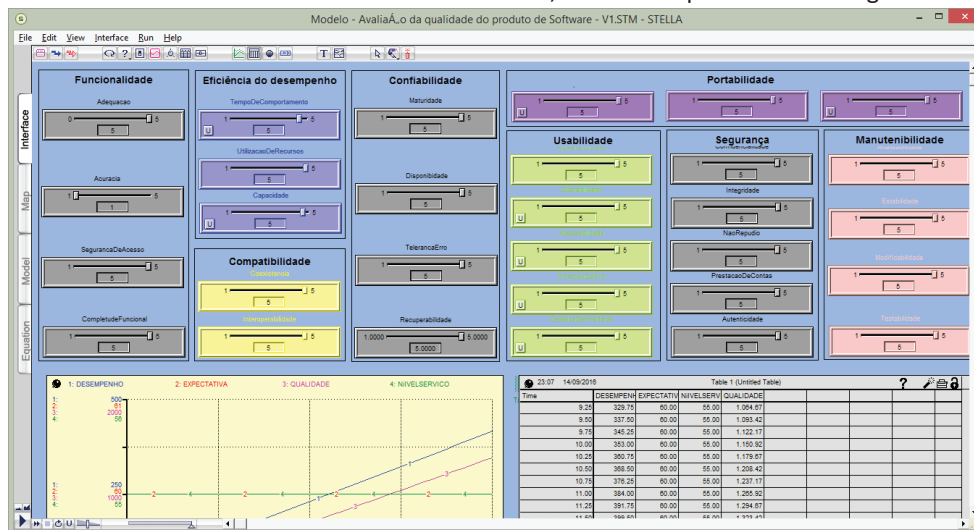


Figura 6. Tela de configuração de informações

Fonte: O(s) próprio(s) autor(es)

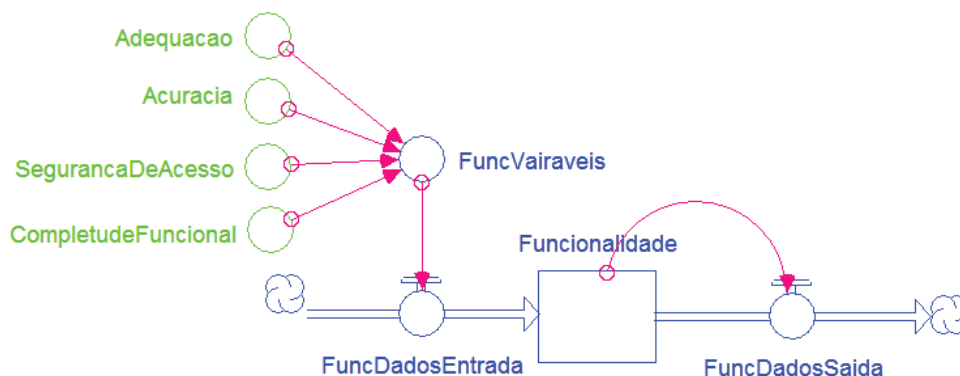


Figura 7. Configurações das características de funcionalidades

Fonte: O(s) próprio(s) autor(es)

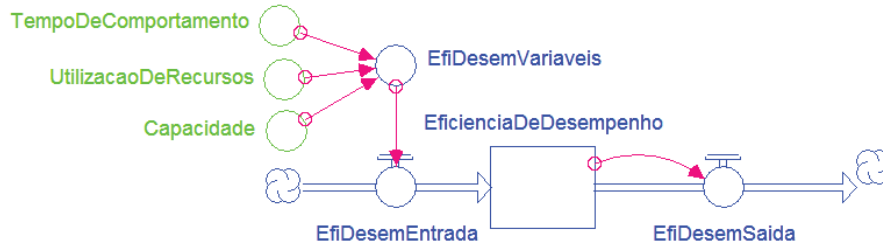


Figura 8. Configurações das características de eficiência de desempenho
 Fonte: O(s) próprio(s) autor(es)

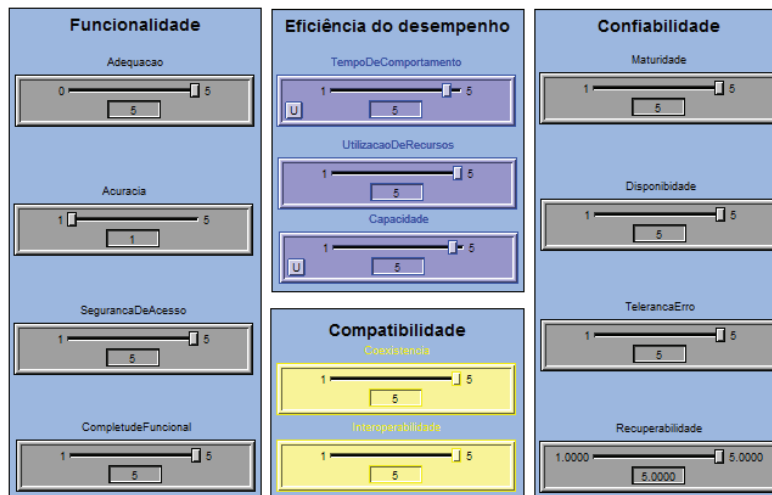


Figura 9. Interface com os parâmetros de configuração
 Fonte: O(s) próprio(s) autor(es)

