



ACCEPTANCE – TEST

.....

true

Driven Development

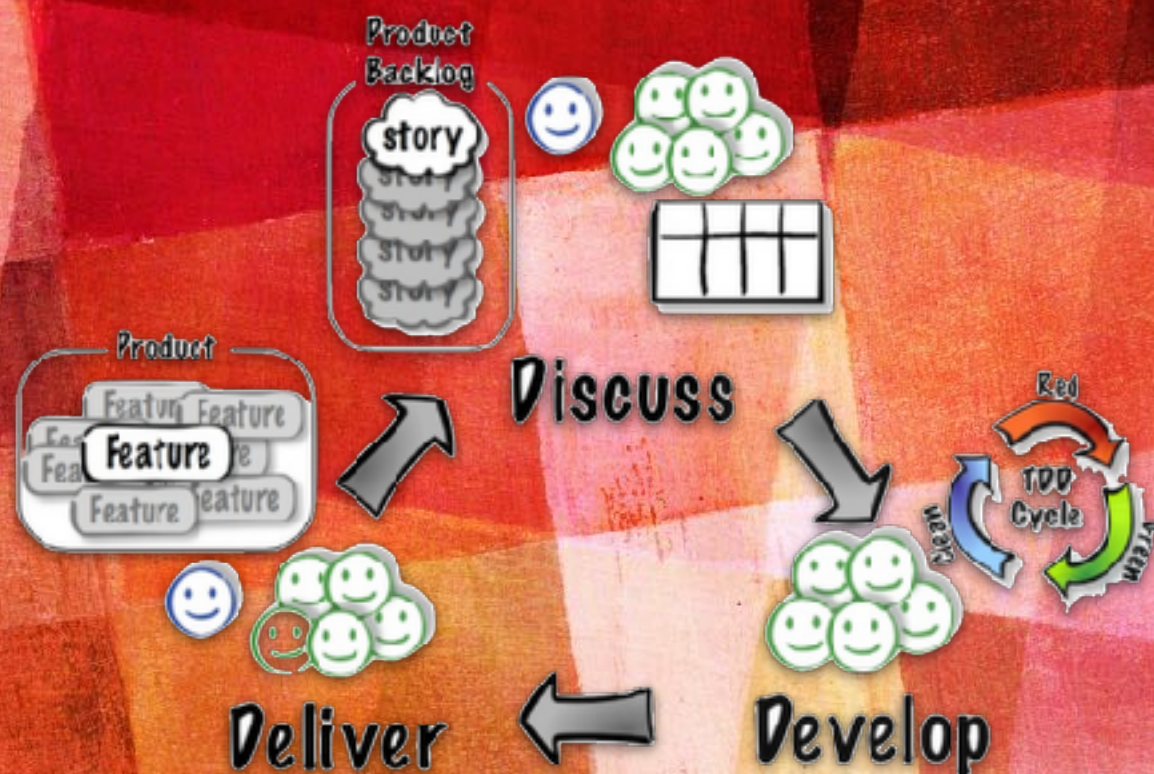


WHAT IS ATDD?

.....

ATDD หรือ Acceptance Test Driven Development คือ การสร้างความเข้าใจที่ตรงกันกับลูกค้าและทีมที่ใช้พัฒนาโปรแกรม เพื่อใช้ในการพัฒนาโปรแกรม ซึ่งจำเป็นต้องอาศัยความร่วมมือกันของทุกฝ่ายเพื่อให้เกิดความเข้าใจเกี่ยวกับงานที่ตรงกัน นั่นคือ A(Acceptance) ซึ่งจะใช้การพัฒนาโปรแกรมในรูปแบบของ TDD(Test Driven Development) โดย TDD จะอยู่ในองค์ประกอบย่อยของ ATDD ซึ่งมี 3 ส่วนคือ Discuss, Develop, Deliver

