

Engenharia De Software

AULA 6 E 7 - TESTES

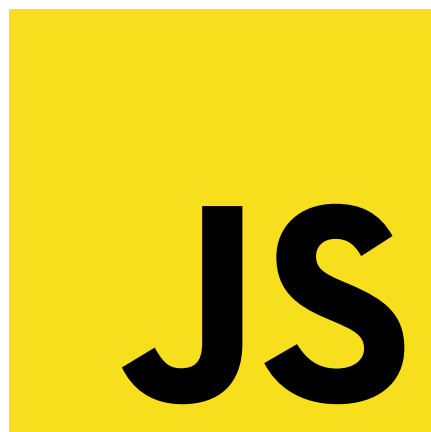
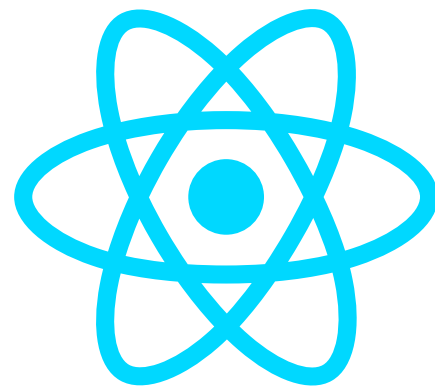
Inatel

Conteúdo da aula

- Conhecendo o Jest
- Testes Unitários
- Testes Mocks
- Como testar um site?
- Como testar um aplicativo?
- Outras linguagens

JEST

- Jest é um poderoso Framework de Testes em JavaScript com um foco na simplicidade.



JEST

Por que usar Teste Unitário ?

Lembre-se:

“O teste é o primeiro cliente do seu código e o ditado diz que o cliente sempre tem a razão.”

Inatel

JEST

Vamos implementar testes unitários para um exemplo de Calculadora em Node.js

- Crie uma aplicação node.js

```
E:\Desktop\Projetos\Monitoria>yarn init
yarn init v1.22.5
question name (Monitoria): calculadora-test
question version (1.0.0):
question description: Testes Unitários para Calculadora em Node.js
question entry point (index.js):
question repository url:
question author: Vanessa Swerts
question license (MIT):
question private: n
success Saved package.json
Done in 53.90s.
```

JEST

- Como instalar o JEST ?

```
yarn add --dev jest
```

```
npm install --save-dev jest
```

- Gerando um arquivo de configuração básico

```
jest --init
```

- Documentação do JEST

<https://jestjs.io/docs/pt-BR/getting-started>

JEST

- O arquivo **package.json**

```
{
  "name": "aula06-testes",
  "version": "1.0.0",
  "description": "Projeto para a Aula 06 do Lab de C214",
  "main": "src/index.js",
  "author": "Vanessa Swerts",
  "license": "MIT",
  ▶ Debug
  "scripts": {
    "test": "jest",
    "coverage": "jest --coverage"
  },
  "devDependencies": {
    "jest": "^26.6.3"
  }
}
```

