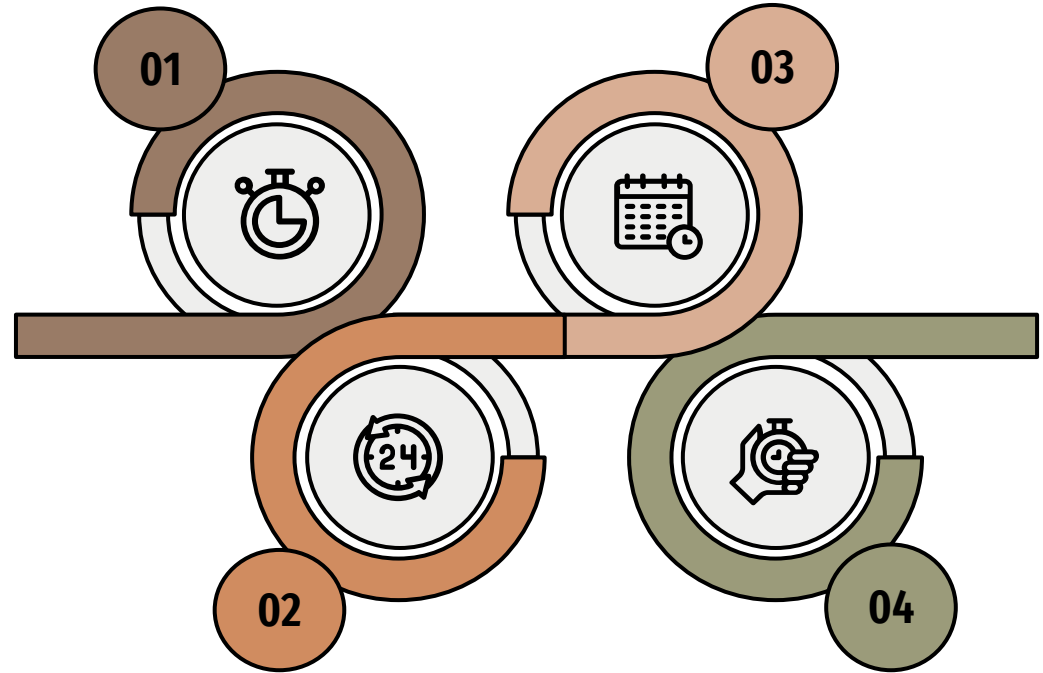


Business Agility

Gestão Ágil

Unidade 02
Prof. Daniel Caixeta



4. Os pilares da Gestão Ágil/*Business Agility*

4.1. Os pilares [...].

5. Metodologias ágeis

5.1. Metodologias ágeis & Gestão ágil.

5.2. *SCRUM*.

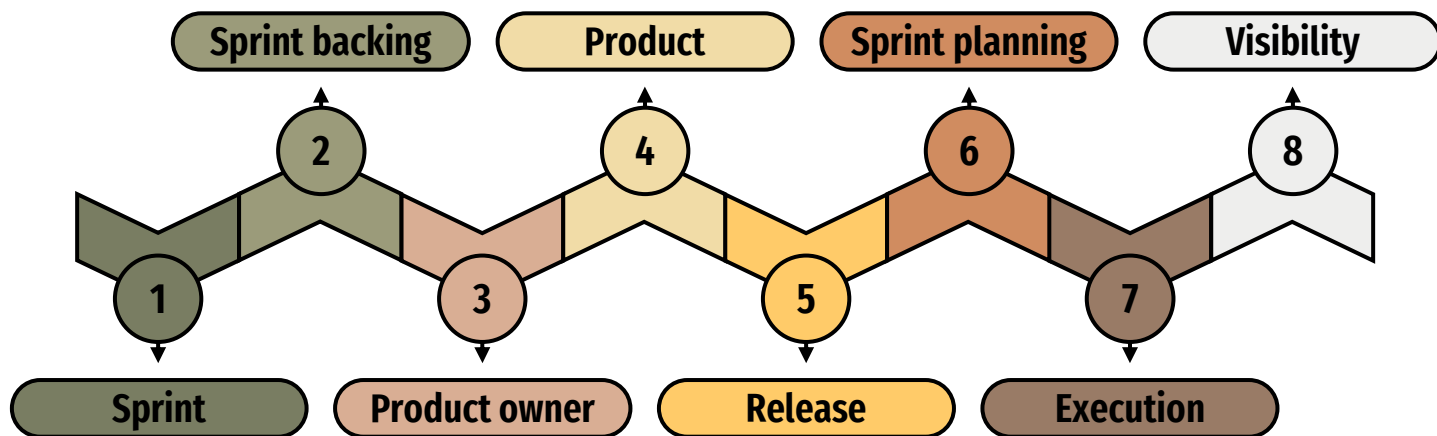
5.3. *Extreme Programming* (XP).

5.3.1. Os processos XP.

5.4. *Kanban*.

5.5. *Lean Startup*.

