



**UNIVERSIDADE DO SUL DE SANTA CATARINA  
WILIAN VARGAS DE LIMA**

**SCRUM NO BRASIL**

**Florianópolis  
2011**

**WILIAN VARGAS DE LIMA**

**SCRUM NO BRASIL**

Monografia apresentada ao Curso de Especialização em Engenharia de Projetos de Software da Universidade do Sul de Santa Catarina, como requisito à obtenção do título de Especialista em Engenharia de Projetos de Software.

Orientador: Prof. Dr. Nikolai Albuquerque

Florianópolis  
2011

**WILIAN VARGAS DE LIMA**

**SCRUM NO BRASIL**

Esta Monografia foi julgada adequada à obtenção do Título de Especialista em Engenharia de Projetos de Software e aprovada em sua forma final pelo Curso de Especialização em Engenharia de Projeto de Software da Universidade do Sul de Santa Catarina.

Florianópolis, 27 de junho de 2011.

---

Professor e orientador Nikolai Albuquerque, Dr.  
Universidade do Sul de Santa Catarina

---

Professora Vera R. N. Schuhmacher, Msc.  
Universidade do Sul de Santa Catarina

## **AGRADECIMENTOS**

Aos meus pais que sempre foram exemplo para minha vida, e sempre incentivaram a continuar estudando e aperfeiçoamento meus conhecimentos.

A Sandra minha companheira que sempre me apoiou nas dificuldades.

Ao professor e orientador Nikolai Albuquerque, pela grande colaboração.

“A vida não dá, nem empresta, não se comove, nem se apieda.  
Tudo quanto ela faz é retribuir e transferir aquilo que nós lhe oferecemos.”  
**(Albert Einstein)**

## RESUMO

Este estudo visa demonstrar a importância da metodologia ágil Scrum nos ambientes de desenvolvimento de softwares brasileiros. Fundamentando os conhecimentos sobre as metodologias ágeis bem como os Princípios Ágeis. Além disso, demonstrando claramente o funcionamento da metodologia Scrum, em todas as suas partes e processos para o seu bom uso e todo proveito das opções que o mesmo disponibiliza no desenvolvimento dos projetos.

A investigação se deu pelo uso da técnica de questionário, os resultados são apresentados apontando o potencial no modelo catarinense.

Palavras-chave: Metodologias ágeis. Scrum. Desenvolvimento de softwares.

## **ABSTRACT**

This study aims to demonstrate the importance of the Scrum agile software development environments in Brazil. Basing knowledge about agile methodologies and agile principles. Moreover, clearly demonstrating the operation of the Scrum methodology, in all its parts and processes for its proper use and advantage of all options that it offers in project development.

The investigation took place by using the technique of the questionnaire, the results are presented in the model indicating the potential of Santa Catarina.

Keywords: Agile methodology. Scrum. Software development.

## LISTA DE FIGURAS

Figura 1: Exemplo de Planejamento dos itens do Product Backlog .....	24
Figura 2: Exemplo de Planejamento dos itens do Product Backlog .....	25
Figura 3: Exemplo de Product Backlog .....	27

## LISTA DE GRÁFICOS

Gráfico 1: Número de funcionários da empresa .....	33
Gráfico 2: Tipo de produto desenvolvido .....	34
Gráfico 3: Utiliza metodologias ágeis no desenvolvimento de softwares .....	35
Gráfico 4: Tempo que a empresa utiliza essa metodologia.....	36
Gráfico 5: Metodologia utilizada .....	37
Gráfico 6: Utiliza o Scrum com alguma outra metodologia – framework .....	39
Gráfico 7: Metodologias utilizadas com Scrum.....	40
Gráfico 8: Práticas Scrum utilizadas na empresa .....	41
Gráfico 9: Práticas de testes utilizadas.....	42
Gráfico 10: Melhora na qualidade do produto .....	43
Gráfico 11: Satisfação da equipe antes da utilização do Scrum.....	45
Gráfico 12: Satisfação da equipe com a utilização do Scrum.....	46
Gráfico 13: Satisfação da gerência antes da adoção do Scrum.....	47
Gráfico 14: Satisfação da gerência após a adoção do Scrum.....	48

## SUMÁRIO

<b>1 INTRODUÇÃO</b> .....	<b>11</b>
1.1 OBJETIVOS .....	12
1.1.1 Objetivo geral .....	12
1.1.2 Objetivos específicos.....	12
1.2 APRESENTAÇÃO DO PROBLEMA .....	12
1.3 JUSTIFICATIVA.....	13
1.4 ESTRUTURA DO TRABALHO.....	13
<b>2 REVISÃO BIBLIOGRAFICA</b> .....	<b>14</b>
2.1 FUNDAMENTOS .....	14
2.2 OS 12 PRINCÍPIOS ÁGEIS .....	15
2.3 MODELO SCRUM .....	17
2.3.1 Papéis No Scrum.....	19
2.3.2 Artefatos do Scrum.....	26
2.3.3 Pronto.....	29
<b>3 METODOLOGIA DA PROPOSTA</b> .....	<b>30</b>
3.1 ASPECTOS METODOLÓGICOS .....	30
3.2 METODOLOGIA PROPOSTA PARA EXECUÇÃO DO PROJETO .....	31
<b>4 DESENVOLVIMENTO</b> .....	<b>32</b>
4.1 IDENTIFICAÇÃO DO PÚBLICO ALVO .....	32
4.2 CONSIDERAÇÕES FINAIS SOBRE OS RESULTADOS DO .....	49
QUESTIONÁRIO .....	49
<b>5 CONCLUSÃO</b> .....	<b>51</b>
<b>REFERÊNCIAS</b> .....	<b>52</b>
<b>APÊNDICE A – QUESTIONÁRIO PARA COLETA DE DADOS</b> .....	<b>53</b>

## 1 INTRODUÇÃO

Desde que os computadores passaram a ser utilizados tenta-se criar softwares como meios de utilizar esse hardware que foi apresentado à disposição do ser humano.

A grande diferença no passar das décadas é que procura-se não mais somente realizar a tarefa desejada, mas sim realizá-la com clareza, rapidez e qualidade.

Empresas de desenvolvimento de software começaram a surgir para atender o mercado brasileiro, e com a demanda e procura, desenvolver apenas “sentando e programando” não é mais uma opção viável, pelo menos para quem quer crescer e oferecer o melhor produto.

Precisa-se ter organização e planejamento para que as coisas simplesmente não fujam o controle, nesse momento encaixam-se as metodologias de desenvolvimento de softwares.

Existem diversas metodologias hoje no mundo, classificadas entre tradicionais e ágeis, todas tendo grande importância e utilização dependendo do que as empresas buscam.

O Scrum trata-se de um framework de práticas ágeis de gerenciamento de projeto, um processo a seguir para o desenvolvimento de softwares, funcionando como as “rédeas” do projeto para que este tenha metas bem definidas, buscando manter o cliente ciente da situação do sistema ao qual requisitou.

As metodologias ágeis procuram ter uma estrutura e organização apenas no nível de sua necessidade, para assim ser realmente ágil e não confuso.

## 1.1 OBJETIVOS

Os objetivos do projeto estão classificados em objetivo geral e objetivos específicos.

### 1.1.1 Objetivo geral

Esse projeto busca identificar como a metodologia ágil Scrum pode contribuir em empresas de desenvolvimento de softwares brasileiras.

### 1.1.2 Objetivos específicos

Os objetivos específicos deste trabalho são:

- Aprofundar os conhecimentos sobre metodologias ágeis e os 12 princípios ágeis.
- Compreender o processo de desenvolvimento baseado no modelo Scrum.
- Realizar uma pesquisa identificando quais práticas do Scrum são aplicadas na integra em empresas de desenvolvimento de software.
- Identificar características facilitadoras e de desafio que o Scrum promove nos ambientes empresariais brasileiros.

## 1.2 APRESENTAÇÃO DO PROBLEMA

O desenvolvimento de software no Brasil tem crescido de forma espantosa, e realizar o mesmo sem uma forma de controlar e gerenciar os projetos tem se mostrado falho. Diante das constantes mudanças mundiais é necessário controlar e racionalizar os recursos disponíveis.

O presente trabalho apresenta o modelo de desenvolvimento ágil Scrum, utilizado no gerenciamento dos projetos de softwares, demonstrando quais são suas vantagens e desvantagens e o que leva uma empresa a adotá-lo.

### 1.3 JUSTIFICATIVA

Este trabalho busca esclarecer e ampliar o conhecimento sobre essa metodologia ágil que é o Scrum, e que tem se tornado tão popular no mundo e no Brasil.

Também identificar no mercado de desenvolvimento de softwares o que está sendo utilizado nos planejamentos dos projetos, e que está dando certo na maioria das empresas. Além disso, verificar quais são as facilidades e dificuldades que se apresentam ao implantar o Scrum em empresas que tinham alguma ou nenhuma metodologia para os seus projetos.

### 1.4 ESTRUTURA DO TRABALHO

A monografia é apresentada em quatro capítulos. No primeiro capítulo foram apresentados informações de caráter introdutório.

No capítulo dois são abordados os principais tópicos que dão fundamentação teórica ao trabalho como os fundamentos sobre metodologias ágeis, os princípios ágeis, e o modelo Scrum.

No capítulo três é discutida a metodologia proposta para o trabalho e posteriormente o relato da aplicação dela.

O capítulo quatro apresenta as conclusões da monografia.

## 2 REVISÃO BIBLIOGRAFICA

O planejamento e organização dos projetos de softwares tem se mostrado de grande importância, independente de qual seja o tamanho do programa a ser desenvolvido.

Neste ponto serão abordados os conceitos presentes neste trabalho, suas fundamentações e definições, apresentando as metodologias ágeis, os 12 princípios ágeis e principalmente a compreensão dos papéis e como funciona o processo de desenvolvimento no modelo Scrum.

### 2.1 FUNDAMENTOS

Atualmente os softwares têm assumido um papel de extrema importância em nossa sociedade, pode ser um simples controle de estoque, ou um sistema de controle aéreo, as opções e utilidades têm demonstrado ser ilimitadas. Nesse contexto fica difícil ter aquele programador, trabalhando em uma sala fechada, decidindo sozinho como desenvolver o software, essa cena foi substituída por equipes capacitadas e que buscam trabalhar com processos bem definidos.

Neste ponto podemos mencionar as metodologias de desenvolvimento de software, que tratam “do que”, “como” e “o que” cada um vai fazer no projeto.

Existem duas linhas nas metodologias que são as tradicionais, e as ágeis no qual o Scrum (tema desse projeto) se encaixa. Basicamente nas metodologias tradicionais procura-se entregar o software na sua totalidade, mesmo que existam mudanças no escopo ou qualquer outra mudança importante no percurso, o projeto só é definido como bem sucedido no final do mesmo.

Nas metodologias ágeis procura-se entregar partes funcionais para o cliente ao longo do projeto todo, entregando “valor” a ele, e caso o projeto venha a ser descontinuado ou cancelado o cliente já recebeu algo funcional, não vai perder seu investimento até então, como definem Shore e Warden (2008, p.18):

Os métodos ágeis também determinam expectativas no início do projeto, portanto, se seu projeto não será um sucesso organizacional, você vai descobrir cedo o suficiente para cancelá-lo antes que sua empresa tenha gastado muito dinheiro.

Mas de todo modo as metodologias ágeis não podem ser pensadas como a solução para todos os problemas, é apenas uma maneira de fazer as coisas, uma maneira de trabalhar, e até que toda uma equipe adapte-se, ela vai ser mais lenta e menos produtiva do que tenha sido até então.

Shore e Warden (2008, p.15) sugerem que para antes de cogitar a idéia de utilização de desenvolvimento ágil deve-se se ter em mente a resposta para a pergunta: “O desenvolvimento ágil nos ajudará a ter mais sucesso?”.

Esse sucesso pode ser classificado ou dividido no sucesso organizacional tanto da empresa, como o valor do seu produto.

O sucesso técnico, que nada mais é do que os programadores poderem considerar seus códigos de excelente qualidade.

O sucesso pessoal, no qual cada um deve encontrar-se satisfeito e realizado com o seu trabalho na utilização deste tipo de metodologia.

