

UNIVERSIDADE DO VALE DO RIO DOS SINOS - UNISINOS
UNIDADE ACADÊMICA DE EDUCAÇÃO CONTINUADA
MBA EM ADMINISTRAÇÃO DA TECNOLOGIA DA INFORMAÇÃO

MARIA ISABEL GEWEHR MELO

**FRAMEWORK ÁGIL PARA CONTROLE DE DEMANDAS DE SOFTWARE DE
UMA UNIVERSIDADE**

SÃO LEOPOLDO

2013

Maria Isabel Gewehr Melo

**FRAMEWORK ÁGIL PARA CONTROLE DE DEMANDAS DE SOFTWARE DE
UMA UNIVERSIDADE**

Trabalho de Conclusão de Curso apresentado como requisito parcial para a obtenção do título de Especialista em Administração de Tecnologia da Informação, pelo MBA em Administração da Tecnologia da Informação, da Universidade do Vale do Rio dos Sinos - UNISINOS

Orientador: Prof. Edson Wobeto

São Leopoldo – RS

2013

RESUMO

O objetivo deste trabalho é propor melhorias no método de análise e encaminhamento de demandas de software, enviadas ao setor de desenvolvimento de sistemas de uma Instituição de Ensino Superior. A proposta é utilizar as melhores práticas das metodologias ágeis aplicáveis a ambientes dinâmicos como o de uma instituição de ensino. Inicialmente, foi realizada a análise do cenário atual, coletando informações dos gestores das áreas de negócio e desenvolvimento de sistemas. Logo após, foi realizada uma pesquisa bibliográfica buscando identificar prós e contras da utilização de metodologias ágeis. Não poderia deixar de ser destacada também a importância da gestão de conhecimento, que cada dia se torna mais necessária nas organizações. Dessa forma, foi possível identificar que métodos ágeis, aliado a gestão de conhecimento se complementam e podem trazer muitos benefícios para as organizações. Esse trabalho traz uma proposta de implantação da metodologia Scrum adaptado à realidade do ambiente e atendendo a expectativa das áreas de negócio e desenvolvimento de sistemas.

Palavras-chave: Scrum. Metodologias Ágeis. Gestão de Conhecimento.

SUMÁRIO

1 INTRODUÇÃO	8
1.1 SITUAÇÃO PROBLEMÁTICA.....	8
1.2 OBJETIVOS.....	13
1.2.1 Objetivo geral.....	13
1.2.2 Objetivos específicos.....	13
1.3 JUSTIFICATIVA	14
2 FUNDAMENTAÇÃO TEÓRICA.....	15
2.1 Metodologias de desenvolvimento ágil x tradicional	15
2.1.1 Manifesto Ágil.....	18
2.1.2 Scrum.....	20
2.1.3 Utilização do Scrum.....	28
2.1.4 Extreme Programming (XP)	29
2.2 MÉTODOS ÁGEIS X GERENCIAMENTO DE PROJETOS	31
2.3 GESTÃO DE CONHECIMENTO	35
2.3.1 Conhecimento Organizacional	35
2.3.1.1 Infraestrutura organizacional.....	37
2.3.1.2 Pessoas.....	37
2.3.1.3 Tecnologia	37
2.3.2 Tipos de conhecimento	38
2.3.3 Liderança nas organizações do conhecimento	39
3 MÉTODO	41
3.1 DELINEAMENTO DA PESQUISA.....	41
3.2 DEFINIÇÃO DA UNIDADE DE ANÁLISE	41
3.3 TÉCNICAS DE COLETA DE DADOS	41
3.4 TÉCNICAS DE ANÁLISE DOS DADOS	42
3.5 LIMITAÇÕES DO MÉTODO	43
4 APRESENTAÇÃO E ANÁLISE DOS DADOS	44
4.1 ÁREA DE TI	45
4.2 ÁREAS DE NEGÓCIO.....	46
4.3 HISTÓRICO	46
4.4 PROPOSTA PARA IMPLANTAÇÃO DA METODOLOGIA ÁGIL	50
5 CONCLUSÃO.....	55

REFERÊNCIAS	57
APÊNDICE A - ROTEIROS DE ENTREVISTAS	61

LISTA DE FIGURAS

Figura 1 - Cenas de uma empresa de <i>software</i>	12
Figura 2 – Ciclos iterativos e incrementais	16
Figura 3 - Modelo Cascata (Clássico).	17
Figura 4 - Custo da mudança de projeto no modelo cascata.	18
Figura 5 - Qual o método ágil que você mais segue.....	20
Figura 6 - Exemplo de Product Backlog	23
Figura 7 - Exemplo de quadro de tarefas.....	26
Figura 8 - Visão geral da dinâmica do Scrum	27
Figura 9 - Diferenças entre escopo e visão do projeto	33
Figura 10 - Espiral do conhecimento.....	39

LISTA DE QUADROS

Quadro 1 – Quadro de Comparação entre os métodos de gerenciamento de projetos	34
Quadro 2 – Características de comparação de tipos de conhecimento.....	38
Quadro 3 – Solicitações enviadas 2012.....	44

LISTA DE ABREVIATURAS

- TI – Tecnologia de Informação
- IES – Instituições de Ensino Superior
- PO – Product Owner
- CMMI – Capability Maturity Model Intregation
- XP – Extreme Programming
- PDE - Plano de Desenvolvimento da Educação
- APM - Agile Project Management
- PMI – Project Management Institute

1 INTRODUÇÃO

Desenvolvimento baseado em métodos ágeis surgiu na década de 90, mas ganhou notoriedade em 2001 através do chamado “Manifesto para o Desenvolvimento Ágil de *Software*”, que foi assinado por Kent Beck e outros 16 notáveis desenvolvedores. Basicamente esse manifesto foi criado visando suprir fraquezas percebidas e reais da engenharia de *software* convencional. Essa engenharia combina uma filosofia e um conjunto de diretrizes de desenvolvimento. A filosofia encoraja a entrega incremental de *software* e a satisfação do cliente logo no início do projeto, e as diretrizes de desenvolvimento, dão prioridade a entrega e a comunicação constante com os clientes e não à análise e projeto que são à base da engenharia de *software* convencional. Apesar de não serem prioritárias, é importante destacar que análise e projeto não são descartados no desenvolvimento ágil. A base para a criação do manifesto ágil é o moderno mundo dos negócios, que necessita de *software* de forma rápida e é constantemente alterado. (PRESSMAN, 2010).

Neste capítulo será apresentado o problema deste projeto de pesquisa, objetivos e a justificativa para realização do estudo.

1.1 SITUAÇÃO PROBLEMÁTICA

Em virtude das mudanças sócio econômicas a população brasileira tem buscado qualificação, desta forma o ensino superior apresentou e ainda apresenta o desafio de receber um número maior de alunos. Em resposta a essa procura, ocorre a expansão do ensino superior, buscando atender também a ampliação de acesso solicitada através do PDE (Plano de Desenvolvimento da Educação). Esse processo faz com que mudanças significativas sejam realizadas nas Instituições, como por exemplo: diversificação, flexibilização na estrutura dos cursos de graduação, relação teoria-prática, avaliações periódicas, dentre outros. A TI (Tecnologia da Informação) hoje é essencial em IES (Instituições de Ensino Superior). As instituições estão sendo pressionadas cada vez mais a atuarem de acordo com os novos conceitos de ensino e em função disso, são realizados diversos investimentos visando atender tanto a área acadêmica (ensino, pesquisa e extensão) quanto administrativa. (ALVES, 2005).

As IES, por se constituírem em organizações que contam, no seu quadro, com especialistas de alto nível, devem posicionar-se de forma flexível, adotando instrumentos de gestão, que propiciem sua inserção num ambiente dinâmico, dotando-as de condições que possibilitem um desempenho capaz de fazer frente às demandas da sociedade atual. (ALVES, 2005, p. 22).

Holderegger (2010) relata que as IES precisam apoiar-se em sistemas de informação capazes de garantir qualidade na tomada de decisões estratégicas, mas destacam que existem alguns fatores que prejudicam o processo de construção desses sistemas. Dentro os fatores prejudiciais, podemos destacar:

- pouca documentação formal de processos;
- gestores desconexos com os objetivos centrais da Instituição.

Visando apoiar a IES foco desse estudo de caso na gestão adequada das suas demandas de sistemas, pretende-se propor o uso de métodos ágeis na área de desenvolvimento. Segundo Holderegger (2010), esse método é recomendado para ambientes dinâmicos, com grande variabilidade e volume de solicitações. O que reflete a realidade desse estudo de caso.

Para a identificação dos problemas da IES frente à área de desenvolvimento foram realizadas entrevistas com representantes da TI e das áreas de negócio sendo que foram destaque em suas respostas os seguintes itens:

- a área de desenvolvimento de sistemas é essencial ao negócio;
- a área de desenvolvimento de sistemas esta sobrecarregada;
- não há centralização do envio de demandas dentro das áreas de negócio;
- falta documentação/treinamento;
- o tempo para atendimento das necessidades das áreas de negócio é a principal reclamação, principalmente quando se trata de algo novo (melhoria).

De acordo com a pesquisa realizada pela Universidade de São Paulo em 2012, que visava medir, a nível nacional, o estado da prática de métodos ágeis, os principais benefícios observados foram:

- aumentou em 76% a habilidade de gerenciar mudanças de prioridade e produtividade da equipe;
- aumento de 75%, em qualidade, moral do time e simplificação do processo de desenvolvimento. (MELO, 2012).

A unidade de estudo desse trabalho é uma equipe de desenvolvimento de sistemas, que conta com aproximadamente 22 colaboradores que atendem toda a demanda de sistemas da IES que será referenciada neste trabalho como Empresa X.

Demanda é o ato de demandar, buscar, procurar. Ou seja, são as necessidades das áreas de negócio encaminhadas ao setor de desenvolvimento de sistemas em busca de solução/adequação a realidade do negócio naquele momento. (DICIONÁRIO, 2013).

Os sistemas desenvolvidos internamente dão conta de diferentes e importantes áreas de negócio, como por exemplo, contas a receber, gestão acadêmica, gestão de estágios, dentre outras.

Essas demandas crescem a cada dia. Além das demandas relacionadas aos sistemas desenvolvidos internamente, existe também a contratação de terceiros para soluções específicas como: *Enterprise Resource Planning* (ERP), *Business Intelligence* (BI) e *Student Relationship Management* (SRM). Apesar da contratação desses terceiros, a equipe de desenvolvimento de sistemas acompanha as implantações e realiza parte do trabalho, como por exemplo, integrações com sistemas internos, suporte após a implantação, customizações, etc.

O funcionamento adequado de todas essas aplicações, desde a sua implantação, passando pelas customizações, integrações entre os diferentes aplicativos, treinamentos e suporte, é a rotina de trabalho da área de desenvolvimento de sistemas.

Customizações segundo Souza (1999) é a modificação de um sistema ERP para que esse se torne adequado a uma determinada situação empresarial.

As demandas são encaminhadas por qualquer funcionário de qualquer outro setor da Instituição utilizando um sistema denominado: site de solicitações.

O site de solicitações é uma ferramenta desenvolvida internamente, disponível na *web*, que permite enviar as solicitações para diversos setores, sendo um deles o setor de desenvolvimento de sistemas.

Vários movimentos de conscientização já foram desenvolvidos para que os colaboradores do setor exijam uma solicitação para qualquer demanda de trabalho que recebam. Nessa solicitação há informações básicas quanto ao demandante (nome, setor, ramal), uma descrição do que está sendo solicitado, uma ideia de prazo, e, uma ideia de criticidade através da consideração de urgência. Para auxiliar na interpretação também é possível que sejam enviados arquivos/documentos anexos à solicitação.

Em geral as demandas recebidas diariamente se referem a erros de aplicações, dúvidas sobre regras de negócio, erros de utilização das ferramentas, customizações simples e

customizações que necessitam de uma análise mais aprofundada. Mesmo que a equipe seja responsável por gerenciar as solicitações, não existe um procedimento padrão para a priorização do atendimento dessas solicitações.

Buscando melhorar a forma de classificação dessas solicitações, alguns ajustes já foram feitos e estão sendo praticados a cerca de dois anos. Dentre eles, podemos citar a reestruturação das equipes, ocorrendo a divisão por times. Cada time possui no mínimo dois analistas, além de programadores e estagiários.

Mesmo com o crescente investimento em tentativas de aplicação de metodologias para gerenciar e atender de forma mais satisfatória as demandas, o sucesso total não é alcançado e comumente ocorre a demora no atendimento e reclamações de clientes internos.

Os colaboradores do setor de desenvolvimento também se sentem incomodados com as mudanças de prioridade imposta pelos clientes, isso faz com que muitas vezes percam o ritmo de trabalho, pois, um projeto que está sendo executado é substituído por outro.

Os clientes relatam que muitas vezes solicitam uma mudança/evolução do sistema e, quando essas são entregues já não atendem mais aos requisitos do negócio, não sendo portanto válidas. Ou ainda, algo que precisa ser desenvolvido e que, quando chega o momento de atendimento, os requisitos levantados já estão diferentes ou o projeto não é mais necessário.

Diante do exposto, a ideia central deste trabalho é identificar dentro da metodologia ágil, princípios que possam ser implantados nesse ambiente, e propor um método de aplicação que possa trazer melhorias nos seguintes quesitos:

- melhor gerenciamento da priorização das demandas de desenvolvimento, buscando agilidade no atendimento e maior alinhamento com o planejamento estratégico da Instituição;
- gestão de conhecimento da equipe;
- percepção dos setores demandantes sobre o trabalho realizado pela equipe de desenvolvimento;
- agilidade no atendimento de dúvidas e revisões de regras de negócio.

Esse movimento se faz necessário, pois, de acordo com Couto (2007), as empresas estão sendo exigidas a mostrar competência técnica, operacional e gerencial e isso requer demonstrações de controles de atividades internas e externas. Dessa forma, cresce a demanda de sistemas que suportem esses processos. Em uma IES ocorrem demandas provenientes de órgãos governamentais nos diversos âmbitos, além de adequações a melhores práticas de

mercado para competitividade. Essas necessidades fazem com que o ambiente de desenvolvimento de sistemas seja muito parecido com o representado através da Figura 1, onde podemos notar um acúmulo de trabalho, entrega de produtos sem a devida qualidade e o abandono dos procedimentos visando ganho de tempo.

