



"Antes de imprimir pense em sua responsabilidade e compromisso com o **MEIO AMBIENTE.**"

Engenharia de Software

Metodologias Ágeis de Desenvolvimento de Software



Givanaldo Rocha de Souza

givanaldo.rocha@ifrn.edu.br

<http://docente.ifrn.edu.br/givanaldorochoa>

Material original gentilmente cedido pelo professor Fábio Procópio



Introdução

- ❑ As Metodologias Ágeis de Desenvolvimento de Software são indicadas como sendo uma opção às abordagens tradicionais para desenvolver softwares;
- ❑ Comparadas a outras metodologias, produzem pouca documentação. É recomendado documentar o que realmente será útil;
- ❑ São recomendadas para projetos que:
 - ✓ existem muitas mudanças;
 - ✓ os requisitos são passíveis de alterações;
 - ✓ a recodificação do programa não acarreta alto custo;
 - ✓ a equipe é pequena;
 - ✓ as datas de entrega curtas acarretam alto custo;
 - ✓ o desenvolvimento rápido é fundamental.
- ❑ Em essência, as Metodologias Ágeis foram desenvolvidas com o objetivo de vencer as fraquezas percebidas e reais da Engenharia de Software (Pressman, 2010).

Metodologias Ágeis

- ➔ Introdução
- Manifesto Ágil
- Os 12 princípios
- Algumas metodologias
- XP
 - Planejamento
 - Projeto
 - Codificação
 - Teste
- DAS
- DSDM
 - Ciclo de vida
- Scrum
 - Papéis
 - Fluxo
- Família *Crystal*
- FDD
- TDD
- Seminários
- Referências



O Manifesto Ágil

- ❑ Em 2001, Kent Beck e mais 16 desenvolvedores, produtores e consultores de software, que formavam a Aliança Ágil, assinaram o Manifesto de Desenvolvimento Ágil de Software, declarando:



Estamos descobrindo melhores modos de desenvolvimento de software fazendo-o e ajudando outros a fazê-lo. Por meio desse trabalho, passamos a valorizar:

- ❑ *Indivíduos e interações* ao invés de processos e ferramentas.
- ❑ *Software funcionando* ao invés de uma documentação abrangente.
- ❑ *Colaboração do cliente* ao invés de negociação de contratos.
- ❑ *Resposta a modificações* ao invés de seguir um plano.

Isto é, ainda que haja valor nos itens à direita, valorizamos mais os itens à esquerda.

Metodologias Ágeis

Introdução

➔ Manifesto Ágil

Os 12 princípios

Algumas metodologias

XP

Planejamento

Projeto

Codificação

Teste

DAS

DSDM

Ciclo de vida

Scrum

Papéis

Fluxo

Família *Crystal*

FDD

TDD

Seminários

Referências



Os 12 princípios do Manifesto Ágil

1. Garantia da satisfação do consumidor com entrega rápida e contínua de softwares funcionais.
2. Mudanças de requisitos, mesmo no fim do desenvolvimento, ainda são bem-vindas.
3. Frequentemente são entregues softwares funcionais (semanas, ao invés de meses).
4. Desenvolvedores e pessoas relacionadas aos negócios devem trabalhar, em conjunto, até o fim do projeto.
5. Construir projetos com indivíduos motivados, dar-lhes ambiente e suporte necessários e confiar que farão seu trabalho.
6. Uma conversa face a face é o método mais eficiente e efetivo de transmitir informações para e dentro de uma equipe de desenvolvimento.

Metodologias Ágeis

Introdução

Manifesto Ágil

→ Os 12 princípios

Algumas metodologias

XP

Planejamento

Projeto

Codificação

Teste

DAS

DSDM

Ciclo de vida

Scrum

Papéis

Fluxo

Família *Crystal*

FDD

TDD

Seminários

Referências



Os 12 princípios do Manifesto Ágil

7. Software em funcionamento é a principal medida de progresso.
8. Desenvolvimento sustentável, de modo a manter um ritmo constante indefinidamente.
9. Atenção contínua para com a excelência técnica e para com bons projetos aumenta a agilidade.
10. Simplicidade – a arte de maximizar a quantidade de trabalho não efetuado – é essencial.
11. As melhores arquiteturas, requisitos e projetos emergem de equipes auto-organizáveis.
12. Em intervalos regulares, a equipe deve refletir sobre como se tornar mais eficiente.

Metodologias Ágeis

Introdução

Manifesto Ágil

→ Os 12 princípios

Algumas metodologias

XP

Planejamento

Projeto

Codificação

Teste

DAS

DSDM

Ciclo de vida

Scrum

Papéis

Fluxo

Família *Crystal*

FDD

TDD

Seminários

Referências



Algumas metodologias

□ Em seu livro, Pressman (2010) apresenta as seguintes metodologias:

- *XP (Extreme Programming)*
- *DAS (Desenvolvimento Adaptativo de Software)*
- *DSDM (Dynamic Software Development Method)*
- *Scrum*
- *Crystal*
- *FDD (Feature Driven Development)*
- *Modelagem Ágil (AM)*
- *Processo Unificado Ágil (AUP)*

Metodologias Ágeis

Introdução

Manifesto Ágil

Os 12 princípios

→ Algumas metodologias
XP

Planejamento

Projeto

Codificação

Teste

DAS

DSDM

Ciclo de vida

Scrum

Papéis

Fluxo

Família *Crystal*

FDD

TDD

Seminários

Referências



XP (*Extreme Programming*)