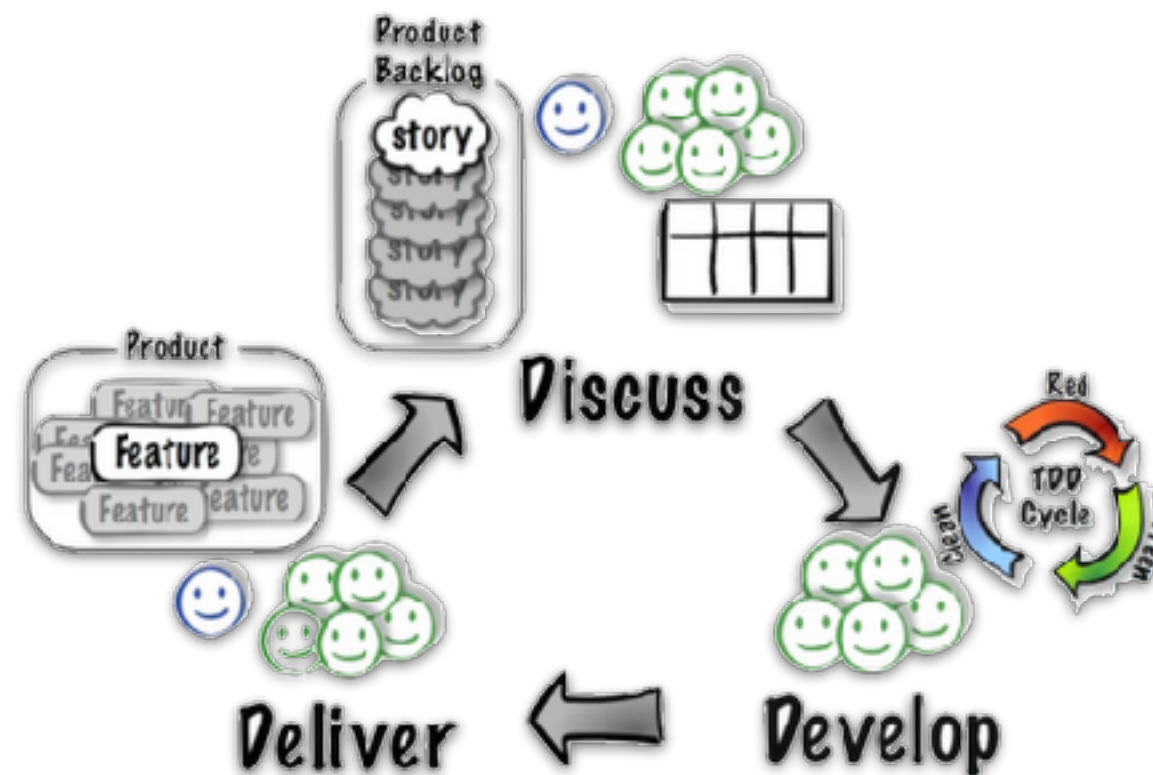
Acceptance Test Driven Development (ATDD) Cycle



WHAT IS ATDD?



1. Discuss เป็นขั้นตอนในการจัดการกับปัญหาเกี่ยวกับ Requirement โดยจะทำการวิเคราะห์ Story ที่อยู่ใน Backlog เพื่อให้เกิดความเข้าใจที่ตรงกันของทุกคนภายในทีม เพื่อให้เกิดการพัฒนาที่ถูกต้องและดีที่สุด



3. Deliver/Demo ขั้นตอนนี้เป็นการส่งมอบงาน หรือแสดงการทำงานของ Features ต่างๆที่ได้ทำการพัฒนา ให้กับผู้ที่เกี่ยวข้องกันได้เห็น ซึ่งก็คือการ Demo งานนั่นเอง ซึ่งจะแสดงให้เห็นว่าการพัฒนาโปรแกรมนั้นได้เป็นไปตามข้อตกลงที่ได้คุยกันไว้แล้ว ซึ่งก็อาจจะมีความต้องการเพิ่มเติมของลูกค้าใน Features เดิมหรือ Features ใหม่ๆเข้ามา ซึ่งจะนำไปพูดคุยกันต่อในการ Discuss ครั้งต่อไป

