

**SIDADE CATÓLICA DE MINAS GERAIS**

**ENGENHARIA DE SOFTWARE - PRAÇA DA LIBERDADE**

**TESTE DE SOFTWARE**

**Alunos:**  Lucas Lima

**Testes em Node.Js usando Tape**

**A linguagem de programação Node.Js**

Inicialmente podemos falar sobre especificamente o que é, de onde vem e para o que é normalmente utilizado. O Node.Js é um ambiente de execução que trabalha em javascript, tal linguagem que é padrão para a manipulação de páginas HTML e teve seu surgimento em 1995. Inicialmente ela foi utilizada para o desenvolvimento de client-side, mas conforme ocorreu a evolução e propagação da internet ela começou a ser aplicada no ambiente de server-side.

O Node.Js possui colaboradores em seu repositório principal nodejs/node no GitHub o qual é mantido por colaboradores que são nomeados continuamente pelo Technical Steering Committe(TSC).

Com o Node.Js é possível se criar aplicações completas, sites APIs e microsserviços, isso é possível graças à união ao ambiente de execução de JavaScript fornecido pelo próprio Node.Js e o motor de interpretação e execução de JavaScript presente no Google Chrome, chamado de V8.

**O framework de testes Tape(github -** [**https://github.com/substack/tape**](https://github.com/substack/tape))

O Tape é um framework utilizado para testes, ele é comparado a outros frameworks como por exemplo, Mocha, Jasmine ou NodeUnit. O Tape pode oferecer uma estrutura de testes unitários muito simples e de fácil entendimento.

O framework possui uma configuração muito básica, necessitando da instalação utilizando o gerenciador de pacotes do NPM, e adicionando pequenas dependências no código que será testado. Um defeito a ser falado devido a sua estrutura simples é que se um controle maior por parte de testes seja necessária a utilização de frameworks mais complexos e completos é indicada, como por exemplo o Jest.

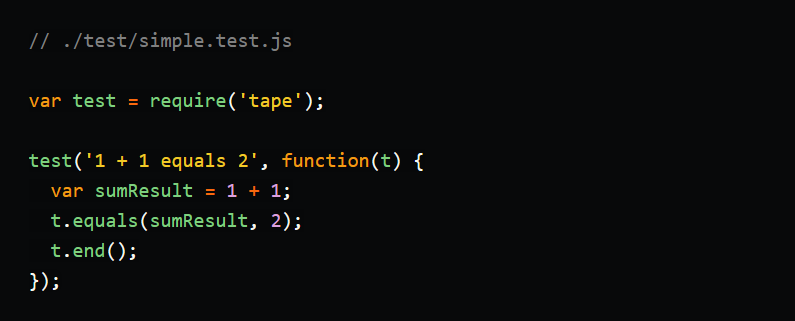
Pontos como muita configuração, poluição do ambiente global e compartilhamento de estado(ex. beforeEach ou afterEach), são onde o Tape se encontra mais acessível que seus competidores(ex. Mocha, Jasmine), tendo um módulo de exportação simples, instalação rápida e intuitiva se auxiliada da documentação e chamas de configuração e rotinas de modo individual onde contém todas as variáveis de teste de estado para local.

**Test Anything Protocol**

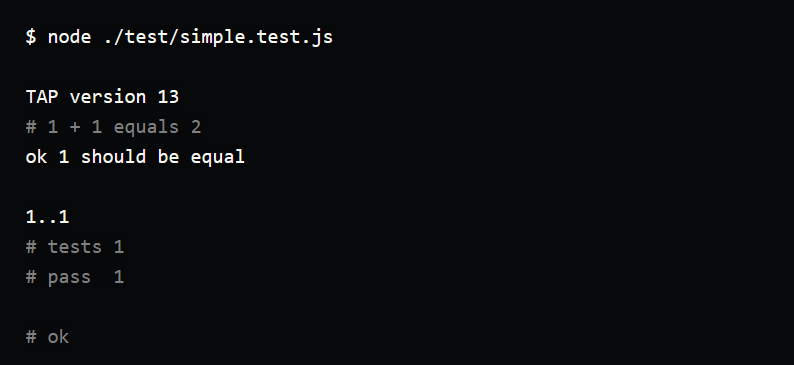
TAP, ou Test Anything Procotol é a definição de ‘tried & true’ que já existe desde 1987 e já foi implementada várias vezes em diversas linguagens diferentes.

Esse formato simples significa que é possível combinar ferramentas comuns. Onde as ferramentas não se conectam diretamente com o Tape. Sendo a única coisa que importa ser o protocolo TAP, para assim poder canalizar os resultados. É possível canalizar um produto TAP em uma outra linguagem para o protocolo e ele funciona da mesma forma.

Aqui podemos ver uma configuração simples e exemplo de teste utilizando o Tape:



Aqui podemos ver um exemplo daquilo que será apresentado na tela após os testes do projeto serem executados:



O Tape pode ser utilizado para testes unitários ou até mesmo teste de integração, podendo assim produzir testes para nós ou browsers. Possui suporte para saídas de xUnit. É caracterizado por ser client-side e server-side fazendo assim o teste de qualquer parte do front-end ou back-end. Pode ser adicionada a biblioteca redTape que estende o Tape e com isso disponibilizando funcionalidades como beforeEach e afterEach e assim se ter uma ferramenta completa.

**REFERÊNCIAS BIBLIOGRÁFICAS**

# MACWRIGHT, Tom: TAP & Tape, the awesome way to test JavaScript. Macwright, 2014. Disponível em: <<https://macwright.com/2014/03/11/tape-is-cool.html>>. Acesso em: 16 de março de 2022.

* ELLIOTT, Eric: Why I use Tape Instead of Mocha & So Should You. Medium, 2015. Disponível em: <<https://medium.com/javascript-scene/why-i-use-tape-instead-of-mocha-so-should-you-6aa105d8eaf4>>. Acesso em: 16 de março de 2022.
* Tape. Testling. Disponível em: <<https://ci.testling.com/guide/tape>>. Acesso em: 16 de março de 2022.

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