

AGILE CULTURE: MAKING
PROCESSES FASTER
METODOLOGIAS
ÁGEIS

ANDRE LUIZ DIAS RIBEIRO



LISTA DE FIGURAS

Figura 1.1 - Analogia a gestão de projetos ágeis.....	8
Figura 1.2 – Processo de gestão com SCRUM	13
Figura 1.3 - Analogia a partes envolvidas	15
Figura 4.4 - Site do Google Ventures	25
Figura 1.5 - O sprint dá às equipes um atalho para aprender sem construir e lançar	26
Figura 1.6- Etapa do <i>Design Sprint</i>	26



SUMÁRIO

1 METODOLOGIAS ÁGEIS	4
1.1 Manifesto Ágil	4
1.2 Conceitos ágeis modernos.....	6
1.3 Gestão ágil de projetos	7
1.4 Extreme Project Management (XPM)	9
1.4.1 Características da XPM	9
1.5 Gestão de projetos com SCRUM	12
1.5.1 Características do Scrum	13
1.6 Análise da abordagem tradicional e ágil.....	15
1.7 Análise comparativa entre o gerenciamento tradicional e o gerenciamento ágil.....	17
1.8 Diferenças e Similaridades entre as abordagens	18
1.8.1 Gestão do escopo	19
1.8.2 Gestão do tempo	19
1.8.3 Gestão de custos	20
1.8.4 Gestão da qualidade	20
1.8.5 Gestão de riscos	21
1.8.6 Gestão da comunicação	21
1.8.7 Gestão de recursos humanos	22
1.8.8 Gestão da aquisição	22
1.8.9 Gestão da integração	22
1.8.10 Gestão das partes envolvidas	23
1.9 Design Sprint	24
1.9.1 O que é <i>design sprint</i> , afinal?	25
1.9.2 Etapas.....	27
1.9.2.1 Entender	27
1.9.2.2 Divergir	27
1.9.2.3 Decidir.....	27
1.9.2.4 Prototipar	28
1.9.2.5 Validar.....	28
REFERÊNCIAS	30

1 METODOLOGIAS ÁGEIS

Seja bem-vindo ao SHIFT **Agile Culture: Making Processes Faster**.

Ao longo de 4 capítulos você aprenderá métodos ágeis que vão do conceito à prática, tornando sua equipe mais ágil e elevando estas técnicas a nível organizacional.

1.1 Manifesto Ágil

O manifesto ágil foi publicado em 2001, quando Kent Beck e 16 outros notáveis desenvolvedores, produtores e consumidores de software, conhecidos como Aliança Ágil, assinaram o Manifesto para o Desenvolvimento Ágil de Software com o objetivo de apontar novas direções na execução de projetos.

Esse manifesto está publicado em www.manifestoagil.com.br

A base para elaboração desse manifesto ágil considera que os requisitos não são estáveis durante todo o projeto, que as mudanças vão ocorrer em qualquer cenário e que as execuções sequenciais das fases de um projeto não são tão previsíveis como gostaríamos que fossem. Isso posto, ficam definidos quatro fundamentos básicos que guiam o manifesto:

- Indivíduos e interações, em vez de processos e ferramentas.
- Software funcionando, em vez de documentação abrangente.
- Colaboração do cliente, em vez de negociação de contratos.
- Resposta às modificações, em vez de seguir um plano.

Esses fundamentos são a base de todos os frameworks ágeis, estabelecendo o alicerce sobre os quais são desenvolvidos os padrões de agilidade: foco no time, cliente fazendo parte do projeto, entregas durante o ciclo do projeto e mudanças são bem-vindas em qualquer momento do projeto.

Como resultado desses quatro fundamentos, o manifesto estabeleceu doze princípios básicos que detalham um pouco mais as diretrizes que devem ser seguidas, a saber:

- **Satisfação do cliente por meio de entrega contínua:** ao avaliar o produto durante todo o projeto, o cliente percebe a evolução e verifica a qualidade dos produtos.
- **Modificação de requisitos são bem-vindas:** mudanças sempre vão ocorrer. A forma de tratá-las deve ser feita de maneira transparente.
- **Entrega de software funcionando frequentemente:** ao verificar que o produto funciona, a equipe e o cliente tornam-se mais confiantes.
- **Pessoal de negócio e desenvolvedores trabalhando juntos:** o conceito do "projeto é nosso", não somente da equipe ou só do cliente, pertence a todos os envolvidos.
- **Indivíduos motivados:** para fazer um projeto, todos devem estar dispostos na execução e comprometidos com os resultados.
- **Equipes auto-organizadas:** um dos fundamentos da agilidade. Não há chefes, mas sim decisões do grupo.
- **Conversas face a face para levantar informações:** com a participação efetiva do cliente como parte do time, as dúvidas e requisitos são discutidos na hora.
- **Software funcionando como medida de progresso:** à medida que as entregas são realizadas, a evolução de completude é percebida claramente.
- **Ritmo constante de desenvolvimento sustentável:** equipe e demais envolvidos percebem os benefícios da agilidade, se motivam e se comprometem com ele.
- **Excelência técnica:** fazer certo da primeira vez permite a construção efetiva e com qualidade.
- **Simplicidade:** fazer o essencial para entregar o produto.

- **Equipe reflete sobre como se tornar mais efetiva, então sintoniza e ajusta adequadamente seu comportamento:** por meio de lições aprendidas o time identifica os erros e acertos para melhorar o desempenho a cada novo ciclo.

Esses princípios, quando seguidos pelo time, realmente tornam o resultado final um diferencial para todos os envolvidos, proporcionando a satisfação do cliente e a motivação do time de construção.

1.2 Conceitos ágeis modernos

Hoje, quinze anos após a publicação do Manifesto Ágil, como vocês podem observar, muita coisa evoluiu e diversos conceitos se consolidaram no mercado como práticas essenciais para se obter sucesso nos projetos, tais como: as entregas parciais, time auto-organizado, aceitação de mudanças, entre outras.