4. OS PILARES DA GESTÃO ÁGIL/BUSINESS AGILITY

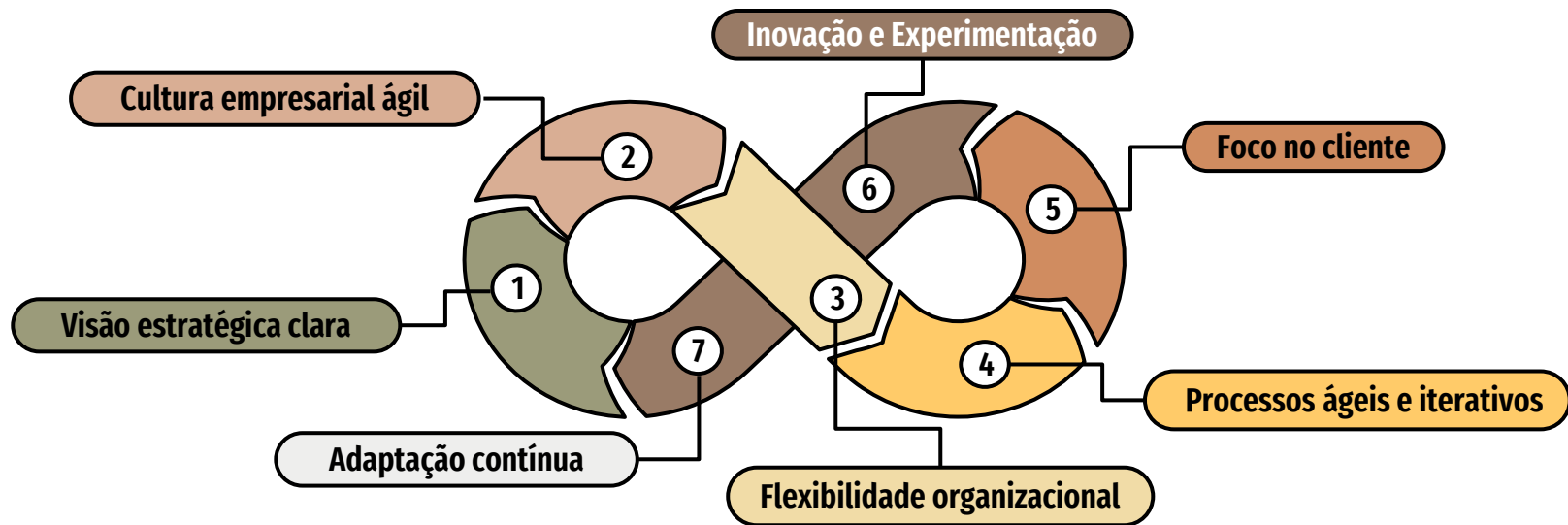


4.1. OS PILARES

- No atual cenário empresarial, marcado pela rápida evolução tecnológica e mudanças imprevisíveis nos mercados, a Gestão Ágil emerge como uma abordagem essencial. Ela capacita as empresas a prosperarem em ambientes dinâmicos e complexos, mantendo-se ágeis, inovadoras e resilientes.
- Os pilares da Gestão Ágil/*Business Agility* são os fundamentos sobre os quais essa abordagem é construída, fornecendo a base para a transformação cultural, de processos e operações das organizações.
- Representam princípios orientadores que ajudam as empresas a enfrentar as incertezas do mercado, possibilitando antecipar mudanças, adaptar-se rapidamente e entregar valor consistente aos clientes.



- Ao entender e adotar esses pilares, as empresas passam a desenvolver uma mentalidade organizacional mais ágil, promover uma cultura de inovação e colaboração, e implementar práticas e processos que permitam uma resposta eficaz às mudanças do mercado.
- É fundamentada em alguns pilares que fornecem a estrutura necessária que as empresas necessitam. Apresentemos alguns deles:



1

Visão estratégica clara

- Uma visão estratégica clara e compartilhada é essencial para orientar as decisões e ações da empresa. Isso inclui definir metas e objetivos claros, identificar as necessidades dos clientes e as tendências do mercado e estabelecer uma direção clara para o futuro.

2

Cultura empresarial ágil

- Uma cultura organizacional que valoriza a experimentação, a inovação e a aprendizagem contínua é fundamental para a Gestão Ágil. Isso envolve promover uma mentalidade de curiosidade, tolerância ao risco e colaboração entre equipes e departamentos.

3

Flexibilidade organizacional

- As empresas ágeis são caracterizadas por estruturas organizacionais flexíveis e adaptáveis. Isso inclui a capacidade de formar equipes multifuncionais e auto-organizadas, implementar processos de tomada de decisão descentralizados e responder rapidamente às mudanças nas condições do mercado.

4

Processos ágeis e iterativos

- A adoção de metodologias ágeis, como *Scrum*, *Kanban* e *Lean*, é fundamental para a *Business Agility*. Esses métodos promovem ciclos de desenvolvimento curtos e incrementais, *feedback* contínuo dos clientes e uma abordagem iterativa para a entrega de produtos e serviços.

5

Foco no cliente

- Colocar o cliente no centro de todas as decisões e atividades é essencial para o sucesso da Gestão Ágil. Isso envolve entender as necessidades e expectativas, coletar informações regularmente e ajustar as estratégias e práticas da empresa com base em suas preferências.

6

Inovação e Experimentação

- A capacidade de inovar e experimentar novas ideias e abordagens é crucial. Isso inclui promover uma cultura de inovação, investir em pesquisa e desenvolvimento e estar aberto a tentar coisas novas e aprender com os resultados.

