

Projeto de Software

Gerenciamento de projetos ágeis

Me Marco Ikuro Hisatomi

1

- Unidade de Ensino: 02
- Competência da Unidade: Conhecer e compreender as outros modelos de projetos de softwares.
- Resumo: Modelar e aplicar técnicas ágeis e de Design para projetos de software.
- Palavras-chave: metodologia ágil; SCRUM; XP; FDD; Lean; MVP; design thinking;
- Título da Teleaula: Gerenciamento de projetos ágeis
- Teleaula nº: 02

2

Manifesto e métodos ágeis para gerenciamento de softwares

3

Histórico

- Meados de 1990, motivados por uma reação adversa aos chamados "métodos pesados" de desenvolvimento de software;
- 2001 surgimento e a propagação de paradigmas de desenvolvimento de software ágeis;
- Manifesto ágil.

4

Manifesto ágil – valores

- I. Indivíduos e interações mais que processos e ferramentas;
- II. Software em funcionamento mais que documentação abrangente;
- III. Colaboração com o cliente mais que negociação de contratos;
- IV. Responder às mudanças mais que seguir um plano.



Fonte: do autor

5

Manifesto ágil – princípios

- i. A prioridade é satisfazer ao cliente;
- ii. Mudanças de requisitos são bem-vindas;
- iii. Entregar software funcionando frequentemente;
- iv. As equipes de negócio e de desenvolvimento devem trabalhar juntas;
- v. Construa projetos em torno de indivíduos motivados;
- vi. A maneira mais eficiente de a informação circular entre a equipe de desenvolvimento é por uma conversa cara a cara;



Fonte: do autor

6

Manifesto ágil – princípios

- vii. Software funcionando é a medida primária de progresso;
- viii. Processos ágeis promovem um desenvolvimento sustentável;
- ix. Atenção contínua à excelência técnica e a um bom design aumenta a agilidade;
- x. Simplicidade é essencial;
- xi. As melhores arquiteturas, os melhores requisitos e projetos emergem de times auto-organizáveis;
- xii. Em intervalos regulares, a equipe deve refletir sobre como se tornar mais eficaz.



Fonte: do autor

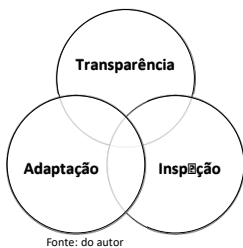
7

Motivações

- Menos centradas em documentação;
- São adaptativas em vez de predeterminantes;
- Aceitam mudanças ao longo do desenvolvimento de maneira natural;
- Orientados a pessoas e não a processos;

8

Empiricismo



Fonte: do autor

Melhoria contínua: a experiência é a base da construção de soluções

- Apresentar Fatos
- Equipe inspeciona
- Melhoria com base na inspeção

9

Tipos de metodologias ágeis Parte I

10

Extreme Programming (XP)

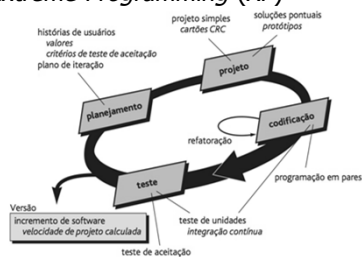
- Rápido desenvolvimento;
- Requisitos se alteram constantemente;
- Valores do XP:
 - Comunicação;
 - Simplicidade;
 - Feedback;
 - Coragem.



Fonte: do autor

11

Extreme Programming (XP)



Fonte: Precaman (2014)

12

Extreme Programming (XP)

Equipe de trabalho:

- Gerente de Projeto;
- Coach;
- Analista de teste;
- Redator técnico;
- Desenvolvedor.

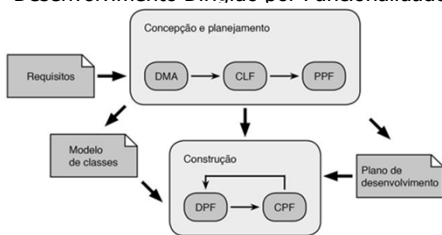
13

Desenvolvimento Dirigido por Funcionalidade - FDD

- Enfatiza o uso de orientação a objetos.
- Duas grandes fases:
 - Concepção e planejamento
 - Construção

14

Desenvolvimento Dirigido por Funcionalidade - FDD



15

Tipos de metodologias ágeis Parte II

16

Crystal Clear

Para projetos flexível e pequeno porte:

1. Foco (sem distração)
2. Entrega frequentes
3. O processo de reflexão
4. Comunicação permanente (oral)
5. Confiança entre os membros
6. Ambiente técnico favorável
7. Peritos envolvidos e disponíveis



Fonte: do autor

17

ASD – Adaptive Software Development

- É baseado em ciclos iterativos de 4 a 8 semanas;
- Os prazos são pré-fixados (*timeboxing*);
- É tolerante à mudança e à adaptação;
- É orientado a desenvolver primeiramente os elementos de maior risco.

