



## IMPACTO DA GESTÃO SISTÊMICA NO DESENVOLVIMENTO DE PROJETOS: UM ESTUDO EM UMA *SOFTWARE HOUSE*

Jefferson de Souza Pinto<sup>1</sup>, Daniel de Azevedo Figueiredo<sup>1</sup>, Rafael Barbosa Lucena Cruz<sup>1</sup>,  
Tiago Vinícius de Oliveira Brachini<sup>1</sup>, Poliana de Jesus Oliveira<sup>1</sup>

1 Instituto Federal de Educação, Ciência e Tecnologia de São Paulo

### RESUMO

O mercado do século XXI exige das empresas a capacidade de se autogerir para atender de forma eficiente e eficaz as necessidades de cada cliente; assim, para obter estes resultados, notou-se a importância de minimizar esforços e eliminar o retrabalho. Nesse contexto, se faz necessária uma visão abrangente que reflita o cenário globalizado e revolução tecnológica, como a Teoria Geral dos Sistemas, que surgiu, no início do século XX, como uma forma de integração entre as diversas áreas do conhecimento, gerando subsídios para um modelo de gestão que permite ao gestor uma visão holística do negócio, com enfoque nas inter-relações entre os elementos da organização. A partir do exposto, o trabalho objetiva analisar a aplicação da Teoria Sistêmica em uma empresa de *Software House* na região bragantina. Para desenvolver o trabalho, utilizou-se o método de estudo de caso e pesquisa bibliográfica e, no levantamento dos dados, foi realizada a observação participante, tendo o desenvolvimento de projetos da empresa como objeto de estudo. A partir da análise do caso é possível verificar, como benefício do modelo de gestão sistêmica, uma tendência a maior efetividade na resolução de conflitos, bem como estímulos a ganhos de produtividade, decorrente da percepção holística dos processos.

**Palavras-chaves:** Gestão Sistêmica; Sistemas; Desenvolvimento de Projetos; Software House.

### 1. INTRODUÇÃO

A complexidade da organização é resultante das forças e pressões que ela sofre do seu ambiente (Donaires, 2006). Portanto, o cenário atual exige que os gestores considerem o problema da globalização na tomada de decisões.

Globalização significa combinar o comportamento das grandes civilizações e dissolver suas fronteiras de maneira historicamente inédita (Shimizu, 2011). O resultado desse processo está na criação de problemas complexos, que exigem uma forma de pensamento específica. Pois, segundo Senge *et* Sterman (1994) *apud* Andrade (1997), uma nova forma de pensamento deveria ajudar a mapear, desafiar e melhorar os modelos mentais, visando ações mais efetivas na realidade organizacional. No entanto, existe uma expectativa de que a realidade siga um modelo ordenado, que

apresente regularidade e repetitividade, como as típicas de uma máquina. Essa expectativa é uma consequência da formação cartesiana dos profissionais e do legado da administração clássica que permeia o pensamento gerencial (Donaires, 2009). Nesse contexto, se faz necessária uma visão abrangente que gere subsídios para um modelo de gestão que permita uma visão holística do negócio.

Onde encontrar, porém, tal paradigma que permita lidar com os problemas profissionais inseridos no ambiente complexo e dinâmico do mercado atual? De acordo com Martinelli (2002), os gerentes práticos frequentemente são confrontados com desafios que requerem novas maneiras de pensar e pedem ajuda ao mundo acadêmico para que este lhes forneça um 'par de lentes sistêmicas', com uma



perspectiva geral que seja genérica, abrangente, holística e concreta, capaz de descrever a atuação administrativa. Assim, baseada na Teoria Geral dos Sistemas, que surgiu, no início do século XX, como uma forma de integração entre as diversas áreas do conhecimento, a visão sistêmica (ou perspectiva sistêmica) passou a caracterizar numerosas pesquisas no campo da administração, principalmente na busca de diretrizes e metodologias para a chamada administração sistêmico-evolutiva, integrada, holística (Martinielli, 2006).

A fim de apresentar esta realidade no desenvolvimento de *software*, deve-se notar que a complexidade do *software* é fruto da complexidade do processo que o gerou, que por sua vez é um reflexo da complexidade da organização. Organizações que se propõem a desenvolver *software* num mercado globalizado, variado ou fragmentado são elas mesmas organizações complexas. Isso quer dizer que tais organizações não podem ser compreendidas por meio da ótica simplificadora do mecanicismo (Donaires, 2006). De forma que, a partir do exposto, este trabalho objetiva analisar a aplicação da teoria sistêmica no desenvolvimento de projetos de uma *Software House* na Região Bragantina. A ferramenta utilizada no estudo de caso para a implantação da gestão sistêmica foi o Método Ágil *Scrum*.

Este trabalho visa demonstrar as consequências oriundas da prática do pensamento sistêmico, descrevendo os seus princípios teóricos e contextualizando-os, a partir da análise do caso, à realidade da empresa objeto de estudo. Tais consequências incluem benefícios, como maior efetividade na resolução de conflitos e e como estímulos a ganhos de produtividade decorrente da percepção holística dos processos.

## 2. REVISÃO BIBLIOGRÁFICA

Nesta seção é apresentada a revisão da literatura por meio da análise de referenciais teóricos publicados em artigos, livros, sites e teses, com o intuito de verificar o estado da arte sobre o tema do trabalho.