Nessa evolução constante e na vivência prática da aplicação dos princípios ágeis, surgiram novos princípios que amadurecem e permitem o refinamento constante dessa abordagem, destacando-se quatro princípios básicos que são aplicados em grandes organizações como Google, Amazon, entre outras grandes corporações, a saber:

- **Torne as pessoas incríveis:** todos os envolvidos em um projeto têm expectativas. Fique atento! Observe e discuta suas necessidades. Traga à mesa. Torne as pessoas parte da solução e criadoras do projeto.
- **Faça da segurança um pré-requisito:** o ambiente atual é digital e as ameaças estão ativas em todos os ramos e lugares; preze em construir um produto seguro e que proporcione essa sensação aos usuários. Não podemos construir soluções vulneráveis a qualquer pessoa mal-intencionada.
- **Erre rápido e aprenda rápido:** para aprender é necessário errar. Não tenha medo de errar! Mas aprenda com o erro e não torne a repeti-lo. Experimentar faz parte do aprendizado contínuo e seu refinamento leva à excelência.

- **Entregue valor constantemente:** as entregas parciais são um dos pilares da agilidade. No entanto, não basta entregar, o produto deve ter valor para o cliente. Ele deve ser capaz de reconhecer cada entrega como parte de um todo, mas que faz sentido sozinho, atingindo um objetivo específico de negócio.

Detalhes sobre os conceitos ágeis modernos podem consultados em:
<http://modernagile.org/>

Esse conjunto de conceitos e práticas aplicadas (em conjunto, isoladamente ou em partes) em projetos 100% ágeis ou não, são benéficas em todos os sentidos. Procure aplicá-las para maximizar seus resultados e experimentar na prática os benefícios dessa abordagem inovadora e desafiadora.

1.3 Gestão ágil de projetos

A gestão ágil de projetos surgiu com a criação dos processos ágeis de desenvolvimento de software. O principal objetivo é permitir o controle das atividades de construção que utilizam técnicas ágeis para gerar informações gerenciais sobre o andamento e sobre a evolução do projeto.

Processos ágeis possuem características específicas que precisam ser identificadas para terem o tratamento adequado.

Pressman (2006) destaca três características principais:

- Os requisitos de um projeto ágil são instáveis por natureza. O cliente inclui e remove requisitos à medida que o projeto evolui, adequando-o às suas necessidades imediatas.
- Há uma interposição das fases de projeto e de implementação fazendo com que a arquitetura e requisitos técnicos sejam tratados à medida que são criados.
- As fases de desenvolvimento do projeto não são sequenciais como de costume. Ocorrem de acordo com as condições do projeto.

Sob essas condições, a gestão de projetos é desafiada a estabelecer processos ou reforçar os já existentes para conseguir controlar e acompanhar a evolução do desenvolvimento do software.



Figura 1.1 - Analogia a gestão de projetos ágeis
Fonte: Banco de imagens Shutterstock (2017)

Duas técnicas se destacam para a gestão de projetos ágeis: a definição da estratégia e a construção incremental do software, contendo as funcionalidades definidas pelo cliente e construídas com a qualidade esperada (entrega de valor) e o feedback constante do cliente em todas as etapas da construção, desde os requisitos até o aceite final do incremento, proporcionando agilidade na resolução de pendências e na avaliação dos requisitos.

No mercado, destacam-se dois métodos de gestão de projetos ágeis que incorporam essas técnicas e outras mais, permitindo uma condução controlada do software: uma é o método Extreme Project Management (XPM), e o outro, o SCRUM, processo da indústria automobilística adequado à construção de software.

1.4 Extreme Project Management (XPM)

O primeiro método criado para a gestão de projetos ágeis foi o Extreme Project Management (XPM) com o objetivo de gerenciar os projetos que utilizam o método de desenvolvimento Extreme Programming (XP), criado por Beck (1999). É uma técnica construída sobre os princípios do XP, adequada para equipes pequenas e projetos com requisitos imprecisos e instáveis.

A principal diferença entre XPM e a metodologia tradicional é a abertura às mudanças. Em XPM, as mudanças são rotineiras e fazem parte do processo. Devem ser tratadas e priorizadas de acordo com as necessidades do cliente, sempre com foco em satisfazer suas necessidades (BECK, 1999). Além da abertura às mudanças, a XP possui outras características importantes, como a participação efetiva do cliente, a comunicação constante e o planejamento por cenários.

1.4.1 Características da XPM

As características principais da XPM são a base do processo ágil de desenvolvimento, cabendo à gestão do projeto se adaptar para cuidar do projeto de forma diferente, não mais com base em processos repetitivos, mas na capacidade de se adaptar ao ambiente do projeto. Essas principais características são apresentadas a seguir:

a) Participação efetiva do cliente:

O cliente participa do começo ao fim do projeto. Ele tem total autonomia de conversa com o gerente de projeto para realizar modificações, corrigir erros e/ou incrementar ou retirar qualquer requisito do projeto. O cliente e o gerente de projeto devem caminhar juntos durante todo o projeto.

b) Planejamento por cenários:

Todos os projetos apresentam riscos e estão constantemente sujeitos a mudanças. Essa característica é maximizada em projetos ágeis. Por isso, torna-se uma condição básica para o gerente de projeto fazer o planejamento por cenários (ou incremental) para atender ao projeto de acordo com as necessidades fornecidas pelo

cliente. Em XPM não é feito o planejamento macro do projeto, o que não permite a visualização do seu final.

c) Mudanças bem-vindas:

O escopo de um projeto ágil é altamente mutável. As mudanças devem ser registradas e controladas em reuniões diárias entre o gerente de projetos e os *stakeholders*, com o objetivo de decidir se serão feitas ou não, bem como sua prioridade dentro do processo de construção. O gerente de projetos ainda deve analisar o impacto para avaliar se houve alteração em expectativas de sucesso no escopo, nos objetivos, nos custos, nos riscos, na qualidade ou em projetos relacionados.

d) Documentação mínima:

O XP prega a não utilização de documentação. Porém, na XPM um conjunto mínimo e estritamente necessário de artefatos para o projeto deve ser definido entre o gerente, a equipe e o cliente. Esses artefatos devem ter finalidades técnicas e de forma alguma burocratizar o desenvolvimento. Entre eles destacam-se: as histórias das funcionalidades, modelo de dados, roteiro de testes, entre outros.

e) Comunicação constante:

