

การทดสอบระบบบูรณาการข้อมูลสำหรับศูนย์บริการข้อมูลลูกค้า

TESTING CONSOLIDATION AND BUSINESS PROCESS  
REDESIGN SYSTEM FOR CUSTOMER CONTACT CENTER



โครงการสหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร  
ปริญญาวิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์)  
ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2559

การทดสอบระบบบูรณาการข้อมูลสำหรับศูนย์บริการข้อมูลลูกค้า

TESTING CONSOLIDATION AND BUSINESS PROCESS  
REDESIGN SYSTEM FOR CUSTOMER CONTACT CENTER



นางสาวชมพูนุช ภูวันนา  
นางสาวพรธีรา มีจรูญสม

โครงการสหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร  
ปริญญาวิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์)  
ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2559

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TESTING CONSOLIDATION AND BUSINESS PROCESS  
REDESIGN SYSTEM FOR CUSTOMER CONTACT CENTER



COOPERATIVE EDUCATIONSUBMITTED IN PARTIAL FULFILLMENT OF  
THE REQUIREMENTS FOR  
THE DEGREE OF BACHELOR OF SCIENCE (COMPUTER SCIENCE)  
DEPARTMENT OF COMPUTER SCIENCE, FACULTY OF SCIENCE  
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG  
ACADEMIC YEAR 2559

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงการสหกิจศึกษา	การทดสอบระบบบูรณาการข้อมูลสำหรับศูนย์บริการข้อมูลลูกค้า	
	Testing Consolidation and Business Process Redesign System for Customer Contact	
ชื่อนักศึกษา	นางสาวชมพูนุช ภูวันนา	รหัสนักศึกษา 56050229
	นางสาวพรธีรา มีจรูญสม	รหัสนักศึกษา 56050321
ปริญญา	วิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์)	
ภาควิชา	วิทยาการคอมพิวเตอร์	
ปีการศึกษา	2559	
อาจารย์ที่ปรึกษา	ผศ.กฤษฎา บุศรา	

คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.) อนุมัติให้โครงการสหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์) ประจำปีการศึกษา 2559

คณะกรรมการสอบ	ลายมือชื่อ
ผศ.กฤษฎา บุศรา อาจารย์นิเทศและอาจารย์ที่ปรึกษา	

ลิขสิทธิ์ของคณะวิทยาศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

T1495๖๑

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงการสหกิจศึกษา	การทดสอบระบบบูรณาการข้อมูลสำหรับศูนย์บริการข้อมูลลูกค้า		
ชื่อนักศึกษา	นางสาวชมพูช ภูวันนา	รหัสนักศึกษา	56050229
	นางสาวพรธีรา มีจรรย์สม	รหัสนักศึกษา	56050321
ปริญญา	วิทยาศาสตร์บัณฑิต (วิทยาการคอมพิวเตอร์)		
ภาควิชา	วิทยาการคอมพิวเตอร์		
คณะ	วิทยาศาสตร์		
มหาวิทยาลัย	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.)		
ปีการศึกษา	2559		
อาจารย์ที่ปรึกษา	ผศ.กฤษฎา บุศรา		

### บทคัดย่อ

การทดสอบระบบบูรณาการข้อมูลสำหรับศูนย์บริการข้อมูลลูกค้า เป็นโครงการสหกิจศึกษาของบริษัท Fusionware Company Limited โดยวัตถุประสงค์ของโครงการคือ เพื่อทดสอบความถูกต้องในการทำงานของเว็บแอปพลิเคชัน ซึ่งในการทดสอบนั้นจะตรวจสอบกระบวนการทำงาน และการแสดงผลของระบบ โดยแบ่งประเภทของการทดสอบออกเป็น 2 ประเภท คือ การทดสอบซอฟต์แวร์ด้วยมนุษย์ (Manual test) และการทดสอบซอฟต์แวร์ด้วยระบบอัตโนมัติ (Automated test) ในการทดสอบจะเป็นการตรวจสอบความถูกต้องในการทำงานของระบบทั้งหมด 2 ส่วน ได้แก่ การแสดงผลหน้าบ้าน (Front office) และการแสดงผลหลังบ้าน (Back office) สำหรับการทดสอบด้วยระบบอัตโนมัติ จะมีการพัฒนาการทดสอบโดยการนำเครื่องมือ (Tools) มาประยุกต์ใช้ ได้แก่ เครื่องมือ Selenium IDE เป็นส่วนเสริมของการทดสอบเว็บแอปพลิเคชันที่ทำงานบนเบราว์เซอร์ Firefox และ Robot framework ซึ่งเป็นเฟรมเวิร์คโอเพนซอร์สภาษา Python ที่สนับสนุนแนวคิดในการพัฒนาแบบ Acceptance Test Driven Development (ATDD) และ Selenium-Webdriver โดยทั้งหมดนี้เป็นเครื่องมือในการปฏิสัมพันธ์กับเว็บแอปพลิเคชันแทนมนุษย์ เพื่อลดเวลาที่ใช้ในการทดสอบ เพิ่มประสิทธิภาพ เพิ่มความแม่นยำ และลดข้อผิดพลาดก่อนที่จะถูกส่งมอบไปถึงมือของลูกค้า ทำให้การทดสอบมีความครอบคลุมมากยิ่งขึ้น

**คำสำคัญ :** การทดสอบซอฟต์แวร์ด้วยมนุษย์ การทดสอบซอฟต์แวร์ด้วยระบบอัตโนมัติ การพัฒนาแบบขับเคลื่อนจากการยอมรับของผู้ใช้ โรบอทเฟรมเวิร์ค เว็บแอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<b>Title</b>	Testing Consolidation and Business Process Redesign System for Customer Contact
<b>Students</b>	Miss Chompunooch Puwanna Student ID 56050229 Miss Porntheera Meejaroonsom Student ID 56050321
<b>Degree</b>	Bachelor of Science (Computer Science)
<b>Department</b>	Computer Science
<b>Faculty</b>	Science
<b>University</b>	King Mongkut's Institute of Technology Ladkrabang (KMITL)
<b>Academic Year</b>	2016
<b>Advisor</b>	Asst.Prof.Krudsada Budsara

### Abstract

This paper is about the cooperative education project that examined the testing consolidation and business process redesign system for customer contact center of Fusionware Company Limited. The main purpose is to test the validity of the web application operations. This has been actively completed by examining the operating process and the display system. In this case, there are two types of web application examination included: manual software test and automated software test which were totally examined in both front office and back office of operating system. For that automated software test, the Selenium IDE tool, an optional web testing which performs on Firefox browser and Robot framework that is an open framework Python language to support Acceptance Test Driven Development (ATDD) and Selenium-Webdriver theories, were applied to develop the web application examination. Upon the examination, all those tools were utilized instead to increase the web application examination efficiency and accuracy. Moreover, the tools also reduced time and the budget.

**Keywords :** Acceptance Test Driven Development (ATDD), Automated software test, Manual software test, Web application

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

รายงานสำหรับโครงการสหกิจศึกษาฉบับนี้สำเร็จลงได้ด้วยดีเนื่องด้วยความอนุเคราะห์ช่วยเหลือเป็นอย่างดีจาก อาจารย์ ผศ.กฤษฎา บุศรา ที่กรุณาเป็นอาจารย์ที่ปรึกษา ให้คำปรึกษา และประสานงานต่างๆ เพื่อให้การทำโครงการสหกิจศึกษา และการมาสหกิจศึกษาที่บริษัท Fusionware Company Limited เป็นไปอย่างดี ผู้จัดทำขอกราบขอบพระคุณอาจารย์ทุกท่านเป็นอย่างสูงไว้ ณ ที่นี้

นอกจากนี้ผู้จัดทำขอกราบขอบพระคุณ

- |                 |                   |         |                    |
|-----------------|-------------------|---------|--------------------|
| 1. คุณปรีชากร   | ตินงาม            | ตำแหน่ง | Project Manager    |
| 2. คุณภาสพงศ์   | ไทยทัตกุล         | ตำแหน่ง | Solution Architect |
| 3. คุณสาคร      | แสนเมือง          | ตำแหน่ง | Developer Leader   |
| 4. คุณภาณุพัฒน์ | ช่อลัดดา          | ตำแหน่ง | Developer          |
| 5. คุณจรงค์     | กองโปธิ           | ตำแหน่ง | Quality Assurance  |
| 6. คุณปริตต์    | เบ็ญจสุพัฒน์นันท์ | ตำแหน่ง | Software Tester    |
| 7. คุณธนเทพ     | มินาภา            | ตำแหน่ง | Software Tester    |
| 8. คุณสุมิตรา   | โพธิลา            | ตำแหน่ง | Business Analyst   |

เป็นอย่างสูงที่คอยดูแลเอาใจใส่ผู้จัดทำอย่างจริงจัง ให้คำปรึกษาในทุกๆด้าน และช่วยให้การมาสหกิจของผู้จัดทำสำเร็จไปได้ด้วยดี และข้าพเจ้าขอขอบพระคุณพี่ๆสมาชิกในทีม SSCRN Tester ทุกคนที่คอยให้คำปรึกษา สั่งสอน และให้ความรู้ในทุกๆด้าน ซึ่งมีส่วนทำให้ตัวผู้จัดทำได้รับความรู้ และประสบการณ์ในการทำงานจริงอันมีค่า

สุดท้ายนี้ผู้จัดทำขอกราบขอบพระคุณ คุณพ่อ คุณแม่ และคนในครอบครัวที่คอยเป็นกำลังใจ ให้เสมอมาตลอดช่วงเวลาที่ได้ออกมาทำโครงการสหกิจในครั้งนี้ รวมไปถึงรุ่นพี่ และเพื่อนๆที่ให้คำปรึกษาในทุกๆด้านตลอดช่วงเวลาที่ผ่านมานี้

นางสาว ชมพูนุช ภูวันนา

นางสาว พรธีรา มีจรรยาสม

# สารบัญ

	หน้า
บทคัดย่อภาษาไทย .....	ก
บทคัดย่อภาษาอังกฤษ .....	ข
กิตติกรรมประกาศ .....	ค
สารบัญ .....	ง
สารบัญตาราง .....	ช
สารบัญรูป .....	ฉ
<b>บทที่ 1 บทนำ .....</b>	<b>1</b>
1.1 ความเป็นมาและความสำคัญของงานปัญหา .....	1
1.2 รายละเอียดเกี่ยวกับบริษัท FusionWare Company Limited .....	2
1.3 วัตถุประสงค์ของงานวิจัย .....	2
1.4 ขอบเขตของงานวิจัย .....	3
1.5 ประโยชน์ที่คาดว่าจะได้รับ .....	3
1.6 ขั้นตอนการดำเนินงาน .....	4
1.7 อุปกรณ์และเครื่องมือในการดำเนินงาน .....	4
<b>บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง .....</b>	<b>6</b>
2.1 การทดสอบระบบ (Testing) .....	6
2.1.1 ความแตกต่างระหว่าง Verification และ Validation .....	7
2.1.2 เทคนิคที่ใช้ในการทดสอบระบบ .....	7
2.1.3 ประเภทของการทดสอบโปรแกรม .....	8
2.1.4 ระดับ Software Testing .....	8
2.1.5 รูปแบบการทดสอบระบบ .....	9
2.1.6 ความสำคัญของข้อผิดพลาด (Defect Priority) .....	9
2.1.7 ระดับในการทดสอบระบบ .....	10
2.2 การทดสอบอัตโนมัติ (Automated Testing) .....	10
2.2.1 ประเภทของ Automated Test .....	11
2.2.2 ลักษณะของเทสเคสที่เหมาะสมในการทำ Automated Testing .....	11
2.2.3 ข้อดีของการทดสอบอัตโนมัติ .....	12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
2.3 เครื่องมือที่ใช้ในการทดสอบอัตโนมัติ .....	12
2.3.1 Selenium.....	13
2.3.2 Selenium IDE.....	20
2.3.3 Selenium WebDriver.....	23
2.3.4 Robot framework .....	29
2.4 ATDD .....	34
2.4.1 ลักษณะเด่นๆ ของ ATDD.....	34
2.4.2 ATDD cycle .....	34
2.5 TDD.....	36
2.6 เครื่องมือที่ใช้ในการทดสอบอัตโนมัติ .....	37
2.6.1 วิธีการค้นหาด้วย ID.....	38
2.6.2 วิธีการค้นหาด้วย Name.....	39
2.6.3 วิธีการค้นหาด้วย Link Text.....	39
2.6.4 วิธีการค้นหาด้วย CSS.....	40
2.6.5 วิธีการค้นหาด้วย XPath.....	42
2.7 ภาษาจาวา (Java programming language).....	44
2.7.1 จาวาแพลตฟอร์ม .....	45
2.7.2 ข้อดีของภาษาจาวา .....	45
2.8 ภาษาไพทอน (Python).....	46
2.8.1 ไพทอนแพลตฟอร์ม .....	46
2.8.2 ข้อดีของภาษาไพทอน.....	47
2.9 XML (Extensible Markup Language).....	48
2.9.1 Element.....	49
2.9.2 Attribute.....	49
2.9.3 โครงสร้างของเอกสาร XML.....	50
2.9.4 การประกาศ XML .....	50
2.9.5 จุดเด่นของ XML .....	50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
2.10 ระบบบริหารความสัมพันธ์ลูกค้า.....	51
2.10.1 ประโยชน์ของ CRM ต่อธุรกิจ .....	52
2.10.2 การเพิ่มประสิทธิภาพของเจ้าหน้าที่ Call Center ด้วยระบบ CRM.....	53
2.11 เว็บเซอร์วิส (Web service) .....	54
2.11.1 มาตรฐานหลักของเว็บเซอร์วิส .....	54
2.11.2 คุณลักษณะของ Web Service .....	55
2.12 Jira .....	56
2.12.1 ประเภทของงานใน Jira.....	56
2.12.2 ความสำคัญ (Priority) .....	58
<b>บทที่ 3 วิธีการดำเนินงานวิจัย.....</b>	<b>59</b>
3.1 การศึกษาระบบบูรณาการข้อมูลสำหรับศูนย์บริการข้อมูลลูกค้า .....	59
3.2 การทำความเข้าใจจุดประสงค์ และขอบเขตในการทดสอบระบบ .....	61
3.3 การออกแบบกรณีทดสอบตามกรณีทดสอบที่ได้รับ .....	68
3.3.1 การออกแบบกรณีทดสอบสำหรับพีเจอร์ ระบบ SSCRN สามารถ แสดงข้อมูลพื้นฐานของลูกค้าในหน้า Customer information ได้อย่างถูกต้อง.....	71
3.3.2 การออกแบบกรณีทดสอบสำหรับพีเจอร์ ระบบ SSCRN สามารถแสดง ชื่อ-นามสกุล เมื่อทำการค้นหาเบอร์โทรศัพท์ บนหน้า Home ได้อย่างถูกต้อง.....	74
3.3.3 การออกแบบกรณีทดสอบสำหรับพีเจอร์ ระบบ SSCRN สามารถแสดง Package ได้อย่างถูกต้องเมื่อทำการค้นหา Main Package ในหน้า Customer informatio.....	77
3.3.4 การออกแบบกรณีทดสอบสำหรับพีเจอร์ ระบบ SSCRN สามารถออก Order (ซื้อแพคเกจ หรือ ยกเลิกแพคเกจ) ได้อย่างถูกต้อง....	81
3.4 การพัฒนาโปรแกรมทดสอบอัตโนมัติ (Automated Testing).....	85
3.4.1 การพัฒนาโปรแกรมเพื่อทดสอบพีเจอร์ ระบบ SSCRN สามารถแสดงข้อมูลพื้นฐานของลูกค้าในหน้า Customer information ได้อย่างถูกต้อง ด้วยเครื่องมือ Selenium IDE.....	86

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
3.4.2 การพัฒนาโปรแกรมเพื่อทดสอบพีเจอร์ ระบบ SSCRN สามารถแสดง ชื่อ-นามสกุล เมื่อทำการค้นหาเบอร์โทรศัพท์ บนหน้า Home ได้อย่างถูกต้อง ด้วยเครื่องมือ Robot framework .....	86
3.4.3 การพัฒนาโปรแกรมเพื่อทดสอบพีเจอร์ ระบบ SSCRN แสดง Package ได้ถูกต้องเมื่อทำการค้นหา Main Package ในหน้า Customer information ด้วยเครื่องมือ Robot framework ....	111
3.4.4 การพัฒนาโปรแกรมเพื่อทดสอบพีเจอร์ ระบบ SSCRN สามารถออก Order (ซื้อแพคเกจ หรือ ยกเลิกแพคเกจ) ได้อย่างถูกต้อง ด้วยเครื่องมือ Robot framework .....	122
3.5 วิเคราะห์ และจัดการกับข้อผิดพลาดที่เกิดขึ้น .....	136
3.5.1 ขั้นตอนการติดตามปัญหา (Defect Tracking) .....	136
<b>บทที่ 4 ผลการวิจัยและการอภิปรายผล .....</b>	<b>141</b>
4.1 ผลการพัฒนาซอฟต์แวร์ .....	141
4.1.1 การพัฒนาซอฟต์แวร์อัตโนมัติด้วยเครื่องมือ Selenium IDE.....	141
4.1.2 การพัฒนาซอฟต์แวร์อัตโนมัติด้วยเครื่องมือ Robot Framework.....	145
<b>บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ .....</b>	<b>151</b>
5.1 สรุปผลการวิจัย .....	151
5.2 ปัญหาและข้อจำกัด.....	151
5.3 ข้อเสนอแนะ .....	152
เอกสารอ้างอิง.....	153
ภาคผนวก ก.....	156
ภาคผนวก ข.....	198

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญตาราง

ตารางที่	หน้า
2.1 แนวความคิดในการทดสอบระบบระหว่าง Programmer และ Tester.....	7
2.2 สรุปข้อดีและข้อเสียของการใช้งานร่วมกับ Selenium RC .....	15
2.3 แสดงความแตกต่างหลัก ๆ ระหว่าง Selenium Grid 1 และ 2 .....	17
2.4 สรุปข้อดีและข้อเสียของการใช้งานร่วมกับ Selenium IDE .....	18
2.5 สรุปข้อดีและข้อเสียของการใช้งานร่วมกับ WebDriver.....	19
2.6 แสดงสัญลักษณ์และรายละเอียดการทำงานของ XPath.....	42
2.7 แสดงตัวอย่างการใช้สัญลักษณ์และรายละเอียดการทำงานของ XPath .....	43
3.1 ตารางแบบฟอร์ม Test Case ที่ใช้กรอกเงื่อนไขต่างๆในการทดสอบระบบ .....	69
3.2 กรณีทดสอบของพีเจอร์ ระบบ SSCRN สามารถแสดงข้อมูลพื้นฐานของลูกค้า ในหน้า Customer information ได้อย่างถูกต้อง.....	71
3.3 Scenario ที่ใช้สำหรับการแสดงผลของ Feature ระบบ SSCRN สามารถแสดงข้อมูลลูกค้าได้อย่างถูกต้อง .....	74
3.4 กรณีทดสอบของพีเจอร์ ระบบ SSCRN สามารถแสดงชื่อลูกค้า เมื่อทำการค้นหาเบอร์โทรศัพท์บนหน้า Home ได้อย่างถูกต้อง .....	75
3.5 Scenario ที่ใช้สำหรับการแสดงผลของ Feature ระบบ SSCRN สามารถแสดงข้อมูลลูกค้าได้อย่างถูกต้อง.....	76
3.6 กรณีทดสอบของพีเจอร์ ระบบ SSCRN สามารถแสดง Package ได้อย่างถูกต้อง เมื่อทำการค้นหา Main Package ในหน้า Customer information.....	78
3.7 Scenario ที่ใช้สำหรับการแสดงผลของ Feature ระบบ SSCRN สามารถแสดง Package ได้อย่างถูกต้องเมื่อทำการค้นหา Main Package ในหน้า Customer information .....	80
3.8 กรณีทดสอบของพีเจอร์ ระบบ SSCRN สามารถออก Order (ซื้อแพคเกจ) ได้อย่างถูกต้อง .....	82
3.9 Scenario ที่ใช้สำหรับการแสดงผลของ Feature ระบบ SSCRN สามารถออก Order (สามารถซื้อแพคเกจ) ได้อย่างถูกต้อง .....	85

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป

รูปที่	หน้า
2.1 System Development Life Cycle.....	6
2.2 V Mode .....	9
2.3 กลุ่มของเครื่องมือทดสอบที่ใช้ใน Selenium .....	13
2.4 แนวคิดในการทำงานที่เรียกว่า Same Origin Policy .....	14
2.5 แสดงการทำงาน ของ Selenium Grid .....	16
2.6 เปรียบเทียบการทำงานระหว่าง Selenium RC และ WebDriver .....	18
2.7 Selenium 2 เกิดจากการรวมตัวกันของ Selenium RC และ WebDriver .....	20
2.8 แสดงส่วนประกอบของคำสั่งที่ใช้ใน Selenium IDE .....	21
2.9 แสดง Standard Library.....	30
2.10 แสดง External Library .....	30
2.11 แสดง Other Library .....	30
2.12 แสดงรูปแบบคำสั่งการกำหนดค่า Library .....	31
2.13 แสดงรูปแบบคำสั่งการสร้าง Keyword.....	32
2.14 แสดงรูปแบบคำสั่งการสร้างตัวแปร.....	33
2.15 แสดงรูปแบบคำสั่งการสร้าง Test Case .....	34
2.16 แสดงวงจรการพัฒนาแบบ ATDD .....	36
2.17 วงจรการพัฒนาแบบ TDD .....	36
2.18 ระบบบริหารความสัมพันธ์ลูกค้า (CRM Software) .....	52
2.19 Jira.....	56
2.20 Jira Screen.....	58
3.1 แผนผังการทำงานของวงจรการพัฒนาซอฟต์แวร์แบบ Scrum model.....	61
3.2 Flow ขั้นตอนการทดสอบระบบ.....	64
3.3 สร้าง Folder ที่ใช้ในการทดสอบในโฟลเดอร์หลัก.....	86
3.4 สร้างโฟลเดอร์ Report ที่ใช้ในโฟลเดอร์ PostBos .....	86
3.5 สร้างโฟลเดอร์ Script ที่ใช้ในโฟลเดอร์ PostBos .....	87
3.6 สร้างโฟลเดอร์ Snapshot ที่ใช้ในโฟลเดอร์ PostBos .....	87
3.7 สร้างโฟลเดอร์ Report ที่ใช้ในโฟลเดอร์ PostNon .....	87
3.8 สร้างโฟลเดอร์ Script ที่ใช้ในโฟลเดอร์ PostNon .....	87

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
3.9 สร้างโฟลเดอร์ Snapshot ที่ใช้ในโฟลเดอร์ PostNon .....	87
3.10 สร้างโฟลเดอร์ Report ที่ใช้ในโฟลเดอร์ PreBos .....	88
3.11 สร้างโฟลเดอร์ Script ที่ใช้ในโฟลเดอร์ PreBos .....	88
3.12 สร้างโฟลเดอร์ Snapshot ที่ใช้ในโฟลเดอร์ PreBos .....	88
3.13 สร้างโฟลเดอร์ Report ที่ใช้ในโฟลเดอร์ PreNon .....	88
3.14 สร้างโฟลเดอร์ Script ที่ใช้ในโฟลเดอร์ PreNon .....	88
3.15 สร้างโฟลเดอร์ Snapshot ที่ใช้ในโฟลเดอร์ PreNon .....	89
3.16 สร้างโฟลเดอร์ Report ที่ใช้ในโฟลเดอร์ PreINS .....	89
3.17 สร้างโฟลเดอร์ Script ที่ใช้ในโฟลเดอร์ PreINS .....	89
3.18 สร้างโฟลเดอร์ Snapshot ที่ใช้ในโฟลเดอร์ PreINS .....	89
3.19 การกำหนด Base URL สำหรับเข้าทดสอบระบบ .....	90
3.20 การกำหนดความเร็วในการ Execute เป็น Fast .....	90
3.21 การสร้าง Store สำหรับเก็บค่า Path ของรูปภาพที่ทำการจับหน้าจอ .....	90
3.22 การแสดงผลของคำสั่งใน Table .....	90
3.23 การสร้าง Store สำหรับเก็บค่าเบอร์โทรศัพท์ .....	90
3.24 การแสดงผลของคำสั่งใน Table .....	91
3.25 แสดงการใช้คำสั่ง assertTitle .....	91
3.26 การแสดงตรวจสอบด้วยคำสั่ง AssertTitle .....	91
3.27 แสดงการใช้คำสั่ง Type .....	92
3.28 ค่าของ HTML attribute ที่ใช้คำสั่ง Type .....	92
3.29 แสดงการใช้คำสั่ง Click .....	92
3.30 ค่าของ HTML attribute ที่ใช้คำสั่ง Click .....	93
3.31 แสดงการใช้คำสั่ง waitForText .....	93
3.32 ค่าของ HTML attribute ที่ใช้คำสั่ง waitForText .....	93
3.33 แสดงการใช้คำสั่ง verifyText .....	94
3.34 ค่าของ HTML attribute ที่ใช้คำสั่ง verifyText .....	95
3.35 แสดงการใช้คำสั่ง Pause .....	95
3.36 แสดงการใช้คำสั่ง storeEval .....	95

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
3.37 แสดงการใช้คำสั่ง captureEntirePageScreenshot.....	95
3.38 แสดงการใช้คำสั่ง clickAndWait.....	96
3.39 ค่าของ HTML attribute ที่ใช้คำสั่ง clickAndWaitBos.....	96
3.40 แสดงการใช้คำสั่ง Pause .....	96
3.41 แสดงการใช้คำสั่ง waitForTex เพื่อรอข้อความ Alert แสดงบนหน้าจอ .....	97
3.42 ค่าของ HTML attribute ที่ใช้คำสั่ง waitForText .....	97
3.43 แสดงการใช้คำสั่ง assertTitle .....	97
3.44 การแสดงตรวจสอบด้วยคำสั่ง AssertTitle .....	98
3.45 แสดงการใช้คำสั่ง verifyValue.....	98
3.46 ค่าของ HTML attribute ที่ใช้คำสั่ง verifyValuev .....	98
3.47 แสดงการตรวจสอบการแสดงผลโดยใช้คำสั่ง verifyText.....	99
3.48 ค่าของ HTML attribute ที่ใช้คำสั่ง verify สำหรับตรวจสอบ Use .....	99
3.49 ค่าของ HTML attribute ที่ใช้คำสั่ง verify สำหรับตรวจสอบ Network Type .....	100
3.50 ค่าของ HTML attribute ที่ใช้คำสั่ง verify สำหรับตรวจสอบ Service Year.....	100
3.51 ค่าของ HTML attribute ที่ใช้คำสั่ง verify สำหรับตรวจสอบ Mobile Status.....	100
3.52 ค่าของ HTML attribute ที่ใช้คำสั่ง verify สำหรับตรวจสอบ Mobile Segment .....	100
3.53 ค่าของ HTML attribute ที่ใช้คำสั่ง verify สำหรับตรวจสอบ Migrate to BOS.....	100
3.54 ค่าของ HTML attribute ที่ใช้คำสั่ง verify สำหรับตรวจสอบ Customer Type.....	101
3.55 ค่าของ HTML attribute ที่ใช้คำสั่ง verify สำหรับตรวจสอบ PW Verified .....	101
3.56 ค่าของ HTML attribute ที่ใช้คำสั่ง verify สำหรับตรวจสอบ Secret Password .....	101
3.57 ค่าของ HTML attribute ที่ใช้คำสั่ง verify สำหรับตรวจสอบ Billing Cycle.....	101
3.58 ค่าของ HTML attribute ที่ใช้คำสั่ง verify สำหรับตรวจสอบ Credit Limit.....	101
3.59 Test script หน้า Home ของกลุ่มลูกค้าประเภท Postpaid NonBos.....	102
3.60 Test script หน้า Customer infomation ของกลุ่มลูกค้าประเภท Postpaid NonBos .	102
3.61 การเปลี่ยนแปลง Test script เพื่อใช้งานกับกลุ่มลูกค้าประเภท Postpaid Bos.....	103
3.62 Test script หน้า SSCRN Home สำหรับ กลุ่มลูกค้าประเภท Prepaid ทั้งหมด.....	104
3.63 Test script หน้า Customer infomation สำหรับกลุ่มลูกค้าประเภท Prepaid ทั้งหมด	104
3.64 การกำหนด Library ที่ใช้ในการทดสอบ.....	105

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
3.65 ตัวแปรที่ใช้ในการทดสอบ .....	105
3.66 การสร้าง Keyword Open Browser and select User .....	106
3.67 ค่าของ HTML attribute ที่ใช้คำสั่ง Wait until Element Is Visible .....	106
3.68 การสร้าง Keyword Enter Mobile Number .....	106
3.69 ค่าของ HTML attribute ที่ใช้คำสั่ง Input text.....	107
3.70 ค่าของ HTML attribute ที่ใช้คำสั่ง Click Element .....	107
3.71 ค่าของ HTML attribute ที่ใช้คำสั่ง Wait until Element Is Visible .....	107
3.72 การสร้าง Keyword Verify Customer Name at Home.....	107
3.73 ค่าของ HTML attribute ที่ใช้คำสั่ง Get Text.....	108
3.74 การสร้าง Keyword End call.....	108
3.75 ค่าของ HTML attribute ที่ใช้คำสั่ง Click Element .....	108
3.76 Testcase สำหรับ Scenario 1 ลูกค้ำประเภท Postpaid Bos.....	109
3.77 Testcase สำหรับ Scenario 2 ลูกค้ำประเภท Postpaid Non Bos.....	109
3.78 Testcase สำหรับ Scenario 3 ลูกค้ำประเภท Prepaid Bos.....	109
3.79 Testcase สำหรับ Scenario 4 ลูกค้ำประเภท Prepaid Non Bos.....	110
3.80 Testcase สำหรับ Scenario 5 ลูกค้ำประเภท Prepaid INS .....	110
3.81 ตัวแปรที่ใช้ในการทดสอบ .....	111
3.82 การสร้าง Keyword Open Browser and select User .....	112
3.83 ค่าของ HTML attribute ที่ใช้คำสั่ง Wait until Element Is Visible .....	112
3.84 การสร้าง Keyword Enter Mobile Numbe.....	112
3.85 ค่าของ HTML attribute ที่ใช้คำสั่ง Input text.....	113
3.86 ค่าของ HTML attribute ที่ใช้คำสั่ง Click Element .....	113
3.87 ค่าของ HTML attribute ที่ใช้คำสั่ง Wait until Element Is Visible .....	113
3.88 การสร้าง Keyword Verify Customer Name at Home.....	113
3.89 ค่าของ HTML attribute ที่ใช้คำสั่ง Get Text.....	114
3.90 การสร้าง Keyword Click promotion service button .....	114
3.91 ค่าของ HTML attribute ที่ใช้คำสั่ง Click Element .....	114
3.92 การสร้าง Keyword Verify Customer name .....	114

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
3.93 ค่าของ HTML attribute ที่ใช้คำสั่ง Get Text.....	115
3.94 การสร้าง Keyword Verify Mobile status.....	115
3.95 ค่าของ HTML attribute ที่ใช้คำสั่ง Get Text.....	115
3.96 การสร้าง Keyword Click Button Change Main promotion .....	115
3.97 ค่าของ HTML attribute ที่ใช้ใน Keyword Verify Mobile status.....	116
3.98 การสร้าง Keyword Enter Empty Main Promotion.....	116
3.99 ค่าของ HTML attribute ที่ใช้ใน Keyword Enter Empty Main Promotion.....	116
3.100 การสร้าง Keyword Verify Package Main promotion .....	116
3.101 ค่าของ HTML attribute ที่ใช้ใน Keyword Verify Package Main promotion .....	117
3.102 การสร้าง Keyword Enter Package Main Promotion.....	117
3.103 ค่าของ HTML attribute ที่ใช้ใน Keyword Enter Package Main .....	117
3.104 การสร้าง Keyword Click Search Package Main Promotion .....	117
3.105 ค่าของ HTML attribute ที่ใช้ใน Keyword Click Element.....	118
3.106 การสร้าง Keyword Click Search Package Main Promotion .....	118
3.107 Testcase สำหรับ Scenario 1 .....	118
3.108 Testcase สำหรับ Scenario 2 .....	118
3.109 Testcase สำหรับ Scenario 3 .....	119
3.110 Testcase สำหรับ Scenario 4 .....	119
3.111 Testcase สำหรับ Scenario 5 .....	120
3.112 Testcase สำหรับ Scenario 6 .....	120
3.113 Testcase สำหรับ Scenario 7 .....	120
3.114 Testcase สำหรับ Scenario 8 .....	120
3.115 Testcase สำหรับ Scenario 9 .....	120
3.116 Testcase สำหรับ Scenario 10 .....	121
3.117 Testcase สำหรับ Scenario 11 .....	121

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
3.118 Testcase สำหรับ Scenario 12 .....	121
3.119 ตัวแปรที่ใช้ในการทดสอบ .....	122
3.120 การสร้าง Keyword Open Browser and select User .....	122
3.121 ค่าของ HTML attribute ที่ใช้คำสั่ง Wait Until Element Is Visible.....	123
3.122 การสร้าง Keyword Enter Mobile Number .....	123
3.123 ค่าของ HTML attribute ที่ใช้คำสั่ง Input text.....	123
3.124 ค่าของ HTML attribute ที่ใช้คำสั่ง Click Element .....	124
3.125 ค่าของ HTML attribute ที่ใช้คำสั่ง Wait Until Element Is Visible.....	124
3.126 การสร้าง Keyword Verify Customer Name at Home .....	124
3.127 ค่าของ HTML attribute ที่ใช้คำสั่ง Get Text.....	124
3.128 การสร้าง Keyword Click promotion service button.....	124
3.129 ค่าของ HTML attribute ที่ใช้คำสั่ง Click Element .....	125
3.130 การสร้าง Keyword Verify Customer name.....	125
3.131 ค่าของ HTML attribute ที่ใช้คำสั่ง Wait Until Element Is Visible.....	125
3.132 ค่าของ HTML attribute ที่ใช้คำสั่ง Get Text.....	125
3.133 การสร้าง Keyword Verify Mobile status.....	126
3.134 ค่าของ HTML attribute ที่ใช้คำสั่ง Get Text.....	126
3.135 การสร้าง Keyword Click go to Package Consultation page .....	126
3.136 ค่าของ HTML attribute ที่ใช้คำสั่ง Click Element .....	126
3.137 การสร้าง Keyword Enter Package .....	127
3.138 ค่าของ HTML attribute ที่ใช้คำสั่ง Input text.....	127
3.139 การสร้าง Keyword Enter Package .....	127
3.140 ค่าของ HTML attribute ที่ใช้คำสั่ง Click Element .....	127
3.141 การสร้าง Keyword Verify Result Package .....	128
3.142 ค่าของ HTML attribute ที่ใช้คำสั่ง Get text.....	128
3.143 การสร้าง Keyword Select Effective Date Next Day .....	128

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
3.144 ค่าของ HTML attribute ที่ใช้ใน Keyword Select Effective Date Next Day (1) ....	129
3.145 ค่าของ HTML attribute ที่ใช้ใน Keyword Select Effective Date Next Day .....	129
3.146 การสร้าง Keyword Click Add Package.....	129
3.147 ค่าของ HTML attribute ที่ใช้ใน Keyword Click Add Package.....	129
3.148 การสร้าง Keyword Click Add Package.....	130
3.149 ค่าของ HTML attribute ที่ใช้ใน Keyword Verify Package in Shopping cart.....	130
3.150 การสร้าง Keyword Click Go to Package Management.....	130
3.151 ค่าของ HTML attribute ที่ใช้ใน Keyword Click Go to Package Management....	131
3.152 การสร้าง Keyword Verify Package in Package Management Page .....	131
3.153 ค่าของ HTML attribute ที่ใช้ใน Keyword Click Go to Package Management....	131
3.154 การสร้าง Keyword Input Description Over 1000 Character.....	132
3.155 การสร้าง Keyword Click Submit Order .....	132
3.156 ค่าของ HTML attribute ที่ใช้ใน Keyword Click Submit Order .....	132
3.157 การสร้าง Keyword Go to Customer information page .....	133
3.158 ค่าของ HTML attribute ที่ใช้ใน Keyword Go to Customer information page ....	133
3.159 การสร้าง Keyword End call.....	133
3.160 ค่าของ HTML attribute ที่ใช้ใน Keyword End call.....	133
3.161 Testcase สำหรับ Scenario 1 .....	134
3.162 Testcase สำหรับ Scenario 2 .....	134
3.163 Testcase สำหรับ Scenario 3 .....	134
3.164 Testcase สำหรับ Scenario 4 .....	134
3.165 Testcase สำหรับ Scenario 5 .....	135
3.166 Testcase สำหรับ Scenario 6 .....	135
3.167 Testcase สำหรับ Scenario 7 .....	135
3.168 Testcase สำหรับ Scenario 8 .....	135
3.169 Testcase สำหรับ Scenario 9 .....	135
3.170 Create Issue (1) .....	136
3.171 Create Issue (2) .....	137
3.172 Create Issue (3) .....	138

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
3.173 Create Issue (4) .....	139
3.174 Create Issue (5) .....	140
4.1 ตัวอย่างรายงานแสดงผลการทดสอบด้วยเครื่องมือ Selenium IDE .....	142
4.2 ตัวอย่างรูปที่บันทึกหน้าจออัตโนมัติด้วยเครื่องมือ Selenium IDE .....	142
4.3 ผลลัพธ์การทดสอบกรณีไม่มีข้อผิดพลาด ด้วยเครื่องมือ Selenium IDE .....	143
4.4 ผลลัพธ์การทดสอบในรูปแบบของตารางกรณีไม่มีข้อผิดพลาด .....	143
4.5 ผลลัพธ์การทดสอบกรณีเกิดข้อผิดพลาด ด้วยเครื่องมือ Selenium IDE .....	144
4.6 ผลลัพธ์การทดสอบในรูปแบบของตารางกรณีเกิดข้อผิดพลาด .....	145
4.7 ตัวอย่างรายงานแสดงผลการทดสอบด้วยเครื่องมือ Robot Framework .....	146
4.8 ผลลัพธ์การทดสอบกรณีไม่มีข้อผิดพลาด ด้วยเครื่องมือ Robot Framework .....	147
4.9 ผลลัพธ์การทดสอบกรณีเกิดข้อผิดพลาด ด้วยเครื่องมือ Robot Framework .....	148
4.10 การแสดงรายละเอียดการทดสอบ Test detail .....	148
4.11 การแสดงผลของ File log ด้วยเครื่องมือ Robot Framework .....	149
4.12 การบันทึกผลด้วยรูปภายใน File log .....	150
ก.1 แสดงการ Download Python .....	157
ก.2 แสดงไฟล์ python ที่ทำการ Download เสร็จ .....	157
ก.3 แสดงการเลือก Run เพื่อติดตั้ง .....	158
ก.4 แสดงการ Install (1) และ (2) .....	158
ก.5 แสดงการเลือก directory ที่ต้องการติดตั้ง (3) .....	159
ก.6 แสดงการเลือก features (4) .....	159
ก.7 แสดงการรอติดตั้ง Python .....	160
ก.8 แสดงการติดตั้ง (5) .....	160
ก.9 หลังจากติดตั้ง Python เสร็จ .....	161
ก.10 แสดงการทดสอบ Python ผ่าน command line .....	161
ก.11 แสดงกรณีผลการทำงาน Error .....	161
ก.12 แสดงการกำหนด Environment variable แบบชั่วคราว (1) .....	162
ก.13 แสดงการกำหนด Environment variable แบบชั่วคราว (2) .....	162
ก.14 แสดงการกำหนด Environment variable (1) .....	162
ก.15 แสดงการกำหนด Environment variable (2) .....	163

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
ก.16 แสดงการกำหนด Environment variable (3).....	163
ก.17 แสดงการกำหนด Environment variable (4).....	164
ก.18 แสดงการกำหนด Environment variable (5).....	164
ก.19 แสดงการกำหนด Environment variable (6).....	165
ก.20 แสดงการทดสอบ Python ผ่าน command line.....	166
ก.21 แสดงการพิมพ์คำสั่ง pip ใน command line.....	166
ก.22 แสดงผลการทำงานของคำสั่ง pip.....	167
ก.23 แสดงคำสั่งการติดตั้ง robot framework ด้วย pip.....	167
ก.24 แสดงการทำงานหลังการติดตั้ง pip.....	168
ก.25 แสดงคำสั่งการติดตั้ง library.....	168
ก.26 แสดงการตรวจสอบการติดตั้ง selenium2library.....	169
ก.27 แสดงการตรวจสอบผลการติดตั้ง robot framework.....	169
ก.28 แสดงตัวอย่าง Text editor (1).....	170
ก.29 แสดงตัวอย่างของการใช้ Sublime ในการเขียน Robot framework (2).....	170
ก.30 แสดงการดาวน์โหลด Sublime.....	171
ก.31 แสดงการเลือก Run เพื่อติดตั้ง.....	171
ก.32 แสดงการติดตั้ง Sublime (1).....	172
ก.33 แสดงการติดตั้ง Sublime (2).....	172
ก.34 แสดงการติดตั้ง Sublime (3).....	173
ก.35 แสดงการติดตั้ง Sublime (4).....	173
ก.36 แสดงการติดตั้ง Sublime (5).....	174
ก.37 แสดงการติดตั้ง Sublime (6).....	174
ก.38 แสดงการติดตั้ง Sublime (7).....	175
ก.39 แสดง Console.....	175
ก.40 แสดงการนำซอร์สโค้ดที่คัดลอกไว้ใน มาวางใน Console (8).....	176
ก.41 แสดงภาพการติดตั้ง Package Control สำเร็จ.....	176
ก.42 แสดงการ install package Robot Framework (9).....	176

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
ก.43 แสดงการ install package Robot Framework (10) .....	177
ก.44 install package Robot Framework สำเร็จ.....	177
ก.45 แสดงการเลือก Robot Framework Assistant (11) .....	178
ก.46 เลือก Robot Framework Assistant สำเร็จ .....	178
ก.47 แสดงการเลือกดาวน์โหลด Chrome driver .....	179
ก.48 แสดงการติดตั้ง Chrome driver (1) .....	179
ก.49 แสดงการติดตั้ง Chrome driver (2) .....	180
ก.50 แสดงการติดตั้ง Chrome driver .....	180
ก.51 แสดงการ Setup Path chromedriver (1) .....	180
ก.52 แสดงการ Setup Path chromedriver (2) .....	181
ก.53 แสดงการ Setup Path chromedriver (3) .....	181
ก.54 แสดงการ Setup Path chromedriver (4) .....	182
ก.55 แสดงการเลือกดาวน์โหลด IE Driver .....	183
ก.56 แสดงการติดตั้ง IE Driver (1) .....	183
ก.57 แสดงการติดตั้ง IE Driver (2) .....	184
ก.58 แสดงการ Setup Path IE driver (1) .....	184
ก.59 แสดงการ Setup Path IE driver (2) .....	185
ก.60 แสดงการ Setup Path IE driver (3) .....	186
ก.61 แสดงการตั้งค่าที่ IE Browser (1) .....	187
ก.62 แสดงการตั้งค่าที่ IE Browser (2) .....	188
ก.63 แสดงการตั้งค่าที่ IE Browser (3) .....	189
ก.64 แสดงการตั้งค่าที่ IE Browser (4) .....	190
ก.65 แสดงการเลือกดาวน์โหลด Gecko web driver .....	191
ก.66 แสดงการติดตั้ง Gecko Driver .....	191
ก.67 แสดงการติดตั้ง Gecko Driver .....	192
ก.68 แสดงการ Setup Path Gecko driver (1) .....	192
ก.69 แสดงการ Setup Path Gecko driver (2) .....	193
ก.70 แสดงการ Setup Path Gecko driver (3) .....	193
ก.71 ตัวอย่าง Test case ที่ใช้ในการทดสอบ .....	194

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
ก.72 Process การทำงานของ Test case .....	194
ก.73 การใช้ Snippets ใน Robot framework.....	195
ก.74 ตัวอย่าง Test script ที่ใช้ทดสอบ .....	195
ก.75 การ Run Test Case (1).....	196
ก.76 การ Run Test Case (2).....	196
ก.77 การ Run Test Case (3).....	196
ก.78 ตัวอย่าง Report ที่เกิดจากการ Run .....	197
ข.1 การ Download โปรแกรม Selenium IDE.....	199
ข.2 การ Install โปรแกรม Selenium IDE .....	199
ข.3 การ Restart โปรแกรมหลังจากทำการติดตั้ง .....	200
ข.4 การ Customize เพื่อให้โปรแกรมขึ้นมาแสดงบน Tool bar .....	200
ข.5 การ Drag and Drop โปรแกรมมาแสดงบน Tool bar .....	201
ข.6 การเรียกใช้งาน Selenium IDE (1).....	201
ข.7 การเรียกใช้งาน Selenium IDE (2).....	202
ข.8 ใส่ URL ที่ต้องการทดสอบ .....	203
ข.9 เปิดโปรแกรม Selenium IDE.....	203
ข.10 การเลือก Show All Available Command .....	204
ข.11 การใช้ Command (1).....	204
ข.12 การใช้ Command (2).....	205
ข.13 การใช้ Command (3).....	205
ข.14 การสั่ง Save Test Case .....	206
ข.15 การสั่ง Export Test Case .....	207
ข.16 การ Run Test Script บน Selenium IDE (1).....	207
ข.17 การ Run Test Script บน Selenium IDE (2).....	208
ข.18 การ Run Test Script บน Selenium IDE (3).....	209

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

การดำเนินธุรกิจในปัจจุบันขององค์กรนั้น จะประสบผลสำเร็จมากน้อยเพียงใด ลูกค้ามีส่วนสำคัญมาก การเน้นการบริการลูกค้า และการสร้างความสัมพันธ์กับลูกค้าให้เหนียวแน่นขึ้นการรักษาลูกค้าให้อยู่ได้นานเป็นสิ่งสำคัญที่องค์กรต่างให้ความสนใจ CRM จึงเป็นเครื่องมือหนึ่งที่ต้องการหลายๆ องค์กรนำมาใช้ ดังนั้นบริษัท Fusionware Company Limited ได้พัฒนาระบบบูรณาการสำหรับศูนย์บริการข้อมูลลูกค้า (SSCRN) เพื่อใช้ในเครือข่ายโทรศัพท์เคลื่อนที่แห่งหนึ่ง ซึ่งเป็นระบบที่มีคุณสมบัติเชื่อมโยงระบบมากกว่า 10 ระบบมาไว้ในระบบเดียว โดยระบบบูรณาการสำหรับศูนย์บริการข้อมูลลูกค้า (SSCRN) รองรับการทำงานสำหรับส่วนส่งเสริมการขายในประเทศ (Promotion) ซึ่งมีความสามารถในการทำงาน (Feature) เช่น แสดงข้อมูลลูกค้า (Customer information) การซื้อโปรโมชั่น (Package Consultation) ตั้งแต่โปรโมชั่นหลัก โปรโมชั่นเสริม หรือบริการเสริมต่างๆ การแนะนำโปรโมชั่นที่เหมาะสมกับพฤติกรรมการใช้งานของลูกค้า (Offer Package) หรือโปรโมชั่นยอดนิยมที่มีคนใช้งานจำนวนมาก (Hot hit) มีการแจ้งเตือน (Alert) ในเรื่องต่างๆ รวมถึงการตรวจสอบการทำรายการต่างๆ ทั้งหมดเมื่อมีการทำรายการในระบบ (Order monitor) นอกจากนั้นระบบยังมีการจัดการส่วนส่งเสริมการขาย ต่างประเทศ (International Roaming) และมีการทำระบบหลังบ้าน (Back office) เพื่อให้ผู้ใช้งานสามารถเข้ามาปรับแต่งการตั้งค่าต่างๆภายในระบบได้อีกด้วย ทั้งหมดนี้เพื่ออำนวยความสะดวกให้แก่ผู้รับสาย (Agent) ในระบบบูรณาการ สำหรับศูนย์บริการข้อมูลลูกค้า (Agent) และผู้ที่เกี่ยวข้องกับการเข้าใช้งานระบบ ดังนั้นด้วยความสามารถในการทำงานของระบบ จึงมีความจำเป็นต้องทดสอบระบบให้ครอบคลุมและเกิดข้อผิดพลาดในการทำงานน้อยที่สุด

โดย ณ ปัจจุบันนี้ระบบระบบบูรณาการสำหรับศูนย์บริการข้อมูลลูกค้า (SSCRN) กำลังอยู่ในช่วงทดสอบประสิทธิภาพการทำงานของระบบ จึงจำเป็นต้องมีการเคลื่อนย้ายซอร์สโค้ดไปมาระหว่างส่วนพัฒนา และส่วนระบบจริง ซึ่งก่อนจะทำการเคลื่อนย้ายไปยังระบบจริงนั้น จะต้องทำการตรวจสอบว่าซอร์สโค้ดใหม่ที่ทำกรเขียนขึ้น หรือซอร์สโค้ดที่ถูกแก้ไขดัดแปลงนั้น จะไม่ไปกระทบกับซอร์สโค้ดส่วนเดิมที่ทำงานได้ถูกต้องอยู่แล้ว ซึ่งอาจทำให้ระบบมีความผิดพลาด หรือเกิด Defect ขึ้น โดยการทดสอบนั้นนักทดสอบระบบจะต้องทำการทดสอบด้วยมือ (Manual testing) ซึ่งหลังจากการเอกสารนี้เป็นเอกสารที่ส่งมอบไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำสอบได้สักระยะ นักทดสอบได้เกิดปัญหาคือ การทำสอบด้วยมือนั้นครอบคลุมการทำงานในทุกส่วนของระบบอย่างไม่ทั่วถึง ใช้เวลานาน และตรวจสอบได้ไม่ครอบคลุม ซึ่งอาจทำให้ไม่สามารถแก้ไขข้อผิดพลาดที่เกิดขึ้นได้ทันเวลา อีกทั้งยังพบข้อผิดพลาดที่ไม่ถูกตรวจสอบหลุดรอดออกไป นอกจากนี้ยังทำให้สิ้นเปลืองงบประมาณ และบุคลากรที่ใช้ในการทดสอบแต่ละครั้งอย่างไม่คุ้มค่า

จึงทำให้มีโครงการทดสอบซอฟต์แวร์ทดสอบอัตโนมัติขึ้น เพื่อทดสอบการทำงานของระบบบูรณาการสำหรับศูนย์บริการข้อมูลลูกค้า (SSCRN) แทนมนุษย์ทุกครั้งที่มีการเปลี่ยนแปลงหรือแก้ไขช่วยลดค่าใช้จ่ายและจำนวนบุคลากรที่ใช้ในการทดสอบแต่ละครั้งลง และเพิ่มประสิทธิภาพในการทดสอบให้มีความครอบคลุม เพิ่มแม่นยำ และรวดเร็วมากยิ่งขึ้น

## 1.2 รายละเอียดเกี่ยวกับบริษัท FusionWare Company Limited

ซึ่งเป็นกลุ่มบริษัทที่ให้บริการครบวงจรด้านเทคโนโลยีสารสนเทศ เช่น การพัฒนาแอปพลิเคชัน (Application) การบริหารจัดการระบบฐานข้อมูล การประยุกต์ใช้ Software Solution ต่าง ๆ ในการแก้ปัญหาธุรกิจ การออกแบบและพัฒนาเว็บไซต์ หรือการพัฒนาโปรแกรมสำหรับ Mobile และ Tablet Devices ซึ่งเครือ Cloud Group เป็นผู้เชี่ยวชาญในการเชื่อมต่อระบบสารสนเทศ ตามแนวคิดสถาปัตยกรรมเชิงบริการ (Service-Oriented Architecture) โดยได้ทำการแบ่งระบบสารสนเทศเพื่อให้สอดคล้องกับการบริหารจัดการข้อมูลสารสนเทศแบบองค์รวมขององค์กร และยังเป็นผู้ที่ทำกรให้บริการด้านเทคโนโลยีสารสนเทศทั้งองค์กรภาครัฐและเอกชนมาเป็นระยะเวลากว่า 10 ปี

## 1.3 วัตถุประสงค์ของงานวิจัย

- 1) เพื่อลดความผิดพลาดในการทำงานของระบบ ทำงานระบบทำงานอย่างมีประสิทธิภาพ
- 2) สามารถจัดการการทดสอบ โดยการนำเทคนิคต่าง ๆ ในการทดสอบเข้ามาใช้ได้อย่างเหมาะสม
- 3) ศึกษาวิธีพัฒนาซอฟต์แวร์ทดสอบอัตโนมัติ
- 4) พัฒนาซอฟต์แวร์ทดสอบอัตโนมัติเพื่อใช้ทดสอบระบบบูรณาการสำหรับศูนย์บริการข้อมูลลูกค้า (SSCRN) แทนมนุษย์
- 5) พัฒนาซอฟต์แวร์ที่มีความยืดหยุ่น และเข้าใจได้ง่าย เพื่อให้ผู้อื่นนำไปใช้พัฒนาต่อยอดได้
- 6) วิเคราะห์ และจัดการปัญหาที่เกิดขึ้นระหว่างการทดสอบ เพื่อนำไปแก้ไขได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.4 ขอบเขตของงานวิจัย

- 1) ระบบสามารถแสดงข้อมูลต่าง ๆ ได้ถูกต้องตามที่กำหนด โดยมีการทดสอบระบบโดยการใช้เครื่องมือในการทดสอบ (Automated Test) และการทดสอบด้วยมนุษย์ (Manual Test)
- 2) สร้างเหตุการณ์ในการทดสอบที่ครอบคลุมเหตุการณ์หลัก ที่เกิดขึ้นในการทำงานจริง
- 3) การทดสอบทุกครั้งจะมีรายงานผลการทดสอบเพื่อบ่งชี้ถึงจุดที่เกิดข้อผิดพลาด
- 4) การพัฒนาซอฟต์แวร์ เป็นซอฟต์แวร์ทดสอบอัตโนมัติสำหรับทดสอบระบบบูรณาการสำหรับศูนย์บริการข้อมูลลูกค้า (SSCRN) เท่านั้น
- 5) ซอฟต์แวร์ทดสอบอัตโนมัติพัฒนาด้วยภาษา Python
- 6) สร้างชุดคำสั่ง (Test Script) สำหรับใช้ในการทดสอบกับระบบบูรณาการสำหรับศูนย์บริการข้อมูลลูกค้า (SSCRN) เท่านั้น
- 7) ทดสอบการทำงานของระบบบนเว็บเบราว์เซอร์ Google Chrome, Internet Explorer, Mozilla Firefox
- 8) ใช้ทดสอบระบบหลังจากที่มีการเปลี่ยนแปลงหรือแก้ไข เพื่อหาข้อผิดพลาดก่อนจะนำไปใช้งานบนระบบจริง
- 9) ซอฟต์แวร์ทดสอบอัตโนมัติพัฒนามาจากกรณีทดสอบเดียวกันกับการทดสอบด้วยมนุษย์

## 1.5 ประโยชน์ที่คาดว่าจะได้รับ

- 1) ผู้พัฒนา
  - เข้าใจหลักการของการทดสอบด้วยมนุษย์
  - เข้าใจหลักการของการทดสอบอัตโนมัติ
  - สามารถนำการทดสอบอัตโนมัติไปประยุกต์ใช้ในการพัฒนาซอฟต์แวร์ในอนาคตได้
  - ได้ประยุกต์ใช้ความรู้ที่เรียนมาในการทำงานจริง
- 2) ผู้ใช้ซอฟต์แวร์ทดสอบอัตโนมัติ
  - ใช้ซอฟต์แวร์อัตโนมัติในการทดสอบการทำงานของระบบ แทนการทดสอบด้วยมนุษย์
  - ประหยัดเวลาและต้นทุนในการทดสอบแต่ละครั้ง
  - การทดสอบมีความถูกต้อง แม่นยำ และครอบคลุมมากขึ้น
  - เวลาที่ใช้ในการหาข้อผิดพลาดลดลง
  - สามารถสร้างกรณีทดสอบเพิ่มเติมได้ง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3) ผู้ใช้งานระบบ

- ผู้ใช้งานระบบสามารถใช้งานระบบได้อย่างถูกต้อง และตรงต่อความต้องการ
- ผู้ใช้สามารถลดเวลาในการรอการประมวลผลของข้อมูลมาแสดงบนหน้าจอ เพื่อความสะดวกในการให้บริการแก่ลูกค้า
- ผู้ใช้สามารถลดเวลาในการศึกษาระบบ และใช้ระยะเวลาในการเข้าถึงระบบลดน้อยลง
- ผู้ใช้สามารถดูรายงานต่าง ๆ เพื่อประกอบการตัดสินใจในการบริหารจัดการงานได้

## 1.6 ขั้นตอนการดำเนินงาน

- 1) ศึกษาขอบเขตโครงการสหกิจศึกษา และข้อมูลต่าง ๆ ที่เกี่ยวข้อง รวมไปถึงโปรแกรม และอุปกรณ์ที่นำมาใช้
- 2) ศึกษาหน้าที่และความรับผิดชอบต่าง ๆ ของตนเอง
- 3) ทดสอบและลองใช้งานเว็บแอปพลิเคชันบนอุปกรณ์ที่นำมาใช้งานจริง
- 4) นำปัญหาจากการทดสอบมาพัฒนาเป็นซอฟต์แวร์ทดสอบอัตโนมัติ เพื่อใช้ในการทดสอบ
- 5) หาข้อผิดพลาด และผลกระทบต่าง ๆ ที่ตามมา เมื่อระบบมีการทำงานผิดพลาด
- 6) รวบรวมและบันทึกปัญหาจากการทดสอบแอปพลิเคชันจากกลุ่มตัวอย่างผู้ใช้งาน
- 7) ทำความเข้าใจกับปัญหาที่เกิดขึ้นและสามารถอธิบายได้ เพื่อนำให้ส่วนที่เกี่ยวข้องนำไปแก้ไข
- 8) จัดทำคู่มือการใช้งาน
- 9) จัดทำรูปเล่มโครงการเตรียมนำเสนอ
- 10) นำเสนอโครงการสหกิจศึกษา

## 1.7 อุปกรณ์และเครื่องมือในการดำเนินงาน

- 1) ฮาร์ดแวร์
  - เครื่องคอมพิวเตอร์แบบพกพา จำนวน 2 เครื่อง
  - จอคอมพิวเตอร์ จำนวน 2 เครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2) ซอฟต์แวร์

- ระบบปฏิบัติการ Window 8 (64 bits) และ Window 8.1 (64 bit)
- Microsoft Office 2013
- Oracle SQL Developer
- Internet Explorer (IE)
- Google Chrome
- Mozilla Firefox
- Microsoft Paint
- Atlassian JIRA
- Selenium IDE
- Robot framework

## 3) ภาษาที่ใช้

- Java (Java Programming Language)
- Python (Python Programming Language)
- SQL (Structure Query Language)
- XML (Extensible Markup Language)
- HTML (Hyper Text Markup Language)

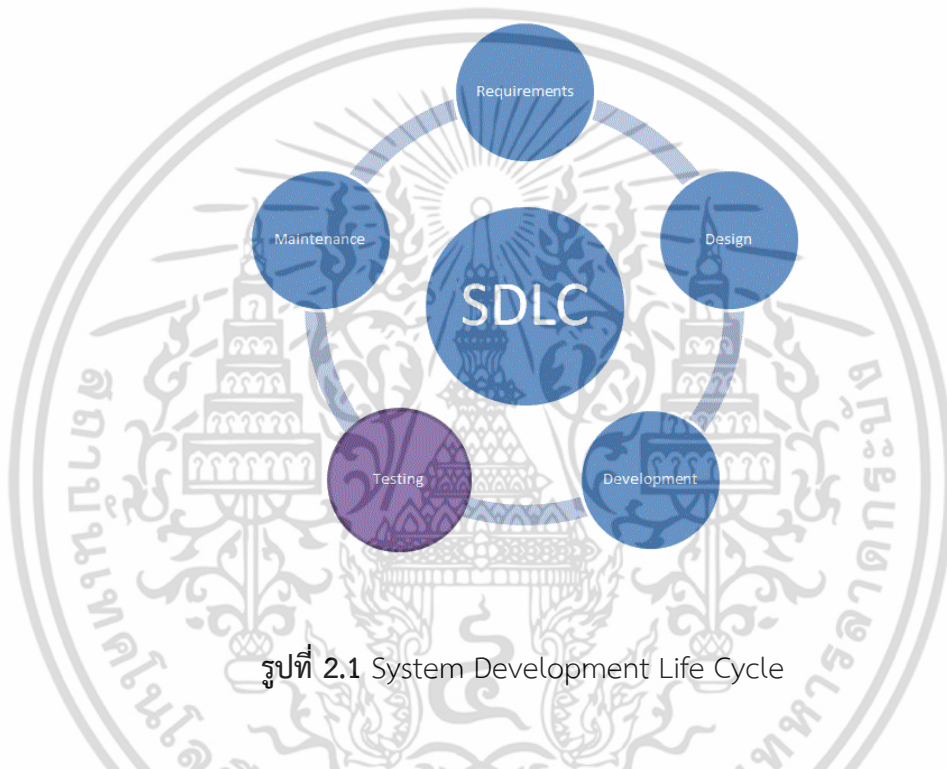
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

# ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงความรู้และทฤษฎีที่เกี่ยวข้องกับการโครงการงานสหกิจนี้ โดยผู้จัดทำได้ศึกษารวบรวม ข้อมูลและนำความรู้เหล่านี้ไปประยุกต์ใช้ในการวิเคราะห์และพัฒนาโปรแกรม ซึ่งทฤษฎีที่เกี่ยวข้องที่ได้รวบรวมไว้มีดังนี้

### 2.1 การทดสอบระบบ (Testing)



รูปที่ 2.1 System Development Life Cycle

เนื่องจากการสร้างระบบในแต่ละครั้งอาจเกิดข้อผิดพลาดต่าง ๆ ได้มากมาย จึงจำเป็นต้องมีผู้ทดสอบระบบ เพื่อให้ระบบมีความพร้อมในการใช้งาน และเกิดความสมบูรณ์ถูกต้องที่สุดก่อนที่จะส่งมอบให้กับผู้ใช้งาน โดยจะอยู่ในขั้นตอน Testing ในกระบวนการของ System Development Life Cycle (SDLC) ซึ่งการทดสอบระบบในแต่ละครั้งจะมีขั้นตอนมาตรฐานในการทดสอบ และควรทดสอบโดยผู้ทดสอบหลาย ๆ คน เพื่อไม่ให้เกิดการสับสน และให้ครอบคลุมทุก ๆ กรณี โดยจะทดสอบจากเอกสาร Software Requirement หรือเรียกอีกอย่างว่า Program Specification หรือ Detail Design ที่ถูกเขียนขึ้นโดย System Analysis (SA) ซึ่งในเอกสาร Program Specification จะเป็นรายละเอียดเกี่ยวกับความต้องการของผู้ใช้งาน โดยจะใช้ภาษาในการเขียนที่เข้าใจง่าย เรียกว่า ภาษาในระดับ High Level โดยการทดสอบระบบนั้นสามารถแบ่งได้เป็น 2 มุมมอง คือ การทดสอบระบบในมุมมองของ Programmer และการทดสอบระบบในมุมมองของ Tester ซึ่งความแตกต่างสามารถแบ่ง

ได้ตามตารางที่ 2.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 แนวความคิดในการทดสอบระบบระหว่าง Programmer และ Tester

Programmer	Tester
ทดสอบเฉพาะในส่วนของการเขียนโปรแกรม จะไม่ทดสอบในเชิง Business	ทดสอบในเชิง Business ซึ่งมีการ รายละเอียดต่าง ๆ เข้ามาเกี่ยวข้อง
ทดสอบเฉพาะชุดคำสั่ง (Code) ที่มีการ เปลี่ยนแปลง	ทดสอบการทำงานของระบบอย่างเป็น ขั้นตอน และทดสอบในส่วนที่จะเกิด ข้อผิดพลาด
ถ้ามีการแก้ไขเล็กน้อย จะไม่สนใจทำการ ทดสอบ และจะไม่ตรวจสอบว่ามีผลกระทบ ต่อส่วนอื่นหรือไม่	ทดสอบทั้งในส่วนที่มีการเปลี่ยนแปลง และ ไม่มีการเปลี่ยนแปลง
สร้างข้อมูลเพื่อใช้ในการทดสอบ โดยที่ไม่ ต้องเอามาจากระบบอื่น ๆ	ทดสอบระบบในทุก ๆ ด้านไม่ว่าจะเป็นเชิง บวก หรือเชิงลบ

### 2.1.1 ความแตกต่างระหว่าง Verification และ Validation [1]

- *Verification* คือ กระบวนการประเมิน ตรวจสอบ และควบคุมให้ Software ที่ถูกออกแบบ และพัฒนานั้นถูกต้องตรงตาม Requirement ของ User ซึ่งขั้นตอนนี้ต้องทำการตรวจสอบก่อนการพัฒนา Software
- *Validation* คือ กระบวนการตรวจสอบ Software ว่าถูกพัฒนาได้ตรงตาม Requirement ของ User หรือไม่ ซึ่งขั้นตอนนี้จะทำการตรวจสอบหลังจาก ได้พัฒนา Software ไปแล้ว

### 2.1.2 เทคนิคที่ใช้ในการทดสอบระบบ

- *Black Box* คือ การทดสอบโดยผู้ทดสอบระบบ (Tester) โดยที่จะสนใจ เพียงว่า ระบบมี Function การทำงานอย่างไร และผลลัพธ์ที่เกิดขึ้น ไม่สนใจชุดคำสั่ง (Code) ต่างๆในระบบ
- *White Box* คือ การทดสอบที่ผู้ทดสอบจำเป็นต้องมีความรู้เรื่อง Programming ซึ่งจะทดสอบการทำงาน และการออกแบบระบบต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.1.3 ประเภทของการทดสอบโปรแกรม

