



# Introdução ao TDD



**Certified  
Developer**  
The Ultimate Tech Degree

**DigitalHouse** >  
Coding School



## O que é TDD (Test Driven Development)?

TDD é uma técnica de desenvolvimento de software baseado em um processo iterativo, incremental e com feedback imediato.

Foi desenvolvido por Kent Beck em seu próprio livro "TDD By Example" (2002).

TODO O CÓDIGO É  
CULPADO  
ATÉ PROVAR SUA  
INOCÊNCIA

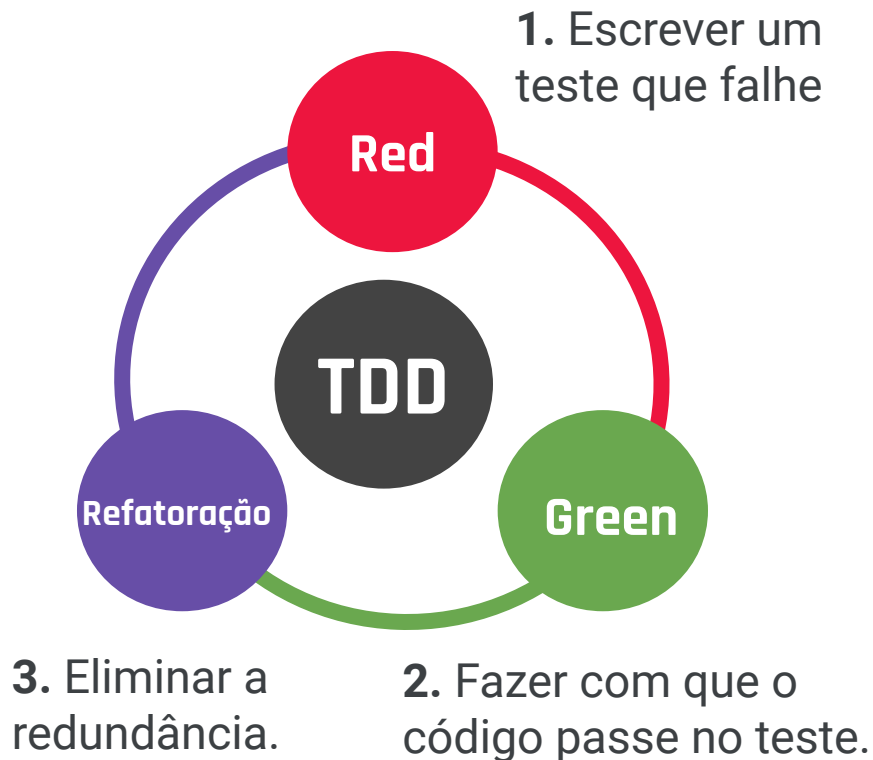


## OS 3 PASSOS DO TDD

**Etapá vermelha:** escreva um teste simples que falhe ao ser executado.

**Etapá verde:** programe o código necessário para a aprovação do teste.

**Etapá de refatoração:** nesta etapa, o objetivo é remover a duplicação criada na etapa 2.





## Vantagens e desvantagens quando aplicamos TDD

- Remove o medo de quebrar o código (teste frequente).
- Oferece desenvolvimento apenas do código necessário e simples.

- Curva de aprendizado considerável.
- Falsa crença na segurança (podemos esquecer os cenários para testar).



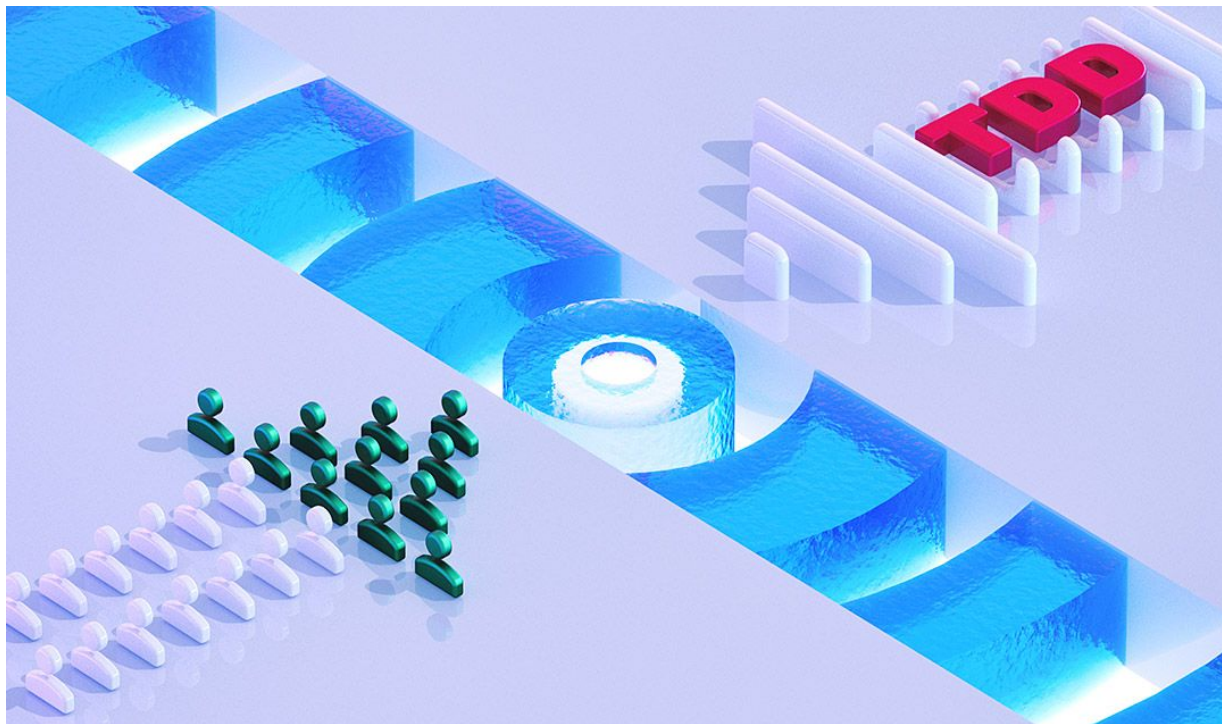
## Quando desenvolvemos TDD, precisamos ter uma equipe de controle de qualidade?

Fazer TDD não substitui o QA, pois há certas partes do sistema que não podem ser testadas fazendo TDD - porque existem certos testes que devem ser automatizados por ferramentas de alto nível funcional e é sempre necessário que pelo menos um ser humano use o sistema.

A utilidade de fazer TDD é que o QA receberá um sistema com menos erros, o que permitirá que você se concentre em fazer testes funcionais de alto nível em vez de testar tópicos simples.



# TDD vs. TESTERS





## 2 boas razões para adicionar testadores ao processo TDD

- Os testadores dedicados são mais propensos do que os programadores a dedicar um tempo para fazer testes exploratórios (ad-hoc). Eles podem encontrar bugs na funcionalidade ou nos serviços que o software deve fornecer - o que os programadores não previram durante o desenvolvimento do código.
- Os testadores também dão uma nova olhada no produto, em comparação com os programadores que estão imersos no software por muitas horas.



# BIBLIOGRAFIA

- Rasmusson, J. (2010). Chapter 14. En *The agile samurai: How agile masters deliver great software*. Pragmatic Bookshelf.
- TDD - Test Driven Development - <https://medium.com/tableless/tdd-test-driven-development-71ad9a69d465>



DigitalHouse>  
Coding School