Figura 1 - Cenas de uma empresa de *software*



Fonte: COUTO (2007)

Sabe-se que as empresas precisam se moldar para atender a demanda de mercado, e que agilidade, segundo Highsmith (2002), é a habilidade de criar e responder a mudanças de forma a manter a lucratividade em um turbulento ambiente de negócios. Diante disso, é necessário que a TI também se molde e fique adequada às necessidades organizacionais. A TI precisa ser peça chave para o atendimento das metas organizacionais. Conforme Weill (2010), ter conhecimento em TI faz com que esse setor seja um ativo estratégico e para que se tenha vantagem, a empresa e seus gestores precisam ter habilidade para usá-la de forma consistente, obtendo assim vantagens competitivas frente à concorrência.

Este trabalho pretende contribuir com os processos organizacionais, buscando um melhor desempenho da equipe de desenvolvimento de sistemas, estudando maneiras que permitam que a equipe tenha melhores condições de entregar produtos com maior qualidade e, em menor tempo, de forma a favorecer a tomada de decisão de seus gestores e contribuir de maneira efetiva com o crescimento e desenvolvimento desta Instituição, fazendo com que esse

setor não seja considerado apenas um centro de custo, onde se investe muito e de onde não se obtém retorno.

Com base no descrito acima, a questão de pesquisa deste trabalho é:

Como ajustar os processos do setor de desenvolvimento de sistemas da Empresa X buscando maior agilidade, melhor comunicação entre clientes internos e equipe de desenvolvimento e maior alinhamento do trabalho do setor aos objetivos estratégicos da IES, contribuindo com o desenvolvimento da organização?

1.2 OBJETIVOS

Abaixo são expostos os objetivos deste projeto de pesquisa. A ideia base é buscar moldar uma metodologia de desenvolvimento ágil que possa ser implantada, aceita pela equipe e pela direção, e que possa rapidamente gerar benefícios passíveis de medição.

1.2.1 Objetivo geral

Identificar as fragilidades e virtudes do setor de desenvolvimento de sistemas da Empresa X e a partir desta identificação, propor melhorias através do uso de metodologias de desenvolvimento ágil, que possam contribuir com o processo de atendimento de demandas do setor, tornando o atendimento mais ágil, assertivo e contribuindo com os objetivos estratégicos.

1.2.2 Objetivos específicos

- analisar a demanda existente, identificando os maiores demandantes e suas principais necessidades;
- identificar pontos fortes e fracos no setor;
- propor um modelo de gerenciamento de demandas adaptado as necessidade do ambiente;
- desenvolver um modelo prático de gestão de conhecimento;
- propor metodologia de melhoria continua.

1.3 JUSTIFICATIVA

O manifesto ágil, segundo Highsmith (2002), tem em sua essência, o desenvolvimento de *software* baseado na agilidade, flexibilidade e na habilidade de comunicação. A metodologia se baseia no princípio de oferecer *software* de qualidade e com valor agregado ao negócio em curtos períodos de tempo.

Diante das premissas do manifesto ágil e visando que as entregas realizadas pelo setor de desenvolvimento de sistemas da Empresa X sigam essas definições, ao final desse projeto de pesquisa, pretende-se entregar uma sugestão de processos baseados em desenvolvimento ágil, que agreguem valor ao modelo de negócio apresentado. Pretende-se aumentar ainda mais a eficiência do setor de desenvolvimento de TI e contribuir de forma significativa com os objetivos estratégicos da Instituição.

Acredita-se que é o momento oportuno para que mudanças sejam propostas visto que algumas melhorias já implantadas foram aceitas pela Instituição e houve o entendimento de que essas melhorias, mesmo gerando um desconforto inicial, geraram benefícios que já foram percebidos pelos clientes internos conforme relatos colhidos na entrevista realizada.

Os clientes já perceberam um aumento na quantidade de atendimentos realizados e, mesmo considerando que ainda não é o suficiente, concordam que precisam participar mais dos projetos e veem essa participação positivamente, vislumbrando como resultado, produtos mais adequados a sua necessidade.

O desenvolvimento ágil de *software* propõe entregas rápidas e que geram valor ao cliente. Diante disso, nota-se uma grande oportunidade de obter sucesso com a implantação de métodos ágeis, visto que a demora no atendimento é a principal queixa dos clientes.

As entrevistas também apontam que os clientes entendem como positivo o recebimento parcial que traga benefícios perceptíveis.

Os próximos capítulos deste trabalho irão tratar da fundamentação teórica envolvida neste estudo e do método de pesquisa utilizado.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo trata da revisão bibliográfica feita para este trabalho. Na primeira parte é vista uma revisão sobre a metodologia ágil, metodologia tradicional, principais frameworks ágeis utilizados no mercado. A seguir, são destacados alguns pontos a respeito de gestão de conhecimento nas organizações.

2.1 METODOLOGIAS DE DESENVOLVIMENTO ÁGIL X TRADICIONAL

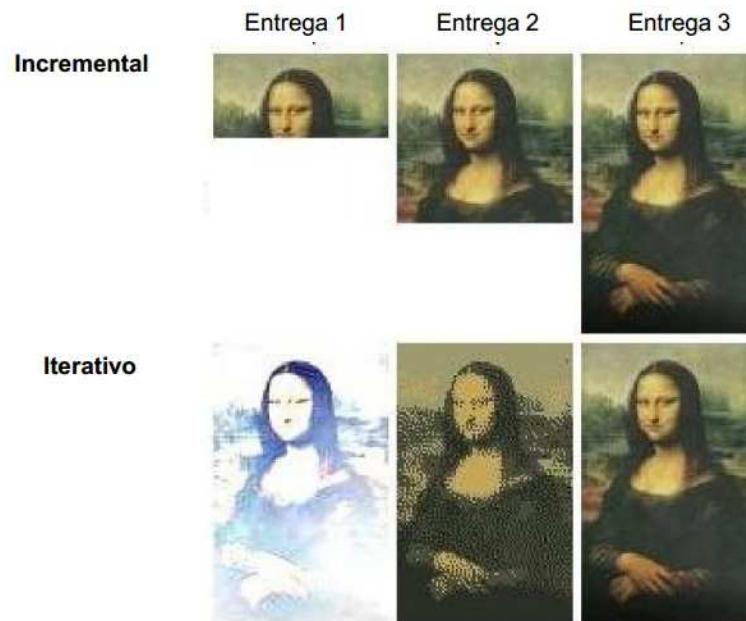
Segundo Carvalho (2009), na década de 90 começaram a surgir técnicas de desenvolvimento ágil. De acordo com os autores, essa disciplina surgiu fortemente influenciada pelas melhores práticas da indústria japonesa, pelos princípios de manufatura enxuta e também pelas estratégias de gestão de conhecimento de Nonaka e Takeuchi e Peter Senge.

Segundo Marçal (2009) o surgimento de vários métodos ágeis no final da década de 90 contribuiu para que a partir de 2000 acompanhássemos a tendência de desenvolvimento ágil de aplicações. Citando alguns métodos ágeis que surgiram na época, a autora destaca: Adaptive Software Development de HighSmith, Crystal de CockBurn, Dynamic Systems Development de Cohen, et al, eXtreme Programming de Beck e Feature Driven Development de HighSmith e Scrum.

Os métodos ágeis empregam princípios de ciclo iterativos, entrega rápida de software funcionando e simplicidade, conforme publicado no manifesto ágil para o desenvolvimento de software. (MARÇAL, 2009).

Ciclos iterativos e incrementais podem ser interpretados mais didaticamente através da explicação de Santos (2013) visualizada na figura abaixo:

Figura 2 – Ciclos iterativos e incrementais



Fonte: SANTOS (2013)

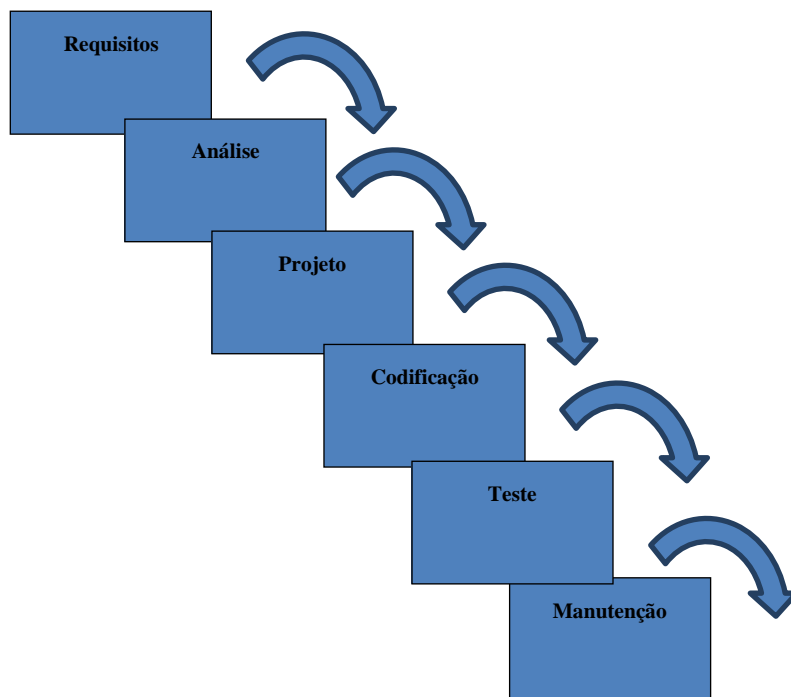
Iteração pode ser vista como o grande diferencial frente à teoria tradicional, visto que a iteração busca planejar apenas o curto prazo. O método introduz ciclos curtos de realização e testes, dessa forma interpreta-se ciclos iterativos e incrementais, fazendo com que o projeto seja executado em fases onde a construção ocorre aos poucos e o resultado pode ir sendo percebido já no início do processo. (AMARAL, 2011).

Segundo Santos (2013), desenvolvimento ágil de software é baseado na linha de pensamento de dividir para conquistar. O autor destaca que em virtude de complexidade, tamanho, mudanças de requisitos e a urgência de utilizar um recurso de software é praticamente inconcebível a utilização do modelo cascata tradicional (clássico) de desenvolvimento de software, onde a ideia é a entrega total do produto de uma única vez.

O modelo clássico, é baseado em um contexto de desenvolvimento de software muito diferente do atual, foi concebido baseado em utilização de mainframes e terminais burros. Alterações de software nesta arquitetura eram muito caras, pois o acesso as máquinas era limitado e ferramentas de apoio ao desenvolvimento de software como depuradores e analisadores de código eram inexistentes. Esse modelo foi o primeiro processo de desenvolvimento de software publicado e é utilizado até hoje. Em resumo, é uma sequência de passos que deve ser seguida, acompanhada de uma documentação padrão que deve ser aprovada para que se inicie a próxima etapa. Esse modelo dominou o desenvolvimento de

software até o início da década de 90, contrariando a opinião de diversos pesquisadores e desenvolvedores que já identificavam problemas na adoção da visão sequencial de tarefas, destacando Fred Brooks em seu artigo “No Silver Bullet: Essence and Accidents of Software Engineering”, esse autor declara que especificar totalmente um software antes do início de sua implementação é impossível. Tom Gilb, também destaca que o uso do modelo clássico em grandes softwares apresenta maior risco e menores possibilidades de sucesso. (SOARES, 2004).

Figura 3 - Modelo Cascata (Clássico).



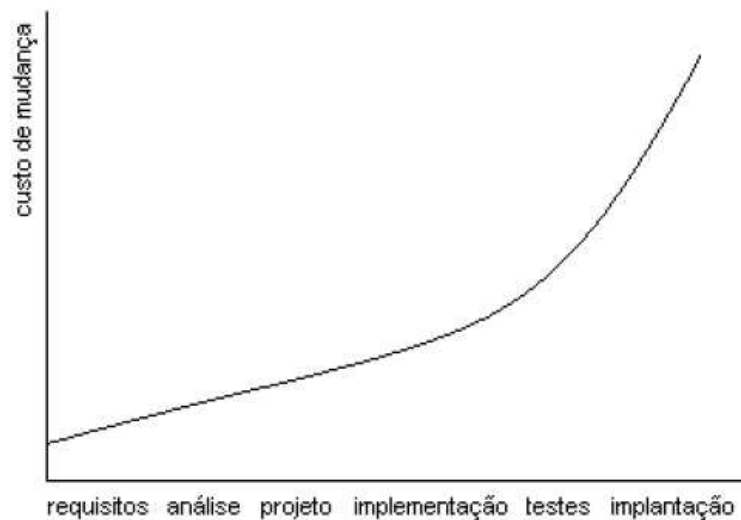
Fonte: Adaptado pelo autor (2013).

No modelo cascata assume-se que todos os requisitos são conhecidos na primeira fase do projeto. Se algum detalhe ficar de fora desse levantamento, e é comum que isso ocorra, em função de que os solicitantes dificilmente conseguem visualizar um sistema no início do projeto, as alterações nos requisitos tornam-se onerosas e por consequência não são bem vindas, porque elas obrigam que o desenvolvedor reinicie todas as fases do projeto. (MALLMANN, 2011).

Abaixo pode ser visto um gráfico que demonstra o crescimento do custo de alterações de requisitos durante as fases de um projeto de desenvolvimento de sistemas. Evidencia-se o alto crescimento do custo em função do andamento das fases do projeto. Dessa forma, indica-

se o método cascata apenas para situações onde os requisitos são estáveis e requisitos futuros são previsíveis. Essas situações são difíceis de serem atingidas em função da dinâmica das organizações, das alterações de legislação e da dificuldade dos solicitantes em definir escopo futuro de um sistema. (SOARES, 2004).

Figura 4 - Custo da mudança de projeto no modelo cascata.



Fonte: SOARES (2004).

