

ศึกษาการใช้ระบบอัตโนมัติทดสอบซอฟต์แวร์เว็บแอปพลิเคชัน : กรณีศึกษา
บริษัทด้านเทคโนโลยีสารสนเทศและการสื่อสาร

เศรษฐพงษ์ อิมสุวรรณ

สารนิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาการจัดการทางวิศวกรรม วิทยาลัยนวัตกรรมการด้านเทคโนโลยีและ
วิศวกรรมศาสตร์ มหาวิทยาลัยธุรกิจบัณฑิตย์

พ.ศ. 2561

**The Study of Using Automation Testing Test
the Software Web Application : Case Study Use Information and
Communication Technology Company**

Setthapong Imsuwan

**A Thematic paper Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering
Department Engineering Management.
Dhurakij Pundit University**

2018



ใบรับรองสารนิพนธ์

วิทยาลัยนวัตกรรมการด้านเทคโนโลยีและวิศวกรรมศาสตร์ มหาวิทยาลัยธุรกิจบัณฑิตย์

ปริญญา วิศวกรรมศาสตรมหาบัณฑิต

หัวข้อสารนิพนธ์ ศึกษาการใช้ระบบอัตโนมัติทดสอบซอฟต์แวร์เว็บแอปพลิเคชัน : กรณีศึกษา
บริษัทด้านเทคโนโลยีสารสนเทศและการสื่อสาร

เสนอโดย เศรษฐพงษ์ อิ่มสุวรรณ

สาขาวิชา การจัดการทางวิศวกรรม

อาจารย์ที่ปรึกษาสารนิพนธ์ ผู้ช่วยศาสตราจารย์ ดร.ศุภรัชชัย วรรัตน์

ได้พิจารณาเห็นชอบโดยคณะกรรมการสอบสารนิพนธ์แล้ว



.....ประธานกรรมการ
(อาจารย์ ดร.ประศาสน์ จันทราทิพย์)



.....กรรมการและอาจารย์ที่ปรึกษาสารนิพนธ์
(ผู้ช่วยศาสตราจารย์ ดร.ศุภรัชชัย วรรัตน์)



.....กรรมการ
(อาจารย์ ดร.ณัฐพัชร์ อารีรัชกุลกานต์)

วิทยาลัยนวัตกรรมการด้านเทคโนโลยีและวิศวกรรมศาสตร์ รับรองแล้ว



.....
(ผู้ช่วยศาสตราจารย์ ดร.ณรงค์เดช กิรติพรานนท์)

คณบดีวิทยาลัยนวัตกรรมการด้านเทคโนโลยีและวิศวกรรมศาสตร์

วันที่ 17 เดือน กรกฎาคม พ.ศ. 2567

หัวข้อสารนิพนธ์	การใช้ระบบอัตโนมัติทดสอบซอฟต์แวร์เว็บแอปพลิเคชัน : กรณีศึกษาบริษัทด้านเทคโนโลยีสารสนเทศและการสื่อสาร
ชื่อผู้เขียน	เศรษฐพงษ์ อัมสุวรรณ
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร.ศุภรัชชัย วรรณ
สาขาวิชา	การจัดการทางวิศวกรรม
ปีการศึกษา	2560

บทคัดย่อ

งานวิจัยเรื่องนี้มีวัตถุประสงค์เพื่อตรวจสอบการทำงานของซอฟต์แวร์ให้ถูกต้องตามข้อกำหนดของผู้ใช้งานเป็นขั้นตอนที่สำคัญและจำเป็นต้องทดสอบ ผู้วิจัยดำเนินการใช้ Automation Testing เข้ามาทดสอบซอฟต์แวร์เปรียบเทียบกับทดสอบแบบ Manual Testing เพื่อศึกษาและพัฒนากระบวนการ Automation Testing มาประยุกต์ใช้ในการทดสอบซอฟต์แวร์เว็บแอปพลิเคชันทำให้ลดระยะเวลาในการทดสอบ ผลที่ตามมาคือ สามารถลดแรงงานคน ค่าใช้จ่าย ความผิดพลาดในการบันทึกข้อมูล ผู้วิจัยได้ทำการศึกษาโดยใช้ผู้ทดสอบ ทั้งหมด 9 คนภายในทีม ทำการทดสอบซอฟต์แวร์ด้วยวิธี Manual Testing 20 Test case จำนวน 5 ครั้ง แล้วหาค่าเฉลี่ยของเวลาที่ใช้ในการทดสอบ และบันทึกเวลาที่ใช้ในการทดสอบแบบ Automation Testing 20 Test case เดียวกันจำนวน 5 ครั้ง พบว่าเวลาเฉลี่ยที่ใช้ในการทดสอบแบบ Automation Testing มีเวลาน้อยกว่าเวลาที่ใช้ในการทดสอบแบบ Manual Testing อย่างมีนัยสำคัญทุกครั้ง ค่าเฉลี่ยของเวลาที่ใช้ในการทดสอบแบบ Automation Testing คิดเป็น ร้อยละ 19.26 ของเวลาที่ใช้แบบ Manual Testing ผลการศึกษานี้เป็นไปตามวัตถุประสงค์ของการนำ Automation Testing ทำให้ลดระยะเวลาในการทดสอบ ซอฟต์แวร์เว็บแอปพลิเคชัน และยังได้ผลการทดสอบที่ Manual Testing ทำไม่ได้ได้แก่ สามารถสั่งงานแบบ Schedule run ได้ รายงานผลการทดสอบ Test Result Log ได้รวดเร็วและใช้เป็นเครื่องมือติดตาม Test case ว่าถูกต้องครบถ้วนตามข้อกำหนด

Thematic Paper Title The Study of Using Automation Testing Test the Software Web
Application : Case Study Use Information and Communication
Technology Company
Author Setthapong Imsuwan
Thematic Paper Advisor Assistant Professor Dr. SuparatchaiVorarat
Department Engineering Management
Academic Year 2016

ABSTRACT

The objective of this research is to study the using Automation Testing test the operation of the software according to user requirements. It is an important step and is required to investigate the accomplishment of software. Researcher use Automation Testing to examine software web application compare to Manual Testing. Using Automation Testing to prove that time spent in testing software web application can reduce, with more advantages such as human labor costs, data recording errors. The study was conducted by using nine manual testers test twenty cases compare to automation testing The manual testing process takes a significant amount of time as compared to automation testing. In the experiment, automation testing took 19.26 percent of the time allocated for manual testing on the average. The results of the study were based on the purpose of Automation Testing, which reduced the testing time. The more test results are not available in the Manual Testing, such as : can be run on schedule run and report test results quickly. Test result Log is a test case that is accurate to meet the requirements.

กิตติกรรมประกาศ

ผู้วิจัยได้ศึกษาและจัดทำสารนิพนธ์ เรื่องการใช้ Automation Testing ทดสอบการทำงานของซอฟต์แวร์เว็บแอปพลิเคชันทำให้ลดเวลาที่ใช้ในการทดสอบ สารนิพนธ์ฉบับนี้สำเร็จได้อย่างสมบูรณ์ ด้วยการติดตามให้คำแนะนำของอาจารย์ที่ปรึกษา ผู้ช่วยศาสตราจารย์ ดร.ศุภรัชชัย วรรัตน์ ตรวจสอบความถูกต้อง อธิบายความรู้ ให้คำปรึกษาเสนอข้อคิดเห็นงานที่ควรเพิ่มเติมหรือแก้ไขซึ่งผู้ทำสารนิพนธ์ได้นำความรู้ใช้ปรับปรุงงานวิจัยให้สมบูรณ์ และใช้เวลาแก่ผู้ทำสารนิพนธ์ทุกครั้งขอคำปรึกษา

ขอขอบพระคุณอาจารย์ที่ปรึกษาร่วม อาจารย์ณัฐพัชร อารีรัชกุลกานต์ ให้คำแนะนำเพิ่มเติมผู้ทำสารนิพนธ์ได้ใช้คำแนะนำและความรู้มาประยุกต์ใช้กับทำสารนิพนธ์ฉบับนี้ ขอขอบคุณคุณ เสาวลักษณ์ คู่คำที่ได้ดูแลและอำนวยความสะดวกในเรื่องกิจกรรมเกี่ยวกับการเรียนและงานสารนิพนธ์ ตั้งแต่เริ่มต้นการศึกษาจนทำงานสารนิพนธ์ เสร็จสมบูรณ์

ความสำเร็จที่เป็นผลเนื่องมาจากความตั้งใจ อุทิศเวลาเพียรในการศึกษาเล่าเรียน คุณค่าประโยชน์ และความดีที่อาจมีในสารนิพนธ์ฉบับนี้ ผู้วิจัยขอขอบพระคุณของบิดามารดา ครูอาจารย์และผู้มีพระคุณทุกท่าน ที่มีส่วนให้การศึกษา แก่ผู้วิจัยทุกระดับ ให้การสนับสนุนในเวลาและโอกาสที่แตกต่างกันผู้วิจัยขอขอบพระคุณทุกท่าน

เศรษฐพงษ์ อิ่มสุวรรณ

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	๗
บทคัดย่อภาษาอังกฤษ.....	๘
กิตติกรรมประกาศ.....	๑
สารบัญตาราง.....	๗
สารบัญภาพ.....	๗
บทที่	
1. บทนำ.....	1
1.1 ความสำคัญและความเป็นมาของปัญหา.....	1
1.2 วัตถุประสงค์ของการวิจัย.....	2
1.3 ขอบเขตของการวิจัย.....	2
1.4 ประโยชน์ที่ได้รับ.....	2
1.5 ขั้นตอนการดำเนินการวิจัย.....	2
2. แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง.....	4
2.1 แนวความคิด และทฤษฎีเรื่อง การควบคุมคุณภาพ.....	4
2.2 งานวิจัยที่เกี่ยวข้อง.....	6
2.3 การทดสอบซอฟต์แวร์ (Software Testing).....	8
2.4 Acceptance Test Driven Development (ATDD).....	16
2.5 Test Case	17
2.6 Testing Technique : Robot Framework	19
2.7 Regression Testing	21
3. ระเบียบวิธีการวิจัย.....	25
3.1 ศึกษาการทดสอบซอฟต์แวร์ของบริษัทที่เป็นกรณีศึกษา.....	25
3.2 อุปกรณ์และเครื่องมือที่ใช้ในการวิจัย.....	26
3.3 วิเคราะห์ปัญหาการทดสอบแบบซอฟต์แวร์ Manual Testing.....	27

สารบัญ (ต่อ)

บทที่	หน้า
3.4 เหตุผลที่ใช้ Automation Testing.....	29
3.5 พัฒนา Robot Frame work สำหรับการทดสอบแบบ Automation Testing	31
3.6 เปรียบเทียบเวลาทดสอบแบบ Manual Testing และ Automation Testing	40
3.7 ทำ Regression Testing	54
4. ผลการวิจัย.....	55
4.1 ผลการออกแบบและพัฒนาระบบ.....	55
4.2 วิเคราะห์เปรียบเทียบเวลาที่ใช้ในการทดสอบทั้ง 2 แบบ	63
4.3 ผลของการทำ Regression Testing.....	65
5. สรุปผลการวิจัย.....	66
5.1 สรุปผลการวิจัย.....	66
5.2 อภิปรายผล.....	68
5.3 ข้อเสนอแนะ	68
5.4 ข้อเสนอแนะในการทำวิจัยครั้งต่อไป	69
บรรณานุกรม.....	70
ภาคผนวก.....	73
ประวัติผู้เขียน.....	96

สารบัญตาราง

ตารางที่	หน้า
3.1 ตารางแสดงเวลาการทำทดสอบแบบ Manual Testing 15-03-2018 แสดงเวลาหน่วยเป็น ชั่วโมง:นาที:วินาที).....	42
3.2 ตารางแสดงเวลาการทำทดสอบแบบ Manual Testing 18-03-2018 แสดงเวลาหน่วยเป็น ชั่วโมง:นาที:วินาที).....	43
3.3 ตารางแสดงเวลาการทำทดสอบแบบ Manual Testing 21-03-2018 แสดงเวลาหน่วยเป็น ชั่วโมง:นาที:วินาที).....	44
3.4 ตารางแสดงเวลาการทำทดสอบแบบ Manual Testing 27-03-2018 แสดงเวลาหน่วยเป็น ชั่วโมง:นาที:วินาที).....	46
3.5 ตารางแสดงเวลาการทำทดสอบแบบ Manual Testing 29-03-2018 แสดงเวลาหน่วยเป็น ชั่วโมง:นาที:วินาที).....	47
3.6 ตารางรายงานเวลาที่ใช้ในการทดสอบ 15-03-2018 ของการทดสอบแบบ Automation Testing และแบบ Manual Testing.....	49
3.7 ตารางรายงานเวลาที่ใช้ในการทดสอบ 18-03-2018 ของการทดสอบแบบ Automation Testing และแบบ Manual Testing.....	50
3.8 ตารางรายงานเวลาที่ใช้ในการทดสอบ 21-03-2018 ของการทดสอบแบบ Automation Testing และแบบ Manual Testing.....	51
3.9 ตารางรายงานเวลาที่ใช้ในการทดสอบ 27-03-2018 ของการทดสอบแบบ Automation Testing และแบบ Manual Testing.....	52
3.10 ตารางรายงานเวลาที่ใช้ในการทดสอบ 29-03-2018 ของการทดสอบแบบ Automation Testing และแบบ Manual Testing.....	53
4.1 ตารางเปรียบเทียบเวลาที่ใช้ในการทดสอบ.....	63
4.2 แสดงผลการทดสอบ Automation Testing จำนวน 10 ครั้ง.....	64

สารบัญภาพ

ภาพที่	หน้า
2.1 ขั้นตอนการทดสอบแบบ Manual Testing.....	10
2.2 ขั้นตอนการทดสอบอัตโนมัติ.....	14
2.3 Functional Testing Process.....	15
2.4 Robot Framework Architecture and Test Flow.....	20
2.5 ขั้นตอนการทำงานของ Regression Testing.....	24
3.1 ภาพแสดงโครงสร้างการทำงานของ Robot Framework.....	31
3.2 ภาพแสดงโครงสร้างการทำงานของ Robot Framework	32
3.3 แสดงขั้นตอนการทำงานของ Keyword Click Web Element	34
3.4 แสดงขั้นตอนการทำงานของ Keyword Input Web Text	35
3.5 แสดงขั้นตอนการทำงานของ Keyword Web Element Should Be Visible	36
3.6 แสดงขั้นตอนการทำงานของ Keyword Web Element Should Be Not Visible	37
3.7 แสดงขั้นตอนการทำงานของ Keyword Verify	38
3.8 แสดงขั้นตอนการทำงานของ Keyword Common Input	39
4.1 แสดงโครงสร้างการเก็บข้อมูล	56
4.2 แสดงโครงสร้างการเก็บข้อมูลของ Test cas	56
4.3 แสดงโครงสร้างการเก็บข้อมูลของ File Resource	57
4.4 แสดงโครงสร้างการเก็บข้อมูลของ RedefineKeywords	58
4.5 แสดงหน้าจอสำหรับใช้ในการเขียน Test case	60
4.6 แสดง Robot Framework Log	61
4.7 ภาพแสดง Robot Framework Report	62

บทที่ 1

บทนำ

1.1 ความสำคัญและความเป็นมาของปัญหา

ในปัจจุบันนี้คอมพิวเตอร์ และเทคโนโลยีสารสนเทศถูกนำมาใช้งาน เพื่อตอบสนองความต้องการของหน่วยงานและเข้ามามีส่วนสำคัญสำหรับการดำรงชีวิตของมนุษย์เป็นอย่างมาก การใช้ระบบคอมพิวเตอร์และเทคโนโลยีสารสนเทศได้พัฒนาอย่างต่อเนื่อง เรื่องการอำนวยความสะดวกในการทำงาน ทำให้การทำงานในชีวิตประจำวันใช้เวลาน้อยลงได้ผลงานและผลผลิตมากขึ้น การนำคอมพิวเตอร์มาใช้งานมีพัฒนาการมาตามลำดับ ยุคแรกของการเริ่มใช้ให้คอมพิวเตอร์ทำงานใช้การเขียนโปรแกรมเป็น โค้ด และนำไปทดลองใช้เพื่อตรวจสอบว่าโปรแกรมที่เขียนขึ้นทำงานได้ ต่อมาความต้องการของผู้ใช้งานคอมพิวเตอร์มากขึ้น และงานซับซ้อนมากขึ้น การเขียนโปรแกรมพัฒนาเป็นซอฟต์แวร์ ซอฟต์แวร์ได้ถูกพัฒนาขึ้นเพื่อใช้งานในเรื่องต่าง ๆ เช่น งานในสำนักงาน หรืองานเฉพาะเรื่องของธุรกิจต่าง ๆ การพัฒนาซอฟต์แวร์มีขั้นตอนที่สำคัญ คือ การทดสอบซอฟต์แวร์ที่พัฒนาขึ้นก่อนนำไปใช้งาน การทดสอบซอฟต์แวร์ มีพัฒนาการตามลำดับ การทดสอบเริ่มจากทดสอบเพื่อให้แน่ใจว่าซอฟต์แวร์ทำงานได้ตามข้อกำหนดทางเทคนิค ใช้แรงงานให้คอมพิวเตอร์ทำงานได้ตามความต้องการของผู้ใช้งาน กรณีที่มีข้อผิดพลาดผู้เขียนโปรแกรมจะทดสอบเพื่อค้นหาข้อบกพร่องภายในโปรแกรมแล้วแก้ไขการทำงานของโปรแกรมให้แรงงาน ได้ถูกต้อง การทดสอบซอฟต์แวร์จะช่วยค้นหาข้อบกพร่อง และลดข้อผิดพลาดจากการทำงานของซอฟต์แวร์ให้เหลือน้อยที่สุด เป็นสิ่งที่จะช่วยเพิ่มคุณภาพให้กับซอฟต์แวร์ การค้นพบข้อบกพร่องก่อนทำให้แก้ปัญหาได้ก่อนการนำไปใช้งาน ในปัจจุบันหน่วยงานที่ใช้ซอฟต์แวร์จึงให้ความสำคัญกับการทดสอบซอฟต์แวร์มากขึ้น

จากวิกิพีเดีย สารานุกรมเสรี การทดสอบซอฟต์แวร์เป็นกระบวนการ เพื่อช่วยให้ซอฟต์แวร์ที่พัฒนาขึ้นใช้งาน มีความถูกต้อง ความสมบูรณ์ ปลอดภัย และมีคุณภาพที่ดี

การทำงานวิจัยเรื่องนี้ผู้วิจัยต้องการศึกษาเรื่องการทดสอบซอฟต์แวร์ที่ใช้ในการทำงานของบริษัทที่เป็นกรณีศึกษา เพื่อศึกษาว่าการทดสอบโดยใช้แบบ Manual Testing ในการทดสอบระบบโดยจะนำคนเข้ามาจำลองเป็น ผู้ใช้งานทดสอบการทำงานของระบบหรือค้นหาข้อบกพร่องของซอฟต์แวร์ การทดสอบระบบการทำงานขั้นเริ่มต้นใช้การทดสอบด้วยวิธีการทดสอบแบบ Manual Testing ในช่วงแรก ๆ ของการพัฒนาซอฟต์แวร์ที่เขียนขึ้นเพื่อใช้งานนั้น สามารถทำได้ เพราะซอฟต์แวร์มีองค์ประกอบน้อย และมีการทำงานที่ไม่ซับซ้อนมากนัก การทำงานทุกงานต้องมีการพัฒนาระบบงานและมีการทดสอบอย่างต่อเนื่อง การพัฒนาระบบงานทำให้องค์ประกอบ

เพิ่มขึ้น เมื่อซอฟต์แวร์นั้นได้รับการพัฒนาหรือเพิ่มฟีเจอร์ต่าง ๆ เข้ามาในระบบต้องทดสอบว่าการพัฒนาระบบกระทบกับระบบการทำงานที่ทำอยู่หรือไม่ เมื่อมีการพัฒนาหรือเปลี่ยนแปลงเพื่อยืนยันว่าฟีเจอร์ที่เพิ่มขึ้นมานั้นไม่กระทบกับระบบการทำงานเดิมที่ได้ทำการทดสอบไปแล้ว ทำให้มีงานบางส่วนนั้นต้องทำซ้ำ ๆ กับส่วนที่ได้ทดสอบไปแล้ว ผู้วิจัยจึงเห็นควรนำการทดสอบแบบ Automation Testing เข้ามาใช้ทดสอบเมื่อมีการปรับปรุงกระบวนการพัฒนาซอฟต์แวร์ เพื่อช่วยลดระยะเวลาการทดสอบแบบ Manual Testing และแรงงานที่ใช้ในการทดสอบ รวมถึงแก้ปัญหากระบวนการทดสอบความถดถอย (Regression Testing) เป็นการทดสอบซ้ำ ซึ่งต้องป้อนข้อมูลชุดเดิมซ้ำ วัตถุประสงค์ของการทดสอบถดถอย คือ การค้นหาข้อบกพร่อง เพื่อรับรองว่าการทำงานของซอฟต์แวร์ไม่มีข้อบกพร่องที่เกิดขึ้นจากกิจกรรมต่าง ๆ ในการพัฒนาระบบและไม่มีฟังก์ชันการทำงานใดมีความผิดพลาดเกิดขึ้นการทดสอบความถดถอยต้องทำ ทุกวัน

1.2 วัตถุประสงค์ของการวิจัย

1. เพื่อศึกษาและพัฒนาระบบ Automation Testing มาประยุกต์ใช้ในการทดสอบซอฟต์แวร์เว็บแอปพลิเคชัน เพื่อลดระยะเวลาในการทดสอบ

1.3 ขอบเขตของการวิจัย

1. พัฒนาเครื่องมือสำหรับสร้างกรณีทดสอบแบบ Automation Testing
2. ใช้ Automation Testing ทดสอบการทำงานของซอฟต์แวร์เว็บแอปพลิเคชันของบริษัทที่เป็นกรณีศึกษา
3. ทำ Regression Testing

1.4 ประโยชน์ที่ได้รับ

การใช้ Automation Testing ทดสอบการทำงานของซอฟต์แวร์เว็บแอปพลิเคชัน

1. สามารถทำงานแทนคนได้รวดเร็วและมีความแม่นยำมากกว่าคน
2. ลดปัญหาการใช้แรงงานคนทำงานที่ต้องป้อนข้อมูลที่ซ้ำ ๆ กัน
3. เป็นเครื่องมือต้นแบบสามารถนำไปพัฒนาต่อยอดได้
4. ช่วยในการทำ Regression Testing

1.5 ขั้นตอนในการดำเนินงานวิจัย

1. ศึกษาการทดสอบซอฟต์แวร์เว็บแอปพลิเคชันของบริษัทที่เป็นกรณีศึกษา
2. ศึกษาเครื่องมือที่จะนำมาใช้ในการวิจัย

3. วิเคราะห์ปัญหาการทดสอบซอฟต์แวร์แบบ Manual Testing และเหตุผลที่นำการทดสอบแบบ Automation Testing มาใช้
4. พัฒนา Automation Testing Tool สำหรับใช้ในการสร้างกรณิทดสอบแบบ Automation
5. เปรียบเทียบเวลาทำการทดสอบซอฟต์แวร์ด้วยแบบ Manual Testing และแบบ Automation Testing
6. วิเคราะห์ผลการทดสอบ ทำตารางเปรียบเทียบเวลาที่ใช้ในการทดสอบ
7. ทำ Regression Testing
8. ทำรายงานผลการทดสอบซอฟต์แวร์



บทที่ 2

แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

งานวิจัยเรื่องนี้ ผู้วิจัยต้องการศึกษา เรื่องการใช้ Automation Testing ทดสอบซอฟต์แวร์ ที่พัฒนาขึ้นใช้งานของบริษัทที่เป็นกรณีศึกษา บริษัทด้านเทคโนโลยีสารสนเทศและการสื่อสาร หนึ่งบริษัท สำหรับองค์กรหรือบริษัทธุรกิจ การทำการทดสอบซอฟต์แวร์จึงมีความสำคัญ เพื่อให้ซอฟต์แวร์ที่พัฒนาขึ้นใช้งานได้ตรงตามข้อกำหนดตามความต้องการของผู้ใช้งาน และได้ผลลัพธ์ ตามที่คาดหวัง Automation Software Testing ได้ถูกนำมาใช้งานแทน Manual Testing มากขึ้น ช่วย ในการทดสอบข้อมูลที่มีจำนวนมาก ในเวลาที่จำกัด ตรวจสอบฟังก์ชันการทำงานด้วยวิธีอัตโนมัติ และตรวจสอบผลกระทบที่เกิดขึ้นจากการเปลี่ยนแปลงหรือแก้ไขหรือพัฒนาซอฟต์แวร์

ผู้วิจัยได้ทำการศึกษาโดยอ้างอิงแนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง ดังนี้

2.1 แนวคิดและทฤษฎีที่เกี่ยวข้อง

2.2 งานวิจัยที่เกี่ยวข้อง

2.3 การทดสอบซอฟต์แวร์

2.4 Acceptance Test Driven Development (ATDD)

2.5 Test Case

2.6 Testing Technique : Robot Framework

2.7 Regression Testing

2.1 แนวความคิด และทฤษฎีเรื่อง การควบคุมคุณภาพ

2.1.1 Lean Software Development (LSD)

Lean Software Development (LSD) (Filip Kiss.2017) เกิดและพัฒนามาจากโรงงาน อุตสาหกรรม มุ่งเน้นการกำจัดความสิ้นเปลืองในการผลิต เพื่อให้การผลิตมีประสิทธิภาพ สูงสุด การใช้ทรัพยากรเพื่อให้เกินประโยชน์สูงสุด ซึ่งต่อมาได้มีบทบาทในการนำ Lean มาใช้ ใน การพัฒนา Software Lean มีแนวคิดอยู่ 7 ประการ และมี tools ในการปฏิบัติทั้งหมด 22 อย่าง

แนวคิดอยู่ 7 ประการ

1.Eliminate waste กำจัดความสิ้นเปลือง

2.Amplify Learning หลักการเบื้องต้นที่นำพาสู่การ เรียนรู้ความต้องการลูกค้า

3.Decide ad Late as Possible เป็นการคงตัวเลือก คงการตัดสินใจที่เด็ดขาดไว้

4.Deliver as fast as possible ส่งมอบงานให้ลูกค้าอย่างรวดเร็ว

5. Empower the Team องค์กรที่เติบโตแล้ว

6. Build Integrity In สร้างความสมบูรณ์ของซอฟต์แวร์

7. See the whole ถ้าเป็นระบบที่ซับซ้อนให้เริ่มจากการแบ่งส่วนของระบบให้เป็นระบบย่อยๆแล้วจัดการทีละส่วน

การพัฒนาซอฟต์แวร์ขึ้นใช้งานทุกงาน มีความมุ่งหมายให้ใช้งานได้ถูกต้องและตรงตามข้อกำหนดของผู้ใช้งาน ผู้วิจัยได้ประยุกต์ใช้แนวคิดของ Lean Software Development ตามกฎข้อที่ 1 มุ่งเน้นไปที่ Eliminate waste การกำจัดความสิ้นเปลืองในการพัฒนาซอฟต์แวร์ ความสิ้นเปลืองของงานพัฒนาซอฟต์แวร์ ได้แก่ การผลิตซอฟต์แวร์ ที่ไม่มีคุณภาพ เป็นการผลิตของเสียที่ต้องการแก้ไขอยู่บ่อยครั้งทำให้สิ้นเปลือง เวลา แรงงาน และค่าใช้จ่าย เพื่อลดความสูญเสีย เวลา แรงงาน ค่าใช้จ่าย ทำได้โดยค้นหาข้อบกพร่องของซอฟต์แวร์ก่อนนำไปใช้งานด้วยโปรแกรมทดสอบแบบอัตโนมัติ เพื่อลดความสูญเสียเวลาในการติดตามตรวจสอบและแก้ไขข้อบกพร่องแต่ละข้ออย่างรวดเร็วเป็นการสร้างความสมบูรณ์ ให้กับซอฟต์แวร์ก่อนส่งมอบงานให้ผู้ใช้งาน

2.1.2 The 7 Wastes

ขั้นตอนพื้นฐานในการพัฒนาซอฟต์แวร์นั้น ประกอบไปด้วยการวางแผน เขียนโค้ด และการทดสอบซอฟต์แวร์ โดย 7 Wastes จะมุ่งเน้นไปที่ความสิ้นเปลือง 7 ประการในการพัฒนาซอฟต์แวร์ประกอบไปด้วย(สุรัส ตังไพบุลย์ 2547)

1. Over Production ขยะจากการผลิตมากเกินไป
2. Over Processing ขยะจากการมีขั้นตอนการทำงานมากเกินไป
3. Transportation ขยะจากการเคลื่อนย้ายงาน
4. Inventory ขยะจากการมีของคงคลังมากเกินไป
5. Motion ขยะจากการเคลื่อนไหวของคนทำงาน
6. Waiting ขยะจากการรอคอย
7. Defects ขยะจากผลิตภัณฑ์ที่บกพร่อง

การกำจัดความสิ้นเปลือง ผู้วิจัยได้เลือกขั้นตอนที่ 7 ในการพัฒนาซอฟต์แวร์คือ Defects ข้อบกพร่องที่เกิดขึ้นขณะที่มีการพัฒนาซอฟต์แวร์เป็นสิ่งที่หลีกเลี่ยงไม่ได้แต่จำนวนข้อบกพร่องสามารถจัดการได้จากโดยใช้วิธีการที่เหมาะสม การกำจัดข้อบกพร่องทีมพัฒนาซอฟต์แวร์จะเน้นไปที่การป้องกันการเกิดขึ้นของข้อบกพร่องตั้งแต่ขั้นตอนแรกๆของการพัฒนา เพื่อให้สามารถค้นหาข้อบกพร่องตั้งแต่ขั้นตอนแรกๆของการพัฒนา การค้นพบข้อผิดพลาดได้เร็วจะลดผลกระทบที่อาจจะเกิดขึ้นในขั้นตอนการพัฒนาในระบบในลำดับต่อไป ข้อมูลที่เกี่ยวข้องกับการ

วิเคราะห์ข้อบกพร่อง หรือรายงานผลการแก้ไขข้อบกพร่อง ควรเป็นการทำงานแบบอัตโนมัติและสื่อสารกันระหว่างทีมพัฒนา ทีมผู้ทดสอบ และผู้ใช้งาน

2.2 งานวิจัยที่เกี่ยวข้อง

รัชนก ไชยประเสริฐ (2013) ได้ทำการวิจัย เรื่องกรอบการทดสอบระบบอัตโนมัติในระบบ POCT (Point of Care Testing) งานวิจัยนี้เสนอการทดสอบระบบแบบอัตโนมัติในกระบวนการพัฒนาซอฟต์แวร์ ซึ่งช่วยเพิ่มประสิทธิภาพในการทดสอบซอฟต์แวร์และช่วยลดต้นทุนในการพัฒนาซอฟต์แวร์ด้านทรัพยากร คน เวลา และสามารถรายงานผลการทดสอบได้อย่างรวดเร็ว

ชนพล สิทธิคุณฤกษ์ (2554) ได้ทำการวิจัยเรื่อง การพัฒนาระบบจัดการกรณีทดสอบซอฟต์แวร์ งานวิจัยเรื่องนี้ได้พัฒนาเครื่องมือระบบจัดการกรณีทดสอบซอฟต์แวร์ เพื่อนำมาช่วยผู้ทดสอบในการสร้างกรณีทดสอบ นอกจากนั้นยังช่วยในการเก็บข้อมูล กรณีทดสอบ การตรวจสอบ การติดตาม การแก้ไข เพิ่ม-ลด กรณีทดสอบที่ผู้ทดสอบใช้ในการทดสอบทั้งหมด เพื่อให้การทดสอบทำได้สะดวกยิ่งขึ้นและเป็นการลดภาระของผู้ทดสอบ

สกรณ์ บุญบง (2556) ได้ทำการวิจัยเรื่องการสร้างกรณีทดสอบ สำหรับการทดสอบระดับรวมหน่วยเพิ่มทีละหน่วยโดยอัตโนมัติ จากกรณีทดสอบระดับหน่วย งานวิจัยนี้ได้เสนอแนวทางแก้ไขปัญหาค่าซ้ำ และยุ่งยากของการสร้างกรณีทดสอบระดับรวมหน่วย โดยพัฒนาเครื่องมือที่ใช้ในการสร้างกรณีทดสอบระดับรวมหน่วยอัตโนมัติจากกรณีทดสอบระดับหน่วยโดยใช้วิธีเพิ่มทีละหน่วยจากกรณีทดสอบระดับหน่วย งานวิจัยเรื่องนี้อธิบายการสร้างความสัมพันธ์ระหว่างโมดูลเพื่อลดความยุ่งยากซับซ้อน และเวลาในการสร้างกรณีทดสอบ

สิวิมล เย็นไสว (2558) ได้ทำการวิจัยเรื่องแนวทางแก้ไข ปัญหาการทดสอบซอฟต์แวร์ด้วยวิธี ATDD และการบริหารผลการปฏิบัติงาน งานวิจัยเรื่องนี้มีวัตถุประสงค์เพื่อศึกษาปัญหาที่ส่งผลกระทบต่อประสิทธิภาพการทดสอบ ซอฟต์แวร์และนำเสนอแนวทางการแก้ปัญหาที่คาดว่าจะส่งผลกระทบต่อประสิทธิภาพของซอฟต์แวร์ แนวทางแก้ไขโดยเลือกแบบ ATDD Acceptance Test Driven Development มาใช้ร่วมกับการบริหารผลการปฏิบัติงาน โดยที่ขั้นตอนการทำงานของ ATDD ช่วยในด้านของการออกแบบก่อนเริ่มการทดสอบซอฟต์แวร์ พบว่า การนำเครื่องมืออัตโนมัติมาใช้แก้ปัญหาด้านเครื่องมือช่วยลดเวลาในการทดสอบซอฟต์แวร์ ช่วยค้นพบจุดบกพร่องในการทดสอบทำให้แก้ไขได้รวดเร็ว

ณัฐรัตน์ หาญวรวงศ์ (2556) ได้ทำการวิจัยเรื่อง การออกแบบและพัฒนาระบบการสร้างกรณีทดสอบสำหรับการทดสอบซอฟต์แวร์แบบอัตโนมัติ โดยใช้โครงสร้าง UI user interface ผู้วิจัยได้