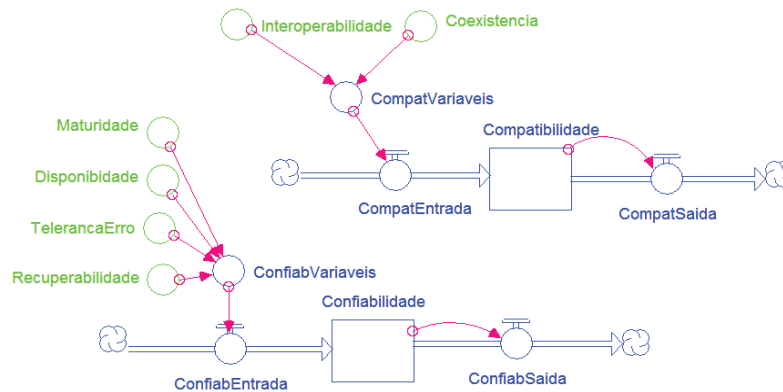


Figura 10. Configuração das características de Confiabilidade e Compatibilidade
 Fonte: O(s) próprio(s) autor(es)

Confiabilidade e Compatibilidade

A qualidade do software poderá variar consideravelmente a depender: da característica de compatibilidade, que avalia coexistência, interoperabilidade; e da confiabilidade, que avalia as características de maturidade, disponibilidade, tolerância ao erro e recuperabilidade (Figura 10).

Usabilidade e Segurança

Nesta etapa, o gestor deve informar parâmetros referentes às características de usabilidade e segurança. A usabilidade aborda subcaracterísticas relacionadas com apreensibilidade, operabilidade, acessibilidade, proteção de erro do usuário e estética da interface de usuário, enquanto a segurança aborda os aspectos de confidencialidade, inte-

gridade, não-repúdio, prestação de contas e autenticidade (Figura 11).

Manutenibilidade e Portabilidade

Finalmente, o gestor irá fornecer as últimas informações para avaliação do software, que são as características de manutenibilidade, que avalia os aspectos relacionados com analisabilidade, estabilidade, modificabilidade, testabilidade, portabilidade e, finalmente, as características de portabilidade, que irão considerar os aspectos de adaptabilidade, instalação e substituição (Figura 12), completando assim, os itens de avaliações recomendados pela ISO/IEC 25.010.

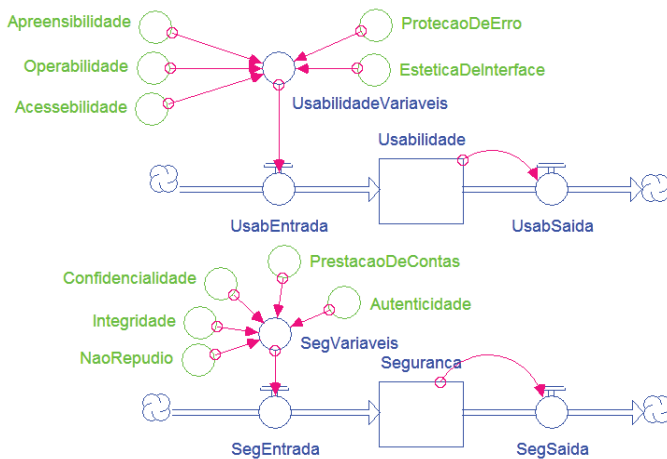


Figura 11. Configuração das características de usabilidade e segurança

Fonte: O(s) próprio(s) autor(es)

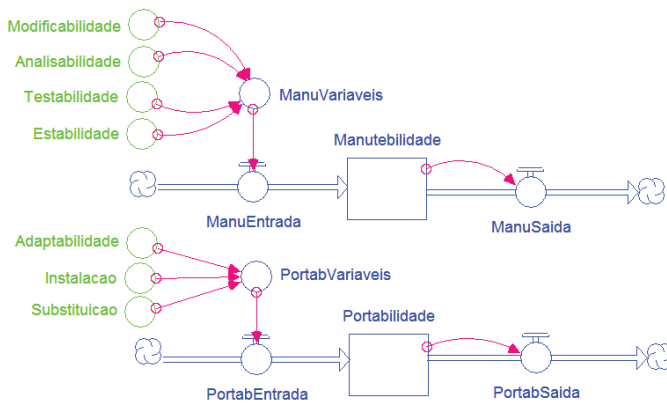


Figura 12. Configuração de manutenibilidade e portabilidade

Fonte: O(s) próprio(s) autor(es)

