

# **Uma breve visão sobre a metodologia scrum dos discentes de sistema de informação da faculdade projeção de Sobradinho/DF**

Douglas Martins Neves  
Leonardo Paiva Campos de Melo  
Rogério Oliveira da Silva

**Resumo:** Esse artigo aborda sobre o modelo de desenvolvimento ágil e flexível, denominado Scrum, que pode ser aplicado a qualquer produto ou no gerenciamento de qualquer atividade complexa no âmbito de engenharia de software, proporcionando um excelente entendimento entre as equipes de desenvolvimento. O propósito é apresentar o framework de processo ágil utilizado para gerenciar e controlar o desenvolvimento de um produto de software através de práticas interativas e incrementais.

**Palavras-chave:** Scrum. Desenvolvimento. Ágil.

**Abstract:** This article addresses the issue of the development model and is flexible, called Scrum, which can be applied to any product or non-management of any complex of software engineering activity, providing an excellent understanding as development teams. The purpose of presenting the process of elaborating a document for the management and control of the development of a software product through interactive and incremental practices.

**Keywords:** Scrum. Development. Agile.

## **1. INTRODUÇÃO**

O livro *The Enterprise and Scrum* de Ken Schwaber (2007) relata que projetos comuns gastam cerca de 50% do tempo com requisitos, arquitetura e especificação, e são realizados antes de construir qualquer funcionalidade. Todavia, os requisitos mudam por volta de 35% e 65% das funcionalidades levantadas por meio dos requisitos nunca ou raramente serão utilizadas.

O Scrum é de suma importância, pois discorre de metodologias ágeis para a gestão de projetos de desenvolvimento de software, a qual expressa dinamismo no ambiente de negócios já que a Tecnologia da Informação (TI) transpassa pela constante mudança e consequentemente por pressões por inovações e resultados melhores do produto final (Machado & Medina, 2017).

A metodologia ágil Scrum, como modelo para o gerenciamento dos projetos, permite que os stakeholders realizem o trabalho de forma integrada e comunicativa (Souza, 2016).

É fundamentada por seis características: “flexibilidade de resultados e prazos, times pequenos, revisões frequentes, colaboração e orientação a objetos”, que constituem como um guia para boas práticas com o intuito de alcançar o sucesso com práticas de gerenciamento (Schwaber & Sutherland, 2013).

Parte da origem de que o processo de desenvolvimento é imprevisível e estabelece que é um conjunto de atividades que convencionam conhecimento, ferramentas e

técnicas, juntamente com o melhor que a equipe pode conceder (Carvalho & Mello, 2009).

O Scrum é uma biblioteca voltada para a gerência, seu funcionamento é estruturado por ciclos chamados de Sprints e possui quatro fundamentos: (1) regras, (2) papéis ajudam identificar o que será desenvolvido e quais serão os responsáveis por quais atividades, (3) cerimônias são as reuniões e como resultado são gerados artefatos (4) (Schwaber & Sutherland, 2013). É um framework (biblioteca) de processo ágil que agrega mais valor e otimiza o tempo.

Dentro deste contexto, o propósito deste artigo é apresentar esse framework de processo ágil utilizado para gerenciar e controlar o desenvolvimento de um produto de software através de práticas interativas e incrementais.