ศึกษาเรื่อง กรอบการทำงานที่ช่วยให้การพัฒนาซอฟต์แวร์รวดเร็วและสามารถวิเคราะห์กรณีทดสอบ เพื่อหาส่วนของซอฟต์แวร์ที่ควรได้รับการทดสอบ ผู้วิจัยจึงพัฒนาเครื่องมือสร้างโครงสร้าง UI ของหน้าจอของซอฟต์แวร์ที่ต้องการทดสอบ เครื่องมือที่พัฒนาจากโครงสร้างของ UI สามารถสร้างกรณีทดสอบสำหรับการทดสอบแบบอัตโนมัติได้จากข้อมูลของโครงสร้าง UI

Juraj Huska (2012) ได้ทำวิจัยเรื่อง Automated Testing of the Component-based Web Application User Interface เพื่อศึกษาแนวโน้ม การทดสอบ Web Application ขององค์กรธุรกิจ ผู้วิจัยสำรวจเครื่องมือของ Automated Testing เพื่อเลือกใช้กับ Web Application และเปรียบเทียบการใช้เครื่องมือในประเด็นการใช้งาน ซึ่งจะทำให้การทดสอบได้ผลสำเร็จ ผลการวิจัยพบว่า การประยุกต์ใช้ Application Programming Interface (API) สำหรับใช้ทดสอบองค์ประกอบของ Web Application ทำให้ผลการทดสอบเป็นผลสำเร็จ

Phat Chau Tan (2016) ได้ทำงานวิจัยเรื่อง Automation Testing With Robot Framework ผลงานวิจัยพบว่า Automation Testing ช่วยลดเวลาที่ใช้ในการทดสอบได้มาก ในขณะที่เดียวกันก็ทำ Regression Testing ให้ได้เพราะ Automation Test สามารถ Run Test ในเวลากลางคืนก็ได้ วันหยุดสุดสัปดาห์ก็ได้ ผลการทำวิจัยเรื่องนี้ Test Case ทุกกรณี run ได้ผลเสร็จสมบูรณ์ และประสบความสำเร็จ ถึงแม้ว่าจะมีปัญหาให้แก้ไขหรือมีสิ่งที่ไม่เข้าใจยาก และสลับซับซ้อนในบางขั้นตอน ทำให้ผลการทดสอบน่าเชื่อถือ

Alazar Seyoum Haile (2011) ทำงานวิจัยเรื่อง Automation of Test Cases for Web Application of CRM Test Cases (Customer Relationship Management) ผู้ดำเนินการวิจัยนำกรณีทดสอบ 10 Test Cases ผลการทดสอบด้วยแบบ Automation Testing ใช้เวลาในการทดสอบเปรียบเทียบกับเวลาที่ทำการด้วยแบบ Manual Testing ใน 10 Test Cases เดียวกัน กรณีต่อกรณี กระบวนการทดสอบด้วยแบบ Manual Testing ใช้เวลามากกว่าการทดสอบแบบ Automation Testing ทุกกรณี เมื่อเปรียบเทียบกันโดยเฉลี่ยของการทดสอบ 5 กรณี เวลาที่ใช้ในการทดสอบแบบ Automation Testing ใช้เวลาเพียง 16% โดยเฉลี่ยของเวลาที่ใช้ทดสอบแบบ Manual Testing ผลงานวิจัยที่ได้ก็อีกอย่าง คือ การใช้ Automation ทดสอบ Web ของ Customer Service ลดจำนวน Error ที่อาจจะเกิดจากการป้อนข้อมูลที่มีความสลับซับซ้อน เมื่อต้องป้อนข้อมูลซ้ำ ๆ ในการทำการทดสอบถดถอย

2.3 การทดสอบซอฟต์แวร์ (Software Testing)

รังสิต ศิริรังษี (2557) กล่าวว่า การทดสอบซอฟต์แวร์ เป็นกระบวนการของการวิเคราะห์การทำงานของซอฟต์แวร์ เพื่อตรวจสอบว่า ซอฟต์แวร์ที่พัฒนาขึ้นใช้งานทำงานได้ตรงตามความต้องการตามข้อกำหนดของผู้ใช้งาน และไม่มีข้อบกพร่องของระบบ การทดสอบช่วยเพิ่มความเชื่อมั่นให้กับ โปรแกรมที่ได้รับการทดสอบที่ได้รับการแก้ไขข้อบกพร่อง ขณะทำการทดสอบ

การทดสอบซอฟต์แวร์ เป็นสิ่งที่เกิดขึ้นพร้อมกับการนำคอมพิวเตอร์มาใช้งาน ผู้ใช้คอมพิวเตอร์ต้องเขียน โปรแกรมสั่งให้คอมพิวเตอร์ทำงานตามลักษณะของงาน เป็นงานในสำนักงานหรืองานเฉพาะเรื่อง โปรแกรมที่เขียนขึ้นใช้งานเพื่อสั่ง และควบคุมคอมพิวเตอร์ให้ทำงานตามจุดประสงค์ของการเขียนโปรแกรม ต่อมาได้พัฒนาการเขียนโปรแกรมขึ้นเป็นการเขียนซอฟต์แวร์ การทดสอบซอฟต์แวร์ที่พัฒนาขึ้นใช้งานในช่วงแรก ๆ ทำการทดสอบด้วยแบบ Manual Testing เป้าหมายของการทดสอบซอฟต์แวร์เพื่อตรวจสอบความถูกต้องการทำงานของ โปรแกรม เพื่อค้นหาข้อบกพร่องและลดความผิดพลาดที่อยู่ภายในโปรแกรม

ปัจจุบันนี้การพัฒนาซอฟต์แวร์ เพื่อใช้งานมีความซับซ้อนมากขึ้น ตามกระบวนการทำงานเฉพาะเรื่อง เครื่องมืออัตโนมัติ Automation Testing จึงถูกคิดค้นขึ้น เพื่อใช้แทนการทดสอบด้วยมือ เพราะการใช้เครื่องมืออัตโนมัติสามารถนำกลับมาใช้ใหม่ได้ (Reusability) โดยไม่ต้องป้อนข้อมูลใหม่ มีความคงที่ (Consistency) มีความเร็ว (Speed) มากกว่าการทดสอบด้วยแบบ Manual Testing ช่วยเพิ่มความครอบคลุมในการทดสอบ (Productivity) จากเหตุผลที่กล่าวมานี้ จึงเป็นข้อตกลงของทีมงานในบริษัทที่เป็นกรณีศึกษาที่จะทำการทดสอบซอฟต์แวร์ด้วยแบบ Automation Testing

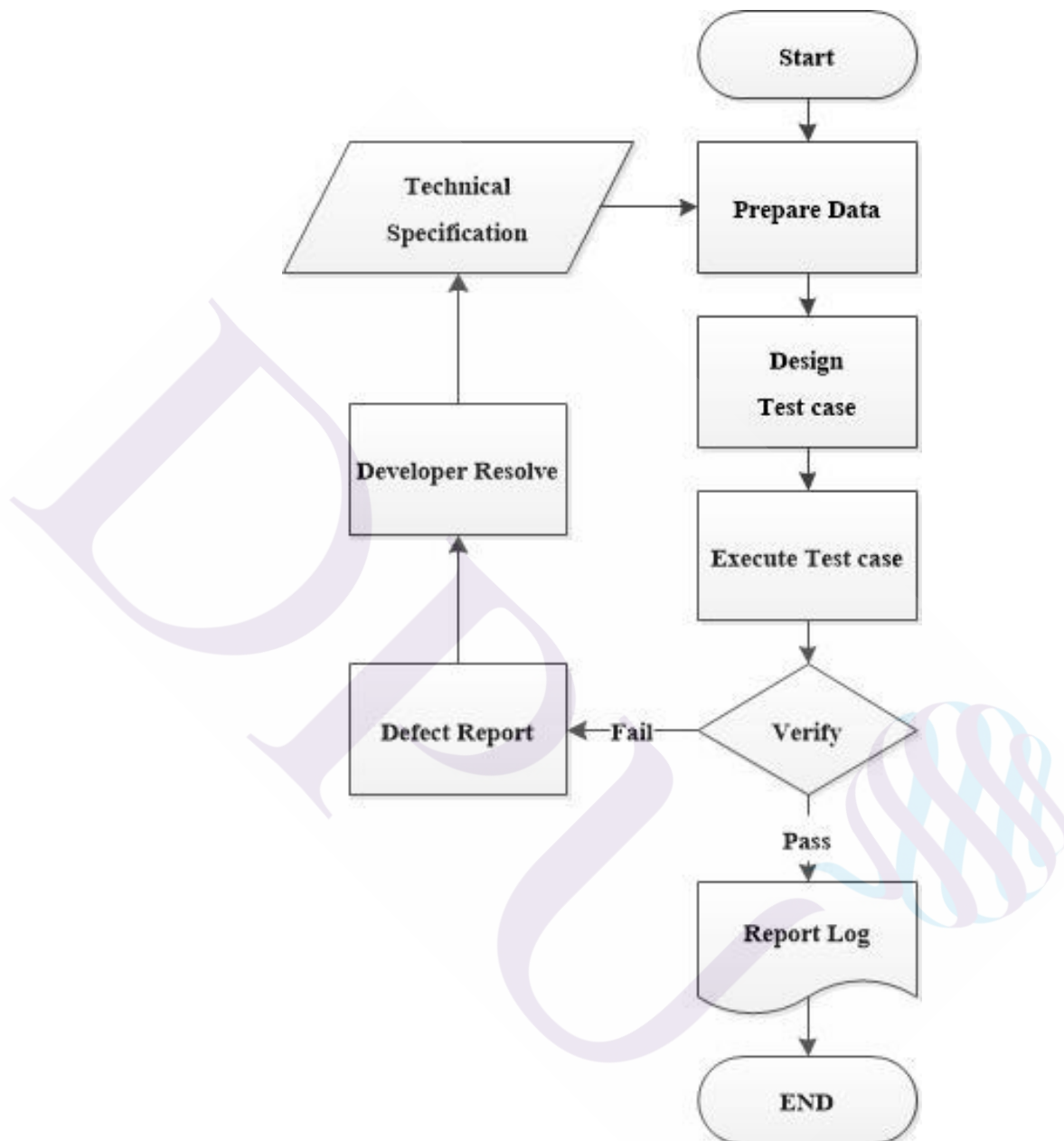
การทดสอบซอฟต์แวร์นั้น ผู้ทดสอบจะเลือกเทคนิควิธีที่เหมาะสมและมีประสิทธิภาพ มาใช้ในการทดสอบ วิบูลย์ ชัยจิราภรณ์ (2556) นำเสนอความรู้เรื่อง แนวโน้มการทดสอบซอฟต์แวร์ในอนาคตจะเป็นการทดสอบโดยนำเครื่องมือ Automation Testing Tools เข้ามาใช้ในการทดสอบ เพราะลดระยะเวลา แรงงาน และยังใช้ทำการทดสอบความถดถอย (Regression Testing) เพื่อตรวจสอบความเชื่อมั่นการทำงานของระบบว่า สามารถทำงานได้ตามความต้องการที่กำหนดไว้ เมื่อมีการแก้ไข พัฒนา เปลี่ยนแปลงซอฟต์แวร์

การทดสอบซอฟต์แวร์แบบ Manual Testing

การทดสอบด้วยมือ (Manual Testing) เป็นการทดสอบด้วยผู้ทดสอบ โดยไม่ใช้การเขียนโปรแกรมคอมพิวเตอร์ หรือเครื่องมืออัตโนมัติเข้ามามีส่วนร่วมในการตรวจสอบซอฟต์แวร์ การทดสอบแบบนี้ผู้ทดสอบจะประมวลผลกรณีทดสอบด้วยมือใช้ความรู้พื้นฐานในการทดสอบ เป็นกระบวนการในการหาข้อบกพร่องในโปรแกรมให้ทำงานได้ถูกต้องตามข้อกำหนดที่ต้องการ และไม่มีข้อผิดพลาด โดยผู้ทดสอบที่ทำการทดสอบแบบ Manual Testing จะทำการทดสอบตามกระบวนการทดสอบใช้ Technical Specification ดังกระบวนการต่อไปนี้

1. การทดสอบจะเริ่มต้นจากการ Prepare Data เตรียมข้อมูล สำหรับการทดสอบ ได้แก่ file, username, password ชุดข้อมูลสำหรับการทดสอบ
2. ขั้นตอนที่สอง ผู้ทดสอบ ทำ design test case จะ design ให้สอดคล้องกับ Technical specification เพื่อใช้ทดสอบส่วนประกอบของ design test case ประกอบด้วย
 - 2.1 test case number เพื่อจำแนก test case ให้เป็นหมวดหมู่
 - 2.2 test description เพื่ออธิบายว่า กระบวนการ การทดสอบของข้อนี้ๆ ต้องการทดสอบอะไร
 - 2.3 test step ระบุขั้นตอนการทดสอบ
 - 2.4 expected result ระบุผลลัพธ์ที่คาดหวัง
 - 2.5 actual result บันทึกผลที่ได้จากการทดสอบจริง เพื่อตรวจสอบว่าได้ผลตรงกับ expected result หรือไม่
 - 2.6 test result ผลการทดสอบ pass / fail
3. ขั้นตอนต่อจากการทำ design test case เรียบร้อยแล้ว ทำการ execute test case โดยจะทดลองกรอกข้อมูลจริง เพื่อทดสอบว่าการทำงานเป็นไปตาม Technical Specification ถูกต้องหรือไม่ ด้วยการ verify การทวนสอบ ประเมินผลการทำงานภายในวงจรเพื่อตรวจสอบว่า ผลงานตรงกับความต้องการที่กำหนดไว้ พร้อมบันทึกผล actual result กรณีที่ผลการทดสอบได้ผลตรงกับ expected result จะบันทึกผลการทดสอบ pass ลงในช่อง test result กรณีที่ผลการทดสอบได้ผลไม่ตรงกับ expected result จะบันทึกผล fail ลงในช่อง test result แล้วทำการบันทึกผลลงใน defect report เพื่อส่งให้ developer ตรวจสอบหาปัญหา ทำการแก้ปัญหา และทำการ execute test case จนได้ผล ตรงกับ Technical Specification แล้วบันทึกผลลง Technical Specification อีกครั้ง จนได้ผลว่า pass
4. บันทึกข้อมูลรายละเอียดผลการทดสอบ

ขั้นตอนการทดสอบแบบ Manual Testing



ภาพที่ 2.1 ขั้นตอนการทดสอบแบบ Manual Testing

การทดสอบซอฟต์แวร์อัตโนมัติ (Automation Testing)

การทดสอบซอฟต์แวร์ เป็นกระบวนการที่ใช้ระบุความถูกต้องในการทำงานของซอฟต์แวร์ที่พัฒนาขึ้นใช้งานได้ตามข้อกำหนดของความต้องการและค้นหาข้อบกพร่องที่เกิดขึ้นขณะทำการทดสอบ เพื่อทำการแก้ไขระหว่างขั้นตอนการทดสอบ การทดสอบอัตโนมัติ (Automation Testing) เป็นการทดสอบโดยนำเครื่องมือ (Automation Testing Tool) เข้ามาช่วยในการทดสอบเป็นการประมวลผลโปรแกรมภายใต้การทดสอบด้วยคอมพิวเตอร์ เป้าหมายหลักของการทดสอบแบบอัตโนมัติ คือ ความพยายามที่จะลดระยะเวลาที่ใช้ในการทดสอบในกรณีที่ผู้ทดสอบต้อง Execute Test Case ซ้ำ ๆ กันทุกวัน Automation Testing จะสามารถช่วยลดระยะเวลาในการป้อนข้อมูลซ้ำแบบทำด้วยมือได้มาก ขั้นตอนต่อจากการทดสอบซอฟต์แวร์ ผู้ทดสอบต้องทำ Regression Test เพื่อตรวจสอบฟังก์ชัน การทำงานของระบบให้ทำงานได้ผล ตรงตามความต้องการของผู้ใช้งานและไม่มีผลผิดพลาดเกิดขึ้น Automation Testing ช่วยให้ผู้ทดสอบทำ Regression Testing ได้อย่างรวดเร็ว และสามารถทดสอบได้จากเครื่องคอมพิวเตอร์พร้อมกันได้มากกว่าหนึ่งเครื่อง

ในปัจจุบันมีนักทดสอบซอฟต์แวร์พัฒนา Test Tool ขึ้นมาใช้งานหลายประเภท

ขั้นตอนแรก Automated Test Tool Selection ของการทดสอบอัตโนมัติ คือ การเลือกเครื่องมือที่สามารถใช้งานได้ตรงตามความต้องการกับระบบที่ต้องการทดสอบ พิจารณาจากปัจจัยที่มีส่วนเกี่ยวข้อง ได้แก่ ภาษาที่ใช้เขียนโปรแกรม ราคา ความซับซ้อนในการใช้งาน ความสามารถของเครื่องมือ การแสดงผลที่ดูได้ง่าย และการสนับสนุน อาทิเช่น มีการอัปเดตอยู่ตลอดเวลาเพื่อแก้ปัญหาต่างๆ ของเครื่องมือ

ขั้นตอนที่สอง Define Scope of Automation พิจารณาจาก จำนวนข้อมูลทดสอบ ฟังก์ชันการทำงาน ความซับซ้อนของกรณีทดสอบ

ขั้นตอนที่สาม Planning, Design, Development การวางแผนทดสอบจาก Requirement ที่ได้รับมา วางแผนว่าจะทดสอบเรื่องอะไรบ้าง จัดหาทรัพยากรที่จำเป็น เช่น Hardware, Software, Browsers, Database System เลือกเทคนิคการทดสอบ การเตรียมข้อมูล กำหนดระยะเวลาที่ใช้ในการทดสอบ

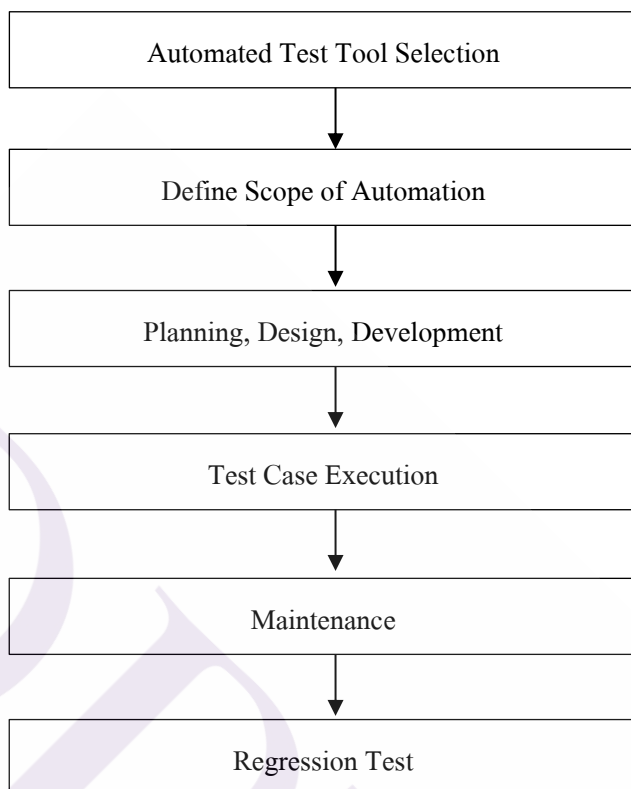
ขั้นตอนที่สี่ Test case Execution ขั้นตอนประมวลผลการทดสอบ สร้าง Test Script สคริปต์เป็น โปรแกรมที่ถูกเขียนขึ้น เพื่อใช้ทดสอบส่วนที่เป็นฟังก์ชันการทำงานของระบบ เครื่องมือทดสอบอัตโนมัติ สามารถแสดงรายงานผลการทดสอบประมวลผลได้ทันที

ขั้นตอนที่ห้า Maintenance ขั้นตอนการบำรุงรักษา สคริปต์ที่ใช้ในการทดสอบอัตโนมัติ ผู้ทดสอบจำเป็นต้องแก้ไขเพิ่มเติมและบำรุงรักษาอย่างต่อเนื่องเพื่อให้ทำงานร่วมกับฟังก์ชันการทำงานที่ถูกพัฒนาขึ้นในแต่ละช่วงของการอัปเดต

ขั้นตอนที่หก Regression Test การทดสอบความถดถอยเป็นการทดสอบซ้ำด้วยกรณีทดสอบที่ผ่านการประมวลผลอย่างถูกต้อง เพื่อตรวจสอบผลกระทบที่เกิดขึ้นจากการเปลี่ยนแปลงหรือแก้ไขที่เกิดขึ้นระหว่างขั้นตอนการพัฒนาและการบำรุงรักษา การทดสอบความถดถอยต้องทำทุกวัน



ขั้นตอนการทดสอบอัตโนมัติ



ภาพที่ 2.2 ขั้นตอนการทดสอบอัตโนมัติ

ข้อดีของการทดสอบอัตโนมัติ (Automation Testing)

1. ช่วยลดระยะเวลาและค่าใช้จ่ายในการทดสอบ
2. ช่วยลดความผิดพลาดในการป้อนข้อมูลหรือการทำงานซ้ำ
3. การทดสอบแบบอัตโนมัติสามารถประมวลผล Test Case ที่มีความซับซ้อนและมีจำนวนมากได้
4. สามารถทำการทดสอบที่ทำแบบ Manual Testing ไม่ได้
5. ช่วยให้ทำ Regression Testing ได้ทุกวัน
6. นักทดสอบพัฒนาการทำงานตนเองทุกครั้งที่มีการเปลี่ยนแปลง Source Code

Functional Testing

การทดสอบฟังก์ชันเป็นการทดสอบระบบซอฟต์แวร์ว่าทำงานได้ตรงตามข้อกำหนด และได้ผลลัพธ์ตรงตามความต้องการของผู้ใช้งาน

(<https://charathbank.wordpress.com/2010/10/04/qa-knowledge-testing-type/>)

การทดสอบฟังก์ชัน หมายความว่า การทดสอบแอปพลิเคชันต่อความต้องการหรือข้อกำหนดของธุรกิจ การทดสอบฟังก์ชัน คือ กระบวนการทดสอบซอฟต์แวร์ที่พัฒนาขึ้นในใช้งาน ซึ่งซอฟต์แวร์ที่พัฒนาขึ้น จะต้องถูกตรวจสอบให้เชื่อมั่นว่าสอดคล้องกับข้อกำหนดความต้องการทุกข้อ การทดสอบฟังก์ชัน คือ วิธีการตรวจสอบซอฟต์แวร์ เพื่อให้แน่ใจว่ามีฟังก์ชันการทำงานที่ตรงตามความต้องการทั้งหมด ที่ระบุไว้ในข้อกำหนดด้านการทำงานของซอฟต์แวร์

การทดสอบฟังก์ชันประกอบด้วย การประเมิน และเปรียบเทียบฟังก์ชันของซอฟต์แวร์แต่ละอย่างกับความต้องการหรือข้อกำหนดทางธุรกิจ ซอฟต์แวร์ได้รับการทดสอบโดยให้อินพุตที่เกี่ยวข้องกับบางส่วน เพื่อให้เอาต์พุตที่สามารถถูกประเมินว่าสอดคล้อง มีความสัมพันธ์หรือแตกต่างกัน เมื่อเปรียบเทียบกับความต้องการพื้นฐาน นอกจากนี้ การทดสอบฟังก์ชันยังตรวจสอบสำหรับการใช้งาน เช่น ตรวจสอบให้แน่ใจว่าฟังก์ชันการทำงานตามแนวทางของการทำงานที่ต้องการ เทคนิคการทดสอบฟังก์ชันที่ผู้วิจัยใช้คือ SDLC (Software Development Life Cycle) การวิเคราะห์ค่าขอบเขต วิเคราะห์การจัดการข้อบกพร่อง การทดสอบ UI

การทดสอบฟังก์ชันดำเนินการตามขั้นตอนต่อไปนี้

- ระบุฟังก์ชันที่ซอฟต์แวร์ถูกคาดหวังว่าจะดำเนินการนำเสนออินพุตหรือสร้างอินพุต ตามข้อกำหนดของฟังก์ชัน ทำให้ได้เอาต์พุต ตามข้อกำหนดของฟังก์ชันการทำงานของกรณีทดสอบ เปรียบเทียบเอาต์พุตที่เกิดขึ้นจริงและเอาต์พุตที่คาดหวังตรวจสอบว่า แอปพลิเคชันทำงานตามความต้องการของลูกค้า

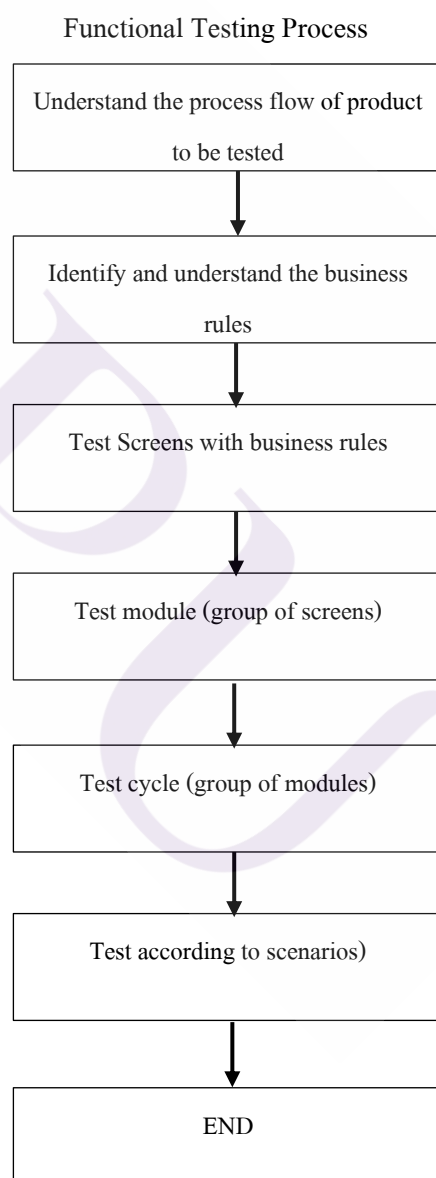
- การทดสอบฟังก์ชันจะประกอบด้วย กรณีทดสอบหลายกรณี แต่ละกรณีมุ่งเน้นผลการทดสอบต่อ Business Rules, Data Mapping, Data Validation และ Data Re-engineering

เอกสารต่าง ๆ จะถูกสร้างขึ้นระหว่างการทดสอบฟังก์ชันเอกสารต่าง ๆ ได้แก่ Test Plan, Test Design Spec, Test Scenario, Test Log และ Test Summary

Functional Testing Process

1. ทำความเข้าใจกับกระบวนการของ product ที่จะทำการทดสอบ
2. จำแนกและทำความเข้าใจกับลักษณะของธุรกิจที่จะทำการทดสอบ
3. ทดสอบการทำงานของระบบย่อยให้ทำงานได้ถูกต้องตามลักษณะของธุรกิจ

4. การทดสอบแบบทีละโมดูล เพื่อให้แน่ใจว่าส่วนประกอบย่อยของระบบแต่ละโมดูลทำงานได้อย่างถูกต้อง
5. ตรวจสอบระบบของโมดูลทุกโมดูลทำงานร่วมกันได้ถูกต้อง ตามความต้องการที่ตกลงกันไว้
6. การทดสอบระบบรวมการทำงานของแต่ละขั้นตอนทำได้ถูกต้องเป็นไปตามสถานการณ์ที่กำหนดไว้



ภาพที่ 2.3 Functional Testing Process

ที่มา: <http://sdlcservices.com/functional-testing.html>

2.4 Acceptance Test Driven Development (ATDD)

Elisabeth Henrickson (2008) นำเสนอเรื่อง ATDD ในการประชุมวิชาการเรื่อง Driving Development with Tests: ATDD and TDD การทดสอบซอฟต์แวร์แบบ ATDD หมายถึง แนวปฏิบัติระหว่างทีมพัฒนาเครื่องมือทดสอบ (Developer) และผู้ใช้งานซอฟต์แวร์ (Customer) ทำข้อตกลง ทำความเข้าใจให้ตรงกันเรื่องเกณฑ์สำหรับใช้ตรวจรับงานที่ทำการทดสอบ ATDD มีแนวทางปฏิบัติงานมุ่งเน้นการมีส่วนร่วมของลูกค้า

ลักษณะของ ATDD

1. Common Understanding เป็นการสร้างความเข้าใจที่ตรงกันให้กับทีมพัฒนาเครื่องมือทดสอบ และลูกค้าโดยการใช้ตัวอย่างในการแสดง Requirements
2. Ubiquitous Language ใช้ภาษาที่ให้ความหมายเดียวกัน ในการสื่อสารเดียวกัน ภาษาที่ใช้ในการเขียน Test Case และ Requirement เป็นภาษาเดียวกัน ทำให้ง่ายต่อการตีความและการทำความเข้าใจ
3. Executable Requirements / Examples Living Documentation เป็นการสร้าง Requirement ที่สามารถประมวลผล โดยคอมพิวเตอร์ได้
4. มีชื่อเรียกอย่างอื่นอีก เช่น Story Test Driven Development, Behavior Driven Development, Executable Specifications, Specification by example

ATDD Cycle

ATDD มีการทำงานเป็นแบบวนซ้ำ ซึ่งประกอบด้วย 3 ขั้นตอน Discuss Development และ Deliver (<https://agilethailand.wordpress.com/2012>)

Discuss

เป็นขั้นตอนของการถกเถียงถึงปัญหาหรือ Feature ที่นำมาจาก Product Backlog เพื่อหา Solution โดยทีมงานทั้งหมดจะร่วมกันหาวิธีการแก้ปัญหาที่ดีที่สุด ขั้นตอนนี้ช่วยเพิ่มความเข้าใจ และคำศัพท์ที่ใช้ร่วมกัน เพื่อสื่อความหมายเดียวกัน รวมถึงมีการกำหนดตัวอย่างที่ชัดเจนสำหรับ Solution เพื่อใช้ในการสร้างเป็น Test Cases ซึ่งสามารถประมวลผลโดยคอมพิวเตอร์ได้ต่อไป

Develop

ตัวอย่างที่กำหนดไว้ในขั้นตอน Discuss จะเป็นตัวช่วยและแนะแนวทางในการพัฒนา Feature โดยในระหว่างพัฒนา Feature ก็สามารถสร้าง Automation Test Cases สำหรับ Feature นั้น

ในเวลาเดียวกัน ซึ่งทั้งทีมงานต้องร่วมกันรับผิดชอบการ Automation ร่วมกัน และในตอนสุดท้าย เมื่อทุก Test Cases ผ่าน การอภิปรายและการแก้ไขปรับปรุงเป็นอันเสร็จขั้นตอน Develop

Deliver

ขั้นตอนการส่งมอบงาน คือ การแสดงการทำงานของ Feature นั้น ให้กับทุก ๆ Stakeholders และทำการ Execute Test Cases เพื่อทดสอบการทำงานของ Feature ที่เสร็จแล้ว เป็นที่ยืนยันว่า Feature นี้ ทำงานตรงตามความต้องการของลูกค้า ดังที่ได้ตกลงกันในช่วง Discuss หากได้รับคำแนะนำหรือข้อคิดเห็นก็นำไปพิจารณาในช่วง Discuss ครั้งต่อไป

2.5 Test Case

จากข้อกำหนดของ IEEE Standard For Software Test

IEEE Std 829-1998 การเขียน Test case ประกอบด้วย

1. Structure ของ Test Case

กรณีทดสอบประกอบด้วย 3 ส่วนที่สำคัญ ดังนี้

1.1 Information ข้อมูลประกอบด้วย ข้อมูลเกี่ยวกับกรณีทดสอบ ได้แก่ Test number, Tester name, Test version, Test case name, วัตถุประสงค์ของ Test case และรายการที่ต้องการทดสอบ

1.2 Activity กิจกรรม จัดลำดับ กิจกรรมที่ต้องทำก่อน ระบุกิจกรรมที่ต้องดำเนินการหลังจากสิ้นสุดการทำงานของกรณีทดสอบ ซึ่งเป็นการกระทำที่ละขั้นตอนขณะที่มีการทดสอบ รวมถึงข้อมูลอินพุตที่เตรียมไว้สำหรับการทดสอบ

1.3 Result ผลลัพธ์ ที่ได้จากการทำงานของกรณีทดสอบ ผลลัพธ์ที่แท้จริงจากการทำงานของระบบ

2. Component ของ Test Case

รายการของการกระทำพร้อมผลลัพธ์ที่คาดหวัง ซึ่งต้องสอดคล้องกับข้อกำหนดของระบบ กรณีทดสอบ ประกอบด้วย

2.1 วัตถุประสงค์ของการทดสอบ หรือรายการความต้องการของระบบที่ต้องการทดสอบ

2.2 คำอธิบายของวิธีการที่ใช้ในการทดสอบ

2.3 คิดตั้งส่วนประกอบที่จะใช้ในการทดสอบ ได้แก่ ฮาร์ดแวร์ ซอฟต์แวร์ ระบบปฏิบัติการที่ใช้ ข้อมูล คำอธิบายประกอบการทดสอบ

2.4 อินพุต เอาท์พุต วิธีทดสอบ ผลลัพธ์ที่คาดหวัง เงื่อนไขของการทดสอบ

3. Creating test case

การสร้าง กรณีทดสอบประกอบด้วยขั้นตอน ดังต่อไปนี้

3.1 ระบุขั้นตอนที่จะดำเนินการทดสอบ โดยใช้คำอธิบายที่กระชับและระบุรายการให้ชัดเจน

3.2 ขั้นตอนแรกของการสร้าง test case พิจารณากระบวนการของแต่ละขั้นตอน ตั้งแต่ขั้นตอนเริ่มต้นจนถึงขั้นตอนแล้วเสร็จ

3.3 การเขียน Test Case เพื่อทดสอบ Feature Systems ต้องศึกษารายละเอียดและทำความเข้าใจกับ ข้อกำหนดของระบบให้ถูกต้องชัดเจน

3.4 ถ้าต้องการเพิ่มเติมส่วนประกอบเฉพาะกรณีต้องแน่ใจว่า ขั้นตอนการปฏิบัติที่เพิ่มขึ้นจะช่วยให้การทำ Test Case ประสบความสำเร็จ

