

# Curso: Aprendizagem de Máquina em Inteligência Artificial

## Disciplina: Internet das Coisas

### Aula: Modelos Ágeis de Projetos



# Agenda

1. Seleção Ciclo de Vida

2. Princípios e pensamento ágil

4. Frameworks ágeis

5. Gestão de times ágeis

6. Iniciando projetos ágeis

7. Planejamento Adaptativo

8. Executando projetos ágeis

9. Monitorando e Controlando projetos ágeis

10. Práticas de melhoria contínua



# Agenda

1. Visão 6º Ed. Guia PMBOK

2. Seleção Ciclo de Vida

3. Princípios e pensamento ágil

4. Frameworks ágeis

5. Gestão de times ágeis

6. Iniciando projetos ágeis

7. Planejamento Adaptativo

8. Executando projetos ágeis

9. Monitorando e Controlando projetos ágeis

10. Práticas de melhoria contínua



# O que é um projeto?

**PROJETO** é um esforço temporário empreendido para criar um produto, serviço ou resultado único (Guia

PMBOK, 2017, 6ª ed.)

## Empreendimento temporário

- Um início e um término definidos e não necessariamente de curta duração.
- O final do projeto é alcançado quando:
  - **Objetivos** do projeto **foram alcançados**;
  - **Objetivos não serão** ou não poderão ser cumpridos;
  - Os **recursos estão esgotados** ou **não estão mais disponíveis** para alocação ao projeto;
  - A necessidade do projeto **não existe mais**;
  - O **projeto é finalizado** por motivo legal ou por conveniência.

## Criar um produto, serviço ou resultado exclusivo (ENTREGA).

- Um **produto único** que pode ser um componente de outro item, um aprimoramento ou um novo item final;
- Um **serviço único** capaz de realizar um serviço;
- Um **resultado único**, como um produto ou documento;
- Uma **combinação única** de um ou mais produtos, serviços ou resultados (ex. um aplicativo de software, documentação associadas e serviços de centrais de atendimentos).



# 5 Grupos de Processos de Gerenciamento de Projetos

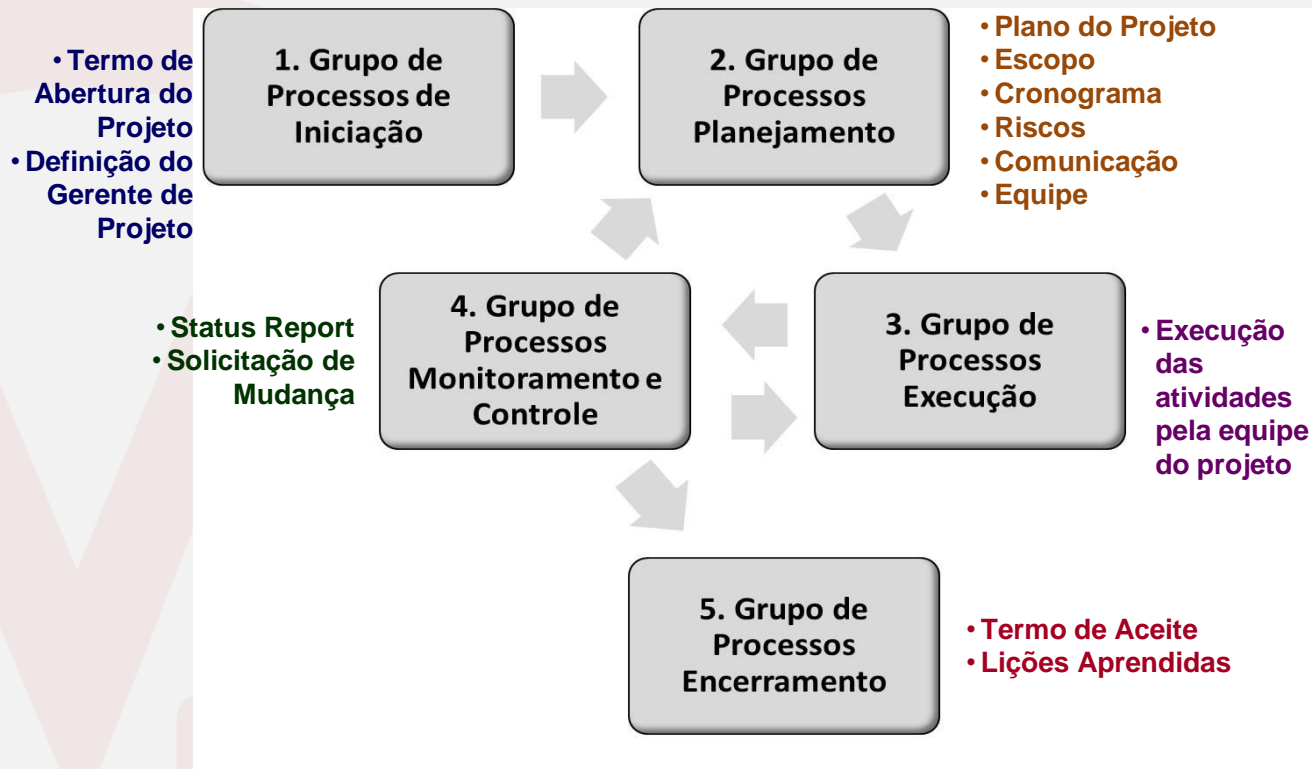


Figura: Cinco grupos de processos de gerenciamento de projetos

Fonte: baseado no Guia PMBOK®, 6º ed. 2017.

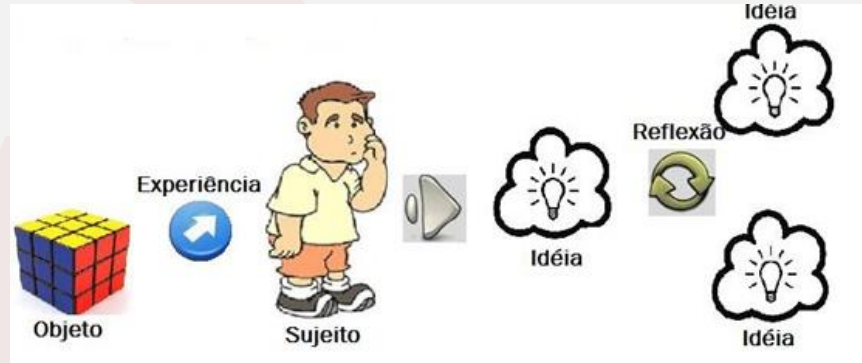


# 10 Áreas de Conhecimento do Guia PMBOK® - 6ª ed.



# Teoria do Empirismo

- A teoria do empirismo é baseada no conceito que **problemas complexos são difíceis de serem previstos**. O empirismo ajuda as pessoas a **tratarem problemas complexos**;
- Isso requer dar um **passo de cada vez**, a fim de realizar **uma pequena quantidade de trabalho para adquirir experiência**.



Fonte: <https://andersonyankee.wordpress.com/2010/11/16/distincao-de-como-se-da-o-conhecimento-em-david-hume-e-john-locke/>

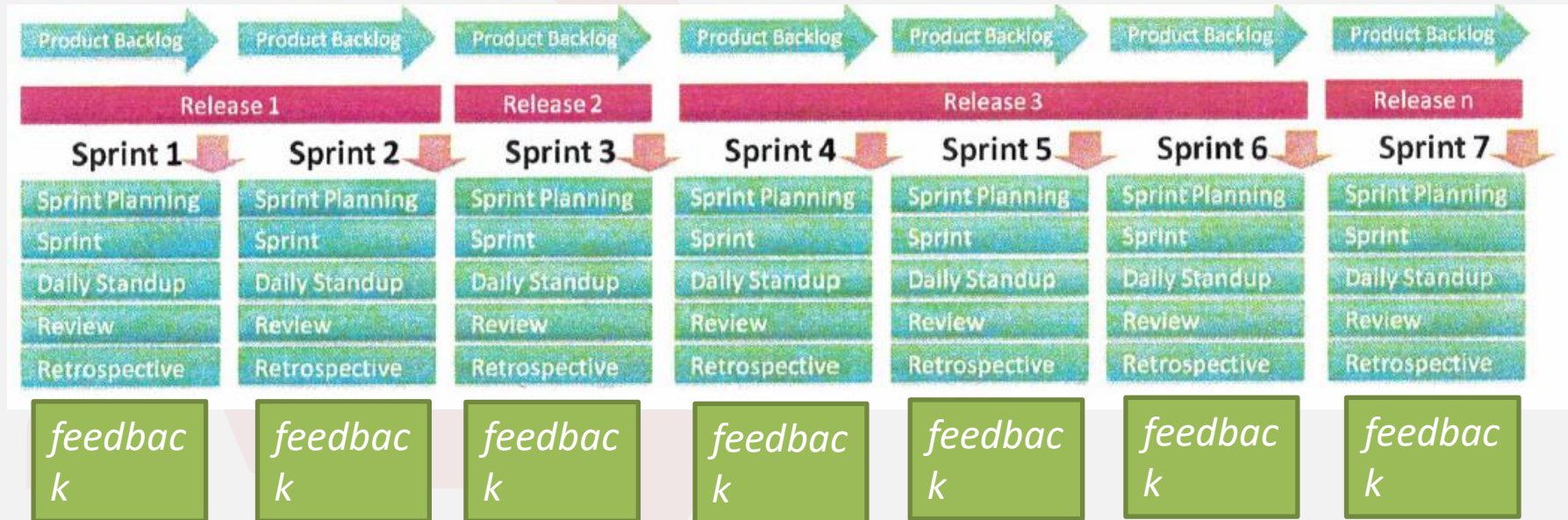
**Empirismo** é a escola do pensamento filosófico relacionada à **teoria do conhecimento**, que pensa estar na **experiência a origem de todas as ideias**. O nome empirismo vem do latim: *empiria* (experiência) e *-ismo* (sufixo que determina, entre outras coisas, uma corrente filosófica). Temos, assim, a “corrente filosófica da experiência”.

Fonte: <https://mundoeducacao.bol.uol.com.br/filosofia/empirismo.htm>

# Ciclo de Vida Ágil

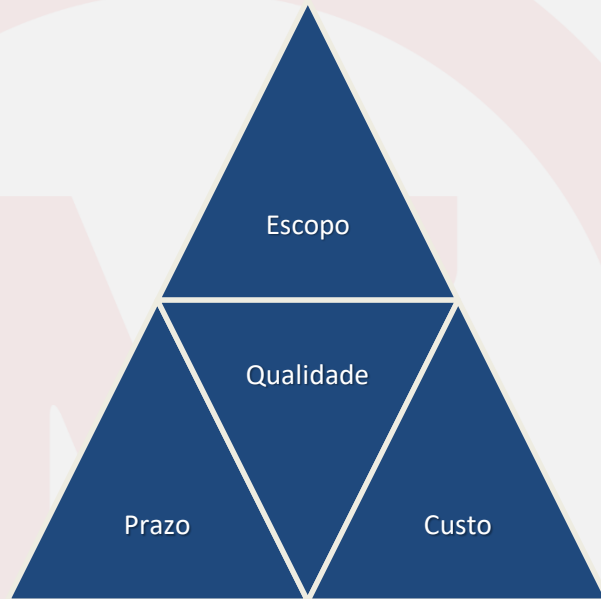
No Ciclo de Vida Ágil, os requisitos são dinâmicos, as atividades são repetidas até as entregas estarem corretas, produz e refina **entregas em lotes pequenos frequentes** por meio do **recebimento de feedbacks** das partes. Tem como objetivo **entregar valor ao cliente, priorizando as entregas**. É adaptativo e orientado a mudança.