Especulação

Colaboração

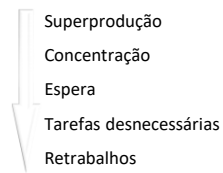
Aprendizado

Fonte: adaptado de Wazlawick (2013)

18

Lean

- Elimine o desperdício;
- Desenvolver com qualidade;
- Crie o conhecimento;
- Melhore na clima organizacional;
- Entregue rapidamente;
- Valorize a equipe;
- Otimize o todo.



19

XP

20

Desafio

Como se daria cada situação a seguir, ao adotar o XP:

- Levantar pontos frágeis da metodologia atual
- Planejar a introdução de práticas do XP relacionadas aos princípios da comunicação e *feedback*
- Adotar práticas contínuas de aprimoramento do modelo e de encantamento de novos clientes

21

Resolução

Levantar pontos frágeis da metodologia atual:

- Identificar falta de comunicação com o cliente
- Verificar que o cliente não faz parte do processo, logo não conhece as funcionalidades
- Certificar-se que as etapas do processo devem ser integralmente concluídas

22

Resolução

Planejar a introdução de práticas do XP relacionadas aos princípios da comunicação e *feedback*:

- Chamar o cliente ao projeto
- Nova maneira de coletar e tratar os requisitos
- O código coletivo, a *stand up meeting* e o uso de metáforas também inspiram relativa facilidade em suas implementações

23

Resolução

Adotar práticas contínuas de aprimoramento do modelo e de encantamento de novos clientes:

- Colher percepções e sugestões da equipe sobre sua rotina
- Treinamento dos novos desenvolvedores que venham a ingressar na instituição

24

Dúvidas?

25

SCRUM Parte I

26

SCRUM

- Método ágil mais usado atualmente
- Aplica-se não só ao desenvolvimento de softwares como a qualquer ambiente de trabalho
- É um framework utilizado na gestão de projetos e **desenvolvimento ágil de software**



Fonte: do autor

27

SCRUM

Objetivos:

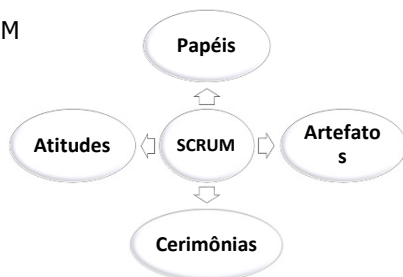
- Desenvolvidos complexos em que os requisitos mudam rapidamente e constantemente
- Gerenciar e controlar o desenvolvimento do trabalho
- Tornar a equipe autogerenciável e funcional
- Implementar o conceito iterativo e incremental no desenvolvimento de software e/ou produtos
- Identificar causas de problemas e remover impedimentos



Fonte: do autor

28

SCRUM



Fonte: adaptado de Starocco (2012)

29

SCRUM

Equipe de trabalho (papéis):

- **Scrum Master:** trata-se de um facilitador do projeto
- **Product Owner:** é a pessoa responsável pelo projeto propriamente dito. Ele tem a missão de indicar os requisitos mais importantes a serem tratados nos Sprints
- **Scrum Team:** é a equipe de desenvolvimento, composta normalmente por seis a dez pessoas



Fonte: do autor

30

SCRUM

- Gestão e planejamento de Software
- Reuniões Regulares (*Stand-up Meeting*)
- Projetos são divididos em ciclos
- Entrega incremental
- Envolvimento do cliente



Fonte: do autor

31

SCRUM Parte II

32

SCRUM

- *Product Backlog*: trata-se da lista que contém todas as funcionalidades desejadas para o produto.
- *Sprint Backlog*: lista de tarefas que a equipe deverá executar naquele Sprint.
- *Sprint*: divide o processo de efetiva construção do software em ciclos regulares, que variam de duas a quatro semanas.



Fonte: do autor

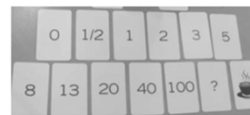
33

SCRUM

- Kanban: "quadro de trabalho", organizar as atividades dos itens de Backlog da Sprint;
- Pôquer do planejamento;



Fonte: <https://bit.ly/39Ruc2Y>



Fonte: Sbrocco (2012)

34

SCRUM



Fonte: Sbrocco (2012)

35

SCRUM

Cerimônias:

- Planejamento da sprint (Sprint Planning Meeting);
- Reunião diária (Daily Meeting ou Daily SCRUM);
- Revisão da sprint (Sprint Review);
- Retrospectiva da sprint (Sprint Retrospective).



