

# Capítulo 6 - Configurando o Cypress-Axe para seus testes específicos

---

Neste capítulo, obteremos mais informações técnicas e configuraremos o Cypress e o Cypress-Axe para começar a automatizar alguns de nossos testes de acessibilidade.

Nós vamos:

- Criar um projeto de teste simples.
- Ver como podemos instalar o Cypress e o Cypress-Axe neste projeto
- Importar o Cypress-Axe para o nosso projeto para que possamos estender os comandos padrões do Cypress

## Pré requisitos

---

Como pré-requisito, para instalar o Cypress, você precisa ter o Node e o NPM instalados em sua máquina, bem como o Visual Studio Code.

Se você não tiver esses programas instalados, vá em frente e volte quando tudo estiver pronto.

Você pode consultar os recursos abaixo para encontrar os links sobre como instalar esses programas.

## Instalação e configuração

---

Em primeiro lugar, vamos fazer a instalação e configuração.

Como já tenho o Node instalado, não preciso fazer a parte da instalação.

Se você quiser descobrir se você instalou o Node com sucesso, basta digitar:

```
node -v
```

Será informado o número da versão que você está usando.

O que faremos agora é criar um novo projeto de teste, que chamarei de cypress-axe-project.

Uma vez que este diretório é criado, vamos prosseguir e ir para este diretório digitando cd, seguido do nome do projeto.

```
cd cypress-axe-project
```

Em seguida, vou digitar o comando

```
npm init -y
```

Este comando irá simplesmente inicializar nosso projeto e criar o pacote que usaremos para o arquivo.

Isso gerenciará todas as dependências do nosso projeto.

Se desejar, você pode pensar neste arquivo de forma semelhante a um arquivo Maven, se você estiver vindo de um ambiente Java.

```
→ Documents node -v
v12.16.3
→ Documents mkdir cypress-axe-project
→ Documents cd cypress-axe-project
→ cypress-axe-project npm init -y
Wrote to /Users/mariedrake/Documents/cypress-axe-project/package.json:

{
  "name": "cypress-axe-project",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC"
}
```

Excelente. Agora que estamos prontos para instalar o Cypress, tudo o que precisamos fazer agora é digitar os comandos:

```
npm i -D cypress@4.8.0 cypress-axe@0.8.1
```

**i** é um atalho para instalar, **-D** porque queremos salvá-lo em nossas dependências de desenvolvimento neste projeto específico.

Em seguida, o nome do pacote, cypress, seguido por uma versão específica.

E, claro, também adicionaremos o pacote cypress-axe.

Eu escolhi codificar os números de versão para esses dois pacotes, então eu sei que este curso sempre funcionará no recurso no caso de novas versões apresentarem problemas de compatibilidade.

Agora, vamos apenas esperar que o processo de instalação seja concluído, e isso pode levar mais tempo para alguns, especialmente se esta for a primeira vez que você instala o Cypress.

```
→ cypress-axe-project npm i -D cypress@4.8.0 cypress-axe@0.8.1

> cypress@4.8.0 postinstall /Users/mariedrake/Documents/cypress-axe-project/node_modules/cypress
> node index.js --exec install

Cypress 4.8.0 is installed in /Users/mariedrake/Library/Caches/Cypress/4.8.0

npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN cypress-axe-project@1.0.0 No description
npm WARN cypress-axe-project@1.0.0 No repository field.

+ cypress@4.8.0
+ cypress-axe@0.8.1
added 205 packages from 146 contributors and audited 205 packages in 22.492s

3 packages are looking for funding
  run `npm fund` for details

found 1 low severity vulnerability
  run `npm audit fix` to fix them, or `npm audit` for details
```

Agora que tudo está instalado com sucesso, vamos prosseguir e abrir o Visual Studio Code para carregar este projeto.

Nosso projeto foi carregado, mas onde está nossa pasta Cypress?

Se você não sabe, o Cypress vem embutido com uma estrutura de pastas predefinida para você começar imediatamente.

No entanto, precisamos executar mais um comando, então vamos voltar ao nosso terminal (ou uso o próprio terminal do Visual Studio Code).