Figura - Ciclo de Vida Ágil

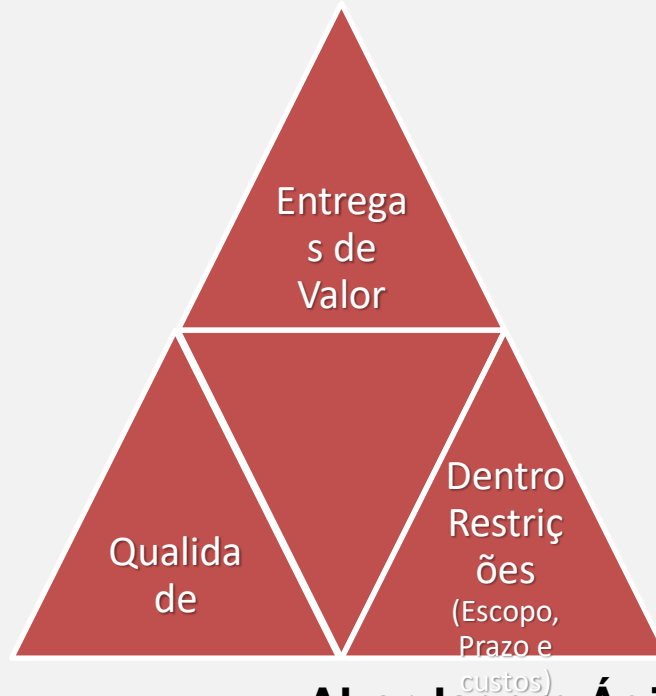




# Restrições de projetos ágeis



**Abordagem  
Preditiva**



**Abordagem Ágil**



# Agenda

1. Visão 6º Ed. Guia PMBOK

2. Seleção Ciclo de Vida

3. Princípios e pensamento ágil

4. Frameworks ágeis

5. Gestão de times ágeis

6. Iniciando projetos ágeis

7. Planejamento Adaptativo

8. Executando projetos ágeis

9. Monitorando e Controlando projetos ágeis

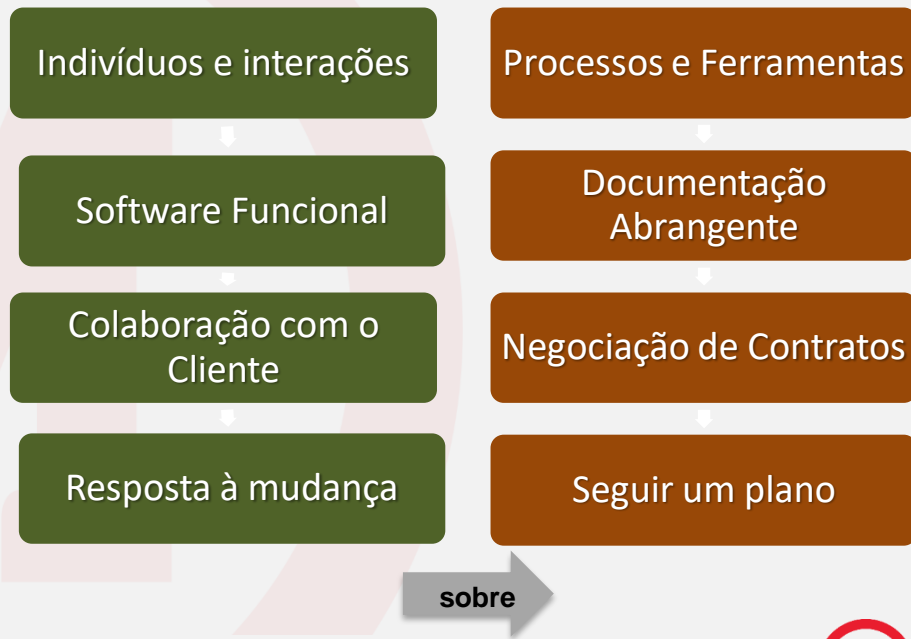
10. Práticas de melhoria contínua



# Manifesto Ágil

Em 2001, 17 gurus do desenvolvimento de software reuniram-se na cidade de *Snowbird* para discutirem porque os **projetos de software sempre fracassam ou atrasam ou são problemáticos** sempre pelos mesmos motivos. Daí nasceu o **Manifesto Ágil**.

**Figura - Valores do Manifesto Ágil**



## 12 Princípios do Manifesto Ágil

1. Nossa maior prioridade é **satisfazer o cliente** por meio da contínua e adiantada **entrega de software de valor**;
2. As **mudanças** de requisitos **são bem-vindas**, mesmo no **fim do desenvolvimento**. Processos ágeis se adequam a mudanças para que o cliente possa ter vantagens competitivas;
3. **Entregar software funcional frequentemente**, desde algumas semanas até alguns meses, com preferência a períodos mais curtos;
4. Pessoas de negócios e desenvolvedores devem **trabalhar diariamente** e em conjunto **ao longo de todo o projeto**;
5. Construa projetos ao **redor de indivíduos motivados**. Dê a eles o ambiente e o suporte que necessitem para ter o trabalho bem feito;
6. O método mais eficiente e eficaz de transmitir informações para , e entre a equipe de desenvolvimento, é através de conversação **cara a cara (face to face)**;
7. **Software funcional** é a **medida primeira do progresso**;
8. Processos ágeis promovem **desenvolvimento sustentável**. Os patrocinadores, desenvolvedores e usuários devem ser capazes de manter ritmos constantes (cadência) indefinidamente;
9. **Contínua atenção à excelência técnica e bem design** melhoram a agilidade;
10. **Simplicidade**: a arte de maximizar a quantidade de trabalho não realizado é essencial;
11. As melhores arquiteturas, requisitos e *designs* **emergem de equipes auto organizadas**;
12. Em intervalos regulares, **a equipe reflete em como se tornar mais efetiva**, então, se **ajustam e refinam** seu comportamento adequadamente.



# O que é ser Ágil?

- A palavra “Ágil” é um adjetivo, que significa:

“Para mover-se rapidamente e facilmente, ou de pensar rapidamente de uma forma inteligente”

Fonte: Dicionário Aurélio, 2018

- Ter **rapidez** e **desembaraço**;
- Entregar aquilo que **gera o maior valor para o cliente**, de forma cada vez mais **rápida** e com **qualidade**;
- Fazer as coisas **complexas** de forma **simples**;
- Ter a **equipe** toda **comprometida** com os objetivos do **projeto**

Ter capacidade de responder rapidamente as mudanças

## LEMBRE-SE!

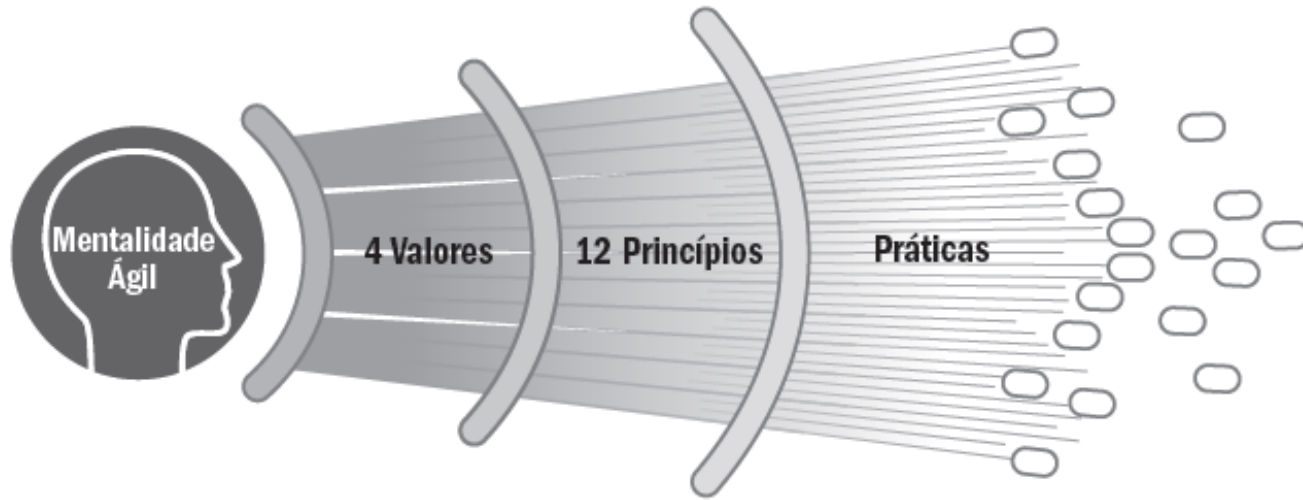
- Para ser ágil, deverá adotar os **valores e princípios** do Manifesto Ágil
- Realizar **primeiro aquilo que trará mais retorno para o cliente e não o que é mais fácil desenvolver**
- **Ser transparente com equipes, clientes, fornecedores, etc.**
- **Se você adota o modelo de trabalho ágil, mas não é transparente com seu cliente, VOCÊ NÃO É ÁGIL.**





**AGILIDADE**  
=  
Ser ágil  
+  
Ser adaptativo  
+  
Usar um *framework*

# Mindset Ágil ou Mentalidade Ágil



Ágil é uma mentalidade definida por valores, orientada por princípios e manifestada por meio de muitas práticas diferentes. Os praticantes ágeis selecionam as práticas com base em suas necessidades.

Fonte: Guia PMBOK, 6º ed. p. 10, seção 2



# Agenda

1. Visão 6ª Ed. Guia PMBOK

2. Seleção Ciclo de Vida

3. Princípios e pensamento ágil

4. Frameworks ágeis

5. Gestão de times ágeis

6. Iniciando projetos ágeis

7. Planejamento Adaptativo

8. Executando projetos ágeis

9. Monitorando e Controlando projetos ágeis

10. Práticas de melhoria contínua





# Contexto da Aplicação Ágil

A aplicação dos métodos ágeis é recomendada para contextos “**complicados**” e “**complexos**”, onde existe considerável incerteza em relação aos requisitos e/ou tecnologia:

- **Simple**: Abordagem preditiva
- **Complicated**: abordagem ágil
- **Complex**: Abordagem ágil
- **Chaos**: risco iminente

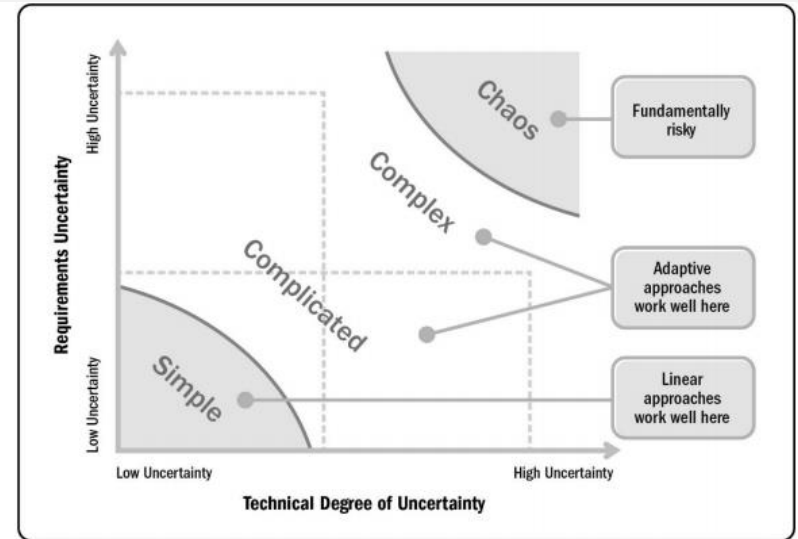


Figura 2-5. Modelo de incerteza e complexidade inspirado no Modelo de Complexidade de Stacey