2.1.1 Manifesto Ágil

O manifesto ágil para o desenvolvimento de software foi criado em fevereiro de 2001, quando Kent Beck e outros desenvolvedores, gerentes e entusiastas se reuniram em Utah buscando alternativas para os processos de desenvolvimento de sistemas tradicionais baseados em pesada documentação. Nesse manifesto são destacados os princípios da metodologia ágil conforme segue:

- nossa maior prioridade é satisfazer o cliente, através da entrega adiantada e contínua de software de valor;
- aceitar mudanças de requisitos, mesmo no fim do desenvolvimento. Processos ágeis se adequam a mudanças, para que o cliente possa tirar vantagens competitivas;

- entregar software funcionando com frequência, na escala de semanas até meses, com preferência aos períodos mais curtos;
- pessoas relacionadas a negócios e desenvolvedores devem trabalhar em conjunto e diariamente, durante todo o curso do projeto;
- construir projetos ao redor de indivíduos motivados. Dando a eles o ambiente e suporte necessário, e confiar que farão seu trabalho;
- o método mais eficiente e eficaz de transmitir informações para, e por dentro de um time de desenvolvimento, é através de uma conversa cara a cara;
- software funcional é a medida primária de progresso;
- processos ágeis promovem um ambiente sustentável. Os patrocinadores, desenvolvedores e usuários, devem ser capazes de manter indefinidamente, passos constantes;
- contínua atenção a excelência técnica e bom design, aumenta a agilidade;
- simplicidade: a arte de maximizar a quantidade de trabalho que não precisou ser feito;
- as melhores arquiteturas, requisitos e designs emergem de times auto-organizáveis;
- em intervalos regulares, o time reflete em como ficar mais efetivo, então, se ajustam e otimizam seu comportamento de acordo.

Em 2013, completou-se 12 anos de Manifesto Ágil. Foram dezessete especialistas em processo de desenvolvimento de *software* que se reuniram e montaram as bases da metodologia ágil que traz nos seus princípios os seguintes conceitos:

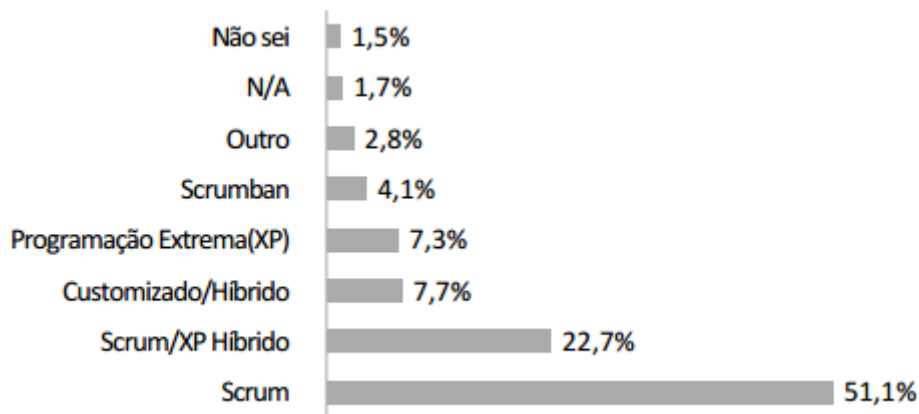
- a) maior valorização de indivíduos e interação do que processos e ferramentas;
- b) *software* em funcionamento é melhor do que documentação abrangente;
- c) colaboração com o cliente é maior do que negociação de contratos e
- d) responder a mudanças é mais valioso do que seguir um plano.

Marçal (2009) relata que fatores relacionados a competição de mercado têm favorecido os métodos ágeis. Cresce muito a necessidade das empresas de entregar produtos e serviços cada vez melhores com prazos cada vez mais curtos e preços mais atrativos. Marçal (2009 apud BOEHM; TURNER, 2004) descreve que o ambiente onde o software é imaginado, criado e especificado está sofrendo profundas mudanças, os sistemas de informação estão cada vez maiores e mais complexos. Requisitos mudam em um ritmo

acelerado e o software está presente em toda a parte. “Processos prescritivos e padronizados com práticas invariáveis não são mais recomendados para o ambiente volátil de desenvolvimento de software e produtos”. (MARÇAL, 2009, p. 20).

Acreditando que os conceitos de desenvolvimento ágil podem contribuir de forma favorável com as estratégias da empresa, pretende-se realizar um levantamento das melhores práticas dentro deste conceito e propor a utilização delas. Para tanto, a revisão bibliográfica vai tratar dos modelos SCRUM e XP, por serem os mais utilizados no mercado brasileiro, conforme demonstrado abaixo pela pesquisa realizada pelo grupo de pesquisa em métodos ágeis da Universidade de São Paulo em 2012.

Figura 5 - Qual o método ágil que você mais segue



Fonte: MELO (2012)

2.1.2 Scrum

De acordo com Carvalho (2009), Scrum foi desenvolvido por Jeff Sutherland em 1993 baseado em um artigo de Nonaka e Takeuchi de 1986. Esse artigo abordava as vantagens de pequenos times de desenvolvimento de produtos.

Carvalho também relata que o nome Scrum surgiu da comparação entre desenvolvedores e jogadores de Rugby. Scrum é o nome dado a reunião rápida que ocorre quando os jogadores vão iniciar um lance.

Como esse método foi desenvolvido por pesquisadores ligados a área de desenvolvimento de software, foi nessa área que o Scrum teve mais penetração, mas na verdade pode ser utilizado em qualquer trabalho iterativo e incremental, podendo ser aplicado ao desenvolvimento de produtos em geral. (RISING, 2000).

O método Scrum segundo Schwaber (2011) é um framework para desenvolvimento de software fundamentado nas teorias empíricas de controle de processos. O empirismo afirma que o conhecimento vem da experiência. Scrum é baseado em ciclos iterativos e incrementais e busca prever e controlar os riscos. O método baseia-se em três etapas:

- transparência: todos os aspectos do processo que influenciam no resultado devem ser visíveis para aqueles que gerenciam. Quando alguém tiver que inspecionar o processo, aquilo que foi feito deve ser igual à definição, ou seja, igual ao que foi pedido.
- inspeção: o processo deve ser inspecionado (iterações) com frequência suficiente para que as mudanças possam ser detectadas. A frequência deve levar em consideração que os processos possam ser modificados no ato da inspeção. Um fator importante é a habilidade e a dedicação das pessoas envolvidas no processo.
- adaptação: se durante a inspeção os envolvidos detectarem um ou mais aspectos que possam prejudicar o resultado final, medidas de ajuste devem ser tomadas o mais rapidamente possível para minimizar o problemas. Há três momentos de inspeção no Scrum: a reunião diária (daily scrum), reunião de planejamento (Sprint Review Meeting) e a reunião de retrospectiva do sprint (Sprint Review Meeting).

Schwaber (2011) descreve os papéis do framework, como segue:

- scrum master – responsável por fazer com que o processo do Scrum seja entendido e seguido no time e na organização. É uma espécie de líder, pode ser um membro do próprio time e pode variar de um projeto para outro, a ideia é que o Scrum Master figure como um treinador fazendo com que o time seja produtivo e produza com qualidade. O Scrum Master nunca deve ser o dono do produto.
- product owner – responsável pelos requisitos, representa o interesse de quem está pedindo o produto (cliente). É uma pessoa apenas, não um comitê e pode ser algum membro do time. Sua função é gerenciar o product backlog, manter a equipe informada e gerenciar as prioridades. Schwaber (2011) declara que é a responsabilidade desse papel a maximização do valor do trabalho executado pelo time. Para que esse papel tenha sucesso, todos na organização devem respeitar suas decisões. Ninguém tem autorização para mudar as prioridades da equipe além do product owner.

- time – equipe de desenvolvedores, responsável pela execução do projeto. É do time a responsabilidade sobre o sucesso ou fracasso de cada iteração e sobre o projeto como um todo. A equipe precisa ser auto gerenciável, o scrum master pode orientar e auxiliar, mas a equipe precisa descobrir a melhor forma de maximizar sua produtividade. Quando a equipe assume o Sprint, precisam fazer o que for necessário para cumprir o seu compromisso. No final do Sprint a equipe demonstra a funcionalidade ao proprietário do produto.

Kniberg (2007) relata que Scrum não é uma metodologia, Scrum é um framework, e por isso não vai dizer exatamente o que deve ser feito. Destaca também que você é forçado a adaptar o processo pela sua situação específica.

Carvalho (2012), afirma que o método Scrum não define nenhuma técnica ou método específico para a fase de desenvolvimento de software, são estabelecidas na verdade um conjunto de técnicas e práticas gerenciais que devem ser adotadas para o sucesso do projeto, essas práticas são nomeadas da seguinte forma:

- product backlog: prática responsável pela coleta dos requisitos. São marcadas reuniões com todos os envolvidos, investidores e parceiros do projeto. São apontados os itens com todas as necessidades do negócio e os requisitos técnicos a serem desenvolvidos, ou seja, uma lista de atividades que provavelmente serão desenvolvidas durante o projeto. De acordo com Kniberg (2007), product backlog é o coração do Scrum, é onde tudo começa. Uma lista de requisitos, estórias, ou seja, os desejos dos clientes. Abaixo um exemplo de product backlog, onde os campos apresentados na tabela tem a seguinte definição:
- ID: é uma identificação única da estória; não tem dois pontos depois da palavra ID
- nome: é considerando um nome curto para identificação que seja facilmente interpretado pelos membros da equipe;
- imp: é a importância da estória em pontos para o product owner, quanto mais pontos mais importante.
- est: é a estimativa inicial da equipe quanto ao tempo de implementação da estória. Corresponde mais ou menos a relação homem/dia.
- como demonstrar: é uma descrição de como demonstrar o funcionamento da estória. Podemos resumir como a especificação de um teste simples.

- notas: informações adicionais curtas e relevantes ao desenvolvimento.

Figura 6 - Exemplo de Product Backlog

PRODUCT BACKLOG (exemplo)					
ID	Nome	Imp	Est	Como demonstrar	Notas
1	Depósito	30	5	Logar-se, abrir a página de depósito, depositar R\$ 10,00, ir para a página do meu saldo e verificar que este aumentou em R\$ 10,00.	Precisa de uma diagrama UML de sequência. Não é necessário se preocupar com criptografia por enquanto.
2	Verificar seu próprio histórico de transações	10	8	Logar-se, clicar em “transações”. Fazer um depósito. Voltar para transações, verificar se o novo depósito é listado.	Usar paginação para evitar consultas muito grandes ao banco de dados. Projetar de forma similar à página de visualização de usuários.

Fonte: KNIBERG (2007)

Estórias são os requisitos, é o que o cliente deseja escrito utilizando a linguagem do cliente. Normalmente são descritas pelo product owner sendo possível que outras pessoas adicionem requisitos, mas é responsabilidade do product owner entender cada estória e saber priorizá-las. (KNIBERG, 2007).

- Sprint Planning Meeting – é a reunião onde o time faz o planejamento do Sprint. Essa reunião segundo Kniberg (2007) deve respeitar um princípio básico do Scrum, o tempo, o autor preza por reuniões que respeitem o tempo planejado para acontecer e que sejam eficientes. É importante também que sejam estabelecidos prazos realistas. Se a reunião não rende conforme o esperado, o autor defende que o Sprint deve sofrer as consequência da falta de planejamento e com isso a equipe aprende e a próxima reunião será mais eficiente. Podemos associar o pensamento desse autor com ideias sobre gestão de conhecimento, onde o conhecimento tácito de cada integrante é compartilhado com a equipe e o conhecimento organizacional é gerado. De acordo com o autor, de uma reunião de planejamento de Sprint os seguintes resultados devem ser alcançados:

- objetivo do Sprint;
- lista de membros da equipe e seus níveis de comprometimento;
- um Sprint backlog;
- data para apresentação do Sprint;
- data e local para a reunião diária.

Conforme Schwaber (2004) essa reunião deve durar 8 horas sendo composta por duas partes de 4h cada, uma delas para seleção dos itens do product backlog e outra para preparar o sprint backlog.

- Sprint Backlog – Lista de atividades que devem ser desenvolvidas no Sprint. A lista é montada a partir do product backlog que é devidamente priorizado pelo product owner. A equipe decide quantas estórias do product backlog farão parte do Sprint backlog. O Sprint backlog fica organizado pela importância da estória e na reunião de planejamento a equipe define conforme sua percepção a quantidade de estórias que consegue atender pelo tamanho do sprint e a velocidade da equipe. Como visto cada estória tem uma estimativa de tempo, no Sprint backlog a equipe seleciona os itens prioritários conforme a estimativa de tempo e a produtividade da equipe. A estimativa de tempo pode ser realizada utilizando o método planning poker. Nesse método todos os membros da equipe devem estimar o tempo de execução de uma estória, isso faz com que todos os membros acabem se envolvendo e entendendo algo sobre a estória mesmo não sendo a pessoa responsável pela sua implementação, essa atividade também favorece a troca de conhecimento e o surgimento de questões importantes sobre a estória. No planning poker cada membro precisa selecionar através de um baralho de 13 cartas a sua estimativa de tempo para execução da estória, as cartas são reveladas simultaneamente, fazendo com que cada membro da equipe realize a sua estimativa sem a influência da escolha de outro colega. Se houver muita divergência entre as estimativas, a equipe precisa discutir e chegar a uma visão comum do trabalho envolvido na estória e a estimativa é realizada novamente, isso é repetido até que as estimativas sejam próximas;
- Sprint – é considerada a principal prática do Scrum onde são implementados os itens de trabalho definidos no Product Backlog. É na reunião de planejamento do Sprint (Sprint Planning Meeting) que o tamanho do Sprint é definido, não existe

uma regra quanto à definição de tamanho, mas Kniberg (2007) relata que sprints longos são os preferidos pelos desenvolvedores em função de um maior tempo para adaptação. Já os product owners preferem sprints mais curtos pela entrega e feedback mais frequente, com isso a possibilidade de realização de ajustes caso algo não esteja indo bem. O principal desse processo é o compromisso, a equipe fica com a responsabilidade de realizar a entrega dentro do tempo que ficou estimado na reunião. Kniberg (2007), afirma que não existe discussão quanto às datas de entrega. O tamanho do Sprint precisa ser experimentado e definido pela equipe (auto gerenciável), esse tamanho corresponde ao número de semanas que vai durar o Sprint. Kniberg (2007) sugere a experimentação para a definição do tamanho ideal do Sprint para cada caso. Deve ser observada a questão de que um Sprint não deve ser tão longo e nem tão curto, a equipe deve adequar o tamanho a ponto de possuir a agilidade adequada ao ambiente e também conseguir dar fluidez e solucionar eventuais problemas em um Sprint. Após a escolha do tamanho adequado, esse deve ser mantido a ponto de a organização não questionar o andamento do trabalho, pois sabe o tempo que leva para receber um produto. A velocidade da equipe é sua capacidade de produção. Já Schwaber (2004) defende que a regra básica do Scrum é a liberação de produtos em ciclos de 30 dias;

- Daily Scrum – reunião diária rápida entre os membros do time para definição das tarefas do dia e avaliação dos resultados do dia anterior. Essa reunião é chamada de Stand Up Meeting (realizada em pé) e três perguntas devem ser respondidas por cada membro da equipe sobre suas responsabilidades:
 - O que foi feito ontem?
 - O que será feito hoje?
 - Há algum obstáculo para a realização da atividade?