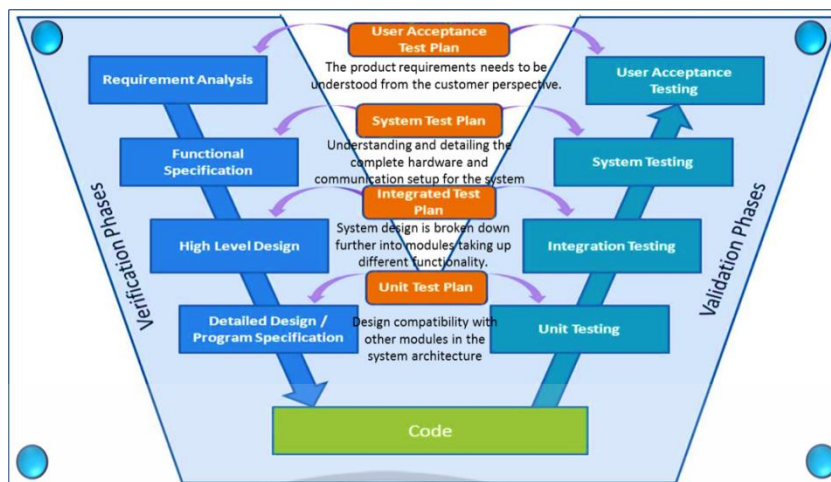
- *Functional Test* คือ จะใช้เทคนิคการทดสอบแบบ Black Box ซึ่งจะทำการทดสอบโดยผู้ทดสอบระบบ (Tester) โดยจะทดสอบว่าระบบนั้นๆ สามารถทำงานอะไรได้บ้าง
- *Non – Functional Test* คือ เป็นการทดสอบประสิทธิภาพ ความถูกต้อง ความสามารถ และความน่าเชื่อถือของระบบ ทั้งทางด้าน Software และ Hardware ซึ่งตัวอย่างของ Non – Function เช่น Performance Test (ทดสอบประสิทธิภาพของระบบ) Load Test (ทดสอบการรองรับปริมาณผู้ใช้งาน) เป็นต้น

### 2.1.4 ระดับ Software Testing มี 4 ระดับ คือ

- *Unit Test*: ทดสอบโดย Developer ซึ่งเป็นการทดสอบเบื้องต้น ขณะที่เขียนโปรแกรมในส่วนของตนเอง เช่น การทดสอบไวยากรณ์ (Syntax) หรือ Format ต่างๆ เป็นต้น เพื่อตรวจสอบผลการทำงานของแต่ละ Module ย่อยๆ ทั้งหมดของระบบให้ตรงตามทีออกแบบไว้ ซึ่งเทคนิคที่ใช้ในการทดสอบ คือ White Box และ Black Box
- *Integration Test*: ทดสอบส่วนของโปรแกรมโดยมี Function หรือ Feature มากกว่า 2 ขึ้นไป แต่โปรแกรมยังไม่ใหญ่มาก ซึ่งเป็นการทดสอบการส่งค่าตัวแปรต่างๆ ไปยังแต่ละหน้าว่าทำงานถูกต้องหรือไม่
- *System Test*: เป็นการทดสอบระบบที่ใกล้เคียงกับระบบจริงมากที่สุด ซึ่งจะมีการเรียกใช้งานค่าต่างๆ มาจากหลากหลายระบบ เพื่อตรวจสอบ (Verify) การทำงานของระบบว่าทำงานถูกต้องตาม Requirement หรือไม่ โดยสามารถทดสอบได้ทั้งแบบ Functional และ Non-Functional ขึ้นอยู่กับว่าระบบเหมาะสมกับแบบใด
- *E2E (End-to-End)*: เป็นการทดสอบก่อนส่งมอบให้ผู้ใช้งาน เรียกอีกอย่างว่า Acceptance Testing

*Regression Test* : จะเป็นการทดสอบเพื่อดูผลจากการเปลี่ยนแปลงในบางส่วน ซึ่งอาจจะไปกระทบกับส่วนอื่น ๆ ที่ไม่ได้ทำการเปลี่ยนแปลง โดย Regression Test นี้เปรียบเสมือนการทดสอบอีกครั้งใน Function เดิมที่ได้ทำการทดสอบไปแล้ว เพื่อตรวจสอบว่า Function นั้น ๆ ยังสามารถใช้งานได้เช่นเดิม ซึ่งการทำ Regression Test จะเกิดขึ้นก็ต่อเมื่อมี Requirement แจ้งมาว่าให้ทดสอบใหม่ทั้งระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 V Mode

### 2.1.5 รูปแบบการทดสอบระบบ สามารถแบ่งได้ 2 แบบ คือ

- *Manual Test*: เป็นการทดสอบระบบทีละขั้นตอน เมื่อทดสอบแล้วต้องนำมาเขียนเป็น Test Case เพื่อให้ผู้ทีนำไปอ่านต่อเข้าใจได้ง่าย
- *Automated Test*: เป็นการทดสอบระบบโดยการรัน Script หรือทดสอบซ้ำๆ เพื่อให้ได้ผลลัพธ์เหมือนเดิม ซึ่งถ้ามีการเปลี่ยนแปลงส่วนใดส่วนหนึ่งของโปรแกรมต้องทำการทดสอบใหม่ทั้งหมด

### 2.1.6 ความสำคัญของข้อผิดพลาด (Defect Priority)

- *Critical*: เป็นข้อผิดพลาดที่กระทบต่อระบบมากที่สุด ต้องได้รับการแก้ไขทันที ไม่เช่นนั้นระบบจะทำงานต่อไม่ได้
- *High/ Major*: เป็นข้อผิดพลาดที่กระทบต่อระบบมาก แต่ไม่จำเป็นต้องแก้ไขทันที โดยอาจเกิดจากการกรอกข้อมูลที่ถูกต้อง แต่ระบบแสดงผลการทำงานผิดพลาด เช่น การเกิด Error
- *Minor*: เป็นข้อผิดพลาดที่กระทบต่อระบบ แต่ระบบยังสามารถทำงานต่อได้ เช่น การส่งค่าผิด
- *Low*: เป็นข้อผิดพลาดที่กระทบต่อระบบน้อยที่สุด เช่น Design ผิดพลาดสีของหน้า Web Page เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.1.7 ระดับในการทดสอบระบบ [2]

- การทดสอบบนเครื่องผู้พัฒนา หรือเรียกอีกชื่อว่า Dev คือ การทดสอบโดยใช้เครื่องทดสอบที่มีประสิทธิภาพน้อยที่สุด เช่น Ram น้อย เป็นต้น ถ้าทดสอบผ่านจะไปทดสอบในระดับถัดไป
- การทดสอบในระดับ Staging หรือเรียกอีกชื่อว่า UAT คือ การทดสอบโดยให้ผู้ใช้งานเป็นผู้ทดสอบ ผู้พัฒนาโปรแกรมไม่สามารถทดสอบได้ เครื่องที่ใช้ในการทดสอบมีประสิทธิภาพเพิ่มมากขึ้น ถ้าทดสอบผ่านจะไปทดสอบในระดับถัดไป
- การทดสอบในระดับ Production หรือเรียกอีกชื่อว่า Prod คือ การทดสอบในระดับสุดท้าย ซึ่งเครื่องที่ใช้ในการทดสอบจะมีประสิทธิภาพมากที่สุด เช่น Ram เยอะที่สุด เป็นต้น

หมายเหตุ : การทดสอบในแต่ละระดับต้องมีจำนวนเครื่องในการทดสอบที่เท่ากัน

### 2.2 การทดสอบอัตโนมัติ (Automated Testing) [3]

Automate Testing คือการทดสอบซอฟต์แวร์ด้วยคอมพิวเตอร์แบบอัตโนมัติแทนการทดสอบโดยมนุษย์ โดยจะใช้เครื่องมือซึ่งช่วยในการรันชุดคำสั่ง ที่ใช้ในการทดสอบซอฟต์แวร์ที่ต้องการ และช่วยเปรียบเทียบผลลัพธ์จริงที่ได้จากระบบกับผลลัพธ์ที่ควรได้รับ เพื่อตรวจสอบว่าซอฟต์แวร์ที่ทดสอบนี้ทำงานได้ถูกต้องหรือไม่ โดยปกติในการทดสอบด้วยมือ (Manual Testing) นั้น ผู้ทดสอบ 1 คนสามารถทดสอบได้ที่ละ 1 เทสเท่านั้น ทำให้ต้องใช้เวลานานหรือใช้คนทดสอบจำนวนมาก ซึ่งจุดประสงค์หลักในการทำ Automated Testing ก็เพื่อลดจำนวนการทดสอบที่ต้องใช้คนลง เพื่อประหยัดเวลาและต้นทุนที่ใช้ในการทดสอบในแต่ละครั้ง โดยใช้เครื่องมือในการรันเทสอย่างอัตโนมัติ ไม่จำเป็นต้องมีคนคอยคุมขณะรัน และยังสามารถรันเทสหลายเทสได้พร้อมกันในเวลาเดียว ผู้ทดสอบเพียงแค่สั่งรันเทสแล้วรอผลลัพธ์ของการทดสอบ ซึ่งอยู่ในรูปของรายงานสรุปผลการทดสอบ

### 2.2.1 ประเภทของ Automated Test [4]

1) Functional Automated Test คือ การนำ Test Tool มาช่วยในการทดสอบระบบ หรือโปรแกรมว่าทำงานได้ตรงตาม Requirement/Functional หรือ Business Flow หรือไม่ โดยใช้ Tool เข้าไปจดจำการทำงานหรือการ Record จากหน้าจอโปรแกรมหรือ Web หลังจากนั้น Tool จะ Generate Script ขึ้นมา Script ที่ถูกสร้างขึ้นมานำมาปรับปรุง เพื่อใช้ในการ Execute ซ้ำ แล้วจะบอกได้ว่าที่ Execute ไปนั้น Pass หรือ Fail

2) Performance Automated Test คือ Tool จะทำการ Record Scenario ตามที่เรารับบันทึกไว้ และจะจับ Data ที่รับส่งกันระหว่าง Client กับ Server ว่ามีการส่งอะไร กันในระดับ Protocol (TCP, UDP, HTTP, SMTP) จากนั้น ก็จะสร้าง Virtual User เพื่อจำลองจำนวน User ในระบบ แล้วยิงข้อมูลเหล่านี้เข้าไปที่ Server เพื่อดูว่า Server จะ ทำงานได้จริงหรือไม่ นอกจากนั้นยังกำหนด Scenario ต่าง ๆ ของการ Test ได้ด้วย เช่น ทำ Stress Test โดยเพิ่ม User ทีละ 10 คน เพิ่มทุก ๆ 10 นาทีจนถึง 1000 คน เพื่อ ดูว่าระบบ หรือโปรแกรม จะล้มตอนไหนหรือจำลองช่วงเวลา ที่ User จะ Peak ขึ้นมา เช่น ระบบค้าหลักทรัพย์ จะมีลูกค้า เข้าระบบเยอะช่วง 9:45-10:00 (ก่อนตลาดเปิด ) 12:25-12:30 (ช่วงพักกินข้าว) 14:25-14:30 (ช่วงเปิดบ่าย) 16:25-16:40 (ก่อนปิดตลาด) เป็นต้น ทำให้เราสามารถ Test ระบบ Load Balancing ได้ว่ารองรับการทำงานได้จริงหรือไม่

3) Security Automated Test Tool ที่ช่วยทดสอบ เรื่องความปลอดภัยของระบบ หรือ เพื่อดูเรื่องความปลอดภัยของการเข้าใช้งานระบบ ว่ามีช่อง (Vulnerable) ให้ผู้ไม่ประสงค์ ดี เข้ามาลักลอบนำข้อมูลออก หรือเข้ามาทำลายระบบ

4) Test Management เป็น Tool สำหรับบริหารจัดการ Test Resource และ จัดการ Progress ของการ Test สามารถใช้เก็บ Requirement หรือ ติดตามงาน แล้วยังสามารถนำเอา Test Script ของ Automation ขึ้นเก็บได้ด้วย คือแทนที่จะมานั่ง Run script แล้วกลับมาบอกว่า Test ผ่านหรือไม่ผ่าน ก็เอา Test Script ใส่เข้าไป แล้วตั้งเวลา Execute ตอนเย็น หรือจะสั่งเป็น Schedule Run ตอนเช้า แล้วค่อยมาดูผลว่า Pass หรือ เท่าไหร่

### 2.2.2 ลักษณะของเทสเคสที่เหมาะสมในการทำ Automated Testing

- 1) เทสเคสที่มีการทดสอบซ้ำ ๆ และไม่บ่อยมีการเปลี่ยนแปลง
- 2) เทสเคสที่ทดสอบด้วยมือนั้นเป็นไปได้ยาก
- 3) เทสเคสที่ต้องทดสอบเป็นเวลานาน
- 4) เทสเคสที่มีความสำคัญมากและมีความเสี่ยงสูง ถ้าการทดสอบด้วยมืออาจผิดพลาด
- 5) เทสเคสที่มีการทดสอบฟังก์ชันเดิมๆ แต่ข้อมูลที่ใช้ในการทดสอบมีหลายชุด เช่น ฟังก์ชันล็อกอิน ฟังก์ชันการค้นหา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.2.3 ข้อดีของการทดสอบอัตโนมัติ

- 1) ช่วยลดเวลาและค่าใช้จ่ายในการทดสอบ ในกรณีที่มีการแก้ไขซอร์สโค้ดจำเป็นต้องมีการทดสอบซ้ำเสมอ ซึ่งการทดสอบอัตโนมัติหลังจากที่มีการสร้างการทดสอบแล้วสามารถนำไปใช้ซ้ำได้โดยไม่ต้องมีค่าใช้จ่ายเพิ่มเติม
- 2) ช่วยเพิ่มความถูกต้องในการทำงาน การทดสอบอัตโนมัติสามารถทำงานได้อย่างถูกต้องได้ผลลัพธ์เท่ากันทุกครั้ง
- 3) ช่วยเพิ่มความครอบคลุมการทดสอบ การทดสอบอัตโนมัติเพิ่มระดับความลึกและขยายขอบเขตของการทดสอบได้มากขึ้น โดยมีความสามารถประมวลผล Test case ที่มีความซับซ้อนและมีจำนวนมากได้โดยง่าย
- 4) สามารถทำงานที่การทดสอบด้วยมือทำไม่ได้ เช่นในกรณีที่ต้องการทดสอบเว็บแอปพลิเคชันที่ต้องการรองรับผู้ใช้จำนวนมาก และผู้ใช้มีการเข้าถึงแอปพลิเคชันในเวลาใกล้เคียงกัน เป็นต้น

## 2.3 เครื่องมือที่ใช้ในการทดสอบอัตโนมัติ

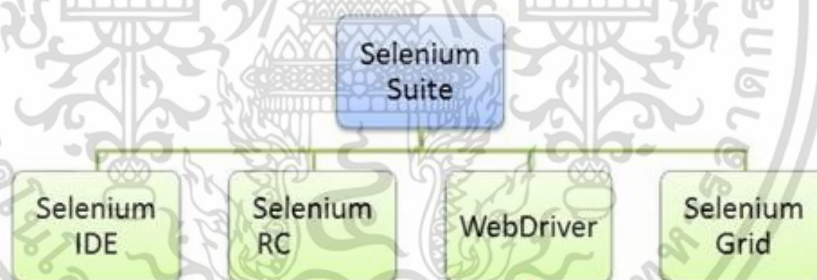
ในทางปฏิบัติแล้วการทดสอบฟังก์ชันการทำงานของระบบใดก็ตามเริ่มต้นจากการสร้าง Test Scenario ก่อนจากนั้นจึงถูกแปลงให้อยู่ในรูปของข้อมูลที่ใช้ในการทดสอบด้วยมือ ซึ่งประกอบไปด้วยหลายขั้นตอนของการทดสอบที่ถูกสร้างขึ้นในรูปของภาษาเขียนที่ใช้ทั่วไป ซึ่งแต่ละขั้นตอนของการทดสอบส่วนใหญ่นั้นจะเป็นชุดคำสั่งเฉพาะที่ใช้ในการทดสอบการทำงานของระบบ ที่อยู่ภายใต้การทดสอบเป็นหลัก แต่อย่างไรก็ตามการทดสอบด้วยมือที่ปกติจะถูกดำเนินการ โดยนักทดสอบที่นั่งอยู่หน้าจอคอมพิวเตอร์และประมวลผลการทดสอบตามขั้นตอนที่กำหนดไว้ ซึ่งการทำงานในลักษณะนี้อาจก่อให้เกิดความเบื่อหน่ายในการทำงาน โดยเฉพาะอย่างยิ่งการทดสอบที่จำเป็นต้องทำซ้ำๆ ติดต่อกัน เป็นเวลานาน นอกจากนั้นการทดสอบด้วยวิธีการนี้ยังมีค่าใช้จ่ายในการลงทุนด้านทรัพยากรมนุษย์ค่อนข้างสูง เนื่องจากปริมาณงานที่มากจำเป็นต้องใช้นักทดสอบมากขึ้นตามไปด้วย และข้อจำกัดในเรื่องของเวลาที่บ่อยครั้งส่งผลให้การทดสอบด้วยมือในทุกๆ ฟังก์ชันการทำงานของระบบก่อนการส่งมอบเป็นสิ่งที่เป็นไปไม่ได้และบ่อยครั้งที่ผู้ใช้ระบบอาจพบว่าข้อบกพร่องที่ยังค้นหาไม่พบจากการทดสอบในลักษณะดังกล่าวยังคงหลงเหลืออยู่ภายในระบบเสมอ ในเวลาต่อมาจึงได้มีการคิดค้นกระบวนการทดสอบอัตโนมัติขึ้นเพื่อช่วยในการแก้ไขปัญหาดังกล่าว โดยการแปลงขั้นตอนการทดสอบด้วยมือให้เป็นขั้นตอนการทดสอบอัตโนมัติโดยอยู่ในรูปของสคริปต์ที่สามารถนำไปประมวลผลบนเครื่องมือทดสอบแบบอัตโนมัติ

### 2.3.1 Selenium [5]

Selenium เป็นกลุ่มเครื่องมือทดสอบต่าง ๆ ซึ่งแต่ละชนิดมีวิธีการทำงานร่วมกับการทดสอบอัตโนมัติที่แตกต่างกัน โดยนักทดสอบส่วนใหญ่จะเน้นไปที่การใช้เครื่องมือเพียงหนึ่งหรือสองชนิดเท่านั้นเพื่อให้เป็นไปตามความต้องการในการทดสอบ แต่อย่างไรก็ตามการใช้เครื่องมือทั้งหมดร่วมกับ การทดสอบอัตโนมัติจะช่วยทำให้การทดสอบสามารถครอบคลุมการทำงานของเว็บแอปพลิเคชันได้ทุก ๆ มิติ ข้อดีประการหนึ่งของการใช้ Selenium ในการทดสอบอัตโนมัติคือความสามารถในการรองรับการทำงานร่วมกันกับบราวเซอร์ และระบบปฏิบัติการที่แตกต่างกันได้ ส่งผลให้นักทดสอบสามารถทดสอบระบบที่มีความหลากหลาย ได้มากขึ้น ดังที่ได้กล่าวมาแล้วว่า Selenium ถูกออกแบบมาเพื่อวัตถุประสงค์ในการใช้งานที่แตกต่างกัน

กลุ่มของเครื่องมือในการทดสอบ (Selenium Suite)

- 1) Selenium Integrated Development Environment (IDE)
- 2) Selenium Remote Control (RC)
- 3) WebDriver
- 4) Selenium Grid



รูปที่ 2.3 กลุ่มของเครื่องมือทดสอบที่ใช้ใน Selenium

Selenium ถูกพัฒนาขึ้นเป็นครั้งแรกในปี 2004 และมีพัฒนาการต่อเนื่องมาจนถึงปัจจุบัน ซึ่งในแต่ละช่วงเวลาของการพัฒนาได้มีการเพิ่มส่วนประกอบต่างๆ เข้าไปเพื่อให้มีความสามารถ ในการทำงานมากยิ่งขึ้น ในขณะที่เดียวกันได้มีการตัดบางส่วนของที่ล้าสมัยหรือไม่จำเป็นออกไปเพื่อให้ เครื่องมือมีความทันสมัยมากขึ้น ซึ่งพัฒนาการดังกล่าวสามารถแบ่งออกได้ตามช่วงเวลาที่สำคัญๆ ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.3.1.2 Selenium Core

ในปี 2004 Jason Huggins วิศวกรของบริษัท ThoughtWorks ซึ่งทำหน้าที่ในการพัฒนาเว็บแอปพลิเคชันที่ต้องการทดสอบบ่อยๆ ซึ่งจากการทำงานดังกล่าวจะทำให้ก่อให้เกิดแนวคิดที่ว่า การทดสอบด้วยมือซ้ำๆ ส่งผลให้ประสิทธิภาพในการทดสอบลดลง ดังนั้นจึงได้พัฒนาโปรแกรมด้วยภาษา สคริปต์โดยใช้ชื่อว่า JavaScriptTestRunner เพื่อใช้สำหรับควบคุมการกระทำของบราวเซอร์โดย อัตโนมัติ ซึ่งในเวลาต่อมาได้ถูกเปลี่ยนชื่อเป็น Selenium Core และถือเป็นพื้นฐานการทำงานที่สำคัญ สำหรับ Selenium Remote Control (RC) และ Selenium IDE ที่ถูกพัฒนาขึ้นในเวลาต่อมา

#### Same Origin Policy

เป็นชื่อของแนวคิดด้านความปลอดภัยที่ใช้กับจาวาสคริปต์ในการโปรแกรมบนฝั่งไคลเอนต์ โดยป้องกันไม่ให้จาวาสคริปต์สามารถเข้าถึงส่วนประกอบใด ๆ ของโดเมนที่แตกต่าง ไปจากโดเมนที่ถูกเรียกใช้ครั้งแรก ยกตัวอย่างเช่น โค้ด HTML ภายใน www.google.com ถูกใช้งาน ร่วมกับจาวาสคริปต์ ซึ่งกลไกที่ใช้ Same Origin Policy จะยอมให้จาวาสคริปต์ดังกล่าวสามารถเข้าถึงเว็บได้เฉพาะภายในโดเมน google.com เท่านั้น ซึ่งในกรณีนี้ได้แก่ google.com/mail, google.com/login, เป็นต้น แต่ในทางตรงกันข้ามกลไกดังกล่าวจะไม่ยอม ให้จาวาสคริปต์ดังกล่าวสามารถเข้าถึงเว็บเพจจากโดเมนที่แตกต่างกันไปจากwww.google.comได้ เช่น www.yahoo.com/ หรืออื่น ๆ เป็นต้น ทั้งนี้เนื่องจากเป็นโดเมนที่แตกต่างไปจากโดเมน ที่ถูกเรียกใช้ครั้งแรกว่านเอง ซึ่งแนวคิดในการทำงานดังกล่าวสามารถนำเสนอได้ดังรูปต่อไปนี้



รูปที่ 2.4 แนวคิดในการทำงานที่เรียกว่า Same Origin Policy

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.3.1.3 Selenium RC

Selenium RC ย่อมาจากคำว่า Selenium Remote Control ซึ่งถูกพัฒนาขึ้นเพื่อแก้ไขข้อจำกัดจากแนวคิด แบบ Same Origin Policy โดย Paul Hamant วิศวกรอีกคนหนึ่งจากบริษัท ThoughtWork ได้ พัฒนาโปรแกรมที่ใช้เป็นเซิร์ฟเวอร์เพื่อทำหน้าที่เป็น HTTP proxy เพื่อใช้สำหรับบล็อกเบราว์เซอร์ ให้เชื่อว่า Selenium Core และเว็บแอปพลิเคชันได้ถูกทดสอบจากโดเมนเดียวกัน ซึ่งการทำงานใน ลักษณะดังกล่าวจึงเป็นที่มาของ Selenium RC หรือ Selenium 1 ในเวลาต่อมา นอกจากนี้ ยังส่งผล ให้นักทดสอบที่ใช้ Selenium Core จำเป็นต้องติดตั้งทั้งเว็บแอปพลิเคชันที่ ต้องการทดสอบและ Selenium RC ที่เป็นเซิร์ฟเวอร์ลงในเครื่องเดียวกันตามไปด้วย

Selenium RC นอกจากจะใช้เป็นเครื่องมือทดสอบอัตโนมัติสำหรับเว็บ แอปพลิเคชันในช่วงต้นๆ แล้ว ยังสนับสนุนการทำงานร่วมกับโปรแกรมภาษาต่างๆ ได้แก่ Java, C#, PHP, Python, Perl, Ruby เป็นต้น นอกจากนี้ Selenium RC อาศัยการทำงานแบบโคลเอนต์เซิร์ฟเวอร์ ที่ยอมให้ผู้ใช้ควบคุมการทำงานของ บราวเซอร์จากภายในเครื่องเดียวกันหรือจากเครื่อง คอมพิวเตอร์อื่น ๆ ได้ สำหรับ ข้อดีและข้อเสียของการใช้ Selenium RC สามารถสรุปได้ดังตาราง ต่อไปนี้

ตารางที่ 2.2 สรุปข้อดีและข้อเสียของการใช้งานร่วมกับ Selenium RC

ข้อดี	ข้อเสีย
สามารถทำงานร่วมกับเบราว์เซอร์และระบบปฏิบัติการที่แตกต่างกันได้	การติดตั้งมีความยุ่งยากกว่า Selenium IDE
สนับสนุนการทำงานแบบวนซ้ำ (Loop) และเงื่อนไขต่าง ๆ (if/switch)	นักทดสอบต้องมีประสบการณ์ทางด้าน การโปรแกรมมาก่อน
สนับสนุนการทำงานแบบ Data Driven Testing	ต้องการ Selenium RC เซิร์ฟเวอร์สำหรับรัน การทำงาน
มี API ที่สมบูรณ์และพร้อมใช้งาน	มี API มากเกินจำเป็นและบางคำสั่งซ้ำซ้อนกัน ก่อให้เกิดความสับสน
สนับสนุนการทำงานของเบราว์เซอร์รุ่นใหม่ ๆ	ไม่สามารถลำดับการรันการทดสอบในกรณีที่มี หลาย ๆ Test Case

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.3.1.4 Selenium Grid

Grid Selenium Grid ถูกพัฒนาขึ้นโดย Patrick Lightbody เพื่อจุดประสงค์ในการลดเวลาที่ใช้ในการประมวลผล การทดสอบ โดยใช้แนวคิดของการประมวลผลการทดสอบแบบขนาน นั่นคือการรันหลาย ๆ การทดสอบที่แตกต่างกันในเวลาเดียวกันสามารถเกิดขึ้นได้ในบราวเซอร์และระบบปฏิบัติการที่ แตกต่างกัน โดยการผ่านคำสั่ง Selenium ไปยังเครื่องคอมพิวเตอร์อื่น ๆ เพื่อการประมวลผลการ ทดสอบในเวลาเดียวกันได้ นั่นคือผู้ใช้สามารถใช้ Selenium Grid เป็นเครื่องมือที่ใช้ในการทดสอบร่วมกับ Selenium RC เพื่อรันการทำงานแบบขนานได้ ปัจจุบัน Selenium Grid สามารถแบ่งออก ได้เป็น 2 เวอร์ชัน ได้แก่ Grid 1 ซึ่งเป็นเวอร์ชันเก่าและ Grid 2 เป็นเวอร์ชันที่ใช้งานอยู่ในปัจจุบัน Selenium Grid ใช้แนวคิดในการทำงานแบบ Hub-Node โดยผู้ใช้สามารถรันการทดสอบเฉพาะบนเครื่องคอมพิวเตอร์ ซึ่งทำหน้าที่เป็น Hub และผ่านคำสั่งไปยังคอมพิวเตอร์ซึ่งทำหน้าที่เป็น Nodes เพื่อประมวลผลการทดสอบในเวลาเดียวกันดังรูปต่อไปนี้



รูปที่ 2.5 แสดงการทำงานของ Selenium Grid

การเลือกใช้งานของ Selenium Grid จะพิจารณาจากการทำงานสองลักษณะดังต่อไปนี้

- รันการทดสอบร่วมกับบราวเซอร์ ระบบปฏิบัติการ และเครื่องที่แตกต่างกันในช่วงเวลาเดียวกัน ทั้งนี้เพื่อให้แน่ใจว่าเว็บแอปพลิเคชันที่ถูกทดสอบสามารถทำงานร่วมกับ บราวเซอร์และระบบปฏิบัติการที่แตกต่างกันได้
- ต้องการลดเวลาในการประมวลผลการทดสอบ เช่น สำหรับในกรณีที่กำหนดให้ Selenium Grid รัน 4 การทดสอบในเวลาเดียวกัน ผู้ใช้จะสามารถทดสอบได้เร็วขึ้น 4 เท่านั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.3 แสดงความแตกต่างหลัก ๆ ระหว่าง Selenium Grid 1 และ 2

Grid 1	Grid 2
มี remote control ของตัวเองที่แตกต่างไปจาก Selenium RC เซิร์ฟเวอร์	ได้มีการรวบรวม Selenium Server jar ไฟล์ไว้ด้วยกัน
ผู้ใช้จำเป็นต้องติดตั้งและคอนฟิก Apache Ant ก่อนการเรียกใช้ Grid 1	ไม่จำเป็นต้องติดตั้ง Apache Ant ใน Grid 2
สนับสนุนการทำงานเฉพาะคำสั่งหรือสคริปต์ Selenium RC	สนับสนุนการทำงานทั้ง Selenium RC และ WebDriver สคริปต์
ผู้ใช้สามารถรันอัตโนมัติได้เฉพาะหนึ่ง browser ต่อหนึ่ง remote control	หนึ่ง remote control สามารถรันอัตโนมัติได้ 5 browser

### 2.3.1.5 Selenium IDE

Selenium IDE ถูกพัฒนาขึ้นโดยชาวญี่ปุ่นที่ชื่อ Shinya Kasatani ในรูปส่วนต่อขยายของ Firefox ที่สามารถใช้ในการทำงานแบบอัตโนมัติร่วมกับ browser ผ่านเทคนิคการทดสอบที่เรียกว่า Record & Playback โดยมีจุดประสงค์เพื่อให้เป็นเครื่องมือทดสอบที่ใช้งานได้ง่าย โดยผู้ใช้ไม่จำเป็นต้องมีพื้นฐานความรู้ด้านการโปรแกรมมาก่อน นอกจากนั้นยังเป็นเครื่องมือที่ช่วยเพิ่มความเร็วในการสร้าง Test Case ซึ่งในเวลาต่อมาจึงได้มอบ Selenium IDE ให้กับโครงการพัฒนา Selenium ในปี 2006 Selenium IDE จัดเก็บสคริปต์สำหรับการทดสอบในรูปแบบของตารางที่สามารถนำกลับมาใช้ใหม่เพื่อการประมวลผลซ้ำได้ในภายหลัง แต่อย่างไรก็ตามตัวโปรแกรมเองยังคงขาดคุณสมบัติที่สำคัญอื่น ๆ ได้แก่ คุณสมบัติในการรันผลลัพธ์การทดสอบหรือความสามารถในการสร้างการทดสอบอัตโนมัติทั้งหมดที่ผู้ใช้ต้องการ นอกจากนั้น Selenium IDE ยังไม่สนับสนุนการทำงาน ประเภททำซ้ำหรือเงื่อนไขต่าง ๆ สำหรับการใช้งานร่วมกับสคริปต์สำหรับการทดสอบ ทั้งนี้ เนื่องจากเหตุผลบางส่วนที่เป็นปัญหาทางเทคนิคและบางส่วนจากตัวนักพัฒนาเอง อย่างไรก็ตาม เนื่องจากความง่ายต่อการใช้งาน Selenium IDE จึงถูกจำกัดให้เป็นได้เฉพาะเครื่องมือต้นแบบที่สามารถใช้ในการทดสอบแบบพื้นฐานเท่านั้น ส่วนในกรณีที่ต้องการสร้าง Test Case ที่มีการทำงานแบบซ้ำซ้อนเป็นต้องเลือกใช้ Selenium RC หรือ WebDriver แทน

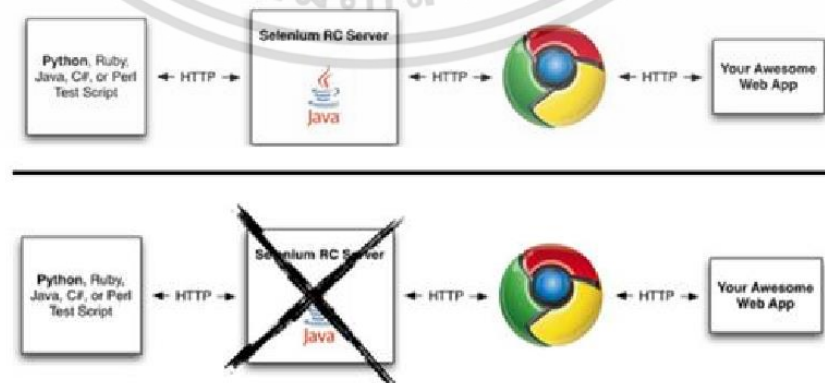
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.4 สรุปข้อดีและข้อเสียของการใช้งานร่วมกับ Selenium IDE

ข้อดี	ข้อเสีย
สะดวกต่อการติดตั้งและใช้งาน	ใช้ได้เฉพาะกับบราวเซอร์ Firefox
ไม่จำเป็นต้องมีประสบการณ์ด้านโปรแกรมมาก่อน ใช้ความรู้เฉพาะ HTML และ DOM	ออกแบบมาเพื่อใช้เป็นเครื่องมือต้นแบบ เท่านั้น
สามารถส่งออกการทดสอบไปยังรูปแบบที่ใช้ได้กับ Selenium RC และ WebDriver	ไม่สนับสนุนการทำงานแบบทำซ้ำหรือแบบมีเงื่อนไข
มี help และโมดูลที่ถูกสร้างไว้แล้ว สำหรับรายงานผลการทดสอบ	การประมวลผลการทดสอบช้าเมื่อเปรียบเทียบกับ selenium RC และ WebDriver

### 2.3.1.6 WebDriver

ในปี 2006 Simon Stewart ได้พัฒนา WebDriver ขึ้นโดยเพิ่ม API ที่เป็นทางเลือกสำหรับ ฟังก์ชันการทำงานที่ไม่ได้รับการสนับสนุนจาก Selenium-RC โดยการทำงานของ WebDriver จะ ไม่ขึ้นอยู่กับจาวาสคริปต์ที่ฝังตัวอยู่ในบราวเซอร์ ดังนั้นจึงสามารถลดข้อจำกัดที่เกิดขึ้นจากการ ทำงานของ Selenium ได้ นอกจากนี้ WebDriver ยังมีเป้าหมายหลักในการพัฒนาเครื่องมือทดสอบอัตโนมัติเพื่อให้สามารถทำงานข้ามแพลตฟอร์มได้รวมถึงความสามารถในการควบคุมการทำงานร่วมกับบราวเซอร์ได้ในระดับระบบปฏิบัติการ สำหรับความแตกต่างระหว่าง Selenium RC และ WebDriver สามารถนำเสนอได้จากรูปดังต่อไปนี้



รูปที่ 2.6 เปรียบเทียบการทำงานระหว่าง Selenium RC และ WebDriver

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่ขึ้นด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

WebDriver ได้ถูกออกแบบมาเพื่อแก้ไขข้อจำกัดต่าง ๆ ที่เพิ่มมากขึ้นจากการใช้ จาวาสคริปต์ รวมไปถึงข้อบกพร่องต่าง ๆ ที่เกิดขึ้นทั้งจาก Selenium IDE และ Selenium RC โดยมี ชุดคำสั่งที่ถูกออกแบบมาเป็นแบบเชิงวัตถุส่งผลให้การนำไปใช้ สามารถทำงานได้ง่ายยิ่งขึ้น รวมทั้งยังสนับสนุนการทำงานกับโปรแกรมภาษาต่างๆ เช่นเดียวกับ Selenium RC นอกจากนี้ยังมีการทำงานที่เสถียรกว่าในการทดสอบ อัตโนมัติร่วมกับบราวเซอร์ชนิดต่างๆ

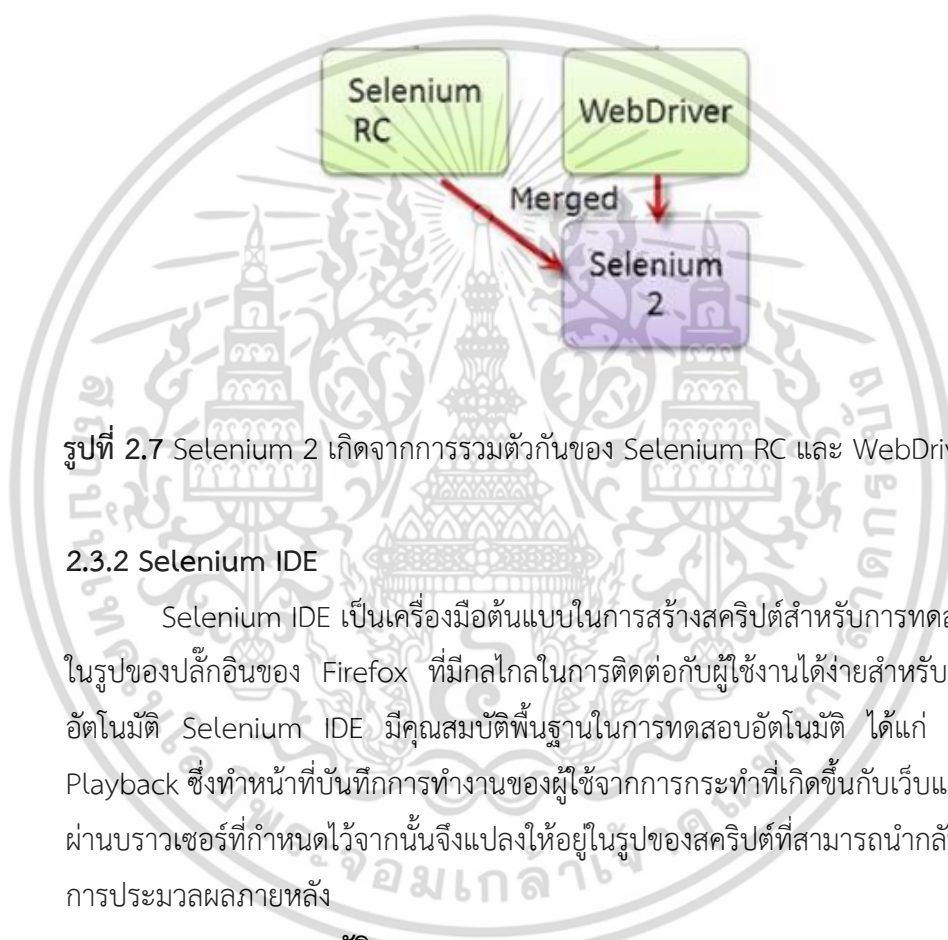
### ตารางที่ 2.5 สรุปข้อดีและข้อเสียของการใช้งานร่วมกับ WebDriver

ข้อดี	ข้อเสีย
การติดตั้งง่ายกว่า Selenium RC	การติดตั้งยุ่งยากกว่า Selenium IDE
ติดต่อโดยตรงกับบราวเซอร์	ต้องการความรู้ด้านโปรแกรมมาก่อน
การติดต่อกับบราวเซอร์มีความเสมือนจริง	ไม่สนับสนุนบราวเซอร์รุ่นใหม่ ๆ
ไม่ต้องใช้คอมพิวเตอร์อื่น เช่นเดียวกับใน Selenium RC Server	ไม่มีกลไกสำหรับแสดงข้อความเตือนในช่วงเวลา รันใหม่และสร้างผลลัพธ์การทดสอบ
ประมวลผลได้เร็วกว่า Selenium IDE และ Selenium RC	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.3.1.7 Selenium 2

ในปี 2008 ทีมพัฒนา Selenium ทั้งหมดได้ตัดสินใจรวม WebDriver และ Selenium RC เข้าด้วยกันเพื่อสร้างเครื่องมือชนิดใหม่ที่มีประสิทธิภาพมากขึ้นเรียกว่า Selenium 2 โดยใช้ WebDriver เป็นแกนหลักในการทำงาน ปัจจุบัน Selenium RC ไม่มีการพัฒนาเวอร์ชันใหม่อีกต่อไปแล้ว มีแต่เพียงส่วนของการบำรุงรักษาเท่านั้น ดังนั้นแนวโน้มของกาพัฒนาเครื่องมือทดสอบปัจจุบันจะ เน้นไปที่ Selenium 2 เสมอ



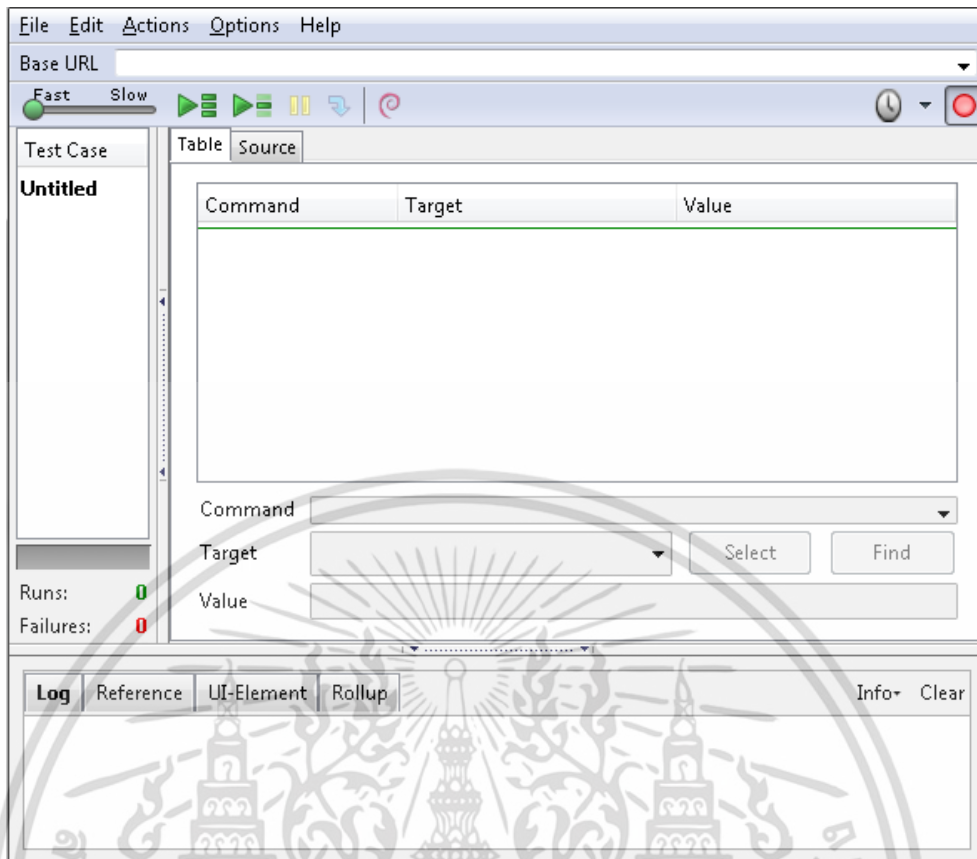
รูปที่ 2.7 Selenium 2 เกิดจากการรวมตัวกันของ Selenium RC และ WebDriver

### 2.3.2 Selenium IDE

Selenium IDE เป็นเครื่องมือต้นแบบในการสร้างสคริปต์สำหรับการทดสอบ โดยอยู่ในรูปของปลั๊กอินของ Firefox ที่มีกลไกในการติดต่อกับผู้ใช้งานได้ง่ายสำหรับการทดสอบอัตโนมัติ Selenium IDE มีคุณสมบัติพื้นฐานในการทดสอบอัตโนมัติ ได้แก่ Record & Playback ซึ่งทำหน้าที่บันทึกการทำงานของผู้ใช้จากการกระทำที่เกิดขึ้นกับเว็บแอปพลิเคชันผ่านเบราว์เซอร์ที่กำหนดไว้จากนั้นจึงแปลงให้อยู่ในรูปของสคริปต์ที่สามารถนำกลับไปใช้ในการประมวลผลภายหลัง

#### 2.3.2.1 คุณสมบัติของ Selenium IDE

ดังที่กล่าวมาแล้วว่า Selenium IDE เป็นเครื่องมือที่ใช้สำหรับการทดสอบอัตโนมัติร่วมกับเว็บแอปพลิเคชัน ดังนั้นจึงประกอบไปด้วยเครื่องมือที่ช่วยอำนวยความสะดวกในการทำงานต่างๆ ดังต่อไปนี้



รูปที่ 2.8 แสดงส่วนประกอบของคำสั่งที่ใช้ใน Selenium IDE

#### 1) เมนูบาร์ (Menu Bar)

เป็นเมนูหลักที่ประกอบไปด้วยเมนูต่างๆ โดยเมนูแรกได้แก่ File ซึ่งประกอบไปด้วยเมนูย่อยที่เป็นทางเลือกต่างๆ ที่ใช้สำหรับการทำงาน ของ Test Case และ Test Suite ตัวอย่างเช่น การสร้างใหม่ เปิด และ บันทึก รวมไปถึงการส่งออก Test Case ส่วนถัดมา ได้แก่ เมนู Edit ซึ่ง ยอมให้ผู้ใช้สามารถ Copy, Paste, Delete, Undo และเลือกการทำงาน ทั้งหมดรวมไปถึงความสามารถในการแทรกคำสั่งหรือคอมเมนต์ใหม่ลงใน Test Case เมนูถัดมาได้แก่ เมนู Action ซึ่งใช้สำหรับการ Record และ Play สคริปต์ที่ถูกบันทึกไว้ นอกจากนี้ยังประกอบไปด้วยเมนูย่อยที่ใช้ ร่วมกับการดีบั๊ก

เมนูถัดมาได้แก่ เมนู Options ซึ่งยอมให้ผู้ใช้สามารถ เปลี่ยนแปลงออปชันในการติดตั้งได้ตัวอย่างเช่น ผู้ใช้สามารถกำหนดค่า TimeOut สำหรับคำสั่งที่ต้องการ เพิ่มส่วนขยายเพื่อช่วยในการทำงาน

ส่วนเมนูลำดับสุดท้ายได้แก่ เมนู Help ซึ่งเกี่ยวข้องกับ ความช่วยเหลือที่อยู่ในรูปของเอกสารออนไลน์เป็นหลัก และส่วนที่เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มาตรฐานและถูกกำหนดโดย Firefox มีเพียงหนึ่งรายการเท่านั้นที่เป็นของ Selenium IDE ได้แก่ UI-Element Documentation

## 2) ทูลบาร์ (Toolbar)

เป็นเมนูหลักที่จะประกอบไปด้วยเมนูต่างๆ ซึ่งมีหน้าที่สำหรับควบคุมประมวลผล Test Case รวมไปถึงคุณสมบัติที่ใช้ในการดีบั๊กต่างๆ

## 3) Test Case Pane

สคริปต์จากการ Record จะถูกนำเสนอภายใน Test Case ที่ประกอบไปด้วยสอง Tab ได้แก่ Table Tab ที่ใช้แสดงคำสั่งในการทำงาน และ Source Tab ที่ใช้แสดงผลในรูปของโค้ดที่ถูกระบุ

Selenium IDE จะแบ่ง Table ออกเป็นสองส่วนตามลำดับการนำเสนอ โดยส่วนแรกด้านบนใช้สำหรับการนำเสนอชุดคำสั่งที่ใช้ โดยแต่ละแถวของชุดคำสั่งดังกล่าวจะถูกแบ่งออกเป็นสามคอลัมน์ซึ่ง ได้แก่ Command, Target และ Value ตามลำดับ ส่วนที่สองด้านล่างเป็นการนำเสนอรายละเอียดของชุดคำสั่งเช่นเดียวกันแต่อยู่ในรูปของฟิลด์กรอก ข้อมูลที่ถูกนำเสนอพร้อมกับชุดคำสั่งปัจจุบันที่ใช้ ซึ่งจากฟิลด์ดังกล่าวผู้ใช้สามารถแก้ไขคำสั่งปัจจุบันได้ตามต้องการ อย่างไรก็ตามการนำเสนอคำสั่งที่ใช้ในการทำงานทั้งสองส่วนภายใน Table จะต้องมีความสัมพันธ์กันเสมอ

ส่วน Tab ในลำดับถัดไปได้แก่ Source ซึ่งเป็นรูปแบบปกติของการจัดเก็บ Test Case ไว้ภายในไฟล์ โดยชนิดของไฟล์ที่เป็นค่า default ได้แก่ ไฟล์ HTML แต่เนื่องจาก Selenium สนับสนุนการทำงานร่วมกับโปรแกรมภาษาได้แก่หลายชนิด ดังนั้นผู้ใช้จึงสามารถเลือกชนิดของการจัดเก็บไฟล์ได้ตามโปรแกรมภาษาที่กำหนดไว้ เช่น Java, Php, C#, หรือภาษาสคริปต์เช่น Python เป็นต้น นอกจากนั้นแล้วส่วนที่เป็น Source tab ยังยอมให้ผู้ใช้สามารถแก้ไข Test Case ได้บางส่วนรวมถึงการ Copy, Cut และ Paste เป็นต้น

## 4) Log/Reference/UI-Element/Rollup Pane

เป็นส่วนที่อยู่ล่างสุดของ Selenium IDE ซึ่งถูกใช้สำหรับนำเสนอฟังก์ชันการทำงาน 4 แบบที่แตกต่างกัน ได้แก่ Log, Reference, UI-Element, Rollup Pane ตามลำดับ

Log เมื่อผู้ใช้การประมวลผล Test Case ที่ถูกบันทึกการทำงานไว้แล้ว Selenium IDE จะแสดงรายละเอียดการทำงานและข้อมูลที่แสดงถึง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความผิดพลาดของแต่ละคำสั่งที่ใช้ตามลำดับความก้าวหน้าของการทำงาน Reference โดยปกติจะเป็นค่า default ที่ถูกเลือกไว้อัตโนมัติ เมื่อมีการใช้หรือแก้ไขคำสั่งและพารามิเตอร์ใน Table Tap Reference Tap จะแสดงรายละเอียดการทำงานในรูปเอกสารของคำสั่งปัจจุบัน เมื่อมีการเรียกใช้หรือแก้ไขคำสั่งไม่ว่าจะอยู่ใน Table หรือ Source Tab

UI-Element เป็นส่วนการทำงานเพิ่มเติมที่ใช้สำหรับ Selenium IDE และ Selenium RC เพื่อให้ความเป็นไปได้ในการแปลงค่าระหว่างโดยใช้ชื่อที่สื่อความหมายได้ตามส่วนประกอบที่ถูกระบุที่อยู่ภายในเว็บเพจ การแปลงค่าดังกล่าวจะถูกจัดเก็บไว้ในไฟล์โดยใช้โดยใช้สัญชาตญาณลักษณะของ จาวาสคริปต์ ไฟล์ดังกล่าวอาจใช้ร่วมกันโดย Selenium IDE และเป็นตัวขับเคลื่อนการทำงานผ่าน Selenium เซิร์ฟเวอร์

Rollup ยอมให้ผู้ใช้สามารถประมวลผลกลุ่มของคำสั่งในขั้นตอนเดียว โดยกลุ่มของคำสั่งดังกล่าวถูกเรียกว่า “Rollup” ที่ประกอบไปด้วยการใช้จาวาสคริปต์เป็นหลัก และแนวคิดของ UI-Element ในการแปลงกลุ่มของคำสั่งที่มีลักษณะคล้ายฟังก์ชันในโปรแกรมภาษานั้นเอง Rollups สามารถนำกลับมาใช้ใหม่ได้ นั่นคือความสามารถในการใช้งานได้หลายครั้งภายในหนึ่ง Test Case

### 2.3.3 Selenium WebDriver

WebDriver ถูกพัฒนาขึ้นโดยมีวัตถุประสงค์เพื่อสร้าง API ที่สามารถใช้งานง่าย ไม่ยึดติดกับเฟรมเวิร์คอื่นๆ ที่ใช้ในการทดสอบ ผู้ใช้สามารถเลือกการสร้าง Test Case และการรัน การทดสอบโดยใช้งานร่วมกับ JUnit หรือเมธอด main() ในการควบคุมการประมวลผลเช่นเดียวกับโปรแกรมทั่วไป

WebDriver ติดต่อกับเบราว์เซอร์ในระดับของระบบปฏิบัติการผ่านไดรฟ์เวอร์ที่ ถูกออกแบบไว้สำหรับเบราว์เซอร์แต่ละชนิด สนับสนุนการทดสอบบนอุปกรณ์เคลื่อนที่ต่างๆ เช่น Iphone, Android เป็นต้น

#### 2.3.3.1 คำสั่งใน WebDriver

WebDriver ถูกออกแบบมาในรูปแบบของ Interface ที่ใช้สำหรับการทดสอบผ่านเว็บเบราว์เซอร์และเพื่อตอบสนองความต้องการพื้นฐานสำหรับการทดสอบเว็บ แอปพลิเคชันโดยปกติแล้วการทดสอบเว็บแอปพลิเคชันต้องการฟังก์ชันการทำงานที่สามารถ แบ่งได้เป็น 3 ชนิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชันควบคุมการทำงานของบราวเซอร์ ประกอบด้วย

- `get` โหลดเว็บเพจโดยผ่านค่า URL ในรูปแบบของ String ที่ต้องการในกรณีเว็บเบราว์เซอร์ไม่ได้เปิดอยู่จะทำการเปิดเว็บเบราว์เซอร์ `navigate()`; โหลดเว็บเพจโดยระบุค่า URL รวมไปถึงจากประวัติการเข้าถึงผ่านเบราว์เซอร์  
`driver.get("www.google.co.th");`
  - `close()`; ปิดหน้าต่างเบราว์เซอร์ปัจจุบัน  
`driver.close();`
  - `quite()`; ปิดทุกๆ หน้าต่างของเบราว์เซอร์ที่เกี่ยวข้องกับ `webdriver`  
`driver.quit();`
  - `switchTo()`; สลับหน้าต่างปัจจุบันไปยังหน้าต่างที่ต้องการ  
`driver.switchTo().window("WindowName");`
  - `manage()`; ยอมให้มีการเข้าถึงเมนูในเบราว์เซอร์ เช่น add cookie เป็นต้น
- ฟังก์ชันสำหรับการเลือก WebElement
- `findElement()` คืนค่า 'WebElement' ที่พบ
  - `findElements()` คืนค่าทุกๆ WebElement ที่พบภายในเว็บเพจปัจจุบัน
  - ฟังก์ชันที่ช่วยในการดีบั๊ก
  - `getCurrentUrl` คืนค่า URL ปัจจุบันที่ถูกโหลดภายในเบราว์เซอร์ session
  - `getPageSource()` คืนค่า HTML source ของเพจปัจจุบันที่ถูกโหลดภายในเบราว์เซอร์
  - `getTitle()` คืนค่าชื่อของเว็บเพจปัจจุบัน
  - `getWindowHandle()` คืนค่า String ที่ใช้ระบุตัวหน้าต่างปัจจุบัน
- ไดรฟ์เวอร์อ็อบเจกต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.3.3.2 ไดรฟ์เวอร์สำหรับบราวเซอร์ที่ใช้ใน WebDriver

Driver เป็นอ็อบเจกต์ของ WebDriver ที่ใช้สำหรับสำหรับการทำงานร่วมกับ บราวเซอร์ชนิดต่างๆ ซึ่งการใช้งานของแต่ละบราวเซอร์จะมีความแตกต่างกัน ดังนั้นในการเลือกใช้ driver สำหรับบราวเซอร์ที่แตกต่างกันย่อมมีรายละเอียดของการติดตั้งใช้งานที่แตกต่างกันตามไปด้วย สำหรับ driver ซึ่งเป็นที่นิยมในการใช้งานในปัจจุบันมี ดังต่อไปนี้

```
WebDriver driver = new HtmlUnitDriver();
```

นอกจากนั้น HTMLUnit ยังสนับสนุนการทำงานของจาวาสคริปต์ รวมไปถึงการทำงานร่วมกับ Ajax ที่มีความซับซ้อนได้เช่นกัน ในทางปฏิบัติแล้วไดรฟ์เวอร์แบบนี้สามารถใช้ได้ดีในกรณีที่ต้องการทดสอบอัตโนมัติที่มีการใช้งานร่วมกับข้อมูลทดสอบจำนวนมาก นอกจากนี้ข้อได้เปรียบในเรื่องของความเร็วแล้วยังสามารถลดการทำงานกับบราวเซอร์ที่ใช้ตามไปด้วย

### 2.3.3.3 ไดรฟ์เวอร์สำหรับ Firefox

Driver ถูกติดตั้งมาพร้อมกับ WebDriver เรียบร้อยแล้วและพร้อมสำหรับการใช้งานทันที โดยไม่ต้องมีการดาวน์โหลดไฟล์เพื่อนใช้งานเพิ่มเติมแต่อย่างใด ส่วนการเรียกใช้งานของ Driver ดังกล่าวสามารถทำได้โดยใช้คำสั่งดังต่อไปนี้

```
WebDriver driver = new FirefoxDriver();
```

### 2.3.3.4 ไดรฟ์เวอร์สำหรับ Internet Explorer

Driver แบบนี้สามารถทำงานได้เฉพาะระบบปฏิบัติการวินโดวส์ เนื่องจากไม่ได้เป็นบราวเซอร์หลักที่ใช้งานร่วมกับ Selenium โดยตรง ดังนั้นการเรียกใช้ driver สำหรับบราวเซอร์ Internet Explorer จะต้องดาวน์โหลด InternetExplorerDriver และใช้คำสั่ง System.setProperty() กำหนดรายละเอียดของบราวเซอร์ที่ใช้พร้อมทั้งระบุ path ของไฟล์ จากนั้นจึงสามารถเรียกใช้ driver ได้โดยใช้คำสั่งต่อไปนี้

```
WebDriver driver = new InternetExplorerDriver();
```

### 2.3.3.5 ไดรฟ์เวอร์สำหรับ Chrome

WebDriver สามารถทำงานร่วมกับบราวเซอร์ Chrome ผ่าน chromedriver ผู้ใช้จำเป็นต้องติดตั้งทั้ง chromedriver และ chrome บราวเซอร์ นอกจากนี้ผู้ใช้จำเป็นต้องกำหนดค่า Property ก่อนการเรียกใช้งานเช่นเดียวกับบราวเซอร์ InternetExplorerDriver การเรียกใช้ driver สำหรับบราวเซอร์สามารถทำได้ดังคำสั่งต่อไปนี้

```
System.setProperty("webdriver.chrome.driver", "C:\\ Path of  
\\chromedriver.exe");
```

```
WebDriver driver = new ChromeDriver();
```

### 2.3.3.6 WebElement ที่ใช้ร่วมกับ WebDriver

WebElement เป็น interface ที่สืบทอดมาจาก SearchContext ใช้สำหรับการ นำเสนอส่วนประกอบต่างๆ ของ HTML ที่ปรากฏอยู่ในเว็บเพจ โดยปกติแล้วการทำงานส่วนใหญ่ที่มีการโต้ตอบกับเว็บเพจจะทำงานผ่านเมธอดที่กำหนดไว้ภายใน WebElement ซึ่ง interface ดังกล่าวได้จัดเตรียมเมธอดต่างๆ ไว้พร้อมสำหรับการทำงานดังรายละเอียดต่อไปนี้

- Clear() ลบค่าปัจจุบันออกทั้งหมดในกรณีที่ Element มีการใช้งานเกี่ยวข้องกับ ข้อความ
- Click() จำลองการทำงานของเมาส์คลิกลงบน element
- getAttribute(String name) คืนค่าที่เกี่ยวข้องกับชื่อ attribute ที่ถูกระบุหรือ null (ในกรณีหาไม่พบ)
- getTagName() คืนค่า tag name ของ element ที่ถูกระบุ
- getText() คืนค่า text ที่อยู่ใน element (รวมถึง subelement) ที่ไม่ถูกซ่อนโดย css
- getValue() คืนค่า attribute จาก element
- isEnabled() คืนค่า true สำหรับ input element ที่ปัจจุบัน enable ไว้
- isSelected() คืนค่า true หาก element (radio button, select, และ checkboxes)
- sendKeys(CharSequenceKeyToSend) จำลองการกรอกข้อมูลลงใน element
- setSelected() เลือก element (radio button, select, และ checkboxes)
- submit() Submits ค่าบล็อกเดียวกันในกรณีที่ element เป็นฟอร์ม (หรือ อยู่ในฟอร์ม) Blocks จนกว่าเพจใหม่จะถูกโหลด
- toggle() toggles สถานะของ element ที่เป็น checkbox

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเข้าถึงส่วนต่างๆ บนเว็บเพจ เพื่อให้สามารถปฏิสัมพันธ์กับเว็บเพจได้ทำ  
ได้จากการเรียกใช้เมธอด `findElement()` ที่คืนค่า `WebElement` อ็อบเจกต์ อย่งไร  
ก็ตามเมธอด ดังกล่าวจะรับค่าพารามิเตอร์ที่ต้องการค้นหาผ่านคีย์เวิร์ด “By” ซึ่งมี  
รูปแบบของการค้นหาที่ สามารถแบ่งออกเป็นประเภทต่างๆ ได้ดังต่อไปนี้

- Id attribute  
`By.id(“myElementID”)`
- Name attribute  
`By.name(“myElementName”)`
- XPATH  
`By.xpath(“//input[@id=’myElementID’]”)`
- Class name  
`By.className(“even-table-row”)`
- CSS Selector  
`By.cssSelector(“h1[title]”)`
- Link Text  
`By.linkText(“Click Me!”)`, `By.partialLinkText(“ck M”)`
- Tag Name  
`By.tagName(“td”)`

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.3.4 Robot framework [6]

Robot framework เป็นเครื่องมือในการทดสอบตัวหนึ่งที่ได้รับคามนิยมในการเขียน testing และการทำ TDD (Test-Driven Development) เนื่องจากสคริปต์ที่เขียนออกมา สามารถอ่านแล้วเข้าใจได้ง่าย ถึงแม้จะไม่เคยเขียนโปรแกรมมาก่อนก็ตาม ดังนั้นจึงสามารถบอกได้ทันทีว่าสคริปต์นั้น ๆ ทำงานอย่างไร และทดสอบอะไรอยู่ ตัว core framework นั้นถูกเขียนด้วย python (รองรับทั้ง Python 2 และ Python 3) และถูกรันบน Jython (JVM) กับ IronPython เพื่อให้สามารถรองรับทั้ง Java และ .NET ได้ด้วย ดังนั้นมันจึงมีความยืดหยุ่นสูง และมีเทสสคริปต์ต่อการทดสอบขั้นพื้นฐานค่อนข้างครบถ้วน ซึ่งจะถูกแยกเป็น Library อย่างเป็นระเบียบ สามารถ import นำไปใช้ตามงานที่ต้องการทดสอบได้ ไม่ว่าจะเป็น Android, iOS, Website, Rest API, Windows GUI, Java Swing GUI, Soup Web Service, MongoDB

โครงสร้างหลัก ๆ ของ Robot framework ได้แก่

- 1) Settings
- 2) Variables
- 3) Test Cases
- 4) Keywords

โดยโครงสร้างที่สำคัญในการรันเทสเคสคือ Setting และ Test Case ซึ่งในการสร้างไฟล์สำหรับเทสด้วย Robot Framework ทุกครั้งจะต้องมี 2 ส่วนนี้มิฉะนั้นจะไม่สามารถทำงานได้ ในส่วนของ Keyword กับ Variables อาจจะมีหรือไม่มีก็ได้

#### 2.3.4.1 Settings

Settings ในส่วนนี้จะเป็นการกำหนด Library ที่เราต้องการจะหยิบ Keyword ใน Library ไหนมาใช้งาน ซึ่ง Library เปรียบเสมือน Dictionary ที่รวมเอา Keyword ทั้งหมดที่จะนำไปเขียนคำสั่ง Test แบ่งเป็น Library Standard, External และ other

STANDARD	EXTERNAL	OTHER
<p><b>Builtin</b> Provides a set of often needed generic keywords. Always automatically available without imports.</p> <p><b>Dialogs</b> Provides means for pausing the test execution and getting input from users.</p> <p><b>Collections</b> Provides a set of keywords for handling Python lists and dictionaries.</p> <p><b>Process</b> Library for running processes in the system. New in Robot Framework 2.8.</p>	<p><b>OperatingSystem</b> Enables various operating system related tasks to be performed in the system where Robot Framework is running.</p> <p><b>Remote</b> Special library acting as a proxy between Robot Framework and test libraries elsewhere. Actual test libraries can be running on different machines and be implemented using any programming language supporting XML-RPC protocol.</p> <p><b>Screenshot</b> Provides keywords to capture screenshots of the desktop.</p>	<p><b>String</b> Library for generating, modifying and verifying strings.</p> <p><b>Telnet</b> Makes it possible to connect to Telnet servers and execute commands on the opened connections.</p> <p><b>XML</b> Library for generating, modifying and verifying XML files.</p> <p><b>DateTime</b> Library for date and time conversions. New in Robot Framework 2.8.5.</p>