- ❑ Utiliza OO como paradigma de desenvolvimento;
- ❑ Inclui um conjunto de regras e práticas com base nas seguintes atividades:
 - ❑ Planejamento
 - ❑ Projeto
 - ❑ Codificação
 - ❑ Teste



Metodologias Ágeis

Introdução
Manifesto Ágil
Os 12 princípios
Algumas metodologias
→ XP
Planejamento
Projeto
Codificação
Teste
DAS
DSDM
Ciclo de vida
Scrum
Papéis
Fluxo
Família *Crystal*
FDD
TDD
Seminários
Referências



XP: Planejamento

- ❑ Criação de um conjunto de “histórias de usuários” descrevendo as características e funcionalidades requeridas pelo software que será construído;
- ❑ As histórias (semelhantes aos casos de uso) são escritas pelos clientes e colocadas em cartões de indexação;
- ❑ O cliente atribui uma prioridade à cada história;
- ❑ Os desenvolvedores analisam cada história e atribuem um custo a cada uma delas, com base em número de semanas necessárias para o seu desenvolvimento;

Metodologias Ágeis

Introdução
Manifesto Ágil
Os 12 princípios
Algumas metodologias

XP
→ Planejamento
Projeto
Codificação
Teste

DAS
DSDM
Ciclo de vida

Scrum
Papéis
Fluxo

Família *Crystal*
FDD
TDD

Seminários
Referências



XP: Planejamento (Kanban)



Metodologias Ágeis

- Introdução
- Manifesto Ágil
- Os 12 princípios
- Algumas metodologias
- XP
 - Planejamento
 - Projeto
 - Codificação
 - Teste
- DAS
- DSDM
 - Ciclo de vida
- Scrum
 - Papéis
 - Fluxo
- Família *Crystal*
- FDD
- TDD
- Seminários
- Referências



XP: Planejamento (Kanban - Trello)

Metodologias Ágeis

Introdução
Manifesto Ágil
Os 12 princípios
Algumas metodologias
XP

Planejamento
Projeto
Codificação
Teste

DAS

DSDM

Ciclo de vida

Scrum

Papéis

Fluxo

Família *Crystal*

FDD

TDD

Seminários

Referências



XP: Planejamento

- ❑ Se a história precisar de mais de 3 semanas para desenvolvimento, é solicitado ao cliente que ela seja dividida em histórias menores;
- ❑ Desenvolvidas em 3 modos:
 - 1) Todas as histórias serão implementadas imediatamente (dentro de poucas semanas).
 - 2) As histórias com valor mais alto serão antecipadas no cronograma e implementadas primeiro.
 - 3) As histórias de maior risco serão antecipadas no cronograma e implementadas primeiro.
- ❑ Com o avanço do projeto, o cliente pode adicionar novas histórias, mudar a sua prioridade, subdividi-la e eliminá-las.

Metodologias Ágeis

- Introdução
- Manifesto Ágil
- Os 12 princípios
- Algumas metodologias
- XP
 - Planejamento
 - Projeto
 - Codificação
 - Teste
- DAS
- DSDM
 - Ciclo de vida
- Scrum
 - Papéis
 - Fluxo
- Família *Crystal*
- FDD
- TDD
- Seminários
- Referências



XP: Projeto

- ❑ Segue rigorosamente o KIS (*keep it simple*)
- ❑ Estimula o uso de cartões CRC (*Classe, Responsabilidade e Colaboração*) para a identificação e organização das classes OO relevantes para o incremento do software
- ❑ Cartões CRC permitem a descrição dos conceitos identificados na metáfora na forma de classes.
 - ❑ Responsabilidades são identificadas para cada classe.
 - ❑ As colaborações determinam as interações entre classes.
 - ❑ Os cartões permitem que o todo o time possa colaborar com o design.

ID: 1	Classe: Usuários	Super Classe:
Responsabilidades		Colaborações
Conhecer seus dados		
Conhecer se está ativo		
Conhecer seu estado civil		Civil
Conhecer sua renda		Renda
Conhecer sua escolaridade		Escolaridade

Metodologias Ágeis

Introdução
 Manifesto Ágil
 Os 12 princípios
 Algumas metodologias
 XP
 Planejamento
 Projeto
 Codificação
 Teste
 DAS
 DSDM
 Ciclo de vida
 Scrum
 Papéis
 Fluxo
 Família *Crystal*
 FDD
 TDD
 Seminários
 Referências



XP: Projeto

- ❑ Os cartões CRC são o único produto de trabalho do projeto;
- ❑ Caso seja identificado um problema difícil na história, recomenda-se a criação imediata de um protótipo operacional daquela parte do projeto. Denominado *Solução de Ponta*.
- ❑ Encoraja a **refatoração**.
 - ✓ Técnica que altera a estrutura do sistema sem modificar o comportamento externo.