# Frameworks Ágeis

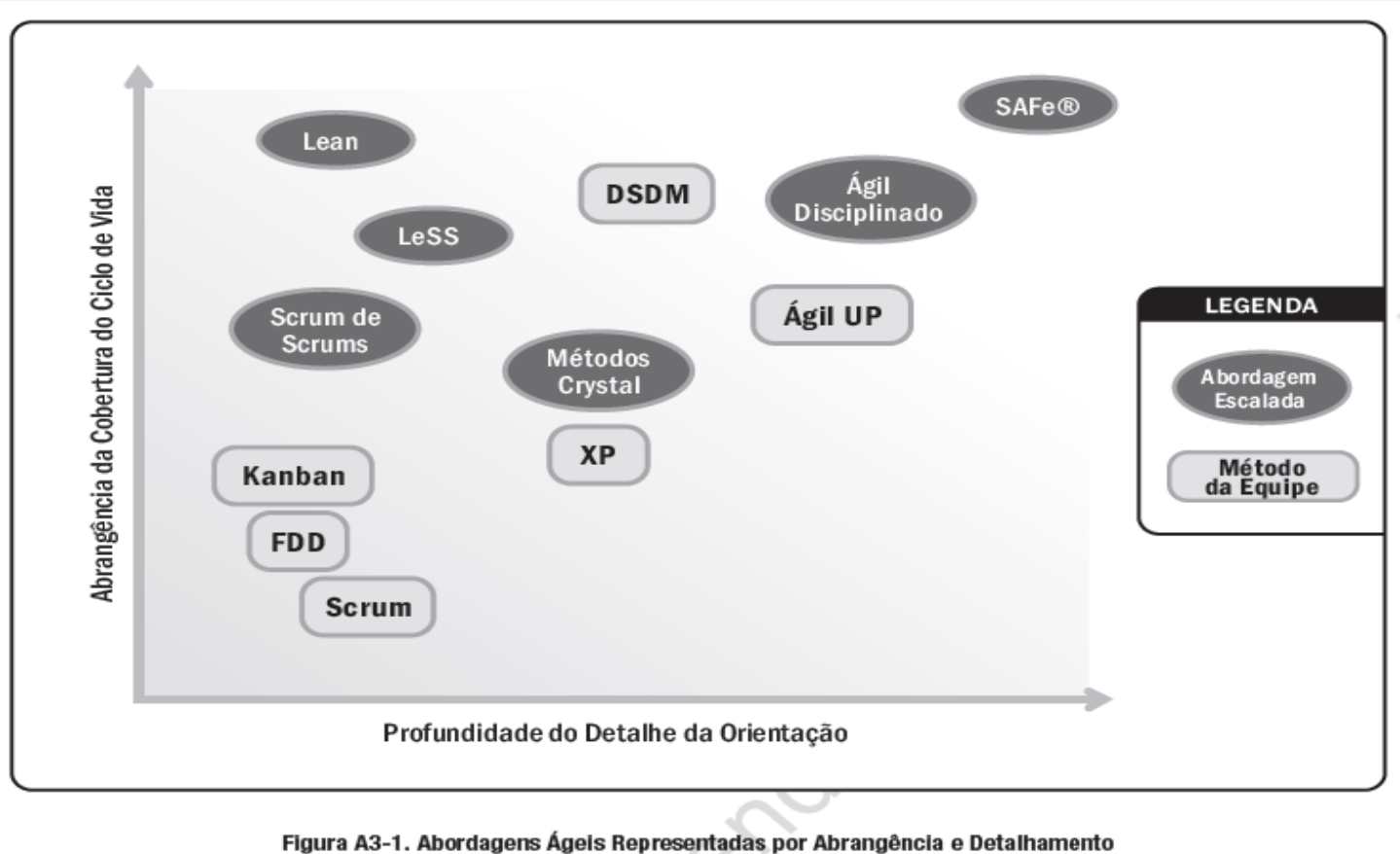
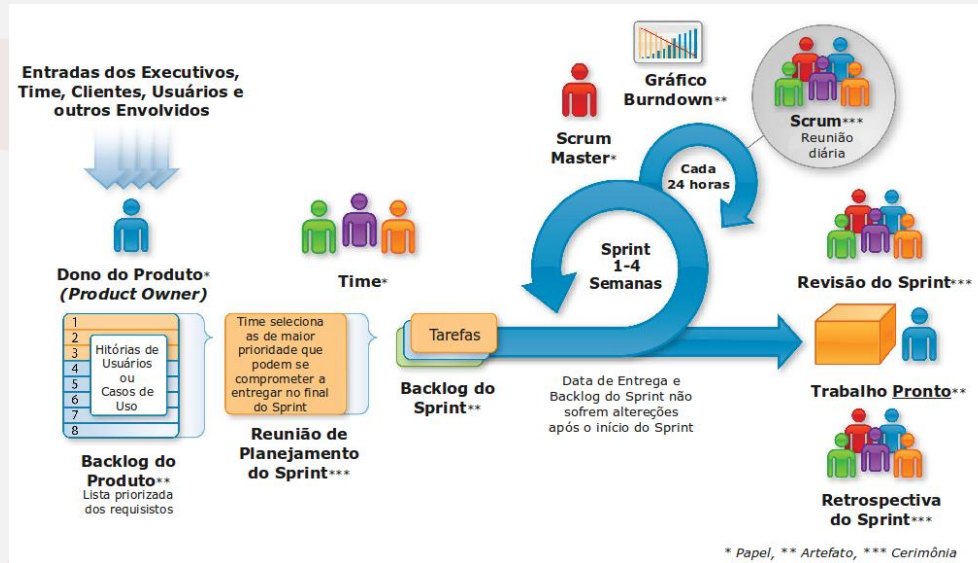


Figura A3-1. Abordagens Ágeis Representadas por Abrangência e Detalhamento

Fonte: Guia PMBOK, 6º ed.

# O que é o SCRUM



- É um framework para **gerenciamento de projetos e desenvolvimento ágil de software**;
- Não é um processo detalhado e **prescritivo** para todas as situações;
- É baseado no desenvolvimento **iterativo e incremental**;
- É um **framework ágil voltado a uma única equipe** para gerenciar o desenvolvimento de um produto em **entregas incrementais por meio de ciclos interativos**;
- É executado em *timeboxes* de **1 mês** ou menos com durações constantes;
- A cada sprint são produzidos **incrementos do produto potencialmente liberáveis / utilizáveis**;
- É **fácil de ser entendido** e **difícil de ser aplicado**.

# SCRUM - Pilares do Processo Empírico

## Transparência

- Todo e qualquer fator ou acontecimento relacionado ao processo de entrega, que possa impactar o resultado final do projeto (produto), deve ser visível e do conhecimento de todos envolvidos, inclusive do cliente (ex. Kanban)

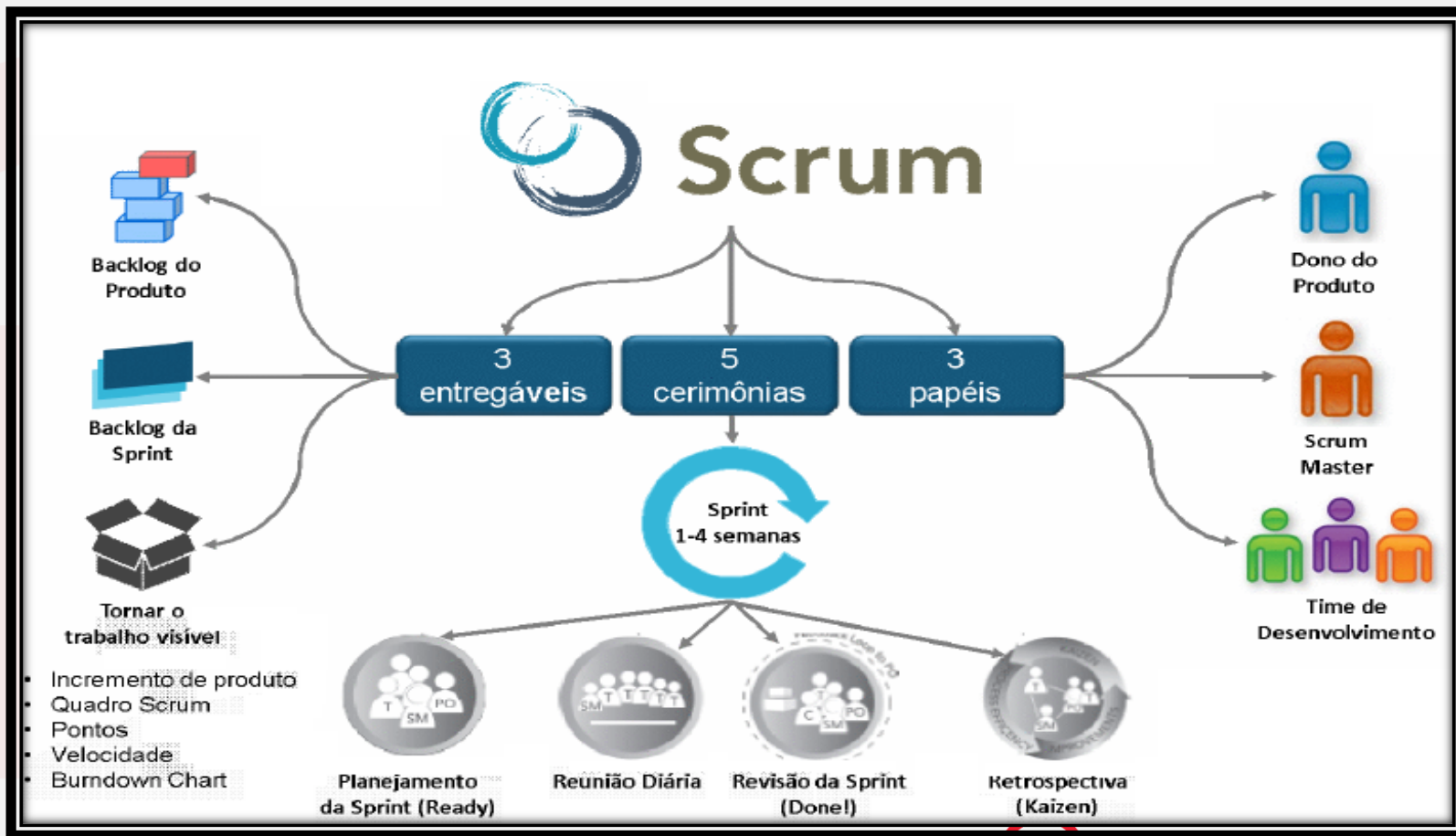
## Inspeção

- Todos os aspectos da entrega que possam impactar o resultado final do projeto devem ser inspecionados frequentemente para que qualquer variação prejudicial possa ser identificada e corrigida o mais o mais rápido possível

## Adaptação

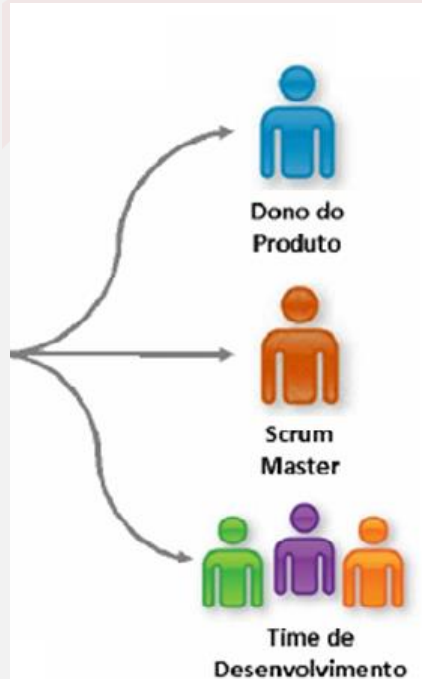
- Toda vez que uma variação prejudicial é identificada, o processo deve ser ajustado imediatamente, como forma de evitar outros desvios (retrospectiva)

# Estrutura do SCRUM



# Estrutura do SCRUM

## 3 papéis



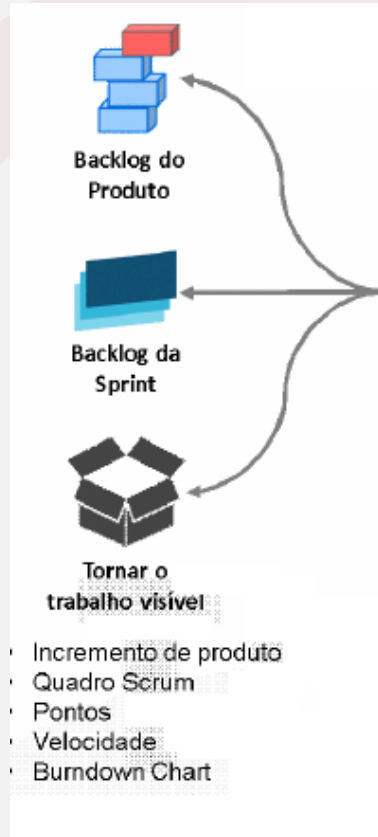
Elabora a visão do projeto, define e gerencia o **backlog do produto**, **priorizando**, maximizando e garantindo valor, entendimento e visibilidade do produto.

