

Stay Focus

Mode Brain Fuck

ON

Définition: Procédure permettant de **vérifier** que le produit répond aux attentes de l'utilisateur. On vérifie le produit dans son **ensemble**, avec des **scénarios** réalistes d'utilisation.

Annotation @Test

Méthode de Test de
la Story Homepage

```
// Test de la Story #1-homepage (https://trello.com/c/WKTneu9o/1-homepage)
@Test
public void testHomepage() throws Exception {
    // Scénario: Inscription depuis la homepage
    driver.get("https://www.meetup.com/fr-FR/");
    assertEquals(driver.findElement(By.cssSelector("button-register")), "Inscription");
    driver.click(By.cssSelector("button-register"));
    assertEquals(driver.getCurrentUrl(), "https://secure.meetup.com/register/?_locale=fr-FR");

    // Scénario suivant
    // TODO
    // To Be Completed By Coders From Coding Factory
}
```

Assertion de l'url de la
seconde page

Acceptance Test du
scénario d'inscription

Acceptance Test

Ecrit par le Développeur/Testeur

Basé sur les critères d'acceptation (écrit par le PO)

Le développeur documente le test d'acceptation puis le développe.

Tests orientés Full Stack

Étiquettes

Homepage Feature

+

Description [Éditer](#)

Homepage

Story

EN TANT QUE

Utilisateur non connecté de Meetup

JE VEUX

accéder à une page d'accueil

AFIN DE

connaître les différentes fonctionnalités du site



Critères D'acceptation

[Supprimer...](#)

0%

☐

Le titre de la page d'accueil doit être: *Partagez vos passions | Meetup*

☐

La description de la page d'accueil doit être: *Partagez vos passions et faites bouger votre ville ! Meetup vous aide à rencontrer des personnes près de chez vous, autour de vos centres d'intérêt.*

☐

La punchline (h1) du site doit être: *Vous avez une passion ?*

☐






La sous punchline doit être: *Lancez-vous avec Meetup*

☐





Un bouton d'inscription rouge doit être présent en haut de la page, juste après la punchline, il doit contenir le texte : *Inscription* et doit contenir un lien qui me mène vers la page "<https://secure.meetup.com/register>"

[Ajouter un élément...](#)

Ajouter

-  Membres
-  Étiquettes
-  Checklist
-  Échéance
-  Pièce jointe

Actions

-  Déplacer
-  Copier
-  S'abonner
-  Archiver

[Partagez et plus...](#)

Story homepage

Le critère d'acceptation est écrit par le PO

Un scénario d'acceptation est rédigé par le développeur:

Scénario d'inscription:

- Lorsqu'un utilisateur est sur la page d'accueil et souhaite s'inscrire
- Un bouton «inscription» est présent
- Lorsqu'il clic il est redirigé vers la page d'inscription.

Problématique Test Fonctionnel & Acceptance

Interprétation par développeur

Réécriture des critères d'acceptation en scénarios = double travail

Doublon de spécifications (critères + scénarios)

Scénarios peu compréhensibles par le PO

ATDD

Acceptance Test Driven Development

Spécifier par le test

Passage d'une story basée sur des critères d'acceptation à une story basée sur des tests d'acceptation

Rédaction des Tests en langage naturel par le PO

Sans ATDD – Exemple

Story: Partage de vidéos

Critères d'acceptation:

- L'utilisateur peut partager une vidéo
- Pas de doublon de partage possible
- Si doublon, rejet du partage

Acceptance Test pour valider ces critères

- Démarrer sur la page Vidéo
- Soit la vidéo «Daft Hands - Harder, Better, Faster, Stronger» partagée avec «Juste Leblanc»
- Le titre de la page devrait être «Vidéos»
- Rechercher «Daft Hands»
- Le titre de la page devrait être «Résultat de recherche - Daft Hands»
- Cliquer sur le lien «Daft Hands - Harder, Better, Faster, Stronger»
- Cliquer sur le bouton partager
- Le titre de la page devrait être «Partager - Daft Hands - Harder, Better, Faster, Stronger»
- Sélectionner le groupe d'amis «Amis du lycée»
- Saisir dans le champs prénom «Juste»
- Saisir dans le champs nom «Leblanc»
- Cliquer sur le bouton partager la vidéo
- Le message la vidéo «Daft Hands - Harder, Better, Faster, Stronger» est partagée avec «Juste Leblanc»
- Les personnes partageant la vidéo sont listés de la façon suivante
- | groupe | prénom | nom |
- | Amis du lycée | Juste | Leblanc |
- | Collectionneur | François | Pignon |
- Cliquer sur le bouton partager
- Le titre de la page devrait être «Partager - Daft Hands - Harder, Better, Faster, Stronger»
- Sélectionner le groupe d'amis «Amis du lycée»
- Saisir dans le champs prénom «Juste»
- Saisir dans le champs nom «Leblanc»
- Cliquer sur le bouton partager la vidéo
- Le message d'erreur doit être le suivant la vidéo «Daft Hands - Harder, Better, Faster, Stronger» est déjà partagée avec «Juste Leblanc»