Kniberg (2007) relata que sua equipe segue a risca a agenda diária de reuniões, elas são realmente realizadas em pé, para evitar o risco de ultrapassar o tempo determinado (15 minutos) e em frente ao quadro de tarefas. Caso haja algum impedimento, é tarefa do Scrum master administrar o problema e identificar uma solução.

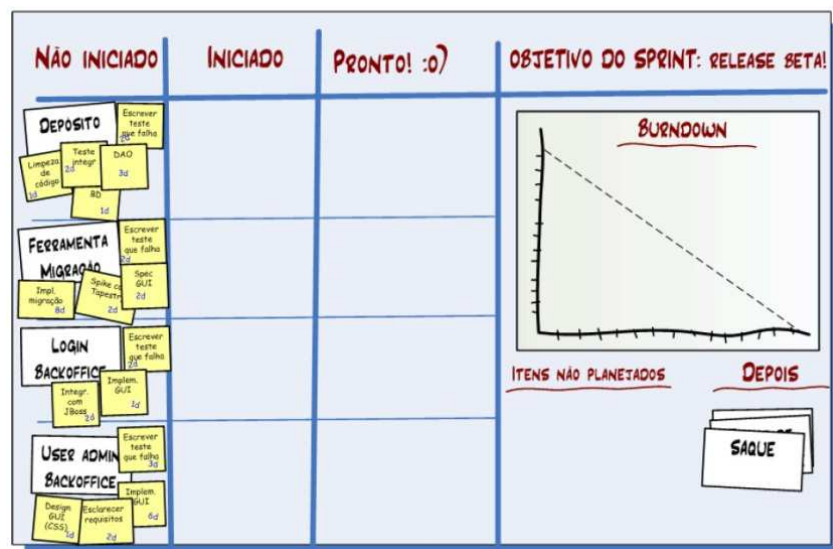
- Quadro de tarefas - o quadro de tarefas é proposto pelo autor como o método mais produtivo para controle do Sprint backlog. É um recurso do framework que permite que todos os membros da equipe possam visualmente ter a ideia real do

que está sendo feito, segundo o autor a atualização do quadro geralmente ocorre durante a reunião diária, enquanto cada pessoa responde os questionamentos propostos para a reunião. Neste momento itens não planejados também são incluídos no quadro e ajustes de estimativa de prazo também vão sendo atualizadas. O autor relata que algumas equipes usam o método de atualizar o quadro antes da reunião, o importante segundo ele é adotar um método e segui-lo.

Um exemplo de quadro de tarefas, pode ser visto na figura abaixo e suas colunas são:

- Não iniciado: são inseridas todas as atividades que ninguém irá realizar hoje.
- Iniciado: significa as atividades que alguém está executando hoje.
- Pronto: são os itens que não serão mais trabalhados.
- Itens não planejados: são itens não previstos no Sprint.
- Gráfico burndown: mostra o andamento da Sprint de acordo com a estimativa da equipe.
- Depois: são os itens que serão trabalhados após a conclusão de todos os itens do Sprint.

Figura 7 - Exemplo de quadro de tarefas



Fonte: KNIBERG (2007)

- Sprint Review Meeting – Reunião que ocorre após cada Sprint. Nesse momento o time discute seus erros, acertos e lições aprendidas. Para KNIBERG (2007), é

muito importante que esse momento aconteça, mesmo que as equipes estejam propensas a não fazê-lo, todos concordam que esse momento é de extrema importância. O autor destaca que esse momento é o segundo evento mais importante do Scrum (o primeiro é a reunião de planejamento do sprint). Esse é o momento onde todos os membros da equipe podem contribuir com suas ideias e percepções revisando seus erros e acertos e assim aprendendo e não repetindo as mesmas atitudes que geraram problemas. Mais uma vez o Scrum contribui para gerar conhecimento organizacional.

A finalização de um projeto Scrum requer a conclusão das atividades previstas no backlog e a validação de uma lista de critérios de aceitação definidos pelo Product Owner. Esses critérios de aceitação se referem aos itens que devem obrigatoriamente estar inseridos na entrega. (KNIBERG, 2007).

Para Schwaber (2011) o backlog do produto nunca está completo. Ele evolui tanto quanto o ambiente e o produto. Ou seja, é um ciclo de desenvolvimento, onde as entregas são realizadas na conclusão do sprint. O time deve ter um entendimento único do que significa o estado de “pronto” com relação ao projeto que estão atuando. A definição de “pronto” deve refletir que existe um incremento de produto apto a ser entregue. Essa mesma definição inclusive, orienta a equipe nas reuniões de planejamento de sprint. O propósito de pronto é a entrega de funcionalidades potencialmente utilizáveis.

Abaixo uma figura que mostra um resumo de todas as práticas do método Scrum.

Figura 8 - Visão geral da dinâmica do Scrum



Fonte: CARVALHO (2009)

2.1.3 Utilização do Scrum

Com o objetivo de aumentar a conscientização e compreensão do Scrum, promover melhorias necessárias para o contínuo sucesso do framework, foi criado o Scrum Alliance, que é uma organização profissional sem fins lucrativos que tem por missão promover o uso de Scrum de forma generalizada e prática. Essa entidade também certifica pessoas e empresas que possam repassar o conhecimento sobre o framework dentro dos padrões especificados.

Realizando um rápido resumo sobre a utilização do Scrum, abaixo são destacados pontos positivos e negativos para utilização do framework:

Kamei (2011) em seu artigo relata pontos positivos e negativos da implantação do Scrum nas Secretarias Estaduais de Fazenda do estado de Alagoas, sendo observados os seguintes aspectos como pontos positivos:

- melhor envolvimento do cliente;
- melhor comprometimento da equipe pelos resultados;
- melhoria de comunicação entre equipe e clientes;
- melhoria na qualidade da entrega dos produtos.

Como ponto negativo são destacadas as seguintes situações:

- resistência dos membros de alguns projetos inviabilizaram a implantação;
- falta de motivação e abertura a mudanças dos membros da equipe.

Santos Júnior (2009), descreve em seu artigo a experiência da implantação do Scrum em uma multinacional com processos certificados de qualidade (CMMI e ISO). A empresa adotou Scrum em 2006 objetivando mais agilidade além de manter a qualidade dos produtos de software. Neste exemplo os seguintes aspectos negativos podem ser relatados:

- falta de treinamento adequado a equipe quanto ao framework;
- modificações nos processos padrões do framework, como por exemplo, reuniões diárias sendo feitas dia sim e dia não, e utilizando um tempo maior que o indicado;
- não realização do product backlog por problemas na estimativa de tempo;
- erros no produto entregue;

- falta de atualização diária do Sprint backlog.

Mesmo relacionando pontos negativos, os seguintes pontos positivos foram apontados:

- time se sentia motivado pela liberdade de definir as atividades e pela agilidade;
- planilhas para controle do product backlog e Sprint backlog que permitiam maior controle sobre os requisitos e tarefas;
- time trabalhando de forma integrada e com facilidade de comunicação.

Além desses aspectos o autor declara em seu artigo que essas percepções eram relacionadas ao primeiro projeto da empresa utilizando a metodologia. Foi realizado um investimento em capacitação de um Scrum Master que repassou o conhecimento a outros membros da equipe, dessa forma, resolvendo os principais problemas que tinham sido encontrados na primeira fase do projeto. A organização aprendeu que a flexibilidade e o dinamismo da ferramenta ágil trazem benefício quando utilizadas com disciplina e após os ajustes realizados entregaram além dos requisitos previstos ainda outros requisitos adicionais ao software.

2.1.4 Extreme Programming (XP)

Astels (2002) acredita que tanto pessoas de negócio como desenvolvedores veem as técnicas tradicionais de desenvolvimento de software como muito lentas. O pessoal da área de negócios visualiza no software uma vantagem competitiva e vantagem competitiva não pode ser lenta.

O XP (eXtreme Programming) segundo Teles (2005), teve origem nas tradições do desenvolvimento em Smalltalk e datam de meados da década de 80 quando Kent Beck e Ward Cunningham trabalhavam na Tektronix. Nesta época, práticas como refatoração, programação em par, mudanças rápidas, feedback constante aos clientes, desenvolvimento iterativo, testes automatizados, entre outras, são elementos centrais da cultura da comunidade Smalltalk. Segundo o autor, seguindo este ponto de vista, o XP pode ser considerado o modo de agir do Smalltalk generalizado para outros ambientes. Alguns outros autores realizaram suas contribuições ao XP ao longo de uma década até que, em 1996 todas as partes começam a se juntar através de um trabalho realizado pelo idealizador Kent Beck na Chrysler. Neste projeto, conhecido por C3 ou Chrysler Comprehensive Compensation System, Kent utilizou

pela primeira vez o conjunto de práticas que formam atualmente a estrutura do eXtreme Programming.

Valores do XP, segundo Teles (2005):

- feedback;
- comunicação;
- simplicidade;
- coragem.

Práticas:

- cliente presente – nesta prática, talvez a mais extrema segundo Astels (2002), há a insistência de que se tenha um cliente real, uma pessoa que deve ser um dos usuários do sistema e que deve representar os demais usuários. Ela deve responder perguntas e tomar decisões relativas a prioridade de recursos, riscos, etc. É tarefa do cliente a descrição e priorização das estórias;
- metáfora – serve para que todas as pessoas envolvidas no projeto tenham uma mesma visão sobre o quadro geral do sistema e do problema a ser solucionado. (ALSTELS, 2002);
- jogo do planejamento – realização de um planejamento mínimo para se ter ideia do tamanho do projeto e do tempo, levando em consideração a capacidade da equipe;
- stand up meeting - assim como no Scrum, são reuniões que devem ser realizadas em pé a fim de serem rápidas e objetivas. São valiosos fóruns de comunicação e o momento da equipe dividir o que fez, o que fará e toda e qualquer outra situação que seja importante compartilhar com os demais;
- programação em par – cada código é escrito por um par de desenvolvedores. Um deles, o piloto, mantém o controle através do teclado e do mouse e o parceiro observa, buscando contribuir através de correção de erros de ortografia, sintaxe e oferecendo ideias e sugestões. (ASTELS, 2002);
- código coletivo – qualquer desenvolvedor pode ter acesso a qualquer fonte e modificá-lo sem que precise esperar para obter acesso a gravação. (ASTELS, 2002);
- código padronizado;
- design simples – ser simples é a ordem. O projeto deve ser simples e limpo;

- desenvolvimento orientado a testes – é uma das peças chaves da filosofia XP. Um dos momentos é o teste de unidade que é descrito pelo desenvolvedor antes mesmo dele realizar a codificação. É uma maneira de gerar documentação a respeito do comportamento desejado para o sistema, e também de dizer com certeza, quando o teste é aprovado, de que está pronta aquela funcionalidade;
- refatoração – significa a reescrever o código a fim de deixa-lo mais claro e simples. Astels (2002) citando Fowler, o autor descreve que essa é uma forma disciplinada de limpar o código e minimizar as chances de bugs;
- integração contínua – integração pelo menos diária do trabalho realizado ao código base da produção. Os testes garantem se você pode ou não liberar um novo release e se suas alterações não estão gerando conflito com as de outro desenvolvedor;
- releases curtos – realizar liberações menores e colocar a disposição do usuário uma funcionalidade o mais cedo possível;
- ritmo sustentável – ninguém trabalha bem se estiver cansado, sob pressão ou estressado. Empresas que utilizam XP devem trabalhar com semanas de 8 horas e preocupar-se com a qualidade de vida de seus colaboradores. A pressão e o estresse tão comuns em empresas de desenvolvimento de software, devem ser tratadas pelas práticas de planejamento existentes no método. (ASTELS, 2002).

Extreme Programming (XP) segundo CockBurn (2006) é um bom método ágil para ser analisado.

Comparando as descrições dos frameworks Scrum e XP, pode-se notar diversas práticas comuns e objetivos convergentes associados a entregas de valor, entregas rápidas e compartilhamento de conhecimento entre os membros da equipe. Dessa forma é possível admitir que a utilização de um dos modelos ou adoção de práticas mistas utilizando as duas metodologias possam auxiliar na solução de alguns dos problemas enfrentados atualmente neste estudo de caso.

2.2 MÉTODOS ÁGEIS X GERENCIAMENTO DE PROJETOS

Jim Highsmith (2004) vislumbrando a questão de gerenciamento de projetos, criou o APM (Agile Project Management) que segundo Benassi (2009), é um conjunto de práticas que auxiliam a equipe a entregar produtos e serviços de valor em ambientes complexos, instáveis e desafiadores. O APM surgiu também em 2001 junto com o movimento pelo

desenvolvimento ágil de software e vem a contribuir para um olhar diferenciado sobre gerenciamento de projetos, fugindo da maneira clássica, que tem foco no planejamento detalhado e é resistente a mudanças.

Os princípios do APM são definidos em duas categorias:

- Categoria de valor ao cliente:
 - entregas de valor;
 - entregas e iterações baseadas nas especificações do produto;
 - excelência técnica.

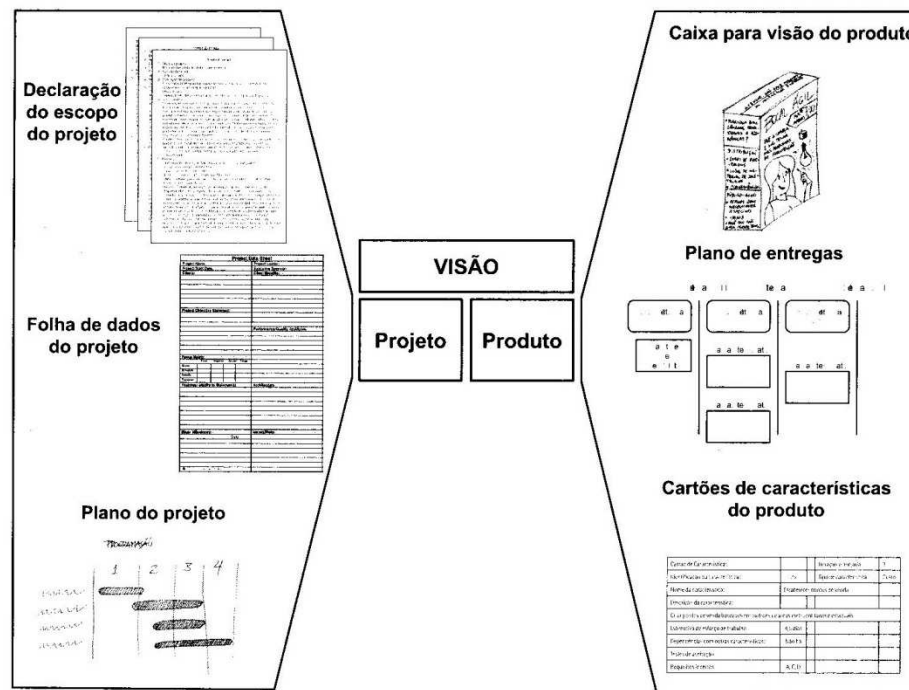
- Categoria de estilo de gerenciamento baseado na liderança e colaboração:
 - encorajar a exploração;
 - construir times adaptativos;
 - simplificação.

