

MÉTRICAS PARA EQUIPES QUE UTILIZAM O SCRUM COMO METODOLOGIA ÁGIL NO GERENCIAMENTO DO PROJETO

Rafael P. Mugno¹
Cláudia Soave²

Resumo: O Scrum, como metodologia ágil, vem sendo utilizado em diversas equipes de desenvolvimento de sistemas. Saber utilizar a metodologia e buscar uma constante evolução do projeto e dos profissionais é fundamental para se obter sucesso na organização, tendo em vista, também, a busca em diminuir os custos e entregar o produto com qualidade final. Para apoiar nessa evolução e entregar um produto com excelência, é necessário possuir indicadores que auxiliem no processo de acompanhamento das pessoas e da equipe, o que permite melhores tomadas de decisões e monitoramento durante a gestão dos projetos. Visto isso, este trabalho tem como objetivo propor um conjunto de métricas para projetos de desenvolvimento de softwares que utilizam o Scrum como metodologia ágil, a partir de uma revisão da literatura. São apresentadas sete métricas fundamentais que podem ser aplicadas na gestão de times que utilizam o Scrum, demonstrando como a métrica deve ser aplicada e monitorada, para garantir uma evolução da equipe.

Palavras-chave: Métricas, Scrum, Metodologias ágeis, Desenvolvimento de sistemas.

1 INTRODUÇÃO

As metodologias ágeis, como o Scrum, estão bastante presentes para apoiar no gerenciamento de projetos de desenvolvimento de software, contudo, para que se tenha uma boa gestão da equipe e do projeto, é fundamental que se possua indicadores bem definidos para apoiar nesse papel. Segundo Downey e Sutherland (2013), a falta de atenção adequada às métricas pode impedir que as equipes melhorem sistematicamente e alcancem uma boa produtividade.

Nesse sentido, este trabalho tem como problema de pesquisa: *Como a utilização de métricas pode contribuir para o monitoramento de projetos e gestão de equipes, que utilizam o Scrum como metodologia ágil no contexto da inovação?* O objetivo geral do estudo é propor um conjunto de métricas para projetos de desenvolvimento de softwares que utilizam o Scrum como metodologia ágil. Neste contexto, pretende-se em termos de objetivos específicos, compreender as métricas adequadas para equipes de desenvolvimento de sistemas, elencar os principais indicadores e descrever como cada métrica pode ser aplicada.

Conceituando, métricas são medidas quantitativas que demonstram alguma informação, enquanto os indicadores informam sobre o desempenho ou o estado de uma área

¹ Discente do curso de Especialização em Gestão e Inovação do IFRS Câmpus Farroupilha. Tecnólogo em Análise e Desenvolvimento de Sistemas IFRS. E-mail: rafamugno@gmail.com

² Docente do curso de Especialização em Inovação e Gestão. Doutora em Educação. Mestra em Filosofia. Especialista em Estratégias de Marketing e em Desenvolvimento Humano e Arteterapia. Bacharela e Licenciada em Administração de Empresas.

de interesse, com base nas informações providas das métricas. As métricas podem ser usadas para calcular indicadores, ajudando na avaliação e no monitoramento de resultados.

Este artigo justifica-se, uma vez que a evolução de uma equipe é um fator muito importante para se ter sucesso no que se está desenvolvendo e para isso a gestão das pessoas precisa de ferramentas de apoio para conseguir guiar o time com metodologias corretas e indicadores para conseguir visualizar o progresso. Segundo Ramos et al. (2020), o gerenciamento de projetos tem grande importância para o sucesso e qualidade no que está sendo entregue e, para isso ser possível, o gerenciamento deve estar aliado a boas metodologias e indicadores.

O setor de desenvolvimento de sistemas tem grande importância para a transformação digital por trazer ferramentas de apoio que proporcionam uma evolução para as empresas. Área que está em constante crescimento, teve um aumento de quase 40% no Brasil entre 2022 e 2023, mesmo em um cenário de crise em grandes empresas mundiais de tecnologia (Batalha, 2023).

Outrossim, as equipes ágeis têm dificuldade em medir o desempenho. Pesquisas globais mostram que 50% das equipes não conhecem sua velocidade de produção e têm dificuldade em encontrar maneiras de melhorar e medir essa taxa. Mesmo quando as equipes conhecem sua velocidade, a gestão não consegue comparar o desempenho de duas equipes com as métricas atuais (Downey; Sutherland, 2013).

2 REFERENCIAL TEÓRICO

Nesta seção são apresentados os principais conceitos abordados neste trabalho, relacionados à gestão de equipes dentro do desenvolvimento de sistemas e gestão de projeto com a utilização de metodologias ágeis, especificando o Scrum.

2.1 GESTÃO ÁGIL DE PROJETOS

Em 2001, o Manifesto Ágil foi criado por um grupo de 17 representantes de desenvolvimento de software, buscando inovar, com alternativas mais leves e rápidas às metodologias tradicionais. O Manifesto destaca quatro valores-chave: priorizar indivíduos e iterações, software funcionando, colaboração com o cliente e resposta a mudanças. Os

métodos ágeis enfatizam flexibilidade e adaptabilidade, minimizando a documentação, em contraste com a metodologia tradicional, onde o planejamento é central (Almeida; Carneiro, 2023).

