



BDD - Testing - TestCafe

Manuel Armillas Hernández
Pablo Bande Sánchez-Girón



Contents

1. BDD

1.2 what is BDD?

1.3 Important considerations on BDD

1.4 Some advantages of BDD

1.5 More advantages of BDD

1.6 How to implement the BDD approach?

1.7 BDD or TDD?



Contents

2. Functional testing

2.1 What do you test?

2.2 Steps

2.3 Do's and Don'ts

2.4 Visual Regression Testing



Contents

3. TestCafé

3.1 Install TestCafé

3.2 Simple Test

3.3 Test results

3.4 Fixture Class and Test Object

3.5 Expect

3.6 Interesting TestCafé options

3.7 VSCode: TestCafé Test runner

Behaviour Driven Development

The background features a series of dark grey, three-dimensional rectangular planes that create a sense of depth and perspective, receding towards the right. Two distinct parallelogram shapes are integrated into the design: a light green one positioned higher and further back, and a blue one positioned lower and closer to the foreground. The overall aesthetic is modern and technical.



1.2 What is BDD?

- Behaviour Driven Development
- Development strategy
- Branch of Test Driven Development (TDD)
- Defines a common vocabulary between stakeholders and engineers.



1.3 Important considerations on BDD

- Each consideration **MUST** become into user stories, defining concrete examples
- Each example **MUST** be a scenario of a user on the system
- User behaviour and not Unit Test
- 'Given-When-Then' and 'Role-Feature-Reason'

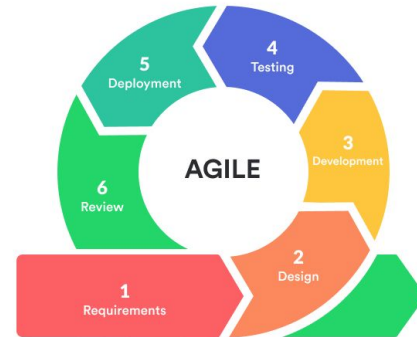


1.4 Some advantages of BDD

1. Defines behaviour and not tests
2. Improve communication between the members of the team
3. Sort learning curve

1.5 More advantages of BDD

1. It's not technical
2. Let you accept functionalities before development
3. Fits well in agile methodologies





1.6 How to implement the BDD approach?

Gherkin: 'Given-When-Then'

Feature: Refund item // *software feature description*

Scenario: Jeff returns a faulty microwave // *concrete example*

Given Jeff has bought a microwave for \$100 // *initial context*

And he has a receipt

When he returns the microwave // *event*

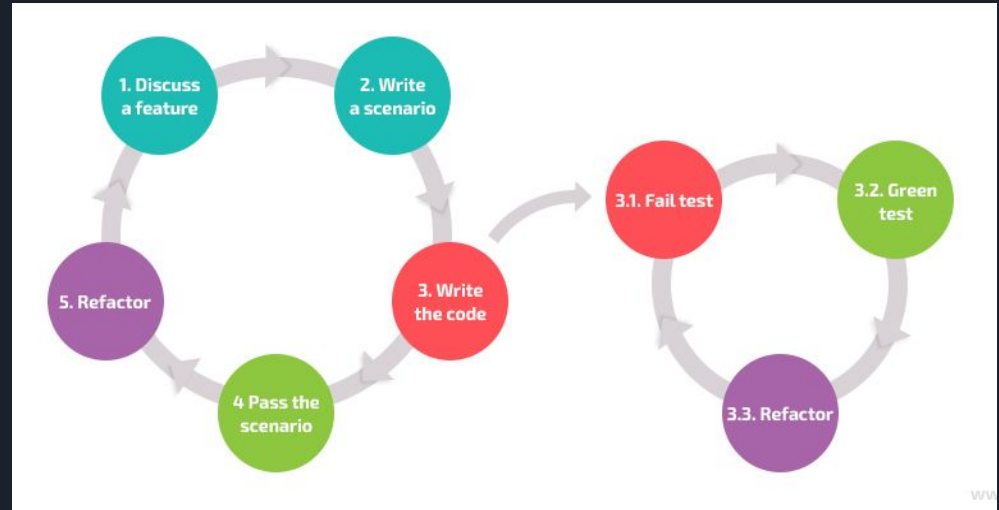
Then Jeff should be refunded \$100 // *expected outcome*

'Role - Feature - Reason'

As a seller, **I want to** receive notification for order creation **so I can** arrange delivery as fast as I can.