Como salientam Shore e Warden (2008, p.15):

Sem o sucesso pessoal, você terá problemas em motivar a si e aos seus funcionários. Sem o sucesso técnico, seu código fonte um dia irá desmoronar sob seu próprio peso. Sem o sucesso organizacional, sua equipe pode achar que não é mais desejada na empresa.

O desenvolvimento ágil concentra-se em atingir esses pontos, então é possível que ele possa ajudar a empresa que estiver com problemas em atingir algum desses “sucessos”.

## 2.2 OS 12 PRINCÍPIOS ÁGEIS

Os doze princípios ágeis são parte do Manifesto Ágil, o qual foi criado por desenvolvedores e gestores muito respeitados pelo mundo, os quais tentaram passar a visão de uma equipe de desenvolvimento de software. O manifesto esta disponível na internet em pelo menos 20 línguas, e disponível para ser traduzido em qualquer país que tenha interesse.

O manifesto antes dos princípios determina quatro valores, que como descrito no manifesto, fazendo software por si mesmo e ajudando outros a fazer, passaram a valorizar (<http://manifestoagil.com.br>, 2001):

1. **Indivíduos e interação entre eles** mais que processos e ferramentas.
2. **Software em funcionamento** mais que documentação abrangente.
3. **Colaboração com o cliente** mais que negociação de contratos.
4. **Responder a mudanças** mais que seguir um plano.

São estabelecidos então os princípios que servem como alicerce para todos que utilizam desenvolvimento ágil, os quais podem ser encontrados no site <http://manifestoagil.com.br>. Os doze princípios são:

- 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.

Tanto os valores como os princípios apenas demonstram que chegou-se a conclusão de que a solução para um software de qualidade, que agrada tanto o cliente como a equipe que o desenvolveu não esta apenas em se ater aos papéis, documentos e relatórios gerados. Trata-se de manter o cliente atualizado, envolvido ao projeto, e vendo os resultados, tornando o software de certa forma palpável o mesmo. E claro o querer de fazer a excelência no que esta se trabalhando.

## 2.3 MODELO SCRUM

As informações a cerca do modelo Scrum, foram principalmente baseadas no Scrum Guide, que como o nome define serve de guia para todos que tem o interesse de utilizar o framework, o qual foi desenvolvido e é mantido por Ken Schwaber e Jeff Sutherland, gestores de grande conhecimento e respeito em todo o mundo, sendo grandes contribuidores do Scrum e do manifesto ágil.

O Scrum é utilizado desde a década de 90, formado por práticas e técnicas comprovadas por anos. Foi realmente apresentado e publicado na OOPSLA de 1995, a qual é uma conferência dedicada à programação orientada a objetos, e desde então vem sendo aprimorado.

Trata-se de uma metodologia ágil, a qual funciona como um guia para o “caminho das pedras” no gerenciamento e desenvolvimento de projetos de softwares, não é um processo que pode ser seguido e vai dizer exatamente como fazer, ou como deve ser trabalhado na empresa, e sim um framework.

Empregando uma abordagem incremental e iterativa para otimização e controle dos riscos, o Scrum tem como base a teoria de controle de processos empíricos, os quais têm três pontos de base que são a Transparência, a Inspeção e a Adaptação. (Schwaber e Sutherland, 2010).

Transparência: Os processos devem ser claros e visíveis para quem esta gerenciando os resultados.

Inspeção: Como o nome determina, é necessário realizar inspeções nos processos, para que possa ser detectado o quanto antes qualquer problema nos mesmos. A reunião diária trata-se de um tipo de inspeção.

Adaptação: Determina através da inspeção, se os processos apresentam problemas que podem levar a um produto fora das especificações, deve tentar adaptar e ajusta o processo para que isso não venha a ocorrer.

Além disso Schwaber e Sutherland(2010, p.4) salientam que:

Existem três pontos para inspeção e adaptação em Scrum.

A Daily Scrum é uma reunião utilizada para inspecionar o progresso em direção à Meta da Sprint e para realizar adaptações que otimizem o valor do próximo dia de trabalho.

Além disso, as reuniões de Revisão da Sprint e de Planejamento da Sprint são utilizadas para inspecionar o progresso em direção à Meta da Release e para fazer as adaptações que otimizem o valor da próxima Sprint. Finalmente, a Retrospectiva da Sprint é utilizada para revisar a Sprint passada e definir que adaptações tornarão a próxima Sprint mais produtiva, recompensadora e gratificante.

O Scrum é basicamente dividido por Equipes Scrum com seus membros tendo papéis definidos, pelos Times Boxes que são as reuniões de planejamento, releases, sprints e por fim os Artefatos e Regras.

De acordo com Schwaber e Sutherland(2010, p.4) as Equipes de Scrum tem em sua formação três papéis:

O Product Owner: define os itens que devem ser realizados pela Equipe Scrum.

O ScrumMaster: assegura que a equipe vai seguir as práticas e processos do Scrum.

A Equipe Scrum: são as pessoas que realmente colocam a “mão na massa”, que executam o trabalho e tornam os requisitos em um produto.

A partir dos Times Boxes é que se tem uma regularidade no projeto, os quais são os elementos fixos como: Reunião de Planejamento da Release, reunião de Planejamento da Sprint, a Sprint em si, as Reuniões Diárias(Daily Scrum), a Revisão da Sprint e a Retrospectiva da Sprint.

Dentre todas as Times Boxes a Sprint pode-se considerar a mais importante, pois é a iteração previamente definida, em que durante esse período é feito o esforço para o desenvolvimento do que foi planejado para essa iteração e sendo uma parcela do produto final busca-se que sejam partes funcionais. As sprints funcionam uma após a outra, e mesmo que um produto não tenha sido concluído em sua iteração estipulada, pode continuar sendo trabalhada na próxima.

Como definem Schwaber e Sutherland(2010) os artefatos são formados por:

Product Backlog – Trata-se de uma lista contendo as funcionalidades desejadas.

Sprint Backlog – É uma lista de tarefas feita a partir do Product Backlog para ser desenvolvida dentro da Sprint.

Burndown de Release – mensura o Product Backlog ao longo do tempo.

Burndown de Sprint – mensura os itens da Sprint Backlog restante ao longo das sprints.

Já as Regras, funcionam como o próprio nome determina, são regras para manter todos os pontos funcionando bem e ligados.

Como exemplo, uma regra do Scrum diz que “somente membros do Time - as pessoas comprometidas em transformar o Product Backlog em um incremento - podem falar durante uma Daily Scrum”. (Schwaber e Sutherland, 2010, p. 5).

### 2.3.1 Papéis No Scrum

a) **ScrumMaster:** É aquele que faz com que a equipe siga as práticas, valores e regras do Scrum. Também está na equipe para ajudar e tirar do caminho da equipe qualquer obstáculo que venha a comprometer o trabalho e entregas da sprint.

Busca fazer com que a equipe torne-se cada vez mais auto-organizável e multidisciplinar.

Schwaber e Sutherland(2010, p.6) sugerem que:

O ScrumMaster pode ser um membro do Time; por exemplo, um desenvolvedor realizando tarefas da Sprint. No entanto, isso frequentemente leva a conflitos quando o ScrumMaster deve escolher entre remover impedimentos e realizar tarefas. O ScrumMaster nunca deve ser o Product Owner.

b) **Product Owner:** O Product Owner determinar os itens que compõe o Product Backlog, também priorizando os itens da mesma, qualquer repriorização deve passar pelo Product Owner e para que isso funcione corretamente, suas decisões devem ser respeitadas por todos os envolvidos no projeto.

Segundo a Improve It (2007) O Product Owner além de definir os itens que compõem o Product Backlog deve se comprometer a não trazer novos requisitos enquanto a equipe esta no meio de uma sprint.

Schwaber e Sutherland(2010, p.6) comentam que:

O Product Owner pode ser um membro do Time, trabalhando também em desenvolvimento. Mas essa responsabilidade adicional pode reduzir a sua habilidade de lidar com as partes interessadas. No entanto, o Product Owner nunca pode ser o ScrumMaster.

c) **O Time ou Equipe:** São as pessoas que realizam todos os itens do Product Backlog, finalizando e entregando uma parte dele em cada Sprint.

Normalmente são formados por desenvolvedores, com variados conhecimentos que se complementam para o desenvolvimento das tarefas.

O time ainda pode dividir-se em times menores, para a realização de testes e garantir a qualidade e a análise de negócios.

Normalmente são auto-organizáveis, e o como o Scrum Guide (2010,8) recomenda, “nem mesmo o ScrumMaster – diz ao time como transformar o Product Backlog em incrementos de funcionalidades entregáveis.”

Em relação ao tamanho que as equipes devem ter, Schwaber e Sutherland( 2010, p. 8), afirmam como um tamanho “ótimo”, o de 7 pessoas. Menor que 5 pessoas pode haver falta de conhecimento e menor interação, já mais do que nove membros torna-se um grupo muito grande para ser gerenciado.

d) **Time Boxes:** São os eventos com duração fixa, como as Reuniões de planejamento de Sprints e Release, Daily Scrum(Reuniões diárias), Revisão da Sprint, Retrospectiva da Sprint, e a própria Sprint.

e) **Reunião de Planejamento da Release:** Nas reuniões de planejamento da release são estabelecidos os planos e metas com a equipe Scrum e o restante da organização, buscando encontrar as melhores maneiras de atingir ou até superar as expectativas do cliente, bem como o melhor retorno sobre o investimento.

Segundo Schwaber e Sutherland( 2010, p. 9), “o plano de release estabelece a meta de release, as maiores prioridades do Product Backlog, os principais riscos e as características gerais e funcionalidades que estarão contidas na release.” Estabelece também uma data de entrega e custos prováveis. Assim torna-se possível inspecionar o progresso e realizar mudanças no plano da release em cada sprint.

Boa parte das organizações que possuem um processo de planejamento de release tradicional, é realizado no início da execução da release, e provavelmente não será mais alterado. Já fazendo o planejamento de release com o Scrum, provavelmente vai demandar um carga de esforço maior, pois o planejamento acontece nas reuniões de revisão de sprint e de planejamento de sprint, bem como nas reuniões diárias.

f) **Sprint:** Trata-se das iterações definidas previamente, as quais têm duração fixa, e trata-se do momento que são executadas as tarefas estabelecidas para essa iteração.

Segundo Rising e Janoff (2000, p.2, tradução nossa), “Uma sprint produz um produto visível e utilizável, que implementa uma ou mais interações do usuário com o sistema. A idéia central por trás de cada Sprint é a entrega de funcionalidades importantes.”

Durante a sprint a equipe e os objetivos a serem desenvolvidos não devem sofrer alteração, e quem garante isso é o Scrum Master, ele que deve garantir que não vão ocorrer mudanças imprevistas na sprint e assim não afetar o resultado da mesma.

Schwaber e Sutherland (2010) definem que: “as Sprints contêm e consistem na reunião de Planejamento de Sprint, o trabalho de desenvolvimento, a Revisão da Sprint e a Retrospectiva da Sprint. As Sprints correm uma após a outra, sem intervalos entre elas.”

As empresas em geral adotam um padrão recomendado pelo Scrum Guide(2010, p.11) de sprints de duas semanas principalmente para as que estão começando a utilizar o Scrum, o que com o tempo pode ser mudado, pois à medida que a equipe vai adquirindo experiência no Framework isto pode ser redefinido para o prazo que mais se encaixa com as tarefas a serem executadas e a equipe.

Uma equipe experiente, consegue perceber se é melhor sprints maiores ou menores que duas semanas definindo um tamanho bom que fique aproximado do que o Product Owner gostaria e o que o Time prefere.

De acordo com Kniberg (2007, p.25):

Bem, *sprints* curtos são bons. Eles permitem que a equipe seja “ágil”, isto é, mude de direção freqüentemente. *Sprints* curtos = ciclo curto de *feedback* = entregas mais freqüentes = *feedback* mais freqüente do cliente = menos tempo perdido, indo na direção errada = aprender e melhorar rápido, etc.

Entretanto, *sprints* longos são bons também. A equipe tem mais tempo para ganhar ritmo, ela tem mais espaço para se recuperar dos problemas, e conseguir atingir o objetivo do sprint, você tem menos *overhead* em termos de reuniões de planejamento, apresentações, etc.

