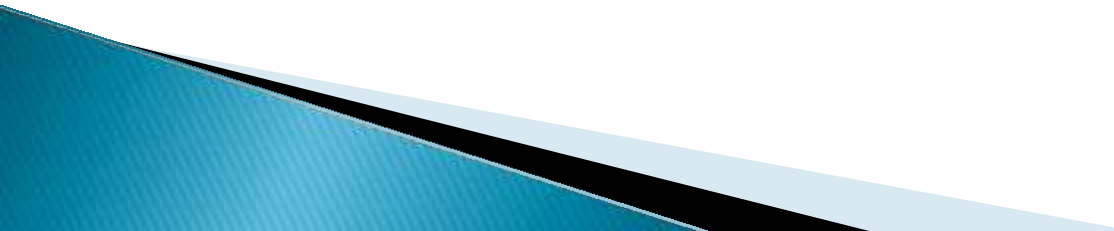


Métodos Ágeis

*Rodrigo Narvaes
Figueira*

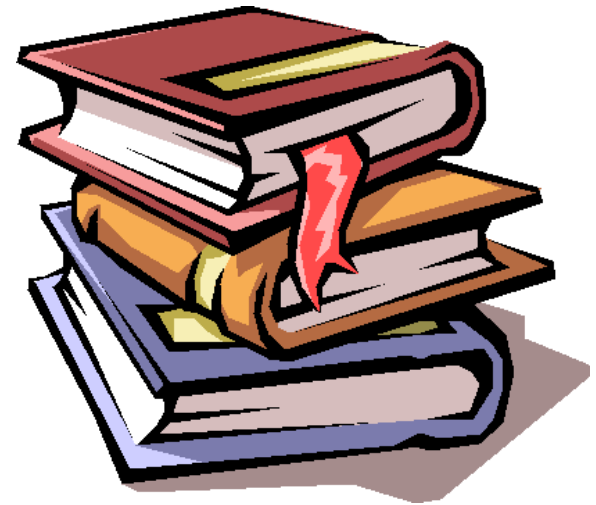


Objetivos

- Apresentação da metodologia.
 - O que é metodologia?
 - O que é metodologia ágil?
 - Comparativo entre metodologias ágeis e tradicionais
 - Apresentação do SCRUM
- 

Abordagem Teórica

- Definição de metodologias ágeis;
- Conceitos gerais sobre as principais metodologias utilizadas;
- Termos comuns utilizados;
- Técnicas comuns utilizadas.



Abordagem prática

- Estudo de caso que utilize alguns dos conceitos desenvolvidos.

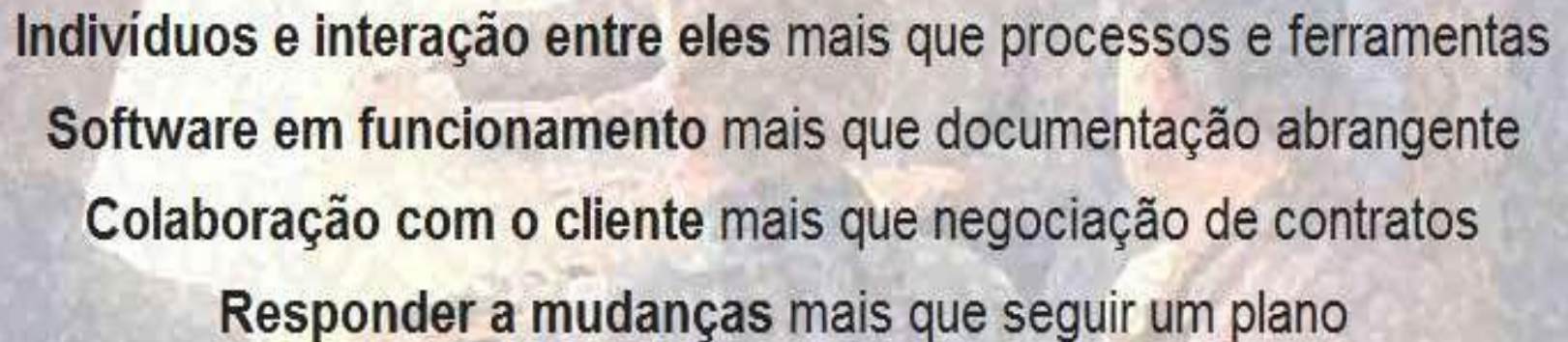


O que é uma metodologia?

- É o estudo de um método
public static void main(String[] args){}??
- Abordagem estruturada para desenvolver software com objetivo de **facilitar a produção** de alta qualidade dentro de **custos adequados**.
- “Como fazer”?



Metodologia Ágil



Indivíduos e interação entre eles mais que processos e ferramentas
Software em funcionamento mais que documentação abrangente
Colaboração com o cliente mais que negociação de contratos
Responder a mudanças mais que seguir um plano

Manifesto Ágil - 2001

Metodologia Ágil

Indivíduos e interação entre eles mais que processos e ferramentas

Software em funcionamento mais que documentação abrangente

Colaboração com o cliente mais que negociação de contratos

Responder a mudanças mais que seguir um plano

Manifesto Ágil - 2001

Os doze princípios

Nossa maior prioridade é satisfazer o cliente, através da entrega adiantada e contínua de software de valor.

Aceitar mudanças de requisitos, mesmo no fim do desenvolvimento. Processos ágeis se adequam a mudanças, para que o cliente possa tirar vantagens competitivas.

Entregar software funcionando com frequência, na escala de semanas até meses, com preferência aos períodos mais curtos.

Pessoas relacionadas a negócios e desenvolvedores devem trabalhar em conjunto e diariamente, durante todo o curso do projeto.

Construir projetos ao redor de indivíduos motivados. Dando a eles o ambiente e suporte necessário, e confiar que farão seu trabalho.

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.

Software funcional é a medida primária de progresso.

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

Continua atenção à excelência técnica e bom design, aumenta a agilidade.

Simplicidade: a arte de maximizar a quantidade de trabalho que não precisou ser feito.

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

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

Os doze princípios

Prioridade, satisfazer o cliente com entrega adiantada e contínua de SW de valor

Aceitar e adequar-se a mudanças para o cliente tirar vantagens competitivas.

Entregar SW funcionando com frequência, na escala de horas, dias ou semanas.

Pessoas de negócio e desenvolvimento trabalham em conjunto diariamente.

Projetos com indivíduos motivados, ambiente e suporte, fazem a diferença.

O Método mais eficaz de se comunicar inter e intra, é uma conversa cara a cara.

Os doze princípios

Software funcional é a medida primária de progresso.


Todos são capazes de manter passos constantes em um ambiente sustentável.

Contínua atenção à excelência técnica e bom design, aumenta a agilidade.

A arte de maximizar a quantidade de trabalho que não precisou ser feito.

As melhores arquiteturas e artefatos emergem de times auto-organizáveis.

Em intervalos regulares, o time reflete em como ficar mais efetivo e melhora.



Metodologia Ágil

- ✓ Conjunto de metodologias de desenvolvimento de *software*;
- ✓ Tirar o foco do Processo;
- ✓ Mais ênfase à contribuição das pe



Metodologia Ágil vs Tradicional

- Metodologias ágeis são adaptativas em vez de preditivas;

“O problema não é a mudança em si, mesmo porque ela ocorrerá de qualquer forma. O problema é como receber, avaliar e responder às mudanças.”



Metodologia Ágil vs Tradicional

- Metodologias tradicionais são recomendadas quando os requisitos de software são estáveis e requisitos futuros são facilmente previsíveis.

O cliente sabe o que quer?



Metodologia Ágil vs Tradicional

- Metodologias ágeis propiciam entregas constantes enquanto que nas metodologias tradicionais este tempo é mais espaçado.



