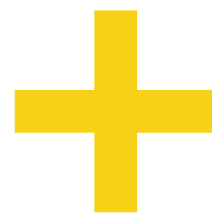


Testes automatizados com Appium no iOS



Appium

- Appium é uma estrutura de automação de teste de código aberto para uso com aplicativos da Web nativos, híbridos e móveis.
- Ele dirige aplicativos iOS, Android e Windows usando o protocolo WebDriver.

CodeceptJS

- O CodeceptJS é uma estrutura moderna de teste de ponta a ponta com uma sintaxe especial no estilo BDD.

Softwares necessários

- Node
- Appium
- CodeceptJS
- Allure - Ferramenta de relatório de teste multilíngue leve e flexível
- HomeBrew - Gerenciador de pacotes

Processo de instalação

- **HomeBrew**

- `> /usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"`

- **Node**

- > brew install node

- **Allure**

- > brew install allure

- **Appium**
 - > npm install -g appium
- **Appium Client**
 - > npm install wd

- **iOS-deploy**
 - O iOS-deploy é um pequeno utilitário para instalar e depurar aplicativos do iPhone a partir da linha de comando, sem usar o Xcode.
 - > brew install ios-deploy

- **iDeviceInstaller**

- O iDeviceInstaller é uma ferramenta para interagir com o installation_proxy de um dispositivo iOS que permite instalar, atualizar, desinstalar, arquivar, restaurar e enumerar aplicativos instalados ou arquivados. Também é necessário para executar testes em dispositivos reais.
- > brew install ideviceinstaller

- **iOS WebKit Debug Proxy**
- O Appium usa essa ferramenta para acessar WebViews em dispositivos reais
 - > brew install ios-webkit-debug-proxy

- **Appium Doctor**
- O Appium doctor é utilizado para verificar se todos os pré-requisitos do Appium estão instalados corretamente
 - > npm install -g appium-doctor
- **Execute o Appium Doctor para verificar se o ambiente está em conformidade**
 - > appium-doctor

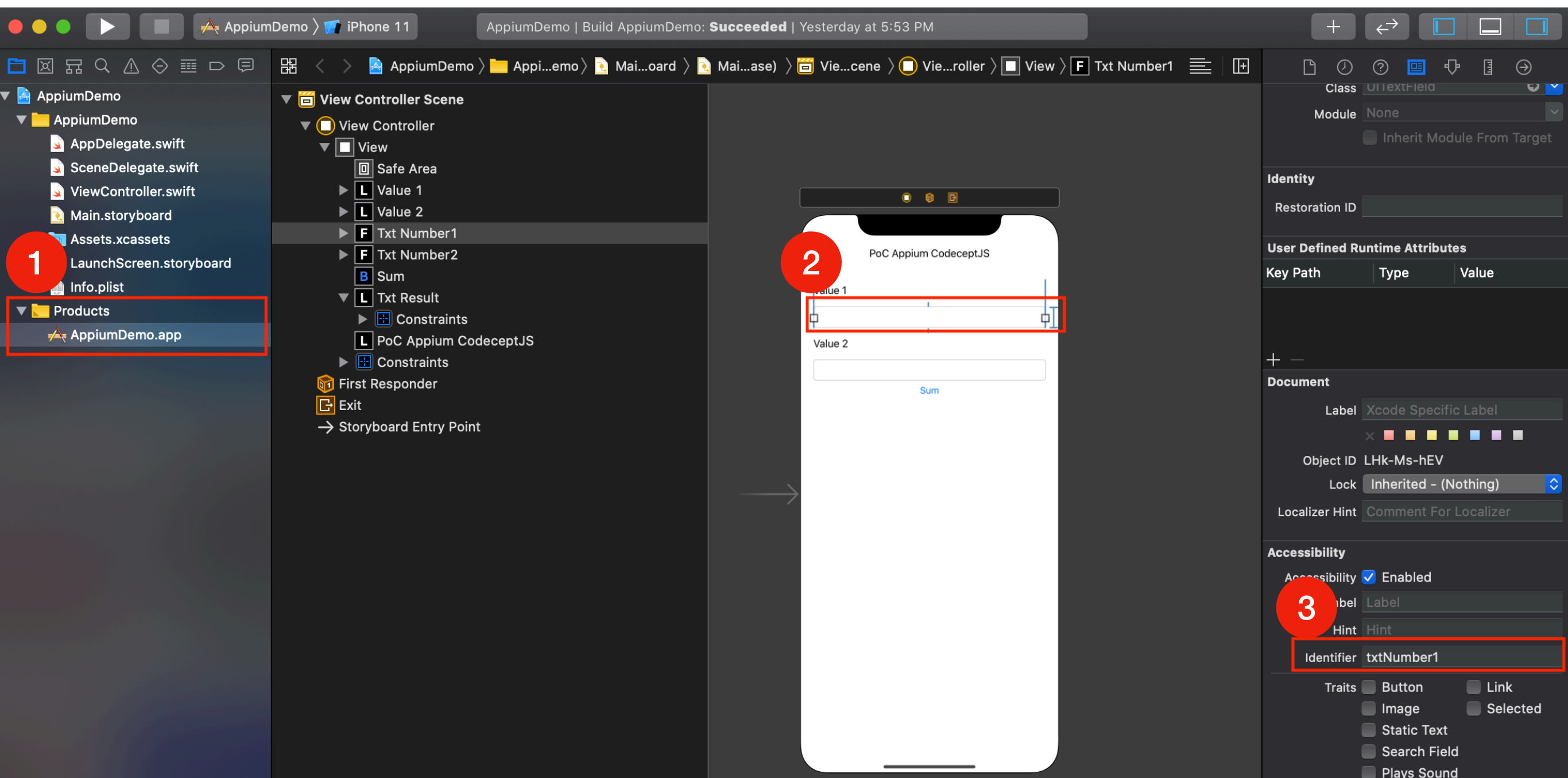
- **iOS WebKit Debug Proxy**
- O Appium usa essa ferramenta para acessar WebViews em dispositivos reais
 - `> brew install ios-webkit-debug-proxy`