2. Develop เป็นขั้นตอนที่ต้องพัฒนาโปรแกรม โดยนำเอา Story ที่ได้ผ่านการ Discuss มาแล้วมาทำการพัฒนา ซึ่งจะทำการพัฒนาโดยใช้ TDD (Test Driven Development) ซึ่งมีกระบวนการในการพัฒนาไปพร้อมกันกับการเขียน Testcase เพื่อทดสอบโปรแกรมไปด้วย ซึ่งก็มีการนำ Automated testing เข้ามาช่วยในการทดสอบเช่นกัน

WHAT IS ROBOTFRAMEWORK?

Robot Framework คือ Generic test automation framework for acceptance testing and ATDD ซึ่งการใช้งานก็ค่อนข้างง่าย (สำหรับคนเขียนโปรแกรมไม่เป็น แต่พอจะรู้จักโครงสร้างของโปรแกรมมาบ้าง ก็ทำได้) Robot Framework implement มาจากภาษา Python ทั้ง version 2 และ version 3 ซึ่งวิธีการใช้งาน Robot Framework รองรับทั้ง 2 version แต่เท่าที่เห็น ปัจจุบัน คนไทยนิยมใช้กับ Python 2.7 (Python จะ support จนถึงปี 2020)

ROBOT FRAMEWORK

Generic test automation framework for acceptance testing and ATDD



WHAT IS SELENIUM?

Selenium คือ Software Testing Framework / Plugin ที่มีประสิทธิภาพมากตัวหนึ่ง
ไว้ใช้สำหรับทำ Automated Testing เขียน Test Case เพื่อทดสอบ Web Application นั้นเอง

Programming Lang.



Web Browser.



REPORT AND LOG



SuiteSetup Test Report

Generated
20180828 09:37:30 GMT+07:00
11 hours 7 minutes ago

Summary Information

Status: All tests passed

Start Time: SuiteSetup Test Log

End Time:

Elapsed Time: Test Statistics

Log File:

Generated
20180828 09:37:30 GMT+07:00
11 hours 8 minutes ago

Test Statistics

Critical Tests

All Tests

No Tags

SuiteSetup

Test Details

Totals

Tags

Name:

Total Statistics	Total	Pass	Fail	Elapsed	Pass / Fail
Critical Tests	2	2	0	00:00:18	<div></div>
All Tests	2	2	0	00:00:18	<div></div>

Statistics by Tag	Total	Pass	Fail	Elapsed	Pass / Fail
No Tags					<div></div>

Statistics by Suite	Total	Pass	Fail	Elapsed	Pass / Fail
SuiteSetup	2	2	0	00:00:18	<div></div>

Test Execution Log

SUITE SuiteSetup

Full Name: SuiteSetup
Source: /Users/jpac/RobotFW/Lab/SuiteSetup.robot
Start / End / Elapsed: 20180828 09:37:12.302 / 20180828 09:37:30.039 / 00:00:17.737
Status: 2 critical test, 2 passed, 0 failed
2 test total, 2 passed, 0 failed

TEST Test Suite Setup case 1

Full Name: SuiteSetup.Test Suite Setup case 1
Start / End / Elapsed: 20180828 09:37:12.375 / 20180828 09:37:25.619 / 00:00:11.244
Status: PASS (critical)

- SETUP SeleniumLibrary.Open Browser https://www.google.com, chrome
- KEYWORD SeleniumLibrary.Click Element id=lst-ib
- KEYWORD SeleniumLibrary.Input Text id=lst-ib, robotframework
- KEYWORD SeleniumLibrary.Press Key id=lst-ib, W3
- KEYWORD BuiltIn.Sleep 5

TEST Test Suite Setup case 2

Full Name: SuiteSetup.Test Suite Setup case 2
Start / End / Elapsed: 20180828 09:37:23.615 / 20180828 09:37:30.039 / 00:00:06.423
Status: PASS (critical)

