

Behavioral-Driven Development

Table of Contents

1. Principles	1
2. Examples	1
2.1. Some review activity (No code)	1
2.2. This course materials! (No code)	2
2.3. Android example	2
2.4. Angular example	3
3. Supported languages	5
4. Build	5
5. Find a plugin for your IDE	5
6. Gherkin tips	6
6.1. Avoid duplication	6
6.2. Grouping step definitions	6
7. Scenario outline (template)	7
8. Language support	7
Appendix A: Useful links	7

1. Principles

- Given
- When
- Then
- And

2. Examples

2.1. Some review activity (No code)

```
Feature: Reviewing a Ph.D. Thesis
  Every PhD thesis review has some recurrent steps

Scenario: A reviewer, being an expert on the field, should be cited somewhere
  Given A PhD thesis to review
  And a reviewer John Smith
  Then The thesis should cite John Smith's work

Scenario: A Ph.D. thesis should be an original work
  Given A PhD thesis to review
  And a reviewer John Smith
  Then The PhD.pdf should be checked against plagiarism
```

Figure 1. Review PhD feature

2.2. This course materials! (No code)

```
#-----  
# Checking URLs  
# JMB - 2020  
#-----  
# language: en  
Feature: Teaching Materials Quality Assessment  
    Every material should have correct links  
  
Scenario: The URLs mentioned in an AsciiDoc document should be verified (non 404)  
    Given An AsciiDoc file  
    Then All the URLs should be active
```

Figure 2. Check URLs feature

2.3. Android example

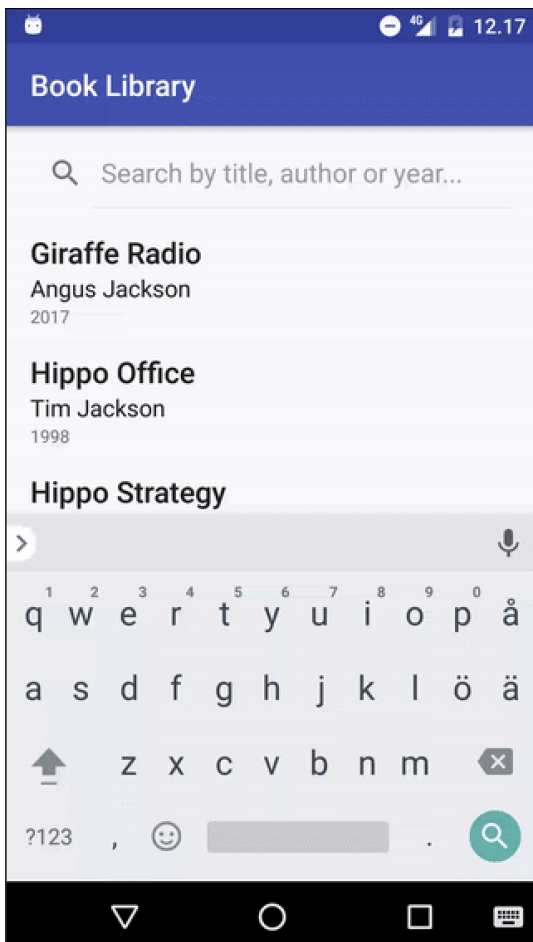


Figure 3. An Android app (source [here](#))

A feature for this app (source [here](#))

Feature: Book Search

Scenario: Search books by author

Given there's a book called "Tips for job interviews" written by "John Smith"

And there's a book called "Bananas and their many colors" written by "James Doe"

And there's a book called "Mama look I'm a rock star" written by "John Smith"

When an employee searches by author "John Smith"

Then 2 books should be found

And Book 1 has the title "Tips for job interviews"

And Book 2 has the title "Mama look I'm a rock star"

2.4. Angular example

A "Getting Started" app

<https://github.com/petermorlion/angular-getting-started>

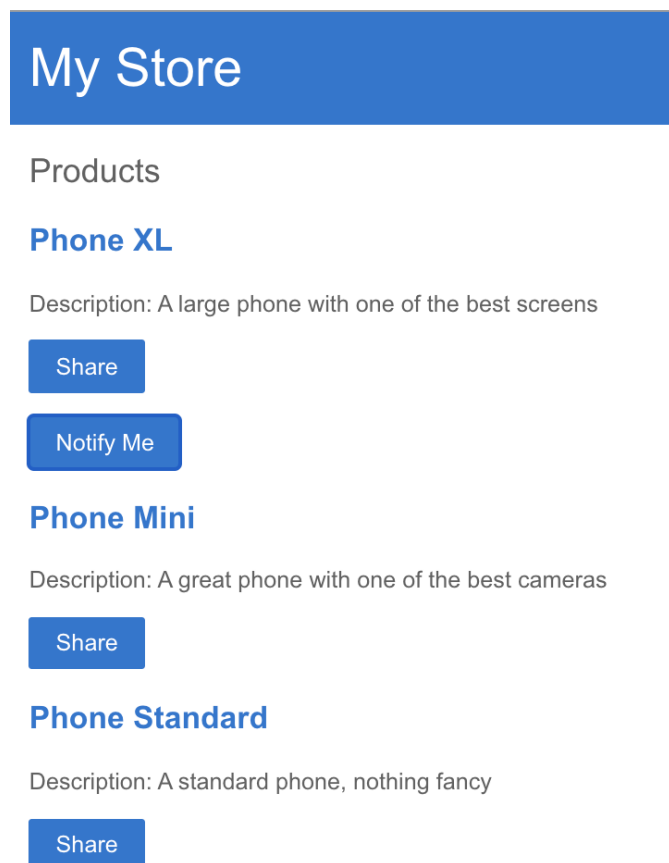


Figure 4. An Angular example app

Some feature description

```

Feature: Automatic discounts for premium customers
  Premium customers should automatically get a
  discount of 10% on purchases over $100.

  Scenario: Purchase over $100
    Given a premium customer
    And an order containing
      | item | amount | price |
      | pencil | 100 | 1 |
      | paper | 1 | 35 |
    When the customer checks out
    Then the total price should be 121.5