### 2.1. Definição de sistema

Existem muitas definições para sistema na literatura. Ribeiro (2004) define que é um conjunto de elementos, interagentes e interdependentes, cada qual com sua função específica, que trabalha em sintonia para atingir determinado objetivo comum; ao passo que Faria (1980) define sistema como um superfluxo integrador dos diversos fluxos setoriais que resultam nas atividades dos órgãos especializados, responsáveis pelo desempenho das funções típicas de um organismo. Um grande sistema coordena fluxo e os subsiste-

mas comportam fluxos específicos. O sistema é, enfim, uma organização em ação. Ele afirma que esse conceito é subentendido em decorrência das seguintes propriedades:

- a) Aspectos: um sistema é um conjunto de elementos em interação; um conjunto de objetos e de relações entre os objetos e entre seus atributos; e, por fim, um sistema é, também, um todo organizado composto de muitas partes, um conjunto de atributos;
- b) Critérios e características: um sistema precisa ser definível no sentido de poder situar-se com certa precisão no tempo e no espaço; um sistema é considerado como tal, nos casos em que uma variedade de operações executadas preferivelmente por várias disciplinas (órgão) leva à conclusão que existe um sistema específico; um sistema tem de manifestar diferenças significativas nas escalas de estrutura e processo; e,
- c) Fundamentos existentes no conceito de sistema: os objetivos totais do sistema, e mais especificamente, as medidas de rendimento do sistema inteiro; o ambiente do sistema e as coações fixas; os recursos e os componentes do sistema, suas atividades, finalidades e medidas de rendimento e também sua administração.

Por fim, Faria (2002) conceitua o sistema como sendo complexo, e para compreendê-lo, deve-se conhecer suas características, seus tipos, parâmetros, entre outros aspectos.

### 2.2. Teoria Geral de Sistemas

A administração é uma das ciências que mais utiliza o enfoque da Teoria Sistêmica (Faria, 2002). O autor esclarece que a Teoria Geral de Sistemas estabeleceu uma dependência recíproca e a necessidade de sua integração, e que, desde então, todos os ramos do conhecimento passaram a tratar seus estudos - quaisquer que fossem as áreas - como sistemas.

A partir do surgimento dessa teoria por meio dos trabalhos de Ludwing Von Bertalanffy, Faria (2002) menciona que ele teve a preocupação de apresentar a teoria e conceitos, com estabelecimentos de condições reais e atuais:

- a) Fundamentos estabelecidos: existem subsistemas; os sistemas são abertos, e existem muitas interações externas; as funções dependem da estrutura desses sistemas;
- b) Características principais: conjunto organizado e complexo; conjunto de unidades inter-relacionadas; todo sistema tem um objetivo; todo sistema afetado



em uma das suas partes será afetado nas demais; há um equilíbrio entre as partes de um sistema.

Faria (2002) evidencia que sistemas são caracterizados por seus parâmetros, conforme a Figura 1. Tais parâmetros são: entrada ou insumo (*input*); processamento (*throughput*); saída ou resultado (*output*); retroação ou retroalimentação (*feedback*); ambiente. É importante reconhecer também os tipos de sistema existentes, que pode ser:

- Concreto - estrutura física da empresa, podendo-se medir quantitativamente seu desempenho;
- Abstrato - estruturado pelos conceitos, hipóteses;
- Aberto - caracterizado pela grande interação com o meio; e,
- Fechado - caracterizado aos processos limitado a uma entrada.

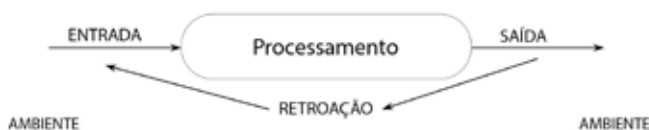


Figura 1. Parâmetros do Sistema.

Fonte: Faria (2002, p.127).

Na aplicabilidade, Ribeiro (2004) concluiu que a teoria de sistemas desempenha papel decisivo na ciência de nosso tempo, pois permite a integração de conhecimento das ciências físicas, biológicas e humanas. Especificamente na administração, Bouding (1956) *apud* Wetherbe (1987) afirma que:

A abordagem sistêmica é a maneira como pensar sobre o trabalho de gerenciar. Ela fornece uma estrutura para visualizar fatores ambientais internos e externos como um todo integrado. Ela permite o reconhecimento da função dos subsistemas, bem como dos supra-sistemas complexos dentro dos quais as organizações têm que operar. Os conceitos sistêmicos criam uma maneira de pensar, a qual, de um lado, ajuda o gerente a reconhecer a natureza de problemas complexos e, por isso, ajuda a operar dentro do ambiente percebido.

Ribeiro (2004) afirma que a teoria sistêmica teve o benefício de apresentar a organização funcionando em partes e ao mesmo tempo em interação com o meio ambiente, conforme apresentado na Figura 2.



Figura 2. Interação organização/ambiente.

Fonte: Ribeiro (2004, p. 108).

Por fim, Wetherbe (1987) conclui que:

A abordagem sistêmica é uma maneira de pensar quando analisando ou gerenciando sistemas. Ela não está interessada apenas nas partes de um sistema; ao contrário, considera o efeito total criado quando as partes funcionam como um todo – o próprio sistema. O projeto de sistemas consiste em estabelecer os objetivos dos sistemas, selecionar para inclusão aquelas entidades que possuam atributos que possam contribuir para os objetivos dos sistemas e estruturar adequadamente as entidades incluídas no sistema.

Dessa forma, nos últimos 50 anos, um amplo conjunto de metodologias sistêmicas foi desenvolvido com o objetivo de resolver com problemas mal estruturados, algumas das quais podem ser usadas para abordar problemas comuns em administração, em especial nas questões ligadas às atividades de negociação no dia a dia das pessoas nas organizações (Martinelli, 2006).