3.5 กำหนดผลที่คาดหวังของโปรแกรม

4. The Test Case IEEE 829 ประกอบด้วย

4.1 Test case ID : เลขที่ของ test case ซึ่งเป็นเลขเฉพาะของแต่ละเคส เพื่อใช้ระบุลำดับของเคส

4.2 Test case description : คำอธิบายของกรณีทดสอบ

4.3 Test prerequisite: การจัดลำดับตามเงื่อนไขที่จะถูกทดสอบ

4.4 Test steps: นำเสนอในการทดสอบระบบ รายละเอียดรวมอยู่ในกระบวนการทดสอบ

4.5 Expected result: เป็นการคาดคะเนผลลัพธ์ของระบบล่วงหน้า หรือสิ่งที่ผู้ใช้คาดหวังจากฟังก์ชันการทำงานของระบบ

4.6 Actual result: เป็นผลลัพธ์ที่แท้จริงจากการกระทำตามอินพุตที่กำหนดไว้ ซึ่งกรณีผลลัพธ์ที่คาดหวังตรงกับผลลัพธ์ที่แท้จริง ถือว่า pass หากไม่ใช่ถือว่า fail

(<http://en.wikipedia.org/wiki/Test-driven-development>)

2.6 Robot Framework

Robot Framework เป็นการทดสอบซอฟต์แวร์ที่เป็น Open Source ได้รับการพัฒนาบนพื้นฐานของภาษา Python สามารถใช้ Test Libraries ที่ถูกสร้างขึ้นมาด้วยภาษา Python หรือ Java ได้ Robot Framework เป็นเครื่องมือสำหรับทำ Acceptance Test โดยใช้วิธีการที่เรียกว่า Keyword-Driven Testing Approach Robot Framework เป็น Open Source ที่ได้รับการเผยแพร่ภายใต้เงื่อนไขของ Apache License 2.0 ให้ใช้ได้ฟรี ปรับแต่งได้ License ประเภทนี้ไม่มีการผูกขาด ทุกคนสามารถใช้ได้โดยไม่มีค่าใช้จ่าย Robot Framework นั้น ได้รับการพัฒนาขึ้นโดยบริษัท Nokia Networks (<https://networks.nokia.com/>) และในปัจจุบันได้รับการพัฒนาและทำงานต่อโดย Robot Framework Foundation เป็นหน่วยงานที่จะทำหน้าที่ตรวจสอบและแก้ไขข้อบกพร่องต่าง ๆ ที่เกิดขึ้นจากกลุ่มผู้ใช้งาน Robot Framework จุดเด่นของ Robot Framework นั้น คือ ถูกออกแบบมาเพื่อให้มี syntax ที่ตรงไปตรงมา สามารถเข้าใจได้ง่าย ไม่ซับซ้อน ไม่จำเป็นต้องเข้าใจภาษา Programming ทำให้ง่ายต่อการเขียน Test Case นอกจากนี้ยังมี เครื่องมือที่สำหรับเขียน Test Case อาทิเช่น RIDE (Robot Framework Integrated Development Environment) Editor ที่สร้างโดยทีมพัฒนา Robot Framework หรือ Eclipse IDE โดยจะมี Robot Framework Plugin ให้ใช้งาน

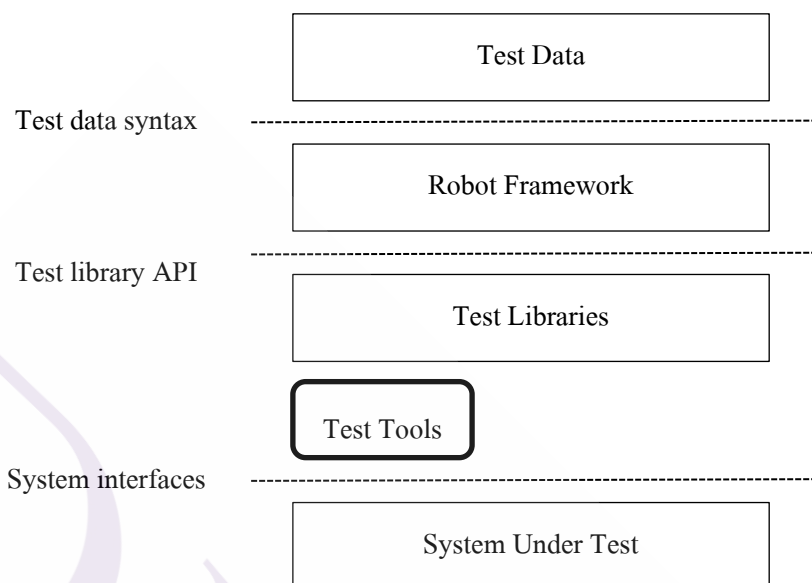
Robot Framework มีจุดศูนย์กลางอยู่บน Git Hub, Apache 2 License เมื่อ User ต้องการหาข้อมูลเพิ่มเติม หรือเอกสารคู่มือการใช้งาน รวมไปถึง Source Code และติดตามปัญหาการใช้งานต่าง ๆ ได้ โครงสร้างการทำงานของ Robot Framework สามารถใช้ Test Library ทั่วไป หรือ Tools ที่ใช้ช่วยการทำงานของ Robot Framework เช่น RIDE, RED On Eclips

Robot Framework ถูกออกแบบด้วย Modular Architecture สามารถเขียน Library เพิ่มเติมนอกเหนือจาก Standard Libraries

เมื่อดำเนินการทดสอบแล้วกรอบแรกจะแยกวิเคราะห์ข้อมูลทดสอบ จากนั้นจะใช้คำหลักจาก Library ทดสอบเพื่อโต้ตอบกับระบบที่กำลังทดสอบ Library สามารถสื่อสารกับระบบได้โดยตรงหรือใช้เครื่องมือทดสอบอื่น ๆ เป็นไคร์เวอร์

การดำเนินการทดสอบเริ่มต้นจากบรรทัดคำสั่งเป็นผลให้ผู้ทดสอบได้รับรายงานและเข้าสู่ระบบในรูปแบบ HTML เช่นเดียวกับการส่งออก XML เหล่านี้ ให้รูปลักษณ์ที่กว้างขวาง

Robot Framework Architecture and Test Flow



ภาพที่ 2.4 Robot Framework Architecture and Test Flow

ที่มา: <http://robotframework.org/#documentation>

คำอธิบาย

ขั้นตอนที่ 1 เมื่อเริ่มต้นการทำงาน Robot Framework ขั้นตอนแรกจะแยกวิเคราะห์ข้อมูลการทดสอบ

ขั้นตอนที่ 2 จากนั้นระบบจะใช้ Keyword ที่ Import มาจาก Libraries ไปยังการโต้ตอบกับระบบที่ต้องการทดสอบ

ขั้นตอนที่ 3 Libraries สามารถสื่อสารกับระบบได้ทั้งทางตรงหรือใช้ Test Tools Driver อื่น ๆ

ขั้นตอนที่ 4 โดยระบบจะเริ่มทำงานจาก Command Line หลังจากจบการทำงาน Robot Framework จะสรุป Report log ในรูปแบบของ HTML Robot Framework บน Python Platform

เหตุผลที่ใช้ Robot Framework

ในการทดสอบซอฟต์แวร์ของบริษัทที่เป็นกรณีศึกษาใช้ Robot Framework เพราะมีข้อดี ดังนี้

1. Test Speed ทำได้ค่อนข้างเร็ว เพราะเป็นภาษาที่เป็น Interpreter คือ เขียนแล้วสามารถรันได้
2. มี Editor Support หลายอัน เช่น RIDE (Robot Framework Integrated Development Environment), RED on Eclipse
3. External Test Library มีให้เลือกใช้มากมายจากการดูที่ Python Package Index เช่น Selenium 2 Library, SSH Library, Database Library, etc.,
4. สามารถสร้าง Keyword ขึ้น มาใช้งานให้สัมพันธ์กับการทำงานของระบบ โดยเฉพาะ

ข้อจำกัดบางกรณีสำหรับ Robot Framework

1. Test Library Dependencies Management ทำได้ไม่ค่อยดี ไม่มี Dependency Management Tool ช่วยทำได้เต็มที่ คือ สร้าง Requirements เก็บ List ของ Library เอาไว้ทำ pip install
2. Test Library บางตัวเขียนด้วยภาษา C (C Python) ทำให้การเอาไปรันบน OS ที่แตกต่างกันมักจะมีปัญหาหรือ install ได้ยาก
3. การ Install ค่อนข้างยาก เพราะมี Dependency สูง เช่น Python version มี support ทั้ง 2.x และ 3.x แต่ Test Library หลายตัวยังเขียน support 2.x

2.7 Regression Testing

Regression testing การทดสอบความถดถอยเป็นการทดสอบชนิดหนึ่งใน Functional Testing ใช้สำหรับการตรวจสอบปัญหาที่เกิดขึ้นจากการเพิ่มหรือแก้ไข Code ของระบบและตรวจสอบการทำงานของซอฟต์แวร์ว่ามีปัญหาหรือไม่ ต้องการความเชื่อมั่นการทำงานของซอฟต์แวร์ เมื่อมีการเพิ่มหรือแก้ไข Code สามารถทำงานได้ตามความต้องการตามข้อกำหนดเดิมและไม่ส่งผลกระทบต่อการทำงานของระบบ เป็นการทดสอบหลังการแก้ไขที่เกิดขึ้นในระหว่างขั้นตอนการพัฒนาและการบำรุงรักษา

Regression testing เป็นการทดสอบเพื่อทดสอบส่วนของระบบหรือ application ที่มีการเปลี่ยนแปลงโดยเลือกฟังก์ชันเป็นหลัก การทำ regression test ต้องทำทุกวัน เพื่อตรวจสอบฟังก์ชัน (<http://charathbank.wordpress.com>)

Regression Testing เป็นการทดสอบที่สำคัญมากของการพัฒนาซอฟต์แวร์ จะทำการทดสอบ Regression Testing เมื่อ 1. ทำการ Release ระบบหรือเพิ่ม Feature ใหม่ และ 2. ทำการแก้ไขข้อผิดพลาดที่สำคัญ การทดสอบ Regression Testing ควรทำการทดสอบทุกครั้งที่มีการเปลี่ยนแปลง Code เพื่อ Test Function การทำงานของซอฟต์แวร์

วัตถุประสงค์ของการทดสอบความถดถอย คือ การสร้างความเชื่อมั่นกับระบบที่มีการเปลี่ยนแปลง แก้ไขและพัฒนาว่าสามารถทำงานได้ตามความต้องการของระบบที่กำหนดไว้

วัตถุประสงค์ของการทดสอบความถดถอย มีดังนี้

1. เพื่อรับรองว่าไม่มีข้อบกพร่องที่เกิดขึ้นจากกิจกรรมต่างๆ ในการบำรุงรักษา
2. เพื่อตรวจสอบการแก้ไขข้อบกพร่องหรือเพื่อให้แน่ใจว่า ซอฟต์แวร์ทำงานตามที่ระบุไว้ในข้อกำหนดของการออกแบบระบบ

3. เพื่อรักษาระดับของความน่าเชื่อถือของระบบที่มีการเปลี่ยนแปลงเกิดขึ้น

ขั้นตอนที่สำคัญในการทดสอบความถดถอย คือ การเลือกกรณีทดสอบ ให้พิจารณาการเลือกกรณีทดสอบ ดังนี้

1. กรณีทดสอบที่มีข้อผิดพลาดเกิดขึ้นระหว่างการเขียน โปรแกรม
2. กรณีทดสอบที่มีการเปลี่ยนแปลงของ Code ครั้งล่าสุด
3. กรณีที่ต้องการค้นหาผลกระทบที่เกิดจากการเปลี่ยนแปลงฮาร์ดแวร์

ขั้นตอนการทำงาน Regression Testing

การทำงานทดสอบ Regression Testing มีการดำเนินการทดสอบ ดังต่อไปนี้

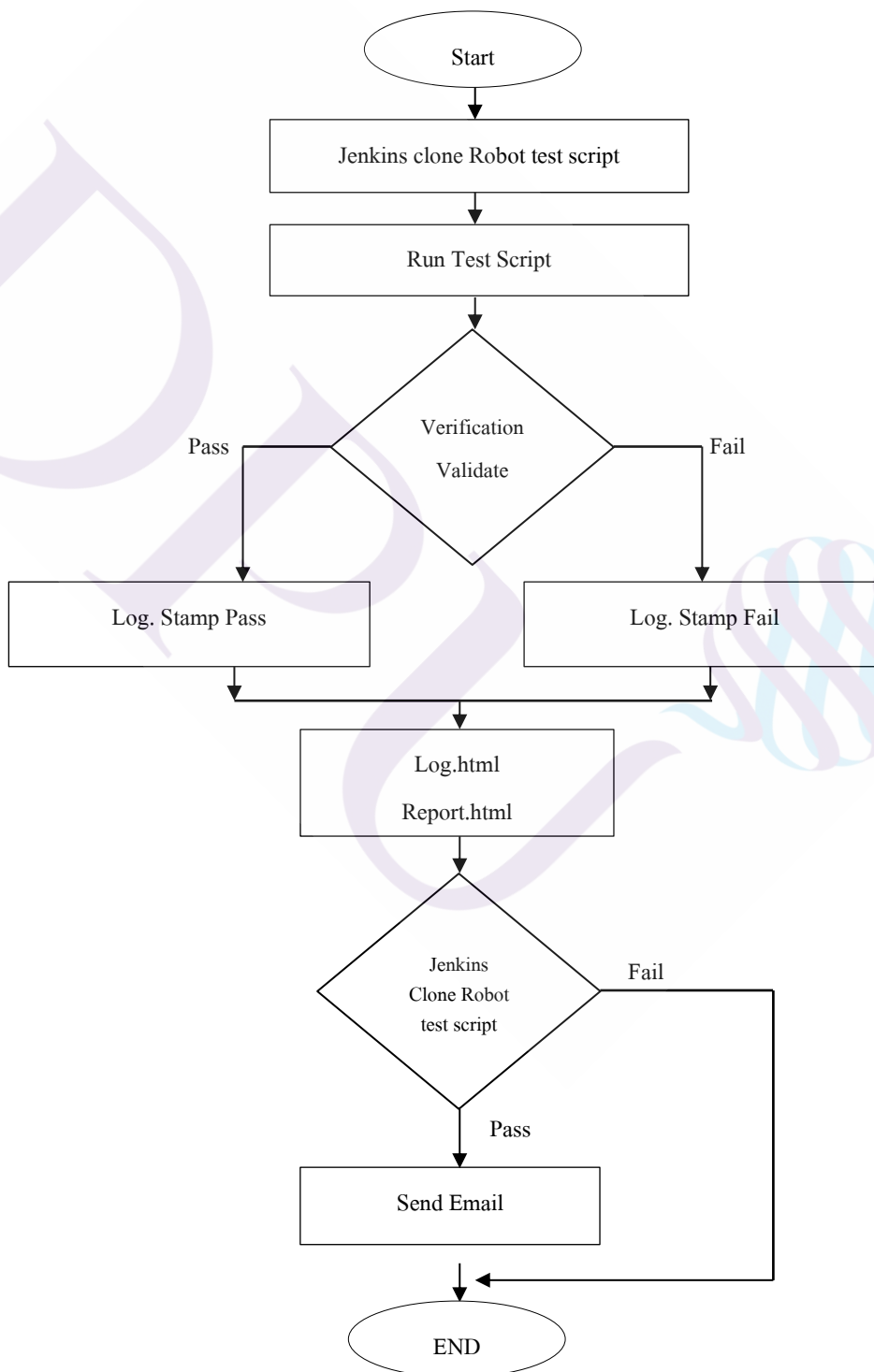
เริ่มต้นจากการใช้ซอฟต์แวร์ Jenkins Jenkins เป็นซอฟต์แวร์โอเพ่นซอร์ส เป็นซอฟต์แวร์ที่ทำงานแบบ continuous integration tools เขียนด้วย ภาษา Java และ support control version แบบ Git และ SVN โดย Jenkins จะทำหน้าที่รวมโค้ด การทำงานทั้งหมดเข้าด้วยกัน และสั่งการทำงานอัตโนมัติตามเวลาที่ตั้งไว้ หลังจากนั้น Jenkins จะทำการ Clone Robot Framework script จาก Git ลงมาเพื่อสั่งรันทำงาน

ขั้นต่อไป Jenkins จะสั่งรัน Robot Framework test script

เมื่อซอฟต์แวร์รันตาม test script แล้วจะทำการ Verification การทวนสอบเพื่อให้ผู้ทดสอบแน่ใจว่า ซอฟต์แวร์ที่ถูกพัฒนาขึ้นเป็นไปตามข้อกำหนดทางเทคนิคที่กำหนดไว้ ต่อจากกระบวนการทวนสอบ ผู้ทดสอบจะทำการ validation การยืนยันผลการตรวจสอบว่า ซอฟต์แวร์สามารถทำงานได้ตามความต้องการของผู้ใช้ กรณีที่ผลการทำการทวนสอบและการยืนยันผลบันทึกผลการทดสอบว่า ผ่าน กรณีที่ผ่าน ซอฟต์แวร์จะทำการบันทึกว่า log stamp pass กรณีที่ไม่ผ่าน ซอฟต์แวร์จะทำการบันทึกว่า log stamp fail แล้วบันทึกผลลง log.html และสรุปผลทั้งหมดลง

ใน report.html ก่อนจะทำการส่งสรุปผลลง Email เพื่อแจ้งผู้เกี่ยวข้องทุกคน ผู้ทดสอบจะทำการตรวจสอบว่า Jenkins สามารถ build Robot Framework test script สำเร็จหรือไม่ เพื่อป้องกันการส่งข้อมูลที่ผิดพลาดให้แก่ผู้เกี่ยวข้อง กรณีที่ผลการตรวจสอบแล้วผ่าน ผู้ดำเนินการทดสอบจะส่ง Email ไปยังผู้เกี่ยวข้อง กรณีที่ไม่ผ่าน จะแจ้งทีมพัฒนาระบบ

ขั้นตอนการทำงานของ Regression Testing



ภาพที่ 2.5 ขั้นตอนการทำงานของ Regression Testing

ที่มา: <https://charathbank.wordpress.com/2010/10/04/qa-knowledge-testing-type/>



บทที่ 3

วิธีการดำเนินการวิจัย

การวิจัยครั้งนี้ ผู้วิจัยมีวัตถุประสงค์เพื่อศึกษาและพัฒนาการใช้งาน Automation Testing Tool ทดสอบซอฟต์แวร์เว็บแอปพลิเคชัน ของบริษัทที่เป็นกรณีศึกษา และศึกษาการใช้ Automation Testing ทดสอบการทำงานของซอฟต์แวร์ช่วยลดเวลาที่ใช้ในการทดสอบซอฟต์แวร์ และใช้ Automation Testing ทำ Regression Testing เพื่อตรวจสอบการทำงานของซอฟต์แวร์ เมื่อมีการแก้ไข Code หรือปรับปรุง Feature ว่า ยังสามารถทำงานได้ตามความต้องการของระบบที่กำหนดไว้ ผู้วิจัยได้กำหนดขั้นตอนการดำเนินการวิจัย เครื่องมือที่ใช้ในการวิจัย และกระบวนการต่าง ๆ ของการวิจัย ดังนี้

3.1 ขั้นตอนการดำเนินการวิจัย



ขั้นตอนการดำเนินการวิจัย

ศึกษาการทดสอบซอฟต์แวร์ของบริษัทที่เป็นกรณีศึกษา บริษัทที่เป็นกรณีศึกษาทำงานเกี่ยวกับการออกแบบและแก้ปัญหาด้านซอฟต์แวร์ สำหรับองค์กรหรือบริษัทธุรกิจความรู้ความเข้าใจเรื่องการทดสอบซอฟต์แวร์จึงมีความสำคัญต่อการทำงานของทีมงานที่ต้องดำเนินการตรวจสอบให้ซอฟต์แวร์ทำงานได้ผลลัพธ์ตรงตามความต้องการที่กำหนดไว้ ค้นหาข้อบกพร่องและค้นพบข้อผิดพลาดที่เกิดขึ้นระหว่างทำการทดสอบ หรือมีการเปลี่ยนแปลงข้อมูล ซึ่งบ่งความแตกต่างระหว่างผลลัพธ์ที่คาดหวัง และผลลัพธ์ที่เกิดขึ้นจริง

ผู้วิจัยศึกษาและทดลองทำ POC (Proof of Concept) ซึ่งหมายถึง การทดสอบการใช้ซอฟต์แวร์หรือการใช้โปรแกรม หรือระบบคอมพิวเตอร์ เพื่อตรวจสอบว่า ซอฟต์แวร์ หรือโปรแกรม หรือระบบนั้นๆ สามารถทำงานได้ถูกต้อง ตรงตามความต้องการของผู้ใช้ก่อนที่ผู้วิจัยจะนำ Robot Framework มาใช้ ผู้วิจัยได้ดำเนินการตามทฤษฎี POC โดยการนำ Robot Framework เข้ามาทดลองใช้งานเบื้องต้น กับซอฟต์แวร์ของบริษัทที่เป็นกรณีศึกษา โดยตั้งเงื่อนไขของซอฟต์แวร์ที่จะนำเข้ามาช่วยทดสอบว่าต้องสามารถทำงานร่วมกับซอฟต์แวร์ที่ใช้ในกรณีทดสอบได้ว่า

1. ต้องทำงานร่วมกับ Fire Fox Browser ได้
2. สามารถแก้ไขหรือเพิ่มเติมการทำงานนอกเหนือจาก Library ที่มีให้ได้
3. มีหน้า summary report
4. สามารถดูรายละเอียดของการรัน test script เป็นข้อๆ ได้
5. สามารถตรวจสอบข้อมูล Database ได้
6. Verify ทำการทวนสอบ เพื่อให้แน่ใจว่า ซอฟต์แวร์ที่ถูกพัฒนาขึ้นเป็นไปตามข้อกำหนดทางเทคนิคที่กำหนดไว้

7. สามารถส่ง capture screen ให้ดูได้หากมีข้อบกพร่องเกิดขึ้น

ผู้วิจัยศึกษาการทำงานของ Robot Framework เพื่อตรวจสอบให้แน่ใจว่า ซอฟต์แวร์สามารถทำงานได้ตรงตามความต้องการแล้วจึงเลือก Robot Framework นำมาตรวจสอบและพัฒนา ระบบซอฟต์แวร์

3.2 อุปกรณ์และเครื่องมือที่ใช้ในการวิจัย

3.2.1 อุปกรณ์ฮาร์ดแวร์ที่จะนำมาใช้

เครื่องคอมพิวเตอร์โน้ตบุ๊ก

- หน่วยประมวลผล : Intel Core i3
- ความเร็ว : 2.5 GHZ

- หน่วยความจำ (RAM) : 8GB
- ฮาร์ดดิสก์ความจุ : 500 GB

3.2.2 ซอฟต์แวร์ที่จะนำมาใช้

- ระบบปฏิบัติการ Microsoft Windows 10 Home 64 bit
- Robot Framework RIDE โปรแกรมสำหรับเขียน Test Scrip และ Keyword
- Firefox เว็บเบราว์เซอร์สำหรับรัน Robot Framework
- Visual Studio Code โปรแกรมสำหรับแก้ไข library Python
- Selenium เป็นซอฟต์แวร์ สำหรับ Automation Testing เพื่อทดสอบเว็บแอปพลิเคชัน
- Robot Framework ซอฟต์แวร์ สำหรับการทำให้ Automation Testing โดยมีรูปแบบ Syntax เป็นภาษาเขียนทำความเข้าใจได้ง่าย
- Python เป็นภาษาพื้นฐานที่ใช้สำหรับเขียน Robot Framework

3.2.3 Robot Framework Library

- Selenium 21. Library ใช้สำหรับเชื่อมต่อกับเว็บแอปพลิเคชัน
- Builtin ใช้สำหรับจัดการส่วนที่เป็นคำหลักทั่วไปที่ต้องการใช้เกี่ยวกับ Python
- Date Time ใช้สำหรับแปลงเวลาและวันที่
- Auto It Library ใช้สำหรับจัดการส่วนที่เกี่ยวข้องกับ Window Application
- Custom Excel XIS Library ใช้สำหรับอ่านเขียนและแก้ไขข้อมูลบน Excel
- Archive Library ใช้สำหรับจัดการ File ที่เป็น zip และ bar
- HttpLibrary.HTTP สำหรับทดสอบ HTTP
- XML ใช้สำหรับสร้างและแก้ไข XML ไฟล์
- Database Library ใช้สำหรับเชื่อมต่อ Database
- Ca Plections ใช้สำหรับสร้างข้อมูลชนิด Python lists and dictionaries

3.3 วิเคราะห์ปัญหาการทดสอบแบบ Manual Testing

วิเคราะห์ปัญหาของการทดสอบซอฟต์แวร์แบบ Manual Testing หลังจากขั้นตอนการพัฒนาซอฟต์แวร์เสร็จแล้ว ผู้ทดสอบจะต้องดำเนินการทดสอบโดยการรัน โปรแกรมและอินพุตข้อมูลเข้าสู่ระบบ เพื่อทดสอบฟังก์ชันการทำงานที่ต้องการและในกรณีที่ซอฟต์แวร์มีการปรับปรุงหรือแก้ไขเพิ่มเติม ผู้ทดสอบจำเป็นต้องทำการทดสอบซ้ำใหม่ตั้งแต่ต้น เพื่อทดสอบและตรวจหาผลกระทบที่มีต่อซอฟต์แวร์ ซึ่งขั้นตอนการทดสอบต้องอินพุตข้อมูลซ้ำตั้งแต่ต้นและใช้เวลา

ค่อนข้างมาก การทำงานซ้ำส่งผลให้ผู้ทดสอบเกิดความเบื่อหน่ายและมีแนวโน้มในการเกิดความผิดพลาดจากตัวผู้ทดสอบจากผลการทำงานวิจัย(ศิริมล เย็นใสว 2558) ผู้วิจัยวิเคราะห์ปัญหาเพื่อแก้ปัญหาที่เกิดขึ้นจากการทำการทดสอบแบบ Manual Testing

วิเคราะห์ปัญหาของการทดสอบซอฟต์แวร์แบบ Manual Testing

ปัญหาด้านคน	<ol style="list-style-type: none"> (1) ผู้ทดสอบขาดความรู้ที่ถูกต้อง เมื่อซอฟต์แวร์มีความซับซ้อนมากขึ้นหรือเพิ่มฟีเจอร์ ผู้ทดสอบจะทำผิดพลาด (2) กรณีที่มีการปรับปรุงซอฟต์แวร์หรือเพิ่มฟีเจอร์ ผู้ทดสอบจะต้องทำการทดสอบข้อมูลใหม่ทั้งหมด ทำให้เกิดกระบวนการป้อนข้อมูลซ้ำ ทำให้งานล่าช้าหรือป้อนข้อมูลผิด (3) การทดสอบแต่ละครั้งทำงานไม่เหมือนเดิม ถ้างานเป็นโปรเจกเดียวกัน เมื่อมีการปรับปรุงแก้ไข ต้องเริ่มต้นงานใหม่ ขั้นตอนเริ่มต้นงานใหม่จนถึงขั้นตอนปรับปรุงแก้ไขต้องทำเหมือนกัน (4) ผู้ทดสอบขาดความรู้ที่ถูกต้อง ป้อนข้อมูลผิด ทำตามสคริปต์ไม่ครบทุกขั้นตอน สื่อสารกับทีมงานผิดพลาด
ปัญหาด้านเครื่องมือ	<ol style="list-style-type: none"> (1) คอมพิวเตอร์ที่ใช้ทำงานช้า เนื่องจากระบบของเทคโนโลยี (2) ฮาร์ดแวร์ไม่อัปเดต
ปัญหาการทดสอบ	<ol style="list-style-type: none"> (1) เอกสารประกอบการทดสอบไม่สมบูรณ์ (2) ขาดความรวดเร็วในการประมวลผล
ปัญหาสภาพแวดล้อม	<ol style="list-style-type: none"> (1) การใช้คนทำงานกรณีมีงานแทรก ทำให้การดำเนินการทดสอบหยุดชะงัก (2) กรณีคนกลางทำให้ไม่มีคนทำงานแทน
ปัญหาวิธีการทำงาน	<ol style="list-style-type: none"> (1) การสื่อสารระหว่างทีมงานไม่ชัดเจน (2) กรณีทีมงานเข้าใจไม่ตรงกันเรื่องภาษาที่ใช้ในการสื่อสาร ส่งผลกระทบต่อประสิทธิภาพการทดสอบซอฟต์แวร์
ปัญหาการจัดการ	<ol style="list-style-type: none"> (1) กรณีซอฟต์แวร์มีฐานข้อมูลขนาดใหญ่ ถ้าทีมงานไม่ออกแบบการเรียงลำดับข้อมูล ส่งผลต่อประสิทธิภาพการทดสอบ (2) ไม่มีตารางเวลาการทดสอบ ทำให้เวลาในช่วงของการทดสอบไม่เพียงพอ

การแก้ปัญหาการทดสอบซอฟต์แวร์ ตามที่วิเคราะห์ไว้แก้ไขได้ด้วยการนำ Automation Testing มาช่วย ปัจจุบันความต้องการของผู้ทดสอบในตลาดอุตสาหกรรมซอฟต์แวร์จะเปลี่ยนจากการทดสอบแบบ Manual Testing เป็นผู้ทดสอบที่มีความสามารถในการใช้ซอฟต์แวร์ที่เป็นเครื่องมือทดสอบอัตโนมัติ เนื่องจากการทดสอบอัตโนมัติ มีความเร็วสูงกว่าและเกิดข้อผิดพลาดได้น้อยกว่าการทดสอบแบบ Manual Testing ซึ่งหมายถึง การลดต้นทุน เวลา และทรัพยากรบุคคล และได้ผลลัพธ์ที่มีประสิทธิภาพ

3.4 เหตุผลที่ใช้ Automation Testing

3.4.1 ผู้วิจัยศึกษาเหตุผลที่ใช้ Automation Testing จากผลงานวิจัย และเอกสารที่เกี่ยวข้องพบว่าวิบูลย์ ชัยจิราภรณ์ (2557) ได้อธิบายเหตุผลที่ต้องใช้ Automation Testing ดังนี้

3.4.1.1 Reliable สามารถทดสอบได้อย่างแม่นยำ ลด Human Error ที่จะเกิดขึ้นเนื่องจากผู้ทดสอบคือข้อมูลผิด

3.4.1.2 Repeatable ทดสอบ Test Case ที่เคยผ่านการทำ Automate Testing ได้ กรณีที่ต้องการรันการทดสอบหลาย ๆ ครั้ง การทดสอบวิธีอัตโนมัติทำได้

3.4.1.3 Reusable นำมาใช้ทดสอบใหม่ได้ ใช้กับ Application ที่เปลี่ยน Version ใหม่หรือเปลี่ยน User Interface

3.4.1.4 Less Time With Fewer Resources ช่วยลดเวลาและลดทรัพยากรในการทำทดสอบ ลดจำนวนผู้ทดสอบ

3.4.1.5 Cost Reduction ลดค่าใช้จ่ายเป็นผลมาจากลดจำนวนผู้ทดสอบ และลดเวลาการทำงาน

3.4.1.6 ทำ Regression Test ได้ การทดสอบความถดถอยเป็นการทดสอบซ้ำ เพื่อตรวจสอบผลกระทบที่เกิดขึ้นจากการแก้ไข หรือเปลี่ยนแปลง หรือพัฒนา

3.4.2 ศึกษาเรื่องการใช้ Automation Testing จากสื่ออิเล็กทรอนิกส์ พบว่า สำหรับบริษัทที่มุ่งเน้นในเรื่องคุณภาพของการผลิตซอฟต์แวร์ สิ่งที่จะขาดไม่ได้ก็คือ การทดสอบระบบเพื่อให้ได้ซอฟต์แวร์ที่ทำงานได้ตรงตามความต้องการของผู้ใช้งานและไม่มีข้อบกพร่องของระบบ ทำให้ผลิตภัณฑ์มีคุณภาพและความพึงพอใจของลูกค้า (<https://www.medium.com//Automation>)

การทดสอบอัตโนมัติเป็นการทำงาน โดยใช้เครื่องมือทดสอบอัตโนมัติมีความเร็วสูงกว่าการทดสอบด้วยมือ เพิ่มความครอบคลุมการทดสอบ ผู้ทดสอบใช้โปรแกรมที่สร้างขึ้น ขณะทำการทดสอบจึงเห็นส่วนของการใช้หน่วยความจำ ข้อมูลทดสอบ สถานะของโปรแกรมที่กำหนดว่าโปรแกรมมีการทำงานตามที่กำหนดไว้ ระหว่างทำการทดสอบแบบอัตโนมัติสามารถประมวลผล

Test Case ที่มีความซับซ้อนและจำนวนมากได้ ส่งผลให้ครอบคลุมการทดสอบสูงกว่าการทดสอบด้วยมือ และการใช้ Automation test ช่วยลดปัญหาการป้อนข้อมูลซ้ำ เมื่อต้องการทำ Regression test หลังจากการแก้จุดบกพร่องหรือพัฒนาซอฟต์แวร์

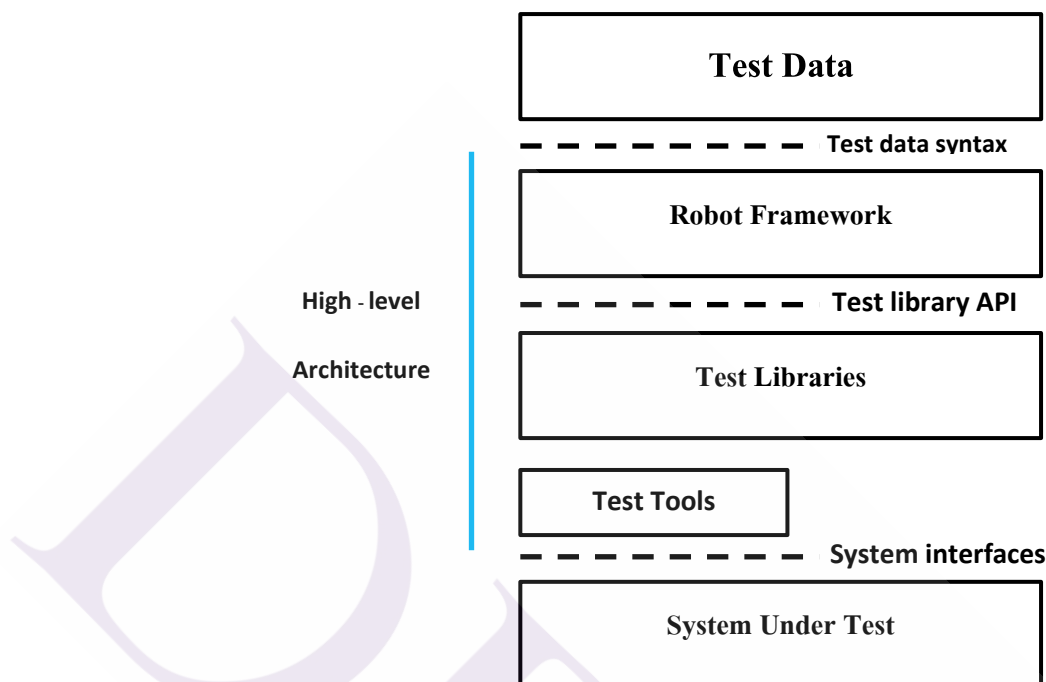
3.4.3 ข้อดีของการทดสอบซอฟต์แวร์อัตโนมัติ

