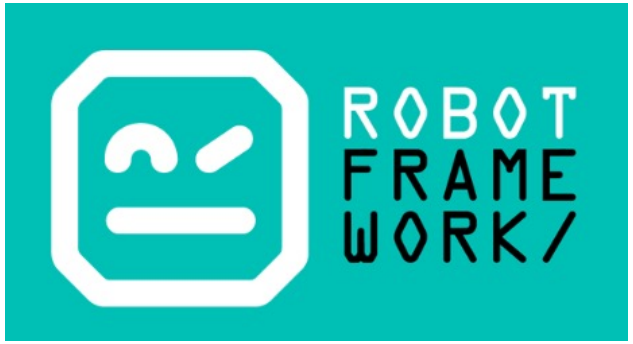


Automated Testing with Robot Framework

indythaitester@gmail.com

https://www.facebook.com/indythaitester

Assoc. Rangsit Sirirangsi

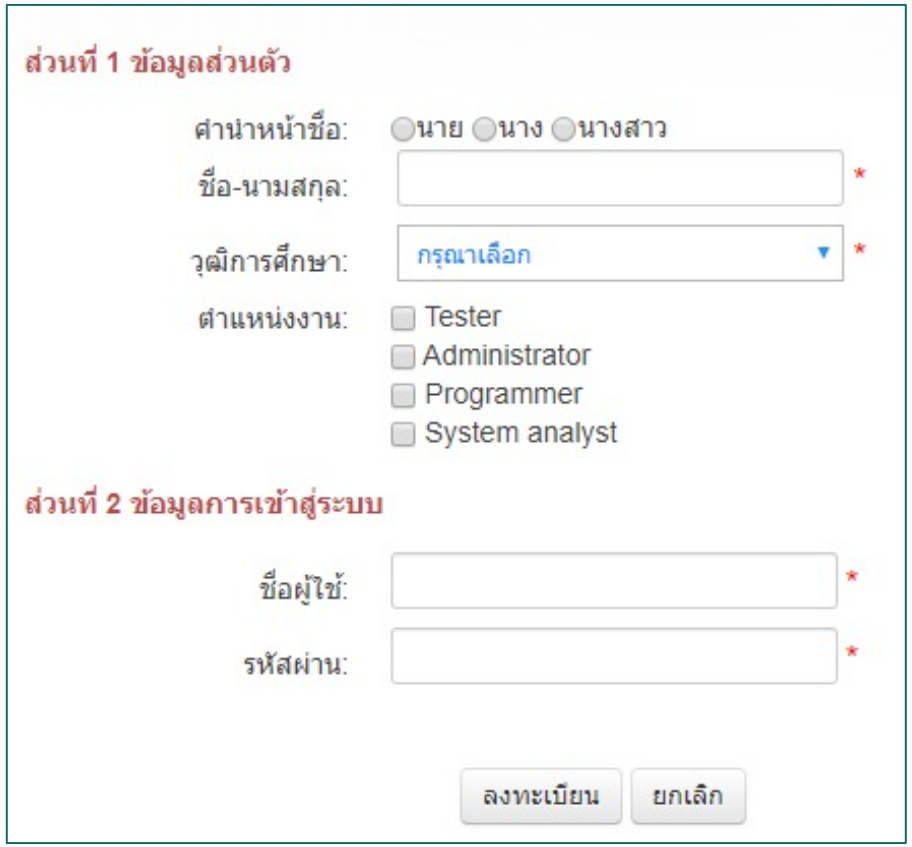
www.indythaitester.com

About Me

ROBOT FRAME WORK

- รศ. รังสิต ศิริรังษี
- Automated Testing Consult
- Automated Testing Trainer : QTP/UFT, Selenium, JMeter, Katalon Studio, Protractor, Puppeteer, Robot Framework, Playwright, Cypress
- Contact : Line ID: S.rangsit
  - Fan page : https://www.facebook.com/indythaitester/

Manual Testing



- ทดสอบการกรอกข้อมูลของ Web App ดังรูป
  - โหลดแบบฟอร์ม
  - กรอกและเลือกข้อมูลลงในฟอร์ม
  - ตรวจสอบแลพบันทึกผลลัพธ์ Pass/Fail
  - กรณี Fail ให้จับภาพหน้าจอไว้

