DOJO de Desenvolvimento orientado a testes - TDD

Cássio Trindade

Arquiteto de Software

AGÊNCIA EXPERIMENTAL DE ENGENHARIA DE SOFTWARE

DIRETÓRIO ACADÊMICO DA INFORMÁTICA
FACIN
SI CC ES
PUCRS

TDD – Test Driven Development

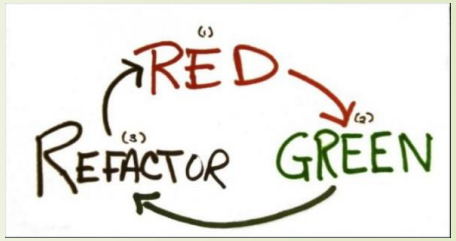
Esta técnica delega a importância dos testes a um nível maior, colocando estes como elemento base do código. A escrita da codificação do sistemas se inicia com os teste para o desenvolvimento de uma funcionalidade.

A ideia principal é escrever um teste, e quando é rodado este deverá falhar.

Quando o teste falhar deve-se verificar a implementação da funcionalidade e modificá-la para que o teste seja valido.

Uma vez válido deve ser escrito um novo teste para melhorar a implementação.

Este é o ciclo a ser percorrido até que todos os requisitos de terminado cenário de teste sejam atendidos.



AGÊNCIA EXPERIMENTAL DE ENGENHARIA DE SOFTWARE

DIRETÓRIO ACADÊMICO DA INFORMÁTICA
FACIN
SI
CC
ES
PUCRS

Pequenos Ciclos de Repetições

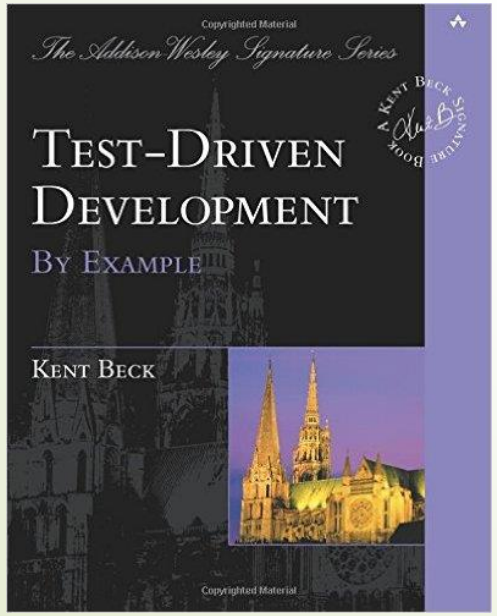
1. Escrevemos um Teste que inicialmente não passa (**Red**)
2. Adicionamos uma nova funcionalidade do sistema e Fazemos o Teste passar (**Green**)
3. Refatoramos o código da nova funcionalidade (**Refactor**)
4. Escrevemos o próximo Teste

AGES
AGÊNCIA EXPERIMENTAL DE ENGENHARIA DE SOFTWARE

DIRETÓRIO ACADÊMICO DA INFORMÁTICA
FACIN
SI CC ES
PUCRS

Por que usar?

- Códigos menores e limpos (Clean);
- Maior produtividade = Menos Bugs;
- Refactoring constante;
- Feedback rápido no impacto de novas funcionalidades;
- Baixo Acoplamento, responsabilidades bem definidas;



AGÊNCIA EXPERIMENTAL DE ENGENHARIA DE SOFTWARE

DIRETÓRIO ACADÊMICO DA INFORMÁTICA
FACIN
SI CC ES
PUCRS

DICA “MOTHERFUCKER”

Escrever um código simples e claro, que seja fácil de modificar e seguro.

AGÊNCIA EXPERIMENTAL DE ENGENHARIA DE SOFTWARE

DIRETÓRIO ACADÊMICO DA INFORMÁTICA
FACIN
SI CC ES
PUCRS

DOJO (do Zen Budismo, "*lugar de iluminação*")

- É estabelecido um problema a ser solucionado durante o dojo;
- Define-se um time de participantes de 10 a 30 componentes;
- Uma dupla de programadores, por vez do time, resolve uma parte do problema programando em par, utilizando o desenvolvimento orientado a testes. Os componentes do par são chamados "piloto" e "copiloto";
- Cada par tem entre 5 e 7 minutos para solucionar uma parte do problema.
- Ao final do período, o par interrompe seu trabalho onde estava e o copiloto assume o teclado. O piloto volta ao time de participantes, e alguém do time assume o posto de copiloto, para mais um turno de programação.