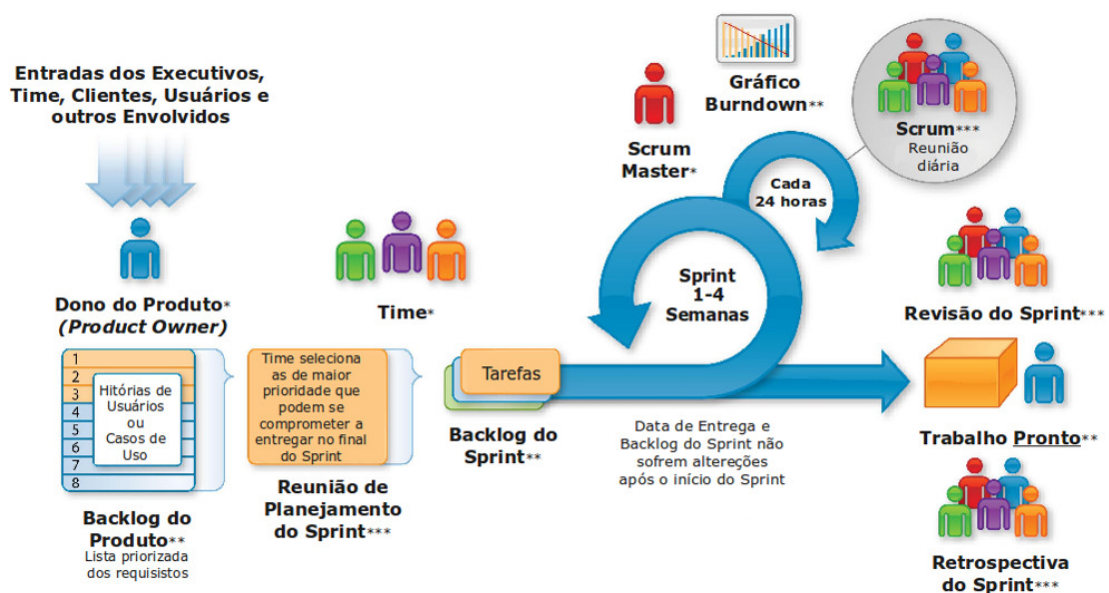
## 2. CARACTERÍSTICAS

Scrum é um framework utilizado para aprimorar a produção de projetos e usar suas características de desenvolvimento ágil para o desenvolvimento de software. Suas características são:

- Agilidade para gerenciar e controlar o desenvolvimento;
- Otimização no tempo de desenvolvimento;
- Desempenho equivalente entre as partes;
- Produtividade aumentada durante o desenvolvimento.

Com isso, o Scrum acaba sendo o mais utilizado no desenvolvimento devido aos benefícios oferecido pelo framework. Por trás desses benefícios oferecidos pelo o Scrum, existe um ciclo de vida que proporciona um melhor resultado para o desenvolvimento, conforme a imagem abaixo.

Figura 1 – Ciclo de vida do Scrum.



\* Papel, \*\* Artefato, \*\*\* Cerimônia

Fonte: <http://ned.unifenas.br/cursosgratuitos/201302/scrum/funcionamento.html>

### 2.1. PRODUCT OWNER

O **Product Owner** pode ser representado por um cliente, gerente, ou até mesmo por um membro da equipe de desenvolvimento, o mesmo é responsável por definir todas as funcionalidades e prioridades para adicionar ao Product Backlog.

O Scrum Team olha para o Product Backlog priorizado, seleciona os itens mais prioritários e se compromete a entregá-los ao final de um Sprint (iteração). Estes itens transformam-se no Sprint Backlog. Contudo, o Product Owner é responsável por:

- Priorizar as funcionalidades;
- Decidir data de liberação do conteúdo release;
- Aceitar ou rejeitar os resultados de trabalho;
- Ajustar funcionalidades e prioridades a cada 30 dias, conforme o necessário;
- Garantir que o time de desenvolvimento entenda o produto Backlog.

## **2.2. SCRUM MASTER**

Compete ao Scrum Master a responsabilidade de tornar o ambiente funcional, garantir que o time seja produtivo e que atenda ao processo estabelecido, bem como certificar que as práticas do Scrum estejam sendo adotadas e, o projeto e a cultura organizacional estejam em harmonia com as metas e o retorno sobre o investimento seja alcançado a cada Sprint (Fonseca, 2009). Scrum Master remove obstáculos e evita interferências, garante colaboração, guardião das práticas do Scrum, ou seja, mantém o foco na meta da Sprint.

“Responsável de conduzir o processo, tendo como função maior a busca pelas condições de trabalho, diretrizes, condução das reuniões, de revisão e retrospectiva. Também é fundamental na solução de problemas e obstáculos” (Fonseca, 2009).

O Scrum Master:

- Representa a gestão do projeto (ligação);
- É responsável por difundir os valores do Scrum e suas práticas;
- Executa as reuniões Daily Scrum;
- Remove os impedimentos;
- Protege o time contra interferências externas;
- Assegura que o time esteja totalmente funcional e produtivo;
- Possibilita uma cooperação estreita entre todos os papéis e funções” (Metodologia de Desenvolvimento de Sistemas, 2017).

## **2.3. ATRIBUIÇÕES DO SCRUM MASTER**

### **2.3.1. O SCRUM MASTER SERVE O PRODUCT OWNER DE VÁRIAS MANEIRAS, INCLUINDO:**

- Encontrando técnicas para o gerenciamento efetivo do Backlog do Produto;
- Comunicar a visão, objetivo e itens do Backlog do Produto;
- Ensinar a Time Scrum a criar itens de Backlog do Produto;
- Compreender a longo-prazo o planejamento do Produto;
- Compreender e praticar a agilidade;
- Facilitar os eventos Scrum conforme exigidos ou necessários (Schwaber & Sutherland, 2013).

