

Métodos Ágeis e Programação Extrema

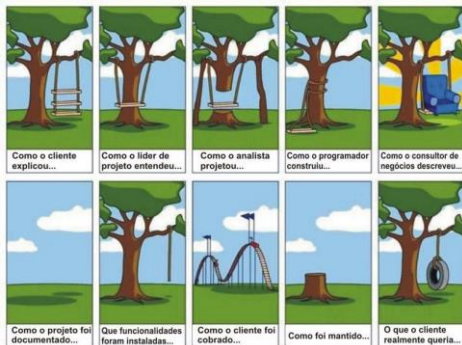
Prof. Me. Fernando Roberto Proença

Passos, 17 de junho de 2016

Roteiro

- Estatísticas sobre o Desenvolvimento de Software
- Métodos Ágeis
 - Releases e Iterações
- Programação eXtrema
 - Definição e Objetivo
 - Os valores do XP
 - Práticas do XP
 - Quando a Programação Extrema não deve ser implantada

Quando a comunicação passa por muitas pessoas, o número de falhas na transmissão da mensagem aumenta na mesma proporção...



Quantidade não define qualidade...



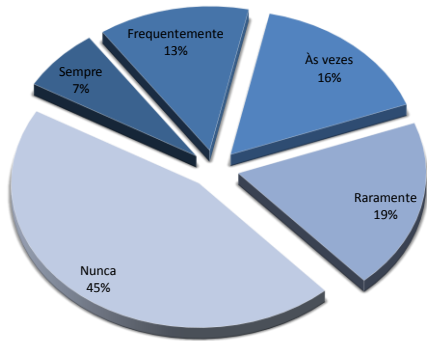
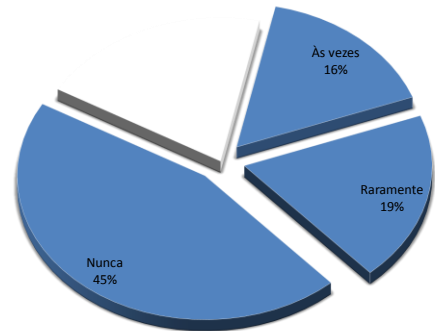
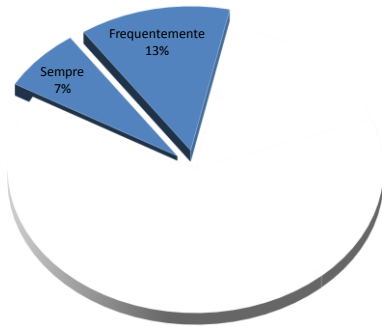
Algumas Estatísticas

- **74%** dos projetos **não terminam** nas condições planejadas;
- **46%** dos projetos **sofrem alterações** de prazo, escopo e orçamento para poder continuar a existir;
- **20%** dos projetos **falham** e não são entregues;

Algumas Estatísticas

- Apenas **1** a cada **5** projetos conquista **satisfação aceitável** dos usuários;
- **51%** das implantações de software **fracassam** como solução;
- **61%** dos usuários de sistemas se dizem **frustrados** em suas expectativas em relação à funcionalidade do software.

Utilização das funcionalidades de um software produzido... 7

80% das funcionalidades **não são relevantes...** 820% das funcionalidades produzidas **realmente importam** 9

Mudar?

Problema

ou Oportunidade



Solução?

Possível Solução: Uso de **Métodos Ágeis**

Indivíduos e interações

Software funcionando

Colaboração do cliente

Resposta às mudanças



Processos e ferramentas

Documentação abrangente

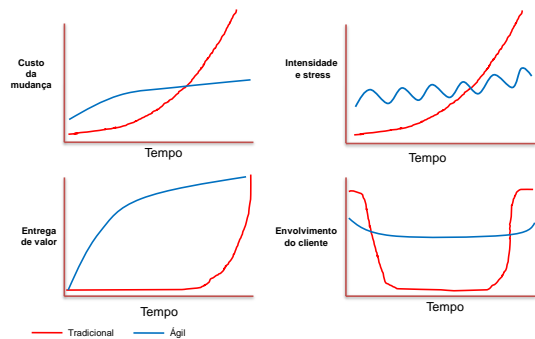
Negociação de contrato

Plano pré-estabelecido

"Embora haja valor nos itens do lado direito, nós valorizamos **mais** os itens do lado esquerdo."