Dentre várias metodologias criadas com base no Manifesto Ágil, o Scrum é uma delas. Segundo Tekin, Yilmaz e Clarke (2023) o Scrum é amplamente utilizado nos times de desenvolvimentos de sistemas. Schwaber e Sutherland (2020) definem o Scrum como um framework que ajuda pessoas, times e organizações a gerarem valor por meio de soluções adaptativas para problemas complexos. Esse framework define alguns papéis que os membros da equipe devem ter, cerimônias que devem ser realizadas e como devem estar organizadas e distribuídas as tarefas.

Todas as atividades existentes no projeto que ainda não foram realizadas ficam em um espaço chamado backlog do produto. As equipes trabalham em iterações (período de tempo) chamadas Sprints, geralmente com duração de 2 a 4 semanas, produzindo incrementos do produto a cada novo ciclo. Antes de uma nova Sprint iniciar, é realizado um planejamento (momento conhecido como planning) para definir quais tarefas do backlog vão ser desenvolvidas na iteração, de acordo com a capacidade do time. Essa capacidade geralmente é medida em pontos de história (story points) que representam uma medida relativa de acordo com a complexidade e tempo para realizar cada tarefa. No final de cada Sprint, são realizadas duas cerimônias, a revisão (ou review) que tem o objetivo de apresentar o que foi desenvolvido e a retrospectiva, voltada para o time analisar acertos e erros que surgiram na iteração, para proporcionar uma melhoria contínua nos próximos desenvolvimentos do projeto (Schwaber; Sutherland, 2020).

2.2 GESTÃO DE EQUIPES

Segundo Demo, Fogaça e Costa (2018) a gestão de pessoas tem um papel fundamental dentro das organizações, assumindo funções estratégicas dentro da empresa, pois deve-se sempre buscar a evolução das pessoas e aumentar o conhecimento, juntamente com os objetivos da organização. A gestão de pessoas passou por grandes evoluções, sempre inovando para se adequar às necessidades do mercado, com qualificação e eficiência, atuando com práticas de atração, valorização, retenção de talentos e capacitação.

As equipes, para desenvolver um sistema, geralmente são compostas por membros multidisciplinares, para contemplar todas as habilidades que se necessita na realização do projeto. Cada membro desempenha um papel específico na equipe, como análise, desenvolvimento, teste, designer ou infraestrutura. É importante que a equipe tenha uma boa gestão de mudanças, para saber lidar facilmente com as alterações que surgem durante a evolução do projeto e saiba lidar com conflitos, devido às diferenças de opinião ou desafios técnicos que podem surgir (Torres *et al.*, 2023).

No Scrum, os times são compostos por três papéis fundamentais: o *Scrum Master*, que atua como facilitador, guardião do processo e apoia na remoção de impedimento; o *Product Owner*, que representa os clientes, define as prioridades, transcreve para o time as regras do negócio, o que e como precisam ser desenvolvidas; e a *Equipe de Desenvolvimento*, que é auto-organizada e composta por profissionais responsáveis por criar o produto. Todos esses papéis colaboram de forma integrada para alcançar os objetivos do projeto, enfatizando a responsabilidade compartilhada e a busca pela eficiência e qualidade (Schwaber; Sutherland, 2020).

3 METODOLOGIA

Para atender com o objetivo deste artigo, foi realizada uma revisão narrativa da literatura, proposta por Rother (2007). A revisão narrativa possui o intuito de utilizar fontes bibliográficas ou eletrônicas para embasar teoricamente um objetivo, descrevendo e discutindo o estado da arte de um assunto de maneira ampla.

As buscas foram realizadas nas bases de dados eletrônicas Periódicos CAPES e Google Acadêmico. Utilizando como palavras-chaves os seguintes termos: ("métricas" OR "indicadores") AND ("scrum") AND ("gestão de projetos" OR "desenvolvimento de sistemas") e sua variação em inglês, para encontrar artigos também no idioma inglês: ("metrics" OR "indicators") AND ("scrum") AND ("project management" OR "systems development"). Essa busca, trouxe 87 artigos na base dos Periódicos CAPES e 3.090 no Google Acadêmico.

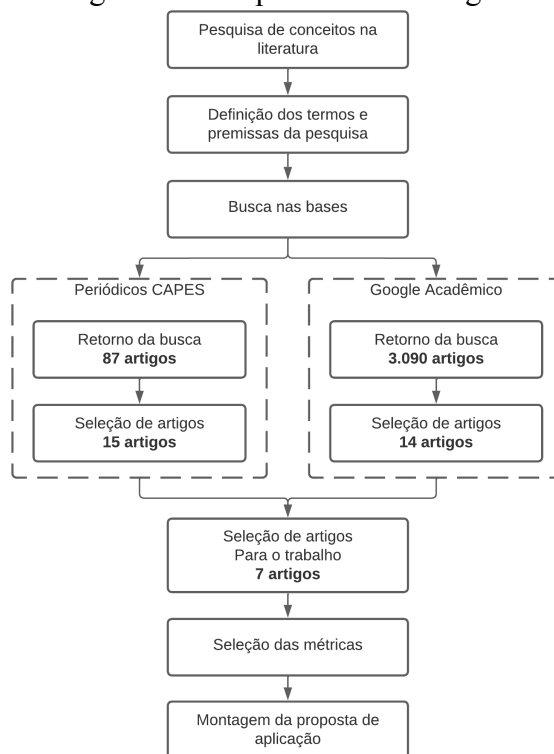
Após esta etapa, foi realizada a leitura dos artigos pelos resumos, para fazer a inclusão dos artigos que possuíam relação ou respondiam ao problema da pesquisa. Foram adicionados artigos publicados no período de 10 anos, entre os anos de 2013 a 2023, artigos que

trouxessem aplicações práticas de métricas usadas em metodologias ágeis, dentro do contexto de equipes de desenvolvimento de sistemas, no idioma inglês ou português.