### **2.3.2. O SCRUM MASTER SERVE O TIME DE DESENVOLVIMENTO DE VÁRIAS MANEIRAS, INCLUINDO:**

- Treinar o Scrum Team em autogerenciamento e interdisciplinaridade;
- Ensinar e liderar o Scrum Team na criação de produtos de alto valor;
- Remover impedimentos para o progresso do Scrum Team;
- Facilitar os eventos Scrum conforme exigidos ou necessários;
- Treinar o Time de Desenvolvimento em ambientes organizacionais nos quais o Scrum não é totalmente adotado e compreendido (Schwaber & Sutherland, 2013).

### **2.3.3. 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;
- Trabalhando com outros Scrum Masters para aumentar a eficácia da aplicação do Scrum nas organizações (Schwaber & Sutherland, 2013).

## **2.4. REUNIÕES**

### **2.4.1. SPRINT**

No Scrum os projetos são divididos em ciclos com duração de 1 mês. Esses ciclos são chamados de Sprint, onde a equipe planeja e define os prazos a ser entregue em reuniões mensais ou semanais, podendo a mesma ser revisada diariamente por meio do Daily Scrum para a priorização da responsabilidade do desenvolvimento da equipe (Metodologias Ágeis de Desenvolvimento de Software, 2017).

### **2.4.2. SPRINT PLANNING MEETING (REUNIÃO DE PLANEJAMENTO DA SPRINT)**

É uma reunião onde se encontra o Product Owner, Scrum Master e todo o Scrum Team. Nessa reunião o Product Owner define as funcionalidades de maior prioridade para equipe.

Na reunião o Product Owner não precisa descrever todo Product Backlog para a equipe, e sim as prioridades no decorrer da reunião, a equipe faz perguntas de modo que sejam capazes de quebrar as funcionalidades em tarefas técnicas, essas tarefas irão dar origem ao Sprint Backlog.

Após a reunião a equipe Scrum encontrasse separadamente para decidir o quanto a equipe pode se comprometer o que fazer na Sprint, sendo de responsabilidade da mesma definir o quanto ela é capaz de se comprometer (Sprint Planning Meeting, 2017).

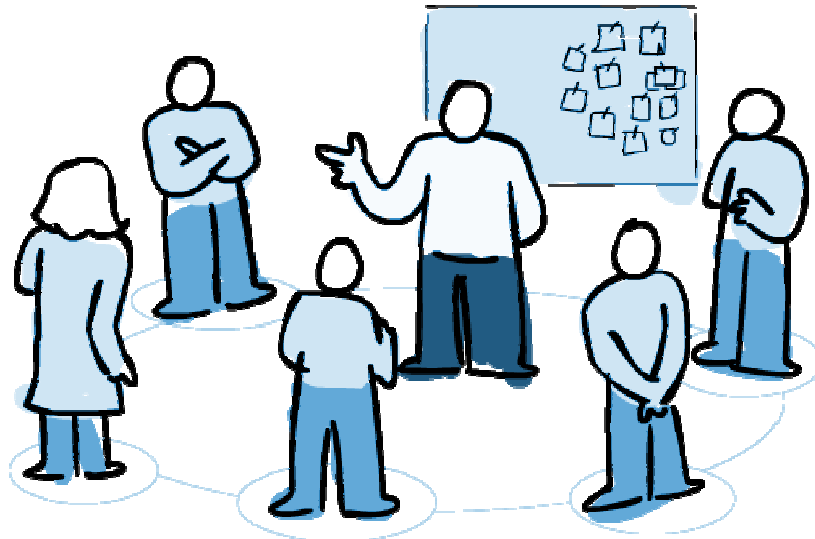
### **2.4.3. DAILY SCRUM**

A cada dia da Sprint a equipe faz reuniões diárias que chama Daily Scrum, normalmente no mesmo lugar e no mesmo horário geralmente feito na parte da manhã para estabelecer prioridades do dia, nesses encontros são feitas perguntas:

- O que você fez ontem?

- O que você vai fazer hoje?
- Existe algum obstáculo?

Figura 2 – Reuniões diárias.