## รูปที่ 2.9 แสดง Standard Library

STANDARD	EXTERNAL	OTHER
<p><b>Android Library</b> Library for all your Android automation needs. It uses Calabash Android internally.</p> <p><b>AnywhereLibrary</b> Library for testing Single-Page Apps (SPA). Uses Selenium Webdriver and Appium internally.</p> <p><b>AppiumLibrary</b> Library for Android- and iOS-testing. It uses Appium internally.</p> <p><b>Archive Library</b> Library for handling zip- and tar-archives.</p> <p><b>Autotest Library</b> Windows GUI testing library that uses Autotest freeware tool as a driver.</p> <p><b>Database Library (Java)</b> Java-based library for database testing. Usable with Jython. Available also at Maven central.</p> <p><b>Database Library (Python)</b> Python based library for database testing. Works with any Python interpreter, including Jython.</p> <p><b>Diff Library</b> Library to diff two files together.</p> <p><b>Django Library</b> Library for Django, a Python web framework.</p> <p><b>Eclipse Library</b> Library for testing Eclipse RCP applications using SWT widgets.</p>	<p><b>robotframework-faker</b> Library for Faker, a fake test data generator.</p> <p><b>FTP Library</b> Library for testing and using FTP steyer with Robot Framework.</p> <p><b>HTTP library (liverest)</b> Library for HTTP level testing using liverest tool internally.</p> <p><b>HTTP Library (Requests)</b> Library for HTTP level testing using Request internally.</p> <p><b>HttpRequestLibrary (Java)</b> Library for HTTP level testing using Apache HTTP client. Available also at Maven central.</p> <p><b>iOS Library</b> Library for all your iOS automation needs. It uses Calabash iOS Server internally.</p> <p><b>ImageHorizon Library</b> Cross-platform, pure Python library for GUI automation based on image recognition.</p> <p><b>MongoDB Library</b> Library for interacting with MongoDB using pymongo.</p> <p><b>MQTT Library</b> Library for testing MQTT brokers and applications.</p> <p><b>Netconf Library</b> NETCONF protocol library based on ncclient</p> <p><b>Rammbock</b> Generic network protocol test library that offers easy way to specify network packets and inspect the results of sent and received packets.</p>	<p><b>RemoteSwingLibrary</b> Library for testing and connecting to a Java process and using SwingLibrary, especially Java Web Start applications.</p> <p><b>SeleniumLibrary</b> Web testing library that uses popular Selenium tool internally. Uses deprecated Selenium 1.0 and is also itself deprecated.</p> <p><b>Selenium2Library</b> Web testing library that uses Selenium 2. For most parts drop-in-replacement for old SeleniumLibrary.</p> <p><b>Selenium2Library for Java</b> Java port of the Selenium2Library.</p> <p><b>ExtendedSelenium2Library</b> Web testing library that uses Selenium2Library internally, providing AngularJS support on top of it.</p> <p><b>SSH Library</b> Enables executing commands on remote machines over an SSH connection. Also supports transferring files using SFTP.</p> <p><b>SudsLibrary</b> A library for functional testing of SOAP-based web services based on Suds, a dynamic SOAP 1.1 client.</p> <p><b>SwingLibrary</b> Library for testing Java applications with Swing GUI.</p> <p><b>waiter-robot</b> Web testing library that uses Watir tool.</p>

## รูปที่ 2.10 แสดง External Library

STANDARD	EXTERNAL	OTHER
<p><b>Creating test libraries</b> Creating test libraries section in Robot Framework User Guide.</p> <p><b>phone-app-robotframework</b> Provides resources and tools for writing functional Selenium tests for Phone CMS and its add-ons.</p>	<p><b>RemoteApplications</b> Special test library for launching Java applications on a separate JVM and taking other libraries into use on them.</p>	<p><b>JavaLibCore</b> Base for implementing larger Java based test libraries for Robot Framework.</p> <p><b>Remote</b> Built-in special library acting as a proxy between Robot Framework and test libraries elsewhere. Actual test libraries can be running on different machines and be implemented using any programming language supporting XML-RPC protocol.</p>

## รูปที่ 2.11 แสดง Other Library

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## รูปแบบคำสั่งการกำหนดค่า Library

Library <<Library Name>>

```

1 # Structure
2 *** Settings ***
3 Library Selenium2Library
4 *** Keywords ***
5
6 *** Variables ***
7
8 *** Test Cases ***

```

รูปที่ 2.12 แสดงรูปแบบคำสั่งการกำหนดค่า Library

### 2.3.4.2 Keywords

Keyword ในส่วนนี้จะใช้ในกรณีที่ต้องการที่จะสร้าง Keyword ของขึ้นมา ซึ่งจะสามารถใช้ได้เฉพาะในไฟล์ปัจจุบันเท่านั้น โดยใน Keyword อาจจะสร้างชุดคำสั่งย่อย ๆ ขึ้นมาอีก

รูปแบบคำสั่งการสร้าง Keyword

<<Keyword>>

[Argument] <<ตัวแปร>>

```

1 # Structure
2 *** Settings ***
3 Library Selenium2Library
4 *** Keywords ***
5 Searching
6     [Arguments]    ${Keyword-Search}
7     Input Text    1st-ib    ${Keyword-Search}
8     Press Key    1st-ib    \\13
9
10 *** Variables ***
11
12 *** Test Cases ***

```

รูปที่ 2.13 แสดงรูปแบบคำสั่งการสร้าง Keyword

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.3.4.3 Variables

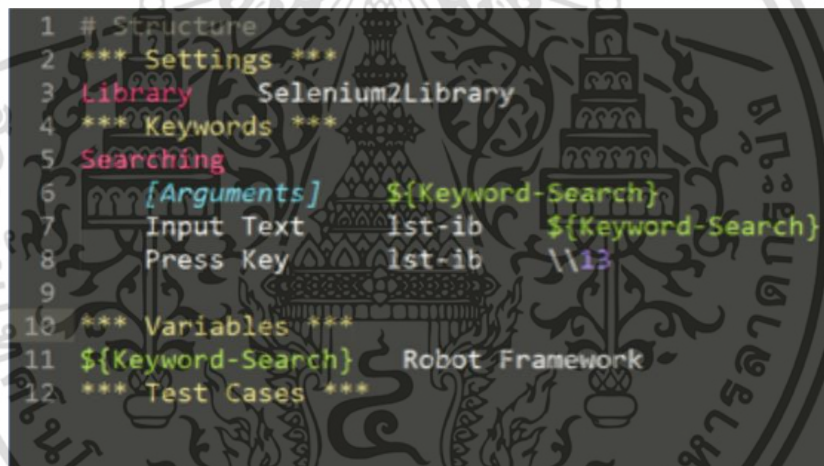
Variables ในส่วนนี้จะเป็นการสร้างหรือประกาศตัวแปรขึ้นมาสำหรับใช้งานสำหรับ Test file เพื่อนำไปใช้ใน Test suite

ชนิดของตัวแปร

- 1) Scalar ใช้งานในรูปแบบ `${Variable name}` สามารถกำหนดค่าได้เพียงค่าเดียวเท่านั้น
- 2) List ใช้งานในรูปแบบ `@{Variable name}` สามารถกำหนดค่าได้มากกว่า 1 ค่าในตัวแปรเดียวกัน

รูปแบบคำสั่งการสร้างตัวแปร

`${<<Variables>>}` `<<Value>>`



```

1 # Structure
2 *** Settings ***
3 Library Selenium2Library
4 *** Keywords ***
5 Searching
6 [Arguments]   ${Keyword-Search}
7 Input Text   lst-ib   ${Keyword-Search}
8 Press Key    lst-ib   \\13
9
10 *** Variables ***
11 ${Keyword-Search}   Robot Framework
12 *** Test Cases ***

```

รูปที่ 2.14 แสดงรูปแบบคำสั่งการสร้างตัวแปร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 2.3.4.4 Test Case

Test Cases คือส่วนของการเขียน Test Case และการนำ Keyword เขียนเป็นชุดคำสั่งในส่วนนี้ ซึ่งสิ่งสำคัญในส่วนนี้คือ ชื่อ Test Cases เพราะไม่มีการตั้งชื่อให้กับ Test Cases ก็จะไม่สามารถ Run คำสั่งทั้งหมดในไฟล์นี้ได้

รูปแบบคำสั่งการสร้าง Test Case

<<ชื่อ Test Case >>

```

1 # Structure
2 *** Settings ***
3 Library Selenium2Library
4 *** Keywords ***
5 Searching
6 [Arguments] ${Keyword-Search}
7 Input Text lst-id ${Keyword-Search}
8 Press Key lst-id \13
9
10 *** Variables ***
11 ${Keyword-Search} Robot Framework
12
13 *** Test Cases ***
14 Test Robot Framework
15 Open Browser https://www.google.com gc
16 Searching Robot Framework
  
```

รูปที่ 2.15 แสดงรูปแบบคำสั่งการสร้าง Test Case

#### 2.4 ATDD [7]

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.4.1 ลักษณะเด่นๆ ของ ATDD [8]

- Common understanding เป็นการสร้างความเข้าใจที่ตรงกันให้กับทีมและลูกค้า โดยการใช้ตัวอย่างในการแสดง requirements
- Ubiquitous language ใช้ภาษาที่ให้ความหมายเดียวกันในการสื่อถึงสิ่งเดียวกัน ภาษาที่ใช้ในการเขียน test cases และ requirements เป็นภาษาเดียวกัน ทำให้ง่ายต่อการตีความและการทำความเข้าใจ
- Executable requirements/examples Living documentation เป็นการสร้าง requirements ที่ไม่ได้เป็นเพียงกระดาษแต่สามารถประมวลผลโดยคอมพิวเตอร์ได้

### 2.4.2 ATDD cycle

ATDD มีการทำงานเป็นแบบวนซ้ำ ซึ่งประกอบด้วย 3 ขั้นตอน

- Discuss
- Develop
- Deliver

#### 2.4.2.1 Discuss

เป็นขั้นตอนของการถกเถียงถึงปัญหาหรือ Feature ที่นำมาจาก Product backlog เพื่อหา Solution โดยทีมทั้งหมดจะร่วมกันหาวิธีการแก้ปัญหาที่ดีที่สุด ขั้นตอนนี้ช่วยเพิ่มความเข้าใจและคำศัพท์ที่ใช้ร่วมกันเพื่อสื่อความหมายเดียวกัน รวมถึงมีการกำหนดตัวอย่างที่ชัดเจนสำหรับ Solution เพื่อใช้ในการสร้างเป็น test cases ซึ่งสามารถประมวลผลโดยคอมพิวเตอร์ได้ต่อไป

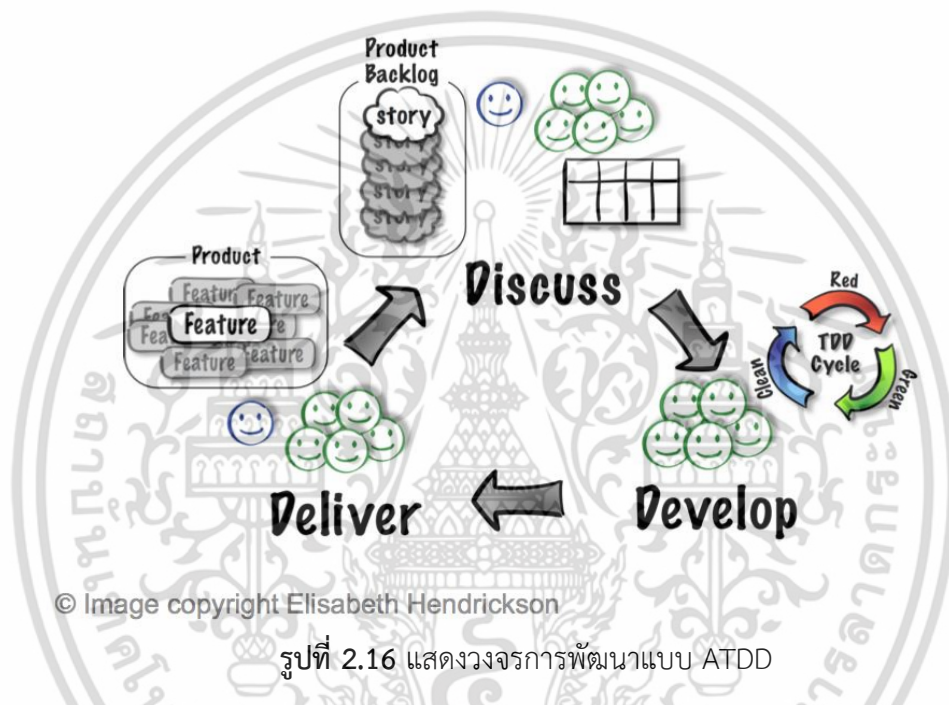
#### 2.4.2.1 Develop

ตัวอย่างที่กำหนดไว้ในขั้นตอน Discuss จะเป็นตัวช่วยและแนะแนวทางในการพัฒนา Feature โดยในระหว่างพัฒนา Feature ก็สามารถสร้าง automated test cases สำหรับ Feature นั้นในเวลาเดียวกัน ซึ่งทั้งทีมต้องร่วมกันรับผิดชอบการ automation ร่วมกัน และในตอนท้ายสุดเมื่อทุก test cases ผ่าน เป็นอันเสร็จขั้นตอน

### 2.4.2.1 Deliver

ขั้นตอนการส่งมอบงานคือการแสดงการทำงานของ Feature นั้นให้กับทุกๆ Stakeholders และทำการ execute test cases กับ Feature ที่เสร็จแล้วเป็นที่ยืนยันว่า Feature นี้ทำงานตรงตามความต้องการของลูกค้าดังที่ได้ตกลงกันในช่วง Discuss หากได้รับคำแนะนำหรือข้อคิดเห็นก็นำไปพิจารณาในช่วง Discuss ครั้งต่อไป

## The ATDD cycle

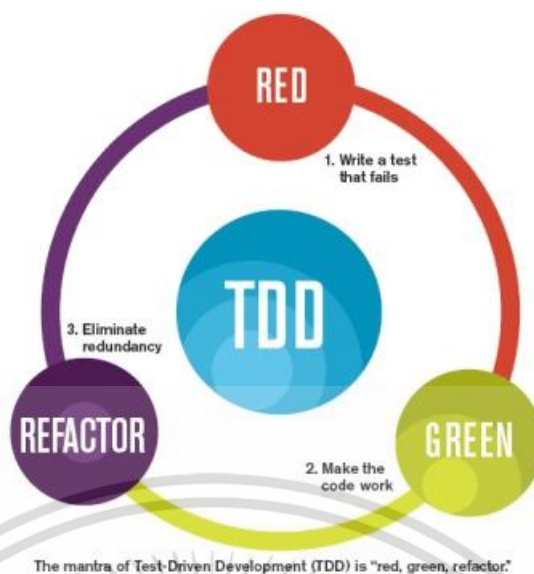


รูปที่ 2.16 แสดงวงจรการพัฒนาแบบ ATDD

## 2.5 TDD (Test Driven Development) [9]

Test Driven Development หรือ TDD คือกระบวนการพัฒนาซอฟต์แวร์รูปแบบหนึ่ง ซึ่งใช้หลักการคือ มีวงจรการพัฒนาที่ทำซ้ำไปเรื่อยๆ โดยวงจรการพัฒนาซอฟต์แวร์แบบ TDD นั้นจะเป็นดังรูปที่ 2.17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.17 วงจรการพัฒนาแบบ TDD

โดยวงจรการพัฒนาซอฟต์แวร์แบบ TDD นั้นจะเริ่มจากการที่ developer ทำการเขียน test case นี้เป็น unit test ที่อธิบายฟังก์ชันใหม่ หรือ ความต้องการของระบบ หลังจากเขียน test case ขึ้นแล้วให้ทำการรัน test case ซึ่งเมื่อทำการรัน test case ขึ้นก็จะพบว่าผลลัพธ์คือ test fail เนื่องจากโปรแกรมที่พัฒนายังไม่มีส่วน implementation หรือยังไม่ได้พัฒนาขึ้นมาหลังจากนั้น ขั้นตอนต่อมาคือพัฒนาโปรแกรม หรือเขียนซอร์สโค้ดหลักของระบบขึ้นมา โดยมีจุดมุ่งหมายคือเขียนโปรแกรมโดยเน้นให้ผลลัพธ์คือการทำให้ test case ที่เขียนขึ้นมาในขั้นตอนแรก รันผ่าน ซึ่งซอร์สโค้ดในขั้นตอนที่สองนี้ควรพยายามเขียนซอร์สโค้ดให้น้อยที่สุดเท่าที่ทำได้ โดยมีจุดประสงค์เพียงเพื่อให้ test case รันผ่านเท่านั้น หลักทำการรัน test case ทั้งหมดผ่านแล้ว ขั้นตอนสุดท้ายในวงจรการพัฒนาแบบ TDD คือการ refactor หรือการปรับปรุง จัดแต่งซอร์สโค้ดที่เขียนขึ้นให้สวยงาม อ่านง่าย ทำการ extract method หรือตั้งชื่อตัวแปรตาม standard เมื่อทำการ refactor เสร็จแล้วจะถือว่าจบ วงจรการพัฒนา 1 รอบหลังจากนั้นก็ทำการวนรอบการพัฒนาไปเรื่อยๆ โดยจะต้องเริ่มต้นจากการเขียน test case สำหรับฟังก์ชันใหม่ที่ต้องการสำหรับระบบและทำการพัฒนาตามขั้นตอนต่อไป จนโปรแกรมเสร็จสมบูรณ์สมบูรณ์ โดยการนำ Test Driven Development ไปประยุกต์ใช้พัฒนาโปรแกรมนั้นไม่มีข้อจำกัดทางด้านภาษาโปรแกรมที่ใช้ในการพัฒนา เนื่องจาก TDD นั้นเป็นแนวคิดที่ไม่ยึดติดกับภาษาโปรแกรมและในส่วนของการเขียน test case ซึ่งเป็น unit test นั้น ได้มีผู้พัฒนา framework ที่ใช้ช่วยเขียน unit test สำหรับภาษาต่างๆ ไว้มากมาย โดย framework ที่เป็นที่ยอมรับใช้ในการพัฒนา unit test นั้น ยกตัวอย่างเช่น Junit สำหรับ java PHPUnit สำหรับภาษา PHP และ COBOLUnit สำหรับภาษา COBOL เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.6 การค้นหาส่วนประกอบของ HTML [10]

สิ่งสำคัญสำหรับการทดสอบอัตโนมัติร่วมกับเว็บแอปพลิเคชันภายใต้การทดสอบคือ วิธีการค้นหาส่วนประกอบต่าง ๆ ของ HTML ที่ต้องการทดสอบ เช่น Link, Buttons, Edit boxes, Drop downs, และอื่น ๆ Selenium ใช้แนวคิดที่เรียกว่า “Locator” สำหรับการค้นหาและจำแนกความแตกต่างระหว่างส่วนประกอบเหล่านั้น ซึ่งเป็นการค้นหาที่ถูกต้องถือเป็นขั้นตอนสำคัญในการสร้างสคริปต์เพื่อใช้สำหรับการทดสอบอัตโนมัติ ในทางทฤษฎีส่วนประกอบต่าง ๆ ของ HTML มักจะใช้ชื่อหรือรหัสที่แตกต่างกัน ทั้งนี้เพื่อให้ผู้ใช้สามารถนำไปใช้ในการอ้างอิงได้อย่างถูกต้อง แต่อย่างไรก็ตาม การทดสอบอัตโนมัติด้วย Selenium จะใช้เทคนิคการทดสอบแบบ Black Box ซึ่งปกปิดรายละเอียดการทำงานภายในไว้ในรูปกล่องดำ ซึ่งผู้ใช้ไม่สามารถเข้าถึงโครงสร้างภายในโค้ดได้ ดังนั้นในทางปฏิบัติการค้นหาส่วนประกอบของ HTML จากชื่อหรือรหัสที่แตกต่างกันอาจไม่สามารถทำได้อย่างถูกต้องทั้งหมด นอกจากนี้ในบางเว็บยังอาจใช้เครื่องมืออัตโนมัติในการสร้างรหัสที่แตกต่างกันทุกครั้งที่เว็บแอปพลิเคชันทำงาน ซึ่งสิ่งเหล่านี้ส่งผลให้วิธีการค้นหาส่วนประกอบต่างๆ ของเว็บเพจที่มีความถูกต้อง 100 เปอร์เซ็นต์เป็นไปได้ยาก

ดังที่ได้กล่าวมาแล้วว่า Selenese เป็นกลุ่มของคำสั่งที่ใช้งานร่วมกับ Selenium ในการทดสอบซึ่งคำสั่งเหล่านี้โดยปกติจะแบ่งออกเป็นสามส่วน ส่วนแรกได้แก่ command เป็นคำสั่งที่ใช้ทำงานทั่วไป เช่น click, type เป็นต้น ส่วนที่สองท้ายสุด Value เป็นค่าที่ต้องการใช้งานในรูปแบบของพารามิเตอร์ ส่วนที่เป็น Target เป็นส่วนที่สำคัญที่สุดที่ใช้สำหรับระบุวิธีการค้นหา ซึ่งหมายถึงการระบุตัวตนของส่วนประกอบต่างๆ ของ HTML ซึ่งการค้นหาดังกล่าวสามารถทำได้หลายวิธี โดยแบ่งออกเป็นประเภทต่างๆ ได้ดังนี้

### 2.6.1 วิธีการค้นหาด้วย ID

เป็นวิธีการค้นหาส่วนประกอบต่างๆ ของ HTML ที่มีค่า ID สอดคล้องกับรูปแบบที่ระบุไว้ ดังนี้

รูปแบบ : id = id of the element

วิธีการค้นหาด้วย ID มีข้อได้เปรียบตรงที่ค่า ID มีโอกาสซ้ำกันได้น้อยมาก ดังนั้นจึงถือเป็นวิธีการที่มีความถูกต้องสูง

ตัวอย่างการทำงานของโค้ด HTML

```
<html>
<body>
<form id = "loginForm">
  <input name= "username" type= "text" />
  <input name= "password" type= "password" />
  <input name= "continue" type= "submit" value= "login" />
  <input name= "continue" type= "button" value= "Clear" />
</form >
</body>
</html>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.6.2 วิธีการค้นหาด้วย Name

เป็นวิธีการค้นหาส่วนประกอบต่างๆ ของ HTML โดยใช้ name

รูปแบบ : name = name of the element

ตัวอย่างการทำงานของโค้ด HTML

```
<html>
<body>
<form id = "loginForm">
  <input name= "username" type= "text" />
  <input name= "password" type= "password" />
  <input name= "continue" type= "submit" value= "login" />
  <input name= "continue" type= "button" value= "Clear" />
</form >
</body>
</html>
```

ในกรณีการค้นหาจาก name อย่างเดียวไม่สามารถระบุตัวตนได้อย่างถูกต้อง อาจกำหนดค่าอื่นๆ เพิ่มเติมเพื่อช่วยในการระบุตัวตนได้อย่างถูกต้องยิ่งขึ้น

## 2.6.3 วิธีการค้นหาด้วย Link Text

เป็นวิธีการค้นหาส่วนประกอบต่างๆ ของ HTML ได้จาก link โดยขึ้นอยู่กับข้อความที่ต้องการเป็นหลัก

รูปแบบ : link = link\_text

ตัวอย่างการทำงานของโค้ด HTML ดังต่อไปนี้

```
<html>

<body>

  <p>Are you sure you want to do this ?</p>

  <a href="continue.html">Continue</a>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<a href="cancel.html">Cancel</a>
```

```
</body>
```

```
</html>
```

```
link=Continue (4)
```

```
link=Cancel (5)
```

## 2.6.4 วิธีการค้นหาด้วย CSS

CSS (Cascading Style Sheets) เป็นภาษาสำหรับการแสดงรายละเอียดการถอดความของ HTML และเอกสาร XML CSS ใช้ Selector สำหรับการกำหนดค่าคุณสมบัติไปยังส่วนประกอบต่างๆ ของ HTML ที่อยู่ภายในเอกสารที่ถูกระบุ มีหลายรูปแบบดังต่อไปนี้

- Element Selector
- ID Selectors
- Class Selectors

ตัวอย่างการทำงานของโค้ด HTML ดังต่อไปนี้

```
<html>
<head>
  <title>CSS Selectors</title>
</head>
<body>
  <h3 id= "moduleHeader">Page Module Header</h3>
  <div class= "primaryContainerModule">
    <h4 id= "pageSub">Subtitle</h4>
  </div>
  <form id= "pageForm" method= "post"
  action="process.php">
    <input type="text" name="username" value="">
    <input type="password" name="password" value="">
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตเห็นใบเซปวีระโยชนทานการค่า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<input type="submit" name="submitButton"
value="Submit Form"/>
```

```
</body>
```

```
</html>
```

#### 2.6.4.1 Element Selector

เป็นการใช้ name เพื่อค้นหาส่วนประกอบต่างๆ ของ HTML ที่ต้องการ จากตัวอย่างโค้ดด้านบนสามารถค้นหา subtitle ได้โดยใช้ CSS selector ดังต่อไปนี้

```
css=h4
```

ID Selectors ที่ประกอบด้วยเครื่องหมาย #

ID ถูกนำมาใช้ร่วมกับเครื่องหมาย # เพื่อค้นหาส่วนประกอบต่าง ๆ ของ HTML ที่ต้องการ จากตัวอย่างโค้ดด้านบนสามารถค้นหา subtitle ได้โดยใช้ CSS selector ดังต่อไปนี้

```
css=#pageSub
```

Class Selector ที่ประกอบด้วยเครื่องหมายจุด (.)

Class selector ถูกนำมาใช้ร่วมกับเครื่องหมายจุด (.) สำหรับเพื่อค้นหาส่วนประกอบต่าง ๆ ของ HTML ที่ต้องการ จากตัวอย่างโค้ดด้านบนสามารถค้นหา Primary Container ได้โดยใช้ CSS selector ดังต่อไปนี้

```
Css=.primaryContainerModule
```

## 2.6.5 วิธีการค้นหาด้วย XPath

XPath มีลักษณะเป็นภาษาที่ถูกใช้ในกรณีที่ต้องการหาตำแหน่งของโหนดภายในเอกสาร XML เนื่องจาก HTML ที่ได้ถูกพิจารณาว่ามีการพัฒนาเช่นเดียวกับ XML ดังนั้นจึงสามารถใช้คุณสมบัติของ XPath ในการค้นหาส่วนประกอบของ HTML ได้เช่นเดียวกัน

รูปแบบ : XPath = //html-tag[@att-name='att-value']

ตารางที่ 2.6 แสดงสัญลักษณ์และรายละเอียดการทำงานของ XPath

นิพจน์	รายละเอียดการทำงาน
/(slash)	เลือกส่วนประกอบต่างๆ ที่มีความสัมพันธ์กับ root โหนด
//(double slash)	เลือกโหนดภายในเอกสารจากโหนดปัจจุบันที่ตรงกับโหนดที่ถูกเลือกไว้จากตำแหน่งที่ถูกระบุ
.(dot)	ใช้สำหรับการนำเสนอโหนดปัจจุบัน
..	เสนอ parent ของโหนดปัจจุบัน
@	นำเสนอค่า attribute
Nodename	เลือกโหนดทั้งหมดที่มีชื่อ "nodename"

ตัวอย่างการทำงานของโค้ด HTML ดังต่อไปนี้

<bib>

<book> <publisher> Addison-Wesley </pubisher>

<author> Serge Abitebiul </author>

<author><first-name> Risk </first-name>

<first-name> Hull </first-name>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
</author>
```

```
<author>Victor Vianu</author>
```

```
<title> Foundation of Databases </title>
```

```
<year>1995</year>
```

```
</book>
```

```
<book price="55">
```

```
<publisher>Freeman</publisher>
```

```
<author>Jeffrey D. Ullman </author>
```

```
<title>Principles of Database and Knowledge Base Systems </title>
```

```
<year>1998</year>
```

```
</book>
```

```
</bib>
```

ตารางที่ 2.7 แสดงตัวอย่างการใช้สัญลักษณ์และรายละเอียดการทำงานของ XPath

นิพจน์	รายละเอียดการทำงาน
/bib/book/year	<year>1995</year> หรือ <year>1998</year>
/bib/paper/year	Empty (there were no paper)
//author	<pre>&lt;author&gt; Serge Abitebiul &lt;/author&gt;</pre> <pre>&lt;author&gt;&lt;first-name&gt; Risk &lt;/first-name&gt;</pre> <pre>&lt;first-name&gt; Hull &lt;/first-name&gt;</pre>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	<pre> &lt;/author&gt;  &lt;author&gt;Victor Vianu&lt;/author&gt;  &lt;author&gt;Jeffrey D. Ulman &lt;/author&gt; </pre>
<code>/bib//first-name</code>	<code>&lt;first-name&gt; Risk &lt;/first-name&gt;</code>
<code>//author/*</code>	<pre> &lt;first-name&gt; Risk &lt;/first-name&gt;  &lt;first-name&gt; Hull &lt;/first-name&gt; </pre>
<code>//bib/book/@price</code>	55 //@price mean that price is has to be an attribute
<code>/bib/book/author[firstname]</code>	<pre> &lt;author&gt;&lt;first-name&gt; Risk &lt;/first-name&gt;  &lt;first-name&gt; Hull &lt;/first-name&gt;  &lt;/author&gt; </pre>

## 2.7 ภาษาจาวา (Java programming language) [11]

ภาษาจาวา (Java programming language) เป็นภาษาโปรแกรมเชิงวัตถุ (Object Oriented Programming) พัฒนาโดย เจมส์ กอสลิง และวิศวกรคนอื่น ๆ ที่ ซัน ไมโครซิสเต็มส์ ภาษาจาวาถูกพัฒนาขึ้นในปี พ.ศ. 2534 (ค.ศ. 1991) โดยเป็นส่วนหนึ่งของ โครงการกรีน (the Green Project) และสำเร็จออกสู่สาธารณะในปี พ.ศ. 2538 (ค.ศ. 1995) ซึ่งภาษานี้มีจุดประสงค์เพื่อใช้แทนภาษาซีพลัสพลัส (C++) โดยรูปแบบที่เพิ่มเติมขึ้นคล้ายกับภาษาออบเจกต์ทีฟซี (Objective-C) แต่เดิมภาษานี้เรียกว่า ภาษาโอ๊ก (Oak) ซึ่งตั้งชื่อตามต้นโอ๊กใกล้ที่ทำงานของ เจมส์ กอสลิง แต่ว่ามีปัญหาทางลิขสิทธิ์ จึงเปลี่ยนไปใช้ชื่อ "จาวา" ซึ่งเป็นชื่อกาแฟแทน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.7.1 จาวาแพลตฟอร์ม

- 1) Java 2 Platform, Standard Edition (J2SE) ใช้สำหรับเขียนโปรแกรมทั่วไปหรือที่เรียกว่า “จาวาแอปพลิเคชัน” (Java application) และโปรแกรมที่ทำงานบนเว็บเบราว์เซอร์หรือที่เรียกว่า “แอปแพลตฟอร์ม” (Java applet)
- 2) Java 2 Platform, Enterprise Edition (J2EE) ใช้สำหรับเขียนโปรแกรมแบบมัลติเทียร์ (multitiered) แพลตฟอร์มที่มีรากฐานมาจาก J2SE และเพิ่มคุณสมบัติสำหรับพัฒนาโปรแกรมในระดับองค์กร
- 3) Java 2 Platform, Micro Edition (J2ME) เป็นแพลตฟอร์มที่ทำงานบนสินค้าอิเล็กทรอนิกส์ เช่น โทรศัพท์มือถือรุ่นเก่า, พีดีเอ (personal digital assistant) และกล่องเคเบิลทีวี (TV set-top box)

### 2.7.2 ข้อดีของภาษาจาวา

- 1) ทำงานบนเว็บเบราว์เซอร์ได้ โปรแกรมจาวาที่มีคุณสมบัติดังกล่าวคือจาวาแอปพลิเคชันหรือเรียกย่อๆ ว่าแอปพลิเคชัน ซึ่งแปลว่าแอปพลิเคชันขนาดเล็กแอปพลิเคชันเป็นโปรแกรมที่ทำงานอยู่บน เว็บเพจ ทำให้สามารถทำงานบนระบบปฏิบัติการใดก็ได้ที่มีได้ใน เว็บเบราว์เซอร์ ที่สนับสนุนจาวาติดตั้งอยู่
- 2) ความปลอดภัยสูง ผู้ใช้สามารถมั่นใจได้ว่าแอปพลิเคชันที่โหลดจากเว็บไซต์ต่างๆ นั้นไม่มาพร้อมกับไวรัสที่เข้าไปสร้างความเสียหาย หรือแอบดูข้อมูลในเครื่อง
- 3) สนับสนุนงานหลายระดับ ตั้งแต่ระดับเซิร์ฟเวอร์ คอมพิวเตอร์ส่วนบุคคล ไปจนถึงอุปกรณ์พกพา
- 4) สามารถทำงานบนเครื่องคอมพิวเตอร์ต่างระบบได้ ขอแค่เครื่องคอมพิวเตอร์นั้นมีเครื่องจักรเหมือนจาวาติดตั้งอยู่
- 5) ภาษาจาวาเป็นภาษาเชิงวัตถุ ภาษาเชิงวัตถุช่วยให้แก้ปัญหาได้ง่ายขึ้น ทั้งนี้เพราะการมองสิ่งต่างๆ เป็นวัตถุนั้นเป็นมุมมองที่พวกเราคุ้นเคยอยู่แล้ว
- 6) ความเรียบง่าย ไวยากรณ์ของภาษาจาวาถูกออกแบบเป็นอย่างดี ทำให้เข้าใจง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 7) ระบบจัดการคืนพื้นที่ในหน่วยความจำอัตโนมัติ (automatic garbage collection) ช่วยให้โปรแกรมเมอร์ไม่ต้องกังวลว่าจะลืมคืนพื้นที่ในหน่วยความจำที่ไม่ได้ใช้
- 8) มีคลาสจำนวนมากมาให้ใช้ ทำให้โปรแกรมเมอร์สามารถพัฒนาโปรแกรมได้อย่างรวดเร็วโดยการต่อยอดจากสิ่งที่มีอยู่แล้ว
- 9) ฟรี เครื่องมือพัฒนาแจกฟรี ตั้งแต่ตัวแปลภาษา, โปรแกรมเครื่องจักรเสมือน, โปรแกรมที่ใช้พัฒนา หรือที่เรียกว่า Integrated Development Environment- IDE, โปรแกรมวาด UML class diagram, ฯลฯ

## 2.8 ภาษาไพทอน (Python) [12]

ภาษาไพทอน (Python programming language) เป็นภาษาโปรแกรมระดับสูง เพื่อใช้งานทั่วไปแบบอินเทอร์พรีเตอร์ ที่สร้างโดย กิโด ฟาน รอสซัม (Guido van Rossum)

Python คือชื่อภาษาที่ใช้ในการเขียนโปรแกรมภาษาหนึ่ง ซึ่งถูกพัฒนาขึ้นมาโดยไม่ยึดติดกับแพลตฟอร์ม กล่าวคือสามารถรันภาษา Python ได้ทั้งบนระบบ Unix, Linux , Windows NT, Windows 2000, Windows XP หรือแม้แต่ระบบ FreeBSD

### 2.8.1 ไพทอนแพลตฟอร์ม

- 1) ซีไพทอน (CPython) คือแพลตฟอร์มภาษาไพทอนดั้งเดิม โปรแกรมอินเทอร์พรีเตอร์ถูกเขียนโดยภาษาซี ซึ่งคอมไพล์ใช้ได้บนหลายระบบปฏิบัติการ เช่น วินโดวส์, ยูนิกซ์, ลินุกซ์ การใช้งานสามารถทำได้โดยการติดตั้งโปรแกรมอินเทอร์พรีเตอร์และแพ็คเกจที่จำเป็นต่างๆ
- 2) ไจทอน (Jython) ซึ่งเป็นแพลตฟอร์มภาษาไพทอนที่ถูกพัฒนาบนแพลตฟอร์มจาวา เพื่อเพิ่มอำนวยความสะดวกในการใช้ความสามารถภาษาสคริปต์ของไพทอนลงในซอฟต์แวร์จาวาอื่นๆ การใช้งานสามารถทำได้โดยการติดตั้งจาวาและเรียกไลบรารีของ ไจทอน ซึ่งมาในรูปแบบไบนารีเพื่อใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 3) ไพทอนดอตเน็ต (Python.NET) เป็นการพัฒนาภาษาไพทอนให้สามารถทำงานบนดอตเน็ตเฟรมเวิร์กของไมโครซอฟท์ได้โดยโปรแกรมที่ถูกเขียนจะถูกแปลงเป็น CLR ปัจจุบัน มีโครงการที่นำภาษาไพทอนมาใช้บน .NET Framework ของไมโครซอฟท์แล้วคือโครงการ IronPython

### 2.8.2 ข้อดีของภาษาไพทอน

- 1) สนับสนุนแนวแบบคิ OOP (Object Oriented Programming)
- 2) เป็น Open Source ทำให้ทุกคนสามารถที่จะนำ Python มาพัฒนาโปรแกรมของเราได้โดยไม่ต้องเสียค่าใช้จ่าย ทำให้มีคนเข้ามาช่วยกันพัฒนาให้ Python มีความสามารถสูงขึ้น และใช้งานได้ครบครันกับทุกลักษณะงาน
- 3) สนับสนุนเทคโนโลยี COM ของ Ms-windows
- 4) Python รวมมาตรฐานการอินเทอร์เฟซ Tkinter ซึ่งสนับสนุนบนระบบ X windows, Ms-windows และ Macintosh การใช้คำสั่ง Tkinter API ช่วยให้โปรแกรมเมอร์ไม่ต้องแก้ไขโค้ดเมื่อนำไปรันบนระบบปฏิบัติการอื่นๆ
- 5) เป็น Dynamic typing คือ สามารถเปลี่ยนชนิดข้อมูลได้ง่ายและสะดวก
- 6) มี Built-in Object Types คือ โครงสร้างของข้อมูลที่สามารถใช้ได้ ใน Python ประกอบด้วย ลิสต์, ดิกชันนารี, สตริง ที่ง่ายต่อการใช้งานและมีประสิทธิภาพสูง
- 7) มีเครื่องมือต่างๆ มากมาย เช่น การประมวลผลเท็กซ์ไฟล์ การเรียงข้อมูล การเชื่อมต่อสตริง การตรวจสอบเงื่อนไขของข้อความ การแทนค่า เป็นต้น
- 8) มีมอดูลที่สร้างขึ้นจากนักพัฒนาสนับสนุนมากมาย ได้แก่ COM, Image, CORBA, ORBs, XML เป็นต้น
- 9) จัดการหน่วยความจำอย่างอัตโนมัติ สามารถจัดการพื้นที่หน่วยความจำที่ไม่ต่อเนื่องให้ทำงานได้อย่างมีประสิทธิภาพ
- 10) อนุญาตให้ฝังชุดคำสั่งของ Python เอาไว้ภายในโค้ดภาษา C/C++ ได้
- 11) อนุญาตให้โปรแกรมเมอร์สร้าง Dynamic Link Library (DLL) เพื่อใช้ร่วมกับ Python
- 12) มีมอดูลสนับสนุนเกี่ยวกับเน็ตเวิร์ก โปรเซส เรกูลาร์ เอ็กสเพรสชัน, xml, GUI และอื่นๆ
- 13) มีฟังก์ชันสนับสนุนฐานข้อมูล เช่น MySQL, Sybase, Oracle, Informix, ODBC และอื่นๆ
- 14) มีไลบรารีสนับสนุนด้านการสร้างภาพกราฟฟิก เช่น ทำภาพเบลอ หรือภาพชัด หรือเขียนข้อความบนภาพ ตลอดจนบันทึกลงไฟล์ในรูปแบบต่างๆ ได้อย่างมีประสิทธิภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.9 XML (Extensible Markup Language) [14]

XML ย่อมาจาก Extensible Markup Language คือภาษาหนึ่งที่ใช้ในการแสดงผลข้อมูล ซึ่งภาษาที่ใช้กำหนดรูปแบบของคำสั่งภาษา HTML หรือที่เรียกว่า Meta Data ซึ่งจะใช้สำหรับกำหนดรูปแบบของคำสั่ง Markup ต่าง ๆ ถ้าเปรียบเทียบกับภาษา HTML จะแตกต่างกันที่ HTML ถูกออกแบบมาเพื่อการแสดงผลอย่างเดียวนั้น เช่น ให้แสดงผลตัวเล็ก ตัวหนา ตัวเอียง เหมือนที่คุณเคยเห็นในเว็บเพจทั่วไป แต่ภาษา XML นั้นถูกออกแบบมาเพื่อเก็บข้อมูล โดยทั้งข้อมูลและโครงสร้างของข้อมูลนั้นๆไว้ด้วยกันส่วนการแสดงผลก็จะใช้ภาษาเฉพาะซึ่งก็คือ XSL (Extensible Stylesheet Language) ภาษา XML มีโครงสร้างที่ประกอบด้วยแท็กเปิด และแท็กปิด เช่นเดียวกับภาษา HTML แต่ภาษา XML คุณสามารถสร้างแท็กรวมทั้งกำหนดโครงสร้างของข้อมูลได้เอง ซึ่งความสามารถตรงนี้ตัวภาษา ทำไม่ได้เพราะภาษา HTML ถูกกำหนดแท็กตายตัวโดย W3C หรือ World Wide Web Consortium อาจกล่าวได้ว่า XML เป็นส่วนเสริมของ HTML เพราะตัว XML ไม่สามารถแสดงผลได้ในตัวของมันเอง หากต้องการแสดงผลที่ถูกต้อง จะต้องมีการใช้ร่วมกับภาษาอื่น เช่น HTML, JSP, PHP, ASP หรือภาษาอื่น ๆ ที่สนับสนุน XML จะมีนามสกุลเป็น .XML สามารถสร้างขึ้นจากโปรแกรมประเภท Text Editor ใดก็ได้ เช่น Notepad, Editplus, DreamWeaver, MS Word เป็นต้น สิ่งที่ต้องได้ว่าเป็นเสน่ห์ของ XML นั้นจะเป็นความสะดวกในการจัดการด้านระบบการติดต่อกับผู้ใช้จากโครงสร้างของข้อมูล เราสามารถนำข้อมูลจากหลายแหล่งมาแสดงผลและประมวลผลร่วมกันได้ ไม่ว่าจะเป็นข้อมูลลูกค้า รายการสั่งซื้อ ผลการวิจัย รายการรับชำระเงินข้อมูลเวชระเบียน รายการสินค้าหรือข้อมูลสารสนเทศอื่นๆ ก็สามารถแปลงให้เป็น XML ได้ และในส่วนของข้อมูลสามารถปรับให้เป็น HTML ได้ สำหรับประโยชน์ในการใช้งานนั้น เราจะสามารถนำมาใช้สำหรับการเข้าถึงระบบข้อมูลขนาดใหญ่ใช้กับระบบเครือข่ายในองค์กร หรืออินเทอร์เน็ตเพื่อดูข้อมูลหรือเรียกใช้ข้อมูลที่ให้การแสดงผลทางหน้าจอที่รวดเร็ว

### 2.9.1 Element

ประกอบไปด้วย แท็กเปิด ข้อมูล และแท็กปิด ยกตัวอย่างเช่น

```
<student> Example_sudent </student>
```

ในที่นี้ <student> คือแท็กเปิด Example\_sudent คือข้อมูล และ </student> คือแท็กปิด โดยแท็กปิดนั้นจะต้องมีชื่อเหมือนแท็กเปิดของมันแต่ตามหลังจากเครื่องหมาย '/' จะสังเกตได้ว่า XML นั้นคล้ายกับ HTML เป็นอย่างมากสำหรับข้อแตกต่างที่ชัดเจนคือ HTML ได้กำหนดแท็กไว้ล่วงหน้าแล้วแต่ XML ไม่ว่าใครก็สามารถกำหนดแท็กของเราเองได้ XML นั้นไม่ใช่ภาษาโดยสมบูรณ์มันเป็นมาตรฐานข้อมูลมากกว่า โดยตัวโปรแกรมประยุกต์จะเป็นผู้กำหนดรูปแบบของตัวเองขึ้นและจะสามารถใช้ได้กับโครงสร้างข้อมูลที่ถูกอนุญาต (เพราะว่ามีรูปแบบของข้อมูลที่เข้ากันได้) XML นั้นเป็นภาษาที่ case sensitive ดังนั้นการที่เราเขียนว่า <student> กับ <Student> จึงถือว่าเป็นคนละแท็กกัน นอกจากนี้แล้ว element ใน XML สามารถบรรจุอยู่ใน element อื่นๆได้ยกตัวอย่างเช่น

```
<student>
  <name>example name</name>
  <id>123456789</id>
</student>
```

### 2.9.2 Attribute

โดย attribute ที่มีรูปแบบดังนี้

```
<student name="example_name"></student>
<student name='example_name'></student>
```

จะเห็นว่าทั้งสองแบบมีความเหมือนกันแตกต่างกันเล็กน้อยคือใช้เครื่องหมาย " กับ ' ซึ่งสามารถใช้ได้ทั้งคู่

### 2.9.3 โครงสร้างของเอกสาร XML

โครงสร้างของเอกสาร XML ถูกกำหนดขึ้นโดยลำดับชั้น โดยเอกสารใด ๆ นั้นต้องมี root element หนึ่งตัวเสมอ เช่นในที่นี้คือ <student>

```
<?xml version="1.0" ?>

<student>

    <name>example name</name>

    <id>123456789</id>

</student>
```

### 2.9.4 การประกาศ XML

ส่วนต่างๆของ XML คือ node โดย node ที่ปรากฏบน XML ทุกฉบับคือ การประกาศ Declaration ซึ่งมีลักษณะเหมือนแท็กแต่มีเครื่องหมาย ? อยู่ด้วย โดยในเอกสาร XML จะต้อง มี node นี้ทุกฉบับ

```
<?xml version="1.0" ?>
```

### 2.9.5 จุดเด่นของ XML

- 1) ดูเอกสารได้ง่าย สะดวก และได้ผลดีเหมือน HTML
- 2) สนับสนุนการประยุกต์ใช้กับงานต่าง ๆ และสนับสนุนโปรแกรมประยุกต์ต่าง ๆ
- 3) เขียนง่าย
- 4) อ่านได้ด้วยมนุษย์ โดยไม่ต้องอาศัยโปรแกรมหรือเครื่องมือช่วยแปล
- 5) การเขียน XML ทำได้ด้วยการใช้ Text editor ทั่วไป และไม่ต้องการเครื่องมือที่ซับซ้อน
- 6) ใช้เป็นตัวควบคุมข้อมูล (Meta data) จึงเป็นแนวทางในการขนส่งข้อมูล และสร้างการเชื่อมโยงระหว่างแอปพลิเคชันได้ง่าย
- 7) สนับสนุน UNICODE ทำให้ใช้ได้กับหลากหลายภาษา และผสมกันได้หลากหลาย

ภาษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 8) ดึงเอกสาร XML มาใช้งานได้ง่าย และใช้ร่วมกับโปรแกรมประยุกต์อื่นได้ง่าย เช่น โปรแกรม DB2, Oracle, SAP เป็นต้น
- 9) นำไปประยุกต์ใช้ในการดำเนินกิจกรรมบนเครือข่ายได้มาก เช่น E-Business, EDI, E-Commerce, การจัดการ Supply chain / Demand chain management, การดำเนินการแบบ intranet และ Web Base Application

## 2.10 ระบบบริหารความสัมพันธ์ลูกค้า (Customer Relationship Management หรือ CRM) [15]

CRM ย่อมาจาก Customer Relationship Management คือ การบริหารความสัมพันธ์กับลูกค้า ซึ่งให้ลูกค้ามีความรู้สึกผูกพันกับสินค้า บริการ หรือองค์กร เมื่อลูกค้ามีความผูกพันในทางที่ดีแล้ว ลูกค้าก็จะไม่เปลี่ยนใจไปจากสินค้า หรือบริการ ทำให้มีฐานลูกค้าที่มั่นคง และนำมาซึ่งความมั่นคงขององค์กร ดังนั้น การที่จะเข้าใจลูกค้าได้นั้น จะต้องอาศัยการสังเกตพฤติกรรมของลูกค้า แล้วนำมาวิเคราะห์หาความเกี่ยวข้องระหว่างพฤติกรรมของลูกค้า กับกลยุทธ์ทางการตลาดขององค์กร ซึ่งกระบวนการทำงานของระบบ CRM มี 4 ขั้นตอนดังนี้

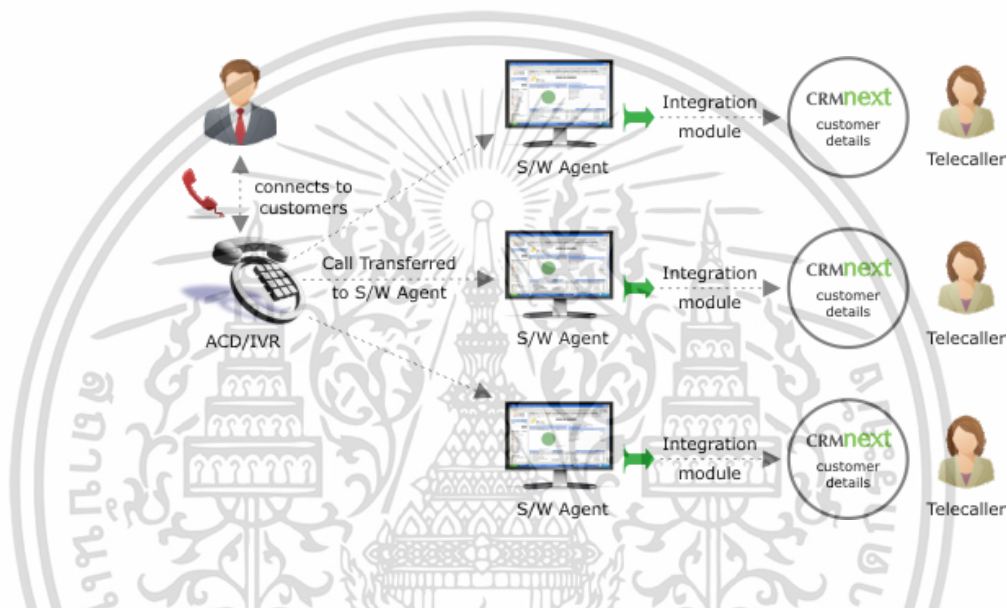
1. *Identify* คือ การเก็บข้อมูลว่าลูกค้าของบริษัทเป็นใคร เช่น ชื่อลูกค้า ข้อมูลสำหรับติดต่อกับลูกค้า เป็นต้น
2. *Differentiate* คือ การวิเคราะห์พฤติกรรมของลูกค้าแต่ละคน และจัดแบ่งลูกค้าออกเป็นกลุ่มตามประเภทลูกค้า
3. *Interact* คือ การมีปฏิสัมพันธ์กับลูกค้าเพื่อเรียนรู้ความต้องการของลูกค้า และเพื่อสร้างความพึงพอใจให้กับลูกค้าในระยะยาว
4. *Customize* คือ การนำเสนอสินค้า หรือบริการที่มีความเหมาะสมเฉพาะตัวกับลูกค้าแต่ละคน

นอกจากนั้นแล้ว CRM ยังมีความสัมพันธ์เกี่ยวกับเทคโนโลยีสารสนเทศในด้านการนำซอฟต์แวร์มาใช้ในการพัฒนาเพื่อดึงดูดลูกค้า สร้างความสัมพันธ์กับลูกค้า เช่น Website เป็นต้น ซึ่ง CRM software จะช่วยในการจัดการฐานข้อมูลลูกค้า วิเคราะห์ข้อมูล และเป็นช่องทางในการติดต่อกับลูกค้า โดย CRM software มักแบ่งออกเป็น 3 ส่วนดังนี้

1. *Operational CRM* เป็นซอฟต์แวร์ Front Office ที่ใช้จัดการกระบวนการทางธุรกิจที่เกี่ยวข้องกับลูกค้า ไม่ว่าจะเป็น Sales, Marketing หรือ Service เช่น การจัดการข้อมูลติดต่อกับลูกค้า การเสนอราคา การบริหารฝ่ายขาย การเก็บข้อมูลพฤติกรรมลูกค้า ระบบบริการลูกค้า เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. *Analytical CRM* ระบบวิเคราะห์ข้อมูลลูกค้าที่รวบรวมได้จากส่วน Operational CRM หรือจากแหล่งอื่นๆ เพื่อแบ่งลูกค้าออกเป็นกลุ่ม และค้นหากลุ่มลูกค้าเป้าหมายที่บริษัทสามารถนำเสนอสินค้า หรือบริการเพิ่มเติมได้
3. *Collaborative CRM* ระบบช่วยสนับสนุนในการติดต่อปฏิสัมพันธ์กับลูกค้าผ่านทางช่องทางต่างๆ เช่น ติดต่อส่วนตัว จดหมาย แฟกซ์ โทรศัพท์ เว็บไซต์ E-Mail เป็นต้น รวมถึงช่วยจัดการทรัพยากรที่บริษัทมี คือพนักงาน กระบวนการทำงาน และฐานข้อมูล (Database) เพื่อนำไปให้บริการแก่ลูกค้า และช่วยรักษาฐานลูกค้าของบริษัทให้ดีขึ้น



รูปที่ 2.18 ระบบบริหารความสัมพันธ์ลูกค้า (CRM Software)

### 2.10.1 ประโยชน์ของ CRM ต่อธุรกิจ

1. CRM ช่วยเพิ่มความสามารถในการให้บริการลูกค้าได้ดีขึ้น เช่น ใช้ Website ในการให้ข้อมูลเกี่ยวกับสินค้า วิธีการใช้สินค้า และให้ความช่วยเหลือแก่ลูกค้าได้ตลอด 24 ชั่วโมง เป็นช่องทางให้ลูกค้าแนะนำติชมต่อบริการของบริษัทได้ง่าย ช่วยให้ลูกค้าสามารถ customize ความต้องการของตนเองได้ทันที เป็นต้น
2. CRM ช่วยพัฒนาความสัมพันธ์ระหว่างบริษัทกับลูกค้าให้ดีขึ้น ช่วยให้บริษัทรู้ความสนใจ ความต้องการ และพฤติกรรมการซื้อของลูกค้า ทำให้บริษัทสามารถนำเสนอสินค้าที่เหมาะสม กับลูกค้าได้ และช่วยให้บริษัทสามารถให้บริการหลังการขายแก่ลูกค้าตามที่ลูกค้าต้องการได้ การสร้างความสัมพันธ์กับลูกค้าในระยะยาวจะช่วยเพิ่มความจงรักภักดี (loyalty) ที่ลูกค้ามีต่อบริษัท ลดการสูญเสียลูกค้า ลดต้นทุนการตลาด เพิ่มรายได้จากการที่ลูกค้าซื้อซ้ำหรือแนะนำให้คนรู้จักซื้อสินค้าของบริษัท และหมายถึงกำไรของบริษัทที่เพิ่มขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.10.2 การเพิ่มประสิทธิภาพของเจ้าหน้าที่ Call Center ด้วยระบบ CRM [2]

ระบบ CRM (Customer Relationship Management Software) สามารถเพิ่มประสิทธิภาพของเจ้าหน้าที่ที่ทำงานในศูนย์บริการลูกค้า (Call Center) ได้ เนื่องจากเป็นระบบที่รวบรวมข้อมูลของลูกค้า และกิจกรรมต่างๆ ที่ลูกค้าได้ติดต่อมายัง Call Center เพื่อให้เจ้าหน้าที่ได้ค้นหาข้อมูลเพิ่มเติม โดยไม่ต้องขอข้อมูลซ้ำจากลูกค้า หากองค์กรมีระบบ CRM ซึ่งได้การออกแบบ ให้ทำงานเข้ากับระบบ call center แล้ว ก็จะสามารถเพิ่มอัตราการแก้ปัญหาในสายแรก (Increase First Call Resolution Rate) ลดระยะเวลาในการคุยสายกับลูกค้า (Reduce Call Time) โดยระบบ CRM จะมีระบบปฏิบัติการเชื่อมต่อระหว่าง call center และระบบของหน่วยงานอื่นๆ สามารถค้นหาข้อมูลแบบ real-time ทำให้เกิดความรวดเร็วในการแก้ปัญหา และพัฒนาความพึงพอใจแก่ลูกค้าได้ เพิ่มโอกาสในการขายได้มากขึ้น (Improve Up Sell and Cross Sell) และสามารถพัฒนาระบบการติดต่อของเจ้าหน้าที่ในองค์กรให้มีประสิทธิภาพดียิ่งขึ้น (Improve Call Center Workflows) ปัญหาที่มักพบในการทำงานของเจ้าหน้าที่เกี่ยวกับการทำงานบนจอแสดงผล (Monitor) ในกรณีที่เจ้าหน้าที่จำเป็นต้องเปิดหน้าจอแสดงผลหลายๆจอ พร้อมกัน เพื่อไว้เพื่อใช้ในการแก้ปัญหาให้กับลูกค้า บ่อนข้อมูลการให้บริการพร้อมทั้งต้องส่งข้อมูลออกไปยังพนักงานอื่นเพื่อหาข้อมูล ตรวจสอบสถานะและจัดการข้อมูลพื้นฐานอื่นๆด้วยหน้าจอแสดงผลของระบบ CRM นั้นจะมีความยืดหยุ่นสูง โดยสามารถออกแบบเครื่องมือระบบจอแสดงผลที่จะดึงเอาข้อมูลจากฐานข้อมูลลูกค้า ใน Application ของ CRM และข้อมูลจาก Application ต่างๆใน call center มาแสดงบนจอแสดงผลที่มีการจัดการไว้ได้ ซึ่งจะช่วยให้เจ้าหน้าที่ เข้าสู่ขั้นตอนการทำงานในส่วนของแต่ละคนได้ในเวลาน้อยที่สุด ทั้งหมดนี้จะช่วยให้เจ้าหน้าที่ทำงานได้รวดเร็วขึ้น พัฒนาการบริการลูกค้าและพนักงาน call center ยังลดความผิดพลาดจากการทำงานได้อีกด้วย

## 2.11 เว็บเซอร์วิส (Web service) [16]

เว็บเซอร์วิส (web service *บริการบนเว็บ*) คือระบบซอฟต์แวร์ที่ออกแบบมา เพื่อสนับสนุน การแลกเปลี่ยนข้อมูลกัน ระหว่างเครื่องคอมพิวเตอร์ผ่านระบบเครือข่าย โดยที่ภาษาที่ใช้ในการ ติดต่อสื่อสารระหว่างเครื่องคอมพิวเตอร์ คือเอกซ์เอ็มแอล(XML) เว็บเซอร์วิสมีอินเทอร์เน็ตเฟส ที่ใช้ อธิบายรูปแบบข้อมูลที่เครื่องคอมพิวเตอร์ประมวลผลได้ เช่น WSDL ระบบคอมพิวเตอร์ใช้งานสื่อสาร โต้ตอบกับเว็บเซอร์วิสตามรูปแบบที่ได้กำหนดไว้แล้ว โดยการส่งสาสน์ตามอินเทอร์เน็ตเฟสของเว็บเซอร์วิส นั้น โดยที่สาสน์ดังกล่าวอาจแนบไว้ใน SOAP Developpe หรือส่งตามอินเทอร์เน็ตเฟสในแนวทาง ของ REST สาสน์เหล่านี้ถูกส่งโดยอาศัย protocol HTTP และใช้ XML ร่วมกับมาตรฐาน เกี่ยวกับเว็บอื่นๆ โปรแกรมประยุกต์ที่เขียนโดยภาษาต่างๆ และทำงานบนแพลตฟอร์มต่างๆกัน สามารถใช้เว็บเซอร์วิสเพื่อแลกเปลี่ยนข้อมูลผ่านทางเครือข่ายคอมพิวเตอร์ เช่น อินเทอร์เน็ต ใน ลักษณะเดียวกับการสื่อสารระหว่างโปรเซส (Inter-process communication) บนเครื่องเดียวกัน ความสามารถในการแลกเปลี่ยนข้อมูลระหว่างระบบที่ต่างกันนี้ เช่น การแลกเปลี่ยนข้อมูลระหว่าง ) โปรแกรมที่เขียนโดยภาษาจาวา และโปรแกรมที่เขียนโดยภาษาไพทอน หรือการแลกเปลี่ยนข้อมูล ระหว่างโปรแกรมประยุกต์ที่ทำงานบนไมโครซอฟท์วินโดวส์และโปรแกรมประยุกต์ที่ทำงานบนลินุกซ์) เกิดขึ้นได้เนื่องจากการใช้มาตรฐานเปิด โดย OASIS และ W3C เป็นคณะกรรมการหลักในการ รับผิดชอบมาตรฐานและสถาปัตยกรรมของเว็บเซอร์วิส

### 2.11.1 มาตรฐานหลักของเว็บเซอร์วิส

มาตรฐานหลักของเว็บเซอร์วิสจะประกอบไปด้วยมาตรฐานต่าง ๆ ดังนี้ XML WSDL SOAP และ UDDI

- 1) Extensible Markup Language (XML)
- 2) Simple Object Access Protocol (SOAP)
- 3) Web Services Description Language (WSDL)
- 4) Universal Description, Discovery and Integration (UDDI)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

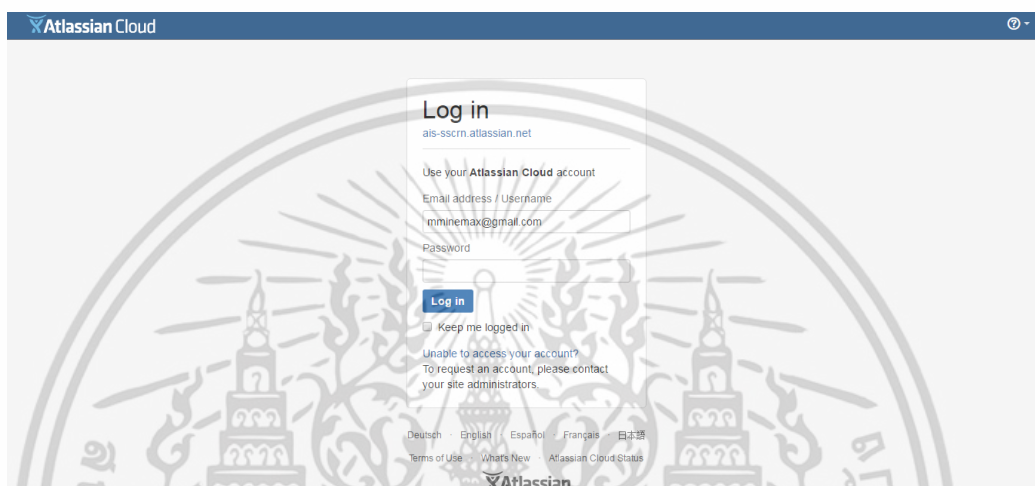
### 2.11.2 คุณลักษณะของ Web Service

- 1) เว็บเซอร์วิสเป็นซอฟต์แวร์คอมพิวเตอร์ที่ระบุตำแหน่งโดยใช้ URI
- 2) อินเทอร์เน็ตและการติดต่อโดยใช้ภาษา XML
- 3) เว็บเซอร์วิสสนับสนุนการเรียกใช้จากซอฟต์แวร์ประยุกต์อื่น ๆ ผ่านโปรโตคอลอินเทอร์เน็ต
- 4) เว็บเซอร์วิสใช้เอกสารแบบ XML ในการส่งข้อมูลระหว่างผู้ให้บริการและผู้ใช้
- 5) เว็บเซอร์วิสช่วยในการเชื่อมโยงโปรแกรมที่ทำงานอยู่ต่างแพลตฟอร์มสามารถทำงานร่วมกันได้ Cross-Platform Integration
- 6) สามารถพัฒนาเว็บเซอร์วิสได้โดยใช้โปรแกรมภาษาคอมพิวเตอร์ต่างๆเช่น Java, C, C#
- 7) สามารถแปลงซอฟต์แวร์คอมพิวเตอร์ให้เป็น Web Service ได้
- 8) เว็บเซอร์วิสจะไม่รวมถึงการจัดการด้านการแสดงผล (Presentation) เหมือน HTML
- 9) เว็บเซอร์วิสจะเป็นซอฟต์แวร์คอมพิวเตอร์แบบ Loosely Couple
- 10) เราสามารถที่จะค้นหาและเรียกใช้เว็บเซอร์วิสจาก registry ที่เป็นแบบ public หรือ private โดยใช้ มาตรฐานกลางเช่น UDDI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.12 Jira

Jira คือโปรแกรมที่ช่วยจัดการงานต่างๆ ใน Project ใน Software Development project ส่วนใหญ่ก็จะประกอบไปด้วยงานหลายๆ งาน งานพัฒนาฟังก์ชันงานต่างๆ หรืองานทดสอบฟังก์ชันงานที่ได้พัฒนาขึ้น ซึ่ง Jira ช่วยในการติดตามงานต่างๆ และยังสามารถแบ่งงานออกเป็นเฟสต่างๆ หรือที่เรียกว่าเป็นแต่ละรีลีส (release) นอกจากนั้นยังสามารถสร้าง release notes ที่จะใช้เป็นเอกสารร่วมกับการ release



รูปที่ 2.19 Jira

### 2.12.1 ประเภทของงานใน Jira

งานในแต่ละงานใน Jira แบ่งเป็นแต่ละประเภท ตัวอย่างเช่น

- Bug ปัญหาการทำงานของโปรแกรม
- Improvement งานที่จะต้องพัฒนาให้ดีขึ้น
- New feature ความสามารถในการทำงานต้องการจะเพิ่มเข้ามาในระบบ
- Task งานทั่วไปที่จะต้องทำให้สำเร็จ

แต่ในการใช้งาน Jira สามารถปรับแต่งประเภทของงานได้ตามที่ต้องการ ไม่จำเป็นต้องใช้ตามนี้ ตัวอย่างการปรับแต่งประเภทของงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### Project management type

- Task หมายถึง งานต่างๆ ไป หรือปัญหาที่ต้องได้รับการแก้ไขใน Project
- Risk หมายถึง งานที่ทำเพื่อป้องกันปัญหาที่จะเกิดขึ้นในอนาคต
- New Feature หมายถึง ความสามารถในการทำงานที่ต้องการให้มีการเปลี่ยนแปลง

### Analysis and Design type

- Story งานที่เกี่ยวข้องกับการวิเคราะห์ระบบ เช่นการทำ Use Case, Activity Diagram
- Model งานที่เกี่ยวข้องกับการออกแบบระบบ เช่น การทำ Class Diagram, Database Diagram

### Development type

- Code หมายถึงงานพัฒนาโปรแกรม

### Testing type

- Test Event หมายถึงการทำ Test Script
- Test Execution หมายถึงการตรวจสอบการทำงานของโปรแกรม โดย Test Script ที่ได้ทำจาก Test Event
- Bug ในกรณีที่ตรวจสอบแล้วพบข้อบกพร่องในการทำงาน ก็จะต้องสร้าง เป็น Defect เพื่อส่งกลับไปให้นักพัฒนาระบบแก้ไข

### Deployment type

- Release Task เมื่อทดสอบและแก้ไขแล้ว ก็จะต้อง release หรือติดตั้งโปรแกรม บน server ซึ่งอาจจะเป็น development server หรือ production server ก็แล้วแต่ เรียกว่าเป็น Release Task

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The screenshot shows the Jira Issue Navigator interface. The main heading is "High priority for 2.0". Below the heading, there is a search bar with the filter criteria: "project = ANGRY AND updated >= -1d OR com". The table below lists various issues with the following columns: Key, Summary, Assignee, Reporter, P (Priority), Status, Resolution, Created, Updated, and Due. The issues listed include ANGRY-232, ANGRY-226, ANGRY-186, ANGRY-168, ANGRY-163, ANGRY-156, ANGRY-148, ANGRY-121, ANGRY-102, ANGRY-87, ANGRY-85, ANGRY-71, ANGRY-67, ANGRY-62, ANGRY-45, ANGRY-41, and ANGRY-34.