Como nos projetos tradicionais, o gerente de projetos deve gastar a maior parte do seu tempo na comunicação entre todos os *stakeholders* do projeto. Quanto melhor entender as necessidades e expectativas deles, maiores as chances de sucesso do projeto. Essa comunicação deve ser pessoal e realizada constantemente, mas como na XPM há pouca documentação, a confiança e a parceria tornam-se essenciais para o sucesso do projeto.

f) Foco no contexto

O contexto é mais importante do que o conteúdo. O gerente de projetos na XPM deve estar focado no aspecto empresarial do projeto, ou seja, nos objetivos e nos benefícios do projeto, em vez dos aspectos técnicos do produto ou serviço que está sendo realizado no projeto. Quanto menos o gerente de projetos souber sobre os assuntos técnicos do projeto, melhor, pois ele deve estar focado em orientar

trabalhadores do conhecimento usando criatividade, flexibilidade e atenção às pessoas.

g) Suporte à implantação:

Em XPM o projeto não termina após a implantação. O acompanhamento e o suporte ao produto pós-implantação são práticas que visam garantir que os resultados do projeto foram alcançados.

h) Gerente de projetos como facilitador:

O gerente de projetos em XPM deve ser mais proativo e facilitador do que nunca. Sua atuação na redução de conflitos, alinhamento de expectativas, motivação das pessoas e posicionamento constante são chaves para o sucesso do projeto. Todas as atividades devem ser feitas em conjunto com a equipe e o cliente, criando um alto grau de comprometimento com as atividades planejadas.

i) Papel dos *stakeholders*:

NA XPM, os *stakeholders* funcionam como gerentes sêniores do projeto. Sua atuação é de parceria e apoio, e não apenas de validação. As seguintes responsabilidades devem ser exercidas por eles:

- Participar das sessões de planejamento do projeto que definem escopo, objetivos e benefícios.
- Apoiar o gerente de projetos nos conflitos internos da organização que podem vir afetar o projeto.
- Monitorar os resultados do projeto, principalmente os relacionados à tríplice restrição – prazo, custo e qualidade.

Em XPM, os critérios de sucesso do projeto são essenciais e devem ser acompanhados e monitorados durante todo o ciclo do desenvolvimento do software. A equipe de projeto deve definir quais são aqueles que serão atingidos no início das atividades. Alguns fatores críticos de sucesso são listados a seguir, mas não se limitam a esses:

- Entregar as funcionalidades definidas no escopo.
- Atender ao prazo e ao custo do projeto.

- Atender às expectativas dos envolvidos.
- Garantir a entrega de produtos com qualidade.
- Satisfazer a equipe do projeto.

1.5 Gestão de projetos com SCRUM

O *framework* SCRUM é um dos mais utilizados para projetos ágeis. Utiliza um processo incremental e iterativo que produz partes do projeto de forma constante, contendo funções definidas e priorizadas pelo cliente, que faz o aceite de cada entrega produzida em períodos de duas a quatro semanas.

O SCRUM está fundamentado nas questões ágeis do cliente como parte da equipe, incrementos em período de tempo fixo e aberto às solicitações de mudanças, sendo adaptativo em relação ao ambiente e colaborativo com as partes interessadas. A figura Processo de gestão com SCRUM ilustra suas principais atividades.

Para mais informações sobre o processo SCRUM e suas certificações acesse:
www.scrum.org ou www.scrumalliance.org

No SCRUM, além da existência das atividades que devem ser realizadas durante o processo de construção, há uma definição clara de papéis que deve ser observada e respeitada. Esses papéis são:

- **Scrum Master:** Exerce a função do gerente de projetos, porém com a ação de facilitador entre a equipe e o cliente. Suas funções básicas são proteger o relacionamento com o cliente, resolver os problemas que ocorrem durante o projeto, manter a equipe focada na execução do sprint e garantir o uso correto das regras SCRUM.
- **Product Owner (PO):** É o representante do cliente no projeto. Deve ter conhecimento e autonomia para decidir e validar os produtos produzidos pelo projeto, conhecer as necessidades do cliente e convocar outros usuários para ajudar na definição dos requisitos.
- **Time:** São as pessoas que realizam a construção do software. A equipe deve ser multidisciplinar e autogerenciada para gerar os produtos com alta

qualidade para o cliente. Além disso, são responsáveis por definir o prazo e as metas dos *sprints*.

1.5.1 Características do Scrum

A gestão ágil com SCRUM é rígida e exige disciplina na sua execução. As regras definidas em cada passo devem ser cumpridas para garantir os resultados propostos pelo método. A seguir são descritas as principais características do SCRUM, na ordem em que são utilizadas:

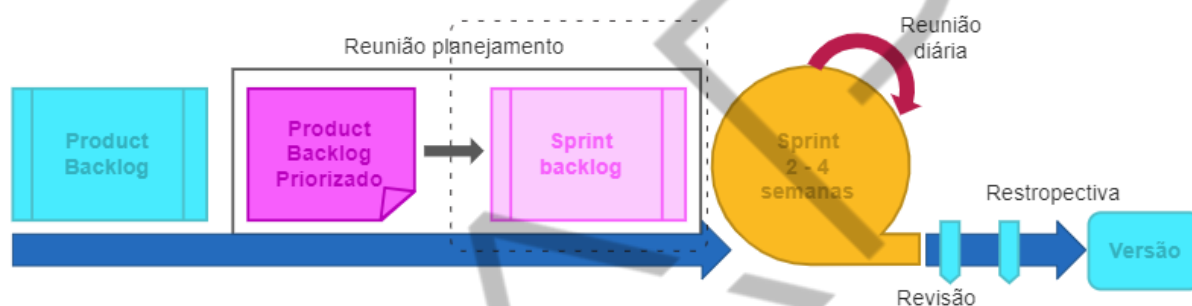


Figura 1.2 – Processo de gestão com SCRUM
Fonte: Elaborada pelo autor, adaptado por FIAP (2018)

a) Product backlog:

O product backlog é um documento elaborado pelo Product Owner que define o que deverá ser feito no projeto, ou seja, define o escopo inicial do projeto. Deve ser elaborado em formato de uma lista das necessidades funcionais e não funcionais que representem as necessidades do negócio, tais como: requisitos, documentação, diagramas, requisitos técnicos, erros, entre outros.