Coach da equipe, líder servidor, remove impedimentos e o facilitador dos eventos. **É o dono do processo**, responsável por garantir que toda a equipe compreenda, adote e siga as regras do Scrum. Também orienta a equipe sobre como remover impedimentos.

**Equipe auto organizada e multifuncional** que é composta por membros (desenvolvedores, dba's, analistas, etc.), trabalham no desenvolvimento e possuem tudo o que necessitam para entregar incrementos de um produto pronto ao final de cada sprint sem dependência externa. Gerenciam o back-log de atividades técnicas (***Sprint backlog***)

# Estrutura do SCRUM

## 3 entregáveis



### Backlog do Produto

- **Lista ordenada e priorizada** de **requisitos** necessários para o **produto**;
- Mantida única e exclusivamente pelo *Product Owner* (PO);
- Constantemente refinada pelo PO (*Backlog Refinement* / **Grooming**)

### Backlog da Sprint

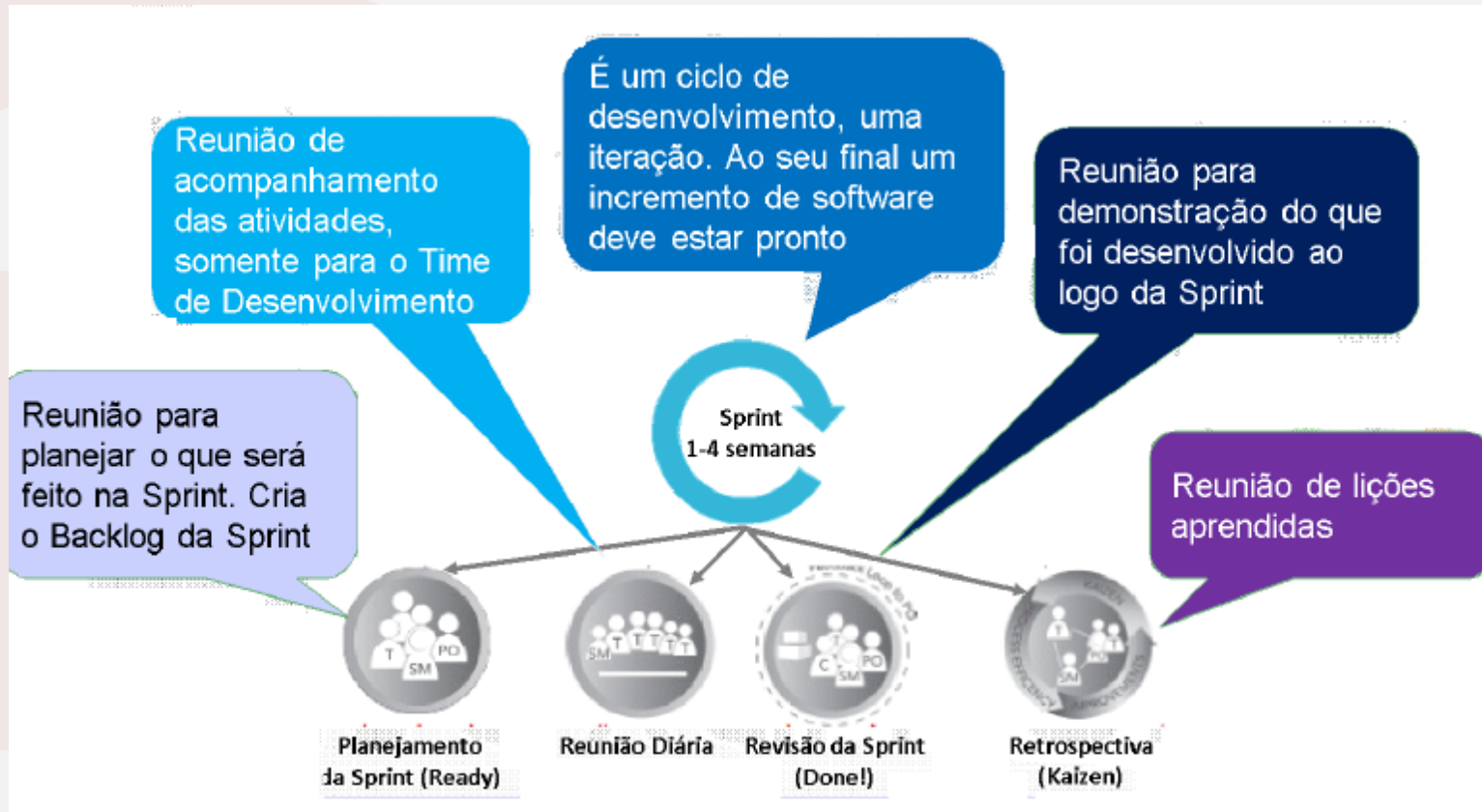
- **Lista de Tarefas técnicas** que deverão ser feitas na sprint para atingir sua meta;
- Mantida única e exclusivamente pela equipe de desenvolvimento.

### Incrementos

- Entregas / Produtos incrementais gerados ao final de cada sprint.

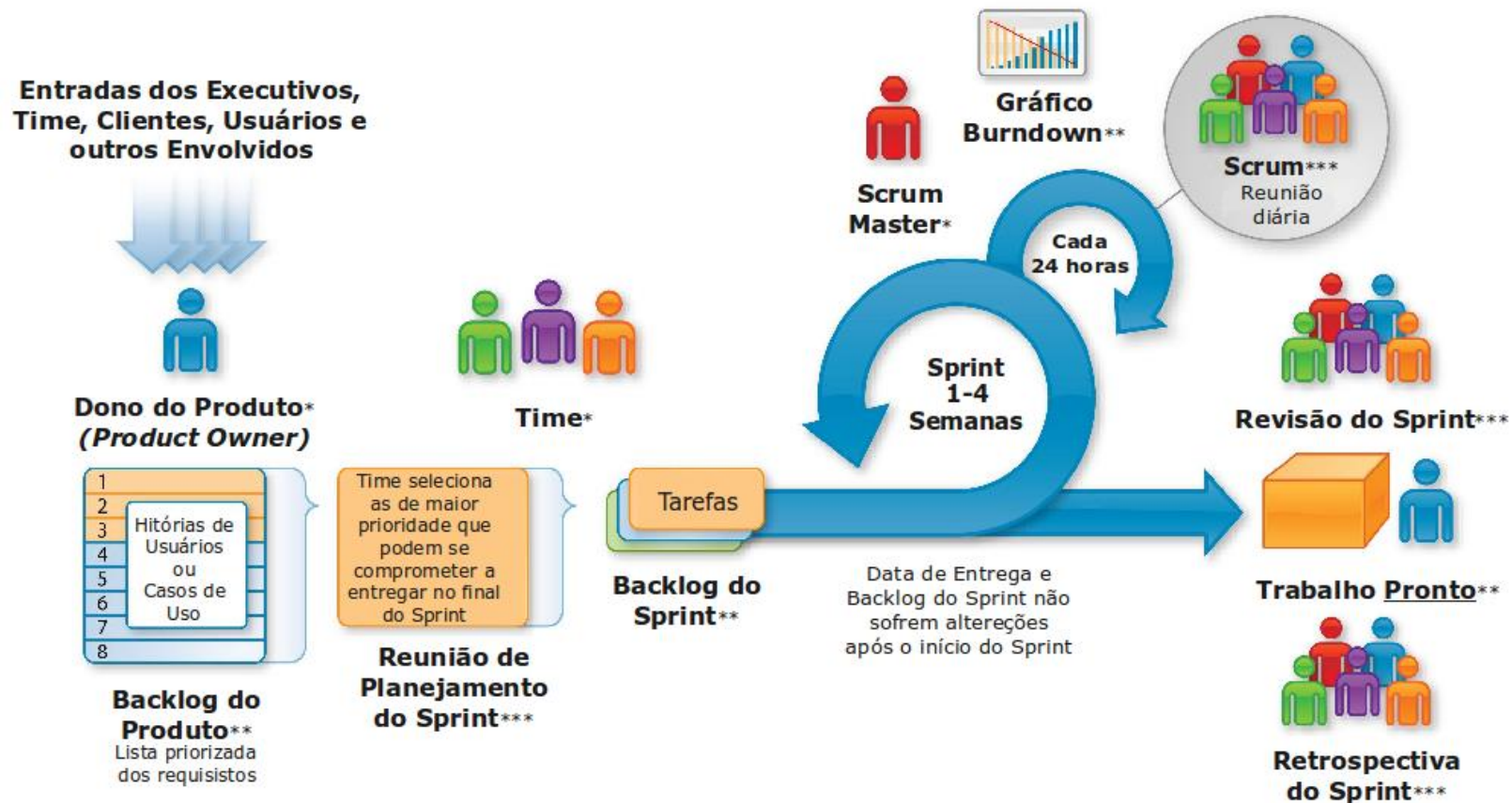
# Estrutura do SCRUM

## 5 Cerimônias





# Como o Scrum funciona



Fonte: <http://www.fabricadejogos.net/posts/scrum-como-metodologia-agil-na-producao-de-jogos-digitais/>

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

# eXtreme Programming (XP)

Do inglês eXtreme Programming (XP) é uma **metodologia ágil** para **equipes pequenas e médias** e que irão desenvolver **software com requisitos vagos e em constante mudança**. Para isso, adota a estratégia de **constante acompanhamento e realização de vários pequenos ajustes** durante o desenvolvimento de software.



## As práticas do XP

Área prática XP	Primária	Secundária
<b>Organizacional</b>	Sentem-se juntos Equipe inteira Espaço de trabalho informativo	Envolvimento Real com o Cliente Continuidade da equipe Ritmo sustentável
<b>Técnica</b>	Programação em pares Programação dos testes primeiro Design incremental	Código compartilhado Documentação de código e testes Refatoração
<b>Planejamento</b>	Histórias de usuário Ciclo semanal Ciclo trimestral Slack (folga/buffer)	Análise de causa-raiz Equipes reduzidas Pagamento por uso Contrato de escopo negociado Reunião diária em pé
<b>Integração</b>	Construção em 10 minutos Integração Contínua Teste primeiro	Base única de código Implantação incremental Implantação diária

Fonte: Agile Guide, 102, 2018.