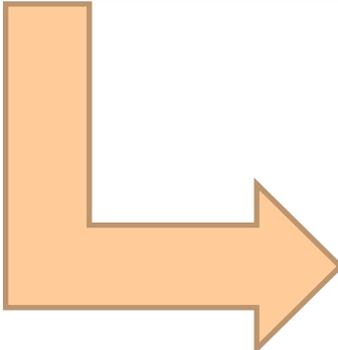
Metodologias Ágeis

Introdução
Manifesto Ágil
Os 12 princípios
Algumas metodologias
XP
Planejamento
Projeto
Codificação
Teste
DAS
DSDM
Ciclo de vida
Scrum
Papéis
Fluxo
Família *Crystal*
FDD
TDD
Seminários
Referências

XP: Projeto - Refatoração (extração de método)



```
/** Salva o produto no banco de dados. */  
public void save()  
{  
    // Verifica propriedades  
    if (this.getName() == null)  
    {  
        throw new Exception("Falta nome");  
    }  
    else if (this.getDescription() == null)  
    {  
        throw new Exception("Falta a descrição");  
    }  
  
    this.getDatabase().save(this);  
}
```



```
/** Salva o produto no banco de dados. */  
public void save()  
{  
    this.checkProperties();  
    this.getDatabase().save(this);  
}  
  
/** Verifica as propriedades do produto. */  
private void checkProperties()  
{  
    if (this.getName() == null)  
    {  
        throw new Exception("Falta nome do produto.");  
    }  
    else if (this.getDescription() == null)  
    {  
        throw new Exception("Falta a descrição do produto.");  
    }  
}
```

Metodologias Ágeis



- Introdução
- Manifesto Ágil
- Os 12 princípios
- Algumas metodologias
- XP
 - Planejamento
 - Projeto
 - Codificação
 - Teste
- DAS
- DSDM
- Ciclo de vida
- Scrum
- Papéis
- Fluxo
- Família *Crystal*
- FDD
- TDD
- Seminários
- Referências



XP: Codificação

- ❑ Depois que as histórias forem desenvolvidas e o início do projeto for feito, recomenda-se não iniciar a programação;
 - ❑ Elemento chave do XP.
- ❑ É recomendado realizar testes unitários sobre cada uma das histórias que serão incluídas na versão atual.
- ❑ Depois de os testes unitários terem sido criados, o desenvolvedor está focado no que deve ser implementado.
- ❑ Programação em pares:
 - ❑ duas pessoas trabalhando juntas na mesma máquina;
 - ❑ cada pessoa fica encarregada de uma atividade;
 - ❑ quando o trabalho dos programadores é completado, é feita uma integração com o trabalho de outros;
 - ❑ existe uma equipe responsável pela integração.

Metodologias Ágeis

Introdução
Manifesto Ágil
Os 12 princípios
Algumas metodologias
XP
Planejamento
Projeto
Codificação
Teste
DAS
DSDM
Ciclo de vida
Scrum
Papéis
Fluxo
Família *Crystal*
FDD
TDD
Seminários
Referências





XP: Teste

- ❑ São aplicados os testes unitários.
- ❑ Os testes de aceitação (ou teste de cliente) são especificados sob a ótica do cliente e abrangem as características e as funcionalidades do sistema global visíveis e passíveis de revisão.
- ❑ *"Resolver pequenos problemas a cada intervalo de umas poucas horas leva menos tempo do que resolver grandes problemas perto da data de entrega", Wells (1999) apud Pressman(2010).*