- SETUP SeleniumLibrary.Open Browser https://www.google.com, chrome
- KEYWORD SeleniumLibrary.Click Element id=lst-ib
- KEYWORD SeleniumLibrary.Input Text id=lst-ib, hello world
- KEYWORD SeleniumLibrary.Press Key id=lst-ib, W3

Elapsed Pass / Fail

00:00:18

00:00:18

Elapsed Pass / Fail

Elapsed Pass / Fail

00:00:18

LIBRARY FOR ROBOTFRAMEWORK



ROBOT
FRAME
WORK/

Robot Framework is a generic test automation framework for acceptance testing and acceptance test-driven development (ATDD). It has easy-to-use tabular test data syntax and it utilizes the keyword-driven testing approach. Its testing capabilities can be extended by test libraries implemented either with Python or Java, and users can create new higher-level keywords from existing ones using the same syntax that is used for creating test cases.

Robot Framework project is hosted on [GitHub](#) where you can find further documentation, source code, and issue tracker. Downloads are hosted at [PyPI](#). The framework has a rich ecosystem around it consisting

STANDARD

Builtin

Provides a set of often needed generic keywords. Always automatically available without imports.

OperatingSystem

Enables various operating system related tasks to be performed in the system where Robot Framework is running.

String

Library for generating, modifying and verifying strings.

Process

Library for running processes in the system. New in Robot Framework 2.8.

EXTERNAL

Dialogs

Provides means for pausing the test execution and getting input from users.

Remote

Special library acting as a proxy between Robot Framework and test libraries elsewhere. Actual test libraries can be running on different machines and be implemented using any programming language supporting XML-RPC protocol.

Telnet

Makes it possible to connect to Telnet servers and execute commands on the opened connections.

DateTime

Library for date and time conversions. New in Robot Framework 2.8.5.

["http://robotframework.org/#libraries"](http://robotframework.org/#libraries)

INSTALL ROBOTFRAMEWORK WITH SELENIUM



➤ Download and Install Python 2.7

[“https://www.python.org/download/releases/2.7/”](https://www.python.org/download/releases/2.7/)