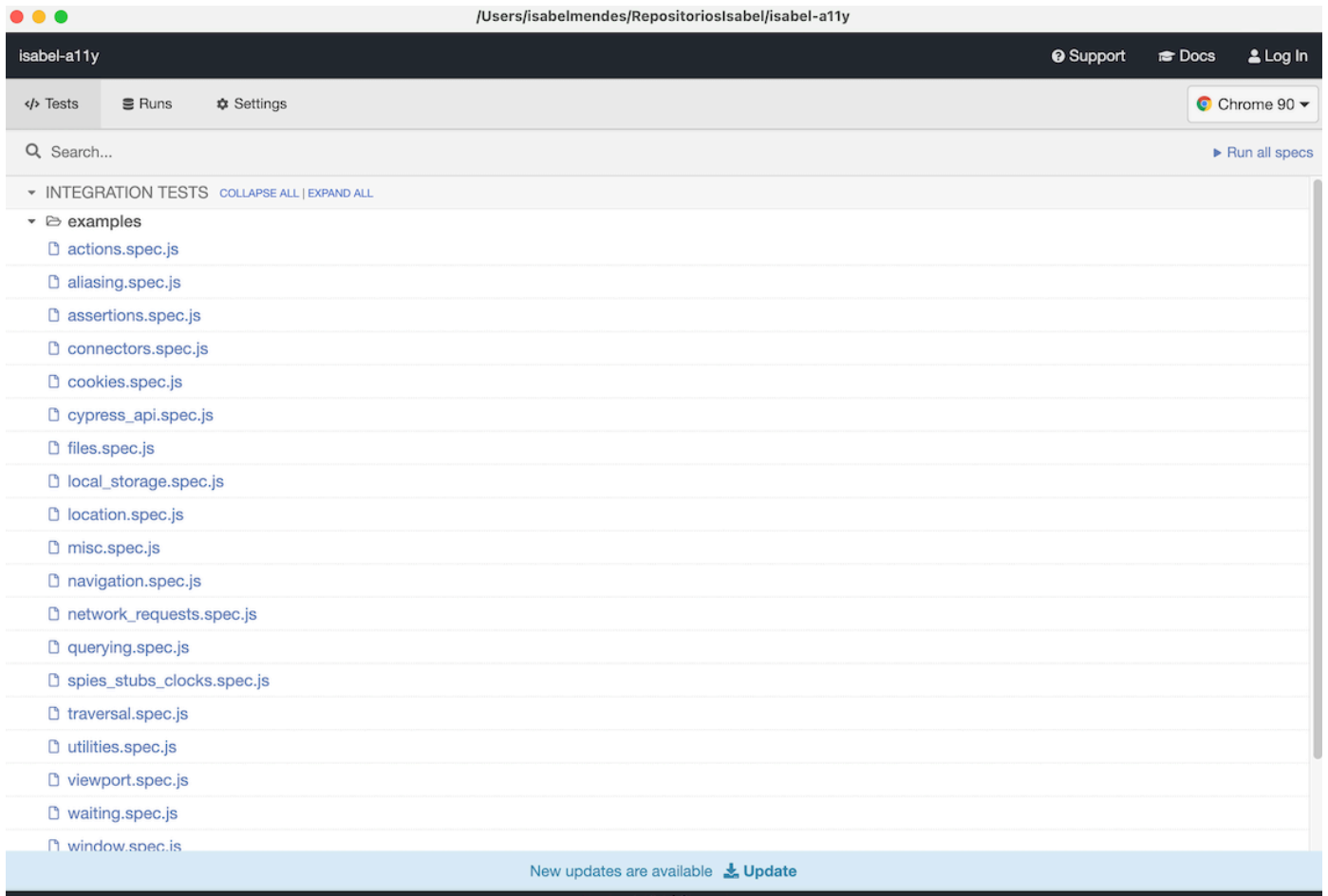
```
npx cypress open
```

Observe que estou usando NPX aqui em vez de NPM.

NPX é basicamente uma notação de atalho para executar um comando específico.

