大学共同利用機関法人 情報・システム研究機構
国立情報学研究所
National Institute of Informatics

トップエスイー ソフトウェア開発実践演習

実践ATDD / TDD

株式会社NTTデータ　加藤 史也
株式会社 日立製作所　相樂 恭宏
日本電気株式会社　梅原 崚

テストにおける問題点

ソフトウェア開発において, JUnitやSelenium等のテスティングフレームワークによるテストの自動実行は一般的な取り組みとなっている. しかし, 本演習の参加メンバが所属する現場では, 設計とテストの間で整合性がとれていないケースや, チームメンバがテストの意義を理解せず, テストが形骸化してしまうといった問題が発生している. また, テストの活動において, 前述のような問題は広く発生しうると考える.

手法・ツールの検証

テストの作成を設計に準ずる活動として位置付ける手法：Acceptance Test Driven Development (ATDD)およびTest Driven Development (TDD)の適用によって前述の問題の解決に繋がるかを考察する. ATDD, TDDを実際に適用したソフトウェア開発を通して, これらの手法によるメリット, デメリットを検証する. また, 得た知見をもとに, 実際の業務に適用するにあたっての留意点について考察する.

演習の実施内容

- **開発対象のアプリケーション**

本演習では, TODOリストのWebアプリケーション開発を対象として, ATDD, TDDを適用した. 今回の演習では4種類の機能(TODOの登録, 一覧表示, 完了, 期限が近いものの強調表示)を開発対象として設定し, 実装を行った.

- **演習のフローと所感**

本演習で実施した開発のフローを図1に示す. 開発の初めに受入テストシナリオを作成し, それを用いて顧客と合意形成を行う. その後, 対象とする受入テストシナリオを合格するようにプログラムを実装した. 受け入れテスト, 単体テストの先行作成は, 設計に相当する活動であり, 実装するプログラムが設計(=テスト)通りであるかを常に確認可能であることがATDD, TDDの利点であると感じた.

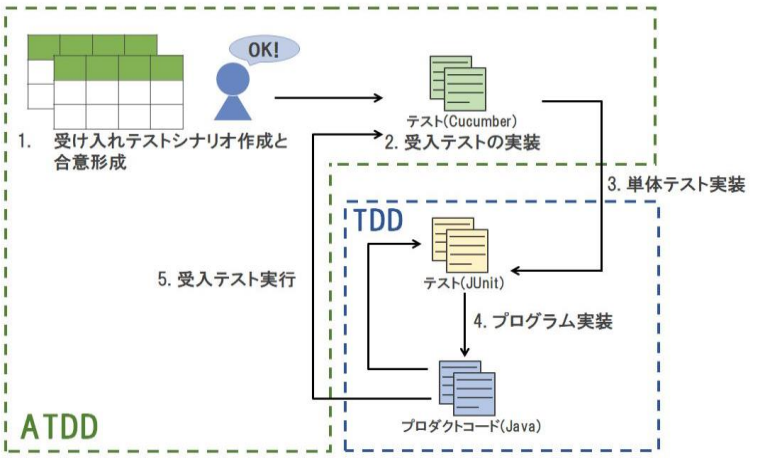

図1. 開発のフロー

期待できる効果

- **有意義なテストの作成**

ウォーターフォール開発では, テストを作成する時点で実装が完了しているため, 機能テストが実装を追認するだけの無意味なものになってしまう懸念がある. ATDDでは, テストを作成する時点で実装が存在していないことを保証できるため, 機能要件の達成を厳密にチェックするテストを作成することができ, テストの形骸化の防止に役立つと考える.

- **機能仕様の品質向上**

ATDDでは, 仕様をテスト項目の集まりとして表現する. ソフトウェアの操作や動作を具体的に検討する必要があるため, 仕様の漏れや曖昧な点を早期に発見することができる.

考慮・工夫が必要な点

- **テスト駆動開発という概念の導入コスト**

これから開発するソフトウェアのテストを作成するためには, 完成品を実装レベルで想像するという独特な感覚が要求される. さらに. テストフレームワークを利用する場合はその習得も必要であり, チーム内に未経験者が多い場合, モブプログラミングなど導入コストを削減する工夫が必要であると感じた.

- **ATDD適用の向き不向き**

どのような開発にもATDDが適しているとは限らないと考える. 例えば, テスト1件ごとの開発規模の見積りが難しいため, 担当の割り振りが必要な大規模開発には不向きである. また, 非機能要件のテストは, 実装が完了していないと一般的に作成が困難である. これらの観点を含め, ATDDの適用可否, 適用範囲の考慮が必要である.