รูปที่ 2.20 Jira Screen

### 2.12.2 ความสำคัญ (Priority)

- Blocker เป็นงานที่ถ้าไม่ทำให้เสร็จ ก็จะกั้นไม่ให้งานอื่นๆ สำเร็จได้ คือเป็นงานที่ block ชาวบ้านนั่นเอง
- Critical เป็นงานที่มีความสำคัญมาก ซึ่งทำให้การทำงานในฟังก์ชันใดๆ ไม่สามารถทำงานได้
- Major เป็นงานที่มีผลกระทบต่อการทำงานโดยรวม
- Minor เป็นงานที่กระทบกับการทำงานแต่น้อย ซึ่งโปรแกรมยังสามารถทำงานต่อ
- Trivial เช่นการแก้ไขคำผิด หรืองานใดๆ ที่ไม่มีผลต่อการทำงานของฟังก์ชัน

งานแต่ละงานใน Jira จะเก็บรายละเอียดดังกล่าวข้างต้นแล้วยังเก็บ หัวข้อ, รายละเอียดของงาน, ความคิดเห็นที่ได้ตอบไปมาเกี่ยวกับงานนั้นๆ และยังสามารถ attach file รายละเอียดหรือข้อมูลเพิ่มเติมในงานแต่ละงานได้ และเก็บเวลาที่ใช้ในการทำงานของแต่ละงานหนึ่งๆ ได้ เพื่อใช้ในการประเมินงาน และเป็นสถิติของโปรเจคต่อไป

Jira ยังมีความสามารถอื่นๆ ที่ช่วยในการทำงานของโปรเจคได้ ดังจะเห็นว่าหลายๆ โปรเจค รวมทั้ง open source project ก็ยังนำโปรแกรมตัวนี้มาช่วยในการทำงาน เนื่องจากเป็น web application ทำให้ user ไม่ต้องลำบากในการลงโปรแกรม และยังเข้าถึงได้ง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

# วิธีการดำเนินงานวิจัย

ในบทนี้จะกล่าวถึงขั้นตอนในการทดสอบซอฟต์แวร์ของระบบบูรณาการข้อมูลสำหรับศูนย์บริการข้อมูลลูกค้า โดยในการทดสอบนั้นจะดำเนินการตามแผนที่ได้วางไว้ดังต่อไปนี้

- 1) การศึกษาระบบบูรณาการข้อมูลสำหรับศูนย์บริการข้อมูลลูกค้า
- 2) การทำความเข้าใจจุดประสงค์ และขอบเขตในการทดสอบระบบ
- 3) การออกแบบกรณีทดสอบตามกรณีทดสอบที่ได้รับ
- 4) พัฒนาโปรแกรมทดสอบอัตโนมัติ (Automated Testing)
- 5) วิเคราะห์ และจัดการกับข้อผิดพลาดที่เกิดขึ้น

### 3.1 การศึกษาระบบบูรณาการข้อมูลสำหรับศูนย์บริการข้อมูลลูกค้า

เนื่องจากการทดสอบซอฟต์แวร์ มีจุดประสงค์เพื่อที่จะทดสอบความถูกต้องในการทำงานของระบบบูรณาการข้อมูลสำหรับศูนย์บริการข้อมูลลูกค้าโดยเฉพาะ ดังนั้นจึงมีความจำเป็นที่ผู้เขียนจะต้องทำความเข้าใจในการทำงานของระบบบูรณาการข้อมูลสำหรับศูนย์บริการข้อมูลลูกค้า ทั้งในเรื่องของมุมมองธุรกิจ และมุมมองทางด้านเทคนิค เนื่องจากระบบจะมีกฎเกณฑ์ในการทำงาน และการแสดงผล ซึ่งถ้าไม่เข้าใจในกฎเกณฑ์หรือพฤติกรรมทางด้านธุรกิจของระบบอาจส่งผลให้พัฒนาซอฟต์แวร์ที่ใช้ในการทดสอบผิดพลาดได้ ซึ่งจะส่งผลเสียอย่างมาก เนื่องจากผลการทดสอบที่ได้ อาจจะออกมาถูกต้องทั้งที่ระบบทำงานผิดพลาด ดังนั้นเพื่อให้ซอฟต์แวร์ที่พัฒนาขึ้นมีความถูกต้อง จึงต้องศึกษาระบบให้เกิดความเข้าใจมากที่สุด เพื่อที่จะสามารถทดสอบระบบที่ครอบคลุมกรณีปัญหาต่างๆที่เกิดขึ้นได้จริงในการทำงานได้อย่างครบถ้วน

โดยระบบบูรณาการข้อมูลสำหรับศูนย์บริการข้อมูลลูกค้า เป็นระบบที่จัดอยู่ในประเภท Web Application ประกอบกับการทำงานกับ Web Service ต่างๆ กับทางบริษัท Fusionware Company Limited ซึ่งมีการนำหลักสถาปัตยกรรมเชิงบริการ (Service Oriented Architecture หรือ SOA) มาใช้เป็นมาตรฐานในการพัฒนาระบบขนาดใหญ่ และมีการทำงานที่ซับซ้อน เพื่อทำให้เกิดความสะดวกสบายต่อการใช้งาน และนำหลักการของระบบบริหารความสัมพันธ์ลูกค้า (Customer Relationship Management หรือ CRM) มาใช้ในการพัฒนา Website ให้ดึงดูดลูกค้า สร้างความสัมพันธ์กับลูกค้า และวิเคราะห์ข้อมูลต่างๆของลูกค้า เพราะฉะนั้นระบบนี้จึงถูกสร้างขึ้นมาเพื่อช่วยอำนวยความสะดวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต่อผู้รับสาย (Agent) ในการให้บริการข้อมูลสำหรับศูนย์บริการข้อมูลลูกค้า ให้มีความรวดเร็ว ใช้งานง่ายขึ้น และสามารถรับสายได้เพิ่มมากขึ้น ทำให้ลูกค้าที่โทรเข้ามาไม่ต้องถือสายรอการทำรายการเป็นเวลานาน

โดยระบบบูรณาการข้อมูลสำหรับศูนย์บริการข้อมูลลูกค้า (Font Office) จะมีรายละเอียดการทำงานของหน้าจอต่างๆดังต่อไปนี้

1) หน้า Home

สำหรับแสดงรายละเอียดรูปแบบ เช่น ชื่อ นามสกุล ของหมายเลขต่างๆ ที่ถูก Query ซึ่งผู้ใช้สามารถเลือก รูปแบบของ Service ที่ลูกค้าสนใจ ทั้งในประเทศและต่างประเทศ

2) หน้า Customer Information

สำหรับแสดงรายละเอียดของลูกค้า การใช้งาน Package และบริการต่างๆที่ใช้ในปัจจุบัน รวมไปถึงแสดงการแจ้งเตือนต่างๆ เช่น แจ้งเตือนการอายุพรวันเกิดลูกค้า หรือการแจ้งเตือนเมื่อยอดเงินใกล้หมด เพื่อให้ผู้ใช้งานระบบ (Agent) ทราบถึงรายละเอียดต่างๆของลูกค้า และสามารถยืนยันตัวตนของลูกค้าได้ ซึ่งในหน้านี้จะเป็นหน้าการแสดงผลที่ดึงข้อมูลมาจากหลายระบบมาไว้ในหน้าจอเดียว

3) หน้า Package Consultation

สำหรับแสดง Package ต่างๆที่เหมาะสมกับลูกค้า และแสดง Package ยอดนิยมในช่วงเวลานั้นๆ เพื่อนำเสนอ Package ที่เหมาะสมต่อการใช้งานให้แก่ลูกค้า โดยจะวิเคราะห์จากข้อมูลการใช้งาน Package ของลูกค้ารายนั้นๆ

4) หน้า Package Management

สำหรับแสดงการทำรายการ เพิ่ม เปลี่ยน และยกเลิก Package ทั้งหมดในการทำรายการนั้นๆ เพื่อเป็นการสรุปการทำรายการก่อนจะยืนยันการทำรายการ โดยจะให้ผู้ใช้งานระบบ (Agent) แจ้งแก่ลูกค้าก่อนยืนยันการทำรายการ และทำการส่งข้อมูลการทำรายการ Promotion ต่างๆ ให้อีกระบบหนึ่ง เพื่อดำเนินการจัดการ Promotion ที่ทำรายการให้แก่ลูกค้า

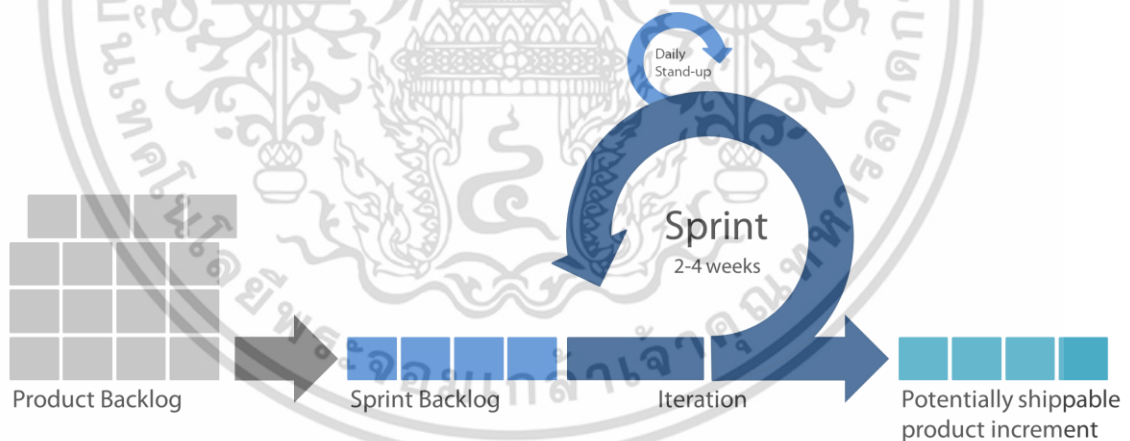
5) หน้า Service Completion

สำหรับแสดงสถานะการทำรายการ เป็น Order Submitted และยังมีแสดง Campaign ต่างๆ เพื่อที่จะแจ้งสถานะการทำรายการว่าเสร็จสมบูรณ์ให้ผู้ใช้งานระบบ (Agent) ทราบ และเมื่อคลิกปุ่ม Finish หรือคลิกปุ่มวางสาย จะเสร็จสิ้นกระบวนการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2 การทำความเข้าใจจุดประสงค์ และขอบเขตในการทดสอบระบบ

จากหัวข้อที่ 3.1 การศึกษาการทำงานของระบบบูรณาการข้อมูลสำหรับศูนย์บริการข้อมูลลูกค้า ซึ่งจะเห็นได้ว่าการทำงานของระบบบูรณาการข้อมูลสำหรับศูนย์บริการข้อมูลลูกค้า ค่อนข้างมีความซับซ้อน เพราะเป็นการรวมระบบต่างๆที่มากกว่า 10 ระบบมาไว้ในระบบบูรณาการข้อมูลสำหรับศูนย์บริการข้อมูลลูกค้าเพียงระบบเดียว เพื่อให้สามารถใช้ได้ภายใน 4 หน้าจอ ดังนั้นในการทดสอบแต่ละครั้งจะต้องมีการแบ่งพีเจอรืในการทดสอบออกเป็นพีเจอรืย่อยๆ ซึ่งจะต้องทำการแบ่งเป็นจำนวนมาก โดย ณ ปัจจุบันการทดสอบความถูกต้องของระบบใช้การทดสอบด้วยมือ หรือ Manual Testing เป็นหลัก จะใช้เวลามากในการทดสอบแต่ละครั้ง เนื่องจากการทดสอบด้วยมือจะต้องทำการ Manual ในแต่ละจุดต่างๆซึ่งอาจจะทำให้การตรวจสอบนั้นทำได้อย่างไม่ทั่วถึง รวมทั้งในการทดสอบแต่ละครั้งสิ่งที่เราทดสอบเป็นการทดสอบฟังก์ชันใหม่ ที่ทำงานต่อจากเดิม เพราะฉะนั้นการทดสอบกรณีแบบนี้ จะต้องกลับไปทดสอบเรื่องเดิมซ้ำๆ เพื่อดูว่ามีพีเจอรืใหม่เพิ่มมาแล้ว ระบบยังสามารถทำงานได้อยู่ โดยไม่ส่งผลกระทบต่อการทำงานเดิมที่สามารถทำงานได้อยู่ก่อนแล้ว อีกข้อสำคัญคือระบบบูรณาการข้อมูลสำหรับศูนย์บริการข้อมูลลูกค้า นั้นถูกพัฒนาขึ้นภายใต้วิธีการพัฒนาซอฟต์แวร์แบบ Scrum model ซึ่งมีรูปแบบในการทำงานดังรูปที่ 3.1



รูปที่ 3.1 แผนผังการทำงานของวงจรการพัฒนาซอฟต์แวร์แบบ Scrum model

โดยจากรูปที่ 3.1 จะเห็นได้ว่าวิธีการพัฒนาแบบ Scrum model นั้นมีกรอบในการพัฒนาในแต่ละรอบค่อนข้างสั้น ประมาณสองถึงสัปดาห์ และในทุกๆ รอบนั้นจะต้องมีการส่งมอบงานที่ใช้งานได้ (Potentially Shippable Product) ซึ่งก่อนที่จะส่งมอบทุกๆครั้งจะต้องมีการตรวจสอบความถูกต้องของระบบว่า ซอร์สโค้ดใหม่ที่พัฒนาขึ้นมาในแต่ละรอบการพัฒนานั้น ไปทำให้ระบบเดิมที่เคยทำงานได้ เกิดข้อผิดพลาดตรงไหนหรือไม่ หรือที่เรียกว่าการทำ Regression testing หรือหากจะทำเอกสารนี้เป็นเอกสารที่ส่งมอบไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การตรวจสอบความถูกต้องโดยรวมว่าระบบสามารถทำงานได้เป็นปกติหรือไม่ ซึ่งในปัจจุบันการทดสอบเหล่านี้จะใช้การทำ Manual Testing ซึ่งถ้ามีการเปลี่ยนแปลงซอร์สโค้ดบ่อยๆ หรือมีกรอบเวลาในการพัฒนาที่สั้นแบบ Scrum model การทำ Manual testing นั้นอาจจะไม่ทันต่อความต้องการ สิ้นเปลืองทรัพยากรมนุษย์ และเวลาอย่างมากในการทดสอบ

ดังนั้น การนำเทคนิค และเครื่องมือในการทดสอบ มาประยุกต์ใช้ในการทดสอบซอฟต์แวร์ สำหรับระบบบูรณาการข้อมูลสำหรับศูนย์บริการข้อมูลลูกค้า จะช่วยให้การทดสอบสามารถทำได้ อย่างถูกต้อง มีคุณภาพ สะดวก รวดเร็ว และครอบคลุม โดยเน้นการเขียนกรณีทดสอบ (Test case) ตามเหตุการณ์หลักที่มักจะถูกดำเนินการในระบบจริง ซึ่งการทดสอบจะใช้การเขียนกรณีทดสอบ ทั้งหมดไว้ล่วงหน้า พร้อมทั้งทำการเตรียมข้อมูลที่ใช้ในการทดสอบ จากนั้นผู้ทดสอบจะทำการทดสอบ ด้วยมือ Manual Testing โดยจะใช้เทคนิคในการทดสอบแบบ Exploratory Test ซึ่งกล่าวไว้แล้วใน บทที่ 2 จากนั้นจะเลือกกรณีทดสอบที่จำเป็นต้องทดสอบทุกครั้ง หรือมีการทดสอบบ่อยๆ (Regression Testing) นำมาสร้างเป็นชุดคำสั่งเพื่อใช้ในการทดสอบ (Test script) โดยทำการสอบอัตโนมัติ Automate Testing ซึ่งเครื่องมือที่นำมาประยุกต์ใช้ ได้แก่ Selenium IDE ใช้เทคนิคการทดสอบแบบ Record and Playback , Selenium WebDriver โดยใช้เทคนิคการทดสอบแบบ TDD และ Robot framework ใช้เทคนิคการทดสอบแบบ ATDD โดยการทดสอบอัตโนมัติจะเป็นการตรวจสอบตั้งแต่ การเข้าสู่ระบบเพื่อการค้นหา และแสดงชื่อของลูกค้าในหน้า Main การตรวจสอบข้อมูลพื้นฐานของลูกค้าในหน้า Customer information และค้นหา Package หลัก ( Main Package ) การทำรายการ โปรโมชัน (การออก Order) จนถึงการวางสายเมื่อทำรายการเรียบร้อยแล้ว ซึ่งการตรวจสอบเหล่านี้จะ ทำด้วยวิธีการทดสอบโดยอัตโนมัติทั้งหมด ทำให้ไม่จำเป็นที่จะต้องคัดลอกเบอร์ หรือคัดลอกแพ็คเกจที่จะ ใช้ทดสอบจาก Excel ทุกครั้ง อีกทั้งไม่จำเป็นที่จะต้องนั่งกดค้นหา และทำรายการออเดอร์ ด้วยมือทั้งหมด

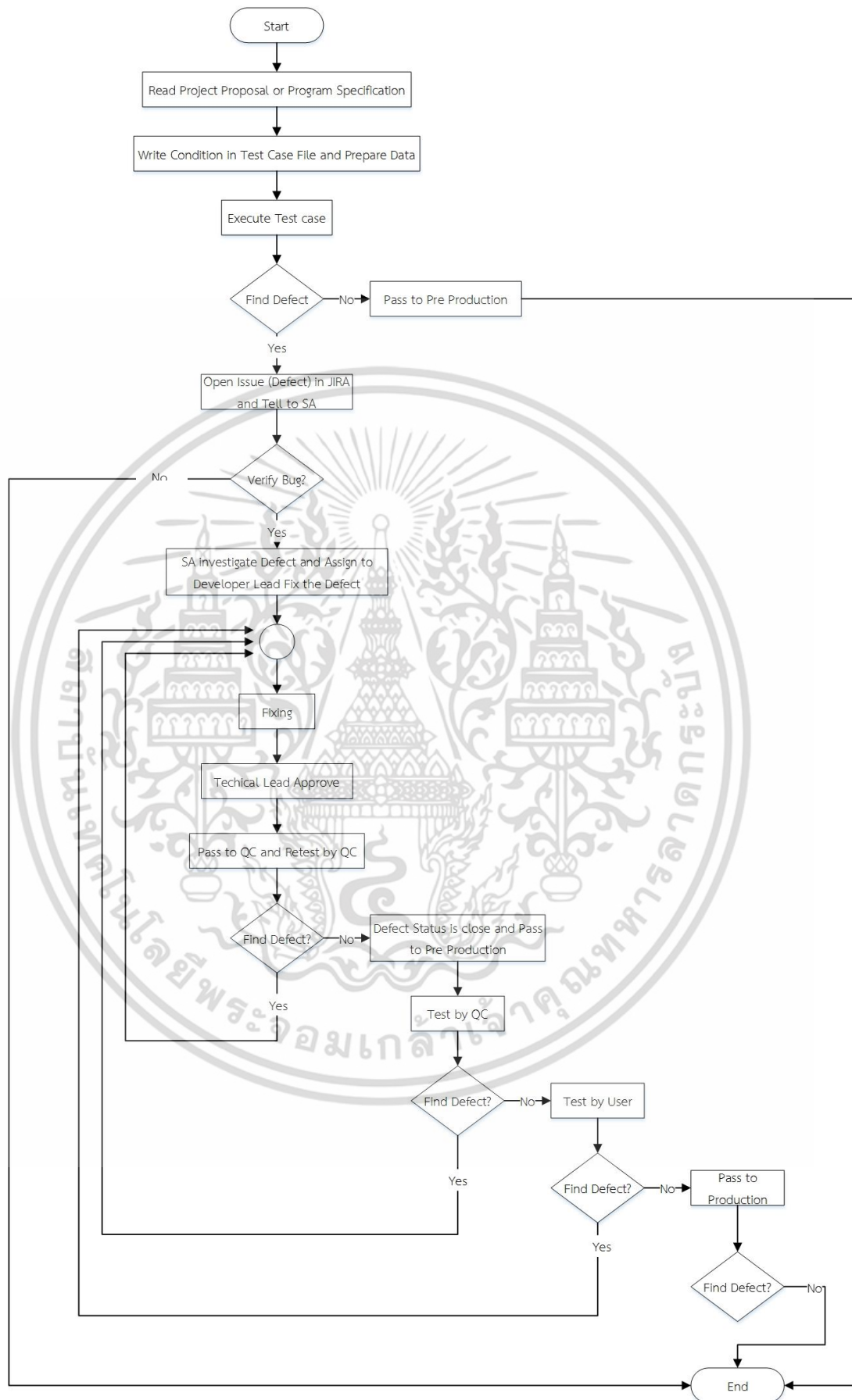
สำหรับส่วนของการทดสอบนั้น จะแบ่งทดสอบออกออกเป็นพีเจอร์ทั้งหมดดังนี้

- 1) ตรวจสอบการแสดงผลข้อมูลพื้นฐานของลูกค้าในหน้า Customer information
- 2) ตรวจสอบการแสดงผลของชื่อลูกค้า เมื่อทำการค้นหาเบอร์โทรศัพท์ในหน้า Home
- 3) ตรวจสอบแสดงผลเมื่อทำการค้นหา Main Package ในหน้า Customer information
- 4) ตรวจสอบการออกออร์เดอร์ว่าสามารถทำงานได้อย่างถูกต้อง

โดยการทดสอบ และพัฒนาซอฟต์แวร์อัตโนมัติ Automated Testing ได้ใช้เครื่องมือในการพัฒนา 3 ตัว เพื่อเปรียบเทียบให้เห็นความแตกต่าง และประสิทธิภาพในการใช้งานเครื่องมือแต่ละตัว คือ Selenium IDE ใช้เทคนิคการทดสอบแบบ Record and Playback , Selenium Web driver ใช้เทคนิคการทดสอบแบบ TDD และ Robot framework ใช้เทคนิคการทดสอบแบบ ATDD ซึ่งการนำเทคนิคต่างๆมาประยุกต์ใช้ในการทดสอบระบบจะทำให้สามารถทดสอบได้สะดวก ง่ายตายและยังมีประสิทธิภาพเพิ่มมากขึ้น

หลังจากเข้าใจวัตถุประสงค์ในการทดสอบซอฟต์แวร์อัตโนมัติแล้ว จากนั้นจึงต้องทำความเข้าใจเกี่ยวกับกระบวนการทดสอบระบบบูรณาการข้อมูลสำหรับศูนย์บริการข้อมูลลูกค้าด้วย เพื่อให้เข้าใจใน Flow การทำงานตั้งแต่เริ่มต้นจนจบ โดยกระบวนการทดสอบจะแสดงด้วยแผนผังในรูปที่

3.2



รูปที่ 3.2 Flow ขั้นตอนการทดสอบระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## อธิบายขั้นตอนการทำงาน

1. ศึกษา Project Proposal หรือที่เรียกว่า Program Specification เพื่อทำความเข้าใจในรายละเอียดการทำงานของ Feature ต่างๆ รวมไปถึงขั้นตอนการทำงานของ Feature นั้นๆ
2. ทำการเขียน Testcase ของการทำงานใน Feature ต่างๆ โดยแบ่งเป็นเงื่อนไขต่างๆ ที่อาจจะเกิดขึ้นได้ พร้อมทั้งเขียนผลลัพธ์ที่คาดหวัง และข้อมูลที่จะนำมาทดสอบ ( Test data )
3. ทำการทดสอบโปรแกรม โดยดำเนินการทดสอบทำการทดสอบตามขั้นตอน และเงื่อนไขที่เขียนใน Testcase ข้อ 2 เพื่อตรวจสอบว่า ได้ผลลัพธ์ตรงตามที่คาดหวังใน Project Proposal หรือ Program Specification หรือไม่
  - 3.1 หากพบปัญหา หรือข้อผิดพลาดที่เกิดจากการทำงานของระบบที่ไม่ตรงตาม Project Proposal หรือ Program Specification จะต้องข้อผิดพลาดทั้งหมดแจ้งให้ Business Analyst (BA) ทราบ และทำการเปิด Defect ใน JIRA เพื่อรอให้ SA มาวิเคราะห์และยืนยันว่าปัญหาที่เกิดขึ้นเป็น Defect ซึ่งเกิดจากระบบทำงานผิดพลาดไม่ตรงกับ Project Proposal หรือ Program Specification
  - 3.2 เมื่อ SA ยืนยันแล้วว่าปัญหาที่เกิดขึ้นเป็น Defect จะทำการตรวจเพื่อทำการวิเคราะห์ปัญหา (Investigate) และจัดการปัญหา โดย Developer Leader จะทำการแบ่งงานให้ Developer ในทีมแต่ละคน แก้ไขข้อผิดพลาดจากการทำงานของระบบ
  - 3.3 จากนั้น Developer ที่ได้รับมอบหมาย จะทำการแก้ไขข้อผิดพลาดที่เกิดขึ้น โดย Defect ข้อที่กำลังถูกแก้ไขนั้นจะมี Status ใน JIRA เป็น Fixing
  - 3.4 หลังจากทำการแก้ไขเรียบร้อยแล้ว Developer จะส่ง Defect ข้อนั้นมาให้ Technical Lead เพื่อทำการยืนยันว่าได้ทำการแก้ไขแล้วเรียบร้อยแล้วพร้อมส่งต่อให้ Tester team ทำการทดสอบในขั้นตอนต่อไปได้ โดยใน JIRA จะมี Status เป็น Technical lead approve

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 เมื่อ Technical Lead ทำการยืนยันแล้วว่า Defect ถูกแก้ไขเรียบร้อยแล้ว Status ใน JIRA จะแสดงเป็น Pass to QC ซึ่งทางทีม Tester จะต้องทำการทดสอบ Defect ที่เกิดขึ้นอีกครั้ง หรือเรียกว่า การทำ Regression Test โดย Tester ที่ทำการทดสอบ จะต้องตรวจสอบอีกครั้งว่า หลังจากแก้ไขเรียบร้อยแล้ว ระบบสามารถทำงานได้ปกติ Defect ที่ทำการแก้ไ้นั้นไม่ส่งผลกระทบต่อการทำงานของระบบ หากมีข้อผิดพลาดเกิดขึ้นจะกลับไปแก้ไขข้อผิดพลาดในขั้นตอนที่ 3.3

4. หลังจากทำการทดสอบจาก Environment Develop เรียบร้อยแล้ว ถัดไปจะเข้าสู่ Environment ถัดไปคือ Pre Production เป็นการทดสอบโดยจำลองเครื่อง และข้อมูลทั้งหมดเสมือนจริง เพื่อตรวจสอบความถูกต้องก่อนจะทำการขึ้นไปทดสอบบน Production จริง ซึ่งในขั้นตอนนี้ Tester จะทำการทดสอบเพื่อหาข้อผิดพลาดอีกครั้ง โดยขั้นตอนต่างๆที่ทำการทดสอบจะทำเหมือนบน Environment Develop จะเปลี่ยนเพียงข้อมูลที่ใช้ในการทดสอบเท่านั้น หากทดสอบแล้วมีข้อผิดพลาดเกิดขึ้น จะต้องกลับไปทำการแก้ไข Defect ในขั้นตอนที่ 3.3 อีกครั้ง และพีเจอร์ที่มีผลกระทบกับ Defect นั้นจะไม่สามารถขึ้น Production ใน Sprint นั้นๆได้ โดยจะถูกเลื่อนไปขึ้น Sprint ถัดไปหากมีการแก้ไขเรียบร้อยแล้ว

5. เมื่อทีมผู้ทดสอบ Tester ทำการทดสอบเรียบร้อยแล้ว ในขั้นตอนการทดสอบนี้จะมีเพียงพีเจอร์ที่ทำงานได้เท่านั้น เนื่องจากพีเจอร์ที่มีข้อผิดพลาดเกิดขึ้นนั้นได้ถูกส่งกลับไปแก้ไขในข้อที่ 3.3 ซึ่งขั้นตอนถัดไป ตัวแทนของ User จะทำการตรวจรับงาน หรือ UAT โดย ตัวแทนของ User จะทำการทดสอบระบบเพื่อหาข้อผิดพลาด ว่ามีข้อผิดพลาดเกิดขึ้นหรือไม่

5.1 หากพบข้อผิดพลาดที่เกิดจากการทำงานของโปรแกรม User ที่ทำการทดสอบจะทำการรายงานข้อผิดพลาดลงในระบบ JIRA เพื่อบันทึกผลของข้อผิดพลาดจากการทำงานของโปรแกรม ใน Environment Pre-production พร้อมทั้งใส่รายละเอียดของปัญหาที่พบนั้นๆ เช่น Location (ส่วนที่เกิดปัญหา), Step (ขั้นตอนการทำให้เกิดปัญหา), Severity (ระดับความรุนแรงของปัญหา), Open Date (วันที่ทดสอบแล้วเกิดปัญหา) เป็นต้นโดยเมื่อเกิดข้อผิดพลาดต้องกลับไปแก้ไข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อผิดพลาดในขั้นตอนที่ 3.3 ต่อไป และนำพีเจอร์ที่พบข้อผิดพลาดมาทดสอบอีกครั้งใน Sprint ถัดไป

5.2 หากไม่พบข้อผิดพลาด จะนำพีเจอร์ที่ทำงานถูกต้องทั้งหมดใน

Environment Pre production นำไปทำการ Deploy เพื่อขึ้นบน

Environment Production ในเวลา 23.00 – 03.00 น.

6. หลังจากการการ Deploy เสร็จเรียบร้อยแล้ว จากนั้น Team Tester จะเริ่มทำการทดสอบครั้งสุดท้าย (Post test) ในเวลา 03.00-06.00 น. และจะเปิดให้ Agent ทำการทดสอบพร้อมกันในเวลา 06.00 น. หากมีข้อผิดพลาดเกิดขึ้น จะรายงานปัญหาหรือข้อผิดพลาดทั้งหมดไปแจ้งให้ Business Analyst (BA) ทราบ เพื่อทำการวิเคราะห์ปัญหา (Investigate) เพื่อหาสาเหตุของปัญหา (Root Cause) และจัดการปัญหา โดย Developer Leader จะทำการแบ่งงานให้ Developer ในทีมแต่ละคนแก้ไขข้อผิดพลาดจากการทำงานของโปรแกรม ดังข้อที่ 3.3
7. เมื่อทำการแก้ไขปัญหาต่างๆที่พบแล้ว หรือมี Feature ใหม่ที่ระบบสามารถทำงานได้ ก็จะมีการขึ้น Production ในรอบถัดไป โดยก่อนขึ้น Production จะทำการทดสอบเช่นเดิม ตั้งแต่ข้อ 3

### 3.3 การออกแบบกรณีทดสอบตามกรณีทดสอบที่ได้รับ

ในด้านการออกแบบกรณีทดสอบนั้น จะพูดถึงการออกแบบกรณีทดสอบสำหรับการทดสอบด้วยมือ (Manual Testing) ซึ่งกรณีทดสอบที่เขียนขึ้นมานั้นจะถูกเขียนออกมาในรูปแบบของเอกสารหรือที่เรียกว่า Test case โดย Test case เป็นเอกสารที่ระบุชุดข้อมูลป้อนเข้าผลลัพธ์ ที่คาดว่าจะได้รับและกลุ่มของเงื่อนไขในการดำเนินการในชุดทดสอบต่าง ๆ โดยการเขียนผ่านโปรแกรม Microsoft Excel โดยการเขียน Test case จะต้องมียูสเซอร์แบบการเขียนโดยใช้รูปแบบเอกสารดังนี้

#### รูปแบบการเขียน Test case

แบบฟอร์มนี้จะมีชื่อไฟล์คือ Testcase\_<ชื่อ Feature>\_<ชื่อ Feature>.xlsx และเป็นแบบฟอร์มที่ใช้ในการเขียน Test Case ในกรณีต่างๆว่า ผู้ทดสอบระบบต้องการที่จะทดสอบระบบในกรณีใดบ้าง โดยกรณีต่างๆที่ใช้ในการทดสอบระบบต้องคิดขึ้นเอง ซึ่งจะยึดข้อมูลตาม Project Proposal หรือที่เรียกว่า Program Specification ซึ่งประกอบด้วย Tab 2 Tab คือ Tab แบบฟอร์ม และ Tab รูปภาพ โดยในแบบฟอร์มจะมีรายละเอียดหัวข้อต่างๆ ดังนี้

- Test Case ID: ชื่อของหัวข้อที่ต้องการจะทดสอบระบบ
- Test Case: หัวข้อที่ต้องการจะทดสอบระบบ ซึ่งอาจจะอธิบายรายละเอียดของส่วนนั้นๆ ว่าอยู่ที่ตำแหน่งใดก็ได้
- Test Condition ID: ชื่อของหัวข้อย่อย หรือเงื่อนไขย่อยที่ใช้ทดสอบระบบ
- Test Condition: จะประกอบไปด้วย เงื่อนไขข้อมูลที่ใช้ในการทดสอบระบบของแต่ละเงื่อนไข ดังนี้
  - Network Type: รูปแบบของการชำระเงิน เช่น Prepaid หรือ Postpaid
  - Charge Type: ประเภทเครือข่าย เช่น 2G, 3G หรือ 3G BOS
  - Condition: หัวข้อย่อย หรือเงื่อนไขย่อยที่ใช้ทดสอบระบบ
  - Test Data: ข้อมูลที่ใช้ในการทดสอบ ประกอบไปด้วย
  - Mobile Number: เบอร์โทรศัพท์
  - Expected Result: ผลลัพธ์ที่คาดหวังในการทดสอบ ตาม Project Proposal หรือที่เรียกว่า Program Specification
  - Actual Result: ผลลัพธ์ที่เกิดขึ้นจริงตามที่ได้ทดสอบระบบ อาจจะเป็นรูปภาพประกอบก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



สำหรับการออกแบบกรณีทดสอบ (Test case) เพื่อทำการทดสอบระบบระบบบูรณาการข้อมูลสำหรับศูนย์บริการข้อมูลลูกค้า จะออกแบบว่าจะต้องทดสอบในส่วนใดบ้างจึงจะถูกต้องและครอบคลุมตามกรณี ทดสอบแต่ละกรณี ซึ่งในขั้นตอนนี้จะเป็นการออกแบบ scenario ที่ใช้สำหรับการแสดงผลของระบบใน หน้าต่างๆ โดยจะแบ่งหน้าจอที่ใช้ในการทดสอบออกทั้งหมด 5 หน้าจอหลักๆ ดังนี้

- 1) Home
- 2) Customer information
- 3) Package consultation
- 4) Package management
- 5) Service completion

เนื่องจากระบบบูรณาการข้อมูลสำหรับศูนย์บริการข้อมูลลูกค้า เป็นระบบที่มีความซับซ้อนเพราะมีการรวบรวมระบบภายนอกที่มากกว่า 10 ระบบมาไว้เพียงระบบเดียว ทำให้การทดสอบแต่ละครั้ง จะต้องมีการกำหนด Test case แต่ละส่วนให้อยู่บนพื้นฐานของ Business requirement และวัตถุประสงค์ของระบบ ซึ่งจะมีการเขียน Test case โดยแบ่งฟังก์ชันการทำงานของซอฟต์แวร์ออกเป็นพีเจอรีย่อย ๆ แทนการเขียนเพื่อทดสอบในแต่ละหน้า โดยพีเจอร์ทันทั้งหมัดที่นำมาเขียนเป็น Test case เพื่อใช้ในการทดสอบระบบแบบ Automated Testing จะมีดังนี้

- 1) ตรวจสอบการแสดงผลข้อมูลพื้นฐานของลูกค้าในหน้า Customer information
- 2) ตรวจสอบการแสดงผลของชื่อ-นามสกุล ชื่อลูกค้า เมื่อทำการค้นหาเบอร์โทรศัพท์ในหน้า SSCRN HOME
- 3) ตรวจสอบแสดงผลเมื่อทำการค้นหา Main Package ในหน้า Customer information
- 4) ตรวจสอบการออกออเดอร์ว่าสามารถทำงานได้อย่างถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.1 การออกแบบกรณีทดสอบสำหรับพีเจอร์ ระบบ SSCRN สามารถแสดงข้อมูลพื้นฐานของลูกค้าในหน้า Customer information ได้อย่างถูกต้อง

ในการออกแบบการทดสอบเพื่อพัฒนาซอฟต์แวร์ทดสอบอัตโนมัตินี้ จะออกแบบกรณีทดสอบจริงที่ผู้ทดสอบใช้ในการทดสอบการทำงานของระบบ SSCRN ได้เสมือนกับมนุษย์เป็นผู้ทดสอบ โดยการออกแบบกรณีทดสอบที่ตีควรรใช้สำหรับทดสอบฟังก์ชันการทำงานต่างๆ โดยเฉพาะ ในขั้นตอนต่อไปนี้จะออกแบบกรณีทดสอบของการออกแบบการทดสอบสำหรับพีเจอร์ ระบบ SSCRN สามารถแสดงข้อมูลลูกค้าได้อย่างถูกต้องด้วยเครื่องมือ Selenium IDE โดยพีเจอร์นี้จะออกแบบไปที่การทดสอบทั้งหมด 2 หน้าดังนี้

- 1) หน้า Home สามารถเรียกข้อมูลลูกค้าได้อย่างถูกต้อง
- 2) หน้า Customer information สามารถแสดงข้อมูลพื้นฐานของลูกค้าได้อย่างถูกต้อง

ซึ่งจะทำการทดสอบความถูกต้องของข้อมูลลูกค้าในแต่ละ Segment ว่าแสดงถูกต้องหรือไม่ โดยสามารถเขียนกรณีทดสอบได้ดังตารางที่

ตารางที่ 3.2 กรณีทดสอบของพีเจอร์ ระบบ SSCRN สามารถแสดงข้อมูลพื้นฐานของลูกค้าในหน้า Customer information ได้อย่างถูกต้อง

ลำดับที่	สิ่งที่ต้องทำ	ผลลัพธ์ที่ถูกต้อง
1	เข้าสู่ระบบ SSCRN	หน้า SSCRN Home แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● Version ปัจจุบันของระบบ SSCRN</li> <li>● User ที่เข้าใช้งานแสดงถูกต้อง</li> <li>● แสดงช่องสำหรับกรอกเบอร์โทรศัพท์</li> <li>● แสดงปุ่มสำหรับกดค้นหา</li> <li>● แสดงเมนูสำหรับเข้าใช้งานครบทั้ง 3 เมนู ได้แก่ Promotion Service, IR Service , Dispute Service</li> </ul>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2	กรอกเบอร์โทรศัพท์ ประเภทรายเดือน แบบ Bos แล้วกดค้นหา	หน้า SSCRN Home แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>• ชื่อ-นามสกุล ของลูกค้าแสดงถูกต้อง</li> <li>• เบอร์โทรศัพท์แสดงถูกต้อง</li> </ul>
3	เข้าสู่เมนู Promotion Service	หน้า Customer information แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>• Service Year</li> <li>• Mobile Status</li> <li>• Mobile Segment</li> <li>• Migrate To BOSCustomer Type</li> </ul>
4	กดวางสาย	หน้า SSCRN Home แสดงผล
	กรอกเบอร์โทรศัพท์ ประเภทรายเดือน แบบ Non Bos แล้วกดค้นหา	หน้า SSCRN Home แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>• ชื่อ-นามสกุล ของลูกค้าแสดงถูกต้อง</li> <li>• เบอร์โทรศัพท์แสดงถูกต้อง</li> </ul>
5	เข้าสู่เมนู Promotion Service	หน้า Customer information แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>• Service Year</li> <li>• Mobile Status</li> <li>• Mobile Segment</li> <li>• Migrate To BOSCustomer Type</li> </ul>
6	กดวางสาย	หน้า SSCRN Home แสดงผล
7	กรอกเบอร์โทรศัพท์ ประเภทเติมเงิน แบบ Bos แล้วกดค้นหา	หน้า SSCRN Home แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>• ชื่อ-นามสกุล ของลูกค้าแสดงถูกต้อง</li> <li>• เบอร์โทรศัพท์แสดงถูกต้อง</li> </ul>
8	เข้าสู่เมนู Promotion Service	หน้า Customer information แสดงผล สิ่งที่ต้องตรวจสอบได้แก่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

		<ul style="list-style-type: none"> <li>● Service Year</li> <li>● Mobile Status</li> <li>● Mobile Segment</li> <li>● Migrate To BOSCustomer Type</li> </ul>
9	กดวงสาย	หน้า SSCRN Home แสดงผล
10	กรอกเบอร์โทรศัพท์ ประเภทเติมเงิน แบบ Non Bos แล้วกดค้นหา	หน้า SSCRN Home แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● ชื่อ-นามสกุล ของลูกค้าแสดงถูกต้อง</li> <li>● เบอร์โทรศัพท์แสดงถูกต้อง</li> </ul>
11	เข้าสู่เมนู Promotion Service	หน้า Customer information แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● Service Year</li> <li>● Mobile Status</li> <li>● Mobile Segment</li> <li>● Migrate To BOSCustomer Type</li> </ul>
12	กดวงสาย	หน้า SSCRN Home แสดงผล
13	กรอกเบอร์โทรศัพท์ ประเภทเติมเงิน แบบ INS แล้วกดค้นหา	หน้า SSCRN Home แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● ชื่อ-นามสกุล ของลูกค้าแสดงถูกต้อง</li> <li>● เบอร์โทรศัพท์แสดงถูกต้อง</li> </ul>
14	เข้าสู่เมนู Promotion Service	หน้า Customer information แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● Service Year</li> <li>● Mobile Status</li> <li>● Mobile Segment</li> <li>● Migrate To BOSCustomer Type</li> </ul>
15	กดวงสาย	หน้า SSCRN Home แสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากกรณีทดสอบดังกล่าว จะสามารถแบ่งเป็น Scenario ต่างๆ ได้ดังตารางที่ 3.3

**ตารางที่ 3.3** Scenario ที่ใช้สำหรับการแสดงผลของ Feature ระบบ SSCRN สามารถแสดงข้อมูลลูกค้าได้อย่างถูกต้อง

ลำดับ Scenario	สิ่งที่จะทดสอบ
1	หน้า SSCRN Home สามารถเรียกข้อมูลลูกค้าได้อย่างถูกต้อง (Postpaid Bos)
2	หน้า Customer information สามารถแสดงลูกค้าได้อย่างถูกต้อง (Postpaid Bos)
3	หน้า SSCRN Home สามารถเรียกข้อมูลลูกค้าได้อย่างถูกต้อง (Postpaid Non Bos)
4	หน้า Customer information สามารถแสดงลูกค้าได้อย่างถูกต้อง (Postpaid Non Bos)
5	หน้า SSCRN Home สามารถเรียกข้อมูลลูกค้าได้อย่างถูกต้อง (Prepaid Bos)
6	หน้า Customer information สามารถแสดงลูกค้าได้อย่างถูกต้อง (Prepaid Bos)
7	หน้า SSCRN Home สามารถเรียกข้อมูลลูกค้าได้อย่างถูกต้อง (Prepaid Non Bos)
8	หน้า Customer information สามารถแสดงลูกค้าได้อย่างถูกต้อง (Prepaid Non Bos)
9	หน้า SSCRN Home สามารถเรียกข้อมูลลูกค้าได้อย่างถูกต้อง (Prepaid INS)
10	หน้า Customer information สามารถแสดงลูกค้าได้อย่างถูกต้อง (Prepaid INS)

**3.3.2 การออกแบบกรณีทดสอบสำหรับพีเจเจอร์ ระบบ SSCRN สามารถแสดงชื่อ-นามสกุล เมื่อทำการค้นหาเบอร์โทรศัพท์บนหน้า Home ได้อย่างถูกต้อง**

ในการออกแบบการทดสอบเพื่อพัฒนาซอฟต์แวร์ทดสอบอัตโนมัตินี้ จะออกแบบกรณีทดสอบ โดยระบบ SSCRN สามารถแสดงข้อมูลชื่อลูกค้า และเบอร์โทรศัพท์ลูกค้า เมื่อทำการค้นหาเบอร์โทรศัพท์ในหน้า Home ได้อย่างถูกต้องได้อย่างถูกต้อง ด้วยเครื่องมือ Robot Framework โดยพีเจเจอร์นี้จะออกแบบไปที่การทดสอบทั้งหมด 1 หน้าดังนี้

1) หน้า Home สามารถแสดงชื่อ-นามสกุล และเบอร์โทรศัพท์ลูกค้าได้อย่างถูกต้อง

ซึ่งจะทำการทดสอบความถูกต้องของชื่อ-นามสกุล ลูกค้าในแต่ละ Segment ว่าแสดงถูกต้องหรือไม่ โดยทำการตรวจสอบหลังจากการกดค้นหาเบอร์ไปแล้ว ซึ่งสามารถเขียนกรณีทดสอบได้ดังตารางที่ 3.3

ตารางที่ 3.4 กรณีทดสอบของพีเจอร์ ระบบ SSCRN สามารถแสดงชื่อลูกค้า เมื่อทำการค้นหาเบอร์โทรศัพท์บนหน้า Home ได้อย่างถูกต้อง

ลำดับที่	สิ่งที่ต้องทำ	ผลลัพธ์ที่ถูกต้อง
1	เข้าสู่ระบบ SSCRN	หน้า SSCRN Home แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● Version ปัจจุบันของระบบ SSCRN</li> <li>● User ที่เข้าใช้งานแสดงถูกต้อง</li> <li>● แสดงช่องสำหรับกรอกเบอร์โทรศัพท์</li> <li>● แสดงปุ่มสำหรับกดค้นหา</li> <li>● แสดงเมนูสำหรับเข้าใช้งานครบทั้ง 3 เมนู ได้แก่ Promotion Service, IR Service , Dispute Service</li> </ul>
2	กรอกเบอร์โทรศัพท์ประเภทรายเดือน แบบ Bos แล้วกดค้นหา	หน้า SSCRN Home แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● ชื่อ-นามสกุล ของลูกค้าแสดงถูกต้อง</li> <li>● เบอร์โทรศัพท์แสดงถูกต้อง</li> </ul>
3	กดวางสาย	เบอร์ และชื่อที่แสดงบนหน้า SSCRN Home หายไป
4	กรอกเบอร์โทรศัพท์ประเภทรายเดือน แบบ Non Bos แล้วกดค้นหา	หน้า SSCRN Home แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● ชื่อ-นามสกุล ของลูกค้าแสดงถูกต้อง</li> <li>● เบอร์โทรศัพท์แสดงถูกต้อง</li> </ul>
5	กดวางสาย	เบอร์ และชื่อที่แสดงบนหน้า SSCRN Home หายไป
6	กรอกเบอร์โทรศัพท์ประเภทเติมเงินแบบ Bos แล้วกดค้นหา	หน้า SSCRN Home แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● ชื่อ-นามสกุล ของลูกค้าแสดงถูกต้อง</li> <li>● เบอร์โทรศัพท์แสดงถูกต้อง</li> </ul>
7	กดวางสาย	เบอร์ และชื่อที่แสดงบนหน้า SSCRN Home หายไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8	กรอกเบอร์โทรศัพท์ ประเภทเติมเงิน แบบ NonBos แล้วกดค้นหา	หน้า SSCRN Home แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>• ชื่อ-นามสกุล ของลูกค้าแสดงถูกต้อง</li> <li>• เบอร์โทรศัพท์แสดงถูกต้อง</li> </ul>
9	กดวางสาย	เบอร์ และชื่อที่แสดงบนหน้า SSCRN Home หายไป
10	กรอกเบอร์โทรศัพท์ ประเภทเติมเงิน แบบ INS แล้วกดค้นหา	หน้า SSCRN Home แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>• ชื่อ-นามสกุล ของลูกค้าแสดงถูกต้อง</li> <li>• เบอร์โทรศัพท์แสดงถูกต้อง</li> </ul>
11	กดวางสาย	เบอร์ และชื่อที่แสดงบนหน้า SSCRN Home หายไป
12	ออกจากระบบ SSCRN	หน้าต่างระบบ SSCRN ถูกปิดอัตโนมัติ

จากกรณีทดสอบดังกล่าว จะสามารถแบ่งเป็น Scenario ต่างๆ ได้ดังตารางที่ 3.5

ตารางที่ 3.5 Scenario ที่ใช้สำหรับการแสดงผลของ Feature ระบบ SSCRN สามารถแสดงข้อมูลลูกค้าได้อย่างถูกต้อง

ลำดับ Scenario	สิ่งที่จะทดสอบ
1	หน้า SSCRN Home สามารถแสดงชื่อ-นามสกุลลูกค้าได้อย่างถูกต้อง
2	หน้า SSCRN Home สามารถแสดงชื่อ-นามสกุลลูกค้าได้อย่างถูกต้อง (Postpaid Non Bos)
3	หน้า SSCRN Home สามารถแสดงชื่อ-นามสกุลลูกค้าได้อย่างถูกต้อง (Postpaid Bos)
4	หน้า SSCRN Home สามารถแสดงชื่อ-นามสกุลลูกค้าได้อย่างถูกต้อง (Postpaid Non Bos)
5	หน้า SSCRN Home สามารถแสดงชื่อ-นามสกุลลูกค้าได้อย่างถูกต้อง (Postpaid INS)

3.3.3 การออกแบบกรณีทดสอบสำหรับพีเจอร์รี่ ระบบ SSCRN สามารถแสดง Package ได้อย่างถูกต้องเมื่อทำการค้นหา Main Package ในหน้า Customer information

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการออกแบบการทดสอบเพื่อพัฒนาซอฟต์แวร์ทดสอบอัตโนมัตินี้ จะออกแบบกรณีทดสอบ โดยระบบ SSCRN สามารถแสดงข้อมูล Main Package ในหน้า Customer information ได้อย่างถูกต้อง เมื่อทำการค้นหาด้วยเงื่อนไขที่กำหนด ด้วยเครื่องมือ Robot Framework โดยพีเจอร์นี่จะออกแบบไปที่การทดสอบทั้งหมด 2 หน้าดังนี้

- 1) หน้า Main สามารถเรียกข้อมูลลูกค้าได้อย่างถูกต้อง
- 2) หน้า Customer information สามารถแสดงข้อมูล และ Mobile Staus ของลูกค้าได้อย่างถูกต้อง
- 3) หน้า Customer information หลังจากกดปุ่ม Change Main เมื่อทำการค้นหา Main Package ด้วยเงื่อนไขต่างๆ ทั้งการค้นหาด้วยการระบุข้อมูล Package ที่ถูกต้อง และการค้นหาด้วยการระบุข้อมูล Package ที่ไม่ถูกต้อง แล้วระบบ SSCRN สามารถแสดง Package หรือข้อความแจ้งเตือนได้อย่างถูกต้อง

การทดสอบในพีเจอร์นี่จะเป็นการตรวจสอบความถูกต้องของการแสดงผล Package ที่ทำการค้นหา เฉพาะเบอร์โทรศัพท์ที่เป็นรายเดือน ประเภท Bos เท่านั้น ว่าแสดงถูกต้องหรือไม่ โดยสามารถเขียนกรณีทดสอบได้ดังตารางที่ 3.5

ตารางที่ 3.6 กรณีทดสอบของพีเจอาร์ ระบบ SSCRN สามารถแสดง Package ได้อย่างถูกต้องเมื่อทำการค้นหา Main Package ในหน้า Customer information

ลำดับที่	สิ่งที่ต้องทำ	ผลลัพธ์ที่ถูกต้อง
1	เข้าสู่ระบบ SSCRN	หน้า SSCRN Home แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● Version ปัจจุบันของระบบ SSCRN</li> <li>● User ที่เข้าใช้งานแสดงถูกต้อง</li> <li>● แสดงช่องสำหรับกรอกเบอร์โทรศัพท์</li> <li>● แสดงปุ่มสำหรับกดค้นหา</li> <li>● แสดงเมนูสำหรับเข้าใช้งานครบทั้ง 3 เมนู ได้แก่ Promotion Service, IR Service , Dispute Service</li> </ul>
2	กรอกเบอร์โทรศัพท์ ประเภทรายเดือน แบบ Bos แล้วกดค้นหา	หน้า SSCRN Home แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● ชื่อ-นามสกุล ของลูกค้าแสดงถูกต้อง</li> <li>● เบอร์โทรศัพท์แสดงถูกต้อง</li> </ul>
3	เข้าสู่เมนู Promotion Service	หน้า Customer information แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● Customer name</li> <li>● Mobile Status</li> <li>● Migrate To BOSCustomer Type</li> </ul>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4	กดปุ่ม Change Main Promotion บน Tab Existing Package	<p>เมนู Search Main Package แสดงผล สิ่งที่ต้องตรวจสอบได้แก่</p> <ul style="list-style-type: none"> <li>● การนำเสนอ Package จาก iCRM</li> <li>● ช่องสำหรับระบุ Main Package</li> <li>● ปุ่มกดสำหรับค้นหา Main Package</li> </ul>
5	ค้นหา Main Package ด้วยการไม่ระบุข้อมูลลงในช่องค้นหา จากนั้นทำการกดปุ่มค้นหา	<p>ระบบแสดงผลการค้นหา Main Package โดยมีสิ่งที่ต้องตรวจสอบได้แก่</p> <ul style="list-style-type: none"> <li>● ข้อความแจ้งเตือน ‘Please enter your search conditions’.</li> </ul>
6	ค้นหา Main Package ด้วยการระบุ Package Name ที่ถูกต้อง ลงในช่องค้นหา จากนั้นทำการกดปุ่มค้นหา	<p>ระบบแสดงผลการค้นหา Main Package โดยมีสิ่งที่ต้องตรวจสอบได้แก่</p> <ul style="list-style-type: none"> <li>● ชื่อ Main Package ที่ทำการค้นหาแสดงถูกต้อง</li> <li>● รายละเอียด Main Package ที่ทำการค้นหาแสดงถูกต้อง</li> </ul>
7	ค้นหา Main Package ด้วยการระบุ Keyword ที่ถูกต้อง ลงในช่องค้นหา จากนั้นทำการกดปุ่มค้นหา	<p>ระบบแสดงผลการค้นหา Main Package โดยมีสิ่งที่ต้องตรวจสอบได้แก่</p> <ul style="list-style-type: none"> <li>● Package ที่แสดงผล ต้องมี Keyword ที่ระบุแสดงอยู่ด้วย</li> </ul>
8	ค้นหา Main Package ด้วยการระบุ Package Description ที่ถูกต้อง ลงในช่องค้นหา จากนั้นทำการกดปุ่มค้นหา	<p>ระบบแสดงผลการค้นหา Main Package โดยมีสิ่งที่ต้องตรวจสอบได้แก่</p> <ul style="list-style-type: none"> <li>● Package ที่แสดงผล ต้องมีรายละเอียด Package ตรงกับที่ทำการค้นหา</li> </ul>
9	ค้นหา Main Package ด้วยการระบุ Package Code ที่ถูกต้อง ลงในช่องค้นหา จากนั้นทำการกดปุ่มค้นหา	<p>ระบบแสดงผลการค้นหา Main Package โดยมีสิ่งที่ต้องตรวจสอบได้แก่</p> <ul style="list-style-type: none"> <li>● Package ที่แสดงผลจะต้องมี Package code ตรงกับที่ระบุ</li> </ul>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10	ค้นหา Main Package ด้วยการระบุ Package Name ที่ไม่ถูกต้องลงในช่องค้นหา จากนั้นทำการกดปุ่มค้นหา	ระบบแสดงผลการค้นหา Main Package โดยมีสิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● แสดงข้อความ 'ไม่พบข้อมูล'</li> </ul>
11	ค้นหา Main Package ด้วยการระบุ Keyword ที่ไม่ถูกต้องลงในช่องค้นหา จากนั้นทำการกดปุ่มค้นหา	ระบบแสดงผลการค้นหา Main Package โดยมีสิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● แสดงข้อความ 'ไม่พบข้อมูล'</li> </ul>
12	ค้นหา Main Package ด้วยการระบุ Package Description ที่ไม่ถูกต้องลงในช่องค้นหา จากนั้นทำการกดปุ่มค้นหา	ระบบแสดงผลการค้นหา Main Package โดยมีสิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● แสดงข้อความ 'ไม่พบข้อมูล'</li> </ul>
13	ค้นหา Main Package ด้วยการระบุ Package Code ที่ไม่ถูกต้องลงในช่องค้นหา จากนั้นทำการกดปุ่มค้นหา	ระบบแสดงผลการค้นหา Main Package โดยมีสิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● แสดงข้อความ 'ไม่พบข้อมูล'</li> </ul>
14	ออกจากการค้นหา Main Promotion	หน้า Customer information แสดงผล
15	กดวางสาย	หน้า SSCRN Home แสดงผล
16	กดปุ่ม Close Browser	หน้าต่างระบบ SSCRN ถูกปิด

จากกรณีทดสอบดังกล่าว จะสามารถแบ่งเป็น Scenario ต่างๆ ได้ดังตารางที่ 3.7

ตารางที่ 3.7 Scenario ที่ใช้สำหรับการแสดงผลของ Feature ระบบ SSCRN สามารถแสดง Package ได้อย่างถูกต้องเมื่อทำการค้นหา Main Package ในหน้า Customer information

ลำดับ Scenario	สิ่งที่จะทดสอบ
1	ตรวจสอบการแสดงผล Customer Name บนหน้า SSCRN Home
2	ตรวจสอบการแสดงผลข้อมูลพื้นฐานลูกค้า บนหน้า Customer information
3	ค้นหาแพคเกจ ด้วยการไม่ระบุข้อความในช่อง แล้วค้นหา )Empty Main Promotion(
4	ค้นหาแพคเกจ Main Promotion ด้วย Package Name

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5	ค้นหาแพคเกจ Main Promotion ด้วย Keyword Package
6	ค้นหาแพคเกจ Main Promotion ด้วย Package Description
7	ค้นหาแพคเกจ Main Promotion ด้วย Package Code
8	ค้นหาแพคเกจ Main Promotion ด้วย Invalid Package Name
9	ค้นหาแพคเกจ Main Promotion ด้วย Invalid Keyword
10	ค้นหาแพคเกจ Main Promotion ด้วย Invalid Package Description
11	ค้นหาแพคเกจ Main Promotion ด้วย Invalid Package Code
12	วางสาย และกดปุ่ม Close Browser

### 3.3.4 การออกแบบกรณีทดสอบสำหรับพีเจอร์ ระบบ SSCRN สามารถออก Order (ซื้อแพคเกจ หรือ ยกเลิกแพคเกจ) ได้อย่างถูกต้อง

ในการออกแบบการทดสอบเพื่อพัฒนาซอฟต์แวร์ทดสอบอัตโนมัตินี้ จะออกแบบกรณีทดสอบ ด้วยเครื่องมือ Robot Framework ซึ่งสำหรับพีเจอร์นี้จะเริ่มทำการทดสอบตั้งแต่หน้าแรก และไปจบในหน้าสุดท้าย ซึ่งก็คือสามารถทำรายการ เช่นการซื้อแพคเกจ หรือการยกเลิกแพคเกจ ได้อย่างถูกต้อง โดยพีเจอร์นี้จะออกแบบไปที่การทดสอบรวมทั้งหมด 5 หน้า ดังนี้

- 1) SSCRN Home
- 2) Customer information
- 3) Package consultation
- 4) Package management
- 5) Service completion

ซึ่งจะทำการทดสอบในพีเจอร์นี้จะเป็นการตรวจสอบการออกออร์เดอร์ เฉพาะเบอร์โทรศัพท์ที่เป็นรายเดือน ประเภท Bos เท่านั้น ว่าสามารถทำรายการซื้อแพคเกจได้อย่างถูกต้อง โดยสามารถเขียนกรณีทดสอบได้ดังตารางที่ 3.8

ตารางที่ 3.8 กรณีทดสอบของพีเจอร์ ระบบ SSCRN สามารถออก Order (ซื้อแพคเกจ) ได้อย่างถูกต้อง

ลำดับที่	สิ่งที่ต้องทำ	ผลลัพธ์ที่ถูกต้อง
1	เข้าสู่ระบบ SSCRN	หน้า SSCRN Home แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● Version ปัจจุบันของระบบ SSCRN</li> <li>● User ที่เข้าใช้งานแสดงถูกต้อง</li> <li>● แสดงช่องสำหรับกรอกเบอร์โทรศัพท์</li> <li>● แสดงปุ่มสำหรับกดค้นหา</li> <li>● แสดงเมนูสำหรับเข้าใช้งานครบทั้ง 3 เมนู ได้แก่ Promotion Service, IR Service , Dispute Service</li> </ul>
2	กรอกเบอร์โทรศัพท์ ประเภทรายเดือน แบบ Bos แล้วกดค้นหา	หน้า SSCRN Home แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● ชื่อ-นามสกุล ของลูกค้าแสดงถูกต้อง</li> <li>● เบอร์โทรศัพท์แสดงถูกต้อง</li> </ul>
3	เข้าสู่เมนู Promotion Service	หน้า Customer information แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● Customer name</li> <li>● Mobile Status</li> <li>● Migrate To BOSCustomer Type</li> </ul>
4	เลือก Tab Package consultation เพื่อค้นหา Package	หน้า Package consultation แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● การนำเสนอ Package จาก iCRM</li> <li>● ช่องสำหรับระบุ Package</li> <li>● ปุ่มกดสำหรับค้นหา Package</li> </ul>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5	ค้นหา Package ด้วยการระบุ Package Name ลงในช่องค้นหา จากนั้นทำการกดปุ่มค้นหา	ระบบแสดงผลการค้นหา Package โดยมีสิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● ชื่อ Package ที่ทำการค้นหาแสดงถูกต้อง</li> <li>● รายละเอียด Main Package ที่ทำการค้นหาแสดงถูกต้อง</li> </ul>
6	กดปุ่ม Read more ที่ Package	ระบบแสดงผลการค้นหา Package โดยมีสิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● รายละเอียดของ Package</li> <li>● ปุ่ม Send SMS</li> <li>● ปุ่ม Add Package สีเขียว</li> <li>● ปุ่ม Service Interact</li> <li>● Effective <ul style="list-style-type: none"> <li>- Immediate</li> <li>- Next Day</li> <li>- Next Bill</li> </ul> </li> <li>● ค่าใช้จ่าย )บาท(</li> <li>● Voice(นาที)</li> <li>● Data(นาที)</li> <li>● Data</li> <li>● รวมทั้งหมด แสดงด้วยตัวเลขสีแดง</li> </ul>
7	เลือก Effective	ปุ่ม Radio button ถูกเลือก
8	กดปุ่ม Add Package	Package ที่เลือกเข้าสู่ตระกร้า แสดงเป็นเลข 1
9	เลือก Tab Package management เพื่อทำการตรวจสอบ Package ที่เลือก	หน้า Package consultation แสดงผล สิ่งที่ต้องตรวจสอบได้แก่ <ul style="list-style-type: none"> <li>● Package Name</li> <li>● Package Fee</li> <li>● Change Fee</li> <li>● Effective Start Date</li> <li>● Action แสดงเครื่องหมายกากบาทสีแดง</li> <li>● Sum</li> </ul>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

		<ul style="list-style-type: none"> <li>● Total</li> <li>● Description</li> </ul>
10	ใส่ Description เกิน 1,000 ตัวอักษร	ระบบแสดงข้อความแจ้งเตือน ‘Description มีขนาดเกิน 1,000 ตัวอักษร’
11	กดปุ่ม Submit Order	<p>หน้า Service Completion แสดงผล สิ่งที่ต้องตรวจสอบได้แก่</p> <ul style="list-style-type: none"> <li>● แสดงข้อความ ‘Order Complete’</li> <li>● จำนวนหมายเลขในตระกร้าเปลี่ยนเป็น 0</li> </ul>
	เลือก Tab Customer information	<p>หน้า Customer information แสดงผล โดยสิ่งที่ต้องตรวจสอบได้แก่</p> <ul style="list-style-type: none"> <li>● เมนู Existing Package มี Package เพิ่มเข้ามา โดยสิ่งที่ต้องตรวจสอบคือ <ul style="list-style-type: none"> <li>- Package Name</li> <li>- Promotion Class เช่น On Top</li> <li>- Start Date</li> <li>- End Date</li> <li>- Attribute</li> </ul> </li> <li>● เมนู Recent Activity สิ่งที่ต้องตรวจสอบคือ <ul style="list-style-type: none"> <li>- Date/Time</li> <li>- Type</li> <li>- Category</li> <li>- Sub Category</li> <li>- Description</li> <li>- Owner</li> <li>- Status</li> </ul> </li> </ul>
12	กดวางสาย	หน้า SSCRN Home แสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากกรณีทดสอบดังกล่าว จะสามารถแบ่งเป็น Scenario ต่างๆ ได้ดังตารางที่ 3.9

**ตารางที่ 3.9** Scenario ที่ใช้สำหรับการแสดงผลของ Feature ระบบ SSCRN สามารถออก Order (สามารถซื้อแพคเกจ) ได้อย่างถูกต้อง

ลำดับ Scenario	สิ่งที่ทดสอบ
1	ตรวจสอบการแสดงผล Customer Name บนหน้า SSCRN Home
2	ตรวจสอบการแสดงผลข้อมูลพื้นฐานลูกค้า บนหน้า Customer information
3	ค้นหาแพคเกจ ที่หน้า Package Consultation
4	ตรวจสอบ การ Add Package
5	ตรวจสอบแพคเกจที่หน้า Package Management
6	ตรวจสอบ Alert message เมื่อใส่ Description เกิน 1,000 ตัวอักษร
7	ตรวจสอบการยืนยันการทำรายการเมื่อ Order complete
8	ตรวจสอบแพคเกจที่เพิ่มเข้าไปในหน้า Customer information
9	กดวางสาย และปิด Browser