As metas e tarefas a serem realizadas geralmente não sofrem alterações durante a sprint, mas se a equipe perceber que não vai conseguir realizar todos os itens estipulados para a iteração, ou ainda, perceber que vai terminar tudo bem antes do que o previsto, pode-se negociar junto ao Product Owner para aumentar ou reduzir o escopo do Product Backlog dessa sprint.

Mesmo com todos os cuidados e esforços de toda equipe pode acontecer de Sprints serem canceladas, e quem pode fazer isso é o Product Owner. Uma sprint pode vir a ser cancelada se os produtos a serem desenvolvidos não tiverem mais importância para o projeto, mudanças de mercado e tecnologia podem influenciar fortemente para isso acontecer.

Caso venha ocorrer um cancelamento de sprint a atitude a ser tomada é verificar os itens que ficaram prontos até então, e se podem ser acrescentados como algo utilizável.

Cancelamentos de sprints são situações incomuns e adversas, pois as mesmas normalmente acontecem em prazos curtos, propiciando a finalização dos itens propostos.

g) **Reunião de Planejamento da Sprint:** Como o próprio nome define é o momento para planejar a Sprint, tendo seu tempo fixado de acordo com o tamanho da iteração, onde, uma iteração de duas semanas leva em torno de 4 horas para realizar uma Reunião de Planejamento da Sprint.

Segundo Kniberg (2007, p.25) “O propósito da reunião de planejamento do sprint é dar à equipe informação suficiente para trabalharem em paz por algumas semanas, e para dar ao product owner confiança suficiente para deixá-los fazerem isto.”

A reunião é dividida entre a definição dos itens que serão feitos na iteração e a busca da equipe para entender como será realizado essas novas funcionalidades.

Inicialmente junto com o Product Owner a equipe define quais itens vão formar a meta para esta iteração baseado na capacidade e velocidade do time.

Conforme Schwaber e Sutherland( 2010, p. 13) :

A Meta da Sprint é o objetivo que será atingido através da implementação do Product Backlog. Ela é uma descrição que fornece orientação ao Time sobre a razão pela qual ele está desenvolvendo o incremento. A Meta da Sprint é um subconjunto da Meta da Release.”

A equipe junto ao Product Owner decide quais histórias vão ser incluídas na Sprint em questão. Kniberg (2007, p.35) cita duas técnicas que são:

**Sentimento/Instinto:** A técnica de sentimento/instinto consiste em o ScrumMaster questionar a equipe sobre o que ela acha que é capaz de fazer durante aquela sprint, apresentando os itens do Product Backlog e tentando extrair deles o que é possível combinar dos itens para essa iteração e o que não seria possível.

**Cálculo Da Velocidade:** Já a técnica de cálculo da velocidade, em uma equipe com alguma experiência anterior tenta-se buscar por base a velocidade que a mesma apresentou anteriormente e relacionar com as histórias previamente estimadas. Nas equipes sem experiência trata-se apenas do que a equipe estimou e o quanto ela aponta que vai precisar para realizar a tarefa. Como o próprio Kniberg (2007) comenta, trata-se de conseguir uma estimativa crua, mas de qualquer maneira é útil por não ter nada a ser comparado. Na próxima Sprint já deve começar a estabelecer alguns dados para servir como base.

Kniberg (2007) salienta ainda um ponto extremamente importante a se considerar, pois as tarefas inacabadas ou “quase prontas” não tem valor algum na velocidade, pois no Scrum é considerado apenas tarefas realizadas, prontas.

Após as metas serem criadas pelo time, começa a ser pensado em como realizar os itens selecionados do Product Backlog. Schwaber e Sutherland(2010, p. 13) comentam que nesta segunda parte da reunião “o Time define como transformará o Product Backlog selecionado durante a Reunião em um incremento pronto.”

Aqui o Product Owner deve fazer-se mais presente para ajudar a equipe a verificar se está realmente de acordo com os itens propostos para a sprint, caso perceba-se que tem itens demais ou de menos o Product Owner pode fazer essas trocas.

Schwaber e Sutherland(2010, p. 13) salientam que:

Geralmente, somente 60-70% do total do Sprint Backlog será definido na Reunião de Planejamento da Sprint. O restante é deixado para ser detalhado mais tarde ou são dadas estimativas grandes que serão decompostas mais tarde na Sprint.

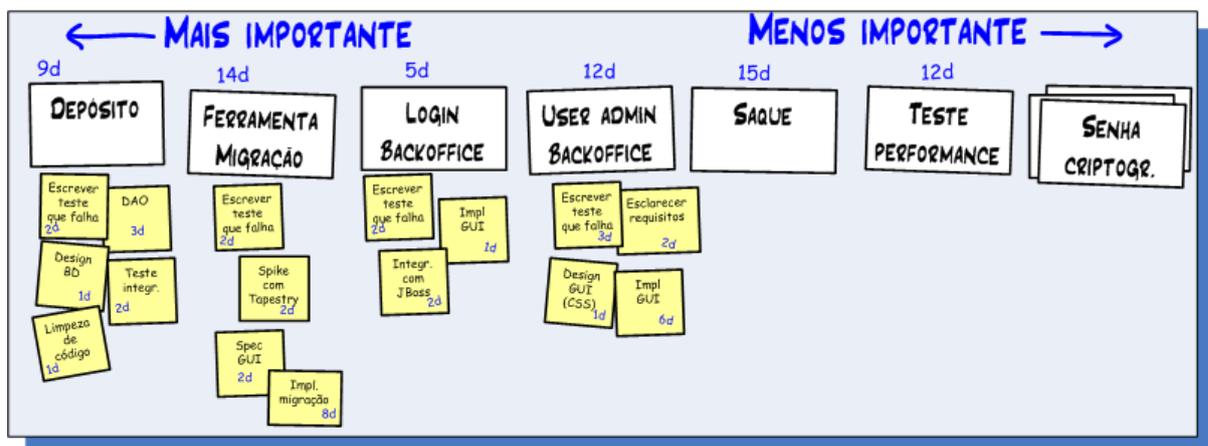


Figura 1: Exemplo de Planejamento dos itens do Product Backlog

Fonte: (Kniberg, 2007, p.43)

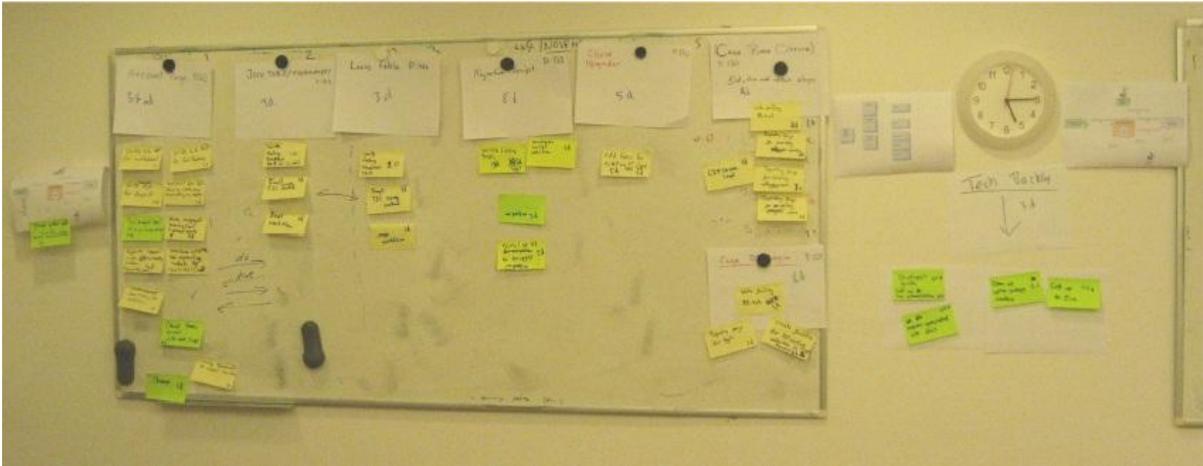


Figura 2: Exemplo de Planejamento dos itens do Product Backlog

Fonte: (Kniberg, 2007, p.43)

h) **Revisão da Sprint:** A revisão da Sprint trata-se de uma reunião que ocorre ao final da sprint, para que todos da equipe possam conversar sobre o que foi ou não foi feito nessa iteração.

Nesta reunião é mostrado às funcionalidades implementadas até então, podendo todos colaborar sobre o que poderá ser feito na próxima sprint. O Product Owner é quem basicamente coordena a reunião ponderando sobre os itens da iteração realizados, os problemas encontrados e discutindo sobre o Product Backlog que se apresenta ao final da iteração.

Segundo Schwaber e Sutherland(2010, p. 14), as revisões tem uma duração fixa de 4 horas para sprints de um mês, sendo proporcionalmente diminuída para sprints menores, onde uma sprint de duas semanas vai ter a metade desse tempo, no caso, apenas duas horas.

i) **Retrospectiva da Sprint:** A retrospectiva ocorre após a revisão de sprint, com uma duração de 3 horas para sprints de um mês, sendo proporcionalmente diminuída para sprints menores.

Busca verificar como transcorreu a iteração em relação aos membros da equipe, das relações entre elas, dos processos e ferramentas. Deve ser identificado tudo que ocorreu bem, e o que pode ser feito como melhoria nos pontos identificados com falhas, ou não satisfatórios.

j) **Daily Scrum:** Ocorrem todos os dias, deve ter uma duração de 10 a 15 minutos, não mais que isso, onde cada membro da equipe vai explanar para os outros membros:

- 1- O que realizou desde a última reunião.
- 2- O que vai fazer antes da próxima reunião.
- 3- E se existe algum obstáculo.

O ScrumMaster deve garantir que a reunião ocorra todos os dias, e que os membros da equipe sejam breves em suas explicações.

### 2.3.2 Artefatos do Scrum

Os artefatos do Scrum trata-se do Product Backlog, o Burndown da Release, o Sprint Backlog e o Burndown da Sprint.

a) **Product Backlog:** É a lista de itens a qual estão estipulados os itens e requisitos a serem desenvolvidos pelo time. Essa lista inicia com determinados requisitos, mas nunca fica por isso, sendo sempre acrescida no decorrer do desenvolvimento do projeto.

Segundo Kniberg (2007, p.17), “O *product backlog* é o coração do *Scrum*. É aqui que tudo começa. O *product backlog* é basicamente uma lista de requisitos, histórias, coisas que o cliente deseja, descritas utilizando a terminologia do cliente.”

Ainda seguindo Schwaber e Sutherland(2010) pode ser dizer que o Product Backlog está sempre em estado de mudança buscando manter o produto atualizado, sendo assim o mesmo nunca deixará de existir enquanto o produto precisar ser melhorado e aprimorado. Dentro desta lista estão às funções, características, tecnologias, melhorias e correções de defeitos a serem feitas no software nas próximas releases.

Os itens constantes do Product Backlog são classificados para determinar o que deve ser priorizado na hora da execução, tendo como atributos sua descrição, a prioridade e estimativa. Por isso os itens que estão com maior prioridade ficam no topo da lista, e para ele estar nessa posição foi por que se pensou, discutiu e chegou-se a um consenso de que determinado Product Backlog tem seu valor alto o suficiente para estar no topo ou no fim da lista.

Segundo Schwaber e Sutherland(2010, p. 18) o Product Backlog de maior prioridade é aquele que:

O Backlog de mais alta prioridade é mais claro e possui mais informações detalhadas do que o Backlog de prioridade mais baixa. Fazem-se melhores estimativas quando baseadas em uma maior clareza e em mais detalhes. Quanto mais baixa a prioridade, menor é o nível detalhe, até que mal se consiga entender o item.

A lista por estar em constante mudança, em decorrência de o produto estar sendo utilizado e aparecer novos requisitos, ou mudanças de mercado e de negócio. Apenas os itens de maior importância geralmente são mantidos com grande detalhamento, isso para que assim evite-se estar sempre fazendo retrabalho.

Todos os itens a serem realizados pela equipe são divididos para que assim se encaixem dentro das sprints e possam ser desenvolvidos por uma ou mais equipes.