### 2.3. Gerenciamento de projetos

Terribili Filho (2011, p. 40) define projeto como qualquer esforço para a criação de um produto ou serviço, com início e fim, utilizando-se de prazos para conclusão de suas etapas e envolvendo orçamentos relativos a recursos humanos e logísticos. Com a conclusão de todas as etapas, feitas com base em requisitos preestabelecidos, tem-se a “saída” entregável, faltando somente aprovação final do seu patrocinador ou cliente. Essa definição reforça as características do gerenciamento de projetos, apresentadas por Svejvig et Andersen (2015), as quais são: executabilidade, simplicidade, temporariedade, linearidade, controlabilidade e instrumentalidade.

Por meio de projetos, é possível construir a combinação de recursos organizacionais necessários, a fim de proporcionar uma capacidade de desempenho elevada na criação e execução de estratégias organizacionais (Cleland et Ireland, 2007).

Por sua vez, o gerenciamento de projetos tem por base a realização das entregas planejadas com qualidade e no tempo esperado sem estourar o orçamento (Terribili Filho, 2011). Sua implementação é feita por meio de procedimentos que consistem nas funções fundamentais de administração, utilizando-se recursos para alcançar os objetivos do projeto (Cleland et Ireland, 2007).



De acordo com Faria (1980), é possível constituir um projeto a partir da abordagem sistêmica, verificando de forma racional a estrutura organizacional disponível e os recursos que existem para a efetivação de um objetivo. Para isso é necessário:

- a) Prescrever um sistema que leve funcionalidade à organização de um processo para a solução geral de problemas;
- b) Estipular um sistema de parâmetros que proporcione o formato necessário para a solução dos problemas; e,
- c) Descrever modelos e propriedades que forneçam os meios necessários para a iteração de saídas alternativas no processo de solução dos problemas.

Assim, Faria (1980) afirma que a concepção sistêmica de organização difere de métodos clássicos por buscar soluções globais, estabelecendo a interação funcional entre os órgãos especializados e procurando anular a tendência de compartimentalização de microestruturas dos órgãos, que tendem a funcionar buscando preservar os objetivos setoriais em detrimento do propósito final. De acordo com Castro (2000), o sucesso do sistema organizacional depende da otimização coordenada de suas dimensões sociais, econômicas e tecnológicas, conforme a Figura 3.



Figura 3. A empresa como um sistema sociotécnico.

Fonte: Castro (2000, p. 37).

## 2.4. Scrum

Scrum é um modelo ágil de gestão de projetos de *software*. Baseia-se no desenvolvimento incremental das aplicações, centrado na equipe, permitindo um maior controle

do processo pelo fato de ter ciclos de iteração muito curtos. Seus processos de baseiam nas metodologias ágeis de desenvolvimento, caracterizando-se como uma forma “não tradicional” de desenvolver *software*, em que a prioridade máxima se centra na satisfação do cliente através da entrega antecipada e contínua de *software* suscetível de avaliação (Ferreira *et al.*, 2005; Sabbagh, 2014; Prikladnicki, Willi, Milani, 2014).

Com o intuito de padronizar as práticas de desenvolvimento de *software*, em fevereiro de 2001, um grupo de representantes das metodologias de desenvolvimento ágil se reuniu e criou o Manifesto Ágil, cujos doze princípios, de acordo com Beck *et al.* (2001), são:

- a) A maior prioridade é satisfazer o cliente por meio da entrega contínua e adiantada de *software* com valor agregado;
- b) Mudanças nos requisitos são bem-vindas, mesmo tardiamente no desenvolvimento;
- c) Processos ágeis tiram vantagem das mudanças visando vantagem competitiva para o cliente;
- d) Entregar frequentemente *software* funcionando, de poucas semanas a poucos meses, com preferência à menor escala de tempo;
- e) Pessoas de negócio e desenvolvedores devem trabalhar diariamente em conjunto por todo o projeto;
- f) É preferível a construção de projetos em torno de indivíduos motivados;
- g) É preciso dar aos colaboradores o ambiente e o suporte necessário e confiar neles para fazer o trabalho;
- h) O método mais eficiente e eficaz de transmitir informações para e entre uma equipe de desenvolvimento é por meio de conversa face a face;
- i) *Software* funcionando é a medida primária de progresso;
- j) Os processos ágeis promovem desenvolvimento sustentável. Os patrocinadores, desenvolvedores e usuários devem ser capazes de manter um ritmo constante indefinidamente;
- k) Contínua atenção à excelência técnica e bom *design* aumentam a agilidade;
- l) Simplicidade - a arte de maximizar a quantidade de trabalho não realizado - é essencial;



- m) As melhores arquiteturas, requisitos e *designs* emergem de equipes auto-organizáveis; e,
- n) Em intervalos regulares, a equipe reflete sobre como se tornar mais eficaz e então refina e ajusta seu comportamento de acordo.

Apesar do manifesto definir novos valores ao desenvolvimento de *software*, suas antigas diretrizes não foram descartadas. No entanto, seu foco foi alterado para suprir novas necessidades observadas dentro do desenvolvimento. Assim, conforme Beck *et al.* (2001), é possível valorizar:

- a) Indivíduos e interações mais que processos e ferramentas;
- b) Software em funcionamento mais que documentação abrangente;
- c) Colaboração com o cliente mais que negociação de contratos; e,
- d) Responder a mudanças mais que seguir um plano.

Focada inicialmente apenas no desenvolvimento de *softwares*, essa metodologia se estendeu ao gerenciamento de projetos e à cultura organizacional. Em decorrência disso, as técnicas do *Scrum* trouxeram assertividade e integração às empresas, o que corrobora com Santos Filho (2012).