### 3.4 การพัฒนาโปรแกรมทดสอบอัตโนมัติ (Automated Testing)

หลังจากออกแบบสิ่งที่ต้องการจะทดสอบในแต่พีเจอร์แล้ว ซึ่งจะอยู่ในรูปของ scenario ต่างๆ เรียบร้อยแล้ว ในขั้นตอนการพัฒนาโปรแกรมทดสอบอัตโนมัตินี้ จะเป็นการพัฒนาซอร์สโค้ดเพื่อให้โปรแกรมทดสอบอัตโนมัติสามารถทำงานได้ตามที่ได้ออกแบบไว้ในแต่ละ scenario ซึ่งจะเป็นการพัฒนาด้วยเครื่องมือมือ 2 ตัว คือ Selenium IDE และ Robot framework โดยมีรายละเอียดในการพัฒนาดังนี้

- 1) เครื่องมือ Selenium IDE ใช้สำหรับทดสอบพีเจอร์
  - ตรวจสอบการแสดงผลข้อมูลพื้นฐานของลูกค้าในหน้า Customer information
- 2) เครื่องมือ Robot framework ใช้สำหรับทดสอบพีเจอร์
  - ตรวจสอบการแสดงผลของชื่อลูกค้า เมื่อทำการค้นหาเบอร์โทรศัพท์ในหน้า Home
  - ตรวจสอบแสดงผลเมื่อทำการค้นหา Main Package ในหน้า Customer information
  - ตรวจสอบการออกออร์เดอร์ว่าสามารถทำงานได้อย่างถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.1 การพัฒนาโปรแกรมเพื่อทดสอบพีเจอร์ ระบบ SSCRN สามารถแสดงข้อมูลพื้นฐานของลูกค้าในหน้า Customer information ได้อย่างถูกต้อง ด้วยเครื่องมือ Selenium IDE

จากตารางที่ 3.2 ตาราง Scenario จะเห็นว่าสามารถแบ่งกลุ่มของลูกค้าได้ 5 ประเภท คือ

- Postpaid Bos
- Postpaid Non Bos
- Prepaid Bos
- Prepaid Non Bos
- Prepaid INS

ดังนั้น เพื่อให้สะดวกในการจัดเก็บไฟล์ในการทดสอบ รายงาน และภาพจากการจับหน้าจอ จะทำการสร้าง Folder ไว้ดังนี้

Post Bos	20/01/2560 20:00	File folder
PostNon	20/01/2560 20:01	File folder
PreBos	07/12/2559 13:30	File folder
PreINS	07/12/2559 13:30	File folder
PreNon	11/12/2559 18:19	File folder

รูปที่ 3.3 สร้าง Folder ที่ใช้ในการทดสอบในโฟลเดอร์หลัก

จากนั้นทำการสร้างโฟลเดอร์เพื่อใช้สำหรับเก็บรายงาน ไว้ในแต่ละโฟลเดอร์ดังนี้

Name	Date modified	Type
report	16/12/2559 10:12	File folder
script	16/12/2559 10:59	File folder
snapshot	18/12/2559 14:50	File folder

รูปที่ 3.4 สร้างโฟลเดอร์ Report ที่ใช้ในโฟลเดอร์ PostBos

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Name	Date modified	Type
report	16/12/2559 10:12	File folder
script	16/12/2559 10:59	File folder
snapshot	18/12/2559 14:50	File folder

รูปที่ 3.5 สร้างโฟลเดอร์ Script ที่ใช้ในโฟลเดอร์ PostBos

Name	Date modified	Type
report	16/12/2559 10:12	File folder
script	16/12/2559 10:59	File folder
snapshot	18/12/2559 14:50	File folder

รูปที่ 3.6 สร้างโฟลเดอร์ Snapshot ที่ใช้ในโฟลเดอร์ PostBos

Name	Date modified	Type
report	16/12/2559 10:12	File folder
script	16/12/2559 10:59	File folder
snapshot	18/12/2559 14:50	File folder

รูปที่ 3.7 สร้างโฟลเดอร์ Report ที่ใช้ในโฟลเดอร์ PostNon

Name	Date modified	Type
report	16/12/2559 10:12	File folder
script	16/12/2559 10:59	File folder
snapshot	18/12/2559 14:50	File folder

รูปที่ 3.8 สร้างโฟลเดอร์ Script ที่ใช้ในโฟลเดอร์ PostNon

Name	Date modified	Type
report	16/12/2559 10:12	File folder
script	16/12/2559 10:59	File folder
snapshot	18/12/2559 14:50	File folder

รูปที่ 3.9 สร้างโฟลเดอร์ Snapshot ที่ใช้ในโฟลเดอร์ PostNon

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Name	Date modified	Type
report	16/12/2559 10:12	File folder
script	16/12/2559 10:59	File folder
snapshot	18/12/2559 14:50	File folder

รูปที่ 3.10 สร้างโฟลเดอร์ Report ที่ใช้ในโฟลเดอร์ PreBos

Name	Date modified	Type
report	16/12/2559 10:12	File folder
script	16/12/2559 10:59	File folder
snapshot	18/12/2559 14:50	File folder

รูปที่ 3.11 สร้างโฟลเดอร์ Script ที่ใช้ในโฟลเดอร์ PreBos

Name	Date modified	Type
report	16/12/2559 10:12	File folder
script	16/12/2559 10:59	File folder
snapshot	18/12/2559 14:50	File folder

รูปที่ 3.12 สร้างโฟลเดอร์ Snapshot ที่ใช้ในโฟลเดอร์ PreBos

Name	Date modified	Type
report	16/12/2559 10:12	File folder
script	16/12/2559 10:59	File folder
snapshot	18/12/2559 14:50	File folder

รูปที่ 3.13 สร้างโฟลเดอร์ Report ที่ใช้ในโฟลเดอร์ PreNon

Name	Date modified	Type
report	16/12/2559 10:12	File folder
script	16/12/2559 10:59	File folder
snapshot	18/12/2559 14:50	File folder

รูปที่ 3.14 สร้างโฟลเดอร์ Script ที่ใช้ในโฟลเดอร์ PreNon

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Name	Date modified	Type
report	16/12/2559 10:12	File folder
script	16/12/2559 10:59	File folder
snapshot	18/12/2559 14:50	File folder

รูปที่ 3.15 สร้างโฟลเดอร์ Snapshot ที่ใช้ในโฟลเดอร์ PreNon

Name	Date modified	Type
report	16/12/2559 10:12	File folder
script	16/12/2559 10:59	File folder
snapshot	18/12/2559 14:50	File folder

รูปที่ 3.16 สร้างโฟลเดอร์ Report ที่ใช้ในโฟลเดอร์ PreINS

Name	Date modified	Type
report	16/12/2559 10:12	File folder
script	16/12/2559 10:59	File folder
snapshot	18/12/2559 14:50	File folder

รูปที่ 3.17 สร้างโฟลเดอร์ Script ที่ใช้ในโฟลเดอร์ PreINS

Name	Date modified	Type
report	16/12/2559 10:12	File folder
script	16/12/2559 10:59	File folder
snapshot	18/12/2559 14:50	File folder

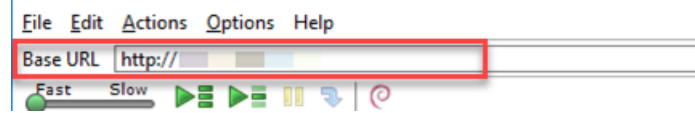
รูปที่ 3.18 สร้างโฟลเดอร์ Snapshot ที่ใช้ในโฟลเดอร์ PreINS

เมื่อทำการสร้าง Folder เรียบร้อยแล้ว จะต้องทำการแบ่ง Test suite ออกเป็น 2 ส่วน คือ Setting ซึ่งใช้สำหรับจัดเก็บ Path หรือตัวแปรที่ใช้ในการทดสอบใน Scenario นี้ และ Verify Customerinfo PostBos ใช้สำหรับทดสอบความถูกต้องในการทำงานโดยส่วนนี้จะมีการเรียกตัวแปรจาก Setting มาใช้งานด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

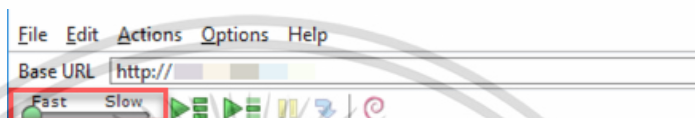
## ตั้งค่าในการทดสอบทั่วไป

เริ่มจากการกำหนด Base URL ที่ใช้สำหรับเข้าทดสอบ Web application



รูปที่ 3.19 การกำหนด Base URL สำหรับเข้าทดสอบระบบ

ตั้งค่าความเร็วในการ Execute



รูปที่ 3.20 การกำหนดความเร็วในการ Execute เป็น Fast

จากนั้นทำการสร้าง Testcase ขึ้นมาใหม่ตั้งชื่อว่า Setting\_PostBos เพื่อใช้สำหรับเก็บค่าตัวแปร หรือเก็บ Path ที่ใช้ในการทดสอบ

### 1) Setting\_PostBos

#### 1.1) ทำการสร้าง Store สำหรับเก็บ Path ดังนี้

Command	store		
Target	D:\Test Script_IDE\CustomerInfo_V2\Postnon\snapshot\	Select	Find
Value	PathPic		

รูปที่ 3.21 การสร้าง Store สำหรับเก็บค่า Path ของรูปภาพที่ทำการจับหน้าจอ

Table	Source
Command	Target
Setting	
1. Set path for save picture	
store	D:\Test Script_IDE\CustomerInfo_V2\Postnon\snapshot\ PathPic

รูปที่ 3.22 การแสดงผลของคำสั่งใน Table

#### 1.2) จากนั้นทำการสร้าง Store สำหรับเก็บค่าต่างๆของตัวแปร เพื่อที่จะสามารถนำไปใช้ใน Testcase อื่นๆ ได้ดังนี้

Command	store		
Target	0812345678	Select	Find
Value	MobileNumber		

รูปที่ 3.23 การสร้าง Store สำหรับเก็บค่าเบอร์โทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ประโยชน์เพื่อการศึกษาดูเท่านั้น เมื่ออนุญาตเห็นไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำการสร้าง Store ทั้งหมดเรียบร้อยแล้ว จะแสดงผลใน Table ดังนี้

2. Set Variable for verify Customer info

store	(Version 2.44.01)	Version
store	Active	MobileStatus
store	08	MobileNumber
store	Ms.	UserName
store	Serenade	MobileSegment
store	3G	NetworkType

รูปที่ 3.24 การแสดงผลของคำสั่งใน Table

2) Verify Customerinfo PostBos

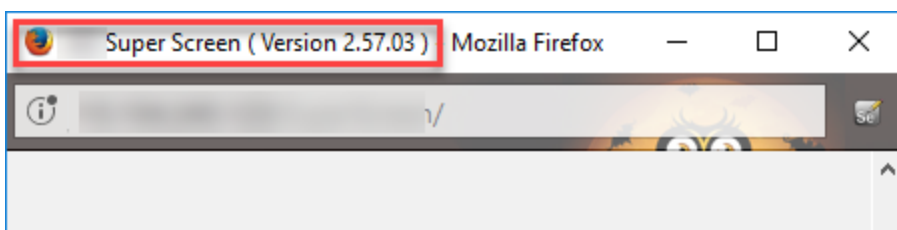
สำหรับขั้นตอนนี้คือการตรวจสอบข้อมูลที่ใช้การทดสอบบนหน้า SSCRN Home ในขั้นตอนนี้จะอธิบายคำสั่งที่ใช้ และแสดง Element ที่ใช้ในการทดสอบแต่ละคำสั่งดังต่อไปนี้

- assertTitle \${Version} เป็นการตรวจสอบเวอร์ชันของระบบที่ทำการ

ทดสอบ หากเวอร์ชันที่ทำการตรวจสอบนี้ไม่ตรงกับ เวอร์ชันใน Store ที่ทำการกำหนดค่าไว้ใน Setting Script จะหยุดทำงานทันที ไม่ทำงานในบรรทัดถัดไปใน Table

Command	Target	Value
Page SSCRN home		
assertTitle	\${Version}	
type	id=searchForm:txtMobileN...	\${MobileNumber}
click	id=btnSearchCust	
waitForText	css=div.profile-info-value ...	\${UserName}

รูปที่ 3.25 แสดงการใช้คำสั่ง assertTitle



รูปที่ 3.26 การแสดงตรวจสอบด้วยคำสั่ง AssertTitle

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Type id=searchForm:txtMobileNumber \${MobileNumber}
- เป็นการลบค่าปัจจุบันออกจากกล่องข้อความ จากนั้นใส่ค่าใหม่เข้าไป คือค่า MobileNumber ที่กำหนดไว้ใน Setting

Command	Target	Value
Page SSCRN home		
assertTitle	\${Version}	
type	id=searchForm:txtMobileN...	\${MobileNumber}
click	id=btnSearchCust	
waitForText	css=div.profile-info-value ...	\${UserName}

รูปที่ 3.27 แสดงการใช้คำสั่ง Type

โดยการเข้าถึง HTML element จะใช้ค่าของ HTML attribute ดังรูปที่ 3.28

```
<div class="profile-info-value" style="margin-left: 25px; width: 80%;">
  <div class="input-group">
    <input id="searchForm:txtMobileNumber" class="form-control mobileInputTextbox"
      name="searchForm:txtMobileNumber" maxlength="10" onkeypress="return
      checkCharacterInputForMobileNumber();" style="background-color: rgb(255, 255, 255);
    >
    <span id="btnSearchCust" class="input-group-addon" style="cursor: pointer;"></span>
    <input id="searchForm:j_idt41" class="jsfBtnSearchCustomer" name="searchForm:j_idt41">
```

รูปที่ 3.28 ค่าของ HTML attribute ที่ใช้คำสั่ง Type

- Click id=btnSearchCust หลังจากใส่ค่าใหม่เรียบร้อยแล้ว ทำการสั่งให้กดปุ่มค้นหา

Command	Target	Value
Page SSCRN home		
assertTitle	\${Version}	
type	id=searchForm:txtMobileN...	\${MobileNumber}
click	id=btnSearchCust	
waitForText	css=div.profile-info-value ...	\${UserName}

รูปที่ 3.29 แสดงการใช้คำสั่ง Click

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยการเข้าถึง HTML element จะใช้ค่าของ HTML attribute ดังรูปที่ 3.30

```
255);" type="text">
▶ <span id="btnSearchCust" class="input-group-addon" style="cursor:
background-color: rgb(238, 238, 238);"></span>
<input id="searchForm:j_idt41" class="jsfBtnSearchCustomer"
name="searchForm:j_idt41" value="btnSearchCustomer" style="displa
```

รูปที่ 3.30 ค่าของ HTML attribute ที่ใช้คำสั่ง Click

- waitForText css=div.profile-info-value >#lbl-username  
\${UserName}

สั่งให้โปรแกรมรอจนกว่าจะมี Username แสดงบนหน้า Home ถ้ายังไม่มี Username แสดงโปรแกรมจะแสดงผล Fail ในบรรทัดนี้ แล้วจึงทำคำสั่งในบรรทัดถัดไปใน Table

Command	Target	Value
Page SSCRN home		
assertTitle	\$(Version)	
type	id=searchForm:txtMobileN...	\$(MobileNumber)
click	id=btnSearchCust	
waitForText	css=div.profile-info-value ...	\$(UserName)

รูปที่ 3.31 แสดงการใช้คำสั่ง waitForText

โดยการเข้าถึง HTML element จะใช้ค่าของ HTML attribute ดังรูปที่ 3.32

```
<div class="profile-info-value" style="margin-left: 25px;">
  <span id="lbl-username" style="font-size: 25px; background-color: transparent;">
    </span>
</div>
```

รูปที่ 3.32 ค่าของ HTML attribute ที่ใช้คำสั่ง waitForText

- verifyText css=div.profile-info-value >#lbl-username  
\${UserName}

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการตรวจสอบ UserName บนหน้า Home ว่าแสดงถูกต้องหรือไม่ เมื่อเปรียบเทียบกับค่า Username ใน Setting

Command	Target	Value
verifyText	css=div.profile-info-value > #lbl-...	{UserName}
pause	2000	
storeEval	var date = new Date();date.getDat...	fullyFormattedDate
captureEntirePageS...	\$(PathPic)Capture_\$(fullyFormatt...	
clickAndWait	css=div.infobox-content	

รูปที่ 3.33 แสดงการใช้คำสั่ง verifyText

โดยการเข้าถึง HTML element จะใช้ค่าของ HTML attribute ดังรูปที่ 3.34

```
<div class="profile-info-value" style="margin-left: 25px;">
  <span id="lbl-username" style="font-size: 25px; background-color: transparent;">
  </span>
</div>
```

รูปที่ 3.34 ค่าของ HTML attribute ที่ใช้คำสั่ง verifyText

- Pause 2000 หยุดรอเป็นเวลา 2 วินาที แล้วค่อยทำการ Execute คำสั่งถัดไปใน Table เพื่อเป็นการหยุดดูการแสดงผลบนหน้าจอ

Command	Target	Value
verifyText	css=div.profile-info-value > #lbl-...	{UserName}
pause	2000	
storeEval	var date = new Date();date.getDat...	fullyFormattedDate
captureEntirePageS...	\$(PathPic)Capture_\$(fullyFormatt...	
clickAndWait	css=div.infobox-content	

รูปที่ 3.35 แสดงการใช้คำสั่ง Pause

- `storeEval var date = new Date();date.getDate() + '-' + (date.getMonth()+1) + '-' + date.getFullYear() + '-' + date.getHours() + '-' + date.getMinutes() + '-' + date.getSeconds(); fullyFormattedDate` คือการสั่งให้ Selenium อ่านค่าวันที่ และเวลาในปัจจุบัน โดยใช้ JavaScript จากนั้นเอามาเก็บไว้ในตัวแปร `fullyFormattedDate`

Command	Target	Value
verifyText	css=div.profile-info-value > #lbl-...	\$(UserName)
pause	2000	
storeEval	var date = new Date();date.getDat...	fullyFormattedDate
captureEntirePageS...	\$(PathPic)Capture_\$(fullyFormatt...	
clickAndWait	css=div.infobox-content	

รูปที่ 3.36 แสดงการใช้คำสั่ง storeEval

- `captureEntirePageScreenshot`  
`$(PathPic)Capture_$(fullyFormattedDate).png`  
 ทำการจับภาพหน้าจอ SSCRN Home โดยให้ทำการบันทึกไฟล์ภาพลงใน Pathpic ที่ทำการกำหนดไว้ใน Setting และตั้งชื่อรูปภาพว่า Capture\_เวลา และวันที่ปัจจุบัน โดยรูปภาพจะถูกกำหนดนามสกุลไฟล์ไว้เป็น .png

Command	Target	Value
verifyText	css=div.profile-info-value > #lbl-...	\$(UserName)
pause	2000	
storeEval	var date = new Date();date.getDat...	fullyFormattedDate
captureEntirePageS...	\$(PathPic)Capture_\$(fullyFormatt...	
clickAndWait	css=div.infobox-content	

รูปที่ 3.37 แสดงการใช้คำสั่ง captureEntirePageScreenshot

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- clickAndWait css=div.infobox-content ทำการกดปุ่ม Promotion service เพื่อเข้าสู่หน้า Customer information

Command	Target	Value
verifyText	css=div.profile-info-value > #lbl-...	\${UserName}
pause	2000	
storeEval	var date = new Date();date.getDat...	fullyFormattedDate
captureEntirePageS...	\${PathPic}Capture_\${fullyFormatt...	
clickAndWait	css=div.infobox-content	

รูปที่ 3.38 แสดงการใช้คำสั่ง clickAndWait

โดยการเข้าถึง HTML element จะใช้ค่าของ HTML attribute ดังรูปที่ 3.39

```

<div class="infobox-icon"></div>
<div class="infobox-data">
  <div class="infobox-content" style="background-color: transparent;">
    Promotion Service</div>
</div>
    
```

รูปที่ 3.39 ค่าของ HTML attribute ที่ใช้คำสั่ง clickAndWait

ถัดมาจะเป็นการตรวจสอบการแสดงผลในหน้า Customer information โดยมีคำสั่งที่ใช้ทั้งหมดดังนี้

- Pause 4000 หยุดรอเป็นเวลา 4 วินาที แล้วค่อยทำการ Execute คำสั่งถัดไปใน Table เพื่อเป็นการหยุดดูการแสดงผลบนหน้าจอ

Command	Target	Value
pause	4000	
waitForText	id=alert-header	Alert
assertTitle	\${Version}	

รูปที่ 3.40 แสดงการใช้คำสั่ง Pause

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- waitForText id=alert-header Alert สั่งให้โปรแกรมรอนกว่าหน้าจจะมีคำว่า Alert แสดงขึ้นมาก่อน จึงจะทำการ Execute คำสั่งถัดไป

Command	Target	Value
pause	4000	
waitForText	id=alert-header	Alert
assertTitle	#{Version}	

รูปที่ 3.41 แสดงการใช้คำสั่ง waitForText เพื่อรอข้อความ Alert แสดงบน

หน้าจอ

โดยการเข้าถึง HTML element จะใช้ค่าของ HTML attribute ดังรูปที่ 3.X

```
<div class="timeline-items" style="display: block;">
  <h5 id="alert-header">
  <div id="error-alert">
    <div class="timeline-item clearfix">
```

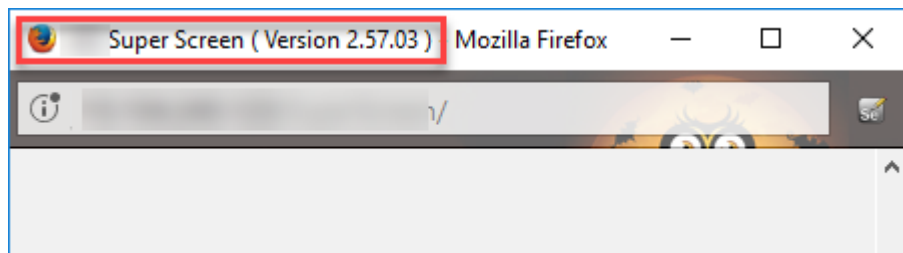
รูปที่ 3.42 ค่าของ HTML attribute ที่ใช้คำสั่ง waitForText

- assertTitle #{Version} เป็นการตรวจสอบเวอร์ชันของระบบที่ทำการทดสอบอีกครั้งในหน้า Customer information หากไม่ถูกต้องโปรแกรมจะหยุดการทำงานทันที

Command	Target	Value
assertTitle	#{Version}	
verifyValue	id=j_idt94:txtMobileNumber	#{MobileNumber}

รูปที่ 3.43 แสดงการใช้คำสั่ง assertTitle

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.44 การแสดงตรวจสอบด้วยคำสั่ง AssertTitle

- verifyValue id=j\_idt94:txtMobileNumber \${MobileNumber} ทำการตรวจสอบค่าที่ input เข้าไปใน Textbox ว่าแสดงถูกต้องหรือไม่

Command	Target	Value
verifyValue	id=j_idt99:txtMobileNumber	\${MobileNumber}
verifyText	id=lbl-username	\${UserName}

รูปที่ 3.45 แสดงการใช้คำสั่ง verifyValue

โดยการเข้าถึง HTML element จะใช้ค่าของ HTML attribute ดังรูปที่ 3.46

```
<div class="input-group">
  <input id="j_idt99:txtMobileNumber" class="form-control
  mobileInputTextbox" name="j_idt99:txtMobileNumber">
```

รูปที่ 3.46 ค่าของ HTML attribute ที่ใช้คำสั่ง verifyValue

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- verifyText จะเป็นการตรวจสอบ ข้อความที่แสดงบนหน้าจอ ปุ่ม แต่ละส่วนว่าแสดงถูกต้องหรือไม่

Table	Source	
Command	Target	Value
verifyText	id=lbl-username	\${UserName}
verifyText	xpath=//i[@id='lbl-BrandMode...	\${NetworkType}
verifyText	id=lbl-ServiceYear	\${ServiceYear}
verifyText	id=lbl-MobileStatus	\${MobileStatus}
verifyText	id=lbl-LastIVR	\${MobileSegment}
verifyText	css=i.icon-remove.bigger-120	
verifyText	id=lbl-CustomerType	\${CustomerType}
verifyText	//span[@id='customer-info-det...	\${PWVerified}
verifyText	//span[@id='customer-info-det...	\${SecretPassword}
verifyText	xpath=//span[@id='lbl-Networ...	\${BillingCycle}
verifyText	//span[@id='customer-info-det...	\${CreditLimit}

รูปที่ 3.47 แสดงการตรวจสอบการแสดงผลโดยใช้คำสั่ง verifyText

โดยการเข้าถึง HTML element จะใช้ค่าของ HTML attribute ซึ่งจะเรียงลำดับการทำงานตั้งแต่รูปที่ 3.48-3.58 ดังนี้

```
break-word;">
▶ <span id="lbl-username" style="background-color: transparent;">
</span>
```

รูปที่ 3.48 ค่าของ HTML attribute ที่ใช้คำสั่ง verify สำหรับตรวจสอบ Username

```

▼<div class="profile-info-value" style="margin-left: 25px; height: 40px;">
  <i id="lbl-BrandModel" title="Network Type" style="color: rgb(209, 91, 71); background-color: transparent;">

```

รูปที่ 3.49 ค่าของ HTML attribute ที่ใช้คำสั่ง verify สำหรับตรวจสอบ Network Type

```

▼<div class="profile-info-value cusinfo-cel2-value">
  <span id="lbl-ServiceYear" style="background-color: transparent;">
    1 Years 9 Months 27 Days</span>

```

รูปที่ 3.50 ค่าของ HTML attribute ที่ใช้คำสั่ง verify สำหรับตรวจสอบ Service Year

```

▼<div class="profile-info-value cusinfo-cel2-value">
  <span id="lbl-MobileStatus">Active </span>
</div>

```

รูปที่ 3.51 ค่าของ HTML attribute ที่ใช้คำสั่ง verify สำหรับตรวจสอบ Mobile Status

```

▼<div class="profile-info-value cusinfo-cel2-value">
  <span id="lbl-LastIVR">Serenade CEO </span>
</div>

```

รูปที่ 3.52 ค่าของ HTML attribute ที่ใช้คำสั่ง verify สำหรับตรวจสอบ Mobile Segment

```

▼<span class="badge badge-primary mini-box-status-N">
  <i class="icon-remove bigger-120" style="background-color: transparent;">

```

รูปที่ 3.53 ค่าของ HTML attribute ที่ใช้คำสั่ง verify สำหรับตรวจสอบ Migrate To BOS

```

▼<div class="profile-info-value cusinfo-cel2-value">
  <span id="lbl-CustomerType" style="background-color: rgba(0, 0, 0, 0);">Thai Citizen </span>
</div>

```

รูปที่ 3.54 ค่าของ HTML attribute ที่ใช้คำสั่ง verify สำหรับตรวจสอบ Customer Type

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<div class="profile-info-name cusinfo-cel3-label">PW Verified</div>
<div class="profile-info-value cusinfo-cel3-value">☰</div>
```

รูปที่ 3.55 ค่าของ HTML attribute ที่ใช้คำสั่ง verify สำหรับตรวจสอบ PW

Verified

```
<span id="customer-info-detail-panel2">
<input id="customerCategory" name="customerCategory" value="R"
type="hidden">
<input id="residentialMessage" name="residentialMessage" value=""
type="hidden">
```

รูปที่ 3.56 ค่าของ HTML attribute ที่ใช้คำสั่ง verify สำหรับตรวจสอบ Secret

Password

```
<div class="profile-info-value cusinfo-cel3-value">
<span id="lbl-NetworkType" Cycle 24-23 &nbsp;  
</span>
```

รูปที่ 3.57 ค่าของ HTML attribute ที่ใช้คำสั่ง verify สำหรับตรวจสอบ Billing

Cycle

```
<div style="min-height: 32px;" class="profile-info-row">
<div class="profile-info-name cusinfo-cel3-label">Available Credit Limit
</div>
```

รูปที่ 3.58 ค่าของ HTML attribute ที่ใช้คำสั่ง verify สำหรับตรวจสอบ Credit

Limit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากการพัฒนาโปรแกรมเพื่อทดสอบพีเจเจอร์ ระบบ SSCRN สามารถแสดงข้อมูลพื้นฐานของลูกค้าในหน้า Customer information ได้อย่างถูกต้อง ด้วยเครื่องมือ Selenium IDE แล้ว 1 ประเภท คือประเภท Postpaid NonBos จะได้ Test script ที่ใช้ในการทำงานทั้งหมดดังรูปที่ 3.59 และ 3.60

Command	Target	Value
<b>Page SSCRN home</b>		
assertTitle	#{Version}	
type	id=searchForm:txtMobileNumber	#{MobileNumber}
click	id=btnSearchCust	
waitForText	css=div.profile-info-value > #lbl-username	#{UserName}
verifyText	css=div.profile-info-value > #lbl-username	#{UserName}
pause	2000	
storeEval	var date = new Date();date.getDate() + '-' + (d... fullyFormattedDate	
captureEntirePa...	\$(PathPic)Capture_\$(fullyFormattedDate).png	
clickAndWait	css=div.infobox-content	

รูปที่ 3.59 Test script หน้า Home ของกลุ่มลูกค้าประเภท Postpaid NonBos

Command	Target	Value
<b>Page Customer information</b>		
pause	4000	
waitForText	id=alert-header	Alert
assertTitle	#{Version}	
verifyValue	id=j_idt99:txtMobileNumber	#{MobileNumber}
verifyText	id=lbl-username	#{UserName}
verifyText	xpath=//*[@id='lbl-BrandModel']][2]	#{NetworkType}
verifyText	id=lbl-ServiceYear	#{ServiceYear}
verifyText	id=lbl-MobileStatus	#{MobileStatus}
verifyText	id=lbl-LastIVR	#{MobileSegment}
verifyText	css=i.icon-remove.bigger-120	
verifyText	id=lbl-CustomerType	#{CustomerType}
verifyText	//span[@id='customer-info-detail-panel2']/div[2]/div	#{PWVerified}
verifyText	//span[@id='customer-info-detail-panel2']/div[2]/div[2]	#{SecretPassword}
verifyText	xpath=//*[@id='lbl-NetworkType']][2]	#{BillingCycle}
verifyText	//span[@id='customer-info-detail-panel2']/div[2]/div[4]/div[2]	#{CreditLimit}
storeEval	var date = new Date();date.getDate() + '-' + (date.getMonth()+1) ... fullyFormattedDate	
captureEntirePageS...	\$(PathPic)Capture_\$(fullyFormattedDate).png	
clickAndWait	css=#endcallink > img.endcall-img	

รูปที่ 3.60 Test script หน้า Customer infomation ของกลุ่มลูกค้าประเภท Postpaid NonBos

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นจะสามารถนำ Test script ไปพัฒนาเพื่อทดสอบกับกลุ่มลูกค้าประเภทอื่นได้ โดยทำการเปลี่ยนส่วน Setting ที่มีไว้สำหรับใส่ข้อมูลนำเข้า เช่น ชื่อ เบอร์โทรศัพท์ของลูกค้าที่ต้องการตรวจสอบ สำหรับ Test script ส่วนของการ Verify สำหรับกลุ่มลูกค้าประเภท Post paid ทั้งหมดจะใช้ Script เดียวกัน จะเปลี่ยน Script เพียงหนึ่งบรรทัด โดยการเปลี่ยนแปลงคำสั่งจะแสดงในรูปที่ 3.60 และส่วน Verify ของกลุ่มลูกค้าประเภท Prepaid จะเหมือนกันทั้งหมด โดยแสดงตัวอย่างดังรูปที่ 3.61

Command	Target	Value
<b>Page Customer information</b>		
pause	4000	
waitForText	id=alert-header	Alert
assertTitle	\${Version}	
verifyValue	id=j_idt99:txtMobileNumber	\${MobileNumber}
verifyText	id=lbl-username	\${UserName}
verifyText	xpath=//i[@id='lbl-BrandModel']	\${NetworkType}
verifyText	id=lbl-ServiceYear	\${ServiceYear}
verifyText	id=lbl-MobileStatus	\${MobileStatus}
verifyText	id=lbl-LastIVR	\${MobileSegment}
verifyText	css=i.icon-bigger-120	
verifyText	id=lbl-CustomerType	\${CustomerType}
verifyText	//span[@id='customer-info-detail-panel2']/div[2]/div	\${PWVerified}
verifyText	//span[@id='customer-info-detail-panel2']/div[2]/div[2]	\${SecretPassword}
verifyText	xpath=//span[@id='lbl-NetworkType']	\${BillingCycle}
verifyText	//span[@id='customer-info-detail-panel2']/div[2]/div[4]/div[2]	\${CreditLimit}
storeEval	var date = new Date();date.getDate() + '-' + (date.getMonth()+1) ...	fullyFormattedDate
captureEntirePageScreenshot	\${PathPic}Capture_\${fullyFormattedDate}.png	
clickAndWait	css=#endcallink > img.endcall-img	

รูปที่ 3.61 การเปลี่ยนแปลง Test script เพื่อใช้งานกับกลุ่มลูกค้าประเภท Postpaid Bos

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Command	Target	Value
<b>Page SSCRN home</b>		
assertTitle	\${Version}	
type	id=searchForm:txtMobileNumber	\${MobileNumber}
click	id=btnSearchCust	
waitForText	css=div.profile-info-value > #lbl-username	\${UserName}
verifyText	css=div.profile-info-value > #lbl-username	\${UserName}
pause	2000	
storeEval	var date = new Date();date.getDate() + '-' + (date.getMonth()+1) ...	fullyFormattedDate
captureEntirePageS...	\${PathPic}Capture_\$(fullyFormattedDate).png	
clickAndWait	css=div.infobox-content	

รูปที่ 3.62 Test script หน้า SSCRN Home สำหรับ กลุ่มลูกค้าประเภท Prepaid ทั้งหมด

Command	Target	Value
<b>Page Customer information</b>		
waitForText	id=alert-header	Alert
assertTitle	\${Version}	
storeText	css=div.profile-info-value.cusinfo-cel2-value	\${AccountStatus}
verifyText	css=b	Prepaid Promotion Ser...
verifyValue	id=j_idt99:txtMobileNumber	\${MobileNumber}
verifyText	css=#customer-info-detail-panel1 > div.profile-info-row > div.pr...	\${UserName}
verifyText	//span[@id='customer-info-detail-panel1']/div[2]/div[2]	\${Age}
verifyText	css=div.profile-info-row > div.profile-info-value.cusinfo-cel2-val...	\${ServiceYear}
verifyText	//span[@id='customer-info-detail-panel2']/div/div[2]/div[2]	\${MobileSegment}
verifyText	//span[@id='customer-info-detail-panel2']/div/div[3]/div[2]	\${NetworkType}
verifyText	id=j_idt1195	\${AccountBalance}
verifyText	//span[@id='customer-info-detail-panel2']/div[2]/div/div[2]	\${BillingSystem}
verifyText	//span[@id='customer-info-detail-panel2']/div/div[5]/div[2]	\${MaxBalance}
verifyTable	//span[@id='customer-info-detail-panel2']/div[2]/div[2]/table.1.0	\${Validity}
verifyTable	//span[@id='customer-info-detail-panel2']/div[2]/div[2]/table.1.1	\${MaxActiveDay}
verifyTable	//span[@id='customer-info-detail-panel2']/div[2]/div[3]/table.1.0	\${ActiveStop}
verifyTable	//span[@id='customer-info-detail-panel2']/div[2]/div[3]/table.1.1	\${SuspendStop}
verifyTable	//span[@id='customer-info-detail-panel2']/div[2]/div[3]/table.1.2	\${DisableStop}
storeEval	var date = new Date();date.getDate() + '-' + (date.getMonth()+1) ...	fullyFormattedDate
captureEntirePageS...	\${PathPic}Capture_\$(fullyFormattedDate).png	

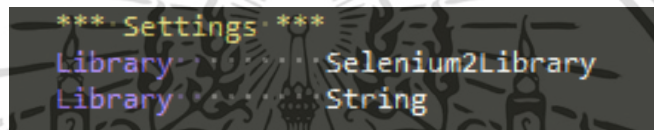
รูปที่ 3.63 Test script หน้า Customer infomation สำหรับกลุ่มลูกค้าประเภท Prepaid ทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.2 การพัฒนาโปรแกรมเพื่อทดสอบพีเจอร์ ระบบ SSCRN สามารถแสดง ชื่อ-นามสกุล เมื่อทำการค้นหาเบอร์โทรศัพท์บนหน้า Home ได้อย่างถูกต้อง ด้วยเครื่องมือ Robot framework

หลังจากออกแบบการทดสอบ ซึ่งอยู่ในรูปแบบ Scenario ต่างๆเรียบร้อยแล้ว ในขั้นตอนการพัฒนาโปรแกรมทดสอบอัตโนมัตินี้จะเป็นการพัฒนาซอร์สโค้ด ด้วยเครื่องมือ Robot framework เพื่อให้โปรแกรมทดสอบอัตโนมัติสามารถทำงานได้ตามที่ออกแบบไว้แต่ละ Scenario ซึ่งเป็นการพัฒนาจะต้องพัฒนาตามโครงสร้างหลักของ Framework 4 ส่วน คือ Settings , Keywords, Variables และ Test case ดังนี้

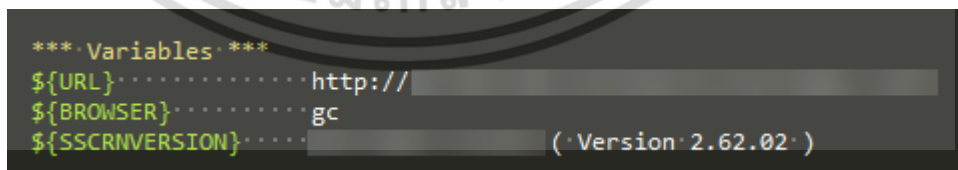
- 1) Settings ในส่วนนี้จะเป็นการกำหนด Library ที่ต้องการจะนำ Keyword ต่างๆในLibrary มาใช้งาน สำหรับ Library ที่เลือกมาใช้งานคือ Selenium2 Library และ String Library



```
*** Settings ***
Library           Selenium2Library
Library           String
```

รูปที่ 3.64 การกำหนด Library ที่ใช้ในการทดสอบ

- 2) Variables ทำการประกาศตัวแปรที่จะใช้งานในการทดสอบ โดยตัวแปรที่ใช้ดังนี้
  - URL ใช้สำหรับ เก็บค่า URL ที่ใช้ในการทดสอบ
  - BROWSER ใช้สำหรับ เก็บค่า เบราเซอร์ที่ต้องการจะเรียกใช้ เช่น Google chrome Internet explorer หรือ Firefox
  - SSCRNVERSION ใช้สำหรับ เก็บค่า Version ของระบบ เพื่อนำไปใช้ในการทดสอบ



```
*** Variables ***
${URL}           http://
${BROWSER}       gc
${SSCRNVERSION}  (Version 2.62.02)
```

รูปที่ 3.65 ตัวแปรที่ใช้ในการทดสอบ

- 3) สร้าง Keyword ขึ้นมาเพื่อใช้ในการเขียน Test script โดย Keyword ที่ใช้ในการทดสอบ พีเจอร์ ระบบ SSCRN สามารถแสดง ชื่อ-นามสกุล เมื่อทำการค้นหาเบอร์โทรศัพท์บนหน้า Home ได้อย่างถูกต้อง มีทั้งหมดดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1) Open Browser and select User เป็นการสั่งให้โปรแกรมเปิดเบราว์เซอร์ที่ใช้ในการทดสอบ และเลือก User ที่จะเข้าใช้งาน โดยการใช้งาน Keyword นี้ ผู้ทดสอบจะต้องทำการเลือก Browser ที่ต้องเปิด และ User เอง สำหรับการทำงานภายใน Keyword Open Browser and select User จะประกอบด้วย Keyword ย่อยต่างๆ ดังรูปภาพที่ 3.66

```

*** Keywords ***
Open Browser and select User
  [Arguments] ..... ${USER}
  Open Browser ..... ${URL} ..... ${BROWSER}
  Wait Until Element Is Visible ..... link= ..... ${USER}
  Click Element ..... link= ..... ${USER}
  Select Window ..... Title = ..... ${SSCRVERSION}
  Title Should Be ..... ${SSCRVERSION}
  Wait Until Element Is Visible ..... /*[@id="searchForm:txtMobileNumber"]

```

รูปที่ 3.66 การสร้าง Keyword Open Browser and select User

ซึ่ง Keyword Open Browser and select User จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังรูปที่ 3.67

```

<div class="profile-info-value" style="margin-left: 25px; width: 80%;">
  <div class="input-group">
    <input id="searchForm:txtMobileNumber" class="form-control mobileInputTextbox"
      name="searchForm:txtMobileNumber" maxlength="10" onkeypress="return
      checkCharacterInputForMobileNumber();" style="background-color: rgb(255, 255, 255);
    <span id="btnSearchCust" class="input-group-addon" style="cursor: pointer;"></span>
    <input id="searchForm:j_idt41" class="jsfBtnSearchCustomer" name="searchForm:j_idt

```

รูปที่ 3.67 ค่าของ HTML attribute ที่ใช้คำสั่ง Wait Until Element Is Visible

3.2) Enter Mobile Number เป็นการสั่งให้โปรแกรมป้อนเบอร์โทรศัพท์ที่ต้องการทดสอบ แล้วกดค้นหา โดยผู้ทดสอบต้องทำการระบุเบอร์ที่ใช้ในการทดสอบเอง

```

Enter Mobile Number
  [Arguments] ..... ${MobileNumber}
  Input text ..... id=searchForm:txtMobileNumber ..... ${MobileNumber}
  Click Element ..... css=i.icon-search.bigger-110
  Wait Until Element Is Visible ..... css=div.profile-info-value>.\#lbl-username

```

รูปที่ 3.68 การสร้าง Keyword Enter Mobile Number

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่ง Keyword Enter Mobile Number จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ทั้งหมดดังนี้

```
<div class="profile-info-value" style="margin-left: 25px; width: 80%;">
  <div class="input-group">
    <input id="searchForm:txtMobileNumber" class="form-control mobileInputTextbox"
      name="searchForm:txtMobileNumber" maxlength="10" onkeypress="return
      checkCharacterInputForMobileNumber();" style="background-color: rgb(255, 255, 255)
    >
    <span id="btnSearchCust" class="input-group-addon" style="cursor: pointer;"></span>
    <input id="searchForm:j_idt41" class="jsfBtnSearchCustomer" name="searchForm:j_idt
```

รูปที่ 3.69 ค่าของ HTML attribute ที่ใช้คำสั่ง Input text

```
<span id="btnSearchCust" class="input-group-addon" style="cursor: pointer;">
  <i class="icon-search bigger-110"></i>
</span>
```

รูปที่ 3.70 ค่าของ HTML attribute ที่ใช้คำสั่ง Click Element

```
<div class="profile-info-value" style="margin-left: 25px;">
  <span id="lbl-username" style="font-size: 25px; background-color: transparent;">
  </span>
</div>
```

รูปที่ 3.71 ค่าของ HTML attribute ที่ใช้คำสั่ง Wait Until Element Is Visible

3.3) Verify Customer Name at Home เป็นการสั่งให้โปรแกรมตรวจสอบชื่อของลูกค้าที่แสดงบนหน้าจอ ว่าตรงกับ Expected result ที่ระบุหรือไม่ โดยผู้ทดสอบจะต้องทำการระบุชื่อที่ใช้ในการตรวจสอบเอง

```
Verify Customer Name at Home
... [Arguments] ... ${Expected CustomerName}
... ${Name} ... Get Text ... css=div.profile-info-value>.\#lbl-username
... ${Name} ... Strip String ... ${Name}
... Should Be Equal ... ${Name} ... ${Expected CustomerName}
```

รูปที่ 3.72 การสร้าง Keyword Verify Customer Name at Home

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่ง Keyword Verify Customer Name at Home จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังรูปที่ 3.73

```
<div class="profile-info-value" style="margin-left: 25px;">
  <span id="lbl-username" style="font-size: 25px; background-color: transparent;">
    </span>
</div>
```

รูปที่ 3.73 ค่าของ HTML attribute ที่ใช้คำสั่ง Get Text

3.4) End call เป็นการสั่งให้โปรแกรมกดปุ่มวางสาย บนหน้า Home

```
End call
Click Element .....css=#endcallink > img.endcall-img
```

รูปที่ 3.74 การสร้าง Keyword End call

ซึ่ง Keyword End call จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังรูปที่ 3.75

```
<a href="#" id="endcallink" onclick="endCall()">
  
</a>
```

รูปที่ 3.75 ค่าของ HTML attribute ที่ใช้คำสั่ง Click Element

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) Testcase คือส่วนของการเขียน Test Cases และการนำ Keyword ที่สร้าง หรือ import เขียนเป็นชุดคำสั่ง สำหรับพีเจอร์ ระบบ SSCRN สามารถแสดง ชื่อ-นามสกุล เมื่อทำการค้นหาเบอร์โทรศัพท์บนหน้า Home ได้อย่างถูกต้อง จะแบ่งออกเป็นทั้งหมด 5 Scenario ตามที่ออกแบบไว้ในตารางที่ 3.4

4.1) Scenario 1 หน้า SSCRN Home สามารถแสดงชื่อ-นามสกุลลูกค้าได้ อย่างถูกต้อง (Postpaid Bos)

```
*** Test Cases ***
Scenario 1 : Verify Customer Name : SSCRN Homepage (Postpaid Bos)
#Test Step                                     Expected Result-----#
.. Open Browser and select User ..
.. Enter Mobile Number ..
.. Verify Customer Name at Home ..
.. Capture Page Screenshot ..
.. End call ..
```

รูปที่ 3.76 Testcase สำหรับ Scenario 1 ลูกค้าประเภท Postpaid Bos

4.2) Scenario 2 หน้า SSCRN Home สามารถแสดงชื่อ-นามสกุลลูกค้าได้ อย่างถูกต้อง (Postpaid Non Bos)

```
Scenario 2 : Verify Customer Name : SSCRN Homepage (Postpaid Non Bos)
.. Enter Mobile Number ..
.. Verify Customer Name at Home ..
.. Capture Page Screenshot ..
.. End call ..
```

รูปที่ 3.77 Testcase สำหรับ Scenario 2 ลูกค้าประเภท Postpaid Non Bos

4.3) Scenario 3 หน้า SSCRN Home สามารถแสดงชื่อ-นามสกุลลูกค้าได้ อย่างถูกต้อง (Prepaid Bos)

```
Scenario 3 : Verify Customer Name : SSCRN Homepage (Prepaid Bos)
.. Enter Mobile Number ..
.. Verify Customer Name at Home ..
.. Capture Page Screenshot ..
.. End call ..
```

รูปที่ 3.78 Testcase สำหรับ Scenario 3 ลูกค้าประเภท Prepaid Bos

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4) Scenario 4 หน้า SSCRN Home สามารถแสดงชื่อ-นามสกุลลูกค้าได้อย่างถูกต้อง (Prepaid Non Bos)

```
Scenario 4 : Verify Customer Name : SSCRN Homepage (Prepaid Non Bos)
... Enter Mobile Number .....
... Verify Customer Name at Home .....
... Capture Page Screenshot .....
... End call
```

รูปที่ 3.79 Testcase สำหรับ Scenario 4 ลูกค้าประเภท Prepaid Non Bos

4.5) Scenario 5 หน้า SSCRN Home สามารถแสดงชื่อ-นามสกุลลูกค้าได้อย่างถูกต้อง (Prepaid INS)

```
Scenario 5 : Verify Customer Name : SSCRN Homepage (Prepaid INS)
... Enter Mobile Number .....
... Verify Customer Name at Home .....
... Capture Page Screenshot .....
... End call
```

รูปที่ 3.80 Testcase สำหรับ Scenario 5 ลูกค้าประเภท Prepaid INS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.3 การพัฒนาโปรแกรมเพื่อทดสอบพีเจอร์ ระบบ SSCRN แสดง Package ได้ถูกต้องเมื่อทำการค้นหา Main Package ในหน้า Customer information ด้วยเครื่องมือ Robot framework การพัฒนาจะต้องพัฒนาตามโครงสร้างหลักของ Framework 4 ส่วน คือ Settings , Keywords, Variables และ Test case ซึ่งในส่วนของ Setting จะใช้ซอร์สโค้ดเดียวกับพีเจอร์ที่ 3.4.2 ดังนั้นจึงจะแสดงการพัฒนาในส่วนที่แตกต่างกันเท่านั้น 3 ส่วน คือ Variables Keywords และ Test case ดังนี้

- 1) Variable ประกาศตัวแปรที่จะใช้งานในการทดสอบ โดยมีตัวแปรที่ใช้ทั้งหมดดังนี้
  - URL ใช้สำหรับเก็บค่า URL ที่ใช้ในการทดสอบ
  - BROWSER ใช้สำหรับเก็บค่า เบราเซอร์ที่ต้องการจะเรียกใช้ คือ Google chrome
  - SSCRNVERSION ใช้สำหรับเก็บค่า Version ของระบบ เพื่อนำไปใช้ในการทดสอบ
  - SEARCH WITH PKG\_NAME ใช้สำหรับเก็บชื่อแพคเกจ
  - SEARCH WITH KEYWORD ใช้สำหรับเก็บชื่อย่อของแพคเกจ
  - SEARCH WITH PKG\_DESC ใช้สำหรับเก็บคำอธิบายของแพคเกจ
  - SEARCH WITH PKG\_CODE ใช้สำหรับเก็บรหัสของแพคเกจ

```

*** Variables ***
${URL} ..... http://10.104.240.123/SuperScreen/login.jsf#
${BROWSER} ..... gc
${SSCRNVERSION} ..... AIS Super Screen (Version 2.62.02)
${SEARCH WITH PKG_NAME} ..... [BOS]_3G_NetSIM 429B 1GB512MB UL Night 1GB Awifi
${SEARCH WITH KEYWORD} ..... free
${SEARCH WITH PKG_DESC} ..... โทรศัพท์มือถือจอพับ 1100 บาท
${SEARCH WITH PKG_CODE} ..... P15084923

```

รูปที่ 3.81 ตัวแปรที่ใช้ในการทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) Keyword ที่ใช้ในการเขียน Test script ระบบ SSCRN แสดง Package ได้ถูกต้องเมื่อทำการค้นหา Main Package ในหน้า Customer information มีทั้งหมดดังนี้

2.1) Open Browser and select User เป็นการสั่งให้โปรแกรมเปิดเบราว์เซอร์ที่ใช้ในการทดสอบ และเลือก User ที่จะเข้าใช้งาน โดยการใช้งาน Keyword นี้ ผู้ทดสอบจะต้องทำการเลือก Browser ที่ต้องเปิด และ User เอง สำหรับการทำงานภายใน Keyword Open Browser and select User จะประกอบด้วย Keyword ย่อยต่างๆ ดังรูปภาพที่ 3.82

```

*** Keywords ***
Open Browser and select User
  [Arguments] ..... ${USER}
  Open Browser ..... ${URL} ..... ${BROWSER}
  Wait Until Element Is Visible ..... link= ..... ${USER}
  Click Element ..... link = ..... ${USER}
  Select Window ..... Title = ..... ${SSCRNVERSION}
  Title Should Be ..... ${SSCRNVERSION}
  Wait Until Element Is Visible ..... /*[@id="searchForm:txtMobileNumber"]

```

รูปที่ 3.82 การสร้าง Keyword Open Browser and select User

ซึ่ง Keyword Open Browser and select User จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังรูปที่ 3.83

```

<div class="profile-info-value" style="margin-left: 25px; width: 80%;">
  <div class="input-group">
    <input id="searchForm:txtMobileNumber" class="form-control mobileInputTextbox"
      name="searchForm:txtMobileNumber" maxlength="10" onkeypress="return
      checkCharacterInputForMobileNumber();" style="background-color: rgb(255, 255, 255);
    <span id="btnSearchCust" class="input-group-addon" style="cursor: pointer;"></span>
    <input id="searchForm:j_idt41" class="jsfBtnSearchCustomer" name="searchForm:j_idt

```

รูปที่ 3.83 ค่าของ HTML attribute ที่ใช้คำสั่ง Wait Until Element Is Visible

2.2) Enter mobile number 10 digit เป็นการสั่งให้โปรแกรมป้อนเบอร์โทรศัพท์ที่ต้องการทดสอบ จำนวน 10 หลัก แล้วกดค้นหา โดยผู้ทดสอบต้องทำการระบุเบอร์ที่ใช้ในการทดสอบเอง

```

Enter mobile number 10 digit
  [Arguments] ..... ${MobileNumber}
  Input text ..... id=searchForm:txtMobileNumber ..... ${MobileNumber}
  Click Element ..... css=i.icon-search.bigger-110
  Wait Until Element Is Visible ..... css=div.profile-info-value> \#lbl-username

```

รูปที่ 3.84 การสร้าง Keyword Enter Mobile Number

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่ง Keyword Enter Mobile Number จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ทั้งหมดดังนี้

```
<div class="profile-info-value" style="margin-left: 25px; width: 80%;">
  <div class="input-group">
    <input id="searchForm:txtMobileNumber" class="form-control mobileInputTextbox"
      name="searchForm:txtMobileNumber" maxlength="10" onkeypress="return
      checkCharacterInputForMobileNumber();" style="background-color: rgb(255, 255, 255)
    >
    <span id="btnSearchCust" class="input-group-addon" style="cursor: pointer;"></span>
    <input id="searchForm:j_idt41" class="jsfBtnSearchCustomer" name="searchForm:j_idt
```

รูปที่ 3.85 ค่าของ HTML attribute ที่ใช้คำสั่ง Input text

```
<span id="btnSearchCust" class="input-group-addon" style="cursor: pointer;">
  <i class="icon-search bigger-110">..</i>
</span>
```

รูปที่ 3.86 ค่าของ HTML attribute ที่ใช้คำสั่ง Click Element

```
<div class="profile-info-value" style="margin-left: 25px;">
  <span id="lbl-username" style="font-size: 25px; background-color: transparent;">
  </span>
</div>
```

รูปที่ 3.87 ค่าของ HTML attribute ที่ใช้คำสั่ง Wait Until Element Is Visible

- 2.3) Verify Customer Name at Home เป็นการสั่งให้โปรแกรมตรวจสอบชื่อของลูกค้าที่แสดงบนหน้าจอ ว่าตรงกับ Expected result ที่ระบุหรือไม่ โดยผู้ทดสอบจะต้องทำการระบุชื่อที่ใช้ในการตรวจสอบเอง

```
Verify Customer Name at Home
... [Arguments] ... ${Expected CustomerName}
... ${Name} ... Get Text ... css=div.profile-info-value> \#lbl-username
... ${Name} ... Strip String ... ${Name}
... Should Be Equal ... ${Name} ... ${Expected CustomerName}
```

รูปที่ 3.88 การสร้าง Keyword Verify Customer Name at Home

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่ง Keyword Verify Customer Name at Home จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ทั้งหมดดังนี้

```
<div class="profile-info-value" style="margin-left: 25px;">
  <span id="lbl-username" style="font-size: 25px; background-color: transparent;">
    </span>
</div>
```

รูปที่ 3.89 ค่าของ HTML attribute ที่ใช้คำสั่ง Get Text

2.4) Click promotion service button เป็นการสั่งให้โปรแกรมกดปุ่ม Promotion service บนหน้า Home

```
Click promotion service button
...Click Element...css=div.infobox-content
```

รูปที่ 3.90 การสร้าง Keyword Click promotion service button

ซึ่ง Keyword Click promotion service button จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ทั้งหมดดังนี้

```
<div class="infobox infobox-green boxMainService">
  <div class="infobox-icon">...</div>
  <div class="infobox-data">...</div>
</div>
```

รูปที่ 3.91 ค่าของ HTML attribute ที่ใช้คำสั่ง Click Element

2.5) Verify Customer name เป็นการสั่งให้โปรแกรมตรวจสอบชื่อลูกค้าที่แสดงบนหน้าจอ โดยการทำงานของ Keyword นี้เริ่มจากรอให้มีปุ่ม Refresh ขึ้นก่อน จากนั้นจึงทำการรับชื่อของลูกค้าด้วย Keyword Get Text และนำมาตัดช่องว่างออก สุดท้ายคือการนำค่าที่รับเข้ามามาเปรียบเทียบกับ Expected result ด้วย Keyword Should Be Equal ว่าตรงกันหรือไม่

```
Verify Customer name
...#Get name
...[Arguments]...${Expected CustomerName}
...${Name}...Get Text...id=lbl-username
...${Name}...Strip String...${Name}
...Should Be Equal...${Name}...${Expected CustomerName}
```

รูปที่ 3.92 การสร้าง Keyword Verify Customer name

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่ง Keyword Verify Customer name จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังนี้

```
<div class="profile-info-value" style="margin-left: 25px; word-wrap: break-word;">
  <span id="lbl-username">
</div>
```

รูปที่ 3.93 ค่าของ HTML attribute ที่ใช้คำสั่ง Get Text

2.6) Verify Mobile status เป็นการสั่งให้โปรแกรมตรวจสอบ Mobile status ที่แสดงบนหน้าจอ โดยการทำงานของ Keyword นี้คือ ทำการรับ Mobile status ด้วย Keyword Get Text และนำมาตัดช่องว่างออก สุดท้ายคือการนำค่าที่รับเข้ามานำมาเปรียบเทียบกับ Expected result ด้วย Keyword Should Be Equal ว่าตรงกันหรือไม่

```
Verify Mobile status
[Arguments] ..... ${Expected MobileStatus}
#Get Status
${Status} ..... Get Text ..... id=lbl-MobileStatus
${Status} ..... Strip String ..... ${Status}
Should Be Equal ..... ${Status} ..... ${Expected MobileStatus}
```

รูปที่ 3.94 การสร้าง Keyword Verify Mobile status

ซึ่ง Keyword Verify Mobile status จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังนี้

```
<div class="profile-info-value cusinfo-cel2-value">
  <span id="lbl-MobileStatus">Active </span>
</div>
```

รูปที่ 3.95 ค่าของ HTML attribute ที่ใช้คำสั่ง Get Text

2.7) Click Button Change Main promotion เป็นการสั่งให้โปรแกรมกดปุ่มเปลี่ยน Main Promotion

```
Click Button Change Main promotion
... Wait Until Element Is Visible ..... //a[@id='changemainpackagelink']/i
... Click Element ..... //a[@id='changemainpackagelink']/i
```

รูปที่ 3.96 การสร้าง Keyword Click Button Change Main promotion

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่ง Keyword Verify Mobile status จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังนี้

```
<a title="Change Package" href="#box-ChangePackage" id="changemainpackagelink" data-toggle="modal" style="text-decoration: none;">
```

รูปที่ 3.97 ค่าของ HTML attribute ที่ใช้ใน Keyword Verify Mobile status

2.8) Enter Empty Main Promotion เป็นการสั่งให้โปรแกรมระบุค่าว่าง ลงไปในช่องค้นหาชื่อแพคเกจ จากนั้นทำการกดปุ่มค้นหา

```
Enter Empty Main Promotion
Click Element ..... id=bt-Search-MainPackage
```

รูปที่ 3.98 การสร้าง Keyword Enter Empty Main Promotion

ซึ่ง Keyword Enter Empty Main Promotion จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังนี้

```
<button id="bt-Search-MainPackage" class="btn-Search" style="margin-top: -1px; width: 120px;">
</a>
```

รูปที่ 3.99 ค่าของ HTML attribute ที่ใช้ใน Keyword Enter Empty Main Promotion

2.9) Verify Package Main promotion เป็นการสั่งให้โปรแกรมรับค่าแพคเกจที่แสดงบนหน้าจอ ทำการตัดช่องว่างทิ้งด้วย Keyword Strip String จากนั้นจึงทำการเปรียบเทียบกับ Expected result ที่ผู้ทดสอบระบุใน Testcase ว่าตรงกันหรือไม่

```
Verify Package Main promotion
[Arguments] ..... ${Expected MainPackage}
${MainPackage} ..... Get Text ..... css=div.box-pac-row1-col > span
${MainPackage} ..... Strip String ..... ${MainPackage}
Should Be Equal ..... ${MainPackage} ..... ${Expected MainPackage}
```

รูปที่ 3.100 การสร้าง Keyword Verify Package Main promotion

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่ง Keyword Verify Package Main promotion จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังนี้

```
▼ <div class=" box-pac-row1">
  ▼ <div class="box-pac-row1-col">
    <div class="packageIU" style="display: none;">P15084923</div>
```

รูปที่ 3.101 ค่าของ HTML attribute ที่ใช้ใน Keyword Verify Package Main promotion

2.10) Enter Package Main Promotion เป็นการสั่งให้โปรแกรมใส่ค่า แพคเกจที่ต้องการค้นหาโดยผู้ทดสอบจะต้องเป็นคนระบุชื่อแพคเกจเอง

```
Enter Package Main Promotion
[Arguments] ..... ${MainPkg}
Input text ..... name=j_idt524:j_idt526 ..... ${MainPkg}
```

รูปที่ 3.102 การสร้าง Keyword Enter Package Main Promotion

ซึ่ง Keyword Enter Package Main Promotion จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังนี้

```
<input type="text" name="j_idt526:j_idt528:j_idt530" value class="
col-xs-12 col-sm-12 inputSearchMain">
```

รูปที่ 3.103 ค่าของ HTML attribute ที่ใช้ใน Keyword Enter Package Main Promotion

2.11) Click Search Package Main Promotion เป็นการสั่งให้โปรแกรมกดปุ่มค้นหาแพคเกจ

```
Click Search Package Main Promotion
Click Element ..... id=mclickSearch2
```

รูปที่ 3.104 การสร้าง Keyword Click Search Package Main Promotion

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่ง Keyword Click Search Package Main Promotion จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังนี้

```
<button class="btn-Search" style="margin-top:
-1px;" id="mclickSearch2">
<i class="icon-search bigger-110 icon-search-bt
```

รูปที่ 3.105 ค่าของ HTML attribute ที่ใช้ใน Keyword Click Element

2.12) Verify Invalid Package Main promotion เป็นการสั่งให้โปรแกรมตรวจสอบการ ค้นหาแพคเกจ กรณีระบุแพคเกจไม่ถูกต้อง

```
Verify Invalid Package Main promotion
[Arguments] ..... ${Expected MainPackage}
Wait Until Page Contains ..... ${Expected MainPackage}
```

รูปที่ 3.106 การสร้าง Keyword Click Search Package Main Promotion

3) Testcase คือส่วนของการเขียน Test Cases และการนำ Keyword ที่สร้าง หรือ import เขียนเป็นชุดคำสั่ง สำหรับระบบ SSCRN แสดง Package ได้ถูกต้องเมื่อทำการค้นหา Main Package ในหน้า Customer information จะแบ่งออกเป็นทั้งหมด 5 Scenario ตามที่ ออกแบบไว้ในตารางที่ 3.6

3.1) Scenario 1 ตรวจสอบการแสดงผล Customer Name บนหน้า SSCRN Home

```
*** Test Cases ***
Scenario 1 : Verify Customer Name : SSCRN Homepage
# Test Step Expected Result-----#
Open Browser and select User
Enter mobile number 10 digit
Verify Customer Name at Home
Capture Page Screenshot
Click promotion service button
```

รูปที่ 3.107 Testcase สำหรับ Scenario 1

3.2) Scenario 2 ตรวจสอบการแสดงผลข้อมูลพื้นฐานลูกค้า บนหน้า Customer information

```
Scenario 2 : Verify value : Customer information page
Verify Customer name
Verify Mobile status Active
Capture Page Screenshot
```

รูปที่ 3.108 Testcase สำหรับ Scenario 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3) Scenario 3 ค้นหาแพคเกจ ด้วยการไม่ระบุข้อความในช่อง แล้วค้นหา (Empty Main Promotion)

```
Scenario 3 : Search Empty Main Promotion
... Click Button Change Main promotion
... Enter Empty Main Promotion
... Verify Package Main promotion ..... Please enter your search conditions.
```

รูปที่ 3.109 Testcase สำหรับ Scenario 3

3.4) Scenario 4 ค้นหาแพคเกจ Main Promotion ด้วย Package Name

```
Scenario 4 : Search Main Promotion With Package Name
... Enter Package Main Promotion ..... ${SEARCH WITH PKG_NAME}
... Click Search Package Main Promotion
... Verify Package Main promotion ..... [BOS]_
```

รูปที่ 3.110 Testcase สำหรับ Scenario 4

3.5) Scenario 5 ค้นหาแพคเกจ Main Promotion ด้วย Keyword Package

```
Scenario 4 : Search Main Promotion With Package Name
... Enter Package Main Promotion ..... ${SEARCH WITH PKG_NAME}
... Click Search Package Main Promotion
... Verify Package Main promotion ..... [BOS]_
```

รูปที่ 3.111 Testcase สำหรับ Scenario 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3.6) Scenario 6 ค้นหาแพคเกจ Main Promotion ด้วย Package Description

```
Scenario 6 : Search Main Promotion With Package Description
... Enter Package Main Promotion ..... ${SEARCH_WITH_PKG_DESC}
... Click Search Package Main Promotion
... Verify Package Main promotion ..... [BOS]
```

รูปที่ 3.112 Testcase สำหรับ Scenario 6

## 3.7) Scenario 7 ค้นหาแพคเกจ Main Promotion ด้วย Package Code

```
Scenario 7 : Search Main Promotion With Package Code
... Enter Package Main Promotion ..... ${SEARCH_WITH_PKG_CODE}
... Click Search Package Main Promotion
... Verify Package Main promotion ..... [BOS]
```

รูปที่ 3.113 Testcase สำหรับ Scenario 7

## 3.8) Scenario 8 ค้นหาแพคเกจ Main Promotion ด้วย Invalid Package Name

