

PROCESSO DE DESENVOLVIMENTO DE SOFTWARE ÁGIL

Prof. Dr. Ivanir Costa
TITULAR DO PROGRAMA DE MESTRADO E DOUTORADO EM INFORMÁTICA E
GESTÃO DO CONHECIMENTO (PPGI)
DA UNINOVE
2020

História

Na opinião dos autores e defensores (Agilistas), os Métodos Ágeis originaram-se (a partir de 1990); pois

Viam os modelos clássicos como o **cascata** como extremamente burocrático e lento;

Contraditórios em relação a forma usual com que os Engenheiros de Software sempre realizaram seus trabalhos com eficiência, no passado.

História

Em **2001**, membros proeminentes da comunidade de software se reuniram em Snowbird ¹ e adotaram o nome, para os seus métodos de trabalho como ***métodos ágeis***.

Dessa reunião nasceu o ***Manifesto Ágil***:

- Documento que reúne os valores, princípios e práticas destes processos de desenvolvimento de software.

1 – The Lodge at Snowbird ski resort, Utah, fevereiro 2001

História

Visão do mercado

Os Métodos Ágeis são algumas vezes **caracterizados como o oposto** de metodologias guiadas pelo planejamento ou denominadas disciplinadas ou preditivas (baseadas no planejamento).

Estudo do mercado atual

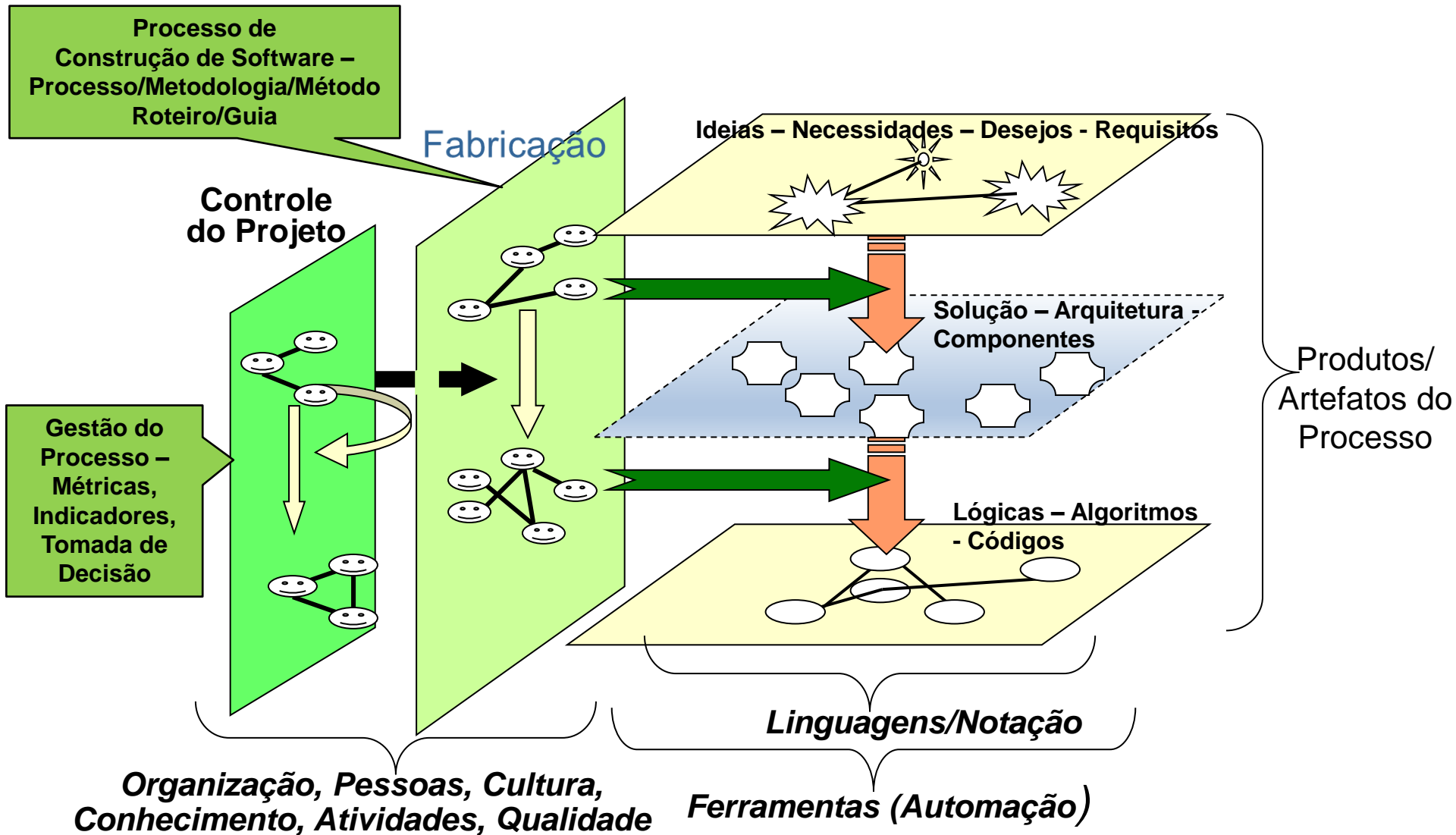
MÉTODO	Sucesso	Desafios	Fracasso
Ágil	42%	50%	8%
Cascata	26%	53%	21%