JEST

- Configurações do JEST

```
Vanessa Swerts@Swerts96 MINGW64 /e/Desktop/Projetos/Monitoria/aula06-testes
$ yarn jest --init
yarn run v1.22.5
$ E:\Desktop\Projetos\Monitoria\aula06-testes\node_modules\.bin\jest --init

The following questions will help Jest to create a suitable configuration for your project

✓ Would you like to use Typescript for the configuration file? ... no
✓ Choose the test environment that will be used for testing » node
✓ Do you want Jest to add coverage reports? ... yes
✓ Which provider should be used to instrument code for coverage? » v8
✓ Automatically clear mock calls and instances between every test? ... yes

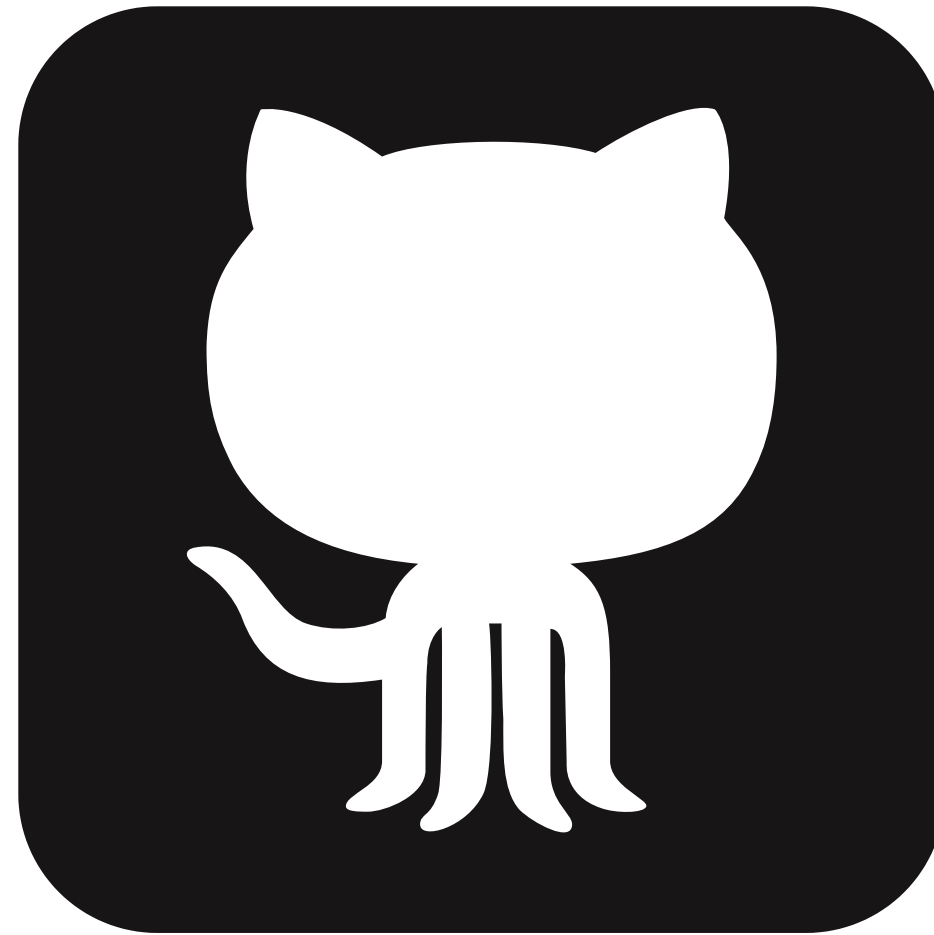
💎 Configuration file created at E:\Desktop\Projetos\Monitoria\aula06-testes\jest.config.js
Done in 48.84s.
```


JEST

- Arquivo **jest.config.js**

```
module.exports = {  
  bail: true,  
  clearMocks: true,  
  collectCoverage: true,  
  collectCoverageFrom: ["src/**"],  
  coverageDirectory: "__tests__/coverage",  
  coverageProvider: "v8",  
  testEnvironment: "node",  
  testMatch: [  
    "**/__tests__/**/*.test.js?(x)"  
  ]  
}
```

Código da Aula



<https://github.com/monitoria-C214/aula06-calculadora>

Alguns métodos ...

- **toBe** : Compara se os valores são iguais.
- **toBeNull** : Verifica se o valor esperado é null.
- **toBeTruthy**: Verifica se o valor esperado é verdadeiro.
- **toBeFalsy**: Verifica se o valor esperado é falso.
- **toBeGreaterThan** : Compara se o valor é maior.
- **toBeLessThanOrEqual**: Compara se o valor é menor ou igual.
- **toBeCloseTo** : Comparação que usa aproximação.
- **toMatch**: Usado para comparar Strings.

Documentação: <https://jestjs.io/docs/pt-BR/using-matchers>

Coverage

- Cobertura de Código
- Cria um relatório sobre o nosso código, mostrando o arquivos testados e não testados.

All files

61.54% Statements 24/39 80% Branches 4/5 57.14% Functions 4/7 61.54% Lines 24/39

Press *n* or *j* to go to the next uncovered block, *b*, *p* or *k* for the previous block.

