

T.D 1

Test-Driven Development (TDD), ou développement piloté par les tests en français, est une méthode de développement de logiciel qui consiste à concevoir un logiciel par des itérations successives très courtes (ou petits pas), telles que chaque itération est accomplie en formulant un sous-problème à résoudre sous forme d'un test avant d'écrire le code source correspondant, et où le code est continuellement remanié dans une volonté de simplification. [wikipedia]

Le fonctionnement du TDD est conseillé par ces points :

<p>Vous ne pouvez pas écrire de code de production tant que vous n'avez pas écrit un test unitaire qui échoue.</p>	<p>"You may not write production code until you have written a failing unit test."</p>
<p>Vous ne pouvez pas écrire plus de code de test qu'il n'en faut pour qu'un test unitaire échoue, et ne pas compiler revient à échouer.</p> <p>Vous ne pouvez pas écrire plus de code de production que nécessaire pour que le test unitaire actuellement en échec réussisse.</p> <p>Les anciens tests doivent toujours passer (non regression)</p>	<p>"You may not write more of a unit test than is sufficient to fail, and not compiling is failing."</p> <p>il n'est permis d'écrire que du code de production permettant directement de faire passer le test unitaire précédent, ni plus ni moins</p>

Le cycle de vie du TDD devient alors :

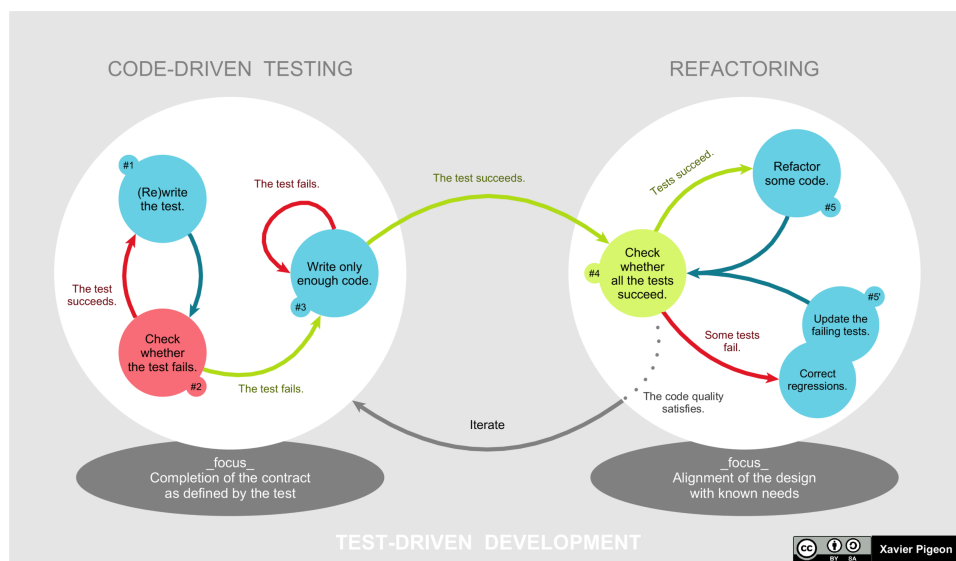


FIGURE 1 – TDD cycle de vie

Mais on trouve aussi 2 écoles pour le TDD : TDD classique (sur classes) et ATDD dès la création des tests d'acceptabilité (acceptance test) voir Figure 2. Si vous êtes responsable de la spécification et codage d'une feature complète, je vous conseille de partir de ATDD (partie gauche de la figure).

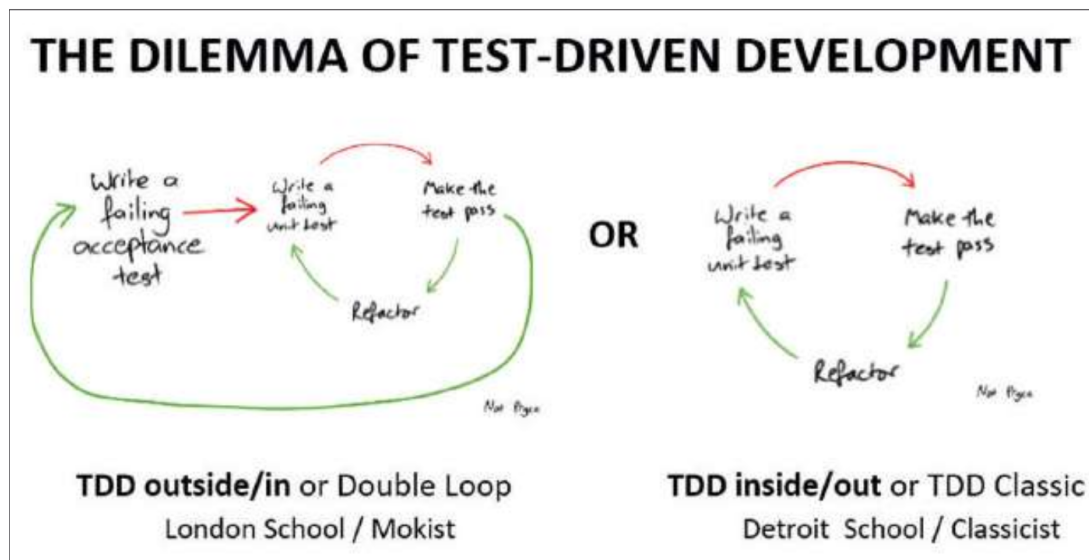


FIGURE 2 – ATDD et TDD

Exercice 1

Voyons comment cela fonctionne avec un exemple. Soit le site d'e-commerce "les_40_voleurs". Et soit la story suivante "Story1 : Un client peut se connecter au site avant de faire un achat".

Question 0.0.1 Décrivez les tests d'acceptabilité, puis, pour chacun, indiquez le type de test à effectuer (conformité, robustesse, sécurité, etc.)

Question 0.0.2 Prenez l'exemple de *TC1 verify that customer with valid username & password can login successfully* et écrivez le failing acceptance test. On décrira les étapes du test en français (de façon algorithmique avec des procédures de votre choix). Si vous connaissez le framework Selenium, vous pouvez directement écrire le code. Selenium sera vu dans un TP.

Passons à l'étape 2 "Write a failing unit test".

Question 0.0.3 Comment se déroule l'étape 2 ?

Question 0.0.4 Donnez l'exemple de tests unitaires (en français, ou PHPunit). Voyez vous où se trouve le dilemme ?

Question 0.0.5 Avant de foncer sur du code, il faut identifier les classes utilisées. Dans une structure de code de site Web utilisant un MVC, quelles peuvent être ces classes ?

Prenons l'exemple de la classe `ModèleUser` et de sa fonction `User loginuser(user,pass)`.

Question 0.0.6 Donnez au moins un test pour la fonction `login(user,pass)` du modèle en PHP.

Question 0.0.7 Peut-t-on faire du test en isolation du modèle ?

Question 0.0.8 Peux t-on faire un test unitaire du contrôleur pour la fonction loginUser() ?

Question 0.0.9 Si le temps le permet refaire un test d'acceptabilité pour le cas ou un mot de passe vide est donné (vulnérabilité dans le TOP10 OWASP).