ปัจจุบันการแข่งขันเป็นสิ่งที่สำคัญที่สุดในวงการพัฒนาซอฟต์แวร์ที่ใช้ในการทดสอบการทำงานของซอฟต์แวร์ชนิดต่าง ๆ เพราะมีผู้ใช้งานซอฟต์แวร์ในการบริหารธุรกิจ หรือระบบงานอุตสาหกรรมการผลิต หรือในสำนักงานกันมากขึ้น การสร้างซอฟต์แวร์ขึ้นมา เพื่อใช้งานจึงต้องมีการทดสอบให้ซอฟต์แวร์ใช้งานได้ตรงตามความต้องการของผู้ใช้งาน การนำเครื่องมือการทดสอบอัตโนมัติมาใช้ทดสอบ มีความจำเป็น เมื่อ (1) มีการทดสอบซ้ำหลายครั้ง (2) เมื่อมีการทดสอบความถดถอย (Regression Testing) ซ้ำบ่อยครั้ง (3) เมื่อต้องการทดสอบเว็บแอปพลิเคชัน ที่รองรับผู้ใช้งานจำนวนมากและผู้ใช้มีการเข้าถึงแอปพลิเคชันในเวลาใกล้เคียงกัน การทดสอบแบบนี้ทำด้วยมือไม่ได้ (<https://www.testing-whiz.com/blog>)

รังสิต ศิริรังษี (2557) สรุปข้อดีของการทดสอบอัตโนมัติไว้ดังนี้ (1) ช่วยลดเวลาและค่าใช้จ่ายในการทดสอบ (2) ช่วยเพิ่มความถูกต้องในการทำงาน (3) ช่วยเพิ่มความครอบคลุมการทดสอบ (4) สามารถทำงานที่การทดสอบทำด้วยมือไม่ได้ (5) ช่วยเพิ่มความสามารถของผู้ทดสอบ

การใช้ Automation Testing สามารถสั่งการทำให้ Execute ในช่วงนอกเวลางาน หรือช่วงเวลาดกลางคืนได้ แล้วสามารถตรวจสอบผลลัพธ์ในช่วงเช้าโดยไม่จำเป็นต้องมี Tester คอยเฝ้าการทำงานอยู่กับเครื่องคอมพิวเตอร์

3.5 พัฒนา Robot Frame work สำหรับการทดสอบแบบ Automation Testing

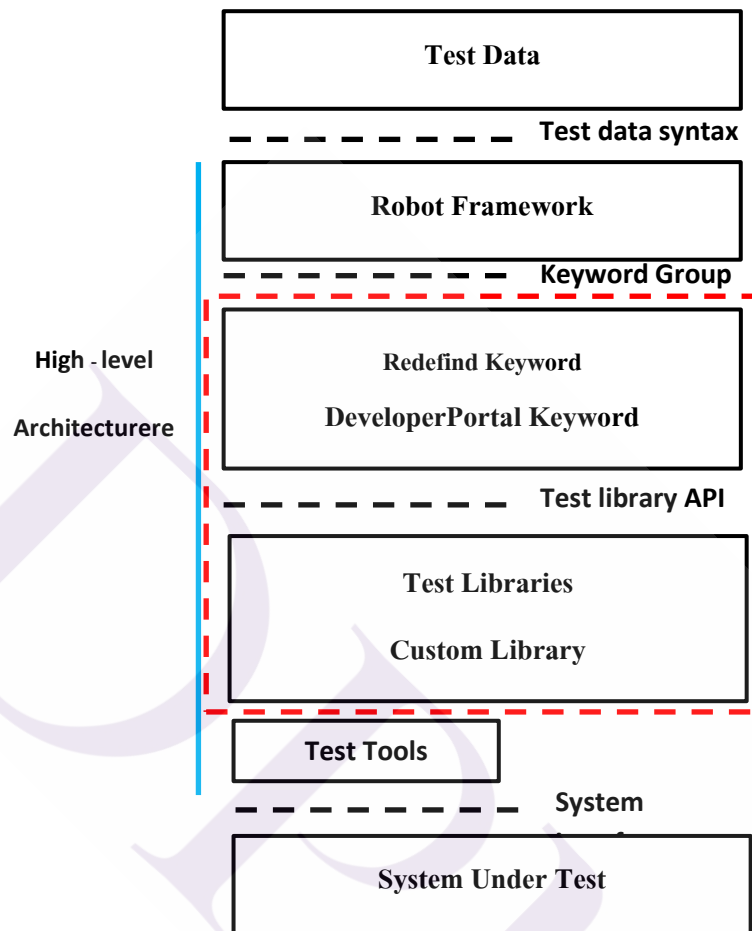


ภาพที่ 3.1 ภาพแสดงโครงสร้างการทำงานของ Robot Framework

ภาพแสดงการทำงานของ Robot Framework ก่อนปรับปรุง เป็นการทำงานโดย Robot Framework นั้นจะเป็นตัวเชื่อมระหว่างการทำงานสองส่วนคือ Test Data เป็นส่วนข้อมูลที่ใช้จัดทำขึ้นเพื่อใช้ในการตรวจสอบความถูกต้องของการทำงานของระบบ และ Test Library เป็นส่วนที่เรียกใช้ Library สำหรับทำงานกับชุดทดสอบ Test Data ก่อนที่จะไปส่งการทำงาน System Under Test

เนื่องจากการทำงานเดิมของ Robot Framework นั้น เป็นกรณีที่ Robot Framework เรียกใช้ Test Library โดยตรง ซึ่ง Library บางส่วนนั้น มีการอัปเดตอยู่ตลอดเวลาทำให้พบปัญหาในการแก้ไข หรือการจัดการ Maintenance นั้นทำได้ยาก เนื่องจากหากมีการ Update หรือแก้ไขนั้น จะส่งผลกระทบต่อ Test Data ของเราทั้งหมด และ Keyword จาก Library ในบางส่วนนั้นไม่สามารถใช้งานได้โดยตรง ต้องมีการใช้ Keyword หลายๆตัวประกอบกันทำให้ ยากต่อการใช้งาน จึงจำเป็นต้องพัฒนา Framework สำหรับ โครงสร้างขึ้นมาใหม่เพื่อช่วยในการจัดการระหว่าง Robot Framework และ Test Library และ ควบรวม Keyword ทั้งหมดเข้ามาใช้งานแทนที่

พัฒนา Robot Framework โดยเพิ่ม Redefind Keyword



ภาพที่ 3.2 ภาพแสดงโครงสร้างการทำงานของ Robot Framework

ภาพแสดงถึงการทำงานของ Robot Framework เป็นชุดข้อมูลที่จะใช้ในการทำการทดสอบ อาทิเช่น Username ,Password และรายละเอียดอื่นๆสำหรับการทดสอบเพื่อให้การทดสอบนั้น เพื่อจำลองการทำงานให้เสมือนการทำงานของ User เข้ามาใช้งานจริงทั้งหมด ในการออกแบบส่วนของ Test Case นั้นมีข้อสำคัญที่ต้องคำนึงถึงคือ การ Reuse ข้อมูลที่เราจะทำการทดสอบ เพื่อทำการ Regression Test Data ที่ Robot Framework Support นั้นมี 4 Format ด้วยกัน ที่มา <http://www.howtoautomate.in.th/robot-framework-how-to-use-test-data/>

1. HTML format (.html)
2. TSV format (tab-separated values .tsv)
3. Plain txt format (.txt)

4. reStructuredText format (.rst,.rest)

ในการ Convert Test case นั้นจะไม่เรียกใช้ Keyword จาก Library โดยตรงเนื่องจากยากต่อการจัดการดูแลในภายหลัง จึงทำการ Group Keyword ขึ้นมาใหม่โดยแบ่งออกเป็น 2 ส่วนหลักๆ ได้แก่

1. Redefind Keyword เป็นกลุ่ม Keyword ที่มีหลักการทำงานคล้ายเดิมแต่ทำการ Group ขึ้นมาใหม่ เพื่อให้ง่ายต่อการใช้งานและการจัดการในภายหลัง

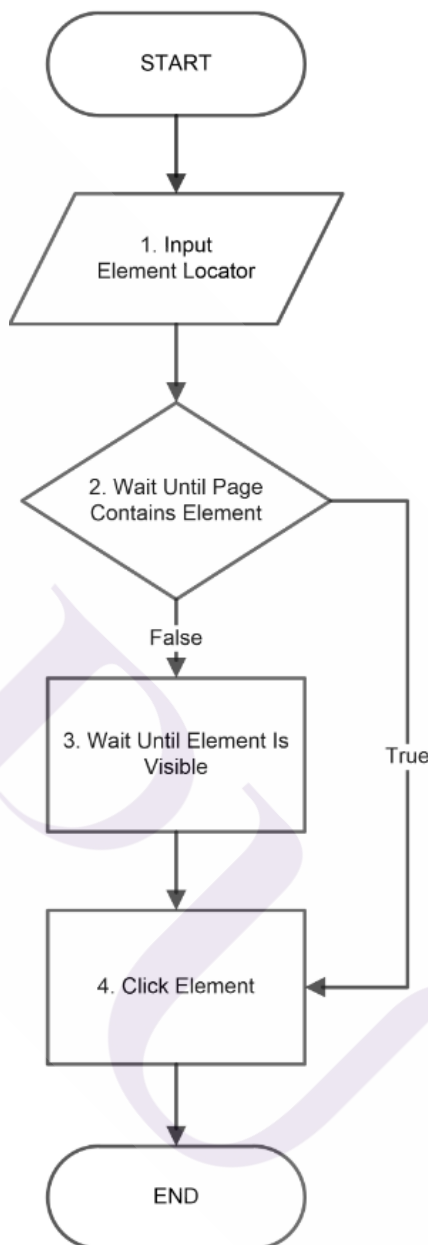
2. Developer Portal Keyword เป็นกลุ่ม Keyword สร้างขึ้นมาเพื่อใช้เฉพาะเจาะจงสำหรับการทำงานให้เข้ากับระบบการทำงานของเว็บไซต์ที่เป็นกรณีทดสอบ เช่น การ Register หรือการคำนวณค่าต่าง ๆ ที่ใช้สำหรับเว็บไซต์ที่เป็นกรณีทดสอบเท่านั้น

ส่วนของ Library นั้นเป็นส่วนของการทำงานพัฒนาขึ้นโดยภาษา Python ภายใน Library นั้นจะประกอบไปด้วย Keyword ต่าง ๆ ที่ใช้เฉพาะทาง เช่น Selenium2Library เป็น Library สำหรับใช้งานเว็บไซต์ โดยจะแบ่งออกเป็น 2 ส่วนหลัก ๆ ได้แก่

1. Test Library เป็นส่วนที่เก็บ Library ที่ดาวโหลดจาก Internal Library ของ Robot Framework เอง และ External Library ที่มีให้ Download ทั่วไป

2. Customer Library เป็น Library ที่ทำขึ้นมาใช้งานเองไม่มีให้ดาวโหลดทั่วไป เมื่อผ่านขั้นตอนการทำงานของ Library เสร็จเรียบร้อย Robot Framework สั่งงานเว็บไซต์ขึ้นมาทำงานในการออกแบบ Keyword

ขั้นตอนการทำงานของ Keyword Click Web Element

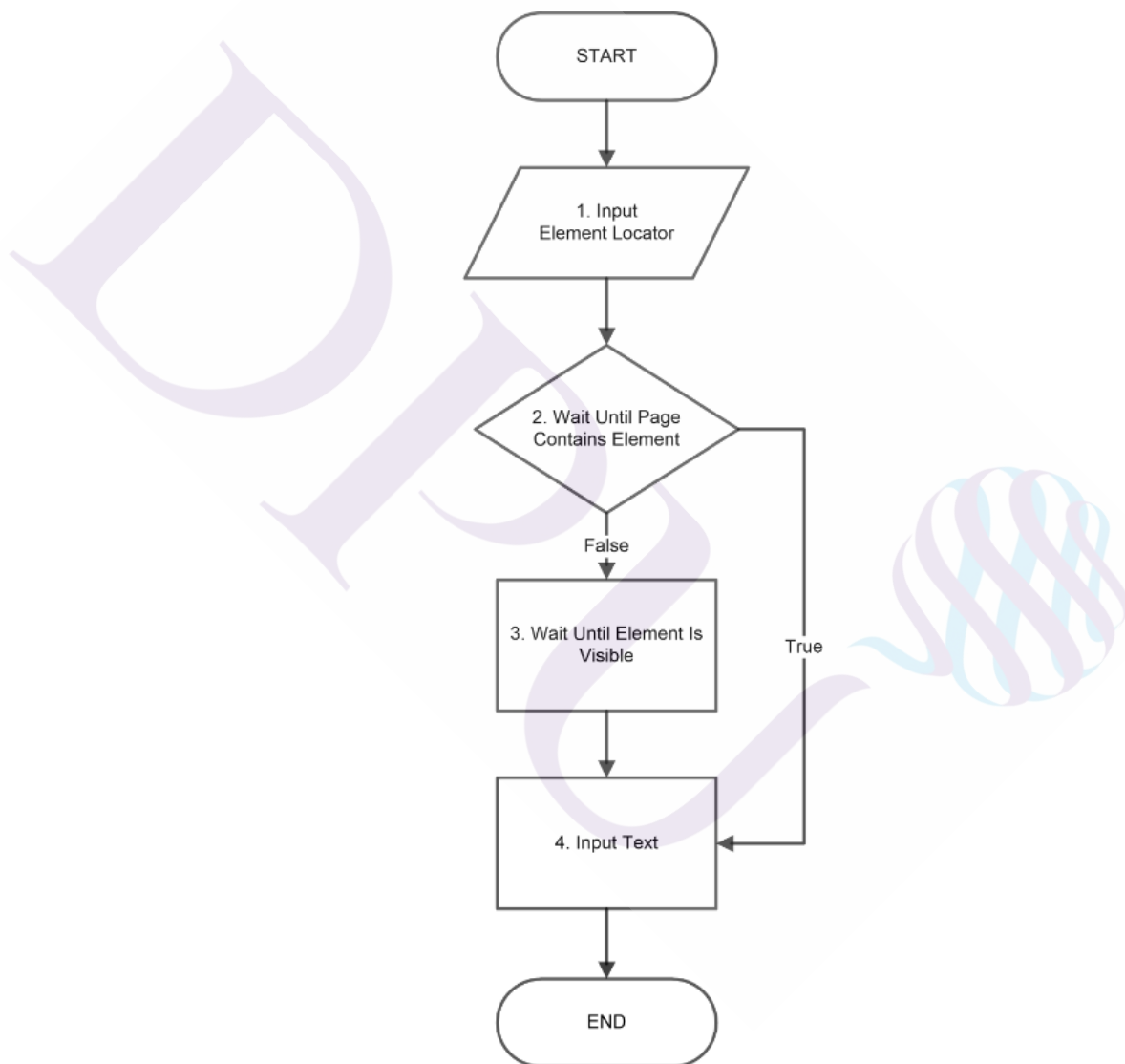


ภาพที่ 3.3 แสดงขั้นตอนการทำงานของ Keyword Click Web Element

1. ผู้ใช้กำหนด ที่อยู่ของ Element Locator ที่ต้องการสั่งการทำงาน
2. รอจนกว่า Element Locator จะแสดงบนหน้า Page จะรีเทินค่าเป็น True , False

3. รอจนกว่า Element Locator จะแสดงบนหน้า Page มีค่าเป็น False Keyword จะทำการ Wait Until Element Is Visible อีกครั้งหนึ่งเพื่อรอให้ Element Locator จะแสดงบนหน้า Page
4. หลังจาก Element Locator จะแสดงบนหน้า Page แล้วระบบจะทำการ Click Web Element และจบการทำงาน

ขั้นตอนการทำงานของ Keyword Input Web Text

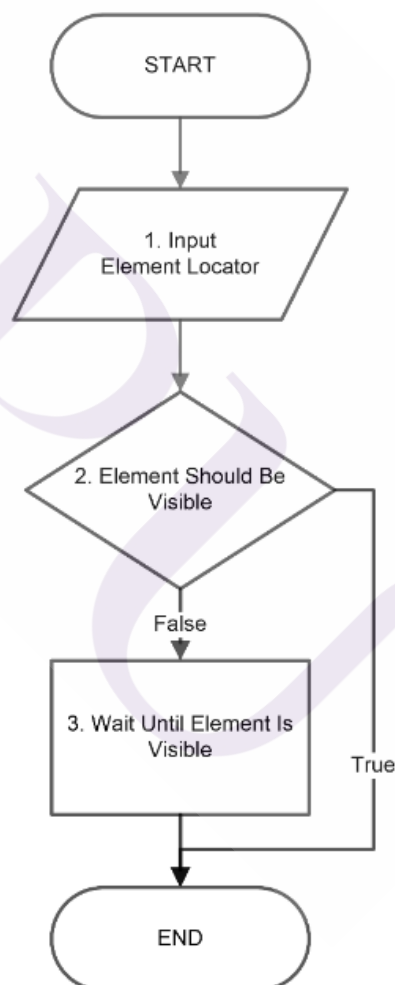


ภาพที่ 3.4 แสดงขั้นตอนการทำงานของ Keyword Input Web Text

จากภาพที่ 8.2 แสดงขั้นตอนการทำงานของ Keyword Input Web Text

1. ผู้ใช้กำหนด ที่อยู่ของ Element Locator ที่ต้องการสั่งการทำงาน
2. รอจนกว่า Element Locator จะแสดงบนหน้า Page จะรีเทิร์นค่าเป็น True , False
3. รอจนกว่า Element Locator จะแสดงบนหน้า Page มีค่าเป็น False Keyword จะทำการ Wait Until Element Is Visible อีกครั้งหนึ่งเพื่อรอให้ Element Locator จะแสดงบนหน้า Page
4. หลังจาก Element Locator จะแสดงบนหน้า Page แล้วระบบจะทำการ Input Text และจบการทำงาน

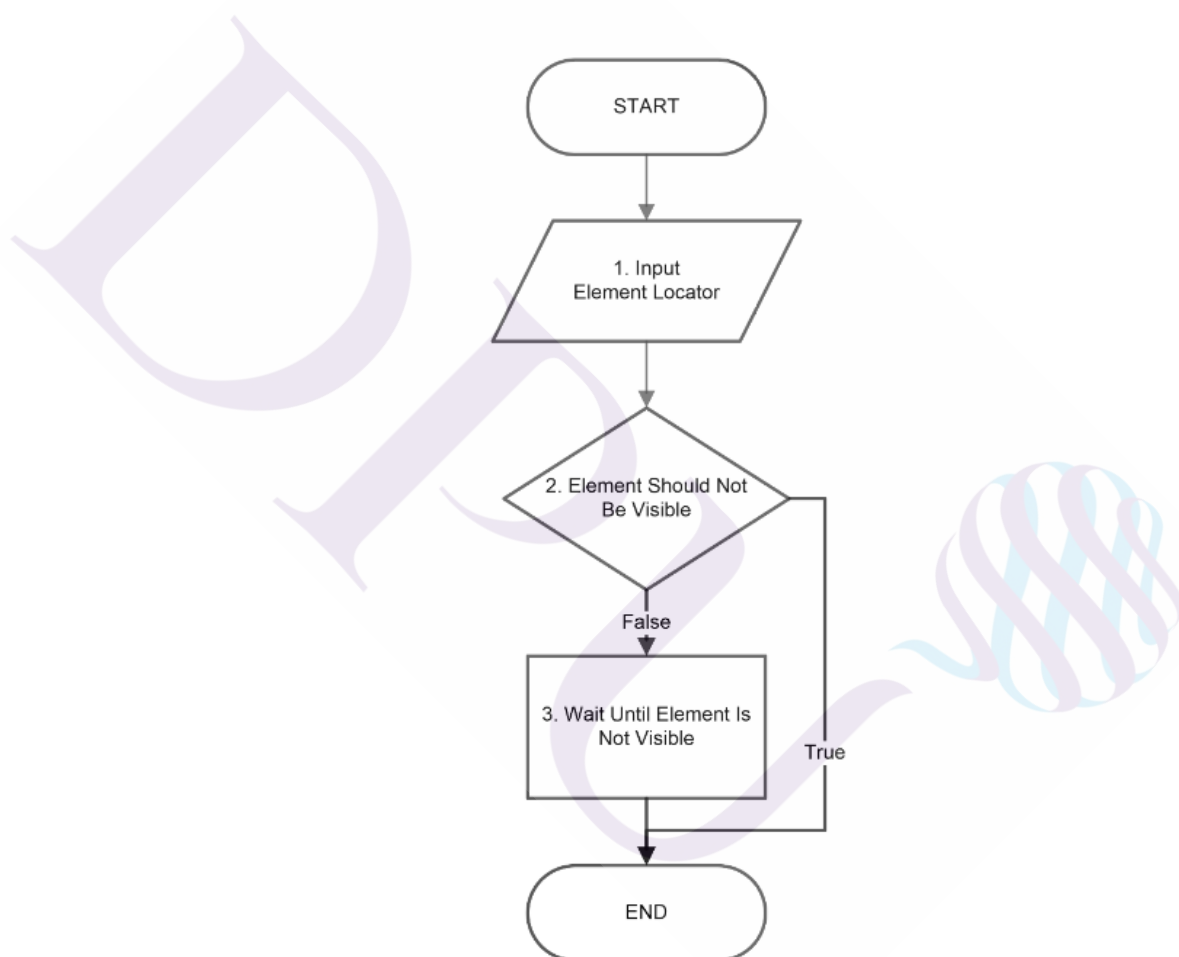
ขั้นตอนการทำงานของ Keyword Web Element Should Be Visible



ภาพที่ 3.5 แสดงขั้นตอนการทำงานของ Keyword Web Element Should Be Visible

จากภาพที่ 8.3 แสดงขั้นตอนการทำงานของ Keyword Web Element Should Be Visible

1. ผู้ใช้กำหนด ที่อยู่ของ Element Locator ที่ต้องการสั่งการทำงาน
2. ตรวจสอบการแสดงผล Element Locator บนหน้า Page จะรีเทิร์นค่าเป็น True , False ถ้าเป็นจริงจะจบการทำงานทันที
3. กรณีตรวจสอบการแสดงผล Element Locator บนหน้า Page เป็น False Keyword สั่ง Wait Until Element Is Visible อีกครั้งเพื่อรอ Element แสดงบนหน้า Page



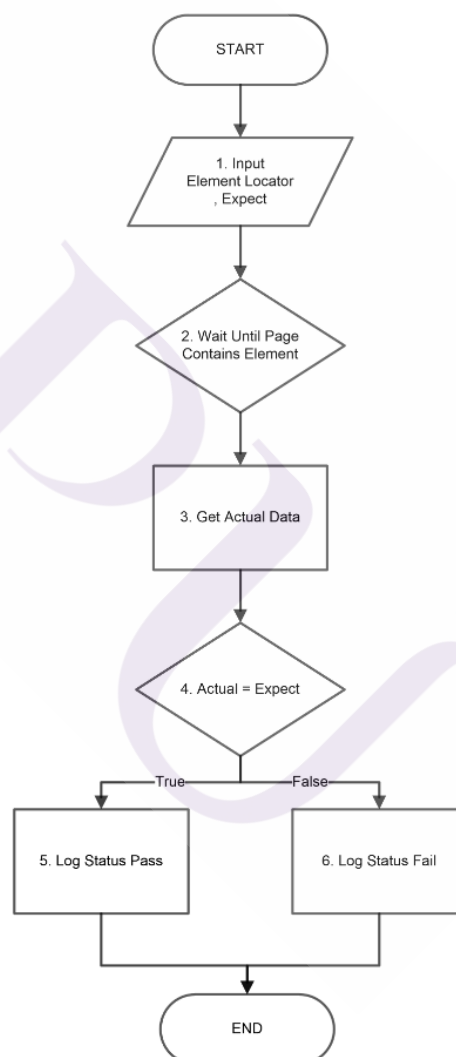
ภาพที่ 3.6 แสดงขั้นตอนการทำงานของ Keyword Web Element Should Be Not Visible

จากภาพที่ 8.4 แสดงขั้นตอนการทำงานของ Keyword Input Web Text

1. ผู้ใช้กำหนด ที่อยู่ของ Element Locator ที่ต้องการสั่งการทำงาน
2. ตรวจสอบ Element Locator จะต้องไม่แสดงบนหน้า Page จะรีเทิร์นค่าเป็น True , False ถ้าเป็นจริงจะจบการทำงานทันที

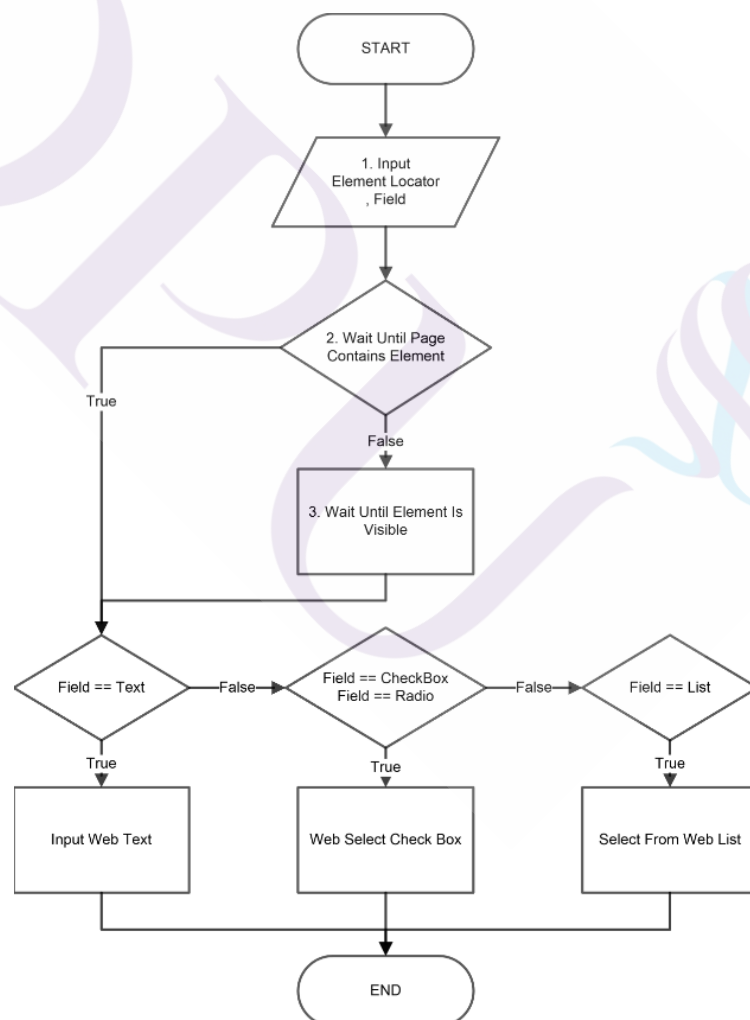
3. กรณีตรวจสอบการแสดงผล Element Locator บนหน้า Page เป็น False Keyword สั่ง Wait Until Element Is Not Visible อีกครั้งเพื่อรอ Element ไม่แสดงบนหน้า Page

ขั้นตอนการทำงานของ Keyword Verify



ภาพที่ 3.7 แสดงขั้นตอนการทำงานของ Keyword Verify

1. ผู้ใช้กำหนด ที่อยู่ของ Element Locator ที่ต้องการสั่งการทำงาน และ Expect เป็นค่าที่คาดหวัง
 2. ตรวจสอบ Element Locator จะต้องแสดงบนหน้า Page
 3. Get Actual Data คือการดึงข้อมูลที่แสดงบนหน้า Page เพื่อนำมาเปรียบเทียบ
 4. เปรียบเทียบข้อมูลที่ได้ระหว่าง ข้อมูลที่แสดงบนหน้า Page ค่า Actual result และ Expected result โดย Keyword จะ Return ค่าเป็น 2 Status คือ Pass และ Fail
 5. ถ้าหากขั้นตอนที่ 4 มีผลเป็น Pass ระบบจะแสดง Log Pass และจบการทำงาน แต่ถ้าหากแสดงผลเป็น Fail Keyword จะแสดง Log Fail และจบการทำงาน
- ขั้นตอนการทำงานของ Keyword Common Input



ภาพที่ 3.8 แสดงขั้นตอนการทำงานของ Keyword Common Input

จากภาพที่ 9 แสดงขั้นตอนการทำงานของ Keyword Verify

1. ผู้ใช้กำหนด ที่อยู่ของ Element Locator ที่ต้องการสั่งการทำงาน
2. รอจนกว่า Element Locator จะแสดงบนหน้า Page จะรีเทิร์นค่าเป็น True , False
3. รอจนกว่า Element Locator จะแสดงบนหน้า Page มีค่าเป็น False Keyword จะทำการ Wait Until Element Is Visible อีกครั้งหนึ่งเพื่อรอให้ Element Locator จะแสดงบนหน้า Page
4. จะทำการเช็คเงื่อนไขตาม Field ที่ต้องการใส่ เพื่อสั่งการทำงาน Keyword ที่จะใช้สำหรับอินพุตให้เหมาะกับ Element Locator นั้นๆ และจบการทำงานของ Keyword

3.6 เปรียบเทียบเวลาทำการทดสอบแบบ Manual Testing และ Automation Testing

การทดสอบซอฟต์แวร์ของบริษัทที่เป็นกรณีศึกษา เป็นการทดสอบแบบ Functional Testing การทดสอบแบบนี้เป็นขั้นตอนที่ถูกสร้างขึ้นมา เพื่อใช้ยืนยันว่าแต่ละส่วนประกอบของระบบทำงานร่วมกันได้อย่างถูกต้อง ตามความต้องการที่ตกลงร่วมกันไว้ระหว่าง Developer และ Customer โดยเน้นที่การทดสอบจาก interface ของระบบงาน ถ้าเป็นระบบเว็บแอปพลิเคชัน ก็เน้นไปที่ user interface บน browser

Regression Testing เป็นการทดสอบชนิดหนึ่งใน Functional Testing ใช้เมื่อทำการ release ระบบหรือเพิ่ม feature ใหม่ เนื่องจากทุกครั้งที่มีการเปลี่ยนแปลงจะเกิดความเสี่ยง เรื่องซอฟต์แวร์นั้น ยังคงทำงานได้ถูกต้อง การทำ Regression Testing ทำให้ค้นหาข้อบกพร่อง ที่เกิดขึ้นจากการแก้ไขหรือเปลี่ยนแปลงได้เร็วขึ้น และแก้ไขปัญหาโดยข้อบกพร่องยังไม่ส่งผลกระทบต่อการทำงานของระบบ

ขั้นตอนแรกของการทำการวิจัย ผู้วิจัยได้ทดสอบซอฟต์แวร์ของบริษัทที่เป็นกรณีศึกษา ใช้การทดสอบแบบ Functional Testing ตาม technical specification จำนวน 20 test cases

ขั้นตอนที่ 2 เลือกผู้ทดสอบซอฟต์แวร์ในทีมงาน 9 คน ทำการทดสอบ 20 test cases ด้วย Manual Testing ทุกคน บันทึกเวลาที่ใช้ในการทดสอบแบบ Manual Testing เวลาที่ทำการทดสอบ คิดเป็นนาที (min.) และวินาที (sec.) ทำตารางเปรียบเทียบเวลา ของผู้ทดสอบทั้ง 9 คน บันทึกเวลาของแต่ละคน คนละ 5 ครั้ง แล้วหาค่าเฉลี่ยของเวลาที่ใช้ในการทดสอบ วิเคราะห์เวลาที่ใช้ในการทดสอบแบบ manual testing จากค่าเฉลี่ย

ขั้นตอนที่ 3 ผู้วิจัยทำ Convert Manual Testing to Automation Test Script แล้ว run Automation Test Script กรณีที่ผ่าน ก็ทำ Rerun Automation Test Script กรณีไม่ผ่านก็กลับไปทำ convert ใหม่ เมื่อ Rerun Automation Test Script ผ่านแล้ว ก็ยอมรับผลการทดสอบ commit to gitlap serve

ดำเนินการทดสอบซอฟต์แวร์แบบ Automation Testing จำนวน 5 ครั้ง บันทึกเวลาที่ใช้ในการทดสอบทุกครั้ง ผู้ทดสอบทำ Regression testing ทุกวัน เพื่อตรวจสอบการทำงานของซอฟต์แวร์ให้แน่ใจว่าไม่มีข้อผิดพลาดเกิดขึ้นจากกิจกรรมการพัฒนาหรือแก้ไขระบบ

แล้วดำเนินการต่อดังนี้

หาค่าเฉลี่ย เวลาที่ได้จากการทดสอบแบบ Automation Testing

ทำตารางเปรียบเทียบ ค่าเฉลี่ยของเวลาที่ใช้ในการทดสอบแบบ Manual Testing และ Automation Testing

วิเคราะห์เวลาที่ใช้ในการทดสอบซอฟต์แวร์แบบ Manual Testing เปรียบเทียบกับเวลาที่ใช้แบบ Automation Testing

ตารางที่ 3.1 ตารางแสดงเวลาการทำทดสอบแบบ Manual Testing 15-03-2018 แสดงเวลาหน่วยเป็น ชั่วโมง:นาที:วินาที