Relatório do Standish Group (2014-2018)

- O Estudo mostra que os projetos de desenvolvimento de software com processos/métodos ágeis têm 42% de sucesso, enquanto os projetos em cascata possuem apenas 26% de chances de sucesso.
- Os projetos com métodos ágeis apresentam somente 8% de fracassos, enquanto os projetos desenvolvidos com o processo Cascata apresentaram 21% de fracassos.
- O estudo reforça a importância que os métodos ágeis vêm obtendo no mercado de desenvolvimento de software, a nível mundial e no Brasil.

PROCESSO DE DESENVOLVIMENTO DE SOFTWARE ÁGIL

MODELO DE PROCESSO DE SOFTWARE



Será que o processo/método ágil atende essa estrutura clássica?

Manifesto Ágil

O que é?

O manifesto ágil é uma **declaração de valores e princípios** que fundamentam o desenvolvimento ágil de software.

Manifesto para o Desenvolvimento Ágil de Software

Publicado por 17 profissionais proeminentes no uso de métodos mais leves de desenvolvimento de software (2001).

- Composto por **4 valores e 12 princípios**:

Os **valores** são:

1. Indivíduos e Interações **mais que (ou acima de)** processos e ferramentas
2. Software em Funcionamento **mais que (ou acima de)** documentação abrangente
3. Colaboração com o Cliente **mais que (ou acima de)** negociação de Contratos
4. Responder as Mudanças **mais que (ou acima de)** seguir um plano

Manifesto para o Desenvolvimento Ágil de Software

Os 12 princípios do Manifesto Ágil



Manifesto para o Desenvolvimento Ágil de Software

Os 12 princípios do Manifesto Ágil



1. A prioridade é satisfazer o cliente, através da entrega adiantada e contínua de **software de valor**.



2. Aceitar **mudanças de requisitos**, mesmo durante o desenvolvimento, para que o cliente possa tirar vantagens competitivas).

Manifesto para o Desenvolvimento Ágil de Software

Os 12 princípios do Manifesto Ágil

3 Entregas frequentes



3. Entregar software **funcionando**, na escala de semanas até meses, com preferência para os períodos mais curtos.

4 Trabalho em equipe



4. Pessoas relacionadas à negócios e desenvolvedores devem **trabalhar em conjunto e diariamente** durante todo o curso do projeto.

Manifesto para o Desenvolvimento Ágil de Software

Os 12 princípios do Manifesto Ágil

5 Confie e apoie



5. Construir projetos ao redor de **indivíduos motivados dando a eles** o ambiente e suporte necessários, e confiar que farão o seu trabalho.

6 Conversas face a face



6. O método mais eficiente e eficaz de transmitir informações para, e por dentro de um time de desenvolvimento, é através de uma **conversa cara a cara (face-to-face)**.

Manifesto para o Desenvolvimento Ágil de Software

Os 12 princípios do Manifesto Ágil

7 Software funcionando



7. **Software funcional** é a medida primária de progresso.

8 Desenvolvimento sustentável



8. Processos ágeis promovem um **ambiente sustentável** (Os patrocinadores, desenvolvedores e usuários, devem ser capazes de manter, indefinidamente passos constantes).

Manifesto para o Desenvolvimento Ágil de Software

Os 12 princípios do Manifesto Ágil



9. **Contínua atenção** à excelência técnica e bom design, aumentam a agilidade.



10. **Simplicidade**, a arte de maximizar a quantidade de trabalho que não precisou ser feito.

Manifesto para o Desenvolvimento Ágil de Software

Os 12 princípios do Manifesto Ágil



11. As melhores arquiteturas, requisitos e designs **emergem** de times auto-organizáveis.



12. Em **intervalos regulares**, o time reflete em como ficar mais efetivo, se ajustam e otimizam seu comportamento de acordo.

Desenvolvimento Ágil de Software

A maioria dos métodos ágeis tenta minimizar o risco do desenvolvimento do software usando **ciclos de curtos períodos**, chamados de **iteração ou sprints**, os quais tipicamente vão de poucos dias, uma semana, e até quatro semanas de duração.

Desenvolvimento Ágil de Software

Cada **iteração (sprint)** é como um projeto de software em miniatura e inclui todas as tarefas necessárias para implantar o mini incremento - nova(s) funcionalidade(s) ou melhoria(s):

- Atividades: planejamento, análise de requisitos, design, codificação, teste e documentação.