Avec ATDD – Exemple

Story: Partage de vidéos

Critères d'acceptation:

- L'utilisateur peut partager une vidéo
- Pas de doublon de partage possible
- Si doublon, rejet du partage

Acceptance Test basé sur des scénarios

- Action
- Point de verification
- Données

Spécification exécutable – Partager une video, rejet des doublons de partage

Soit la vidéo «Daft Hands – Harder, Better, Faster, Stronger» partagée avec «Juste Leblanc»

Lorsque l'utilisateur partage la video «Daft Hands – Harder, Better, Faster, Stronger» avec «Juste» «Leblanc»

Alors la liste des contacts partageant la vidéo est inchangée

Et le message d'erreur suivant est affiché : la vidéo «Daft Hands – Harder, Better, Faster, Stronger» est déjà partagée avec «Juste Leblanc»

BDD

Behaviour Driven Development

Orienter ses tests en fonction des comportements utilisateur

Gherkin : Given – When - Then

Formalisme d'aide à la rédaction des Tests d'Acceptance

Acceptance Test basé sur des scénarios BDD

- Action
- Point de verification
- Données

Spécification exécutable - Partager une video, rejet des doublons de partage

Given

Soit la vidéo «Daft Hands - Harder, Better, Faster, Stronger» partagée avec «Juste Leblanc»

Then

Lorsque l'utilisateur partage la video «Daft Hands - Harder, Better, Faster, Stronger» avec «Juste» «Leblanc»

Alors la liste des contacts partageant la vidéo est inchangée

Et le message d'erreur suivant est affiché : la vidéo «Daft Hands - Harder, Better, Faster, Stronger» est déjà partagée avec «Juste Leblanc»

Description Gherkin

Feature: Fonctionnalités de ma page d'accueil

Scenario: Vérification du titre et de la description

Given je suis sur la homepage

Then le titre doit être "Partagez vos passions | Meetup"

And la description doit être "Partagez vos passions et faites bouger votre ville ! Meetup vous aide à rencontrer des personnes près de chez vous, autour de vos centres d'intérêt."

Story & Scénario

Traduction Cucumber d'un Step

```
@Given("^je suis sur la homepage$")
public void je_suis_sur_la_homepage() throws Throwable {
    driver.get("https://www.meetup.com/fr-FR/");
}
```

```
@Then("^le titre doit être \"([^\"]*)\"$")
public void le_titre_doit_être(String arg1) throws Throwable {
    assertEquals(driver.getTitle(), arg1);
}
```

Assertion du titre de la page

```
@Then("^la description doit être \"([^\"]*)\"$")
public void la_description_doit_être(String arg1) throws Throwable {
    // By CSS Selector
    assertEquals(driver.findElement(By.cssSelector("meta[name='description']")).
        getAttribute("content"), arg1);
}
```

On écrit le code d'un
step qu'une fois ! 😊

Avantage ATDD avec BDD

Pas d'interprétation possible

Pas de réécriture: ~~critère~~ -> test écrit par le PO

Scénarios compréhensibles par le PO car formalisé en langage humain

Avantage ATDD avec BDD

Spécifications exécutables par un outil de test

Modifier les spécifications entraîne la modification du code de test donc du code de l'application

**Spécification & Test & Code
toujours à jour et lié.**

Avantage ATDD avec BDD

Ecriture de scénarios paramétrables

Scenario Outline: Authentification sur un portail

Given Un compte de login "admin" et de mot de passe "admin159" valide

When J'ouvre le formulaire d'authentification

And Je saisis l'identifiant "<login>"

And Je saisi le mot de passe "<password>"

And Je valide le formulaire

Then Le message d'erreur "<message>" s'affiche

Examples:

<i>login</i>	<i>password</i>	<i>message</i>
admin	admin	L'identifiant ou le mot de passe est invalide.
admin2	admin159	L'identifiant ou le mot de passe est invalide.
admin2	admin2	L'identifiant ou le mot de passe est invalide.
admin	admin159	

Avantage ATDD avec BDD

Réutilisation du code de test, une fois le step existant dans la base de code.

```
@Given("^je suis sur la homepage$")
public void je_suis_sur_la_homepage() throws Throwable {
    driver.get("https://www.meetup.com/fr-FR/");
}
```

cucumber



Cucumber, quel intérêt ?