Juntamente com essa seleção de leitura dos artigos pelos resumos, foi realizada a exclusão dos artigos que não tinham relação ou não respondiam ao tema. Também, foram excluídos artigos fora do período de 10 anos de publicação, artigos que não apresentavam métricas para metodologias ágeis na gestão do projeto e que não estavam escritos nos idiomas inglês ou português. Para os artigos retornados pelo Google Acadêmico, por conta da grande quantidade, foi feita a análise dos 60 primeiros resultados, os demais foram excluídos, visto que os títulos e palavras-chave dos artigos já indicavam temas que apontavam menor aderência ao objeto de pesquisa deste trabalho.

Na etapa final, foi realizada uma seleção dos 7 melhores artigos, os quais continham um conteúdo de análise que melhor responderam à questão norteadora. O critério de seleção desses artigos foi a abordagem sobre métricas que se aplicam na metodologia *Scrum*, descrevendo detalhes sobre cada uma delas e dados sobre como são utilizadas. Com base nesses artigos, realizou-se uma análise para seleção das métricas, de acordo com sua frequência, relevância exposta pelo artigo e que apoiasse para responder o problema da pesquisa.

Figura 01 - Etapas da metodologia.



Fonte: Autoria própria.

A Figura 01 apresenta um fluxo visual de todas as etapas que compõem a metodologia de pesquisa utilizada, de acordo com o que está descrito anteriormente. A análise dos dados, para o desenvolvimento deste trabalho, foi realizada a partir da leitura dos artigos selecionados.

4 APRESENTAÇÃO E DISCUSSÃO DOS RESULTADOS

Após a realização da pesquisa, de acordo com as etapas que foram descritas na metodologia, o desenvolvimento foi realizado com base nos 7 artigos selecionados. No Quadro 01, pode-se observar a relação deles, contendo as informações de título, autor e ano de publicação, juntamente com um identificador numérico (primeira coluna) para referenciá-los no Quadro 02.

Quadro 01 - Relação de artigos selecionados.

Número	Título	Autor
01	Métricas de avaliação para abordagens ágeis em projetos de software	Pegoraro, 2014
02	Monitoramento de métricas de qualidade e produtividade em projetos ágeis de software através da integração de dados extraídos de ferramentas de gestão e testes	Lima Ramos, 2023
03	A Novel Approach for Visualization, Monitoring, and Control Techniques for Scrum Metric Planning Using the Analytic Hierarchy Process	Tekin; Yilmaz; Clarke, 2023
04	Perceived Importance of Metrics for Agile Scrum Environments	Almeida; Carneiro, 2023
05	Use of software metrics in agile software development process	Padmini; Bandara; Perera, 2015
06	Scrum metrics for hyperproductive teams: how they fly like fighter aircraft	Downey; Sutherland, 2013
07	Performance metrics in scrum software engineering companies	Almeida; Carneiro, 2021

Fonte: Autoria própria.

4.1 IMPORTÂNCIA DAS MÉTRICAS

Ao implementar o Scrum como metodologia ágil em um time, é essencial entendê-lo como um ecossistema de partes interdependentes, onde a coordenação dessas partes requer uma inspeção diária para manter um estado de alta energia. Para isso, um conjunto simples de métricas fornece um painel semelhante ao cockpit de uma aeronave, que observa a altitude, direção, velocidade e taxa de descida pode mantê-lo no rumo, mesmo em condições climáticas adversas (Downey; Sutherland, 2013). Almeida e Carneiro (2023) complementam, dizendo que compreender a importância das métricas considerando os papéis específicos do Scrum é crucial para aumentar a coesão da equipe e a qualidade do trabalho.

De acordo com a pesquisa realizada por Padmini, Bandara e Perera (2015), 90% das respostas de todo o time, indicaram que as equipes gostam de usar métricas no processo ágil de desenvolvimento de sistemas devido aos seus benefícios. Os benefícios identificados incluem a capacidade de acompanhar o progresso do projeto, monitorar a qualidade do produto e permitir melhores previsões e gerenciamento de projetos. No entanto, alguns participantes indicaram que não conseguem compreender os benefícios do uso de métricas no processo ágil, isto enfatiza que não só é importante identificar um conjunto apropriado de métricas que seja benéfico para o processo de desenvolvimento, mas também os profissionais precisam de ser conscientizados dos benefícios.

Bons indicadores ajudam a equipe a medir seu próprio desempenho e tomar decisões com base em dados e não apenas em sentimentos ou suposições. Além disso, esses indicadores têm significado fora da equipe original, o que permite que a gerência compare o desempenho de várias equipes com dados quantitativos. Eles também estabelecem um padrão consistente para a coleta de dados, tornando a produtividade uma medida clara e amplamente compreendida (Downey; Sutherland, 2013).

Existe uma grande importância do uso de indicadores de desempenho para monitorar o progresso e promover melhorias em projetos. Antes de iniciar o gerenciamento de projetos, é fundamental definir métricas de acompanhamento, que são essenciais para otimizar resultados e orientar a equipe ao longo de cada etapa da implementação do projeto. No contexto da indústria de software, as métricas desempenham um papel crucial em diversas áreas, incluindo planejamento, estimativa, gerenciamento, monitoramento de projetos, compreensão de qualidade e metas de negócios, além de melhorar a comunicação, processos e ferramentas de desenvolvimento de software (Almeida; Carneiro, 2023).