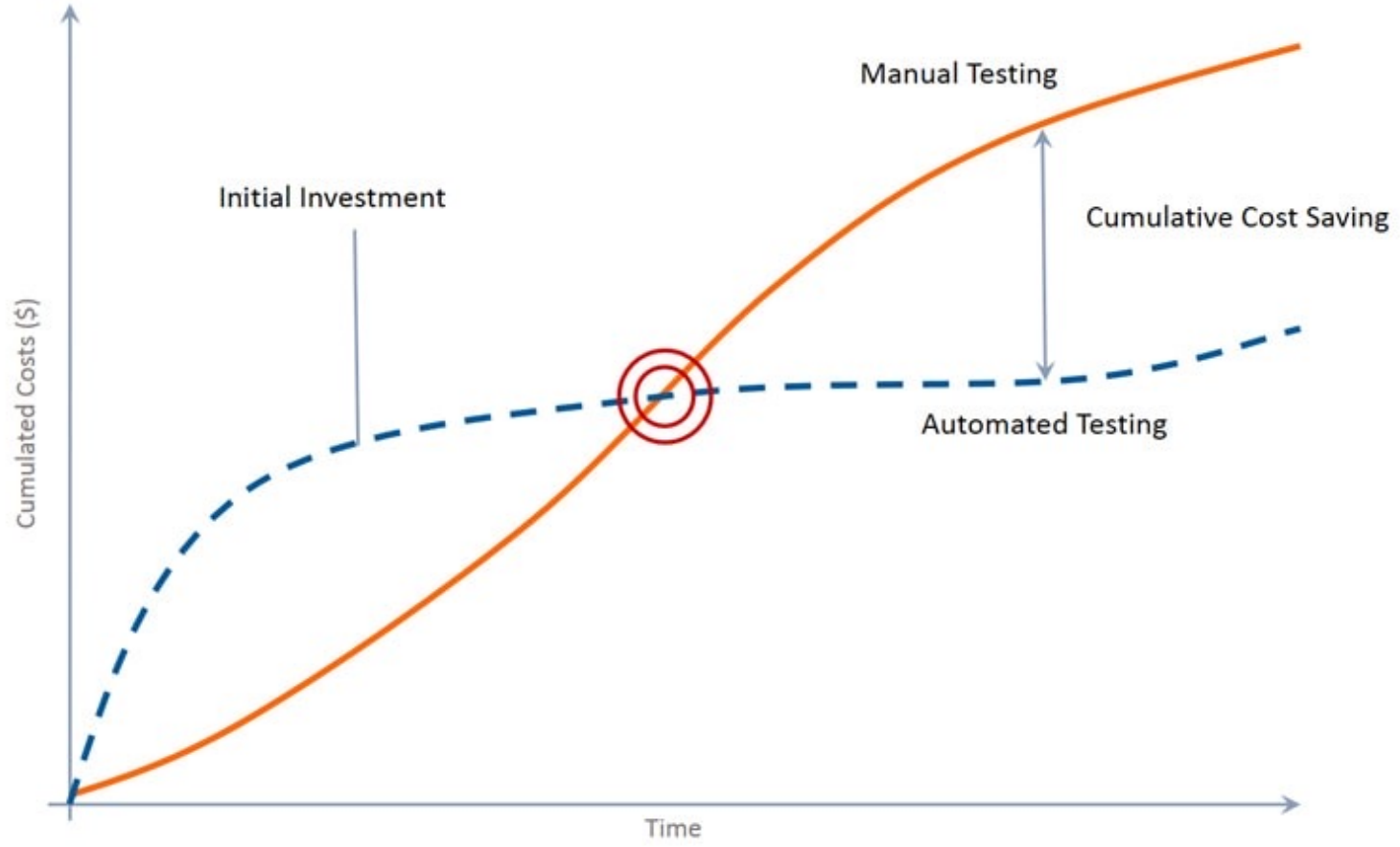
What is Automation Testing

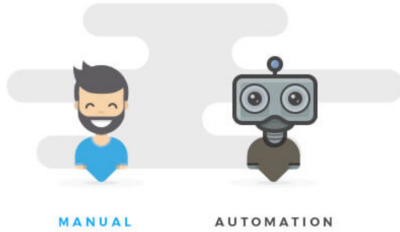
ROBOT FRAME WORK

- **Test automation** เป็นการใช้ซอฟต์แวร์เพื่อควบคุมการประมวลผลการทดสอบ โดยเปรียบเทียบผลลัพธ์ที่เกิดขึ้นจริงกับผลลัพธ์ที่คาดไว้

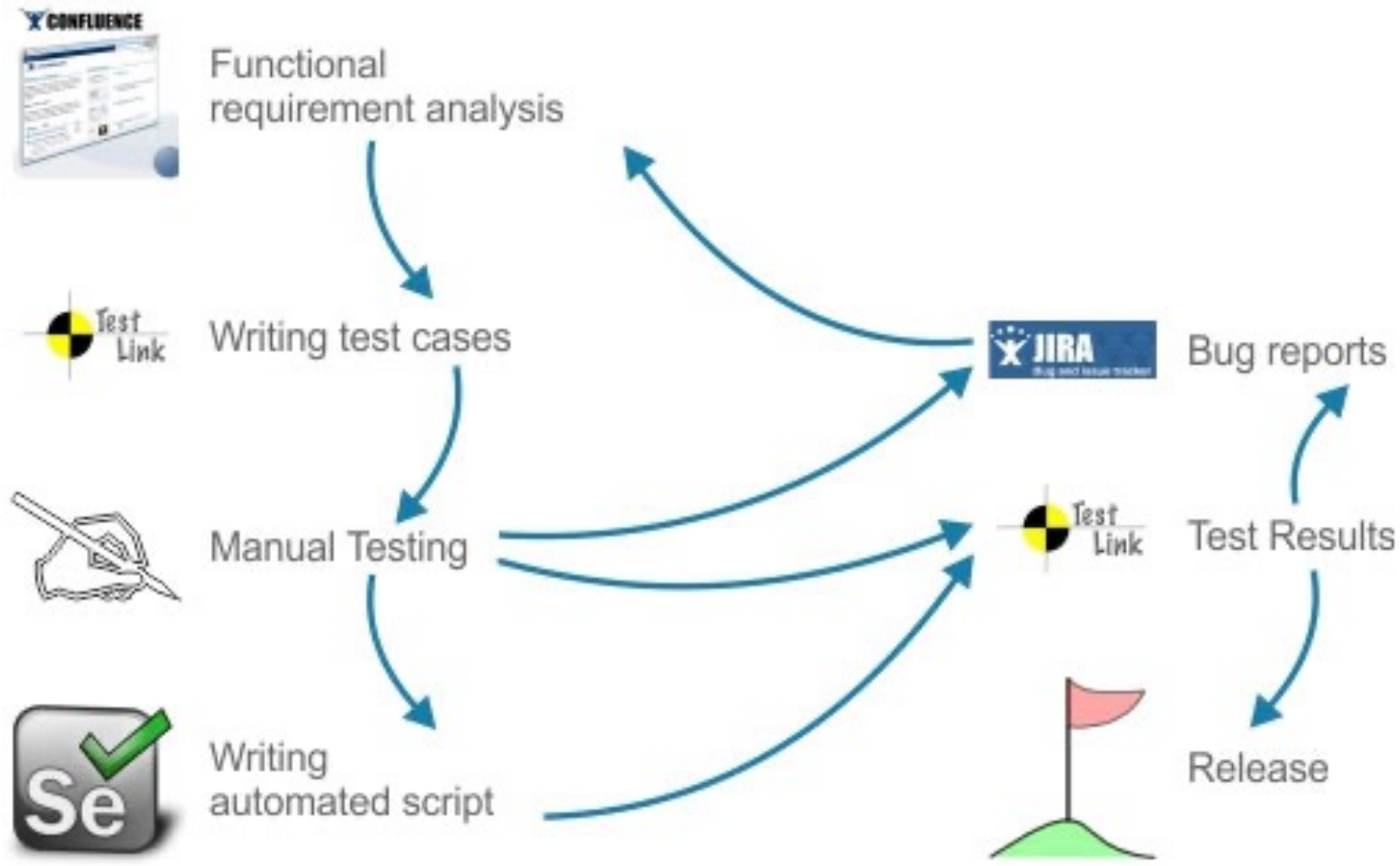
	Manual	**Automated**
Time	Faster for quick tasks	Faster in the long run
Money	Cheaper for quick tasks	Cheaper in the long run
Reliability	Less	More
Limitations	Performance tests	Visual aspects
Reusability	Less	More
Test coverage	Less	More
Human resources	More	Less
Programming knowledge	Unnecessary	Necessary
Control & debugging	Easier	Difficult
For changes	Small	Constant
Tools	Unnecessary	Necessary

