



**UNIVERSIDADE ESTADUAL DO NORTE DO PARANÁ**  
**CAMPUS LUIZ MENEGHEL**

**ANDRESSA APARECIDA MANGOLIN ROSA**

***SCRUM E KANBAN PARA O GERENCIAMENTO DE***  
**COMUNICAÇÃO EM EQUIPES DE**  
**DESENVOLVIMENTO DE *SOFTWARE***

Bandeirantes  
2014

**ANDRESSA APARECIDA MANGOLIN ROSA**

***SCRUM E KANBAN PARA O GERENCIAMENTO DE  
COMUNICAÇÃO EM EQUIPES DE  
DESENVOLVIMENTO DE SOFTWARE***

Trabalho de Conclusão de Curso submetido ao *Campus* Luiz Meneghel da Universidade Estadual do Norte do Paraná, como requisito parcial para a obtenção do grau de Bacharel em Sistemas de Informação.

Orientador: Prof. Wellington Ap. Della Mura

Coorientador: Prof. Me Thiago Adriano Coleti

Bandeirantes  
2014

**ANDRESSA APARECIDA MANGOLIN ROSA**

***SCRUM E KANBAN PARA O GERENCIAMENTO DE  
COMUNICACAO EM EQUIPES DE  
DESENVOLVIMENTO DE SOFTWARE***

Trabalho de Conclusão de Curso submetido à *Campus* Luiz Meneghel da Universidade Estadual do Norte do Paraná, como requisito parcial para a obtenção do grau de Bacharel em Sistemas de Informação.

**COMISSÃO EXAMINADORA**

---

Prof. Wellington Ap. Della Mura  
Universidade Estadual do Norte do Paraná

---

Prof. Me. Thiago Adriano Coleti  
Universidade Estadual do Norte do Paraná

---

Prof. Dra. Daniela de F. G. Trindade  
Universidade Estadual do Norte do Paraná

Bandeirantes, 25 de Junho de 2014

## **AGRADECIMENTOS**

Primeiramente a Deus, a nossa sra. Aparecida que colocou as pessoas certas, nos momentos certos, pela força, coragem, e enfim por mais essa benção em minha vida de poder concluir uma graduação.

Ao meu orientador Wellington Della Mura, pela dedicação, paciência, pelos conselhos, atenção em todos os momentos e em cada reunião realizada uma nova aprendizagem não só referente ao trabalho mas também na vida, uma pessoa que podemos contar em todos os momentos, e ao coorientador Thiago Adriano Coleti que não somente abriu as portas de sua empresa para realizar o estudo, como também foi um grande amigo sempre disposto a ajudar no que precisa-se.

A minha família, pela confiança, força e motivação. E Principalmente a meus pais, pelo apoio, por me ajudar nesse período e atenção de madrugada quando preparava uma xícara de leite para eu levar pra estudar. Ao meu irmão que estimo muito que sempre me levou onde precisava.

Ao meu namorado Felipe que sempre esteve em todas horas, sempre me apoiou e acreditou em mim, que me ajudou a estudar, teve paciência de me ensinar o que sabia e que abdicou de muita coisa, só pra me ajudar, torcendo por mim todo esse tempo, para que realizasse todos meus sonhos. Obrigada Felipe você foi muito compreensível nas minhas ausências, e sempre incentivou a pensar grande e ir atrás dos meu sonhos.

Ao Estevan Braz Brandt Costa que já trabalhei junto, e também foi meu professor na graduação e pela dedicação, amizade, paciência e bom humor contribuiu para alguns dias nublados de trabalho fosse os melhores possíveis.

Aos meus amigos que sempre estiveram presentes em todos os momentos desta minha jornada, a minha amiga Cristiane Martins pela grande amizade e carinho, ao meu amigo José Mário Cheirubin também colega de sala que sempre é um grande conselheiro e mostra o lado bom das coisas.

Aos meus professores do curso, pela aprendizagem adquirada e principalmente paciência e dedicação nesta etapa tão importante de minha vida.

A banca de Defesa com a Daniela de Freitas Guilhermino Trindade, pela contribuição para o aprimoramento deste estudo e sempre disposta a conversar e ajudar.

Muitas pessoas que fizeram parte desta jornada contribuíram para que esse sonho se tornasse realidade, agradeço de coração toda a aprendizagem adquirida nesses anos. A todos que, com boa intenção, colaboraram para a realização e finalização deste trabalho. Confesso que muitos momentos pensei em desistir. Não foi fácil mas hoje vejo que também não era impossível e que no final dá tudo sempre certo.

“Eu não sou tão forte quanto previa, nem tão fraca como eu temia. Não tenho o passo rápido como eu gostaria, nem paraliso como poderia. Aprendi a me equilibrar nos extremos. Se eu não tenho o direito de escolher todos os acontecimentos, me posiciono de acordo com os fatos. No final, o que me move não é o forte o suficiente para me derrubar, mas é intenso o bastante para me fazer ir além”. (*Fernanda Gaona*)

## RESUMO

A comunicação tem uma grande importância nas organizações, é algo indispensável e um dos maiores problemas enfrentados pelas empresas de desenvolvimento de software. Este trabalho apresenta um estudo de caso numa empresa de desenvolvimento de software. O estudo descreve a utilização de duas Metodologias Ágeis, combinando o *Scrum* e o *Kanban* para ajudar a equipe de desenvolvimento a gerenciar a comunicação. A partir dos problemas detectados na empresa, pode-se ver como a empresa estava e a partir disto implantar os métodos. Foi ainda criado um modelo de processo que pudesse representar como seria o “passo a passo” de uma solicitação de um cliente para empresa. Neste sentido, este trabalho tem como objetivo minimizar os problemas de comunicação que ocorria na equipe de desenvolvimento.

**Palavras-chave:** Metodologia Ágil; Comunicação; *Kanban*; *Scrum*.

## **ABSTRACT**

Communication is very important in organizations, it is something essential and one of the biggest problems faced by software development companies. This paper presents a case study in a software development company. The study describes the use of two Agile Metodologias, combining Scrum and Kanban to help the development team to manage the communication. From the problems identified in the company, you can see how the company was and from this deploy methods. It was also created a process model that could represent what would "step by step" of a request from a customer to the company. Thus, this work aims to minimize the communication problems that occurred in the development team.

**Palavras-chave:** Agile; Communication; *Kanban*; *Scrum*.

## LISTA DE ILUSTRAÇÕES

Figura 1 - Modelo do Processo de Comunicação (fonte: FREIRE, 2012) .....	17
Figura 2 - Gráfico de Burndown (fonte: KNIBERG e SKARIN, 2009).....	28
Figura 3 - TaskBoard (fonte: KNIBERG e SKARIN, 2009).....	29
Figura 4 - Modelo simples de sinalizador visual <i>Kanban</i> . (fonte: MARIOTTI, 2008) .	31
Figura 5 - Comparação entre os quadros (fonte: KNIBERG e SKARIN, 2009). .....	33
Figura 6 - Ciclo de vida do BPM (APBMP, 2009).....	37
Figura 7 - Exemplo de processo criado no BizAgi.....	40
Figura 8 - Modelo de Processo de Gerenciamento de Comunicação .....	51
Figura 9 - Subprocesso de Análise de Prioridade .....	51
Figura 10 - Quadro <i>Kanban</i> implantado na empresa .....	53

## LISTA DE TABELAS

Tabela 1 - Comparação entre os Métodos (fonte: KNIBERG e SKARIN, 2009) .....	32
Tabela 2 - Descrição dos principais objetos de modelagem.....	38

## LISTA DE ABREVIATURAS E SIGLAS

<b>BPM</b>	<i>Business Process Management</i>
<b>BPMN</b>	<i>Business Process Modeling Notation</i>
<b>BPMS</b>	<i>Business Process Management System</i>
<b>FDD</b>	<i>Feature Driven Development</i>
<b>LSD</b>	<i>Lean Software Development</i>
<b>PO</b>	<i>Product Owner</i>
<b>XP</b>	<i>Extreme Programming</i>
<b>WIP</b>	<i>Work in process</i>

## SUMÁRIO

LISTA DE ILUSTRAÇÕES .....	9
1 INTRODUÇÃO.....	12
1.1 Contextualização .....	12
1.2 Formulação e Escopo do Problema .....	13
1.3 Justificativa .....	14
1.4 Objetivos.....	14
1.4.1 Objetivo Geral .....	14
1.4.2 Objetivos Específicos .....	14
1.5 Organização do Trabalho .....	15
2 FUNDAMENTAÇÃO TEÓRICA .....	16
2.1 Comunicação.....	16
2.1.1 Elementos da comunicação .....	16
2.1.2 Problemas que envolvem a falha de Comunicação.....	17
2.2 Métodos Ágeis .....	18
2.2.1 Scrum.....	20
2.2.2 Kanban.....	29
2.3 Scrum e Kanban na comunicação.....	32
2.4 Gestão de Processo BPM .....	35
2.4.1 BPMN .....	38
2.4.2 BizAgi .....	38
3 METODOLOGIA .....	41
4 DESENVOLVIMENTO.....	42
4.1 Elaboração do Questionário de Identificação de Problemas de Comunicação.....	42
4.2 A JoinSoft .....	44
4.3 Aplicação do Questionário.....	45
4.4 Treinamento das Metodologias e Implantação dos Métodos Ágeis .....	46
4.5 Modelo de Processo de Gerenciamento de Comunicação .....	48
4.5.1 Planejamento e Análise.....	49
4.5.2 Desenho e modelagem de processos de negócio.....	50
4.6 RESULTADOS OBTIDOS .....	51
5 CONSIDERAÇÕES FINAIS.....	54
REFERÊNCIAS .....	55

APÊNDICE A .....	58
APÊNDICE B .....	62
APÊNDICE C .....	64
Elementos do Processo.....	65
Elementos do Subprocesso.....	66
ANEXO A.....	69
RELATÓRIO DA EMPRESA .....	70

# 1 INTRODUÇÃO

## 1.1 Contextualização

A comunicação se tornou um marco histórico na evolução humana, devido à necessidade de conviver com outros da mesma espécie. Pode-se dizer que comunicação é um processo de troca de informação e conhecimento entre as pessoas.

Essencial em uma organização, a comunicação muitas vezes é esquecida na prática. Um dos grandes problemas no processo de desenvolvimento de software é a falta de comunicação dentro da equipe, que resulta não só em conflitos internos e externos, mas também no atraso da entrega do produto para o cliente.

Segundo Martins e Misaghi (2012) falhas na comunicação ou a falta de comunicação afetam diretamente um projeto, podendo resultar no seu sucesso ou até mesmo no seu fracasso. No decorrer do projeto podem ocorrer muitos problemas relacionados a comunicação entre a equipe de desenvolvimento e com um cliente. Por exemplo, uma pessoa da equipe podem esquecer de comunicar fato importante, uma solicitação que julga simples, uma alteração de código, alteração na agenda da visita ao cliente, dentro outras.

As equipes de projeto necessitam discutir ideias, levantar as vantagens e desvantagens de suas estratégias, para encontrar, assim, melhores soluções para a resolução de problemas (Trindade, 2008).

O envolvimento dos participantes não pode ser ignorado, falhas de comunicação entre o grupo pode comprometer o desempenho da equipe e o sucesso do projeto.

De acordo com Trindade (2008) a comunicação deve ser planejada, executada e controlada com muito rigor e comprometimento, de forma a garantir que os objetivos de grupos de trabalhos sejam atingidos de maneira natural e transparente.

Na busca pela eficiência, qualidade e lucro as empresas desenvolvedoras de software estão à procura de estratégias que possam melhorar o gerenciamento de sua equipe e do seu produto.

Os Métodos Ágeis surgiram como uma alternativa às metodologias tradicionais, que eram consideradas pesadas e extensas demais, devido a grande massa de documentação. A Metodologia Ágil inova, pois consigo traz a eficiência da equipe e do gerenciamento do projeto, mantendo a organização e diminuindo o retrabalho. As Metodologias Ágeis podem acrescentar muita a organização de uma equipe de desenvolvimento, e principalmente, a comunicação entre os membros, reduzindo problemas e os conflitos. Trindade(2008) afirma que “um fator importante é reconhecer que a comunicação gera novos compromissos e ações a serem executadas as quais poderão interferir na continuidade dos projetos”.

## 1.2 Formulação e Escopo do Problema

Muitos das dificuldades que ocorrem no decorrer do projeto são relacionados a comunicação entre a equipe de desenvolvimento. A ausência de comunicação pode acarretar vários problemas como:

- Resistência às mudanças (Martins; Misaghi, 2012);
- Conflito de ideias (Prikladnicki, 2004);
- Omissão de informação (Prikladnicki, 2004).

O presente trabalho aborda a comunicação e os problemas que elas podem provocar em empresas de desenvolvimento de software. As práticas criadas neste trabalho foram aplicadas em um estudo de caso real na empresa JoinSoft , que possui alguns problemas de comunicação entre a equipe. A implantação utiliza a Metodologia Ágil *Scrum* com o suporte do *Kanban*.

Kniberg e Skarin(2009) relatam que o *Scrum* e o *Kanban* são, ambos, altamente adaptativos, mas o *Scrum* é mais prescritivo que *Kanban*, ou seja o *Scrum* lhe dá mais restrições, por conta disso, deixa menos opções abertas.

O objetivo deste trabalho é analisar o que a aplicação das metodologias resultou no ambiente organizacional e demonstrar como as metodologias podem contribuir para melhorar a comunicação dentro da equipe. Assim, buscou elaborar uma estratégia utilizando as Metodologias Ágeis como apoio na gerencia da equipe de desenvolvimento.