➤ Setup Environment

`c:\Python27;c:\Python27\Scripts;C:\Python27\Driver;%PATH%`

➤ Install Plugin for Robotframework

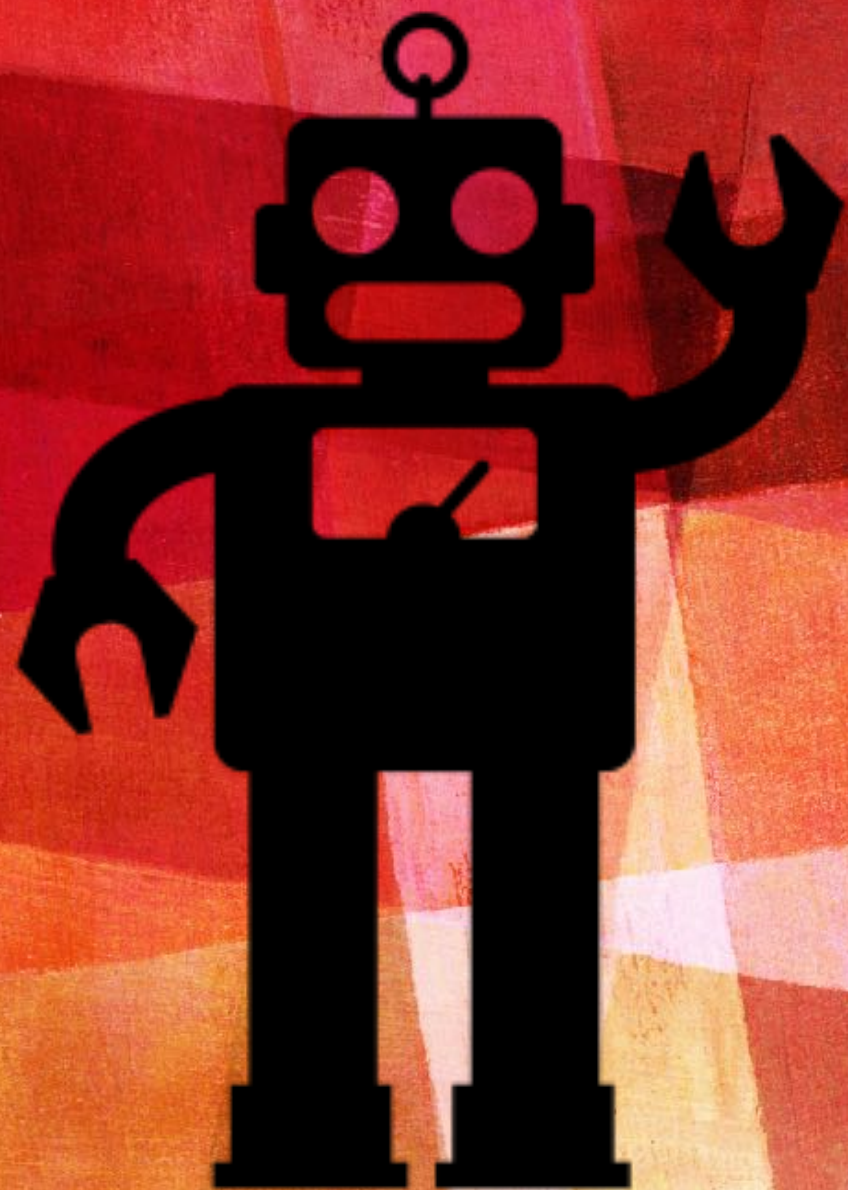
`pip install robotframework`

`pip install robotframework-selenium2library==1.8.0`

`pip install --upgrade robotframework-selenium2library`

➤ Editor recommended

Atom [“https://atom.io/”](https://atom.io/)



ROBOTFRAMEWORK STRUCTURE

.....

➤ *** Settings ***

ในส่วนของ Settings ใช้ในการ include file, ประกาศ document description, การทำ Suite Setup / Teardown / Test Setup / Teardown และการ import library

➤ *** Variables ***

มันก็คือการ ประกาศตัวแปรนั่นเอง

➤ *** Keywords ***

ในส่วนของ Keywords, Keyword ก็เหมือนกับการเขียน Function ในภาษาโปรแกรมมิ่งทั่วไป ดูตัวอย่างจะเข้าใจง่าย

➤ *** Test Cases ***

ส่วนของ Test Cases เป็นที่ที่ใช้ในการเขียน สิ่งที่เราจะ Test ลงไป

true



Workshop

HELLO ROBOT



- Create file hello.robot

```
Hello.robot x
1  *** Settings ***
2  Library    Selenium2Library
3
4
5  *** Testcases ***
6  Hello World
7      Open Browser    https://www.google.com    chrome
8
```

- Run with

`$pybot hello.robot`

- Help for pybot "pybot --help"

BE CAREFULL

true

```
Hello.robot x
1  *** Settings ***
2  Library    Selenium2Library
3
4
5  *** Testcases ***
6  Hello World
7  Open Browser    https://www.google.com    chrome
8
```

*Paragraph on everywhere
don't be use tab*

Between keyword and argument

space more than 2 spacebars

In keyword only 1 spacebar



SETTING – SUITE / TEST SETUP

- Create file SuiteSetup.robot

```
SuiteSetup.robot x
1  *** Settings ***
2  Library    Selenium2Library
3  #Suite setup    Open Browser    https://www.google.com    chrome
4  Test setup    Open Browser    https://www.google.com    chrome
5
6
7  *** Testcases ***
8  Test Suite Setup case 1
9      Click Element    id=lst-ib
10     Input Text    id=lst-ib    robotframework
11     Press Key    id=lst-ib    \\13
12     Sleep    5
13
14  Test Suite Setup case 2
15     Click Element    id=lst-ib
16     Input Text    id=lst-ib    helloworld
17     Press Key    id=lst-ib    \\13
18
```