Os principais métodos ágeis da atualidade

XP - eXtreme Programming,

SCRUM,

Kanban

Lean Software

TDD - Test Driven Development

FDD - Feature Driven Development,

Crystal Clear,

DSDM - Dynamic Systems Development Method,

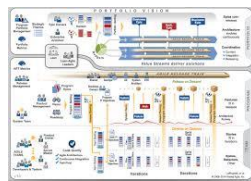
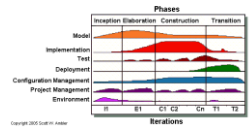
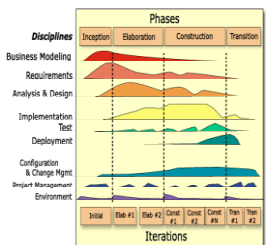
AUP – Agile Unified Process,

ICONIX

HISTÓRIA - EVOLUÇÃO DOS PROCESSOS DE SOFTWARE



RUP



SAFe

Lean Software Development
An Agile Toolkit

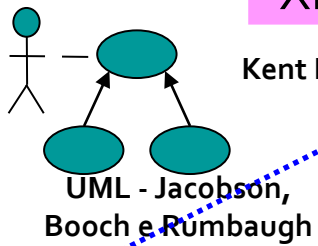
Ken Schwaber
Cockburn

Série ISO para software
(25110)

Manifesto Ágil

XP

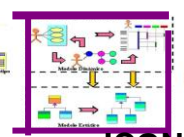
Kent Beck



UML - Jacobson,
Booch e Rumbaugh



SCRUM / KANBAN



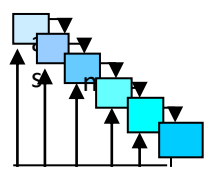
ICONIX



SEI

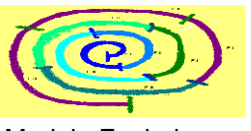
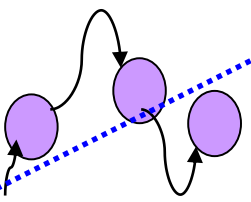


Spotify/
Squads

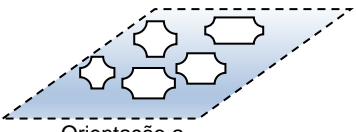


Cascade

Estruturado



Modelo Espiral



Orientação a
Objetos-Componentes

1970 ▶▶▶ 1980 ▶▶▶ 1990 ▶▶▶ 1995 ▶▶▶ 2000 ▶▶▶ 2010 ▶▶▶ 2020

Gane, Tom de
Marco, Palmer &
Mc Menamim
Edward Yourdon

Método Ágil XP (eXtreme Programming)

XP (eXtreme Programming)

XP é uma disciplina **leve** do desenvolvimento de software baseada nos princípios:

- 1.Simplicidade,
- 2.Comunicação,
- 3.Feedback,
- 4.Coragem

XP é desenhada para uso em times pequenos e para quem necessita desenvolver software rapidamente num ambiente de mudanças constantes de requisitos.

XP (eXtreme Programming)

De acordo com Kent Beck, o sucesso metodológico do método ágil XP vem do esforço na satisfação do cliente.



13 de julho de 2006 — Kent Beck gives a talk about feeling at ease in a technical position. Sponsored by Agitar Software -

<http://www.youtube.com/watch?v=JqysHIQHxlg&feature=related>

XP (eXtreme Programming)

O método ágil XP enfatiza o **trabalho em grupo**.

Gerentes, clientes e desenvolvedores **são parte** do time dedicado a entregar um software de qualidade.

Segundo o XP existe uma limitação na definição do grupo, sendo possível um grupo de no máximo **dez pessoas**, incluindo os gerentes e clientes.

12 práticas do XP (eXtreme Programming)

P1 - O processo de planejamento

Jogo do Planejamento

P2 - Pequenas versões

ciclo muito curto

P3 – Metáfora

nome comum e uma linguagem comum

P4 - Design simples

Não existe mais o “construir para o futuro”

P5 - Testes

validação do software durante todo o tempo.

P6 - Desenvolvimento em espiral (ciclos)

Práticas do XP (eXtreme Programming)