Também foi possível, medir a evolução e o desempenho da equipe, ao final do estudo, mostrando as vantagens e desvantagem que as metodologias resultaram na empresa.

### **1.3 Justificativa**

Visto que a comunicação é, sem dúvida, de extrema importância para uma equipe de desenvolvimento, esta pesquisa justifica-se pela necessidade de uma estratégia de organização e gerenciamento da comunicação em uma equipe de desenvolvimento de software.

### **1.4 Objetivos**

#### **1.4.1 Objetivo Geral**

Este trabalho tem como objetivo aplicar a metodologia de gestão estratégica combinada com as Metodologia Ágeis, para solucionar falhas de comunicação dentro de uma equipe de desenvolvimento de software.

#### **1.4.2 Objetivos Específicos**

- Estudar os principais conceitos das Metodologias Ágeis e comunicação, realizar um estudo aprofundado sobre as metodologias *Scrum* e *Kanban*;
- Descrever falhas de comunicação enfrentadas em uma empresa de software;
- Estudar os fatores que ocasionam falhas de comunicação;
- Diagnosticar as dificuldades de comunicação encontradas;
- Explicar o funcionamento das *frameworks Scrum* e *Kanban*;
- Implantar o framework na empresa;
- Analisar o impacto dos resultados e demonstrar a evolução da equipe de desenvolvimento;
- Descrever o desempenho;
- Relatar as pontos positivos e negativos da implantação do framework;

## 1.5 Organização do Trabalho

O trabalho está estruturado da seguinte forma o capítulo 2 descreve a fundamentação teórica, dividida em: Comunicação, Metodologias Ágeis e *Scrum* e *Kanban* na comunicação, Gestão de Processos BPM. No capítulo 3 é apresentada a metodologia desta pesquisa. No capítulo 4 será apresentado o desenvolvimento pesquisa começando pela Elaboração do Questionário de Identificação dos Problemas, A Empresa JoinSoft, Aplicação do Questionário, Treinamento das Metodologias, Implantação dos Métodos Ágeis, Modelo de Processo de Gerenciamento de Comunicação e Resultados Obtidos. No capítulo 5 são apresentadas as Considerações Finais e Trabalhos Futuros.

## 2 FUNDAMENTAÇÃO TEÓRICA

### 2.1 Comunicação

Devido a evolução humana a comunicação se tornou indispensável em todos ambientes. A comunicação é um processo de troca de informação e conhecimento entre as pessoas.

A comunicação pode ser definida como uma transmissão de uma mensagem codificada por um emissor através de um canal para ser decodificada por um receptor (Torres, 2013).

Para assegurar que a mensagem seja entregue corretamente é necessário antes de tudo um planejamento do que seja proferido, um propósito de forma que as pessoas que serão comunicadas tenham entendimento do que foi dito, para que não haja falha na compreensão.

#### 2.1.1 Elementos da comunicação

Comunicação é o ato de compartilhar informações entre duas ou mais pessoas, com a finalidade de persuadir ou de obter um entendimento comum a respeito de um assunto ou de uma situação (Terciotti, 2009).

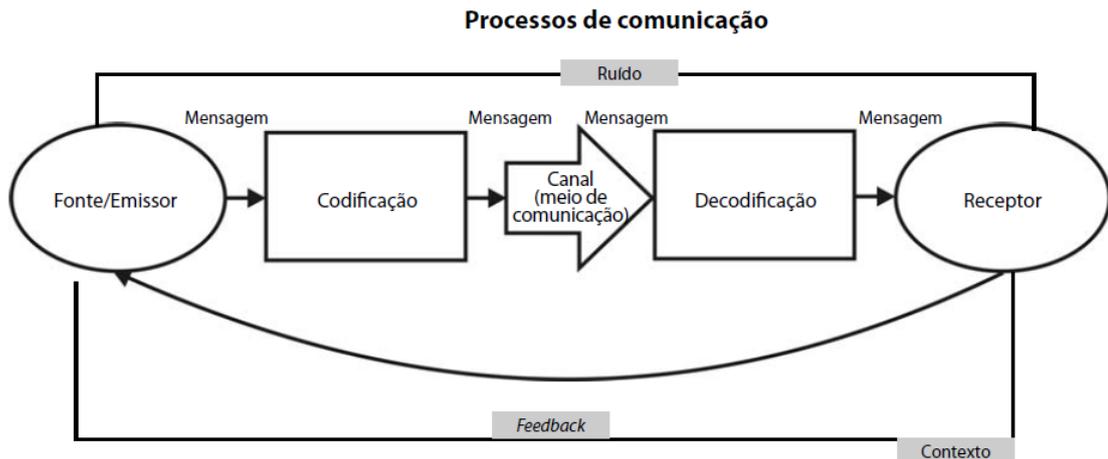
Segundo Arantes (1998) citado por Carvalho e Mirandola(2007), a comunicação é a base para a sobrevivência, o crescimento e a continuidade das organizações. O processo de comunicação está dividido em Emissor, Codificação, Mensagem, Canal, Decodificação, Receptor e o *Feedback*.

Torres (2013) descreve da seguinte forma o processo de comunicação:

- Emissor: Inicia a mensagem codificando um pensamento;
- Codificação: Forma da mensagem transmitida pelo emissor;
- Mensagem: Produto físico decodificado pelo emissor;
- Canal: É o caminho por onde a mensagem é transferida;
- Receptor: É a quem a mensagem é transmitida;
- Decodificação: Interpretação da mensagem pelo receptor;
- Ruído: Barreira da comunicação que distorce a clareza da mensagem;

- **Feedback:** Verificação do sucesso na transmissão de uma mensagem, como originalmente pretendia. Neste momento sabe-se se houve compreensão da mensagem.

Na Figura 1 é ilustrado o processo de comunicação.



**Figura 1 - Modelo do Processo de Comunicação (fonte: FREIRE, 2012)**

### 2.1.2 Problemas que envolvem a falha de Comunicação

No processo de comunicação pode haver algumas barreiras impedindo a execução processo. As barreiras à comunicação podem ser tanto físicas como de significado ou psicológicas. Ferreira e Serra(2006) apud de Carvalho e Mirandola (2007).

As barreiras físicas são obstáculos no processo da transmissão da informação e barreiras de psicológicas são relacionadas às variáveis interpessoais entre emissor / receptor e à utilização de códigos não partilhados.

Os principais métodos da comunicação de acordo com Torres (2013) são a comunicação oral, comunicação escrita e a comunicação verbal, porem apresentam pontos de dificuldades:

- **Comunicação Oral:** Por mais rápidas que sejam tem as suas desvantagens, pois quanto maior o número de receptores, maior será a chance de distorções. Por exemplo, a brincadeira do “telefone sem fio”,

onde cada pessoa interpreta a mensagem da sua própria maneira e quando chega ao final esta totalmente modificada.

- Comunicação escrita: As vantagens são óbvias, vai desde a forma de armazenar até a possibilidade de consultar o documento no caso de dúvidas, mas as desvantagens também existem. A Comunicação escrita consome mais tempo do que a oral e não há um *feedback* instantâneo e talvez em nenhum momento.
- Comunicação não verbal: Este método de comunicação é enviada com a oral, mas também pode ser enviada sozinha. Por exemplo, um olhar, um sorriso, um movimento de sobrancelhas e etc. A desvantagem da comunicação não verbal está no receptor se confundir no que realmente o emissor deseja, pois ele diz algo, mas sua feição ou movimentos “dizem” outra.

## 2.2 Métodos Ágeis

As Metodologias Ágeis surgiram como uma alternativa as abordagens tradicionais, por serem pesadas demais ou orientadas a planejamento.

Conhecidas inicialmente como metodologias leves, as Metodologias Ágeis surgiram como uma resposta à necessidade de mudanças na forma de gerir projetos de desenvolvimento de software e processos envolvidos nestes projetos. (Taniguchi; Correa, 2009).

Segundo Tanaka e Banki, 2008:

Em 2001, com propósito de mudança no processo de desenvolvimento, um grupo de profissionais resolveram projetar um modelo de princípios e valores para às equipes de desenvolvimento produzir rapidamente e responder às mudanças, eles chamaram a si mesmos de Aliança Ágil. Embora as metodologias que compõem o movimento ágil já existissem o Manifesto Ágil é considerado oficialmente como o início do movimento ágil (Tanaka e Banki, 2008).

Segundo Paula (2008) os modelos tradicionais ocultavam uma série de problemas sérios, como:

- Problema para entrega dentro do prazo considerando o escopo de todas as funcionalidades solicitadas;

- Grande número de erros dado que a metodologia tradicional dificulta a execução de alterações durante o processo de desenvolvimento;

As Metodologias Ágeis apresentam novas práticas e mudanças na cultura organizacional da equipe, que pode impactar como um ganho para todos que compõem a equipe de desenvolvimento. A real intenção é fazer com que a equipe seja unida e se sinta motivada para aplicar as ideias da metodologia.

Dentre as Metodologias Ágeis mais conhecidas, destacam-se:

- Crystal - Crystal é o nome de uma família de métodos que devem ser ajustados para melhor se adaptarem a uma determinada equipe e projeto (Siqueira, 2003). Os métodos são moldados um a um para ter quantidade exata de processo para atender os projetos. Os membros são identificados por cores que indicam a intensidade do método. Neste caso, quanto mais escura a cor, maior é a complexidade do projeto (Santos et al., 2011)
- Feature Driven Development (FDD) - Desenvolvimento Guiado por Funcionalidade é uma Metodologia Ágil para gerenciamento e desenvolvimento de software, que visa manter metodologias tradicionais, equilibrando práticas atuais e as tradicionais. Cada membro da equipe pode assumir mais de um papel simultaneamente e um papel pode ser assumido por mais de um membro da equipe (Cunha; Filipakis, 2012).
- *Kanban* – O *Kanban* é uma abordagem que transforma o gerenciamento do projeto com alterações no ciclo de desenvolvimento de software. O *Kanban* tem como principal objetivo transformar o trabalho em andamento visível para toda equipe, criando um sinal visual que indica que o novo trabalho pode ou não ser iniciado e se o limite acordado para cada fase está sendo respeitado (Mariotti, 2008).
- Lean Software Development (LSD) – A LSD é uma Estratégia que busca satisfação dos clientes. Segundo o *Lean Institute* Brasil (2010), atende as necessidades do cliente de maneira mais simples possível, com um menor valor, aproveitando ao máximo todos os recursos disponíveis para a produção e tendo como consequência um melhor custo benefício para o cliente. Conforme Fadel e Silveira (2010), o desenvolvimento de software

Lean é a aplicação dos conceitos do sistema de produção da Toyota para o desenvolvimento de software.

- *SCRUM* - O *SCRUM* visa fornecer software aos clientes de forma rápida e com qualidade. O *SCRUM* é um framework de inspeção e adaptação, dentro do qual é possível aplicar vários processos e técnicas visando o desenvolvimento de produtos complexos, de forma a aumentar a qualidade e previsibilidade dentro dos projetos (SCHWABER; SUTHERLAND, 2009).

- Extreme Programming (XP) - A Metodologia Ágil XP foi desenvolvida para equipes desenvolvimento pequenas ou médias. A base da metodologia XP é formada por quatro valores: comunicação, *feedback*, simplicidade e coragem (Fowler, 2005).

Cada metodologia tem suas características e particularidades, mas todas tem o mesmo propósito é buscar estratégias para aumentar a produtividade e qualidade do produto desenvolvido.

### **2.2.1 Scrum**

O *Scrum* surgiu em 1996 por Ken Schwaber e Jeff Sutherland e destaca-se dos demais métodos, por causa maior ênfase dada ao gerenciamento do projeto. O *Scrum* pode ser definido segundo Schwaber e Sutherland (2011) como *framework* dentro do qual pessoas podem tratar e resolver problemas complexos e adaptativos, enquanto produtiva e criativamente entregam produtos com o mais alto valor possível. O termo *Scrum* surgiu por causa de um jogo chamado de Rugby, pois se organizam em círculos para planejar uma jogada.

O *Scrum* é leve, simples de entender e extremamente difícil de dominar. O *Scrum* possui uma abordagem iterativa e incremental para aperfeiçoar a previsibilidade e o controle de riscos. Possui três pilares que dão suporte a implementação de controle de processo empírico: transparência, inspeção e adaptação.

#### **2.2.1.1 Equipe Scrum**

A Equipe do *Scrum* está associada a seus papéis, eventos, artefatos e regras. Cada componente dentro do *framework* serve a um propósito específico e é

essencial para o uso e o sucesso do *Scrum*. O modelo de equipe no *Scrum* é projetado para aperfeiçoar a flexibilidade, criatividade e produtividade (Schwaber; Sutherland, 2011)

O Time *Scrum* é composto pelo *Product Owner* (PO), a Equipe de Desenvolvimento e o *Scrum Master* e podem ser auto-organizáveis e multifuncionais.

As Equipes auto-organizáveis escolhem qual a melhor maneira para completarem seu trabalho, em vez de serem dirigidos por outros de fora da equipe. Já as Equipes multifuncionais possuem todas as competências necessárias para completar o trabalho sem depender de outros que não fazem parte da equipe.

O *Scrum* tem como premissa trabalhar com equipes pequenas, no máximo sete pessoas, requisitos que são pouco estáveis ou desconhecidos; e iterações curtas (Soares et al).