7

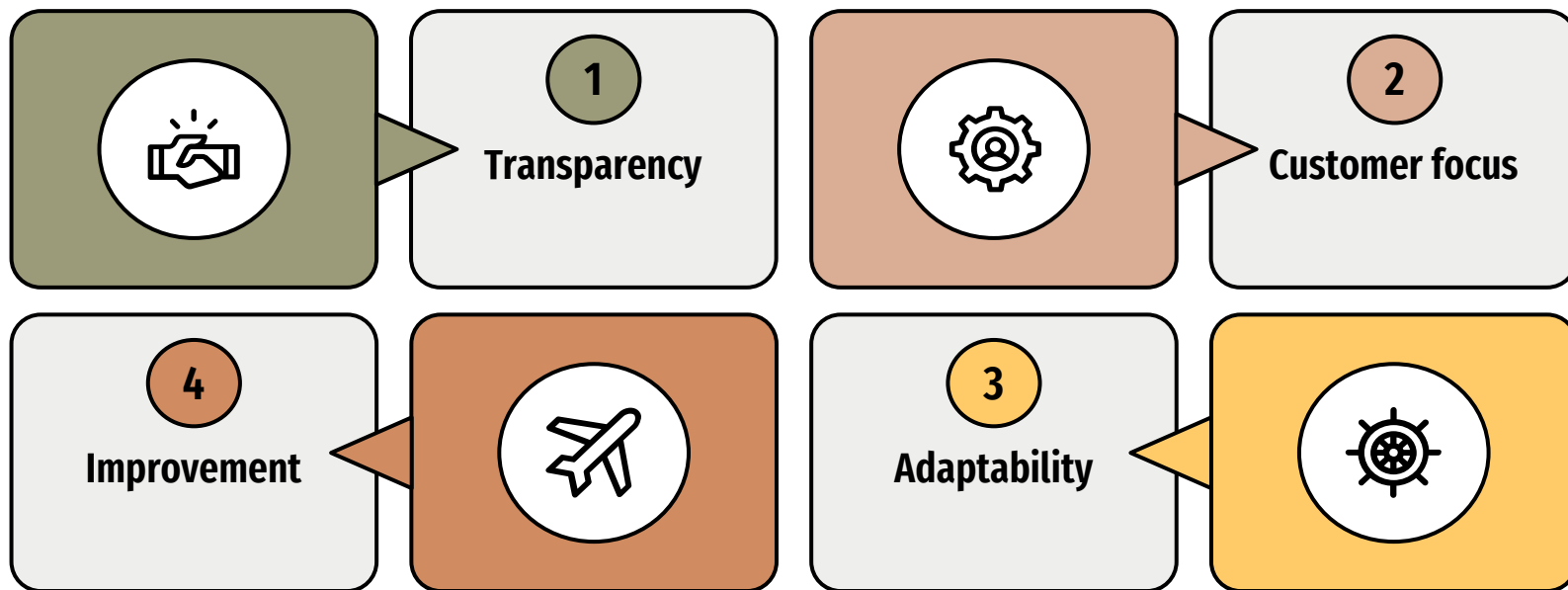
Adaptação contínua

- As empresas ágeis estão sempre em estado de adaptação e melhoria contínuas. Isso envolve monitorar constantemente o ambiente de negócios, identificar oportunidades e ameaças emergentes e ajustar as estratégias e táticas da empresa conforme necessário.

Resumindo, esses são apenas alguns dos principais pilares da Gestão Ágil/*Business Agility*.

Ao adotar esses princípios e práticas, as empresas podem se tornar mais ágeis, adaptáveis e inovadoras, preparando-se para enfrentar os desafios e aproveitar as oportunidades que o futuro reserva.

5. METODOLOGIAS ÁGEIS & GESTÃO ÁGIL



5.1. BREVE INTRODUÇÃO

- As Metodologias Ágeis e a Gestão Ágil estão intrinsecamente interligadas, uma vez que ambas buscam promover a agilidade, flexibilidade e capacidade de resposta rápida em contextos marcados por mudanças constantes.
- Vamos explorar como esses dois conceitos se relacionam:

1 Metodologias ágeis

- Metodologias como *Scrum*, *Kanban* e *Extreme Programming* (XP), são abordagens específicas para desenvolvimento de software e gestão de projetos que enfatizam entregas rápidas.
- Essas metodologias são caracterizadas por processos iterativos e incrementais, colaboração entre equipes multifuncionais, priorização flexível e capacidade de resposta às mudanças nos requisitos do projeto.

2 Gestão ágil

- É uma abordagem mais ampla que se estende além do desenvolvimento de software e se aplica a toda a organização.
- Envolve a implementação de práticas ágeis em todas as áreas da empresa, desde *marketing* e vendas até operações e suporte ao cliente.

- A relação entre as duas está no fato de que as metodologias ágeis fornecem um conjunto de princípios e práticas que podem ser aplicados em toda a organização para promover a agilidade e a capacidade de resposta rápida. Por exemplo, os princípios do *Scrum*, como ciclos de desenvolvimento curtos e *feedback* contínuo, podem ser adaptados para além do desenvolvimento de *software* e aplicados em outras áreas, como *marketing* ou gerenciamento de projetos.
- Além disso, ajudam a promover uma cultura de experimentação e aprendizado, que é fundamental para a gestão ágil.
- Ao permitir iteração rápida e *feedback* contínuo, as empresas podem aprender com seus sucessos e fracassos e ajustar suas estratégias e práticas de acordo.
- Apresentamos algumas das metodologias ágeis mais populares.

5.2. SCRUM

- O *Scrum* é uma das metodologias ágeis mais populares e amplamente utilizadas no desenvolvimento de *software* e gerenciamento de projetos.
- Criado por Jeff Sutherland e Ken Schwaber na década de 1990, se baseia em princípios de transparência, inspeção e adaptação para promover a entrega rápida e iterativa de valor aos clientes.
- Aqui estão alguns dos principais conceitos e práticas do Scrum.



Princípios do *Scrum*.

1

Time Scrum

- É composto por um grupo de profissionais multifuncionais que trabalham juntos para entregar um produto ou projeto. O time é auto-organizado e responsável por planejar, executar e entregar o trabalho.



2

Product Owner (P.O)

- É o responsável por representar os interesses dos *stakeholders* e garantir que o produto atenda às necessidades e expectativas dos clientes. É responsável por definir e priorizar o *backlog* do produto, comunicar as necessidades do cliente para o *Time Scrum* e tomar decisões sobre o produto.



3

Scrum Master

- É o responsável por facilitar o processo Scrum, remover obstáculos que possam impedir o progresso do time e garantir que os princípios e práticas sejam seguidos corretamente. Atua como um mentor e facilitador.



4

Backlog do Produto

- É uma lista de todas as funcionalidades, requisitos e melhorias desejadas para o produto. É de responsabilidade do P.O priorizar e manter o backlog do produto atualizado.



5

Sprint

- É um período de tempo fixo, geralmente de duas a quatro semanas, durante o qual o *Time Scrum* trabalha para entregar um conjunto de funcionalidades ou incrementos de produto. No início de cada *Sprint*, o time seleciona um conjunto de itens do *backlog* do produto para trabalhar durante esse período fixo.