File ▲		Statements ▾		Branches ▾		Functions ▾		Lines ▾	
Calculadora.js	<div><div></div></div>	70.59%	24/34	100%	4/4	66.67%	4/6	70.59%	24/34
index.js	<div><div></div></div>	0%	0/5	0%	0/1	0%	0/1	0%	0/5

Coverage

All files Calculadora.js

70.59% Statements 24/34 100% Branches 4/4 66.67% Functions 4/6 70.59% Lines 24/34

Press *n* or *j* to go to the next uncovered block, *b*, *p* or *k* for the previous block.

```
1 1x class Calculadora {
2 1x
3 1x   add(num1, num2) {
4 1x     return num1 + num2
5 1x   }
6 1x
7 1x   sub(num1, num2) {
8 1x     return num1 - num2
9 1x   }
10 1x
11 1x   mult(num1, num2) {
12 1x     return num1 * num2
13 1x   }
14 1x
15 1x   div(num1, num2) {
16   if (num2 === 0)
17     throw Error("Divisão por zero!")
18
19     return num1 / num2
20   }
21 1x
22 1x   pow(num1, num2) {
23 1x     return Math.pow(num1, num2)
24 1x   }
25 1x
26 1x   sqrt(num) {
27   if (num < 0) {
28     throw 'Indefinido'
29   }
30   return Math.sqrt(num)
31 }
32 1x }
```

Snapshot

- Testes de snapshot são ferramentas bem úteis sempre que você desejar garantir que sua UI não seja alterada inesperadamente.
- <https://jestjs.io/docs/pt-BR/snapshot-testing>

```
it('renders correctly', () => {  
  const tree = renderer  
    .create(<Link page="http://www.facebook.com">Facebook</Link>)  
    .toJSON();  
  expect(tree).toMatchSnapshot();  
});
```

Extensão do Jest no VSCode



Jest orta.vscode-jest

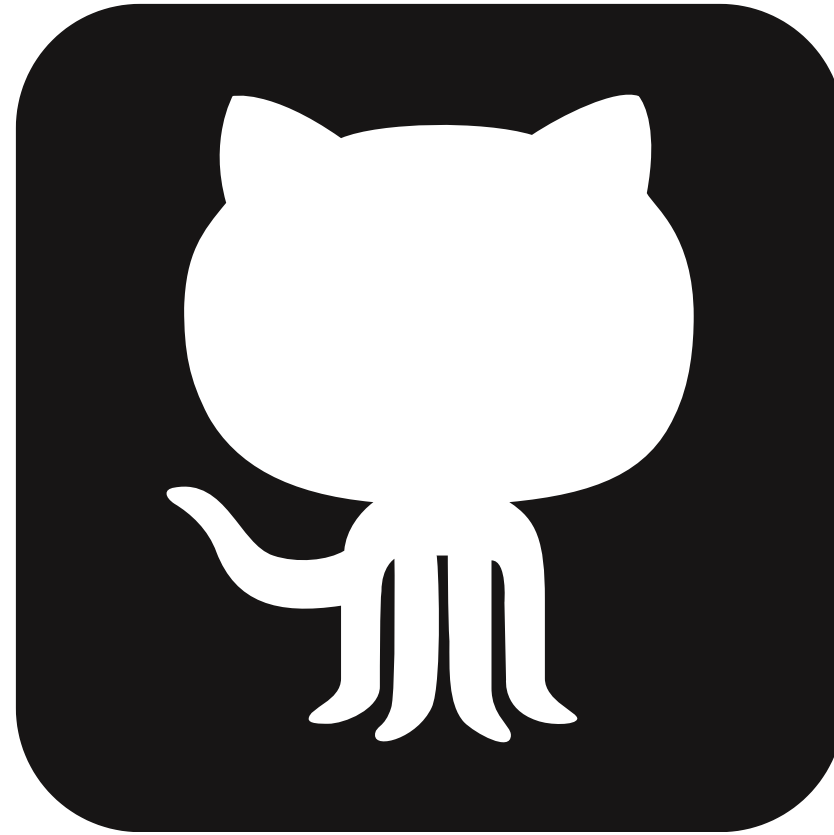
Orta | 562.616 | ★★★★★ | Repository | License | v3.2.0

Use Facebook's Jest With Pleasure.