### ***Product Owner***

O *Product Owner* (PO) ou Dono do Produto é o responsável pelo Product Backlog. Ele que define e prioriza as funcionalidades desejadas para o produto, ou as atividades necessárias ao projeto, descrevendo-as em forma de Histórias no Product Backlog. O PO é o representante do cliente na equipe (Aguanno, 2005), citado por Rose e Mello (2010).

Para Schwaber e Sutherland, 2011o gerenciamento do Backlog do Produto inclui:

- Expressar claramente os itens do Backlog do Produto;
- Ordenar os itens do Backlog do Produto para alcançar melhor as metas e missões;
- Garantir o valor do trabalho realizado pelo Time de Desenvolvimento;
- Garantir que o Backlog do Produto seja visível, transparente, claro para todos, e mostrar o que o Time *Scrum* vai trabalhar a seguir; e,
- Garantir que a Equipe de Desenvolvimento entenda os itens do Backlog do Produto no nível necessário.

## **Scrum Master**

O *Scrum Master* atua como responsável por garantir que o *Scrum* seja entendido e aplicado. O *Scrum Master* faz isso para garantir que o Time *Scrum* adere à teoria, práticas, regras, para que gerenciamento tenha sucesso é considerado ainda um servo-líder para o Time *Scrum*.

De acordo com Schwaber e Sutherland (2011) o *Scrum Master* ajuda aqueles que estão fora do Time *Scrum* a entender quais as suas interações com o Time *Scrum* são úteis e quais não são. O *Scrum Master* ajuda todos a mudarem estas interações para maximizar o valor criado pelo Time *Scrum*.

Schwaber e Sutherland(2011) relata que o *Scrum Master* ajuda aqueles que estão fora do Time *Scrum* a entender quais as suas interações com o Time *Scrum* são úteis e quais não são. O *Scrum Master* ajuda todos a mudarem estas interações para maximizar o valor criado pelo Time *Scrum*. Abaixo é descrito como é trabalho para o *Product Owner*, para a Equipe de Desenvolvimento e para a Organização, abaixo segue a explicação.

### **O *Scrum Master* trabalhando para o *Product Owner***

O *Scrum Master* serve o *Product Owner* de várias maneiras, incluindo:

- Encontrando técnicas para o gerenciamento efetivo do Backlog do Produto;
- Claramente comunicar a visão, objetivo e itens do Backlog do Produto para a Equipe de Desenvolvimento;
- Ensinar a Time *Scrum* a criar itens de Backlog do Produto de forma clara e concisa;
- Compreender a longo-prazo o planejamento do Produto no ambiente empírico;
- Compreender e praticar a agilidade; e,
- Facilitar os eventos *Scrum* conforme exigidos ou necessários.

### **O *Scrum Master* trabalhando para a Equipe de Desenvolvimento**

O *Scrum Master* serve a Equipe de Desenvolvimento de várias maneiras, incluindo:

- Treinar a Equipe de Desenvolvimento em autogerenciamento e interdisciplinaridade;
- Ensinar e liderar a Equipe de Desenvolvimento na criação de produtos de alto valor;
- Remover impedimentos para o progresso da Equipe de Desenvolvimento;
- Facilitar os eventos *Scrum* conforme exigidos ou necessários; e,
- Treinar a Equipe de Desenvolvimento em ambientes organizacionais nos quais o *Scrum* não é totalmente adotado e compreendido.

### **O *Scrum Master* trabalhando para a Organização**

O *Scrum Master* serve a Organização de várias maneiras, incluindo:

- Liderando e treinando a organização na adoção do *Scrum*;
- Planejando implementações *Scrum* dentro da organização;
- Ajudando funcionários e partes interessadas a compreender e tornar aplicável o *Scrum* e o desenvolvimento de produto empírico;
- Causando mudanças que aumentam a produtividade do Time *Scrum*; e,
- Trabalhando com outro *Scrum Master* para aumentar a eficácia da aplicação do *Scrum* nas organizações.

### **Equipe de Desenvolvimento**

A equipe de desenvolvimento é a responsável pelo trabalho de entregar uma versão pronta a cada *Sprint*, somente integrantes da Equipe de Desenvolvimento criam incrementos.

As Equipes de Desenvolvimento são estruturadas e autorizadas pela empresa para organizar e gerenciar seu próprio trabalho. Segundo Schwaber e Sutherland, 2011 as Equipes de Desenvolvimento possuem as seguintes características:

- Eles são auto-organizadas. Ninguém (nem mesmo o *Scrum Master*) diz a Equipe de Desenvolvimento como transformar o Backlog do Produto em incrementos de funcionalidades potencialmente utilizáveis;
- Equipes de Desenvolvimento são multifuncionais, possuindo todas as habilidades necessárias, enquanto equipe, para criar o incremento do Produto.

- O *Scrum* não reconhece títulos para os integrantes da Equipe de Desenvolvimento que não seja o Desenvolvedor, independentemente do trabalho que está sendo realizado pela pessoa; Não há exceções para esta regra.
- Individualmente os integrantes da Equipe de Desenvolvimento podem ter habilidades especializadas e área de especialização, mas a responsabilidade pertence à Equipe de Desenvolvimento como um todo; e,
- Equipes de Desenvolvimento não contém subequipes dedicadas a domínios específicos de conhecimento, tais como teste ou análise de negócios.

O tamanho ideal da Equipe de Desenvolvimento é pequeno o suficiente para se manter ágil e grande o suficiente para completar uma parcela significativa do trabalho.

#### **2.2.1.2 Sprint**

Por ser uma Metodologia Ágil, o *Scrum* é dividido em iterações, cada uma delas é chamada de *Sprint*. O *Sprint* representa um ciclo, ele pode durar de duas a quatro semanas, mas é tipicamente mensal. (Paula, 2008)

No final de cada ciclo é importante ter uma parte do produto concluída para que seja apresentada ao cliente, realizando entregas das partes aos poucos até conclusão do projeto, garante ao cliente que o produto em andamento está sendo desenvolvido conforme foi pedido.

De acordo com Paula (2008) toda *Sprint* tem definido o que será construído, um plano projetado e flexível que irá guiar a construção, o trabalho e o resultado do produto.

A composição das *Sprints* são reuniões de planejamento da *Sprint*, reuniões diárias, trabalho de desenvolvimento, revisão da *Sprint* e a retrospectivas da *Sprint*.

Durante a *Sprint* (Schwaber; Sutherland, 2011):

- Não são feitas mudanças que podem afetar o objetivo da *Sprint*;
- A composição da Equipe de Desenvolvimento permanecem constantes;
- As metas de qualidade não diminuem; e,

- O escopo pode ser clarificado e renegociado entre o *Product Owner* e a Equipe de Desenvolvimento quanto mais for aprendido.

As *Sprints* são limitadas à um mês corrido e quando o ciclo da *Sprint* é muito longo, a definição do que será construído pode alterar a complexidade pode aumentar e o risco pode crescer. *Sprints* permitem previsibilidade que garante a inspeção e adaptação do progresso em direção a meta pelo menos a cada mês corrido. *Sprints* também limitam o risco ao custo de um mês corrido. (Schwaber; Sutherland, 2011)

Uma *Sprint* pode ser cancelada antes do prazo da *Sprint* terminar, porem somente o PO tem a autoridade para cancelar a *Sprint*, uma *Sprint* também pode ser cancelada caso o *Sprint* se torne obsoleto. (Schwaber; Sutherland, 2011)

Geralmente o cancelamento de *Sprints* tende a atrapalhar o desenvolvimento equipe, consome recursos, perdendo tempo em reagrupar em outra reunião de planejamento da *Sprint* para iniciar outra *Sprint*.

### 2.2.1.3 Reunião

No *Scrum* as iterações são chamadas de *Sprints*, ao inicio de cada *Sprint* é necessário uma reunião de planejamento (*Sprint Planning Meeting*), onde o PO e equipe de desenvolvimento decidem o que será feito (*Selected Product Backlog*). (Soares et al., 2007)

A reunião é dividida em duas partes (Soares et al. 2007):

- *Sprint Planning 1* – Na primeira parte da reunião PO apresenta os requisitos mais importantes e verifica os que têm maior prioridade para serem implementados. A equipe de desenvolvimento define o poderá ser implementado no desenvolvimento da próxima *Sprint*.
- *Sprint Planning 2* – Já na segunda, a equipe planeja como será seu trabalho definindo o *Sprint Backlog*, que são tarefas necessárias para implementar as funcionalidades no *Product Backlog*.

Nos encontros diários (*Daily Scrum Meeting*) os membros da equipe de desenvolvimento e *Scrum Master*, discutem o que será trabalhado e possíveis

impedimentos que possam no futuro atrapalhar progresso do trabalho. (Soares et al. 2007)

De acordo Schwaber e Sutherland (2011) a *Daily Scrum Meeting* são objetivas e rápidas, duram em cerca de 15 minutos e é mantida no mesmo horário e local todo dia para reduzir a complexidade. Estas reuniões representam uma maneira de manter as equipes sincronizadas para o andamento do *Sprint*. (ROSE; MELLO, 2010)

Durante a reunião cada integrante da Equipe de Desenvolvimento esclarece (Schwaber; Sutherland, 2011):

- O que foi completado desde a última reunião?
- O que será feito até a próxima reunião?
- Quais os obstáculos que estão no caminho?

Segundo Schwaber e Sutherland (2011), reuniões diárias melhoram as comunicações, eliminam outras reuniões, identificam e removem impedimentos para o desenvolvimento, destacam e promovem rápidas tomadas de decisão, e melhoram o nível de conhecimento da Equipe de Desenvolvimento.

No final de cada *Sprint* é realizada uma revisão para inspecionar o Backlog do Produto, para averiguar e se precisam mudanças nas próximas *Sprints*. A reunião de revisão da *Sprint* é informal, destina-se promover apresentação do incremento. (Schwaber; Sutherland, 2011)

Para Schwaber e Sutherland (2011) “o resultado da Reunião de Revisão da *Sprint* é um Backlog do Produto revisado que define o provável Backlog do Produto para a próxima *Sprint*”.

A retrospectiva da *Sprint* tem como propósito ajudar a equipe desenvolvimento, verificar como esta o andamento e criar melhorias a serem aplicadas na próxima *Sprint*. A retrospectiva ocorre depois da Revisão da *Sprint* e da reunião de planejamento (Schwaber; Sutherland, 2011)

A Retrospectiva da *Sprint* destina-se a (Schwaber; Sutherland(2011):

- Inspecionar como a última *Sprint* foi em relação as pessoas, relações, processos e ferramentas;
- Identificar e ordenar os principais itens que foram bem e as potenciais melhorias; e,

- Criar um plano para implementar melhorias no modo que a equipe de desenvolvimento faz seu trabalho;

#### **2.2.1.4 Artefatos**

Os artefatos em Metodologias Ágeis podem ser denominados como documentação ou gráficos, que auxiliam no desenvolvimento do projeto.

Os principais artefatos do *Scrum* são Backlog do Produto, *Sprint Backlog*, *Burndown Chart* e o *TaskBoard*.

#### **Backlog do Produto**

O Backlog Produto é uma lista elaborada com os principais itens do que deve conter o software, suas funcionalidades e requisitos. Nesta lista cada item tem uma prioridade de entrega que indica o quanto de valor ele gera para o negócio itens no decorrer do projeto. (Paula, 2008)

De acordo com Schwaber e Sutherland (2011), Backlog do Produto é dinâmico, muda constantemente para identificar o que o produto necessita para ser mais apropriado, competitivo, útil só e existirá enquanto o produto também existir.

Os itens do Backlog do Produto apresenta os atributos da descrição, ordem e estimativa. Além de listar todas as características, funções e melhorias e correções o Backlog do Produto lista por ordem de valor, risco, prioridade e necessidade. Os itens no topo da lista ordenada do Backlog do Produto determinam as atividades de desenvolvimento mais imediatas.(Schwaber e Sutherland (2011)

#### ***Sprint Backlog***

No início de cada *Sprint* a equipe seleciona os itens do *Backlog* do Produto, de acordo com as suas prioridades e o tempo que será gasto para completar as suas funcionalidades, essa nova lista é conhecida como *Sprint Backlog* (Paula, 2008)

A lista de *Sprint Backlog* define qual trabalho a equipe de desenvolvimento concretizara para converter os itens de produto em um incremento finalizado. (Schwaber; Sutherland, 2011).

É importante que a equipe identifique os itens e tamanho do *Sprint Backlog*, os seus membros são quem deverão escolher com o que irão se comprometer (Paula, 2008), depois as tarefas maiores serão subdivididas em partes menores.

### **Burndown Chart**

O *Burndown Chart* ou gráfico de *Burndown* é um gráfico onde demonstra como está o andamento do *Sprint*, através dele podemos ver a direção do projeto ao longo do seu ciclo de desenvolvimento. (Paula, 2008).

Para Kniberg e Skarin(2009) no gráfico de *Burndown* demonstra, em uma base diária, o quanto de trabalho resta na iteração atual. Normalmente horas ou dias ou pontos de história. Observe a Figura 2.