E, ainda, para Serrador et Pinto (2015), os métodos ágeis são projetados para usar um mínimo de documentação com o objetivo de facilitar a flexibilidade e a capacidade de resposta às mudanças, o que implica em menos planejamento e mais flexibilidade em projetos ágeis comparado com projetos tradicionais de gestão.

### 3. MÉTODO E OBJETO DE ESTUDO

O procedimento metodológico adotado para realização deste trabalho foi o estudo de caso. Em um primeiro momento, realizou-se uma pesquisa bibliográfica com o propósito de fundamentar e contextualizar o tema. De acordo com Cervo *et al.* (2007):

A pesquisa bibliográfica procura explicar um problema a partir de referências teóricas publicadas em artigos, livros, dissertações e teses. Pode ser realizada independentemente ou como parte da pesquisa descritiva ou experimental. Em ambos os casos, busca-se conhecer e analisar as contribuições culturais ou científicas do passado sobre determinados assuntos, tema ou problema.

Em seguida, por meio de pesquisa de campo, o presente trabalho observou, registrou, analisou e correlacionou fatos, descobrindo suas relações e conexões, sua natureza e suas características (Cervo *et al.*, 2007). O intuito foi obter uma avaliação empírica dos assuntos abordados na revisão bibliográfica, e também propiciar subsídios para inferências acerca do tema.

O estudo de caso ocorreu em uma *Software House* que atua há cinco anos no mercado, situada na cidade de Bragança Paulista, estado de São Paulo, e cuja finalidade é a produção de aplicações customizadas para plataforma *Web*, *Desktop* e *Mobile*. O objeto de estudo foi o desenvolvimento de projetos da própria empresa.

A *Software House* cresceu muito no último ano, o seu quadro de funcionários que era de apenas três colaboradores passou para sete, sendo um diretor geral, um diretor financeiro, um diretor de projetos, um colaborador para o atendimento aos clientes e três programadores. Com este rápido crescimento, a empresa não dispôs de tempo para adaptar-se à nova estrutura organizacional. Em razão de os projetos em andamento não poderem ser paralisados e aos programadores estarem atarefados, não foi possível implementar uma metodologia de desenvolvimento testada e aprovada pelo mercado, que fosse capaz de fazer a gestão do projeto, da equipe de desenvolvimento e do atendimento ao cliente, e que fornecesse informações para que o diretor de projetos e o diretor geral tomassem decisões com maior precisão e em tempo mais curto.

Em face destas condições, a empresa necessita de uma perspectiva dinâmica que lhe permita lidar com as mudanças. Dessa forma, o uso da abordagem sistêmica vem ao encontro das necessidades atuais da empresa. A ferramenta utilizada no estudo de caso para a implantação da gestão sistêmica foi o Método Ágil *Scrum*, que é bastante objetivo, com papéis bem definidos, de fácil adaptação e sua curva de aprendizado é relativamente baixa. Esse método permite aos seus praticantes saber exatamente o que está acontecendo ao longo do projeto e fazer os devidos ajustes para manter o projeto se movendo ao longo do tempo no intuito de alcançar os seus objetivos (Schwaber, 2004 *apud* Pereira *et al.*, 2007).

Para o levantamento de dados, utilizou-se a observação direta intensiva, caracterizando-se, no caso, como observação participante. Conforme Marconi *et* Lakatos (2010), a observação participativa consiste na participação real do pesquisador com a comunidade ou grupo, incorporando-se a ele e participando das atividades normais. No caso, a forma de observação participante utilizada é a natural, pois o observador pertence à mesma comunidade ou grupo que investiga.



A observação do processo de desenvolvimento de *software* foi realizada pelo diretor de projetos, que é responsável pela elaboração do projeto em cima dos dados levantados pelo diretor geral; por definir a estrutura dos módulos da aplicação; pelo controle da execução do projeto; e por estabelecer os ajustes a serem feitos após o *feedback* do cliente.

Para um melhor controle dos dados obtidos durante a observação, realizaram-se relatórios semanais sobre as atividades desenvolvidas no projeto, a fim de documentar todo o processo de coleta de dados. Cabe notar que, não sendo possível acompanhar todo o projeto, devido ao tempo de desenvolvimento ser maior que o prazo para entrega deste artigo, as conclusões terão como base apenas um recorte do projeto.

#### 4. ANÁLISE DE DADOS

Para alcançar os objetivos propostos, a análise de dados foi estruturada da seguinte forma: primeiro, se apresenta a observação da situação atual da empresa objeto de estudo, no que diz respeito a sua estrutura e ao modelo administrativo. Procede-se, a seguir, o levantamento de possíveis falhas nos processos de criação de *softwares* e, por fim, é realizada a proposta da aplicação do *Scrum* como método de gerenciamento, bem como os primeiros resultados da aplicação desta proposta na empresa objeto de estudo.

##### 4.1. Situação atual da *Software House*

Depois de um crescimento em seu quadro de funcionários, a empresa encontra dificuldades para prosseguir com seus projetos. Um ano antes, operava com apenas um desenvolvedor - o qual executava um projeto de cada vez - e o diretor da empresa; posteriormente, com um aumento de aproximadamente 300% em seu quadro de funcionários, a *Software House* precisa se adequar à nova realidade e começar a implantar regras e métodos definidos para controlar seus processos afim de que não ocorram atrasos nos prazos, retrabalho, má elaboração dos projetos e falha no atendimento ao cliente, gerando prejuízos que a empresa não teria como suportar.