Manual vs Automation Testing





Test Automation Process



Adv : Test Automation

- ข้อดีของการทดสอบแบบอัตโนมัติ
  - **ความเร็ว (Fast)** : การทดสอบแบบนี้สามารถในการรันได้เร็วกว่า
  - **ความน่าเชื่อถือ (Reliable)** : การทดสอบแบบนี้มีความน่าเชื่อถือสูง
  - **การทำซ้ำ (Repeatable)** : อาศัยการทำงานแบบซ้ำ ๆ ในการทดสอบ
  - **ความสามารถโปรแกรม (Programmable)** : สามารถโปรแกรมเพื่อช่วยดึงข้อมูลที่ถูกจัดเก็บไว้มาใช้ร่วมกับการทดสอบได้
  - **การนำกลับมาใช้ใหม่ (Reusable)** : นักทดสอบสามารถนำข้อมูลทดสอบกลับมาใช้ใหม่ได้แม้ว่าบางส่วนอาจมีการเปลี่ยนแปลงไปก็ตาม
  - **คุณภาพดีขึ้น (Better Quality)** : การทดสอบอัตโนมัติมีผลลัพธ์ที่มีความถูกต้องสูงถึง 99.9% ช่วยลดเวลาในการทดสอบตลอดจนลดจำนวนสมาชิกในทีมทดสอบลงได้

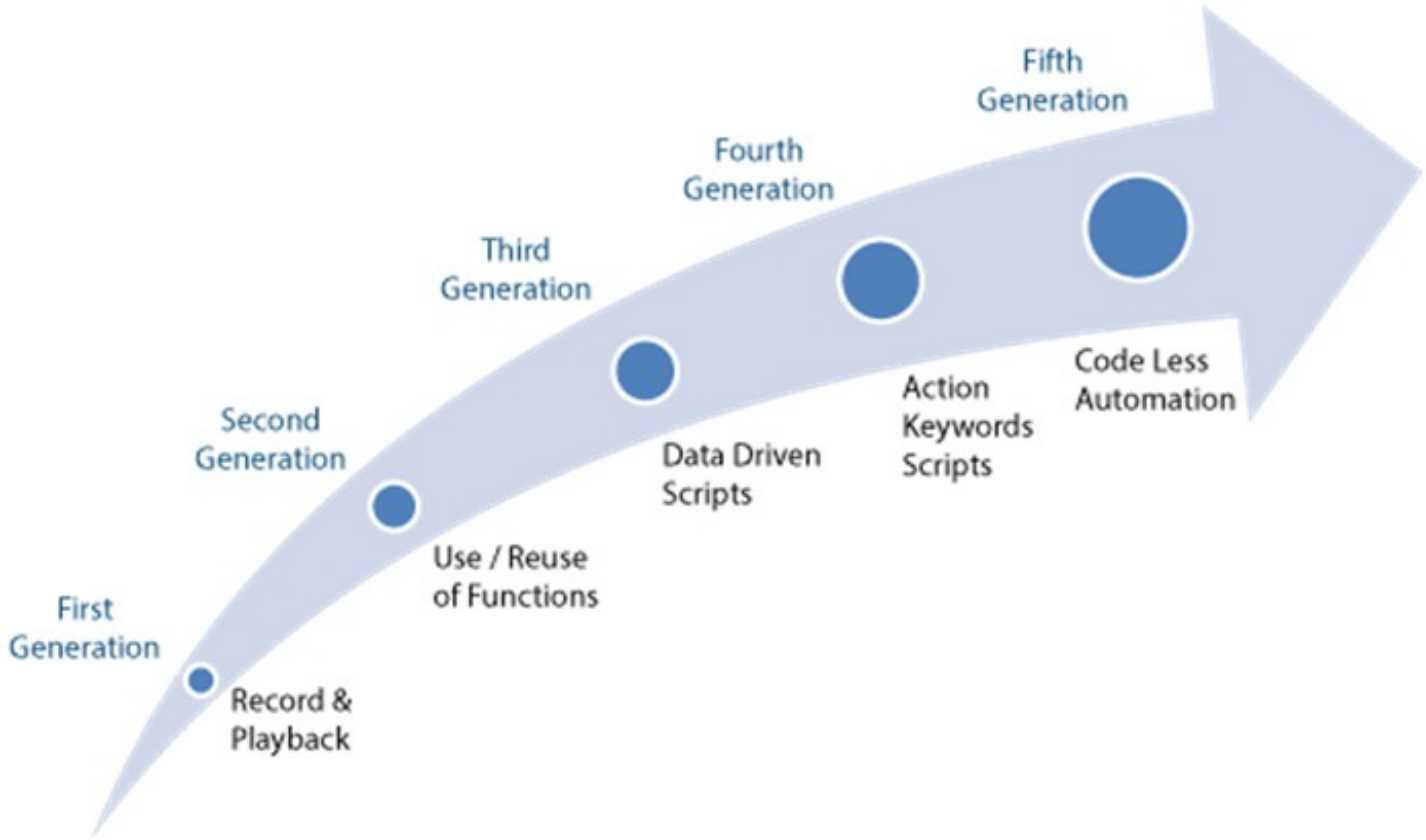
The limitations of test automation

ROBOT FRAME WORK/

- ค่าใช้จ่ายเริ่มต้นสูง ทั้งในส่วนของซอฟต์แวร์และค่าใช้จ่ายของบุคลากร
- เวลาในการสร้างสคริปต์ขึ้นอยู่กับความซับซ้อนของโปรแกรมภายใต้การทดสอบ
- ไม่สามารถนำมาใช้แทนการทดสอบด้วยมือได้ ในกรณีที่
  - การทดสอบที่รันนาน ๆ ครั้ง
  - การทดสอบซอฟต์แวร์ที่มีการประยุกต์ใช้งานด้านภาพและเสียงเป็นหลัก
  - การทดสอบที่รวมถึงการปฏิสัมพันธ์ทางกายภาพระหว่างผู้ใช้กับระบบ
- จำเป็นต้องแก้ไขสคริปต์ทุกครั้งที่มีการเปลี่ยนสภาพแวดล้อมในการทดสอบ
- เครื่องมือทดสอบอัตโนมัติทำงานตามที่ถูกกำหนดไว้เท่านั้น การปรับปรุงแก้ไขใด ๆ ขึ้นอยู่กับความสามารถของนักทดสอบเป็นหลัก

Automation Frameworks Evolution

- Automation Framework ที่ใช้ในการทดสอบปัจจุบันจะประกอบไปด้วยคุณสมบัติดังต่อไปนี้