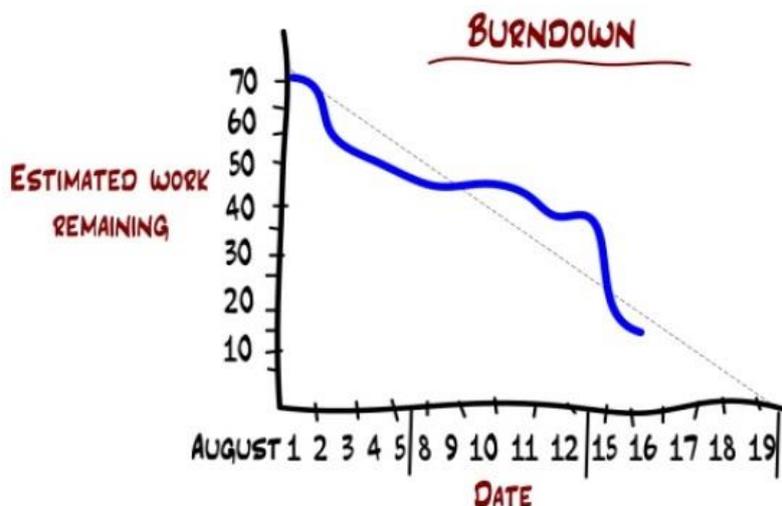


Figura 2 - Gráfico de Burndown (fonte: KNIBERG e SKARIN, 2009).

### **TaskBoard**

É um grande painel onde colocadas várias informações importantes para o acompanhamento do *Sprint*. O *Sprint Backlog*, ou seja, as atividades não iniciadas, as que estão em andamento e as concluídas ficam sempre visíveis e disponíveis para todos os interessados no projeto. (Paula, 2008). Na figura 3 um exemplo de Taskbord.

O *Taskboard* apresenta algumas características como (Paula, 2008):

- Normalmente é desenhado em uma parede e as atividades são descritas em *post-its*;

- Apresenta uma visão geral do *Sprint*;
- Fica acessível a todos os interessados no projeto



Figura 3 - TaskBoard (fonte: KNIBERG e SKARIN, 2009).

### 2.2.2 Kanban

O *Kanban* significa cartão em japonês. O nome surgiu dos sistemas de cartão usados em indústria de produção, que tinha com finalidade gerenciar o fluxo do de trabalho. Baseia na ideia de que as atividades em andamentos devem ser limitadas, um novo item só pode ser iniciado depois que o item em andamento seja finalizado ou quando uma função automática inicia no mesmo instante. (Mariotti, 2008)

O *Kanban* é uma estratégia gerencial de mudança do projeto, seu principal objetivo é deixar o trabalho em andamento sempre visível para equipe toda. O *Kanban* é uma ferramenta do processo, possui um mecanismo de sinalização, que apresenta processo de uma atividade.

Uma das características importantes é o conceito de “puxar tarefa” quando há capacidade de processá-la [...] os requisitos são adicionados a lista de backlog e “puxados” pelos membros que liberam suas atividades correntes e se tornam disponíveis para iniciar uma nova tarefa. (Mariotti,2008)

A implementação do modelo *Kanban* se resume em três etapas que são:

- Visualizar os processos;
- Limitar o trabalho em processo do inglês WIP (*work in progress*);
- Gerenciamento do lead-time, ou seja, tempo que a atividade leva para passar por todas as fases até a sua entrega.

### **2.2.2.1 Sistema *Kanban***

As tarefas são chamadas de cartões, o número de cartões representa a capacidade limite de cada fase de um sistema que está sendo desenvolvido (Mariotti, 2008)

Segundo Mariotti (2008) cada cartão funciona como um mecanismo de sinalização e o sistema só permite iniciar uma nova tarefa quando um cartão está disponível, é de extrema importância respeitar esta regra.

O sistema *Kanban* é um método simples, barato e fácil de implementar, apresentar resultados rapidamente e permite o gerenciamento das atividades em andamento e garantindo o bom desempenho da equipe. (Kniberg ;Skarin, 2009).

O autor Mariotti (2008) cita “O *Kanban* buscar atingir um ritmo sustentável de desenvolvimento para que todos os indivíduos possam alcançar esse objetivo entre vida pessoal e profissional”.

O *Kanban* agrega muito em uma empresa, contribui e altera a cultura organizacional de modo positivo, sistema tende a contribuir para a maturidade da equipe (Mariotti, 2008).

Para aplicação da metodologia é necessário três etapas segundo Mariotti (2008) que são: criar o painel de visualização, limitar os processos WIP e gerenciar o lead-time, aplicando o conceito de puxar uma nova tarefa quando um cartão está disponível.

No *Kanban*, o ideal é focar nos itens de desenvolvimento, evitando priorização de exceções imprevistas. Para que o projeto alcance maturidade, a qualidade deve estar sempre presente para que o desempenho da equipe de desenvolvimento aumente.

### **2.2.2.2 Sinalizador visual *Kanban***

O sinalizador visual segundo Mariotti (2008) funciona como uma ferramenta de sinalização de processo, onde deixa explícito o andamento dos processos.

As colunas representam os diferentes estados de uma tarefa e seu processo de desenvolvimento, o cartão é movido de uma fase para um estado até que esteja finalizado. O quadro que representa o *Kanban* possui: análise, desenvolvimento, aceitação e implantação.

Mariotti (2008) ressalta “o *Kanban*, à primeira vista, pode lembrar o conceito da engenharia de cascata, porém na prática, o *Kanban* não atua como o cascata e evita os problemas decorrente do conceito”. O *Kanban* segue como linha de produção a regra de limitar o processo em andamento, o *WIP*, essa regra evita as falhas apresentadas pela engenharia cascata.

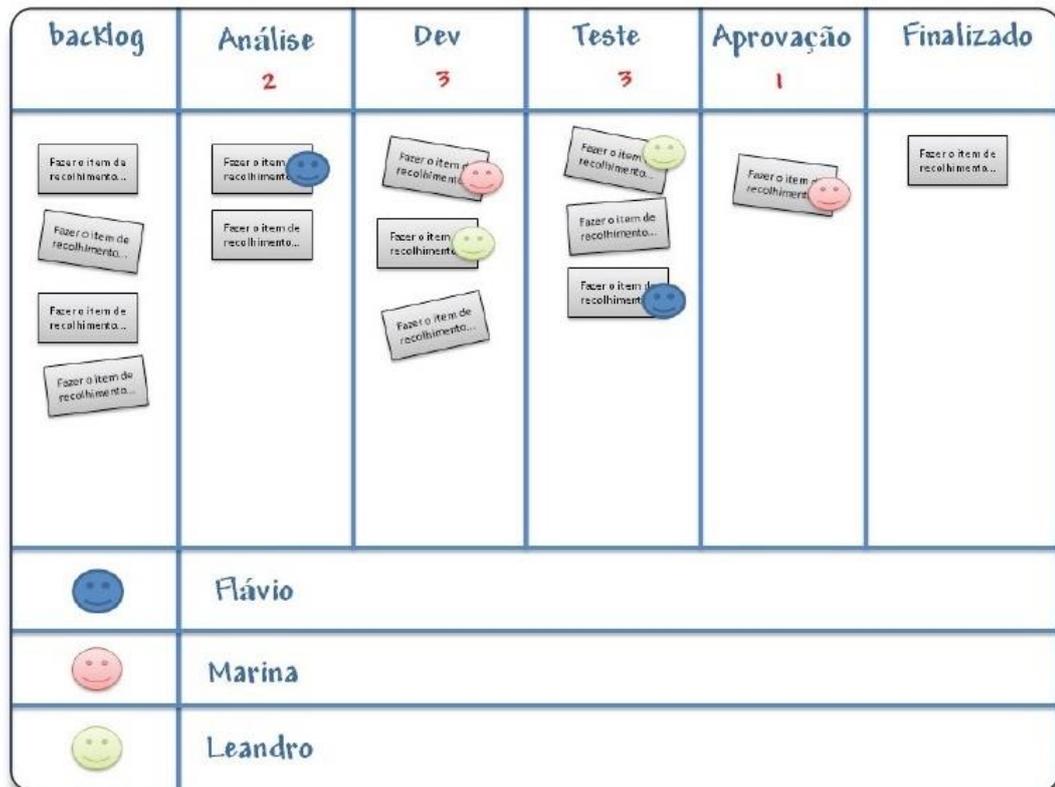


Figura 4 - Modelo simples de sinalizador visual *Kanban*. (fonte: MARIOTTI, 2008)

No quadro visual do sistemas *Kanban* é aplicada a regra em que cada coluna terá um *WIP* estabelecido e que são representados pelo numero máximo de cartões em cada fase (Mariotti, 2008).

O cartão é composto por uma breve descrição das historias que descreve seus requisitos, todo cartão entra na fila de *Backlog* e aguarda sua chamada ate que os próximos cartões sejam finalizados, isso se dá ao nome de “puxar cartões”. Na figura F4, ilustra um modelo simples de sinalizador visual *Kanban*.

Mariotti(2008) diz “no *Kanban*, o conceito de limitar o que deve ser feito é aplicado em todas as colunas do quadro. Essa é uma maneira rápida de reduzir o lead-time.”

### 2.2.2.3 Vantagens da utilização do *Kanban*

O *Kanban* possuem algumas vantagens comparada as outras metodologias: as falhas tornam-se claramente visíveis em tempo real; o benefício de encontrar os “gargalos” faz com que as pessoas passem a colaborar ainda mais para a cadeia de valor em vez de apenas fazerem a sua parte (Mariotti, 2008).

Mariotti ainda cita “O *Kanban* também pode ser aplicado em outros ambientes organizacionais, não só a equipe de desenvolvimento. O intuito dele é aumentar a visibilidade do que esta acontecendo.”

## 2.3 *Scrum* e *Kanban* na comunicação

A Metodologia Ágil citada na pesquisa inova, pois consigo traz eficiência da equipe e do gerenciamento do projeto, mantendo a organização e diminuindo o retrabalho. A intenção deste trabalho é utilizar Metodologia Ágil como gerenciamento da comunicação entre a equipe os motivando no decorrer do projeto.

O *Scrum* utilizam reuniões frequentes para garantir não haja ruído de comunicação no desenvolvimento do projeto. Segundo Paula (2008) um dos principais benefícios obtidos com a prática do *Scrum* é a melhoria comunicação entre os membros da equipe, pois as chances de sucesso em projeto são muito maiores em times onde há uma boa comunicação, onde há entendimento e cooperação mútua.

O *Kanban* por ser uma ferramenta de processo auxilia na estratégia gerencial de mudança do projeto alterando a cultura organizacional. O *Scrum* e *Kanban* utilizam painel ou sinalizador visual que deixa o trabalho em andamento sempre visível para equipe toda. Através de reuniões diárias, sinalizador visual e o gráfico de *burndown* é possível manter uma relação boa entre os participantes, tendo ainda troca de conhecimento e experiências. O trabalho visa alterar a cultura organizacional da empresa, gerenciando a comunicação entre a equipe.

Na Figura 5 pode se notar que ambos quadros são iguais, mas o quadro do *Kanban* tem um item a mais no andamento que é número, o pequeno dois(2) na coluna do meio no quadro *Kanban*. E isto é tudo. Este 2 significa “não pode haver mais de 2 itens nesta coluna ao mesmo tempo.



Figura 5 - Comparação entre os quadros (fonte: KNIBERG e SKARIN, 2009).

Segundo Kniberg e Skarin(2009) o *Scrum* não há nenhuma regra advertindo a equipe para não colocar todas as atividades na coluna “Em execução” ao mesmo tempo.

Abaixo na Tabela 1 descreve as diferenças entre os métodos.

SCRUM	KANBAN
Iterações Timeboxed prescritas.	Iterações Timeboxed opcionais. Pode ter cadência separada para o planejamento, release e melhoria de processos. Pode ser orientada para eventos, em vez de timeboxed.
Equipe compromete-se a uma quantidade específica de trabalho para esta iteração.	Compromisso opcional.
Usa a velocidade como padrão métrico para o planejamento e melhoria de processos.	Usa o <i>Lead time</i> como padrão métrico para o planejamento e melhoria de processos.

<i>SCRUM</i>	<i>KANBAN</i>
Equipes multifuncionais prescritas.	Equipes multifuncionais opcionais. Equipes de Especialistas autorizada.
Os itens devem ser divididos para que possam ser concluídos dentro de 1 <i>sprint</i> .	Nenhum tamanho especial de item é prescrito.
Gráfico Burndown.	Nenhum tipo específico de diagrama é prescrito.
WIP limitado indiretamente (por <i>sprint</i> ).	WIP limitado diretamente (por situação do fluxo de trabalho).
Estimativa prescrita.	Estimativa opcional.
Não poderá adicionar itens à iteração em uso.	Pode adicionar novos itens sempre que houver capacidade disponível.
O <i>sprint backlog</i> é de uma equipe específica	Quadros kanban podem ser compartilhados por várias equipes ou individualmente.
Possui 3 papéis ( <i>Product Owner/ScrumMaster/Team</i> )	Não discrimina nenhum papel.
Um quadro Scrum é reiniciado entre cada <i>sprint</i>	Um quadro kanban continua.
Prescreve um <i>product backlog</i> priorizado	Priorização é opcional.

**Tabela 2 - Comparação entre os Métodos (fonte: KNIBERG e SKARIN, 2009).**

## 2.4 Gestão de Processo BPM

Business Process Management(BPM) é uma metodologias que tem por finalidade melhorar a performance no desempenho no processo de negócio. Para compreensão do BPM, é necessário compreender o significado de processo de negócio. Nesse contexto, segundo ABPMP(2009) um “processo” é um conjunto definido de atividades ou comportamentos executados por humanos ou máquinas para alcançar uma ou mais metas.