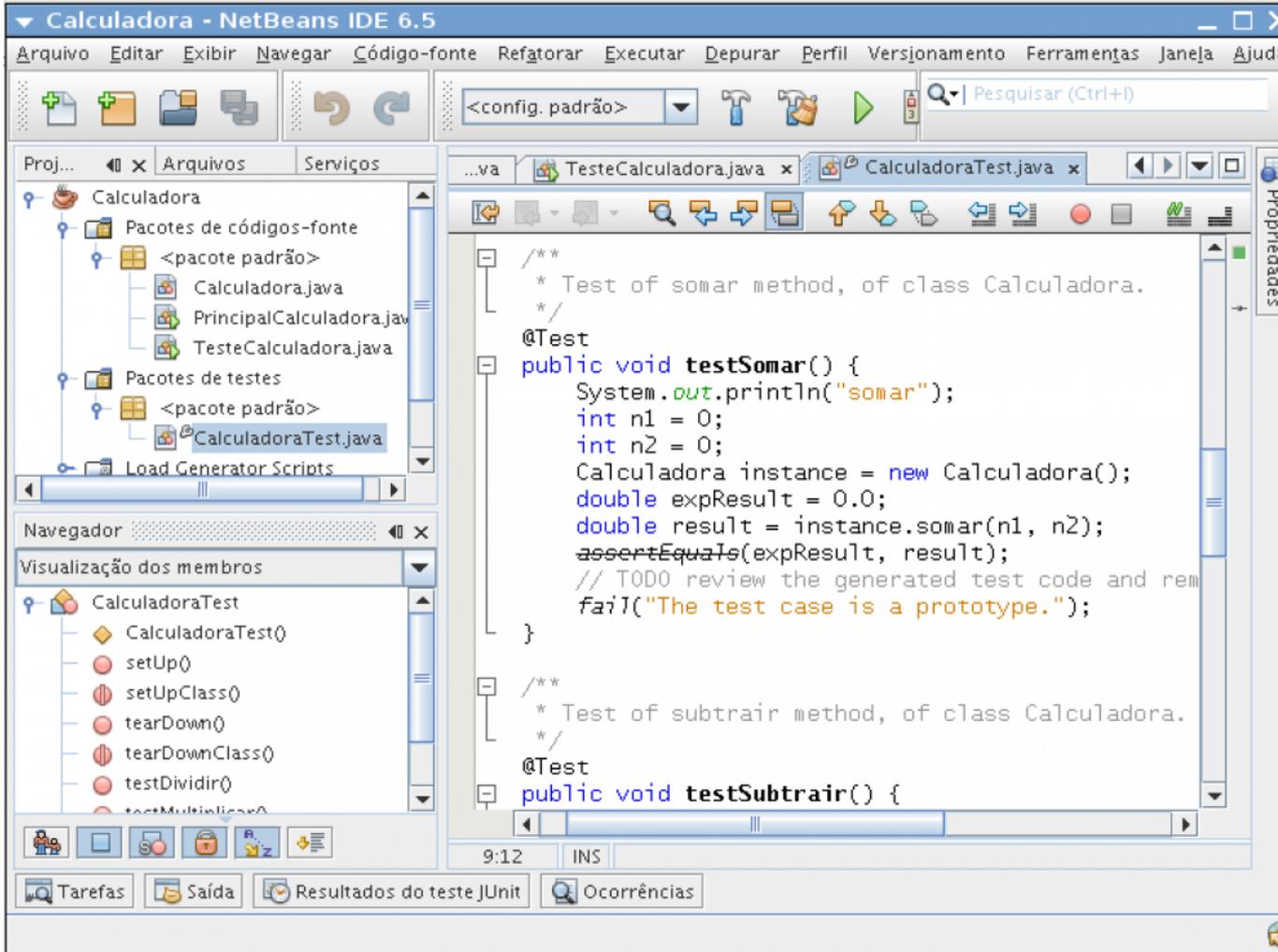
Metodologias Ágeis

Introdução
Manifesto Ágil
Os 12 princípios
Algumas metodologias
XP
Planejamento
Projeto
Codificação
Teste
DAS
DSDM
Ciclo de vida
Scrum
Papéis
Fluxo
Família Crystal
FDD
TDD
Seminários
Referências





XP: Teste Unitário (exemplo)



```

/**
 * Test of somar method, of class Calculadora.
 */
@Test
public void testSomar() {
    System.out.println("somar");
    int n1 = 0;
    int n2 = 0;
    Calculadora instance = new Calculadora();
    double expectedResult = 0.0;
    double result = instance.somar(n1, n2);
    assertEquals(expectedResult, result);
    // TODO review the generated test code and remove the default
    fail("The test case is a prototype.");
}

/**
 * Test of subtrair method, of class Calculadora.
 */
@Test
public void testSubtrair() {
    
```