6

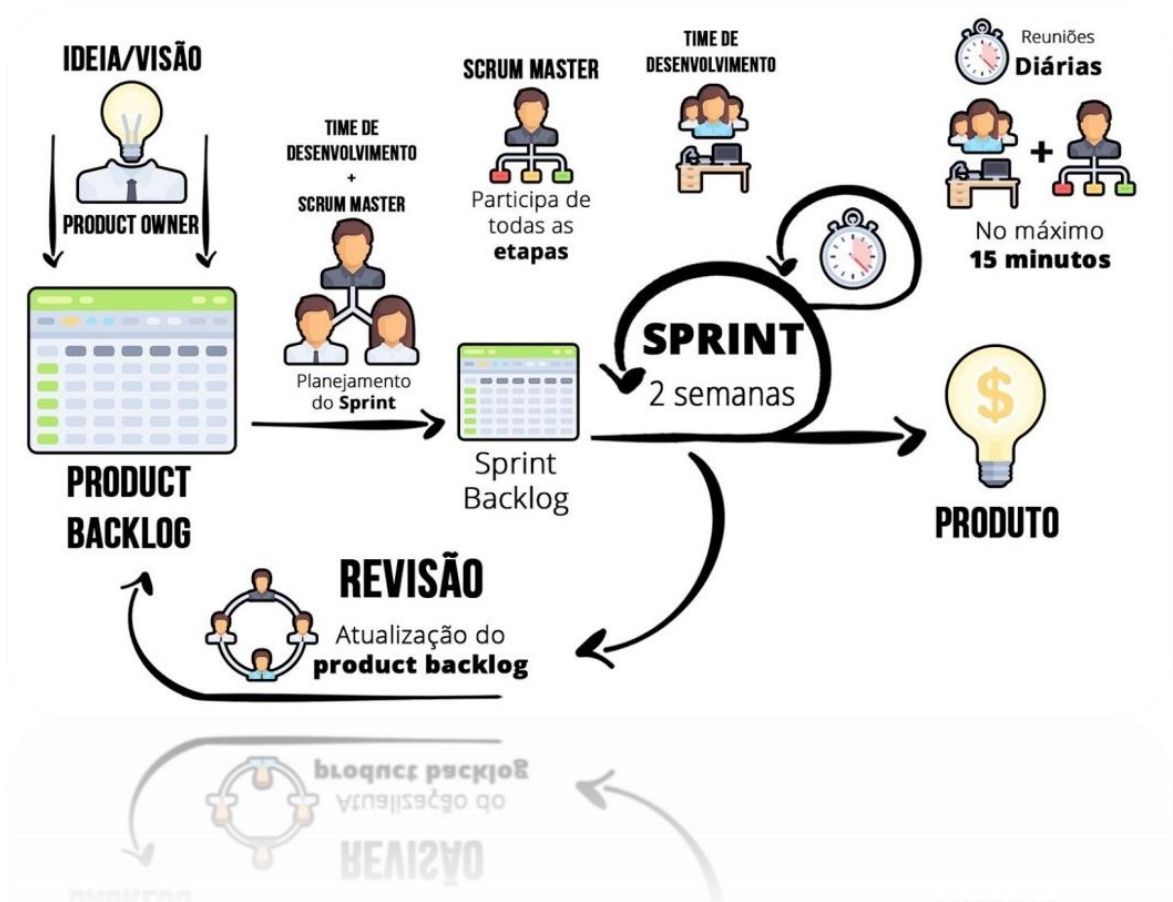
Reuniões Scrum

- O *Scrum* possui várias reuniões, incluindo a *Sprint Planning* (planejamento do *Sprint*), *Daily Scrum* (reunião diária de no máximo 15 minutos), *Sprint Review* (revisão do *Sprint*) e *Sprint Retrospective* (retrospectiva do *Sprint*). Essas reuniões ajudam a garantir a transparência, inspeção e adaptação necessárias para o sucesso do projeto.



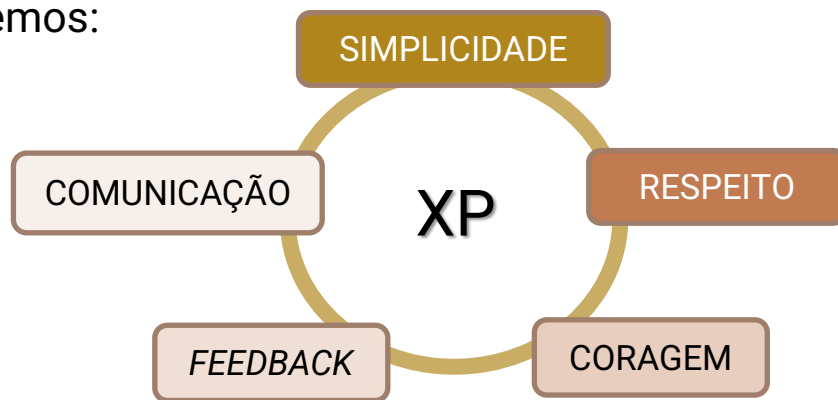
- A metodologia *Scrum* é conhecida por sua simplicidade e flexibilidade, permitindo que equipes de diferentes tamanhos e contextos apliquem seus princípios e práticas para entregar valor aos clientes de forma mais rápida e eficaz.
- Ao promover a colaboração, transparência e responsabilidade, o *Scrum* ajuda as organizações a se tornarem mais ágeis e adaptáveis em mercados cada vez mais competitivos.