Highsmith (2004 apud DIAS, 2005) destaca a base para o gerenciamento ágil de projetos:

- inovação contínua: entrega de produtos atendendo os requisitos e agregando valor ao negócio. Inovação associada a um ambiente organizacional que estimule o autogerenciamento e a autodisciplina.
- adaptabilidade do produto: o desenvolvimento dos produtos deve ser pensando para resolver a necessidade atual, além de poder ser adaptável para uma necessidade futura.
- tempos de entrega reduzidos: foco, direcionamento e capacidade técnica da equipe visando a redução de prazos e utilizando melhores oportunidades de mercado.
- capacidade de adaptação do processo e das pessoas: formação de equipes onde as pessoas se sintam confortáveis com as mudanças, visualizando como desafio e não como problema. Estabelecimento de processos para responder rapidamente a mudanças.
- resultados confiáveis: entrega de produtos que agreguem valor e garantam a operação, suportando o crescimento e lucratividade da empresa.

A figura de Benassi (2009) demonstra de forma clara a diferença de como o projeto é visto em métodos ágeis comparada com o gerenciamento de projetos tradicional. O ágil, apesar de preocupar-se menos com documentação, tem cuidado especial em definir uma visão mais clara para o produto.

Figura 9 - Diferenças entre escopo e visão do projeto



Fonte: AMARAL (2011)

Dias (2005), explica que no gerenciamento clássico (ou tradicional) de projetos a atenção está voltada ao planejamento detalhado do projeto e aos processos formais de controle e monitoramento. Já o gerenciamento ágil de projetos, transfere a importância para a execução, seguindo os preceitos do manifesto ágil. A principal preocupação é entrega rápida e de valor, apresenta-se resultados durante toda a execução do projeto e são permitidas adaptações de escopo a cada iteração visando o alinhamento dos requisitos. Além disso, também destaca que gerenciamento de projetos tradicional, apesar dos ganhos obtidos, não se mostra eficaz quando utilizado para projetos de desenvolvimento de software. Essa diferença está relacionada ao ambiente de negócio bastante dinâmico e sujeito a mudanças constantes, fugindo dos padrões do gerenciamento clássico de projetos. Abaixo, um quadro adaptado por Dias (2005) que demonstra as principais diferenças entre os métodos clássico e ágil de

gerenciamento de projetos, baseada nos grupos de processo proposto pelo PMI e as fases do gerenciamento ágil de projetos.

Quadro 1 – Quadro de Comparação entre os métodos de gerenciamento de projetos

Grupo de processos do gerenciamento clássico de projetos	Fases do gerenciamento ágil de projetos
Iniciação: autorização de um novo projeto ou fase de delimitar o escopo preliminar do projeto	Visão: determinação da visão do produto e do escopo inicial do projeto
Planejamento: planejamento integral e detalhado do projeto	Especulação: desenvolvimento de um plano inicial do projeto, seguido do planejamento individual a cada iteração
Execução: execução do plano do projeto	Exploração: entrega de funcionalidades ou produtos previstos em cada ciclo
Monitoramento e Controle: controle do progresso do trabalho e gerenciamento de mudanças para minimizar os impactos no projeto	Adaptação: revisão dos resultados entregues e abertura de adaptações do escopo, visando o atendimento de novos requisitos do negócio
Encerramento: formalização do aceite final do projeto	Encerramento: aceites do cliente a cada ciclo ou iteração e formalização do encerramento do projeto ao final dos trabalhos

Fonte: DIAS (2005, p. 108-109)

O PMI em fevereiro de 2011, vislumbrando as mudanças de mercado através de suas pesquisas, implantou uma certificação voltada ao gerenciamento ágil de projetos: o PMI-ACP (Agile Certified Practitioner). Essa certificação foi lançada, segundo o PMI, para atender a crescente utilização de práticas ágeis no gerenciamento de projetos, indicada como fator determinante para obtenção de bons resultados.

Gerenciamento de projetos ágil está alinhado a utilização de um framework ágil para desenvolvimento de software, pois essas duas metodologias estão focadas no resultado rápido e de qualidade percebido pelo cliente e portanto, podem ser pensadas como alternativas para aprimorar o atendimento das demandas encaminhadas a área de desenvolvimento de sistemas da Empresa X, foco deste estudo de caso.

2.3 GESTÃO DE CONHECIMENTO

Gestão de conhecimento é o processo de geração de riqueza das organizações a partir de seu conhecimento ou de seu capital intelectual. (BUKOWITZ, 2002).

Para Boff (2001), a gestão do conhecimento é o conjunto de estratégias que visam criar, adquirir, compartilhar e utilizar ativos de conhecimento. De acordo com o autor, gerir conhecimento é estabelecer fluxos que garantam que a informação necessária será recebida no tempo e no formato adequado.

Cavalcanti (2000) argumenta que a economia do conhecimento desloca o eixo da riqueza dos setores industriais, onde matéria prima e mão de obra são fundamentais, para o eixo dos setores que trabalham com produtos e serviços intensivos em tecnologia e conhecimento.

Segundo Sordi (2013), a OECD – Organização para Cooperação e Desenvolvimento Econômico, destaca que 55% da riqueza mundial é gerada pelo conhecimento. Para a autora as organizações precisam desenvolver um ambiente favorável a aprendizagem organizacional e investir em capacitação, mudança cultural, criação de comunidades de prática e implantação de programas que auxiliem no aproveitamento e gestão do conhecimento de seus colaboradores.

2.3.1 Conhecimento Organizacional

Para Nonaka e Takeuchi (1997), conhecimento organizacional é a capacidade que uma empresa tem de criar, disseminar e incorporar conhecimento a produtos, serviços e sistemas.

De acordo com Angeloni (2005), nos dias atuais as organizações além de precisarem suprir as demandas por eficiência, eficácia e efetividade e competirem num mercado global, precisam pensar a gestão como elemento de conhecimento dentro das organizações, um estudo do conhecimento organizacional onde não se observam comportamentos prescritos e sim ações autônomas de mudança.

Para Nonaka e Takeuchi (1997), cultura organizacional é um padrão de pressupostos básicos que foram aplicados e tenham funcionado suficientemente bem a fim de que a organização passe a adotá-los e ensiná-los aos novos membros como a forma correta de perceber, pensar e sentir os problemas.

Segundo Angeloni (2005), cultura organizacional pode ser conceituada através de duas vertentes: mecanicista e holográfica. Na abordagem mecanicista a organização é composta

por elementos distintos, tais como: crenças, histórias, mitos, heróis, tabus, normas e rituais. Nesse conceito, acredita-se que a cultura possa ser manipulada a partir dessas variáveis. Na abordagem holográfica, a cultura organizacional é vista como reflexo do que integrantes entendem sobre a organização, apresentando a característica do todo.

Segundo Salmazo (2004 apud SANCHEZ, 1997, p. 5), conhecimento organizacional pode ser definido como “o conjunto compartilhado de crenças sobre relações causais mantidas por indivíduos dentro de um grupo”.

Consultando Angeloni (2005) sobre um modelo teórico para organizações do conhecimento, a autora cita Donald Schon (1971) e Jean Yves Prax (1997), ambos os autores trazem uma perspectiva histórica e apresentam uma visão ampliada sobre o conhecimento organizacional.

Donald Schon foi pioneiro a tratar organização social como sistema de aprendizagem, segundo ele todo sistema social é composto de uma estrutura, uma tecnologia e uma teoria. A estrutura trata das relações estabelecidas entre os indivíduos, os atributos básicos e os papéis a serem desempenhados. A tecnologia é o conjunto de normas, ferramentas e técnicas que visam otimizar atividades e alcançar metas. Tecnologia neste caso não pode ser vista apenas como recursos da informática, mas técnicas e modos conhecidos para implementar ações organizacionais. Teoria é o conjunto de regras de conhecimento por meio das quais se interpretam a realidade interna e externa do ambiente organizacional. Estas três dimensões apresentadas encontram-se sobrepostas e interagem com as demais. Eventuais modificações numa das dimensões gera efeito sobre as outras. (ANGELONI, 2005).

Na visão de Prax o conhecimento tem três dimensões de análise para que uma organização deixe de ser baseada no paradigma do controle para ser baseada no paradigma do conhecimento, são elas: dimensão do homem e seu conhecimento oral e escrito e as novas técnicas de comunicação das empresas. A dimensão da empresa e seu conhecimento organizacional de comunicação e o conhecimento coletivo. A dimensão de novas tecnologias, com seus recursos tecnológicos que favorecem a engenharia do conhecimento. Para que uma empresa seja voltada ao conhecimento são necessárias interações contínuas e recursivas dessas três dimensões. (ANGELONI, 2005).

A partir do estudo de Schon e Prax, Angeloni propõe um modelo alternativo para gestão do conhecimento levando em consideração a evolução do mundo organizacional. O modelo proposto por Angeloni é composto por três dimensões interagentes e interdependentes: infraestrutura organizacional, pessoas e tecnologia. Essa proposta parte da proposta de Prax com algumas variáveis que foram selecionadas para compor as dimensões

abordadas. Ou seja, o modelo proposto por Angeloni, usa como base os conhecimentos de Schon e Prax, mas algumas premissas essenciais divergem do modelo defendido por esses autores. Na visão de Angeloni, uma organização do conhecimento tem um equilíbrio entre as três dimensões constitutivas, não enfatizando um aspecto dentre os demais.

2.3.1.1 Infraestrutura organizacional

Deve-se desenvolver a infraestrutura organizacional a fim de que se tenha uma visão mais holística da organização. É importante trabalhar a cultura organizacional em busca de um estilo gerencial que não siga a linha do comando e do controle e sim de uma gestão mais participativa.

2.3.1.2 Pessoas

Angeloni (2005), baseada em Turner, afirma que aprender a desaprender é uma qualidade essencial aos colaboradores das organizações da era do conhecimento. É necessária uma visão sistêmica, e o líder assume o papel de incentivar a visão compartilhada e padrões sistêmicos de pensamento. O modelo de empresa onde a equipe diretiva pensa e a operária age, dá lugar ao novo modelo onde ideias e ações podem surgir em qualquer nível da organização.

2.3.1.3 Tecnologia

Todas as ferramentas tecnológicas, incluindo hardware e software utilizados para gerenciar o conhecimento organizacional acumulado e em desenvolvimento.

Segundo Angeloni (2005), as empresas precisam criar um ambiente propício à gestão do conhecimento. Empresas que buscam estar de acordo com a era do conhecimento, precisam incentivar seus colaboradores a estarem integrados à missão, visão e as estratégias da organização. Os colaboradores precisam se sentir parte e entender o valor de sua atividade para o atingimento dos objetivos da organização. Empresas de visão holística estimulam o espírito empreendedor de seus colaboradores, os colaboradores ficam livres para pensar e repensar no seu processo de trabalho e também no processo de tomada de decisão. Sai a figura do gerente autoritário, da rigidez funcional e hierárquica e há espaço agora para o líder, cuja principal habilidade é lidar com pessoas. As empresas da era do conhecimento não contratam

mais mão de obra, simples executores, contratam o que a autora chama de “cérebro de obra”, ou seja, pessoas que devem pensar, refletir, tomar decisões e depois executá-las.

Amaral (2011), afirma que tem crescido nas empresas brasileiras a busca por políticas e processos que visam gerenciar a produção, aquisição, descrição, armazenamento e uso da informação. Segundo ela, um colaborador bem informado é propenso a solucionar problemas do dia a dia das organizações e melhorar os processos organizacionais.

2.3.2 Tipos de conhecimento

Nonaka e Takeuchi (1997) definem conhecimento tácito como sendo o conhecimento pessoal e difícil de ser transferido. Já o conhecimento explícito é o conhecimento transmitido facilmente em linguagem formal ou sistemática. O quadro abaixo compara os tipos de conhecimento segundo o autor.

Nonaka e Takeuchi definem o conhecimento tácito como o mais importante, neste tipo de conhecimento inclui-se valores intangíveis como crenças pessoais e perspectivas e sistemas de valor. Acreditam que esse conhecimento seja o principal motivo de os ocidentais serem que os gerentes japoneses são um enigma.

Quadro 2 – Características de comparação de tipos de conhecimento

Conhecimento Tácito	Conhecimento Explícito
Subjetivo	Objetivo
Experiência	Racionalidade
Simultâneo	Sequencial
Analógico (prática)	Digital (teoria)

Fonte: SALMAZO (2013, p. 20)

Nonaka e Takeuchi (1997) indicam que o conhecimento tácito é à base de todo o conhecimento das organizações. É necessária a socialização do conhecimento tácito como objetivo de compartilhá-lo. Eles definem o chamado “espiral do conhecimento” como um modelo de interação contínua entre conhecimento tácito e explícito. Defendem que essa interação gera um ciclo positivo para a criação do conhecimento organizacional.

Figura 10 - Espiral do conhecimento



Fonte: NONAKA; TAKEUCHI (1997)

2.3.3 Liderança nas organizações do conhecimento

Hoje as organizações não precisam mais de um gerente tradicional, mas sim de um gerente adaptado a nova realidade moderna. O papel que essa pessoa deve desempenhar é um papel de liderança. Isso significa que o processo de gerenciamento moderno passa a ser visualizado como um conjunto de interações humanas, e é esse papel que deve ser desempenhado nas organizações de alto desempenho ou organizações do conhecimento. Esse papel de liderança reflete nas atitudes dos gerentes que nas organizações tradicionais veem sua responsabilidade como a de dar ordens. Nas organizações do conhecimento, a responsabilidade do líder é facilitar o processo, os líderes não dão ordens eles ajudam no sucesso dos colaboradores. Nas organizações tradicionais os gerentes se importam apenas com seu departamento, nas organizações do conhecimento os gerentes se preocupam com toda a companhia. Nas organizações do conhecimento, também muda o papel do colaborador que passa a não ser mais um cumpridor de ordens vindas de seu gerente, que sozinho precisa planejar e dividir as tarefas. Nas organizações do conhecimento os colaboradores executam todas as tarefas em conjunto, por meio de um trabalho em equipe, qualquer um pode falar com qualquer pessoa, sobre qualquer assunto. (PEREIRA, 2008).