Framework BDD n°1

Existe pour pratiquement tous les langages (java, php, ruby, js, .net...)

Méthodes transposables dans d'autres langages

ATDD outillé

Utilisation du site Meetup.com

Base pour tester en ATDD des pages web

✦ Vous épargnez des heures de configs

Code sur mon Github (adesousa)

Liens pratiques

Explication du concept BDD

<http://www.pierregillon.com/2013/08/bdd-behaviour-driven-development.html>

Tableau d'exemples

<https://cucumber.io/docs/reference#data-table>

Options Cucumber

<https://testingneeds.wordpress.com/2015/09/15/junit-runner-with-cucumberoptions/>

Download Cucumber

```
cd /Library/Java/JUNIT
sudo wget http://repo1.maven.org/maven2/info/cukes/cucumber-
    java/1.2.5/cucumber-java-1.2.5.jar
sudo wget http://repo1.maven.org/maven2/info/cukes/cucumber-
    junit/1.2.5/cucumber-junit-1.2.5.jar
sudo wget http://repo1.maven.org/maven2/info/cukes/cucumber-
    core/1.2.5/cucumber-core-1.2.5.jar
sudo wget http://repo1.maven.org/maven2/info/cukes/
    gherkin/2.12.2/gherkin-2.12.2.jar
sudo wget http://repo1.maven.org/maven2/info/cukes/cucumber-
    jvm-deps/1.0.5/cucumber-jvm-deps-1.0.5.jar
sudo chown $USER:$USER /Library/Java/JUNIT/*
sudo chmod 755 /Library/Java/JUNIT/*
```

Selenium configuration for Junit

vim ~/.bash_profile

delete this line at the end of file

```
export CLASSPATH=/Library/Java/JUNIT/junit-4.12.jar:/Library/
Java/JUNIT/hamcrest-all-1.3.jar:/Library/Java/JUNIT/
chromedriver:/Library/Java/JUNIT/client-combined-3.8.1.
jar:/Library/Java/JUNIT/client-combined-3.8.1-sources.jar:
/Library/Java/JUNIT/selenium-server-standalone-3.8.1.jar:.
```

Put this line at the end of file

```
export CLASSPATH=/Library/Java/JUNIT/junit-4.12.jar:/Library/
Java/JUNIT/hamcrest-all-1.3.jar:/Library/Java/JUNIT/
chromedriver:/Library/Java/JUNIT/client-combined-3.8.1.
jar:/Library/Java/JUNIT/client-combined-3.8.1-sources.jar:
/Library/Java/JUNIT/selenium-server-standalone-3.8.1.jar:/
Library/Java/JUNIT/cucumber-java-1.2.5.jar:/Library/Java/
JUNIT/cucumber-junit-1.2.5.jar:/Library/Java/JUNIT/
cucumber-core-1.2.5.jar:/Library/Java/JUNIT/gherkin-2.12.2
.jar:/Library/Java/JUNIT/cucumber-jvm-deps-1.0.5.jar:.
```

First of all, you need to compile your test class (from src folder)

Important, to avoid compile errors
cd src/

Tests

javac -d ../bin/ test/acceptance/Homepage*.java

Then run your test cases (from bin tests folder)

```
cd bin/
```

```
java org.junit.runner.JUnitCore test.acceptance.HomepageTest
```


Testez <<Meetup.com>>

Consultez le Product Backlog :

<https://trello.com/b/dERqaBBR/acceptance-testing-agile-testing-course>

Créer le Gherkin (fichier .feature) et les tests d'acceptance (fichier *Steps.java)

Code de test sur mon github

Objectif

Couverture fonctionnelle à 100%

Chaque test d'acceptation doit être testée par une fonction ATDD en gherkin

Si pas de Test côté Trello, rédiger (.feature) avant d'implémenter (*Step.java).

Point d'attention

La réutilisation du code: essayer d'écrire toujours de la même façon les différents steps.

Given je suis sur la homepage

≠

Given je vais sur la homepage

Cucumber créera alors 2 méthodes
plutôt que de réutiliser la même, vous
devrez alors dupliquer le code
d'implémentation.