Processo Scrum

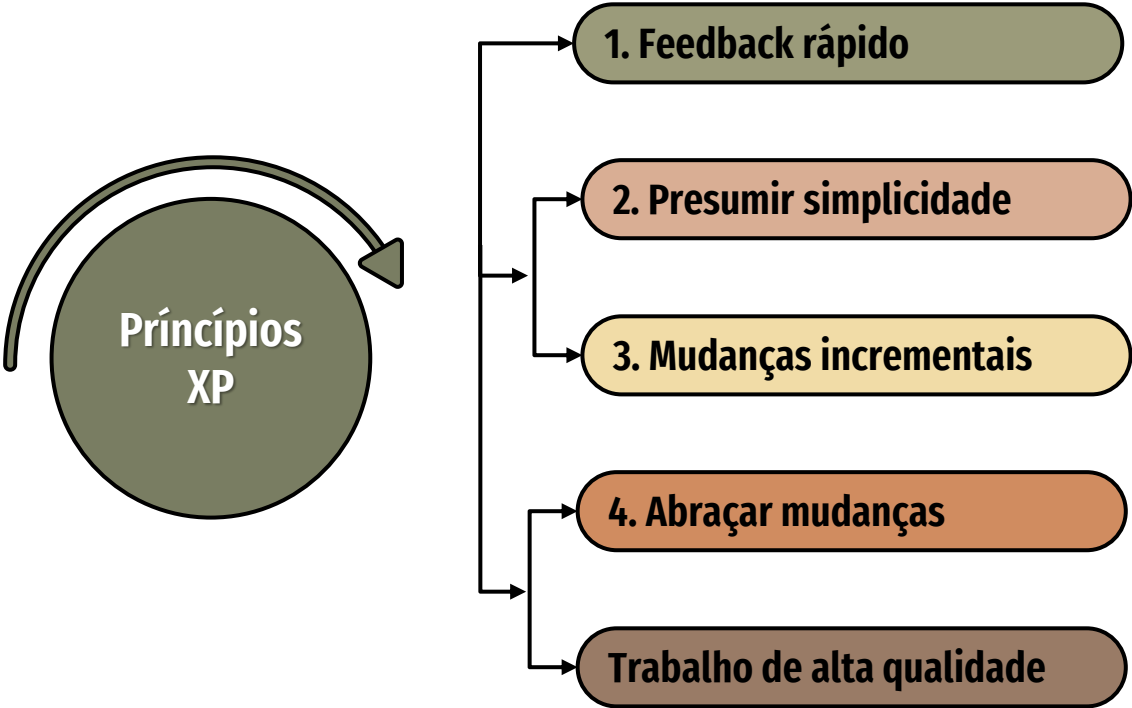


5.3. EXTREME PROGRAMMING (XP)

- É uma metodologia ágil de desenvolvimento que visa melhorar a qualidade do *software* e a satisfação do cliente através de práticas da Engenharia de Software e colaboração eficaz da equipe.
- Desenvolvida por Kent Beck na década de 1990, o XP se baseia em uma série de valores fundamentais e práticas-chave para promover a entrega rápida e contínua de *software* de alta qualidade.
- Entre os principais valores do XP temos:



- A partir desses valores, uma série de princípios básicos foram definidos:



- Para aplicar o XP, é necessário seguir uma série de práticas que dizem respeito ao relacionamento com o cliente, a gerência do projeto, a programação e os testes. (WAZLAWICK, 2013; CAIXETA, 2023).
- Citemos algumas práticas:

1. *Feedback contínuo*

- ✓ Enfatiza a importância do *feedback* contínuo, tanto dos clientes quanto dentro da equipe de desenvolvimento. Isso é alcançado através de práticas como desenvolvimento orientado a testes (TDD), integração contínua, revisões de código e cliente no local.

2. *Metáfora (Metaphor)*

- ✓ É importante conhecer a linguagem do cliente e seus significados. A equipe deve aprender a se comunicar de forma compreensível.

3. *Jogo de planejamento (Planning game)*

- ✓ Reuniões semanais com o cliente para priorizar as funcionalidades a serem desenvolvidas. [...]. Esse tipo de modelo de relacionamento com o cliente é adaptativo.

4. Desenvolvimento orientado a testes (*Test Driven Development – TDD*)

- ✓ No TDD, os testes automatizados são escritos antes do código de produção. Isso ajuda a garantir que o código seja testável, confiável e atenda aos requisitos do cliente desde o início do processo de desenvolvimento.

5. Integração contínua

- ✓ Envolve a integração frequente de código novo no repositório principal, juntamente com a execução automática de testes para verificar se o código introduzido não quebra a funcionalidade existente.

6. Programação em pares (*Pair programming*)

- ✓ O desenvolvimento do código é sempre feito por duas pessoas em cada computador, em geral um programador mais experiente e um aprendiz. [...]. Com isso, o código gerado será sempre verificado por pelo menos duas pessoas, reduzindo as possibilidades de erros.

7. Ritmo sustentável

- ✓ Trabalhar com qualidade não mais que 8 horas diárias.

8. Versões pequenas (*Small releases*)

- ✓ A liberação de pequenas versões do sistema pode ajudar o cliente a testar as funcionalidades de forma contínua. O XP leva esse princípio ao extremo, sugerindo versões ainda menores do que as de outros processos incrementais, como *UP* e *Scrum*.

9. Posse coletiva (*Collective ownership*)

- ✓ O código não tem dono e não é necessário pedir permissão a ninguém para modificá-lo.

10. Padrões de codificação (*Coding standards*)

- ✓ Deve estabelecer e seguir padrões de codificação, de forma que o código pareça ter sido todo desenvolvido pela mesma pessoa, mesmo que tenha sido feito por dezenas delas.

11. Teste de aceitação (*Customer tests*)

- ✓ São testes planejados e conduzidos pela equipe em conjunto com o cliente para verificar se os requisitos foram atendidos.

12. Refatoração (*Refactoring*)

- ✓ Trata-se de um processo de melhoramento do código e não implementação de novas funcionalidades. Essas revisões sistemáticas permitem a diminuição de erros. (CAIXETA, 2023).

13. *Design simples*

- ✓ O XP promove um *design* simples e modular, onde o código é refatorado regularmente para manter a simplicidade e a clareza. Isso ajuda a facilitar a manutenção e evolução do *software* ao longo do tempo.

14. Cliente no local

- ✓ O XP incentiva a presença do cliente ou representante do cliente na equipe de desenvolvimento. Isso ajuda a garantir uma comunicação clara e uma compreensão compartilhada dos requisitos do cliente.