- **CodeceptJS**

- `> npm install codeceptjs webdriverio --save`

Projeto AppiumDemo

- Projeto AppiumDemo



1. App gerado após o build do projeto
2. TextField em storyboard
3. Identificador do TextField selecionado

Inicialização de projeto

- **Criar o projeto CodecepJS**

- `> npm init -y`

- **Iniciar o projeto CodecepJS**

- `> npx codeceptjs init`

→ **ios** **git:(appium)** **x** npx codeceptjs init

npx: installed 197 in 6.405s

[Welcome to **CodeceptJS** initialization tool

It will prepare and configure a test environment for you

Installing to **/Users/rrovaron/raizen/ios**

? **Where are your tests located?** **./*_test.js**

? **What helpers do you want to use?** **Appium**

? **Where should logs, screenshots, and reports to be stored?** **./output**

[? **Would you like to extend the "I" object with custom steps?** **Yes**

? **Where would you like to place custom steps?** **./steps_file.js**

[? **Do you want to choose localization for tests?** **English (no localization)**

[Configure helpers...

[? **[Appium] Application package. Path to file or url** **./input**

? **[Appium] Mobile Platform** **iOS**

? **[Appium] Device to run tests on** **emulator**

[

Steps file created at **/Users/rrovaron/raizen/ios/steps_file.js**

[Config created at **/Users/rrovaron/raizen/ios/codecept.conf.js**

Directory for temporary output files created at **'./output'**

Intellisense enabled in **/Users/rrovaron/raizen/ios/jsconfig.json**

TypeScript Definitions provide autocompletion in Visual Studio Code and other IDEs