Record & Playback

ROBOT FRAME WORK

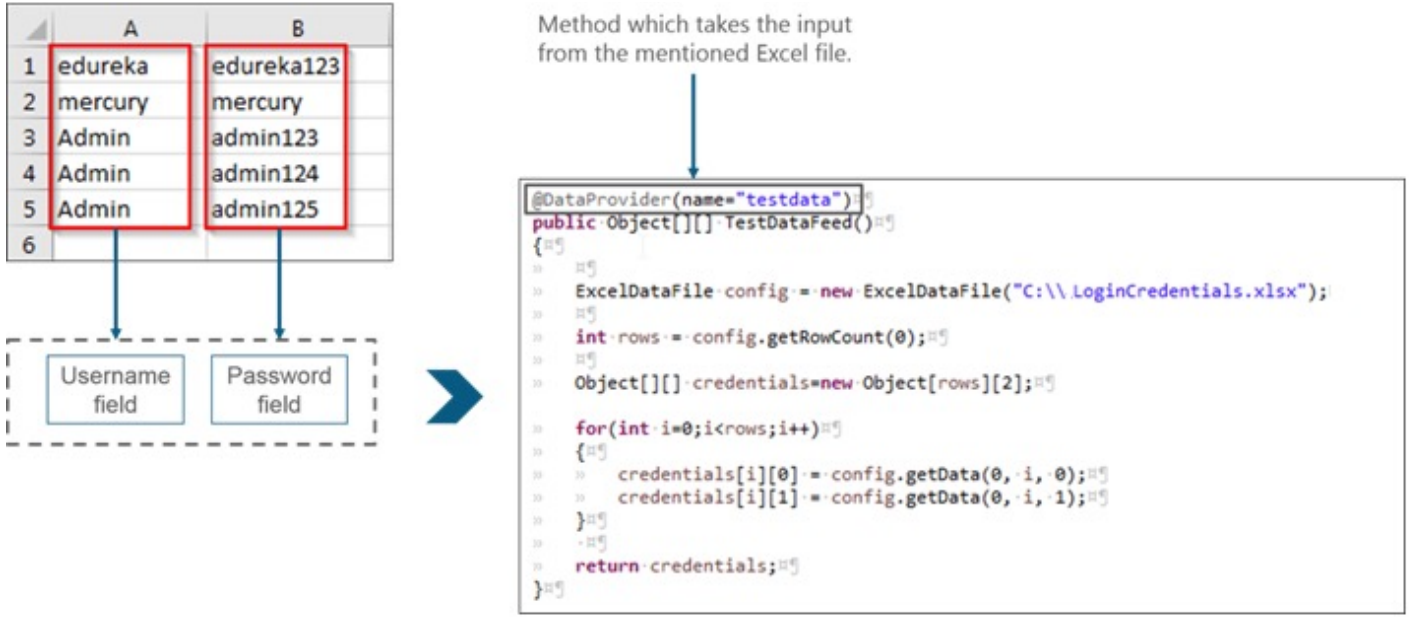
- บันทึก (record) ค่าอินพุตจากผู้ใช้ และการตอบสนองจากระบบให้อยู่ในรูปของ Test Script ที่สามารถนำกลับมาเล่นย้อนกลับ (play back) ได้
- ข้อดี
  - สะดวกรวดเร็วในการทำงาน ค่าใช้จ่ายในการดำเนินการต่ำ
- ข้อเสีย
  - สคริปต์มีลักษณะเป็น hard coded จำเป็นต้องอัพเดททุกครั้งที่มีการเปลี่ยนแปลงข้อมูล
  - การเปลี่ยนแปลงสคริปต์อาจต้องมีการบันทึกใหม่เสมอ ส่งผลทำให้ค่าใช้จ่ายสูงขึ้น
  - ในระหว่างการบันทึกหากมีข้อผิดพลาดใด ๆ เกิดขึ้นจะต้องบันทึกขั้นตอนทั้งหมดใหม่ทุกครั้ง
  - ไม่รองรับการทำงานที่มีความซับซ้อนสูง

ROBOT FRAME WORK

Limitation…

- สคริปต์มีลักษณะเป็น hard coded นักทดสอบจำเป็นต้องอัพเดทสคริปต์ทุกครั้งที่มีการเปลี่ยนแปลงข้อมูล
- สคริปต์เหล่านี้มีการกลไกในการจัดการกับความผิดพลาดที่เกิดขึ้นน้อยมาก บ่อยครั้งจะล้มเหลวเนื่องมาจากการเล่นย้อนกลับสำหรับเหตุการณ์ต่าง ๆ ที่เกิดขึ้น เช่น วินโดวส์ป๊อปอัพ หรือข้อความเตือนแบบต่าง ๆ เป็นต้น
- การเปลี่ยนแปลงสคริปต์อาจต้องมีการบันทึกใหม่เสมอ ดังนั้นจึงส่งผลทำให้ค่าใช้จ่ายในการบำรุงรักษาสคริปต์สูงขึ้น
- ในระหว่างการบันทึกหากมีข้อผิดพลาดใด ๆ เกิดขึ้นจะต้องบันทึกขั้นตอนทั้งหมดใหม่ทุกครั้ง
- สคริปต์สำหรับ Record/Playback จะทำงานได้ดีเฉพาะในกรณีที่โปรแกรมที่ถูกทดสอบได้ผ่านการทำงานมาแล้วนั่นเอง
- ไม่รองรับการทำงานที่มีความซับซ้อนสูง

Data Driven Framework



- เป็นการแยกส่วน Test Data ออกจาก Test Script โดยจัดเก็บไว้ในแหล่งข้อมูลภายนอกเป็นหลัก
- ผู้ใช้สามารถแทนข้อมูลที่เป็น Hard Coded ลงในตัวแปรที่อยู่ภายใน Test Script เพื่ออ่านค่าจากแหล่งข้อมูลที่อยู่ภายนอกได้ ดังนั้นจึงช่วยลดจำนวนสคริปต์ และง่ายต่อการบำรุงรักษา