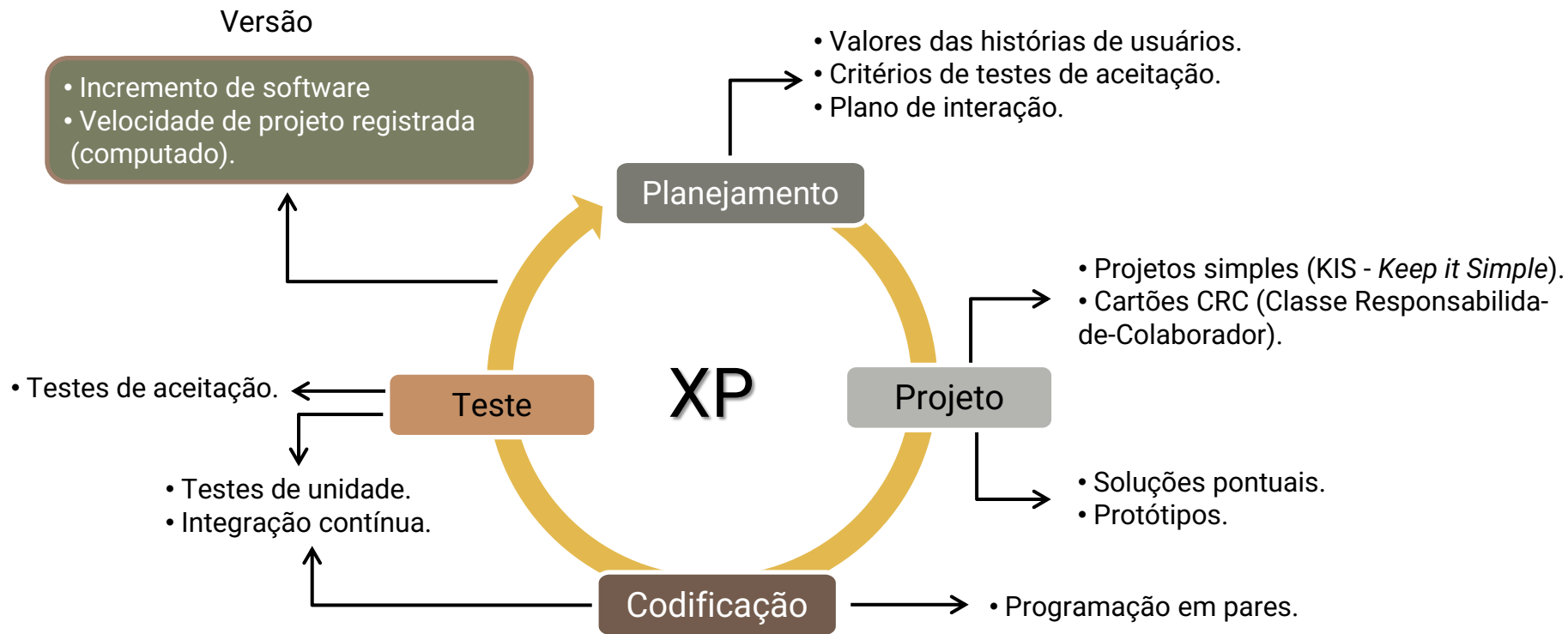
15. Reuniões em pé (*Stand-up meeting*)

- ✓ Como no caso do *Scrum*, reuniões em pé tendem a ser mais objetivas e efetivas.

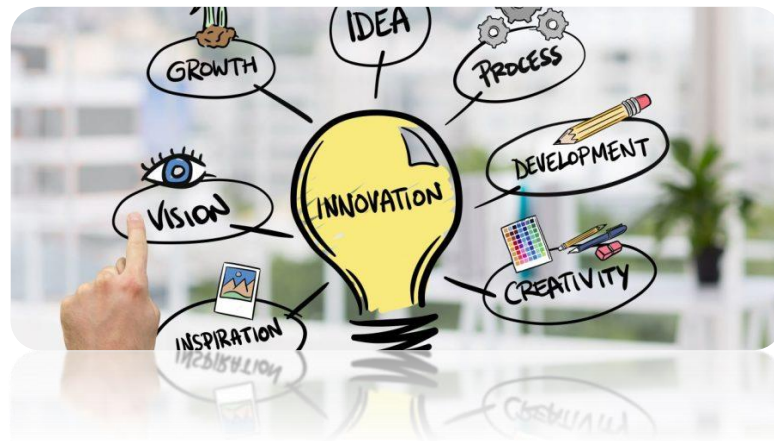
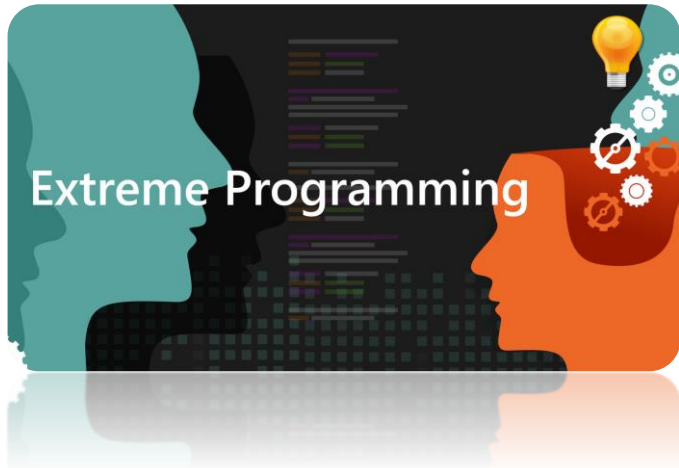
16. Iterações curtas

- ✓ O XP adota iterações curtas de uma ou duas semanas, conhecidas como "iterações ou releases". Isso permite que a equipe entregue valor ao cliente de forma rápida e iterativa, respondendo rapidamente ao feedback do cliente e ajustando os planos conforme necessário.

5.3.1. OS PROCESSOS XP



- Esses são apenas alguns dos princípios e práticas-chave do *Extreme Programming*.
- Ao adotar essas práticas, as equipes de desenvolvimento podem melhorar a qualidade do *software*, aumentar a satisfação do cliente e promover uma cultura de colaboração e excelência técnica.



5.4. KANBAN

- Kanban é uma metodologia de gestão visual que se baseia em cartões físicos ou virtuais para representar o trabalho em um quadro visual.
- O Kanban ajuda as equipes a visualizar seu fluxo de trabalho, limitar o trabalho em progresso e identificar gargalos e oportunidades de melhoria.
- Tem origem no Sistema de Produção da Toyota. É frequentemente usado em contextos onde a demanda e as prioridades mudam frequentemente, como suporte técnico, operações de T.I e desenvolvimento de produtos.
- Apresentaremos nas próximas páginas os principais conceitos e práticas do sobre o Kanban.