<b>PRODUCT BACKLOG (exemplo)</b>					
<b>ID</b>	<b>Nome</b>	<b>Imp</b>	<b>Est</b>	<b>Como demonstrar</b>	<b>Notas</b>
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.

Figura 3: Exemplo de Product Backlog

Fonte: (Kniberg, 2007, p.20)

b) **Burndown da Release:** Bem como Schwaber e Sutherland(2010, p.19) definem, trata-se da soma das estimativas dos esforços restantes do Product Backlog ao longo do tempo.

Sendo o esforço estimado pela unidade de medição escolhida e utilizada pela equipe, onde geralmente as unidades de tempo são as sprints.

O Product Owner deve manter atualizados e publicados os itens do Product Backlog, os quais foram estimados, e se necessário, alterados pela equipe, mesmo que com alguma influência do Product Owner, e o Burndown do Backlog da Release.

c) **Sprint Backlog:** No Sprint Backlog é que são transformados os itens do Product Backlog em incremento para o produto que se classifique como pronto. Tarefas que geralmente são elaboradas na reunião de Planejamento da Sprint.

De acordo com Schwaber e Sutherland(2010, p. 20), “o Sprint Backlog é todo o trabalho que o time identifica como necessário para alcançar a meta da Sprint”.

Os itens da Sprint Backlog geralmente são decompostos, para que possam ser mais bem detalhados e entendidos nas Reuniões Diárias, tendo como um tamanho comum de em torno de um dia para um item da Sprint Backlog.

O Time e somente ele pode modificar o seu Sprint Backlog durante a iteração, de acordo que vai se percebendo se uma tarefa levará mais ou menos tempo do que o previsto, ou ainda se mais ou menos tarefas poderão ser realizadas.

d) **Burndown da Sprint Backlog:** O Burndown da Sprint Backlog é definido por Schwaber e Sutherland(2010, p.20 - 21) como:

Um gráfico da quantidade restante de trabalho do Sprint Backlog em uma determinada Sprint ao longo do tempo da Sprint. Para criar esse gráfico, determine quanto trabalho resta somando as estimativas do Backlog a cada dia da Sprint. A quantidade de trabalho restante para uma Sprint é a soma do trabalho restante para tudo do Sprint Backlog.

Acompanhando o gráfico é possível verificar o trabalho restante na medida em que o tempo vai passando, podendo ser passada uma linha nos pontos do gráfico, e assim demonstrando como o Time esta em completar o trabalho da Sprint.

Assim pode-se verificar se a equipe esta atrasada, em dia, ou adiantada com o que foi planejado para as Sprints.

### 2.3.3 Pronto

Basicamente em Scrum a cada Sprint, procura-se gerar um produto entregável e utilizável, pois caso o Product Owner decida implantar ou entregar a um cliente, este deve estar realmente pronto. Esse incremento deve ser adicionado aos outros anteriormente feitos e previamente muito bem testado, para garantir que tudo continue funcionando após ser adicionado o incremento.

Segundo Schwaber e Sutherland(2010), para ser definido como pronto, todos do Time devem saber o que realmente significa estar pronto, se é apenas o código desenvolvido, ou estar com o código desenvolvido, refatorado, testado exaustivamente, com uma build gerada e passado por testes de aceitação. Independente de qual das duas defina “Pronto” para o Time, todos devem estar com o mesmo pensamento.

Concordando Kniberg(2007, p. 43) define que:

É importante que o product owner e a equipe concordem com uma definição clara de “pronto”. Uma estória está completa quando todo o código está no repositório? Ou está completa apenas quando foi feito deploy em um ambiente de teste e a estória foi verificada por uma equipe de testes de integração?.

De acordo com Schwaber e Sutherland(2010, p.22) “ “Pronto” define o que o Time quer dizer quando se compromete a “aprontar” um item de Product Backlog em uma Sprint”.

Quando chega-se a um estado de “Pronto” em um incremento ele tem toda a análise, projeto, refatoramento, programação, documentação e testes, mas nem sempre o Time é capaz de definir o que é necessário para a implantação, ponto que o Product Owner deve resolver antes da implantação do produto.

### 3 METODOLOGIA DA PROPOSTA

Neste ponto é descrito a metodologia desenvolvida neste projeto.

#### 3.1 ASPECTOS METODOLÓGICOS

Trata-se de uma pesquisa de com procedimentos de cunho bibliográfico, por ter sido baseada em materiais como livros, artigos, e publicações periódicas.

A forma de abordagem da pesquisa é quali-quantitativa. Aos objetivos propostos, trata-se de uma pesquisa exploratória, pois segundo Alves (1991 apud Burigo, 1995, p. 20-21), “através da imersão dos pesquisadores no contexto, uma visão geral e não enviesado do problema considerado, e contribuir para a focalização das questões e identificação de informantes e outras fontes de dados”.

Já para Andrade (2005), a pesquisa exploratória é o passo para obtenção de informações, facilitando a delimitação do assunto, ajudando a definir os objetivos. Constitui-se em uma antecipação da pesquisa. Segundo Mattar (2007, p.7), “pesquisa exploratória visa prover o pesquisador de maior conhecimento sobre o tema ou problema de pesquisa em perspectiva.”

Com os resultados definir as dificuldades e desafios e facilidades encontradas com a adoção do Scrum que possam servir para as empresas de desenvolvimento de software.

Com o intuito de alcançar os objetivos da pesquisa, este trabalho será organizado nas seguintes seções:

- Fundamentação teórica a cerca dos modelos ágeis e o modelo Scrum.
- Reconhecimento do público alvo e exposição dos resultados da pesquisa.
- Análise dos resultados da entrevista.

### 3.2 METODOLOGIA PROPOSTA PARA EXECUÇÃO DO PROJETO

A partir dos estudos efetuados partiu-se para a elaboração de um questionário que permitisse identificar nas empresas de desenvolvimento de software se estão utilizando metodologias para seus projetos, principalmente se estão utilizando o Scrum.

Além disso, nas empresas que utilizam o Scrum identificar o que elas utilizam do Framework, quais as práticas que estão sendo mais adotadas, e qual foi o resultado disso a partir das experiências vivenciadas.

Para a elaboração do questionário seguiu-se os seguintes passos:

- Reconhecimento do público alvo.
- Elaboração do questionário de inspeção.
- Definição da amostra para a pesquisa.
- Aplicação do questionário junto aos profissionais da área de desenvolvimento de software.
- Análise dos resultados da entrevista.

## 4 DESENVOLVIMENTO

Neste capítulo é apresentado o resultado da aplicação da metodologia descrita no capítulo 3.

### 4.1 IDENTIFICAÇÃO DO PÚBLICO ALVO

Para realizar a investigação, foi desenvolvido um questionário e entregue em diferentes empresas de desenvolvimento de software, onde devia ser respondido por um membro da empresa que fosse parte da equipe de desenvolvimento e tivesse conhecimento de como os processos funcionavam.

O questionário conta com questões descritivas, de única e múltiplas opções.

Todas as questões foram respondidas de acordo com o que a empresa utilizava, apenas as questões que não tinham relação com o estilo de desenvolvimento das mesmas não foram respondidas, o que já era esperado pelo pesquisador.

Ouve uma grande dificuldade em conseguir que as empresas respondessem o questionário, demonstrando não querer revelar muito de como trabalham, muitas vezes questionando em que empresa o pesquisador trabalhava, não importando para qual instituição de ensino estava sendo desenvolvida a pesquisa.

A amostra conta com 35 empresas que desenvolvem softwares, localizadas na região da grande Florianópolis, sendo empresas de pequeno a grande porte. O questionário encontra-se no Apêndice A.

Um dos pontos investigados foi a cerca do cargo do entrevistado junto a empresa.

Assim o retorno dos questionários trouxe a seguinte lista de cargos:

- Programador/Desenvolvedor
- Analista de Sistemas

- Líder/Gerente de Projeto
- Analista de Desenvolvimento
- Gerente de Desenvolvimento(Gerente funcional / Projetos)
- Assistente de Gerência de projeto
- Analista de Banco de Dados
- Analista de Testes
- Analista / Gerente de Qualidade
- Coordenador de Pessoas e Gestão
- Coordenação de Desenvolvimento

Um segundo ponto questionado determinou o tamanho da empresa em número de funcionários. Pode ser observado que das empresas pesquisadas 37% possui até 50 funcionários e como menor número empresas com mais de 500 funcionários representando 17%.

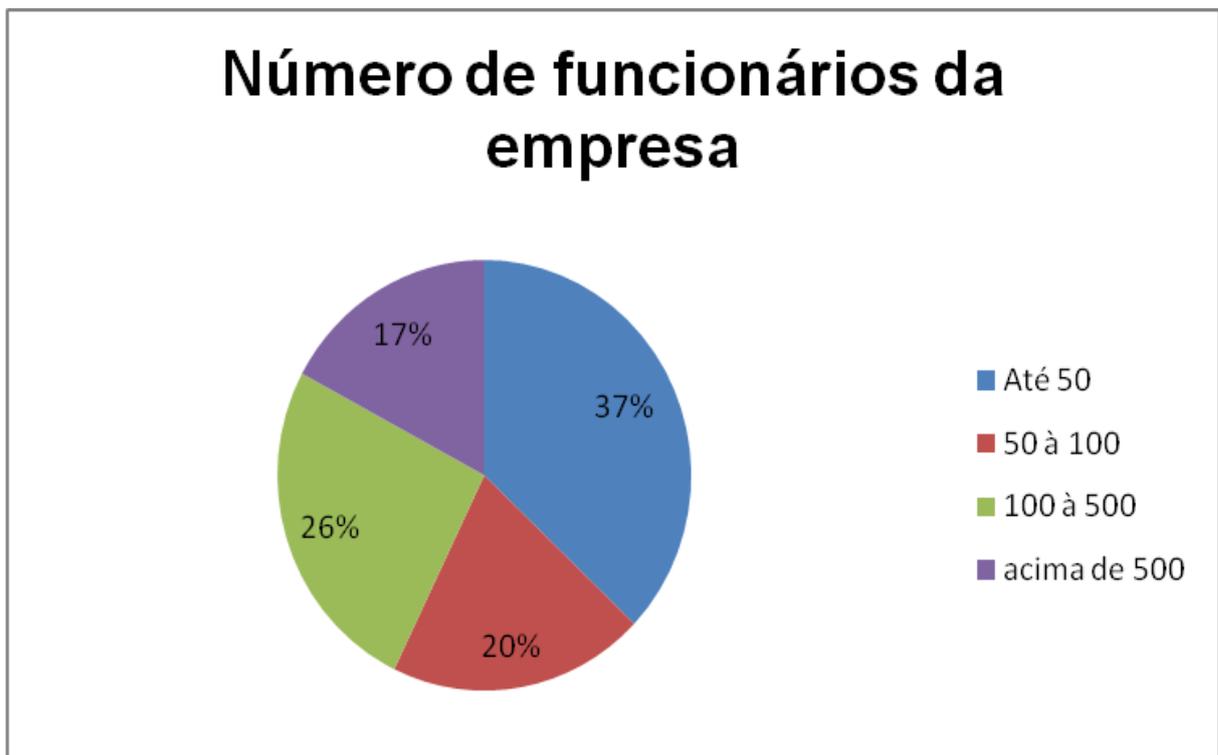


Gráfico 1: Número de funcionários da empresa

Fonte: Elaborado pelo autor

O gráfico 2 apresenta o tipo de produto que é desenvolvido nas empresas questionadas.

A opção “outro” foi a que mais se destacou na pesquisa com 14 respostas. Os outros tipos de softwares mencionados foram:

- Software para engenharia(CAE)
- Software de gerenciamento
- Soluções para internet
- Software para agronegócio e gerencia de requisitos
- Software para marketing digital
- Software para engenharia civil
- Agroindústria e apoio financeiro

Softwares Educacionais e E-Commerce foram os que tiveram o menor número de marcações.



Gráfico 2: Tipo de produto desenvolvido

Fonte: Elaborado pelo autor

O gráfico 3 apresenta o resultado das empresas que utilizam metodologias ágeis do desenvolvimento de software. Onde 66% responderam sim demonstrando ser maioria, como negação aos modelos ágeis tivemos 23% das empresas e outros 11%.

