

# ENGENHARIA DE SOFTWARE

## Fundamentos de engenharia de Software

Ma. Vanessa Matias Leite

1

- Unidade de Ensino: 01
- Competência da Unidade: Conhecer os fundamentos da Engenharia de Software e da metodologia tradicional de desenvolvimento de produtos de software
- Resumo: Aplicar fundamentos de engenharia de software, metodologias ágeis e controle de versões no processo de desenvolvimento
- Palavras-chave: Engenharia de Software, metodologia ágil, XP, SCRUM, gerenciamento de configuração
- Título da Teleaula: Fundamentos de engenharia de Software
- Teleaula nº: 01

2

## Introdução à engenharia de software

3

### Fundamentos da Engenharia de Software

"Engenharia de *software* é uma disciplina cujo objetivo é produzir *software* isento de falhas, entregue dentro do prazo e orçamentos previstos, e que atenda às necessidades do cliente. Além disso, o *software* deve ser fácil de ser modificado quando as necessidades dos usuários mudarem".

4

### Princípios da Engenharia de Software

- Organização hierárquica;
- Formalidade;
- Completeza;
- Dividir para conquistar;
- Ocultação;
- Localização;
- Integridade conceitual;
- Abstração

5

### Crise do Software

- Projetos que estouram o cronograma.
- Projetos que estouram o orçamento.
- Produto final de baixa qualidade ou que não atenda aos requisitos.
- Produtos não gerenciáveis e difíceis de manter e evoluir.

6

### Eternos Mitos

- A existência de um manual de procedimentos e padrões é suficiente para a equipe produzir com qualidade;
- Se o projeto estiver atrasado, sempre é possível adicionar mais programadores para cumprir o cronograma;
- Os requisitos mudam com frequência, mas sempre é possível acomodá-los, pois o software é flexível.
- O teste do software ou sua verificação formal pode remover todos os erros;

7

## Modelos de Processos

9

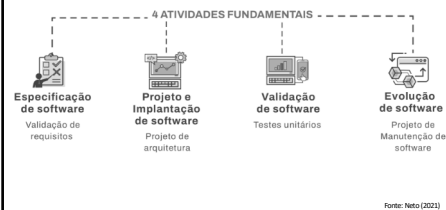
### Processos de Softwares

- Sequência de passos executados com um determinado objetivo.
- Um processo é uma receita que é seguida por um projeto.



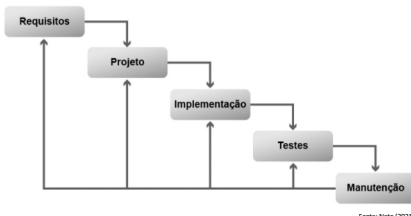
10

### Processos de Softwares



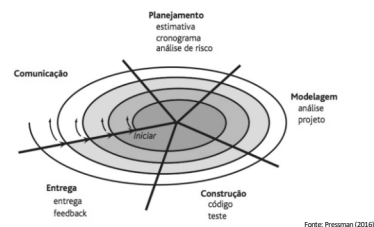
11

### Processos de Softwares

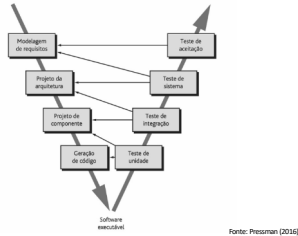


12

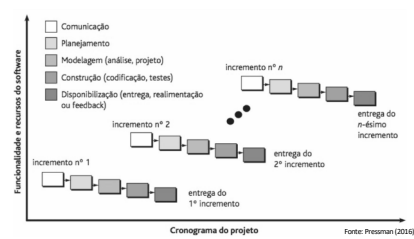
### Modelo Espiral



13

**Modelo V**

14

**Modelo Incremental**

15

**Metodologia Ágil**

16

**Metodologia Ágil**

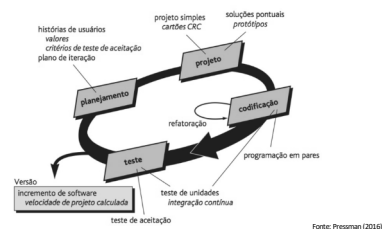
- Planejamento incremental e muito iterativa;
- Menos ênfase nas definições das atividades e mais ênfase nos fatores humanos do desenvolvimento;
- Processos ágeis se adequam a mudanças;
- Mais envolvimento dos clientes;

17

**Metodologia Ágil- Extreme Programming (XP)**

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

18

**Metodologia Ágil- Extreme Programming (XP)**

19

### Metodologia Ágil- *Extreme Programming* (XP)

- Equipe de Trabalho:
  - **Gerente de Projeto:** responsável pelos assuntos administrativos, incluindo relacionamento com o cliente;
  - **Coach:** responsável técnico pelo projeto;
  - **Analista de teste:** ajuda o cliente a escrever os testes de aceitação e fornece *feedback* para a equipe interna;

20

### Metodologia Ágil- *Extreme Programming* (XP)

- Equipe de Trabalho:
  - **Redator técnico:** ajuda a equipe de desenvolvimento a documentar o sistema;
  - **Desenvolvedor:** realiza análise, projeto e codificação do sistema. No XP, não há divisão entre estas especialidades.

21

## SP1- Coleta de Requisitos

22

### Coleta de Requisitos

Sua missão nesta atividade é propor um formulário simples, o qual deverá ser usado como modelo de coleta de requisitos para toda a organização como forma de iniciar a padronização do procedimento.