Metodologia Ágil vs Tradicional

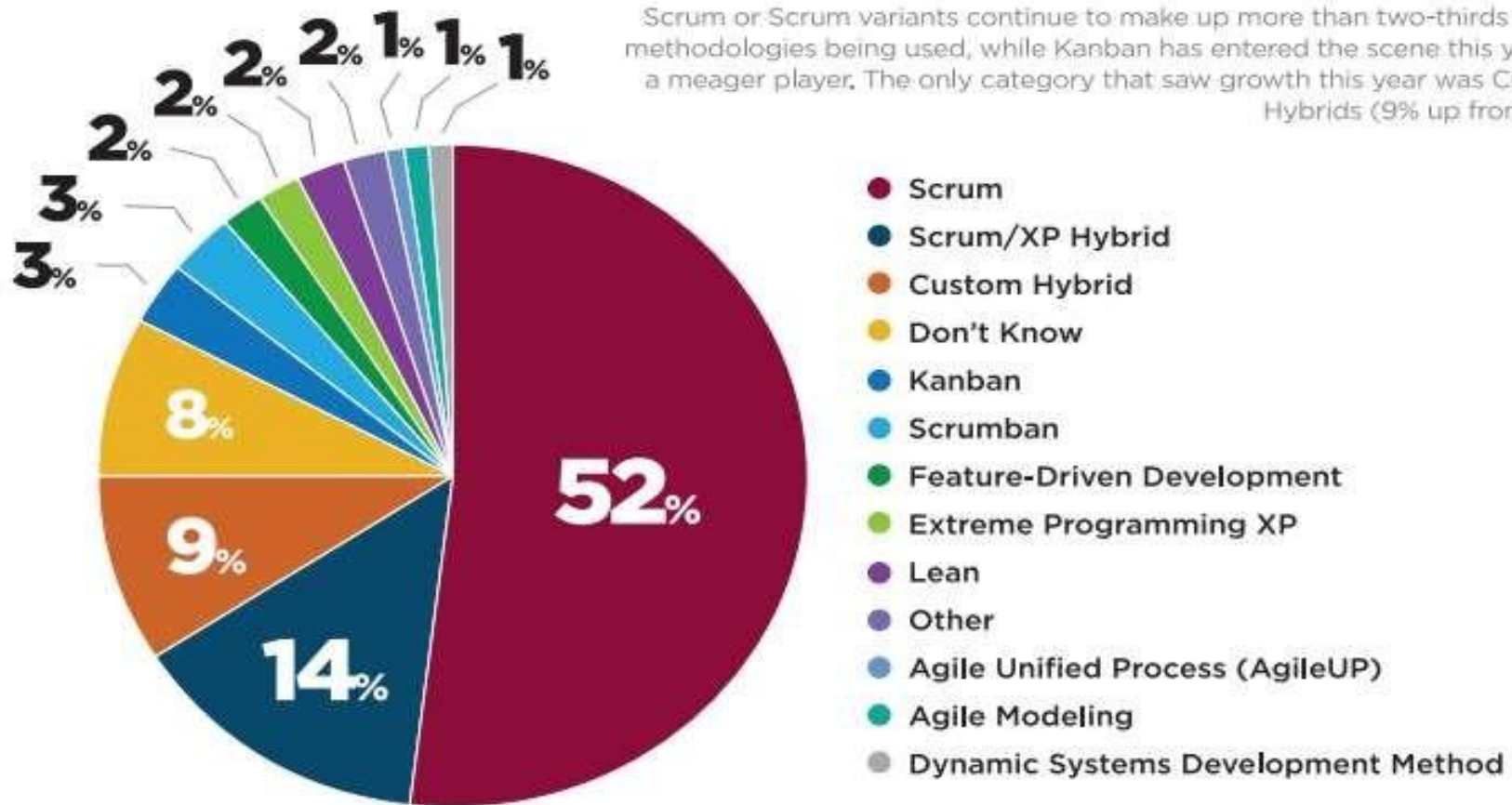
- Metodologias ágeis estão em constante crescimento.



O que é usado?

AGILE METHODOLOGY USED

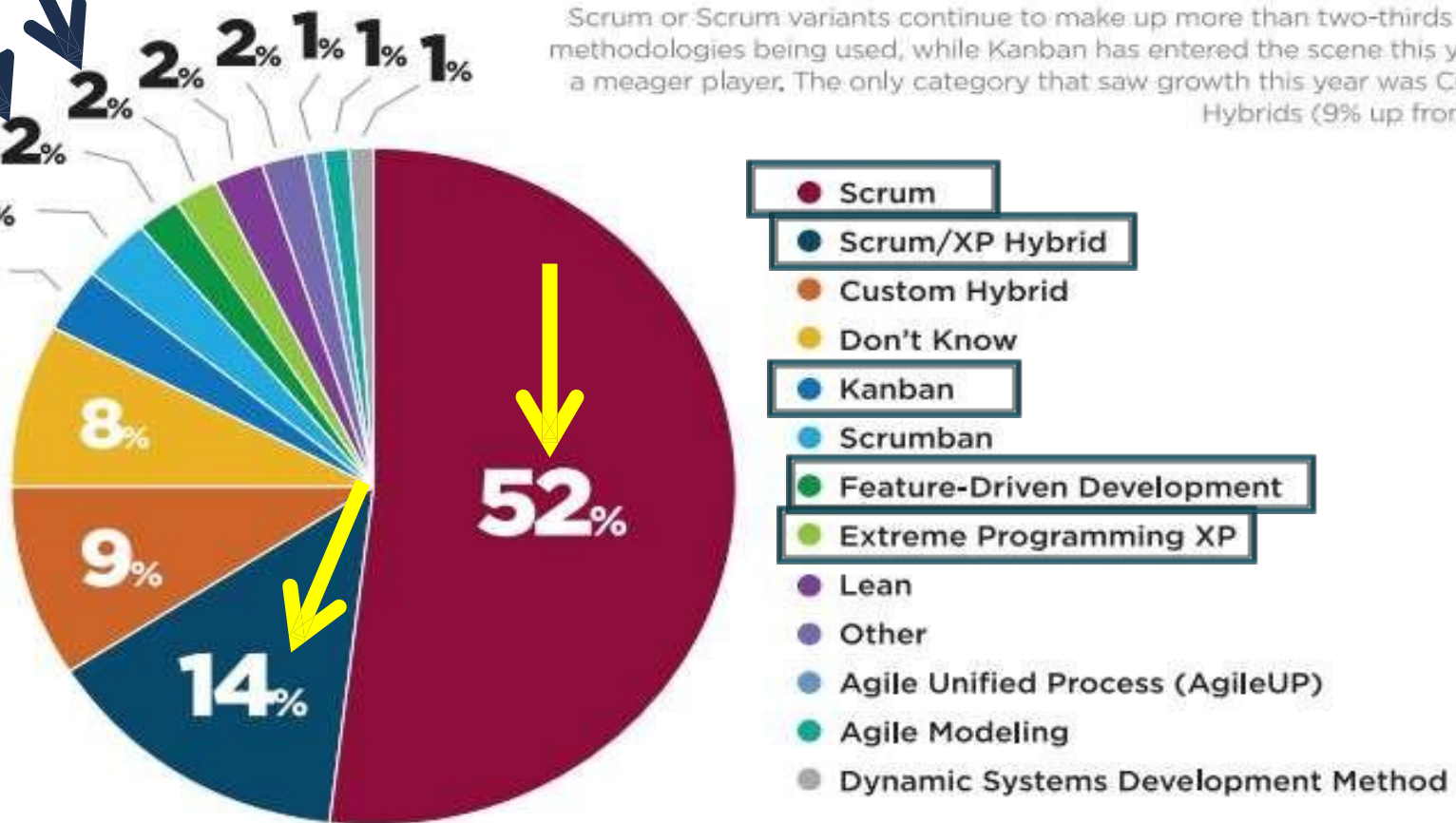
Scrum or Scrum variants continue to make up more than two-thirds of the methodologies being used, while Kanban has entered the scene this year as a meager player. The only category that saw growth this year was Custom Hybrids (9% up from 5%).



O que vamos apresentar

AGILE METHODOLOGY USED

Scrum or Scrum variants continue to make up more than two-thirds of the methodologies being used, while Kanban has entered the scene this year as a meager player. The only category that saw growth this year was Custom Hybrids (9% up from 5%).



Scrum



O que É Scrum?

- É uma metodologia de desenvolvimento iterativo e incremental;

O que NÃO é Scrum?

