



Quem se prepara, não para.

# Arquitetura de Aplicações Web

5º período

Professora: Michelle Hanne

# Sumário

- Tipos de Testes
  - Teste unitário
  - Teste de desempenho
  - Teste de carga
- Revisão de conteúdo para a AV1

## Teste unitário:

- normalmente é realizado pelos próprios desenvolvedores, nas porções menores do software que estão sendo desenvolvidas no momento, de maneira individualizada. Ele serve para verificar se as partes do software funcionam de maneira isolada das demais partes do sistema.

## Teste de caixa branca:

- avalia a parte interna do software, seu código-fonte. Ele serve para identificar problemas na lógica de programação e também na estrutura do programa, observando elementos como as condições usadas, os laços de repetição e o fluxo tomado pelos dados.

## Teste de caixa preta:

- avalia a parte externa do software, o seu modo de funcionamento. Esse tipo de teste serve para identificar se o software está funcionando como deveria, se os dados informados resultam nas informações pretendidas e se, de maneira geral, o sistema faz o que ele deveria fazer.

## Teste de caixa cinza:

- é uma combinação dos testes de caixa branca e de caixa preta, pois ele avalia os aspectos internos e também os aspectos externos do software, as suas entradas, o fluxo dos dados e as saídas.

## Teste de regressão:

- tem a função de testar cada nova versão do software toda vez que uma funcionalidade for modificada. Ou seja, toda a parte pronta do software será testada novamente para verificar se alguma novidade resultou em problema. Esse tipo de teste é muito importante para indicar se erros que já haviam sido corrigidos não voltaram a se manifestar isolada podem apresentar problemas quando tentam, por exemplo, realizar o fluxo de dados de uma parte do sistema para outra.

## Teste de volume:

- esse teste tem como objetivo avaliar até que limites um software pode ser utilizado, ou seja, qual é o seu limite de suporte a informações ou tráfego sem que apresente nenhum problema.

## Teste de performance:

- divide-se em outros três tipos, listados a seguir: Teste de carga, Teste de estresse e Teste de estabilidade.

## Teste funcional ou de funcionalidade:

- verifica se o software como um todo, bem como cada parte dele, faz exatamente o que deveria fazer, ou seja, se os casos de uso foram corretamente descritos e desenvolvidos.

## Teste de segurança:

- verifica se o software permite que os dados sejam acessados somente pelos perfis determinados para cada parte específica do sistema ou para cada funcionalidade.

## Teste de usabilidade:

- é realizado por usuários e não por analistas. O seu propósito é verificar se o software satisfaz as necessidades do usuário.

## Teste de aceitação:

- serve para verificar se o produto de software está pronto para ser entregue ao cliente, ou seja, se ele está pronto para entrar em produção. Geralmente, esse teste é realizado por alguém indicado pelo cliente, ou ainda por um testador especializado que não tenha participado do projeto, mas que tenha grande conhecimento acerca dos requisitos do software.



