

Testes

12 de março de 2019



IME-USP

Código Legado: pré-existente, originário de implementações antigas.



Código Legado: pré-existente, originário de implementações antigas. *Em geral significa que é bem sucedido!*



Código Legado: pré-existente, originário de implementações antigas. *Em geral significa que é bem sucedido!*

Código Belo: Código de longa duração e fácil de evoluir.



Código Legado: pré-existente, originário de implementações antigas. *Em geral significa que é bem sucedido!*

Código Belo: Código de longa duração e fácil de evoluir.

O problema é o código de curta duração inesperada!



A qualidade tem dois aspectos: atender ao cliente e facilidade de criação e manutenção.

Os testes também são de dois tipos:



A qualidade tem dois aspectos: atender ao cliente e facilidade de criação e manutenção.

Os testes também são de dois tipos:

- Verificação — Você construiu corretamente a coisa?



A qualidade tem dois aspectos: atender ao cliente e facilidade de criação e manutenção.

Os testes também são de dois tipos:

- Verificação — Você construiu corretamente a coisa?
- Validação — Você construiu a coisa certa?



- Unidade (função ou método, ou uma classe)
- Módulo (ditto, pode ser classes interrelacionadas)
- Integração (interfaces e uso integrado)
- Sistema (funcionamento total)
- Aceitação (crivo do cliente)

Programação extrema sugere que os testes sejam escritos *antes* do código propriamente dito.



Testes com XP



IME-USP

Test Driven Development (TDD)

Desenvolvimento guiado por testes. Serve para unidade e módulo:
Tenta garantir que o código não possua erros.
Faça os testes, corrija ou implemente cada funcionalidade



Test Driven Development (TDD)

Desenvolvimento guiado por testes. Serve para unidade e módulo:
Tenta garantir que o código não possua erros.
Faça os testes, corrija ou implemente cada funcionalidade

Behaviour Driven Design (BDD)

Projeto guiado por comportamento. Serve para integração, sistema e aceitação:
Tenta garantir que o sistema segue o comportamento esperado.
Especifique o comportamento e construa o sistema de acordo.



- Clareza via concisão (abstração e alto nível)
- Síntese (geração automática de código a partir de uma descrição sintética)
- Reúso (padrões de projetos, orientação a objetos, *DRY*)
- Ferramentas de automação (trabalho chato é mais passível de erros, além de ser chato)



Michael Swaine (Dr. Dobbs, *Fire in the Valley*, InfoWorld) entrevistando Kent Beck (um dos criadores da programação extrema):

Mas se a Programação Extrema é apenas uma nova seleção de práticas antigas, o que existe de tão extremo nisso? A resposta de Kent é que ela leva princípios óbvios do senso comum para níveis extremos. Por exemplo:

- Se iterações curtas são boas, as faça o mais curtas possível — horas, minutos ou segundos ao invés de dias, semanas ou anos.
- Se testar é bom, teste o tempo todo. Escreva o código de teste antes de escrever o código a ser testado.
- Se revisões de código são boas, revise o código com frequência, através da programação em pares, dois programadores para um computador, alternando durante o processo de desenvolvimento.



Princípios do manifesto ágil

- Garantir a satisfação do cliente, entregando rápida e continuamente *software* funcional.
- Até mesmo mudanças tardias de escopo no projeto são bem-vindas
- Software funcional é entregue frequentemente.
- Cooperação constante entre clientes e desenvolvedores.
- Projetos surgem por meio de indivíduos motivados, devendo existir uma relação de confiança.
- A melhor forma de transmissão de informação entre desenvolvedores é pessoalmente.



IME-USP

- *Software* funcional é a principal medida de progresso do projeto;
- Novos recursos de *software* devem ser entregues constantemente.
- Clientes e desenvolvedores devem manter um ritmo até a conclusão do projeto.
- Projeto do *software* deve prezar pela excelência técnica.
- Simplicidade.
- As melhores arquiteturas, requisitos e designs emergem de equipes auto-organizáveis.
- Em intervalos regulares, a equipe reflete sobre como se tornar mais eficaz e então refina e ajusta seu comportamento.