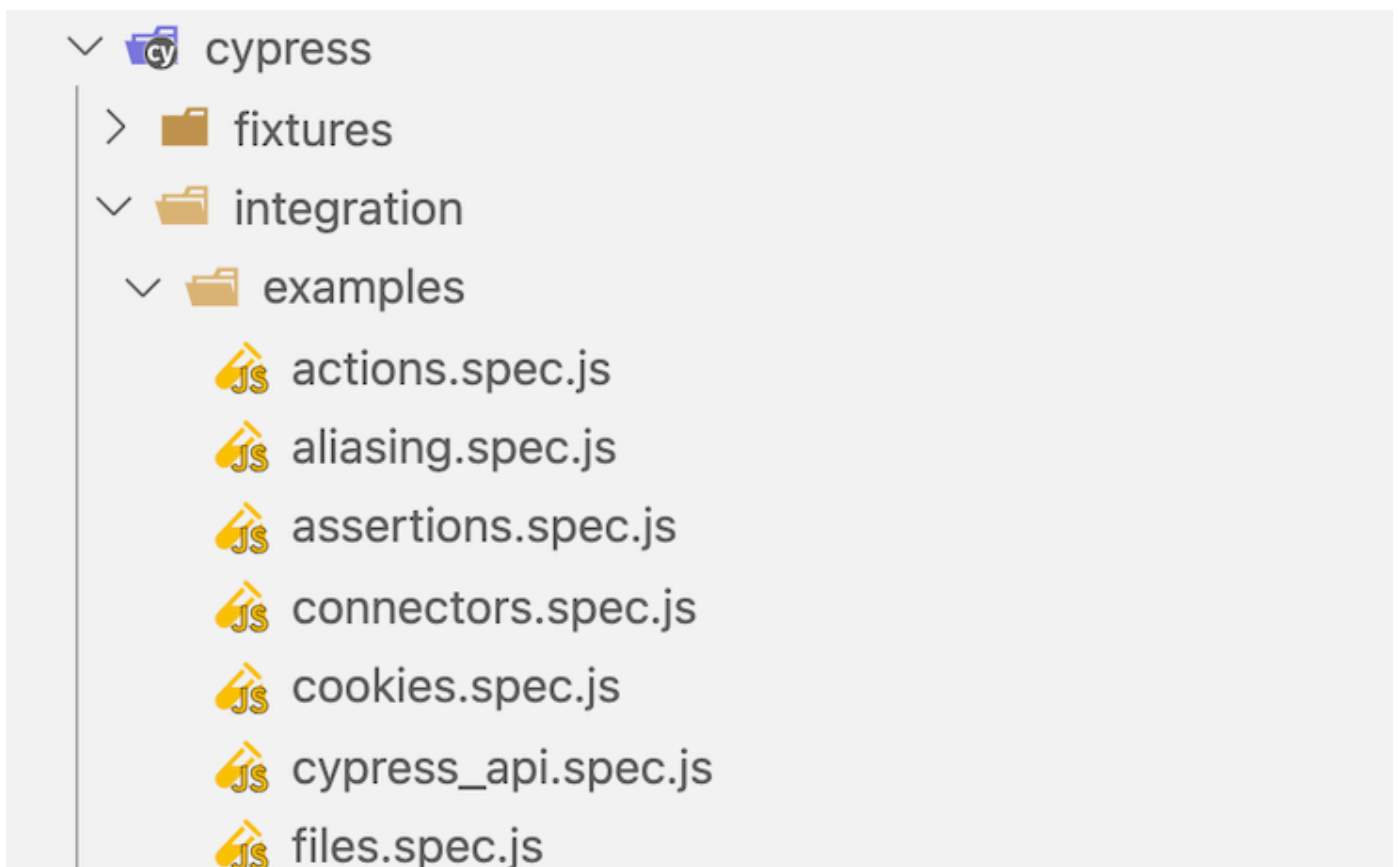
Neste caso, desejo executar o comando cypress open.

Agora, este comando fará duas coisas: primeiro, ele abrirá o executor de teste oficial do Cypress e, em seguida, como você pode ver, carregará o exemplo.



Esses arquivos vêm integrados com o Cypress.

Em segundo lugar, o comando criará a estrutura de pastas padrão para nós, uma vez que ainda não foi criada.



- 📄 local\_storage.spec.js
- 📄 location.spec.js
- 📄 misc.spec.js
- 📄 navigation.spec.js
- 📄 network\_requests.spec.js
- 📄 querying.spec.js
- 📄 spies\_stubs\_clocks.spec.js
- 📄 traversal.spec.js
- 📄 utilities.spec.js
- 📄 viewport.spec.js
- 📄 waiting.spec.js
- 📄 window.spec.js

> 📁 new

> 🧩 plugins

> 📁 support

> 📁 node\_modules

📄 cypress.json

📄 package-lock.json

📄 package.json

Para usar o Cypress-Axe neste projeto, tenho mais uma configuração que preciso mostrar a você.

Precisamos importar os comandos de que precisamos em nosso arquivo support / index.js.

Este comando basicamente nos permitirá estender o comportamento padrão do Cypress e usar os comandos adicionais que precisaremos do Cypress-Axe.

O arquivo deve ser parecido com o abaixo, apenas adicionando as informações no arquivo support/index.js:

JS index.js ●

```
cypress > support > JS index.js
```

```
1 import './commands';  
2  
3 import 'cypress-axe';
```

Então isso é basicamente tudo para nossa configuração.

No próximo capítulo, veremos como podemos escrever nosso primeiro teste de acessibilidade com Cypress-Axe.

## Links

---

- Instalação
  - [Node and NPM](#)
  - [Visual Studio Code](#)
- Github
  - [Git repo for source code](#)
  - [Source code for support/index.js file](#)
- Outros links importantes
  - [Gil Tayar's Introduction to Cypress Course](#)
  - [Cypress-axe Github page](#)
  - [Cypress Documentation site](#)