Caso o cenário avaliado não contemple a necessidade de todas as informações como, por exemplo: os testes da empresa não consideram subcaracterísticas tais como coexistência, interoperabilidade, não-repúdio e prestação de contas, bastará ele deixar zero para esses itens que o sistema irá

desconsiderar os dados referentes a estes parâmetros para geração dos resultados.

5. ASPECTOS METODOLÓGICOS

Visando a avaliação do modelo de acompanhamento da qualidade, foi realizado um estudo de caso múltiplo em três empresas especializadas na criação, desenvolvimento e implantação software.

A pesquisa realizada envolveu as seguintes fases:

- Revisão de literatura;
- Desenvolvimento do modelo;
- Reunião com os gestores das empresas;
- Coleta dos dados;
- Utilização do modelo com diferentes cenários em cada empresa;
- Apresentação e discussão dos resultados;
- Tabulação de resultados;
- Validação do modelo com os gestores/avaliadores;
- Divulgação dos resultados.

A presente pesquisa envolveu uma triangulação entre revisão de literatura, observação e realização de estudo de caso Yin (2005). Ao longo do processo de verificação e validação do modelo, foram utilizadas diferentes calibrações como parte do desenho dos experimentos, permitindo criar um ambiente ideal, reproduzir ambientes reais e testar ambientes alternativos. O aprimoramento do modelo foi definido a partir de estratégias de validação (corretude externa) e verificação (corretude interna), com calibragem realizada por meio da execução de simulações.

O processo de verificação e validação deve se preocupar com a criação de confiança suficiente em um modelo de simulação para que os seus resultados sejam aceitos pelos usuários e stakeholders (Radzicki, 2020). Quanto ao tema validade em modelos de simulação com dinâmica de sistemas (Barlas, 1989), é possível se afirmar que não existem modelos totalmente válidos porque todos são algo menor do que o objeto ou o sistema que está sendo modelado. A priori, é praticamente impossível provar a corretude de um modelo de simulação que visa gerar comportamentos inesperados, que nunca foram observados anteriormente (Rebs *et al.*, 2019). Realizou-se a validação do comportamento

para determinar se os padrões de comportamento gerados pelo modelo estariam próximos o suficiente dos principais padrões exibidos pelo sistema real. Durante a validação de aprendizado houve a avaliação da utilidade do modelo, visando provocar uma reflexão e a ampliação do entendimento sobre o problema.

6. ESTUDO DE CASO E ANÁLISE DOS RESULTADOS

Por razões de sigilo das informações, as empresas avaliadas no estudo de caso serão denominadas Alpha, Beta e Ômega. Cada uma dessas empresas selecionou um produto para submeter ao modelo.

A avaliação da qualidade de software, buscando um consenso comum (por exemplo, que métricas precisam ser consideradas) ainda é um problema desafiador, principalmente quando as pessoas desconhecem que existem modelos mundialmente testados e disponíveis no mercado. Muitas dessas avaliações são baseadas em critérios qualitativos, o que dificulta a medição por não terem uma exatidão numérica.

O modelo proposto teve por objetivo ser uma ferramenta para suporte à tomada de decisão referente a melhorias de software, visando o alcance de um grau de qualidade aceitável. Buscou-se avaliar, por meio de simulações, as métricas de um produto de software, tendo como referência as orientações da norma ISO/IEC 25.010, com a proposição de métodos baseados em Dinâmica de Sistemas.

O modelo proposto possui parâmetros de entrada divididos em oito grupos de características, ou categorias: funcionalidade, eficiência do desempenho, compatibilidade, confiabilidade, usabilidade, segurança, manutenibilidade e portabilidade.

Para que o modelo possa ser utilizado, o gestor precisará fornecer diversos parâmetros que caracterizam seu cenário de qualidade de software a ser alcançada. Existem parâmetros para demandas internas, externas, critérios qualitativos (nível de conhecimento, motivação dos recursos) e critérios subjetivos como taxa de abandono, rotatividade de pessoas e absenteísmo.

A Figura 6 mostra a tela de parametrização do sistema de simulação que automatizou o modelo. Os usuários deram pontuações através de conceitos que variavam de 0 a 5 com as seguintes parametrizações: 0 - sem avaliação, 1 - muito ruim, 2 - ruim, 3 - médio, 4 - bom, 5 - muito bom.