Os indicadores também fornecem dados concretos para embasar feedbacks para o time e individualmente, apoiando e garantindo a evolução de cada colaborador da organização. Como menciona Downey e Sutherland (2013), bons indicadores fornecem ao gestor uma base sólida para os conselhos que oferecem à equipe. Um conjunto simples de métricas resultante é fácil de implementar e tem um impacto poderoso no desempenho das equipes Scrum. Complementando, sem essas métricas, o desempenho da equipe pode ser instável e a perda de controle resultará em uma diminuição da velocidade, além da equipe, as métricas apoiam também para demonstrar à diretoria da empresa como o desenvolvimento do projeto está evoluindo. Downey e Sutherland (2013) citam que as métricas demonstram claramente o impacto de qualquer modificação, desde a introdução de novas ferramentas e tecnologias até mudanças no processo ou na composição da equipe. Isso pode ajudar os gestores a justificarem pedidos de recursos adicionais, explicando como esses recursos serão aplicados e qual impacto a empresa pode esperar se o investimento solicitado for aprovado. Almeida e Carneiro (2023) contribuem, dizendo que a governança do Scrum é uma tarefa desafiadora, pois não pode se concentrar apenas nos processos de desenvolvimento de software, mas deve envolver múltiplos domínios e membros interdisciplinares.

4.2 SELEÇÃO DAS MÉTRICAS

A partir desses artigos selecionados, foi realizada a seleção das métricas, de acordo com sua relevância, frequência que está citada nos trabalhos analisados, aplicabilidade e coerência para atender com o problema de pesquisa, contribuindo para o monitoramento de projetos e gestão de equipes de desenvolvimento de softwares.

No Quadro 02, pode-se observar as métricas selecionadas na primeira coluna, com os artigos que estão citando-as, na segunda coluna, de acordo com o número de cada artigo do Quadro 01.

Quadro 02 - Relação de métricas e suas citações nos artigos.

Métrica	Artigos
Burndown da iteração	01, 03, 04, 05, 07
Burnup do projeto	01, 03, 07
Lead Time	01, 07

Cycle Time	01, 02, 03
Acurácia nas estimativas	01, 04, 05, 06
Velocidade	01, 03, 04, 05, 06, 07
Número de defeitos na iteração	01, 02, 04, 05, 07

Fonte: Autoria própria.

4.3 PROPOSTA PARA UTILIZAÇÃO DE MÉTRICAS

Após indicar a relação das métricas, no Quadro 2, nesta subseção apresenta-se a proposta de utilização das métricas selecionadas, realizando uma explanação de cada uma, com uma explicação de como extrair os indicadores, a contribuição para a gestão dos times e exemplos de utilização.

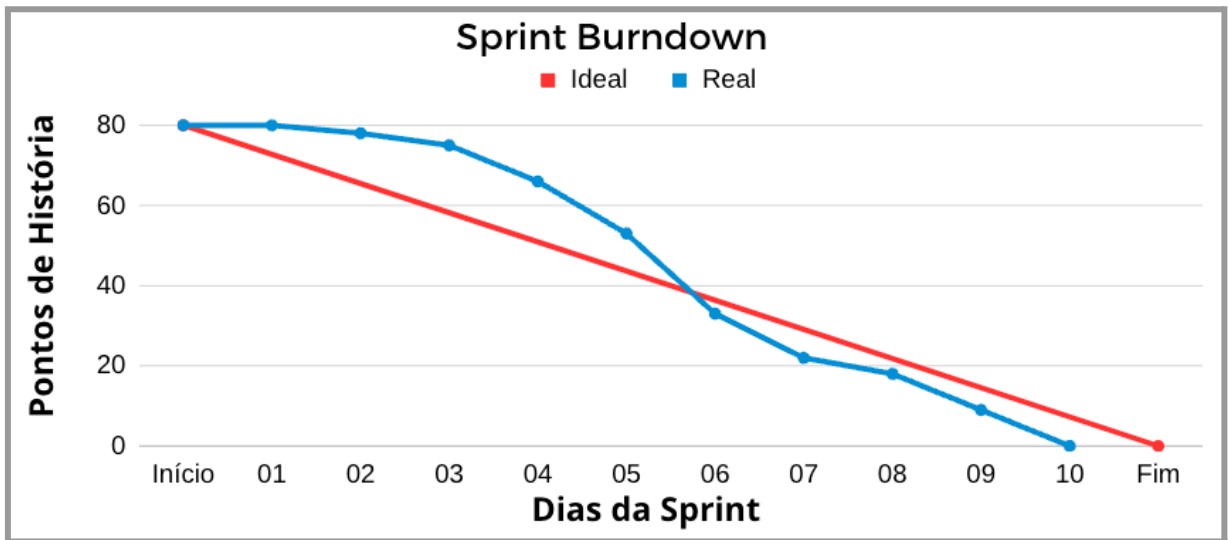
4.3.1 Burndown da Iteração

O burndown da sprint, que pode ser representado graficamente, apresenta a taxa de conclusão do trabalho e quanto resta a ser feito ao longo da sprint (Almeida; Carneiro, 2021). Ele compara o número de pontos (ou horas) planejados na iteração com o número de pontos realizados até um determinado momento. Isso ajuda a equipe a avaliar se está cumprindo as tarefas conforme o planejado. Se houver uma grande diferença entre os pontos estimados e os realizados, pode-se prever um erro na estimativa, indicando que o trabalho realizado está divergente do que foi estimado.