- Não é um processo nem descreve o que se deve fazer em cada situação;

Os papéis

❖ **Product Owner**

❖ **Scrum Master**

❖ **Team**



Product Owner



Product Owner

- Voz do cliente;
- Centrado nos itens do cliente;
- Prioriza o que será feito;
- Adiciona itens ao escopo do projeto;



Team



Team

- Responsável pela entrega do produto;
- Composta por 5-9 pessoas;
- Auto-organizada;
- Auto-conduzida;



Scrum Master



Scrum Master



- ✓ Remoção de impedimentos;
- ✓ Não é necessariamente o líder da equipe;
- ✓ Manter a equipe focada nas tarefas;
- ✓ Facilitador;
- ✓ Negociador;

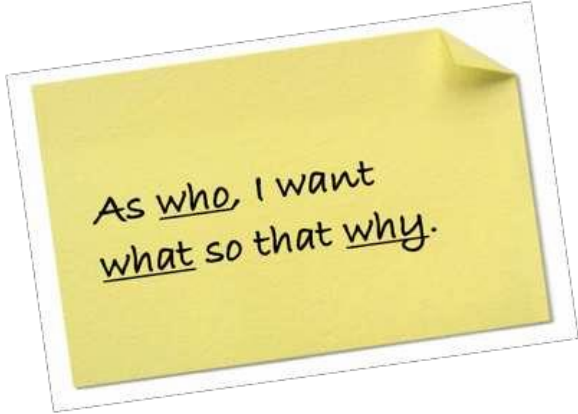
Artefatos

- User Stories
- Product Backlog
- Sprint Backlog
- Reuniões
 - Daily Meeting
 - Sprint Planning Meeting
 - Sprint Review
 - Sprint Retrospective



User Stories (Histórias do Usuário)

- Forma de definir e organizar requisitos do usuário;
- Devem ser curtas, simples e claras;
- Ao contrário de casos de uso, são focados em objetivos do usuário e como o sistema alcança estes objetivos.



As who, I want
what so that why.

User Stories (Histórias do Usuário)

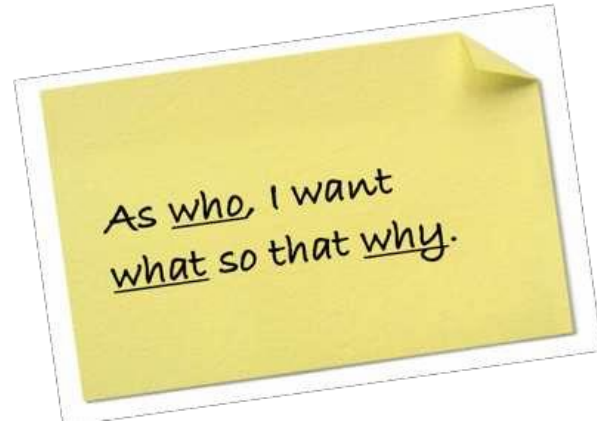
- Fracionam requisitos visando facilitar estimativas de esforço;
- Normalmente apresentada em cartões;

“Se não cabe em um cartão devemos refinar”



User Stories (Histórias do Usuário)

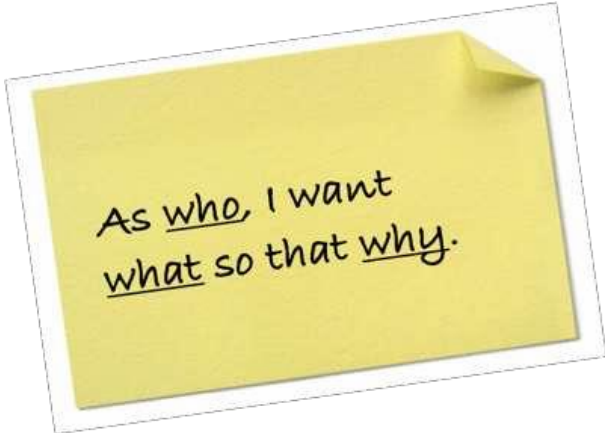
- Contem:
 - *Ator*
 - *Ação*
 - *Funcionalidade*



As who, I want
what so that why.

User Stories (Histórias do Usuário)

- Como um *Usuário Gestor* preciso *cadastrar* os *funcionários de minha equipe* para que se possa *calcular o PPR* destes funcionários



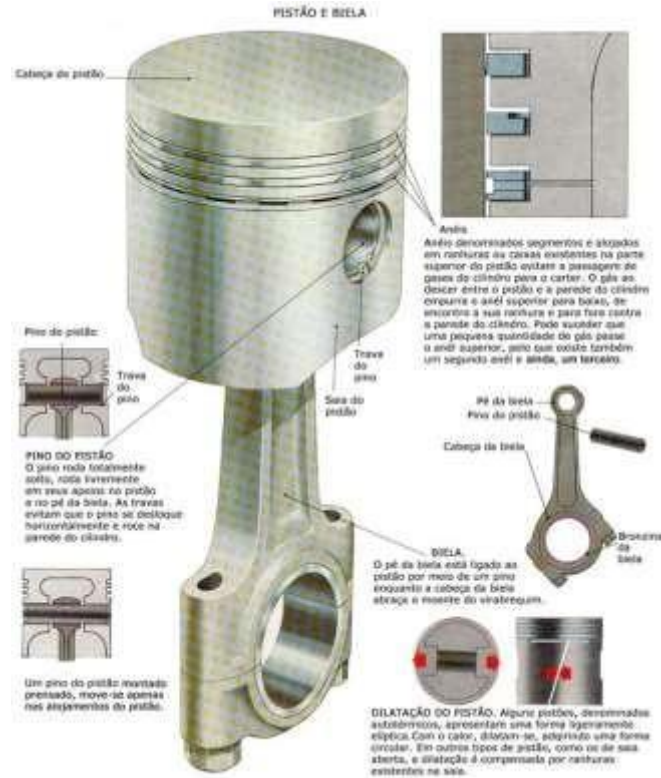
As who, I want
what so that why.

Product Backlog



O conjunto de pendência que formam um produto completo

Sprint Backlog



Seleção de pendências a serem realizadas para uma entrega

Reuniões

- Reunião Diária (*Daily Meeting*)
- Reunião de Planejamento da Sprint (*Sprint Planning Meeting*)
- Reunião de Revisão da Sprint (*Sprint Review*)
- Retrospectiva da Sprint (*Sprint Retrospective*)



Daily Meeting



O que eu fiz?

O que eu farei?

Algo está me impedindo?

Reunião diária, em pé. Todos podem participar, porém poucos podem falar.

Sprint Planning Meeting



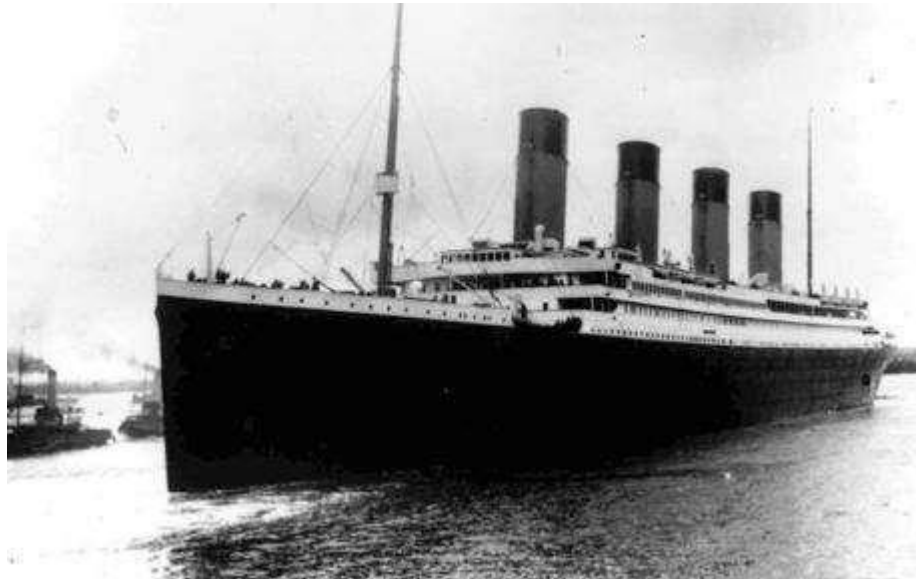
Seleção das pendências que sairão do Product Backlog para o Sprint Backlog