A partir da revisão bibliográfica de métodos ágeis e de gestão de conhecimento, pode-se constatar que esses assuntos são interligados, pois métodos ágeis buscam a valorização de

peças e a troca contínua de experiência, especialmente essas duas questões vão muito ao encontro de organizações do conhecimento, que definem que as empresas precisam criar um ambiente favorável para gerar conhecimento, que precisam valorizar o espírito de equipe e o compartilhamento das informações.

Adotar essas duas práticas nas organizações deve alavancar os resultados, gerando respostas mais rápidas e maior qualidade dos produtos e serviços, dessa forma, podem facilitar e melhorar ainda mais a qualidade de atendimento da Empresa X, foco desse estudo de caso.

3 MÉTODO

Este capítulo descreve o método de pesquisa selecionado, a unidade de estudo e indica as técnicas de coleta e análise de dados utilizados. Ao final, são explicadas algumas limitações do método.

3.1 DELINEAMENTO DA PESQUISA

Nesta pesquisa utilizou-se o método de estudo de caso. De acordo com Yin (2010), não existe uma fórmula que defina quando utilizar ou não o método de pesquisa através de estudo de caso. Mas o autor indica que para responder questões que procuram explicar circunstâncias presentes, por exemplo, “como” ou “por que”, esse método de estudo se mostra relevante. Diante disso e do tema do trabalho, foi escolhido o método de estudo de caso para esse projeto de pesquisa.

A pesquisa realizada será descritiva. Esse método, segundo Gil (1991), tem como objetivo principal a descrição das características de uma determinada população ou fenômeno ou o estabelecimento de relação entre variáveis. Além disso, segundo o mesmo autor, pesquisas descritivas podem também servir para proporcionar uma maior visão do problema, o que as aproxima de pesquisa exploratória e o que parece estar mais associado ao problema que está sendo estudado.

3.2 DEFINIÇÃO DA UNIDADE DE ANÁLISE

A empresa na qual está sendo realizado este estudo de caso é a Empresa X, mais precisamente no setor de desenvolvimento de sistemas. A ideia desse estudo de caso surgiu da necessidade da Empresa X em alinhar o atendimento de suas demandas de sistemas com os objetivos estratégicos da organização e melhorar a percepção dos solicitantes sobre os serviços prestados.

3.3 TÉCNICAS DE COLETA DE DADOS

Para a coleta de dados desse estudo serão utilizadas as seguintes técnicas:

- a) análise de documentação: serão analisados documentos do setor que instituem a forma com a qual as demandas são encaminhadas e a metodologia de atendimento

existente. Esse documento é chamado Plano de Análise e Desenvolvimento de Sistemas, revisado em 2012 e nele será analisada a situação atual de atendimento. Conforme Yin (2010), essa técnica tem como pontos fortes: estabilidade, discrição, exatidão e ampla cobertura, como pontos fracos: recuperabilidade, seletividade, parcialidade e acesso. Tendo em vista que o acesso a essas informações foi liberado, selecionei essa técnica visando enriquecer o trabalho;

- b) registros em arquivo: como já é utilizado há bastante tempo um sistema *web* para envio das demandas do setor, os registros de inclusão dessas demandas, do último ano, serão consultadas e analisadas visando categorizá-las e dessa forma deixar ainda mais personalizada e assertiva as mudanças que serão propostas. Esses registros são autênticos e exatos, mas revelam apenas a situação atual do ambiente, no entanto, se assemelham a outros formatos de recebimento de demanda utilizados por outras empresas, por isso, podem ser utilizados para o estudo e não ferem as observações feitas por Yin (2010) em seu livro quando se refere a utilização de registros em arquivo;
- c) entrevistas: serão realizadas entrevistas com dois gestores do setor de TI (coordenador e gerente de TI), buscando identificar sua visão sobre pontos fortes e fracos e suas pretensões de futuro. Também serão ouvidos cinco gestores dos setores demandantes, buscando identificar a visão atual deles sobre o atendimento e oportunidades de melhoria. Para escolha desses gestores, será levada em consideração, após análise dos registros em arquivo e análise de documentos internos, os setores que mais demandam serviços e quais as principais demandas encaminhadas. Será escolhida a técnica de entrevista em profundidade e essa escolha baseia-se na busca de *insights* para o trabalho. Serão observadas as orientações de Yin (2010), buscando formular questões que façam com que o entrevistado na verdade seja um informante, contribuindo com suas ideias para a realização da pesquisa. As entrevistas terão como base o roteiro que consta no apêndice.

3.4 TÉCNICAS DE ANÁLISE DOS DADOS

A partir das diversas fontes que serão utilizadas para a coleta de dados e delineamento da pesquisa, pretende-se ter uma ideia mais clara e determinar os principais problemas que serão tratados neste trabalho.

Todas as informações colhidas serão analisadas e com isso serão selecionados os itens mais importantes para serem trabalhados, levando em consideração é claro a expectativa dos gestores entrevistados.

3.5 LIMITAÇÕES DO MÉTODO

Essa pesquisa está limitada ao tipo de ambiente de desenvolvimento descrito como objeto de estudo. Há também a limitação das informações que serão liberadas pela coordenação do setor sobre o levantamento e o histórico de suas demandas e a contribuição dos gestores quanto às respostas colhidas através das entrevistas realizadas.

4 APRESENTAÇÃO E ANÁLISE DOS DADOS

Para o desenvolvimento deste trabalho foram realizadas entrevistas focadas nos gestores de TI e nos gestores das áreas de negócio, que em geral demandam a maior parte do trabalho da área de TI, especificamente relacionadas ao desenvolvimento de sistemas. Para cada grupo de gestores foi criado um questionário específico, a fim de obter informações julgadas pertinentes para identificação da situação problema. Nesses questionários procuramos identificar as principais queixas vislumbrando a opinião das duas partes envolvidas: solicitantes e área técnica. Os questionários podem ser vistos no apêndice.

Os devidos questionários foram aplicados aos dois gestores de TI e a cinco (5) gestores das áreas de negócio. A escolha dos gestores das áreas de negócio foi baseada em dois critérios: o primeiro diz respeito ao grau de importância de seu setor perante ao planejamento estratégico da IES e o segundo diz respeito ao número de solicitações feitas pela área de sua responsabilidade.

Dos 9 gestores convidados a participar, apenas 4 tiveram disponibilidade e foram entrevistados no prazo estipulado. Para dispor de um número maior de entrevistados, foi adicionado um gestor da área de planejamento estratégico da Empresa X, visualizando que sua participação poderia contribuir em função da ligação desta área com a área de revisão de processos.

No quadro abaixo são apresentadas as 09 (nove) áreas que mais demandaram solicitações no ano de 2012 junto ao setor de desenvolvimento, servindo de base para a seleção dos gestores conforme descrito acima. Neste período, o setor de desenvolvimento recebeu 5298 solicitações.

Quadro 3 – Solicitações enviadas 2012

continua...

Qtd. Sol. Enviadas	Setor que envia	% Sobre Total
1246	REGISTRO ACADÊMICO	23,5183088
497	FINANCEIRO	9,380898452
354	EXTENSÃO FORMAÇÃO CONTINUADA	6,681766704
328	REGISTRO ACADÊMICO	6,191015478
272	DEPARTAMENTO DE MARKETING	5,134012835

... continuação

260	AGÊNCIA DE TALENTOS	4,907512269
246	CONTROLADORIA	4,643261608
191	RECURSOS HUMANOS	3,605134013
168	ATENDIMENTO	3,171007928
166	PROGRAD	3,133257833

Fonte: Sistema de gestão da Empresa X

4.1 ÁREA DE TI

Das entrevistas realizadas com os gestores de TI puderam ser feitas as seguintes constatações, quanto ao setor de desenvolvimento:

- setor necessita de uma melhor priorização de atividades. Nas entrevistas realizadas é citada a existência de um movimento de conscientização com as áreas solicitantes para que essas priorizem as suas demandas, pois o setor de desenvolvimento muitas vezes não conhece o negócio das áreas a fim de poder identificar qual deve ser primeiramente atendida em detrimento das outras. O objetivo desta priorização é melhorar a organização do setor de desenvolvimento frente as suas demandas;
- o setor esta aberto a implantação de metodologias que possam contribuir com as atividades. É vista de forma muito otimista a implantação de metodologias que possam auxiliar a equipe em seu trabalho diário e que conquistem o time;
- o setor necessita gerir melhor o conhecimento dos colaboradores. Os coordenadores foram questionados quanto a isso e relatam que existe fragilidade, pois depende muito da pessoa responsável pelo processo a construção de documentação que possa ser usada no futuro pelos demais integrantes do setor. Também foi identificada a necessidade de existir mais momentos quanto a troca de informações, a fim de permitir que o conhecimento tácito dos membros do time seja compartilhado, pois a documentação por melhor que seja feita, não consegue transmitir o conhecimento adquirido durante o desenvolvimento da funcionalidade;
- existe pré-disposição dos colaboradores a realização de melhorias. Isso é visualizado no setor pois ocorre o movimento dos próprios colaboradores na busca de melhorias para o processo e produtividade. Os colaboradores trazem

constantemente sugestões de melhoria, sendo uma delas a implantação de metodologias ágeis;

- colaboradores necessitam de uma visão sistêmica sobre as demandas. Isso é uma preocupação tanto das áreas de negócio, quanto da área de desenvolvimento. Deve ser criada uma cultura que favoreça a disseminação de conhecimento e visão de negócio das áreas para toda equipe, a fim de permitir que estes não se atenham puramente ao que está sendo solicitado, mas que possam visualizar a interligação dessa demanda com as demais funcionalidades do sistema, evitando assim possíveis consequências quanto ao que está sendo entregue.

4.2 ÁREAS DE NEGÓCIO

Os gestores das áreas de negócio trouxeram nas entrevistas os seguintes pontos que merecem ser destacados:

- reconhecem a importância do setor de desenvolvimento de sistemas, considerando-a fundamental para execução de suas atividades diárias;
- entendem que o trabalho executado pelo setor é de qualidade, sendo a principal queixa relacionada ao tempo de atendimento, especialmente nas demandas de melhoria, explicadas posteriormente;
- entendem como necessária, e estão dispostos a participar e colaborar com todo o processo de desenvolvimento do sistema, desde seu início;
- percebem que mudanças no processo de atendimento do setor de desenvolvimento trouxeram melhorias;
- solicitam urgência em várias demandas, mas não auxiliam o setor de desenvolvimento quanto a organização destas prioridades.

4.3 HISTÓRICO

A percepção dos gestores da área de TI, com relação ao descontentamento das áreas de negócio no atendimento de suas solicitações, fez com que o método de distribuição das demandas recebidas sofresse algumas modificações nos últimos anos, conforme descrito abaixo:

Num primeiro momento, a equipe estava dividida entre help desk, analistas e desenvolvedores. Help desk é a nomenclatura que foi dada a equipe responsável pelo atendimento de demandas mais gerais. A ideia era que esse atendimento fosse rápido e o tipo de atendimento realizado deveria estar concentrado em criação de pequenos relatórios, consultas e auxílio quanto a dúvidas de usuários, entre outros problemas que exigissem um menor gasto de tempo. A alocação, que é o processo de destinar uma demanda a um determinado colaborador, era realizada pela equipe de help desk. Desta forma, a todo o momento eram alocadas novas demandas, com novas prioridades, sem período de corte de recebimento. O critério adotado para alocar, tomava por base o domínio das aplicações que cada colaborador era responsável.

Este funcionamento gerava muito descontentamento. A equipe de desenvolvimento recebia a todo o momento novas atividades e ficava confusa quanto ao que atender primeiro. Cada colaborador possuía o seu repositório de solicitações, o qual teria que atender, não havendo um repositório centralizado com todas as demandas do setor. Dessa forma crescia o repositório de solicitações atrasadas espalhadas entre os colaboradores, pois além de não saber ao certo qual a prioridade de atendimento, acabavam muitas vezes cedendo à pressão recebida pelas áreas de negócio, que possuíam acesso direto a esses colaboradores.

Em virtude dos problemas relatados, a cerca de dois anos um novo formato foi adotado. Foi mantida a equipe de help desk na avaliação das demandas, mas essa passou a analisar as solicitações juntamente com a uma equipe de analistas sêniores. Também foi criado um período de corte, a fim de restringir as demandas a serem avaliadas diariamente e poder assim alocar a equipe. As demandas encaminhadas após esse período passaram a ser verificadas na próxima avaliação. O período de corte ainda é utilizado atualmente, sendo que para ele foi estipulado um tempo de 24h. A análise diária passou a considerar as solicitações encaminhadas até às 14h, horário esse em que ocorria a reunião diária dos analistas com a equipe de help desk.

Com isso, notou-se então um crescimento do repositório de solicitações na equipe de help desk, que além de prestar suporte imediato, também desenvolvia relatórios e pequenas manutenções no sistema. Resultado: equipe de help desk ficou sobrecarregada de trabalho, não dando conta de suporte imediato.

Com o objetivo de corrigir essa sobrecarga de trabalho na equipe de help desk e estando ciente da contribuição que os métodos ágeis trariam a essa realidade caótica de recebimento de demandas, buscou-se dividir a equipe em times. Deixando de existir a figura

de um help desk único e passando a existir times que desempenham todas as atividades necessárias a um determinado sistema ou grupo de sistemas.

Esses times atualmente são compostos de analistas, desenvolvedores e estagiários. Existem tamanhos distintos entre os times, mas em geral possuem no máximo 05 pessoas. Sendo que a ideia inicial eram 02 analistas, 02 desenvolvedores e 01 estagiário por time.