Segundo Tekin, Yilmaz e Clarke (2023), o gráfico de burndown é uma das métricas que possui maior frequência de utilização, estando presente na maioria dos projetos de desenvolvimento de sistemas. Essa métrica permite acompanhar o progresso, identificar tendências e determinar quanto ainda falta para atingir a conclusão de todas as tarefas que foram planejadas no período (Pegoraro, 2014).

Na Figura 02, pode-se observar a proposta de representação do indicador, em um gráfico que possui dois eixos: o vertical que representa a quantidade de pontos a serem desenvolvidos e o horizontal que representa o número de dias da sprint. Juntamente com duas linhas percorrendo esse gráfico: a vermelha representando o desempenho ideal, uma reta linear que inicia com a soma total de pontos de todas as tarefas da iteração e atinge zero no final do período, e a azul representando o desempenho real da equipe ao longo da iteração.

Figura 02 - Exemplo de gráfico de Burndown da Sprint.



Fonte: Autoria própria.

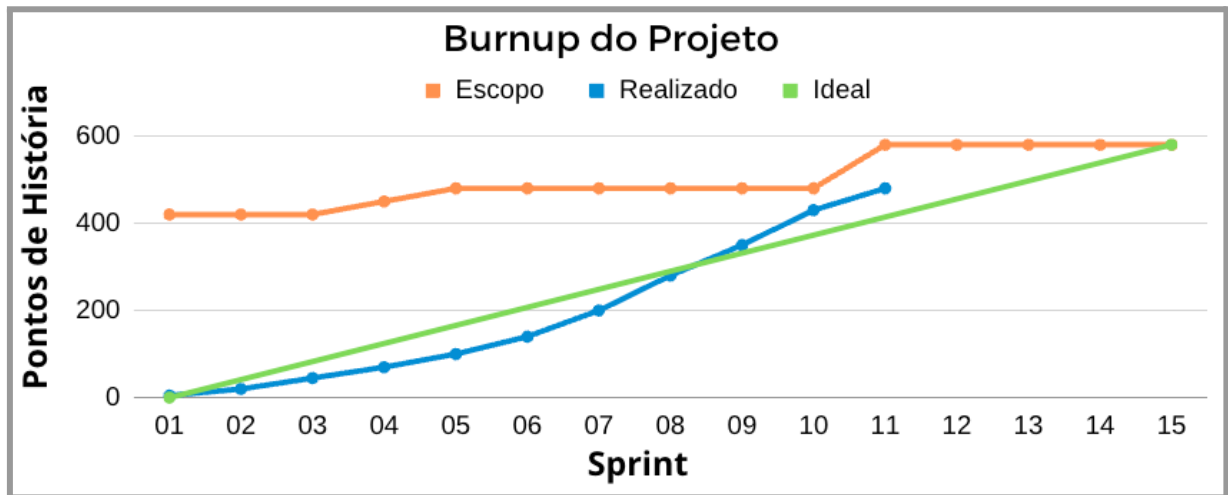
4.3.2 Burnup do Projeto

O burnup é representado por um gráfico que mostra uma comparação entre o trabalho concluído e a meta planejada em todo o projeto, servindo para acompanhar o progresso da equipe (Tekin; Yilmaz; Clarke, 2023). O gráfico de burnup é composto por duas linhas: uma mostra o trabalho total realizado até o momento, geralmente em pontos, enquanto a outra representa a meta da entrega ou do projeto. A diferença entre essas linhas ajuda a equipe a determinar a quantidade de trabalho restante para concluir a meta da entrega ou do projeto. Este gráfico é valioso em reuniões de planejamento de iterações e entregas, e deve ser atualizado ao final de cada sprint (Pegoraro, 2014).

Enquanto o burndown da iteração traz uma visão mais fracionada, olhando apenas para a sprint que se está trabalhando, o burnup observa o backlog como um todo. Servindo assim para trazer uma visão da evolução que se está tendo no projeto até sua conclusão, trazendo um indicador que informa se o que já foi desenvolvido está dentro do esperado para atingir a conclusão do projeto no período e no esforço que foi planejado.

Também é citado por Almeida e Carneiro (2021) com a nomenclatura "burndown do release", mas tem o mesmo objetivo de representar a taxa na qual o trabalho é concluído e quanto trabalho resta a ser feito até finalizar a versão do produto.

Figura 03 - Exemplo de gráfico de Burnup do Projeto.



Fonte: Autoria própria.

Na Figura 03, pode-se observar a proposta de um exemplo de gráfico de Burnup, contendo as duas linhas como mencionado anteriormente e um terceira linha extra (verde) para indicar o andamento ideal para que o projeto seja finalizado dentro do prazo. Portanto, caso a linha do trabalho realizado esteja abaixo da linha ideal, significa que o projeto está atrasado, caso esteja acima, representa um adiantamento. Vale observar também que a linha que indica o escopo tem um aumento em algumas sprints, isso também pode acontecer quando se tem um incremento de escopo durante o projeto.

4.3.3 Lead Time

O Lead Time é o período que uma funcionalidade ou requisito leva desde a sua solicitação até ser entregue ao cliente. Ao analisar antecipadamente a métrica de Lead Time, é possível obter previsibilidade sobre o tempo necessário para concluir uma tarefa. Esses dados podem ser representados graficamente para visualizar a variabilidade no tempo de entrega, sendo que em um processo confiável, essa variabilidade deve ser mínima. Para aplicar essa métrica, é necessário ter um quadro de acompanhamento diário de tarefas, no qual seja registrado o tempo que cada tarefa permanece em cada fase do quadro até sua conclusão final (Pegoraro, 2014).