Sprint Review

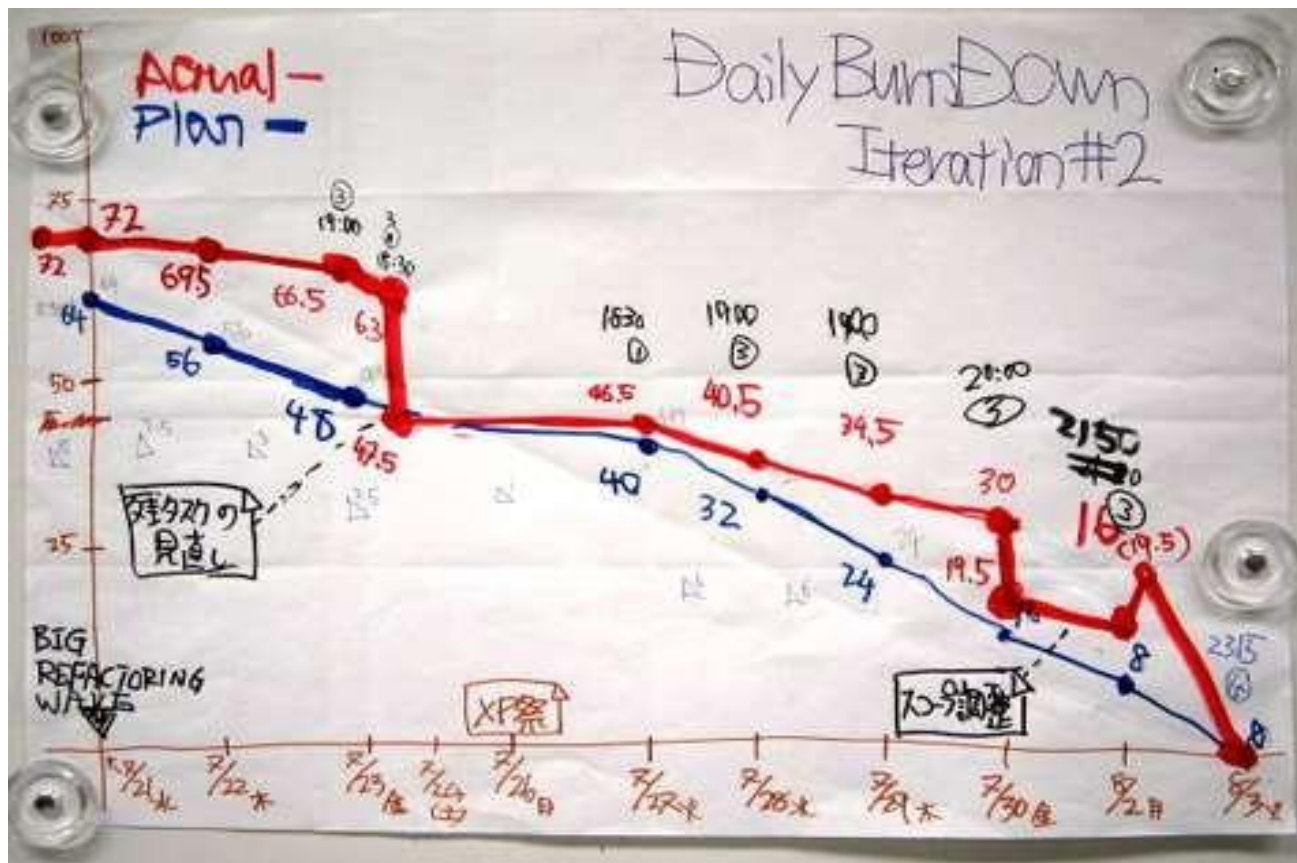


Rever o trabalho que foi concluído e não concluído

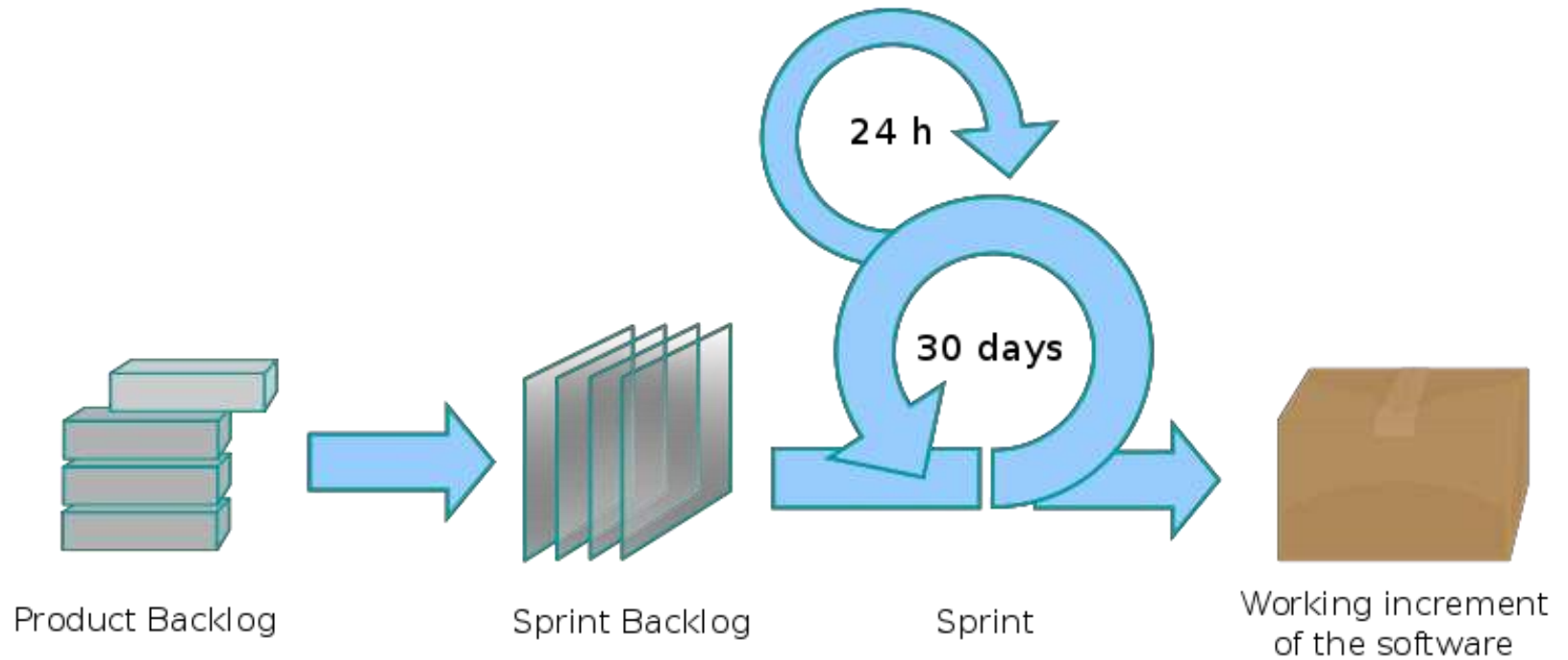
Sprint Retrospective



BurnDown Chart



A metodologia



O que é um Sprint?



O que é um Sprint?