1

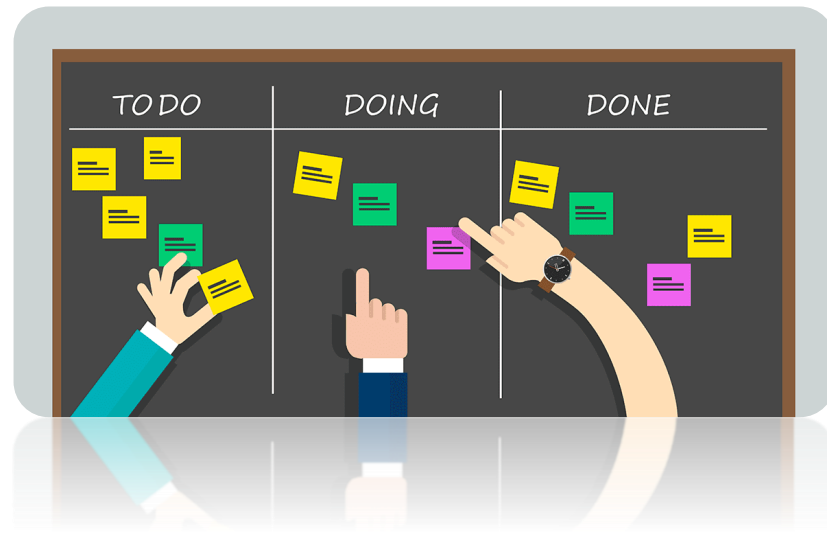
QUADRO KANBAN

- É uma representação visual do fluxo de trabalho da equipe.
- Consiste em colunas que representam diferentes etapas do processo, como "A Fazer" (to DO), "Em Progresso" (DOING) e "Concluído" (DONE).
- Cada tarefa ou item de trabalho é representado por um cartão que se move através das colunas à medida que progride no processo.



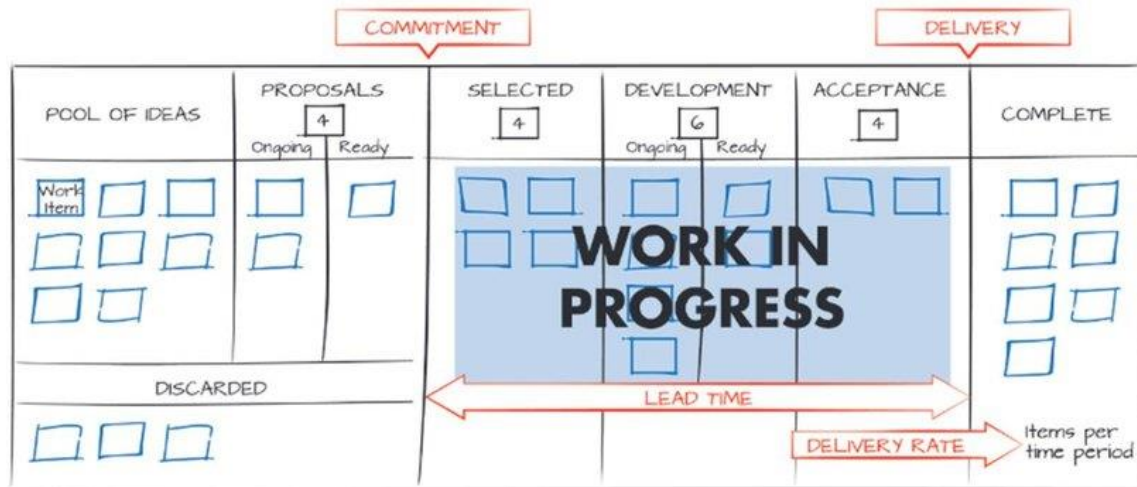
Veja o vídeo no canal do Youtube

<https://www.youtube.com/watch?v=K9b4JC5CsQs>



2 LIMITAÇÃO DO TRABALHO EM PROGRESSO (WIP)

- O Kanban enfatiza a importância de limitar a quantidade de trabalho em progresso em cada etapa do processo. Isso ajuda a evitar o acúmulo de trabalho, reduzir o tempo de espera e manter um fluxo de trabalho equilibrado.



3

GESTÃO DO FLUXO

- O Kanban se concentra em otimizar o fluxo de trabalho, identificando e removendo gargalos e impedimentos que possam prejudicar a eficiência e a produtividade da equipe.
- Isso é alcançado através da análise do fluxo de valor e da implementação de melhorias graduais para suavizar o fluxo.

4

FEEDBACK VISUAL

- Fornece um *feedback* visual instantâneo sobre o status do trabalho, permitindo que a equipe identifique rapidamente problemas e tome medidas corretivas conforme necessário. Isso é facilitado pelo uso de cartões coloridos, etiquetas e indicadores visuais no Quadro Kanban.

5

MELHORIA CONTÍNUA

- Promove uma cultura de melhoria contínua, onde a equipe busca constantemente identificar oportunidades de otimização e implementar mudanças incrementais para aumentar a eficiência e a qualidade do trabalho.

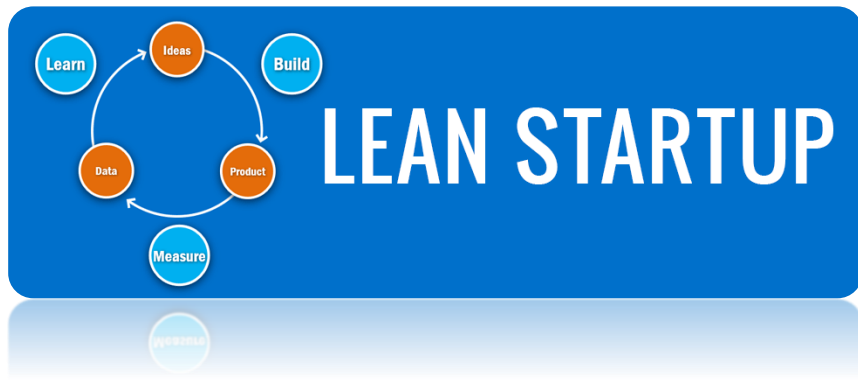
6

FLEXIBILIDADE E ADAPTABILIDADE

- O Kanban é altamente flexível e pode ser adaptado para atender às necessidades específicas de diferentes equipes e projetos.
- Permite que as equipes ajustem seu processo de trabalho conforme necessário e respondam rapidamente às mudanças nas demandas do cliente ou nas condições de mercado.
- Ao adotar o Kanban, as equipes podem melhorar a visibilidade do trabalho, aumentar a eficiência do processo e promover uma cultura de colaboração e melhoria contínua. Essa abordagem ajuda as organizações a entregar valor ao cliente de forma mais rápida e consistente, enquanto mantêm a flexibilidade necessária para se adaptar às mudanças no ambiente empresarial.

