

Testing with CodeceptJS

E2E tests anyone can write

Ivan Herndon

About Me

- Senior Software Engineer at StockX
- Front End Instructor at Grand Circus
- Github: **iherndon**
- Twitter: **@ivanbytes**

Why This Talk?

- Need for automated e2e tests
- A lot of test frameworks not written in JS
- Need for easy to write / understand tests

What is End-to-End testing?

- Test application flow from start to end
- Simulate real user scenario
- Validate system under test
- Validate components for integration and data integrity

Don't we already have tools for this?

- Selenium
 - Not JavaScript
 - Difficulty with environment set up
 - Difficulty writing simple tests
- Protractor
 - JavaScript
 - Testing Api can be hard to follow
 - Difficulty writing tests

Selenium Example

```
public class FirstTestCase {  
    public static void main(String[] args) {  
        // Create a new instance of the Firefox driver  
        WebDriver driver = new FirefoxDriver();  
        //Launch Gmail  
        driver.get("https://www.gmail.com");  
        // Print a Log In message to the screen  
        System.out.println("Successfully opened the website www.gmail.com");  
        //Wait for 5 Sec  
        Thread.sleep(5000);  
        // Close the driver  
        driver.quit();  
    }  
}
```

Protractor Example

```
// spec.js

describe('Protractor Demo App', function() {
  it('should have a title', function() {
    browser.get('https://www.gmail.com');

    expect(browser.getTitle()).toEqual('Gmail');
  });
});
```

Protractor - Interacting with elements

```
describe('Protractor Demo App', function() {  
  it('should add one and two', function() {  
    browser.get('http://juliemr.github.io/protractor-demo/');  
    element(by.model('first')).sendKeys(1);  
    element(by.model('second')).sendKeys(2);  
  
    element(by.id('gobutton')).click();  
  
    expect(element(by.binding('latest')).getText()).  
      toEqual('3');  
  });  
});
```


Enter CodeceptJS

What is CodeceptJS?

- JavaScript
- Framework for e2e acceptance tests
- Engine connects to library of choice - Puppeteer, webdriver, Nightmare, Protractor
- Visual Testing

Why CodeceptJS?

- JavaScript
- API
 - Easy to read, write and develop
- Tests are written from the user's perspective

CodeceptJS

```
Feature('Home Page');
```

```
Scenario('Login to application', (I) => {  
  I.amOnPage('/');  
  I.click('Login');  
  I.see('Welcome');  
});
```

CodeceptJS (selenium example)

```
Feature('First Test Case');

Scenario('Go To Gmail', (I) => {
  I.amOnPage('https://www.gmail.com');
  console.log('Successfully opened the website www.gmail.com');
  I.wait(5);
});
```

CodeceptJS (protractor examples)

```
const assert = require('assert');

Feature('Demo');

Scenario('Check Site Title', async (I) => {
  I.amOnPage('/');
  const title = await I.grabTitle()
  assert.equal(title, 'Demo')
});

Scenario('Addition Works', async (I) => {
  I.amOnPage('/');
  I.fillField('first', 1);
  I.fillField('second', 2);
  I.click('Go');
  const result = I.grabTextFrom('#result');
  assert.equal(result, '3');
});
```

CodeceptJS Quickstart

Install CodeceptJS with Puppeteer

```
npm install codeceptjs puppeteer --save-dev
```

Initialize CodeceptJS in current directory by running:

```
npx codeceptjs init
```

Answer questions with defaults. When asked about helpers select Puppeteer

Create first test

```
npx codeceptjs gt
```

CodeceptJS Quickstart (continued)

Puppeteer starts a browser without showing its window. To see the browser, edit `codecept.json` config and set `show: true` for Puppeteer:

```
{
  "helpers": {
    "Puppeteer": {
      "url": "http://localhost:8000",
      "show": true,
    }
  }
}
```


CodeceptJS Quickstart (continued)

After writing your first test you can run it by using this command:

```
npx codeceptjs run --steps
```

Or add it as as npm script in your package.json:

```
"scripts": {  
  "test": "codeceptjs run --steps"  
},
```

Resources

- <https://codecept.io>
- <https://github.com/iherndon/codecept-demo>
- <https://github.com/iherndon/simple-addition>



Thank you.