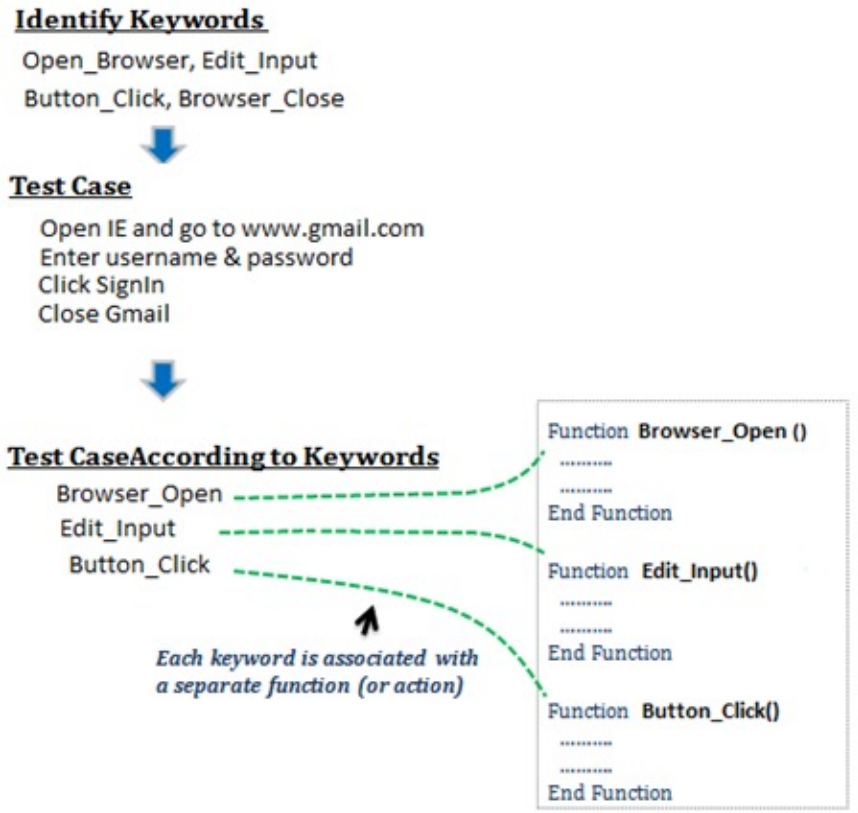
Limitations: Data-Driven Methodology

ROBOT FRAME WORK

- สคริปต์ทดสอบต้องเสร็จสมบูรณ์แล้วก่อนเริ่มต้นการทำงาน
- ต้องการความชำนาญในการสร้างสคริปต์ โดยขึ้นอยู่กับเครื่องมือที่ถูกเลือกใช้
- การทดสอบจะขึ้นอยู่กับ GUI เป็นหลัก ดังนั้นเมื่อ GUI มีการเปลี่ยนแปลง สคริปต์ทดสอบต้องมีการเปลี่ยนแปลงตามไปด้วย
- ต้องการพื้นที่จัดเก็บข้อมูลมากขึ้นสำหรับแต่ละ Test Case
- test script และ data files จำเป็นต้องมีการบำรุงรักษาอย่างมาก

Keyword-Driven Framework

- เป็นส่วนที่พัฒนาขึ้นเพื่อลดข้อจำกัดของ Data Driven Framework
- สคริปต์ทดสอบพัฒนาขึ้นจากทักษะทางโปรแกรมเป็นหลัก ส่งผลให้การบำรุงรักษาทำได้ง่าย สามารถนำกลับมาใช้ใหม่ได้
  - **คีย์เวิร์ด (Keywords)** ในรูปของคำศัพท์พื้นฐานแทนการทำงานในการติดต่อกับผู้ใช้ เช่น "click", "enter", "select"
  - **การกระทำ (Actions)** ที่อยู่ในรูปของโมดูลหรือฟังก์ชัน ซึ่งเป็นผลมาจากการเรียกใช้คีย์เวิร์ดที่กำหนดไว้



Keyword Driven

ROBOT FRAME WORK

- ข้อดีของเทคนิคการทดสอบแบบ Keyword Driven มีดังต่อไปนี้
  - สามารถสร้างสคริปต์ทดสอบไปพร้อม ๆ กับการพัฒนาโค้ด
  - นักทดสอบต้องเรียนรู้เฉพาะคีย์เวิร์ดที่ต้องการเท่านั้น ไม่จำเป็นต้องมีประสบการณ์การโปรแกรม
  - นักทดสอบสามารถนำคีย์เวิร์ด และ test cases ที่ได้ออกแบบไว้แล้วกลับมาใช้ใหม่ได้
  - การแก้ไขเพิ่มเติมโค้ดในส่วนที่เกี่ยวข้องกับ test cases น้อย
  - การบำรุงรักษาทำได้โดยง่าย การเปลี่ยนแปลงฟังก์ชันการทำงานต้องการเท่านั้น
- ข้อเสียของเทคนิคการทดสอบแบบ Keyword Driven มีดังต่อไปนี้
  - สคริปต์ที่ใช้ทดสอบมีลักษณะเฉพาะสูง จำเป็นต้องใช้นักทดสอบที่มีความเชี่ยวชาญในเครื่องมือสร้างสคริปต์ในลักษณะดังกล่าวสูงตามไปด้วย
  - นักทดสอบต้องมีความเชี่ยวชาญในเครื่องมือที่ใช้ในกรณีที่ต้องการแก้ไขเพิ่มเติม Keyword
  - แต่ละเครื่องมือใช้โปรแกรมภาษาแตกต่างกัน เปลี่ยนเครื่องมือต้องเรียนรู้ใหม่เสมอ