Não é um documento estático, sendo atualizado durante todo o processo de construção pelo Product Owner para incluir mudanças que podem ocorrer no projeto.

b) Reunião de planejamento:

É uma reunião que deve ser realizada antes de cada Sprint, com o objetivo de priorizar o que será construído, estimar a duração do Sprint e detalhar as atividades que precisam ser executadas para gerar o produto. Todos os membros do projeto participam.

c) Product backlog priorizado:

Gerado a partir do product backlog pelo Product Owner (PO), com a ajuda do Scrum Master, para definir a prioridade dos requisitos listados inicialmente. Essa lista priorizada é a base para o PO definir quais são as funcionalidades que farão parte dos sprints.

d) Sprint backlog:

A partir da lista priorizada de funcionalidades e da seleção daquelas que devem fazer parte do sprint, a equipe deve fazer uma análise do que foi solicitado junto ao PO para poder estimar o prazo do sprint e definir quais são as atividades necessárias para construir o projeto. É com base nessa estimativa que serão definidos os objetivos do sprint.

e) Sprint:

É a fase de implementação das funcionalidades previstas no sprint backlog. Tem duração predeterminada de 2 a 4 semanas e não pode ser estendido de forma alguma. Ao final, deve ter concluído todo o projeto com qualidade. Diferente dos processos tradicionais, o SCRUM faz entregas constantes de partes do projeto.

f) Reunião diária:

Diariamente, o Scrum Master, junto com o time, realiza uma reunião breve, no local de trabalho, com duração máxima de 15 minutos. O objetivo dessa reunião é acompanhar as atividades que estão sendo realizadas e verificar se existem problemas ou impedimentos que afetem a duração do sprint. De forma alguma deve ser uma reunião para busca de culpados ou de acusações, o foco é a colaboração de todos.

g) Reunião de revisão:

Após a conclusão de um Sprint, deve ser realizada uma reunião para apresentar o resultado do trabalho ao PO e demais envolvidos. O time é o responsável pela apresentação que deve ser feita com o produto funcionando, realizando suas funções. Ao final, os envolvidos podem fazer a sua avaliação e definir se o Sprint será aceito. Eventuais apontamentos de erros e melhorias devem ser inseridos no product backlog.

h) Reunião de retrospectiva:

É uma reunião de lições aprendidas do time, conduzida pelo Scrum Master, na qual são discutidos os pontos positivos e negativos durante a execução do sprint, com o objetivo de melhorar o desempenho dos próximos sprints. Eventualmente, o PO pode participar dessa reunião.

i) Versão:

Ao final do sprint, o time deve ter produzido uma parte do produto a ser entregue com alta qualidade, testado, completo e pronto. Embora essa entrega seja potencialmente uma versão, é o PO que define se será uma versão isolada ou aguarda a produção de novos sprints para formar uma versão.

O SCRUM é o framework ágil mais utilizado no mercado nos dias atuais. Seus conceitos e organização de práticas empíricas conquistaram o mercado e se tornaram referência em gestão ágil de projetos, sendo a base para preceitos ágeis preconizados pelo PMI (*Project Management Institute*) e adotado como referência para a certificação ACP (*Agile Certified Professional*) do próprio PMI.

1.6 Análise da abordagem tradicional e ágil

O ambiente de desenvolvimento de produtos e serviços em geral passa por um processo de mudança. Os projetos tornam-se cada vez mais complexos, sujeitos a constantes alterações e com prazos cada vez mais restritos para atender à demanda do mercado.



Figura 1.3 - Analogia a partes envolvidas
Fonte: Banco de imagens Shutterstock (2017)

O gerenciamento de projetos acompanha essa transformação do ambiente. Se de um lado, o método tradicional é centrado em um processo sequencial e dirigido à qualidade dos artefatos, do outro, o método ágil tem foco na rapidez de adequação às mudanças e conta com elementos como: cliente integrado à equipe e ciclos curtos de desenvolvimento (BOEHM; TURNER, 2002). Esses e outros aspectos trazem muitas discussões e controvérsias entre os especialistas em gestão.

O método de gerenciamento de projetos tradicional é criticado pela sua falta de flexibilidade em se adaptar a essa nova realidade, enquanto o gerenciamento ágil vem ganhando popularidade justamente por sua capacidade de adequação às situações voláteis do ambiente. Porém, também recebe críticas pelo grau de informalidade implícita em seu processo.

Ambas as correntes acreditam estar corretas em seus conceitos e na aplicação de seus métodos, apresentando vantagens e desvantagens para justificarem sua utilização.

Importante: os processos de gerenciamento ágeis envolvem uma série de quebras de paradigmas e equipe e cliente precisam ser treinados para melhor compreensão e efetividade.

O quadro a seguir apresenta um sumário das características do gerenciamento ágil.

Objetivo principal	Orientado por produto e centrado em pessoas.
Tipo de projeto	Projetos com mudanças constantes e que necessitam de respostas rápidas.
Tamanho	Mais efetivo em projetos pequenos (5-10 pessoas).
Gerente de projeto	Papel de facilitador ou coordenador.
Equipe do projeto	Atuação colaborativa em todas as atividades do projeto.

Cliente	É essencial. Deve ser parte integrante da equipe do projeto.
Planejamento	Curto e com a participação de todos os envolvidos na elaboração do planejamento.
Arquitetura	Aplicação de <i>design</i> simples. Evolui junto com o projeto.
Modelo de desenvolvimento	Iterativo e incremental.
Comunicação	Informal.
Controle de mudanças	Dinâmico e com rapidez de incorporação nas iterações.

Quadro 1.1 – Características do gerenciamento ágil
Fonte: Adaptado de Nerur, Mahapatra e Mangalara (2005)

1.7 Análise comparativa entre o gerenciamento tradicional e o gerenciamento ágil

O PMBOK® (2012) é um guia de melhores práticas de gerenciamento de projetos que aplicados adequadamente permitem a realização de um projeto com sucesso. Os processos de gerenciamento de projetos são divididos em nove áreas de conhecimento que organizam a aplicação das técnicas e ferramentas necessárias à realização de um projeto. Essas áreas são: escopo, prazo, custo, qualidade, riscos, comunicação, recursos humanos, aquisição e integração.

Os métodos de gerenciamento tradicional e o gerenciamento ágil de projetos, independentemente de suas características específicas, podem ser suportados pelos conceitos apresentados no PMBOK® (2012) e suas especificidades, em relação a cada área do conhecimento, são apresentadas no quadro a seguir.