Fonte: <http://www.edersonmelo.com/scrum-daily/>

Após as perguntas o Team Scrum obtém uma melhor visão do que foi concluído e o que ainda falta concluir. As reuniões possuem efeitos admiráveis, pois o Team Scrum compreende a importância dos seus compromissos.

#### **2.4.4. SPRINT REVIEW (REUNIÃO DE REVISÃO DA SPRINT)**

Todos da equipe avaliam todo o processo da Sprint. Ao final de cada Sprint é feito um Sprint Review. Nesta reunião, o Scrum Team mostra o que foi desenvolvido durante a Sprint. Os participantes da Sprint Review incluem o Product Owner, o Scrum Team, o Scrum Master, a gerência, os clientes e os engenheiros de outros projetos. Um trabalho incompleto não pode ser mostrado, com isso, a equipe mostra cada um dos itens do Product Backlog trazidos para fazer parte do Sprint.

#### **2.4.5. SPRINT RETROSPECTIVE (RETROSPECTIVA DA SPRINT)**

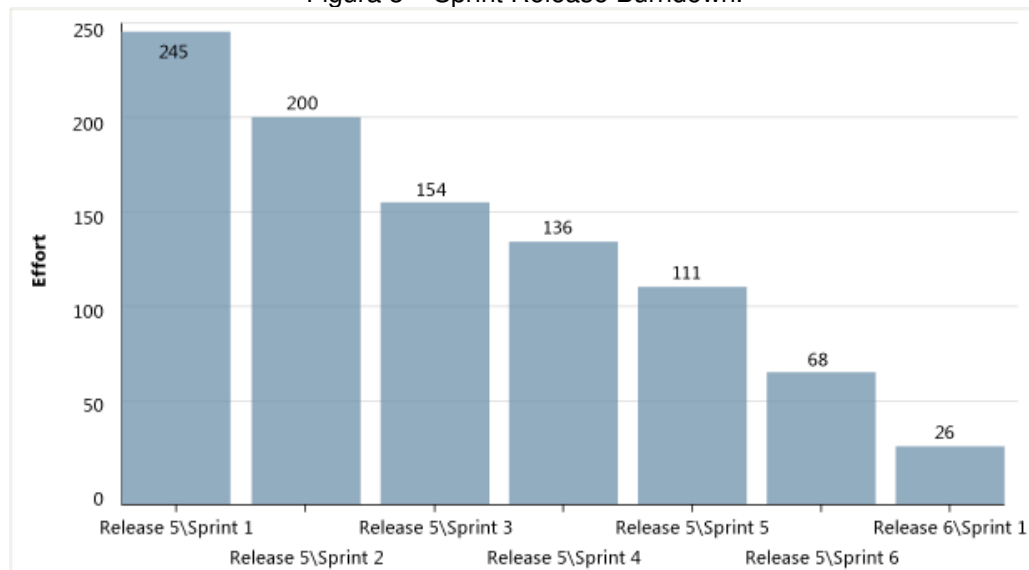
O Sprint Retrospective ocorre ao final da Sprint, onde todos os membros da equipe refletem sobre a Sprint passada e identificam o que funcionou bem e o que podem melhorar nos processos contínuos.

#### **2.4.6. RELEASE BURNDOWN CHART**

Uma equipe monitora o progresso através do Release Burndown ao final de cada sprint. O Scrum Master deve atualizar o gráfico release Burndown no final de cada Sprint e a equipe monitora seu progresso em relação a um plano, atualizando um Release Burndown Chart ao final de cada Sprint (iteração). O eixo horizontal do gráfico de Sprint Burndown mostra os Sprints. O eixo vertical mostra a quantidade

de trabalho restante no início de cada sprint. O trabalho restante pode ser mostrado em qualquer unidade (Release Burndown Chart, 2017).

Figura 3 – Sprint Release Burndown.



Fonte: <https://www.visualstudio.com/en-us/docs/report/sql-reports/release-burndown>

## 2.5. PRODUCT BACKLOG

O Product Backlog é o conteúdo definido no início do desenvolvimento pelo Product Owner, o mesmo não precisa ser totalmente definido no início do projeto, pode ser iniciado com o mais óbvio do projeto podendo ser incrementado com mais detalhes posteriormente durante o desenvolvimento (Product Backlog, 2017). Na imagem abaixo mostra um Product Backlog com as suas prioridades de 0 a 10.

Figura 4 – Lista Backlog.