Neste cenário, a empresa utiliza uma metodologia própria para gestão dos seus projetos, que não mais atende as necessidades esperadas pelo diretor de projetos e pelo diretor geral. Estes têm, atualmente, a intenção de utilizar a experiência obtida em cada projeto como aprendizado para aprimorar seus processos em projetos futuros. Outra necessidade observada é a de realizar o levantamento do custo do projeto com precisão, pois o modelo de gestão atual utilizado pela empresa não fornece subsídios para que se estime o custo de forma correta e tampouco que se estipule o prazo com exatidão.

A metodologia de desenvolvimento de projetos utilizada pela *Software House* é desenvolvida de acordo com as seguintes etapas:

- a) É realizada uma reunião entre o diretor-geral e o cliente para a coleta de dados, com enfoque nas necessidades do negócio. Nesta reunião são levantados os dados e regras de negócios do cliente;
- b) Após a coleta destas informações, o diretor de projetos elabora um pré-projeto dividido em módulos, de acordo com as especificações informadas;
- c) Encaminha-se para o cliente este documento, cuja aprovação ocorrerá após a avaliação de conformidade às regras de negócio requeridas para a aplicação, informadas em reunião;
- d) Com a aprovação do cliente, o diretor de projetos define a estrutura da aplicação e delega aos programadores a execução;
- e) Periodicamente, o diretor de projetos se reúne com os programadores e confere o andamento do projeto;
- f) Com o projeto finalizado, inicia-se um período de testes, em que os programadores testam as funcionalidades da aplicação;
- g) Ao término dos testes internos, passa-se à fase de homologação, durante a qual a aplicação é disponibilizada ao cliente por um período, para que as regras de negócio sejam validadas. Neste período, os desenvolvedores ajustarão o código do programa, se necessário;
- h) Quando a fase de homologação do cliente termina, a aplicação é disponibilizada ao cliente, para que possa ser utilizada em seu negócio;
- i) Desta data em diante, o atendimento da empresa fica responsável em manter o contato com o cliente, encaminhando as solicitações de novas ferramentas e ajustes do projeto realizado, dando início a um novo ciclo de desenvolvimento; e,
- j) Com o projeto finalizado, o diretor de projeto se reúne com o diretor geral para avaliar o trabalho realizado.

A análise das etapas descritas permite vislumbrar um modelo linear e sequencial, focado no encadeamento temporal entre as atividades do projeto. Segundo Jaeger *et al.* (2010, p. 2-3):



Os métodos sequenciais aplicados a projetos de engenharia colocam ênfase maior nas etapas iniciais de projetos, ou seja, em planejar antes de construir. Um dos modelos de maior destaque é conhecido como Cascata, o qual organiza os projetos de *software* em quatro grandes etapas, a serem realizadas sequencialmente, que são: análise, projeto, codificação e testes. No processo em Cascata, cada etapa do ciclo é realizada por completo antes de iniciar a próxima, possuindo assim longos ciclos para a execução de cada etapa e entregando produtos de desenvolvimento somente após a conclusão de todas essas etapas. Ainda hoje, este é o modelo de desenvolvimento mais conhecido e amplamente utilizado no desenvolvimento de *software*.

Falhas percebidas no processo são apontadas posteriormente, para que seja possível selecionar quais variáveis deverão ser manipuladas, a fim de se conseguir o resultado almejado.

#### 4.2. Problemas levantados

Pela situação exposta no item anterior, é possível perceber falhas existentes no processo de criação dos *softwares* que podem resultar no fracasso do projeto, as quais são:

- a) *Feedback* com o cliente: ocorre no final do projeto, de forma que se as informações colhidas no início do projeto não forem precisas, o resultado será grande retrabalho para a empresa desenvolvedora;
- b) Controle do andamento do projeto: é falho, pois embora a divisão do projeto seja em módulos, e mesmo havendo tempo estipulado aos programadores para entrega do projeto, ainda assim não é possível, após a finalização do projeto, mensurar o tempo levado para a realização de cada atividade. Isto prejudica o julgamento do diretor de projetos quanto ao desempenho de sua equipe, sendo difícil, por exemplo, determinar quais colaboradores desempenham o trabalho com maior eficiência e eficácia;
- c) Integração entre os participantes do projeto: poderia ser melhor, pois mesmo sendo realizadas reuniões periódicas, a empresa carece de maior participação da equipe, o que prejudica o andamento do projeto;
- d) *Feedback* com o diretor geral: é falho, pois é realizada uma reunião de status somente após o projeto ser concluído e, com isso, o diretor geral não tem controle sobre a real situação dos projetos em andamento na empresa;
- e) Controle de custo do projeto: se mostra deficiente, pois, nas condições atuais da empresa, há dificuldade de calcular o custo efetivo do projeto. Isto ocorre porque a empresa não realiza o controle da hora trabalhada, que é importante devido à mão de obra dos colaboradores ser o elemento que gera o maior custo.

Tendo em vista estas falhas, salienta-se que sua existência não se deve apenas ao correto cumprimento de atividades e boas práticas, e o sucesso de um projeto não decorre somente desses fatores. Por essa razão, destaca-se, a seguir, a importância da perspectiva sistêmica, que almeja apreender a engenharia de *software* sem fragmentá-la em quaisquer dualidades, uma vez que a empresa é constituída de forma indissociavelmente técnica, social, histórica, econômica, ética e política, aspectos estes inextricavelmente articulados, formando um único e indivisível “tecido” (Santos e Júnior, 2009).