Área do processo	Gerenciamento Tradicional	Gerenciamento Ágil
Escopo	Bem definido nas fases iniciais do projeto e formalizado por meio	Escopo é definido em alto nível e os requisitos são priorizados e

	da EAP (Estrutura Analítica de Projeto).	definidos de forma iterativa. Necessita de maior controle de <i>gold plating</i> .
Tempo	Cronograma detalhado para a realização de todo o projeto.	Uso de controles como o quadro Kanban, com entregas incrementais de 2-4 semanas.
Custo	Monitoração das alterações para que não afete o custo planejado.	Necessita de maior controle em função da rapidez na incorporação de alterações.
Qualidade	Processos de Verificação e Validação e plano de testes.	Programação em pares, testes incrementais.
Riscos	Análise de riscos durante todo o ciclo de vida do projeto.	Gestão de impedimentos no projeto.
Comunicação	Documentado e formal.	Implícita, interpessoal e colaborativa.
Recursos Humanos	Papéis claros e bem definidos.	Confiança nos membros da equipe e ambiente colaborativo.
Aquisição	Controle por contrato e escopo bem definido e documentado.	Presença do cliente, volatilidade de requisitos e pouca documentação tornam o processo um desafio.
Integração	Plano do projeto detalhado e controle total do projeto pelo gerente.	Plano do projeto evolutivo. Gerente do projeto atua como facilitador.
Partes Envolvidas	Processo formal, controlado e atuação pontual.	Fazem parte do projeto com interação constante no projeto.

Quadro 1.2 – Comparação entre o método Tradicional e o método Ágil

Fonte: Elaborado pelo autor

Importante: *Gold plating* é um termo que define a aceitação de solicitações do cliente sem análise ou processo formal, incorporadas ao projeto, sem custos e sem prazo adicional.

1.8 Diferenças e Similaridades entre as abordagens

Vamos discutir agora as diferenças e semelhanças entre as abordagens do gerenciamento tradicional e do gerenciamento ágil dentro de cada área do conhecimento do PMBOK (2012) para que você possa escolher conscientemente o método que melhor atenda às necessidades do projeto que precisa ser gerido

considerando cenários, cultura organizacional e resultados esperados entre outros fatores determinantes.

1.8.1 Gestão do escopo

No gerenciamento tradicional existe uma preocupação em definir o escopo detalhado no início do projeto, formalizando-o com a EAP, por meio da qual são definidos os produtos a serem entregues ao cliente. Toda a informação levantada é documentada na especificação de requisitos que servirá de base para a gestão de mudanças durante o andamento do projeto.

No gerenciamento ágil a preocupação está em definir o escopo em um alto nível para permitir o entendimento do trabalho. Após a definição do escopo, os requisitos são priorizados e definidos com a participação de toda a equipe do projeto, inclusive o cliente, que discute e define as funcionalidades durante cada iteração do ciclo de desenvolvimento.

A possibilidade de existir *gold plating* aumenta em função da documentação mínima que é elaborada e, principalmente, pela interação próxima entre o cliente e a equipe. Para reduzir esse impacto, deve-se aumentar a monitoração sobre os novos requisitos para permitir o controle efetivo do escopo, sem afetar a flexibilidade de incorporação de alterações, característica dos métodos ágeis.

1.8.2 Gestão do tempo

Os processos de definição e estimativa de esforço e duração das atividades são comuns aos dois métodos, que irão diferir na forma de construir o cronograma.

A elaboração de um cronograma detalhado de todas as atividades para a execução do projeto é uma característica do método tradicional. Porém, em função do tempo disponível ou das incertezas que envolvem o projeto, torna-se uma projeção, sujeito a constantes alterações que podem levar à perda do controle do prazo ou insegurança por parte dos envolvidos.

No método ágil, o cronograma é orientado ao produto que será construído em cada iteração, que é planejada de acordo com a prioridade funcional definida pelo

cliente. Essas iterações devem ter duração entre duas e quatro semanas para atender rapidamente às necessidades do cliente. Nessa situação, o prazo final não está claramente definido. No entanto, como o cliente recebe produtos constantes de acordo com sua própria orientação, há uma redução dos conflitos pela cumplicidade no processo.

1.8.3 Gestão de custos

Para o gerenciamento tradicional as alterações são críticas e afetam todo o projeto. Por isso, existe uma ênfase em controlar, monitorar e documentar essas mudanças de forma a obter as aprovações formais do cliente antes de serem incorporadas ao projeto. O foco está em controlar os custos contra o planejamento inicial.

Para o gerenciamento ágil, as alterações são parte do projeto e são incorporadas dentro da iteração mais apropriada e de comum acordo com o cliente. Esse modelo favorece a flexibilidade em atender ao cliente, porém o custo final pode sofrer grandes variações se essas mudanças não forem documentadas e devidamente repassadas ao patrocinador do projeto para recomposição do custo inicial planejado.

1.8.4 Gestão da qualidade

Para ambos os métodos, a qualidade é essencial e os padrões a serem seguidos devem ser estabelecidos no início do projeto. A diferença entre eles está na forma de garantir e controlar a qualidade.

O método tradicional foca na elaboração de planos de testes a partir das especificações de requisitos e nos processos de validação e verificação durante a construção do produto.

O método ágil concentra-se na programação em pares, nos testes incrementais elaborados antes e durante o desenvolvimento do produto.

1.8.5 Gestão de riscos

A identificação, análise e respostas aos riscos do projeto são comuns às duas formas de gerenciamento de projetos.

Os riscos e as incertezas estão presentes em todo tipo de projeto e devem ser gerenciados para evitar que afetem o resultado final do projeto. A monitoração e o controle dos riscos devem ocorrer durante todo o processo de desenvolvimento do produto verificando sua probabilidade e seus impactos sobre o projeto (PRESSMAN, 2006).

Nos métodos ágeis existe a atividade de tratamento e remoção de impedimentos que, de certa forma, pode ser comparado com a análise de riscos. O objetivo dessa remoção de impedimento é evitar que a equipe pare ou que fatores externos influam nas atividades de desenvolvimento do software.