Product Backlog		
User Stories	Business Priority	Story Points
Story A	1	5
Story B	2	8
Story C	3	1
Story D	4	8
Story E	5	2
Story F	6	2
Story G	7	2
Story H	8	8
Story I	9	5
Story J	10	1

Copyright © 2008 Russell Pannone – rpannone@WeBeAgile.com. All rights reserved.

Fonte: <https://pt.slideshare.net/rpannone/creating-a-product-backlog>

## 2.6. SCRUM TEAM

O Scrum Team constrói o produto do cliente através de uma equipe multifuncional onde é usada Sprint para a entrega do produto. A equipe é auto organizável, e possui as próprias responsabilidades.

Figura 6 – Função Scrum Team.



Fonte: [http://mds.cultura.gov.br/#Scrum/roles/scrum\\_team\\_8D8F2B3.html](http://mds.cultura.gov.br/#Scrum/roles/scrum_team_8D8F2B3.html)

### 2.6.1. TAREFAS QUE A EQUIPE DESEMPENHA:

- **DESENVOLVER A ARQUITETURA DA INFORMAÇÃO:** É feita uma estrutura para mostrar os caminhos que o usuário pode percorrer. O objetivo é manter a “visão do todo”.
- **DESENVOLVER BRIEFING CRIATIVO:** Contém uma descrição breve do projeto contendo detalhes e conceitos de criação para o Team seguir em frente.
- **DESENVOLVER WIREFRAME:** O Wireframe é a base para os desenvolvedores que pode ser entregue para o usuário. Podemos definir o Wireframe como a mais importante entrega para o usuário, onde é feita o contato entre o ambiente digital e o usuário, contudo o objetivo do Wireframe é auxiliar o desenvolvedor a entender o que o cliente está querendo. Possuindo um desenho básico de interface, o Wireframe, mostra todos os requisitos funcionais de um sistema.
- **ESTIMANDO O PRODUCT BACKLOG:** O Scrum Team estima o tamanho do Product Backlog, sendo fundamental para Product Owner a tomar decisões de priorização.
- **SPRINT BACKLOG:** O Sprint Backlog é uma lista de tarefas que são extraídos do Product Backlog, com as definições exigidas do Product Owner. As tarefas são realizadas pelo Scrum Team em um Sprint.

Figura 7– Lista de tarefas do Product Backlog.



## Sprint Backlog

Story	To Do		In Process	To Verify	Done
As a user, I... 8 points	Code the... 9	Test the... 8	Code the... DC 4	Test the... SC 6	Code the... SC 4
	Code the... 2	Code the... 8	Test the... SC 8		Test the... SC 4
	Test the... 8	Test the... 4			Test the... SC 6
					Test the... SC 6
As a user, I... 5 points	Code the... 8	Test the... 8	Code the... DC 8		Test the... SC 4
	Code the... 4	Code the... 6			Test the... SC 6



Fonte: <https://www.slideshare.net/NiceMsh/scrum-in-action-60983900>



### **2.6.2. INCREMENTO SCRUM**

A Sprint gera um resultado, que é o incremento de produto potencialmente utilizável (tipo de produto de trabalho). O responsável por essa função é o Scrum Team, que trabalha com as funcionalidades que serão desenvolvidas durante a Sprint. Esse incremento contém arquivo executável elaborado com base em código testado e documentação dos procedimentos operacionais das funcionalidades (Metodologia de Desenvolvimento de Sistemas, 2017).

O incremento é a soma de todos os itens completados do Backlog do Produto. No final de um Sprint, o novo incremento deve estar “pronto”, o que significa que ele está em uma condição utilizável e atende à definição da Equipe do Scrum (Schwaber & Sutherland, 2013).

Na finalização da Sprint o incremento novo deve estar pronto, ou seja, deve estar na condição utilizável e atender os requisitos do status “pronto” (definição estabelecida pela equipe), podendo ou não colocar em produção imediatamente. Por ventura alguma funcionalidade não esteja concluída (pronta) quando a Sprint atingir o seu prazo deverá retornar ao Backlog do Produto para que inclua em uma nova Sprint (Prikladnicki; Willi; Milani, 2014).

### **2.6.3. DEFINIÇÃO DE "PRONTO"**

O Time de Desenvolvimento entrega um incremento de funcionalidade do produto a cada Sprint. Este incremento é utilizável, assim o Product Owner pode escolher por liberá-lo imediatamente. Se a definição de “pronto” para um incremento é parte das convenções, padrões ou diretrizes de desenvolvimento da organização, todos os Times Scrum devem segui-la como um mínimo.