O processo de alocação continua ocorrendo diariamente junto com a reunião diária de análise das solicitações. Mas a composição da reunião agora é de dois representantes de cada time, sendo um analista. As solicitações são encaminhadas entre os componentes do time, ou ficam com o analista para um melhor entendimento/acompanhamento. Na sequência, cada time realiza uma rápida reunião para discutir as demandas recebidas e situação de demandas que estão sendo atendidas.

Essas mudanças possibilitaram uma maior discussão e disseminação de conhecimento entre as equipes em virtude das reuniões que ocorrem diariamente, e que possibilita a troca de ideias e conhecimento entre os participantes. Neste formato é possível uma análise melhor sobre as demandas, que pode até barrar algumas solicitações que possam gerar retrabalho, por serem solicitadas por pessoas muitas vezes sem visão de negócio necessária. Holderegger (2010) cita a falta de comprometimento dos colaboradores com os objetivos centrais das IES, que leva a uma disputa de interesse, principalmente quando uma demanda envolve mais de uma área. Como geralmente falta formalismo pela baixa documentação de processos e falta de objetividade nas solicitações, é comum a ocorrência de atrasos, erros e recursos financeiros e operacionais mal aplicados.

Nesse estudo, foi possível identificar que os gestores das áreas de negócio não veem positivamente a centralização do encaminhamento de demandas em um de seus colaboradores. Acreditam que isso possa travar/burocratizar demais o processo. Podemos destacar que pode não ser necessária a centralização de solicitações de demandas através de um colaborador de cada equipe, mas é preciso tomar cuidado neste ponto, pois Holderegger (2010), alerta que é comum em IES os colaboradores das áreas de negócio não se responsabilizarem por uma definição ou até mesmo encaminharem demandas de última hora por esquecimento. Segundo o autor, esse é o fator gerador do grande número de solicitações urgentes, e a formalização e a documentação dos pedidos faz com que os solicitantes no mínimo busquem respaldo daquilo que estão pedindo. Quanto ao formalismo da documentação das solicitações, a IES foco deste estudo de caso já está preparada, utilizando o site de solicitação, onde são gerenciadas as demandas e que já foi explicado o funcionamento neste trabalho. Quanto à riqueza do detalhamento das solicitações feitas, bem como a urgência

pretendida no atendimento destas, são questões a serem trabalhadas e fazem parte da proposta de implantação de um framework de metodologia ágil, descrito posteriormente.

As mudanças que foram realizadas e relatadas acima no formato de atendimento praticado pela área de desenvolvimento de sistemas já trouxeram benefícios que foram percebidos pelos usuários e foram relatados nas entrevistas, mas mesmo assim, ainda continua ocorrendo um repositório de solicitações espalhadas entre os diversos times e outro repositório denominado de melhorias. Outro problema que persiste diz respeito à demora no atendimento de solicitações, esse fato pode resultar em um produto ou funcionalidade não mais condizente com as necessidades do solicitante.

Atualmente, durante a reunião de análise das demandas e alocação, demandas maiores e que não estavam previstas para serem desenvolvidas, são alocadas em um repositório específico nomeado de melhorias. Neste repositório, aguardam até que possam ser atendidas. Melhorias é uma nomenclatura que foi utilizada para separar as demandas que estão aguardando atendimento. Além dessa classificação, também é possível determinar o processo/sistema ao qual essa melhoria está associada.

Foi relatado pelos gestores de TI, que as demandas de melhorias ficam armazenadas em uma planilha, sendo agrupadas em separado, distribuídas por sistemas que são utilizados na Empresa X. Essa planilha serve de base para que novos grupos de demandas sejam montados. A escolha de qual sistema será colocado em produção é feita em uma reunião semanal que ocorre com todos os coordenadores de área juntamente com a pró-reitoria administrativa. Durante essa reunião, no momento em que se constata uma demanda mais complexa ou de maior envolvimento das equipes, como por exemplo, a reestruturação de um sistema, que é uma demanda que consome mais tempo e que utiliza mais recursos, há a indicação de que primeiramente se encaminhe o assunto com o setor de planejamento estratégico, para que seja realizado o levantamento e descrição de todo o processo. Somente após essa etapa é que a documentação é enviada para o setor de desenvolvimento e é iniciada a análise do sistema. As demandas de melhoria também são distribuídas aos times de desenvolvimento, sendo atendidas concomitantemente com as demais demandas.

A partir da descrição desse cenário atual, pode-se notar que diversos movimentos para adoção de melhores práticas e melhor gerenciamento de demandas já foram iniciados, mas observa-se que muito ainda pode ser feito.

Da opinião dos gestores, puderam ser extraídas as seguintes constatações que reforçam que o momento é propício para a continuidade desse processo:

- existe a percepção de que as mudanças estão trazendo benefícios;
- existe a visão da necessidade de participação integral na construção dos projetos;
- existe descontentamento quanto a prazo de atendimento das demandas, principalmente quanto às demandas de melhorias;
- existe descontentamento quanto à priorização das demandas.

4.4 PROPOSTA PARA IMPLANTAÇÃO DA METODOLOGIA ÁGIL

Diante das constatações sobre as necessidades apontadas tanto pela área de desenvolvimento, quanto pelas áreas de negócio entrevistadas, é possível notar uma forte aderência ao uso de metodologia ágil na Empresa X. A aplicação de metodologia ágil pode contribuir nos seguintes pontos:

- esta metodologia é indicada para ambientes complexos, dinâmicos, onde requisitos mudam constantemente e com grande volume de solicitações. É indicada como o melhor caminho para aumento de produtividade, que repercute em rentabilidade e competitividade organizacional. (HOLDEREGGER, 2010);
- é uma metodologia calcada no trabalho em equipe, permitindo a melhoria na comunicação e ampliando a colaboração. Permite que cada um faça o seu melhor e se sinta bem com o que faz, isso com o tempo eleva a produtividade. (HOLDEREGGER, 2010);

No entanto é preciso observar alguns pontos que podem impactar e servem de premissa para a implantação de métodos ágeis (MELO, 2010):

- a) apoio gerencial;
- b) escolha do método e das práticas;
- c) realização de projeto piloto;
- d) educação e suporte ao time.

Quanto a apoio gerencial, o mesmo foi constatado através das entrevistas, tanto em nível de área de desenvolvimento, quanto em nível das áreas de negócio. É importante agora que esses líderes se incluam nesse novo formato de trabalho, respeitando o tempo que será definido para o atendimento das demandas, contribuindo com as informações necessárias para

o seu atendimento e com a priorização das atividades. É importante que a cultura da organização passe a funcionar de forma holística, onde todos desempenhem seus papéis da melhor forma possível e visualizem sua participação como fundamental, olhando para o todo e tendo certeza de que não existe um único responsável pelo sucesso ou fracasso no atendimento das demandas, todos os envolvidos serão. Também cabe as lideranças a criação de um ambiente favorável à troca de conhecimento, onde os colaboradores possam partilhar seus conhecimentos tácitos, a fim de contribuir com a aprendizagem organizacional e ainda com a criação de produtos de maior qualidade, prática essa incentivada pela metodologia ágil.

A sugestão deste trabalho é a implantação do framework Scrum. Os motivos desta escolha são explicados abaixo e se baseiam no autor Holderegger (2010):

- é a metodologia mais indicada para o tipo de ambiente do estudo de caso. Dinâmico, intenso, e com continuas mudanças de direção e prioridades;
- pode ser aplicada tanto em projetos pequenos como em projetos grandes;
- adapta-se constantemente ao “caos” de interesses e necessidades.

Para a implantação do framework do Scrum é sugerido que o mesmo seja primeiramente implantado em uma das equipes, e num projeto piloto. Para a realização de um projeto piloto, é indicada a escolha de um sistema não crítico e sem restrições severas de prazo. (MELO, 2010). Como o Scrum visa o envolvimento das pessoas, é necessário o envolvimento da equipe em todo o processo de transição. Dessa forma é interessante que a própria equipe escolha o projeto que servirá como piloto. A escolha pode ser feita a partir de uma reunião com a participação de todos os membros da equipe.

O Scrum também favorece a disseminação do conhecimento, a partir das diversas reuniões previstas durante e após a conclusão de uma iteração, o que vai ao encontro das necessidades do setor de gerir melhor o conhecimento dos colaboradores, constatado nas entrevistas realizadas com os gestores da área de desenvolvimento. Além disso, ele respeita os preceitos de organizações do conhecimento no sentido de valorizar as pessoas, de apoiar o trabalho em equipe, de permitir que os colaboradores façam parte das decisões, do entendimento do projeto como um todo e, conseqüentemente, entendendo melhor a organização. O envolvimento de diversas pessoas com papéis e responsabilidades diferentes também favorece o pensamento holístico sobre as necessidades da organização e não apenas para solução específica de um problema isolado do seu setor.

Outro aspecto importante para a escolha do Scrum, diz respeito à dificuldade enfrentada pela Empresa X quanto à priorização de demandas, fato esse ressaltado pelos gestores das áreas de negócio e desenvolvimento de sistemas. Para o uso do framework será necessário que a organização eleja um ou mais colaboradores com conhecimento em negócio e na área de sistemas, para que possam desempenhar o papel de Product Owner (PO). É desse papel a responsabilidade por criar e manter o product backlog, onde deverão constar todas as demandas da Empresa X frente ao setor de desenvolvimento, já devidamente priorizadas, em função do valor que estas representam ao negócio.

Para garantir a observação do item educação e suporte ao time, é necessária a escolha de um Scrum Master e de investimento em capacitação dessa pessoa. Kamei (2011) e Santos Júnior (2009). Também é necessário investir na capacitação da equipe, a fim de familiarizá-los com os conceitos do desenvolvimento ágil de software e com o framework sugerido. Essa familiarização pode ser individual num primeiro momento, buscando que as pessoas já tenham um conhecimento prévio e possam realizar um melhor aproveitamento de uma capacitação formal, que posteriormente é realizada pelo colaborador que vai desempenhar o papel de Scrum Master. Essa capacitação interna garante a disseminação do conhecimento da metodologia e serve para preparar e mobilizar a equipe a fim de torna-los motivados a aderirem as práticas e a disciplina necessária para utilização do Scrum. Melo (2010), Kamei (2011), Santos Júnior (2009).

Mesmo observando esses pontos, a autora Melo (2010) relata que a adoção de métodos ágeis nas organizações é lenta e complexa. Em função disso a proposta de iniciar com um projeto piloto. A autora constatou que a principal dificuldade esta relacionada a cultura organizacional, toda a organização precisa pensar e agir de forma ágil e não apenas a área de desenvolvimento de sistemas. Para isso é muito importante que o apoio gerencial seja cumprido e que além da área de desenvolvimento, todas as demais áreas que encaminham solicitações para a área de desenvolvimento sejam informadas e estejam aptas a participar do processo de implantação de métodos ágeis. Dessa forma, é necessária a capacitação dos membros das áreas de negócio que devem participar ativamente na obtenção e priorização das demandas, juntamente com o product owner. Para o projeto piloto o ideal é que a escolha do product owner seja feita em conjunto com as áreas de negócio do projeto que será executado, fazendo com que esse tome ciência de seu envolvimento e da importância do seu papel frente à equipe, frente às áreas que representa e frente à implantação da metodologia ágil.

Bassi Filho (2008) defende o tripé de conhecimento, onde são destacados três ingredientes essenciais: conhecimento técnico, visão de negócios e vivência com o problema,

ele acredita que precisam ser identificados os detentores desses conhecimentos e garantir que eles tenham o envolvimento na medida certa dentro do projeto.

Alguns pontos importantes para a implantação de métodos ágeis, podemos citar que já estão sendo trabalhados:

Conforme indicado por Holderegger (2010), deve existir um formulário simplificado, utilizado para a formalização do envio da solicitação. A Instituição já possui esse formulário, através do site de solicitação e a cultura de envio de demandas através dessa ferramenta já está disseminada. Acredita-se que essa ferramenta de envio de solicitação tem praticamente todos os recursos para uma melhor comunicação com os solicitantes. Apenas podem ser sugeridas melhorias na classificação das demandas no momento que são enviadas. Fazendo com que os solicitantes indiquem o projeto e classifiquem o atendimento de acordo com sua necessidade:

- erro que impede continuidade do processo;
- erro que não impede a continuidade;
- dúvida esclarecimento;
- desenvolvimento de melhoria;
- suporte à aplicação.

A partir das sugestões de classificação acima, o PO terá melhores condições para avaliar e discutir a importância da demanda frente ao negócio da Empresa X, conseguindo atribuir o valor de negócio adequado a ela, fazendo com que os solicitantes tenham mais clareza quanto ao prazo de atendimento. De acordo com o valor de negócio atribuído a demanda, a mesma poderá ser atendida o mais rápido possível, como por exemplo, no próximo sprint da equipe. Assim, durante o andamento do sprint, novas demandas podem ser adicionadas ao product backlog da equipe, porém somente no final do sprint que esta em andamento, as mesmas serão avaliadas, pois nesse momento é que irá ocorrer uma nova revisão de prioridades do que está no product backlog e com isso, pode-se analisar e determinar quais demandas serão adicionadas ao novo sprint de acordo com as prioridades selecionadas. Assim, pode-se evitar que os colaboradores que se comprometeram com o sprint backlog em andamento, recebam novas demandas, preservando a execução das atividades previstas no sprint.

Outro item que é importante de ser destacado é a visão da Empresa X em descrever seus processos. Esse ponto forte deve ser mantido e reforçado, pois segundo Holderegger

(2010), a partir da documentação de processos o desenvolvimento de sistemas de informação torna-se mais rápido, aderente e de baixo custo.

5 CONCLUSÃO

Faz-se necessária agora a apresentação desse projeto de implantação do framework Scrum aos interessados na Empresa X. Essa é uma das limitações desse trabalho. Foi realizado o estudo de caso e agora é necessário que a empresa decida se deseja ou não implantar as melhorias que foram indicadas. Para isso, devem ser observados os pontos relevantes que foram levantados, ressaltando:

- cultura da organização;
- envolvimento de toda a equipe;
- e o apoio gerencial.

A implantação de métodos ágeis na Empresa X pode melhorar a percepção dos solicitantes quanto aos serviços, especialmente por estarem recebendo de forma mais rápida, melhorias nos sistemas de acordo com as suas necessidades e prioridades, dessa forma, pretende-se elevar o grau de satisfação dos solicitantes.