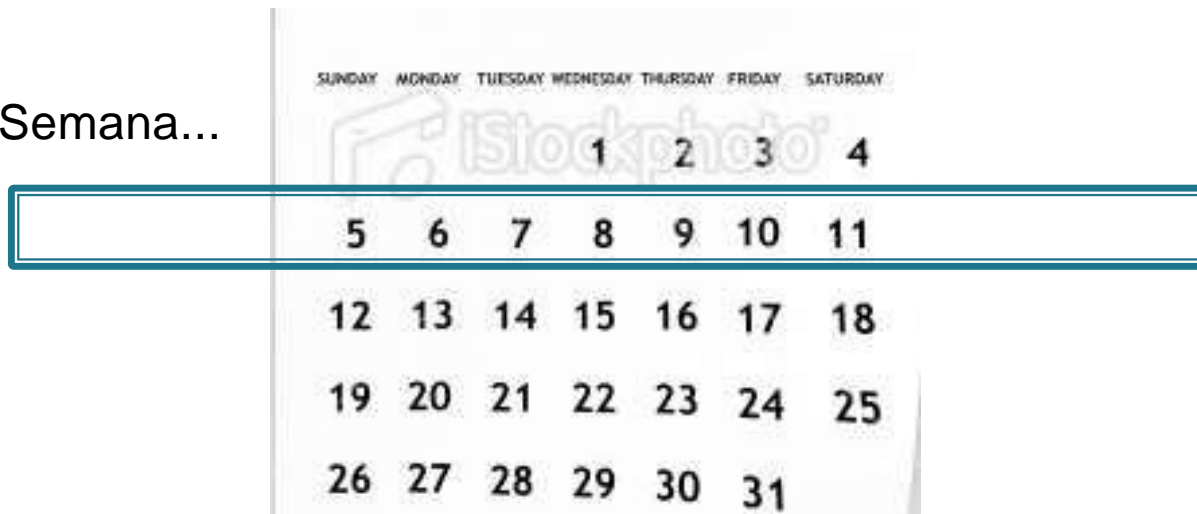
O período de tempo até que um resultado funcional seja entregue



O que é um Sprint?

Duração:

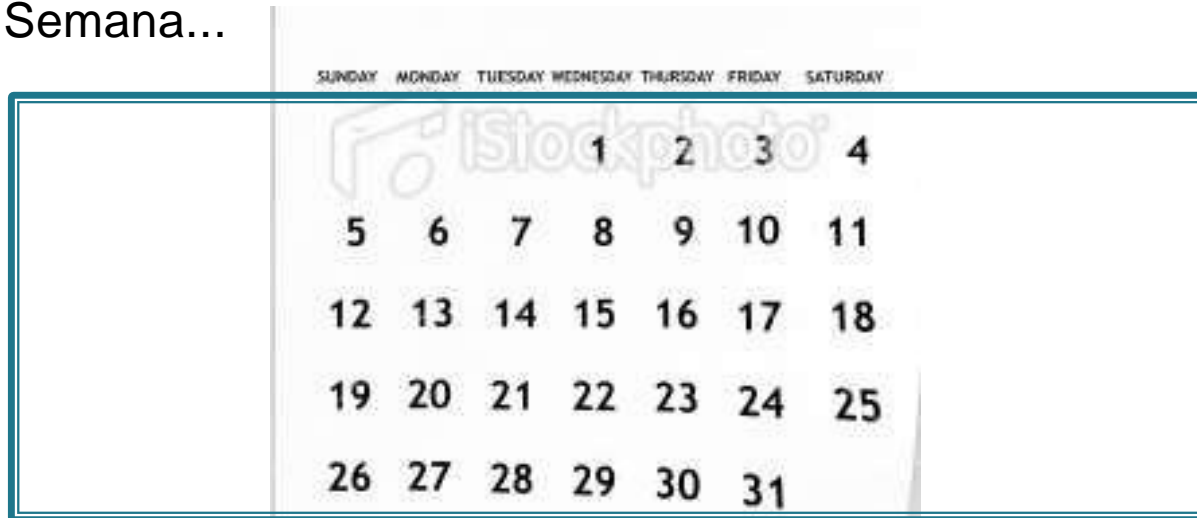
De uma Semana...



O que é um Sprint?

Duração:

De uma Semana...



... A um mês

Resultado: Um incremento no produto

Kanban

- ✓ O **Kanban**, é um método ágil de desenvolvimento de software baseado nas práticas Lean, e que tem como objetivo otimizar o processo de desenvolvimento de software.

Visualização do fluxo



Um exemplo de quadro



Razões para adotar o Kanban

- Entregas a qualquer momento
- Mudar a prioridade a qualquer instante...
- Visualização do fluxo de trabalho (transparência)
- Não quer se preocupar com iterações
- Não quer se preocupar com estimativas

Razões para Não adotar o Kanban

- Entregas baseadas em iterações com tamanho fixos
- Foco no desenvolvimento de produto
- Visualização da Sprint (Sprint Burndown)
- Estórias devem ser estimadas
- Necessidade ter papéis bem definidos, tais como Product Owner

XP – eXtreme Programming

- **Valores**

- Comunicação
- Simplicidade
- *Feedback*
- Coragem
- Respeito

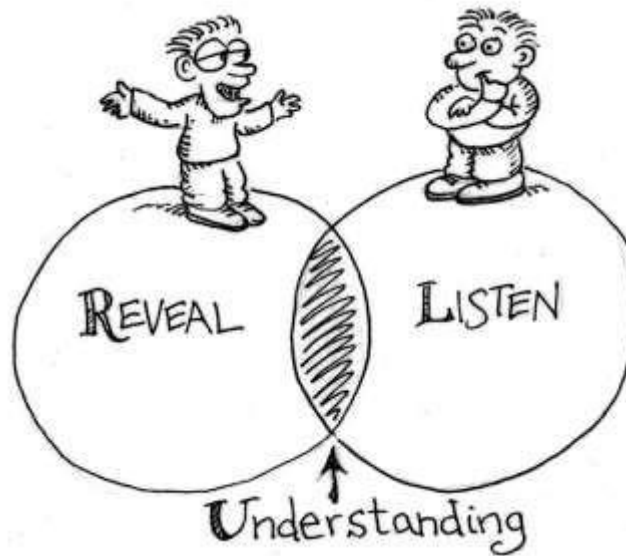
- **Princípios básicos**

- *Feedback* rápido
- Presumir simplicidade
- Mudanças incrementais
- Abraçar mudanças
- Trabalho de alta qualidade.

Práticas

Jogo de Planejamento (*Planning Game*)
Pequenas Versões (*Small Releases*)
Metáfora (*Metaphor*)
Projeto Simples (*Simple Design*)

0	$\frac{1}{2}$	1	2
3	5	8	13



Práticas

Time Coeso (*Whole Team*)

Testes de Aceitação (*Customer Tests*)

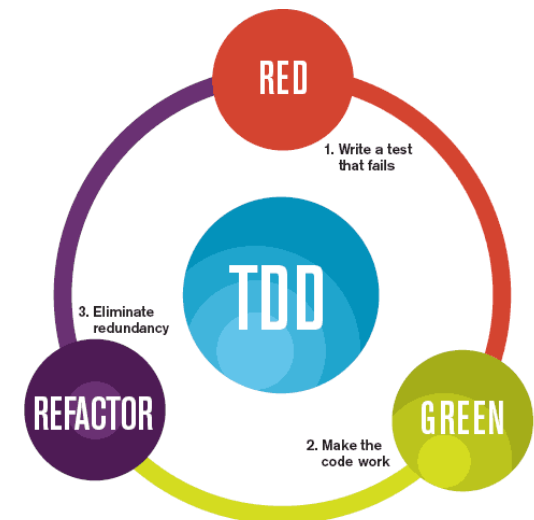
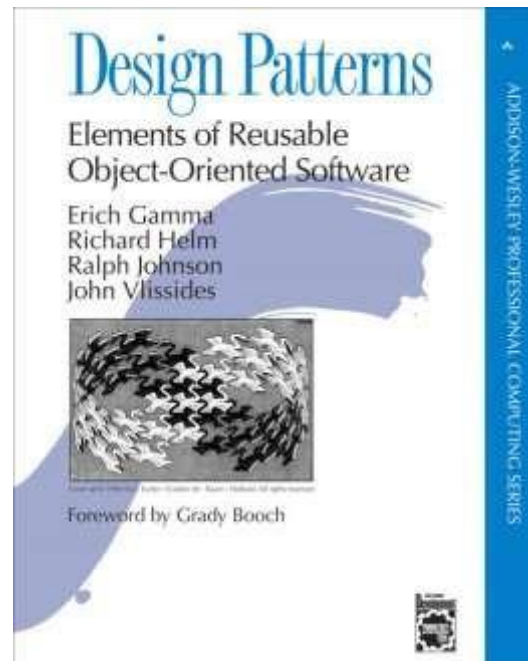
Ritmo Sustentável (*Sustainable Pace*)

Reuniões em pé (*Stand-up Meeting*)



Práticas

Posse Coletiva (*Collective Ownership*)
Programação em Pares (*Pair Programming*)
Padrões de Codificação (*Coding Standards*)
Desenvolvimento Orientado a Testes (*Test Driven Development*)



The mantra of Test-Driven Development (TDD) is "red, green, refactor."