<http://www.agilemanifesto.org> – 2001

Métodos Ágeis – O que muda?



Principais Métodos Ágeis

13

- Scrum
- Programação eXtrema (XP)

Métodos Ágeis

14

Modelo iterativo



Modelo incremental



Métodos Ágeis

Releases e Iterações

16

Releases e Iterações

- Uma *Release* pode ter várias Iterações
- Exemplo de *Releases* e Iterações

Release - 4 Semanas

Release 1	Release 2	Release 3	Release 4
I 1	I 2	I 3	I 4
I 5	I 6		

Iteração - 1 Semana

Iteração – Ciclo semanal

17



Jogo do Planejamento



Iteração – Ciclo semanal

18



Cliente escreve histórias



Descrição:
Como vendedor eu gostaria de
verificar se um livro está disponível
na estante para venda.

Iteração – Ciclo semanal

19



Desenvolvedores estimam



Iteração – Ciclo semanal

20

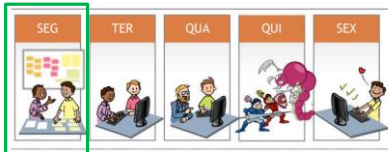


Cliente prioriza



Iteração – Ciclo semanal

21

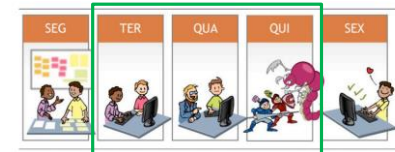


Quadro de funcionalidades



Iteração – Ciclo semanal

22

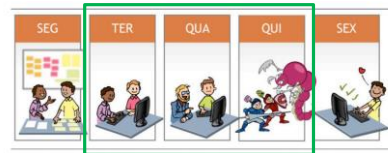


Reunião diária



Iteração – Ciclo semanal

23

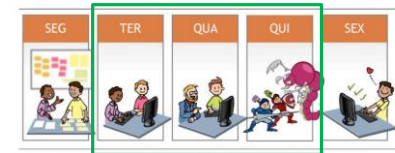


Cronograma

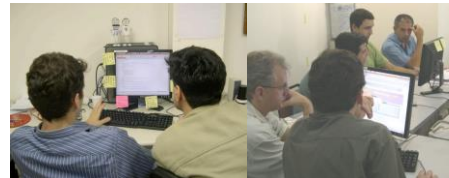


Iteração – Ciclo semanal

24



Desenvolvimento da aplicação (em par)



Iteração – Ciclo semanal

25



Acompanhamento do cliente



Iteração – Ciclo semanal

26

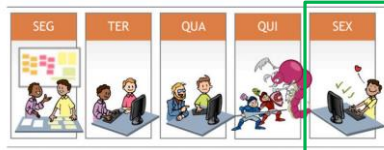


Funcionalidades terminam



Iteração – Ciclo semanal

27



Encerramento da iteração



Iteração – Ciclo semanal

28



Retrospectiva da iteração



Iteração – Ciclo semanal

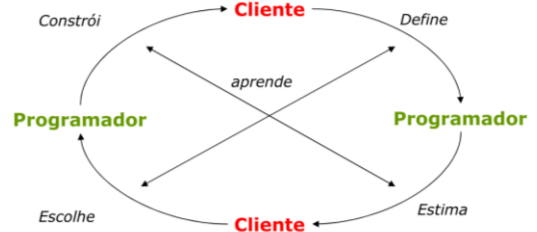
29



Inicia nova iteração

Resumindo o Ciclo de Vida de uma Release

30



Cliente **sempre** pode ver o **retorno** ao seu **investimento**

31



eXtreme Programming

O XP está sempre tentando fazer somente o que importa!

O que é eXtreme Programming (XP)?

33

É um **processo** de **desenvolvimento** de **software** voltado para:

- Projetos cujos **requisitos** são **vagos** e **mudam** com frequência;
- **Equipes pequenas** (até 12 desenvolvedores);
- Desenvolvimento **incremental** ou **iterativo**;
- Desenvolvimento de sistemas **orientados a objeto**.

XP – Objetivo

34

Desenvolver com **qualidade** e **aumentar a satisfação** do cliente.