#### 4.3. Scrum como solução sistêmica

É possível estabelecer uma relação entre o *Scrum* e os conceitos da Teoria Geral dos Sistemas, na medida em que os elementos constituintes do *Scrum*, a saber, as cerimônias, os artefatos e os papéis, permitem a visualização de um modelo sistêmico dotado de elementos dinamicamente inter-relacionados, exercendo funções com objetivos em comum:

- a) *Product Backlogs*: determinada atividade ou função específica e importante para o sistema como um todo a fim de se alcançar um objetivo comum. Trata-se da lista de todas as atividades a serem desenvolvidas. Caracterizam-se dentro da teoria sistêmica como entradas, recursos e informação ao sistema (*inputs*);
- b) *Sprint Backlogs*: subsistema que processa ou converte suas entradas por meio de suas especialidades. Trata-se da lista de todas as funcionalidades a serem desenvolvidas durante o projeto completo, sendo bem definido e detalhado no início do trabalho - deve ser listado e ordenado por prioridade de execução;
- c) *Sprint Review/Retrospective*: ao final de cada *Sprint*, uma Reunião *Sprint Review* é realizada. Durante esta reunião, o *Scrum Team* apresenta o que foi realizado durante a *Sprint*. Tipicamente, esta apresentação é feita na forma de uma demonstração das novas funcionalidades. Dentro da teoria sistêmica, seria o processamento;
- d) *Potentially Shippable Product Increment*: no *Scrum* é necessário que em cada *Sprint* seja entregue um in-



cremento de produto potencialmente utilizável. Este incremento deve consistir de um arquivo executável, gerado a partir de código exaustivamente testado, e documentação dos procedimentos operacionais das funcionalidades, seja na forma de “Ajuda” ou “Documentação de Usuário”. São os *outputs*.

Na aplicação do *Scrum* como solução sistêmica para os problemas elencados anteriormente, pode-se destacar, como características principais deste processo, alguns fatores, descritos a seguir.

A divisão do processo de desenvolvimento é feita com base em pequenos ciclos chamados *Sprint*, em que um conjunto de funcionalidades é pré-definido. A cada conclusão são feitas entregas ao cliente, em intervalos regulares, ideia que corrobora com Inayata *et al.* (2015).

As equipes trabalham em conjunto para atingir o resultado desejado. Como forma de monitoramento do progresso do projeto, e a fim de refletir sobre o trabalho da equipe e sobre o andamento do projeto, são acompanhadas, no fim de cada ciclo de desenvolvimento, as finalizações das etapas por meio de um quadro *Kanban*.

Todos os envolvidos no processo fazem parte do projeto, o que transforma o cliente em um integrante da equipe, que o mantém com *feedbacks* constantes a respeito do andamento do sistema.

No método *Scrum*, o controle de custo é realizado com base em relatórios de finalizações das atividades, pois por meio deles é possível saber o tempo exato gasto em cada tarefa. Com isso, é realizado o cálculo para se obter este custo por meio das variáveis folha de pagamento *versus* tempo gasto, sendo possível também mensurar o custo final do projeto.

#### 4.4. Projeto de um sistema com *Scrum*

Neste estudo de caso, não foi possível acompanhar todo o desenvolvimento do projeto. O motivo disto é que decorrem, de acordo com o diretor de projetos, cerca de seis meses até a conclusão e homologação. Desta maneira, é realizado o registro dos primeiros resultados obtidos, bem como uma previsão para os demais resultados do projeto.

A *Software House* iniciou o primeiro projeto a utilizar a metodologia *Scrum* em junho de 2012, sem um treinamento prévio dos membros da equipe, que contam apenas com conhecimentos obtidos através da leitura de como utilizar o *Scrum*. Em decorrência desta situação, mesmo sendo estipulados padrões corretos, processos falhos foram adotados na metodologia implantada:

- a) *Daily meetings*, que, de acordo com o padrão *Scrum*, deveriam ser realizadas todos os dias e com duração de 15 minutos, foram realizadas duas ou três vezes na semana, sendo que o tempo de duração, em média, era de 40 minutos;
- b) Tarefas definidas no *sprint backlog* estavam sendo desenvolvidas com um tempo muito diferente do indicado pelos membros do time. Em certas ocasiões, o tempo definido era pouco para se realizar a tarefa e, em outras, o tempo era exagerado;
- c) Tasks adicionais que foram identificadas no desenvolvimento, e que deveriam ser adicionadas pelos membros, não estavam ocorrendo, o que gerou um resultado falso na contagem dos tempos das tarefas; e,
- d) Os membros da equipe não estavam atualizando diariamente seus *sprint backlogs* no quadro *Kanban* como pedido – esperavam um ou dois dias para fazê-lo.

Apesar dos primeiros resultados não terem sido tão favoráveis, o diretor de projetos relatou uma mudança na equipe, e apontou mudanças no comportamento e motivação dos membros:

- a) Os membros estão motivados por participarem do desenvolvimento do projeto e da criação das tarefas e com a agilidade com que estas tarefas são realizadas;
- b) Foi adotado um programa para controle do tempo gasto para a execução das tarefas;
- c) A equipe trabalha integrada no mesmo ambiente. Isso facilitou a comunicação entre eles, agilizando a resolução de problemas mais urgentes; e,
- d) O time está motivado pelo desafio de aprender e implantar uma nova metodologia de desenvolvimento.