P7 - Programação em pares

validação do software durante todo o tempo

P8 - Propriedade coletiva

Todo o código pertence a todos da equipe

P9 - Integração contínua

integram software muitas vezes ao dia

P10 - 40 horas de trabalho semanal

Programadores cansados cometem mais erros

P11 - Cliente fazendo parte da equipe

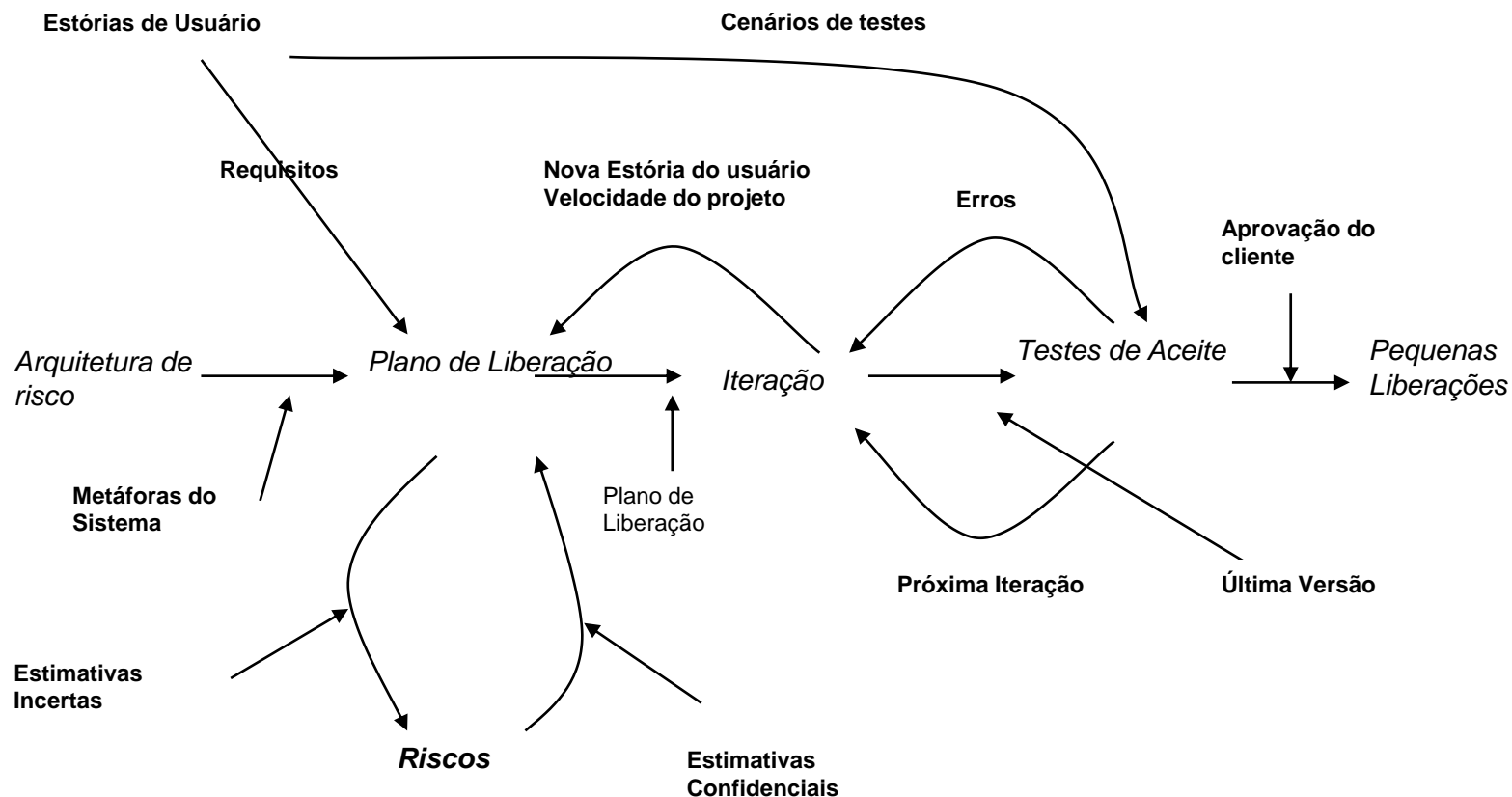
Um projeto XP é conduzido por um cliente

P12 - Codificação padrão

os códigos são escritos da mesma forma

XP (eXtreme Programming)

Modelo do Processo XP



Framework Ágil Scrum

Framework (Método Ágil Scrum)

Inicialmente, o Scrum foi concebido como um estilo de gerenciamento de projetos em empresas de fabricação de automóveis e produtos de consumo (Toyota), por Takeuchi e Nonaka (1986).

O **Scrum** é utilizado como um *framework* Ágil para projetos de Software desde a década de 1990.

Framework (Método Ágil Scrum)