Gerenciamento de Processos de Negócio (BPM) é uma abordagem disciplinada para identificar, desenhar, executar, documentar, medir, monitorar, controlar e melhorar processos de negócio. (ABPMP, 2009)

Conforme a ABPMP(2009) existem vários conceitos fundamentais, básicos, que definem BPM, incluindo noções como:

- BPM é uma disciplina de gerenciamento e um conjunto de tecnologias habilitadoras.
- BPM aborda um trabalho ponta-a-ponta e diferencia entre conjuntos de subprocessos, tarefas, atividades e funções.
- BPM é um conjunto contínuo, em curso, de processos com o foco no gerenciamento de processos de negócio ponta-a-ponta nas organizações.
- BPM inclui modelagem, análise, desenho e medição de processos de negócio de uma organização.
- BPM requer um compromisso significativo da organização que frequentemente introduz novos papéis, responsabilidades e estruturas às organizações tradicionais orientadas a funções.
- BPM é habilitada por tecnologia através de ferramentas para modelagem, simulação, automação, integração, controle e monitoramento de processos de negócio e de sistemas de informação que suportam esses processos.

O ciclo de vida do BPM, pode ser sumarizada por um conjunto gradual e interativo de atividades que incluem (ABPMP, 2009):

•Planejamento - ciclo de vida BPM começa com o desenvolvimento de um plano e uma estratégia dirigida a processos para a organização. O plano inicia por um entendimento das estratégias e metas da organização desenhadas para assegurar uma proposição de valor atrativa para clientes. Essa fase estabelece a

estratégia e o direcionamento do processo BPM. Também identifica papéis e responsabilidades organizacionais apropriadas de BPM, patrocínio executivo, metas, expectativas de medições de desempenho e metodologias.

- Análise - A análise de processos de negócio incorpora várias metodologias com a finalidade de entender os atuais processos organizacionais no contexto das metas e objetivos desejados. A análise assimila informações oriundas de planos estratégicos, modelos de processo, medições de desempenho, mudanças no ambiente externo e outros fatores, a fim de entender completamente os processos de negócio no escopo da organização como um todo.

- Desenho e Modelagem - As atividades de desenho de processos focam no desenho intencional e cuidadoso de como o trabalho ponta-a-ponta ocorre de modo a entregar valor aos clientes. A sequência de atividades é documentada, incluindo o desenho do trabalho realizado, em que tempo, em qual local, por quais atores de processo e utilizando qual metodologia. O desenho define o que a organização quer que o processo seja e responde questões como: o quê, quando, onde, quem e como o trabalho ponta-a-ponta é realizado.

- Implementação de processos - Implementação de processos de negócio é a realização do desenho aprovado de processo de negócio em procedimentos e fluxo de trabalho documentados, testados e operacionais. Também inclui a implementação de políticas e procedimentos novos ou revisados. Durante atividades de implementação assume-se que as fases de análise, modelagem e desenho criaram e aprovaram um conjunto completo de especificações, então, somente pequenos ajustes devem ocorrer durante a implementação.

- Monitoramento e Controle dos processos - No contexto do ciclo BPM, medição e monitoramento provêm informações chave de desempenho de processos através de métricas relacionadas às metas e ao valor para a organização. A análise de informações de desempenho de processos pode resultar em atividades de melhoria, redesenho ou reengenharia.

- Refinamento – A contínua medição e monitoramento de processos de negócio fornecem a informação necessária para que gestores de processo ajustem recursos a fim de atingir objetivos dos processos. O refinamento trata aspectos de ajustes e melhorias pós-implementação de processos com base nos indicadores e informações-chave de desempenho.

Conforme mostra a Figura 6 abaixo, os processos de negócio se movem através do ciclo de vida citado acima e são habilitados ou restringidos por uma variedade de fatores incluindo os quatro fatores primários de valores, crenças, liderança e cultura conforme a Figura 6.

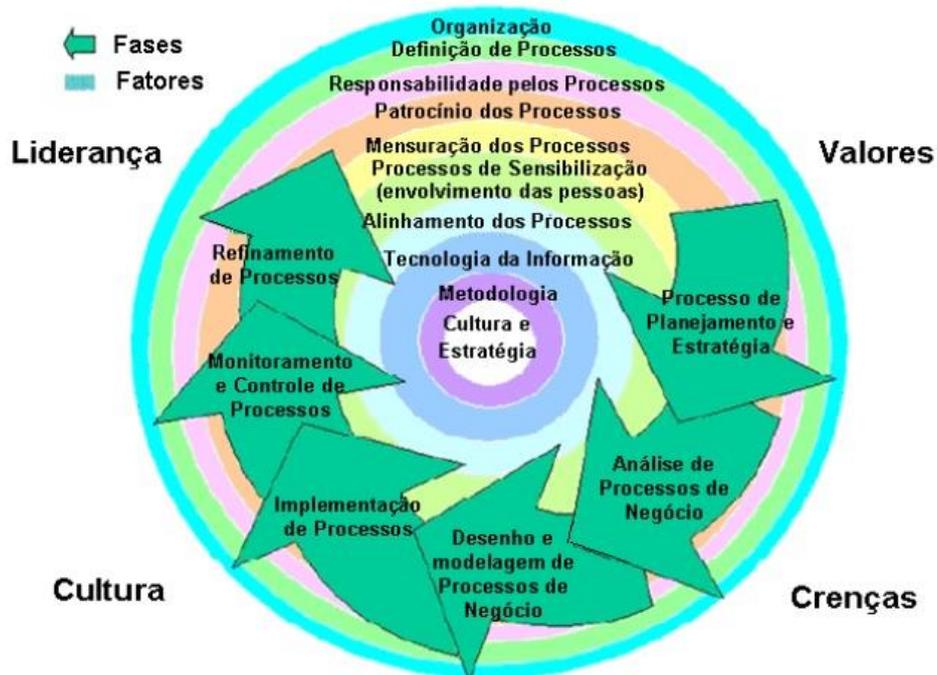


Figura 6 - Ciclo de vida do BPM (APBMP, 2009).

Há três tipos de processos de negócio: primários, de suporte e de gerenciamento (ABPMP, 2009):

- Processos primários são de natureza interfuncional e compõem a cadeia de valor.
- Processos de suporte, tais como recursos humanos e TI, habilitam outros processos.
- Processos de gerenciamento são utilizados para medir, monitorar e controlar atividades de negócio. Garantem que processos primários e de suporte atinjam metas operacionais, financeiras, regulatórias e legais.

### 2.4.1 BPMN

De acordo com ABPMP(2009) a Modelagem de Processos de Negócio é um conjunto de atividades envolvidas na criação de representações de um processo de negócio existente ou proposto. A seguir, alguns dos benefícios da modelagem:

- Modelos são relativamente rápidos, fáceis e baratos de completar.
- Modelos são fáceis de entender quando comparados a outras formas de documentação.
- Modelos fornecem uma linha-base para a medição.
- Modelos facilitam o processo de simulação e análise do impacto.
- Modelos tiram vantagem de vários padrões e um conjunto comum de técnicas.

*Business Process Modeling Notation* (BPMN) é um notação gráfica criada pelo *Business Process Management Initiative* e tem como objetivo ajudar na modelagem e padronização dos processos de negócio. A BPMN fornece uma simbologia simples, mas robusta, para modelar todos os aspectos de processos de negócio. (ABPMP, 2009)

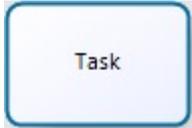
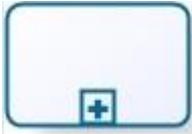
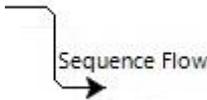
### 2.4.2 BizAgi

O BizAgi é um BPMS (Business Process Management System) uma ferramenta gratuita que permite modelagem de processos de negócio e documentação. O modelador permite que você visualize processos de negócios diagrama, modelo e documentos em padrão da indústria BPMN (Business Process model Notation), é um formato aceito mundialmente para modelagem de processos.

A modelagem de processos é um método para analisar, projetar e diagramar um fluxo de processos de negócios. Modelagem de um processo de forma iterativa, clara, transparente e simples irá permitir-lhe compreender, analisar e fazer uma mudança positiva para o processo de negócio.

Modelo de Negócio e Processos Notation, ou BPMN é uma notação gráfica criado para fornecer uma linguagem unificada de aceitação mundial. Desde que ingressou no grupo em 2007, BizAgi continua a ser um membro ativo do grupo

dentro da OMG que é responsável pela definição do padrão BPMN. BizAgi Process Modeler suporta a versão atual, BPMN 2.0.

ELEMENTOS	DESCRIÇÃO	NOTAÇÃO
Tarefa	Atividade atômica dentro de um fluxo de processo.	
Subprocesso	Identifica um ponto no processo em que um processo pré-definido é utilizado. A Subprocesso reutilizável é chamado de Atividade Chamada em BPMN. O elemento tem uma borda espessa.	
Evento Início	Indica onde um determinado processo é iniciado. Ele não tem qualquer comportamento particular.	
Evento Intermediário	Indica onde acontece alguma coisa em algum lugar entre o início e o fim de um processo. Vai afetar o fluxo do processo, mas não arranca ou (diretamente) encerrar o processo.	
Evento Final	Indica quando o processo termina.	
Gateway Convergência	Como: É usado para mesclar caminhos alternativos.	
Gateway Divergência	É usado para criar caminhos alternativos dentro do processo, mas somente um é escolhido.	
Fluxo de Sequência	Um fluxo de sequência é usado para mostrar o fim de que as atividades serão realizados no processo.	

**Tabela 2 - Descrição dos principais objetos de modelagem**

A ferramenta pode ser utilizada em modelagens de processos, permitindo exportação para arquivo do Word, PDF, XML, dentre outros formatos. Os processos podem ser facilmente importados e exportados a partir de Visio ou XML, e outras ferramentas.

Modelagem de um processo é uma tarefa iterativa e simples para representar as condições de negócios em um diagrama de fluxo, usando símbolos e formas. Para melhor entendimento na Tabela 2 é descrito os principais elementos da modelagem de processo no BizAgi.

BizAgi possibilita automatizar os processos de negócio de forma ágil e em um ambiente gráfico intuitivo, podem ser facilmente modificados, proporcionando para as organizações a agilidade necessária para atingir melhorar o desempenho dos processos de negócio nas organizações.

Foi desenvolvido para proporcionar resultados imediatos com as ferramentas para: diagramar processos em BPMN, definir regras de negócio, orquestrar outras aplicações, definir interface do usuário, otimização e balanceamento de carga de trabalho, portal web de trabalho, indicadores de desempenho de processos, monitor de atividades e muito mais. Na Figura 7 é ilustrado um exemplo de um processo de negócio na ferramenta BizAgi.

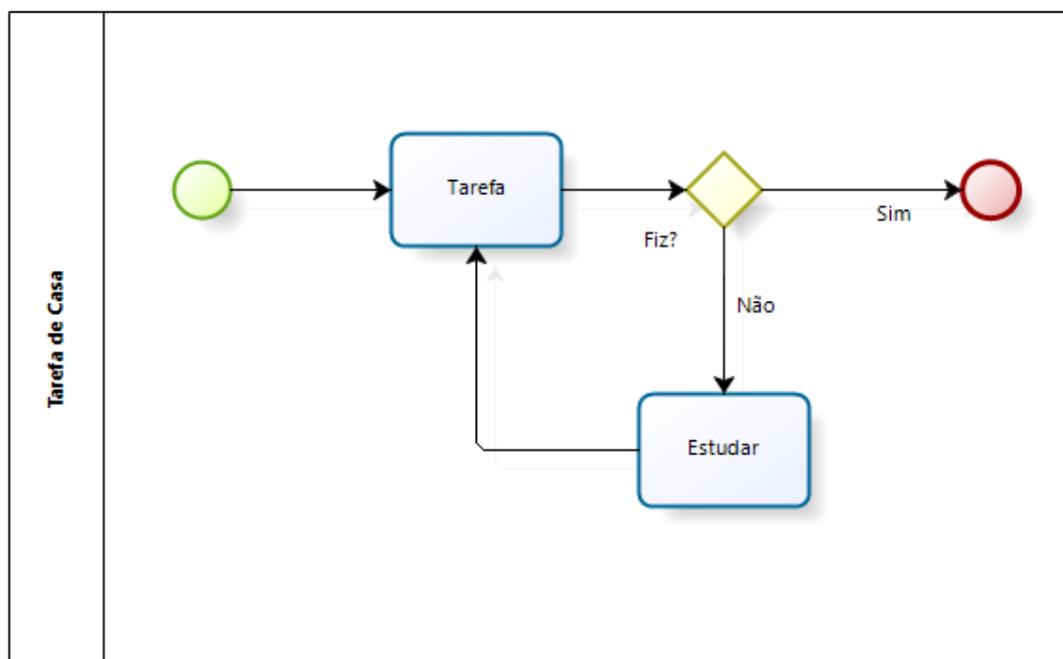


Figura 7 - Exemplo de processo criado no BizAgi

### 3 METODOLOGIA

A metodologia empregada neste trabalho foi estudo de caso. O estudo de caso foi realizado em uma empresa de desenvolvimento de software. Para embasamento do estudo foi realizado um levantamento bibliográfico, para fornecer subsídios e aprofundar o conhecimento sobre as Metodologias Ágeis. Caracteriza-se por um estudo mais aprofundado sobre os assuntos citados acima, permitindo um melhor conhecimento sobre a pesquisa e seus objetivos.

Os passos metodológicos para o desenvolvimento deste estudo serão:

- Estudar conceitos *Scrum* e *Kanban*;
- Analisar as metodologias *Scrum* e *Kanban*;
- Diagnosticar Empresa JoinSoft, para identificar os problemas ligados à comunicação;
- Estratégias para solucionar os problemas;
- Verificar os benefícios e evolução da equipe com a utilização da Metodologias Ágeis.