# Feature-Driven Development (FDD)

Um projeto de desenvolvimento orientado a funcionalidade (FDD) é organizado em torno de **cinco processos ou atividades** realizados de modo **iterativo**.

- Desenvolva um modelo geral;
- Crie uma lista de funcionalidades;
- Planeje de acordo com a funcionalidade;
- Design de acordo com a funcionalidade;
- Construa de acordo com a funcionalidade.

Práticas recomendadas de **Engenharia de Software**

- Modelagem de objeto de domínio;
- Propriedade de classe individual;
- Equipe de funcionalidades
- Inspeções;
- Gerenciamento de configuração;
- Construções regulares;
- Visibilidade do andamento e dos resultados.

O FDD promove o “faseamento” por **FUNCIONALIDADE** (Domínio de

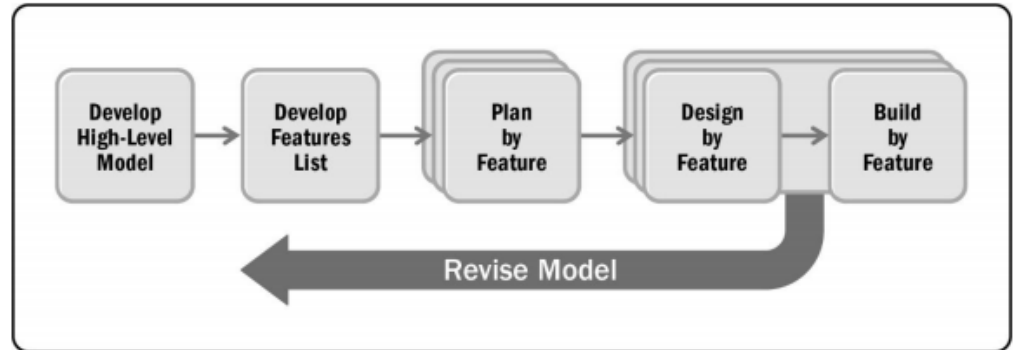
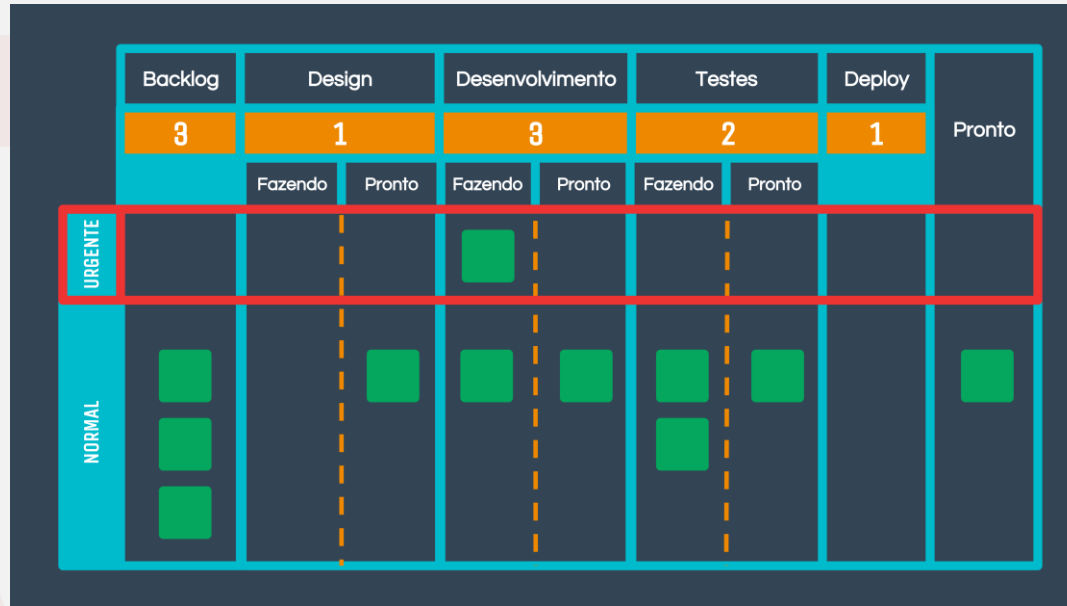


Figura A3-4. Ciclo de vida do Projeto de Desenvolvimento Orientado a Funcionalidades (FDD)

Fonte: Guia PMBOK, 6, ed. 2017

# KANBAN

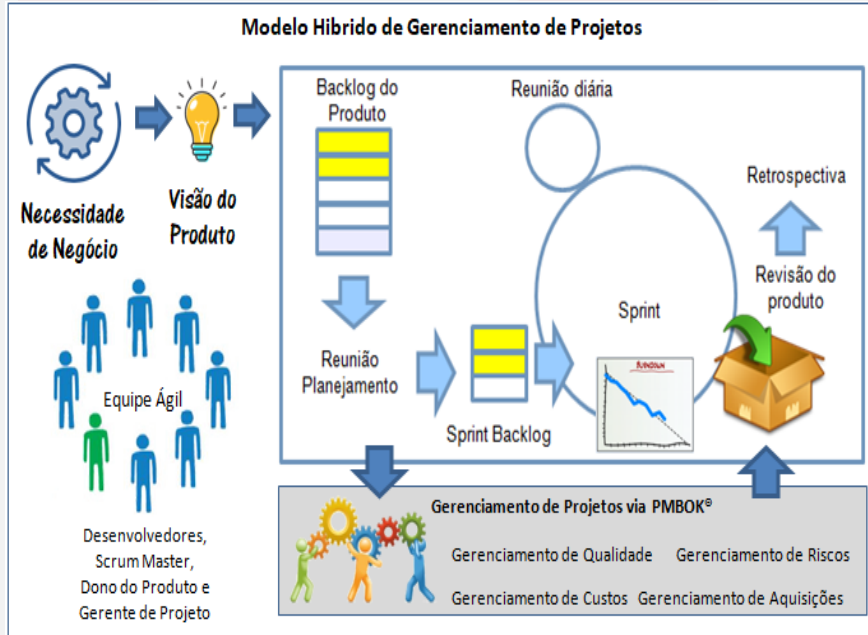


Fonte: <https://targetteal.com/pt/blog/metodo-kanban/>

- É uma palavra japonesa que significa “**tabuleta**” e foi adotado como metodologia de desenvolvimento inspirado no sistema de fábrica da Toyota;
- Trata-se de um grande **quadro branco** onde as **tarefas são escritas em post-its** e passam por vários estágios definidos;
- Sua principal finalidade é **gerenciar um fluxo de trabalho**, e não prover estimativas sobre o trabalho que está sendo realizado.

# Tailoring

**Tailoring** é identificar qual é o **melhor método** a utilizar ou mesmo como extrair de cada método de acordo com o projeto que está sendo conduzido. Para fazer um processo de tailoring, é utilizado o **conceito Shu-Ha-Ri**, de Alistair Cockburn.



Fonte: [https://www.sympla.com.br/palestra-gestao-de-projetos-ageis---gerenciando-projeto-com-modelo-hibrido\\_\\_277432](https://www.sympla.com.br/palestra-gestao-de-projetos-ageis---gerenciando-projeto-com-modelo-hibrido__277432)

**Shu – Obedecer.**  
Utilizar a metodologia ou framework do jeito original, sem grandes adaptações

**Ha - Romper.**  
Experimentar a combinação com outras metodologias em prol do seu projeto e vê o resultado

**Ri – Superar.**  
Criar o seu framework para utilizar no projeto combinando as melhores práticas, técnicas e metodologias.

**Figura Conceito Shu-Ha-Ri**

# Agenda

1. Visão 6ª Ed. Guia PMBOK

2. Seleção Ciclo de Vida

3. Princípios e pensamento ágil

4. Frameworks ágeis

5. Gestão de times ágeis

6. Iniciando projetos ágeis

7. Planejamento Adaptativo

8. Executando projetos ágeis

9. Monitorando e Controlando projetos ágeis

10. Práticas de melhoria contínua



# Gestão de Times Ágeis

As abordagens ágeis enfatizam a **liderança servidora como forma de empoderar as equipes**.

Liderança servidora é a **prática de liderar pelo serviço à equipe**, concentrando-se na compreensão e no atendimento das necessidades e no desenvolvimento dos membros da equipe, para permitir que esta **tenha o melhor desempenho possível**.

## Os líderes servidores abordam o trabalho:

- **Propósito**. Trabalhe com a equipe para definir o “por que” para se engajar ao objetivo do projeto
- **Pessoas**. Incentivar a equipe a criar um ambiente onde todos possam ter sucesso
- **Processo**. Buscar resultados, não importa como a equipe chama seu processo.

## Responsabilidades do líder servidor:

- **Facilita** colaboração da equipe
- **Removem** impedimentos organizacionais
- **Abrem caminhos** para a contribuição dos outros
- **Educa** as partes interessadas sobre o porque ser ágil
- Apoiar a equipe por meio da **mentoria**
- Ajuda a equipe com **atividade técnicas** de gerenciamento de projetos
- Comemore os **sucessos da equipe** e apoie atividades de construção de relacionamento com grupos externos.

O papel de gerente de projetos é um tanto desconhecido em um projeto ágil, pois **muitos frameworks e abordagens ágeis não incluem este papel**. Alguns praticantes acham que o papel de um GP não é mais necessário, pois as equipes autogerenciáveis assumiriam as responsabilidades de um GP. No entanto, praticantes ágeis percebem que os GP's podem agregar valor significativo em muitas situações. A principal diferença que seus papeis e responsabilidades **parecem um pouco diferentes**.



# Agenda

1. Visão 6º Ed. Guia PMBOK

2. Seleção Ciclo de Vida

3. Princípios e pensamento ágil

4. Frameworks ágeis

5. Gestão de times ágeis

6. Iniciando projetos ágeis

7. Planejamento Adaptativo

8. Executando projetos ágeis

9. Monitorando e Controlando projetos ágeis

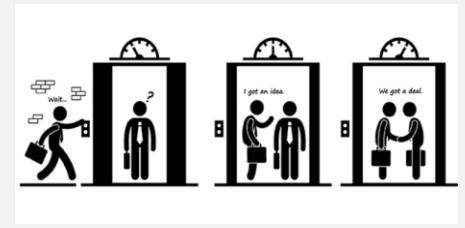
10. Práticas de melhoria contínua





# 5. Iniciando um projeto ágil

## 5.2 Elevator Statement



### Elevator Statement

- O Termo de Abertura também pode ser elaborado no formato **Elevator Statement** - é uma analogia à oportunidade que você teria de apresentar um projeto quando encontra um executivo no elevador, quando você teria, no máximo, 2 minutos para fazer a sua apresentação:

- Para (FOR)** – público alvo do projeto
- Quem (WHO)** – expectativas do público alvo
- O / A (THE)** – produto do projeto
- É um / uma (IS A)** – descrição objetiva do produto do projeto
- Que (THAT)** – objetivo do projeto
- Ao contrário de (UNLIKE)** – produto concorrente ou anterior
- Nós (WE)** – diferenciais do produto do projeto

### Exemplo:

**Para / FOR.** Qualquer pessoa.

**Quem / WHO.** Que queira aprender projetos de internet das coisas.

**O / A / THE** o curso ofertado pela Universidade Presbiteriana Mackenzie.

**É um / uma / IS A.** É um curso de 32 horas.

**Que / THAT.** Que explica internet das coisas e também projetos

**Ao contrário / UNLIKE.** Ao contrário da concorrência.

**Nós / WE.** Teremos palestrantes e profissionais de mercado que agregam valor a disciplina

Fonte: <http://frontroyaldailygrind.com/elevator-pitch-template.html/elevator-pitch-examples-teller-resume-sample-in-elevator-pitch-template>



# 5. Iniciando um projeto ágil

## 5.2 Team Charter

### Nomeação da equipe

- No contexto ágil, as equipes necessitam de **normas e entendimentos sobre como trabalhar em equipe**. Nesta situação configura o **Team Charter (Termo de Nomeação da equipe) ou Contrato Social**.