Práticas

Refatoração (*Refactoring*)

Integração Contínua (*Continuous Integration*)

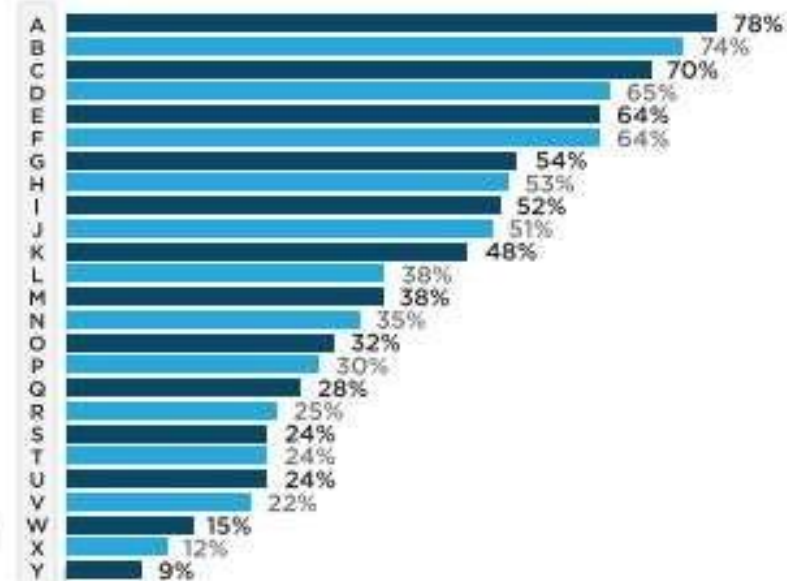


As práticas mais usadas

AGILE TECHNIQUES EMPLOYED

Core agile tenets currently in use are* Daily Standup, Iteration Planning and Unit Testing. Most notable is the increasing use of Kanban (24%). *Respondents were able to select multiple options.

A	Daily Standup	N	Story Mapping
B	Iteration Planning	O	Digital Taskboard
C	Unit Testing	P	Pair Programming
D	Release Planning	Q	Collective Code Ownership
E	Burndown	R	Automated Acceptance Testing
F	Retrospectives	S	Kanban
G	Continuous Integration	T	Onsite Customer
H	Automated Builds	U	Continuous Deployment
I	Velocity	V	Analog Taskboard
J	Coding Standards	W	Agile Games
K	Refactoring	X	Cycle Time
L	Test-Driven Development TDD	Y	Behavior-Driven Development BDD
M	Open workarea		

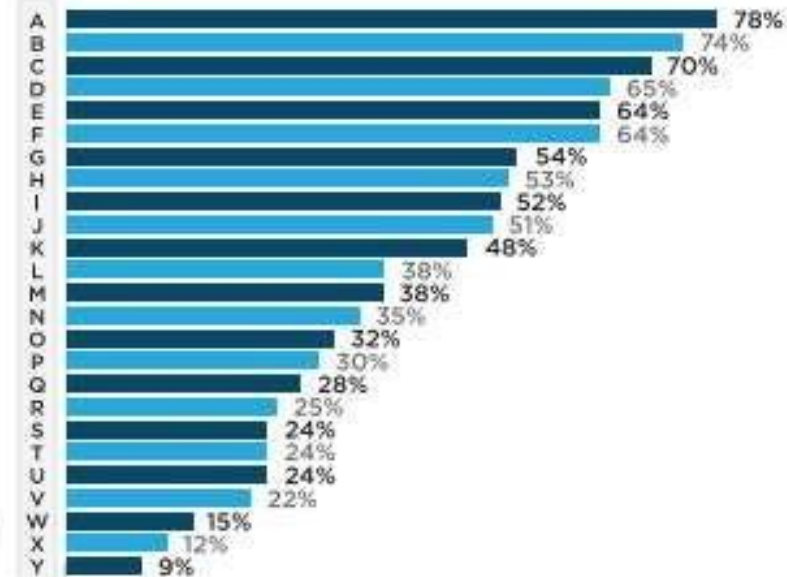


As práticas mais usadas

AGILE TECHNIQUES EMPLOYED

Core agile tenets currently in use are* Daily Standup, Iteration Planning and Unit Testing. Most notable is the increasing use of Kanban (24%). *Respondents were able to select multiple options.

- | | |
|---|---|
| <input checked="" type="radio"/> Daily Standup | N Story Mapping |
| <input checked="" type="radio"/> Iteration Planning | O Digital Taskboard |
| C Unit Testing | <input checked="" type="radio"/> Pair Programming |
| <input checked="" type="radio"/> Release Planning | Q Collective Code Ownership |
| <input checked="" type="radio"/> Burndown | R Automated Acceptance Testing |
| F Retrospectives | S Kanban |
| G Continuous Integration | T Onsite Customer |
| H Automated Builds | U Continuous Deployment |
| I Velocity | V Analog Taskboard |
| <input checked="" type="radio"/> Coding Standards | W Agile Games |
| K Refactoring | X Cycle Time |
| L Test-Driven Development TDD | Y Behavior-Driven Development BDD |
| M Open workarea | |



Um exemplo usando

- Kanban
- XP
- Scrum

Pedidos dos clientes

Cliente A

**Necessida
de A**

Cliente B

**Necessida
de A**

Cliente C

**Necessida
de A**

Pedidos dos clientes

Cliente A

**Necessida
de A**

Cliente B

**Necessida
de A**

Cliente C

**Necessida
de A**

Os pedidos são avaliados

Cliente A

A1

A2

Cliente B

B1

Cliente C

C1

C2

C3

C4

C5

C6

C7

C8

Remoção por tamanho

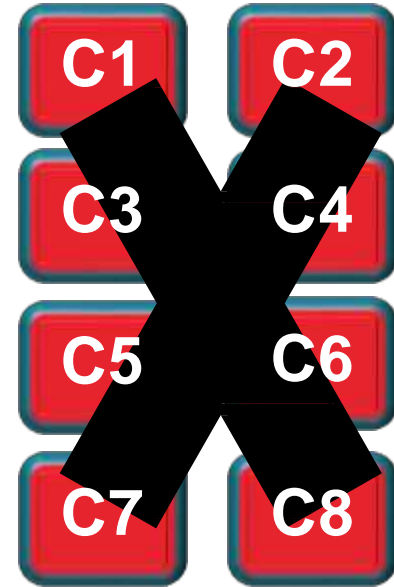
Cliente A



Cliente B



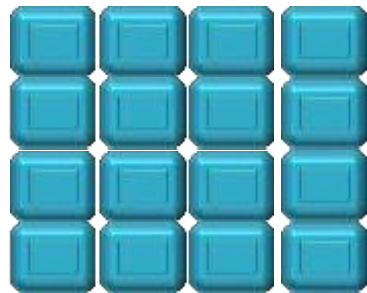
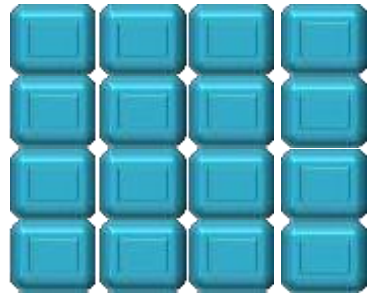
Cliente C