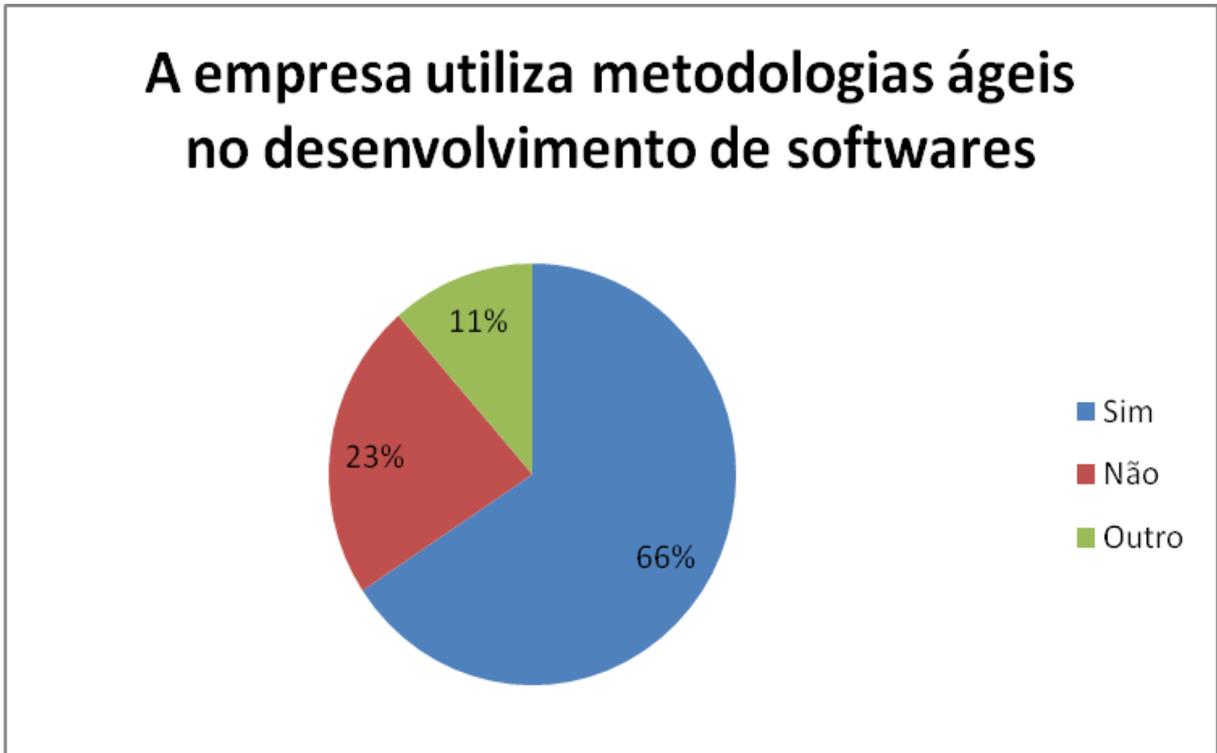


Gráfico 3: Utiliza metodologias ágeis no desenvolvimento de softwares

Fonte: Elaborado pelo autor

O gráfico 4 representa a temporalidade dos modelos usados, a grande maioria fazem uso da metodologia de 2 a 5 anos (14 empresas). Nenhum dos entrevistados usa o modelo a mais de 5 anos.

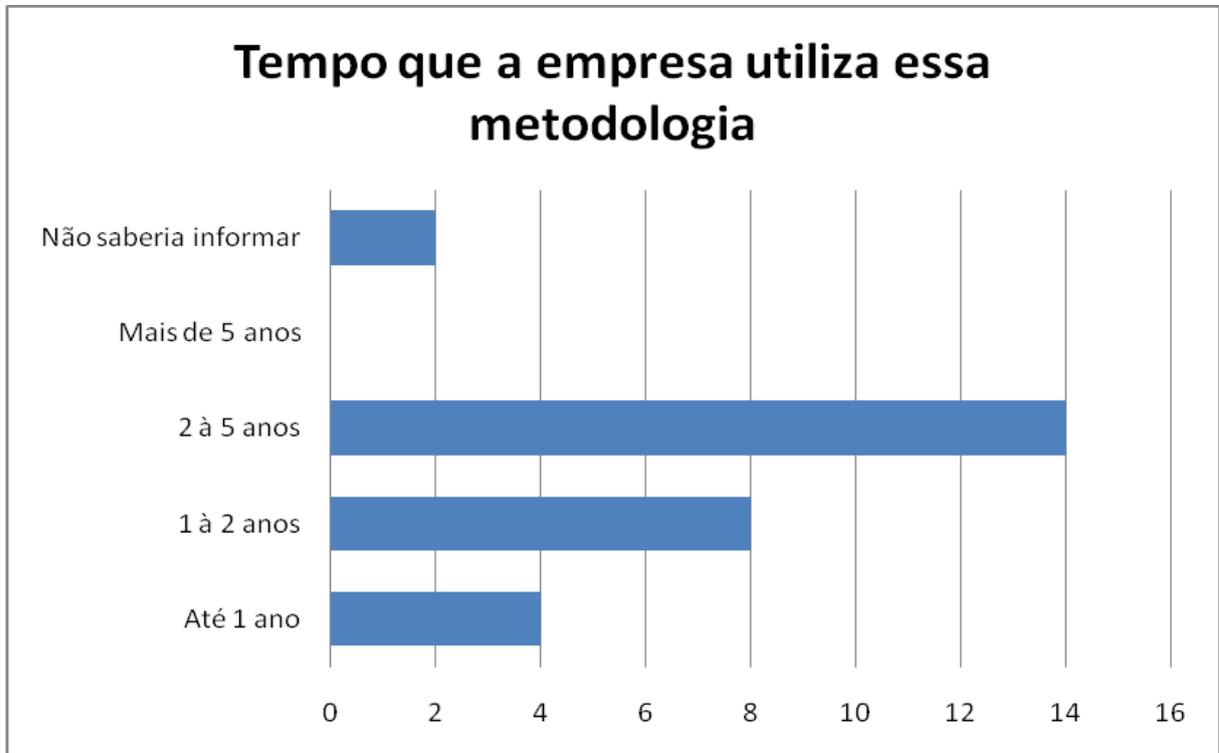


Gráfico 4: Tempo que a empresa utiliza essa metodologia

Fonte: Elaborado pelo autor

No gráfico 5 a partir dos entrevistados que fazem uso de alguma metodologia, temos 64% utilizando o Scrum, e 36% utilizam outra metodologia. Em respostas marcadas como outro foram citadas as metodologias:

- Metodologia própria
- Metodologia tradicional

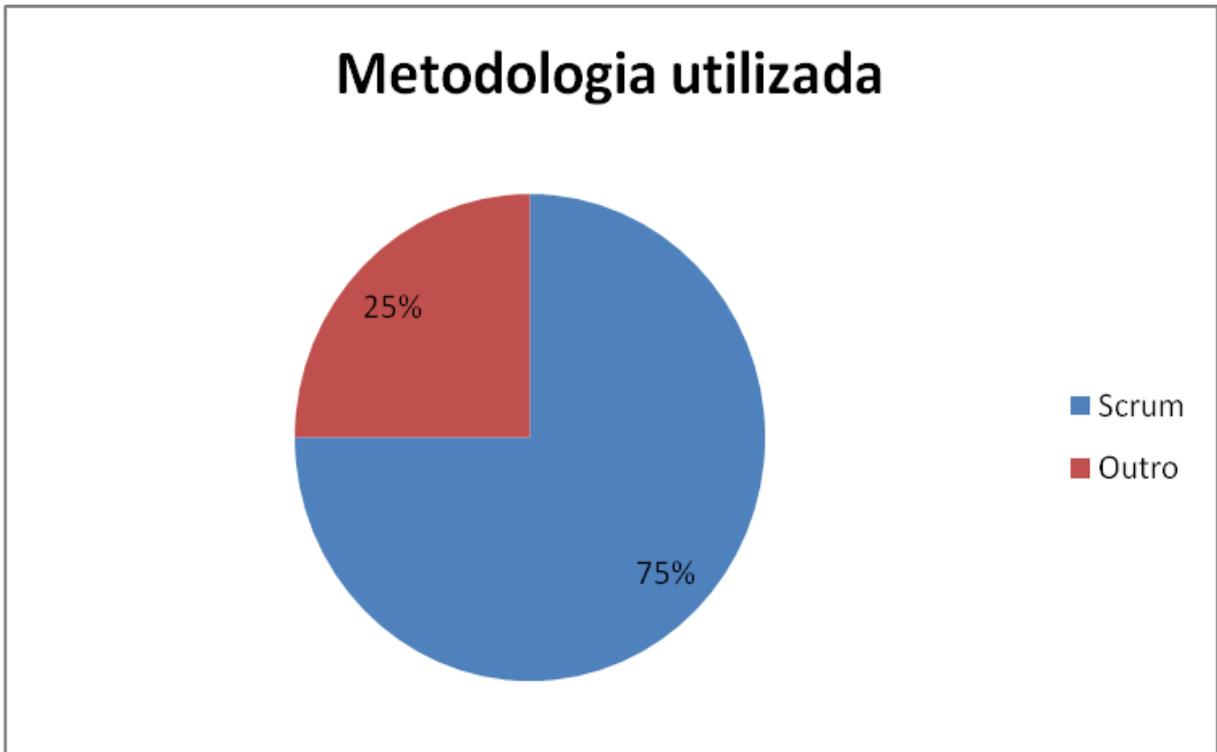


Gráfico 5: Metodologia utilizada

Fonte: Elaborado pelo autor

Questionou-se então para empresas se existia algum movimento ou interesse em mudar a metodologia utilizada e para qual.

Quase por unanimidade as empresas não desejam mudar o uso da metodologia, alguns descrevem que estão se adaptando para conseguir adquirir o Mps.Br procurando a melhoria de seus processos de desenvolvimento de software.

Foi então questionada a motivação pelo uso do modelo Scrum. Apesar de alguns tendo respondido que não tinham conhecimento do motivo da adoção por ter entrado na empresa após a implantação, obteve essas respostas:

- A necessidade de controlar o desenvolvimento dos projetos.

- Melhora na qualidade do processo de desenvolvimento de software.
- Por estar muito próximo da realidade da empresa e não ter uma metodologia anterior.
- A crença de que a adoção de métodos ágeis traria aumento da performance dos projetos.
- A aplicação do processo de desenvolvimento iterativo e incremental, a melhora das atividades de gestão, a satisfação do cliente com as entregas intermediárias periódicas.
- No estudo de diversas alternativas, o Scrum mostrou-se a forma mais adequada de representar nosso processo de desenvolvimento.
- Teve como grande fator motivacional acreditar que teria aumento da qualidade do produto através de metodologias que facilitem o controle dos projetos.
- Divisão do projeto em ciclos iterativos.
- Muitos atrasos nas entregas dos softwares.
- O grande volume de publicações e cases disponíveis. Um outro fator que deve ser informado, é que antes não havia uma metodologia definida, cada equipe fazia conforme sua vontade.
- O cenário anterior ao Scrum era muito caótico, sem qualquer tipo de metodologia.
- O sucesso nas empresas concorrentes.
- Otimização de tempo com as reuniões diárias rápidas de 15 minutos.
- Pelo grande desempenho que teve em outras equipes da empresa.
- A mudança de escopo freqüente.

No gráfico 6 é demonstrado se as empresas adeptas do Scrum, o utilizam de forma parcimoniosa com algum outro modelo. Com maioria de 68% respondida negativamente, e 38% responderam positivamente.