```
Scenario 8 : Search Main Promotion With Invalid Package Name
... Enter Package Main Promotion ..... feewify
... Click Search Package Main Promotion
... Verify Invalid Package Main promotion ..... ไม่พบข้อมูล
```

รูปที่ 3.114 Testcase สำหรับ Scenario 8

## 3.9) Scenario 9 ค้นหาแพคเกจ Main Promotion ด้วย Invalid Keyword

```
Scenario 9 : Search Main Promotion With Invalid Keyword
... Enter Package Main Promotion ..... พระบาทสมเด็จ
... Click Search Package Main Promotion
... Verify Invalid Package Main promotion ..... ไม่พบข้อมูล
```

รูปที่ 3.115 Testcase สำหรับ Scenario 9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.10) Scenario 10 ค้นหาแพคเกจ Main Promotion ด้วย Invalid Package Description

```
Scenario 10 : Search Main Promotion With Invalid Package Description
... Enter Package Main Promotion ..... สว่างสดด้วยจอเงินแท้
... Verify Invalid Package Main promotion ..... ไม่พบข้อมูล
... Click Search Package Main Promotion
```

รูปที่ 3.116 Testcase สำหรับ Scenario 10

3.11) Scenario 11 ค้นหาแพคเกจ Main Promotion ด้วย Invalid Package Code

```
Scenario 11 : Search Main Promotion With Invalid Package Code
... Enter Package Main Promotion ..... P123456789X
... Click Search Package Main Promotion
... Verify Invalid Package Main promotion ..... ไม่พบข้อมูล
```

รูปที่ 3.117 Testcase สำหรับ Scenario 11

3.12) Scenario 12 วางสาย และกดปุ่ม Close Browser

```
Scenario 12 : End Call And Close Browser
... Close Search Package
... End call
... Close Browser
```

รูปที่ 3.118 Testcase สำหรับ Scenario 12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.4 การพัฒนาโปรแกรมเพื่อทดสอบพีเจอร์ ระบบ SSCRN สามารถออก Order (ซื้อแพคเกจ หรือ ยกเลิกแพคเกจ) ได้อย่างถูกต้อง ด้วยเครื่องมือ Robot framework

สำหรับพีเจอร์นี้สุดท้ายนี้จะการพัฒนาตามโครงสร้างหลักของ Framework 4 ส่วน คือ Settings , Keywords, Variables และ Test case เหมือนกับพีเจอร์อื่นๆที่พัฒนาด้วย Robot framework ซึ่งสำหรับพีเจอร์นี้จะเริ่มทำการทดสอบตั้งแต่หน้าแรก และไปจบในหน้าสุดท้าย ซึ่งจะมีมีการนำซอร์สโค้ดจากพีเจอร์อื่นๆมาประยุกต์ใช้ด้วย ดังนั้นการพัฒนาจะมีการเปลี่ยนแปลงส่วน Keywords, Variables และ Test case ดังต่อไปนี้

- 1) Variable ประกาศตัวแปรที่จะใช้งานในการทดสอบ โดยมีตัวแปรที่จะใช้จะเหมือนกับการพัฒนาในหัวข้อที่ 3.4.2 แต่จะมีเพิ่มอีก 1 ตัวแปร เพื่อใช้สำหรับเก็บค่าแพคเกจที่ใช้ในการทดสอบ ดังรูปที่ 3.119

```
*** Variables ***
${URL}      http://
${BROWSER}
${USER}
${SSCRNVERSION}    Super Screen (Version 2.62.02)
${PACKAGE}    One Time Free Call 120Min 10Days Ontop BOSCorp
```

รูปที่ 3.119 ตัวแปรที่ใช้ในการทดสอบ

- 2) Keyword ที่ใช้ในการเขียน Test script ของพีเจอร์ ระบบ SSCRN สามารถออก Order (ซื้อแพคเกจ หรือ ยกเลิกแพคเกจ) ได้อย่างถูกต้อง มีทั้งหมดดังนี้

2.1) Open Browser and select User เป็นการสั่งให้โปรแกรมเปิดเบราว์เซอร์ที่ใช้ในการทดสอบ และเลือก User ที่จะเข้าใช้งาน โดยการใช้งาน Keyword นี้ ผู้ทดสอบจะต้องทำการเลือก Browser ที่ต้องเปิด และ User เอง สำหรับการทำงานภายใน Keyword Open Browser and select User จะประกอบด้วย Keyword ย่อยต่างๆ ดังรูปภาพที่ 3.120

```
*** Keywords ***
Open Browser and select User
    [Arguments] ..... ${USER}
    Open Browser ..... ${URL} ..... ${BROWSER}
    Wait Until Element Is Visible ..... link= ..... ${USER}
    Click Element ..... link= ..... ${USER}
    Select Window ..... Title = ..... ${SSCRNVERSION}
    Title Should Be ..... ${SSCRNVERSION}
    Wait Until Element Is Visible ..... /*[@id="searchForm:txtMobileNumber"]
```

รูปที่ 3.120 การสร้าง Keyword Open Browser and select User

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้ดูแลระบบเว็บไซต์มีการอัปเดตหรือแก้ไขข้อมูลใดๆ ไม่สามารถรับประกันได้ว่าเอกสารฉบับนี้จะไม่มีการเปลี่ยนแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่ง Keyword Open Browser and select User จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังรูปที่ 3.121

```
<div class="input-group">
  <input id="searchForm:txtMobileNumber" type="text" name="
    searchForm:txtMobileNumber" value="0911000665" class="form-
```

รูปที่ 3.121 ค่าของ HTML attribute ที่ใช้คำสั่ง Wait Until Element Is Visible

2.2) Enter mobile number 10 digit เป็นการสั่งให้โปรแกรมป้อนเบอร์โทรศัพท์ที่ต้องการทดสอบ จำนวน 10 หลัก แล้วกดค้นหา โดยผู้ทดสอบต้องทำการระบุเบอร์ที่ใช้ในการทดสอบเอง

```
Enter mobile number 10 digit
... [Arguments] ... ${MobileNumber}
... Input text ... id=searchForm:txtMobileNumber ... ${MobileNumber}
... Click Element ... css=i.icon-search.bigger-110
... Wait Until Element Is Visible ... css=div.profile-info-value > #lbl-username
```

รูปที่ 3.122 การสร้าง Keyword Enter Mobile Number

ซึ่ง Keyword Enter Mobile Number จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ทั้งหมดดังนี้

```
<div class="profile-info-value" style="margin-left: 25px; width: 80%;">
  <div class="input-group">
    <input id="searchForm:txtMobileNumber" class="form-control mobileInputTextbox"
      name="searchForm:txtMobileNumber" maxlength="10" onkeypress="return
      checkCharacterInputForMobileNumber();" style="background-color: rgb(255, 255, 255);
```

รูปที่ 3.123 ค่าของ HTML attribute ที่ใช้คำสั่ง Input text

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

▼<span id="btnSearchCust" class="input-group-addon" style="cursor: pointer;">
  ▶<i class="icon-search bigger-110">...</i>
</span>

```

รูปที่ 3.124 ค่าของ HTML attribute ที่ใช้คำสั่ง Click Element

```

▼<div class="profile-info-value" style="margin-left: 25px;">
  <span id="lbl-username" style="font-size: 25px; background-color: transparent;">
    <span>
  </span>
</div>

```

รูปที่ 3.125 ค่าของ HTML attribute ที่ใช้คำสั่ง Wait Until Element Is Visible

2.3) Verify Customer Name at Home เป็นการสั่งให้โปรแกรมตรวจสอบชื่อของลูกค้าที่แสดงบนหน้าจอ ว่าตรงกับ Expected result ที่ระบุหรือไม่ โดยผู้ทดสอบจะต้องทำการระบุชื่อที่ใช้ในการตรวจสอบเอง

```

Verify Customer Name at Home
[Arguments] ..... ${Expected CustomerName}
${Name} ..... Get Text ..... css=div.profile-info-value > \#lbl-username
${Name} ..... Strip String ..... ${Name}
Should Be Equal ..... ${Name} ..... ${Expected CustomerName}

```

รูปที่ 3.126 การสร้าง Keyword Verify Customer Name at Home

ซึ่ง Keyword Verify Customer Name at Home จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ทั้งหมดดังนี้

```

▼<div class="profile-info-value" style="margin-left: 25px;">
  <span id="lbl-username" style="font-size: 25px; background-color: transparent;">
    <span>
  </span>
</div>

```

รูปที่ 3.127 ค่าของ HTML attribute ที่ใช้คำสั่ง Get Text

2.4) Click promotion service button เป็นการสั่งให้โปรแกรมกดปุ่ม Promotion service บนหน้า Home

```

Click promotion service button
Click Element ..... css=div.infobox-content

```

รูปที่ 3.128 การสร้าง Keyword Click promotion service button

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่ง Keyword Click promotion service button จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ทั้งหมดดังนี้

```
<a href="#" id="endcalllink" onclick="endCall()">
  
</a>
```

รูปที่ 3.129 ค่าของ HTML attribute ที่ใช้คำสั่ง Click Element

2.5) Verify Customer name เป็นการสั่งให้โปรแกรมตรวจสอบชื่อลูกค้าที่แสดงบนหน้าจอ โดยการทำงานของ Keyword นี้เริ่มจากรอให้มีปุ่ม Refresh ขึ้นก่อน จากนั้นจึงทำการรับชื่อของลูกค้าด้วย Keyword Get Text และนำมาตัดช่องว่างออก สุดท้ายคือการนำค่าที่รับเข้ามามาเปรียบเทียบกับ Expected result ด้วย Keyword Should Be Equal ว่าตรงกันหรือไม่

```
Verify Customer name
... Wait Until Element Is Visible ..... css=a[title="Alert Information Refresh"] > .i.icon-refresh
#Get name
[Arguments] ..... ${Expected CustomerName}
${Name} ..... Get Text ..... id=lbl-username
${Name} ..... Strip String ..... ${Name}
Should Be Equal ..... ${Name} ..... ${Expected CustomerName}
```

รูปที่ 3.130 การสร้าง Keyword Verify Customer name

ซึ่ง Keyword Verify Customer name จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ทั้งหมดดังนี้

```
margin-top: -25px; cursor: pointer;"/>
  <i class="icon-refresh">
  ::before
```

รูปที่ 3.131 ค่าของ HTML attribute ที่ใช้คำสั่ง Wait Until Element Is Visible

```
<div class="profile-info-value" style="margin-left: 25px;"/>
  <span id="lbl-username" style="font-size: 25px; background-color: transparent;"/>
  </span>
</div>
```

รูปที่ 3.132 ค่าของ HTML attribute ที่ใช้คำสั่ง Get Text

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6) Verify Mobile status เป็นการสั่งให้โปรแกรมตรวจสอบ Mobile status ที่แสดงบนหน้าจอ โดยการทำงานของ Keyword นี้คือ ทำการรับ Mobile status ด้วย Keyword Get Text และนำมาตัดช่องว่างออก สุดท้ายคือการนำค่าที่รับเข้ามานำมาเปรียบเทียบกับ Expected result ด้วย Keyword Should Be Equal ว่าตรงกันหรือไม่

```
Verify Mobile status
... [Arguments] ..... ${Expected MobileStatus}
... #Get status
... ${Status} ..... Get Text ..... id=lbl-MobileStatus
... ${Status} ..... Strip String ..... ${Status}
... Should Be Equal ..... ${Status} ..... ${Expected MobileStatus}
```

### รูปที่ 3.133 การสร้าง Keyword Verify Mobile status

ซึ่ง Keyword Verify Mobile status จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังนี้

```
<div class="profile-info-value cusinfo-cel2-value">
  <span id="lbl-MobileStatus" Active
    &nbsp;
  </span>
</div>
```

### รูปที่ 3.134 ค่าของ HTML attribute ที่ใช้คำสั่ง Get Text

2.7) Click go to Package Consultation page เป็นการสั่งให้โปรแกรมกดปุ่มเพื่อไปยังหน้า Package Consultation

```
Click go to Package Consultation page
... Click Element ..... css=span.wizard-title.wizard-mouse-pointer
```

### รูปที่ 3.135 การสร้าง Keyword Click go to Package Consultation page

ซึ่ง Keyword Click go to Package Consultation page จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังนี้

```
<div class="wizard-1">...</div>
<span class="wizard-title wizard-mouse-pointer" onclick="gotopage('/
SuperScreen/pages/services/promotion/step-package-consultation.jsf', 1)">
PACKAGE CONSULTATION </span>
```

### รูปที่ 3.136 ค่าของ HTML attribute ที่ใช้คำสั่ง Click Element

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8) Enter Package เป็นการสั่งให้โปรแกรมใส่ค่าแพคเกจที่ต้องการค้นหาลงในช่องโดยผู้ทดสอบจะต้องเป็นคนกำหนดว่าจะให้โปรแกรมใส่ค่าอะไรเพื่อทำการทดสอบ

```
Enter Package
... [Arguments] ..... ${Pkg}
... Input: text ..... id=tempSearchPackageName ..... ${Pkg}
```

รูปที่ 3.137 การสร้าง Keyword Enter Package

ซึ่ง Keyword Enter Package จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังนี้

```
id="txt-Search-MainPackage" /&gt;-->


```

รูปที่ 3.138 ค่าของ HTML attribute ที่ใช้คำสั่ง Input text

2.9) Click Search Package เป็นการสั่งให้โปรแกรมกดปุ่มค้นหาแพคเกจ

```
... Click Element ..... tq=pfuzseulcuj
Click: search package
```

รูปที่ 3.139 การสร้าง Keyword Enter Package

ซึ่ง Keyword Click Search Package จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังนี้

```
><button style="margin-top: -1px; width: 120px;" class="btn-
Search" id="btnSearch1" onclick="javascript:searchPackage();
return false;">...</button>
```

รูปที่ 3.140 ค่าของ HTML attribute ที่ใช้คำสั่ง Click Element

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10) Verify Result Package เป็นการสั่งให้โปรแกรม ตรวจสอบผลลัพธ์หลังจาก การค้นหาแพคเกจ โดยมีการทำงานคือเริ่มจากรับค่าแพคเกจที่แสดงบนหน้าจอ หลังจากทำการค้นหา และทำการตัดช่องว่างทิ้งด้วย Keyword Strip String จากนั้น ทำการเปรียบเทียบกับ Expected result ที่ผู้ทดสอบระบุใน Testcase ว่าตรงกันหรือไม่

```
Verify Result Package
... [Arguments] ..... ${Expected Package}
... ${Package} ..... Get Text ..... css=div.box-pac-row1-col > span
... ${Package} ..... Strip String ..... ${Package}
... Should Be Equal ..... ${Package} ..... ${ExpectedPackage}
```

รูปที่ 3.141 การสร้าง Keyword Verify Result Package

ซึ่ง Keyword Verify Result Package จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังนี้

```
<div class="box-pac">
  <div class="box-pac-row1">
    <div class="box-pac-row1-col"></div>
  </div>
```

รูปที่ 3.142 ค่าของ HTML attribute ที่ใช้คำสั่ง Get text

2.11) Select Effective Date Next Day เป็นการสั่งให้โปรแกรม เลือกวันที่ ที่ ต้องการให้แพคเกจมีผลเริ่มใช้งาน โดยเลือกเป็นวันถัดไป

```
Select Effective Date Next Day
... Click Element ..... id=Search-Offering-Result-tr-0-Icon
... Click Element ..... id=Search-Offering-Result-tr-0-box-Effective-2-R
```

รูปที่ 3.143 การสร้าง Keyword Select Effective Date Next Day

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่ง Keyword Verify Result Package จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ทั้งหมดดังนี้

```
<span style="float: left;">
```

```
<i id="Search-Offering-Result-tr-0-Icon" class="bigger-100  
readmore icon-chevron-up" onclick="javascript:  
mOpen_offer('Search-Offering-Result-tr-0', false);">
```

รูปที่ 3.144 ค่าของ HTML attribute ที่ใช้ Keyword Select Effective Date Next Day (1)

```
<div class="td-cel-prorate-2-1" style="width: 20%; float: left;">  
<input id="Search-Offering-Result-tr-0-box-Effective-2-R" type=  
"radio" name="Search-Offering-Result-tr-0-postpaid" onclick=  
"mselectOffer('Search-Offering-Result-tr-0');">
```

รูปที่ 3.145 ค่าของ HTML attribute ที่ใช้ Keyword Select Effective Date Next Day (2)

2.12) Click Add Package เป็นการสั่งให้โปรแกรมกดปุ่ม Add Package เพื่อให้แพคเกจที่เลือกเข้าสู่ตระกร้า

Click Add Package  
Click Element  
css=i.icon-plus.bigger-110

รูปที่ 3.146 การสร้าง Keyword Click Add Package

ซึ่ง Keyword Click Add Package จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังนี้

```
<span class="badge badge-primary mini-box-status-Y mini-box-  
status-Y" id="bt-Search-Offering-Result-tr-0">  
<i class="icon-plus bigger-110" style="margin-left: 1px;">  
::before  
</i>
```

รูปที่ 3.147 ค่าของ HTML attribute ที่ใช้ใน Keyword Click Add Package

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.13) Verify Package in Shopping cart เป็นการสั่งให้โปรแกรม ตรวจสอบผลลัพธ์ หลังจากการกดปุ่ม Add Package ลงในตระกร้า ว่าภายในตระกร้ามีแพคเกจตรงกันหรือไม่ โดยการทำงานจะเริ่มจาก กดปุ่มดูรายการภายในตระกร้าสินค้า จากนั้นโปรแกรมจะรับค่าแพคเกจที่แสดงบนหน้าจอ มาเก็บไว้ในตัวแปร Package และทำการตัดช่องว่างทิ้งด้วย Keyword Strip String จากนั้นทำการเปรียบเทียบกับ Expected result ที่ผู้ทดสอบระบุใน Testcase ว่าตรงกันหรือไม่

```
Verify Package in Shopping cart
... [Arguments] ..... ${Expected Package}
... Click Element ..... css=i.icon-shopping-cart.bigger-170
... ${Package} ..... Get Text ..... css=div.clearfix.order1
... ${Package} ..... Strip String ..... ${Package}
... Should Be Equal ..... ${Package} ..... ${ExpectedPackage}
```

รูปที่ 3.148 การสร้าง Keyword Click Add Package

ซึ่ง Keyword Verify Package in Shopping cart จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังนี้

```
<li class="purple">
  <a data-toggle="dropdown" class="dropdown-toggle" href="#">
    <i class="icon-shopping-cart bigger-170" style="font-size: 22px;">
      <span>
        </span>
      </i>
    </a>
  </li>
```

รูปที่ 3.149 ค่าของ HTML attribute ที่ใช้ใน Keyword Verify Package in Shopping cart

2.14) Click Go to Package Management เป็นการสั่งให้โปรแกรมกดปุ่ม Go to Package Management เพื่อ Redirect ไปยังหน้า Package management

```
Click Go to Package Management
... Click Element ..... link=Go to Package Management
```

รูปที่ 3.150 การสร้าง Keyword Click Go to Package Management

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่ง Keyword Click Go to Package Management จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังนี้

```

<li>
  <div class="clearfix order1"
    ::before
    > <a href="#">...</a>
  
```

รูปที่ 3.151 ค่าของ HTML attribute ที่ใช้ใน Keyword Click Go to Package Management

2.15) Verify Package in Package Management Page เป็นการสั่งให้โปรแกรมตรวจสอบความถูกต้องของการแสดงผลแพคเกจบนหน้า Package Management Page โดยการทำงานจะเริ่มจากรับค่าแพคเกจที่แสดงบนหน้าจอ มาเก็บไว้ในตัวแปร Pack และทำการตัดช่องว่างทิ้งด้วย Keyword Strip String จากนั้นทำการเปรียบเทียบกับ Expected result ที่ผู้ทดสอบระบุใน Testcase ว่าตรงกันหรือไม่

```

Verify Package in Package Management Page
[Arguments]    ${Expected Package}
${Pack}        Get Text          link=${PACKAGE}
${Pack}        Strip String      ${Pack}
Should Be Equal    ${Pack}        ${ExpectedPackage}

```

รูปที่ 3.152 การสร้าง Keyword Verify Package in Package Management Page

ซึ่ง Keyword Verify Package in Package Management Page จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังนี้

```

<li style="font-size: 22px ! important;">
  <a href="#"
    style="margin-top: 15px; font-size: 21px;">...</a>
</li>

```

รูปที่ 3.153 ค่าของ HTML attribute ที่ใช้ใน Keyword Click Go to Package Management

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.16) Input Description Over 1000 Character เป็นการสั่งให้โปรแกรมตรวจสอบข้อความAlert message เมื่อมีการระบุ Description มากกว่า 1000 ตัวอักษร ที่แสดงบนหน้า Package Management หากใส่ไม่เกิน 1000 ตัวอักษรจะไม่มี Alert message ปรากฏขึ้นมาบนหน้าจอ

```
Input Description Over 1000 Character
... [Arguments] ..... ${Description}
... Input text ..... id=description ..... ${Description}
```

รูปที่ 3.154 การสร้าง Keyword Input Description Over 1000 Character

ซึ่ง Keyword Input Description Over 1000 Character จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังนี้

```
<td colspan="2" rowspan="2" style="text-align: center;">
<textarea id="description" name="description" cols="34"
onkeyup="maximum()" style="width: 100%;" onblur=
"mojarra.ab(this,event,'blur','@this',0)"></textarea>
</td>
```

รูปที่ 3.155 ค่าของ HTML attribute ที่ใช้ใน Keyword Input Description Over 1000 Character

2.17) Click Submit Order เป็นการสั่งให้โปรแกรมตรวจสอบกดปุ่มยืนยันการทำรายการ

```
Click Submit Order
... Click Element ..... //a[@id='submitorder:submitorderbutton']/button
```

รูปที่ 3.155 การสร้าง Keyword Click Submit Order

ซึ่ง Keyword Click Submit Order จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังนี้

```
<span id="submitorder:submitorderbutton" name="submitorder:submitorderbutton">
<button data-last="Finish" class="btn-disable" disabled="disabled" style=
"width: 150px;">
```

รูปที่ 3.156 ค่าของ HTML attribute ที่ใช้ใน Keyword Click Submit Order

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.18) Go to Customer information page เป็นการสั่งให้โปรแกรมกดแท็บ Customer information เป็นการ Redirect กลับไปยังหน้าแรก เพื่อตรวจสอบแพคเกจที่เพิ่มเข้ามาว่าแสดงถูกต้องหรือไม่

```
Go to Customer information page
Click Element .....css=span.wizard-title.wizard-mouse-pointer
```

รูปที่ 3.157 การสร้าง Keyword Go to Customer information page

ซึ่ง Keyword Go to Customer information page จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังนี้

```
<div class="wizard-1">...</div>
<span class="wizard-title wizard-mouse-pointer" onclick="gotopage('/
SuperScreen/pages/services/promotion/step-customer-information.jsf', 3)">
CUSTOMER INFORMATION</span>
```

รูปที่ 3.158 ค่าของ HTML attribute ที่ใช้ใน Keyword Go to Customer information page

2.19) End call เป็นการสั่งให้โปรแกรมกดปุ่มวางสายเพื่อจบการทำงาน

```
End call
Click Element .....css=#endcalllink > img.endcall-img
```

รูปที่ 3.159 การสร้าง Keyword End call

ซึ่ง Keyword Click Submit Order จะมีการเข้าถึง HTML element โดยใช้ค่าของ HTML attribute ดังนี้

```
<a href="#" id="endcalllink" onclick="endCall()">

</a>
```

รูปที่ 3.160 ค่าของ HTML attribute ที่ใช้ใน Keyword End call

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) Testcase คือส่วนของการเขียน Test Cases และการนำ Keyword ที่สร้าง หรือ import เขียนเป็นชุดคำสั่ง สำหรับพีเจอรระบบ SSCRN สามารถออก Order (ซื้อแพคเกจ หรือ ยกเลิกแพคเกจ) ได้อย่างถูกต้องจะแบ่งออกเป็นทั้งหมด 9 Scenario ตามที่ออกแบบไว้ในตารางที่ 3.8

3.1) Scenario 1 ตรวจสอบการแสดงผล Customer Name บนหน้า SSCRN Home

```

*** Test Cases ***
Scenario 1 : Verify Customer Name : SSCRN Homepage
# Test Step Expected Result-----#
... Open Browser and select User ..... PATTARAN
... Enter mobile number 10 digit ..... 0911000665
... Verify Customer Name at Home ..... Ms. SuperScreen GGGBOSAI
... Capture Page Screenshot
... Click promotion service button

```

รูปที่ 3.161 Testcase สำหรับ Scenario 1

3.2) Scenario 2 ตรวจสอบการแสดงผลข้อมูลพื้นฐานลูกค้า บนหน้า Customer information

```

Scenario 2 : Verify value : Customer information page
... Verify Customer name ..... Ms. SuperScreen GGGBOSAI
... Verify Mobile status ..... Active
... Capture Page Screenshot

```

รูปที่ 3.162 Testcase สำหรับ Scenario 2

3.3) Scenario 3 ค้นหาแพคเกจ ที่หน้า Package Consultation

```

Scenario 3 : Search Package : Package Consultation page
... Click go to Package Consultation page
... Enter Package ..... ${PACKAGE}
... Click Search Package
... Verify Result Package ..... ${PACKAGE}

```

รูปที่ 3.163 Testcase สำหรับ Scenario 3

3.4) Scenario 4 ตรวจสอบ การ Add Package

```

Scenario 4 : Add Package
... Select Effective Date Next Day
... Click Add Package
... Verify Package in Shopping cart ..... ${PACKAGE}

```

รูปที่ 3.164 Testcase สำหรับ Scenario 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3.5) Scenario 5 ตรวจสอบแพคเกจที่หน้า Package Management

```
Scenario 5 : Go to Package Management
... Click Go to Package Management
... Verify Package in Package Management Page ..... ${PACKAGE}
```

รูปที่ 3.165 Testcase สำหรับ Scenario 5

## 3.6) Scenario 6 ตรวจสอบ Alert message เมื่อใส่ Description เกิน 1,000 ตัวอักษร

```
Scenario 6 : Verify Alert message Over 1000 Description
... Input Description Over 1000 Character ..... Test By Chompunooch
... Alert Should Be Present
```

รูปที่ 3.166 Testcase สำหรับ Scenario 6

## 3.7) Scenario 7 ตรวจสอบการยืนยันการทำรายการเมื่อ Order complete

```
Scenario 7 : Confirm Order
... Click Submit Order
```

รูปที่ 3.167 Testcase สำหรับ Scenario 7

## 3.8) Scenario 8 ตรวจสอบแพคเกจที่เพิ่มเข้าไปในหน้า Customer information

```
Scenario 6 : Verify Alert message Over 1000 Description
... Input Description Over 1000 Character ..... Test By Chompunooch
... Alert Should Be Present
```

รูปที่ 3.168 Testcase สำหรับ Scenario 8

## 3.9) Scenario 9 กดวางสาย และปิด Browser

```
Scenario 9 : End Call And Close Browser
... End call
... Capture Page Screenshot
... Close Browser
```

รูปที่ 3.169 Testcase สำหรับ Scenario 9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.5 วิเคราะห์ และจัดการกับข้อผิดพลาดที่เกิดขึ้น

เมื่อเกิดปัญหาขึ้นในช่วงของการทดสอบ ผู้ทดสอบจะมีการประชุม และวิเคราะห์ปัญหาที่เกิดขึ้นระหว่าง ผู้พัฒนาและผู้ใช้งานระบบ เพื่อหาแนวทางการแก้ไขปัญหาต่างๆ เมื่อหาต้นเหตุและสิ่งที่เป็นปัญหาได้แล้วจะนำมาแก้ไขโดยผู้พัฒนา และจะมีการใช้เครื่องมือมาช่วยในการติดตามปัญหาต่างๆ หรือเรียกว่า การติดตามปัญหา (Defect Tracking) ซึ่งมีเครื่องมือที่ระบบนี้ใช้ก็คือ Jira ซึ่งเป็น Web Application โดยมีขั้นตอนการติดตามปัญหาดังนี้

#### 3.5.1 ขั้นตอนการติดตามปัญหา (Defect Tracking)

1. เมื่อวิเคราะห์และสรุปการประชุมได้รายละเอียดของปัญหาแล้วทำการใส่รายละเอียดต่างๆลงในไฟล์ Summary Defect
2. นำรายละเอียดของปัญหาต่างๆมาลงใน Create ลง ใน Web Jira ซึ่งสามารถอธิบาย Text Box ต่างๆ ได้ดังนี้

รูปที่ 3.170 Create Issue (1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Project: ชื่อของProject ที่ทำอยู่
- Issue Type: ลักษณะของปัญหา
  - Bug: ปัญหาที่เกิดจากความผิดพลาดของระบบ ต้องถูกนำไปแก้ไข
  - Feature: ความสามารถในการทำงานของระบบที่ต้องใส่เพิ่มเข้าไป
  - CR (Change Request): ปัญหาที่เกิดจากการเปลี่ยนแปลงความต้องการของผู้ใช้งานระบบ
  - Performance Tuning: ปัญหาการแก้ไขประสิทธิภาพของระบบ
- Environment: ตำแหน่งทรัพยากรของระบบที่เกิดปัญหา
  - Production: ทรัพยากรส่วนของการใช้งานจริง
  - Pre-Production: ทรัพยากรส่วนที่ทดลองก่อนการทำงานจริง
  - Development: ทรัพยากรส่วนที่ผู้พัฒนาใช้พัฒนาระบบ
- Issue Group: ส่วนของการเกิดปัญหาในหน้าจอ UI
- Module (Service): ประเภทของการให้บริการที่เกิดปัญหา
  - Promotion: บริการต่างๆ ภายในประเทศ
  - IR: บริการต่างๆ ภายนอกประเทศ
  - All: บริการ ทั้งในประเทศและต่างประเทศ
- Phase: ตำแหน่งของช่วงการพัฒนา

The screenshot shows a 'Create Issue' form with the following fields and options:

- Summary:** A text input field for the issue title.
- Description:** A rich text editor with a toolbar containing options for style, bold, italic, underline, text color, background color, link, unlink, list, and image.
- Priority:** A dropdown menu currently set to 'Medium'.
- Assignee:** A dropdown menu currently set to 'Automatic'.
- Buttons:** 'Create another' (checkbox), 'Create' (blue button), and 'Cancel' (grey button).

รูปที่ 3.171 Create Issue (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Summary: รายละเอียดหัวข้อของปัญหาแบบย่อ
- Description: ช่องการอธิบายรายละเอียดของปัญหาทั้งหมดสามารถใส่รูปภาพ และ ไฟล์วิดีโอที่ไม่ใหญ่เกินไปได้
- Priority: ลำดับความสำคัญของปัญหา
  - High: ปัญหาที่มีความสำคัญมาก และเสี่ยงที่จะทำให้เกิดผลกระทบมาก
  - Medium: ปัญหาที่มีความสำคัญปานกลางไม่เป็นอันตรายต่อระบบ
  - Low: ปัญหาที่มีความสำคัญน้อย แต่ก็ยังต้องแก้ไข
- Assignee: ชื่อของผู้ที่ดูแลและรับผิดชอบปัญหา

The screenshot shows the 'Create Issue' form in JIRA. It features several input fields: 'Original Estimate' (with a help icon and example 'eg. 3w 4d 12h'), 'Remaining Estimate' (with a help icon and example 'eg. 3w 4d 12h'), 'Due Date' (with a calendar icon and Thai text 'วันที่คาดว่าจะเสร็จ'), 'Sprint' (with a dropdown arrow and 'JIRA Software sprint field'), and 'Fix Version/s' (with a dropdown arrow and 'Start typing to get a list of possible matches or press down to select'). At the bottom right, there are three buttons: 'Create another' (with a checkbox), 'Create', and 'Cancel'.

รูปที่ 3.172 Create Issue (3)

- Original Estimate: การประเมินระยะเวลาตั้งต้นในการแก้ไขปัญหา
- Remaining Estimate: การประเมินระยะเวลาที่ยังคงเหลือในการแก้ไข
- Due Date: วันคาดหวังที่ปัญหาจะถูกแก้ไขจนเสร็จ
- Sprint: ช่วงการพัฒนาที่เร่งรีบแก้ไขปัญหาให้เสร็จ
- Fix Version/s: เวอร์ชันที่มีการแก้ไข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Create Issue**
⚙️ Configure Fields ▾

Deploy PRE-PROD\*  🗑️  
date  
วันที่คาดว่าจะขึ้น pre production

Attachment 📎 Drop files to attach, or [browse](#).

Reopen Count   
สำหรับนับว่า issue นี้ถูกเปิดมากี่ครั้งแล้ว

Feature List None ▾  
รายการของ feature ที่มีทั้งหมด ณ ตอนนี  
เนื่องจาก issue ที่เกิดขึ้นใหม่นั้นอาจมีความสัมพันธ์กับ feature แต่ละเรื่อง

Feature Name

is impact\*  Yes  
 No  
bug ที่เกิดขึ้นมันเป็น bug ที่เกิดขึ้นจาก impact จากการพัฒนาในส่วนอื่นใช่หรือไม่

Create another
 Create
Cancel

### รูปที่ 3.173 Create Issue (4)

- Deploy PRE-PROD Date: วันที่ปัญหาจะใช้งานในช่วง Pre-Production
- Attachment: ไฟล์ต่างๆที่เกี่ยวข้องกับปัญหา
- Reopen Count: จำนวนที่ปัญหานี้ถูก Create ไปแล้ว
- Feature List: ความสามารถในการทำงานที่มีความสัมพันธ์กับปัญหา
- Feature Name: ความสามารถในการทำงานที่เกิดปัญหา
- Is Impact: ปัญหานี้เป็นปัญหาเคยที่ใช้งานได้แล้ว แต่ถูกผลกระทบทำให้เกิดปัญหานี้ขึ้นอีกครั้ง หรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Create Issue**
⚙️ Configure Fields ▾

Reopen Count   
สำหรับนับว่า issue นี้ถูกเปิดมากี่ครั้งแล้ว

Feature List ▾  
  
รายการของ feature ที่มีทั้งหมด ณ ตอนนี  
 เนื่องจาก issue ที่เกิดขึ้นมาใหม่นั้นอาจมีความสัมพันธ์กับ feature แต่ละเรื่อง

Feature Name

is impact\*  Yes  
 No  
bug ที่เกิดขึ้นมาเป็น bug ที่เกิดขึ้นจาก impact จากการพัฒนาในส่วนอื่นใช่หรือไม่

Mobile Number\*   
เบอร์ที่เกิดปัญหา

Mobile Sector\*  Prepaid nonBOS  
 Prepaid BOS  
 Postpaid BOS  
 Postpaid nonBOS  
mobile sector ที่พบปัญหา

Create another
 **Create**
Cancel

### รูปที่ 3.174 Create Issue (5)

- Mobile Number: หมายเลขโทรศัพท์ที่เกิดปัญหา
- Mobile Sector (Segment): ประเภทการใช้งานของหมายเลขปัญหา
  - Prepaid nonBOS: ประเภทการใช้งานแบบ เต็มเงิน ระบบ Billing เก่า
  - Prepaid BOS: ประเภทการใช้งานแบบ เต็มเงิน ระบบ billing ใหม่
  - Postpaid BOS: ประเภทการใช้งานแบบ รายเดือน ระบบ billing ใหม่
  - Postpaid nonBOS: ประเภทการใช้งานแบบรายเดือนระบบBillingเก่า

3. เมื่อCreate ปัญหาต่างๆลงใน Jira แล้ว หัวหน้าทีมพัฒนา หรือ Senior Developer ก็จะนำปัญหาต่างๆ ไปจัดการและมอบหมายให้กับ Developer แก้ไข

4. เมื่อปัญหาถูกแก้ไขเสร็จแล้ว ก็จะถูกทดสอบด้วยทีมทดสอบ หรือ Tester

5. เมื่อผ่านการทดสอบจากฝั่งของผู้พัฒนาแล้ว จะถูกนำไปทดสอบอีกที จากฝั่งผู้ใช้งานระบบ ก่อนนำปัญหาต่างๆที่ถูกแก้ไขแล้วขึ้นใช้งานจริง

6. ในการเปลี่ยนแปลงสถานะต่างๆของปัญหา จะถูก Update ในไฟล์ Summary Defect

7. เมื่อเกิดปัญหา หรือ ทีมพัฒนาต้องการข้อมูลเพิ่มเติม จะจัดให้มีการประชุมเพื่อวิเคราะห์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### ผลการดำเนินงาน

หลังจากที่พัฒนาซอฟต์แวร์ทดสอบอัตโนมัติเสร็จสิ้น ทางทีมผู้พัฒนาและทีมผู้ทดสอบ ระบบ SSCRN ก็ได้นำซอฟต์แวร์ทดสอบอัตโนมัตินี้ไปใช้ในการทดสอบการทำงานของเว็บแอปพลิเคชัน หลังจากที่มีการปรับปรุงแก้ไขข้อผิดพลาดทุกครั้ง เพื่อทำการหาข้อผิดพลาดและทำการแก้ไขก่อนจะนำข้อผิดพลาดที่ปรับปรุงใหม่ไปใช้งานจริง หลังจากที่ได้นำซอฟต์แวร์ทดสอบอัตโนมัติมาใช้แทนมนุษย์ พบว่าเวลาที่ใช้ในการทดสอบแต่ละครั้งลดลงเป็นอย่างมาก เนื่องจากซอฟต์แวร์ทดสอบอัตโนมัติทำงานได้รวดเร็วกว่ามนุษย์ และสามารถทดสอบได้หลายชุดในเวลาเดียวกัน ทำให้สามารถทดสอบได้ครอบคลุมทุกเว็บเบราว์เซอร์ ส่งผลให้การหาข้อผิดพลาดและแก้ไขเป็นไปอย่างรวดเร็ว และช่วยให้มั่นใจว่าไม่มีข้อผิดพลาดเกิดจากการทดสอบที่ไม่ครอบคลุม

#### 4.1 ผลการพัฒนาซอฟต์แวร์

หลังจากที่ซอฟต์แวร์ทดสอบอัตโนมัติทำงานเสร็จสิ้นแล้ว จะทำการสร้างรายงานแสดงผลผลการทดสอบออกมาในรูปแบบของไฟล์ HTML ทั้งการพัฒนาด้วยเครื่องมือ Selenium IDE และ Robot Framework โดยในรายงานจะแสดงผลผลการทดสอบของทุก scenario ซึ่งจากการพัฒนาซอฟต์แวร์ด้วยเครื่องมือ 2 ชนิดจะมีผลการทดสอบต่างกัันดังนี้

##### 4.1.1 การพัฒนาซอฟต์แวร์อัตโนมัติด้วยเครื่องมือ Selenium IDE

สำหรับการพัฒนาซอฟต์แวร์อัตโนมัติด้วยเครื่องมือ Selenium IDE จะแสดงผลการทดสอบในรูปแบบของตาราง ไม่แสดงรูปภาพเว็บเพจที่ได้ทดสอบบนแต่ละ scenario โดยรูปภาพที่ทำการ Capture จะสามารถเข้าไปดูที่โฟลเดอร์ที่ตั้งค่าให้บันทึกรูปภาพเท่านั้น ไม่สามารถดูในรายงานได้ ตัวอย่างรายงานแสดงผลการทดสอบดังรูปที่ 4.1

## Summary

× FAILED

2017/06/27 01:37:18

Done	Failed	Undefined	Total
16	2	25	43

### Post\_Cus\_Setting

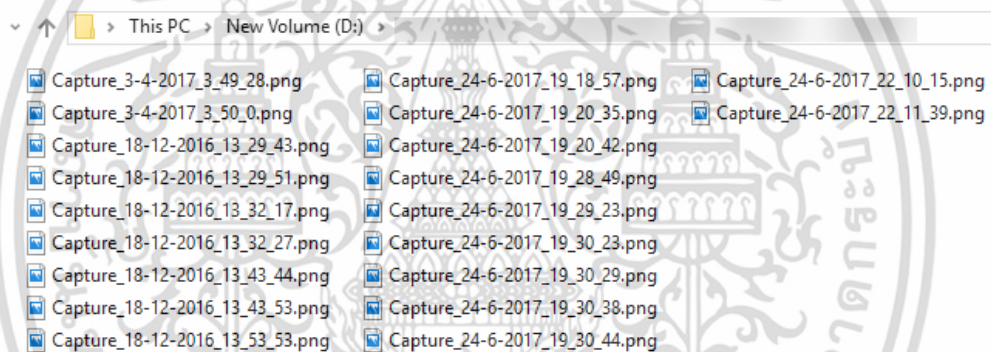
- ✓ 1. Set path for save save picture
- ✓ 2. Set variable for verify Customer info

### Post\_Cus\_Verify

× Page SSCRN home

	Command	Target	Value
✓	assertTitle	\${Version}	
✓	type	id=searchForm.txtMobileNumber	\${MobileNumber}
✓	click	id=btnSearchCust	
×	waitForText	css=div.profile-info-value > #lbl-username	\${UserName}

รูปที่ 4.1 ตัวอย่างรายงานแสดงผลการทดสอบด้วยเครื่องมือ Selenium IDE



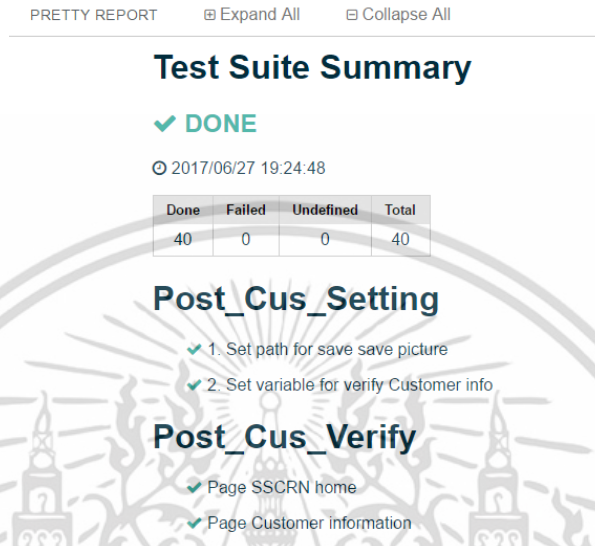
รูปที่ 4.2 ตัวอย่างรูปที่บันทึกหน้าจออัตโนมัติด้วยเครื่องมือ Selenium IDE

ข้อมูลที่แสดงในรายงานผลการทดสอบ ได้แก่ ชื่อ feature ที่ทดสอบ , วันที่และเวลาที่ทำการทดสอบ , จำนวน Test case ทั้งหมด (total) , จำนวน Testcase ที่ทดสอบเรียบร้อยแล้ว (Done) , จำนวน Testcase ที่ยังไม่ได้ทำการทดสอบ (Undefined) , และ จำนวน Testcase ที่เกิดข้อผิดพลาด (Fail) โดยรายงานผลการทดสอบจะแบ่งออกเป็นได้ 2 กรณีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.1.1 ผลลัพธ์การทดสอบกรณีไม่มีข้อผิดพลาด

ถ้าสามารถทำงานได้ปกติและผลลัพธ์ถูกต้องตามที่กำหนดทุก scenario รายงานจะแสดงข้อความ Done และเครื่องหมายถูกเป็นเครื่องหมายสีเขียวทั้งหมด ดังรูปที่ 4.3



รูปที่ 4.3 ผลลัพธ์การทดสอบกรณีไม่มีข้อผิดพลาด ด้วยเครื่องมือ Selenium IDE

โดยสามารถเรียกดูรายละเอียดการทดสอบได้ โดยการกดที่ชื่อ Testcas จากนั้น รายงานจะแสดงผลรายละเอียดการทำงานออกมาในรูปแบบของตาราง ดังรูปที่ 4.4

Post\_Cus\_Setting

- ✓ 1. Set path for save save picture
- ✓ 2. Set variable for verify Customer info

Post\_Cus\_Verify

- ✓ Page SSCRN home

	Command	Target	Value
✓	assertTitle	\$(Version)	
✓	type	id=searchForm.txtMobileNumber	\$(MobileNumber)
✓	click	id=btnSearchCust	
✓	waitForText	css=div.profile-info-value > #lbl-username	\$(UserName)
✓	verifyText	css=div.profile-info-value > #lbl-username	\$(UserName)
✓	pause	2000	
✓	storeEval	var date = new Date();date.getDate() + '/' + (date.getMonth()+1) + '/' + date.getFullYear() + '_' + date.getHours() + '_' + date.getMinutes() + '_' + date.getSeconds();	fullyFormattedDate
✓	captureEntirePageScreenshot	\$(PathPic)Capture_\$(fullyFormattedDate).png	
✓	clickAndWait	css=div.infobox-content	

รูปที่ 4.4 ผลลัพธ์การทดสอบในรูปแบบของตารางกรณีไม่มีข้อผิดพลาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.1.2 ผลลัพธ์การทดสอบกรณีที่มีข้อผิดพลาด

ในกรณีที่มีข้อผิดพลาดเกิดขึ้น รายงานจะแสดงข้อความ FAILED พร้อมด้วยเครื่องหมายกากบาทสีแดง สำหรับภายในตาราง จะแสดงจำนวน Case ที่ผิดพลาดด้วย ดังรูปที่ 4.5

### Test Suite Summary

✖ FAILED

© 2016/11/03 10:26:52

Done	Failed	Undefined	Total
54	5	8	67

### Transaction Report

✔ Menu Transaction Report

### Backend Transaction Report

✖ Menu Backend Transaction Report

### Offer Report\_BasicSearch

✖ Menu Offer Report\_Basic

รูปที่ 4.5 ผลลัพธ์การทดสอบกรณีเกิดข้อผิดพลาด ด้วยเครื่องมือ Selenium IDE

โดยสามารถเรียกดูรายละเอียดการทดสอบได้ โดยการกดที่ชื่อ Testcase จากนั้นรายงานจะแสดงผลรายละเอียดการทำงานออกมาในรูปแบบของตาราง หากภายใน Testcase มี เกิดกรณี Fail ขึ้นมา โปรแกรมจะไม่ทำงานในบรรทัดถัดไป Test case นั้นรายงานผลเป็น Fail ทั้งนี้ สำหรับบรรทัดที่ยังไม่ถูก Execute จะแสดงเครื่องหมายคำถาม ด้านหน้าบรรทัด ดังรูปที่ 4.6

## Backend Transaction Report

✖ Menu Backend Transaction Report

	Command	Target	Value
✓	open	/SuperScreenBackOffice/	
✓	clickAndWait	//form[@id='j_idt13']/a[9]/button	
✓	type	id=fromInputId	11/11/2016 00:00:00
✓	type	name=j_idt19	12/11/2016 23:59:59
✓	click	id=j_idt37	
✖	assertAlert	Created Date-From must be Not Over today	
?	captureEntirePageScreenshot	D:\selenium_java_IDE_To_Webdriver\SanpShot\Sprint26\SSCRN-1403\BackendTransactionReport.jpg	
?	clickAndWait	css=small	

รูปที่ 4.6 ผลลัพธ์การทดสอบในรูปแบบของตารางกรณีเกิดข้อผิดพลาด

### 4.1.2 การพัฒนาซอฟต์แวร์อัตโนมัติด้วยเครื่องมือ Robot Framework

จะแสดงผลการทดสอบในรูปแบบของ HTML โดยรายงานของ Robot Framework จะถูกสร้างหลังจากทำการ Execute Test case เรียบร้อยทั้งหมดโดยอัตโนมัติ ซึ่งมีทั้งหมด 2 File คือ Report และ Log โดย ไฟล์ Log จะแสดงรูปภาพเว็บเพจที่ได้ทดสอบบนแต่ละ scenario ด้วย ช่วยให้สามารถตรวจสอบหาข้อผิดพลาดได้ง่ายขึ้น โดยการดูผลลัพธ์การแสดงผลในรูปภาพแทนการทดสอบซ้ำด้วยมืออีกรอบเพื่อหาข้อผิดพลาด ตัวอย่างรายงานแสดงผลการทดสอบดังรูปที่ 4.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**LOG**  
Generated  
20170624 00:32:40 GMT+07:00  
4 days 2 hours ago

## SentOrder Test Report

**Summary Information**

Status: **1 critical test failed**  
 Start Time: 20170624 00:31:02.145  
 End Time: 20170624 00:32:40.302  
 Elapsed Time: 00:01:38.157  
 Log File: [log.html](#)

**Test Statistics**

Total Statistics	Total	Pass	Fail	Elapsed	Pass / Fail
Critical Tests	8	7	1	00:01:30	<span style="color: green;">7</span> / <span style="color: red;">1</span>
All Tests	8	7	1	00:01:30	<span style="color: green;">7</span> / <span style="color: red;">1</span>

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

Statistics by Suite	Total	Pass	Fail	Elapsed	Pass / Fail
SentOrder	8	7	1	00:01:38	<span style="color: green;">7</span> / <span style="color: red;">1</span>

**Test Details**

Totals Tags Suites Search

Type:  Critical Tests  All Tests

Status: 8 total, 7 passed, 1 failed  
 Total Time: 00:01:29.700

Name	Crit.	Status	Message	Elapsed	Start / End
SentOrder Step 5 : Verify Alert message Over 1000 Description	yes	FAIL	There were no alerts	00:00:02.683	20170624 00:32:30.262 20170624 00:32:32.945
SentOrder Step 1 : Verify Customer Name : SSCRN Homepage	yes	PASS		00:00:31.433	20170624 00:31:10.598 20170624 00:31:42.031
SentOrder Step 2 : Verify value : Customer information page	yes	PASS		00:00:09.595	20170624 00:31:42.032 20170624 00:31:51.627
SentOrder Step 3 : Search Package : Package Consultation page	yes	PASS		00:00:14.845	20170624 00:31:51.628 20170624 00:32:06.473
SentOrder Step 4 : Add Package	yes	PASS		00:00:18.476	20170624 00:32:06.473 20170624 00:32:24.949
SentOrder Step 5 : Go to Package Management	yes	PASS		00:00:05.312	20170624 00:32:24.950 20170624 00:32:30.262
SentOrder Step 6 : Delete Package in Package Management	yes	PASS		00:00:00.232	20170624 00:32:32.945 20170624 00:32:33.177
SentOrder Step 7 : End Call And Close Browser	yes	PASS		00:00:07.124	20170624 00:32:33.178 20170624 00:32:40.302

### รูปที่ 4.7 ตัวอย่างรายงานแสดงผลการทดสอบด้วยเครื่องมือ Robot Framework

สำหรับข้อมูลที่แสดงในรายงานผลการทดสอบ ได้แก่ ชื่อ feature ที่ทดสอบ , ส่วนของ Summary Information ซึ่งประกอบด้วย สถานะการทดสอบ (Status) , วันที่และเวลาที่เริ่มทดสอบ (Start Time) , วันที่และเวลาที่สิ้นสุดการทดสอบ (End Time) , เวลาทั้งหมดที่ใช้ในการทดสอบ (Elapsed Time) , ไฟล์ log สำหรับดูรายละเอียดการทดสอบ , ส่วนของสถิติในการทดสอบ (Test Statistics) ซึ่งประกอบด้วย Total Statistics , Statistics by tag , Statistics by suite ซึ่งแต่ละอัน จะแสดงจำนวน Test case ทั้งหมด (Total) , จำนวนที่ผลการทดสอบถูกต้อง (Passed) , จำนวนผลการทดสอบกรณีไม่ถูกต้อง (Failed) , เวลาที่ใช้ในการทดสอบทั้งหมด (Elapsed) และสถิติการทดสอบทั้งหมด(Pass/Fail) ส่วนสุดท้ายคือ การแสดงรายละเอียดการทดสอบ (Test Detail)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.2.1 ผลลัพธ์การทดสอบกรณีไม่มีข้อผิดพลาด

ถ้าสามารถทำงานได้ปกติและผลลัพธ์ถูกต้องตามที่กำหนด รายงานจะแสดงผลเป็นพื้นหลังสีเขียว โดยส่วนของ Summary Information จะแสดง Status ด้วยข้อความสีเขียวว่า All tests passed และในส่วนของ Test Statistics จะแสดงผลสัดส่วน Pass/Fail ภายในตารางด้วยแถบสีเขียวทั้งหมด ดังรูปที่ 4.8

**Report** Generated  
20170624 00:23:03 GMT+07:00  
3 days 0 hours ago

**Summary Information**

Status: All tests passed  
 Start Time: 20170624 00:20:08.843  
 End Time: 20170624 00:23:03.318  
 Elapsed Time: 00:02:54.475  
 Log File: log.html

**Test Statistics**

Total Statistics	Total	Pass	Fail	Elapsed	Pass / Fail
Critical Tests	11	11	0	00:02:44	<div style="width: 100%; height: 10px; background-color: green;"></div>
All Tests	11	11	0	00:02:44	<div style="width: 100%; height: 10px; background-color: green;"></div>

**Statistics by Tag**

Total	Pass	Fail	Elapsed	Pass / Fail
No Tags				<div style="width: 0%; height: 10px; background-color: green;"></div>

**Statistics by Suite**

Total	Pass	Fail	Elapsed	Pass / Fail	
SearchMobileNumber	11	11	0	00:02:54	<div style="width: 100%; height: 10px; background-color: green;"></div>

**Test Details**

Totals Tags Suites Search

Type:  Critical Tests  All Tests

รูปที่ 4.8 ผลลัพธ์การทดสอบกรณีไม่มีข้อผิดพลาด ด้วยเครื่องมือ Robot Framework

#### 4.1.2.2 ผลลัพธ์การทดสอบกรณีที่มีข้อผิดพลาด

ในกรณีที่มีข้อผิดพลาดเกิดขึ้น รายงานจะแสดงผลเป็นสีชมพู โดยส่วนของ Summary Information จะแสดง Status เป็นจำนวนที่ทำการทดสอบแล้วไม่ผ่าน (Critical test failed) และในส่วนของ Test Statistics จะแสดงผลสัดส่วน Fail ภายในตารางด้วยแถบสีแดง ดังรูปที่ 4.9

## Report

Generated  
20170402 19:42:25 GMT+07:00  
85 days 5 hours ago

### Summary Information

Status: **1 critical test failed**  
 Start Time: 20170402 19:41:28.459  
 End Time: 20170402 19:42:25.405  
 Elapsed Time: 00:00:56.946  
 Log File: [log.html](#)

### Test Statistics

Total Statistics	Total	Pass	Fail	Elapsed	Pass / Fail
Critical Tests	12	11	1	00:00:56	<div style="width: 91.67%;"><div style="width: 91.67%;"></div></div>
All Tests	12	11	1	00:00:56	<div style="width: 91.67%;"><div style="width: 91.67%;"></div></div>

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

Statistics by Suite	Total	Pass	Fail	Elapsed	Pass / Fail
Test	12	11	1	00:00:57	<div style="width: 91.67%;"><div style="width: 91.67%;"></div></div>

### Test Details

Totals Tags Suites Search

Type:  Critical Tests  All Tests

รูปที่ 4.9 ผลลัพธ์การทดสอบกรณีเกิดข้อผิดพลาด ด้วยเครื่องมือ Robot Framework

### Test Details

Totals Tags Suites Search

Name: Test

Status: 12 critical test, 11 passed, **1 failed**  
12 test total, 11 passed, **1 failed**

Start / End Time: 20170402 19:41:28.459 / 20170402 19:42:25.405  
 Elapsed Time: 00:00:56.946  
 Log File: [log.html#s1](#)

Name	Crit.	Status	Message	Elapsed	Start / End
Test.Step 1 : Verify Customer Name : SSCRN Homepage	yes	PASS		00:00:12.638	20170402 19:41:28.973 20170402 19:41:41.611
Test.Step 2 : Verify value : Customer information page	yes	PASS		00:00:08.396	20170402 19:41:41.612 20170402 19:41:50.008
Test.Step 3 : Search Empty Main Promotion	yes	FAIL	Keyword 'Verify Package Main promotion' failed after retrying for 10 seconds. The  Please enter your search conditions.	00:00:13.053	20170402 19:41:50.009 20170402 19:42:03.062

รูปที่ 4.10 การแสดงรายละเอียดการทดสอบ Test detail

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.2.3 รายงานผลลัพธ์การทดสอบด้วย Log file

นอกจาก File Report แล้ว Robot framework จะมีการ Generate log file ทั้งกรณี มีข้อผิดพลาด และไม่มีข้อผิดพลาด โดยรายละเอียดภายใน log จะแสดงผลอย่างละเอียด โดย หากพบข้อผิดพลาดในการทดสอบ จะแสดงด้วยสัญลักษณ์ หรือตัวอักษรสีแดง หากทำงานได้ถูกต้องจะแสดงเป็นสีเขียว ดังตัวอย่างที่ 4.11 และภายในรายงานสามารถแสดงรูปภาพที่สั่งจับภาพหน้าจอ (Capture) ไปได้ ดังตัวอย่างที่ 4.12

#### Test Execution Log

<div style="display: flex; justify-content: space-between;"> <span>[-] SUITE Test</span> <span>00:00:56.946</span> </div>	
Full Name:	
Source:	
Start / End / Elapsed: 20170402 19:41:28.459 / 20170402 19:42:25.405 / 00:00:56.946	
Status: 12 critical test, 11 passed, 1 failed 12 test total, 11 passed, 1 failed	
-----	
[+] TEST	Step 1 : Verify Customer Name : SSCRN Homepage 00:00:12.638
-----	
[+] TEST	Step 2 : Verify value : Customer information page 00:00:08.396
-----	
[-] TEST Step 3 : Search Empty Main Promotion 00:00:13.053	
Full Name: Test.Step 3 : Search Empty Main Promotion	
Timeout: 1 minute 30 seconds	
Start / End / Elapsed: 20170402 19:41:50.009 / 20170402 19:42:03.062 / 00:00:13.053	
Status: FAIL (critical)	
Message: Keyword 'Verify Package Main promotion' failed after retrying for 10 seconds. The last error was: [BOS] [REDACTED] != Please enter your search conditions.	
[+] KEYWORD	Butn.Wait Until Keyword Succeeds 30s, 2s, Click Button Change Main promotion 00:00:00.199
[+] KEYWORD	Butn.Wait Until Keyword Succeeds 30s, 2s, Enter Empty Main Promotion 00:00:02.283
[-] KEYWORD Butn.Wait Until Keyword Succeeds 10s, 2s, Verify Package Main promotion, Please enter your search conditions. 00:00:10.569	

รูปที่ 4.11 การแสดงผลของ File log ด้วยเครื่องมือ Robot Framework

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

+	<b>KEYWORD</b>	Build.Wait Until Keyword Succeeds 30s, 2s, Click Button Change Main promotion	00:00:00.199
+	<b>KEYWORD</b>	Build.Wait Until Keyword Succeeds 30s, 2s, Enter Empty Main Promotion	00:00:02.283
-	<b>KEYWORD</b>	Build.Wait Until Keyword Succeeds 10s, 2s, Verify Package Main promotion, Please enter your search conditions.	00:00:10.569
	Documentation:	Runs the specified keyword and retries if it fails.	
	Start / End / Elapsed:	20170402 19:41:52.493 / 20170402 19:42:03.062 / 00:00:10.569	
-	<b>KEYWORD</b>	Verify Package Main promotion Please enter your search conditions.	00:00:00.133
	Start / End / Elapsed:	20170402 19:41:52.494 / 20170402 19:41:52.627 / 00:00:00.133	
-	<b>KEYWORD</b>	\$(MainPackage) = Selenium2Library.Get Text css=div.box-pac-row1-col > span	00:00:00.133
	Documentation:	Returns the text value of element identified by 'locator'.	
	Start / End / Elapsed:	20170402 19:41:52.494 / 20170402 19:41:52.627 / 00:00:00.133	
-	<b>KEYWORD</b>	Selenium2Library.Capture Page Screenshot	00:00:00.097
	Documentation:	Takes a screenshot of the current page and embeds it into the log.	
	Start / End / Elapsed:	20170402 19:41:52.529 / 20170402 19:41:52.626 / 00:00:00.097	
	19:41:52.626	INFO	



รูปที่ 4.12 การบันทึกผลด้วยรูปภายใน File log

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

# สรุปผลการวิจัยและข้อเสนอแนะ

### 5.1 สรุปผลการวิจัย

- 1) สามารถพัฒนาซอฟต์แวร์ทดสอบอัตโนมัติเพื่อใช้สำหรับทดสอบระบบ SSCRN ได้สำเร็จตามเป้าหมายที่ตั้งไว้
- 2) ซอฟต์แวร์ทดสอบอัตโนมัติสามารถทดสอบความถูกต้องในการทำงาน และการแสดงผลของ SSCRN ได้จริง อีกทั้งยังสามารถนำไปประยุกต์ใช้กับชุดทดสอบอื่นๆ ของ SSCRN ได้
- 3) การนำซอฟต์แวร์ทดสอบอัตโนมัติมาใช้แทนการทดสอบด้วยมนุษย์ช่วยลดเวลาที่ใช้ในการทดสอบลงอย่างมาก ส่งผลให้สามารถหาข้อผิดพลาดได้อย่างรวดเร็ว และทดสอบระบบได้อย่างครอบคลุมทุกกรณี และทุกเว็บเบราว์เซอร์
- 4) กรณีทดสอบอยู่ในรูปแบบของภาษาที่เข้าใจง่าย ทำให้เข้าใจข้อผิดพลาดที่เกิดขึ้นได้ทันที และช่วยให้ผู้ที่ไม่ใช่ Developer สามารถเข้าใจได้ว่ากรณีทดสอบแต่ละกรณีเป็นการทดสอบอะไร และสามารถเขียนกรณีทดสอบเพิ่มเติมด้วยตัวเองได้

### 5.2 ปัญหาและข้อจำกัด

- 1) ความเร็วของเครือข่าย และความหนาแน่นของเซิร์ฟเวอร์มีผลต่อการทดสอบ เนื่องจากส่วนที่ต้องการจะทดสอบอาจจะยังโหลดไม่เสร็จ ทำให้กรณีทดสอบนั้นไม่ผ่าน
- 2) เนื่องจากการทดสอบเว็บแอปพลิเคชัน ซึ่งต้องอ้างอิงกับโครงสร้างของ HTML element เมื่อมีการเปลี่ยนแปลงโครงสร้าง HTML element หรือแก้ไข HTML attribute ต่างๆจะต้องแก้ไขกรณีทดสอบบางส่วนที่มีการอ้างอิง HTML element ดังกล่าวด้วย
- 3) โครงสร้างของ HTML บางส่วนไม่ได้ออกแบบรองรับการทดสอบอัตโนมัติ หรือไม่มีค่า HTML attribute ที่มีความเป็นเอกลักษณ์ ทำให้เขียน xpath เพื่อเข้าถึงส่วนที่ต้องการได้ยาก และต้องอ้างอิง HTML element อื่น เกินความจำเป็น ทำให้ xpath ไม่มีความยืดหยุ่น และกลับมาแก้ไขทุกครั้งที่โครงสร้างของ HTML element ที่อ้างอิงมีการเปลี่ยนแปลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 5.3 ข้อเสนอแนะ

- 1) ผู้พัฒนาเว็บแอปพลิเคชันควรออกแบบให้ HTML element ต่างๆ มี attribute ที่สามารถใช้อ้างอิงได้ และมีความเป็นเอกลักษณ์ เพื่อให้สามารถอ้างอิงถึง element ที่จะตรวจสอบได้โดยตรง เพื่อหลีกเลี่ยงการที่ต้องกลับมาแก้ไขซอฟต์แวร์ทดสอบอัตโนมัติทุกครั้งที่มีการเปลี่ยนแปลงโครงสร้าง HTML
- 2) ก่อนจะสร้าง Keyword ที่ทำงานด้วยเครื่องมือ Robot framework ผู้พัฒนาซอฟต์แวร์ทดสอบอัตโนมัติควรตรวจสอบก่อนว่ามี Keyword ที่ทำงานคล้ายกันอยู่แล้วหรือไม่ เพื่อหลีกเลี่ยงการพัฒนาซอร์สโค้ดที่ซ้ำซ้อน
- 3) สามารถต่อยอดการพัฒนาซอฟต์แวร์ทดสอบอัตโนมัติได้โดยเพิ่มการทำงาน ให้เชื่อมต่อกับเครื่องมือติดตามข้อผิดพลาด เช่น JIRA ซึ่งจะทำให้ผู้ทดสอบไม่ต้องทำการเปิด Defect เองจะสามารถทำได้โดยให้ซอฟต์แวร์ทดสอบอัตโนมัติทำงานแทน