1.7 TDD or BDD?

- Different scope
- TDD: development practice
- PDD: team methodology
- Combine both



Functional Testing





2. Functional testing

- Validates functional specifications
- Black box testing
- Simulate user behavior



2.1 What do you test?

- Mainline functions
- Basic usability
- Basic accessibility
- Error conditions





2.2 Steps

1. User requirements
2. Test case
 - 2.1. Test input
 - 2.2. Test output
3. Execute test cases and validate results
4. Log defects and get them fixed

2.3 Do's and Don'ts



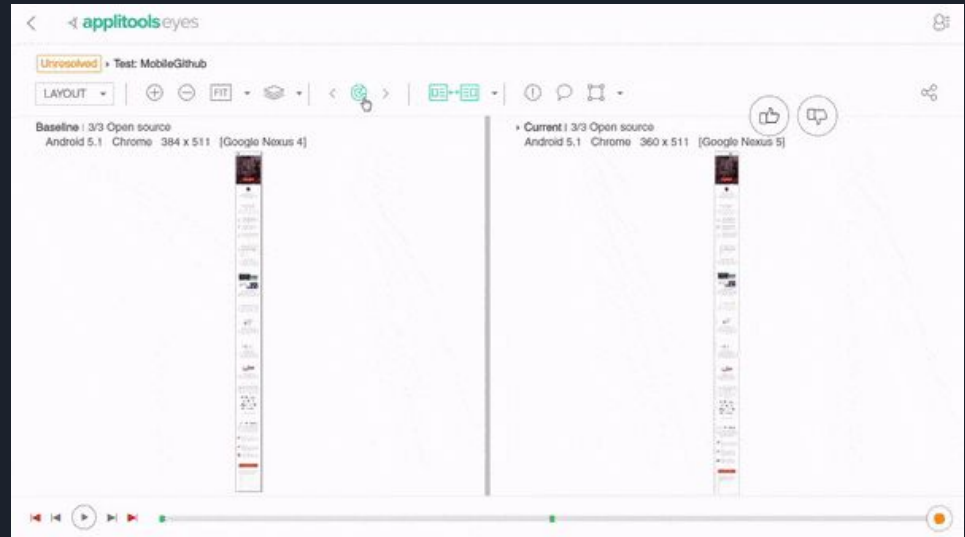
DO's



DONT's

2.4 Visual Regression Testing

- Validate appearance
- Front end
- Detect changes





TestCafé®



3. TestCafé

- E2E testing tool for websites and app's
- Parallel Testing
- Supports many browsers
 - Internet Explorer and Microsoft Edge
 - Google Chrome
 - Mozilla Firefox





3.1 Install TestCafé

Global

```
npm install -g testcafe
```

Local

```
npm install --save-dev testcafe
```



3.2 Simple test

```
import { Selector } from 'testcafe';

fixture `Getting Started`

  .page `http://devexpress.github.io/testcafe/example`;

test('My first test', async t => {

  await t

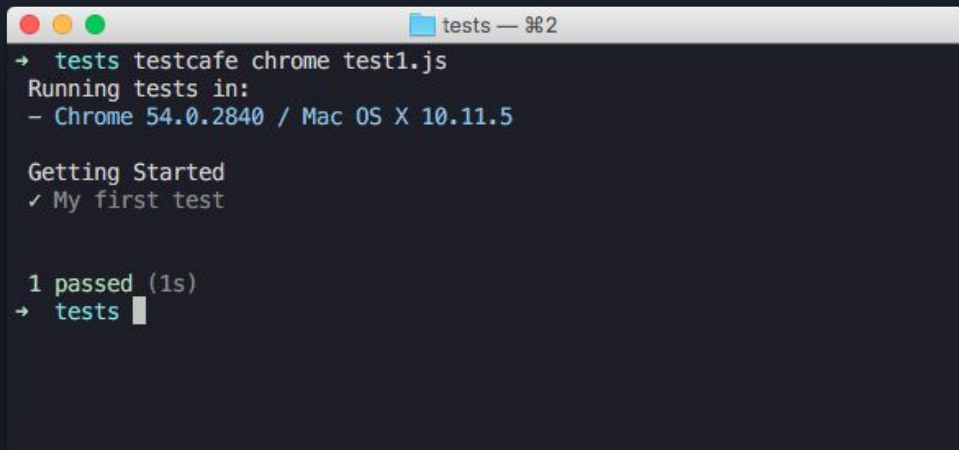
    .typeText('#developer-name', 'John Smith')

    .click('#submit-button');

  .expect(Selector('#article-header').innerText).eql('Thank you, John Smith!');

});
```

3.3 Test results

A terminal window with a title bar containing three colored circles (red, yellow, green) and the text "tests — 902". The terminal content shows a command being executed, the environment being used, a test starting, and a successful result.

```
→ tests testcafe chrome test1.js
Running tests in:
- Chrome 54.0.2840 / Mac OS X 10.11.5

Getting Started
✓ My first test

1 passed (1s)
→ tests █
```



3.4 Fixture Class and Test Object

test:

`.after()` `.before()` `.page()` `.meta()` `.skip()` `.only()`

fixture: `.aftereach()` `.before.each()`



3.5 Expect

```
.expect(Selector('#text').innerText).eql('Thank you, John Smith!');  
  
.expect(Selector('#text').innerText).contains('Thank you');  
  
.expect(Selector('#element').offsetHeight).gt(400);  
  
.expect(Selector('#element').exists).ok();
```