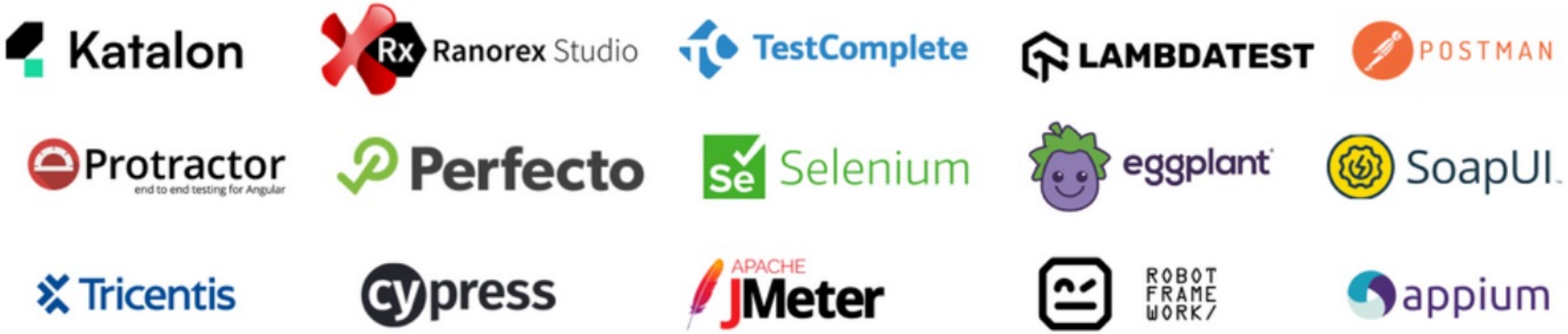
Codeless Automation

ROBOT FRAME WORK

- เป็นการทดสอบฟังก์ชันการทำงานอัตโนมัติที่มีการเขียนโค้ดน้อยที่สุด หรือ Model Based Testing
  - ไม่จำเป็นต้องสร้าง Automation Frameworks แบบซับซ้อนขึ้นมาใหม่
  - Manual Tester/QA สามารถเข้าฝึกอบรมการใช้งานได้โดยง่าย
  - มีค่าใช้จ่ายเฉพาะเครื่องมือ ไม่รวมค่าใช้จ่ายในส่วนของนักทดสอบ
  - สนับสนุนวิธีการพัฒนาแบบใหม่ ๆ อาทิ Agile
  - สะดวกต่อการตั้งค่าโครงแบบ (Configuration)
  - สคริปต์สามารถทำความเข้าใจได้โดยง่าย

Automated Test Scripts

- เป็นโปรแกรมที่ถูกพัฒนาขึ้นตามเกณฑ์มาตรฐานเดียวกับการพัฒนาซอฟต์แวร์ เพื่อให้ได้ผลลัพธ์ในการนำไปใช้ได้เป็นอย่างดี จำเป็นต้องมีการฝึกอบรมผู้ใช้ เช่นเดียวกับโปรแกรมเมอร์
- Test scripts ถูกเขียนขึ้นตามรูปแบบและวิธีการที่กำหนดไว้ในเครื่องมือทดสอบโดยเฉพาะ เช่น Unified Functional Testing (QTP), Rational Software, Selenium ใช้โปรแกรมภาษาที่แตกต่างกันไป เช่น Basic, Java, Python หรือ JavaScript เป็นต้น



What is Robot Framework?

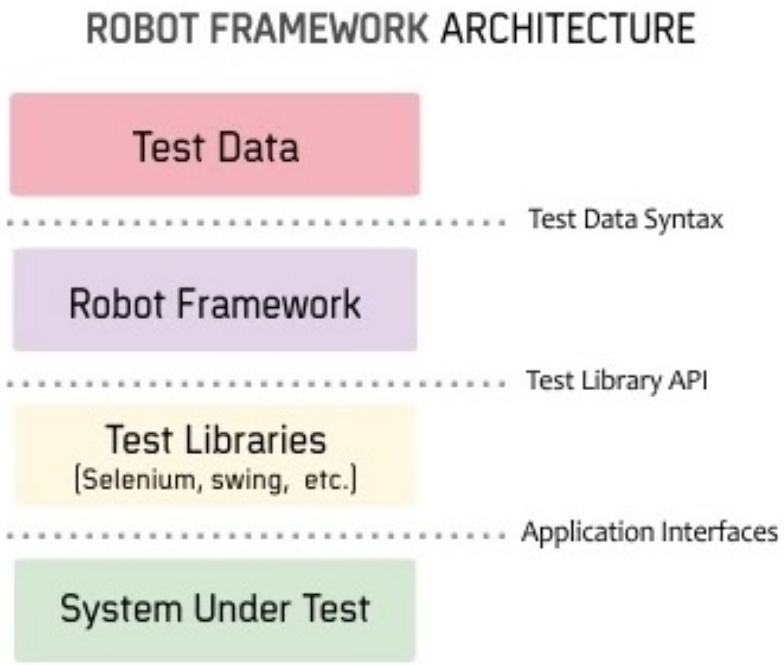
- โดย Pekka Klärck ได้แนวคิดจากวิทยานิพนธ์ระดับปริญญาโท "Data-Driven and Keyword-Driven Test Automation Frameworks“ ที่ Helsinki University Of Technology ในปี 2004
- การพัฒนา Robot Framework เวอร์ชั่นแรกเริ่มขึ้นที่ Nokia Networks เพื่อใช้เป็นเครื่องมือทดสอบอัตโนมัติ ในปี 2005 โดยใช้โปรแกรมภาษา Python
- จากนั้นมีการปรับแก้และ Robot Framework 2.0 ได้ถูกเผยแพร่ออกมาภายใต้ Apache License เป็นซอฟต์แวร์รหัสเปิดในปี 2008
- ใช้ Selenium เป็นพื้นฐานในการทำงานผ่านหลักการที่เรียกว่า Keyword-Driven Frameworks
- ปัจจุบัน Robot Framework พัฒนามาถึงเวอร์ชั่น 6.1

Why Robot Framework?

ROBOT FRAME WORK

- เป็นซอฟต์แวร์รหัสเปิดที่มีข้อดีต่าง ๆ อาทิ ค่าใช้จ่าย มี community สนับสนุน
- เป็นเครื่องมือที่เป็นอิสระจาก platform และ application สำหรับการทดสอบ
- ใช้การสร้าง Keyword ที่มีรูปแบบง่ายต่อการใช้ตลอดจนการทำความเข้าใจ ดังนั้นจึงสะดวกต่อการนำกลับไปใช้ใหม่ได้
- ดังนั้นจึงเป็นเครื่องมือทดสอบอัตโนมัติที่เหมาะสมกับผู้ที่ไม่มีประสบการณ์ด้านการโปรแกรม
- Library จำนวนมากที่พร้อมนำมาใช้ได้กับ Robot Framework อาทิ Android, database, etc. เพื่อใช้ทดสอบ application ในทุก ๆ รูปแบบ
- นอกจากนี้ยังสามารถในการใช้ร่วมกับโปรแกรมภาษาต่าง ๆ อาทิ Python, Java

Robot Framework Architecture



- เป็น Abstraction Layer ที่พัฒนาบน Test Library ต่าง ๆ
- ดังนั้นจึงสามารถใช้ทดสอบได้ทั้ง web, desktop, mobile แอปพลิเคชัน รวมถึง RESTful และ SOAP-based services
- ส่วน Test Case สำหรับ Mobile สามารถทดสอบได้ทั้ง Android และ iOS app ส่วนการทดสอบบนเว็บสามารถรันได้ทั้ง Chrome, Firefox, Edges, IE และ Safari

Test Data Sections

- ใน Robot Framework ข้อมูลที่ใช้ในการทดสอบจะถูกแบ่งออกเป็น Section จำนวน 4 ส่วนหลัก ๆ ในรูปของตารางที่ประกอบไปด้วยสัญลักษณ์ *** พร้อมข้อความที่กำหนดไว้ดังต่อไปนี้