O Lead Time pode ser calculado levando em consideração a quantidade de dias que passaram entre a data de abertura da solicitação e a data de finalização, em que foi disponibilizada no produto final. Com esse dado, pode-se por exemplo, fazer uma média do Lead Time de todas as tarefas já finalizadas e, assim, fornecer uma estimativa para o solicitante com um tempo médio que o time precisa para finalizar uma demanda.

4.3.4 Cycle Time

O Cycle Time é uma métrica que mede o tempo decorrido desde o início do trabalho em uma tarefa até a sua conclusão. Essa métrica se baseia no princípio *Lean* (Pegoraro, 2014), onde se busca entregar produtos de trabalho em ciclos curtos, minimizando o tempo entre receber uma solicitação e entregar uma funcionalidade de software conforme solicitado. Além disso, o Cycle Time revela a capacidade do processo de desenvolvimento, auxiliando na previsão da quantidade de trabalho em andamento que será concluída (Pegoraro, 2014).

Fazendo uma comparação com o Lead Time, ambas métricas trazem a medida de tempo para a evolução da tarefa. Enquanto o Lead Time considera o tempo total desde a solicitação até a disponibilidade para o usuário, o Cycle Time apresenta apenas o tempo em que a tarefa ficou em execução, isso pode trazer vários indicadores, como o tempo médio que o time precisa para desenvolver uma tarefa ou indicar quais foram as tarefas que mais necessitam de tempo de desenvolvimento. Para obter o Cycle Time de uma tarefa, pode-se fazer um cálculo somando os dias que a tarefa ficou em desenvolvimento dentro da iteração.

4.3.5 Acurácia das Estimativas

Uma das questões essenciais em um projeto de software é determinar, antes de sua execução, a quantidade de esforço necessária para sua conclusão, essa informação serve como base para o planejamento do projeto. A realização de estimativas desempenha um papel fundamental em projetos ágeis, embora seja uma tarefa desafiadora e sujeita a erros. A taxa de acerto nas estimativas possibilita avaliar a porcentagem de tarefas concluídas dentro do prazo estipulado. Para aplicar essa métrica, é necessário ter um quadro de acompanhamento de tarefas, no qual são registrados o tempo estimado para a execução das tarefas e o tempo efetivamente gasto (Pegoraro, 2014).

A bibliografia traz diferentes indicadores para medir a assertividade nas estimativas. Downey e Sutherland (2013) trazem uma versão simplificada, onde o cálculo da acurácia se dá com base na relação do tempo estimado da tarefa com o tempo efetivamente utilizado para executar a tarefa, fazendo uma divisão entre esses dois valores. Então, por exemplo, uma tarefa que foi estimada em 20 horas e foi realizada em 22 horas teve uma acurácia de 90% de assertividade.

Consegue-se também calcular a assertividade geral da sprint realizando um somatório de todas as estimativas e trabalho realizado em todas as tarefas da iteração. Segundo Downey e Sutherland (2013), esse número, idealmente, deve permanecer em torno de 80% em equipes saudáveis que enfrentam desafios em seu trabalho.

4.3.6 Velocidade

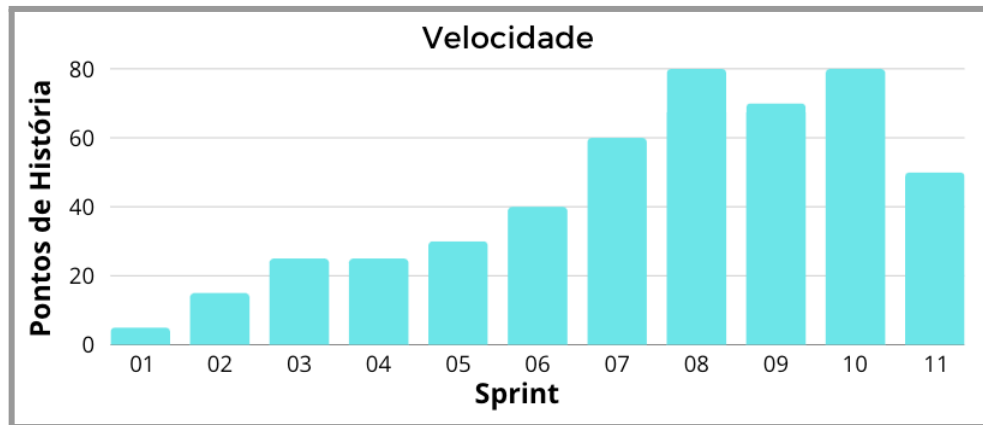
A velocidade do time é definida pela quantidade de trabalho que uma equipe de desenvolvimento pode realizar durante um sprint, pode ser calculada considerando os pontos de história que foram entregues na iteração (Downey; Sutherland, 2013). Segundo resultados da pesquisa de Tekin, Yilmaz e Clarke (2023), velocidade é o indicador que ficou em 3º lugar em maior frequência de utilização entre seus entrevistados, ressaltando a relevância que essa métrica possui nas equipes.