Caso um dos parâmetros não faça parte do cenário avaliado, como por exemplo: proteção de erro do usuário, bastará deixar zero na informação, que esta será desconsiderada no processo de avaliação.

O modelo proposto gera resultados que buscam simular o resultado esperado dos produtos de software em termos de qualidade em relação aos dados das entradas.

Caso 1 - Empresa Alpha

A empresa *Alpha* possui um quadro de mais de 150 funcionários realizando especificamente trabalhos de desenvolvimento de sistemas comerciais, sendo que destes, 10 são formados exclusivamente pelo time de avaliação de teste e qualidade de software.

Em relação à norma ISO/IEC 25.010, a empresa *Alpha* considera os critérios apresentados na Tabela 1 para a sua avaliação.

As informações na Tabela 1 foram reportadas pelo avaliador/gestor da área, que considerou o cenário atual da empresa *Alpha* para o reporte de informações. Ao informar estes dados no modelo, o resultado gerado com o atual cenário é apresentado no gráfico da Figura 13.

Considerando o cenário atual da empresa *Alpha*, existe uma deficiência de qualidade do software nas características de *Eficiência e Confiabilidade*.

A empresa *Alpha* deve, de forma prioritária, trabalhar aspectos relacionados como tempo comportamento, utilização de recursos, capacidade, maturidade, disponibilidade, tolerância ao erro e recuperabilidade. Isso fará com que alcance um equilíbrio em todas as características de qualidade do produto de software, conforme as orientações da norma ISO/IEC 25.010.

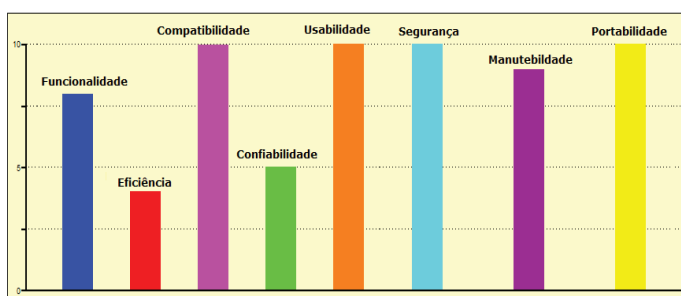


Figura 13. Cenário de avaliação da empresa Alpha

Fonte: O(s) próprio(s) autor(es)

Tabela 1. Cenário da Empresa Alpha

Parâmetros		Informação inicial do avaliador	
Recursos de TI		150	
Quantidade para testes		10	
Produto submetido		Sistema de contabilidade	
Critérios de Avaliação conforme norma ISO/IEC 25.010			
Funcionalidade			
Adequação	5	Segurança de Acesso	1
Acurácia	1	Completo funcional	1
Eficiência do desempenho			
Tempo comportamento	2	Capacidade	1
A utilização de recursos	1		
Compatibilidade			
Coexistência	5	Interoperabilidade	5
Confiabilidade			
Maturidade	1	Tolerância ao erro	2
Disponibilidade	2	Recuperabilidade	0
Usabilidade			
Apreensibilidade	5	Acessibilidade	5
Operabilidade	5	Proteção erro do usuário	5
Estética interface usuário	5		
Segurança			
Confidencialidade	5	Não-repúdio	5
Integridade	5	Prestação de contas	5
Autenticidade	5		
Manutenibilidade			
Analisabilidade	5	Modificabilidade	5
Estabilidade	5	Testabilidade	5
Portabilidade			
Adaptabilidade	5	Instalação	5
Substituição	5		

Fonte: O(s) próprio(s) autor(es)

Caso 2 - Empresa Beta

A empresa *Beta* atende clientes do segmento de logística e distribuição. Possui um quadro de 80 funcionários, com 8 destes trabalhando especificamente com qualidade de software.

As informações na Tabela 2 foram reportadas pelo avaliador/gestor da área, que considerou o cenário atual do produto avaliado da empresa Beta. Ao se informar estes dados ao modelo, foi obtido o resultado mostrado na Figura 14.

Se, por um lado, a qualidade do produto de software da empresa *Beta* apresenta um ótimo desempenho nos cri-

térios de funcionalidade, eficiência, compatibilidade, confiabilidade e segurança, existe uma deficiência em termos de usabilidade, maneabilidade e portabilidade. A acessibilidade, proteção de erro do usuário, estética da interface de usuário, estabilidade testabilidade e adaptabilidade são algumas das subcaracterísticas que necessitam ser melhoradas para diminuir essa considerável diferença entre as características de qualidade do software.