## 4 DESENVOLVIMENTO

Neste capítulo será apresentado e descrito o desenvolvimento do trabalho. Foi desenvolvido um estudo sobre os problemas de comunicação em equipes de desenvolvimento de software.

As práticas criadas neste trabalho foram aplicadas em um estudo de caso real na empresa JOINSOFT, uma empresa que trabalha com o desenvolvimento de softwares, situada em Ibaiti, Paraná. A empresa possui alguns problemas de comunicação entre a equipe.

Após o levantamento bibliográfico sobre o tema do trabalho, foi desenvolvido um questionário para colaborar com a identificação problemas ocorridos na empresa. A finalidade deste questionário é de identificar os problemas comunicação que ocorrem na empresa JOINSOFT. Por meio deste estudo foi possível identificar analisar e diagnosticar as falhas comunicação que vem atrapalhando a comunicação entre os membros.

A priori foi importante o levantamento das informações necessárias para dar prosseguimento ao estudo.

### 4.1 Elaboração do Questionário de Identificação de Problemas de Comunicação

O questionário de identificação de problemas de comunicação foi aplicado na empresa citada no trabalho. Este levantamento pode comprovar e verificar os problemas de comunicação citados. Segue abaixo o questionário.

1) Quantas pessoas compõem a equipe?

3

5

8

10

Outros: \_\_\_\_\_

2) Quais cargos são definidos na equipe de desenvolvimento?

Programador

Designer de Telas

- Tester
  - DBA
  - Suporte
  - Gerente de Projeto
  - Outros: \_\_\_\_\_
- 

3) Qual é a carga horária da empresa?

- 40 horas semanais
- 44 horas semanais

4) Fazem reuniões com frequência?

1 2 3 4 5

Nunca      Sempre

5) Como é a distribuição das tarefas?

---



---



---



---



---

6) Quais os problemas de comunicação são mais frequentes?

- Omissão de informação
  - Rejeição entre a equipe
  - Ruído nas informações
  - Metas e prazos inalcançáveis
  - Falta de padronização da linguagem
  - Desunião da equipe
  - Omissão do gerente ou da equipe
  - Falta de uma hierarquia na recepção das informações
  - Outros: \_\_\_\_\_
- 

7) Quais são as dificuldades encontradas em trabalhar em equipe?

- Omissão de informação
  - Falta de compromisso
  - Desunião da equipe
  - Duplicação de esforços
  - Outros: \_\_\_\_\_
-

8) Como fazem para resolver problemas de comunicação?

---



---



---



---



---

9) A empresa utiliza alguma ferramenta ou mecanismo de comunicação entre os participantes da equipe?

Reuniões

Chat

E- mail

Videoconferência

Outros: \_\_\_\_\_

---

10) A empresa trabalha com alguma Metodologia Ágil para gerenciamento do projeto?

*Extreme Programming (XP)*

*SCRUM*

*Lean Software Development (LSD)*

*Kanban*

Outros: \_\_\_\_\_

---

## 4.2 A JoinSoft

Foi realizada uma análise na empresa JoinSoft, para obter dados que pudessem ajudar na realização da melhoria. Por meio da aplicação do questionário, pode-se conhecer a rotina da empresa e seus principais problemas de comunicação na página 57 Apêndice A está a resposta do questionário de identificação.

A empresa relatou na reunião os problemas enfrentados e os que aconteciam com mais frequência:

- Ruído nas informações, metas e prazos inalcançáveis, falta de uma hierarquia na recepção das informações.
- As dificuldades encontradas em trabalhar em equipe estavam relacionadas ao não planejamento de atividades.

- A resolução dos problemas de comunicação eram feitas por meio de reuniões periódicas (embora poucas). Reunião de equipe, Documentação(escrever) das decisões tomadas.
- A empresa utiliza ferramenta Skype como um mecanismo de comunicação entre os participantes da equipe.
- Utilizavam o *Scrum*, mas não seguiam corretamente.

A rotina de trabalho da equipe é definida por uma carga horária de quarenta horas semanais e trabalha-se com o desenvolvimento de softwares.

A JoinSoft é uma empresa que trabalha com desenvolvimento de soluções de tecnologia de informação. A organização é composta por especialistas nas áreas de Engenharia de Software de Desenvolvimento de Sistemas. Sobre o aspecto de comunicação, os integrantes utilizam o Skype, software que permite a comunicação online através de voz e vídeo.

A empresa atualmente conta com três funcionários com cargos definidos como programador, gerente de projeto e analista de negócios. A distribuição de tarefas funciona da seguinte forma:

- Analista de Negócios: Comercial, contato direto com o cliente, levantamento de requisitos e necessidades, ajuste do conceito software e necessidade organizacional. Teste Funcional de recursos implementados para validar se está de acordo com a solicitação do cliente, Suporte ao Usuário.
- Engenheiro de Software: Análise, modelagem, teste e gestão do projeto de software.
- Programador: Desenvolvimento, Teste de Recursos Computacionais e Suporte ao usuário.

### **4.3 Aplicação do Questionário**

Logo após a reunião, foram analisadas as respostas dos questionários e foram encontradas os seguintes problemas:

- Problemas de solicitações feitas diretamente ao programador que não são repassadas para equipe:

- Clientes solicitam modificações diretamente ao programador que, em muitos casos, não transfere a informação para o restante da equipe para fins de análise e aplica alguma resolução incorreta. Em alguns casos, ao encontrar problemas, o cliente entra em contato com a gerência da empresa que desconhece a solicitação.
- Problemas de Conflito de decisões:
  - A empresa é administrada por 3 pessoas (2 analistas de sistemas e 1 administradora de empresas). A administradora trabalha na função de consultora para os clientes e como ponte para filtrar solicitações “exóticas”. Os 2 analistas são os responsáveis por autorizar a execução de determinada solicitação. Entretanto, os 2 analistas, em várias situações, não se encontram ao mesmo tempo na empresa e, em determinados, casos um analista solicita ao programador que execute a tarefa de uma maneira, e, após, o segundo analista (muitas vezes sem saber da ordem do primeiro) manda executar a tarefa de outra forma o que gera diversos entraves no processo e falhas no sistema e nas consultorias às empresas.
- Problemas de falta de conhecimento de solicitações
  - Semelhante ao apresentado no item 1, em muitos casos as solicitações não são repassadas às equipes de desenvolvimento, além do fato de existir dificuldades no acompanhamento e na medição dos trabalhos das solicitações conhecidas.
- “Telefone sem fio”
  - Recados, solicitações e situações diversas que não são repassadas as pessoas corretas ou são repassadas com ruídos.

## **4.4 Treinamento das Metodologias e Implantação dos Métodos Ágeis**

Com o auxílio dos dados do questionário, pode-se ter uma visão melhor do ambiente organizacional, auxiliando a implantação das metodologias e como eles

usaria para melhorar a comunicação na equipe. O treinamento foi realizado no mesmo dia, a fim de explicar o funcionamento dos métodos.

Em todo momento a empresa esteve disposta a seguir todas tarefas para que pudesse melhorar a comunicação entre eles. Antes de implantar as metodologias foi necessário refletir sobre problemas que existiam na empresa, treinar bem a equipe para que não existisse ruídos de informações.

Seguir corretamente Sprint do Scrum não se encaixava muito bem para realidade da empresa, pois estavam tendo que fazer novas priorizações diariamente, então o Kanban foi utilizado como ponto de partida. Isso significa que a equipe deve começar a dizer “não” a trabalhos que não tem como concluir, a atentar para a qualidade e remover todas as coisas não prioritárias do Backlog, através formulário descrito na Apêndice B e análise das prioridades.

Após a implantação os clientes da empresa, foram avisados a respeito das mudanças na empresa.

- Como deveria se portar quando precisassem da empresa:
- Mostrar ao cliente se a solicitação é necessária e ele aguardar o tempo. O cliente deve sempre esperar pela sua vez,
- O analista verificará a solicitação e logo após dar o feedback ao cliente se esta solicitação será válida ou não. Por exemplo a solicitação é sobre um bug, que está atrapalhando cadastro e isso afeta o relatório do cliente, visto que é uma prioridade de correção a equipe de desenvolvimento fará a análise de tudo que está ocorrendo.

O problema mais significativo da empresa era que eles estavam sobrecarregados de trabalho e por serem poucas pessoas na equipe. Além disso, eles lidavam diretamente com demandas de problemas do cliente e a empresa tem o compromisso de responder a todas as demandas.

Após mudar a relação com clientes e a solicitações, começa-se limitar a quantidade de trabalho em andamento. As reuniões não poderiam ocorrer com muita frequência como no Scrum devido a um membro da empresa não poder estar lá, e assim ficou limitado. As Sprints foram definidas de acordo com 15 dias, onde o quadro foi montado, e qualquer dúvida problema seriam reportados e que todos soubessem o andamento, para não ter falhas de retrabalho ou esquecimento, foi utilizado como mecanismo de comunicação o Skype e o email.

As questões que surgiram nessa fase foram:

- Como administrar as solicitações dos clientes?
- Quem será responsável para liderar?
- Quais serão as funções do restante da equipe?
- Como organizar no quadro?
- Quanto tempo manteria o quadro?

Foi identificado que precisaria de uma auto-organização, aprendizagem constante e transferência de conhecimento pelos membros da equipe, pois é, essenciais para melhorar a comunicação e minimizar problemas futuros.

A princípio a equipe possui três membros e manterão sempre a comunicação, mesmo quando não estiverem presencialmente, ficou definido ainda liderança o analista de negócios e o engenheiro de software trabalhariam com as prioridades e após a análise seria repassado para o desenvolvedor. Foi apresentado as intenções deste trabalho para empresa, os benefícios esperados e as possíveis consequências.

A empresa sempre esteve disposta a seguir os métodos a risca, compram um quadro para gerenciar as solicitações e definiram as prioridades, estão utilizando a framework desde fevereiro de 2014.

## **4.5 Modelo de Processo de Gerenciamento de Comunicação**

A Modelagem de Processos de Negócio é um conjunto de atividades envolvidas na criação de representações de um processo de negócio existente ou proposto (ABPMP, 2009).

Segundo ABPMP(2009) “Modelo” é uma representação simplificada que suporta o estudo e o desenho de algum aspecto de algo, conceito ou atividade.

Neste estudo, pautado sobre o Gerenciamento da Comunicação, foi desenvolvido um modelo de processo de gerenciamento da Comunicação seguindo rigidamente a estrutura dos Métodos Ágeis *Scrum* e *Kanban*. A modelagem do processo foi essencial para a compreensão, documentação, análise, desenho, automatização e medição de atividade de negócio. O objetivo deste modelo é criar uma representação do processo que o descreva de forma necessária e suficiente para empresa poder entender e implanta-la.

### 4.5.1 Planejamento e Análise

Os processos possuem atributos e características que descrevem propriedades, comportamento, propósito, ou outros elementos de processo ABPMP(2009).

Os atributos do processo são capturados em uma ferramenta a fim de organizar, analisar e gerenciar os processos da organização. A análise de processos é realizada por várias técnicas que compreendem mapeamento, entrevista, simulações e diversas outras técnicas analíticas e metodologias assim como foi feita através da conversa com a empresa e com o questionário de identificação do problemas de comunicação.

Um dos primeiros passos do processo é estabelecer e atribuir os papéis, no qual foram definidos são: Analista de Negócios, Engenheiro de Software e o Programador.

A Sprint foi definida a cada quinze dias, para que pudesse ser conversado sobre as novas solicitações e como elas seriam classificadas. A classificação das prioridades das solicitações ficaram da seguinte forma:

1. Correção
2. Normas ou Leis
3. Solicitações comuns
4. Ordem de Chegada

O processo foi detalhado com ênfase da classificação das prioridades. Depois de ser detalhado o processo, foi desenvolvido um desenho na ferramenta BizAgil para criar um modelo.

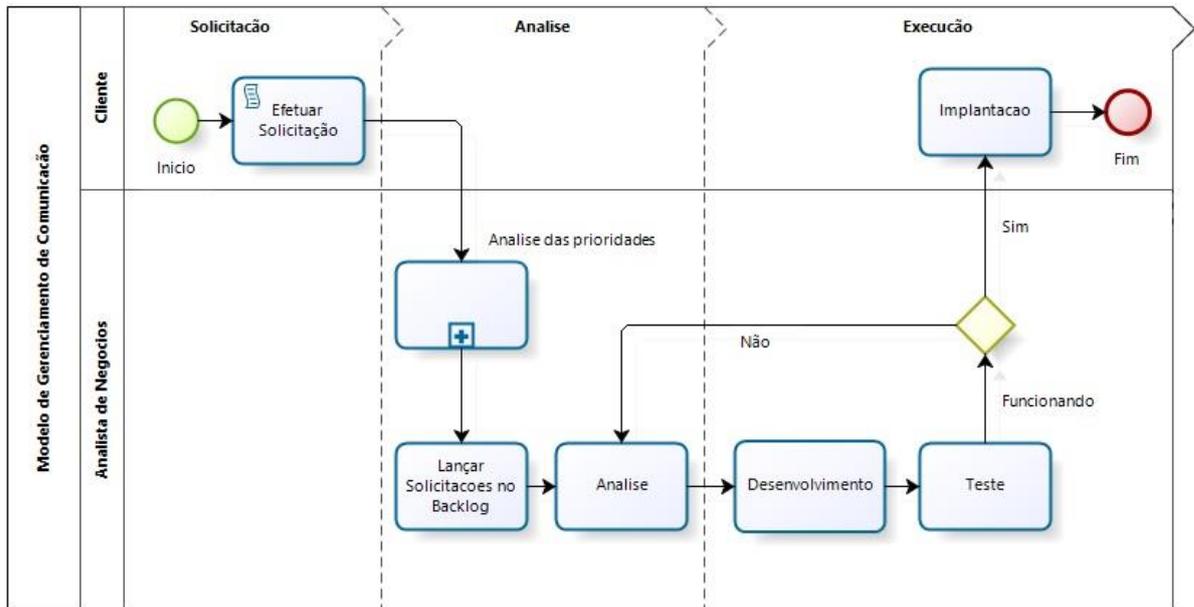
O cliente envia uma solicitação através do e-mail a qual, será adicionada a base de solicitações e a cada Sprint a equipe classificará de acordo com a prioridade e logo após essa classificação será passada para o quadro na fase do Backlog, e assim prosseguir para próximas fases.