## เอกสารอ้างอิง

- [1] รู้หรือไม่ว่า Verification กับ Validation เหมือนหรือต่างกันอย่างไร?. [Online]. Available: <http://www.fusionidea.biz/%E0%B8%A3%E0%B8%B9%E0%B9%89%E0%B8%AB%E0%B8%A3%E0%B8%B7%E0%B8%AD%E0%B9%84%E0%B8%A1%E0%B9%88%E0%B8%A7%E0%B9%88%E0%B8%B2-verification-%E0%B8%81%E0%B8%B1%E0%B8%9A-validation-%E0%B9%80%E0%B8%AB%E0%B8%A1/>
- [2] แนวทางการทำ Load Testing แบบง่ายๆ. [Online]. Available: <http://www.somkiat.cc/think-before-load-testing/>
- [3] Automated Software Testing. [Online]. Available: [http://www.tutorialspoint.com/software\\_testing\\_dictionary/automate\\_software\\_testing.htm](http://www.tutorialspoint.com/software_testing_dictionary/automate_software_testing.htm)
- [4] Why Automated Testing?. [Online]. Available: [http://wcm.g-able.com/portal/page/portal/g-able/thai/it\\_talks/Y2013/it\\_talks\\_V35\\_01/G-Magz\\_V35\\_1.pdf](http://wcm.g-able.com/portal/page/portal/g-able/thai/it_talks/Y2013/it_talks_V35_01/G-Magz_V35_1.pdf)
- [5] รศ.รังสิต ศิริรังษี. 2557. การทดสอบฟังก์ชันอัตโนมัติด้วย Selenium. ครั้งที่พิมพ์ 1. เชียงใหม่
- [6] Robot Framework เบื้องต้น [Online]. Available: <http://www.stream.co.th/2016/02/ทำความเข้าใจกับ-robot-framework-เป็น/>
- [7] ATDD คืออะไร. [Online]. Available: <https://swiftlet.co.th/atdd-คืออะไร/>
- [8] Agile Thailand. Acceptance Test Driven Development (ATDD). [Online]. Available : <https://agilethailand.wordpress.com/2012/02/07/acceptance-test-driven-development-atdd/>
- [9] Test-driven development. [Online]. Available : [http://en.wikipedia.org/wiki/Test-driven\\_development](http://en.wikipedia.org/wiki/Test-driven_development)
- [10] การค้นหาส่วนประกอบของ HTML รศ.รังสิต ศิริรังษี. 2557. การทดสอบฟังก์ชันอัตโนมัติด้วย Selenium. ครั้งที่พิมพ์ 1. เชียงใหม่
- [11] Java คืออะไร. [Online]. Available : <http://www.mindphp.com>
- [12] ภาษาไพทอน [Online]. Available : <https://th.wikipedia.org/wiki/ภาษาไพทอน>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- [13] ข้อดีของภาษาไพทอน [Online]. Available : <http://www.mindphp.com/คู่มือคือ-73/-อะไร/2417python-คืออะไร.html>
- [14] XML คืออะไร [Online]. Available : <http://www.mindphp.com/คู่มือ/73-คืออะไร/2129-xml-คืออะไร.html>
- [15] CRM คืออะไร [Online]. Available : <http://www.mindphp.com/%E0%B8%84%E0%B8%B9%E0%B9%88%E0%B8%A1%E0%B8%B7%E0%B8%AD/73-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3/2072-crm-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3.html>
- [16] เว็บเซอร์วิส (Web Service). [Online]. Available : <http://www.oknation.net/blog/Siraprapa/2010/08/30/entry-2>
- [17] Jira. [Online]. Available : <http://it-madmonster.blogspot.com/2009/07/jira-overview.html>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ภาคผนวก ก

## การติดตั้ง และการใช้งาน

### โปรแกรม Robot Framework

Robot Framework คือซอฟต์แวร์ Open Source ที่สำหรับการทำ Acceptance Testing หรือการทดสอบการทำงานของระบบที่สร้างขึ้น ให้ง่ายและรวดเร็วด้วยการทำ Automate Testing และ ATDD (Acceptance Test-Driven Development) โดยมีรูปแบบ Syntax ที่เป็นภาษาเขียนธรรมดา คนทั่วไปสามารถทำความเข้าใจได้ง่ายแม้ไม่ใช่ Developer

#### วิธีติดตั้ง Robot Framework

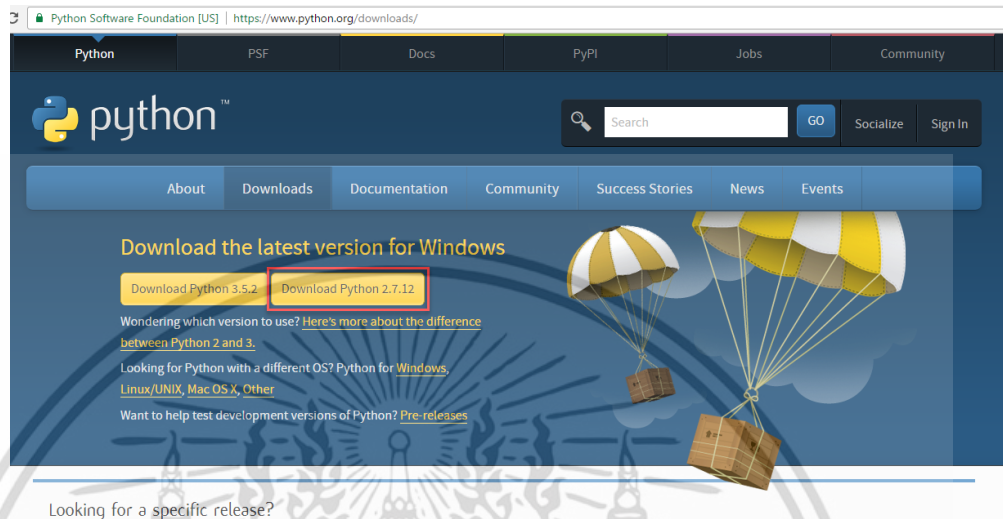
Software ที่ใช้ในการติดตั้งประกอบไปด้วย

- Python 2.7.12
- Text editor ที่ใช้เขียน script ของ Robot framework เช่น Sublime
- Default browser คือ Firefox

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

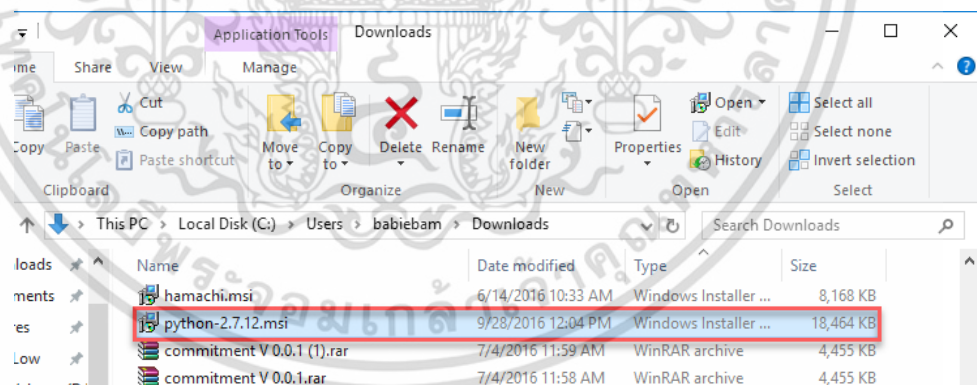
## การติดตั้ง Python

1. Download Python for Window จาก <https://www.python.org/downloads> > เลือก Download Python for Windows version 2.7.12 ( Version ล่าสุด ณ วันที่ 29/09/2016 )



รูปที่ ก.1 แสดงการ Download Python

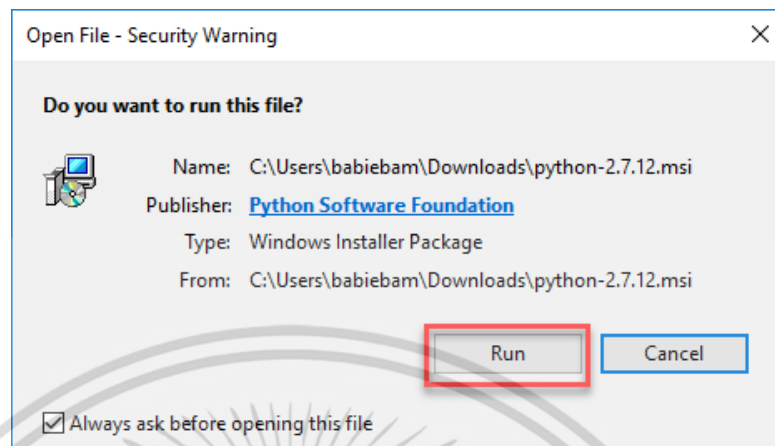
2. หลังจาก Download เสร็จ จะได้ไฟล์ python-2.7.12.msi



รูปที่ ก.2 แสดงไฟล์ python ที่ทำการ Download เสร็จ

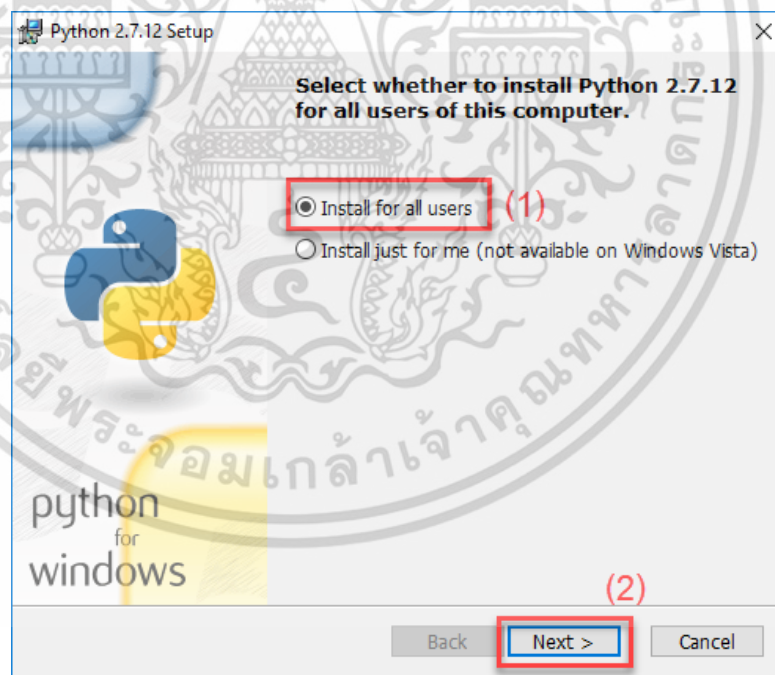
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. จากนั้นสามารถ Double Click > Run เพื่อทำการติดตั้ง



รูปที่ ก.3 แสดงการเลือก Run เพื่อติดตั้ง

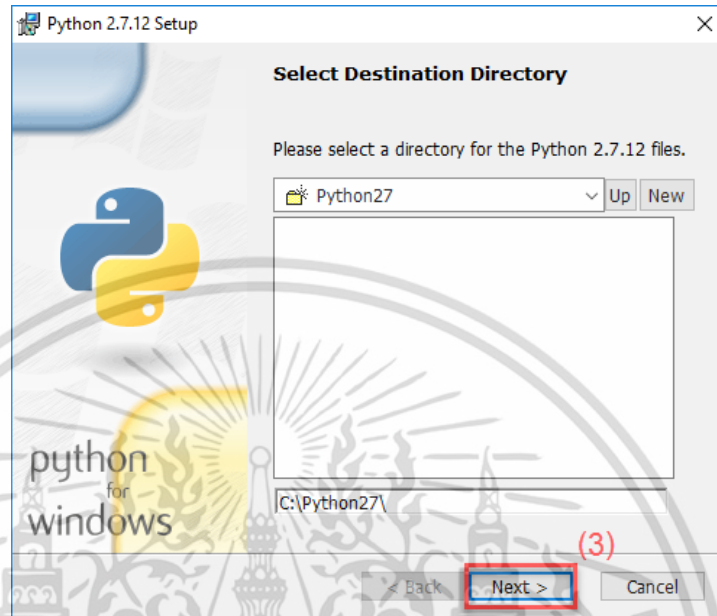
4. เลือกการติดตั้งตามขั้นตอน > Install for all users > Next



รูปที่ ก.4 แสดงการ Install (1) และ (2)

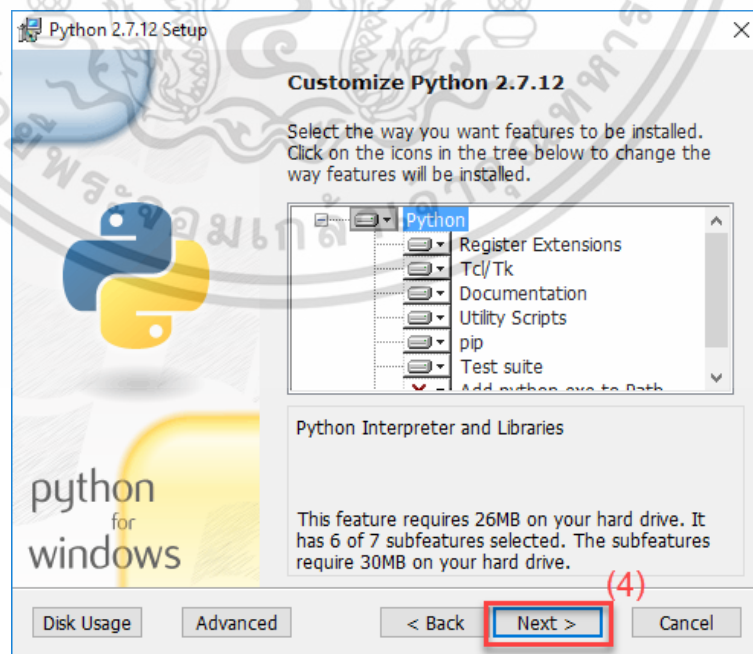
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. เลือก directory ที่ต้องการจะติดตั้ง จากภาพจะเป็นการเลือกติดตั้งที่ C:\Python27\



รูปที่ ก.5 แสดงการเลือก directory ที่ต้องการติดตั้ง (3)

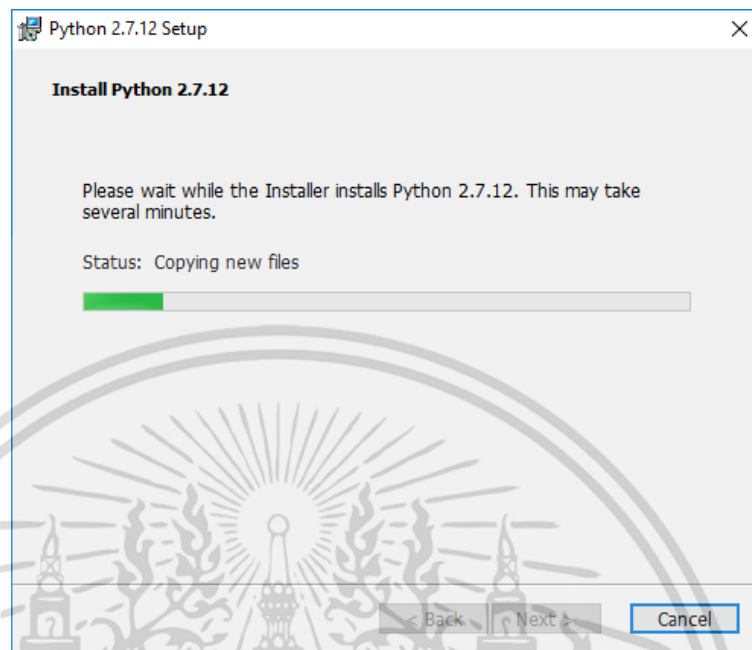
6. จากนั้นเลือก Features ที่ต้องการจะติดตั้ง เลือก Python > Next



รูปที่ ก.6 แสดงการเลือก features (4)

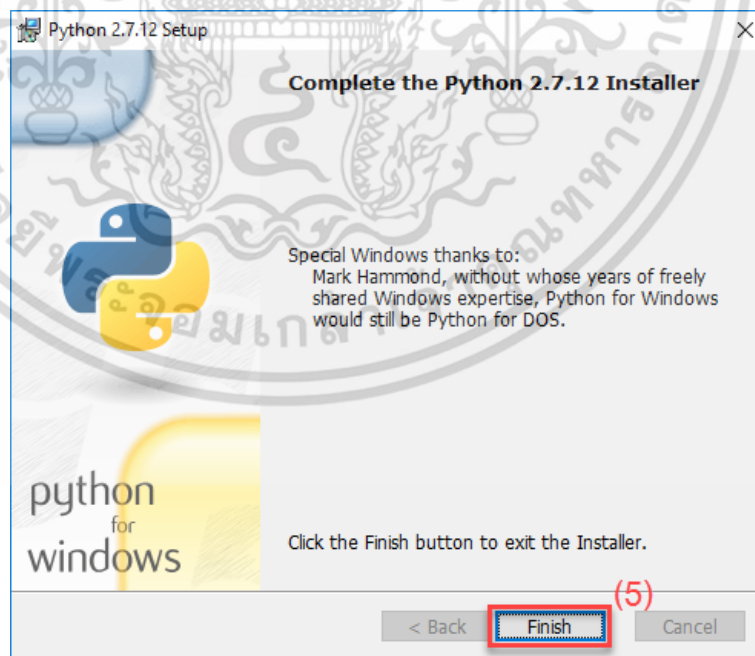
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 7. รอจนทำการติดตั้งเสร็จ



รูปที่ ก.7 แสดงการรอติดตั้ง Python

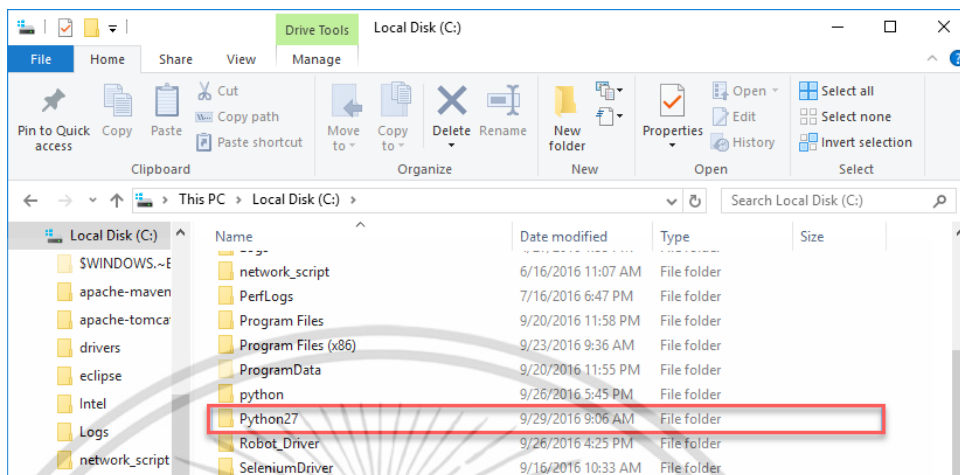
## 8. เมื่อโหลดเสร็จเรียบร้อย จากนั้นกดปุ่ม Finish เพื่อทำการออก



รูปที่ ก.8 แสดงการติดตั้ง (5)

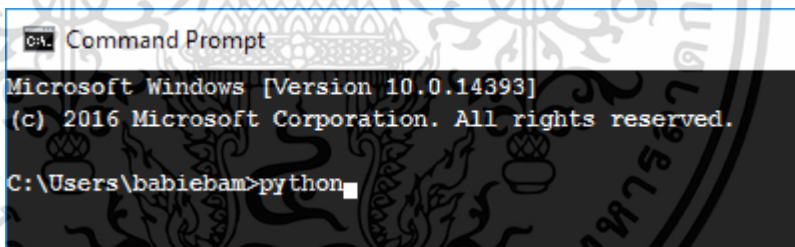
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. หลังจากติดตั้งเสร็จ Python จะอยู่ที่ c:\Python27



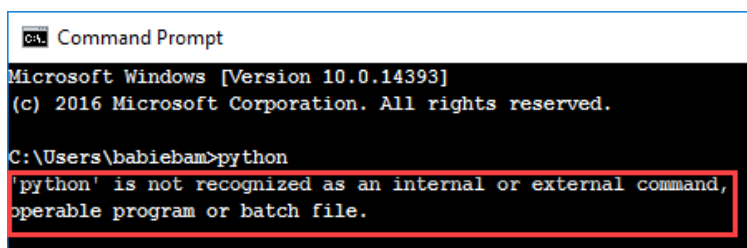
รูปที่ ก.9 หลังจากติดตั้ง Python เสร็จ

10. ทดสอบว่า python สามารถทำงานผ่าน command line ได้หรือไม่ ด้วยการ ไปที่ Start > พิมพ์ในช่องว่างว่า cmd > เลือกที่ cmd > พิมพ์คำว่า python ลงไป



รูปที่ ก.10 แสดงการทดสอบ Python ผ่าน command line

11. ถ้าผลการทำงานเกิด Error จะแสดงข้อความว่า python is not recognized as an internal or external command



รูปที่ ก.11 แสดงกรณีผลการทำงาน Error

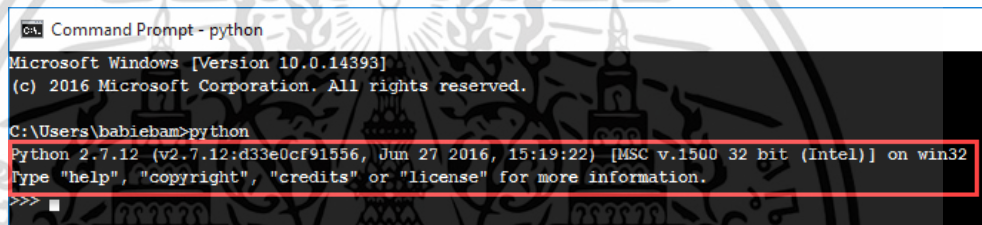
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

12. กรณีผลการทำงานเกิด Error ให้ทำการแก้ไข คือทำการกำหนด Environment variable แบบชั่วคราวใน command line หรือ กำหนดแบบถาวร สำหรับการกำหนดแบบชั่วคราวให้พิมพ์คำสั่งดังนี้

```
set PYTHON_HOME=c:\Python27
set PATH=;%PYTHON_HOME%;%PYTHON_HOME%\scripts;%PATH%
```

รูปที่ ก.12 แสดงการกำหนด Environment variable แบบชั่วคราว (1)

13. จากนั้นให้พิมพ์คำสั่ง python อีกครั้ง บน Command line จะไม่แสดง error อีก

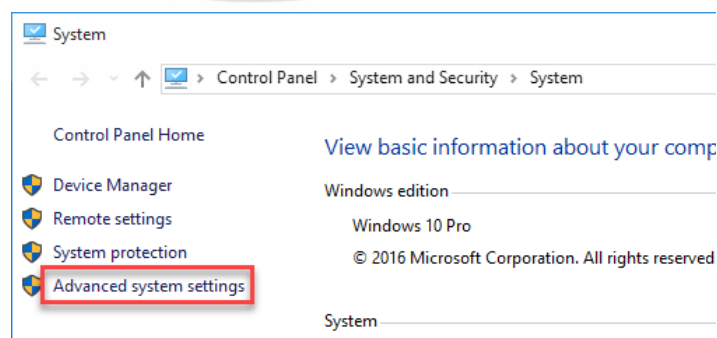


```
Command Prompt - python
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Users\babiebam>python
Python 2.7.12 (v2.7.12:d33e0cf91556, Jun 27 2016, 15:19:22) [MSC v.1500 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

รูปที่ ก.13 แสดงการกำหนด Environment variable แบบชั่วคราว (2)

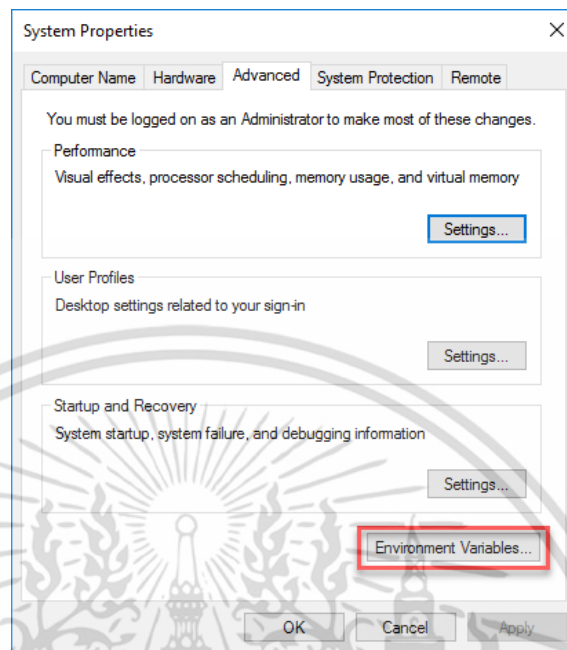
14. แต่เมื่อทำการปิด command line ไป แล้วเปิดขึ้นมาใหม่ จะเกิด error เช่นเดิม เนื่องจากการกำหนด Environment variable แบบชั่วคราวเท่านั้น
15. ดังนั้น สามารถทำการกำหนด Environment variable แบบถาวร ได้โดยเข้าแก้ไขโดยไปที่ System Properties เข้าไปด้วย Click ขวาที่ Control Panel -> เลือก System Security



รูปที่ ก.14 แสดงการกำหนด Environment variable (1)

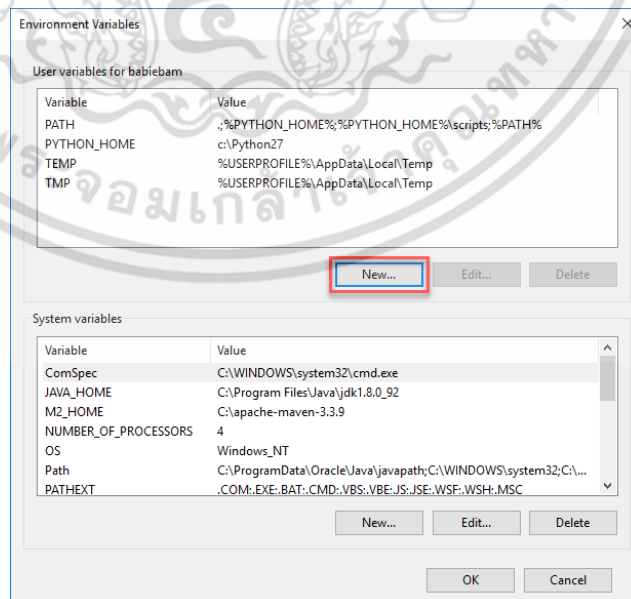
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

16. เลือก tab Advance- > Environment Variable



รูปที่ ก.15 แสดงการกำหนด Environment variable (2)

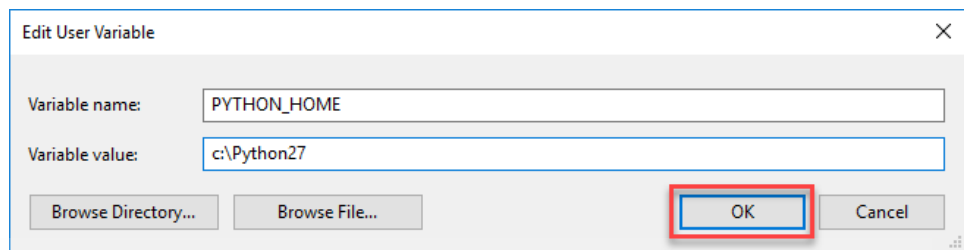
17. ให้ทำการสร้าง Environment Variables ในส่วนของ User Variable โดยการกด New เพื่อสร้าง



รูปที่ ก.16 แสดงการกำหนด Environment variable (3)

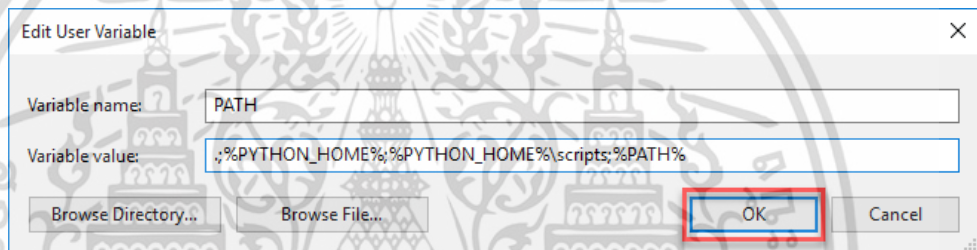
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

18. จากนั้นทำการกำหนด PYTHON\_HOME ดังนี้



รูปที่ ก.17 แสดงการกำหนด Environment variable (4)

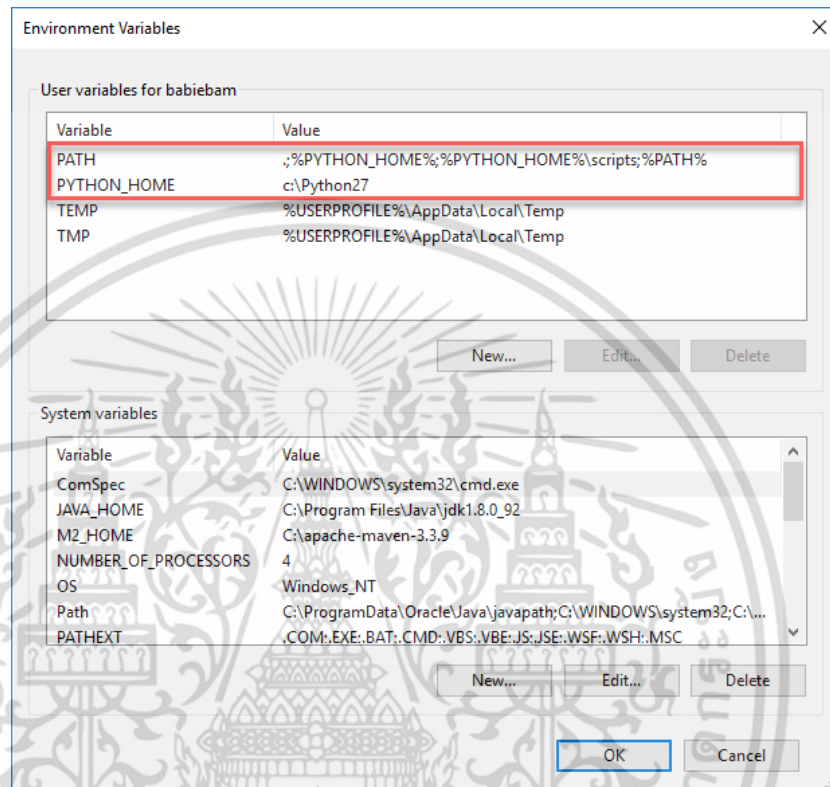
19. ทำการกำหนด PATH ด้วยการตั้งค่าดังรูปภาพที่



รูปที่ ก.18 แสดงการกำหนด Environment variable (5)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

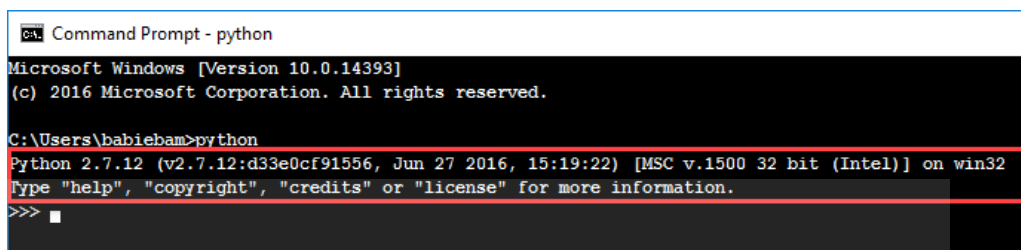
20. จะได้ PATH และ PYTHON\_HOME ดังรูปภาพ จากนั้นกดปุ่ม OK และกลับสู่ Desktop Windows



รูปที่ ก.19 แสดงการกำหนด Environment variable (6)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

21. ทดสอบอีกครั้งด้วยการเปิด command line ขึ้นมาใหม่ -> พิมพ์คำสั่ง python จะไม่เกิด error อีกแล้ว



```

C:\Users\babiebam>python
Python 2.7.12 (v2.7.12:d33e0cf91556, Jun 27 2016, 15:19:22) [MSC v.1500 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>

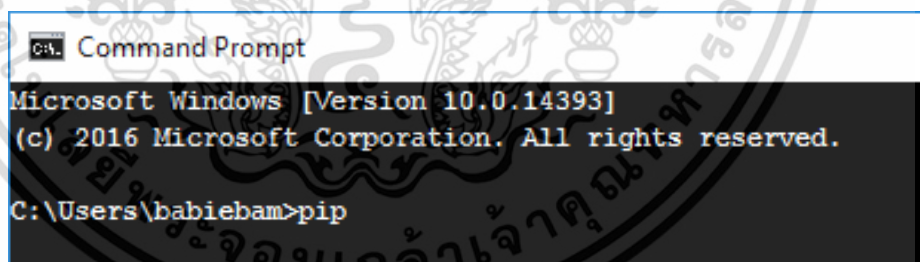
```

รูปที่ ก.20 แสดงการทดสอบ Python ผ่าน command line

22. ทำการทดสอบใช้งาน pip ใน command line

Pip คือตัวจัดการ package/library ต่าง ๆ สำหรับภาษา Python โดย pip จะมีมา กับ Python เวอร์ชัน 2.7.9 เป็นต้นมา ดังนั้นไม่จำเป็นต้องทำการติดตั้ง pip เองแต่อย่างใด ซึ่ง pip จะอยู่ใน folder %PYTHON\_HOME%\Scripts

23. ทำการเปิด command line แล้วพิมพ์คำสั่ง pip



```

C:\Users\babiebam>pip

```

รูปที่ ก.21 แสดงการพิมพ์คำสั่ง pip ใน command line

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

24. หากติดตั้ง pip สำเร็จ จะต้องไม่เกิด error ใด ๆ ขึ้นมาแสดงผลการทำงานดังนี้

```

C:\> Command Prompt
Usage:
  pip <command> [options]

Commands:
  install           Install packages.
  download          Download packages.
  uninstall         Uninstall packages.
  freeze            Output installed packages in requirements format.
  list              List installed packages.
  show              Show information about installed packages.
  search            Search PyPI for packages.
  wheel             Build wheels from your requirements.
  hash              Compute hashes of package archives.
  completion        A helper command used for command completion
  help              Show help for commands.

General Options:
  -h, --help          Show help.
  --isolated          Run pip in an isolated mode, ignoring
                     environment variables and user configuration.
  -v, --verbose       Give more output. Option is additive, and can be
                     used up to 3 times.
  -V, --version       Show version and exit.
  -q, --quiet         Give less output.
  --log <path>       Path to a verbose appending log.
  --proxy <proxy>    Specify a proxy in the form
                     [user:passwd@]proxy.server:port.
  --retries <retries> Maximum number of retries each connection should
                     attempt (default 5 times).
  --timeout <sec>    Set the socket timeout (default 15 seconds).
  --exists-action <action> Default action when a path already exists:
                     (s)witch, (i)gnore, (w)ipe, (b)ackup.
  --trusted-host <hostname> Mark this host as trusted, even though it does
                     not have valid or any HTTPS.
  --cert <path>      Path to alternate CA bundle.
  --client-cert <path> Path to SSL client certificate, a single file
                     containing the private key and the certificate
                     in PEM format.
  --cache-dir <dir>  Store the cache data in <dir>.
  --no-cache-dir     Disable the cache.
  --disable-pip-version-check
                     Don't periodically check PyPI to determine
                     whether a new version of pip is available for
                     download. Implied with --no-index.

```

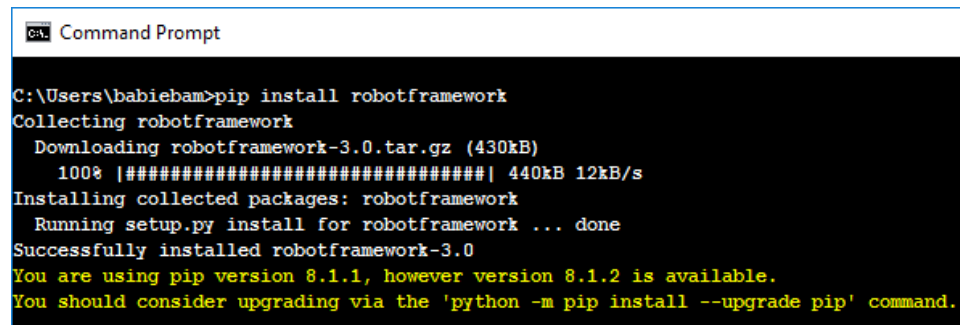
รูปที่ ก.22 แสดงผลการทำงานของคำสั่ง pip

25. ทำการติดตั้ง Robot Framework ผ่าน pip โดยทำการติดตั้ง robot framework ซึ่งเป็น library หนึ่งที่พัฒนาด้วยภาษา Python สำหรับการติดตั้ง จะติดตั้งผ่าน pip ใน command line ด้วยคำสั่ง pip install robotframework ดังนี้

```
pip install robotframework
```

รูปที่ ก.23 แสดงคำสั่งการติดตั้ง robot framework ด้วย pip

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



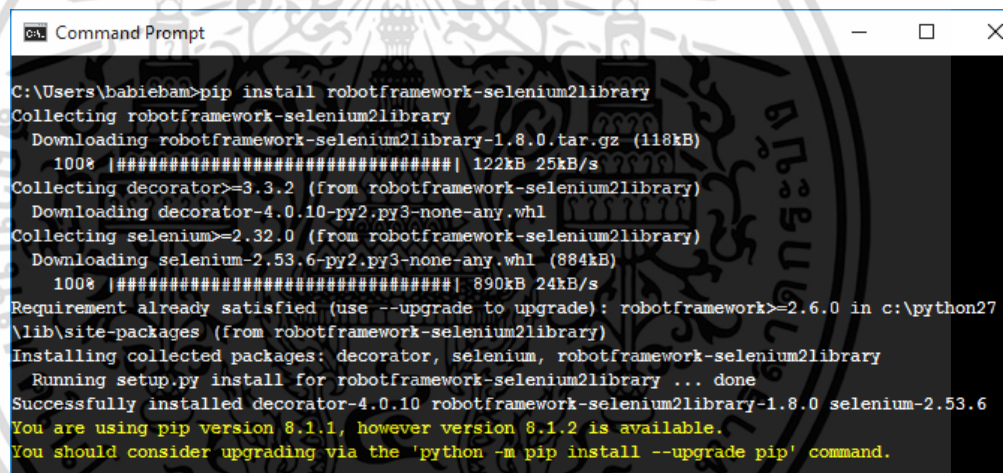
```

C:\Users\babiebam>pip install robotframework
Collecting robotframework
  Downloading robotframework-3.0.tar.gz (430kB)
    100% |#####| 440kB 12kB/s
Installing collected packages: robotframework
  Running setup.py install for robotframework ... done
Successfully installed robotframework-3.0
You are using pip version 8.1.1, however version 8.1.2 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.

```

รูปที่ ก.24 แสดงการทำงานหลังการติดตั้ง pip

26. ต่อมาทำการติดตั้ง library สำหรับการทดสอบระบบ web application ชื่อว่า robotframework-selenium2library ด้วยคำสั่ง pip install robotframework-selenium2library



```

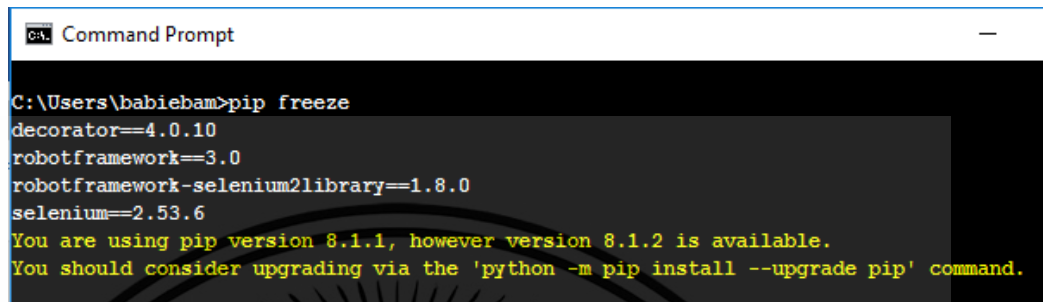
C:\Users\babiebam>pip install robotframework-selenium2library
Collecting robotframework-selenium2library
  Downloading robotframework-selenium2library-1.8.0.tar.gz (118kB)
    100% |#####| 122kB 25kB/s
Collecting decorator>=3.3.2 (from robotframework-selenium2library)
  Downloading decorator-4.0.10-py2.py3-none-any.whl
Collecting selenium>=2.32.0 (from robotframework-selenium2library)
  Downloading selenium-2.53.6-py2.py3-none-any.whl (884kB)
    100% |#####| 890kB 24kB/s
Requirement already satisfied (use --upgrade to upgrade): robotframework>=2.6.0 in c:\python27\lib\site-packages (from robotframework-selenium2library)
Installing collected packages: decorator, selenium, robotframework-selenium2library
  Running setup.py install for robotframework-selenium2library ... done
Successfully installed decorator-4.0.10 robotframework-selenium2library-1.8.0 selenium-2.53.6
You are using pip version 8.1.1, however version 8.1.2 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.

```

รูปที่ ก.25 แสดงคำสั่งการติดตั้ง library

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

27. ตรวจสอบการติดตั้ง selenium2library ด้วยการพิมพ์คำว่า pip freeze หลังจากนั้น จะมี list แสดงขึ้นว่าเราติดตั้ง robotframework-selenium2library ใน environment นี้หรือไม่



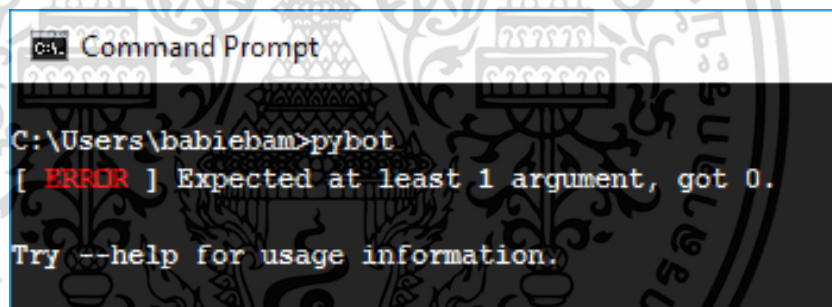
```

C:\Users\babiebam>pip freeze
decorator==4.0.10
robotframework==3.0
robotframework-selenium2library==1.8.0
selenium==2.53.6
You are using pip version 8.1.1, however version 8.1.2 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.

```

รูปที่ ก.26 แสดงการตรวจสอบการติดตั้ง selenium2library

28. ทำการตรวจสอบผลการติดตั้ง robot framework ด้วยคำสั่ง pybot



```

C:\Users\babiebam>pybot
[ ERROR ] Expected at least 1 argument, got 0.
Try --help for usage information.

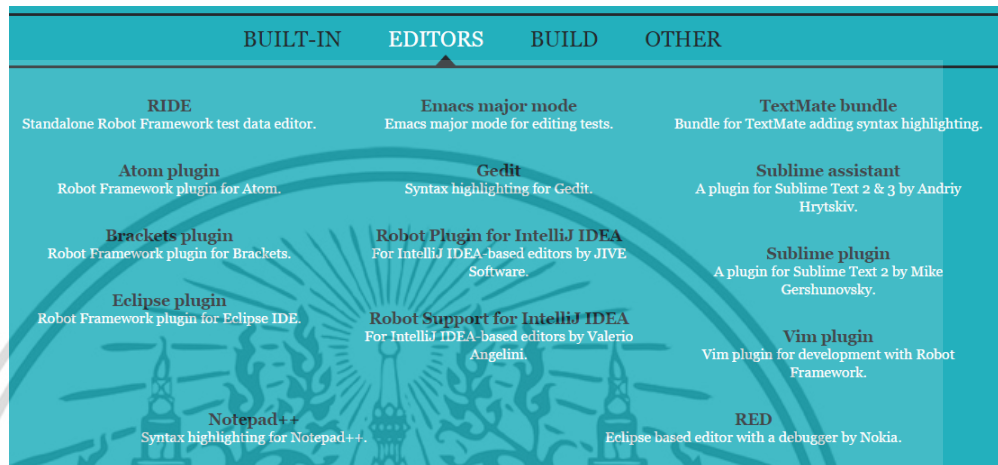
```

รูปที่ ก.27 แสดงการตรวจสอบผลการติดตั้ง robot framework

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## การติดตั้ง Text Editor : Sublime

การเขียน Test case สามารถใช้ Text editor ได้หลายตัว โดยสามารถเข้าไปดูได้ที่เว็บไซต์ของ Robot framework ด้วยลิงค์ต่อไปนี้ <http://robotframework.org/#tools>



รูปที่ ก.28 แสดงตัวอย่าง Text editor (1)

สำหรับตัวอย่างต่อไปนี้จะเป็นการติดตั้ง Text editor คือ Sublime เนื่องจากมีพวก Syntax highlight และ auto-complete ให้ใช้งานได้สะดวก โดยสามารถเริ่มทำการติดตั้งได้ดังนี้

```

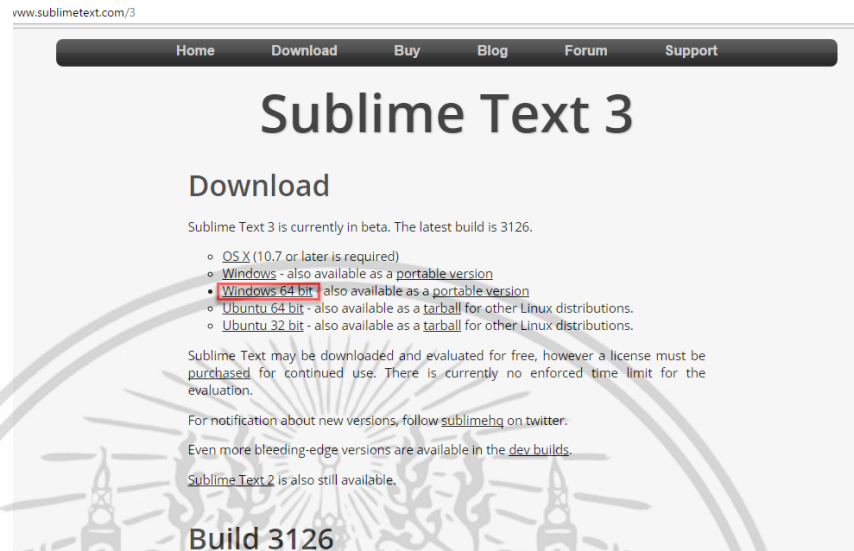
1  *** Settings ***
2  Library    Selenium2Library
3
4  *** Variables ***
5  ${HOMEPAGE}    http://www.google.com
6  ${BROWSER}    chrome
7
8  *** Test Cases ***
9  Go To homepage
10     ... Open Browser    ${HOMEPAGE}    ${BROWSER}
11
12  Capture Page
13     ... [Arguments]    ... ${OTHER_DIR}
14     ... Capture Page Screenshot    ... ${OTHER_DIR}${/}Search Main-{index:06}.png

```

รูปที่ ก.29 แสดงตัวอย่างของการใช้ Sublime ในการเขียน Robot framework (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

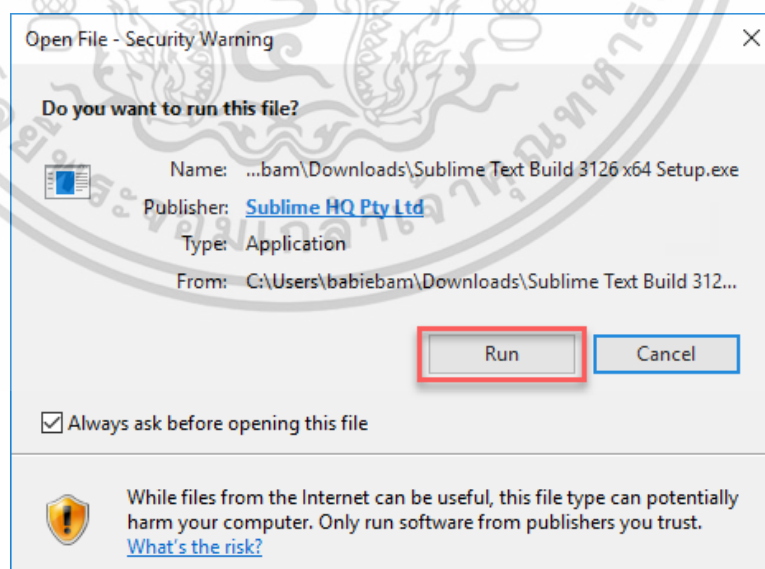
1. ทำการติดตั้ง Text Editor :Sublime สำหรับเขียน script ด้วย Robot Framework โดยเข้าไป Download ได้ที่ <https://www.sublimetext.com/3>



### Build 3126

รูปที่ ก.30 แสดงการดาวน์โหลด Sublime

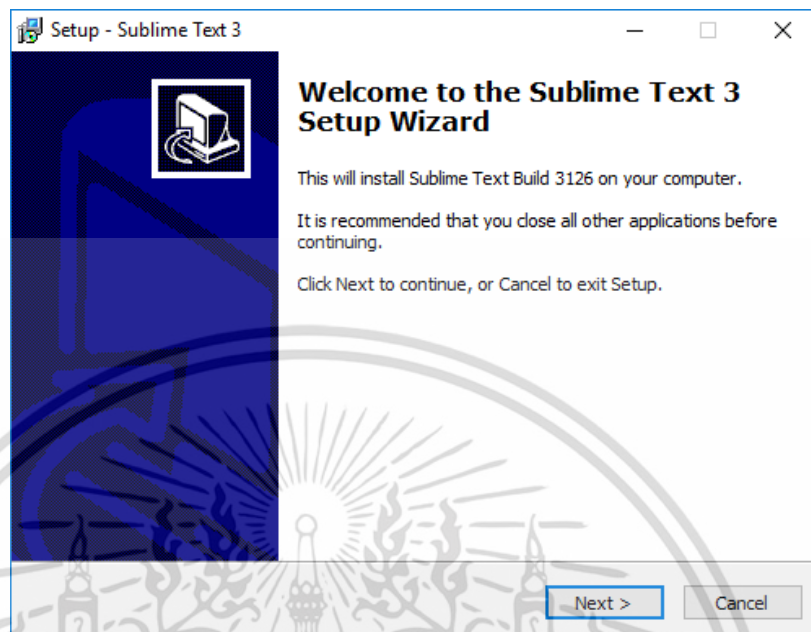
2. เมื่อดาวน์โหลดเรียบร้อยแล้วให้ทำการ Run โดยกดปุ่ม Run ดังภาพ



รูปที่ ก.31 แสดงการเลือก Run เพื่อติดตั้ง

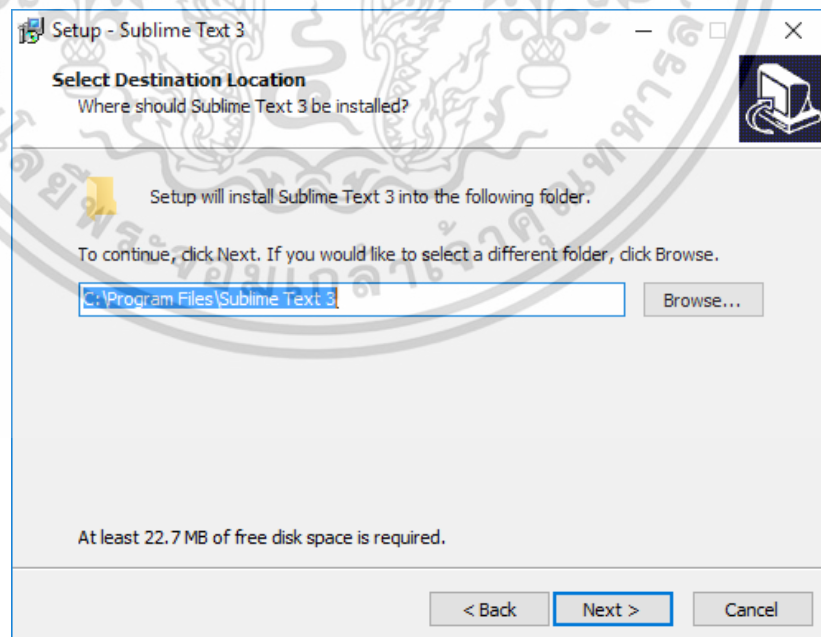
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ทำการติดตั้งตามขั้นตอนโดยเริ่มจาก กด Next เพื่อเข้าสู่ขั้นตอนการติดตั้งขั้นต่อไป



รูปที่ ก.32 แสดงการติดตั้ง Sublime(1)

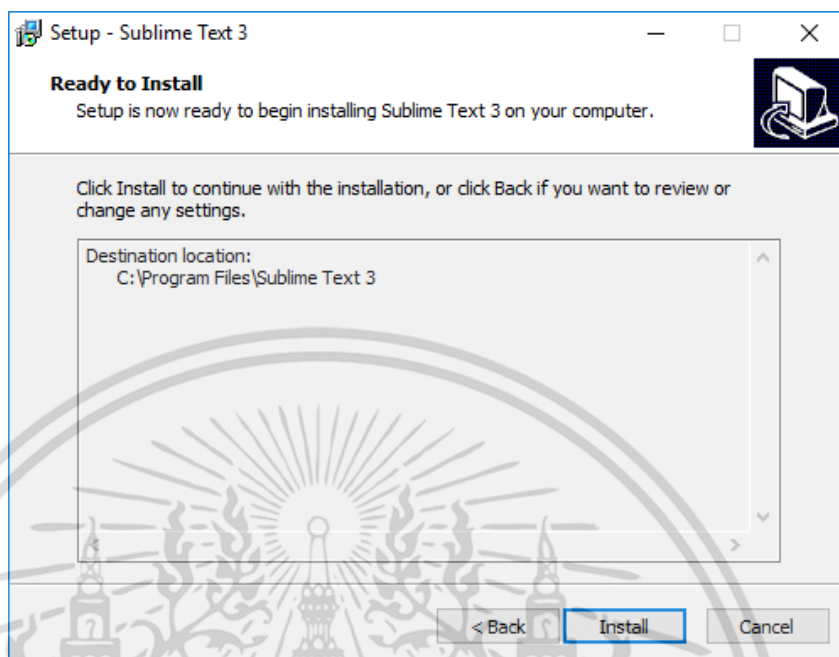
4. กำหนด Path สำหรับทำการติดตั้งโปรแกรม



รูปที่ ก.33 แสดงการติดตั้ง Sublime (2)

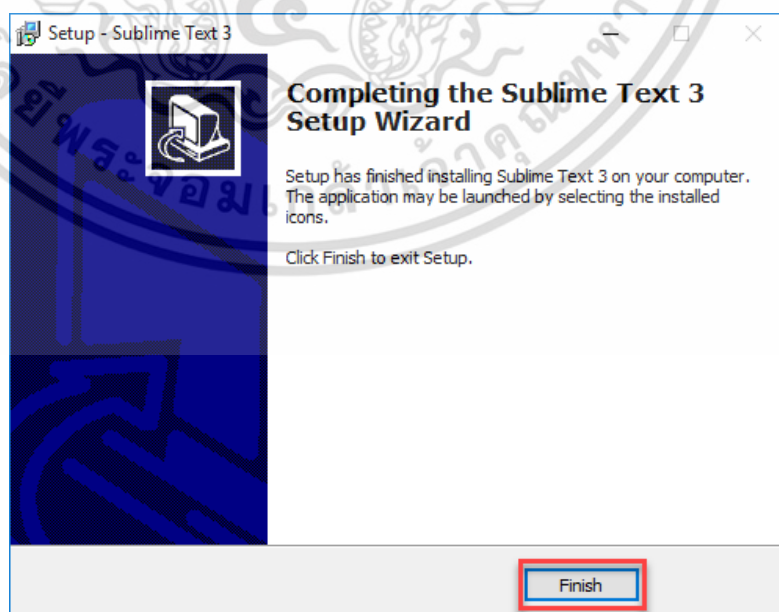
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. กด Install เพื่อไปยังขั้นตอนถัดไป



รูปที่ ก.34 แสดงการติดตั้ง Sublime(3)

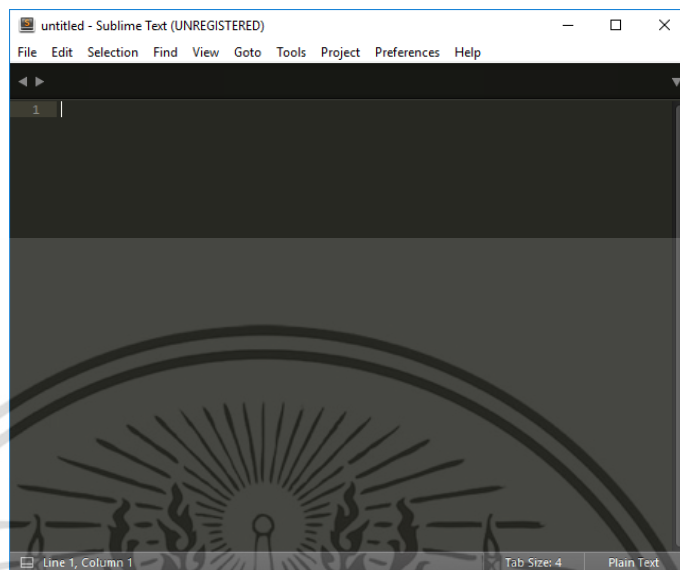
6. เมื่อโหลดเสร็จเรียบร้อย จากนั้นกดปุ่ม Finish เพื่อทำการออก



รูปที่ ก.35 แสดงการติดตั้ง Sublime(4)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. เมื่อทำการติดตั้งเสร็จเรียบร้อยแล้ว จะได้ Text Editor ที่มีหน้าตาดังภาพ



รูปที่ ก.36 แสดงการติดตั้ง Sublime(5)

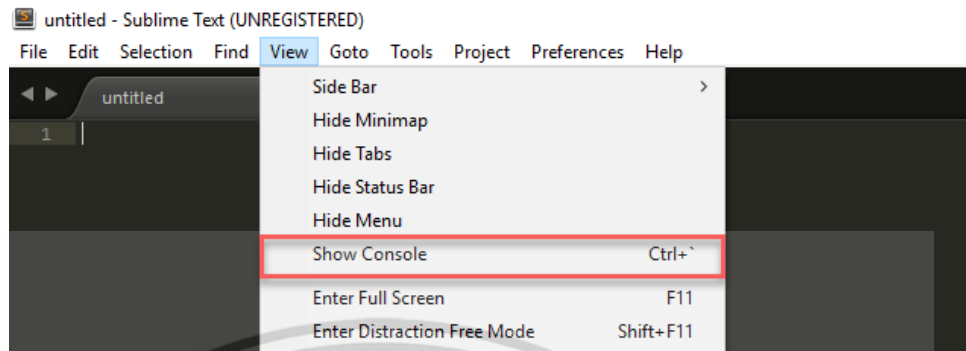
8. จากนั้นทำการติดตั้ง Package Control โดยเข้าไปคัดลอกซอร์สโค้ดจาก <https://packagecontrol.io/installation#st3>



รูปที่ ก.37 แสดงการติดตั้ง Sublime (6)

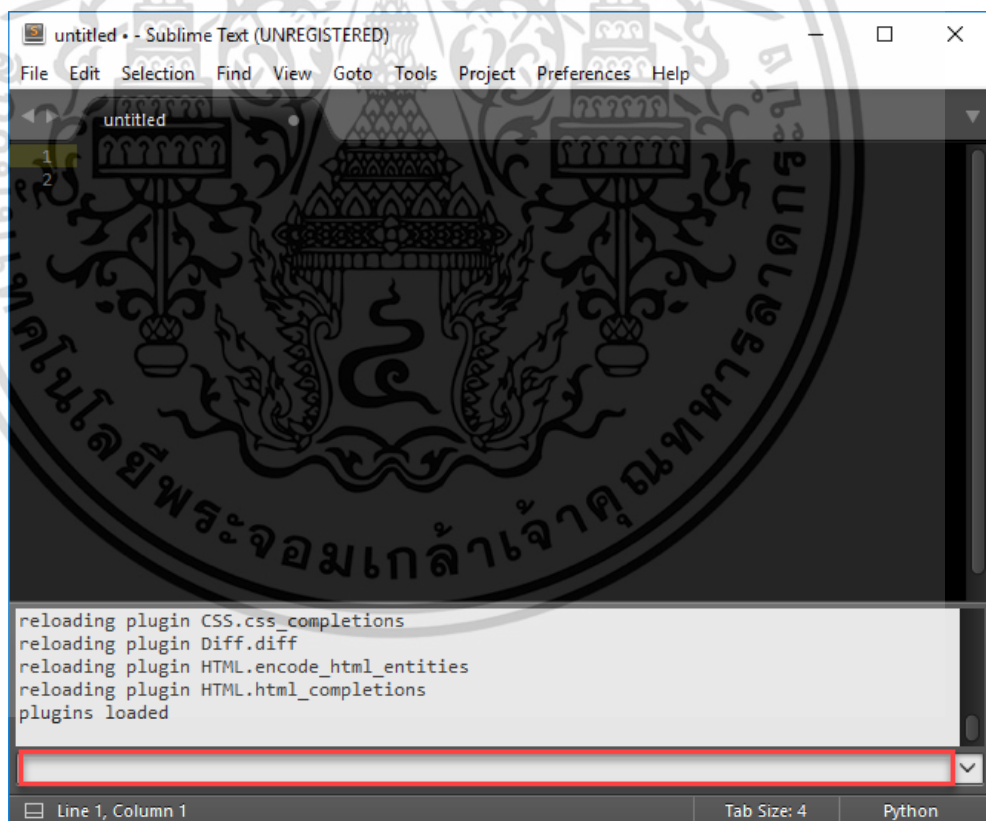
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. ทำการเปิดโปรแกรม Sublime Text ขึ้นมา จากนั้นเลือก View เลือก Show Console



รูปที่ ก.38 แสดงการติดตั้ง Sublime (7)

10. หลังจากเลือก Show Console แล้ว จะมีกล่องให้ใส่คำสั่งดังภาพ



รูปที่ ก.39 แสดง Console

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. จากนั้นนำซอร์สโค้ดที่คัดลอกไว้ใน มาวางใน Console หลังจากนั้นรอให้มันทำการติดตั้ง Package Control สักครู่

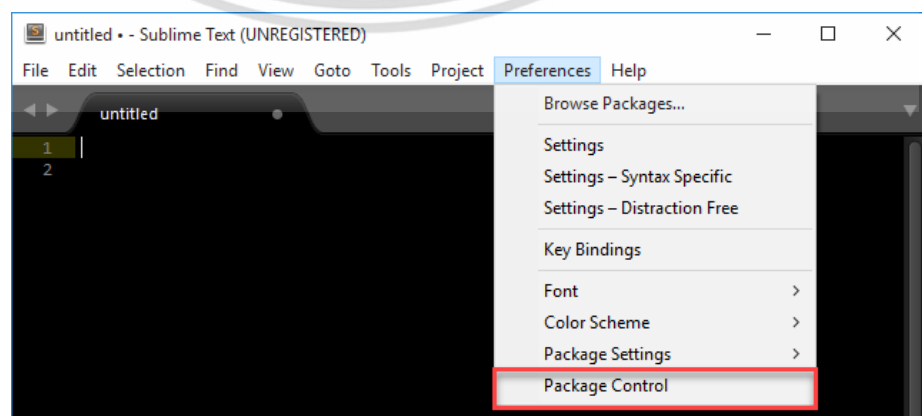
```
reloading plugin CSS.css_completions
reloading plugin Diff.diff
reloading plugin HTML.encode_html_entities
reloading plugin HTML.html_completions
plugins loaded
curl -s https://raw.githubusercontent.com/jensens/sublime-package-control/master/install.py | python -
```

รูปที่ ก.40 แสดงการนำซอร์สโค้ดที่คัดลอกไว้ใน มาวางใน Console (8)