```

Figure 5. Some feature example

Running Cucumber (fail)

```

[→ /tmp/angular-getting-started git:(master) * cucumber features/products-list.feature
Feature: Products List

  Scenario: Load the products list # features/products-list.feature:3
    When we request the products list # features/products-list.feature:4
    Then we should receive # features/products-list.feature:5
      | name | description |
      | Phone XL | A large phone with one of the best screens |
      | Phone Mini | A great phone with one of the best cameras |
      | Phone Standard | A standard phone, nothing fancy |

1 scenario (1 undefined)
2 steps (2 undefined)
0m0.051s

You can implement step definitions for undefined steps with these snippets:

When('we request the products list') do
  pending # Write code here that turns the phrase above into concrete actions
end

Then('we should receive') do |table|
  # table is a Cucumber::MultilineArgument::DataTable
  pending # Write code here that turns the phrase above into concrete actions
end

```

Share your Cucumber Report with your team at <https://reports.cucumber.io>

Command line option: `--publish`
 Environment variable: `CUCUMBER_PUBLISH_ENABLED=true`
 cucumber.yml: `default: --publish`

More information at <https://reports.cucumber.io/docs/cucumber-ruby>

To disable this message, specify `CUCUMBER_PUBLISH_QUIET=true` or use the `--publish-quiet` option. You can also add this to your `cucumber.yml`:
`default: --publish-quiet`

Figure 6. First run of the tests

Write Steps definitions

This is code linking the assertions with the running code.

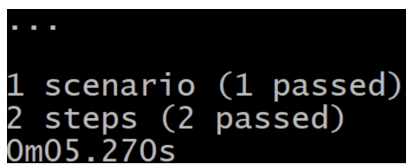
Steps definition

```
import { When, Then } from 'cucumber';

When('we request the products list', function () {
  // Write code here that turns the phrase above into concrete actions
  return 'pending';
});

Then('we should receive', function (dataTable) {
  // Write code here that turns the phrase above into concrete actions
  return 'pending';
});
```

Running Cucumber (pass)



```
...
1 scenario (1 passed)
2 steps (2 passed)
0m05.270s
```

Figure 7. New run of the tests

3. Supported languages

Ruby (origin), Java, JavaScript,

- [IntelliJ Cucumber for Java plugin](#)
- [Cucumber Eclipse plugin](#)

4. Build

Make sure to integrate BDD in your build.

For maven/gradle, see <https://cucumber.io/docs/tools/java/#build-tools>

5. Find a plugin for your IDE

Feature: Products List

Scenario: Load the products list

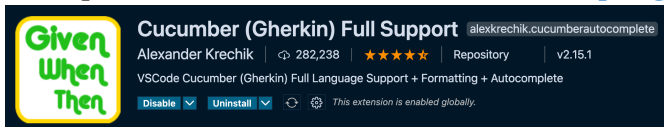
When we request the products list

Then we should receive

<code>name</code>	<code>description</code>
<code>Phone XL</code>	<code>A large phone with one of the best screens</code>
<code>Phone Mini</code>	<code>A great phone with one of the best cameras</code>
<code>Phone Standard</code>	<code>A standard phone, nothing fancy</code>

Figure 8. Cucumber plugin in use in VS Code

Example for VS Code: <https://github.com/alexxkrechik/VSCucumberAutoComplete>



6. Gherkin tips

6.1. Avoid duplication



```
Given I go to the home page
Given I check the about page of the website
Given I get the contact details
```



```
Given I go to the {} page
```

Generic Step definition

```
@Given("I go to the {string} page")
public void i_want_to_open_page(String webpage) {
    webpageFactory.openPage(webpage);
}
```

6.2. Grouping step definitions

One file for each major domain object.

```
EmployeeStepDefinitions.java
EducationStepDefinitions.java
ExperienceStepDefinitions.java
AuthenticationStepDefinitions.java
```

7. Scenario outline (template)

```
Scenario Outline: eating
  Given there are <start> cucumbers
  When I eat <eat> cucumbers
  Then I should have <left> cucumbers
```

Examples:

start	eat	left
12	5	7
20	5	15

8. Language support



Don't forget the `# language: fr!`

```
# language: fr
Fonctionnalité: Servir du café
  Afin de gagner de l'argent
  Les clients doivent être capables
  d'acheter du café à toutes heures

  Scénario: Acheter le dernier café
    Etant donné qu'il reste 1 café dans la machine
    Et que j'ai mis 1 dollar
    Quand j'appuie sur le bouton de la machine
    Alors je devrai recevoir un café
```

Figure 9. Example of feature in French ([source](#))

Appendix A: Useful links

- [Cucumber reference site](#)
- [A 10' tutorial](#)
- [An Android example](#)
- [An Angular example](#)