### Sugestões que podem compor um Team Charter (Social Contract)

- Valores da equipe**, como ritmo sustentável e horas ideais;
- Acordos de trabalho** como “Definição de Preparado” (DoR) e “Definição de Pronto” (DoD);
- Preservar o **timebox**;
- Uso dos limites do trabalho em progresso ou **Work in Progress (WIP)**;
- Regras básicas como uma pessoa **pode falar por vez nas reuniões**;
- Normas do grupo**, como a equipe lida com os horários da reunião.

**Team Canvas**

Version 0.8 | theteamcanvas.com | hello@theteamcanvas.com

Most important things to talk about in the team to make sure your work as a group is productive, happy and stress-free

Team name  Date

<b>PEOPLE &amp; ROLES</b> What are our names and the roles we have in the team?	<b>COMMON GOALS</b> What do we as a group really want to achieve? What is our key goal that is feasible, measurable and time-bounded?	<b>VALUES</b> What do we stand for? What are guiding principles? What are our common values that we want to be at the core of our team?	<b>RULES &amp; ACTIVITIES</b> What are the rules we want to introduce after doing this session? How do we communicate and keep everyone up to date? How do we make decisions? How do we execute and evaluate what we do?
<b>PURPOSE</b> Why are we doing what we are doing in the first place? What are we called as a team?		<b>NEEDS &amp; EXPECTATIONS</b> What each one of us needs to be successful? What are our personal needs towards the team to be at our best?	
<b>PERSONAL GOALS</b> What are our individual personal goals? Are there personal agendas that we want to open up?			
<b>STRENGTHS &amp; ASSETS</b> What are the skills we have in the team that will help us achieve our goals? What are interpersonal/soft skills that we have? What are we good at, individually and as a team?		<b>WEAKNESSES &amp; RISKS</b> What are the weaknesses we have, individually and as a team? What our teammates should know about us? What are some obstacles we see ahead of us that we are likely to face?	

Team: \_\_\_\_\_  
Date: \_\_\_\_\_

Fonte: <http://theteamcanvas.com/>

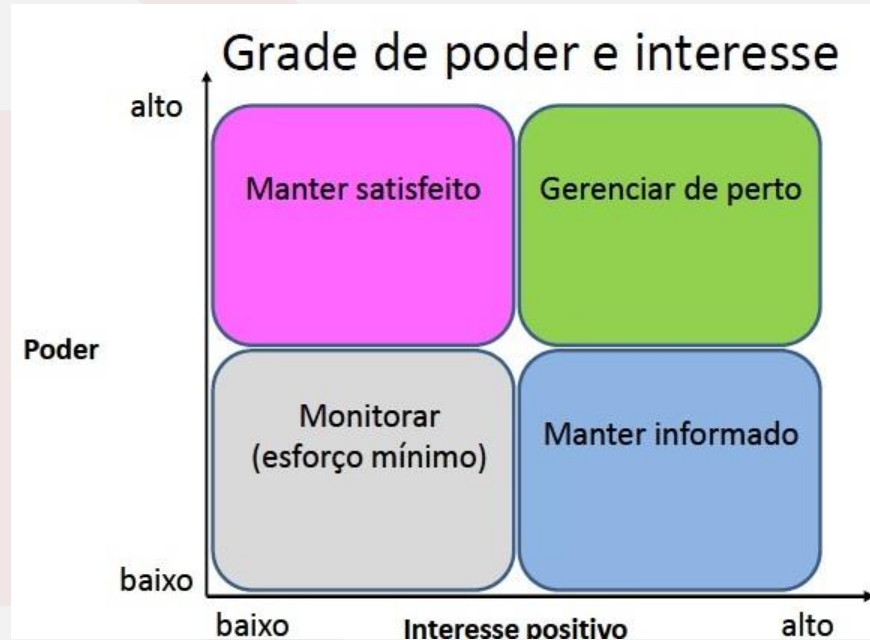
This work is licensed under the Creative Commons Attribution-Share Alike 4.0. To view a copy of this license, visit: <http://creativecommons.org/licenses/by-sa/4.0/>

## 5. Iniciando um projeto ágil

### 5.3 Identificar as partes interessadas

#### Identificar as partes interessadas

- Processo executado na fase inicial do projeto e é uma das maiores causas de fracasso de projeto quando não é feito corretamente.



# Agenda

1. Visão 6ª Ed. Guia PMBOK

2. Seleção Ciclo de Vida

3. Princípios e pensamento ágil

4. Frameworks ágeis

5. Gestão de times ágeis

6. Iniciando projetos ágeis

7. Planejamento Adaptativo

8. Executando projetos ágeis

9. Monitorando e Controlando projetos ágeis

10. Práticas de melhoria contínua

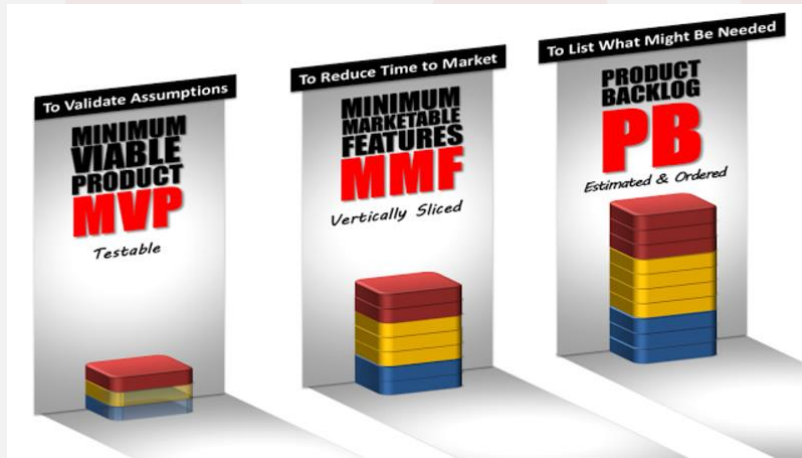


# 6. Planejando Projetos Ágeis

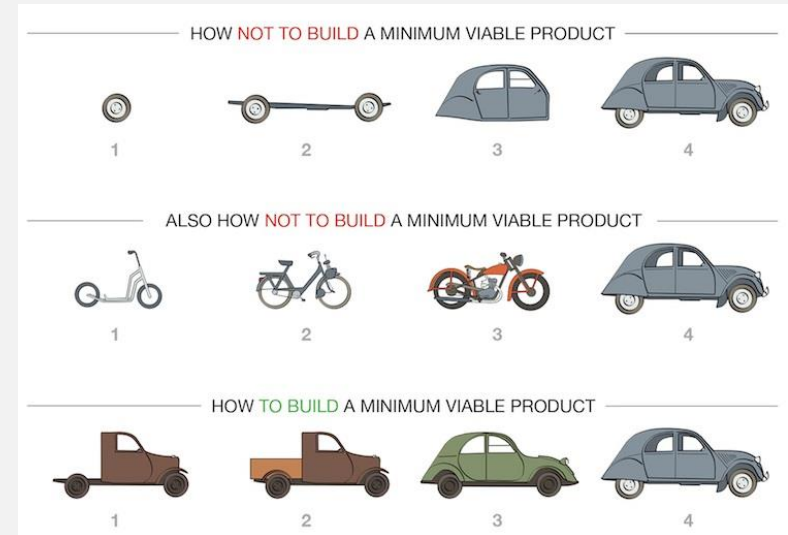
## 6.1 Conceitos de Planejamento Adaptativo

### *Minimally Marketable Feature (MMF) / Minimum Viable Product (MVP)*

- *Minimally Marketable Feature (MMF)* significa **mínima funcionalidade comercializável**, também é conhecido como *Minimum Viable Product (MVP)*, ou seja **Mínimo Produto Viável**;
- Representa o conjunto mínimo de funcionalidades que **torna o produto utilizável pelo cliente**, embora **não represente o produto final a ser gerado pelo projeto**.



Fonte: <http://tracks.roojoom.com/r/89204>



Fonte: <https://sugoiilabs.com/blog/startups/framework-to-build-an-mvp-minimum-viable-product>

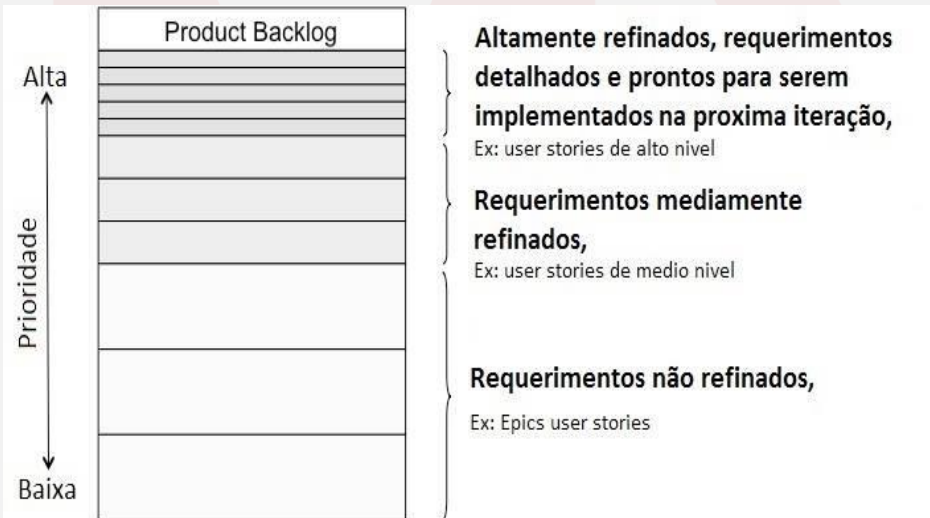


# 6. Planejando Projetos Ágeis

## 6.1 Backlog do Produto

### Backlog do Produto ou Product Backlog

- É uma **lista ordenada, criada, mantida e refinada pelo Product Owner** dos requisitos do produto apresentado na forma de **user stories** para uma equipe. Em sua criação, ele é composto por **épicos (histórias grandes e sem detalhes)**, que mais tardes serão detalhados e refinados.
- **Não é necessário criar todas as histórias para todo o projeto antes do início do trabalho.**



### Exemplo: Product Backlog

Funcionalidade	Benefício	Penalidade	Business_Value	Estimativa	Complexidade
Func_01	1	4	5	5	1,0
Func_02	2	1	3	6	0,5
Func_03	3	5	8	2	4,0
Func_04	4	4	8	8	1,0
Func_05	2	1	3	1	3,0

Fonte: <https://millenniumnetworkblog.wordpress.com/2014/05/06/product-owner-e-a-aplicacao-da-tecnica-de-beneficios-e-complexidade-na-priorizacao-do-product-backlog/>



# 6. Planejando Projetos Ágeis

## 6.2 Coleta de Requisitos

### User Stories

- *User Stories* são **textos simples que descrevem uma funcionalidade**. Geralmente são escritas em pequenos cartões ou em *post its*, para que o autor da **user story** seja **objetivo**.