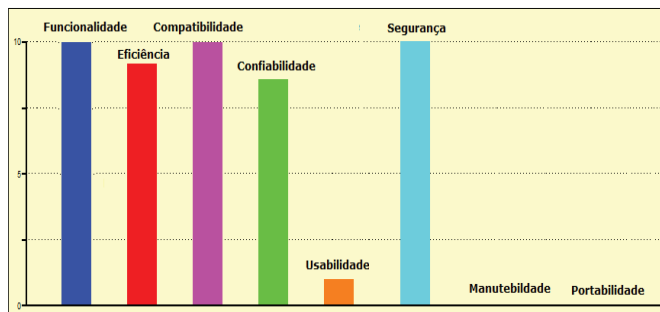


Figura 14. Cenário de avaliação da empresa Beta

Fonte: O(s) próprio(s) autor(es)

Tabela 2. Cenário da Empresa Beta

Parâmetros		Informação inicial do avaliador	
Recursos de TI		80	
Quantidade para testes		08	
Produto submetido		Sistema de contabilidade	
Critérios de Avaliação conforme norma ISO/IEC 25.010			
Funcionalidade			
Adequação	5	Segurança de Acesso	5
Acurácia	0	Completo funcional	5
Eficiência do desempenho			
Tempo comportamento	4	Capacidade	3
A utilização de recursos	5		
Compatibilidade			
Coexistência	5	Interoperabilidade	5
Confiabilidade			
Maturidade	5	Tolerância ao erro	4
Disponibilidade	4	Recuperabilidade	4
Usabilidade			
Apreensibilidade	1	Acessibilidade	1
Operabilidade	1	Proteção erro do usuário	2
Estética interface usuário	2		
Segurança			
Confidencialidade	5	Não-repúdio	5
Integridade	5	Prestação de contas	5
Autenticidade	5		
Manutenibilidade			
Analisabilidade	0	Modificabilidade	0

Estabilidade	0	Testabilidade	0
Portabilidade			
Adaptabilidade	0	Instalação	0
Substituição	0		

Fonte: O(s) próprio(s) autor(es)

Caso 3 - Empresa Ômega

A empresa Ômega possui médio porte, com um quadro superior a 200 funcionários, dos quais 20 desses estão lotados na área de TI, sendo 3 da área de testes e qualidade. A empresa possui softwares próprios e terceirizados. O seu sistema de ERP é terceirizado. Alguns produtos, como o software de atendimento ao cliente, avaliado nesta pesquisa, são de desenvolvimento próprio.

As informações na Tabela 3 foram reportadas pelo avaliador/gestor da área, que considerou o cenário atual para avaliação da qualidade do produto de software. Ao contrário das empresas Alpha e Beta, é importante salientar que TI não é a atividade fim da empresa Ômega, fato que foi considerado na análise de resultados. Ao informar os dados da empresa no modelo, o resultado do cenário atual de qualidade do produto.

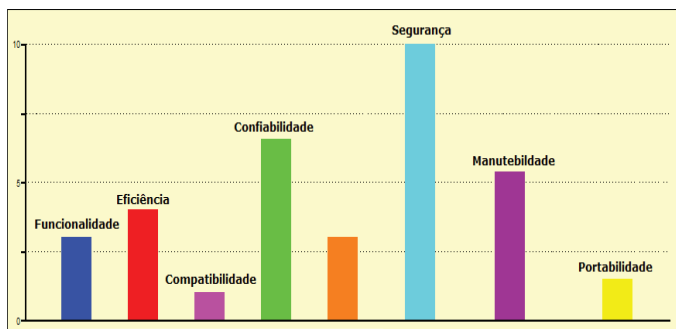


Figura 15. Cenário de avaliação da empresa Ômega

Fonte: O(s) próprio(s) autor(es)

Conforme indicado na Figura 15, com exceção da característica de segurança, a empresa Ômega deverá trabalhar em todas as outras sete características de qualidade para produto de software orientadas pela norma ISO/IEC 25.010.

Neste cenário, indica-se a revisão do processo de software. Neste caso, a adoção de modelos como o *Capability Maturity Model – Integration (CMMI)* e o *Melhoria do Processo de Software Brasileiro (MPS.BR)* poderão influenciar consideravelmente os resultados em uma próxima simulação.