ผู้ทดสอบ No	คนที่ 1	คนที่ 2	คนที่ 3	คนที่ 4	คนที่ 5	คนที่ 6	คนที่ 7	คนที่ 8	คนที่ 9	AVG
1	0:02:47	0:02:39	0:02:11	0:02:21	0:02:30	0:04:03	0:04:35	0:02:37	0:03:35	0:03:02
2	0:02:00	0:02:30	0:02:00	0:02:30	0:02:25	0:04:20	0:05:11	0:03:00	0:04:00	0:03:06
3	0:01:30	0:03:00	0:01:25	0:02:30	0:02:00	0:04:00	0:04:25	0:02:00	0:03:00	0:02:39
4	0:01:30	0:02:00	0:01:30	0:03:00	0:01:59	0:02:00	0:04:50	0:03:30	0:02:00	0:02:29
5	0:02:00	0:03:00	0:01:30	0:01:20	0:01:35	0:04:00	0:05:15	0:02:30	0:02:50	0:02:40
6	0:01:39	0:02:37	0:01:50	0:01:00	0:01:32	0:04:00	0:05:22	0:02:30	0:04:00	0:02:43
7	0:01:36	0:01:39	0:01:30	0:01:00	0:01:50	0:02:30	0:04:00	0:02:00	0:03:30	0:02:11
8	0:01:47	0:01:57	0:01:30	0:01:20	0:01:37	0:02:30	0:04:00	0:02:00	0:03:30	0:02:15

ตารางที่ 1 (ต่อ)

ผู้ทดสอบ No	คนที่ 1	คนที่ 2	คนที่ 3	คนที่ 4	คนที่ 5	คนที่ 6	คนที่ 7	คนที่ 8	คนที่ 9	AVG
9	0:01:30	0:01:30	0:01:30	0:02:00	0:01:23	0:03:00	0:04:00	0:02:00	0:03:50	0:02:18
10	0:04:30	0:03:00	0:02:00	0:05:00	0:03:10	0:02:00	0:05:30	0:03:00	0:02:00	0:03:21
11	0:04:40	0:03:00	0:02:30	0:05:00	0:04:30	0:02:00	0:05:30	0:03:00	0:02:50	0:03:40
12	0:04:50	0:03:00	0:03:00	0:05:00	0:04:54	0:02:00	0:05:30	0:03:00	0:03:00	0:03:48
13	0:05:00	0:03:00	0:02:50	0:05:00	0:05:46	0:02:00	0:05:50	0:03:00	0:03:00	0:03:56
14	0:04:35	0:03:00	0:02:45	0:05:20	0:04:59	0:02:00	0:06:00	0:02:43	0:03:00	0:03:49
15	0:04:35	0:03:00	0:01:55	0:05:20	0:04:58	0:02:30	0:06:00	0:02:20	0:01:50	0:03:36
16	0:04:40	0:03:57	0:04:55	0:05:20	0:05:00	0:02:20	0:06:00	0:02:35	0:03:50	0:04:17
17	0:02:30	0:02:00	0:01:47	0:01:20	0:01:30	0:03:00	0:05:00	0:02:00	0:03:20	0:02:30
18	0:02:15	0:02:30	0:01:45	0:02:10	0:01:30	0:04:00	0:05:20	0:01:30	0:04:40	0:02:51
19	0:02:00	0:03:00	0:02:20	0:02:50	0:02:56	0:03:30	0:04:00	0:02:00	0:03:30	0:02:54
20	0:03:30	0:03:30	0:03:20	0:03:20	0:01:43	0:04:00	0:04:30	0:02:00	0:04:00	0:03:19

จากตารางที่ 1 ผู้ทดสอบ 9 คนทำการทดสอบ Test case คนละ 20 Test case ผู้ทดสอบคนที่ 7 ใช้เวลาในการทดสอบมากที่สุดทุกเคส ผู้ทดสอบที่ใช้เวลาน้อยที่สุดมีหลายคน บางเคสมีคนที่ใช้เวลาน้อยที่สุด 2 คน ระยะเวลาที่ต่างกันมากที่สุด เคส no.6 ผู้ทดสอบใช้เวลาน้อยที่สุด 0:01:00 ผู้ทดสอบใช้เวลามากที่สุด 0:05:22 ต่างกัน 0:04:22 เคสที่ระยะเวลาต่างกันอย่างน้อยที่สุด เคส no.19 ต่างกัน 0:02:00 ผู้ใช้เวลาน้อยที่สุดมีสองคนใช้เวลา 0:02:00 ผู้ที่ใช้เวลามากที่สุดหนึ่งคน 0:04:00

ตารางที่ 3.2 ตารางแสดงเวลาการทำทดสอบแบบ Manual Testing 18-03-2018 แสดงเวลาหน่วยเป็น ชั่วโมง:นาที:วินาที

ผู้ทดสอบ No	คนที่ 1	คนที่ 2	คนที่ 3	คนที่ 4	คนที่ 5	คนที่ 6	คนที่ 7	คนที่ 8	คนที่ 9	AVG
1	0:02:45	0:02:25	0:01:48	0:02:27	0:02:35	0:04:05	0:04:30	0:01:57	0:03:30	0:02:54
2	0:02:12	0:02:20	0:02:11	0:02:27	0:02:25	0:04:20	0:05:11	0:03:00	0:04:31	0:03:11
3	0:01:33	0:03:11	0:02:25	0:02:32	0:02:08	0:04:09	0:04:25	0:02:31	0:03:23	0:02:55
4	0:01:47	0:02:24	0:01:28	0:02:48	0:01:59	0:03:47	0:04:50	0:03:37	0:02:02	0:02:45
5	0:02:12	0:02:58	0:01:08	0:01:25	0:01:35	0:03:57	0:05:15	0:02:33	0:02:47	0:02:39
6	0:01:33	0:01:33	0:01:55	0:01:17	0:01:32	0:04:08	0:05:22	0:02:33	0:03:55	0:02:39
7	0:01:29	0:01:24	0:01:57	0:01:37	0:01:50	0:02:30	0:03:57	0:01:58	0:03:30	0:02:15
8	0:01:28	0:01:30	0:01:53	0:01:22	0:01:37	0:02:31	0:04:09	0:02:20	0:03:30	0:02:16
9	0:01:34	0:01:30	0:01:57	0:02:08	0:01:23	0:02:55	0:03:55	0:02:09	0:03:50	0:02:22
10	0:04:34	0:02:44	0:02:21	0:04:35	0:03:10	0:02:16	0:05:30	0:03:01	0:01:57	0:03:21
11	0:04:44	0:02:57	0:02:35	0:04:57	0:04:30	0:02:07	0:05:21	0:03:30	0:02:53	0:03:44
12	0:04:25	0:03:03	0:03:00	0:04:47	0:04:54	0:02:02	0:05:37	0:03:10	0:02:55	0:03:46
13	0:04:47	0:03:22	0:02:45	0:04:29	0:05:46	0:02:09	0:05:37	0:03:20	0:03:07	0:03:56
14	0:04:28	0:03:11	0:02:43	0:05:12	0:04:59	0:02:15	0:05:27	0:02:43	0:03:21	0:03:49
15	0:04:33	0:03:01	0:01:50	0:05:22	0:04:58	0:02:30	0:05:57	0:02:20	0:01:50	0:03:36
16	0:04:43	0:03:25	0:03:51	0:04:57	0:05:21	0:03:20	0:06:05	0:02:35	0:03:52	0:04:14
17	0:02:32	0:03:17	0:01:41	0:02:20	0:01:37	0:03:03	0:05:11	0:02:09	0:03:20	0:02:48
18	0:02:18	0:02:13	0:01:42	0:02:10	0:01:31	0:04:11	0:05:20	0:01:30	0:04:40	0:02:51
19	0:02:11	0:02:34	0:02:23	0:02:50	0:02:56	0:03:30	0:04:00	0:02:00	0:03:30	0:02:53
20	0:03:21	0:03:45	0:03:11	0:01:20	0:01:43	0:04:00	0:04:30	0:02:00	0:04:00	0:03:06

จากตารางที่ 2 ผู้ทดสอบ 9 คนทำการทดสอบ Test case คนละ 20 Test case ผู้ทดสอบคนที่ 7 ใช้เวลาในการทดสอบมากที่สุดทุกเคส ผู้ทดสอบที่ใช้เวลาน้อยที่สุดมีหลายคน บางเคสมีคนที่ใช้เวลาน้อยที่สุด 2 คน

ระยะเวลาที่ต่างกันมากที่สุด เคส no.14 เวลาต่างกัน 0:04:34

เคสที่ระยะเวลาต่างกันน้อยที่สุด เคส no.19 ต่างกัน 0:01:52

ตารางที่ 3.3 ตารางแสดงเวลาการทำทดสอบแบบ Manual Testing 21-03-2018 แสดงเวลาหน่วยเป็น ชั่วโมง:นาที:วินาที

ผู้ทดสอบ No	คนที่ 1	คนที่ 2	คนที่ 3	คนที่ 4	คนที่ 5	คนที่ 6	คนที่ 7	คนที่ 8	คนที่ 9	AVG
1	0:02:25	0:02:05	0:01:48	0:02:25	0:02:35	0:03:57	0:04:12	0:01:59	0:03:12	0:02:44
2	0:02:05	0:02:09	0:02:11	0:02:22	0:02:25	0:04:10	0:05:05	0:02:57	0:04:28	0:03:06
3	0:01:23	0:03:05	0:02:25	0:02:30	0:02:08	0:03:58	0:04:15	0:02:32	0:03:13	0:02:50
4	0:01:45	0:02:27	0:01:28	0:02:37	0:01:59	0:03:37	0:04:35	0:03:32	0:02:00	0:02:40
5	0:02:02	0:02:38	0:01:08	0:01:23	0:01:35	0:03:47	0:05:27	0:02:23	0:02:45	0:02:34
6	0:01:43	0:01:23	0:01:45	0:01:16	0:01:32	0:04:08	0:05:22	0:02:27	0:03:57	0:02:37
7	0:01:29	0:01:24	0:01:47	0:01:35	0:01:50	0:02:57	0:03:57	0:01:58	0:03:32	0:02:17
8	0:01:32	0:01:30	0:01:43	0:01:20	0:01:31	0:02:21	0:04:19	0:02:12	0:03:33	0:02:13
9	0:01:24	0:01:30	0:01:43	0:02:08	0:01:13	0:02:45	0:03:32	0:02:19	0:03:27	0:02:13
10	0:04:37	0:02:34	0:02:18	0:04:35	0:02:45	0:02:26	0:04:57	0:03:00	0:02:21	0:03:17
11	0:04:24	0:02:57	0:02:25	0:04:57	0:04:30	0:02:11	0:05:11	0:03:37	0:02:31	0:03:38
12	0:04:15	0:03:03	0:03:12	0:04:47	0:04:54	0:01:47	0:05:32	0:03:10	0:02:45	0:03:43
13	0:04:31	0:03:22	0:02:47	0:04:29	0:05:46	0:02:47	0:05:52	0:03:20	0:03:11	0:04:01
14	0:04:18	0:03:21	0:02:45	0:05:12	0:04:59	0:02:17	0:05:01	0:03:43	0:03:01	0:03:51

ตารางที่ 3 (ต่อ)

ผู้ทดสอบ No	คนที่ 1	คนที่ 2	คนที่ 3	คนที่ 4	คนที่ 5	คนที่ 6	คนที่ 7	คนที่ 8	คนที่ 9	AVG
15	0:04:33	0:03:11	0:01:55	0:05:22	0:04:58	0:02:27	0:05:37	0:02:25	0:01:57	0:03:36
16	0:04:23	0:03:45	0:03:55	0:04:57	0:05:21	0:03:20	0:05:59	0:02:34	0:03:39	0:04:13
17	0:02:12	0:03:07	0:01:47	0:02:20	0:01:37	0:03:03	0:05:14	0:02:19	0:03:15	0:02:46
18	0:02:18	0:02:13	0:01:55	0:02:02	0:01:31	0:04:11	0:05:13	0:01:22	0:04:32	0:02:49
19	0:02:21	0:02:24	0:02:20	0:02:50	0:02:56	0:03:30	0:03:53	0:02:01	0:03:21	0:02:51
20	0:03:31	0:03:35	0:03:13	0:01:25	0:01:43	0:03:55	0:04:17	0:02:01	0:03:51	0:03:03

จากตารางที่ 3 ผู้ทดสอบ 9 คนทำการทดสอบ Test case คนละ 20 Test case ผู้ทดสอบคนที่ 7 ใช้เวลาในการทดสอบมากที่สุดทุกเคส ผู้ทดสอบที่ใช้เวลาน้อยที่สุดมีหลายคน บางเคสมีคนที่ใช้เวลาน้อยที่สุด 2 คน

ระยะเวลาที่ต่างกันมากที่สุด เคส no.12 ต่างกัน 0:04:45

เคสที่ระยะเวลาต่างก็น้อยที่สุด เคส no.19 ต่างกัน 0:01:52

ตารางที่ 3.4 ตารางแสดงเวลาการทำทดสอบแบบ Manual Testing 27-03-2018 แสดงเวลาหน่วยเป็น ชั่วโมง:นาที:วินาที

ผู้ทดสอบ No	คนที่ 1	คนที่ 2	คนที่ 3	คนที่ 4	คนที่ 5	คนที่ 6	คนที่ 7	คนที่ 8	คนที่ 9	AVG
1	0:02:22	0:02:15	0:01:38	0:02:15	0:02:25	0:03:47	0:03:59	0:01:37	0:03:09	0:02:36
2	0:02:05	0:02:05	0:02:05	0:02:12	0:02:11	0:04:01	0:04:36	0:02:57	0:04:20	0:02:57
3	0:01:23	0:02:55	0:02:15	0:02:28	0:02:08	0:03:58	0:04:15	0:02:30	0:03:23	0:02:48
4	0:01:43	0:02:25	0:01:37	0:02:25	0:01:59	0:03:32	0:04:35	0:03:18	0:02:03	0:02:37
5	0:02:02	0:02:38	0:01:28	0:01:33	0:01:35	0:03:47	0:04:55	0:02:32	0:02:42	0:02:35
6	0:01:47	0:01:28	0:01:45	0:01:26	0:01:32	0:04:08	0:05:01	0:02:23	0:03:55	0:02:36
7	0:01:27	0:01:22	0:01:45	0:01:37	0:01:45	0:02:52	0:03:52	0:01:53	0:03:30	0:02:14
8	0:01:32	0:01:32	0:01:43	0:01:25	0:01:31	0:02:24	0:04:11	0:02:22	0:03:30	0:02:14
9	0:01:27	0:01:32	0:01:43	0:02:18	0:01:27	0:02:42	0:04:27	0:02:09	0:03:27	0:02:21
10	0:04:33	0:02:32	0:02:38	0:04:35	0:02:45	0:02:26	0:04:52	0:02:59	0:02:21	0:03:18
11	0:04:28	0:02:34	0:02:25	0:04:57	0:04:27	0:02:17	0:05:01	0:03:32	0:02:03	0:03:32
12	0:04:17	0:03:35	0:03:22	0:04:43	0:04:38	0:01:42	0:05:12	0:03:10	0:02:45	0:03:43
13	0:04:27	0:03:17	0:02:42	0:04:21	0:05:37	0:02:42	0:05:39	0:03:15	0:03:17	0:03:55
14	0:04:19	0:03:19	0:02:45	0:04:52	0:04:39	0:02:27	0:05:11	0:03:23	0:03:11	0:03:47
15	0:04:20	0:03:13	0:02:10	0:04:37	0:04:55	0:02:32	0:05:20	0:02:27	0:01:57	0:03:30
16	0:04:12	0:03:40	0:03:52	0:04:52	0:05:37	0:03:20	0:05:59	0:02:32	0:03:39	0:04:11
17	0:02:22	0:03:17	0:02:11	0:02:20	0:01:45	0:03:22	0:05:27	0:02:17	0:03:12	0:02:55
18	0:02:15	0:02:17	0:01:59	0:02:12	0:01:37	0:04:02	0:05:11	0:01:25	0:04:31	0:02:50
19	0:02:19	0:02:26	0:02:21	0:02:51	0:02:56	0:03:35	0:03:55	0:02:01	0:03:21	0:02:52
20	0:03:31	0:03:32	0:03:00	0:01:35	0:01:44	0:03:56	0:04:16	0:02:19	0:03:37	0:03:03

จากตารางที่ 4 ผู้ทดสอบ 9 คนทำการทดสอบ Test case คนละ 20 Test case ผู้ทดสอบคนที่ 7 ใช้เวลาในการทดสอบมากที่สุดทุกเคส ผู้ทดสอบที่ใช้เวลาน้อยที่สุดมีหลายคน บางเคสมีคนที่ใช้เวลาน้อยที่สุด 2 คน

ระยะเวลาที่ต่างกันมากที่สุด เคส no.18 ต่างกัน 0:03:46

เคสที่ระยะเวลาต่างกันน้อยที่สุด เคส no.19 ต่างกัน 0:01:54

ตารางที่ 3.5 ตารางแสดงเวลาการทำทดสอบแบบ Manual Testing 29-03-2018 แสดงเวลาหน่วยเป็น ชั่วโมง:นาที:วินาที

ผู้ทดสอบ No	คนที่ 1	คนที่ 2	คนที่ 3	คนที่ 4	คนที่ 5	คนที่ 6	คนที่ 7	คนที่ 8	คนที่ 9	AVG
1	0:02:10	0:02:10	0:01:28	0:02:00	0:02:25	0:03:47	0:03:57	0:01:47	0:02:57	0:02:31
2	0:02:12	0:02:01	0:02:15	0:02:11	0:02:11	0:03:57	0:04:09	0:02:55	0:04:01	0:02:52
3	0:01:32	0:02:37	0:02:22	0:02:29	0:02:02	0:03:37	0:03:57	0:02:37	0:03:11	0:02:43
4	0:01:43	0:02:31	0:01:31	0:02:10	0:01:59	0:03:22	0:04:12	0:03:11	0:02:13	0:02:32
5	0:02:12	0:02:31	0:01:38	0:01:32	0:01:32	0:03:42	0:04:37	0:02:31	0:02:32	0:02:32
6	0:01:42	0:01:20	0:01:41	0:01:28	0:01:32	0:04:01	0:05:11	0:02:27	0:03:30	0:02:32
7	0:01:29	0:01:21	0:01:42	0:01:32	0:01:42	0:02:32	0:03:47	0:01:55	0:03:29	0:02:10
8	0:01:37	0:01:22	0:01:41	0:01:43	0:01:03	0:02:27	0:04:11	0:02:22	0:03:30	0:02:13
9	0:01:23	0:01:31	0:01:48	0:02:28	0:01:29	0:02:22	0:03:22	0:02:19	0:03:11	0:02:13
10	0:04:32	0:02:38	0:02:37	0:04:39	0:02:39	0:02:22	0:04:51	0:02:52	0:02:17	0:03:16
11	0:04:29	0:02:31	0:02:26	0:04:39	0:04:27	0:02:17	0:04:57	0:03:11	0:02:13	0:03:28
12	0:04:27	0:03:31	0:03:23	0:04:38	0:04:38	0:01:41	0:04:52	0:03:10	0:02:57	0:03:42
13	0:04:17	0:03:02	0:02:59	0:04:20	0:05:32	0:02:41	0:05:41	0:03:11	0:03:12	0:03:53
14	0:04:17	0:03:09	0:02:42	0:04:51	0:04:32	0:02:22	0:05:01	0:03:26	0:03:16	0:03:44

ตารางที่ 5 (ต่อ)

ผู้ทดสอบ No	คนที่ 1	คนที่ 2	คนที่ 3	คนที่ 4	คนที่ 5	คนที่ 6	คนที่ 7	คนที่ 8	คนที่ 9	AVG
15	0:04:23	0:03:17	0:02:17	0:04:29	0:04:33	0:02:31	0:05:12	0:02:23	0:01:42	0:03:25
16	0:04:17	0:03:41	0:03:42	0:04:47	0:05:32	0:03:21	0:05:51	0:02:35	0:03:27	0:04:08
17	0:02:20	0:03:13	0:02:22	0:02:25	0:01:25	0:03:12	0:05:15	0:02:37	0:03:07	0:02:53
18	0:02:12	0:02:01	0:01:51	0:02:11	0:01:37	0:03:57	0:05:17	0:01:37	0:04:23	0:02:47
19	0:02:11	0:02:26	0:02:27	0:02:47	0:02:54	0:03:31	0:03:53	0:02:11	0:03:11	0:02:50
20	0:03:30	0:03:29	0:02:55	0:01:37	0:01:40	0:03:50	0:04:12	0:02:15	0:03:31	0:03:00

จากตารางที่ 5 ผู้ทดสอบ 9 คนทำการทดสอบ Test case คนละ 20 Test case ผู้ทดสอบคนที่ 7 ใช้เวลาในการทดสอบมากที่สุดทุกเคส ผู้ทดสอบที่ใช้เวลาน้อยที่สุดมีหลายคน บางเคสมีคนที่ใช้เวลาน้อยที่สุด 2 คน

ระยะเวลาที่ต่างกันมากที่สุด เคส no.6 ต่างกัน 0:03:51

เคสที่ระยะเวลาต่างกันอย่างน้อยที่สุด เคส no.19 ต่างกัน 0:01:42

จากตารางที่ 1 ถึง ตารางที่ 5 สรุปได้ว่าการทดสอบแบบ Manual Testing ผู้ทดสอบทั้ง 9 คนเป็นผู้ทดสอบซอฟต์แวร์เว็บแอปพลิเคชันของบริษัทที่เป็นกรณีศึกษา ทำการทดสอบ Test case คนละ 20 Test case ด้วยวิธี Manual Testing พบว่า เวลาทดสอบที่แต่ละคนใช้ใน เคสเดียวกันไม่เท่ากัน

จากตารางที่ 1 เก็บเวลาในการทดสอบครั้งที่ 1 ผู้ทดสอบคนที่ 7 ใช้เวลาทดสอบมากกว่าผู้ทดสอบคนอื่นทุกเคส ผู้ทดสอบที่ใช้เวลาน้อยที่สุดมีหลายคน ช่วงเวลาที่ต่างกันมากที่สุด 4 นาที 20 วินาที ช่วงเวลาที่ต่างกันอย่างน้อยที่สุด 2 นาที

จากตารางที่ 2, 3, 4 และ 5 เป็นผลจากการทำการทดสอบครั้งที่ 2, 3, 4 และ 5 ผู้ทดสอบคนที่ 7 คนเดิม กับตารางที่ 1 ใช้เวลาทดสอบมากกว่าผู้ทดสอบคนอื่นทุกเคส ผู้ทดสอบอีก 8 คน แต่ละคนทำเวลาน้อยที่สุดของบางเคสไม่ซ้ำกัน ผู้ทดสอบที่ทำเวลาน้อยที่สุดจะเป็นเคสเดิม เช่น ผู้ทดสอบคนที่ 1 ทำ Test case no.3 น้อยที่สุด ถึง 4 ครั้ง จากการทดสอบ 5 ครั้ง

ช่วงเวลาที่ผู้ทดสอบใช้เวลาทำต่างกันมากที่สุดมีตั้งแต่ 3 นาที 46 วินาที ถึง 4 นาที 45 วินาทีแต่เป็น Test Case no. ไม่ซ้ำกัน

ช่วงเวลาที่ผู้ทดสอบใช้เวลาทำต่างกันน้อยที่สุด มีตั้งแต่ 2 นาที ถึง 1 นาที 42 วินาที และเป็นช่วงเวลาของการทำ Test case no.19 ต่างกันน้อยที่สุดทุกครั้งทั้ง 5 ครั้ง

ตารางที่ 3.6 ตารางรายงานเวลาที่ใช้ในการทดสอบ 15-03-2018 ของการทดสอบแบบ Automation Testing และแบบ Manual Testing

Test case Name	Time take by Automation Testing	Time take by Manual Testing	Duration Minute
TST_F2_0_1_013_ApplyToBePartner_CompanyNameEng_Format	0:00:33	0:03:02	0:02:29
TST_F2_0_1_095_ApplyToBePartner_Marketing_Email_Duplicate	0:00:43	0:03:06	0:02:23
TST_F2_0_1_219_ApplyToBePartner_PhotocopyOfFormal_Format	0:00:51	0:02:39	0:01:48
TST_F2_1_1_231_ApplyToBePartner_TermsOfUsage_General_ClickAgree_Checkbox	0:01:26	0:02:29	0:01:03
TST_F9_0_1_009_Login_General_ClickLogin	0:00:17	0:02:40	0:02:23
TST_F1_1_1_029_Register_ThaiIDCardNoPassportNo_Require	0:00:22	0:02:43	0:02:21
TST_F1_1_1_030_Register_ThaiIDCardNoPassportNo_Length	0:00:21	0:02:11	0:01:50
TST_F1_1_1_031_Register_ThaiIDCardNoPassportNo_Length	0:00:25	0:02:15	0:01:50
TST_F1_1_1_036_Register_Address_Format	0:00:21	0:02:18	0:01:57
TST_F1_1_1_046_Register_TermsOfUsage_ClickCheckbox_Enable	0:00:25	0:03:21	0:02:56
TST_F1_1_1_046_Register_TermsOfUsage_ClickCheckbox_Enable	0:00:25	0:03:40	0:03:15
TST_F1_1_1_046_Register_TermsOfUsage_ClickCheckbox_Enable	0:00:25	0:03:48	0:03:23
TST_F1_1_1_046_Register_TermsOfUsage_ClickCheckbox_Enable	0:00:25	0:03:56	0:03:31
TST_F1_1_1_050_Register_TermsOfUsage_General_ClickAgree	0:00:29	0:03:49	0:03:20
TST_F1_1_1_051_Register_TermsOfUsage_General_ClickClose	0:00:29	0:03:36	0:03:07
TST_F1_1_1_058_Register_TermOfUsage_General_ClickClose	0:00:42	0:04:17	0:03:35
TST_F2_1_1_009_ApplyToBePartner_CompanyProfile_Format	0:00:31	0:02:30	0:01:59
TST_F2_0_1_010_ApplyToBePartner_CompanyProfile_Require	0:00:29	0:02:51	0:02:22
TST_F2_1_1_011_ApplyToBePartner_CompanyProfile_Length	0:00:41	0:02:54	0:02:13
TST_F2_0_1_013_ApplyToBePartner_CompanyNameEng_Format	0:00:33	0:03:19	0:02:46
รวม	0:10:53	1:01:25	0:50:32

จากตารางที่ 6 เวลาที่ใช้ในการทดสอบแบบ Automation Testing น้อยกว่าเวลาที่ใช้ในการทดสอบแบบ Manual Testing ทุกเคส ผลรวมของเวลาที่ใช้ในการทดสอบแบบ Automation Testing เท่ากับ 0:10:53 ผลรวมของเวลาที่ใช้ในการทดสอบแบบ Manual Testing เท่ากับ 1:01:25 เวลาที่ใช้ในการทดสอบแบบ Automation Testing คิดเป็นร้อยละ 17.72 ของเวลาที่ใช้ในการทดสอบแบบ Manual Testing

ตารางที่ 3.7 ตารางรายงานเวลาที่ใช้ในการทดสอบ 18-03-2018 ของการทดสอบแบบ Automation Testing และแบบ Manual Testing

Test case Name	Time take by Automation Testing	Time take by Manual Testing	Duration Minute
TST_F2_0_1_013_ApplyToBePartner_CompanyNameEng_Format	0:00:34	0:02:54	0:02:20
TST_F2_0_1_095_ApplyToBePartner_Marketing_Email_Duplicate	0:00:44	0:03:11	0:02:27
TST_F2_0_1_219_ApplyToBePartner_PhotocopyOfFormal_Format	0:00:51	0:02:55	0:02:04
TST_F2_1_1_231_ApplyToBePartner_TermsOfUsage_General_ClickAgree_Checkbox	0:01:23	0:02:45	0:01:22
TST_F9_0_1_009_Login_General_ClickLogin	0:00:17	0:02:39	0:02:22
TST_F1_1_1_029_Register_ThaiIDCardNoPassportNo_Require	0:00:25	0:02:39	0:02:14
TST_F1_1_1_030_Register_ThaiIDCardNoPassportNo_Length	0:00:24	0:02:15	0:01:51
TST_F1_1_1_031_Register_ThaiIDCardNoPassportNo_Length	0:00:26	0:02:16	0:01:50
TST_F1_1_1_036_Register_Address_Format	0:00:23	0:02:22	0:01:59
TST_F1_1_1_046_Register_TermsOfUsage_ClickCheckbox_Enable	0:00:23	0:03:21	0:02:58
TST_F1_1_1_046_Register_TermsOfUsage_ClickCheckbox_Enable	0:00:23	0:03:44	0:03:21
TST_F1_1_1_046_Register_TermsOfUsage_ClickCheckbox_Enable	0:00:23	0:03:46	0:03:23
TST_F1_1_1_046_Register_TermsOfUsage_ClickCheckbox_Enable	0:00:23	0:03:56	0:03:33
TST_F1_1_1_050_Register_TermsOfUsage_General_ClickAgree	0:00:28	0:03:49	0:03:21
TST_F1_1_1_051_Register_TermsOfUsage_General_ClickClose	0:00:29	0:03:36	0:03:07
TST_F1_1_1_058_Register_TermOfUsage_General_ClickClose	0:00:40	0:04:14	0:03:34
TST_F2_1_1_009_ApplyToBePartner_CompanyProfile_Format	0:00:29	0:02:48	0:02:19
TST_F2_0_1_010_ApplyToBePartner_CompanyProfile_Require	0:00:31	0:02:51	0:02:20
TST_F2_1_1_011_ApplyToBePartner_CompanyProfile_Length	0:00:38	0:02:53	0:02:15
TST_F2_0_1_013_ApplyToBePartner_CompanyNameEng_Format	0:00:32	0:03:06	0:02:34
รวม	0:10:46	1:01:56	0:51:10

จากตารางที่ 7 เวลาที่ใช้ในการทดสอบแบบ Automation Testing น้อยกว่าเวลาที่ใช้ในการทดสอบแบบ Manual Testing ทุกเคส ผลรวมของเวลาที่ใช้ในการทดสอบแบบ Automation Testing เท่ากับ 0:10:46 ผลรวมของเวลาที่ใช้ในการทดสอบแบบ Manual Testing เท่ากับ 1:01:56 เวลาที่ใช้ในการทดสอบแบบ Automation Testing คิดเป็นร้อยละ 17.38 ของเวลาที่ใช้ในการทดสอบแบบ Manual Testing

ตารางที่ 3.8 ตารางรายงานเวลาที่ใช้ในการทดสอบ 21-03-2018 ของการทดสอบแบบ Automation Testing และแบบ Manual Testing

Test case Name	Time take by Automation Testing	Time take by Manual Testing	Duration Minute
TST_F2_0_1_013_ApplyToBePartner_CompanyNameEng_Format	0:00:40	0:02:44	0:02:04
TST_F2_0_1_095_ApplyToBePartner_Marketing_Email_Duplicate	0:00:50	0:03:06	0:02:16
TST_F2_0_1_219_ApplyToBePartner_PhotocopyOfFormal_Format	0:00:54	0:02:50	0:01:56
TST_F2_1_1_231_ApplyToBePartner_TermsOfUsage_General_ClickAgree_Checkbox	0:01:33	0:02:40	0:01:07
TST_F9_0_1_009_Login_General_ClickLogin	0:00:22	0:02:34	0:02:12
TST_F1_1_1_029_Register_ThaiIDCardNoPassportNo_Require	0:00:27	0:02:37	0:02:10
TST_F1_1_1_030_Register_ThaiIDCardNoPassportNo_Length	0:00:26	0:02:17	0:01:51
TST_F1_1_1_031_Register_ThaiIDCardNoPassportNo_Length	0:00:29	0:02:13	0:01:44
TST_F1_1_1_036_Register_Address_Format	0:00:26	0:02:13	0:01:47
TST_F1_1_1_046_Register_TermsOfUsage_ClickCheckbox_Enable	0:00:30	0:03:17	0:02:47
TST_F1_1_1_046_Register_TermsOfUsage_ClickCheckbox_Enable	0:00:30	0:03:38	0:03:08
TST_F1_1_1_046_Register_TermsOfUsage_ClickCheckbox_Enable	0:00:30	0:03:43	0:03:13
TST_F1_1_1_046_Register_TermsOfUsage_ClickCheckbox_Enable	0:00:30	0:04:01	0:03:31
TST_F1_1_1_050_Register_TermsOfUsage_General_ClickAgree	0:00:33	0:03:51	0:03:18
TST_F1_1_1_051_Register_TermsOfUsage_General_ClickClose	0:00:34	0:03:36	0:03:02
TST_F1_1_1_058_Register_TermOfUsage_General_ClickClose	0:00:49	0:04:13	0:03:24
TST_F2_1_1_009_ApplyToBePartner_CompanyProfile_Format	0:00:37	0:02:46	0:02:09
TST_F2_0_1_010_ApplyToBePartner_CompanyProfile_Require	0:00:36	0:02:49	0:02:13
TST_F2_1_1_011_ApplyToBePartner_CompanyProfile_Length	0:00:45	0:02:51	0:02:06
TST_F2_0_1_013_ApplyToBePartner_CompanyNameEng_Format	0:00:40	0:03:03	0:02:23
รวม	0:12:41	1:01:01	0:48:20

จากตารางที่ 8 เวลาที่ใช้ในการทดสอบแบบ Automation Testing น้อยกว่าเวลาที่ใช้ในการทดสอบแบบ Manual Testing ทุกเคส ผลรวมของเวลาที่ใช้ในการทดสอบแบบ Automation Testing เท่ากับ 0:12:41 ผลรวมของเวลาที่ใช้ในการทดสอบแบบ Manual Testing เท่ากับ 1:01:01 เวลาที่ใช้ในการทดสอบแบบ Automation Testing คิดเป็นร้อยละ 20.78 ของเวลาที่ใช้ในการทดสอบแบบ Manual Testing

ตารางที่ 3.9 ตารางรายงานเวลาที่ใช้ในการทดสอบ 27-03-2018 ของการทดสอบแบบ Automation Testing และแบบ Manual Testing