#### 4.5.2 Desenho e modelagem de processos de negócio

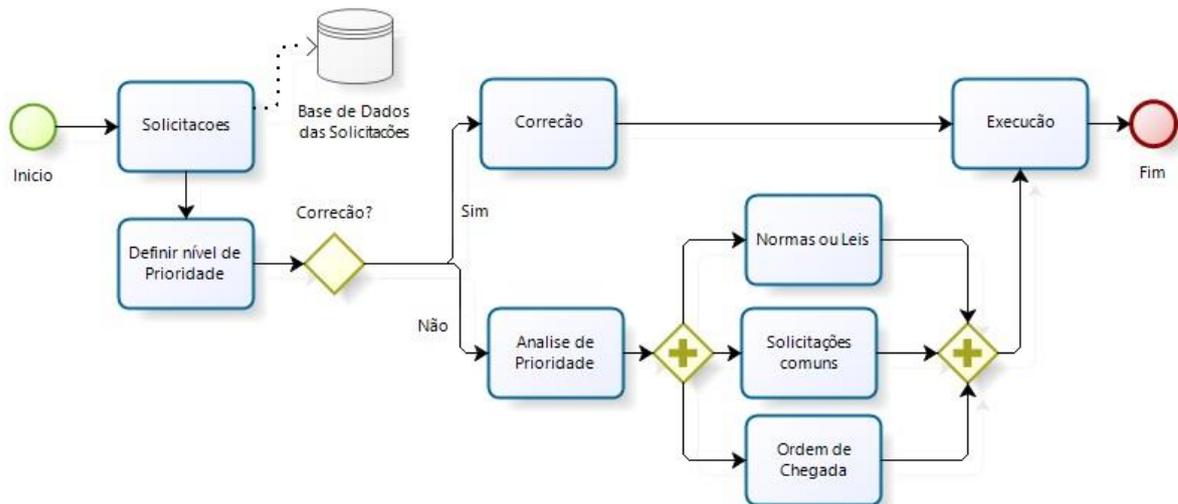
De acordo com ABPMP(2009), o desenho de processos envolve a criação de especificações para processos de negócio novos. Uma vez que o objetivo do gerenciamento de processos de negócio é garantir que os processos de uma organização sejam eficazes, ágeis e eficientes, a importância da fase de desenho não pode ser subestimada.

Antes de iniciar a Sprint, é preciso definir que histórias do *Product Backlog* serão implementadas no ciclo que está para ser iniciado. Para isso as histórias no *Product Backlog* devem estar em uma ordem de prioridade definida. A seguir estão os passos a serem realizados no processo desenho e modelagem seguindo as metodologia, a Figura 8 mostra como ficou o modelo de processo de gerenciamento de comunicação e a Figura 9 o subprocesso da análise de prioridades:

1. Cliente envia um arquivo de solicitação por *email*, no qual é especificado o assunto da solicitação e sua prioridade.
2. A empresa arquiva todas solicitações em uma base de dados.
3. A equipe classifica as solicitações por prioridades (correção, leis e normas, solicitações comuns e ordem de chegada).
4. Uma vez definido o *Sprint Backlog*, a *Sprint* pode então ser iniciada.
5. A *Sprint* será realizada a cada quinze dias.
6. Durante a execução da *Sprint*, a equipe executa as tarefas para a realização das histórias seguindo a ordem de prioridade.
7. Ao final da *Sprint* é realizada a reunião de Revisão. Nesta reunião, a equipe apresenta as histórias implementadas na *Sprint*, mostrando o seu resultado.
8. Quando o *Sprint* acaba, o quadro é limpo – todos os itens são removidos. O *Sprint* novo é iniciado e depois da reunião de planejamento do *Sprint*, temos um novo quadro, com novos itens na coluna mais à esquerda.



**Figura 8 - Modelo de Processo de Gerenciamento de Comunicação**



**Figura 9 - Subprocesso de Análise de Prioridade**

## 4.6 RESULTADOS OBTIDOS

A implantação das metodologias Scrum e Kanban na empresa JoinSoft teve um resultado considerado ótimo, eles executaram tudo que foi proposto para melhorar a comunicação, gerenciar os seus projetos, melhorar o relacionamento com clientes referente as solicitações. Foi comprado um quadro para implantação. A Figura 10 mostra o quadro já implantando.

Segundo a JoinSoft, com a metodologia, foi possível estabelecer um controle maior sobre os requisitos, além de organizar a execução dos mesmos, pois, em uma reunião, ao estabelecer os requisitos e quais fases eles estariam, os requisitos eram alterados somente na próxima reunião e, devido a isso, a equipe de gerência da empresa tinha total controle de quais atividades estavam em desenvolvimento, a empresa elaborou um relatório dos resultados obtidos da implantação empresa, que está no Anexo A.

A metodologia permitiu a melhoria da comunicação na empresa:

- Otimizou o controle de solicitações, todos sabem o que foi solicitado;
- Controle de implementação e gestão das solicitações;
- Análise pela equipe de gerência de projetos.

Sobre os problemas identificados no questionário de identificação que eram problemas de solicitações feitas diretamente ao programador que não são repassadas para equipe, problemas de conflito de decisões, problemas de falta de conhecimento de solicitações e Telefone sem fio.

Através da implantação muita coisa melhorou sobre questão da auto-organização da empresa, claro que no início gerou alguns problemas com clientes, pois estavam acostumados a sempre pedir solicitações urgentes sem necessidade, e querer resultados rápidos e no tempo de deles.

Já os problemas de conflitos de decisões e problemas de falta de conhecimento, ficou definido que todos deveriam estar a parte de tudo que estava acontecendo e sempre comunicar a equipe inteira da decisão para não haver problemas, mesmo as solicitações pequenas deveriam ser registradas, assim minimizando também problemas do “telefone sem fio”.

A empresa passou por organização nova, para se adequar aos métodos isso mudou muito a cultura organizacional, pois para o andamento dos métodos precisavam seguir os passos do trabalho. Todo ou qualquer trabalho enfrenta grandes problemas, pois lidam com pessoas e é complicado mudar de uma hora pra outra toda organização da empresa, visto que foi por justa causa, pois o objetivo era melhorar a comunicação da empresa.

O resultado da implantação demonstrou que ações foram benéficas e uma das grandes a dificuldade era de utilizar o quadro visto que a equipe está dispersa, ou seja, o programador, a analista de negócios e o engenheiro de software não se

encontram no mesmo ambiente físico com frequência. Esta situação dificulta a atualização do *Backlog* e a inclusão de mais dados que poderiam melhorar a gestão do projeto. Logo, propõe-se a criação de um software para automatizar este processo.



Figura 10 - Quadro Kanban implantado na empresa

## 5 CONSIDERAÇÕES FINAIS

A metodologia utilizada neste trabalho foi um estudo de caso, aplicado na empresa JoinSoft e o objetivo principal foi implantar as Metodologias Ágeis para gerenciar melhor a comunicação entre a equipe de desenvolvimento.

Todo trabalho foi baseado nas informações obtidas através do questionário, passando por análise, treinamento, implantação e resultados. A principal contribuição desta pesquisa para a empresa foi criação de estratégias de mudança cultural da organização, mas que também influenciará na diminuição das falhas, e manutenção, na melhoria da produtividade, na qualidade do produto e na satisfação dos clientes.

Foi ainda elaborado o modelo do processo de negócio, utilizando como base os métodos ágeis *Scrum* e *Kanban*, a fim de representar sistematicamente como funcionaria o gerenciamento da comunicação.

Conclui-se que o modelo de processo de negócio elaborado a partir dos métodos ágeis atinge seu objetivo de melhorar o gerenciamento de comunicação em equipes de desenvolvimento de software.

Sugere-se como trabalhos futuros, aplicar o modelo em outras empresas de desenvolvimento com números maiores de participantes na equipe ou até mesmo do mesmo tamanho para verificar se o método que foi utilizado pode ajudar outras empresas; criar um novo protótipo adequando com a necessidade da empresa, pode-se ainda acrescentar ou combinar outras metodologias para auxiliar a equipe como a metodologia XP, Lean e Crystal.

## REFERÊNCIAS

ABPMP (Associação Internacional de Profissionais de Gerenciamento de Processos de Negócio). Guia para o Gerenciamento de Processos de Negócio Corpo Comum de Conhecimento. Versão 2.0 – Terceira liberação em Português. 2009.

BIZAGI PROCESS MODELER. Disponível em:  
<<http://help.bizagi.com/processmodeler/en/>> Acesso em: 15 Maio 2014.

CUNHA, Cristiane Ribeiro da. FILIPAKIS, Cristina D'Ornellas. Proposta de Utilização de FDD e APF para Melhoria do Processo de Software. 2012. Disponível em:  
<[http://ulbra-to.br/encoinfo/artigos/2012/Proposta\\_de\\_Utilizacao\\_de\\_FDD\\_e\\_APF\\_para\\_Melhoria\\_do\\_Processo\\_de\\_Software.pdf](http://ulbra-to.br/encoinfo/artigos/2012/Proposta_de_Utilizacao_de_FDD_e_APF_para_Melhoria_do_Processo_de_Software.pdf)>. Acesso em: 30 set. 2013.

CARVALHO, Marly Monteiro de. MIRANDOLA, Daniela. A comunicação em projetos de TI: uma análise comparativa das equipes de sistemas e de negócios. Produção, v. 17, n. 2, p. 330-342, Maio/Ago. 2007. Disponível em: <<http://ppgep.poli.usp.br/2846/A-comunicacao-em-projetos-de-TI-uma-analise-comparativa-das-equipes-de-sistemas-e-de-negocios.h>> Acesso em: 01 out. 2013.

FADEL, Aline Cristine. SILVEIRA, Henrique da Mota. Metodologias Ágeis no contexto de desenvolvimento de software: XP, *Scrum* e Lean. 2010. Disponível em: <[http://www.ceset.unicamp.br/liag/Gerenciamento/monografias/Lean%20Agil\\_v8.pdf](http://www.ceset.unicamp.br/liag/Gerenciamento/monografias/Lean%20Agil_v8.pdf)>. Acesso em: 01 out. 2013.

FERREIRA, José Maria Carvalho. SERRA, Helena. Psicossociologia das organizações. 2006. Disponível em:<<http://www.iseg.utl.pt/aula/cad89/programa.pdf>> Acesso em: 29 out. 2013.

FOWLER, Martin. *The New Methodology*. 2005. Disponível em:  
<<http://martinfowler.com/articles/newMethodology.html>>. Acesso em: 30 set. 2013.  
FREIRE, Márcio Cerbella. Alegria + Comunicação = RESULTADO, 2012. Disponível em: <<http://blog.emecf.com.br/tag/efetividade-da-comunicacao/>> Acesso em: 01 out. 2013.

FREIRE, Márcio Cerbella. Alegria + Comunicação = RESULTADO. Disponível em:<<http://blog.emecf.com.br/tag/efetividade-da-comunicacao/>>. Acesso em: 30 out. 2013.

KNIBERG, Henrik. SKARIN. *Kanban e Scrum - Obtendo o melhor de ambos*, 2009. Disponível em: <<http://www.infoq.com/br/minibooks/Kanban-Scrum-minibook>> Acesso em: 31 out. 2013.

MARTINS, c. V. MISAGHI, D. M. Comunicação: Sucesso ou Fracasso em Projetos de TI? 2012. Disponível em: <[http://www.convibra.org/upload/paper/2012/32/2012\\_32\\_5111.pdf](http://www.convibra.org/upload/paper/2012/32/2012_32_5111.pdf)>. Acesso em: 30 ago. 2013.

MARIOTTI, Flavio S. *Kanban: o ágil adaptativo: Introduzindo Kanban na equipe ágil* Revista: Engenharia de Software Magazine. 2008. Disponível em: <<http://www.garcia.pro.br/EngenhariadeSW/artigosMA/A6%20-%2045-6-%20Kanbam.pdf>>. Acesso em: 01 set. 2013.

PAULA, Renata de Souza Alves. *Scrum na Melhoria do Gerenciamento de Projetos de Software: Um estudo sobre a implantação de Scrum para otimizar o processo de gerenciamento de projetos de software*. Engenharia de Software Magazine. 2008. Disponível em: <<http://www.garcia.pro.br/EngenhariadeSW/artigosMA/A5%20-%2023-8%20-%20Scrum.pdf>>. Acesso em: 01 set. 2013.