- Run with
\$pybot SuiteSetup.robot
- Then unmark Suite setup and mark Test setup



SETTING - TAG



- Create file Tag.robot

```
Tag.robot
1  *** Settings ***
2  Library    Selenium2Library
3  Suite setup    Open Browser    https://www.google.com    chrome
4  Default Tags    Search Robotframework
5
6  *** Testcases ***
7  Test Suite Setup case 1
8      [Tags]    SearchRobotframework
9      Click Element    id=lst-ib
10     Input Text    id=lst-ib    robotframework
11     Press Key    id=lst-ib    \13
12     Sleep    5
13
14  Test Suite Setup case 2
15      [Tags]    Searchhelloworld
16      Click Element    id=lst-ib
17      Input Text    id=lst-ib    helloworld
18      Press Key    id=lst-ib    \13
19
```

- Run with
\$pybot Tag.robot

```
x ipao@macbookpro ~/RobotFW/Lab pybot -i SearchRobotframework Tag.robot
=====
Tag
=====
Test Suite Setup case 1                                     | PASS |
-----
Tag                                                         | PASS |
1 critical test, 1 passed, 0 failed
1 test total, 1 passed, 0 failed
=====
Output:  /Users/ipao/RobotFW/Lab/output.xml
Log:     /Users/ipao/RobotFW/Lab/log.html
Report:  /Users/ipao/RobotFW/Lab/report.html
ipao@macbookpro ~/RobotFW/Lab pybot -e SearchRobotframework Tag.robot
=====
Tag
=====
Test Suite Setup case 2                                     | PASS |
-----
Tag                                                         | PASS |
1 critical test, 1 passed, 0 failed
1 test total, 1 passed, 0 failed
=====
Output:  /Users/ipao/RobotFW/Lab/output.xml
Log:     /Users/ipao/RobotFW/Lab/log.html
Report:  /Users/ipao/RobotFW/Lab/report.html
```



VARIABLES



- Create file Variable.robot

```
Variable.robot x
1  *** Settings ***
2  Library    Selenium2Library
3
4  *** Variables ***
5  ${URL}     https://www.google.com
6  ${Keyword} robotframework
7  *** Testcases ***
8  Variables Test Case
9      Open Browser    ${URL}    chrome
10     Click Element    id=lst-ib
11     Input Text       id=lst-ib    ${Keyword}
12     Press Key        id=lst-ib    \\13
13     Sleep            5
14
```

- Run with
`$pybot Variable.robot`



KEYWORDS



- Create file Keyword.robot

```
Keyword.robot
1  *** Settings ***
2  Library    Selenium2Library
3
4  *** Testcases ***
5  Goto Search engine
6      Open Google Search Engine    robotframework
7
8  *** Keywords ***
9  Open Google Search Engine
10     [Arguments]    ${KEYWORD}
11     Open Browser    https://www.google.com    chrome
12     Click Element    id=lst-ib
13     Input Text    id=lst-ib    ${KEYWORD}
14     Press Key    id=lst-ib    \\13
15     Sleep    5
16     Close Browser
17
```

- Run with
`$pybot Keyword.robot`



RESOURCE



- Create file Resource.robot and Resource.txt

```
Resource.robot  x
1  *** Settings ***
2  Library      Selenium2Library
3  Resource     Resource.txt
4
5  *** Testcases ***
6  Resource Test Case
7      Open Browser    ${URL}    chrome
8      Click Element   id=lst-ib
9      Input Text      id=lst-ib    ${Keyword}
10     Press Key       id=lst-ib    \\13
11

Resource.txt  x
1  *** Variables ***
2  ${URL}      https://www.google.com
3  ${Keyword}  robotframework
4
```

- Run with
`$pybot Resource.robot`