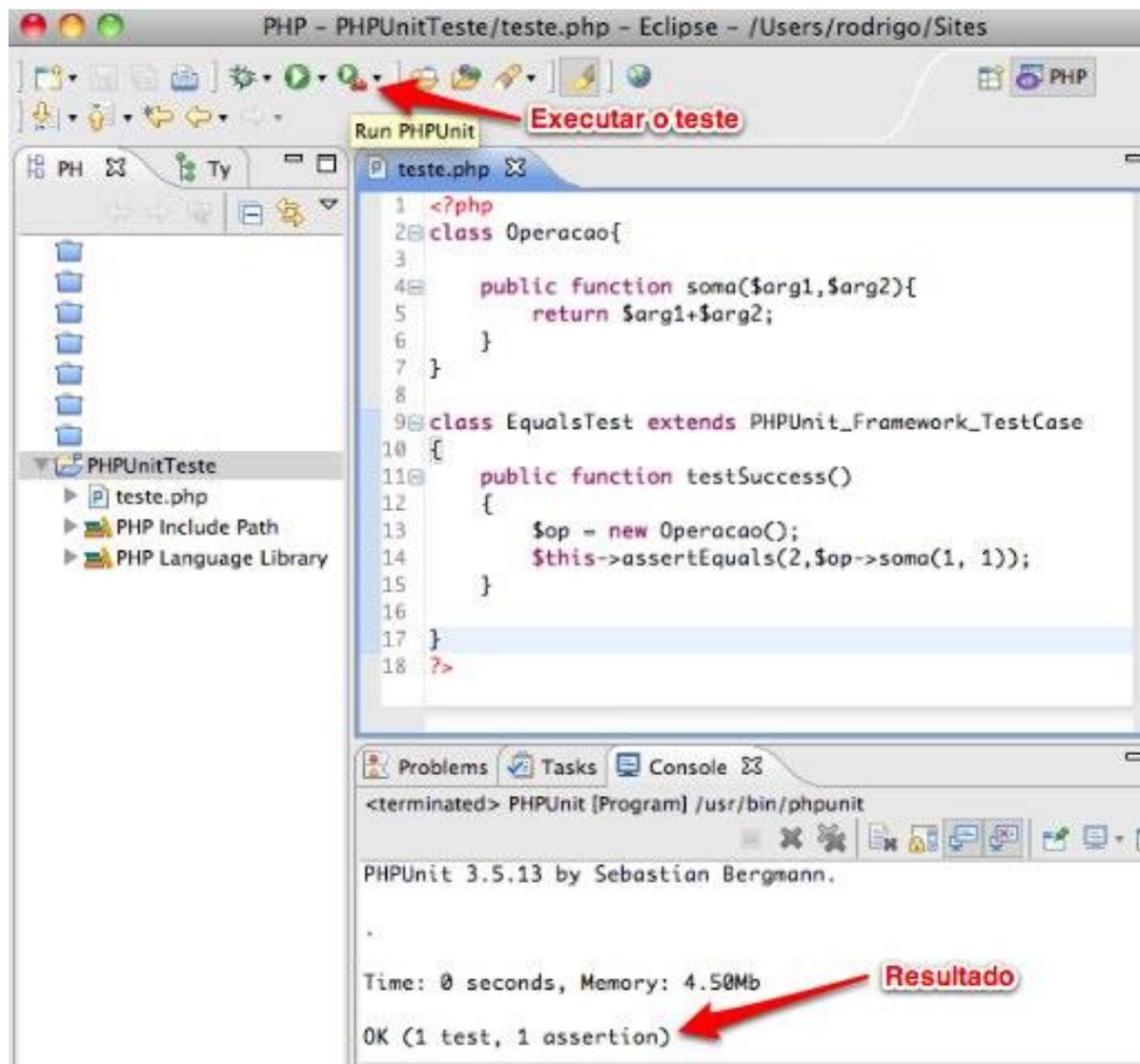
Metodologias Ágeis

- Introdução
- Manifesto Ágil
- Os 12 princípios
- Algumas metodologias XP
 - Planejamento
 - Projeto
 - Codificação
 - Teste
- DAS
- DSDM
- Ciclo de vida
- Scrum
- Papéis
- Fluxo
- Família Crystal
- FDD
- TDD
- Seminários
- Referências





XP: Teste Unitário (exemplo)



```

1 <?php
2 class Operacao{
3
4     public function soma($arg1,$arg2){
5         return $arg1+$arg2;
6     }
7 }
8
9 class EqualsTest extends PHPUnit_Framework_TestCase
10 {
11     public function testSuccess()
12     {
13         $op = new Operacao();
14         $this->assertEquals(2,$op->soma(1, 1));
15     }
16
17 }
18 ?>
    
```

Time: 0 seconds, Memory: 4.50Mb

OK (1 test, 1 assertion)

Metodologias Ágeis

- Introdução
- Manifesto Ágil
 - Os 12 princípios
- Algumas metodologias
 - XP
 - Planejamento
 - Projeto
 - Codificação
 - Teste
 - DAS
 - DSDM
 - Ciclo de vida
 - Scrum
 - Papéis
 - Fluxo
 - Família *Crystal*
 - FDD
 - TDD
 - Seminários
 - Referências



DAS

- ❑ Acrônimo de Desenvolvimento Adaptativo de Software ou, ASD (*Adaptative Software Development*).
- ❑ Proposto por *Highsmith* para auxiliar no desenvolvimento de sistemas e softwares complexos.
- ❑ Concentra-se na colaboração humana e na auto-organização da equipe.
 - ✓ É uma propriedade de sistemas adaptativos.

Metodologias Ágeis

- Introdução
- Manifesto Ágil
- Os 12 princípios
- Algumas metodologias
- XP
 - Planejamento
 - Projeto
 - Codificação
 - Teste
- ➔ DAS
- DSDM
 - Ciclo de vida
- Scrum
 - Papéis
 - Fluxo
- Família *Crystal*
- FDD
- TDD
- Seminários
- Referências



DAS

- ❑ Incorpora três fases:
 - ❑ Especulação
 - ✓ Declara a missão do projeto;
 - ✓ Identifica as restrições do projeto;
 - ✓ Realiza o levantamento dos requisitos básicos.
 - ❑ Colaboração
 - ✓ Filosofia de que pessoas motivadas trabalhando juntas multiplicam seus talentos e resultados.
 - ❑ Aprendizado
 - ✓ Clientes/usuários informam *feedback*;
 - ✓ Revisão dos componentes de software desenvolvidos;
 - ✓ Avaliação do desempenho da equipe DAS.

Metodologias Ágeis

- Introdução
- Manifesto Ágil
- Os 12 princípios
- Algumas metodologias
- XP
 - Planejamento
 - Projeto
 - Codificação
 - Teste
- ➔ DAS
- DSDM
 - Ciclo de vida
- Scrum
 - Papéis
 - Fluxo
- Família *Crystal*
- FDD
- TDD
- Seminários
- Referências



DSDM

- ❑ Acrônimo de *Dynamic Systems Development Method*;
- ❑ "*Fornece um arcabouço para construir e manter sistemas que satisfazem às restrições de prazo apertadas por meio do uso de prototipagem incremental em um ambiente controlado de projeto*", CS3 Consulting Services apud Pressman(2010);
- ❑ O DSDM *Consortium* é um grupo mundial de empresas que definiu um modelo ágil de processo chamada Ciclo de vida DSDM, com as seguintes atividades:
 - ❑ Estudo de viabilidade;
 - ❑ Estudo do negócio;
 - ❑ Iteração do modelo funcional;
 - ❑ Iteração de projeto e construção e;
 - ❑ Implementação.