Uma maneira eficiente de avaliar o desempenho da equipe é considerando as métricas de velocidade e capacidade de trabalho. A velocidade é uma métrica amplamente conhecida e usada no Scrum que mede a taxa de trabalho, no entanto, usar essa métrica isoladamente para comparar o desempenho entre diferentes equipes pode trazer resultados injustos e distorcidos. As diferenças no ponto de partida das equipes e nas estimativas feitas para cada tarefa de usuário podem influenciar significativamente na velocidade, portanto, comparar a velocidade de diferentes equipes pode ter efeitos negativos e gerar situações desconfortáveis. É importante considerar o contexto e outras métricas ao avaliar o desempenho entre diferentes equipes. (Almeida; Carneiro, 2023)

Com este indicador pode-se montar um gráfico de velocidade do time entre as iterações, permitindo realizar conjecturas sobre a quantidade de trabalho que a equipe pode realizar em futuras sprints. O gráfico também ajuda nas decisões sobre o quanto é possível comprometer-se de forma viável e, portanto, pode ser útil durante as reuniões de planejamento do sprint (Tekin; Yilmaz; Clarke, 2023). Esse gráfico também possibilita acompanhar a

evolução do time, onde a tendência nos projetos é que a velocidade se incremente com o passar do tempo, devido ao aumento da maturidade e domínio do projeto. Caso a velocidade passe a diminuir com o decorrer do tempo, é um ponto de atenção que deve ser investigado para entender e tratar os motivos que levam a um possível retrocesso da equipe.

Figura 04 - Exemplo de gráfico de velocidade do projeto.



Fonte: Autoria própria.

A Figura 04 representa a proposta de um gráfico de barras da velocidade, onde cada barra representa a quantidade de pontos de história entregues na Sprint, distribuídos em todas as Sprints finalizadas. Nesse exemplo podemos observar que a velocidade aumentou com o passar do tempo, indicando uma evolução da equipe, que conseguiu desenvolver quantidades maiores de pontos conforme o projeto progrediu.

4.3.7 Número de Defeitos Encontrados na iteração

Essa métrica apresenta o número de defeitos que surgiram em uma sprint (Almeida; Carneiro, 2023). Métrica bastante simples de calcular e representar pois trata da soma de defeitos ou *bugs* que surgem em uma iteração, mas é um indicador bem importante pois permite o acompanhamento da quantidade de falhas que são geradas pelo time, durante o desenvolvimento. Caso esse número for muito elevado é um ponto de atenção, onde a equipe está deixando muitos defeitos em aberto no projeto.

Ademais, como afirma Pegoraro (2014), essa métrica pode ser utilizada para comparar a variação de quantidade de defeitos encontrados entre as iterações passadas, para efeito de avaliação se houve uma melhoria na qualidade do produto durante a evolução do projeto. Para

isso, é necessário conseguir visualizar o histórico de defeitos que surgiram no projeto ao longo do tempo.

Segundo Lima Ramos (2023), essa métrica pode ser utilizada para olhar em um escopo individualizado de cada membro do time, para avaliar qual foi o membro que mais ocasionou falhas no sistema e realizar um feedback direcionado para cada colaborador da equipe. Mas nem sempre se deve levar em conta apenas a quantidade de defeitos, Almeida e Carneiro (2021) defendem que se torna pertinente analisar em conjunto as causas da ocorrência de cada defeito, que podem não estar focadas apenas no componente técnico, mas sim relacionado à especificação da necessidade do cliente ou definição equivocada da regra de negócio.

4.4 ACOMPANHAMENTO DAS MÉTRICAS

Almeida e Carneiro (2023) falam sobre a importância de olhar várias métricas em conjunto, e não apenas cada uma individualmente, pois pode ser realizado o cruzamento dos dados e fazer combinações entre elas, para buscar um resultado mais assertivo e identificar a causa dos problemas com mais facilidade. Por exemplo, pode-se combinar a métrica da velocidade com o Cycle Time e/ou Lead Time, para facilitar a previsão de futuras datas de entrega ou marcos de lançamento, ajudando assim, as partes interessadas e os proprietários do produto a planejar e definir as expectativas com mais precisão.

Para facilitar a extração das métricas é recomendado usar uma ferramenta de apoio. Como cita Padmini, Bandara e Perera (2015), ferramentas são úteis porque simplificam a coleta e análise de parâmetros relacionados às métricas. Existem diversas ferramentas no mercado que possuem a função de apoiar em todo fluxo do Scrum, e juntamente já trazem a exibição ou a possibilidade de criação dos indicadores de acordo com a métrica que se deseja acompanhar. Os resultados da pesquisa de Padmini, Bandara e Perera (2015) relataram que muitas empresas acabam iniciando com uma planilha digital básica para controle e monitoramento das atividades, mas o ideal é que seja utilizada uma ferramenta específica para a gestão de projetos ágeis, por trazer muitos mais recursos e facilidades para esse controle.

Segundo resultados da pesquisa realizada por Almeida e Carneiro (2023), as métricas que mais têm relevância para os projetos são as que estão relacionadas à performance do time, como a velocidade e assertividade nas estimativas. Isso se deve pois essas são as métricas que ajudam a avaliar o andamento do projeto, se está progredindo como esperado ou se está

demandando mais tempo e recursos do que o planejado, para, a partir desses dados, conseguir prever e tomar ações para minimizar os impactos.

Por fim, a proposta de aplicação das métricas se dá com base na utilização das 7 métricas relatadas e detalhadas neste artigo, seguindo uma análise em conjunto e periodicamente, buscando sempre a evolução do time e dos colaboradores.