## Mas o que são testes unitários?

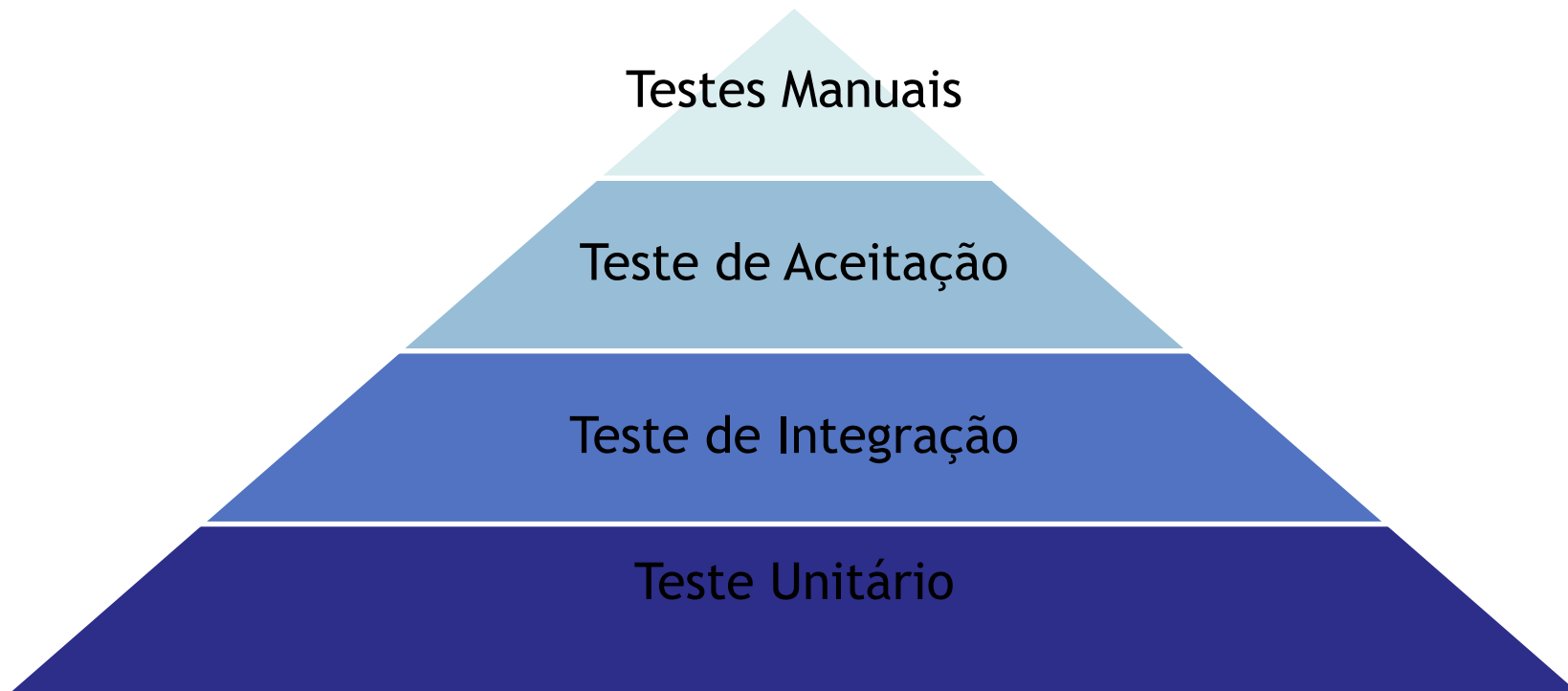
- **Um teste unitário basicamente é o teste da menor parte testável de um programa.**
- Se você programa em uma linguagem que suporte paradigma funcional por exemplo, a menor parte testável do seu código deve ser uma função. Então um teste unitário seria o teste de qualquer função. No caso de orientação a objetos seria o teste de um método de seu objeto

# Testes Unitários

```
function soma(a, b){  
  return a + b;  
}
```

- Para testar este código, só o que precisamos fazer é executar a função e verificar se seu valor de saída é o que esperamos.
- `var resultado = soma(1, 2);`  
`expect(resultado).toEqual(3);`

# Pirâmides de Teste



- Testes unitários não são os únicos tipos de testes que trazem benefícios a um projeto. Também podemos utilizar outros tipos de testes automatizados como **testes de integração** e **testes de aceitação**.
- Os Testes Manuais continuam tendo um papel importante no processo de controle de qualidade. De preferência, os testes manuais devem se concentrar nas áreas que não podem ser automatizadas, como testes de usabilidade.

- Tipos de testes contidos nas técnicas de Teste Estrutural e Funcional.

Teste Funcional (Black Box)	Teste Estrutural (White Box)
Regressão	Desempenho
Requisitos	Carga
Controle	Stress
Usabilidade	Conformidade
Aceitação	Contingência
Integração	Segurança

- Teste de Desempenho:** A aplicação suporta 1.000 transações por minuto com 1.000 usuários simultâneos?
- Teste de Carga:** Quantas transações serão suportadas por minuto quando aumentarmos os usuários simultâneos para 2.000, 3.000, 4.000?
- Teste de Stress:** Quantas transações por minuto solicitadas por 5.000, 6.000, 7.000 usuários simultâneos, serão suportadas pela aplicação sob condições não especificadas do software e até mesmo do próprio hardware?

## Testes unitários em Node usando a abordagem TDD

- <https://www.luiztools.com.br/post/tdd-como-criar-unit-tests-em-node-js-com-tape/>

## Teste de integração

- <https://medium.com/desenvolvimento-com-node-js/testes-de-integra%C3%A7%C3%A3o-para-aplica%C3%A7%C3%B5es-node-js-com-mocha-e-chai-610a1ba15e1b>

## Teste de Carga

- <https://www.npmjs.com/package/autocannon>

# Revisão de conteúdo para a AV1

Três questões relacionadas às unidades 1, 2 e 3. Uma questão relacionada à unidade 4.

## **Recomendações:**

- Ler as unidades 1, 2, 3 e 4, com os principais focos:
- Conceito de Microserviço
- Conceito de Python
- Conceitos de GitHub e GitLab
- Frameworks Rest
- Conceitos de NoSQL e MongoDB
- Conceito de Testes Unitários

- GONÇALVES, Fátima., P. D., BARRETO, Santos., J. D., ZENKER, Maciel., A., FAGUNDES, Rubem. *Testes de software e gerência de configuração*. [Minha Biblioteca]. Retirado de <https://integrada.minhabiblioteca.com.br/#/books/9788595029361/>
- PRESSMAN, R. S. Engenharia de software: uma abordagem profissional. 7. ed. Porto Alegre: AMGH, 2011.