1.8.6 Gestão da comunicação

A comunicação no gerenciamento tradicional é formal e documentada, com fluxos bem definidos de aprovação, acompanhamento e divulgação dos resultados do trabalho. O objetivo é documentar todos os fatos para evitar conflitos entre os envolvidos do projeto.

O método ágil traz uma significativa melhoria nas comunicações e na interação entre todos os envolvidos, promovendo *feedback* constante e transparência durante o desenvolvimento do software (MAURER; MELNIK, 2005). A comunicação é colaborativa e direta entre todos os membros da equipe, proporcionada pela proximidade entre todos os envolvidos do projeto. Esse processo informal exige um certo grau de maturidade por parte da organização, do cliente e da equipe, pois podem ser fontes de conflitos entre as partes.

Independentemente do método aplicado, as decisões e assuntos críticos que ocorrem durante um projeto devem ser documentados e publicados a todos os envolvidos.

1.8.7 Gestão de recursos humanos

No gerenciamento tradicional a definição clara de papéis e responsabilidade dos membros da equipe é uma premissa para a condução adequada do projeto, pois cada pessoa é treinada e guiada pelo processo na execução de suas tarefas.

A confiança e a colaboração na equipe são os atributos do gerenciamento ágil. Todos os participantes do projeto executam suas tarefas, planejam e tomam decisões em conjunto, compartilhando suas experiências. Essa atuação diferenciada exige a participação de profissionais mais experientes e qualificados, embora não necessite que toda a equipe tenha o mesmo nível de conhecimento e experiência.

As premiações e as comemorações pela realização de um projeto são fatores motivacionais comuns aos métodos tradicional e ágil.

1.8.8 Gestão da aquisição

Para a execução de um processo de aquisição de software, é necessário que a descrição do produto a ser adquirido seja detalhada e permita aos fornecedores avaliarem adequadamente o prazo e o esforço para a realização do trabalho.

No gerenciamento tradicional, o processo de aquisição ocorre a partir do escopo bem definido e da documentação detalhada produzida, permitindo o controle e acompanhamento das atividades e do fornecedor durante a construção do software.

Embora existam outras formas de contrato, a contratação por preço fixo é a mais comum. Nesse caso, quando utilizado o gerenciamento ágil, o processo de aquisição deve ser evitado em função da volatilidade dos requisitos, a documentação mínima que é produzida e da presença do cliente como membro da equipe.

1.8.9 Gestão da integração

O Plano do projeto é formal e detalhado no início do projeto para o gerenciamento tradicional, enquanto para o gerenciamento ágil deve ser menos detalhado e sofre mais alterações durante a execução do projeto.

O controle no gerenciamento tradicional tem alto grau de formalização e todas as ocorrências são documentadas como informações históricas. Para o método ágil, a atuação colaborativa da equipe com o cliente favorece um maior grau de informalidade e o conhecimento implícito é privilegiado.

Outra diferença entre os métodos na integração é o papel do gerente de projetos que no método tradicional tem o controle total do projeto, enquanto no método ágil sua atuação é voltada para o papel de facilitador ou coordenador das atividades.

1.8.10 Gestão das partes envolvidas

No gerenciamento tradicional, o gerenciamento das partes envolvidas é formal e o seu envolvimento se faz em partes específicas do projeto como no levantamento de requisitos, validações, reuniões de posicionamento e no aceite dos produtos.

Na gestão ágil de projetos, a participação do cliente é colaborativa e constante no projeto. O preceito básico é o cliente como parte da equipe do projeto.

Embora a gestão tradicional busque aproximar os envolvidos do projeto, esse aspecto é uma das maiores diferenças entre as duas formas de gerenciamento.

Nessa análise comparativa entre as abordagens de gerenciamento de projetos tradicional e ágil, a partir das dez áreas de conhecimento do PMBOK® (2012) foi descrito o modo como essas são tratadas nos dois métodos de gerenciamento.

O método tradicional está consolidado no mercado e os resultados de sua aplicação são evidentes nos diversos ambientes de projeto. O método ágil apresenta-se como uma opção às organizações que buscam alternativas de melhoria em seu processo de desenvolvimento, embora a literatura e as evidências empíricas dos seus resultados sejam reduzidas se comparadas ao método tradicional (NERUR; MAHAPATRA; MANGALARA, 2005).

Para as organizações conservadoras, existe ainda a possibilidade de mesclar as características de ambos os métodos, permitindo uma avaliação gradativa dos pontos fortes e das principais restrições das duas abordagens.

Os atributos comparativos apresentados, associados à análise do ambiente do projeto, visam estabelecer parâmetros que auxiliem na definição de qual a melhor

estratégia a ser aplicada no gerenciamento de projeto de software e permita às organizações atender as necessidades e expectativas do cliente. (RIBEIRO; ARAKAKI; 2006).

1.9 Design Sprint

Você já percebeu que, com frequência, somos obrigados a fornecer soluções rápidas para problemas não são tão claros assim? Outras vezes, encontramos uma solução para o problema, mas não temos como validá-la antes de ter o produto propriamente dito? Geralmente, isso acontece por falta tempo ou de orçamento, ou até porque não há, ainda, uma cultura de *Design Thinking* tão estabelecida. Mas, mesmo quando essa cultura *Design Thinking* já está mais madura na organização, ainda sim, alcançar resultados é um processo lento, pois depende de um trabalho conjunto de diferentes pessoas e perfis, o que gera gargalo ao processo. Foi buscando melhorar essa equação – tempo *versus* processo de *design thinking* – que a Google Ventures criou o *Design Sprint*.

A Google Ventures é um braço do Google criado em 2009, que atua de forma independente e procura acelerar empresas que possam ter futuro em diversas áreas, como em Internet, software, hardware, biotecnologia e cuidados de saúde. Para você ter ideia, a Google Ventures já investiu em cerca de 300 empresas de diversos tipos de atuações, entre elas, uma das *startups* que mais tem feito barulho no mercado mundial nos últimos tempos – a Uber.