Fonte: do autor

36

Abordagens inovadoras e ágeis para a gestão de software

37

Design Thinking

- Centrado no ser humano;
- Um processo de aprendizagem iterativo;
- Provedor de fases divergentes e convergentes;
- Uma abordagem que permite a prototipagem rápida.

38

Design Thinking

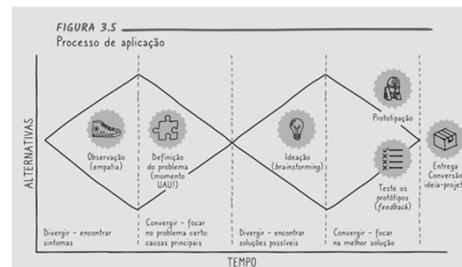
Benefícios:

- Ótimo custo-benefício;
- Proporciona respostas;
- Incentiva a criatividade;
- Estimula a empatia.



Fonte: <https://bit.ly/3ckq9h3>

39



40

Como utilizar o Design Thinking na TI

- Definição do projeto
- Investigação
- Idealização
- Prototipagem
- Teste



Fonte: do autor

41

Comparando Scrum e Design Thinking

	👍	👎
SCRUM	<ul style="list-style-type: none"> • Foco no cliente • Equipe multidisciplinar • Iteratividade entre sprints 	<ul style="list-style-type: none"> • Sem estrutura bem definida • Duração mínima de duas semanas • Sem etapa específica de teste
DESIGN THINKING	<ul style="list-style-type: none"> • Iteratividade • Etapa específica de teste 	<ul style="list-style-type: none"> • Sem estrutura bem definida • Sem duração bem definida
DESIGN SPRINT	<ul style="list-style-type: none"> • Foco no cliente • Equipe multidisciplinar • Iteratividade 	<ul style="list-style-type: none"> • Estrutura bem definida • Duração de cinco dias • Etapa específica de teste

Fonte: Paes(2020)

42

Produto viável mínimo

- Uma versão com funcionalidades básicas que permite analisar se ele cumpre seu propósito, considerando eficiência, usabilidade, aceitação no mercado e comparado à concorrência
- O MVP ajuda a fazer processos internos de forma mais rápida e eficaz
- *Minimum viable product* muito utilizado por empresas da área da inovação e por *startups*
- Validar um modelo de negócio



Fonte: do autor

43

Pivotagem

44

Desafio

Você foi delegado para adotar um recurso desafiador numa organização em que se depara por uma série de incertezas:

- O mercado está instável
- Clientes estão perdendo da confiança
- As vendas já estão conforme as projeções e histórico de anos anteriores

O que fazer nessa situação?

45

Pivotagem = mudar a direção do negócio

PARAR → FIRMAR NA BASE → PIVOTAR

- Girar em torno do próprio eixo é manter a base que está funcionando
- **Não** é fazer pequenas mudanças
- Estar preparado para perceber e reconhecer que é necessário Pivotar
- Encontrar o melhor momento para o giro do negócio
- Planejar para a criação de nova estratégia



Fonte: do autor

46

Pivotagem – motivos

Motivos para pivotar:

- Diminuição pelo interesse dos Serviços
- Tendência imposta pelas Novas tecnologias
- Queda brusca na venda do produto principal
- Produto ou serviço se tornará obsoleto



Fonte: do autor

47

Pivotagem – planejamento

Planejar o Pivot considerando:

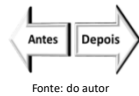
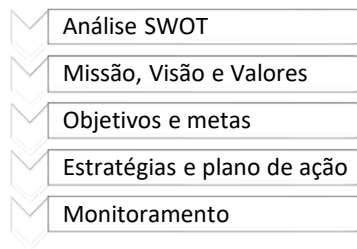
- Muitos colaboradores e maior tempo de mercado
- Colaboradores não preparados para mudanças
- Pouco recurso para suportar o tempo da pivotagem
- Grau de certeza para o Retorno do novo produto/serviço
- Fonte de informações: clientes, prospects, concorrentes, fornecedores, parceiros
- Legislações que favorecerem/impedem o novo negócio



Fonte: do autor

48

Pivotagem – passo a passo



49

Podemos utilizar duas metodologias ágeis em um projeto?

50

Recapitulando

51

Recapitulando

- Manifesto ágil
- XP
- FDD
- Crystal Clear
- Lean
- Scrum
- Design Thinking
- MVP
- Kanban

52