5.5. LEAN STARTUP

- A metodologia *Lean Startup* é uma abordagem para criar e gerenciar *startups* e produtos, com o objetivo de maximizar o aprendizado do cliente, minimizar o desperdício de recursos e acelerar o ciclo de desenvolvimento.
- Enfatiza a construção de um "mínimo produto viável" (MVP) para testar hipóteses de negócios e validar ideias antes de investir recursos significativos.
- Desenvolvida por Eric Ries, a *Lean Startup* se baseia nos princípios do *Lean Manufacturing* e se concentra em validar rapidamente hipóteses de negócios através de experimentação e iteração.



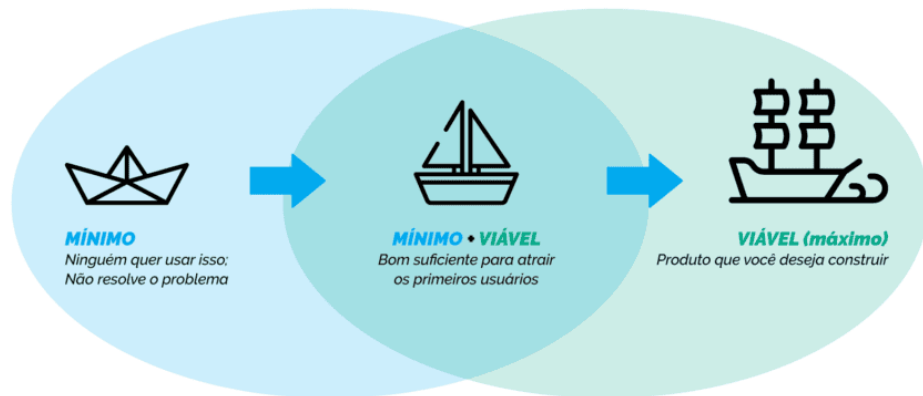
- Apresento aqui alguns princípios e práticas da *Lean Startup*.

1 Construir-Medir-Aprender

- O ciclo de construir – medir - aprender é o “coração” da metodologia *Lean Startup*. Em vez de gastar meses ou anos desenvolvendo um produto antes de lançá-lo no mercado, os empreendedores criam um MVP (*Minimum Viable Product* - Produto Mínimo Viável) inicial para testar suas hipóteses junto aos clientes o mais rápido possível. Então medem e analisam os resultados, aprendem com o feedback dos clientes e itera no produto com base nesse aprendizado.
- + info no link abaixo:



O ANALISTA DE
MODELOS DE NEGÓCIOS



2 Validação de hipóteses

- A *Lean Startup* enfatiza a importância de validar rapidamente as hipóteses de negócios, ou seja, testar as suposições sobre o mercado, os clientes e o produto o mais cedo possível. Isso é feito através de experimentos controlados, como testes A/B¹, entrevistas com clientes e lançamentos de MVPs.

3 Feedback do cliente

- O *feedback* dos clientes é fundamental para o processo *Lean Startup*. Os empreendedores procuram entender as necessidades e desejos dos clientes, coletar informações sobre o produto e utilizá-las para orientar o desenvolvimento e as decisões de negócios.

1. Acesse o link <http://twixar.me/CLCm> e saiba mais sobre o que é Teste A/B.

4

Desenvolvimento enxuto

- A *Lean Startup* promove um modelo de desenvolvimento enxuto, onde os recursos são alocados de forma eficiente e o desperdício é minimizado. Isso inclui a utilização de processos ágeis de desenvolvimento de *software*, a automação de tarefas repetitivas e a eliminação de recursos desnecessários.

5

Métricas de sucesso

- A *Lean Startup* enfatiza o uso de métricas mensuráveis e objetivas para avaliar o progresso e o sucesso do produto. Em vez de se concentrar apenas em métricas tradicionais, como receita e lucro, os empreendedores também consideram métricas de aprendizado, como taxa de conversão, tempo de ciclo e taxa de retenção de clientes.

- No geral, a *Lean Startup* promove uma abordagem experimental e orientada para dados, afim de gerenciar *startups* e produtos.
- Ao adotar os princípios e práticas, os empreendedores podem reduzir o risco de fracasso, otimizar o uso de recursos e criar produtos que atendam às necessidades reais do mercado.
- Em síntese, as Metodologias ágeis e a Gestão ágil são interligadas, sendo as metodologias responsáveis por fornecerem as ferramentas e práticas necessárias para impulsionar a agilidade em toda a organização. Por sua vez, a gestão expande essas práticas além do desenvolvimento de *software*, abarcando todas as áreas da empresa.

REFERÊNCIAS

CAIXETA, Daniel. Aula de Engenharia de Software. Unidade 4. Metodologias ágeis. FBr., 2023. (Apresentação).

CARSTENS, Danielle; FONSECA, Edson. Gestão da tecnologia e inovação. Curitiba, PR: Intersaberes (BVU), 2019. (Biblioteca virtual).

CRUZ, Fábio. *Scrum e Agile* em projetos. 2. ed. Rio de Janeiro: Brasport, 2018. E-book. Disponível em: <https://plataforma.bvirtual.com.br>. Acesso em: 14 dez. 2023. (Biblioteca virtual).

MASSARI, Vitor L.; VIDAL, André. Gestão Ágil de Produtos com *Agile Think Business Framework*: guia para certificação EXIN Agile Scrum Product Owner. Rio de Janeiro, RJ: Brasport (BVU), 2018. (Biblioteca virtual).

MUNIZ, Antônio et. al. Jornada Business Agility: Entenda como a agilidade nos negócios colabora para adaptabilidade contínua e resultados de valor aos clientes. Rio de Janeiro, RJ: Brasport (BVU), 2021. (Biblioteca virtual).

WAZLAWICK, R. S. Engenharia de software: conceitos e práticas. Rio de Janeiro: Elsevier, 2013.