Metodologias Ágeis

- Introdução
- Manifesto Ágil
- Os 12 princípios
- Algumas metodologias
- XP
 - Planejamento
 - Projeto
 - Codificação
 - Teste
- DAS
- ➔ DSDM
 - Ciclo de vida
- Scrum
 - Papéis
 - Fluxo
- Família *Crystal*
- FDD
- TDD
- Seminários
- Referências



Ciclo de vida DSDM

- ❑ Estudo de viabilidade
 - ✓ Define requisitos básicos e restrições do negócio;
 - ✓ Avalia se é viável desenvolver a aplicação usando DSDM.
- ❑ Estudo do negócio
 - ✓ Identifica os requisitos funcionais;
 - ✓ Define a arquitetura básica da aplicação;
 - ✓ Identifica os requisitos de manutenibilidade da aplicação.
- ❑ Iteração do modelo funcional
 - ✓ Constrói um conjunto de protótipos incrementais.
 - ✓ Objetiva adquirir requisitos adicionais com *feedbacks* dos usuários à medida que usam o protótipo.
- ❑ Iteração de projeto e construção
 - ✓ Verifica os protótipos construídos garantindo que cada um tenha passado por engenharia.
- ❑ Implementação
 - ✓ Coloca o último incremento do software no ambiente de produção, apesar de ele poder ainda não estar completo e novas modificações serem solicitadas.

Metodologias Ágeis

Introdução
Manifesto Ágil
Os 12 princípios
Algumas metodologias
XP
Planejamento
Projeto
Codificação
Teste
DAS
DSDM
Ciclo de vida
Scrum
Papéis
Fluxo
Família *Crystal*
FDD
TDD
Seminários
Referências





Scrum

- ❑ Apresenta princípios coerentes com os do Manifesto Ágil;
- ❑ Desenvolvido na década de 90 por Jeff Sutherland.
- ❑ Enfatiza o uso de um conjunto de “padrões de processo de software” ideais para projetos com prazos apertados, requisitos mutantes e negócios críticos;
- ❑ Baseado em ciclos de 30 dias (chamados de *Sprints*), o *Scrum* trabalha para alcançar objetivos bem definidos;
- ❑ Os objetivos são representados no *Product Backlog*, que é uma lista de atividades a serem realizadas;
- ❑ Papéis:
 - ❑ Time
 - ❑ *Product owner*
 - ❑ *Scrum master*

Metodologias Ágeis

- Introdução
- Manifesto Ágil
 - Os 12 princípios
- Algumas metodologias
 - XP
 - Planejamento
 - Projeto
 - Codificação
 - Teste
 - DAS
 - DSDM
 - Ciclo de vida
 - ➔ **Scrum**
 - Papéis
 - Fluxo
 - Família *Crystal*
 - FDD
 - TDD
 - Seminários
 - Referências



Scrum: papéis

- ❑ Time
 - ❑ grupo pequeno, normalmente, entre 5 e 9 pessoas;
 - ❑ deve ser comprometido com o trabalho a fim de atingir a meta de um *Sprint*;
 - ❑ desenvolve e produz com qualidade;
 - ❑ deve ser cada vez mais auto-gerenciável e multidisciplinar.
- ❑ Product Owner
 - ❑ faz o intermédio entre o cliente e o fornecedor do produto
 - ❑ boa noção do produto e das necessidades do cliente;
 - ❑ responsável por atualizar o *Product Backlog*.
- ❑ Scrum Master
 - ❑ um líder, mediador e facilitador;
 - ❑ remove impedimentos da equipe assegurando que as práticas *Scrum* estão sendo executadas com eficiência.

Metodologias Ágeis

Introdução
Manifesto Ágil
Os 12 princípios
Algumas metodologias
XP
Planejamento
Projeto
Codificação
Teste
DAS
DSDM
Ciclo de vida
Scrum
Papéis
Fluxo
Família *Crystal*
FDD
TDD
Seminários
Referências





Scrum: Fluxo de processo

- ❑ Definição do *Backlog*
 - ❑ funcionalidades ou mudanças no produto são feitas pelo *Product Owner* no *Product Backlog*;
 - ❑ a lista é priorizada para refletir a necessidade dos clientes ou demandas do mercado;
- ❑ Sprints
 - ❑ unidades de trabalho necessárias para atender os itens do *Product Backlog*, normalmente, são 30 dias.
- ❑ Reuniões
 - ❑ diárias, aproximadamente 15 minutos;
 - ❑ todos os membros do time respondem às perguntas:
 1. *O que você fez ontem?*
 2. *O que pretende fazer hoje?*
 3. *Que impedimentos estão lhe atrapalhando?*
- ❑ Revisões
 - ❑ o time apresenta os resultados ao *Product Owner* e demais interessados;
 - ❑ os itens do *backlog* são considerados prontos e inicia-se um novo *sprint*.

Metodologias Ágeis

Introdução
Manifesto Ágil
Os 12 princípios
Algumas metodologias
XP
Planejamento
Projeto
Codificação
Teste
DAS
DSDM
Ciclo de vida
Scrum
Papéis
Fluxo
Família *Crystal*
FDD
TDD
Seminários
Referências





Scrum: Fluxo de processo



Metodologias Ágeis

- Introdução
- Manifesto Ágil
- Os 12 princípios
- Algumas metodologias
- XP
 - Planejamento
 - Projeto
 - Codificação
 - Teste
- DAS
- DSDM
 - Ciclo de vida
- Scrum
 - Papéis
 - Fluxo
- Família *Crystal*
- FDD
- TDD
- Seminários
- Referências



Família *Crystal*

- ❑ Criada por Cockburn e Highsmith, contempla um conjunto de metodologias, cada qual com elementos centrais que são comuns a todas, papéis, padrões de processos, produtos de trabalho e práticas específicas de cada uma (Pressman, 2010);
- ❑ Tem como objetivo permitir que equipes ágeis selecionem o membro da família mais apropriado para o seu projeto e ambiente.
- ❑ Cada método *Crystal* é caracterizado por uma cor.
- ❑ 4 parâmetros determinam o método de desenvolvimento:
 - ✓ Tamanho da equipe.
 - ✓ Localização geográfica.
 - ✓ Criticalidade/Segurança
 - ✓ Recursos.