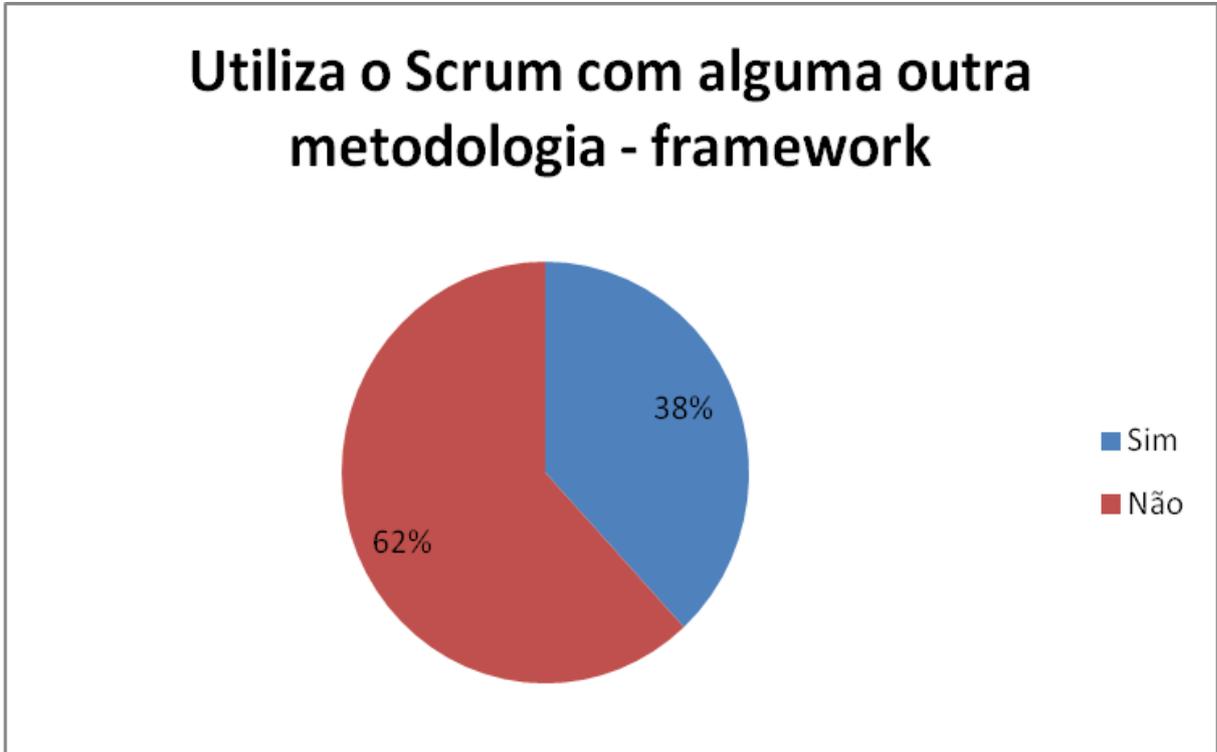


Gráfico 6: Utiliza o Scrum com alguma outra metodologia – framework

Fonte: Elaborado pelo autor

Do resultado obtido e representado no gráfico 6, questionou-se quais as metodologias que os questionados que responderam positivamente, utilizam com Scrum. Obteve-se como resultado 3 empresas usando o Extreme Programming, e 5 empresas que definiram a opção “outro” por não ter sido citada pelo autor da pesquisa como:

- Adaptações com os processos do departamento.
- Coletânea de diversas práticas.
- Processo SERPRO de desenvolvimento de soluções.
- Auxílio da ferramenta IceScrum.
- Scrum associado ao RUP.

A representação do resultado está no gráfico 7.

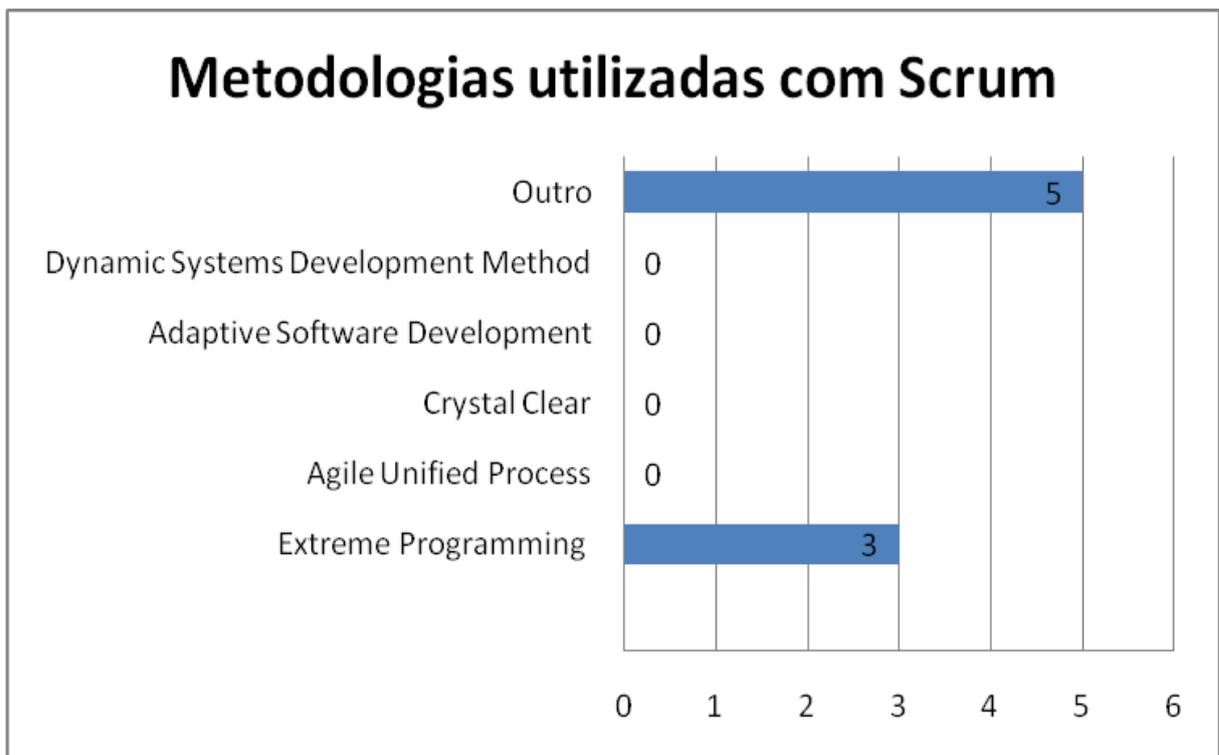


Gráfico 7: Metodologias utilizadas com Scrum

Fonte: Elaborado pelo autor

O gráfico 8 mostra quais práticas Scrum são utilizadas nas empresas que adotam esse framework. As reuniões diárias foram as que tiveram maior número de marcações, sendo 19, e como menor índice as regras Scrum com 3 marcações.

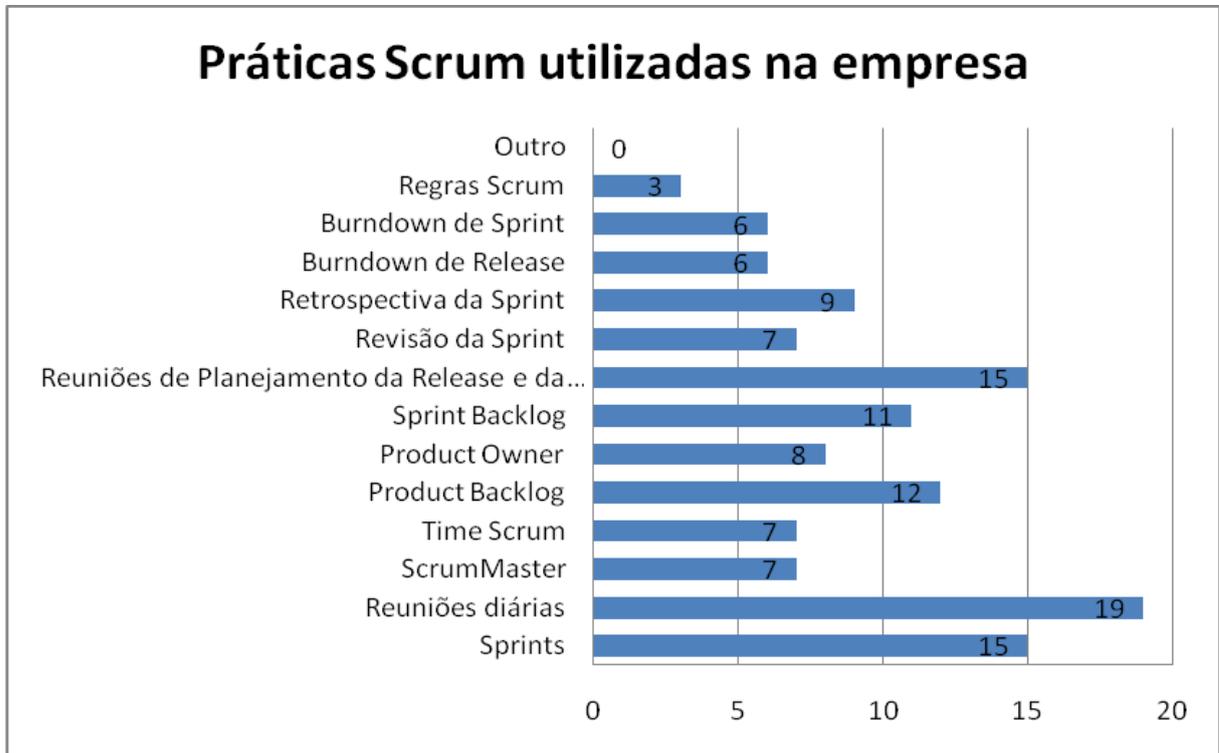


Gráfico 8: Práticas Scrum utilizadas na empresa

Fonte: Elaborado pelo autor

No gráfico 9 apresentam-se as práticas de testes utilizadas. Os testes automatizados aparecem com o maior número de incidências, com 14 marcações, em seguida os testes de aceitação (11) e, os testes unitários (5).

Ainda na possibilidade de variação outros tem-se:

- Testes exploratórios.
- Testes fumaça.
- Testes de integração.
- Testes de homologação.

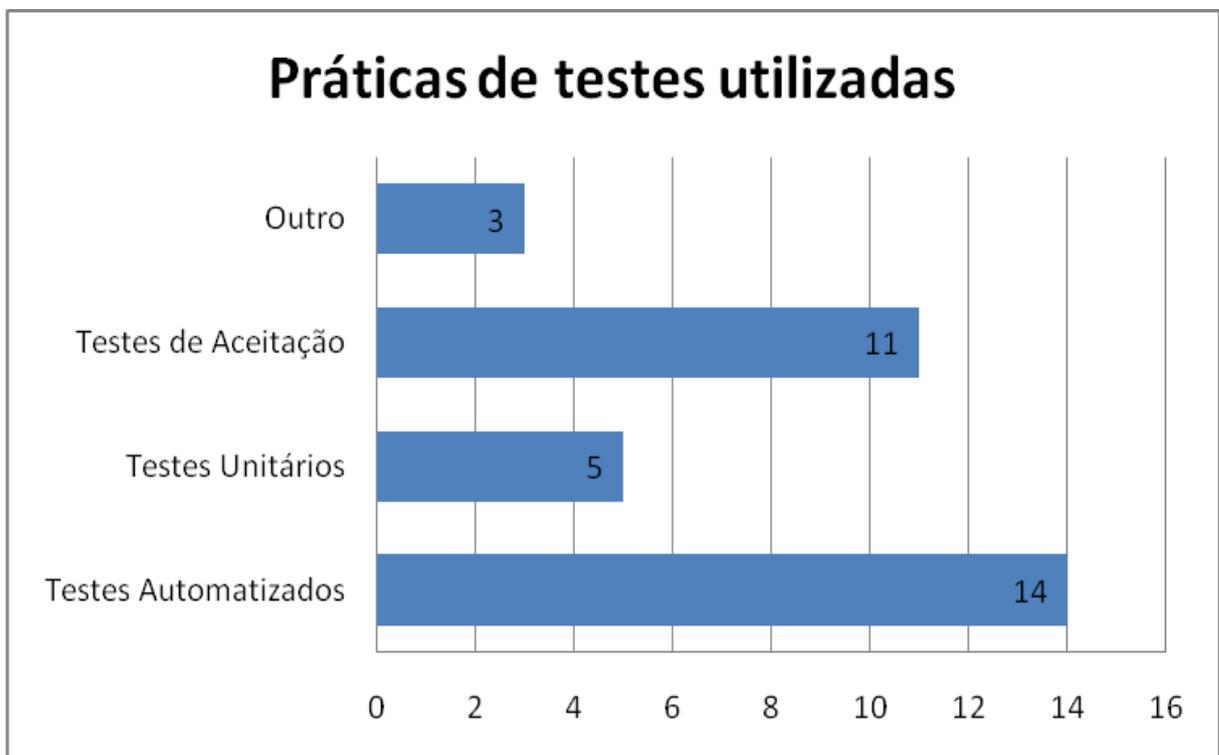


Gráfico 9: Práticas de testes utilizadas

Fonte: Elaborado pelo autor

No gráfico 10 são representados os resultados da questão que tinha como objetivo identificar se, os projetos utilizando o Scrum tiveram uma melhora que possa ser classificada como significativa na qualidade do produto. A resposta “sim” teve 90% dos resultados totalizando 18, e a resposta “não” com 10% representado por 2 dos 21 entrevistados que utilizam Scrum.