23

### Finalidade do sistema

- Site para uma empresa de vestuário, que pretende comercializar os seus produtos online.

24

### Características gerais do sistema e público-alvo.

- Site deve apresentar a empresa e colocar sua marca em evidência;
- Site deve apresentar os produtos comercializados pela empresa;
- O site deve permitir o cadastramento do usuário para conceder-lhe acesso a conteúdo exclusivo.
- O público alvo é feminino.

25

### **Ambiente operacional**

- Todos os navegadores;
- Site responsivo;
- Futuramente transformar em app;

26

### **Descrição das funcionalidades**

- Realizar compra;
- Realizar devolução;
- Cadastrar cliente;
- Alterar cadastro;

27

### **Restrições do sistema**

- Utilizar o BD MySQL;
- O site deve estar pronto em 10 meses;
- O projeto não deve ultrapassar R\$ 6000,00;

28

**Para você, existe alguma metodologia/processo que sempre será e melhor opção ?**

29

## **SCRUM**

30

### **Metodologia Ágil- Scrum**

- Método ágil mais usado atualmente.
- Aplica-se não só ao desenvolvimento de softwares como a qualquer ambiente de trabalho.
- Focado na gestão de projetos, tem como base o planejamento iterativo e incremental.
- É um framework utilizado na gestão de projetos e desenvolvimento ágil de software.

31

### Metodologia Ágil- Scrum

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

32

### Metodologia Ágil- Scrum

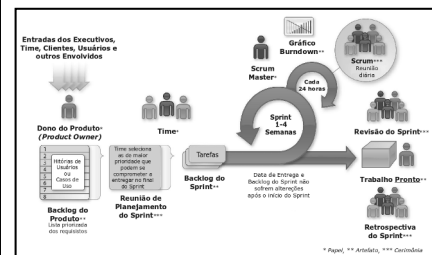
- Equipe de trabalho:
  - **Scrum Master**: trata-se de um facilitador do projeto;
  - **Product Owner**: é a pessoa responsável pelo projeto propriamente dito. Ele tem a missão de indicar os requisitos mais importantes a serem tratados nos Sprints.
  - **Scrum Team**: é a equipe de desenvolvimento,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

33

### Metodologia Ágil- 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.

34



35

### Metodologia Ágil- Scrum

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



Fonte: <https://bit.ly/39Ruc27>



Fonte: Sbracco (2012)

36

## Gestão de Configuração – Parte I

37

### Gestão de Configuração

"Arte de identificar, organizar e controlar modificações no software que está sendo criado, maximizando a produtividade e reduzindo os erros" [Pressman, 2011]

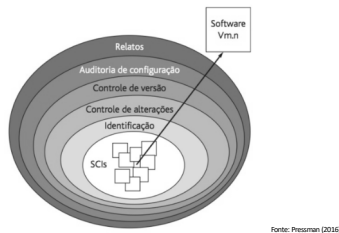
38

### Planejamento de Gerenciamento de Configuração

- Definir o que será gerenciado e o esquema que será usado para identificar as entidades;
- Estabelecer o responsável;
- Estabelecer políticas de gerenciamento de configuração;
- Especificar de ferramentas;
- Descrever a estrutura do banco de dados;

40

### Gerenciamento de Configuração



41

### Gerenciamento de Configuração

- Repositório;
- *Baselines* (linhas de base);
- *Branches*;

42

### Item de configuração

Elemento unitário ou um grupo de elementos para efeito de controle de versão.

- Código
- Documentação
- Diagramas, planos, ferramentas, casos de teste e etc

43

### Identificação de item de configuração

- Manter a rastreabilidade;
- Documentos que podem ser úteis devem ser controlados pelo gerenciamento de configuração;
- Esquemas de nomes hierarquizados;

44

## Gestão de Configuração – Parte II

45

### Banco de Dados de Configuração

- Registrar todas as informações relevantes sobre a configurações de sistemas e os itens de configuração;
- Avaliar o impacto das mudanças no sistema;
- Registro e recuperação de informações de projeto;

46

### Gerenciamento de Versões e Releases

- Preocupa-se com a identificação e a rastreabilidade das versões de um sistema;
- Uma versão é uma instância de um sistema, que difere, de alguma maneira, de outras instâncias;
- Versões com pequenas diferenças são algumas vezes chamadas de variantes;

47

### Gerenciamento de Versões e Releases

- Releases: Versão distribuída aos clientes;
- Os releases devem incorporar novas funcionalidades ou ser planejado para uma plataforma diferente de hardware;
- Há normalmente muito mais versões de um sistema do que liberações;

48

### Identificação de Versões

- Numeração de versões;
- Identificação baseada em atributos;
- Identificação orientada a mudanças;

49

### Ferramentas



Fonte: <https://bit.ly/2Wt4qpl>

50



## SP2- Kanban

51

### Alguns pontos que podemos abordar

- Ideias;
- Desenvolvimento;
- Análise de viabilidade;
- Testes;
- Selecionado;
- Finalizado;



Fonte: Camargo (2019)

52

**É possível utilizar  
duas metodologias  
ágeis em conjunto?**

53

## Recapitulando

54

### Recapitulando

- Conceito da Engenharia de Software;
- Modelos de Processos;
- Metodologia ágil:
  - XP
  - SCRUM
- Gestão de Configuração;

55



56