Jeff Sutherland, John Scumniotales e Jeff McKenna

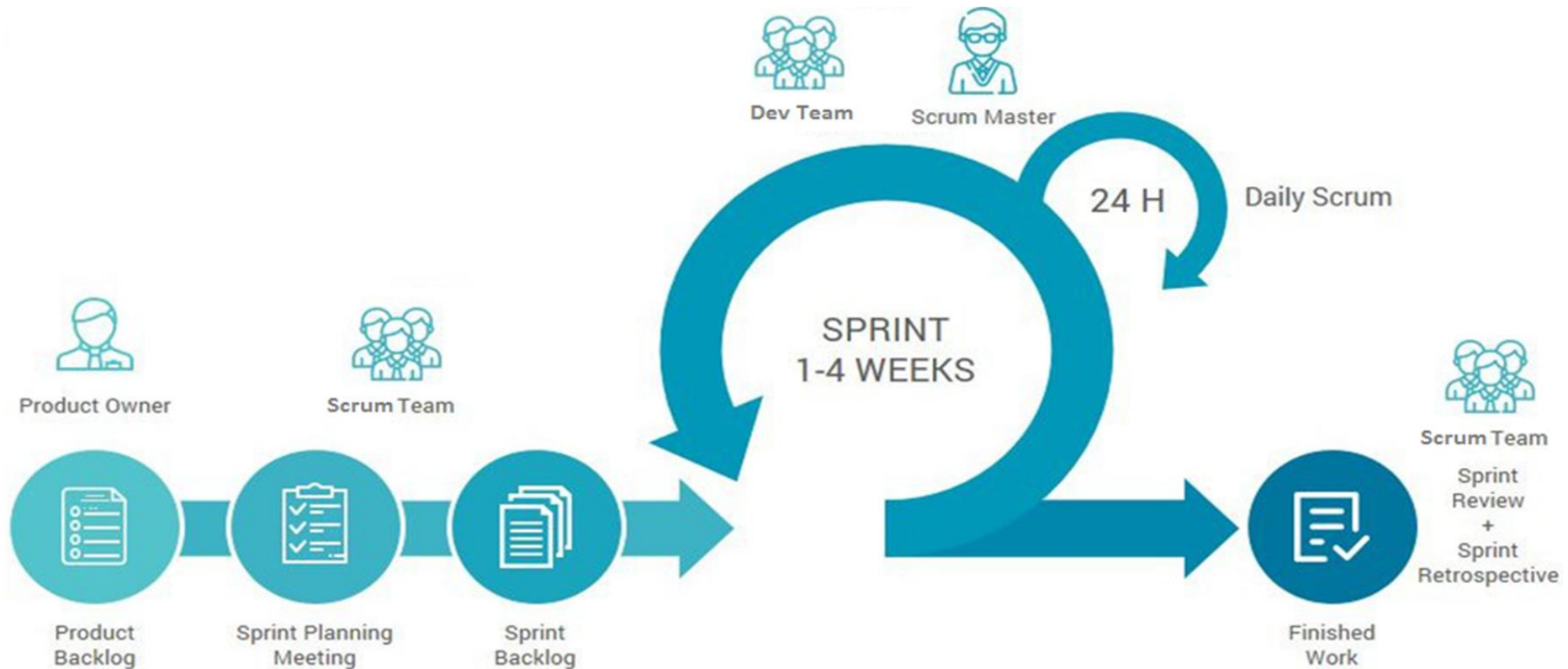
- Conceberam, documentaram e implementaram o SCRUM na empresa Easel Corporation em 1993;
- Incorporando os estilos de gerenciamento observados por Takeuchi e Nonaka na Toyota ao desenvolvimento de projetos de software.

Framework (Método Ágil Scrum)

Em 1995, Ken Schwaber formalizou a definição de *SCRUM* e ajudou a implantá-lo no desenvolvimento de softwares ágeis em todo o mundo.

[illegible]