Test case Name	Time take by Automation Testing	Time take by Manual Testing	Duration Minute
TST_F2_0_1_013_ApplyToBePartner_CompanyNameEng_Format	0:00:42	0:02:36	0:01:54
TST_F2_0_1_095_ApplyToBePartner_Marketing_Email_Duplicate	0:00:51	0:02:57	0:02:06
TST_F2_0_1_219_ApplyToBePartner_PhotocopyOfFormal_Format	0:00:58	0:02:48	0:01:50
TST_F2_1_1_231_ApplyToBePartner_TermsOfUsage_General_ClickAgree_Checkbox	0:01:30	0:02:37	0:01:07
TST_F9_0_1_009_Login_General_ClickLogin	0:00:30	0:02:35	0:02:05
TST_F1_1_1_029_Register_ThaiIDCardNoPassportNo_Require	0:00:28	0:02:36	0:02:08
TST_F1_1_1_030_Register_ThaiIDCardNoPassportNo_Length	0:00:26	0:02:14	0:01:48
TST_F1_1_1_031_Register_ThaiIDCardNoPassportNo_Length	0:00:28	0:02:14	0:01:46
TST_F1_1_1_036_Register_Address_Format	0:00:25	0:02:21	0:01:56
TST_F1_1_1_046_Register_TermsOfUsage_ClickCheckbox_Enable	0:00:29	0:03:18	0:02:49
TST_F1_1_1_046_Register_TermsOfUsage_ClickCheckbox_Enable	0:00:29	0:03:32	0:03:03
TST_F1_1_1_046_Register_TermsOfUsage_ClickCheckbox_Enable	0:00:29	0:03:43	0:03:14
TST_F1_1_1_046_Register_TermsOfUsage_ClickCheckbox_Enable	0:00:29	0:03:55	0:03:26
TST_F1_1_1_050_Register_TermsOfUsage_General_ClickAgree	0:00:34	0:03:47	0:03:13
TST_F1_1_1_051_Register_TermsOfUsage_General_ClickClose	0:00:37	0:03:30	0:02:53
TST_F1_1_1_058_Register_TermOfUsage_General_ClickClose	0:00:48	0:04:11	0:03:23
TST_F2_1_1_009_ApplyToBePartner_CompanyProfile_Format	0:00:39	0:02:55	0:02:16
TST_F2_0_1_010_ApplyToBePartner_CompanyProfile_Require	0:00:35	0:02:50	0:02:15
TST_F2_1_1_011_ApplyToBePartner_CompanyProfile_Length	0:00:45	0:02:52	0:02:07
TST_F2_0_1_013_ApplyToBePartner_CompanyNameEng_Format	0:00:38	0:03:03	0:02:25
รวม	0:12:50	1:00:35	0:47:45

จากตารางที่ 9 เวลาที่ใช้ในการทดสอบแบบ Automation Testing น้อยกว่าเวลาที่ใช้ในการทดสอบแบบ Manual Testing ทุกเคส ผลรวมของเวลาที่ใช้ในการทดสอบแบบ Automation Testing เท่ากับ 0:12:50 ผลรวมของเวลาที่ใช้ในการทดสอบแบบ Manual Testing เท่ากับ 1:00:35 เวลาที่ใช้ในการทดสอบแบบ Automation Testing คิดเป็นร้อยละ 21.18 ของเวลาที่ใช้ในการทดสอบแบบ Manual Testing

ตารางที่ 3.10 ตารางรายงานเวลาที่ใช้ในการทดสอบ 29-03-2018 ของการทดสอบแบบ Automation Testing และแบบ Manual Testing

Test case Name	Time take by Automation Testing	Time take by Manual Testing	Duration Minute
TST_F2_0_1_013_ApplyToBePartner_CompanyNameEng_Format	0:00:35	0:02:31	0:01:56
TST_F2_0_1_095_ApplyToBePartner_Marketing_Email_Duplicate	0:00:49	0:02:52	0:02:03
TST_F2_0_1_219_ApplyToBePartner_PhotocopyOfFormal_Format	0:00:55	0:02:43	0:01:48
TST_F2_1_1_231_ApplyToBePartner_TermsOfUsage_General_ClickAgree_Checkbox	0:01:30	0:02:32	0:01:02
TST_F9_0_1_009_Login_General_ClickLogin	0:00:18	0:02:32	0:02:14
TST_F1_1_1_029_Register_ThaiIDCardNoPassportNo_Require	0:00:23	0:02:32	0:02:09
TST_F1_1_1_030_Register_ThaiIDCardNoPassportNo_Length	0:00:22	0:02:10	0:01:48
TST_F1_1_1_031_Register_ThaiIDCardNoPassportNo_Length	0:00:24	0:02:13	0:01:49
TST_F1_1_1_036_Register_Address_Format	0:00:23	0:02:13	0:01:50
TST_F1_1_1_046_Register_TermsOfUsage_ClickCheckbox_Enable	0:00:27	0:03:16	0:02:49
TST_F1_1_1_046_Register_TermsOfUsage_ClickCheckbox_Enable	0:00:27	0:03:28	0:03:01
TST_F1_1_1_046_Register_TermsOfUsage_ClickCheckbox_Enable	0:00:27	0:03:42	0:03:15
TST_F1_1_1_046_Register_TermsOfUsage_ClickCheckbox_Enable	0:00:27	0:03:53	0:03:26
TST_F1_1_1_050_Register_TermsOfUsage_General_ClickAgree	0:00:32	0:03:44	0:03:12
TST_F1_1_1_051_Register_TermsOfUsage_General_ClickClose	0:00:32	0:03:25	0:02:53
TST_F1_1_1_058_Register_TermOfUsage_General_ClickClose	0:00:42	0:04:08	0:03:26
TST_F2_1_1_009_ApplyToBePartner_CompanyProfile_Format	0:00:32	0:02:53	0:02:21
TST_F2_0_1_010_ApplyToBePartner_CompanyProfile_Require	0:00:30	0:02:47	0:02:17
TST_F2_1_1_011_ApplyToBePartner_CompanyProfile_Length	0:00:40	0:02:50	0:02:10
TST_F2_0_1_013_ApplyToBePartner_CompanyNameEng_Format	0:00:33	0:03:00	0:02:27
รวม	0:11:28	0:59:25	0:47:57

จากตารางที่ 10 เวลาที่ใช้ในการทดสอบแบบ Automation Testing น้อยกว่าเวลาที่ใช้ในการทดสอบแบบ Manual Testing ทุกเคส ผลรวมของเวลาที่ใช้ในการทดสอบแบบ Automation Testing เท่ากับ 0:11:28 ผลรวมของเวลาที่ใช้ในการทดสอบแบบ Manual Testing เท่ากับ 0:59:25 เวลาที่ใช้ในการทดสอบแบบ Automation Testing คิดเป็นร้อยละ 19.29 ของเวลาที่ใช้ในการทดสอบแบบ Manual Testing

ค่าเฉลี่ยของเวลาที่ใช้ในการทดสอบซอฟต์แวร์แอปพลิเคชันแบบ Manual Testing มากกว่าเวลาในการทดสอบแบบ Automation Testing อย่างมีนัยสำคัญทุกครั้ง ค่าเฉลี่ยของเวลาที่ใช้ในการทดสอบแบบ Automation Testing คิดเป็นร้อยละ 19.26 ของเวลาที่ใช้ในการทดสอบแบบ Manual Testing เป็นไปตามวัตถุประสงค์ การใช้ Automation Testing ทำให้ลดเวลาที่ใช้ในการทดสอบซอฟต์แวร์แอปพลิเคชัน

3.7 ทำ Regression Testing

เนื่องจากระบบของเว็บไซต์ที่เป็นกรณีทดสอบนั้นต้องทำงานตลอด 24 ชั่วโมงจึงทำให้มีโอกาสทำงานผิดพลาดได้ตลอดเวลา และระบบมีการอัปเดตอยู่อย่างต่อเนื่อง ดังนั้น

ผู้ทำวิจัยดำเนินการทดสอบ ถดถอย ทุกครั้งในทุกๆวันและวันที่มีการเปลี่ยนแปลงเกิดขึ้นกับโค้ดภายในระบบ เพื่อให้แน่ใจว่าระบบส่วนอื่นๆที่ไม่มีการเปลี่ยนแปลงสามารถทำงานร่วมกับส่วนที่มีการเปลี่ยนแปลงได้ถูกต้อง ทำให้ผู้วิจัย สามารถตรวจสอบการแก้ไข และผลกระทบที่เกิดขึ้นจากการแก้ไขของทีมพัฒนา การทดสอบความ ถดถอย ผู้วิจัยดำเนินการโดยใช้เทคนิคการทดสอบแบบ Automation Testing เข้ามาช่วยในการทดสอบระบบในทุกๆวันตามขั้นตอนการทำงาน ของ Regression Testing ขั้นตอนที่สำคัญ คือ ผู้วิจัยได้ทำการ Verification การทวนสอบ Test case เพื่อตรวจสอบว่า ซอฟต์แวร์ที่มีการเปลี่ยนแปลงและทำงานตลอด 24 ชั่วโมงนั้นทำงานได้ตาม ข้อกำหนดที่ระบุไว้ แล้วทำการ Validation การยืนยันผล เพื่อรักษาระบบให้เป็นไปตามความต้องการของผู้ใช้งานได้อย่างถูกต้องและเกิดข้อผิดพลาดน้อยที่สุด

บทที่ 4

ผลการวิจัย

การทำวิจัย เรื่องนี้มีวัตถุประสงค์ เพื่อศึกษาและพัฒนาการใช้งาน Automation Testing ทดสอบซอฟต์แวร์เว็บแอปพลิเคชัน บริษัทที่เป็นกรณีศึกษา และ เพื่อศึกษาการใช้ Automation Testing ช่วยลดเวลาที่ใช้ในการทดสอบเว็บแอปพลิเคชัน

เพื่อให้บรรลุวัตถุประสงค์ ผู้วิจัยได้ดำเนินการทดสอบและพัฒนาโปรแกรมสำหรับทำ Automation Testing ด้วย Robot Framework เพื่อใช้สำหรับสร้างกรณีทดสอบระบบเว็บแอปพลิเคชัน ซึ่งจะนำมาช่วยในการทดสอบแบบ Automation Testing และลดระยะเวลาที่ใช้ในการทดสอบ ผลการพัฒนาระบบ ผู้วิจัยได้ใช้ในการทดสอบซอฟต์แวร์เว็บแอปพลิเคชัน และได้นำเสนอผลการทดสอบระบบในภาคผนวก ก ผู้วิจัยได้พัฒนาโปรแกรมเป็นโปรแกรมสำหรับทำ Automation Testing ด้วย Robot Framework เพื่อใช้สำหรับสร้างกรณีทดสอบระบบเว็บแอปพลิเคชัน

4.1 ผลการออกแบบและพัฒนาระบบ

จากแนวคิดการพัฒนาโปรแกรม สำหรับทำ Automation Testing ด้วย Robot Framework เพื่อใช้สำหรับสร้างกรณีทดสอบระบบเว็บแอปพลิเคชัน จึงได้นำมาสู่การออกแบบ ใช้บันทึกข้อมูลที่เกี่ยวข้อง และพัฒนาระบบ โดยงานวิจัยนี้ได้พัฒนาระบบพัฒนาโปรแกรม สำหรับทำ Automation Testing ด้วย Robot Framework เพื่อใช้สำหรับสร้างกรณีทดสอบระบบเว็บแอปพลิเคชัน ผลที่ได้จากการออกแบบและพัฒนาประกอบไปด้วย ส่วนของการใช้งาน ทั้งหมด 3 ส่วนประกอบไปด้วย

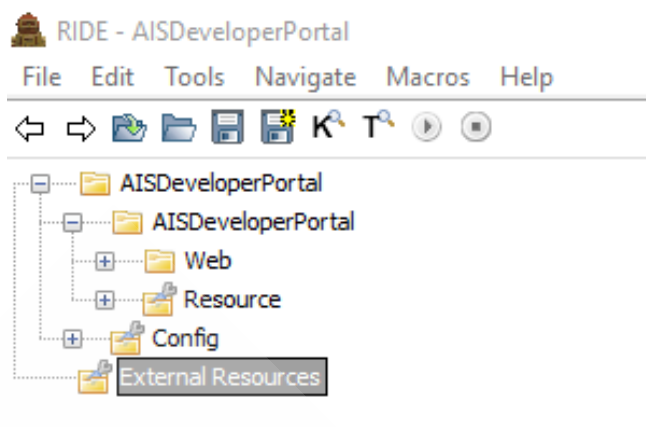
4.1.1. ส่วนของโครงสร้างสำหรับการเก็บข้อมูล

หน้าี่จะเป็นส่วนของโครงสร้างสำหรับการเก็บข้อมูลของ โปรแกรม Automation Testing With Robot Framework เป็นส่วนที่ใช้ในการเก็บข้อมูลจะแบ่งออกเป็น 3 ส่วนหลัก ๆ

4.1.1.1. Web เป็นส่วนที่ใช้ในการเก็บ Test Script สำหรับรัน Automation

4.1.1.2. Resource เป็นส่วนที่ใช้ในการเก็บข้อมูลที่เกี่ยวข้องกับระบบทั้งหมด

4.1.1.3. Config เป็นส่วนที่ใช้ในการเก็บข้อมูลที่จะเปลี่ยนแปลงไปตามเฉพาะเครื่อง



ภาพที่ 4.1 แสดงโครงสร้างการเก็บข้อมูล

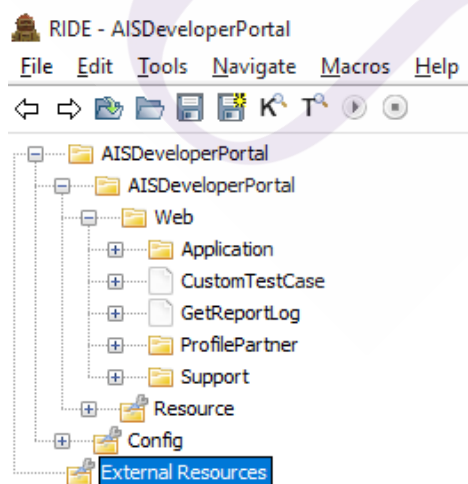
4.1.2 ส่วนที่ใช้ในการเก็บข้อมูล Test case

เป็นส่วนที่ใช้ในการเก็บ Test case สำหรับรัน Automation ภายในจะแบ่งออกเป็น 3 ส่วนหลักๆ

4.1.2.1 จะเป็น Test Case ที่แบ่งตาม Role ที่ใช้ในการทดสอบตามเว็บแอปพลิเคชัน

4.1.2.2 Custom Test Case เป็นส่วนที่เก็บข้อมูลของ Test Script ที่ใช้ในการทดสอบไม่เกี่ยวข้องการทำงานของเว็บแอปพลิเคชัน

4.1.2.3 GetReportLog เป็นส่วนที่ใช้ในการดึงข้อมูลจาก Log ของ Robot Framework ที่ใช้ในการส่ง Email Report



ภาพที่ 4.2 แสดงโครงสร้างการเก็บข้อมูลของ Test cas

4.1.3 ส่วนที่ใช้ในการเก็บ File Resource

Resource เป็นส่วนที่ใช้ในการเก็บข้อมูลที่เกี่ยวข้องกับการทดสอบเว็บแอปพลิเคชันทั้งหมดสามารถแบ่งออกเป็น 6 ส่วนหลักๆ ได้แก่

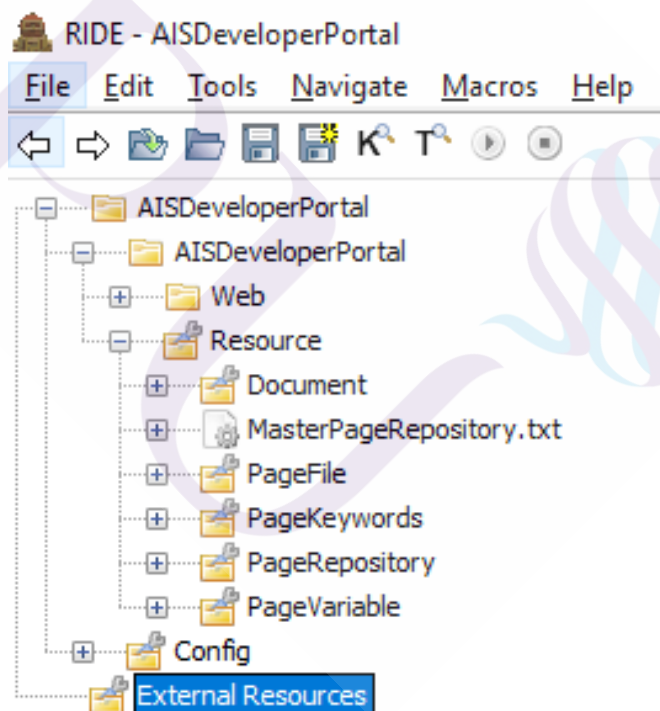
4.1.3.1 Document เป็นส่วนที่เก็บข้อมูลการใช้งานของเว็บแอปพลิเคชัน

4.1.3.2 PageFile เป็นส่วนที่เก็บไฟล์ที่ใช้ในการทดสอบระบบเว็บแอปพลิเคชัน

4.1.3.3 PageKeywords เป็นส่วนที่เก็บ Keyword ที่ใช้สำหรับการทดสอบเว็บแอปพลิเคชัน

4.1.3.4 PageRepository เป็นส่วนที่เก็บข้อมูลของ Xpath ที่บ่งชี้ถึง Element นั้นๆ โดยจะแบ่งเป็นสามส่วนหลักคือ Repository เฉพาะหน้า และ Repository ที่ใช้ร่วมกันทั้งเว็บแอปพลิเคชัน

4.1.3.5 PageVariable เป็นส่วนที่เก็บตัวแปรสำหรับการสร้างกรณีทดสอบเว็บแอปพลิเคชัน



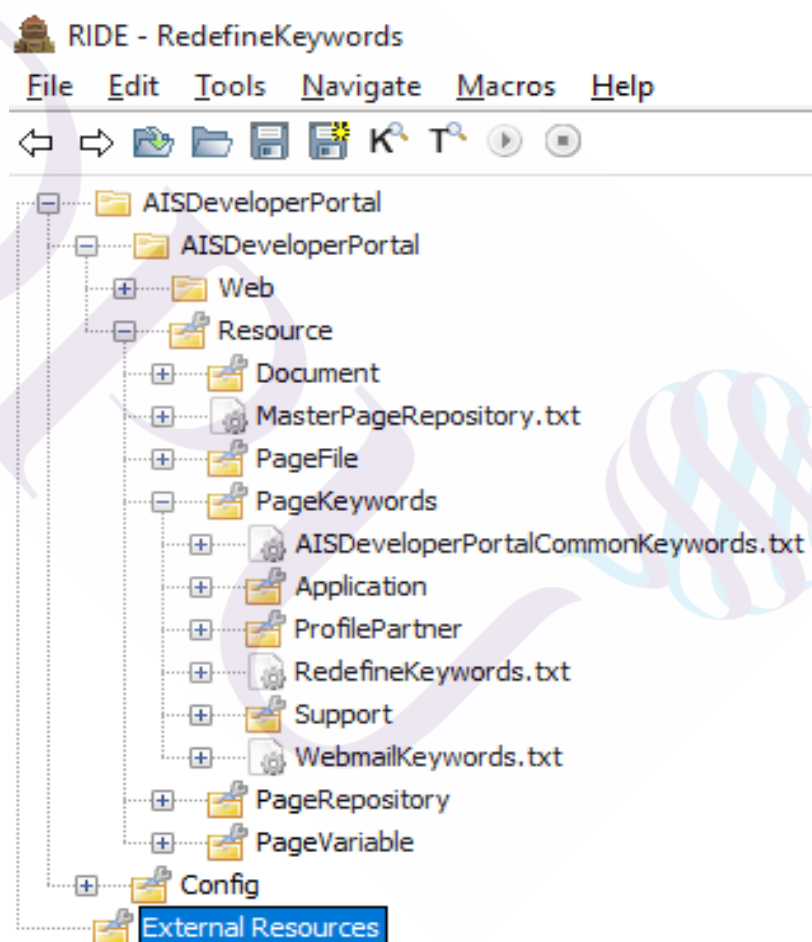
ภาพที่ 4.3 แสดงโครงสร้างการเก็บข้อมูลของ File Resource

4.1.4 RedefineKeywords เป็นส่วนที่เก็บ Keyword การทำงานที่เกี่ยวข้องกับเว็บแอปพลิเคชันทั้งหมดโดยแบ่งออกเป็นสามส่วนหลักๆ ได้แก่

4.1.4.1 AISDeveloperPortalKeyword ที่ทำงานได้เฉพาะหน้าภายในเว็บแอปพลิเคชันที่เป็นกรณีทดสอบ

4.1.4.2 RedefineKeyword เป็น Core Keyword หรือ Keyword ที่ดึงชื่อเดิมมาจาก Library แล้วนำมาทำใหม่

4.1.4.3 ส่วนอื่นๆจะเป็น Keyword ที่ทำขึ้นมาใช้งานเฉพาะหน้าภายในเว็บแอปพลิเคชันเท่านั้น

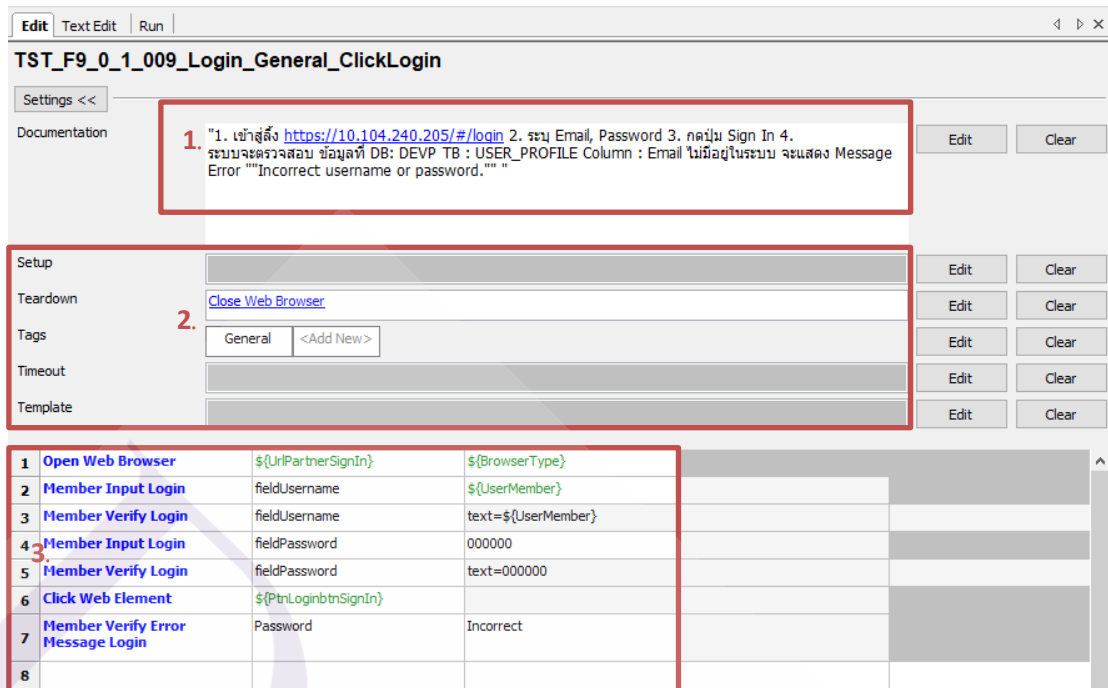


ภาพที่ 4.4 แสดงโครงสร้างการเก็บข้อมูลของ RedefineKeywords

4.1.5 ส่วนของ Keyword ที่ถูกพัฒนาขึ้นจาก Core Keyword ทั้งหมด ดังที่แสดงตามรายชื่อด้านล่าง

Check Exist Database, Check Not Exist Database, Click Web Button, Click Web Element, Click Web Image, Close Firefox Browser, Close Web Browser, Common Input Web Element, Common Split Field And Index, Common Verify Error Message, Common Verify Field, Count Element, Delete All Directory, Delete All File, Delete File In Directory, Double Click Web Element, Download File, Find Xpath, Get All Data From Database, Get Data From Database, Get Locator From Position, Get Row Count From Database, Get Web Text, Get Web Value, Input Web Text, Open Firefox Profile Browser, Open Web Browser, Select From Web List, Select From Web List By Label, Select From Web List By Value, Set Web Window Size, Split Equal String, Unzip File, Upload File, Verify CheckBox, Verify Database, Verify Directory Should Exist, Verify DropDownList, Verify Enable, Verify File Name In Directory, Verify Length, Verify Placeholder, Verify Radio, Verify Text, Wait Web Until Page Contains Element, Web Element Get Matching Xpath Count, Web Element Mouse Over, Web Element Should Be Disabled, Web Element Should Be Enabled, Web Element Should Be Not Visible, Web Element Should Be Visible, Web Element Text Should Be, Web Element Text Should Be, Web Scroll Element Into View, Web Select Checkbox, Web Unselect CheckBox, Get Data From Excel By Name

4.1.6 ผลการพัฒนา Keyword ที่เป็นตัวกลางสำหรับใช้ในการสร้างกรณีทดสอบ ทำให้มีรูปแบบการทำงานที่ไม่ซับซ้อนเข้าใจง่าย สะดวกในการแก้ไขและจัดการในการดำเนินการทดสอบ จากขั้นตอนการพัฒนาที่กล่าวมาข้างต้นนั้น ผู้วิจัยได้นำ Keyword ที่ถูกพัฒนาขึ้นไปใช้ในการสร้างกรณีทดสอบซอฟต์แวร์แอปพลิเคชันได้ในทุก Test Case



ภาพที่ 4.5 แสดงหน้าจอสำหรับการเขียน Test case

4.1.6.1 Documentation เป็นส่วนที่ระบุขั้นตอนการทำงานของ Test case

4.1.6.2 Setup เป็นการสั่งการทำงานก่อนที่จะเริ่มการทำงาน Test case ทุกครั้ง Teardown จะเริ่มสั่งการทำงานหลังจากที่ Test case ทำเสร็จ ในที่นี้ เมื่อ Test case ทำงานเสร็จ ระบบจะสั่ง Close Web Browser ทุกครั้งเพื่อปิดการทำงานของเว็บไซต์ Tag เป็นตัวกรองเพื่อใช้ในการเลือกรันเฉพาะบางส่วน Timeout เป็นตัวตั้งเวลาการทำงานของ Test case เพื่อสั่งให้จบการทำงานของ Test case ทั้งที่ไม่ว่าจะทำงานเสร็จหรือไม่เมื่อครบรอบเวลาระบบจบการทำงานทันที Template เป็นการดึงข้อมูลส่วนอื่นเข้ามาใช้จากภายนอก เช่น Excel file, Text file

4.1.6.3 ส่วนที่ 3 จะเป็นการทำงานของ Test case จะประกอบไปด้วย Keyword ที่สร้างขึ้นมาใช้เป็น Core หลักในที่นี้ ในกรณีของ Test case ข้อนี้ต้องการจะตรวจสอบว่าหากใส่ Username และ Password ผิดระบบจะต้องมีข้อความแสดง Error Incorrect โดยขั้นตอนการทำงานของ Keyword เป็น 7 ขั้นตอนดังนี้

4.1.6.3..1 Open Web Browser ใส่ Url ที่ต้องการและเลือก ชนิดของ Browser

4.1.6.3.2 กรอกข้อมูลที่ช่อง Username

4.1.6.3.3 ตรวจสอบว่ากรอก Username ถูกต้อง

4.1.6.3.4 กรอกข้อมูลที่ช่อง Password

4.1.6.3.5 ตรวจสอบว่ากรอก Password ถูกต้อง

4.1.6.3.6 คลิกปุ่ม Sign in

4.1.6.3.7 ตรวจสอบข้อความ Error Incorrect

4.17 หลังจากทำการรันผลการทดสอบเสร็จสิ้นระบบตัว Robot Framework จะ generate Robot Framework test report และ test log 3 ไฟล์ ไฟล์ที่ 1 คือ Log.html

AISDeveloperPortal Test Log Generated: 20180315 19:14:21 GMT+07:00
68 days 3 hours ago

Test Statistics

Total Statistics	Total	Pass	Fail	Elapsed	Pass / Fail
Critical Tests	17	17	0	00:09:44	██████████
All Tests	17	17	0	00:09:44	██████████

Statistics by Tag	Total	Pass	Fail	Elapsed	Pass / Fail
Duplicate	1	1	0	00:00:44	██████████
Enable	1	1	0	00:00:26	██████████
Format	5	5	0	00:02:50	██████████
General	5	5	0	00:03:24	██████████
Length	3	3	0	00:01:29	██████████
Require	2	2	0	00:00:51	██████████

Statistics by Suite	Total	Pass	Fail	Elapsed	Pass / Fail
AISDeveloperPortal	17	17	0	00:09:50	██████████
AISDeveloperPortal AISDeveloperPortal	17	17	0	00:09:50	██████████
AISDeveloperPortal AISDeveloperPortal.Web	17	17	0	00:09:49	██████████
AISDeveloperPortal AISDeveloperPortal.Web Thesis.TestCase	17	17	0	00:09:49	██████████

Test Execution Errors

20180315 19:13:46:258 WARN Multiple test cases with name 'TST_F2_0_1_013_ApplyToBePartner_CompanyNameEng_Format' executed in test suite 'AISDeveloperPortal.AISDeveloperPortal.Web.Thesis.TestCase'.

Test Execution Log

+ SUITE AISDeveloperPortal

ภาพที่ 4.6 แสดง Robot Framework Log

จากภาพที่ 14 จะเป็นหน้าที่แสดง Report ของ Robot Framework จะประกอบไปด้วย ส่วนที่ 1 Test Statistic แสดงจำนวน Test case ทั้งหมดที่ทำการทดสอบ ส่วนที่ 2 Statistics by Tag แสดงจำนวน Test case ทั้งหมดที่ทำการทดสอบโดยระบบจะทำการกรู๊ปรวม และแยกตามจำนวน Tag ที่ทำการทดสอบ ส่วนที่ 3 Statistics by Suite เป็นการแสดง root ตามแหล่งที่มา ที่ใช้ในการทำการทดสอบ

4.1.8 หลังจากทำการรันผลการทดสอบเสร็จสิ้นระบบตัว Robot Framework จะ generate Robot Framework test report และ test log 3 ไฟล์ ไฟล์ที่ 2 คือ report.html

AISDeveloperPortal Test Report						Generated 20180315 19:14:21 GMT+07:00 68 days 3 hours ago
Summary Information						
Status:	All tests passed					
Start Time:	20180315 19:04:29.702					
End Time:	20180315 19:14:19.316					
Elapsed Time:	00:09:49.614					
Log File:	log-20180315-191419.html					
Test Statistics						
Total Statistics	Total	Pass	Fail	Elapsed	Pass / Fail	
Critical Tests	17	17	0	00:09:44		
All Tests	17	17	0	00:09:44		
Statistics by Tag	Total	Pass	Fail	Elapsed	Pass / Fail	
Duplicate	1	1	0	00:00:44		
Enable	1	1	0	00:00:26		
Format	5	5	0	00:02:50		
General	5	5	0	00:03:24		
Length	3	3	0	00:01:29		
Require	2	2	0	00:00:51		
Statistics by Suite	Total	Pass	Fail	Elapsed	Pass / Fail	
AISDeveloperPortal	17	17	0	00:09:50		
AISDeveloperPortal . AISDeveloperPortal	17	17	0	00:09:50		
AISDeveloperPortal . AISDeveloperPortal . Web	17	17	0	00:09:49		
AISDeveloperPortal . AISDeveloperPortal . Web . Thesis TestCase	17	17	0	00:09:49		

ภาพที่ 4.7 ภาพแสดง Robot Framework Report

จากภาพที่ 15 จะเป็นหน้าที่แสดง Report ของ Robot Framework จะประกอบไปด้วย ส่วนที่ 1 Summary Information แสดง Status การรัน Start Time เวลาเริ่มการทำงาน End Time คือเวลาที่จบการทำงาน Elapsed Time คือเวลาที่ใช้ในการทดสอบทั้งหมด และ Log file ส่วนที่ 2 Test Statistic แสดงจำนวน Test case ทั้งหมดที่ทำการทดสอบ ส่วนที่ 3 Statistics by Tag แสดงจำนวน Test case ทั้งหมดที่ทำการทดสอบโดยระบบ จะทำการกรู๊ปรวม และแยกตามจำนวน Tag ที่ทำการทดสอบ

ส่วนที่ 4 Statistics by Suite เป็นการแสดง root ตามแหล่งที่มา ที่ใช้ในการทำการทดสอบ

เพื่อให้บรรลุวัตถุประสงค์เรื่องการลดเวลา ผู้วิจัยได้คัดเลือกผู้ทดสอบ 9 คน ซึ่งเป็นผู้ทดสอบซอฟต์แวร์เว็บแอปพลิเคชัน ดำเนินการทดสอบ 20 Test Cases คนละ 5 ครั้ง ทุกครั้งที่มีการ Update Test Case แล้วผู้วิจัยนำเวลาที่ผู้ทดสอบใช้ในการทดสอบมาหาค่าเฉลี่ย แสดงไว้ในตารางที่ 1 ถึงตารางที่ 5 และนำค่าเฉลี่ยของเวลาที่ได้นี้ไปวิเคราะห์เปรียบเทียบกับเวลาที่ใช้ในการทดสอบซอฟต์แวร์เว็บแอปพลิเคชันของ 20 Test Case นี้ด้วยแบบ Automation ซึ่งหาได้จากค่าเฉลี่ยการทดสอบ 5 ครั้ง ดังแสดงไว้ในตารางที่ 6 ถึงตารางที่ 10

4.2 วิเคราะห์เปรียบเทียบเวลาที่ใช้ในการทดสอบทั้ง 2 แบบ

การวิเคราะห์เปรียบเทียบเวลาที่ใช้ในการทดสอบซอฟต์แวร์เว็บแอปพลิเคชันของบริษัทที่เป็นกรณีศึกษา ผู้วิจัยได้นำผลของค่าเฉลี่ยของเวลาที่ใช้ในการทดสอบทั้ง 2 แบบ จากตารางที่ 6 ถึงตารางที่ 10 มารายงานเป็นตารางที่ 11

ตารางที่ 4.1 ตารางเปรียบเทียบเวลาที่ใช้ในการทดสอบ

ครั้งที่	Test Case	AVG Time Take by Automation Testing Hr. : min. : sec.	AVG Time Take by Manual Testing Hr. : min. : sec.	Duration
1	กรณีทดสอบ 20 กรณี	0 : 10 : 53	1 : 01 : 25	0:50:32
2	กรณีทดสอบ 20 กรณี	0 : 10 : 46	1 : 01 : 56	0:51:10
3	กรณีทดสอบ 20 กรณี	0 : 12 : 41	1 : 01 : 01	0:48:20
4	กรณีทดสอบ 20 กรณี	0 : 12 : 50	1 : 00 : 35	0:47:45
5	กรณีทดสอบ 20 กรณี	0 : 11 : 28	0 : 59 : 20	0:47:57