Gráfico 10: Melhora na qualidade do produto

Fonte: Elaborado pelo autor

Após a questão sobre a qualidade do produto, foi pedido aos entrevistados o relato **das principais dificuldades encontradas na adoção do Scrum:**

- Dar pontuação para as tarefas.
- Compreensão por parte da equipe.
- Não houve grandes dificuldades na etapa de adoção. A equipe se adaptou facilmente as reuniões diárias, de planejamento e de retrospectiva e ao desenvolvimento baseado em iterações.

- Principalmente estabelecer critérios de estimativas que pudessem ser aplicados a diferentes equipes e diferentes projetos, como o mesmo significado para a alta gerência.
- Período de transição e treinamento.
- Manter o comprometimento da equipe para continuar aplicando o Scrum.
- Na verdade o Scrum caiu tão bem no ambiente da empresa que só percebemos benefícios.
- Convencimento dos chefes.
- Conhecimento técnico da equipe.
- A equipe tem que atender diversas outras equipes que possuem um planejamento não alinhado, logo tem semanas que a equipe fica sobrecarregada e muitas das práticas do SCRUM são deixados de lado.
- Resistência a mudança de cultura.

Solicitou-se dos entrevistados o relato **dos principais benefícios encontrados na adoção do Scrum:**

- Maior controle da duração dos projetos.
- Maior controle, execução, histórico e planejamento.
- Diminuição do custo de gestão em favor do aumento da produtividade global da equipe.
- Melhora na gestão do projeto, na comunicação da equipe e entre a equipe e o cliente, aumento da satisfação do cliente e da produtividade da equipe.
- Foco nas entregas (mesmo parciais) e visibilidade do progresso.
- Agilidade em protótipos.
- Melhora no atendimento a problemas urgentes e integração da equipe na resolução dos problemas.
- Autogerenciamento.
- Visão "do que" está sendo desenvolvido.
- Maior controle no desenvolvimento.

- Melhora na comunicação e na visibilidade do andamento do projeto.
- Motivação da equipe e melhora na qualidade do produto.
- Redução de falhas do produto no cliente.
- Entregas programadas.
- Troca de conhecimento entre a equipe.
- Redução de interrupções durante a sprint.

O gráfico 11 apresenta a satisfação da equipe com os processos de desenvolvimento antes da utilização do Scrum, onde 62% classificaram como razoável, e 24% como Bom.

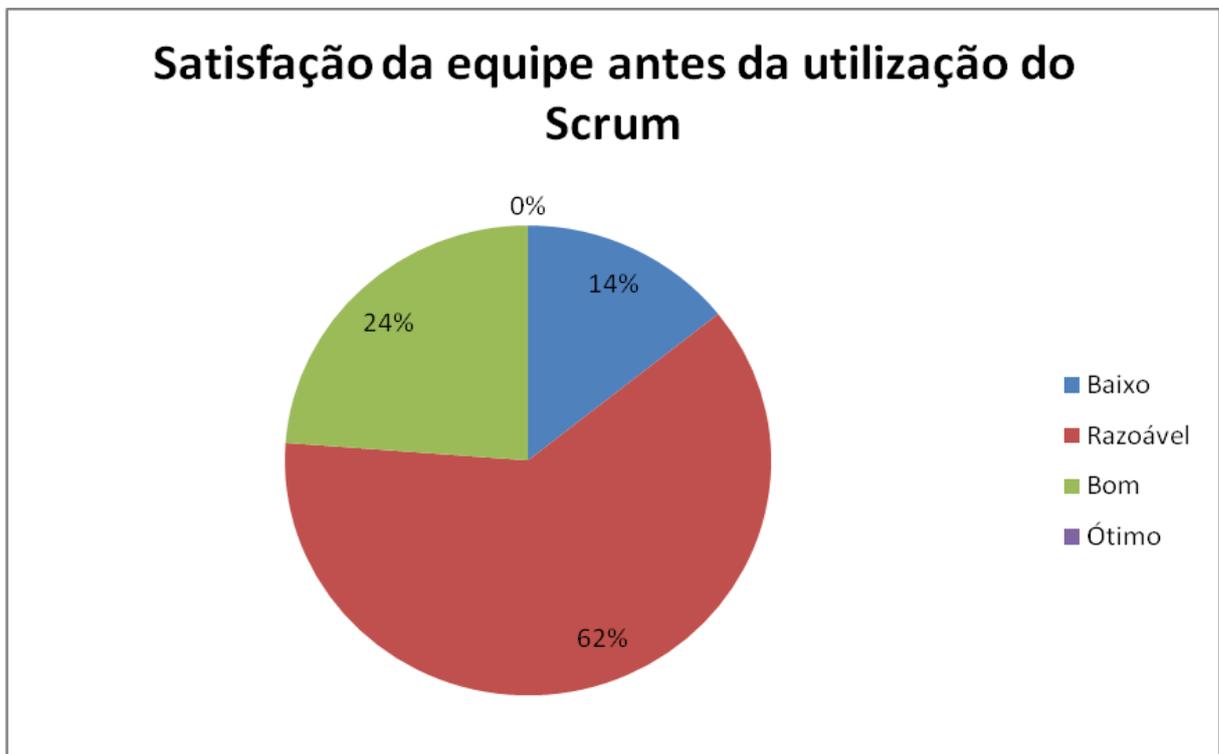


Gráfico 11: Satisfação da equipe antes da utilização do Scrum

Fonte: Elaborado pelo autor

Já no gráfico 12 é apresentado o grau de satisfação da equipe após a utilização do Scrum, onde temos 57% das respostas como Bom e 38% como Ótimo.

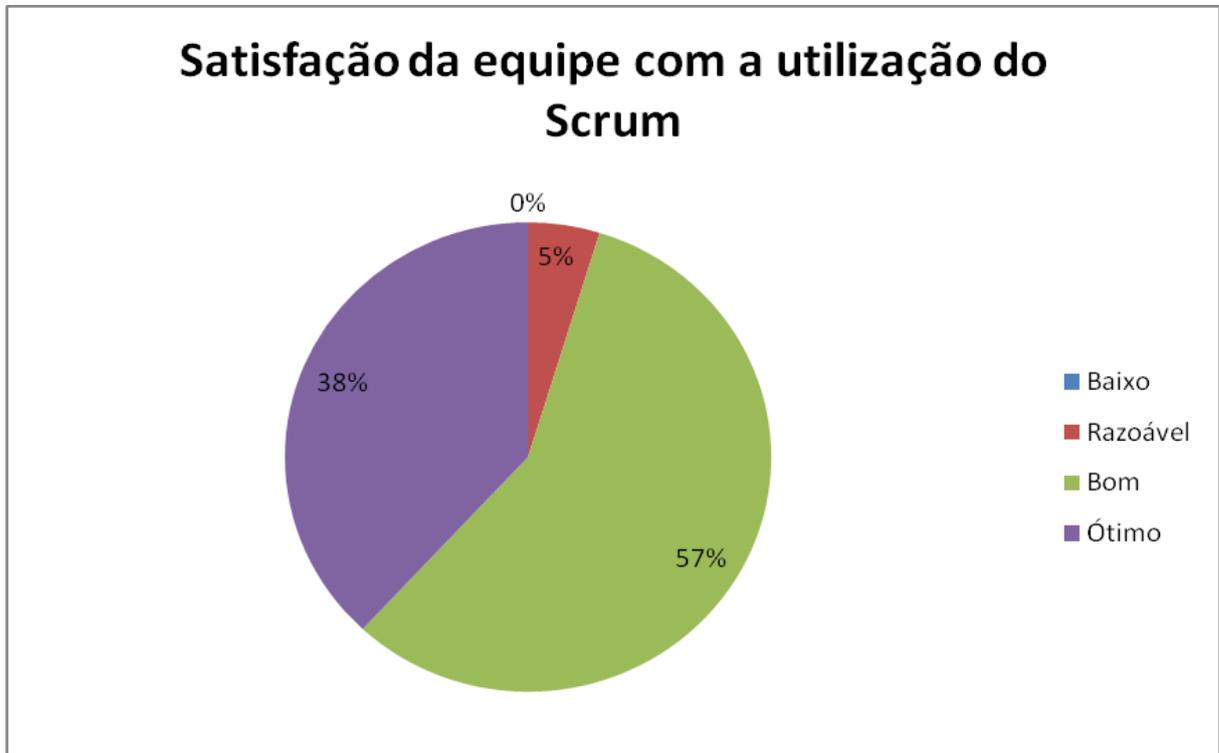


Gráfico 12: Satisfação da equipe com a utilização do Scrum

Fontes: Elaborado pelo autor

O gráfico 13 demonstra a satisfação da gerência antes da adoção do Scrum, sendo 67% em razoável demonstrando que a gerência até então nunca esteve plenamente satisfeita com os resultados.

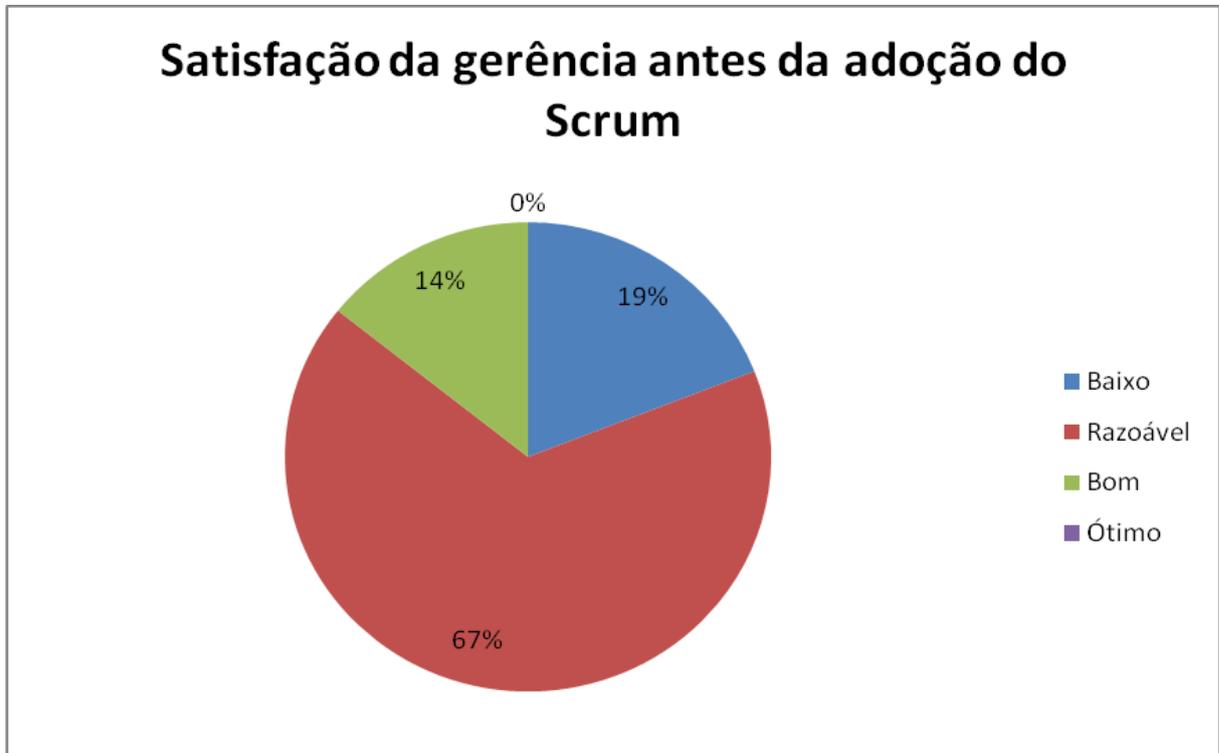


Gráfico 13: Satisfação da gerência antes da adoção do Scrum

Fonte: Elaborado pelo autor

Após a implantação a gerência em 57% dos casos responde como Bom, demonstrando grande satisfação da gerência com a mudança e implementação da metodologia Scrum.