As prioridades, por estarem relacionadas aos itens de maior valor ao negócio, trarão benefícios rapidamente aos solicitantes, que também terão uma ideia real da data de entrega das demandas. Além disso, participando de todo o processo e recebendo entregas parciais, é possível ter uma noção adequada do produto final, inclusive podendo realizar mudanças de requisitos que tragam maior valor. Esse fato traz fundamental valor a implantação, principalmente quando estamos falando de ambientes dinâmicos como o estudado.

É importante ressaltar que foram previstos também os principais pontos que podem impactar negativamente, desta forma, destaca-se a necessidade de realizar um trabalho preventivo a fim de minimizar esses pontos.

Pela importância da participação de toda a equipe, foram previstas capacitações e envolvimento de todos desde a tomada de decisão quanto ao projeto piloto. Acredita-se que é fundamental, além dessas capacitações, o apoio das lideranças a fim de criar uma cultura favorável para a implantação de metodologias ágeis e de troca de conhecimento entre os membros da equipe.

É necessário prever a adoção de um método de avaliação da implantação da metodologia, analisando os resultados do projeto piloto. Com base nestes resultados, é importante a estruturação de um ciclo de melhoria contínua do método, buscando sempre

atingir melhores patamares. Essa é uma das premissas do Scrum, não existe rigidez quanto aos processos, sendo possível a adaptação frequente até atingir os resultados esperados.

É recomendada também a adoção de metodologia ágil para gerenciamento de projetos e não adoção da metodologia tradicional. Esse passo também contribui com o sucesso da implantação do framework e está alinhado com os objetivos da Empresa X.

Como projeto futuro, é apropriado o estudo e adoção de CMMI. O CMMI pode complementar o que está sendo proposto por sua abordagem de melhoria contínua de processos.

REFERÊNCIAS

ALVES, C. F. M. **Gestão da tecnologia da informação nas instituições de ensino superior**. Disponível em: http://tede.unifacs.br/tde_busca/arquivo.php?codArquivo=23. Acesso em: 31 jul. 2013.

AMARAL, Cléa Gomes do. **Gestão do conhecimento: como fazer bom uso do capital intelectual dos profissionais nas organizações**. Disponível em: http://www.pucminas.br/iec/iec_com_voce/materia.php?codigo=7367&PHPSESSID=416f1244cb6cbaec909c18cc6f2dc5bc. Acesso em: 27 jun. 2013.

AMARAL, D. C. et al. **Gerenciamento ágil de projetos: aplicação em produtos inovadores**. São Paulo: Saraiva, 2011.

ANGELONI, M. T. **Organizações do conhecimento: infra-estrutura, pessoas e tecnologias**. São Paulo: Saraiva, 2005.

ASTELS, D. G. M. **Extreme programming: guia prático**. Rio de Janeiro: Campus, 2002.

BASSI FILHO, Dairton Luiz. **Experiências com desenvolvimento ágil**. Dissertação de mestrado. Disponível em: <http://www.teses.usp.br/teses/disponiveis/45/45134/tde-06072008-203515/pt-br.php>. Acesso em: 28 jul. 2013.

BENASSI, J.L.G. **Avaliação de modelos e proposta de método para representação da visão do produto na gestão ágil de projetos**. Dissertação de mestrado. Disponível em: <http://www.teses.usp.br/teses/disponiveis/18/18140/tde-06072009-162430/pt-br.php>. Acesso em: 02 jul. 2013.

BOFF, L. H. **Gestão do conhecimento**. Porto Alegre: Programa de Pós-Graduação em Computação. Notas de aula, 2001.

BUKOWITZ, W. R. **Manual de gestão do conhecimento: ferramentas e técnicas que criam valor para a empresa**. São Paulo: Bookman, 2002.

CARVALHO, B. V.; MELLO, C.H.P. **Aplicação de método ágil scrum no desenvolvimento de produtos de software em uma pequena empresa de base tecnológica**. Disponível em: http://www.scielo.br/scielo.php?pid=S0104-530X2012000300009&script=sci_arttext. Acesso em: 29 jun. 2013.

_____.; _____. **Revisão, análise e classificação da literatura sobre o método de desenvolvimento de produtos ágil SCRUM**. Anais do SIMPOI. 2009. Disponível em: http://www.simpoi.fgvsp.br/arquivo/2009/artigos/E2009_T00109_PCN92031.pdf. Acesso em: 15 jun. 2013.

CAVALCANTI, Marcos. **A nova riqueza das organizações: os capitais do conhecimento**. Disponível em:

<http://portal.crie.coppe.ufrj.br/portal/data/documents/storedDocuments/%7B93787CAE-E94C-45C7-992B-9403F6F40836%7D/%7B4348A601-CD08-4972-B822-D00D470C0D4C%7D/A%20Nova%20Riqueza%20das%20Organiza%C3%A7%C3%B5es.pdf>. Acesso em: 27 jun. 2013.

COCKBURN, A. **Agile software development: the cooperative game**. 2. ed. New Jersey: Safari, 2006.

COUTO, A. B. **CMMI – integração dos modelos de capacitação e maturidade de sistemas**. Rio de Janeiro: Ciência Moderna Ltda., 2007.

DIAS, M. V. B. **Um novo enfoque para o gerenciamento de projetos de desenvolvimento de software**. Dissertação de mestrado. Disponível em: <http://www.teses.usp.br/teses/disponiveis/12/12139/tde-03012006-122134/pt-br.php>. Acesso em: 02 jul. 2013.

DICIONÁRIO do Aurélio online. 2008-2013. Disponível em: <http://www.dicionariodoaurelio.com/demanda.html>. Acesso em: 10 ago. 2013.

GIL, A. C. **Como elaborar projetos de pesquisa**. São Paulo: Atlas, 1991.

HIGHSMITH, J. **Agile Project Management**. Boston: Addison-Wesley, 2002.

HOLDEREGGER, R. JR. A.C. **A dinâmica do desenvolvimento de projetos de sistemas de informação na instituição de ensino superior privada: uma proposta de metodologia**. Disponível em: <http://www.centropaulasouza.sp.gov.br/pos-graduacao/workshop-de-pos-graduacao-e-pesquisa/anais/2010/Trabalhos/gestao-e-desenvolvimento-de-tecnologias-da-informacao-aplicadas/Trabalhos%20Completo/HOLDEREGGER,%20Ricardo.pdf>. Acesso em: 31 jul. 2013.

KAMEI, F. K. et al. **Scrum no serviço público: um relato de implantação nas secretarias estaduais da fazenda e da gestão pública do estado de Alagoas**. Disponível em: <http://www.aedb.br/seget/artigos11/36914380.pdf>. Acesso em: 28 jul. 2013.

KNIBERG, H. **Scrum e XP direto das trincheiras: como nós fazemos Scrum**. 2007. Disponível em: <http://infoq.com/br/minibooks/scrum-xp-fromthe-trenches>. Acesso em: 03 jan. 2013.

MALLMANN, P. R. **Um modelo abstrato de gerência de software para métodos ágeis**. Dissertação de mestrado. Disponível em: <http://biblioteca.asav.org.br/vinculos/tede/PauloRoberoMallmannComputacao.pdf>. Acesso em: 10 ago. 2013.

MARÇAL, A. S. C. **SCRUMMI: um processo de gestão ágil baseado no Scrum e aderente ao CMMI**. Dissertação de mestrado. Fortaleza 2009.

MELO, C. de O. et al. **Métodos ágeis no Brasil: estado da prática em times e organizações.** Relatório técnico RT – MAC – 2012 – 03. São Paulo: Departamento de Ciência da Computação. IME – USP, 2012.

_____; FERREIRA, G. R. M. **Adoção de métodos ágeis em uma instituição pública de grande porte.** Disponível em:
http://ccsl.ime.usp.br/agilcoop/files/WBMA_Melo_e_Ferreira.pdf. Acesso em: 29 jul. 2013.

NONAKA, Ikujiro. TAKEUCHI, Hirotaka. **Criação de conhecimento na empresa.** 20. ed. Rio de Janeiro: Elsevier, 1997.

PEREIRA, M. F.. A gestão organizacional: em busca do comportamento holístico. In: ANGELONI, Maria Terezinha (Org.). **Organizações do conhecimento.** 2. ed. São Paulo: Saraiva, 2008. v. 1.

PRESSMAN, R. S. **Engenharia de software.** Porto Alegre: AMGH, 2010.

RISING, L.; JANOFF, N.S. **The scrum software development process for small teams.** Disponível em: <http://web.lindarising.info/uploads/IEEEScrum.pdf>. Acesso em: 29 jun 2013.

SALMAZO, Paulo Rogério. **Um estudo em gestão do conhecimento para uma empresa de tecnologia da informação.** Disponível em: <http://hdl.handle.net/10183/5970>. Acesso em: 27 jun. 2013.

SANTOS JÚNIOR, Antonio F. dos; SANTOS, R. P.. **Aspectos sociotécnicos do desenvolvimento de software utilizando scrum em um caso prático.** Disponível em:
<http://www.uspleste.usp.br/jporto/woses2009/anais/04-WOSES-2009.pdf>. Acesso em: 28 jul. 2013.

SANTOS, Rildo F. **Scrum Experience.** Disponível em:
<http://www.etecnologia.com.br/scrum/Scrum%20Experience%20%5BO%20Tutorial%20SCRUM%5D%20v16.pdf>. Acesso em: 25 jul. 2013.

SCHWABER, K. **Agile project management with Scrum.** Washington: Microsoft Press, 2004.

_____. **Scrum guide.** 2011. Disponível em:
<http://static.squarespace.com/static/51e3f87ce4b0031a73dac256/t/51f1711de4b06fe045e0764f/1374777629920/Scrum%20Guide%20-%20Portuguese%20BR.pdf#zoom=100>. Acesso em: 10 ago. 2013.

SOARES, M. S. Metodologias ágeis extreme programming e scrum para o desenvolvimento de software. **Revista eletrônica de sistemas de informação.** v. 3, n. 1, 2004. Disponível em:
<http://revistas.facecla.com.br/index.php/reinfo/article/view/146/0>. Acesso em: 10 ago. 2013.

_____. **Publicado no Journal Of Computer Science**, v. 3, n. 2, 2004. Disponível em: <http://www.dcc.ufla.br/infocomp/artigos/v3.2/art02.pdf>. Acesso em: 25 jul. 2013. falta o título da obra

SORDI, Neide Alves Dias de. **Como a gestão do conhecimento pode contribuir para o desenvolvimento econômico**. Disponível em: <http://www.innovagestao.com.br/2013/03/como-a-gestao-do-conhecimento-pode-contribuir-para-o-desenvolvimento-economico/>. Acesso em: 27 jun. 2013.

SOUZA, C. A.; ZWICKER, R. **Um modelo de ciclo de vida de sistemas erp: aspectos relacionados à sua seleção, implementação e utilização**. Disponível em: http://www.ead.fea.usp.br/semead/4semead/artigos/mqi/Souza_e_Zwicker.pdf. Acesso em: 31 jul. 2013.

TELES, V. M. **Um estudo de caso da adoção das práticas e valores do extreme programming**. Disponível em: <http://desenvolvimentoagil.com.br/xp/dissertacaoXP.pdf>. Acesso em: 30 jun. 2013.

WEILL, P.; ROSS, Jeanne W. **Conhecimento em TI. O que os executivos precisam saber para conduzirem com sucesso a TI em suas empresas**. São Paulo: M. Books do Brasil, 2010.

YIN, R. K. **Estudo de caso: planejamento e métodos**. Porto Alegre: Bookman, 2010.

APÊNDICE A - ROTEIROS DE ENTREVISTAS

Roteiro de entrevista que será aplicada aos gestores do setor de TI buscando identificar as reais necessidades na visão deles:

- 1) Como você avalia a estrutura de trabalho do seu setor atualmente? Quais são os principais problemas e as maiores virtudes?
- 2) Quais as principais mudanças já realizadas em sua gestão?
- 3) Qual delas trouxe mais retorno positivo?
- 4) Quais não trouxeram benefícios?
- 5) Quais são as maiores demandas frente às equipes de desenvolvimento?
- 6) Como são gerenciadas as prioridades vindas dessas demandas?
- 7) Existe pressão por parte dos setores demandantes para entrega das soluções?
- 8) Você acredita que a implantação de métodos ágeis possa auxiliar no gerenciamento das demandas do setor?
- 9) Você acredita que seja válida a implantação de CMMI?
- 10) O que é feito hoje no quesito de gestão de conhecimento?
- 11) Quando um colaborador sai da empresa, é possível dizer que os demais membros da equipe a qual esse pertencia, terão o conhecimento necessário para continuar a evolução, construção, suporte, etc.?
- 12) O que pode ser melhorado?
- 13) Você acha prudente a implantação de um plano periódico de solicitações de demandas de *software*. Esse plano seria algo parecido com o atual plano de investimentos, onde os setores administrativos poderiam incluir suas necessidades de *software* para os próximos períodos (semestre/ano) e esses pedidos seriam selecionados e aprovados?
- 14) Como você acredita que seja a visão dos clientes internos sobre o setor?
- 15) Você acredita que a alta direção aceitaria e apoiaria mudanças que trouxessem desconforto aos clientes internos. Como, por exemplo, a implantação deste plano de investimentos em *software* e também a centralização e priorização de solicitações sob responsabilidade de um determinado colaborador do setor?

Roteiro de entrevista que será aplicada aos gestores dos setores demandantes buscando identificar as reais necessidades na visão deles:

- 1) Quando e como você faz solicitações ao setor de desenvolvimento de TI?

- 2) Qual a importância desse setor para a Instituição no seu ponto de vista?
- 3) Qual a importância desse setor para o seu setor no seu ponto de vista?
- 4) Qual a sua percepção sobre o atendimento prestado pelo setor?
- 5) O que você considera adequado?
- 6) O que você gostaria que mudasse?
- 7) Como são feitas as prioridades de demandas solicitadas pelo seu setor?
- 8) As funcionalidades do sistema solicitadas sempre atendem a todas as regras de negócio?
- 9) Quem do seu setor participa da construção dos requisitos e regras de negócio das funcionalidades solicitadas?
- 10) Como você visualizaria a centralização das suas demandas através de um membro da sua equipe?
- 11) Como você visualiza a participação dos membros da sua equipe na construção efetiva e validação das funcionalidades?
- 16) Como você visualiza a implantação de um plano periódico de solicitações de demandas de *software* (plano de investimento em *software*). Esse plano seria algo parecido com o atual plano de investimentos, onde os setores administrativos poderiam incluir suas necessidades de *software* para os próximos períodos (semestre/ano) e esses pedidos seriam selecionados e aprovados?