Framework Scrum – Visão da Estrutura



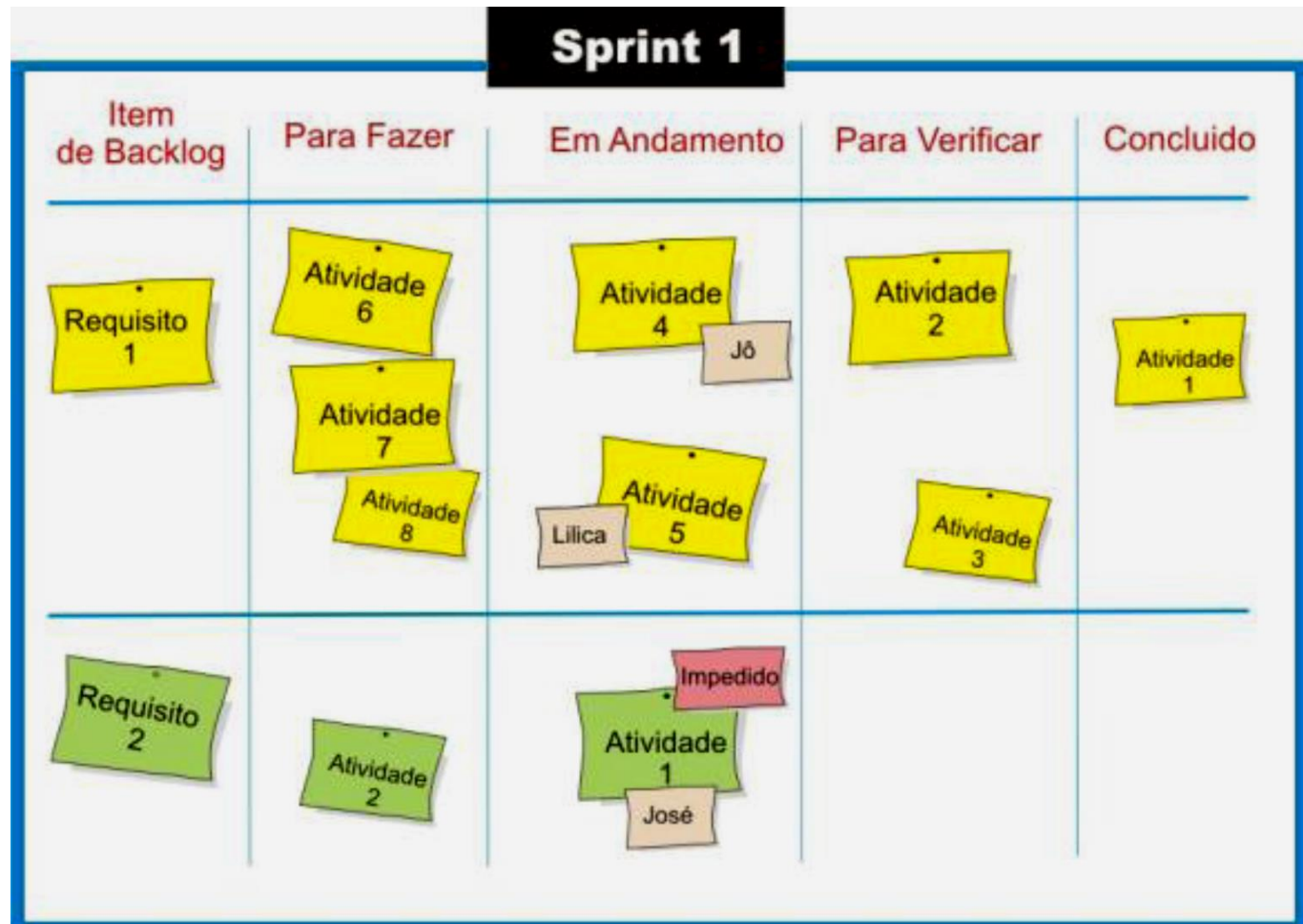
ESTRUTURA SCRUM

Papeis - PO (Product Owner), SM (Scrum Master), ST (Scrum Master), DT (DevTeam)

Eventos/Cerimônias – SPM (Sprint Planning Meeting), DS (Daily Scrum), SR (Sprint Review), SR (Sprint Retrospective)

Artefatos - PB (Product Backlog), SB (Sprint Backlog), Histórias, Tarefas

Quadro Daily Scrum (adaptação do quadro Kanban) – Gestão visual do projeto



Os Itens/Requisitos são as histórias (ou estórias) do usuário e as atividades também são as tarefas do Sprint – As atividades ou tarefas são estimadas e alocadas às pessoas do time Scrum. As cores dos post-its podem indicar situações das tarefas

MÉTODO KANBAN

MÉTODO KANBAN

Criado por Taiichi Ohno na década de 1940, para ser aplicado nas linhas de produção da Toyota.

Criado como um método de planejamento simples, com o objetivo de controlar e gerenciar a mão de obra e inventário (recursos) durante todos os estágios de produção de forma otimizada.

A necessidade de desenvolver o método Kanban surgiu dos **índices inadequados de produtividade e eficiência** da Toyota comparados aos seus rivais americanos.

MÉTODO KANBAN

Com o Kanban a Toyota atingiu a **flexibilidade. a eficiência e aumentou a produtividade**, além de causar redução de custos em todos os processos de trabalho envolvidos.

O método busca controlar toda a linha de produção, do fornecedor ao consumidor.

Através dele, pode-se evitar interrupções no fornecimento, assim como excesso de estoque de mercadorias em vários estágios do processo de produção.

MÉTODO KANBAN

Pode auxiliar na gestão de projetos, ao identificar processos que precisam de mais tempo para serem realizados, evitando-se paradas de membros da equipe aguardando atividades de outros serem concluídas.

A gestão de pessoas também pode ser afetada pelo Kanban, ao otimizar a produtividade da equipe em diversos projetos ao mesmo tempo, otimizando tempo, reduzindo custos diretos e indiretos, e, conseqüentemente, aumentando o lucro real do projeto.

MÉTODO KANBAN

David J. Anderson foi o primeiro a usar este método para processos de desenvolvimento de softwares em 2004.

Em 2010, escreveu o livro “[Kanban: Successful Evolutionary Change for Your Technology Business](#)”, que é considerado o guia mais abrangente para usar o método em processos de trabalho de conhecimento.

David não considera o **Kanban** como uma metodologia do ciclo de vida do projeto, ou de gerenciamento de projetos em si.

Ele exige que alguns processos de trabalho já estejam acontecendo, para então aplicar o método e seguir alterando e otimizando gradualmente os processos do fluxo de trabalho existentes.