Tabela 3. Cenário da Empresa Ômega

Parâmetros	Informação inicial do avaliador		
Recursos de TI		80	
Quantidade para testes		08	
Produto submetido		Sistema de contabilidade	
Critérios de Avaliação conforme norma ISO/IEC 25.010			
Funcionalidade			
Adequação	5	Segurança de Acesso	0
Acurácia	0	Completeness funcional	0
Eficiência do desempenho			
Tempo comportamento	4	Capacidade	3
A utilização de recursos	5		
Compatibilidade			
Coexistência	5	Interoperabilidade	5
Confiabilidade			
Maturidade	5	Tolerância ao erro	4
Disponibilidade	4	Recuperabilidade	4
Usabilidade			
Apreensibilidade	1	Acessibilidade	1
Operabilidade	1	Proteção erro do usuário	2
Estética interface usuário	2		
Segurança			
Confidencialidade	5	Não-repúdio	5
Integridade	5	Prestação de contas	5
Autenticidade	5		
Manutenibilidade			
Analisabilidade	2	Modificabilidade	2
Estabilidade	3	Testabilidade	2
Portabilidade			
Adaptabilidade	2	Instalação	0
Substituição	1		

Fonte: O(s) próprio(s) autor(es)

7. CONCLUSÃO

A presente pesquisa envolveu a triangulação entre revisão da literatura, observação e estudo de caso em três empresas reais. Um modelo de simulação baseado em dinâmica de sistemas foi proposto como solução para lidar com a complexidade inerente ao processo de avaliação de qualidade de produto de software, visando aperfeiçoar o entendimento dos problemas envolvidos e gerar pontos de reflexão que possam aumentar a eficácia da gestão. O modelo apresentado possibilitou a simulação de cenários, visando suportar a avaliação de produtos de software em empresas.

Para testes do modelo, realizou-se um estudo de caso aplicado nas empresas *Alpha*, *Beta* e *Ômega*.

Durante o estudo de caso foi considerada a demanda de trabalho, alterações, testes e comportamento do sistema nos últimos 6 meses. O objetivo foi avaliar as características do produto, tendo como referência de métricas a norma ISO/IEC 25.010. A equipe de pesquisa teve acesso aos registros e relatórios que mostravam a quantidade de chamados relacionados à manutenção dos produtos. Na validação do comportamento, foi avaliado se os padrões de comportamento gerados pelo modelo foram próximos o suficiente dos principais padrões exibidos pelo sistema real. Os seguintes aspectos foram avaliados: replicação de comportamento, adequação visual e sensibilidade dos atributos de saída às mudanças nas variáveis de entrada.

Os testes de sensibilidade das variáveis de entrada e de calibragem do modelo proposto foram executados em três momentos distintos: a) durante a execução dos testes estruturais, com o objetivo de corrigir equações; b) durante os testes de comportamento, com o objetivo de eliminar parâmetros de pouca relevância e calibrar corretamente o modelo e; c) durante a execução de testes de aprendizado, com o objetivo de gerar incerteza nas entradas e analisar riscos. Os resultados obtidos com os testes de comportamento foram apresentados aos usuários e subsidiaram o refinamento do modelo.

Para utilização, manutenção e avaliação da eficácia do modelo, foram realizadas atividades na fase final desta pesquisa, através de sessões de utilização do modelo com os gestores. A síntese de estratégias utilizadas para verificação e validação do modelo, executadas durante esta pesquisa, é apresentada na Tabela 4.

O processo de validação do modelo incluiu entrevistas com dez gestores que trabalham com serviços e qualidade de software e, adicionalmente, 11 gestores de áreas de TI. O exercício incluiu a apresentação do modelo e seus resultados. As três hipóteses analisadas foram relativas à utilidade, preferência e eficácia do modelo proposto. O exercício de validade de aparência (Runerson e Host, 2009) realizado refutou as hipóteses negativas. Para a realização do teste de hipóteses, foi utilizada inferência estatística (Berger, 1987; Casella, 2002), através da execução de um teste estatístico binomial Berger (1987) com 5% de significância, como mostra a Tabela 5.

Como primeira ameaça à validade da pesquisa, aponta-se para restrições do ponto de vista orçamentário e de logística. Em relação à validade de construção, sempre há dúvidas de que as variáveis são bem compreendidas pelos gestores podendo não se obter resultados de simulação que correspondam à realidade. Foi possível observar durante o experimento que envolveu as simulações realizadas, que as empresas primam mais pelos aspectos e características mais relacionados à funcionalidade, eficiência e segurança.