Section	Description
*** Settings ***	ใช้สำหรับระบุส่วนประกอบที่ต้องการใช้ร่วมกับการทดสอบ อาทิ libraries, resource ไฟล์, variable ไฟล์ เป็นต้น
*** Variables ***	ใช้สำหรับประกาศตัวแปรที่ต้องการใช้งานภายในสคริปต์
*** Test Cases ***	ใช้สำหรับการสร้าง test cases จากภาษาง่าย ๆ โดยผู้ที่ไม่มีความรู้ทางด้านเทคนิคสามารถเข้าใจได้
*** Keywords ***	ใช้สร้างรายละเอียดของ Test Case จากคีย์เวิร์ดที่มีอยู่แล้วหรือผู้ใช้สร้างขึ้นมาใหม่ เพื่อใช้ในการทดสอบ

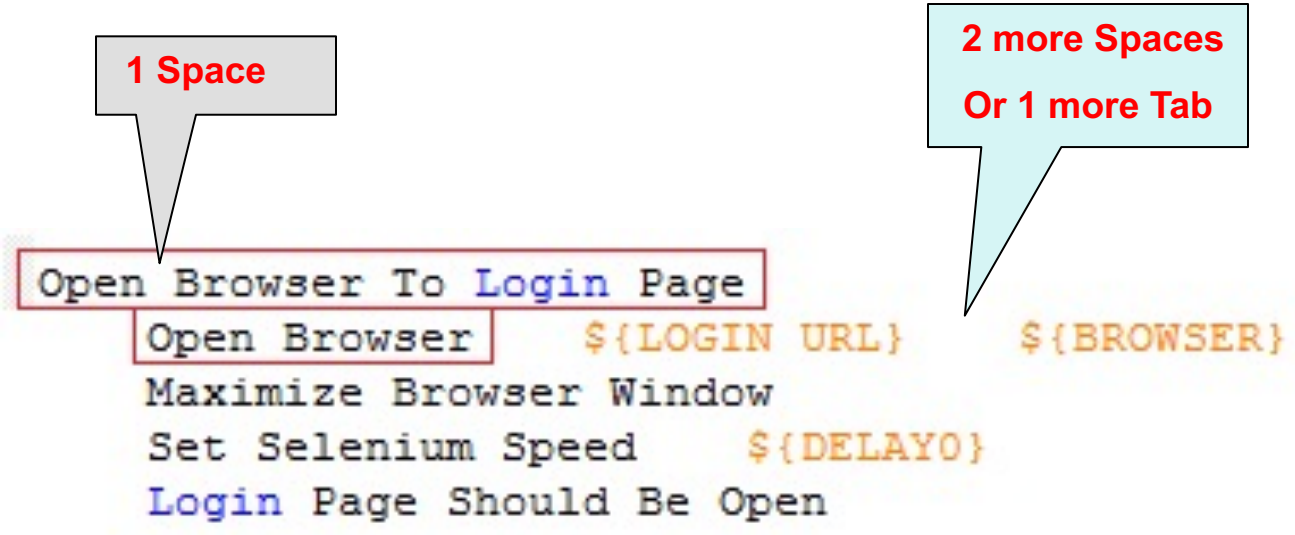
Browser : Basic Keywords

- คีย์เวิร์ดใน Robot Framework ที่เกี่ยวข้องกับ Browser ที่ใช้มีการเรียกใช้งานบ่อย ๆ ได้แก่

Method	Purpose
Open Browser	เปิดบราวเซอร์ตาม Url และชนิดบราวเซอร์ที่ถูกระบุ
Close Browser	ปิดการทำงานของบราวเซอร์ปัจจุบัน
Maximize Browser Window	เปิดหน้าต่างบราวเซอร์เต็มหน้าจอ
Get Location	คืนค่า URL ของบราวเซอร์ปัจจุบัน
Get Title	คืนค่า title ของเพจปัจจุบัน
Sleep	หยุดรอการประมวลผลการทดสอบตามเวลาที่ถูกระบุ

Space separated format

- การตรวจสอบคีย์เวิร์ดหรือคำสั่งที่ใช้ในการทำงานของ Robot Framework จะทำงานโดยการแยกคำสั่งโดยใช้ space อย่างน้อยสอง spaces
- โดยแต่ละคีย์เวิร์ดตลอดจน Argument ต้องมีจำนวนช่องว่างตั้งแต่สองหรือมากกว่าขึ้นไปเสมอ ส่วนในกรณีของ Tab ต้องใช้ตั้งแต่หนึ่งหรือมากกว่าขึ้นไปเช่นเดียวกัน



USB warning in Window 10

- การประมวลผล Robot Framework ร่วมกับ Window 10 และบราวเซอร์ Chrome อาจมีข้อความแจ้งเตือนการทำงานผิดพลาดร่วมกับ USB ได้ เช่น

```
DevTools listening on ws://127.0.0.1:57090/devtools/browser/8203a702-f81a-48a6-8174-6a4dcc319a1a
[5888:5128:0529/094809.966:ERROR:device_event_log_impl.cc(214)] [09:48:09.961] USB: usb_device_handle_win.cc:
ode connection: A device attached to the system is not functioning. (0x1F)
```

- ในทางปฏิบัติแล้วข้อความดังกล่าวไม่สงผลกระทบต่อการทดสอบการทำงานด้วย Robot Framework แต่อย่างใด
- แต่อย่างไรก็ตามหากต้องการแก้ไขเพื่อไม่ให้ระบบแสดงข้อความดังกล่าว ผู้ใช้สามารถทำได้โดยการกำหนด option ใน browser ดังนี้

```
open browser    ${URL}  Chrome    options=add_experimental_option('excludeSwitches', ['enable-logging'])
```

Robot Framework : Output file

- การรัน Test Case ใน Robot Framework จะสร้างไฟล์ผลลัพธ์จากการทดสอบจำนวน 3 ไฟล์ได้แก่
  - Output.xml เป็นไฟล์ที่ประกอบด้วยรายละเอียดผลลัพธ์การทดสอบที่อยู่ในรูป XML format
  - Log.html ใช้สำหรับนำเสนอ Test Statistic รวมถึง Test Execution Log ที่ใช้นำเสนอรายละเอียดการประมวลผลของแต่ละคีย์เวิร์ดที่ใช้ในการทดสอบ
  - Report.html ใช้สำหรับนำเสนอรายละเอียดสรุปผลการทดสอบต่าง ๆ ที่จำเป็นต่อการนำไปใช้ในการวิเคราะห์ต่อไป

```
------------------------------------------------------------------------------
Browser                                                               | PASS |
1 test, 1 passed, 0 failed
==============================================================================
Output:  C:\Record\LAB\temp1\results\output.xml
Log:     C:\Record\LAB\temp1\results\log.html
Report:  C:\Record\LAB\temp1\results\report.html
```