DOJO

Vamos seguir as seguintes regras:

1. O templo por par será de 6 minutos;
2. O par decide o que vai ser feito para solucionar o problema; o próximo par continua a implementação.
3. O time só pode dar sugestões para modificar ou melhorar o código quando já tiver passado mais de 3 minutos, ou quando a dupla solicitar ajuda.

AGÊNCIA EXPERIMENTAL DE ENGENHARIA DE SOFTWARE

DIRETÓRIO ACADÊMICO DA INFORMÁTICA
FACIN
SI CC ES
PUCRS

TECNOLOGIA

- Java 1.8 + Junit
- Eclipse Neon

AGES
AGÊNCIA EXPERIMENTAL DE ENGENHARIA DE SOFTWARE

DIRETÓRIO ACADÊMICO DA INFORMÁTICA
FACIN
SI CC ES
PUCRS

Problemas: **Encontre o telefone**

Em alguns lugares é comum lembrar um número do telefone associando seus dígitos a letras. Dessa maneira a expressão MY LOVE significa 69 5683. Claro que existem alguns problemas, uma vez que alguns números de telefone não formam uma palavra ou uma frase e os dígitos 1 e 0 não estão associados a nenhuma letra.

Sua tarefa é ler uma expressão e encontrar o número de telefone correspondente baseado na tabela abaixo. Uma expressão é composta por letras maiúsculas (A-Z), hifens (-) e os números 1 e 0.

AGÊNCIA EXPERIMENTAL DE ENGENHARIA DE SOFTWARE

DIRETÓRIO ACADÊMICO DA INFORMÁTICA
FACIN
SI CC ES
PUCRS

Problemas: **Encontre o telefone**

Entrada
A entrada consiste de um conjunto de expressões. Cada expressão está sozinha em uma linha e possui C caracteres, onde $1 \leq C \leq 30$.

Saída
Para cada expressão você deve imprimir o número de telefone correspondente.

Exemplo de entrada:
1-HOME-SWEET-HOME
MY-MISERABLE-JOB

Saída correspondente:
1-4663-79338-4663
69-647372253-562

- Letras -> Número
- ABC -> 2
- DEF -> 3
- GHI -> 4
- JKL -> 5
- MNO -> 6
- PQRS -> 7
- TUV -> 8
- WXYZ -> 9

AGES
AGÊNCIA EXPERIMENTAL DE ENGENHARIA DE SOFTWARE

DIRETÓRIO ACADÊMICO DA INFORMÁTICA
FACIN
SI CC ES
PUCRS

Problemas: **Encontre o numero p/ Romano**

O sistema de numeração romana (ou números romanos) desenvolveu-se na Roma Antiga e utilizou-se em todo o seu Império. Neste sistema as cifras escrevem-se com determinadas letras, que representam os números. As letras são sempre maiúsculas, já que no alfabeto romano não existem as minúsculas, as letras são I, V, X, L, C, D e M.

Sua tarefa é desenvolver um programa que converta números indo-arábicos para o formato romano e vice-versa. As regras para a formação dos números romanos são apresentadas a seguir.

- I = 1
- V = 5
- X = 10
- L = 50
- C = 100
- D = 500
- M = 1000

AGES
AGÊNCIA EXPERIMENTAL DE ENGENHARIA DE SOFTWARE

DIRETÓRIO ACADÊMICO DA INFORMÁTICA
FACIN
SI CC ES
PUCRS

Problemas: **Encontre o numero p/ Romano**

Com exceção de V, L e D, os outros numerais podem se repetir no máximo três vezes:

III = 3
XXX = 30
CCC = 300
MMM = 3000

Quando escritos à direita de numerais maiores, I, X e C somam-se aos valores dos primeiros:

IV = 5 - 1 = 4
IX = 10 - 1 = 9
XC = 100 - 10 = 90

Mas se os numerais I, X e C estiverem à esquerda dos maiores, seus valores são subtraídos como, por exemplo, em:

VIII = 5 + 1 + 1 + 1 = 8
LXII = 50 + 10 + 1 + 1 = 62
CLVIII = 158
MCXX = 1000 + 100 + 10 + 10 = 1120

AGES
AGÊNCIA EXPERIMENTAL DE ENGENHARIA DE SOFTWARE

DIRETÓRIO ACADÊMICO DA INFORMÁTICA
FACIN
SI CC ES
PUCRS