Tabela 4. Estratégias para Verificação e Validação

Verificação da implementação do modelo	Avaliar se a implementação do modelo estava livre de erros	Estratégia adotada
Consistência dimensional	Dimensões das variáveis estão consistentes e as unidades estão corretas	Testes do modelo e implementação em software
Validação da Sintaxe	Comportamento - as estimativas de gestão do modelo estão livres de erros de sintaxe	Testes do modelo e implementação em software
Validação semântica	Comportamento - as estimativas de gestão do modelo estão livres de erros semânticos	Testes do modelo e implementação em software
Validação conceitual do modelo	Avaliar a estrutura do modelo	Testes do modelo e implementação em software
Confirmação da estrutura	As estimativas do modelo correspondem às relações reais do processo	Revisão de literatura, observação e validade de aparência Runerson (2009)
Confirmação dos parâmetros	Avaliação dos parâmetros do modelo em função do conhecimento do processo	Revisão de literatura, observação e validade de aparência Runerson (2009)
Condições extremas	Avaliar o comportamento do modelo sobre condições extremas	Revisão de literatura, observação e validade de aparência Runerson (2009)
Sensibilidade do comportamento	Identificação dos parâmetros de processos nos quais o modelo é altamente sensível	Revisão de literatura, observação e validade de aparência Runerson (2009)

Fonte: O(s) próprio(s) autor(es)

Tabela 5. Hipóteses para Testar a Validade de Aparência do Modelo.

Hipóteses	% quem acredita	Existe evidência estatística suficiente para aceitar a hipótese?
Preferência: os gestores preferiram o modelo apresentado em relação à forma atual utilizada.	90	sim
Utilidade: Os gestores consideraram o modelo útil.	100	sim
Eficácia: Os gestores consideraram o modelo eficaz para suporte à tomada de decisão	92	sim

Fonte: O(s) próprio(s) autor(es)

Entre as limitações da pesquisa, pode-se destacar que houve um acesso difícil a empresas e profissionais que conhecessem a norma ISO/IEC 25.010 ou que, ao menos, já tivessem algum tipo de ciência sobre ela. Percebeu-se durante a fase de observação um conhecimento um pouco vago de alguns envolvidos referente aos modelos de processo de desenvolvimento CMMI e MPS.BR. Destaca-se também a percepção dos usuários e avaliadores em relação às métricas de avaliação. Alguns gestores/avaliadores, quando questionados sobre quais métricas poderiam ser aplicadas para avaliar a qualidade do produto de software, demonstraram incerteza, por conta da limitação de seus testes à recuperação de registros do banco de dados e à precisão de cálculos.

Como trabalhos futuros, pretende-se estender o modelo para realizar uma sugestão de pontos, características e subcaracterísticas que possam ser utilizados para incremento da qualidade, visando maior satisfação dos usuários.

REFERÊNCIAS

Office of Government Commerce – OCG, (2007), ITIL Vol. 3 (Information Technology Infrastructure Library)/Service Operation, London: TSO.

COBIT 5, (2014), “Modelo corporativo para governança e gestão de TI da organização”, edição em português, São Paulo, Chapter with the permission of ISACA.

Guerra, A. C.; Colombo, R. M. T. (2009), “Qualidade de produto de software”, *PBQP Software*, Rio de Janeiro.

Bartolini C.; Stefanelli, C. (2011), “Business-driven IT management, proceedings of IFIP/IEEE international symposium on integrated network management (IM), Vol. 1, pp. 963–969.

ABNT NBR ISO/IEC 9000:2015. (2015), Rio de Janeiro: ABNT. Subcomitê de software da ABNT (SC-21:10), <http://docplayer.com.br/12433555-Subcomite-de-software-da-abnt-sc-21-10.html> (acesso em set. 2016).

html (acesso em set. 2016).

da Silva, M. C., (2016), “CMMI para iniciantes”, <http://www.linhadecodigo.com.br/artigo/1401/cmmi-para-iniciantes.aspx> (acesso em set. 2016).

PROMOVE, “Como usar o MPS.Br”, (2016), <http://promovesolucoes.com/mps> (acesso em set. 2016).

Mecenas, I.; de Oliveira, V. (2005), “Qualidade em software - uma metodologia para homologação de sistemas”, Editora Alta Books, Rio de Janeiro.

Sterman, J. (2000), “Business dynamics: systems thinking and modeling for a complex world”, Irvin-McGraw-Hill, Chicago, IL.

Rodrigues, A.; Bowers, J. (1996), “The role of system dynamics in project management”, *International Journal of Project Management*, Vol. 14, pp. 213-220.

BPM CBOK Versão 3.0. (2013), 1ª ed. em português. Association of Business Process Management Professional, ABPMP, São Paulo.

Fernandes, A. A.; Abreu, V. F. (2012), *Implantando a governança de TI – da estratégia à gestão dos processos e serviços*, 3ª edição, Brasport, Rio de Janeiro:.

ISO/IEC 25010 (2011). <https://www.iso.org/obp/ui/#iso:std:iso-iec:25010:ed-1:v1:en> (acesso em set. 2016).

Lahtela, A.; Jantti, M.; Kaukola, J. (2010), “Implementing an ITIL - based IT service management measurement system”, *Proceedings of fourth international conference on digital society*, St. Maarten, pp. 249-254.