5 CONSIDERAÇÕES FINAIS

Este estudo buscou oferecer contribuições relevantes para ampliar a literatura em relação a aplicação de métricas que auxiliam na evolução de times e projetos, melhorando o processo de desenvolvimento de produtos que utilizam o Scrum como metodologia ágil. Com o apoio de métricas bem definidas a gestão consegue avaliar o nível do time, ou individualmente de acordo com a produtividade, e também possibilitam a comparação em diferentes períodos de tempo, para indicar uma evolução ou recorrência de falhas, auxiliando na avaliação e feedback dos colaboradores. No ponto de vista prático, esse estudo também contribuiu para apoiar as organizações, que querem adotar ou já utilizam o Scrum, e desejam um auxílio na definição de métricas, ou necessitam de sugestão para melhorar a performance da equipe. A aplicação das métricas proporciona que as empresas avaliem seus processos de desenvolvimento de software, garantindo que as equipes Scrum entreguem o máximo valor ao buscar uma evolução com o passar do tempo.

Com isso, observa-se que os objetivos da pesquisa foram atendidos, onde o desenvolvimento deste artigo apresentou a proposta de um conjunto de aplicação de 7 métricas para utilizar no Scrum, descrevendo como cada uma pode ser aplicada e demonstrando que indicadores podem ser extraídos com base nesses dados.

Esse estudo teve algumas limitações, como a dificuldade de buscar artigos que atendessem a necessidade do objetivo, devido à baixa quantidade de obras relacionadas com esse tema. Por isso optou-se por incluir artigos em inglês, aumentando consideravelmente a quantidade de material. Outro ponto, foi a quantidade máxima de páginas, que limitou um pouco a quantidade de material e exemplos que poderiam estar presentes neste artigo. Contudo, mesmo com essas dificuldades, não se teve um grande impacto no desenvolvimento deste trabalho.

A partir desse trabalho, pode-se elencar possibilidade de futuras pesquisas, para contribuir com o tema, como uma pesquisa de campo, buscando compreender como as empresas estão organizadas em relação às métricas em projetos de desenvolvimento de sistemas, entendendo a maturidade a abertura que existe para evolução dos processos e equipe. Outra oportunidade também, é a realização do acompanhamento em uma organização ou equipe que não possui métricas definidas, orientando a aplicação das que foram propostas neste trabalho, para posteriormente verificar a evolução que a equipe obteve com base nos indicadores.

REFERÊNCIAS

- ALMEIDA, Fernando; CARNEIRO, Pedro. Performance metrics in scrum software engineering companies. **International Journal of Agile Systems and Management**, v. 14, n. 2, p. 205-223, jan. 2021.
- ALMEIDA, Fernando; CARNEIRO, Pedro. Perceived Importance of Metrics for Agile Scrum Environments. **DOAJ Directory of Open Access Journals Information (Basel)**, v. 14, n. 6, p. 327, 11 jun. 2023.
- BATALHA, Victoria. Brasil teve um crescimento de 80% na criação de vagas na área de TI, revela pesquisa. **JC Concursos 2023**. Disponível em: <<https://jcconcursos.com.br/noticia/empregos/brasil-teve-um-crescimento-de-80-na-criacao-de-vagas-na-area-de-ti-revela-pesquisa-107156>>. Acesso em: 15 mar. 2023.
- DEMO, Gisela; FOGAÇA, Natasha; COSTA, Ana Carolina. Políticas e práticas de gestão de pessoas nas organizações: cenário da produção nacional de primeira linha e agenda de pesquisa. **Cadernos Ebape. BR**, v. 16, p. 250-263, 2018.
- DOWNEY, Scott; SUTHERLAND, Jeff. Scrum metrics for hyperproductive teams: how they fly like fighter aircraft. **46th Hawaii International Conference on System Sciences. IEEE**, p. 4870-4878, 18 mar. 2013.
- PADMINI, KV Jeeva; BANDARA, HMN Dilum; PERERA, Indika. Use of software metrics in agile software development process. **2015 Moratuwa Engineering Research Conference (MERCon). IEEE**, p. 312-317, 2015.
- PEGORARO, Raquel Aparecida. Métricas de avaliação para abordagens ágeis em projetos de software. **UFRGS LUME Repositório Digital**, 2014.
- LIMA RAMOS, Rafael Anderson. Monitoramento de métricas de qualidade e produtividade em projetos ágeis de software através da integração de dados extraídos de ferramentas de gestão e testes. **IFPB Repositório**, 2023.
- RAMOS, Gabriel Russo et al. Comparação entre metodologias de gerenciamento de projetos ágil e tradicional: Um estudo de caso em uma Startup de pequena escala. **Brazilian Journal of Development**, v. 6, n. 3, p. 15025-15047, 2020.
- ROTHER, Edna Terezinha. Revisão Sistemática X Revisão Narrativa. **Acta Paul Enferm**, 20(2): v-vi, jul. 2007.
- SCHWABER, Ken; SUTHERLAND, Jeff. The Scrum Guide. **scrum.org**, nov. 2020.
- TEKIN, Nesib; YILMAZ, Murat; CLARKE, Paul. A Novel Approach for Visualization, Monitoring, and Control Techniques for Scrum Metric Planning Using the Analytic Hierarchy Process. **Journal of Software: Evolution and Process**, v. 35, n. 8, 2023.
- TORRES, Fernando Ibarra *et al.* Knowledge Transfer in Software Development Teams - Systematic Mapping. **IEEE Revista Iberoamericana de Tecnologías del Aprendizaje**, v. 18, n. 2, mai, 2023.