Se “pronto” para um incremento não é uma convenção de desenvolvimento da organização, o Time de Desenvolvimento do Time Scrum deve estabelecer uma definição de “pronto” apropriada para o produto. Se há múltiplos Times Scrum trabalhando no lançamento do sistema ou produto, os Times de desenvolvimento de todos os Times Scrum devem mutuamente determinar a definição de “Pronto” (Schwaber & Sutherland, 2013).

Quando um produto é descrito como "Pronto", todos devem entender o que significa. Somente é considerado se estiver codificado, passando nos testes unitários, estimado, trabalhado em sessões de refinamento do Product Backlog, passando nos testes de aceitação, com o manual do usuário atualizado.

## **3. CONSIDERAÇÕES FINAIS**

O estudo apresentado neste artigo demonstra como o scrum é uma ferramenta essencial, pois possibilita observar potenciais problemas no início, meio e fim da produção do produto, porém apenas indica que há um gargalo e não permite estabelecer uma solução para o problema apresentado. Já o Scrum Master, o Product Owner e membros do time têm o objetivo de encontrar uma solução a fim de obter resultados. Dessa forma, é provável/previsto que a implementação do método possa desenvolver resistência pelas pessoas envolvidas devido ao fato de propor mudanças (Scrum Essencial, 2017). A partir dessas mudanças a organização garante o crescimento no desenvolvimento de metodologias ágeis na gestão de projetos de desenvolvimento de software, oferecendo entrega rápida, com flexibilidade e de qualidade e consequentemente um diferencial entre os concorrentes e satisfação do cliente (Boehm, 2003).

## **REFERÊNCIAS**

MACHADO, Marcos; MEDINA, Sérgio Gustavo. **SCRUM – Método Ágil: uma mudança cultural na Gestão de Projetos de Desenvolvimento de Software**. Revista Científica Intr@ciência, UNIESP, 2017.

SOUZA, Francenila Rodrigues Júnior et al. **O Uso do Scrum Aplicado no Planejamento do Processo de Ensino-Aprendizagem**. III Congresso Internacional das Licenciaturas, 2016.

SCHWABER, Ken; SUTHERLAND, Jeff. **Guia do Scrum - Um guia definitivo para o Scrum: As regras do jogo**. Julho de 2013.

SCHWABER, K.; SUTHERLAND, J. **Guia do Scrum**. Disponível em: <<https://www.scrum.org/Portals/0/Documents/Scrum%20Guides/2013/Scrum-Guide-Portuguese-BR.pdf>>. Acesso em: 17.04.2017.

SCHWABER, Ken. **The Enterprise and Scrum**, Microsoft Press, 2007. TEAMSYSTEM, Scrum for team system, 2005. Disponível em <<http://www.scrumforteamsystem.com>> Acesso em: 20.04.2017.

CARVALHO, B.; MELLO, C. **Revisão, Análise e Classificação da Literatura sobre o método de desenvolvimento de produtos ágil SCRUM**. In: ANAIS SIMPOI, 2009.

FONSECA, Isabella. **Engenharia de Software Conference**. São Paulo: DevMedia, 2009.

BRQ - **Metodologias Ágeis De Desenvolvimento De Software**. Disponível em: <<http://www.brq.com/metodologias-ageis/>>. Acesso em: 17.04.2017.

DESENVOLVIMENTO AGIL - **Sprint Planning Meeting**. Disponível em: <[http://www.desenvolvimentoagil.com.br/scrum/sprint\\_planning\\_meeting](http://www.desenvolvimentoagil.com.br/scrum/sprint_planning_meeting)>. Acesso em: 17.04.2017.

MOUNTAIN GOAT SOFTWARE - **Release Burndown Chart**. Disponível em: <<https://www.mountangoatsoftware.com/agile/scrum/scrum-tools/release-burndown>>. Acesso em: 17.04.2017.

DESENVOLVIMENTO AGIL - **Product Backlog**. Disponível em: <[http://www.desenvolvimentoagil.com.br/scrum/product\\_backlog](http://www.desenvolvimentoagil.com.br/scrum/product_backlog)>. Acesso em: 18.04.2017.

PRIKLADNICKI, Rafael; WILLI, Renato; MILANI, Fabiano. **Métodos ágeis para desenvolvimento de software**. – Porto Alegre: Bookman, 2014.

BOEHM, B.; Turner, R. **Balancing Agility and Discipline A Guide for the Perplexed**. Addison Wesley, 2003.