Segundo o relato do diretor de projetos, um ponto que agregou maior confiabilidade ao trabalho da equipe foi a adoção da metodologia *Test Driven Development* (TDD) associada à metodologia *Scrum*. A TDD é uma maneira de desenvolver, na qual o programador escreve um teste para a solução implantada. Essa metodologia vem melhorar a qualidade final do *software* entregue pela empresa desenvolvedora, pois, como afirma Lewis (2004), qualidade não pode ser alcançada por meio da avaliação de um produto já feito. O objetivo, portanto, é prevenir defeitos de qualidade ou deficiências, em primeiro lugar, tornando os produtos avaliáveis através de medidas de garantia de qualidade.

De acordo com o relatado pelo observador deste trabalho, mesmo o cliente sendo cobrado e chamado para par-





ticipar das iterações, ele ainda não participa inteiramente do andamento do projeto e das atividades realizadas, pois ainda não se sente à vontade e não vê importância em sua participação nas fases do projeto. Embora este não seja o resultado desejado, é importante ressaltar que, com o *Scrum*, o cliente tem participado mais do projeto agora que antes, quando se usava outra metodologia, reduzindo, assim, as chances de haver retrabalho no projeto por disparidade nos requisitos e regras de negócio do sistema.

Por fim, nota-se que a essência do funcionamento de um projeto ágil está na sua capacidade de aprender e se adaptar com esse aprendizado. Para os Métodos Ágeis, as correções sistemáticas de pequenos erros cometidos, em razão de falsas suposições, são a chave para o funcionamento dos seus ciclos de especulação, colaboração e aprendizado (Highsmith, 2002 *apud* Jaeger *et al.*, 2010; Brhela *et al.*, 2015).

## 5. CONCLUSÕES

Conforme abordado, e corroborando com Faria (2002), a Teoria Geral dos Sistemas deu origem a uma nova perspectiva na administração, na qual a organização se configura como um sistema complexo, interagente com seu meio, e cujos gestores lidam com problemas, influenciando variáveis imprevisíveis. Desta maneira, o projeto de desenvolvimento de um *software* necessita, dentro desta concepção, de uma abordagem que considere todas as questões envolvidas no seu ciclo de vida como um todo e de forma integrada.

Neste sentido, a solução adotada para solucionar este desafio foi o uso da metodologia *Scrum* como método sistêmico, corroborando com os princípios apresentados por Beck *et al.* (2001). O *Scrum* caracteriza-se como um modelo de desenvolvimento que prioriza a comunicação, a participação coletiva e a divisão do projeto em ciclos de iterações. Sendo que este método possui artefatos que permitem a todos efetuar ajustes e correções necessárias aos processos de desenvolvimento tratados pela *Software House*.

Assim, este trabalho objetivou mostrar quais as consequências da prática do pensamento sistêmico aplicados ao desenvolvimento de projetos de uma *Software House* da Região Bragantina. O método empregado foi a observação participante, efetuada pelo diretor de projetos da empresa. O estudo de caso centrou-se nas necessidades encontradas na empresa durante esta observação, que no geral foram: problemas com o *feedback* do cliente e do diretor geral e controle de custos e do andamento do projeto.

Além da influência de fatores técnicos nas falhas processuais, a *Software House* também teve a necessidade de adaptar sua cultura organizacional ao *Scrum*, sendo que foram diagnosticadas falhas nesta implantação, como a atua-

lização incorreta de informações importantes, a improdutividade na criação e divisão das tarefas, e os desnecessários prolongamentos das reuniões diárias. No entanto, depois da implantação do *Scrum* houve uma maior integração, quais sejam: as equipes trabalham em conjunto para atingir o resultado desejado; os membros se mostraram motivados com a agilidade em que as tarefas eram realizadas; e, o cliente participou mais ativamente do projeto.

Antecipa-se ainda que, com o aprendizado individual e em conjunto dos participantes, a adaptação poderá ser cada vez melhor. Seguindo o exemplo da área de desenvolvimento da empresa, a metodologia *Scrum* se evidencia como solução para outros setores, como o setor de criação (*Design*), o que apresenta possibilidade de novas pesquisas neste campo de estudo.

Cabe, por fim, salientar que houve uma limitação temporal em relação à análise do processo de desenvolvimento e aplicação do método *Scrum* no projeto, isto se deve ao fato da duração do projeto ser superior ao intervalo de análise dispendido ao estudo de caso no *Software House*.

## REFERÊNCIAS

- Beck, K. et al. (2001), "Manifesto para o desenvolvimento Ágil de software", disponível em: <http://agilemanifesto.org/iso/ptbr> (Acesso em: 10 de junho de 2012).
- Bouding, K. (1956), "General systems theory - the skeleton of science", Management Science, *apud* Wetherbe, J. C. (1987), "Análise de Sistemas para Sistemas de Informação por Computador", 3. ed, Rio de Janeiro: Campus.
- Brhela, M., Methb, H., Maedchea, A., Werdera, K. (2015) "Exploring principles of user-centered agile software development: a literature review", Information and Software Technology, Vol. 61, pp. 163-181.
- Castro, D. M. (2000), Desenvolvimento organizacional, Campinas: [s.n.], disponível em: <http://pt.scribd.com/doc/34463834/Desenvolvimento-Organizacional> (Acesso em: 26 de junho de 2012).
- Cervo, A. L., Bervian, P. A., Silva, R. (2007), Metodologia científica, 6.ed., São Paulo: Pearson.
- Cleland, D. I. et Ireland, L. R (2007), Gerenciamento de projetos, 2. ed., Rio de Janeiro: LTC.
- Donaires, O. S (2006), "Programando na complexidade: um modelo sistêmico-cibernético de desenvolvimento e melhoria de software", artigo apresentado no 2º. Congresso Brasileiro de Sistemas, Ribeirão Preto.
- Donaires, O. S. (2009), "Uma abordagem sistêmica ao mapeamento e melhoria do processo de desenvolvimento de software", Revista FACEF Pesquisa, Franca, Vol. 12, n. 2, mai-ago.