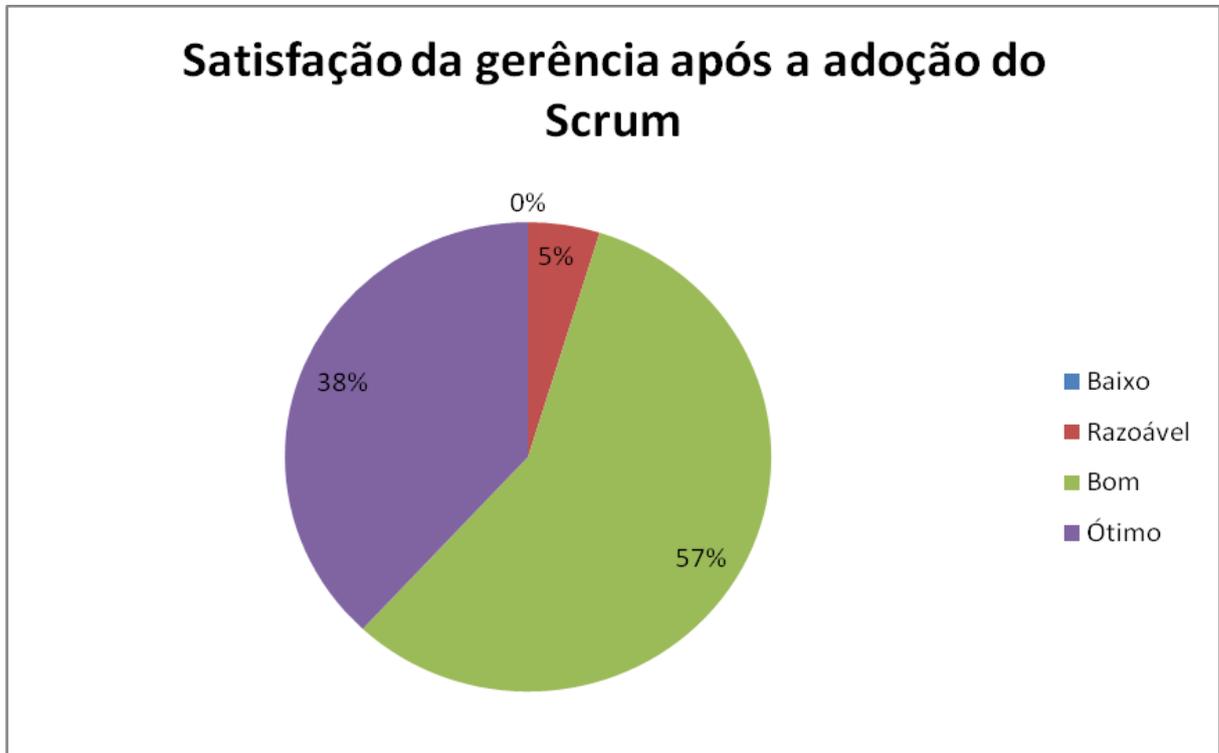


Gráfico 14: Satisfação da gerência após a adoção do Scrum

Fonte: Elaborado pelo autor

## 4.2 CONSIDERAÇÕES FINAIS SOBRE OS RESULTADOS DO QUESTIONÁRIO

Através dos dados obtidos, constata-se a melhora em vários pontos das empresas pesquisadas, como a satisfação da equipe bem como da gerência. O Scrum demonstra ter sido de grande importância e facilitado chegar a esses resultados, além do empenho dos envolvidos na utilização deste framework.

Em contra partida, em ser a metodologia com os melhores resultados, obteve-se informações de empresas que demonstram utilizar a metodologia sem ter conhecimento satisfatório. Alguns dos entrevistados que responderam não utilizar o Scrum, posteriormente no decorrer do questionário, demonstraram utilizá-lo, mesmo que não completamente.

Desta maneira, foram destacadas respostas de usuários da metodologia, por trabalharem com a mesma diariamente e apresentarem o desconhecimento de conceitos considerados básicos, como o “Time Scrum”, ao qual não foi selecionado por muitos na questão relacionada às práticas Scrum, sendo este demonstrado ao longo do projeto um fator de grande importância por se tratar da equipe, as pessoas que estão trabalhando no projeto.

Foi verificado através do questionário que as maiores dificuldades apresentadas foram com a adaptação para o Scrum, por ser necessário que os membros das equipes cumpram os processos, realizem as reuniões, e aceitem a mudança por uma metodologia nova. Outros pontos problemáticos encontrados foram com o convencimento da chefia, pela dificuldade para mensurar o esforço da implantação de uma metodologia financeiramente, e demonstrar quanto valor pode agregar essa mudança a longo prazo, onde inicialmente e durante a adaptação as equipes apresentam um rendimento menor do que se obteve até então.

Após a fase de implantação, os resultados dos esforços para a maioria dos pesquisados foi significativo. Com o resultado da questão sobre os principais benefícios encontrados após a adoção do Scrum, foi citado várias vezes o “maior controle”.

Maior controle da duração, execução, histórico e planejamento do projeto, demonstrando o quanto os entrevistados ficaram satisfeitos por saber o que está acontecendo em todo o projeto, independente de qual sua função na equipe ou na empresa.

A reunião diária foi à opção mais selecionada na questão que requisitava para os entrevistados marcar as práticas Scrum utilizadas em sua empresa.

Entrando em afirmação com o benefício de “maior controle”, a reunião diária apresenta todos os dias para os membros da equipe como está à situação e o que está sendo realizado por cada um, se existe alguma dificuldade ou qualquer outra informação que seja relevante na reunião.

Outro ponto mencionado foi às entregas programadas e a melhora na qualidade do produto que chega ao cliente, criando a satisfação na equipe, bem como da gerência que passa a acreditar fortemente na metodologia utilizada.

## 5 CONCLUSÃO

O trabalho atingiu seus objetivos propostos como o da exploração dos conhecimentos a cerca das metodologias ágeis, com o foco em Scrum, ponto principal da pesquisa. Através da metodologia Scrum pode-se verificar o quanto as empresas de softwares estão buscando algo para gerenciar seus projetos, acelerar seus processos e se adequar ao mercado, que busca ter em suas mãos o melhor produto, o mais rápido possível, onde o cliente não quer mais esperar para ter retorno em seu investimento apenas ao final do projeto, quer poder ver suas requisições se tornando em algo de valor.

Demonstrado pela vontade expressa nos 12 princípios ágeis que ao ter sido desenvolvida por gestores de grande conhecimento e respeito no mundo, iniciam os princípios com a prioridade de satisfazer o cliente, englobando trabalhar e fazer funcionar o software independente das mudanças requisitadas ao longo do projeto, fazer com que o cliente sinta-se parte do projeto, entregando valor a ele com partes funcionais do seu produto.

Através dos estudos realizados, pode-se detalhar todo o processo de desenvolvimento utilizado pela metodologia Scrum, demonstrando como ele funciona, em cada uma das suas etapas, reuniões e planejamentos.

Através da pesquisa pode-se ter uma visão mais clara de como as empresas de desenvolvimento de software trabalham e principalmente as que utilizam Scrum. Demonstrando que aderir a uma nova metodologia ágil como o Scrum não é uma tarefa fácil, por sempre existir o fato da resistência pela gerência, ou pela equipe com medo de algo novo, que venha a exigir novos conhecimentos, novo empenho, e que a primeira vista torne a equipe menos produtiva.

Apesar disso os pontos positivos da implantação da metodologia, aparecem com o decorrer do tempo e a experiência que a equipe venha a adquirir.

Os resultados encontrados pela pesquisa apresentam satisfação tanto das equipes, como da gerência, após o retorno do bom controle do projeto, e a valorização no produto bem desenvolvido.

## REFERÊNCIAS

ANDRADE, Maria Margarida de. **Introdução à metodologia do trabalho científico: elaboração de trabalhos na graduação**. 7. ed. São Paulo: Atlas, 2005.

BÚRIGO, Carla Cristina Dutra. **Qualidade de vida no trabalho: um estudo de caso na UFSC**. Projeto de Dissertação: novembro de 1995.

IMPROVE IT. **Scrum**. Disponível em: <<http://improveit.com.br/scrum>>. Acesso em: 11 jun. 2011.

KNIBERG, Henrik. **Scrum e XP direto das Trincheiras**. C4Media, 2007. Disponível em: <http://www.infoq.com/br/minibooks/scrum-xp-from-the-trenches>>. Acesso em: 9 jun. 2011.

MANIFESTO PARA DESENVOLVIMENTO ÁGIL DE SOFTWARE.

**Doze Princípios do Software Ágil**. Disponível em: <<http://agilemanifesto.org/iso/ptbr/manifesto.html>>. Acesso em: 11 jun. 2011.

MATTAR, Fauze Najib. **Pesquisa de marketing**. 4. ed. São Paulo: Atlas, 2007.

RISING, Linda; JANOFF, Norman S. **The Scrum Software Development Process for Small Teams**, *AG Communication Systems, 2000*.

SCHWABER, Ken; SUTHERLAND, Jeff. **Scrum Guide**. Scrum.Org, Fevereiro 2010. Disponível em: <<http://www.scrum.org/scrumguides/>>. Acesso em: 9 jun. 2011.

SHORE, James; WARDEN, Shane. **The Art of Agile Development**. 1ª ed., O'Reilly Media, 2007.



5- A quanto o tempo a empresa utiliza essa metodologia?

- ( ) Até 1 ano ( ) Mais de 5 anos  
( ) 1 a 2 anos ( ) Não saberia informar  
( ) 2 a 5 anos

6- Qual Metodologia é utilizada?

- ( ) Scrum  
( ) Outro (Descreva): \_\_\_\_\_

7 – Existe algum movimento ou interesse em mudar a metodologia utilizada? Para qual?

---

---

**Questões específicas sobre Scrum:**

8 - O que motivou a adoção ao Scrum?

---

---

9 - Utiliza o Scrum com alguma outra metodologia - framework?

- ( ) Sim ( ) Não

Caso sim, qual:

- ( ) Extreme Programming  
( ) Agile Unified Process  
( ) Crystal Clear  
( ) Adaptive Software Development  
( ) Dynamic Systems Development Method  
( ) Outro(Descreva): \_\_\_\_\_

---

10 - Quais práticas do Scrum são utilizadas na empresa?

- Sprints
  - Reuniões diárias
  - Scrum Master
  - Time Scrum
  - Product Backlog
  - Product Owner
  - Sprint Backlog
  - Reuniões de Planejamento da Release e da Sprint
  - Revisão da Sprint
  - Retrospectiva da Sprint
  - Burndown de Release
  - Burndown de Sprint
  - Regras Scrum
  - Outro(Descreva): \_\_\_\_\_
- 

11 - Quais práticas de testes são utilizadas?

- Testes Automatizados
  - Testes Unitários
  - Testes de Aceitação
  - Outro (Descreva): \_\_\_\_\_
- 

12 – Utilizando o Scrum os projetos tiveram melhora significativa na qualidade do produto?

- Sim
- Não

13 - Cite as principais dificuldades encontradas na adoção do Scrum:

---

---

---

---

---

14 - Cite os principais benefícios encontrados na adoção do Scrum:

---

---

---

---

---

15 - Classifique o grau de satisfação da equipe antes da utilização do Scrum:

Baixo                       Razoável                       Bom                       Ótimo

16 - Classifique o grau de satisfação da equipe com a utilização do Scrum:

Baixo                       Razoável                       Bom                       Ótimo

17 – Defina a satisfação da gerência antes da adoção do Scrum:

Baixo                       Razoável                       Bom                       Ótimo

18 – Defina a satisfação da gerência após a adoção do Scrum:

Baixo                       Razoável                       Bom                       Ótimo