São escritas no formato:

- **Como** <ator>
- **Quero** <ação>
- **Para** <resultado ou benefício>
- Deve atender as características do acrônimo **INVEST**
- Deve ter os **critérios de aceite** escritos em seu verso

Letra	Significado	Descrição
I	Independent (Independente)	A história de usuário deve ser auto-suficiente, de uma forma que não há nenhuma dependência inerente em outra história de usuário.
N	Negotiable (Negociável)	Histórias de usuários, até que elas sejam parte de uma iteração, podem sempre ser alteradas ou reescritas.
V	Valuable (Valiosa)	Uma história de usuário deve entregar valor para o usuário final.
E	Estimable (Estimável)	Você deve sempre ser capaz de estimar o tamanho de uma história de usuário.
S	Small (Pequena)	Histórias de usuários não devem ser tão grandes a ponto de se tornar impossível de planejar/priorizar com um certo nível de certeza.
T	Testable (Testável)	A história de usuário ou a sua descrição relacionada devem fornecer as informações necessárias para tornar o teste de desenvolvimento possível.

### Exemplo de USER STORY

Como um vendedor,  
gostaria de solicitar um  
percentual de desconto para  
efetuar uma venda.

O desconto não pode ser igual ou superior ao valor total da venda.

A venda deve ficar com status pendente.

Deverá ser enviado automaticamente um e-mail solicitando aprovação de um gerente.

Fonte <https://pt.slideshare.net/manoelp/exemplos-de-user-stories>



## 6. Planejando Projetos Ágeis

### 6.4 Estimativas Ágeis

#### Planning Poker

- Os membros da equipe **estimam o esforço das *user stories* em *story points*** por meio de um jogo de **baralho**;
- Evita indução ou coerção nas estimativas, uma vez que todos os **membros exibem suas estimativas ao mesmo tempo**;
- A estimativa não é feita individualmente, mas de **forma consensual**;
- É realizada uma **discussão entre a equipe e o cliente sobre cada requisito**, gerando entendimento e estimativas mais precisas.



Fonte: [https://danielettinger.files.wordpress.com/2015/07/img\\_2627.png](https://danielettinger.files.wordpress.com/2015/07/img_2627.png)





# Agenda

1. Visão 6º Ed. Guia PMBOK

2. Seleção Ciclo de Vida

3. Princípios e pensamento ágil

4. Frameworks ágeis

5. Gestão de times ágeis

6. Iniciando projetos ágeis

7. Planejamento Adaptativo

8. Executando projetos ágeis

9. Monitorando e Controlando projetos ágeis

10. Práticas de melhoria contínua

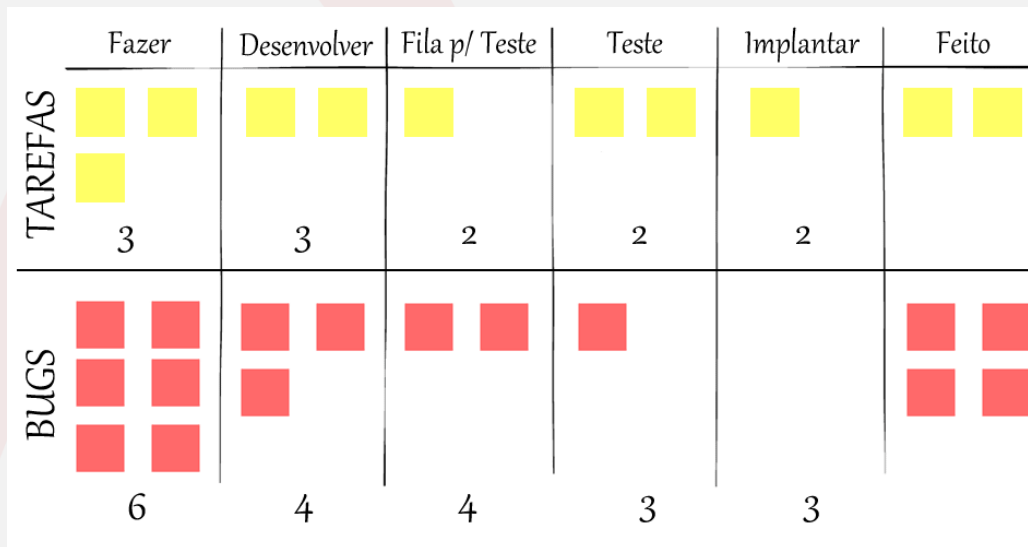


# 7. Executando Projetos Ágeis

## 7.1 Entregando Valor

### Kanban / Quadro de tarefas

- Visibilidade do **fluxo e progresso** do trabalho;
- Identificar **impedimentos**;
- Promover a **interatividade** da equipe;
- Atuar como **irradiador de informações**, deixando-as transparentes.



<https://www.culturaagil.com.br/kanban-do-inicio-ao-fim/>

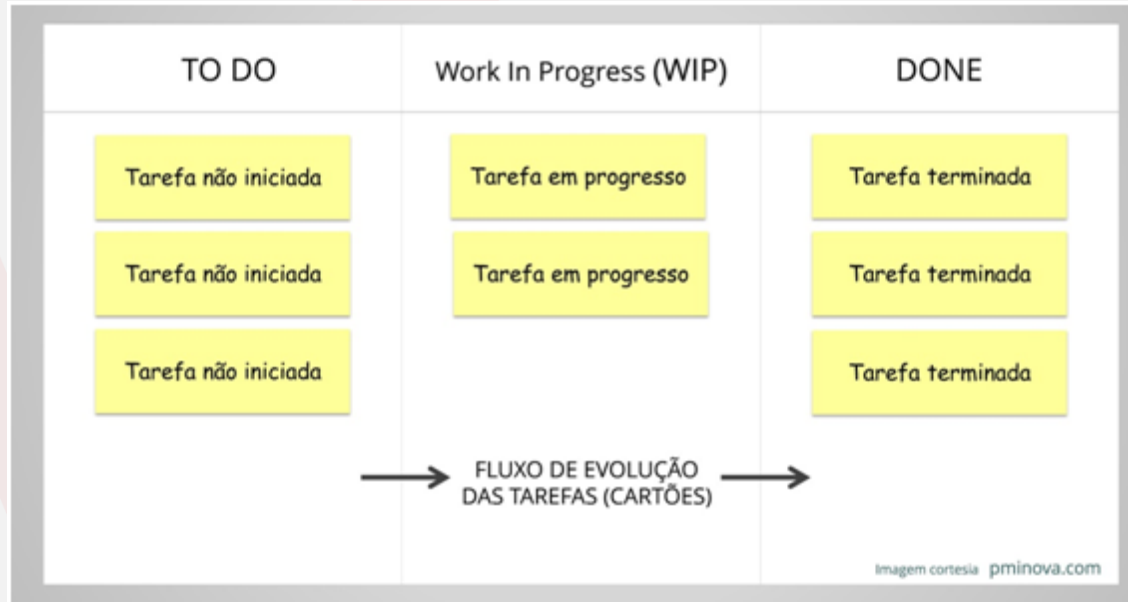


# 7. Executando Projetos Ágeis

## 7.1 Entregando Valor

### Limites de WIP (Work in Progress)

- Encontrar o **número ideal de WIP máximo por cada etapa**, visando **otimizar** o rendimento do trabalho e propiciar uma **melhor entrega de valor**.



<https://www.devmedia.com.br/kanban-4-passos-para-implementar-em-uma-equipe/30218>



# 7. Executando Projetos Ágeis

## 7.2 Engajamento das partes interessadas

### Engajamento das partes interessadas

- **Partes interessadas.** Pessoas ou organizações que **serão impactadas pelo resultado de um projeto** e poderão ter **influência negativa ou positiva** no projeto.

### Planejar o gerenciamento das partes interessadas

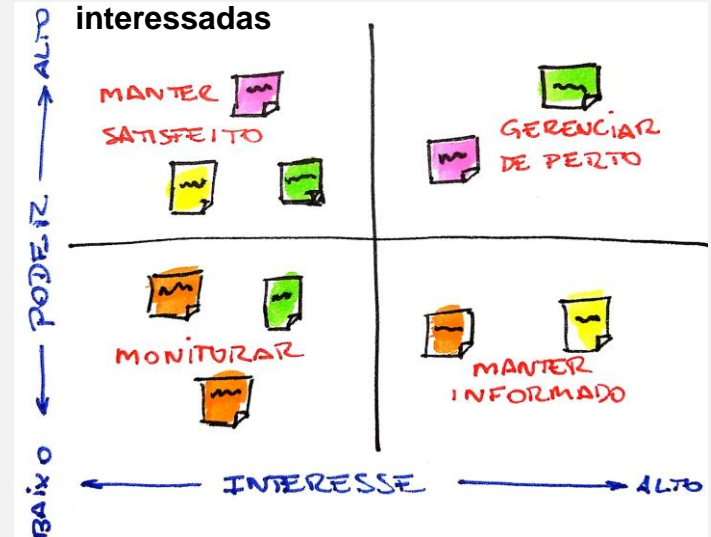
- **Tipo do perfil.** Desinformado, resistente, neutro, apoiador ou líder.
- **Forma e periodicidade de comunicação.** Definir **formato** (e-mail, relatório, etc.) e **periodicidade** (diário, quinzenal, etc.)

### Gerenciar o engajamento das partes interessadas

- **Comunicação transparente.** Dar transparência em tudo o que faz
- **Escuta ativa.** Olhares, empatia, tom de voz, expressão, etc. (comunicação não verbal).

**Controlar o nível de engajamento das partes interessadas.**  
**Engajar, sempre que possível, todas as partes interessadas.**

Figura – Poder x Interesse partes interessadas



Fonte: [https://www.difusaogp.com.br/artigos/scrum-agil/attachment/novo-documento-2017-10-04\\_2/](https://www.difusaogp.com.br/artigos/scrum-agil/attachment/novo-documento-2017-10-04_2/)

# 7. Executando Projetos Ágeis

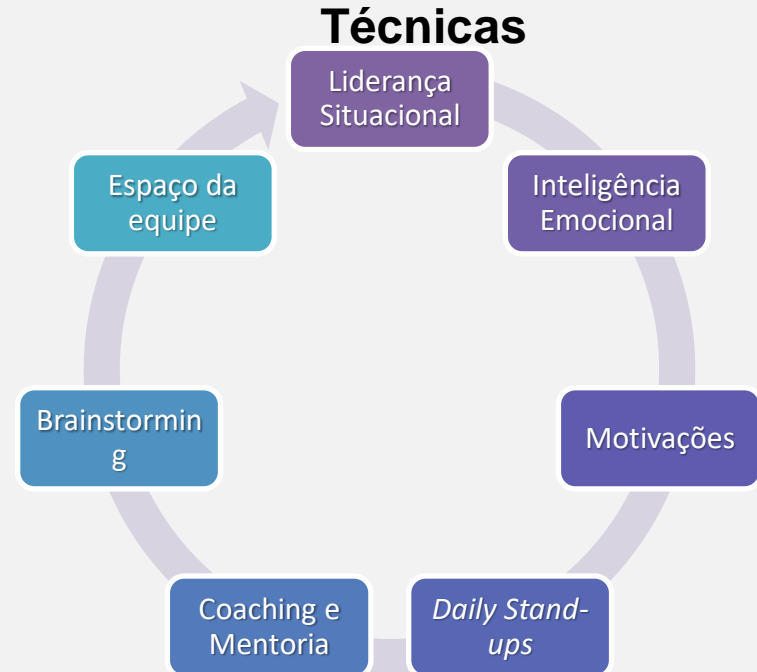
## 7.4 Práticas de Aumento de Desempenho da Equipe

### Aumento do desempenho da equipe

- Aplicar **técnicas** de **gerenciamento** e **liderança** para aumentar o **desempenho** da equipe.