- Faria, A. N. (1980), *Organização de empresas e prática de organização*, 6. ed., Vol. 1., Rio de Janeiro: Livros Técnicos e Científicos.
- Faria, J. C. (2002), *Administração: teorias e aplicações*, São Paulo: Pioneira Thompson.
- Ferreira, D., Costa, F., Alonso, F., Alves, P., Nunes, T. (2005), "Scrum: um modelo ágil para gestão de projectos de software", Faculdade de Engenharia da Universidade do Porto, 11p., disponível em: [http://paginas.fe.up.pt/~aaguiar/es/artigos%20finais/es\\_final\\_19.pdf](http://paginas.fe.up.pt/~aaguiar/es/artigos%20finais/es_final_19.pdf) (Acesso em: 11 de junho de 2012).
- Highsmith, J. A. (2002), "Agile software development ecosystems", Addison-Wesley: Boston, apud Jaeger, J. I., Oliveira, L. R., Pedroso, S. L., (2010), "Análise da aplicação de métodos ágeis na gestão de projetos em empresas desenvolvedoras de software", VII Convibra Administração – Congresso Virtual Brasileiro de Administração, 19-21 nov. 2010, disponível em: [http://www.convibra.com.br/upload/paper/adm/adm\\_1118.pdf](http://www.convibra.com.br/upload/paper/adm/adm_1118.pdf) (Acesso em: 11 de junho de 2012).
- Inayata, I., Salima, S. S., Marczakb, S., Danevac, M., Shamshirbandd, S. (2015) "A systematic literature review on agile requirements engineering practices and challenges", *Computers in Human Behavior*, Vol. 51, Part B, pp. 915-929.
- Jaeger, J. I., Oliveira, L. R., Pedroso, S. L. (2010), "Análise da aplicação de métodos ágeis na gestão de projetos em empresas desenvolvedoras de software", VII Convibra Administração – Congresso Virtual Brasileiro de Administração, 19-21 nov. 2010, disponível em: [http://www.convibra.com.br/upload/paper/adm/adm\\_1118.pdf](http://www.convibra.com.br/upload/paper/adm/adm_1118.pdf) (Acesso em: 11 de junho de 2012).
- Lewis, W. E. (2004), *Software testing and continuous quality improvement*, 2. ed, [S.l.]: Auerbach.
- Marconi, M. A. et Lakatos, E. M. (2010), *Técnicas de pesquisa: planejamento e execução de pesquisas, amostragens e técnicas de pesquisa, elaboração, análise e interpretação de dados*, 7. ed., São Paulo: Atlas.
- Martinelli, D. P. (2002), *Negociação empresarial: enfoque sistêmico e visão estratégica*, São Paulo: Manole.
- Martinelli, D. P. (2006), "Negociação, Administração e Sistemas: três níveis a serem inter-relacionados", *Revista Administração*, São Paulo, Vol. 41, No. 4, pp. 353-368.
- Prikladnicki, R., Willi, R., Milani, F. (Orgs). (2014) *Métodos Ágeis para Desenvolvimento de Software*. Porto Alegre: Bookman.
- Ribeiro, A. D. L. (2004), *Teorias da Administração*, São Paulo: Saraiva.
- Sabbagh, R. (2014) *Scrum: Gestão ágil para projetos de sucesso*. São Paulo: Editora Casa Do Código (Digital).
- Santos Filho, E. V. (2012) *Relação entre gestão de portfólio de projetos de software e desenvolvimento ágil: um caso com o framework Scrum no setor público*. 225f. Dissertação (Mestrado em Informática), Universidade Católica de Brasília, Brasília.
- Santos, R. P. et Santos Júnior, A. F. (2009), "Aspectos sociotécnicos do desenvolvimento de software utilizando Scrum em um caso prático", V Workshop Um Olhar Sociotécnico sobre a Engenharia de Software – Woses, Ouro Preto - Minas Gerais, 02 jun., pp. 38-49.
- Schwaber, K. (2004), "Agile project management with scrum", Microsoft apud Pereira, P., Torreão, P., Marçal, A. S. (2007), "Entendendo Scrum para gerenciar projetos de forma ágil", *Revista Mundo Project Management*.
- Senge, P. M. et Sterman, J. D. (1994), "Systems thinking and organizational learning: acting locally and thinking globally in the organizations of the future", em: Morecroft, John D. W. et Sterman, John D. (1994), *Modeling for learning organizations*, Portland: Productivity Press, pp. 195-216 apud Andrade, A. D. L. (1997), "Pensamento Sistêmico: um roteiro básico para perceber as estruturas da realidade organizacional", *Revista Eletrônica de Administração*, Porto Alegre, Vol. 3, No. 1.
- Serrador, P. et Pinto, J. K. (2015) "Does Agile work? — a quantitative analysis of agile project success", *International Journal of Project Management*, 33, pp. 1040–1051.
- Shimizu, T. (2011), *Sistemas integrados de gestão na economia globalizada*, São Paulo: Atlas.
- Svejvig, P. et Andersen, P. (2015) "Rethinking project management: A structured literature review with a critical look at the brave new world", *International Journal of Project Management*, Vol. 33, pp. 278-290.
- Terribili Filho, A. (2011), *Gerenciamento de projetos em 7 passos: uma abordagem prática*, São Paulo: M. Books.
- Wetherbe, J. C. (1987), *Análise de sistemas: para sistemas de informação por computador*, 3. ed., Rio de Janeiro: Campus.