Log file

- ประกอบด้วยรายละเอียดการประมวลผล Test Case ในรูปของ HTML format โดยนำเสนอตามลำดับชั้นจาก test suite, test case และรายละเอียดของ keyword

REPORT

Generated
20211011 22:19:26 UTC+07:00
15 hours 40 minutes ago

Browser Log

Test Statistics

Total Statistics	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
All Tests	1	1	0	0	00:00:09	

Statistics by Tag	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
No Tags						

Statistics by Suite	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
Browser	1	1	0	0	00:00:09	

Test Execution Log

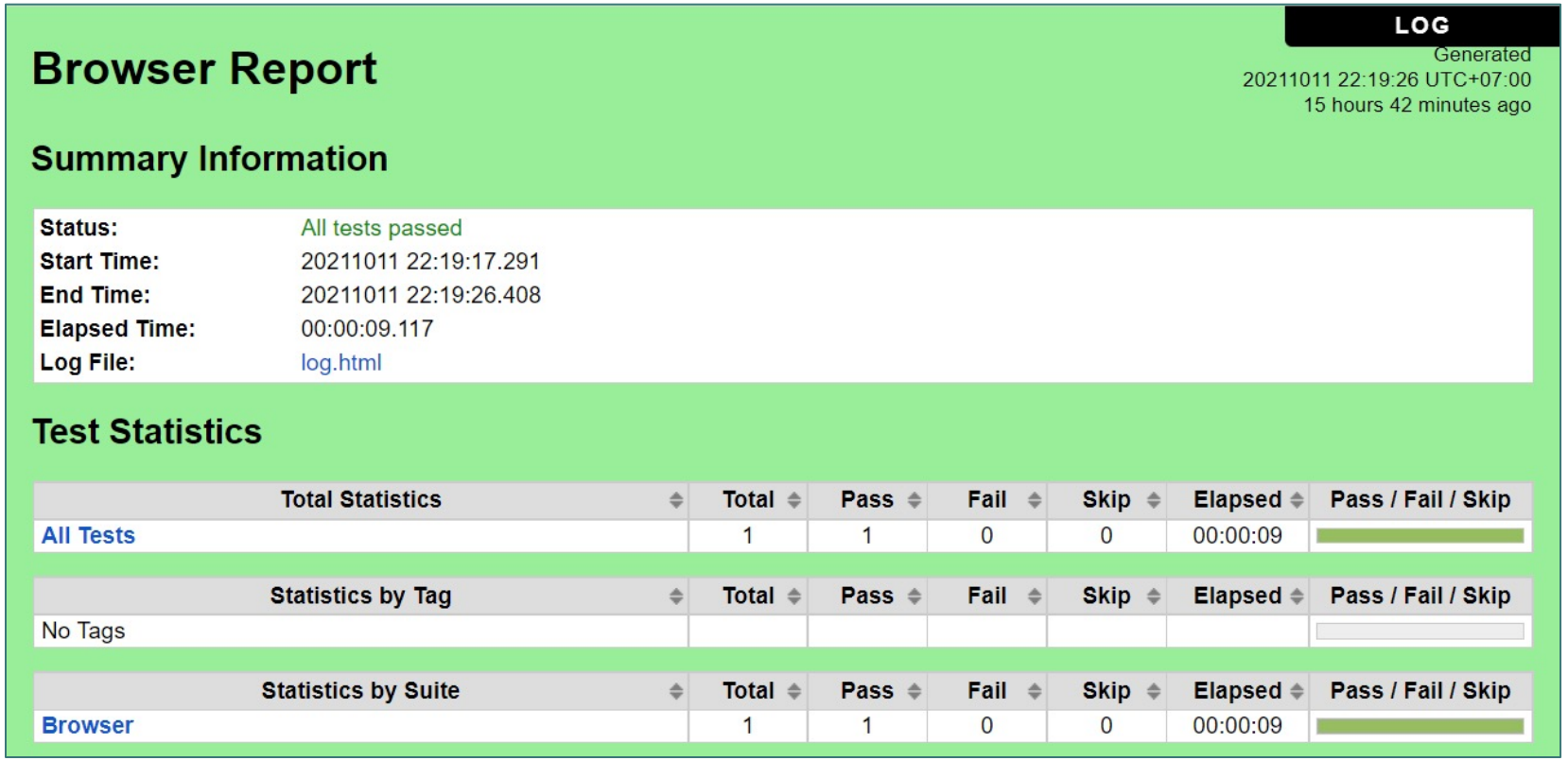
SUITE Browser 00:00:09.117

Full Name: Browser
Source: C:\Record\LAB\temp1\TestCases\browser.robot
Start / End / Elapsed: 20211011 22:19:17.291 / 20211011 22:19:26.408 / 00:00:09.117
Status: 1 test total, 1 passed, 0 failed, 0 skipped

TEST First Test Case 00:00:08.937

Report file

- ประกอบไปด้วยภาพรวมของการประมวลผลทดสอบในรูป HTML format จากการประมวลผล test suites และ test cases สามารถเชื่อมโยงข้อมูลไปยัง Log ไฟล์ได้ในกรณีที่ต้องการข้อมูลเพิ่มเติม



LOG

Browser Report

Generated
20211011 22:19:26 UTC+07:00
15 hours 42 minutes ago

Summary Information

Status:	All tests passed
Start Time:	20211011 22:19:17.291
End Time:	20211011 22:19:26.408
Elapsed Time:	00:00:09.117
Log File:	log.html

Test Statistics

Total Statistics	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
All Tests	1	1	0	0	00:00:09	

Statistics by Tag	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
No Tags						

Statistics by Suite	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
Browser	1	1	0	0	00:00:09	

Close Browser Automatically

ROBOT FRAME WORK

- เนื่องจาก robotframework 6.0 และ robotframework-seleniumlibrary 6.0 มีการสั่งปิดบราวเซอร์โดยอัตโนมัติหลังการทดสอบเสร็จสิ้น แม้ว่าจะไม่ใช้คำสั่ง Close Browser ก็ตาม
- ในกรณีที่ต้องการเปิดบราวเซอร์ค้างไว้หลังการทดสอบ ผู้ใช้สามารถกำหนด options ด้วยคำสั่งดังต่อไปนี้

```
open browser ${URL} ${BROWSER}
... options=add_experimental_option("detach", True)
maximize browser window
```