(TELES, 2004)

Quatro valores do XP

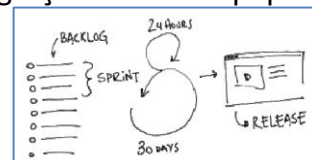
35

1. Comunicação
2. *Feedback*
3. Simplicidade
4. Coragem

1. Comunicação...

36

- Compreensão do negócio
- Satisfação do cliente
- Integração entre a equipe



Exemplo de alguns termos usados no Scrum que são de conhecimento do cliente

2. Feedback...

37

- Retorno do Cliente
- Aprendizado
- Realimentação



Usuário pilotando o sistema apresentado pela equipe, para dar feedback ao cliente. (Case Encceja)
 Imagem obtida de:
<http://visaoagil.wordpress.com/2008/10/14/projeto-agil-de-nivel-nacional-e-no-governo-o-case-encceja/>

3. Simplicidade...

38

- Redução da complexidade
- Aumento da velocidade

— Imagine que o seu cliente “quer apenas usar o telefone para ligar”



Solução 1



Solução 2

4. Coragem...

39

- Melhoria contínua
- Abraço as mudanças

Práticas do XP

40

Cliente Presente

Jogo do Planejamento

Reuniões Diárias

Programação em Par

Testes

Refatoração

Código Coletivo

Padronização

Design Simples

Metáforas

Ritmo Sustentável

Integração Contínua

Releases Curtos

1. Cliente Presente...

41

Todos fazem parte da **equipe**,
 inclusive o **próprio cliente**!

Feedback rápido...

2. Jogo do Planejamento...

42

Planejamento e Estimativas



Estórias

2. Jogo do Planejamento...

43



3. Reuniões Diárias...

44

Reuniões Rápidas

Geralmente em pé...



4. Programação em Pares...

45

Pair Programming...



5. Testes...

46

Teste primeiro! Depois codifique!



6. Refatoração...

47

Melhoria contínua do projeto...

6. Refatoração...

48

Olhando a Interface



Conhecendo o código



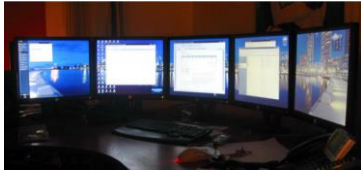
Código Refatorado



7. Código coletivo...

49

**Código
compartilhado...**



8. Padronização...

50

Padronizar é importante...



9. Projeto Simples...

51

**Simplifique sempre
que possível...**

10. Metáforas...

52

**Facilita a
comunicação...**

11. Ritmo Sustentável...

53

**Evita o
cansaço...**

12. Integração Contínua...

54

Sistema integrado

Testes de Unidade e Integração

13. Releases curtas...

55

Entrega frequente de Software Funcional

Quando a Programação Extrema não deve ser Implantada?

Quando XP não deve ser Implantada?

57

- Quando o cliente não aceita as regras do jogo.
- Quando o cliente quer uma especificação detalhada do sistema antes de começar.
- Quando os programadores não estão dispostos a seguir (todas) as regras.
- Se (quase) todos os programadores do time são medíocres.

Quando XP não deve ser Implantada?

58

- Grupos grandes (> 12 programadores).
- Quando *feedback* rápido não é possível:
 - sistema demora 6h para compilar.
 - testes demoram 12h para rodar.
 - exigência de certificação que demora meses.
- Quando o custo de mudanças é muito alto.
- Quando não é possível realizar testes.

Vencendo os Medos

59

- Escrever código
- Mudar de ideia
- Ir em frente sem saber tudo sobre o futuro
- Confiar em outras pessoas
- Mudar a arquitetura de um sistema em funcionamento
- Escrever testes.

Conclusão

60

- XP não é para todo mundo.
- Mas todo mundo pode aprender com ela.

Referências

61

- L.A. Williams and R.R Kessler. All I Really Need to Know About Pair Programming I Learned in Kindergarten. *Commun. ACM*, 43(5): 108-114, May 2000.
- TELES, Vinícius Manhães. *Extreme Programming: Aprenda como encantar seus usuários desenvolvendo software com agilidade e alta qualidade*. 1. ed. São Paulo: Novatec, 2004. 316 p.



Dúvidas?

62



Prof. Me. Fernando Roberto Proença
fernando.proenca@uemg.br