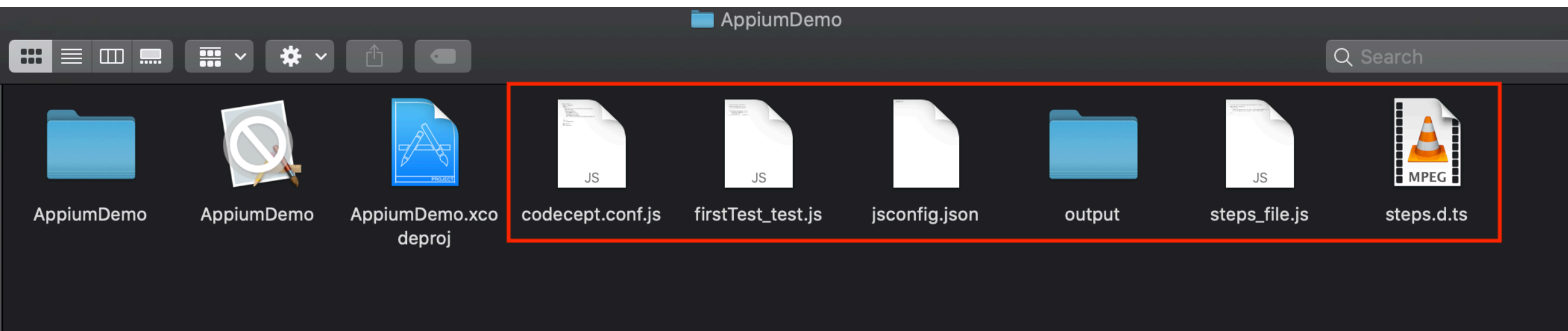
Definitions were generated in **steps.d.ts**

Almost done! Next step:

Create your first test by executing `npx codeceptjs gt` command

- **Criar o primeiro teste**
 - `> npx codecept gt`

- **Itens criados no projeto**



- Edição do codecept.conf.js

JS codecept.conf.js ●

JS codecept.conf.js > [🔗] config

```
1  exports.config = {  
2    tests: '.*_test.js',  
3    output: './output',  
4    helpers: {  
5      Appium: {  
6        1 app: '/Users/rrovaron/Desktop/AppiumDemo/AppiumDemo.app',  
7        platform: 'iOS',  
8        2 desiredCapabilities: {  
9          "platformName": "iOS",  
10         "platformVersion": "13.2",  
11         "deviceName": "iPhone 11",  
12         "automationName": "XCUITest",  
13         "app": "/Users/rrovaron/Desktop/AppiumDemo/AppiumDemo.app"  
14       }  
15     }  
16   },  
17   include: {  
18     I: './steps_file.js'  
19   },  
20   bootstrap: null,  
21   mocha: {},  
22   name: 'AppiumDemo'  
23 }
```


1. Deve ser adicionado o app gerado no projeto (Xcode), com seu caminho absoluto

2. Deve ser adicionado a **KEY** e suas respectivas **propriedades**, mencionando a **plataforma** (iOS), a **versão do iOS** que será testada no simulador/device, o **nome do simulador/device**, o **framework de testes** (XCUITest), e o **caminho absoluto** até o app que será testado

Criação do primeiro teste

- Edição do codecept.conf.js

JS firstTest_test.js ×

JS firstTest_test.js > ...

```
1      1
2      Feature('PoC Appium CodeceptJS');
3
4      2 Scenario('Test Open App', (I) => {
5          3 I.see('PoC Appium CodeceptJS');
6      });
7
8      Scenario('Test create a sum', (I) => {
9          4 I.fillField('~txtNumber1', '1');
10         I.fillField('~txtNumber2', '2');
11         5 I.click('~btnSum');
12         6 I.see('Result is 3', '~txtResult');
13     });
14
```

1. Nome da **feature** a ser testada
2. Nome do **scenerio** a ser testado
3. Comando para buscar determinado texto em algum lugar da tela
4. Comando para preencher texto em determinado campo (com o identificador)
5. Comando para clicar em determinado elemento na tela (com o identificador)
6. Comando para buscar determinado texto em determinado elemento na tela (com o identificador)

Execução do teste

- **Executar o Appium Client**
 - > appium
- **Executar rotina de testes**
 - > npx codeceptjs run

```
/Users/rrovaron/Desktop/AppiumDemo
```

```
[→ AppiumDemo npx codeceptjs run
```

```
CodeceptJS v2.3.6
```

```
Using test root "/Users/rrovaron/Desktop/AppiumDemo"
```

```
68 > npx codeceptjs init
```

```
featureTest --
```

```
69 ✓ test something in 3ms
```

```
70 create first test
```

```
71 OK | 1 passed // 5m
```