Bezerra, T. R.; Moura, J. A. B.; Lima, A. S. (2014), “A system dynamics model to support strategic decision making on IT outsourcing: a case study at a state revenue agency in Brazil”, *Proceedings of NOMS 2014 IEEE/IFIP Network Operations and Management Symposium*, Krakow. 2014 IEEE Network Operations and Management Symposium (NOMS). pp. 1-4.

de Oliveira, J. A. N. (2010), Um modelo formal para avaliar o valor de negócio e sua aplicação no contexto de gestão e governança de TI, Tese de Doutorado, Universidade Federal de Campina Grande, PB.

Lima, A. S.; Sauv e, J.; de Souza, J. N. (2012), “Capturing the quality and business value of IT services using a business-driven model”, *IEEE transactions on network and service management*.

Fernandes, A. C. (2003), “Scorecard Dinâmico - Em direção à integração da dinâmica de sistemas com o balanced scorecard”, Tese, Universidade Federal do Rio de Janeiro, UFRJ, RJ.

Morecroft, J. (2008), “Strategic modeling and business dynamics – a feedback systems approach”, Wiley, New Delhi, India.

Wan, J.; Zhang, H.; Wan, D. (2011), “Evaluation on information technology service management process with AHP”, *Technology and Investment*, Vol. 2, No. 1, pp. 38–46.

- Project Management Institute - PMI. (2012), "A guide to the project management body of knowledge – PMBOK guide", Pennsylvania, USA.
- Runerson, P., Host, M. (2009), "Guidelines for conducting and reporting case study research in software engineering", Vol. 14, pp.31-164, Springer: Empiric Software Eng., DOI 10.1007.
- Lima, A. S.; de Souza, J. N.; Moura, J. A. B.; da Silva, I. P., (2018), "A consensus-based multicriteria group decision model for information technology management committees", *IEEE Transactions on Engineering Management*, Vol. 65, pp. 1-17.
- de Mello, R. F., (2010), "Sistemas dinâmicos e técnicas inteligentes para a previsão de comportamento de processos: uma abordagem para otimização de escalonamento em grades computacionais", 129p, Tese de Livre Docência, Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo, USP, São Carlos, SP
- Isee Systems, (2016), "Stella software". Disponível em <https://www.iseesystems.com/store/products/stella-architect.aspx> (acesso em 17 ago. 2016).
- Barlas, Y. (1989), "Multiple tests for validation of system dynamics type of simulation models", *European Journal of Operational Research*, Vol. 42, No. 1, pp. 59-87.
- Macachy, R. J. (2007), "Software process dynamics", John Wiley & Sons.
- Abdel-Hamid, T. K. (1989), "The dynamics of software project staffing: a system dynamic based simulation approach", *IEEE Transactions on Software Engineering*, Vol. 15, No. 2, pp. 109-119.
- Yin, R. K. (2005), "Estudo de caso: planejamento e métodos", Bookman, São Paulo, SP.
- Berger, J. O.; Delampady, M. (1987), "Testing precise hypotheses", *Statistical Science 2*.
- Casella, G.; Berger, R. L. (2002), "Statistical inference", 2nd ed., Duxbury Advanced Series, California.
- Sousa, A. B.; Lima, A. S.; de Souza, J. N.; Moura, J. A. B.; da Silva, I. P. (2018), "Business risk-based redundancy points identification in synchronous digital hierarchy optical networks", *IEEE Latin America Transactions*, Vol. 16, No. 8, pp. 2254-2260.
- Nóbrega, D. A.; Fenner, G.; Lima, A. S. (2014), "A risk management methodology proposal for information technology projects", *IEEE Latin America Transactions*, Vol. 12, No. 5, pp. 643-656.
- Rebs, T.; Brandenburg, M.; Seuring, S. (2019), "System dynamics modeling for sustainable supply chain management: a literature review and systems thinking approach", *Journal of Cleaner Production*, Vol. 208, pp. 1265-1280.
- Radzicki, M. J. (2020), "System dynamics and its contribution to economics and economic modeling, system dynamics: theory and applications", Springer.

Recebido: 22 abr. 2020

Aprovado: 09 mar. 2021

DOI: 10.20985/1980-5160.2021.v16n1.1625

Como citar: Fenner, G., Lima, A.S., Monteiro Filho, J.M.S., Rocha, L.S., Moura, J.A.B., Souza, J.S. (2021). Um modelo para suporte ao gerenciamento de qualidade de software com base na dinâmica de sistemas. *Revista S&G* 16, 1, 19-33. <https://revistasg.emnuvens.com.br/sg/article/view/1625>