Disable ▼ Uninstall ▼ ⚙️ This extension is enabled globally.

```
✓ test("Soma de 5 + 5", () => {  
  const resultado = calc.add(5, 5)  
  
  expect(resultado).toBe(10)  
})  
  
✓ it("Subtração de 10 - 5", () => {  
  const resultado = calc.sub(10, 5)  
  
  expect(resultado).toEqual(5)  
})
```

Como testar um site?



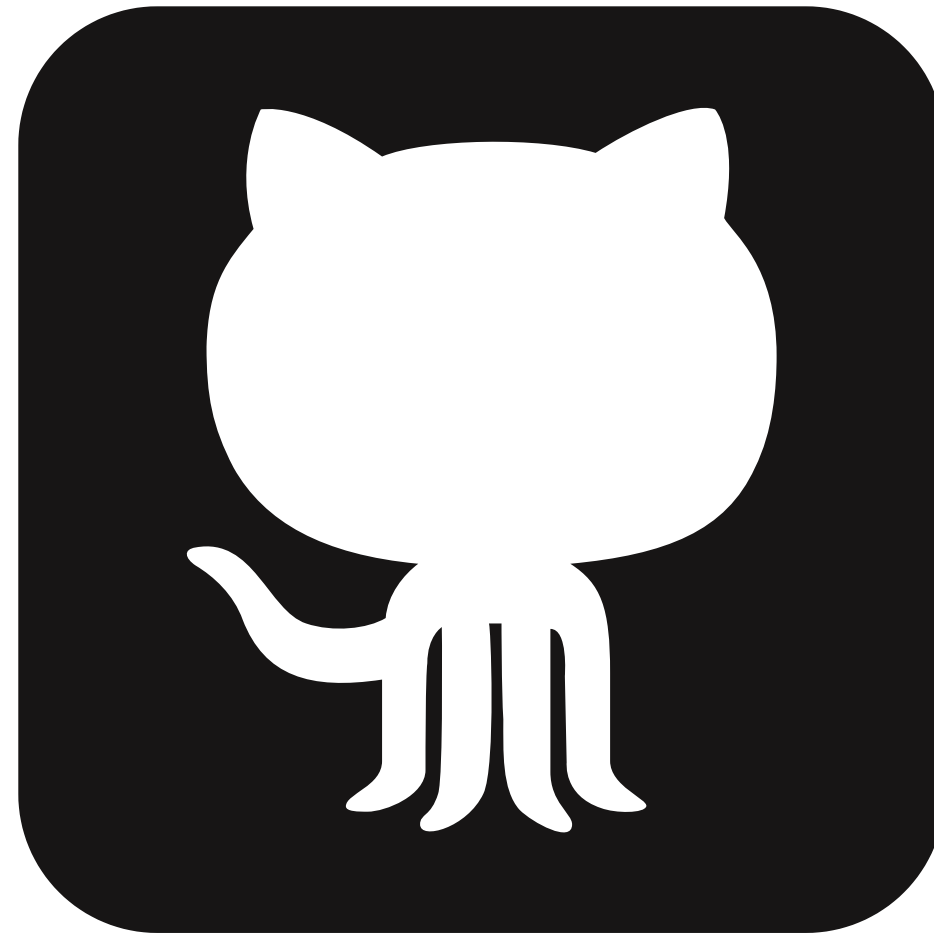
- **Template do Projeto (sem testes)**

<https://github.com/monitoria-C214/github-search-template>

- **Código da Aula (com testes prontos)**

<https://github.com/monitoria-C214/github-search>

Como testar um aplicativo?



<https://github.com/alexanderaugusto/imovel-app>

Outras Linguagens

- **JavaScript:** Qunit
- **Python:** Unittest e Pytest
- **C#:** NUnit , Unit e XUnit
- **PHP:** PHPUnit
- **Flutter:** <https://pub.dev/packages/test>