ค่าเฉลี่ยของเวลาที่ใช้ในการทดสอบซอฟต์แวร์เว็บแอปพลิเคชันแบบ Manual Testing มากกว่าเวลาในการทดสอบแบบ Automation Testing อย่างมีนัยสำคัญทุกครั้ง ค่าเฉลี่ยของเวลาที่ใช้ในการทดสอบแบบ Automation Testing คิดเป็นร้อยละ 19.26 ของเวลาที่ใช้ในการทดสอบแบบ Manual Testing เป็นไปตามวัตถุประสงค์ การใช้ Automation Testing ทำให้ลดเวลาที่ใช้ในการทดสอบซอฟต์แวร์เว็บแอปพลิเคชัน

ตารางที่ 4.2 แสดงผลการทดสอบ Automation Testing จำนวน 10 ครั้ง

ครั้งที่	จำนวน Test case	ผลการทดสอบ	
		Pass	Fail
1	20	19	1
2	20	20	0
3	20	20	0
4	20	20	0
5	20	18	2
6	20	17	3
7	20	20	0
8	20	20	0
9	20	20	0
10	20	0	20

ตารางที่ 12 แสดงผลการทดสอบ Automation Testing

จากตารางแสดงการทดสอบด้วย Automation Testing พบว่ามีการรันทั้งหมด Pass และ Fail สลับกัน โดยจำนวนครั้งที่ผ่านทั้งหมดจะมี ครั้งที่ 1, 3, 4, 7, 8 และ 9 และพบว่า ครั้งที่ทำการทดสอบแล้วไม่ผ่านทั้ง 20 ข้อมีจำนวน 4 ครั้ง ได้แก่ ครั้งที่ 1, 5, 6 และ 10 โดยในครั้งที่ 1 มี 1 ครั้งที่ ไม่ผ่านเนื่องจาก ไม่สามารถเปิด Web Browser ได้ ครั้งที่ 2 และ 3 ไม่สามารถ Login ได้เนื่องจาก ระบบ Authentication มีปัญหา ส่วนในครั้งที่ 10 นั้นไม่สามารถเข้าระบบได้เนื่องจาก Server ไม่สามารถใช้งานได้

จากผลของการทดสอบด้วยวิธี Automation Testing ผู้วิจัย พบว่า เป็นเครื่องมือที่ช่วยในการทดสอบระบบ โดยมีความสามารถต่าง ๆ ที่การทดสอบแบบ Manual Testing ทำไม่ได้ ดังนี้

1. ใช้ในการเก็บข้อมูลความต้องการของระบบ (Requirement)
2. เก็บวิธีการทดสอบ
3. เก็บเงื่อนไขของการทดสอบ (Test case)
4. เก็บข้อมูลการวางแผนกิจกรรม การทดสอบ (Test Plan and Activity Test Plan)
5. ทดแทนการรัน Test Script ด้วยคน สามารถตั้งเวลา Execute หรือสั่งการทำงาน

แบบ Schedule Run ได้

6. รายงานผลการทดสอบ สถานะ หรือคุณภาพของระบบที่ถูกทดสอบ (Test Result Log)

7. ใช้เป็นเครื่องมือในการติดตามว่า Test Case ครอบคลุมกับข้อกำหนดหรือไม่ โดยตรวจสอบจาก Test Coverage Matrix

8. การทำ Defect Management เพื่อใช้ติดตามข้อบกพร่อง และใช้สื่อสารกันระหว่าง Tester Team และ Developer Team สามารถส่ง Defect ในรูปแบบ Excel ที่เกิดขึ้นไปให้ Programmer ทำให้ลดความเข้าใจคลาดเคลื่อน และทำงานได้สะดวกรวดเร็ว

9. ช่วยในการทำ Regression Testing ได้เร็วกว่าทำด้วยแบบ Manual

4.3 ผลของการทำ Regression Testing

การทดสอบความถดถอย เป็นกิจกรรมหนึ่งในการควบคุมคุณภาพของซอฟต์แวร์ การทดสอบความถดถอยซอฟต์แวร์ภายหลังการแก้ไขระบบถือเป็นวิธีการทดสอบที่สำคัญ เนื่องจากฟังก์ชันการทำงานเพิ่มขึ้นในทุกๆรอบการทำงาน การทดสอบการถดถอยซอฟต์แวร์จึงเกิดขึ้นในทุกๆรอบของการทำงานด้วย

ผู้วิจัยได้ทำการทดสอบความถดถอยซอฟต์แวร์ของบริษัทที่เป็นกรณีศึกษาโดยใช้ข้อมูลการทดสอบจากทีมพัฒนาระบบซึ่งเป็นข้อมูลที่เกี่ยวข้องกับจำนวนการเปลี่ยนแปลงแก้ไขที่เกิดขึ้น ผู้วิจัยได้ใช้การทดสอบแบบอัตโนมัติเพราะเป็นการทดสอบซ้ำโดยใช้ข้อมูลชุดเดียวกันสำหรับทดสอบการทำงานในฟังก์ชันการทำงานที่เพิ่มขึ้น

ผลการทดสอบความถดถอยบรรลุวัตถุประสงค์ดังนี้

1. ผลการทดสอบความถดถอยพบว่าข้อบกพร่องที่พบได้ถูกแก้ไขแล้วทำให้ซอฟต์แวร์ทำงานได้ตามที่ระบุไว้ในข้อกำหนด

2. ไม่มีข้อผิดพลาดในการทำงานของซอฟต์แวร์หลังการแก้ไขเพิ่มเติมโค้ดภายในระบบหรือเปลี่ยนแปลงระบบปฏิบัติการ

3. รักษาคุณภาพของซอฟต์แวร์คือคุณสมบัติและฟังก์ชันการทำงานของซอฟต์แวร์ที่ทำงานได้ตรงตามความต้องการของผู้รับบริการและตามข้อกำหนดที่ระบุไว้ในซอฟต์แวร์

บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

ผลการศึกษาการใช้ Automation Testing ในการทดสอบซอฟต์แวร์แอปพลิเคชัน บรรลุวัตถุประสงค์ การพัฒนาขั้นตอนการทดสอบซอฟต์แวร์เว็บแอปพลิเคชันใช้งานได้กับทุกกรณีทดสอบ และการใช้ Automation Testing ทดสอบซอฟต์แวร์แอปพลิเคชันลดเวลาทดสอบได้จริง ผู้วิจัยเปรียบเทียบค่าเฉลี่ยของการทดสอบทั้ง 2 แบบ จำนวนเวลาที่ได้ค่าเฉลี่ยเวลาของการทดสอบแบบ Automation ใช้เวลาคิดเป็นร้อยละ 19.26 ของเวลาที่ใช้ในการทดสอบแบบ Manual Testing

เมื่อตรวจดูรายการกรณี ซึ่งมี 20 กรณีทดสอบเวลาที่ใช้ในการทดสอบเทียบกันแต่ละกรณีทดสอบ จำนวนเวลาที่ใช้ในการทดสอบแบบ Automation Testing น้อยกว่าเวลาที่ใช้ทดสอบแบบ Manual Testing ทุกกรณีอย่างมีนัยสำคัญ ตารางที่ 6 ถึงตารางที่ 10 เวลาที่เฉลี่ยที่ใช้ในการทดสอบแบบ Automation Testing เรียงลำดับจากครั้งที่ 1 ถึงครั้งที่ 5 เท่ากับ

Test case	เวลาเฉลี่ย				
	ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3	ครั้งที่ 4	ครั้งที่ 5
ค่าเฉลี่ยเวลาที่ใช้ในการทดสอบ 20 กรณีทดสอบ	0 : 10 : 53	0 : 10 : 46	0 : 12 : 41	0 : 12 : 50	0 : 11 : 28

เวลาที่เพิ่มขึ้นครั้งที่ 4 และครั้งที่ 5 อธิบายได้ดังนี้

จากการทดสอบ ทั้ง 5 ครั้ง พบว่าเวลาที่ใช้ในการรันทั้งหมดนั้นไม่เท่ากัน โดยครั้งที่ใช้เวลามากที่สุดคือครั้งที่ 4 โดยใช้เวลาในการทดสอบทั้งหมด 12 นาที 50 วินาที เวลาที่ใช้ในการทดสอบเร็วที่สุดคือ ครั้งที่ 2 ใช้เวลา 10 นาที 46 วินาที จากการทดสอบพบว่า ปัญหาส่วนต่างของเวลานั้น เกิดขึ้นจากการที่ ในทุกๆ 1 Test Case จะมีการ Open Browser และ Login เว็บไซต์ที่เป็นกรณีทดสอบอย่างน้อยหนึ่งครั้งทำให้ เวลาในการเปิดปิดเว็บไซต์ในแต่ละครั้งนั้นไม่เท่ากัน และช่วงเวลาที่ทำการ Login นั้นก็มีผลต่อช่วงเวลาที่ผู้ใช้จำนวนมากสามารถทำให้ล็อกอินช้าลงได้

เวลาที่ใช้ในการทดสอบแบบ Manual Testing ของผู้ทดสอบ แต่ละคนทำการทดสอบกรณีทดสอบจำนวน 5 ครั้งพบว่า เวลาเฉลี่ยรวมที่ได้ไม่แตกต่างกันมากนักเนื่องจากการทำ Manual Testing นั้นเวลาที่ใช้ในการทดสอบแต่ละ Test Case ใช้เวลานานกว่า Automation Testing ที่ใช้เวลา

ค่อนข้างเร็ว ทำให้เวลาในการ Open Browser และ Login เว็บไซต์ที่เป็นกรณีทดสอบและจำนวนผู้ใช้งาน ไม่มีผลต่อเวลาที่ใช้ในการทดสอบได้ ค่าเฉลี่ยของผลรวมทั้ง 5 ครั้งลดลง ตามตารางนี้

Test case	ค่าเฉลี่ย				
	ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3	ครั้งที่ 4	ครั้งที่ 5
ค่าเฉลี่ยเวลาที่ใช้ในการทดสอบ 20 กรณีทดสอบ	1 : 01 : 25	1 : 01 : 56	1 : 01 : 01	1 : 00 : 35	0 : 59 : 25

ข้อสังเกตจากเวลาที่ใช้ในการทดสอบ ผู้ทดสอบที่ใช้เวลาน้อยที่สุด ในแต่ละครั้ง แต่ละกรณีทดสอบ มี 2 หรือ 3 คนสลับกัน บางกรณีทดสอบทำเวลาได้น้อยที่สุด มี 2 คน แต่ผู้ทดสอบที่ใช้เวลามากที่สุด ทุกกรณีทดสอบมีคนเดียว ผู้ทดสอบที่ใช้เวลามากที่สุดเป็นอันดับ 2 มีอีก 1 คน แต่สลับเวลาเป็นเร็วขึ้นได้บ้างบางกรณี

การใช้ Automation Testing ทำให้ลดค่าใช้จ่าย

ตารางแสดงค่าใช้จ่ายและเวลาที่ใช้ในการทดสอบซอฟต์แวร์แบบ Automation Testing ลดลง กรณีทดสอบ 20 กรณีต่อวัน

ลำดับ	ประเด็นที่นำเสนอ	ก่อนปรับปรุง	หลังปรับปรุง	ผลต่าง	หน่วย
1	ค่าแรงพนักงานในการปฏิบัติงาน	76,255.2	0	76,255.2	บาท/ปี
2	ระยะเวลาที่ใช้	14607:36	2811:84	11795.52	นาท/ปี

ผลของการเปรียบเทียบเวลาที่ใช้ในการทดสอบ สองแบบได้ผลเวลาดังนี้ เวลาเฉลี่ยที่ใช้ในการทดสอบ Test case 20 case ด้วยวิธี Automation Testing เฉลี่ย 11:44 นาที/วัน และเวลาเฉลี่ยที่ใช้ในการทดสอบแบบ Manual Testing 20 Test case เดียวกัน เฉลี่ย 60:52 นาที/วัน จำนวนครั้งที่ทำการทดสอบเท่ากัน ถ้าคิดในเวลา 1 เดือนทำงาน 20 วันทำ Automation Testing ใช้เวลา 234:32 นาที/เดือน ทำ Manual Testing ใช้เวลา 1217:28 นาที/เดือนค่าจ้างพนักงาน Outsource เป็นรายวัน วันละ เฉลี่ย 2500 บาท หรือชั่วโมงละ 315 บาท

เวลาทดสอบแบบ Automation Testing 11:44 นาที/วัน คิดเป็นเงิน 60.06 บาท/วัน คิดเป็นเงิน 1201.2 บาทต่อเดือน

เวลาทดสอบแบบ Manual Testing 60:52 นาที/วัน คิดเป็นเงิน 317.73บาท/วัน หรือคิดเป็นเงิน 6354.6 บาท/เดือน

จำนวนเงินของการทดสอบแบบ Manual Testing มากกว่าจำนวนเงินของการทดสอบแบบ Automation Testing

5.2 อภิปรายผล

ผลจากการศึกษาแสดงว่า การทดสอบซอฟต์แวร์แอปพลิเคชัน โดยใช้วิธีการทดสอบแบบ Automation Testing ใช้เวลาเพียงร้อยละ 19.26 ของเวลาที่ทดสอบด้วยแบบ Manual Testing การลดเวลาที่ใช้ในการทดสอบทำให้เกิดผลที่ตามมาคือ ลดแรงงานคน ลดค่าใช้จ่าย และลดเวลาการทำงานทดสอบ และผลลัพธ์ที่ดีอีกข้อหนึ่งคือ ลดความผิดพลาดในการบันทึก ข้อมูลด้วยแรงงานคน เมื่อต้องบันทึกข้อมูลเพื่อการทดสอบเป็นจำนวนมาก หรือบันทึกข้อมูลเดิมซ้ำหลายครั้ง

ผลจากการที่ผู้วิจัยได้ พัฒนาโปรแกรมการใช้งานของ Automation Testing Tool ระหว่างการทดสอบ พบว่ามีข้อดีดังนี้

1. เพิ่มความรวดเร็วในการพัฒนาระบบงาน ทดสอบ
2. ทำให้เกิดความร่วมมือที่ดีระหว่าง Tester Team และ Developer Team จากการทำงานร่วมกันระหว่างที่ ดำเนินทดสอบ
3. ช่วยในการบริหารจัดการ Test Resource และการเชื่อมความสัมพันธ์กันระหว่าง Test Case กับข้อกำหนดที่ได้รับมาจากผู้ใช้งานระบบ
4. เพิ่มความมั่นใจว่า Test Asset ต่าง ๆ ถูกนำมาใช้งานเป็น version ล่าสุด
5. ช่วยวางแผนการทำงานล่วงหน้าตั้งเวลา Execute หรือรัน Test Script ได้

5.3 ข้อเสนอแนะ

ความจำเป็นที่บริษัทหรือองค์กรธุรกิจต้องใช้ซอฟต์แวร์เข้ามาช่วยในการทำงาน เพราะปัจจุบันนี้บริษัทหรือองค์กรธุรกิจ มีความเกี่ยวข้องกับ Business Model ที่ซับซ้อนการบริหารงานจึงต้องใช้ซอฟต์แวร์เป็นเครื่องมือเข้ามาช่วย เพิ่มความเร็วในการทำงาน และเป็นเครื่องมือช่วยจัดการลดความเสี่ยงการนำซอฟต์แวร์เข้ามาใช้งาน

การตรวจสอบการทำงานของซอฟต์แวร์ที่พัฒนาขึ้นใช้งานว่าทำงานได้ถูกต้องตรงตามข้อกำหนดของผู้ใช้จึงมีความสำคัญผู้วิจัยมีข้อเสนอแนะในการเลือก Automation Testing Tool ดังนี้

1. เมื่อมีกรณีทดสอบ (Test Case) เป็นจำนวนมากที่ต้องใช้ในการบริหารงาน

2. ในการบริหารงานต้องทำกรณีทดสอบเดียวกันซ้ำหลายครั้ง ในรอบการทดสอบครั้งเดียว การใช้ Automation Testing จะช่วยให้ไม่ต้องบันทึก ข้อมูลซ้ำ ลดเวลา แรงงาน และค่าใช้จ่าย

3. การใช้ Automation Testing จะช่วยเรื่องการแก้ไขกรณีทดสอบตามสถานการณ์การทดสอบที่ต่างกัน

4. เมื่อต้องทำรายงานเป็นเอกสารหรือการวิเคราะห์งาน การใช้ Automation Testing จะทำได้สะดวกรวดเร็วและถูกต้อง

5. จะได้รายงานผลการทดสอบเป็นการบันทึกข้อมูล รายละเอียดผลการทดสอบว่า Pass หรือ Fail วันที่ เวลาทดสอบ ผู้ทดสอบ

6. รูปแบบการใช้งานที่ยังมีบางส่วนเป็น code อยู่บ้างทำให้ผู้ที่ไม่มีพื้นฐานทางการเขียนโปรแกรมใช้งานค่อนข้างยาก

7. Keyword ยังไม่ครอบคลุมการทำงานทั้งหมด ในบางครั้งยังจำเป็นต้องทำ Keyword เฉพาะทางขึ้นมาใช้งาน

5.4 ข้อเสนอแนะในการทำวิจัยครั้งต่อไป

5.4.1 รูปแบบการใช้งานที่ยังมีบางส่วนเป็น Code อยู่บ้างทำให้ผู้ที่ไม่มีพื้นฐานทางการเขียนโปรแกรมใช้งานค่อนข้างยาก จึงควรทำวิจัยครั้งต่อไปศึกษาการใช้งานที่เกี่ยวข้องกับ Code น้อยลง

5.4.2 Keyword ยังไม่ครอบคลุมการทำงานทั้งหมด ในบางครั้งยังจำเป็นต้องทำ Keyword เฉพาะทางขึ้นมาใช้งาน จึงควรทำวิจัยการพัฒนาการเขียน Keyword ที่ใช้ในการทดสอบ

5.4.3 ศึกษา Automation Testing Tool ที่มีการพัฒนาขึ้นใหม่ที่สามารถใช้ทดสอบซอฟต์แวร์ที่มีประสิทธิภาพในการค้นหาข้อบกพร่อง เพื่อเพิ่มความสามารถในการทดสอบที่เพิ่มขึ้น



บรรณานุกรม

บรรณานุกรม

ภาษาไทย

- การเขียน Test Case IEEE Standard for Software Test Documentations, IEE Std829-1998(Sep 16), IEEE press, Newyork, Ny, 1998.
- ชนพล ลิขณุกฤษฎ์. (2554). *A Development of a software test case management system* (วิทยานิพนธ์ปริญญาโทบริหารธุรกิจ). กรุงเทพฯ: จุฬาลงกรณ์มหาวิทยาลัย.
- ณัฐรัตน์ หาญวรวงศ์. (2556). *การออกแบบและพัฒนากาสร้างกรณีทดสอบซอฟต์แวร์แบบอัตโนมัติโดยใช้โครงสร้าง UI* (วิทยานิพนธ์มหาบัณฑิต). กรุงเทพฯ: จุฬาลงกรณ์มหาวิทยาลัย.
- รังสิต ศิริรังษี. (2557). *การทดสอบซอฟต์แวร์ (Software Testing) (พิมพ์ครั้งที่ 1)*.
เชียงใหม่: โรงพิมพ์เชียงใหม่ นพบุรีการพิมพ์.
- รัชชก ชัยประเสริฐ. (2013). *A test automation framework in POCT system using TDD techniques* (วิทยานิพนธ์ปริญญาโทบริหารธุรกิจ). กรุงเทพฯ: มหาวิทยาลัยมหิดล.
- วิบูลย์ ชัยจิราภรณ์. (2556). *Why Automated Testing*. วารสารจีแม็กแซต, 35(10-12).
- วิบูลย์ ชัยจิราภรณ์. (2557). *การทำ Function Test ด้วย Automated Test Tool*. วารสารจีแม็กแซต, 37(12-14).
- ศศิวิมล เย็นไสว. (2558). *แนวทางการแก้ไขปัญหาการทดสอบซอฟต์แวร์ด้วยวิธี ATDD และการบริหารผลการปฏิบัติงาน* (สารนิพนธ์วิทยาศาสตรมหาบัณฑิต). กรุงเทพฯ: มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ.
- สกรณ์ บุษบง. (2556). *การสร้างกรณีทดสอบแบบเพิ่มทีละหน่วยโดยอัตโนมัติจากกรณีทดสอบระดับหน่วย* (วิทยานิพนธ์ปริญญาโทบริหารธุรกิจ). นครราชสีมา: มหาวิทยาลัยเทคโนโลยี สุรนารี.
- สุรัส ตังไพบุลย์. (2547). *เทคนิคการลดความสูญเสียในโรงงานอุตสาหกรรม*. กรุงเทพฯ: ส.เสริมมิตรการพิมพ์
- Jay Heizer' S Barry Rendr. (2005). *Operation Management*. จินตนิย ไพรสมณฑ์ และคณะ. การจัดการการผลิตและการปฏิบัติการ (พิมพ์ครั้งที่ 10). กรุงเทพฯ: เพียร์สัน เอ็ดดูเคชั่น อินโดไชน่า

ภาษาต่างประเทศ

Acceptance Test Driven Development. สืบค้นวันที่ 1 กุมภาพันธ์ 2560, จาก

<http://agilethailand.wordpress.com/2012>

Alazar Seydum Hail. (2011). Automation of test cases for web applications automation of CRM test cases. Helsinki Metropolia University of Applied Sciences..

Automation Testing. . สืบค้นวันที่ 1 กุมภาพันธ์ 2560, จาก <https://www.testing-whiz.com/blog>

Automation Testing. สืบค้นวันที่ 1 กุมภาพันธ์ 2560, จาก <http://red.badger.com/blog/>

Automation Testing. สืบค้นวันที่ 1 กุมภาพันธ์ 2560, จาก <https://www.medium.com//Automation>

Elisabeth Hendrickson. (2008). Driving Development with Tests; ATDD and TDD. Quality

Tree Software, Inc.,

Functional Testing Process. สืบค้นวันที่ 6 กุมภาพันธ์ 2560,

จาก <http://sdllcservices.com/functional-testing.html>

Functional Testing. สืบค้นวันที่ 6 กุมภาพันธ์ 2560, จาก

<https://charathbank.wordpress.com/2010/10/04/qa-knowledge-testing-type/>

Filip Kiss. (2017). Analysis of Lean Software Development. Masaryk University. Czech Republic.

Jusaj Huska. (2012). Automated testing of the component best web application user interface.

Masaryk University. Czech Republic.

Phat Chau. Tan. (2016). Automation testing with robot framework. Helsinki Metropolia

University of Applied Sciences.

Robot Framework Architecture and Test Flow. สืบค้นวันที่ 6 มีนาคม 2560, จาก

[http://robotframework.org/#documation\[online\]](http://robotframework.org/#documation[online])

Robot Framework. สืบค้นวันที่ 6 มีนาคม 2560, จาก <http://robotframework.org/#tools>

Robot Framework. สืบค้นวันที่ 6 มีนาคม 2560, จาก <https://networks.nokia.com/>

Wikipedia, The Free Encyclopedia

<https://en.wikipedia.org/wiki//Automationtesting>

<https://en.wikipedia.org/wiki//SoftwareTesting>

<https://en.wikipedia.org//wili/ManualTesting>



ภาคผนวก

ID	Test Case Number	Testcase Name	Testcase Description	I want to... (Test Step)	So That... (Expected Result)
1	TST_F2_0_1_013	TST_F2_0_1_013_ApplyToBePartner_CompanyNameEng_Format	Verify parameters ตรวจสอบ Format ของ Field: Company Name (Eng) กรณี ระบุข้อมูล ที่ไม่ใช่ ตัวอักษรภาษาอังกฤษ ตัวเลข อักขระพิเศษดังนี้ : @ # \$ % & * () _ - . : ; ," และ space	1. เข้าลิ้งค์ https://10.104.240.205/#/login 2.เข้าสู่ระบบ (Sign in) 3.Click link 'Apply to be Partner' 3.ระบบแสดงหน้า Apply to be Partner 4.Check Format text Field :Company Name (Eng)	ระบบแสดง Error Message Error Code : "DEVP0114" Error Message : "Company Name (Eng) must be contained Eng characters and number only."
2	TST_F2_0_1_095	TST_F2_0_1_095_ApplyToBePartner_Marketing_Email_Duplicate	Verify parameters Department :Marketing ตรวจสอบ Field: Email กรณีระบุ Email ซ้ำกัน ใน Department เดียวกัน	1. เข้าลิ้งค์ https://10.104.240.205/#/login 2.เข้าสู่ระบบ (Sign in) 3.Click link 'Apply to be Partner' 4.ระบบแสดงหน้า Apply to be Partner 5.กรอกข้อมูล Field: Email	ระบบแสดง Error Message Error Code : "DEVP0129" Error Message :" Email is Duplicated."
3	TST_F2_0_1_219	TST_F2_0_1_219_ApplyToBePartner_PhotocopyOfFormal_Format	Verify parameters ตรวจสอบ Field: Photocopy of Formal Company กรณี Upload File ที่ ไม่ใช่ PDF	1. เข้าลิ้งค์ https://10.104.240.205/#/login 2.เข้าสู่ระบบ (Sign in) 3.Click link 'Apply to be Partner' 4.ระบบแสดงหน้า Apply to be Partner 5.ตรวจสอบ Upload Document 6.cilck 'Choose File'	ระบบแสดง Error Message Error Code : "DEVP0111" Error Message : "Please select file in PDF format."

4	TST_F2_1_1_231	TST_F2_1_1_231_ApplyToBePartner_TermsOfUsage_General_ClickAgree_Checkbox	Verify parameters ตรวจสอบ กรณี Click link 'Terms of Usage'	<ol style="list-style-type: none"> 1. เข้าลิ้งค์ https://10.104.240.205/#/login 2.เข้าสู่ระบบ (Sign in) 3.Click link 'Apply to be Partner' 3.ระบบแสดงหน้า Apply to be Partner 4.Click link 'Terms of Usage' 5.ระบบแสดง Popup 'Term and Conditions' 6.Click Agree 	ระบบแสดง Term and Conditions และเมื่อ Click 'Agree' ระบบทำการเลือก Checkbox ให้ Auto
5	TST_F9_0_1_009	TST_F9_0_1_009_Login_General_ClickLogin	Verify Parameter ตรวจสอบ กรณีข้อมูล Username กับ Password ไม่ตรงกัน ระบบจะแสดง Message Error "Incorrect username or password."	<ol style="list-style-type: none"> 1. เข้าลิ้งค์ https://10.104.240.205/#/login 2. ระบุ Email, Password 3. กดปุ่ม Sign In 4. ระบบจะตรวจสอบ ข้อมูลที่ DB: DEVPTB : USER_PROFILE Column : Email ไม่มีอยู่ในระบบ จะแสดง Message Error "Incorrect username or password." 	Email ไม่มีอยู่ในระบบ จะแสดง Message Error "Incorrect username or password." Error Code :: "DEVPO013"
6	TST_F1_0_1_029	TST_F1_1_1_029_Register_ThaiIDCardNoPassportNo_Require	Verify Parameter ตรวจสอบ Mandatory Field : Thai ID Card No. / Passport No. - กรณีไม่ระบุข้อมูล	<ol style="list-style-type: none"> 1. เข้าลิ้งค์ https://10.104.240.205/#/login 2. กดปุ่ม Sign Up 3. ระบบแสดงหน้าจอ Create Account 4. กดปุ่ม Sign Up 5. ตรวจสอบ Message Error Field :: Thai ID Card No. / Passport No. 	ระบบแสดง Message Error : "Please enter your Thai ID Card No./Passport." Error Code :: "DEVPO009 "
7	TST_F1_1_1_030	TST_F1_1_1_030_Register_ThaiIDCardNoPassportNo_Length	Verify Parameter ตรวจสอบ Length Field : Thai ID Card No. / Passport No. - ระบบสามารถรองรับการระบุข้อมูลเท่ากับ 5 ตัวอักษรขึ้นไปแต่ไม่เกิน 20 ตัวอักษร **หากระบุตัวอักษรน้อยกว่า 5 หรือมากกว่า 20 สามารถระบุได้แต่จะแสดง Message**	<ol style="list-style-type: none"> 1. เข้าลิ้งค์ https://10.104.240.205/#/login 2. กดปุ่ม Sign Up 3. ระบบแสดงหน้าจอ Create Account 4. ตรวจสอบ Field :: Thai ID Card No. / Passport No. 	ระบบสามารถรองรับการระบุข้อมูลเท่ากับ 5 ตัวอักษรขึ้นไป แต่ไม่เกิน 20 ตัวอักษร

8	TST_F1_0_1_031	TST_F1_1_1_031_Register_ThaiIDCardNoPassportNo_Length	Verify Parameter ตรวจสอบ Length Field : Thai ID Card No. / Passport No. - กรณีระบุข้อมูลน้อยกว่า 5 หรือเกิน 20 ตัวอักษร	1. เข้าลิ้งค์ https://10.104.240.205/#/login 2. กดปุ่ม Sign Up 3. ระบบแสดงหน้าจอ Create Account 4. ระบุข้อมูลน้อยกว่า 5 หรือเกิน 20 Field :: Thai ID Card No. / Passport No.	ระบบแสดง Message Error : " Thai ID Card No./Passport must have 5-20 characters." "
9	TST_F1_1_1_036	TST_F1_1_1_036_Register_Address_Format	Verify Parameter ตรวจสอบ Format Field : Address - ระบบรองรับการระบุข้อมูลภาษาไทย ภาษาอังกฤษ และอักขระพิเศษได้	1. เข้าลิ้งค์ https://10.104.240.205/#/login 2. กดปุ่ม Sign Up 3. ระบบแสดงหน้าจอ Create Account 4. ตรวจสอบ Field :: Address	ระบบรองรับการระบุข้อมูลภาษาไทย ภาษาอังกฤษ และอักขระพิเศษได้
10	TST_F1_1_1_046	TST_F1_1_1_046_Register_TermsOfUsage_ClickCheckbox_Enable	Verify Parameter ตรวจสอบการแสดงผลข้อมูล Pop Up : "Term and Conditions" (ข้อ 1)	1. เข้าลิ้งค์ https://10.104.240.205/#/login 2. กดปุ่ม Sign Up 3. ระบบแสดงหน้าจอ Create Account 4. ระบุข้อมูล Field ดังนี้ - Firstname - Lastname - Email - Password - Confirm Password - Thai ID Card No. / Passport No. - Telephone No. - Address - Security Check 5. ลิงก์ที่ Check Box : I have read and I Agree with the Terms of Usage Agreement. 6. ตรวจสอบการแสดงผลข้อมูล Pop Up : "Term and Conditions" (ข้อ 1)	ระบบต้องแสดงข้อความ :: 1. The developer hereby represents and warrants that any information or content provided herein by the developer shall not infringe the copyright or intellectual property rights of third parties. If there is a claim by third parties resulting in any and all losses, damages, liabilities or expenses (including reasonable legal fees) whatsoever incurred or suffered by or threatened against AIS, the developer shall use its best efforts to quell such claim at the earliest. Meanwhile, AIS shall consider temporarily suspend the rights and privileges of the developer in regard

<p>11</p>		<p>Verify Parameter ตรวจสอบการแสดงข้อมูล Pop Up : "Term and Conditions" (ข้อ 2)</p>	<ol style="list-style-type: none"> 1. เข้าลิ้งค์ https://10.104.240.205/#/login 2. กดปุ่ม Sign Up 3. ระบบแสดงหน้าจอ Create Account 4. ระบุข้อมูล Field ดังนี้ <ul style="list-style-type: none"> - Firstname - Lastname - Email - Password - Confirm Password - Thai ID Card No. / Passport No. - Telephone No. - Address - Security Check 5. ลักถูกที่ Check Box : I have read and I Agree with the Terms of Usage Agreement. 6. ตรวจสอบการแสดงข้อมูล Pop Up : "Term and Conditions" (ข้อ 2) 	<p>ระบบต้องแสดงข้อความ :: 2. To the extent of the then current protective technology, the developer warrants that the developed software do not and will not contain at the time released by the developer to AIS, any programs, routine, device or other undisclosed feature, including, without limitation, a time bomb, "time-out" feature, virus, software lock, drop dead device, malicious logic, worm, Trojan horse or trap door, which is designed to delete, disable, deactivate, interfere with or otherwise harm AIS' mobile communication networks, systems or AIS' hardware, data or other programs, or which is intended to provide unauthorized access or produce unauthorized modifications (collectively, "disabling procedures").</p>
-----------	--	---	---	--

<p>12</p>			<p>Verify Parameter ตรวจสอบการแสดงผลข้อมูล Pop Up : "Term and Conditions" (ข้อ 3)</p>	<ol style="list-style-type: none"> 1. เข้าลิ้งค์ https://10.104.240.205/#/login 2. กดปุ่ม Sign Up 3. ระบบแสดงหน้าจอ Create Account 4. ระบุข้อมูล Field ดังนี้ <ul style="list-style-type: none"> - Firstname - Lastname - Email - Password - Confirm Password - Thai ID Card No. / Passport No. - Telephone No. - Address - Security Check 5. ลึกถูกที่ Check Box : I have read and I Agree with the Terms of Usage Agreement. 6. ตรวจสอบการแสดงผลข้อมูล Pop Up : "Term and Conditions" (ข้อ 3) 	<p>ระบบต้องแสดงข้อความ :: 3. Regarding a claim by a third party that the developed software infringes its intellectual property rights or other rights, then the developer shall, within thirty (30) days of becoming aware of the claim (whether by AIS notifying the developer of it or otherwise), at its option: 3.1procure promptly for AIS the right to use the license of the developed software as contemplated under this agreement free of any claim or liability for infringement; 3.2develop promptly for AIS with other software which are non-infringing; or 3.3modify the developed software so that they cease to infringe those rights.</p>
-----------	--	--	---	---	---