3.6 Interesting TestCafé options

```
testcafe --list-browsers
```

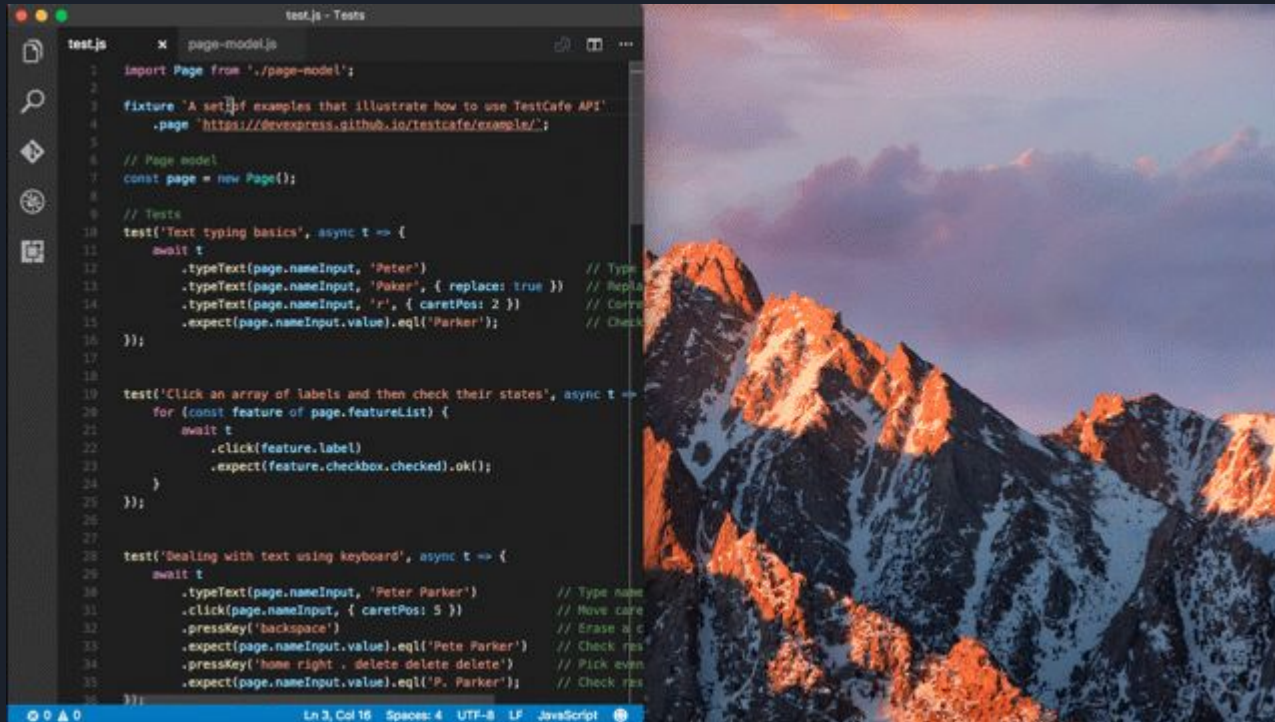
```
testcafe chrome,path:/applications/safari.app  
tests/sample-fixture.js
```

```
testcafe all tests/sample-fixture.js
```

```
testcafe "firefox:headless" tests/sample-fixture.js
```

```
testcafe "chrome:emulation:device=iphone X" tests/sample-fixture.js
```

3.7 VSCode: Test Café Test Runner



Questions



Bibliography

BDD

<https://www.itdo.com/blog/que-es-bdd-behavior-driven-development/>

<https://cucumber.io/docs/gherkin/reference/#example>

<https://steelkiwi.medium.com/how-to-describe-user-stories-using-gherkin-language-8cfc6b888df>

<https://www.visual-paradigm.com/scrum/user-story-role-feature-reason-template/>

<https://medium.com/@nic/writing-user-stories-with-gherkin-dda63461b1d2>

<https://www.javiergarzas.com/2014/12/bdd-behavior-driven-development-1.html>

<https://www.itdo.com/blog/tdd-vs-bdd-expectativas-de-calidad-de-tus-desarrollos/>

<https://www.pluralsight.com/blog/software-development/tdd-vs-bdd>

<https://medium.com/walmartglobaltech/should-we-use-tdd-or-bdd-or-both-801b6a2c6f31>

Functional Test

<https://www.guru99.com/functional-testing.html>

<https://www.simform.com/functional-testing/>

<https://testsigma.com/blog/different-functional-testing-types-explained-in-detail/>

TestCafé

<https://testcafe-discuss.devexpress.com/t/how-to-check-whether-an-element-is-visible/548>

<https://testcafe.io>

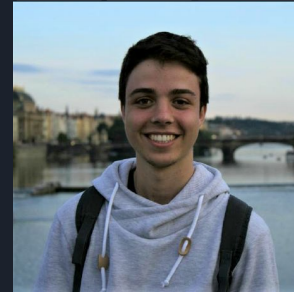
<https://www.youtube.com/watch?v=i5-EUAlcxLA>

<https://www.youtube.com/watch?v=N-cN9VPMcMs>

Contact us



Manuel Armillas Hernández
alu0101243498@ull.edu.es



Pablo Bande Sánchez - Girón
alu0101225296@ull.edu.es