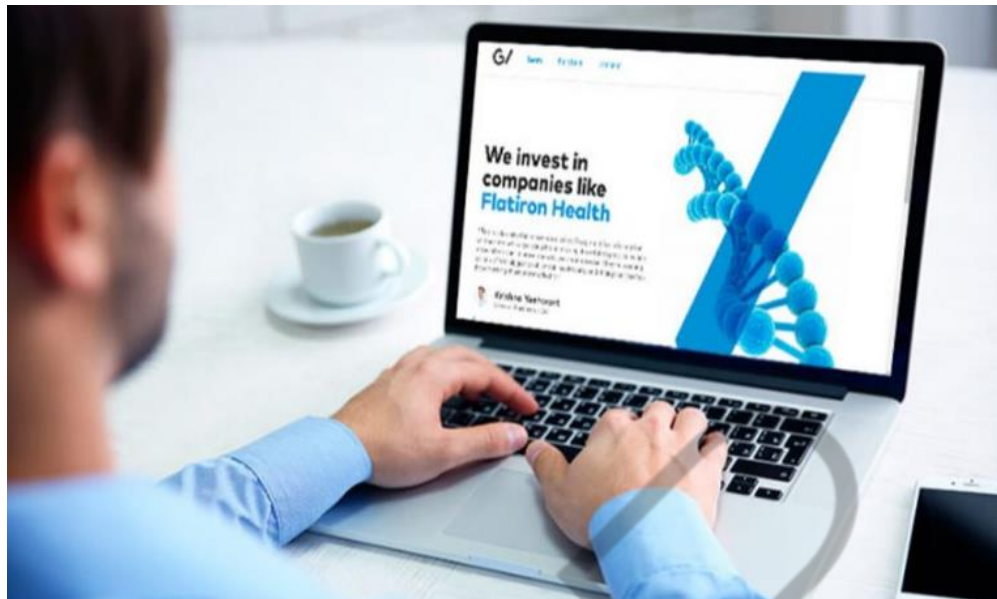


Figura 4.4 - Site do Google Ventures
Fonte: Banco de imagens Shutterstock (2017)

1.9.1 O que é *design sprint*, afinal?

O *Design Sprint* é uma metodologia centrada no usuário, iterativa, prática e colaborativa, resultado de diversas experiências da Google Ventures, formatadas em um guia prático de apenas cinco dias. Baseia-se em *design thinking* e metodologias ágeis para que as equipes possam criar e prototipar soluções de forma rápida e eficaz. Em vez de gastar horas e horas de desenvolvimento para lançar um produto, e só depois conseguir entender se a ideia é boa ou não, o *Design Sprint* oferece um atalho de aprendizado, no qual é possível elaborar e testar praticamente qualquer ideia em apenas 40 horas, sem precisar construir e lançar o produto propriamente dito. Se você parar para pensar, inclusive, esse método concorda com uma das premissas do *Lean Startup*: testar hipóteses rapidamente e acelerar o aprendizado.

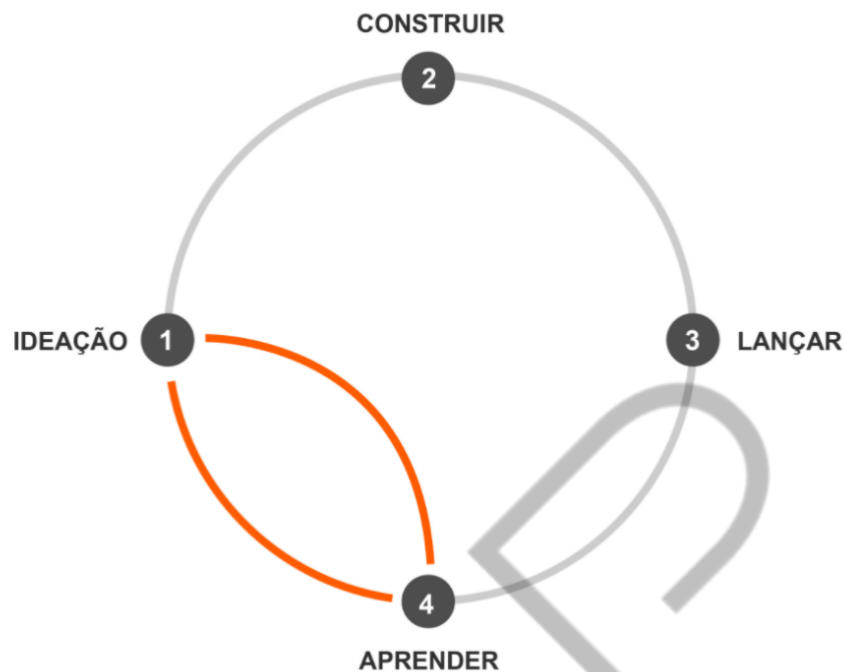


Figura 1.5 - O sprint dá às equipes um atalho para aprender sem construir e lançar
 Fonte: Google Ventures (2016b), adaptado por FIAP (2018)

Mesmo sendo um processo de curta duração, de cinco dias, o *Design Sprint* é formado por etapas que, ora ampliam a sua visão do problema, ora focam em buscar uma solução específica. Em uma visão geral de cada etapa, o que acontece é o seguinte:



Figura 1.6- Etapa do *Design Sprint*
 Fonte: Hoyer (2015), adaptado por FIAP (2018)

Tendo essa visão geral das etapas, vamos, agora, entender como cada uma funciona e qual o *mindset* que é necessário em cada etapa.

1.9.2 Etapas

1.9.2.1 Entender

Na primeira etapa, o foco é buscarmos um melhor entendimento do desafio proposto pela equipe como um todo. Nesta etapa é muito importante que as pessoas compartilhem o seu conhecimento sobre aquele desafio. Isso porque a experiência e o conhecimento do time são sempre assimétricos, por exemplo: a área de vendas tem informações do negócio que a área de desenvolvimento não imagina; o suporte ao cliente conhece processos específicos que o designer da equipe desconhece e assim por diante.

1.9.2.2 Divergir

Busca-se explorar o maior número de ideias, essa é a fase em que tudo é possível. A princípio, parece um processo de *brainstorming*, mas, nesta etapa, cada indivíduo do time vai trabalhando sozinho para desenhar soluções detalhadas no papel e não em grupo, como é um típico *brainstorming*. Isso ocorre assim, pois, como Jake Knapp (criador e evangelista do *Design Sprint*) percebeu, as melhores ideias tendem a vir de indivíduos e não de grupos. O problema é que grupos têm dinâmicas nas quais não necessariamente as melhores ideias sobrevivem, mas as ideias que são vendidas e contadas com mais estratégia e empolgação por alguém do grupo.