<p>13</p>			<p>Verify Parameter ตรวจสอบการแสดงผลข้อมูล Pop Up : "Term and Conditions" (ข้อ 4)</p>	<ol style="list-style-type: none"> 1. เข้าลิ้งค์ https://10.104.240.205/#/login 2. กดปุ่ม Sign Up 3. ระบบแสดงหน้าจอ Create Account 4. ระบุข้อมูล Field ดังนี้ <ul style="list-style-type: none"> - Firstname - Lastname - Email - Password - Confirm Password - Thai ID Card No. / Passport No. - Telephone No. - Address - Security Check 5. ลึกถูกที่ Check Box : I have read and I Agree with the Terms of Usage Agreement. 6. ตรวจสอบการแสดงผลข้อมูล Pop Up : "Term and Conditions" (ข้อ 4) 	<p>ระบบต้องแสดงข้อความ :: 4. The developer shall not provide any developed software or content, which may fall within any of the following manners:</p> <p>(a)affect or insult king institution including heads of foreign countries; (b)cause insult or offensive or disrespectful remark against any country, government, government official or community; (c)insult or treat with disrespect or contempt, or damage any religion, respectful person, plan or objects, or intentionally or inadvertently violate any applicable local law, international law, including but not limited to, regulations promulgated by government authority, and any regulations having the same authorities; (d)conflict public policy, cause public unrest or affect relationship between countries; (e)conflict good moral, culture and customary of</p>
-----------	--	--	---	---	---

						<p>Thailand; (f)contain political content that may cause or affect the security and stability of Thailand; (g)contain pornography, obscene or content that arouses sexual desire; (h)contain content that is unlawful, harmful, threatening, abusive, harassing, tortuous, defamatory, vulgar, libellous, invasive of another's privacy, hateful, or racially, ethnically or otherwise objectionable, or that expresses cruelty, violence or inhumanity; (i)contain content of disillusion, deceit, non-sense which may deceive people, especially kids and youths; (j)contain content regarding abnormal behaviour, which may lead people to commit crime; (k)be rude or insulting any person, goods, services, businesses whether in the same or similar kind of goods, services, or business; (l)contain misstatement or</p>
--	--	--	--	--	--	---

						<p>untrue content; (m)conflict interest of AIS; and (n)infringe the third parties intellectual property rights</p>
--	--	--	--	--	--	--

14	TST_F1_1_1_050	TST_F1_1_1_050_Register_TermsOfUsage_General_ClickAgree	Verify Parameter ตรวจสอบการกดปุ่ม "Agree" - กรณีคลิก Check Box : I have read and I Agree with the Terms of Usage Agreement. *ต้องแสดงเครื่องหมายถูกใน Checkbox	1. เข้าลิงค์ https://10.104.240.205/#/login 2. กดปุ่ม Sign Up 3. ระบบแสดงหน้าจอ Create Account 4. ระบุข้อมูล Field ดังนี้ - Firstname - Lastname - Email - Password - Confirm Password - Thai ID Card No. / Passport No. - Telephone No. - Address - Security Check 5. ลิงก์ที่ Check Box : I have read and I Agree with the Terms of Usage Agreement. 6. ตรวจสอบการแสดงผล Pop Up 7. กดปุ่ม Agree	ระบบจะต้อง Alert หน้าจอ Create Account และคลิกถูกที่ Check Box : I have read and I Agree with the Terms of Usage Agreement.
15	TST_F1_1_1_051	TST_F1_1_1_051_Register_TermsOfUsage_General_ClickClose	Verify Parameter ตรวจสอบการกดปุ่ม "Close" - กรณีคลิก Check Box : I have read and I Agree with the Terms of Usage Agreement. -ต้องไม่มีเครื่องหมายถูกใน Checkbox	1. เข้าลิงค์ https://10.104.240.205/#/login 2. กดปุ่ม Sign Up 3. ระบบแสดงหน้าจอ Create Account 4. ระบุข้อมูล Field ดังนี้ - Firstname - Lastname - Email - Password - Confirm Password - Thai ID Card No. / Passport No. - Telephone No. - Address - Security Check 5. ลิงก์ที่ Check Box : I have read and I Agree with the Terms of Usage Agreement. 6. ตรวจสอบการแสดงผล Pop Up 7. กดปุ่ม Close	ระบบจะต้อง Alert หน้าจอ Create Account

16	TST_F1_1_1_058	TST_F1_1_1_058_Register_TermOfUsage_General_ClickClose	<p>Verify Parameter ตรวจสอบการกดปุ่ม "Close" - กรณีคลิกที่ข้อความ : Terms of Usage -กดปุ่ม [X] ตรวจสอบการแสดงผลข้อความเดิม</p>	<ol style="list-style-type: none"> 1. เข้าลิ้งค์ https://10.104.240.205/#/login 2. กดปุ่ม Sign Up 3. ระบบแสดงหน้าจอ Create Account 4. ระบุข้อมูล Field ดังนี้ <ul style="list-style-type: none"> - Firstname - Lastname - Email - Password - Confirm Password - Thai ID Card No. / Passport No. - Telephone No. - Address - Security Check 5. คลิกที่ข้อความ : Terms of Usage 6. ตรวจสอบการแสดงผล Pop Up 7. กดปุ่ม Close 	ระบบจะด้ง Alert มาหน้า Create Account
17	TST_F2_1_1_009	TST_F2_1_1_009_ApplyToBePartner_CompanyProfile_Format	<p>Verify parameters ตรวจสอบ Format ของ Field: Company Profile</p>	<ol style="list-style-type: none"> 1. เข้าลิ้งค์ https://10.104.240.205/#/login 2.เข้าสู่ระบบ (Sign in) 3.Click link 'Apply to be Partner' 3.ระบบแสดงหน้า Apply to be Partner 4.Check Format text Field :Company Profile 	ระบบรองรับ กรอกข้อมูล Company Profile ตัวอักษร ภาษาอังกฤษ (ใหญ่-เล็ก) ภาษาไทย ตัวเลข อักขระพิเศษ
18	TST_F2_0_1_010	TST_F2_0_1_010_ApplyToBePartner_CompanyProfile_Require	<p>Verify parameters ตรวจสอบ Madatory Field ของ Field: Company Profile กรณีไม่ระบุค่า</p>	<ol style="list-style-type: none"> 1. เข้าลิ้งค์ https://10.104.240.205/#/login 2.เข้าสู่ระบบ (Sign in) 3.Click link 'Apply to be Partner' 3.ระบบแสดงหน้า Apply to be Partner 4.Check Format text Field :Company Profile <ul style="list-style-type: none"> - พิมพ์ข้อความลงใน Text Field แล้วลบข้อความออก 	ระบบแสดง Error Message Error Code : "DEV0075" Error Message : "Company Profile is required."

19	TST_F2_1_1_011	TST_F2_1_1_011_ApplyToBePartner_CompanyProfile_Length	<p>Verify parameters ตรวจสอบ Length ของ Field: Company Profile ระบบรองรับ กรอกข้อมูลได้ไม่ เกิน 1000 Character</p>	<p>1. เข้าลิ้งค์ https://10.104.240.205/#/login 2.เข้าสู่ระบบ (Sign in) 3.Click link 'Apply to be Partner' 3.ระบบแสดงหน้า Apply to be Partner 4.กรอกข้อมูลเกิน Field :Company Profile 6.Check Field :Company Profile</p>	<p>ระบบรองรับ กรอกข้อมูล ได้ไม่ เกิน 1000 Character</p>
20	TST_F2_0_1_013	TST_F2_0_1_013_ApplyToBePartner_CompanyNameEng_Format	<p>Verify parameters ตรวจสอบ Format ของ Field: Company Name (Eng) กรณี ระบุข้อมูล ที่ไม่ใช่ ตัวอักษรภาษาอังกฤษ ตัวเลข อักขระพิเศษดังนี้ ! @ # \$ % & * () _ - . : ; , และ space</p>	<p>1. เข้าลิ้งค์ https://10.104.240.205/#/login 2.เข้าสู่ระบบ (Sign in) 3.Click link 'Apply to be Partner' 3.ระบบแสดงหน้า Apply to be Partner 4.Check Format text Field :Company Name (Eng)</p>	<p>ระบบแสดง Error Message Error Code : "DEVP0114" Error Message : "Company Name (Eng) must be contained Eng characters and number only."</p>

*** Settings ***

Resource ../Resource/PageKeywords/AISDeveloperPortalCommonKeywords.txt
 Resource ../Resource/PageKeywords/ProfilePartner/RoleMember/ApplyToBePartner.txt
 Resource ../Resource/PageKeywords/ProfilePartner/RoleCPC/PartnerProfile.txt
 Resource ../Resource/PageKeywords/ProfilePartner/RoleMember/Register.txt
 Resource ../Resource/PageKeywords/ProfilePartner/RoleCPC/PartnerProfile.txt
 Resource ../Resource/PageKeywords/ProfilePartner/RoleMKT/PartnerProfile.txt
 Library HttpLibrary.HTTP
 Resource ../Resource/PageKeywords/Application/RolePartner/ApplicationManagement.txt
 Library Process
 Resource ../Resource/PageKeywords/Support/Application.txt
 Library CustomXlsxLibrary
 Library ExcelLibrary
 Library CustomLibraryHtmlLog
 Library CustomExcelXlsLibrary
 Resource ../Resource/PageKeywords/ProfilePartner/RoleMember/ApplyToBePartner.txt
 Resource ../Resource/PageKeywords/ProfilePartner/RoleMember/Login.txt
 Resource ../Resource/PageKeywords/ProfilePartner/RoleMember/Register.txt

*** Test Cases ***

TST_F2_0_1_013_ApplyToBePartner_CompanyNameEng_Format

[Documentation] "1. เข้าลิ้งค์ <https://10.104.240.205/#/login>

... 2.เข้าสู่ระบบ (Sign in)

... 3.Click link 'Apply to be Partner'

... 3.ระบบแสดงหน้า Apply to be Partner

... 4.Check Format text Field :Company Name (Eng)"

[Tags] Format

[Timeout]

Open Web Browser \${UrlPartner} \${BrowserType}

Click Web Element \${PtnTopMenubtnSignIn}

Login Partner \${UserMember} \${PassMember}

Click Web Element \${PtnTopMenulinkApplyToBePartner}

Member Input Company Detail fieldCompanyNameEng +๑๒๑+๒

Member Verify Error Message Company Detail CompanyNameEng Format Visible=True

Member Input Company Detail fieldCompanyNameEng กหนดกหนดแบบหอ

Member Verify Error Message Company Detail CompanyNameEng Format Visible=True

Member Input Company Detail fieldCompanyNameEng @\$#กกก

Member Verify Error Message Company Detail CompanyNameEng Format Visible=True

[Teardown] Run Keywords Logout

... AND Close Web Browser

TST_F2_0_1_095_ApplyToBePartner_Marketing_Email_Duplicate

[Documentation] 1.เข้าลิ้งค์ <https://10.104.240.205/#/login>

... 2.เข้าสู่ระบบ (Sign in)

... 3.Click link 'Apply to be Partner'

... 4.ระบบแสดงหน้า Apply to be Partner

... 5.กรอกข้อมูล Field: Email (กรณีระบุ Email ซ้ำกันใน Department เดียวกัน)

[Tags] Duplicate

Open Web Browser \${UrlPartner} \${BrowserType}

Click Web Element \${PtnTopMenubtnSignIn}

Login Partner \${UserMember} \${PassMember}

Click Web Element \${PtnTopMenulinkApplyToBePartner}

Member Input Contact Detail Marketing fieldName[1]=FirstName LastName fieldTel[1]=1175
fieldEmail[1]=TestDevportal@ais.co.th fieldSocial[1]=Marda fieldTimeSupport[1]=09:00-15:00

Member Add Contact listDepartment=Marketing fieldName=David Kuromota
fieldTel=0891209090 fieldEmail=HelloWorld@ais.co.th fieldSocial=Marda
fieldTimeSupport=09:00-15:00

... btnOK

Member Input Contact Detail Marketing fieldEmail[2]=TestDevportal@ais.co.th

Member Verify Error Message Contact Detail Marketing Email[2] Duplicate

[Teardown] Run Keywords Logout

... AND Close Web Browser

TST_F2_0_1_219_ApplyToBePartner_PhotocopyOfFormal_Format

[Documentation] "1. เข้าลิ้งค์ <https://10.104.240.205/#/login>

... 2.เข้าสู่ระบบ (Sign in)

... 3.Click link 'Apply to be Partner'

... 4.ระบบแสดงหน้า Apply to be Partner

... 5.ตรวจสอบ Upload Document

... 6.cilck 'Choose File'" ตรวจสอบ Field: Photocopy of Formal Company

... กรณี Upload File ที่ไม่ใช่ PDF

[Tags] Format

Open Web Browser \${UrlPartner} \${BrowserType}

Click Web Element \${PtnTopMenubtnSignIn}

Login Partner \${UserMember} \${PassMember}

Click Web Element \${PtnTopMenulinkApplyToBePartner}

Member Upload Document PhotocopyOfFormal \${FileNameDoc}

Member Verify Error Message Upload Document PhotocopyOfFormal Format

Member Upload Document PhotocopyOfFormal \${FileNameJpg}
 Member Verify Error Message Upload Document PhotocopyOfFormal Format
 Member Upload Document PhotocopyOfFormal \${FileNameTxt}
 Member Verify Error Message Upload Document PhotocopyOfFormal Format
 Member Upload Document PhotocopyOfFormal \${FileNameXlsx}
 Member Verify Error Message Upload Document PhotocopyOfFormal Format
 [Teardown] Run Keywords Logout
 ... AND Close Web Browser

TST_F2_1_1_231_ApplyToBePartner_TermsOfUsage_General_ClickAgree_Checkbox

[Documentation] "1. เข้าลิ้งค์ <https://10.104.240.205/#/login>

... 2.เข้าสู่ระบบ (Sign in)
 ... 3.Click link 'Apply to be Partner'
 ... 3.ระบบแสดงหน้า Apply to be Partner
 ... 4.Click link 'Terms of Usage'
 ... 5.ระบบแสดง Popup 'Term and Conditions'
 ... 6.Cilck Agree " ระบบแสดง Term and Conditions
 ... และเมื่อ Cilck 'Agree' \ ระบบทำการเลือก Checkbox
 ... ให้ Auto

[Tags] General

Open Web Browser \${UrlPartner} \${BrowserType}
 Click Web Element \${PtnTopMenubtnSignIn}
 Login Partner \${UserMember} \${PassMember}
 Click Web Element \${PtnTopMenulinkApplyToBePartner}
 Member Input Company Detail fieldTaxID 1234567890000
 Member Input Company Detail fieldCompanyProfile test Company Profile
 Member Input Company Detail fieldCompanyNameEng Test Company
 Member Input Company Detail fieldCompanyNameThai บริษัททดลอง
 Member Input Company Detail fieldCommercialNameEng Test Company
 Member Input Company Detail fieldCommercialNameThai บริษัททดลอง
 Member Input Company Detail fieldRegisteredAddress 77/12 11009
 Member Input Company Detail fieldBillingAddress 77/12 11009
 Member Input Company Detail fieldMailingAddress 77/12 11009
 Member Input Company Detail fieldVATAddress 77/12 11009
 Member Input Company Detail fieldCompanyWebsite <http://www.aaa.com>
 Member Upload Document ServiceDetailMarketingPlan \${FileNamePdf}
 Member Input Company Detail fieldMobileTestList 986899908

Member Input Contact Detail ServiceOwner fieldName[1]=Harry Potter
fieldTel[1]=026650022 fieldEmail[1]=Harry@mail.com fieldSocial[1]=HarryFacebook.com
fieldTimeSupport[1]=08:30-17:30

Member Input Contact Detail Marketing fieldName[1]=Luis van fieldTel[1]=026650022
fieldEmail[1]=Luis@mail.com fieldSocial[1]=LuisFacebook.com fieldTimeSupport[1]=08:30-17:30

Member Input Contact Detail FinancialAccounting fieldName[1]=Zara fieldTel[1]=026650022
fieldEmail[1]=Zara@mail.com fieldSocial[1]=ZaraFacebook.com fieldTimeSupport[1]=08:30-17:30

Member Input Contact Detail TechnicalImplementation fieldName[1]=elsa
fieldTel[1]=026650022 fieldEmail[1]=elsa@mail.com fieldSocial[1]=elsaFacebook.com
fieldTimeSupport[1]=08:30-17:30

Member Input Contact Detail TechnicalOperation fieldName[1]=mickey
fieldTel[1]=026650022 fieldEmail[1]=mickey@mail.com fieldSocial[1]=mickeyFacebook.com
fieldTimeSupport[1]=08:30-17:30

Member Input Contact Detail CustomerSupport fieldName[1]=mike fieldTel[1]=026650022
fieldEmail[1]=mike@mail.com fieldSocial[1]=mikeFacebook.com fieldTimeSupport[1]=08:30-
17:30

Member Upload Document PhotocopyOfFormal \${FileNamePdf}
Member Upload Document PhotocopyOfVAT \${FileNamePdf}
Member Upload Document PhotocopyOfCompany \${FileNamePdf}
Input Web Text \${PtnApplyToBePartnerfieldCaptCha} \${CaptchaApplyToBePartner}
Click Web Element \${PtnApplyToBePartnerlinkTermsOfUsage}
Verify Text \${PtnApplyToBePartnerlblTermAndConditions} Term and Conditions
Click Web Element \${PtnApplyToBePartnerbtnAgree}
Selenium2Library.Checkbox Should Be Selected
\${PtnApplyToBePartnerchkbxTermsOfUsagelsChecked}
Selenium2Library.Capture Page Screenshot
[Teardown] Run Keywords Logout
... AND Close Web Browser

TST_F9_0_1_009_Login_General_ClickLogin
[Documentation] "1. เข้าสู่ลิงค์ <https://10.104.240.205/#/login>
... 2. ระบุ Email, Password
... 3. กดปุ่ม Sign In
... 4. ระบบจะตรวจสอบ ข้อมูลที่ DB: DEVP
... TB : USER_PROFILE
... Column : Email ไม่มีอยู่ในระบบ จะแสดง Message Error
... ""Incorrect username or password.""
... "
[Tags] General
Open Web Browser \${UrlPartnerSignIn} \${BrowserType}

Member Input Login fieldUsername \${CreateAccEmailRegister}
 Member Verify Login fieldUsername text=\${CreateAccEmailRegister}
 Member Input Login fieldPassword 000000
 Member Verify Login fieldPassword text=000000
 Click Web Element \${PtnLoginbtnSignIn}
 Member Verify Error Message Login Password Incorrect
 [Teardown] Close Web Browser

TST_F1_1_1_029_Register_ThaiIDCardNoPassportNo_Require

[Documentation] 1.เข้าลิ้งค์ <https://10.104.240.205/#/login>
 ... 2.กดปุ่ม Sign Up
 ... 3.ระบบแสดงหน้าจอ Create Account
 ... 4.กดปุ่ม Sign Up
 ... 5.ตรวจสอบ Message Error Field :: Thai ID Card No. Passport No. กรณีไม่ระบุข้อมูล
 [Tags] Require
 Open Web Browser \${UrlPartner} \${BrowserType}
 Click Web Element \${PtnTopMenubtnSignUp}
 Member Input Create Account fieldThaiIDCardNoPassportNo 1123344565432
 Member Input Create Account fieldThaiIDCardNoPassportNo \${EMPTY}
 Member Verify Create Account fieldThaiIDCardNoPassportNo Text=\${EMPTY}
 Member Verify Error Message Create Account ThaiIDCardNoPassportNo Require
 [Teardown] Close Web Browser

TST_F1_1_1_030_Register_ThaiIDCardNoPassportNo_Length

[Documentation] 1.เข้าลิ้งค์ <https://10.104.240.205/#/login>
 ... 2.กดปุ่ม Sign Up
 ... 3.ระบบแสดงหน้าจอ Create Account
 ... 4.ตรวจสอบ Field :: Thai ID Card No. Passport No. ระบบสามารถรองรับการระบุข้อมูลเท่ากับ 5 ตัวอักษร
 ขึ้นไปแต่ไม่เกิน 20 ตัวอักษร **หากระบุตัวอักษรน้อยกว่า 5 หรือมากกว่า 20 สามารถระบุได้ แต่จะแสดง Message**

[Tags] Length
 Open Web Browser \${UrlPartner} \${BrowserType}
 Click Web Element \${PtnTopMenubtnSignUp}
 Member Input Create Account fieldThaiIDCardNoPassportNo 12345
 Member Verify Create Account fieldThaiIDCardNoPassportNo Length=5
 Member Input Create Account fieldThaiIDCardNoPassportNo 12345678901234567890
 Member Verify Create Account fieldThaiIDCardNoPassportNo Length=20
 [Teardown] Close Web Browser

TST_F1_1_1_031_Register_ThaiIDCardNoPassportNo_Length

[Documentation] 1.เข้าลิ้งค์ <https://10.104.240.205/#/login>

... 2.กดปุ่ม Sign Up

... 3.ระบบแสดงหน้าจอ Create Account

... 4.ระบุข้อมูลน้อยกว่า 5 หรือ เกิน 20 Field :: Thai ID Card No. Passport No.

[Tags] Length

Open Web Browser \${UrlPartner} \${BrowserType}

Click Web Element \${PtnTopMenubtnSignUp}

Member Input Create Account fieldThaiIDCardNoPassportNo 1234

Member Verify Create Account fieldThaiIDCardNoPassportNo Text=1234

Member Verify Error Message Create Account ThaiIDCardNoPassportNo Length

Member Input Create Account fieldThaiIDCardNoPassportNo 123456789012345678901111

Member Verify Create Account fieldThaiIDCardNoPassportNo

Text=123456789012345678901111

Member Verify Error Message Create Account ThaiIDCardNoPassportNo Length

[Teardown] Close Web Browser

TST_F1_1_1_036_Register_Address_Format

[Documentation] "1. เข้าลิ้งค์ <https://10.104.240.205/#/login>

... 2. กดปุ่ม Sign Up

... 3. ระบบแสดงหน้าจอ Create Account

... 4. ตรวจสอบ Format \ Field : \ Address

... - ระบบรองรับการระบุข้อมูลภาษาไทย ภาษาอังกฤษ และอักขระพิเศษได้

[Tags] Format

Open Web Browser \${UrlPartner} \${BrowserType}

Click Web Element \${PtnTopMenubtnSignUp}

Member Input Create Account fieldAddress กกขขคคกงaaaaa@#\$%

Member Verify Create Account fieldAddress Text=กกขขคคกงaaaaa@#\$%

Member Verify Error Message Create Account Address Require Visible=False

[Teardown] Close Web Browser

TST_F1_1_1_046_Register_TermsOfUsage_ClickCheckbox_Enable

[Documentation] "1. เข้าลิ้งค์ <https://10.104.240.205/#/login>

... 2. กดปุ่ม Sign Up

... 3. ระบบแสดงหน้าจอ Create Account

... 4. ระบุข้อมูล Field ดังนี้

- ... - Firstname
- ... - Lastname
- ... - Email
- ... - Password
- ... - Confirm Password
- ... - Thai ID Card No. / Passport No.
- ... - Telephone No.
- ... - Address
- ... - Security Check
- ... 5. ตีถูกที่ Check Box : I have read and I Agree with the Terms of Usage Agreement.
- ... 6. ตรวจสอบการแจ้งเตือน \ Pop Up : ""Term and Conditions"" ระบบต้องแจ้งเตือน (ข้อ 1)" (ข้อ 2)

(ข้อ 3) (ข้อ 4)

[Tags] Enable

Open Web Browser \${UrlPartner} \${BrowserType}

Click Web Element \${PtnTopMenuBtnSignUp}

Click Web Element \${PtnCreateAccchkboxTermsOfUsage}

Verify Text \${PtnCreateAcclblTermAndConditions1} 1.\n\nThe developer hereby represents and warrants that any information or content provided herein by the developer shall not infringe the copyright or intellectual property rights of third parties. If there is a claim by third parties resulting in any and all losses, damages, liabilities or expenses (including reasonable legal fees) whatsoever incurred or suffered by or threatened against AIS, the developer shall use its best efforts to quell such claim at the earliest. Meanwhile, AIS shall consider temporarily suspend the rights and privileges of the developer in regard to such claim until being finalized.

Verify Text \${PtnCreateAcclblTermAndConditions2} 2.\nTo the extent of the then current protective technology, the developer warrants that the developed software do not and will not contain at the time released by the developer to AIS, any programs, routine, device or other undisclosed feature, including, without limitation, a time bomb, "time-out" feature, virus, software lock, drop dead device, malicious logic, worm, Trojan horse or trap door, which is designed to delete, disable, deactivate, interfere with or otherwise harm AIS' mobile communication networks, systems or AIS' hardware, data or other programs, or which is intended to provide unauthorized access or produce unauthorized modifications (collectively, "disabling procedures").

Verify Text \${PtnCreateAcclblTermAndConditions3} 3.\n\nRegarding a claim by a third party that the developed software infringes its intellectual property rights or other rights, then the developer shall, within thirty (30) days of becoming aware of the claim (whether by AIS notifying the developer of it or otherwise), at its option:\n3.1\n\nprocure promptly for AIS the right to use the license of the developed software as contemplated under this agreement free of any claim or liability for infringement;\n3.2\n\ndevelop promptly for AIS with other software which are non-infringing; or\n3.3\n\nmodify the developed software so that they cease to infringe those rights.

Verify Text \${PtnCreateAcclblTermAndConditions4} 4.\n\nThe developer shall not provide any developed software or content, which may fall within any of the following manners:\n(a)\n\naffect or insult king institution including heads of foreign countries;\n(b)\n\ncause insult or offensive or

disrespectful remark against any country, government, government official or community;\n(c)\ninsult or treat with disrespect or contempt, or damage any religion, respectful person, plan or objects, or intentionally or inadvertently violate any applicable local law, international law, including but not limited to, regulations promulgated by government authority, and any regulations having the same authorities;\n(d)\nconflict public policy, cause public unrest or affect relationship between countries;\n(e)\nconflict good moral, culture and customary of Thailand;\n(f)\ncontain political content that may cause or affect the security and stability of Thailand;\n(g)\ncontain pornography, obscene or content that arouses sexual desire;\n(h)\ncontain content that is unlawful, harmful, threatening, abusive, harassing, tortuous, defamatory, vulgar, libellous, invasive of another's privacy, hateful, or racially, ethnically or otherwise objectionable, or that expresses cruelty, violence or inhumanity;\n(i)\ncontain content of disillusion, deceit, non-sense which may deceive people, especially kids and youths;\n(j)\ncontain content regarding abnormal behaviour, which may lead people to commit crime;\n(k)\nbe rude or insulting any person, goods, services, businesses whether in the same or similar kind of goods, services, or business;\n(l)\ncontain misstatement or untrue content;\n(m)\nconflict interest of AIS; and\n(n)\ninfringe the third parties intellectual property rights

Selenium2Library.Capture Page Screenshot

[Teardown] Close Web Browser

TST_F1_1_1_050_Register_TermsOfUsage_General_ClickAgree

[Documentation] "1. เข้าลิ้งค์ <https://10.104.240.205/#/login>

... 2. กดปุ่ม Sign Up

... 3. ระบบแสดงหน้าจอ Create Account

... 4. ระบุข้อมูล Field ดังนี้

... - Firstname

... - Lastname

... - Email

... - Password

... - Confirm Password

... - Thai ID Card No. / Passport No.

... - Telephone No.

... - Address

... - Security Check

... 5. ตีถูกที่ Check Box : I have read and I Agree with the Terms of Usage Agreement.

... 6. ตรวจสอบการแสดงผล Pop Up

... 7. กดปุ่ม Agree"

... ระบบจะต้อง Alert มาหน้า Create Account และตีถูกที่ Check Box : I have read and I Agree with the

Terms of Usage Agreement.

[Tags] General

Open Web Browser \${UrlPartner} \${BrowserType}
 Click Web Element \${PtnTopMenubtnSignUp}
 Member Input Create Account fieldFirstname Test
 Member Input Create Account fieldLastname TestTest
 Member Input Create Account fieldEmail TestAutomate@hotmail.co.th
 Member Input Create Account fieldPassword 456Test!@#%t1278
 Member Input Create Account fieldConfirmPassword 456Test!@#%t1278
 Member Input Create Account fieldThaiIDCardNoPassportNo 1234567890123
 Member Input Create Account fieldTelephoneNo 095651234
 Member Input Create Account fieldAddress 10/100 home bkk 11000
 Member Input Create Account fieldCaptCha \${CaptchaApplyToBePartner}
 Click Web Element \${PtnCreateAccchkboxTermsOfUsage}
 Click Web Element \${PtnCreateAccbtnAgree}
 Selenium2Library.Checkbox Should Be Selected \${PtnCreateAccchkboxTermsOfUsagelsChecked}
 Selenium2Library.Capture Page Screenshot
 [Teardown] Close Web Browser

TST_F1_1_1_051_Register_TermsOfUsage_General_ClickClose

[Documentation] "1. เข้าลิงค์ <https://10.104.240.205/#/login>

... 2. กดปุ่ม Sign Up

... 3. ระบบแสดงหน้าจอ Create Account

... 4. ระบุข้อมูล Field ดังนี้

... - Firstname

... - Lastname

... - Email

... - Password

... - Confirm Password

... - Thai ID Card No. / Passport No.

... - Telephone No.

... - Address

... - Security Check

... 5. ตีถูกที่ Check Box : I have read and I Agree with the Terms of Usage Agreement.

... 6. ตรวจสอบการแสดง Pop Up

... 7. กดปุ่ม Close"ระบบจะต้อง Alert มาหน้า Create Account

... -ต้องไม่มีเครื่องหมายถูกใน Checkbox

[Tags] General

Open Web Browser \${UrlPartner} \${BrowserType}

Click Web Element \${PtnTopMenubtnSignUp}

Member Input Create Account fieldFirstname Test

Member Input Create Account fieldLastname TestTest
 Member Input Create Account fieldEmail TestAutomate@hotmail.co.th
 Member Input Create Account fieldPassword 456Test!@#%t1278
 Member Input Create Account fieldConfirmPassword 456Test!@#%t1278
 Member Input Create Account fieldThaiIDCardNoPassportNo 1234567890123
 Member Input Create Account fieldTelephoneNo 095651234
 Member Input Create Account fieldAddress 10/100 home bkk 11000
 Member Input Create Account fieldCaptCha \${CaptchaApplyToBePartner}
 Click Web Element \${PtnCreateAccchkboxTermsOfUsage}
 Click Web Element \${PtnCreateAccbtnClose}
 Selenium2Library.Checkbox Should Not Be Selected
 \${PtnCreateAccchkboxTermsOfUsagelsNotChecked}
 Selenium2Library.Capture Page Screenshot
 [Teardown] Close Web Browser

TST_F1_1_1_058_Register_TermOfUsage_General_ClickClose

[Documentation] "1. เข้าลิ้งค์ <https://10.104.240.205/#/login>

... 2. กดปุ่ม Sign Up

... 3. ระบบแสดงหน้าจอ Create Account

... 4. ระบุข้อมูล Field ดังนี้

... - Firstname

... - Lastname

... - Email

... - Password

... - Confirm Password

... - Thai ID Card No. / Passport No.

... - Telephone No.

... - Address

... - Security Check

... 5. คลิกที่ข้อความ : Terms of Usage

... 6. ตรวจสอบการแสดงผล Pop Up

... 7. กดปุ่ม Close"

... ตรวจสอบการแสดงผลข้อความเดิม

[Tags] General

Open Web Browser \${UrlPartner} \${BrowserType}

Click Web Element \${PtnTopMenubtnSignUp}

Member Input Create Account fieldFirstname Test

Member Input Create Account fieldLastname TestTest

Member Input Create Account fieldEmail TestAutomate@hotmail.co.th

Member Input Create Account fieldPassword 456Test!@#%t1278
 Member Input Create Account fieldConfirmPassword 456Test!@#%t1278
 Member Input Create Account fieldThaiIDCardNoPassportNo 1234567890123
 Member Input Create Account fieldTelephoneNo 095651234
 Member Input Create Account fieldAddress 10/100 home bkk 11000
 Member Input Create Account fieldCaptCha \${CaptchaApplyToBePartner}
 Click Web Element \${PtnCreateAcclinkTermsOfUsage}
 Click Web Element \${PtnCreateAccbtnClose}
 Member Verify Create Account fieldFirstname Text=Test
 Member Verify Create Account fieldLastname Text=TestTest
 Member Verify Create Account fieldEmail Text=TestAutomate@hotmail.co.th
 Member Verify Create Account fieldPassword Text=456Test!@#%t1278
 Member Verify Create Account fieldConfirmPassword Text=456Test!@#%t1278
 Member Verify Create Account fieldThaiIDCardNoPassportNo Text=1234567890123
 Member Verify Create Account fieldTelephoneNo Text=095651234
 Member Verify Create Account fieldAddress Text=10/100 home bkk 11000
 Member Input Create Account fieldCaptCha \${CaptchaApplyToBePartner}
 Selenium2Library.Capture Page Screenshot
 [Teardown] Close Web Browser

TST_F2_1_1_009_ApplyToBePartner_CompanyProfile_Format

[Documentation] "1. ^{คลิก}เข้าถึง <https://10.104.240.205/#/login>
 ... 2.เข้าสู่ระบบ (Sign in)
 ... 3.Click link 'Apply to be Partner'
 ... 3.ระบบแสดงหน้า Apply to be Partner
 ... 4.Check Format text Field :Company Profile"
 [Tags] Format
 Open Web Browser \${UrlPartner} \${BrowserType}
 Click Web Element \${PtnTopMenubtnSignIn}
 Login Partner \${UserMember} \${PassMember}
 Click Web Element \${PtnTopMenulinkApplyToBePartner}
 Member Input Company Detail fieldCompanyProfile 123456
 Member Verify Company Detail fieldCompanyProfile Text=123456
 Member Input Company Detail fieldCompanyProfile dfdfหกดกหต๒๓๔๓๒๔#@
 Member Verify Company Detail fieldCompanyProfile Text=dfdfหกดกหต๒๓๔๓๒๔#@
 [Teardown] Run Keywords Logout
 ... AND Close Web Browser

TST_F2_0_1_010_ApplyToBePartner_CompanyProfile_Require

[Documentation] "1. ^{คลิก}เข้าถึง <https://10.104.240.205/#/login>

ประวัติผู้เขียน

ชื่อ-นามสกุล

นายเศรษฐพงษ์ อิ่มสุวรรณ

ประวัติการศึกษา

พ.ศ. 2557 ปริญญาตรี คณะวิศวกรรมศาสตร์ สาขาวิศวกรรม
คอมพิวเตอร์ มหาวิทยาลัยธุรกิจบัณฑิตย์

ตำแหน่งและสถานที่ทำงานปัจจุบัน

Automation Software Test Engineer II บริษัท
Aware Technology Solution