PRIKLADNICKI, Rafael. Problemas, Desafios e Abordagens do Processo de Desenvolvimento de Software. Porto Alegre, 13 de fevereiro de 2004. Disponível em: <<http://www.inf.pucrs.br/munddos/docs/TI1.pdf>> Acesso em: 02 set. 2013.

ROSE, Tanara Priscilla Ribeiro. MELLO, Carlos Henrique Pereira. Proposta de sistemática para gestão de projetos baseada na Metodologia Ágil *Scrum*. Enegep 2010. São Carlos, SP, Brasil, 12 a 15 de outubro de 2010. Disponível em: <[http://www.labceo.com.br/bibliografia/archive/files/e-9\\_b805d76db7.pdf](http://www.labceo.com.br/bibliografia/archive/files/e-9_b805d76db7.pdf)>. Acesso em: 28 set. 2013.

SANTOS, Jorge. QUINTILIANO, Leandro Lago. BARBOSA, Lílian da Silva, Paulo Roberto H. Júnior, Wanderley Giralde. Metodologias Ágeis – Parte II. 2011. Disponível em: <<http://www.devmedia.com.br/articles/post-9443-MAgeis-Parte-02.html>>. Acesso em: 28 set. 2013.

SCHWABER, K.; SUTHERLAND, J. Guia do SCRUM. 2011. Disponível em: <<https://www.Scrum.org/Portals/0/Documents/Scrum%20Guides/Scrum%20Guide%20-%20Portuguese%20BR.pdf>>. Acesso em: 29 Set. 2013.

SIQUEIRA, Fábio Levy. Métodos Ágeis 2003. Disponível em: <[http://www.levysiqueira.com.br/artigos/metodos\\_ageis.pdf](http://www.levysiqueira.com.br/artigos/metodos_ageis.pdf)>. Acesso em: 29 set. 2013.

SOARES, Felipe S. Furtado. MARIZ, Leila M. Rodrigues de Sousa. CAVALCANTI. Adoção de *SCRUM* em uma Fábrica de Desenvolvimento Distribuído de Software, 2007. Disponível em: <<http://www.lbd.dcc.ufmg.br/bdbcomp/servlet/Trabalho?id=6894>>. Acesso em: 29 Set. 2013.

STONER, James A. F. e FREEMAN, R. Edward. Administração. Rio de Janeiro: Prentice Hall do Brasil, 1995.

TANAKA, Sérgio Akio. BANKI André Luiz Metodologias Ágeis: Uma Visão Prática. Revista Engenharia de Software Magazine – Metodologias Ágeis. 2008. Disponível em: <<http://www.garcia.pro.br/EngenhariadeSW/artigosMA/A1%20-%204-22%20-%20Metodos%20ageis%20uma%20visao%20pratica.pdf>> Acesso em: 29 Set. 2013.

TERCIOTTI, Sandra Helena. MACARENCO, Isabel. Comunicação Empresarial na Prática. São Paulo: Saraiva, 2009. 204p. Disponível em: <[http://www.4shared.com/zip/cjku-NQp/TERCIOTTI\\_COMUNICAO\\_EMPRESARIA.htm](http://www.4shared.com/zip/cjku-NQp/TERCIOTTI_COMUNICAO_EMPRESARIA.htm)> Acesso em: 29 Set. 2013.

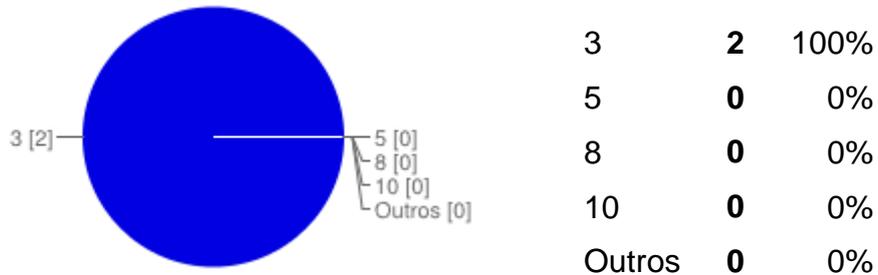
TORRES, Adriano. A Importância Da Comunicação Nas Organizações. 2013. Disponível em: <<http://www.blogdaqualidade.com.br/a-importancia-da-comunicacao-nas-organizacoes/>>. Acesso em: 7 nov. 2013.

TRINDADE, Daniela de Freitas Guilhermino. Uma Ferramenta Para Gerenciar A Comunicação Em Um Ambiente Distribuído De Desenvolvimento De Software. Disponível em: <<http://www.din.uem.br/pos-graduacao/mestrado-em-ciencia-da-computacao/arquivos/dissertacoes-1/Daniela%20Freitas%20Guilhermino%20Trindade.pdf>> Acesso em: 05 jun. 2014.

## APÊNDICE A

## RESPOSTA AO QUESTIONÁRIO DE IDENTIFICAÇÃO DE PROBLEMAS DE COMUNICAÇÃO

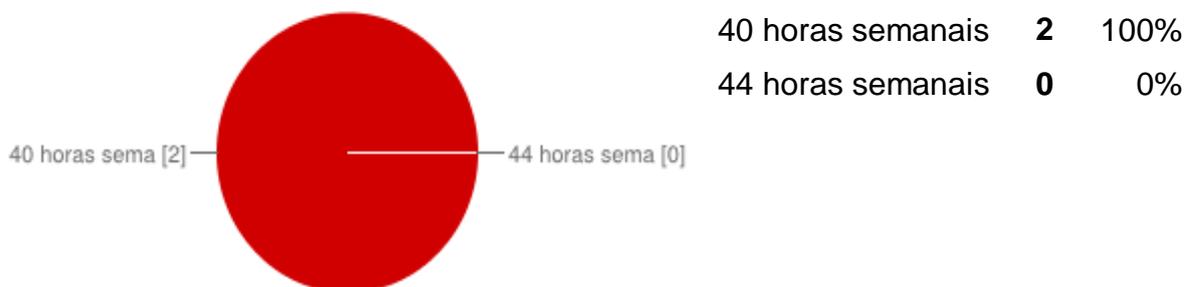
### 1) Quantas pessoas compõem a equipe?



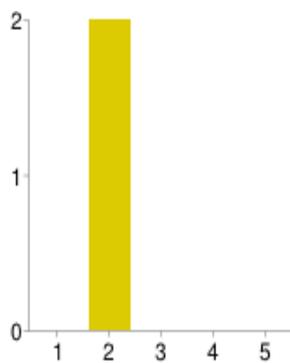
### 2) Quais cargos são definidos na equipe de desenvolvimento?

Programador	2	40%
Designer de Telas	0	0%
Tester	0	0%
DBA	0	0%
Suporte	0	0%
Gerente de Projeto	1	20%
Outros	2	40%

### 3) Qual é a carga horaria da empresa?



#### 4) Fazem reuniões com frequência?



1	0	0%
2	2	100%
3	0	0%
4	0	0%
5	0	0%

#### 5) Como é a distribuição das tarefas?

Analista de Negócios: Identifica problemas nas organizações, faz a comunicação com os clientes, efetua levantamento de requisitos, implanta e analisa resultados do recurso desenvolvido. Gerente de Projeto/Analista de Sistemas: Transforma os requisitos em requisitos de software, efetua modelagem UML, planejamento e distribuição de atividades, Testar funcionalidades do software antes da entrega final. Desenvolvedor/Programador: Implementa solicitações e recursos no software, gerando códigos, efetuar testes unitários e testes funcionais. Importação e transição de sistemas legados. Analista de Negócios: Comercial, contato direto com o cliente, levantamento de necessidades, ajuste do conceito software + necessidade organizacional. Teste Funcional de recursos implementados para validar se está de acordo com a solicitação do cliente, Suporte ao Usuário Engenheiro de Software: Análise, modelagem, teste e gestão do projeto de software. Programador: Desenvolvimento e Teste de recursos Computacionais. Suporte ao usuário.

#### 6) Quais problemas de problemas de comunicação são mais frequentes?

Omissão de informação	0	0%
Rejeição entre a equipe	1	14%
Ruído nas informações	2	29%
Metas e prazos inalcançáveis	1	14%
Falta de padronização da linguagem	0	0%
Desunião da equipe	0	0%
Omissão do gerente ou da equipe	0	0%
Falta de uma hierarquia na recepção das informações	2	29%
Outros	1	14%

### 7) Quais são dificuldades encontradas em trabalhar em equipe?

Omissão de informação	0	0%
Falta de compromisso	0	0%
Desunião da equipe	0	0%
Duplicação de esforços	1	50%
Outros	1	50%

### 8) Como fazem para resolver problemas de comunicação?

Reuniões periódicas (embora poucas). Reunião de equipe, Documentar (escrever) decisões tomadas Reuniões com todos os membros da equipe e pessoas externas colocando o assunto em pauta para ser analisado e identificado possíveis falhas.

### 9) A empresa utiliza alguma ferramenta ou mecanismo de comunicação entre os participantes da equipe?

Reuniões	1	25%
Chat	1	25%
E- mail	1	25%
Videoconferência	0	0%
Outros	1	25%

### 10) A empresa trabalha com alguma metodologia ágil para gerenciamento do projeto?

	0	0%
Extreme Programming (XP)		
SCRUM	1	50%
Lean Software Development (LSD)	0	0%
Kanban	0	0%
Outros	1	50%

## APÊNDICE B



## APÊNDICE C

## DETALHAMENTO DO MODELO DE PROCESSO

### Elementos do Processo

 Início

#### Descrição

Início do Processo de Negócio

 Efetuar Solicitação

#### Descrição

Cliente envia um e-mail com um formulário de solicitação explicando a causa.

 Analise das prioridades

 Lançar Solicitações no *Backlog*

#### Descrição

Lança os cartões no quadro Kanban

 Analise

#### Descrição

Cartões da solicitação

 Desenvolvimento

#### Descrição

A solicitação será entregue ao programador para fazer as devidas correções.

 Teste

#### Descrição

Onde o Tester fará os teste antes para implantação do Cliente

 Funcionando

**Portões****Não****Sim**

Implantação

**Descrição**

Fase de implantação ao cliente.



Fim

**Descrição**

Termino do Processo de Gerenciamento de Comunicação

**Elementos do Subprocesso**

Início

**Descrição**

Início do subprocesso de análise das prioridades



Solicitações

**Descrição**

Solicitação enviada pelo cliente



Definir nível de Prioridade

**Descrição**

A Equipe defini o nível da prioridades



Correção?

**Descrição**

A correção de erros é a definida como prioridade mais importante e urgente

**Portões****Não**

**Sim**

Análise de Prioridade

**Descrição**

Análise das prioridades secundárias

- 1- Normas ou Leis
- 2- Solicitações comuns
- 3- Ordem de chegada

 Gateway Exclusivo

**Portões**

**Ordem de Chegada**  
**Solicitações comuns**  
**ou**

Ordem de Chegada

**Descrição**

Análise das prioridades secundárias

 Gateway Paralelo

Execução

**Descrição**

Execução da solicitação que será enviada para o quadro Kanban

 Fim

**Descrição**

Termino do subprocesso de análise das prioridades

Solicitações comuns

Normas ou Leis

**Descrição**

Análise das prioridades secundárias



Correção

**Descrição**

A correção de erros é definida como prioridade mais importante e urgente



DataStore

**Descrição**

Base de Dados que contém todas as solicitações enviadas pelos clientes.

## ANEXO A

## RELATÓRIO DA EMPRESA

O processo de implantação do modelo na JoinSoft seu iniciou em dezembro de 2013, época em que ocorreram os detalhamentos dos problemas de comunicação e a proposta de aplicação do modelo.

Este período foi especialmente escolhido, pois se iniciava uma fase com funcionários novos, sem costumes e que seriam treinados dentro dos procedimentos da empresa.

Após a implantação do quadro do Kanban foi possível reduzir em 90% os problemas de comunicação existentes na empresa, principalmente os problemas relacionados às solicitações de novos recursos para o sistema.

Com a metodologia, foi possível estabelecer um controle maior sobre os requisitos além de organizar a execução dos mesmos, pois, em uma reunião, ao estabelecer os requisitos e quais fases eles estariam, os requisitos eram alterados somente na próxima reunião e, devido a isso, a equipe de gerencia da empresa tinha total controle de quais atividades estavam em desenvolvimento.

A situação atual do cliente perante a empresa foi outro controle que a metodologia permitiu melhorar. Antes de aplicar a tecnologia, era difícil saber como estava o relacionamento do cliente com a empresa, situação gerada com a falta de comunicação entre programador, analista de negócios e engenheiro de software. Com a metodologia, foi possível implantar uma ação de pós vendas (pós solicitações) melhorando a comunicação e o relacionamento cliente empresa.

Por fim, a metodologia permitiu a melhoria da comunicação na empresa e otimizou as seguintes ações:

- Controle de solicitações: Todos sabem o que foi solicitado.
- Controle de implementação: Gestão das solicitações em desenvolvimento ou em análise pela equipe de gerência de projetos.
- Organização: Com os controle implementados, ocorreu um controle das ações da empresa e assim evitou-se atividades duplicadas, desnecessárias ou fora de ordem.

Como mencionado anteriormente, 90% das ações foram benéficas. Os outros 10% se referem a dificuldade de utilizar o quadro visto que a equipe está dispersa, ou seja, o programador, a analista de negócios e o engenheiro de software não se

encontram no mesmo ambiente físico com frequência. Esta situação dificulta a atualização do *Backlog* e a inclusão de mais dados que poderiam melhorar a gestão do projeto. Logo, propõe-se a criação de um software para automatizar este processo.