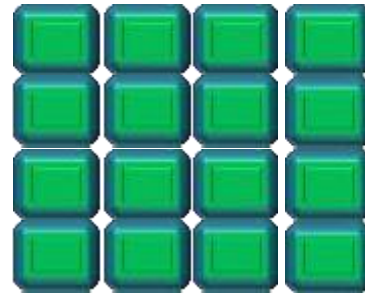
Team + ScrumMaster

Quebra em pequenas tarefas

Cliente A

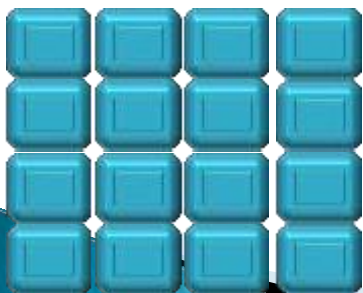
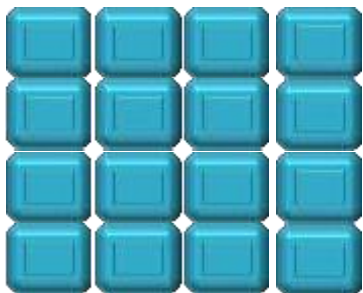
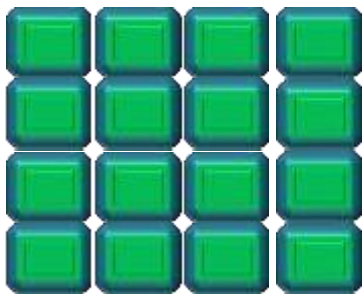


Cliente B



Product
Owner
Scrum
Master

**Product
BackLog**



**Sprint
BackLog**



TO DO

4



Doing

3



QA/Test

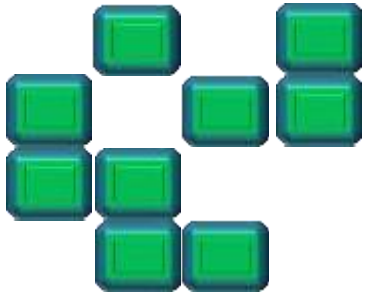
3



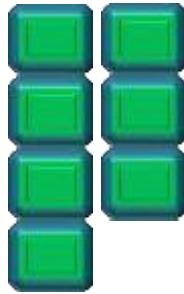
Done



Product
BackLog



Sprint
BackLog



TO DO

4

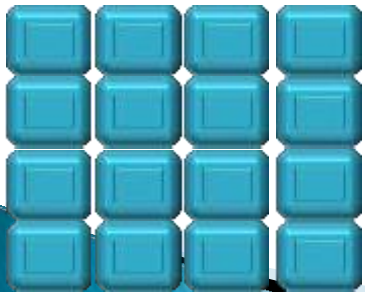
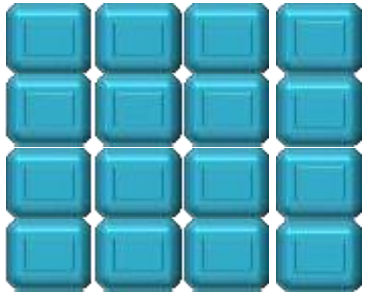
Doing

3

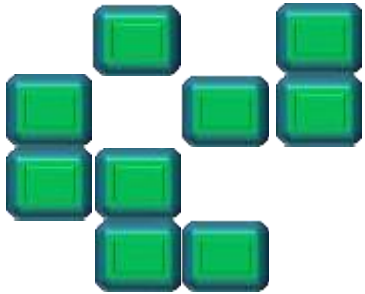
QA/Test

3

Done



Product
BackLog



Sprint
BackLog



TO DO

4



Doing

3



QA/Test

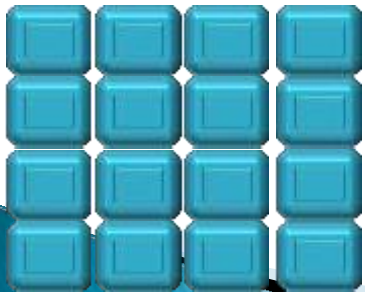
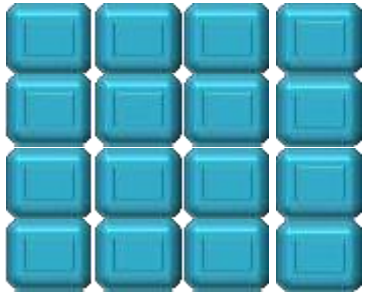
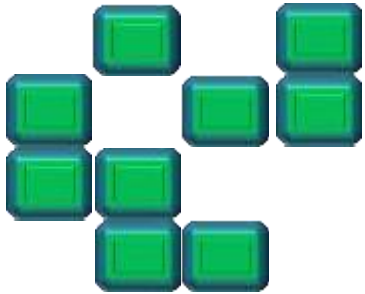
3



Done



Product
BackLog



Sprint
BackLog



TO DO

4



Doing

3

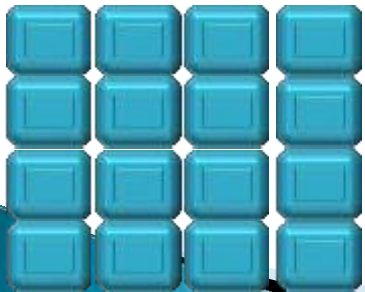
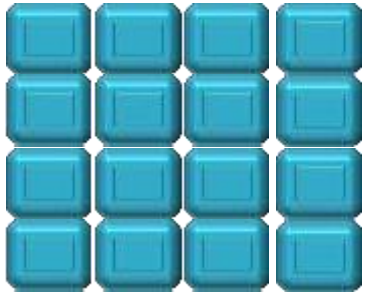
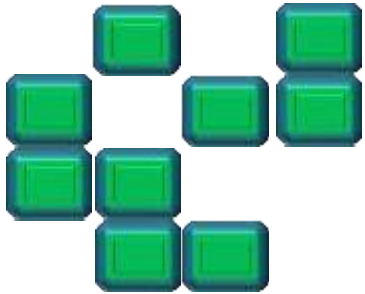


QA/Test

3

Done

Product
BackLog



Sprint
BackLog



TO DO

4

Doing

3



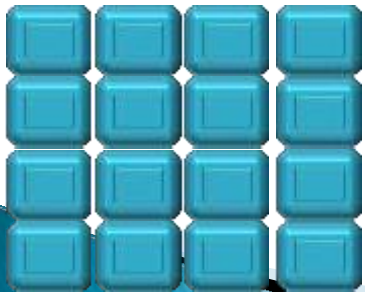
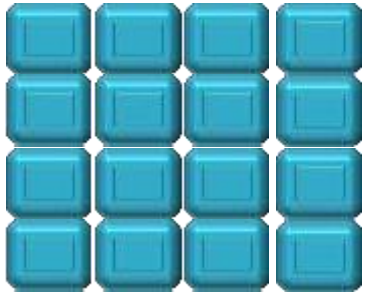
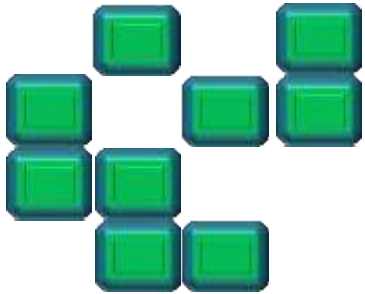
QA/Test

3



Done

Product
BackLog



Sprint
BackLog

TO DO

4



Doing

3

QA/Test

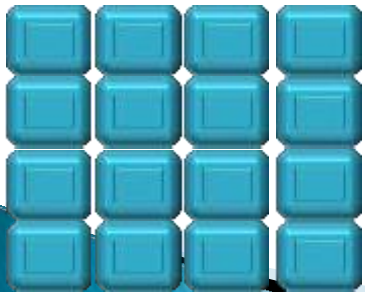
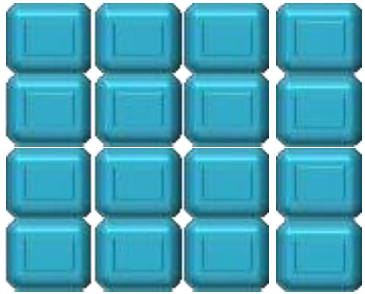
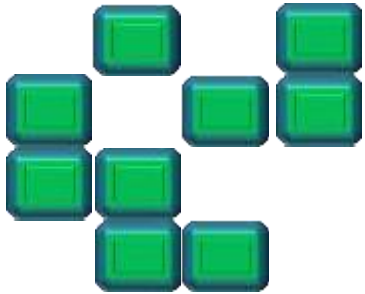
3