Metodologias Ágeis

Introdução

Manifesto Ágil

Os 12 princípios

Algumas metodologias

XP

Planejamento

Projeto

Codificação

Teste

DAS

DSDM

Ciclo de vida

Scrum

Papéis

Fluxo

➔ Família *Crystal*

FDD

TDD

Seminários

Referências



Crystal Clear

- ❑ É uma metodologia leve, para equipes de 1 a 8 pessoas. Podendo chegar até 12 casos especiais.
 - ✓ Yellow: 10 a 20 membros.
 - ✓ Orange: 20 a 50 membros.
 - ✓ Red: 50 a 100 membros.
- ❑ Cada um dos métodos com graus de gerenciamento e de comunicação ajustados de acordo com o tamanho da equipe.
- ❑ Especificação e projeto são feitos informalmente usando quadros publicamente visíveis.
- ❑ A metodologia é propositalmente pouco definida. Para permitir que cada projeto implemente as atividades que lhes pareçam mais adequadas. Fornecendo um mínimo de suporte útil a documentação e comunicação.

Metodologias Ágeis

Introdução

Manifesto Ágil

Os 12 princípios

Algumas metodologias

XP

Planejamento

Projeto

Codificação

Teste

DAS

DSDM

Ciclo de vida

Scrum

Papéis

Fluxo



Família *Crystal*

FDD

TDD

Seminários

Referências



FDD – *Feature Driven Development*

- ❑ Define característica como sendo "*uma função valorizada pelo cliente que pode ser implementada em duas semanas ou menos*", Coad (1999) *apud* Pressman (2010);
- ❑ Os benefícios em se utilizar a filosofia de características são:
 - ✓ os usuários podem descrevê-las mais facilmente;
 - ✓ podem ser organizadas de forma hierárquica;
 - ✓ desenvolvimento a cada duas semanas;
 - ✓ facilidade em analisar projeto e código;
 - ✓ projeto e cronograma são guiados pela hierarquia de características, ao invés de um conjunto de tarefas de engenharia de software adotado arbitrariamente.
- ❑ Definição:
<ação> ○ <resultado> <por | para | de | a> um <objeto>
- ❑ Exemplos:
 - ✓ Gera o relatório de venda para um determinado período;
 - ✓ Exibe as especificações técnicas de um produto;
 - ✓ Adiciona o produto a um carrinho de compras.

Metodologias Ágeis

Introdução

Manifesto Ágil

Os 12 princípios

Algumas metodologias

XP

Planejamento

Projeto

Codificação

Teste

DAS

DSDM

Ciclo de vida

Scrum

Papéis

Fluxo

Família *Crystal*

FDD

TDD

Seminários

Referências



TDD – Test Driven Development

- ❑ Técnica de desenvolvimento de software que baseia em um ciclo curto de repetições:
 - ✓ O desenvolvedor escreve um caso de teste automatizado que define uma melhoria desejada ou uma nova funcionalidade.
 - ✓ É produzido código que possa ser validado pelo teste para posteriormente o código ser refatorado para um código sob padrões aceitáveis.
- ❑ Kent Beck, considerado o criador da técnica, declarou em 2003 que TDD encoraja designs de código simples e inspira confiança.
- ❑ Através de TDD, programadores podem aplicar o conceito de melhorar e depurar código legado desenvolvido a partir de técnicas antigas.