Você, certamente, já participou de trabalhos em grupo e, de alguma forma, percebeu que o indivíduo mais articulado tem vantagem em relação a um indivíduo mais introspectivo em situações de dinâmicas de grupo. O que o Sprint tenta fazer, nessa etapa do processo, é eliminar essa vantagem para que todas as ideias tenham o mesmo peso.

1.9.2.3 Decidir

Na terceira etapa, teremos mais de uma dúzia de soluções para escolher junto com o seu time. É o momento das decisões difíceis, em que as dinâmicas de grupo

também atrapalham a tomada de decisão. O que se passa em um grupo é que, normalmente, as decisões são tomadas por consenso e, na maioria das vezes, o consenso escolhe o caminho mais fácil de aceitar e que gera menos atrito entre o grupo, isso faz com que as ideias mais corajosas ou mais íntegras acabem ficando de lado. Novamente, o *Sprint* tenta eliminar isso, usando uma técnica de votação silenciosa em que ninguém tenta convencer ninguém, chamada *dot voting* (ou *zen-voting*).

1.9.2.4 Prototipar

Nesta fase, o objetivo é tangibilizar a solução criada pelo time. Aqui, é importante ter, ao fim do dia de trabalho, um protótipo de média/alta fidelidade do conceito a ser validado. É importante que o protótipo seja de média/alta fidelidade, pois, como ele será testado com usuários reais, um protótipo de baixa fidelidade nos daria apenas feedbacks e não extrairia reações francas dos seus clientes. Como temos apenas um dia para prototipar, é importante que sejamos muito produtivos e, de preferência, que trabalhemos com ferramentas de prototipagem com as quais nosso time já esteja habituado.

1.9.2.5 Validar

Depois de todo o trabalho nas etapas anteriores, a questão mais importante que ainda nos resta é: “Como podemos saber se nossa ideia é realmente boa?”.

É nesta fase que recebemos a resposta para essa pergunta, já que vamos mostrar protótipos dos conceitos para os potenciais usuários do produto. Nesse momento, o usuário vai interagir com sua solução e dar um feedback real sobre a experiência que ele está tendo. Óbvio que, quando uma ideia arriscada tem êxito nesta etapa, é uma recompensa fantástica para equipe de trabalho, pois gera motivação instantânea. Mas são, na verdade, os fracassos nesse processo que, mesmo dolorosos, fornecem o maior investimento, pois, afinal de contas, a equipe soube que teria que buscar outro caminho ou desistir da ideia em apenas cinco dias de trabalho.

Neste capítulo você viu que para soluções rápidas o Design Sprint nos ajuda a validar ideias em apenas 5 dias! Você teve acesso a métodos baseados no manifesto ágil, enquanto o EXP é focado no universo dos softwares, o Scrum pode ser levado de forma mais ampla, no gerenciamento de qualquer projeto.

Sendo assim, vamos entender a fundo como funciona o framework do Scrum?

No próximo capítulo você vai descobrir porque o Scrum se tornou tão popular e conquistou o mercado.

EXP

REFERÊNCIAS

ABRAHAMSSON, P. et al. New Directions on Agile Methods: A Comparative Analysis. In: **International conference on software engineering**. 2003, Portland. Portland: IEEE, 2003.

BECK, K. **Extreme Programming Explained: Embrace Change**. Boston: Addison Wesley, 1999.

BOEHM, B. W.; TURNER, R. **Balancing agility and discipline: a guide for the perplexed**. Boston: Addison Wesley, 2002.

KERZNER, H. **Gerenciamento de projetos: uma abordagem sistêmica para planejamento, programação e controle**. São Paulo: Edgard Blucher, 2011.

_____. **Gestão de projetos: as melhores práticas**. Porto Alegre: Bookman, 2002.

MAURER, F.; MELNIK, G. What you always wanted to know about agile methods but did not dare to ask. In: **International conference on software engineering**. 2005, St. Louis. St. Louis: IEEE, 2005.

NERUR, S.; MAHAPATRA, R.; MANGALARA, G. **Challenges of Migrating to Agile Methodologies. Communications of the ACM**. v. 48, n. 5, mai. 2005. Disponível em: <<http://faculty.salisbury.edu/~xswang/Research/Papers/SERelated/Agile/p72-nerur.pdf>>. Acesso em: 28 abr. 2015.

OCHNER, J. **Gerência de projetos: uma comparação entre o PMBOK e o XPM**. 2006. 49 f. Monografia de graduação (Bacharelado em Ciência da Computação) – Departamento de Ciência da Computação, Universidade Federal de Lavras, Lavras, 2006. Disponível em: <http://www.bcc.ufla.br/wp-content/uploads/2013/2005/Gerencia_de_projetos_uma_comparacao_entre_o_PMBOK_e_XPM.pdf>. Acesso em: 29 abr. 2015.

PRESSMAN, R. S. **Engenharia de software**. 6. ed. Rio de Janeiro: McGraw-Hill, 2006.

PROJECT MANAGEMENT INSTITUTE (PMI). **Um Guia do Conhecimento em Gerenciamento de Projetos (Guia PMBOK®)**. 5. ed. São Paulo: Saraiva, 2013.

_____. **Sobre o PMI**. Disponível em: <<https://brasil.pmi.org/brazil/AboutUS.aspx>>. Acesso em: 17 out. 2018.

RIBEIRO, A. L. D; ARAKAKI, R. **Gerenciamento de projetos tradicional x gerenciamento de projetos ágil: uma análise comparativa**. In: Congresso internacional de gestão da tecnologia e sistemas de informação, 3, 2006, São Paulo. São Paulo: USP, 2006. p. 1594. Disponível em: <<http://www.tecsi.fea.usp.br/envio/contecsi/index.php/contecsi/3contecsi/paper/viewFile/2014/1128>>. Acesso em: 29 abr. 2015.

SCRUM. **The home of Scrum**. [s.d.]. Disponível em: <<http://www.scrum.org>>. Acesso em: 17 out. 2018.

SCRUM ALLIANCE. **About Scrum Alliance**. [s.d.]. Disponível em: <<https://www.scrumalliance.org>>. Acesso em: 17 out. 2018.

EMENDAS