Done



Product
BackLog



Sprint
BackLog

TO DO

4



Doing

3



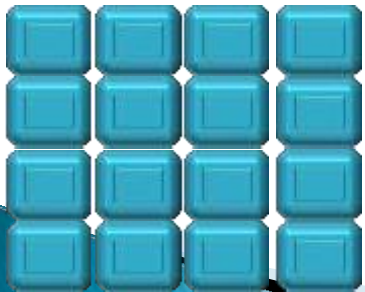
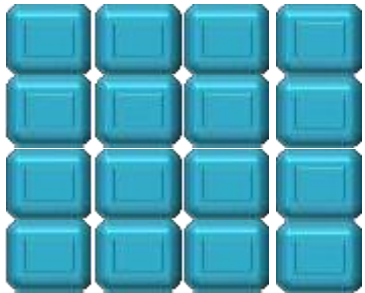
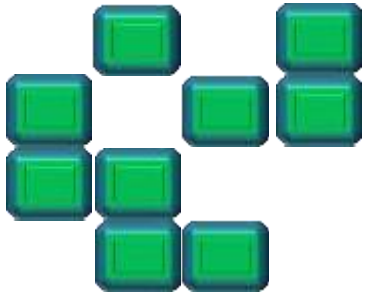
QA/Test

3

Done



Product
BackLog



Sprint
BackLog

TO DO

4

Doing

3

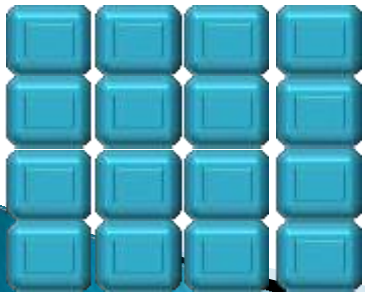
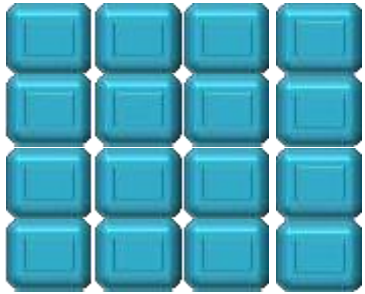
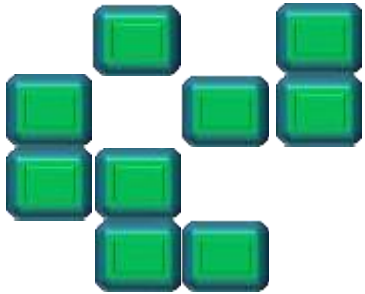
QA/Test

3

Done



Product
BackLog



Sprint
BackLog

TO DO

4

Doing

3

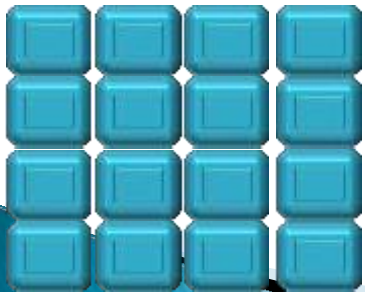
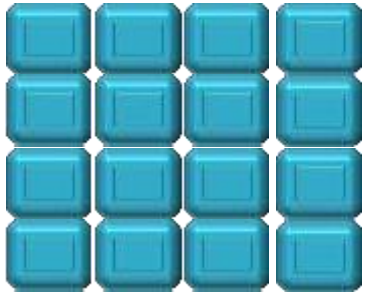
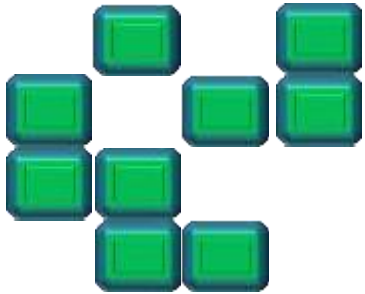
QA/Test

3

Done



Product
BackLog



Sprint
BackLog

TO DO

4

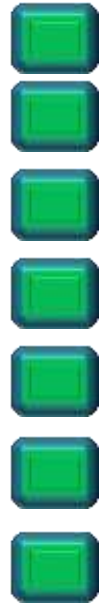
Doing

3

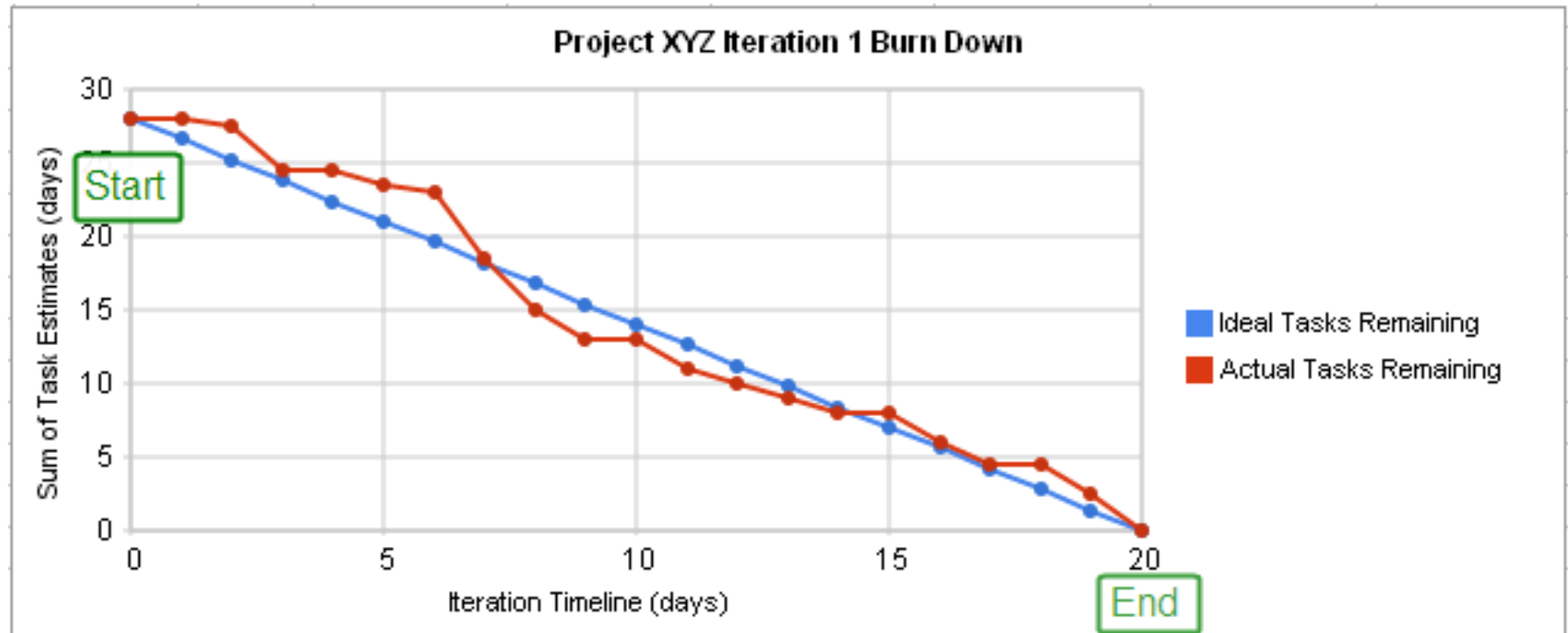
QA/Test

3

Done

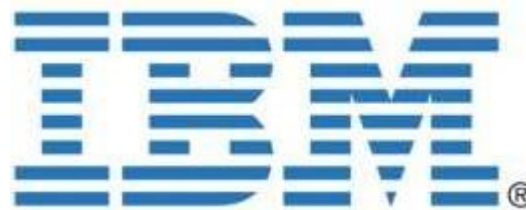


BurnDown Chart



Certificação x Métodos Ágeis

Motorola CMMI Nível 5 – Usa Scrum;
IBM CMMI Nível 5 – Usa Distributed
Scrum;



Casos de sucesso



[Site](#) – [Artigo](#)



[Site](#) - [Artigo](#)