Metodologias Ágeis

Introdução

Manifesto Ágil

Os 12 princípios

Algumas metodologias

XP

Planejamento

Projeto

Codificação

Teste

DAS

DSDM

Ciclo de vida

Scrum

Papéis

Fluxo

Família *Crystal*

FDD

TDD

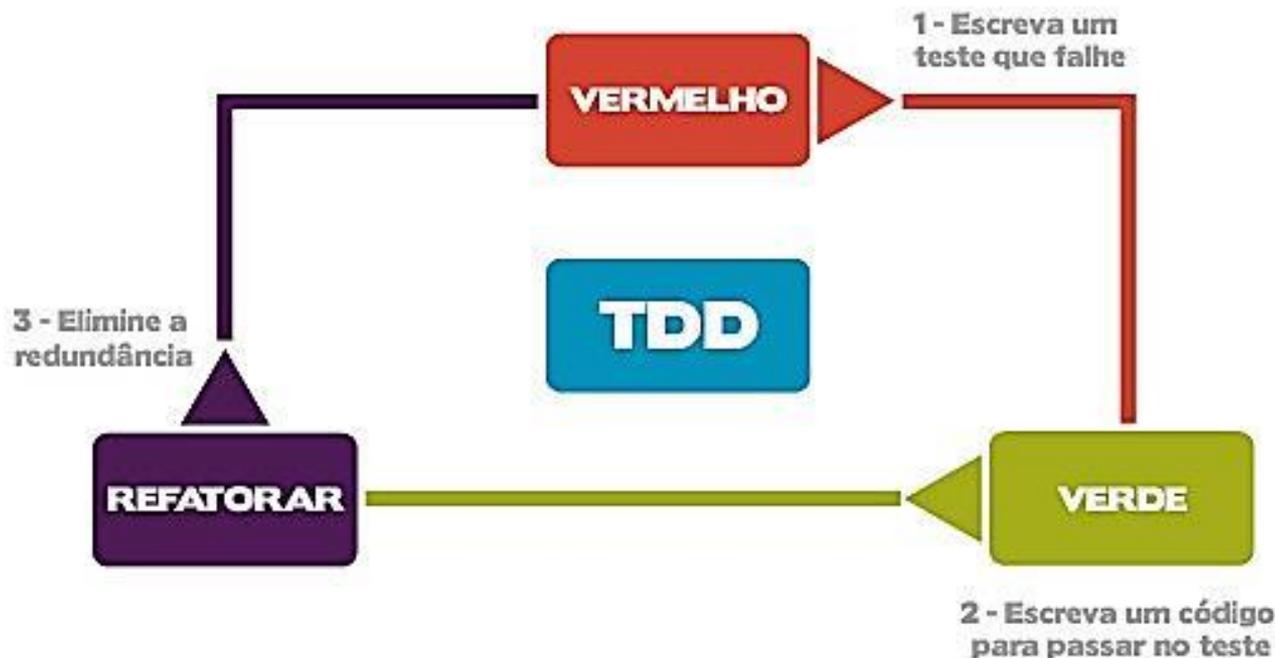
Seminários

Referências





TDD – Test Driven Development



Metodologias Ágeis

- Introdução
- Manifesto Ágil
- Os 12 princípios
- Algumas metodologias
- XP
 - Planejamento
 - Projeto
 - Codificação
 - Teste
- DAS
- DSDM
 - Ciclo de vida
- Scrum
 - Papéis
 - Fluxo
- Família Crystal
- FDD
- TDD
- Seminários
- Referências



- Escreva um teste**, antes mesmo de escrever o código que este teste consome.
- Faça o teste funcionar**, escrevendo o código do qual o teste depende, mesmo que seja um código ruim.
- Refatore**, eliminando duplicações de código, tanto nos testes quanto nas implementações.

Seminários

**APRESENTAÇÃO
19/08 (3,0) PONTOS**



❑ A turma deverá apresentar seminários sobre as seguintes metodologias ágeis:

- 1) SCRUM ()
- 2) XP ()
- 3) FDD ()
- 4) BDD ()
- 5) TDD ()

- ❖ Deve conter mais detalhes do que os mostrados na aula.
- ❖ Devem ser apresentados exemplos reais de aplicação e de documentação gerada.
- ❖ Listar empresas conhecidas que utilizam o respectivo processo.
- ❖ Mostrar como a Orientação a Objetos é utilizada.

Metodologias Ágeis

Introdução
Manifesto Ágil
Os 12 princípios
Algumas metodologias
XP
Planejamento
Projeto
Codificação
Teste
DAS
DSDM
Ciclo de vida
Scrum
Papéis
Fluxo
Família *Crystal*
FDD
TDD
→ Seminários
Referências



Referências

SOARES, Michel dos Santos. **Comparação entre Metodologias Ágeis e Tradicionais para o Desenvolvimento de Software**. Disponível em:

<http://www.dcc.ufla.br/infocomp/artigos/v3.2/art02.pdf>. Acessado em: 26 jan. 2011.

KIIOSKEA.NET. **Métodos ágeis (RAD, XP)**. Disponível em: <http://pt.kioskea.net/contents/genie-logiciel/methodes-agiles.php3>. Acessado em: 26 jan. 2011.

Manifesto para o desenvolvimento ágil de software. Disponível em:

<http://manifestoagil.com.br/index.html>. Acessado em: 26 jan. 2011.

WIKIPÉDIA. **Desenvolvimento ágil de software**. Disponível em:

http://pt.wikipedia.org/wiki/Desenvolvimento_%C3%A1gil_de_software. Acessado em: 26 jan. 2011.

WIKIPÉDIA. **Manifesto ágil**. Disponível em: http://pt.wikipedia.org/wiki/Manifesto_%C3%A1gil.

Acessado em: 26 jan. 2011.

FOWLER, Martin. **A Nova Metodologia**. Disponível em: <http://simplus.com.br/artigos/a-nova-metodologia/#N1F0>. Acessado em: 26 jan. 2011.

CÂMARA, Fábio. **Um cardápio de metodologias ágeis**. Disponível em:

http://imasters.com.br/artigo/7396/gerencia/um_cardapio_de_metodologias_ageis/. Acessado em: 26 jan. 2011.

PRESSMAN, Roger S. **Engenharia de Software**. Mc Graw Hill, 6 ed, Porto Alegre, 2010.

THAMIEL, Thiago. **Entendendo Scrum**. Disponível em:

<http://thiagothamiel.wordpress.com/category/desenvolvimento-agil/page/2/>. Acessado em: 02 fev. 2011.

Metodologias Ágeis

Introdução

Manifesto Ágil

Os 12 princípios

Algumas metodologias

XP

Planejamento

Projeto

Codificação

Teste

DAS

DSDM

Ciclo de vida

Scrum

Papéis

Fluxo

Família *Crystal*

FDD

TDD

Seminários

Referências