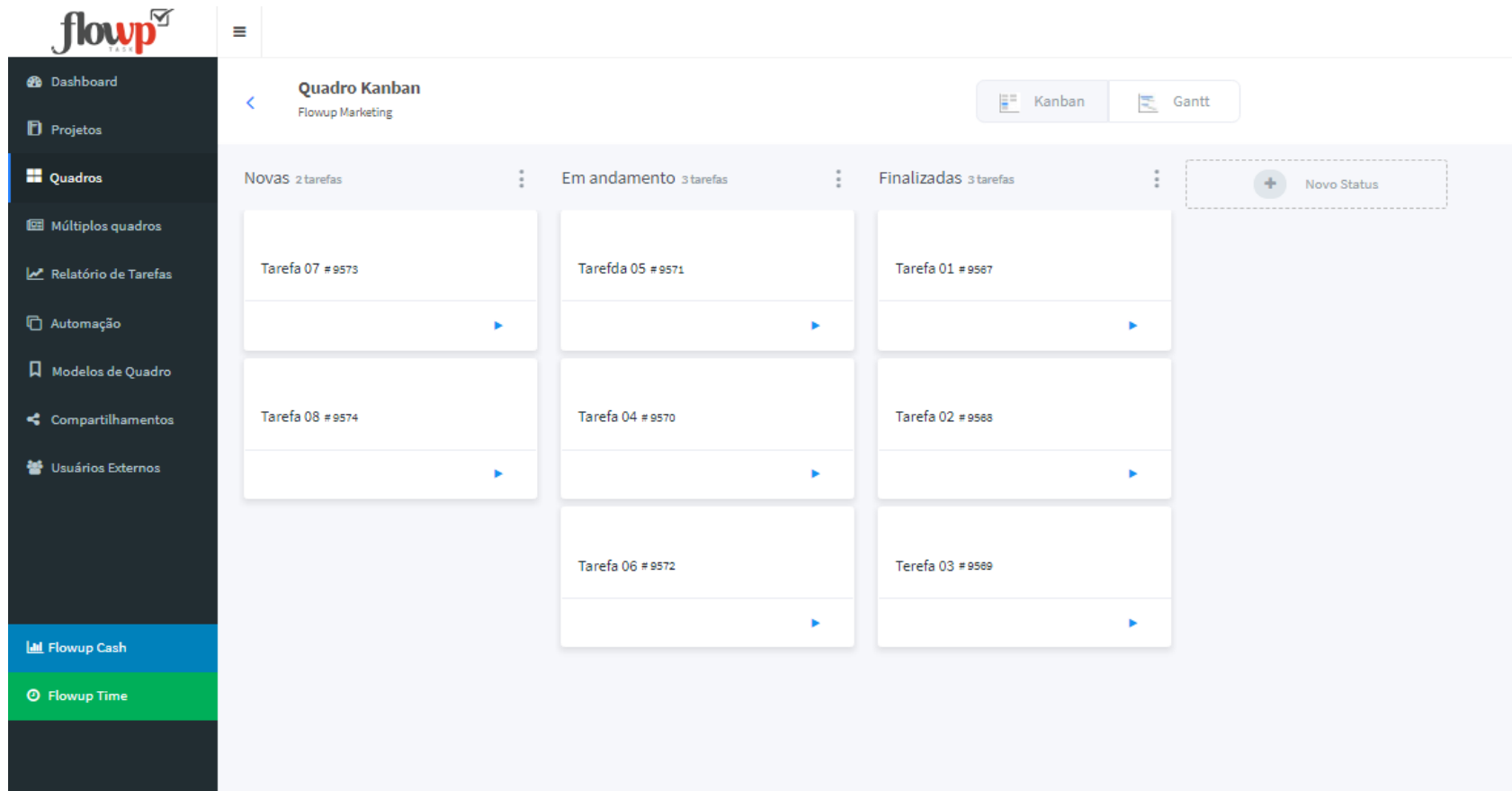
MÉTODO KANBAN

O primeiro passo é criar o quadro Kanban para visualizar o fluxo de trabalho.

Para isso basta

- **Um quadro branco e alguns postits**, caso deseje usar um quadro físico;
- Ou pode-se criar o quadro usando ferramentas de gestão digitais como as oferecidas pelo mercado (Jira, Trello, Clickup, MeuScrum, Bitrix24, entre outros).

MÉTODO KANBAN



Exemplo de um quadro Kanban clássico com apenas três colunas

MÉTODO KANBAN

Por mais simples que seja o método Kanban, **é preciso seguir uma série de princípios e práticas** para que ele traga os resultados desejados ao seu negócio.

Pode-se implementar o método sem seguir estes princípios e práticas, porém, isto **não dará certeza** de que os resultados buscados ao implementar o método serão alcançados.

Esses princípios permitirão que o uso do método Kanban evolua com o tempo, incrementando gradualmente os seus fluxos de trabalho e seus resultados sem causar disrupção em sua equipe e nos processos e resultados atuais da sua empresa.

Seguir estes princípios e práticas **permite o sucesso** para maximizar os benefícios que ele pode trazer aos processos em seu negócio.

MÉTODO KANBAN

Os 4 princípios básicos do Kanban

1. **Comece com o que estiver fazendo agora** – O Kanban deve ser aplicado ao fluxo de trabalho existente;
2. **Concorde em buscar mudanças incrementais** - com o objetivo de evoluir e otimizar seu fluxo de trabalho e o time;
3. **Inicialmente, respeite as funções atuais da sua equipe, responsabilidades e cargos** – O Kanban não impõe mudanças na organização da sua equipe;
4. **Encourage atos de liderança em todos os níveis de trabalho** – Melhorias contínuas nos processos e fluxos de trabalho irão otimizar a forma como se entrega produtos e serviços.

MÉTODO KANBAN

As 6 práticas básicas do Kanban

1. Visualize os fluxos de trabalho

- O Kanban valoriza a visualização do fluxo de trabalho do time através de um quadro físico ou virtual;

2. Limite o trabalho em andamento do fluxo de trabalho

- Evitar o multi-tasking dos especialistas do projeto;
- O Kanban encoraja a equipe a completar tarefas antes de mudar de foco para um novo trabalho;

MÉTODO KANBAN

As 6 práticas básicas do Kanban

3. Gerencie o fluxo de trabalho

- Otimizar o trabalho realizado é o ponto fundamental do método Kanban – Deve-se evitar os gargalos de produção;

4. Deixe as políticas de trabalho explícitas

- A clareza nas políticas de trabalho (regras e orientações) cria padrões para o trabalho a ser realizado.