Gerenciamento	Liderança
Foco em tarefas	Foco em pessoas
Controle da equipe	Fortalecimento da equipe
Foco em eficiência	Foco em efetividade
Fazer certo as coisas	Fazer as coisas certas
Foco em execução	Foco em direcionamento
Foco em práticas	Foco em princípios
Prioriza o comando	Prioriza a comunicação

Fonte: Guia PMBOK, 6. ed.;



# Agenda

1. Visão 6ª Ed. Guia PMBOK

2. Seleção Ciclo de Vida

3. Princípios e pensamento ágil

4. Frameworks ágeis

5. Gestão de times ágeis

6. Iniciando projetos ágeis

7. Planejamento Adaptativo

8. Executando projetos ágeis

9. Monitorando e Controlando projetos ágeis

10. Práticas de melhoria contínua



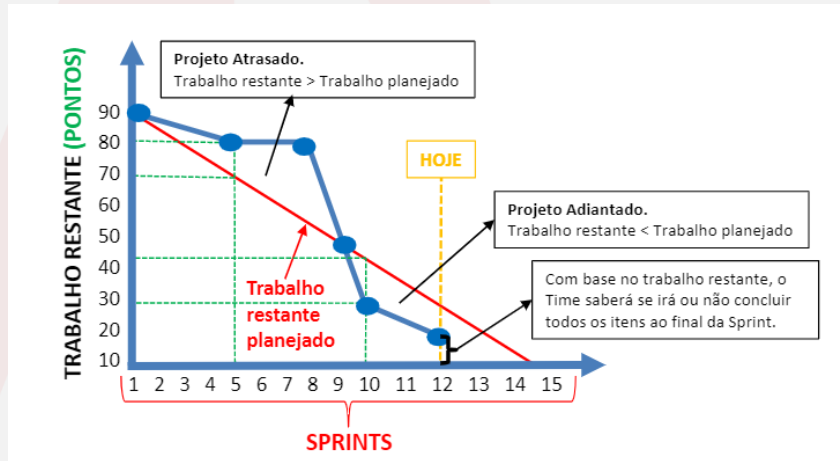
# 8. Monitoramento e Controlando Projetos Ágeis

## 8.1 Controlando Tempo e Escopo

### Gráfico de *Burndown*

- O Gráfico de *Burndown* exibe o **esforço remanescente na linha do tempo** para **finalizar** o projeto, o release ou a iteração. É o trabalho restante, o que falta fazer.

### Exemplo - Gráfico de Burndown



Fonte: <http://docplayer.com.br/82302995-Treinamento-scrum-dia-2.html>



## 8. Monitoramento e Controlando Projetos Ágeis

### 8.2 Controlando Tempo e Escopo

#### Gráfico de *burnup*

- O **Gráfico de *Burnup*** exibe o esforço do projeto, release ou iteração que **já foi concluído** na linha do tempo.

#### Exemplo - Gráfico de *burnup*



Fonte: <https://dionatanmoura.com/o-mantra-da-productividade/gestao-de-tarefas/grafico-de-burnup/>



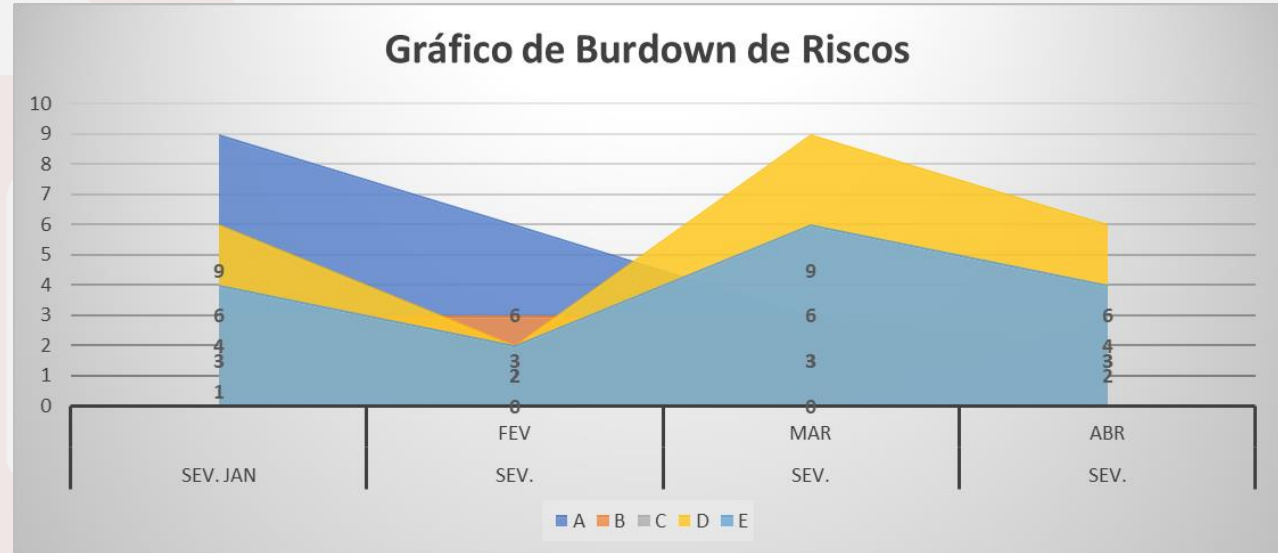
## 8. Monitoramento e Controlando Projetos Ágeis

### 8.3 Controlando Riscos

#### Gráfico de *Burdown* de riscos

- Permitem **monitorar se os riscos e suas severidades** estão **diminuindo** ou **aumentado** no decorrer do tempo.

Risco	Sev. Jan	Sev. Fev	Sev. Mar	Sev. Abr
A	9	6	3	2
B	3	3	3	3
C	1	0	0	3
D	6	2	9	6
E	4	2	6	4



# 8. Monitoramento e Controlando Projetos Ágeis

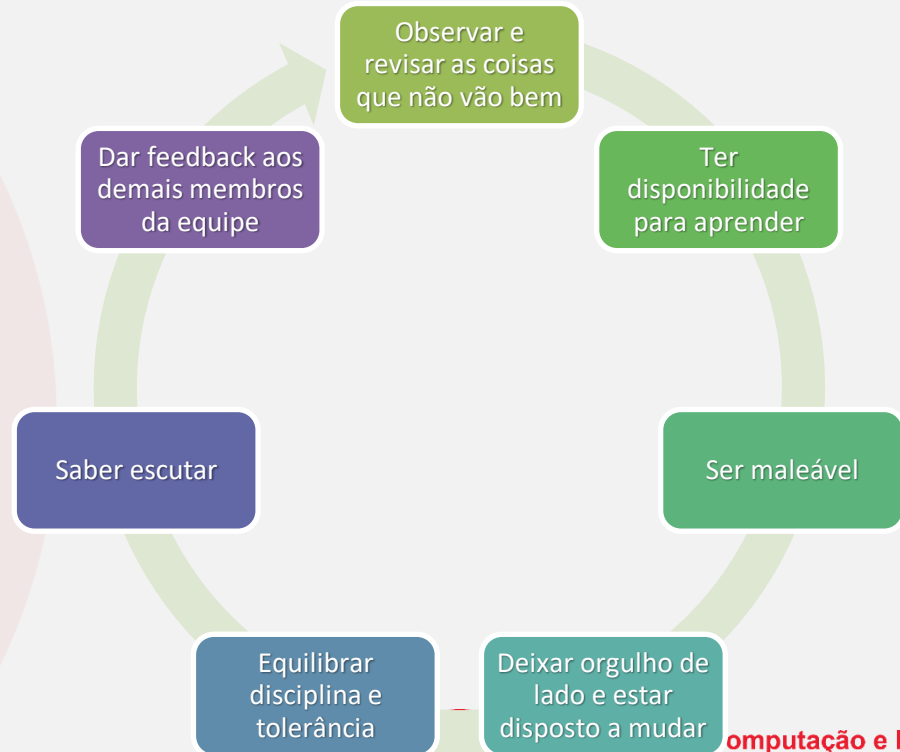
## 8.4 Controlando a Qualidade

### Fator Humano

- As pessoas **cometem enganos** e isso deve ser **gerenciado**.



### Atitudes



# Agenda

1. Visão 6º Ed. Guia PMBOK

2. Seleção Ciclo de Vida

3. Princípios e pensamento ágil

4. Frameworks ágeis

5. Gestão de times ágeis

6. Iniciando projetos ágeis

7. Planejamento Adaptativo

8. Executando projetos ágeis

9. Monitorando e Controlando projetos ágeis

10. Práticas de melhoria contínua



# 9. Práticas de Melhoria Contínua

## 9.1 Retrospectivas

### Retrospectivas

- São **reuniões** que ocorrem ao **final de cada iteração**, onde a equipe do projeto identifica como **melhorar seus processos**;
- Retrospectivas são **reuniões de lições aprendidas**.

Responde as 3 perguntas:

### O que é uma Retrospectiva



Fonte: Helica Karen, Retrospectivas; helica@prodam.am.gov.br

Fonte <https://pt.slideshare.net/kfranklint/gesto-agil-de-tudo-retrospectivas>

O que foi bom?	O que pode ser melhorado?	Quem está no controle?

Fonte: <https://pt.slideshare.net/gustavoqz/participao-do-time-de-teste-em-projetos-scrum-2066606>



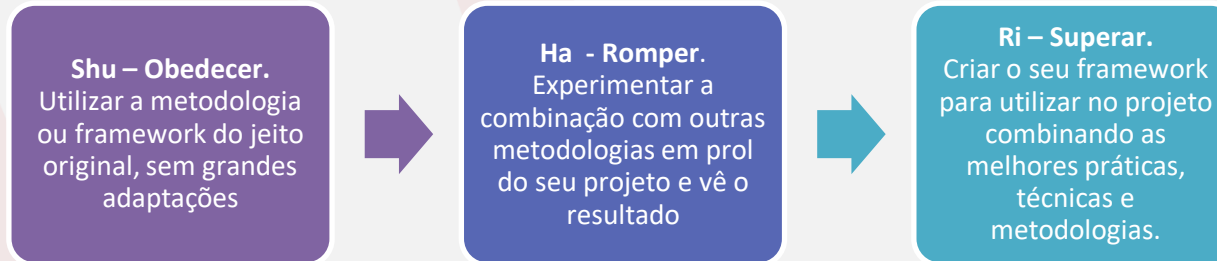
## 9. Práticas de Melhoria Contínua

### 9.5 Aplicando novas práticas ágeis

#### Novas práticas ágeis

- Criar uma **nova prática ágil**;
- Questões a serem respondidas na criação:
  - A nova prática resolverá a causa-raiz que motivou a sua criação?
  - Realmente trará benefícios ao projeto ou trata-se de um benefício pessoal?

#### Figura Conceito Shu-Ha-Ri



Lembre-se dos conceitos do *Shu-ha-ri* e comece a **aplicar a nova prática ágil aos poucos**. Encare a aplicação da prática como um *spike* (prova de conceito), use uma **iteração como experimento e revise os efeitos colaterais** na reunião de retrospectiva.

# **Curso: Aprendizagem de Máquina em Inteligência Artificial**

**Disciplina: Internet das Coisas**

**Aula: Modelos Ágeis de Projetos**

**Obrigada!**

**Profa. Dra. Mônica Mancini, PMP, ASF**