MÉTODO KANBAN

As 6 práticas básicas do Kanban

5. Implemente ciclos de feedback

- Ter espaço para feedback por parte de todos os envolvidos é integral em qualquer sistema de qualidade – em momentos de revisão em torno do quadro Kanban, uso de métricas e relatórios para os **stakeholders**, uso de gráficos e outros métodos de comunicação visual que mostrem o andamento do projeto;

6. Melhore a colaboração da equipe

- O método Kanban é um processo de **melhoria evolutiva** – o Kanban encoraja o uso de métodos científicos onde se cria hipóteses, realiza-se testes sobre esta hipótese e executa-se mudanças baseadas nos resultados dos testes;

FILOSOFIA LEAN (LEAN SOFTWARE)

FILOSOFIA LEAN (LEAN SOFTWARE)

- O termo *Lean*, enxuto em português, está relacionado ao Sistema Toyota de Produção (TPS) que é uma filosofia de administração da produção;
- A Toyota desenvolveu **com sucesso a Filosofia Lean** em sua cultura e, por isso, organizações em todo o mundo basicamente criaram seus próprios sistemas de produção enxuta, baseado no sistema Toyota de produção.
- Também é conhecida como **Lean Thinking ou Pensamento Enxuto**.

FILOSOFIA LEAN (LEAN SOFTWARE)

- O Lean pode ser definido como um **conjunto de práticas**:
 - Cujo o objetivo é **eliminar o desperdício** de forma a reduzir custos e aumentar a produtividade e a lucratividade da empresa.
- A visão da produção enxuta é que todos os recursos da empresa sejam canalizados de maneira que **criem valor** para o usuário final.

FILOSOFIA LEAN (LEAN SOFTWARE)

- Alguns autores afirmam:
 - Que o Lean recentemente está deixando de ser **somente uma filosofia de produção;**
 - Para se tornar uma **ideologia de negócios:**
 - Mais abrangente que incorpora todos os aspectos das correntes de valor, opostas aos processos de produção individuais.

FILOSOFIA LEAN (LEAN SOFTWARE)

Os autores Poppendieck e Poppendieck (2003), em seu livro afirmaram:

❖ Que os princípios Lean são vastamente conhecidos, entretanto, advertem que:

- ☐ As organizações **não obtiveram sucesso** uniforme em aplica-los,
- ☐ Porque o *Lean Thinking* como é conhecido **exige uma mudança de cultura**,
- ☐ Mas, as empresas que entenderam e adotaram a essência do *Lean Thinking* obtiveram **melhorias significativas e sustentáveis** no seu desempenho.

POPPENDIECK, M.; POPPENDIECK, T. *Lean Software Development: An Agile Toolkit*, Crawfordsville, Indiana, 2003.

FILOSOFIA LEAN (LEAN SOFTWARE)

O Lean possui cinco princípios que o norteiam:

1. Valor (*Value*)

- É o foco principal do Lean, deve ser entendido sempre sobre a perspectiva do cliente;

2. Fluxo de Valor (*Value Stream*)

- É uma coleção de ponta a ponta de ações necessárias para trazer um produto para o cliente que lhe agregue o máximo de valor.

FILOSOFIA LEAN (LEAN SOFTWARE)

O Lean possui cinco princípios que o norteiam:

3. Fluxo Contínuo (*Flow*)

- Significa que o fluxo que produz valor, não possui descontinuidade para que as atividades sejam organizadas continuamente, permitindo entregas frequentes e tranquilas.

4. Fluxo Puxado (*Pull*)

- Implica em produzir produto ou partes de um produto somente quando eles são realmente necessários, sob demanda do cliente.

5. Perfeição (*Perfection*)

- Princípio central de melhoria contínua para chegar ao nível de defeitos zero para a entrega do cliente.

- **The Scrum Guide, 2017.** Disponível em: <https://www.scrum.org/resources/scrum-guide?gclid=Cj0KCQjwn8_mBRCLARIsAKxiOGJLf0E2Uy0Mj495FA-bjU6AvzGQKuL7UhCvB-rqs1oaQa8nik076EUaAiliEALw_wcB>
- **Manifesto para o Desenvolvimento Ágil de Software.** Disponível em: <<http://www.manifestoagil.com.br/>>
- **Standish Group, 2015.** Disponível em: https://www.standishgroup.com/sample_research_files/CHAOSReport2015-Final.pdf
- **Governança Digital 4.0. Brasil: Editora Brasport, 2019.**
- **Material dos Cursos da TI Exames:**
 - Professional Scrum Master Essentials
 - Professional Scrum Master Advanced
 - Professional Scrum Product Owner Essentials

FIM DAS CONSIDERAÇÕES SOBRE OS MODELOS ÁGEIS DE DESENVOLVIMENTO DE SOFTWARE