```
Package Control: Not running package cleanup since bootstrapping is not yet complete
reloading settings Packages/Package Control/Package Control.sublime-settings
reloading settings Packages/User/Package Control.sublime-settings
reloading plugin 0_package_control_loader.00-package_control
```

รูปที่ ก.41 แสดงกาการติดตั้ง Package Control สำเร็จ

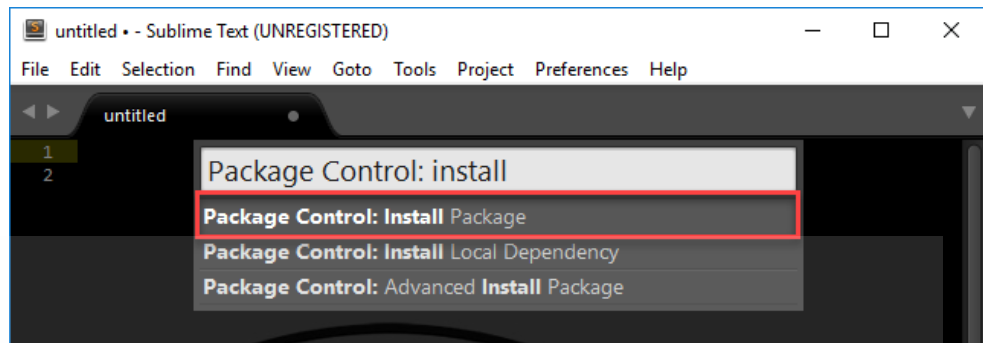
12. ทำการเปิดโปรแกรม Sublime Text ขึ้นมา จากนั้นเลือก View เลือก Show Console เมื่อ Restart Sublime เรียบร้อยแล้วขั้นตอนสุดท้าย คือการ install package Robot Framework เพื่อสามารถรันผ่าน Sublime ไปที่ Preferences > Package Control



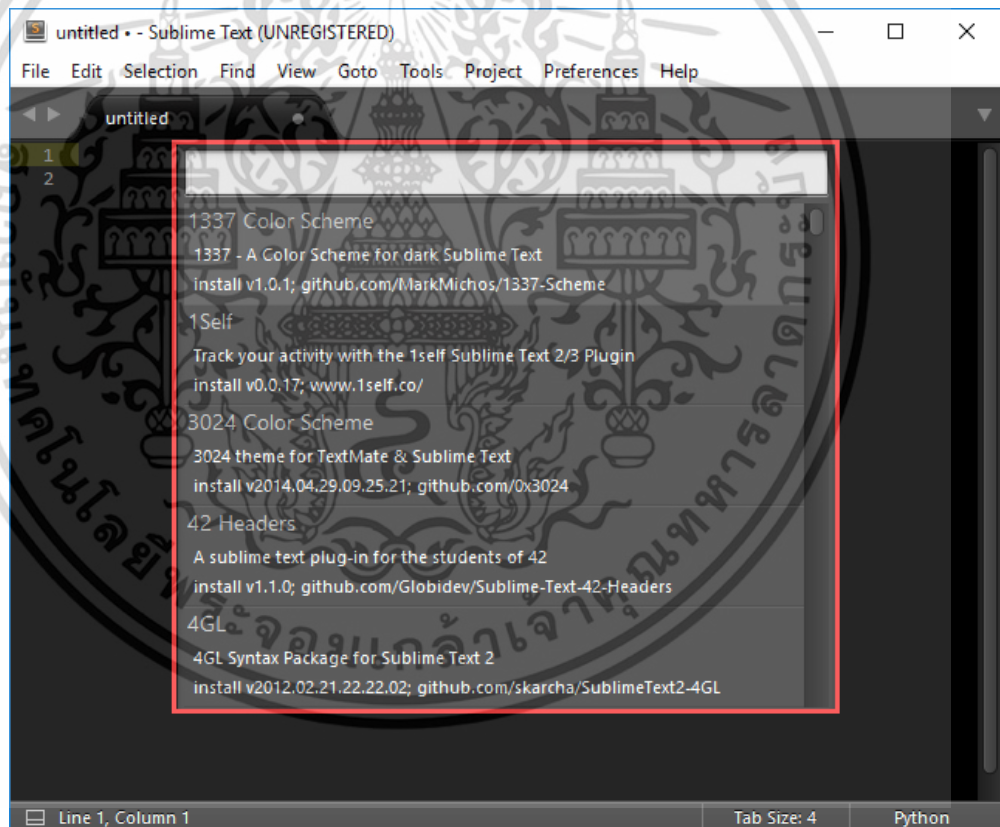
รูปที่ ก.42 แสดงการ install package Robot Framework (9)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานในเชิงวิชาการเท่านั้น ไม่อนุญาตให้นำไปใช้  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

13. จากนั้นพิมพ์คำว่า install เลือก install package



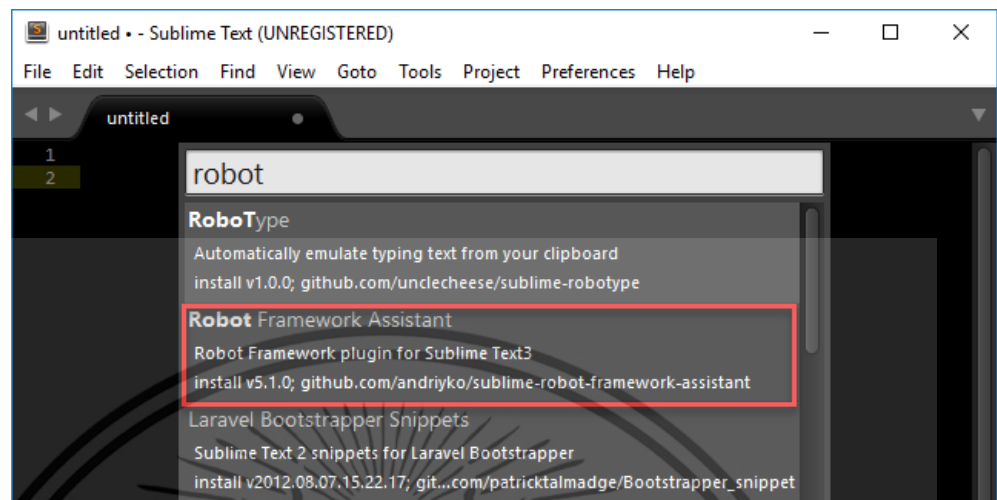
รูปที่ ก.43 แสดงการ install package Robot Framework (10)



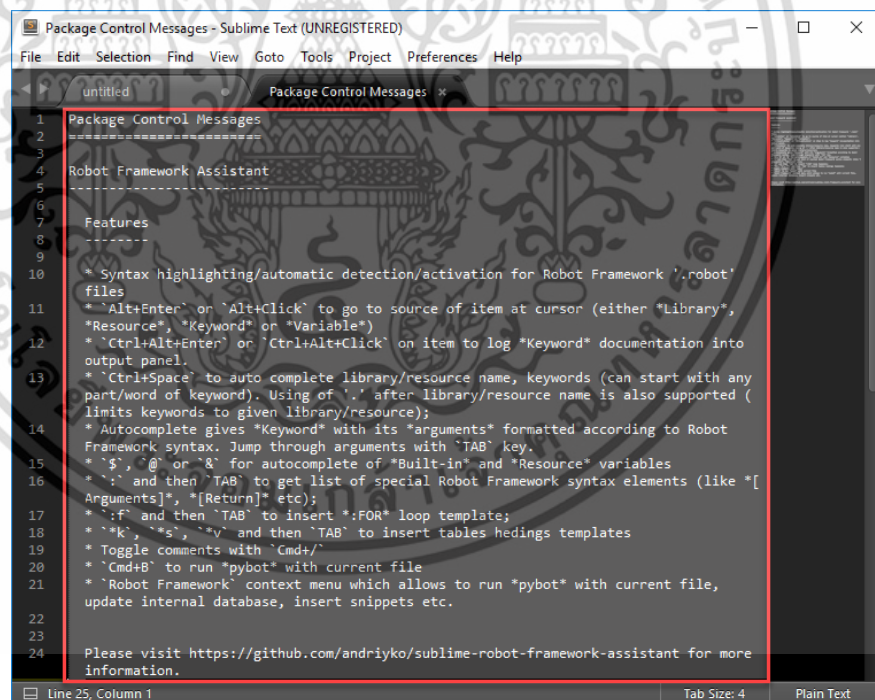
รูปที่ ก.44 install package Robot Framework สำเร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 14. พิมพ์ Robot จากนั้นเลือก Robot Framework Assistant



รูปที่ ก.45 แสดงการเลือก Robot Framework Assistant (11)



รูปที่ ก.46 เลือก Robot Framework Assistant สำเร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

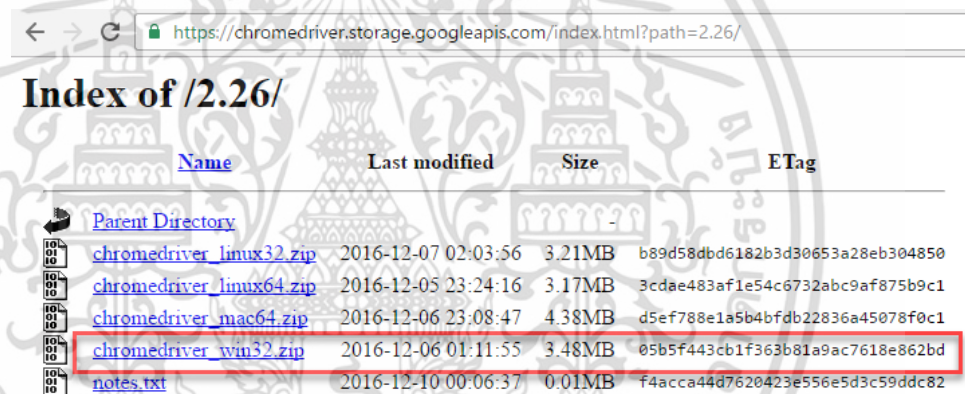
## การติดตั้ง Driver Server เพิ่มเติม

ในการใช้งาน Robot framework เพื่อทดสอบ Web application นั้น จะต้องมีการติดตั้ง Driver เพิ่มเติมเพื่อเรียกใช้งาน Web browser เช่น IE, Chrome , Firefox , Safari , Opera

## การติดตั้ง Chrome drive

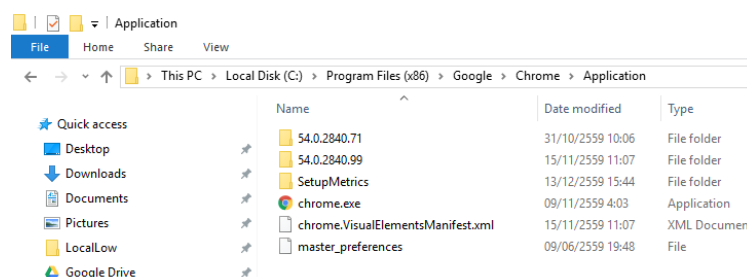
### 1. เข้าไป

<https://chromedriver.storage.googleapis.com/index.html?path=2.26/>  
เพื่อทำการ Download web driver เลือก Download ตามระบบปฏิบัติการของผู้ใช้งาน



รูปที่ ก.47 แสดงการเลือกดาวน์โหลด Chrome driver

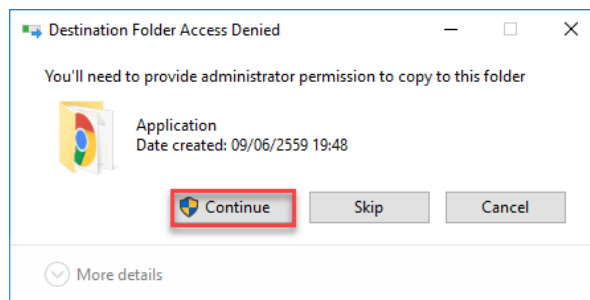
### 2. เมื่อโหลดไฟล์เสร็จแล้ว จากนั้นแตกไฟล์ออกมา แล้ว Copy ไปยัง Folder ที่ติดตั้ง Google Chrome ในที่นี้จะอยู่ใน C:\Program Files (x86)\Google \Chrome\Application



รูปที่ ก.48 แสดงการติดตั้ง Chrome driver (1)

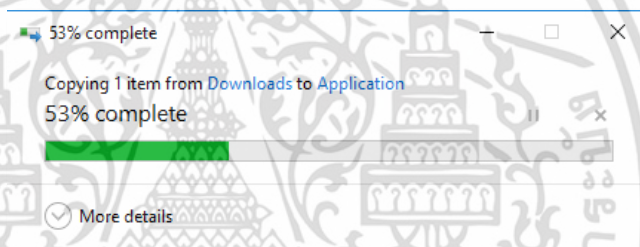
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้โดยไม่หวังกำไรใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เมื่อวางไฟล์ลงใน Folder จะมีแจ้งเตือน ให้เลือก Continue



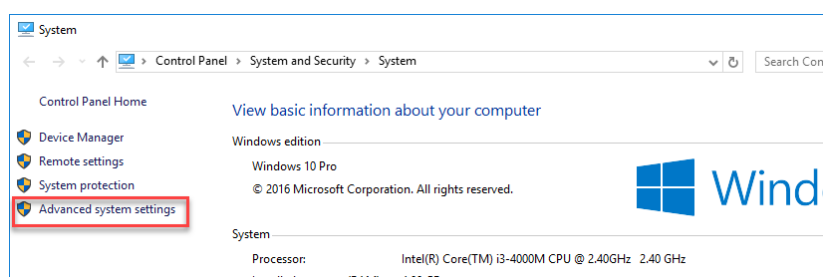
รูปที่ ก.49 แสดงการติดตั้ง Chrome driver (2)

4. รอจนครบ 100%



รูปที่ ก.50 แสดงการติดตั้ง Chrome driver

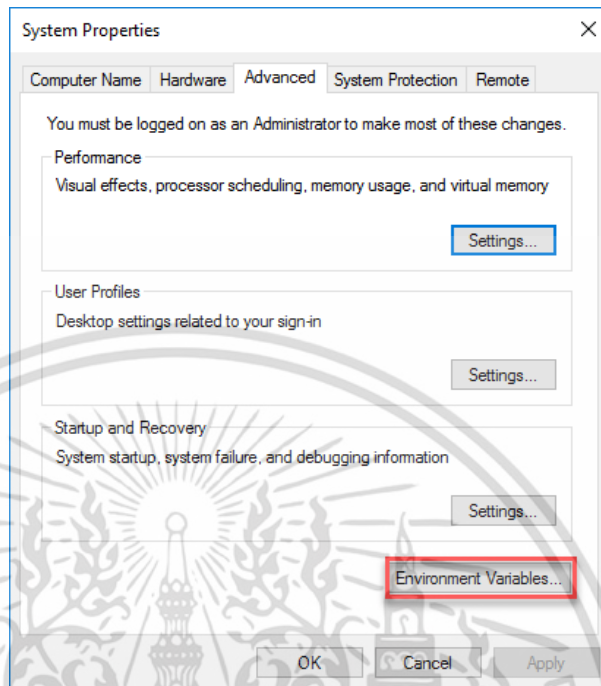
5. หลังจากทำขั้นตอนนี้ให้สั่ง Restart Computer เมื่อกลับมาให้ไปทำการ Setup Path chromedriver จากนั้นจะได้ Driver :chromedriver.exe ภายในโฟลเดอร์ Configuration for System Environments Variables โดยการไปที่ Computer เลือก Properties > Advance System Setting > Environment Variables



รูปที่ ก.51 แสดงการ Setup Path chromedriver (1)

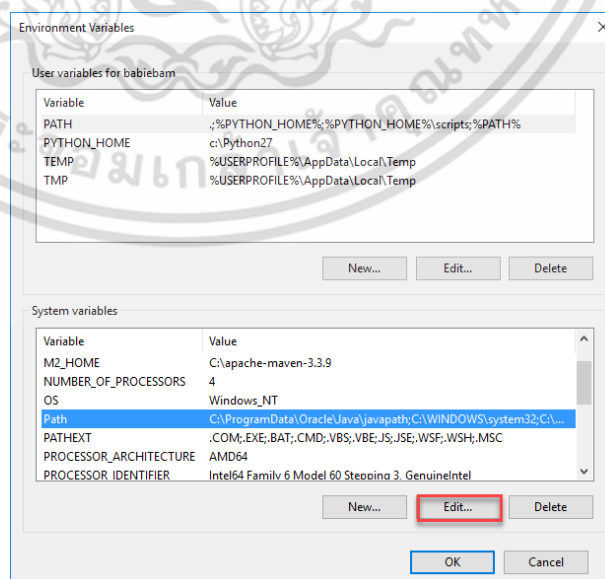
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่บนสื่อออนไลน์โดยไม่ได้รับอนุญาต หากมีข้อผิดพลาดประการใด ขออภัยเป็นอย่างสูง และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 6. เลือก Environment Variables



รูปที่ ก.52 แสดงการ Setup Path chromedriver (2)

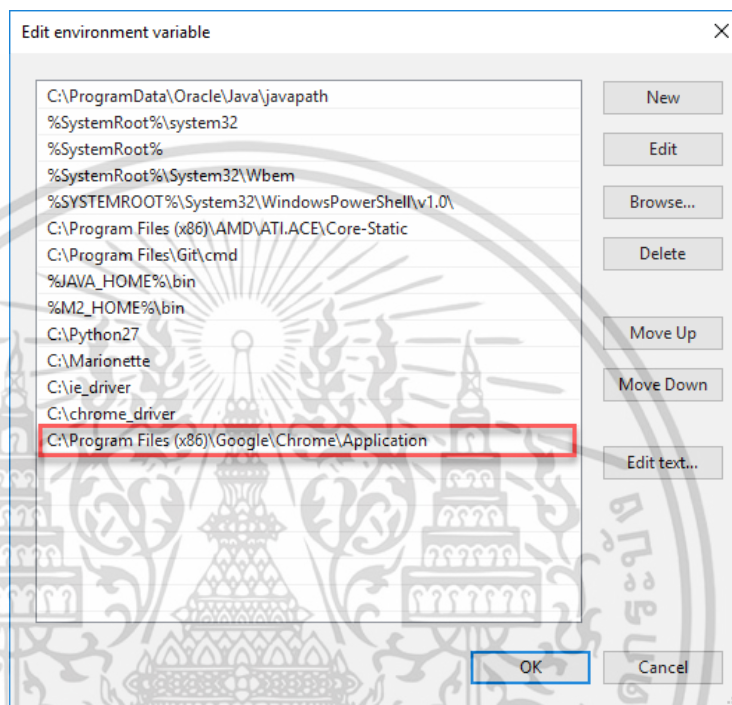
## 7. จากนั้นให้ทำการสร้าง Environment Variables ในส่วนของ System Variable โดยการเลือกที่ path จากนั้นกด Edit



รูปที่ ก.53 แสดงการ Setup Path chromedriver (3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. กดปุ่ม New เพื่อสร้าง Path จากนั้นใส่ค่า Path ที่เก็บ Google Chrome ในที่นี้คือ C:\Program Files (x86)\Google\Chrome\Application เสร็จแล้วกดปุ่ม OK จนครบ และกลับสู่ Desktop Windows



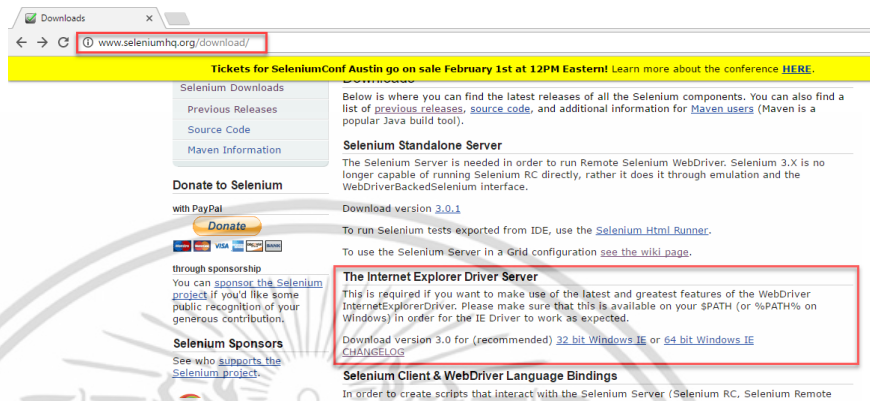
รูปที่ ก.54 แสดงการ Setup Path chromedriver (4)

9. เมื่อทำการตั้งค่า Protected Mode เรียบร้อยแล้ว ให้ทำการ ทดสอบ การติดตั้ง chromedriver ด้วยการสร้าง Testscript ที่ สั่งเปิด browser ด้วย Chrome

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

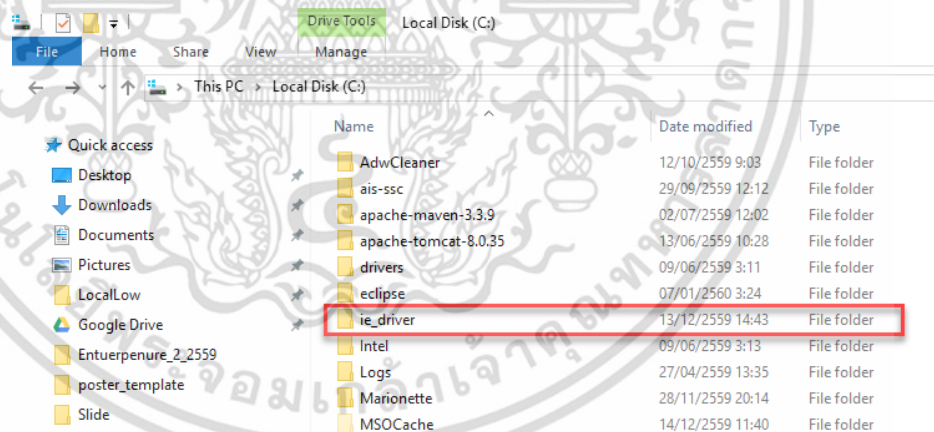
## การติดตั้ง IE driver

1. เข้าไปที่ URL :<http://www.seleniumhq.org/download/> เพื่อทำการ Download IE web driver เลือกดาวน์โหลดตามระบบปฏิบัติการของผู้ใช้



รูปที่ ก.55 แสดงการเลือกดาวน์โหลด IE Driver

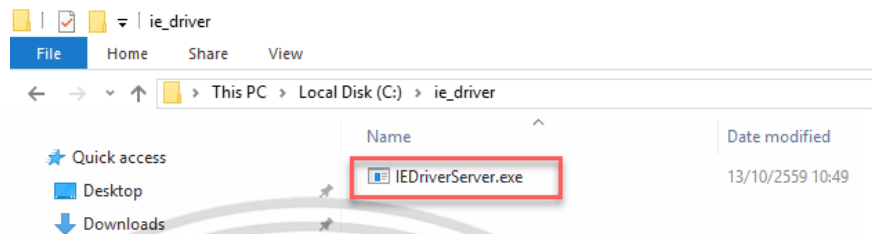
2. สร้าง Folder สำหรับเรียกใช้ IE Driver ไว้ที่ Drive C :



รูปที่ ก.56 แสดงการติดตั้ง IE Driver (1)

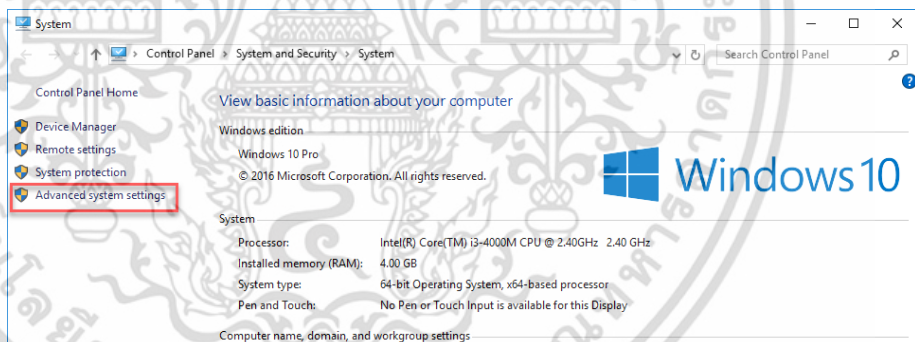
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. แยกไฟล์ออกมา แล้ว Copy ไปยัง Folder ที่สร้างขึ้นสำหรับ IE Driver ในที่นี้จะอยู่ใน C:\ie\_driver



รูปที่ ก.57 แสดงการติดตั้ง IE Driver (2)

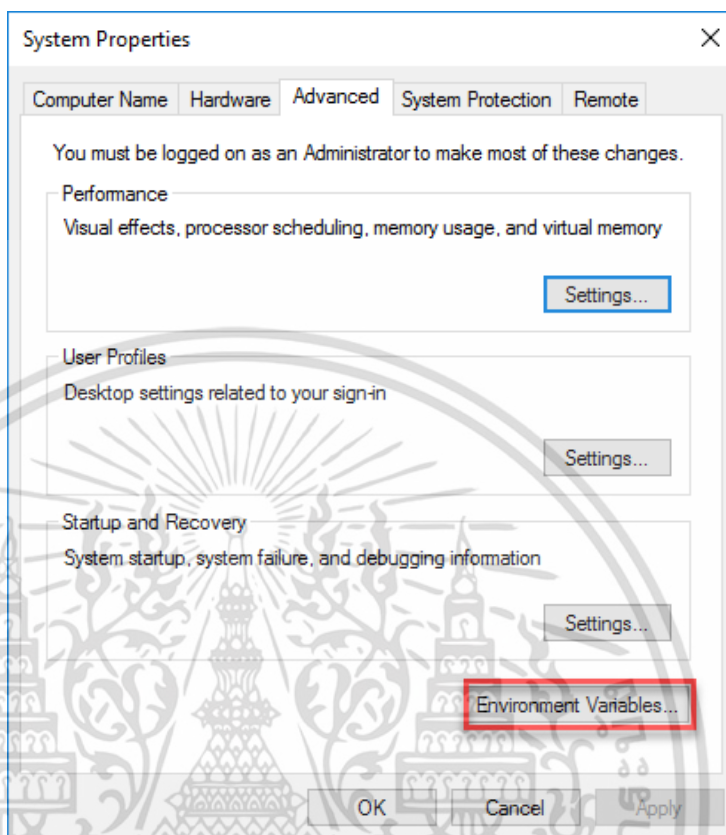
4. ทำการ Setup Path IE driver > Configuration for System Environments Variables โดยไปที่ This Computer เลือก Properties > Advance System Setting > Environment Variables



รูปที่ ก.58 4 แสดงการ Setup Path IE driver (1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

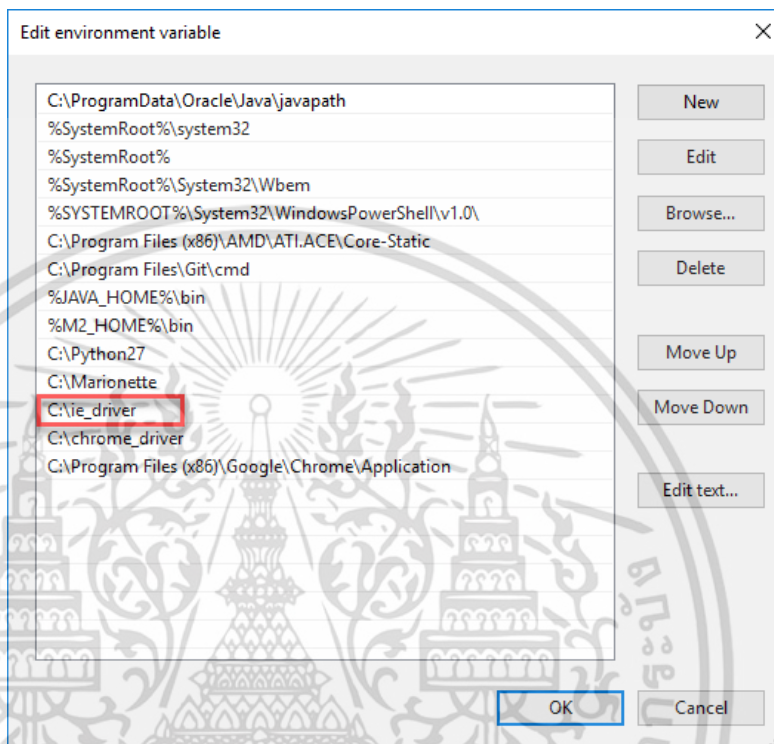
## 5. เลือก Environment Variables



รูปที่ ก.59 แสดงการ Setup Path IE driver (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

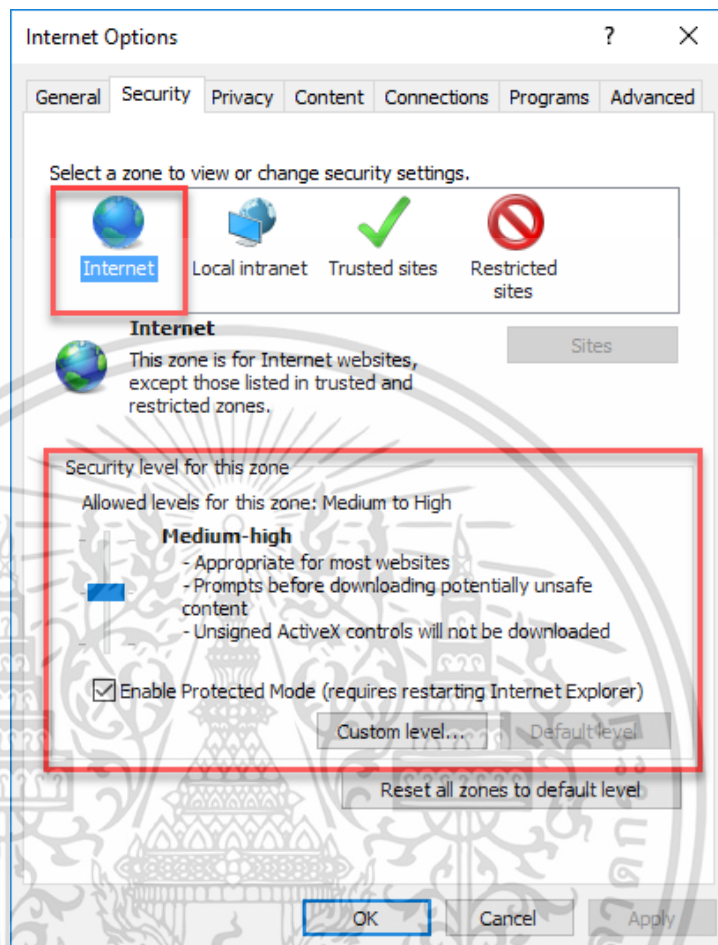
6. จากนั้นให้ทำการสร้าง Environment Variables ในส่วนของ System Variable โดยการเลือกที่ path จากนั้นกด Edit > New > ใส่ Path ของ IE driver ลงไป > กด OK > OK อีกครั้ง



รูปที่ ก.60 แสดงการ Setup Path IE driver (3)

7. หลังจากทำขั้นตอนนี้ให้สั่ง Restart Computer เมื่อกลับมาให้ไปทดสอบ การติดตั้ง IE driver ด้วยการสร้าง Testscript ที่ สั่งเปิด browser ด้วย Internet Explorer ทำการตั้งค่าที่ IE Browser ดังนี้ เปิด IE Browser > กดที่ไอคอนรูปเฟือง > เลือก Internet option

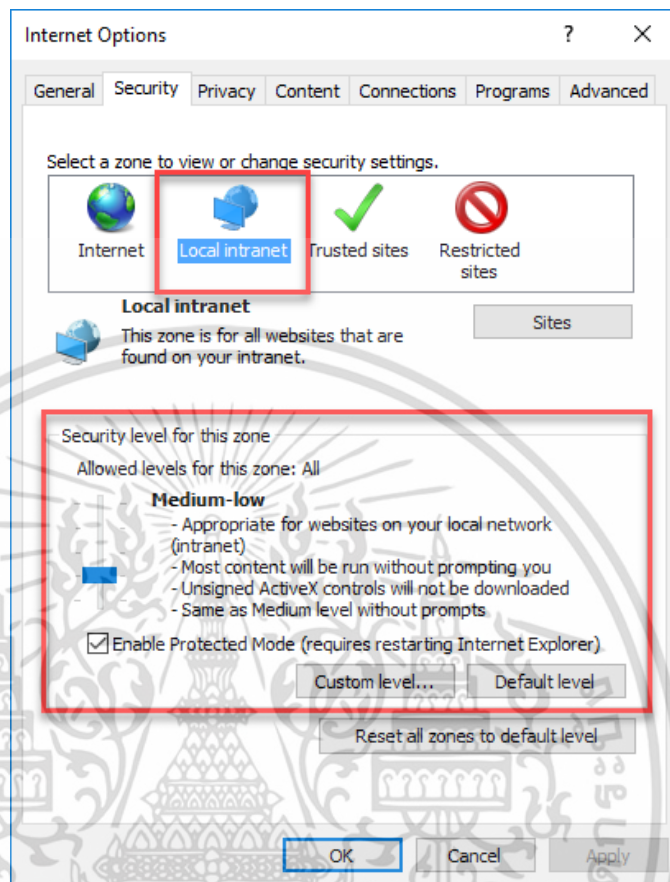
8. เลือก Tab Security > เลือก Internet > ทำการตั้งค่าดังรูปภาพ > กดปุ่ม Apply



รูปที่ ก.61 แสดงการตั้งค่าที่ IE Browser (1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

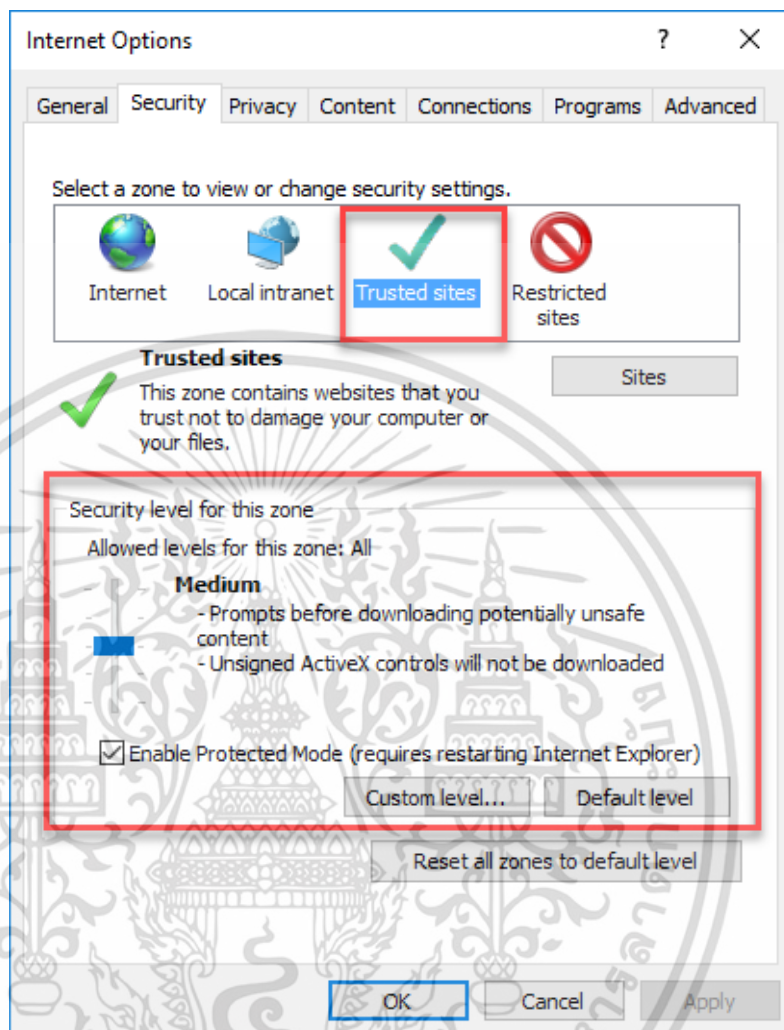
9. เลือก Local Intranet > ทำการตั้งค่าตั้งรูปภาพ > กดปุ่ม Apply



รูปที่ ก.62 แสดงการตั้งค่าที่ IE Browser (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

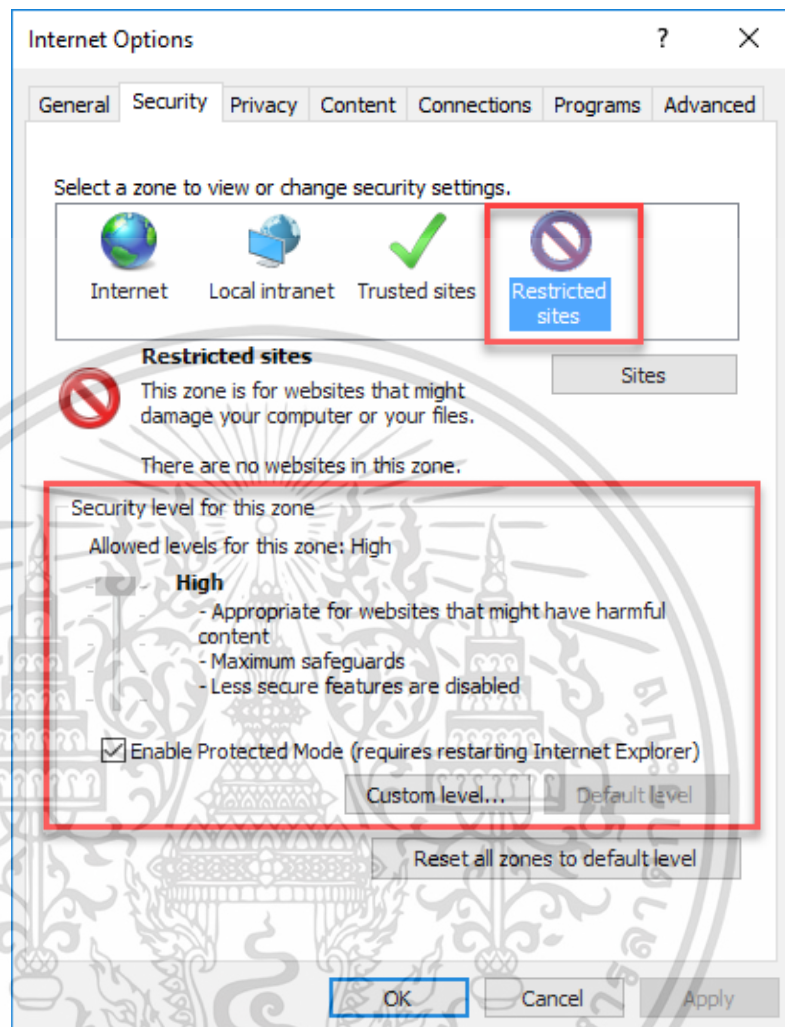
10. เลือก Trusted sites > ทำการตั้งค่าดังรูปภาพ > กดปุ่ม Apply



รูปที่ ก.63 แสดงการตั้งค่าที่ IE Browser (3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. เลือก Restricted sites > ทำการตั้งค่าดังรูปภาพ > กดปุ่ม Apply > กดปุ่ม Ok



รูปที่ ก.64 แสดงการตั้งค่าที่ IE Browser (4)

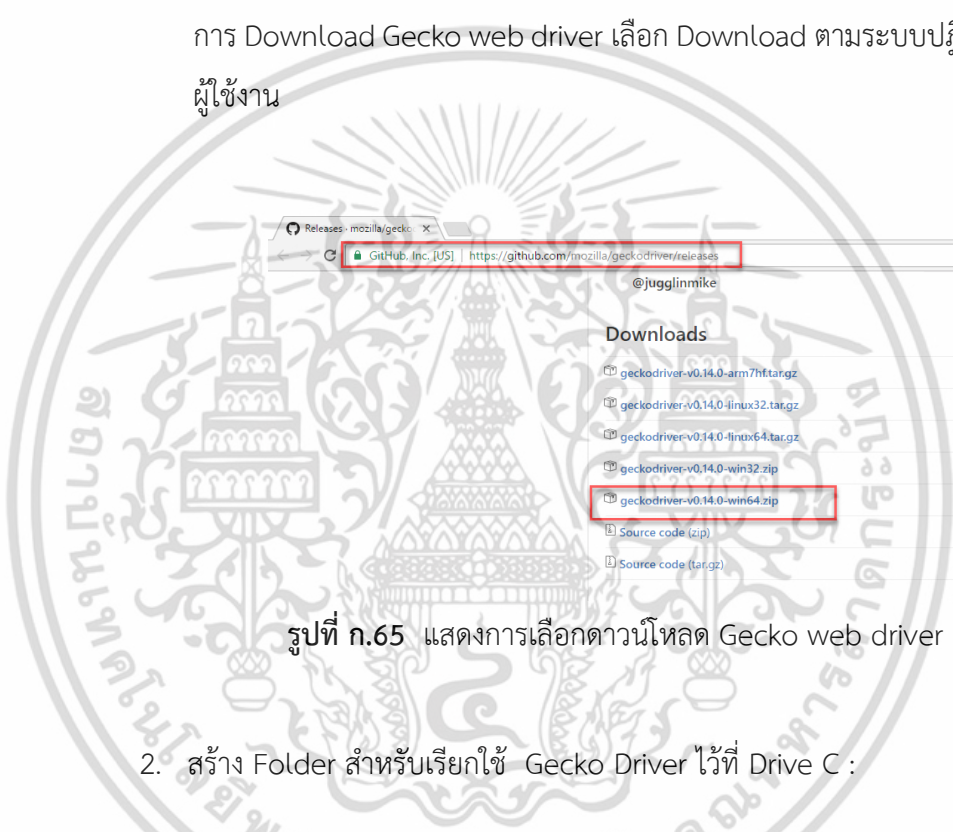
12. เมื่อทำการตั้งค่า Protected Mode เรียบร้อยแล้ว ให้ทำการ ทดสอบ การติดตั้ง chromedriver ด้วยการสร้าง Testscript ที่ สั่งเปิด browser ด้วย IE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### การติดตั้ง Gecko driver เพื่อใช้งาน Browser firefox

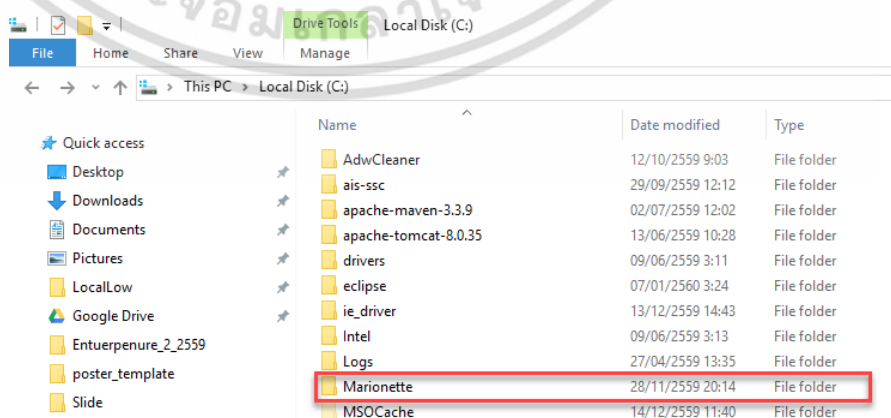
สำหรับการเรียกใช้งานบน Browser Fire fox ที่มี Version ต่ำกว่า 47 จะไม่ต้องติดตั้ง Gecko driver สามารถเรียกใช้งานได้เลย แต่ถ้าเป็น Version ในปัจจุบันส่วนใหญ่แล้วจะ Version สูงกว่า 47 จึงต้องทำการติดตั้ง Gecko driver เพื่อเรียกใช้งาน Browser firefox ดังนี้

1. เข้าไปที่ URL :<https://github.com/mozilla/geckodriver/releases> เพื่อทำการ Download Gecko web driver เลือก Download ตามระบบปฏิบัติการของผู้ใช้งาน



รูปที่ ก.65 แสดงการเลือกดาวน์โหลด Gecko web driver

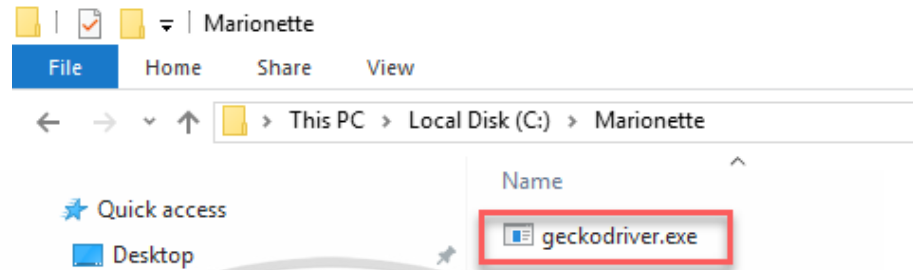
2. สร้าง Folder สำหรับเรียกใช้ Gecko Driver ไว้ที่ Drive C :



รูปที่ ก.66 แสดงการติดตั้ง Gecko Driver

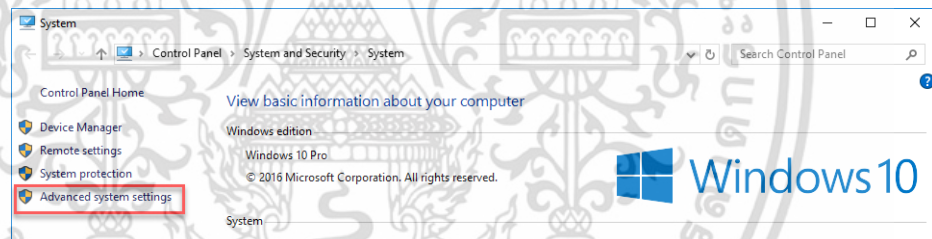
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. แยกไฟล์ออกมา แล้ว Copy ไปยัง Folder ที่สร้างขึ้นสำหรับ Gecko Driver ใน  
ที่นี้จะอยู่ใน C:\Marionette



รูปที่ ก.67 แสดงการติดตั้ง Gecko Driver

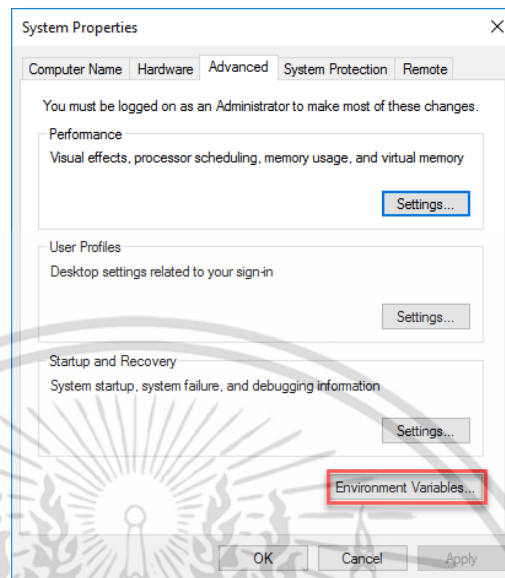
4. ทำการ Setup Path Gecko driver Configuration for System Environments Variables โดยไปที่ This Computer เลือก Properties > Advance System Setting > Environment Variables



รูปที่ ก.68 แสดงการ Setup Path Gecko driver (1)

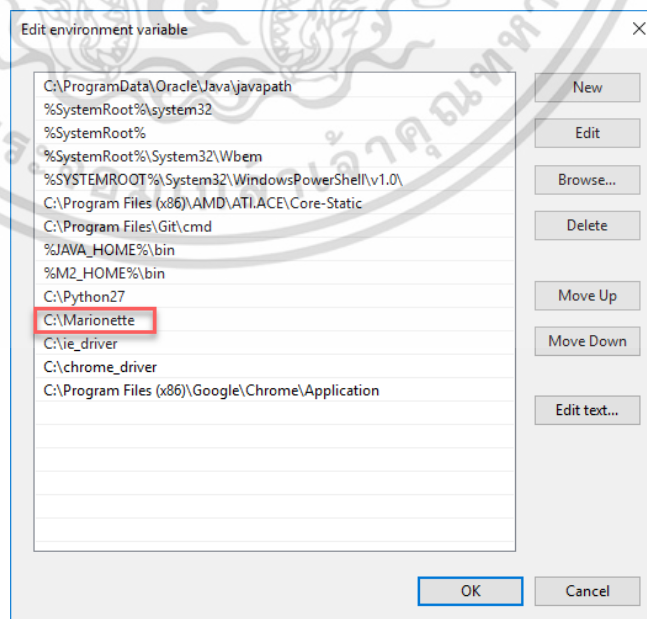
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 5. เลือก Environment Variables



รูปที่ ก.69 แสดงการ Setup Path Gecko driver (2)

6. จากนั้นให้ทำการสร้าง Environment Variables ในส่วนของ System Variable โดยการเลือกที่ path จากนั้นกด Edit > New > ใส่ Path ของ IE driver ลงไป > กด OK > OK อีกครั้ง



รูปที่ ก.70 แสดงการ Setup Path Gecko driver (3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## การใช้งาน Robot Framework

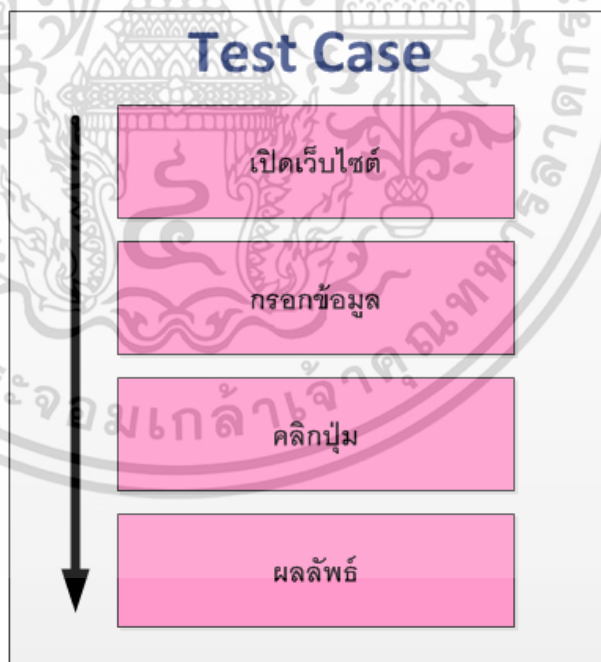
ตัวอย่างประโยคในการเขียน Test Script

1. ลงชื่อเข้าสู่ระบบ Facebook กรณีระบุ Username และ Password ถูกต้อง ระบบจะแสดงหน้าหลักของเว็บไซต์ Facebook

	A	B	C	D	E	F	G	H	I	J	K	L
15	No.	Test Procedure			Expected Result			Test Result (Pass/Fail)		Remark		
	1	เปิดเว็บไซต์http://www.facebook.com			แสดงหน้า Facebook							
16	2	ที่หน้า Facebook กรอกข้อมูลที่ช่อง อีเมลหรือโทรศัพท์ ="xxx@hotmail.com" กรอกข้อมูลที่ช่องรหัสผ่าน ="xxxxxx" คลิกปุ่ม "เข้าสู่ระบบ"			แสดงข้อความ "หน้าหลัก"							
17												

รูปที่ ก.71 ตัวอย่าง Test case ที่ใช้ในการทดสอบ

2. จากตัวอย่าง case ข้างต้นเราก็จะเห็น Process การทำงานที่เรียงเป็นลำดับได้ดังนี้



รูปที่ ก.72 Process การทำงานของ Test case

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. จาก Process ดังกล่าว เราสามารถนำมาเขียนเป็น Script ใน Robot framework ได้ดังนี้ เริ่มที่การวางโครงสร้างโดยใน sublime สามารถเรียกโครงสร้างของ Robot ได้โดยคลิกขวา > Robot Framework > Snippets

```

1  *** Settings ***
2
3  *** Test Cases ***

```

รูปที่ ก.73 การใช้ Snippets ใน Robot framework

4. เมื่อสร้างโครงสร้างเรียบร้อยแล้วก็เริ่มเขียน Test case ได้เลย ในกรอบสี่เหลี่ยมคือคีย์เวิร์ด ที่สั่งให้สคริปทำงาน สามารถเข้าไปดู Keyword ใน Selenium2Library ได้ที่ <http://robotframework.org/Selenium2Library/Selenium2Library.html>

```

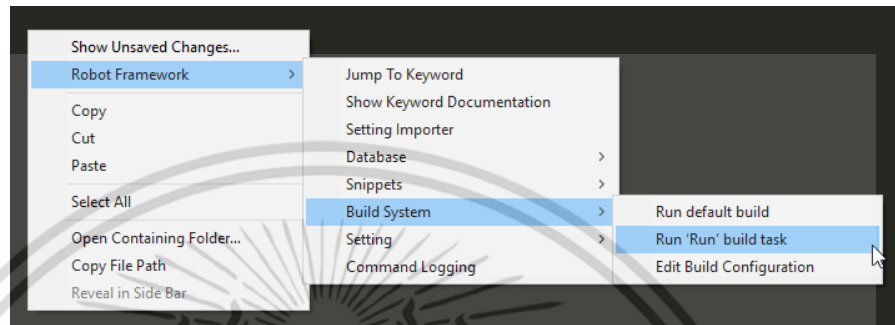
1  *** Settings ***
2  Library    Selenium2Library
3
4  *** Variables ***
5  ${email}
6  ${password}
7
8  *** Test Case ***
9
10 Test Facebook Login
11   Open Browser    https://www.facebook.com    gc
12   Wait Until Page Contains    โลโก้ Facebook
13   Input Text    email    ${email}
14   Input Password    pass    ${password}
15   Click Button    เมื่อกำลังระบบ
16   Wait Until Page Contains    หน้าหลัก
17   Close Browser
18

```

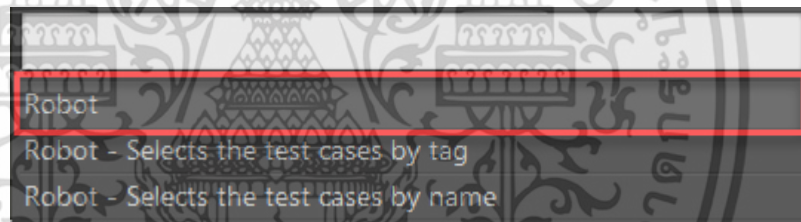
รูปที่ ก.74 ตัวอย่าง Test script ที่ใช้ทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. หลังจากทำการเขียน Script เรียบร้อยแล้ว ให้ทำการ Save ด้วย .robot หรือ .txt ก็ได้ เมื่อทำการ Save เรียบร้อยแล้ว ให้ทำการ Execute Test script โดยการคลิกขวา > เลือก Robot framework > เลือก Build System > เลือก Run 'Run' build task > เลือก Robot



รูปที่ ก.75 การ Run Test Case (1)



รูปที่ ก.76 การ Run Test Case (2)

6. รอผลการ Execut test script เมื่อทำการ Execute เรียบร้อยแล้ว จะได้ผลการทดสอบ และไฟล์ออกมา 3 ไฟล์ คือ Output , Log และ Report โดยโฟลเดอร์ที่ทำการเก็บไฟล์จะอยู่ในโฟลเดอร์เดียวกันกับโฟลเดอร์ Test script

```

=====
Case-2-Facebook-Login
=====
Test Facebook Login | PASS |
-----
Case-2-Facebook-Login | PASS |
1 critical test, 1 passed, 0 failed
1 test total, 1 passed, 0 failed
-----
Output: D:\Robot\output.xml
Log: D:\Robot\log.html
Report: D:\Robot\report.html
[Finished in 235.2s]

```

รูปที่ ก.77 การ Run Test Case (3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเชิงการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Case-1-Facebook-Login Test Log

Generated  
20160616 17:39:07 GMT +08:00  
16 hours 6 minutes ago

### Test Statistics

Total Statistics	Total	Pass	Fail	Elapsed	Pass / Fail
Critical Tests	1	1	0	00:00:08	<span style="color: green;">██████████</span>
All Tests	1	1	0	00:00:08	<span style="color: green;">██████████</span>

Statistics by Tag	Total	Pass	Fail	Elapsed	Pass / Fail
No Tags					<span style="color: gray;">██████████</span>

Statistics by Suite	Total	Pass	Fail	Elapsed	Pass / Fail
<a href="#">Case-1-Facebook-Login</a>	1	1	0	00:00:08	<span style="color: green;">██████████</span>

### Test Execution Log

**SUITE** Case-1-Facebook-Login

Full Name: Case-1-Facebook-Login  
 Source: D:\My Blog\Workshop-robotBlog\case-1-facebook-login.txt  
 Start / End / Elapsed: 20160616 17:38:59.026 / 20160616 17:39:07.120 / 00:00:08.094  
 Status: 1 critical test, 1 passed, 0 failed  
 1 test total, 1 passed, 0 failed

---

**TEST** Facebook Login

Full Name: Case-1-Facebook-Login Facebook Login  
 Start / End / Elapsed: 20160616 17:38:59.255 / 20160616 17:39:07.118 / 00:00:07.863  
 Status: PASS (critical)

- KEYWORD Selenium2Library .Open Browser https://www.facebook.com/, gc
- KEYWORD Selenium2Library .Wait Until Page Contains โฟโต้ Facebook
- KEYWORD Selenium2Library .Input Text email, tasanee\_fai@hotmail.com
- KEYWORD Selenium2Library .Input Password pass, 0882336079

รูปที่ ก.78 ตัวอย่าง Report ที่เกิดจากการ Run

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ภาคผนวก ข

## การติดตั้ง และการใช้งาน

### โปรแกรม Selenium IDE

Selenium IDE คือ Automated Testing Tools ที่ออกแบบมา สำหรับ ใช้งาน ทดสอบ กับ Web Application การทำงานของ Selenium IDE จะเป็นในลักษณะ บันทึก ขั้นตอนการทำงาน ตามที่ เราใช้งานกับ Web Application หลังจากนั้น เราสามารถ Save Test Script ตาม Test Scenario ที่เราออกแบบไว้ และ Export ออกมา เป็น Script ในภาษาต่างๆ ได้ เช่น Python Ruby และนำ Test Script ที่ได้มา นำ ไป Run Test ในการทำ Automated Regression Testing ได้

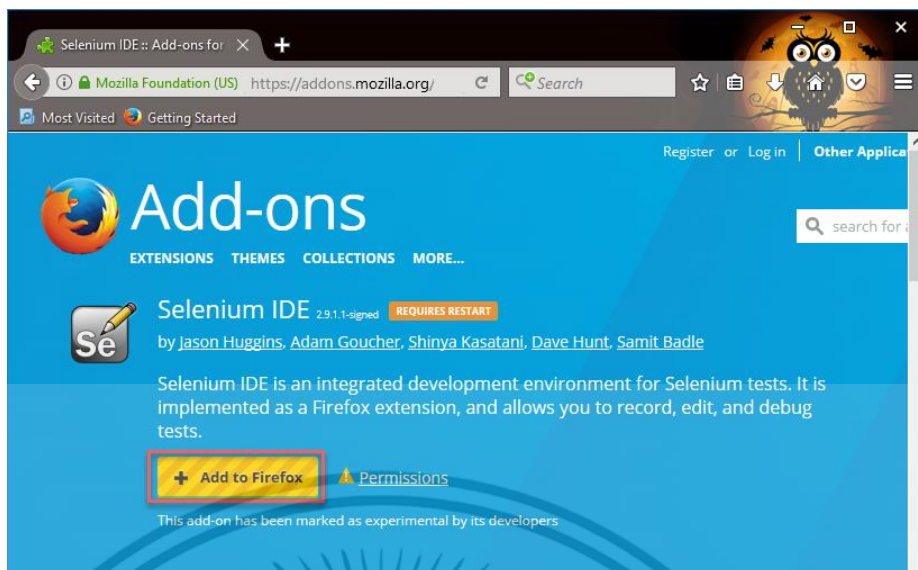
#### วิธีติดตั้ง Selenium IDE

การติดตั้งต้องมี Software ที่ใช้ในการติดตั้งดังนี้

- Browser คือ Firefox
- Java JDK Version 1.7

#### ขั้นตอนการติดตั้ง

1. เปิด Browser : Mozilla Firefox ใส่ URL : <https://addons.mozilla.org/en-us/firefox/addon/selenium-ide/>
2. เลือก Add to firefox เพื่อดาวน์โหลด Plug in เข้าสู่ Web browser



รูปที่ ข.1 การ Download โปรแกรม Selenium IDE

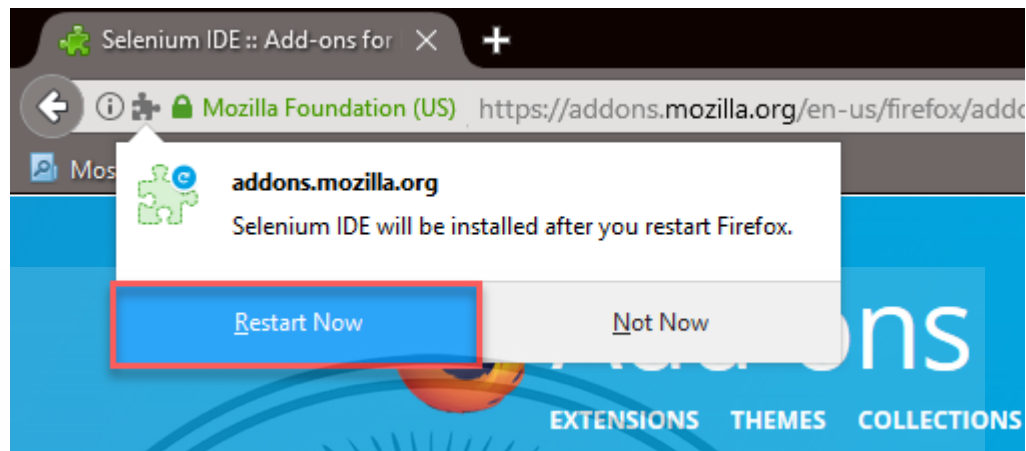
3. หลังจากดาวน์โหลดเสร็จเรียบร้อย โปรแกรมจะขึ้นกล่อง pop up ด้านมุมซ้ายบน จากนั้นให้เลือก install เพื่อทำการติดตั้งโปรแกรม



รูปที่ ข.2 การ Install โปรแกรม Selenium IDE

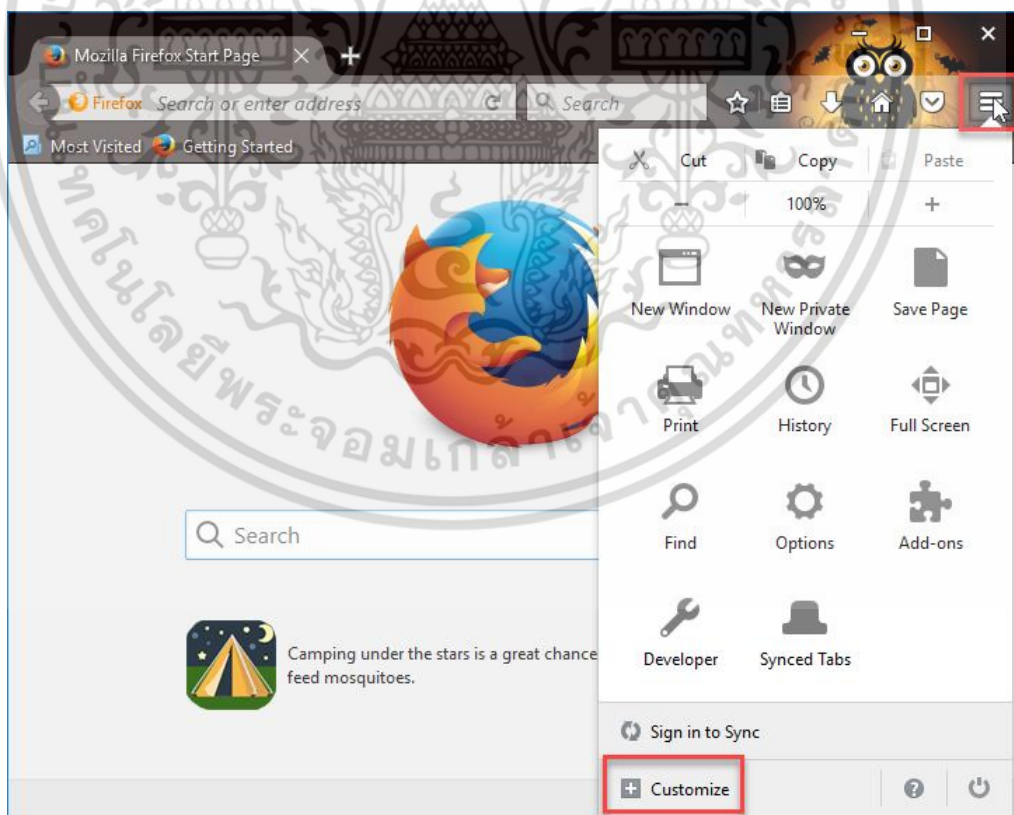
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. จากนั้น จะมีการ Download และติดตั้ง พร้อมกับ ต้อง สั่ง Restart Firefox อีกครั้ง



รูปที่ ข.3 การ Restart โปรแกรมหลังจากทำการติดตั้ง

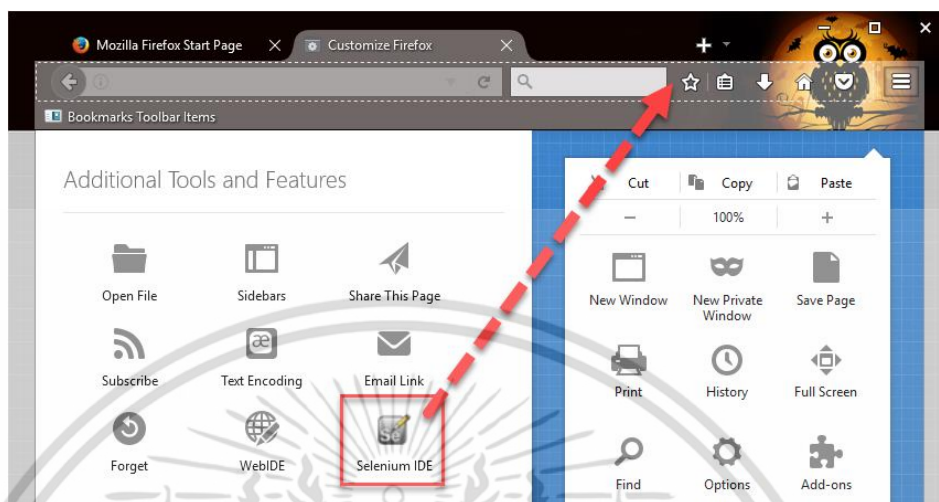
5. หลังจาก Restart เสร็จแล้ว ให้เลือก Open menu ด้านมุมขวาบน จากนั้นเลือก Customize



รูปที่ ข.4 การ Customize เพื่อให้โปรแกรมขึ้นมาแสดงบน Tool bar

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ทำการ Drag and Drop โปรแกรม Selenium IDE มาไว้บน Tool bar เพื่อให้สามารถเรียกใช้งาน Record and play ได้สะดวก



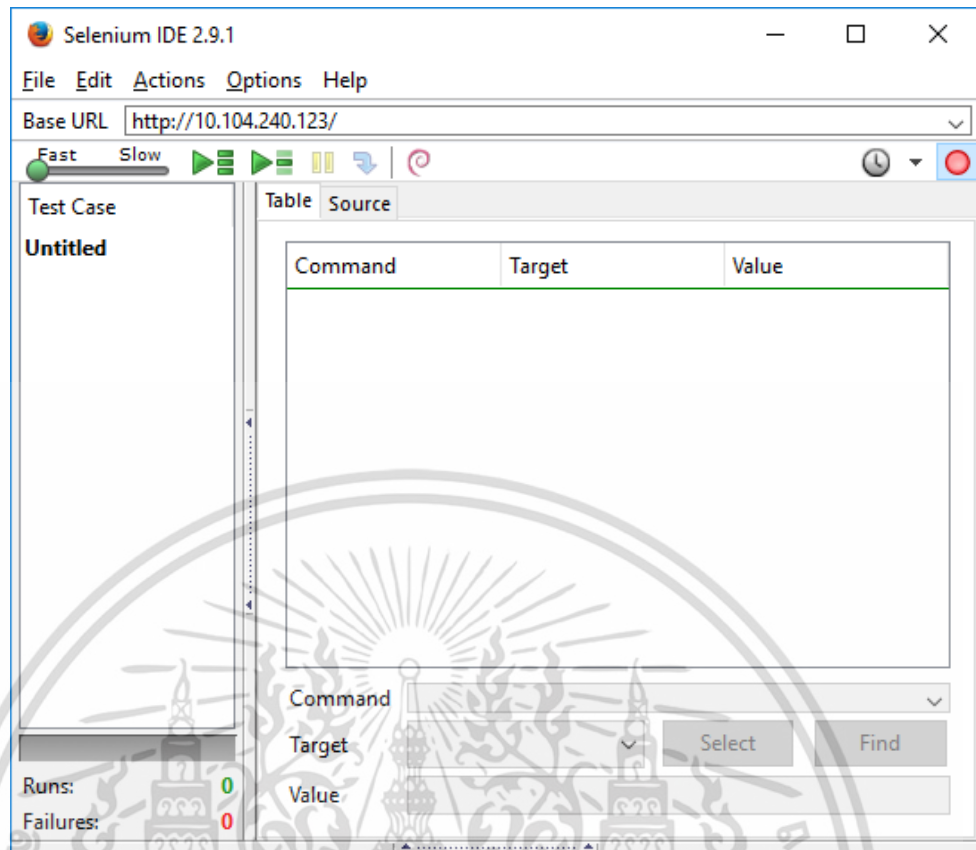
รูปที่ ข.5 การ Drag and Drop โปรแกรมมาแสดงบน Tool bar

7. Selenium IDE จะอยู่มุมขวามือด้านบน กด ที่ shortcut Selenium IDE Record ก็จะสามารถใช้งานได้



รูปที่ ข.6 การเรียกใช้งาน Selenium IDE (1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

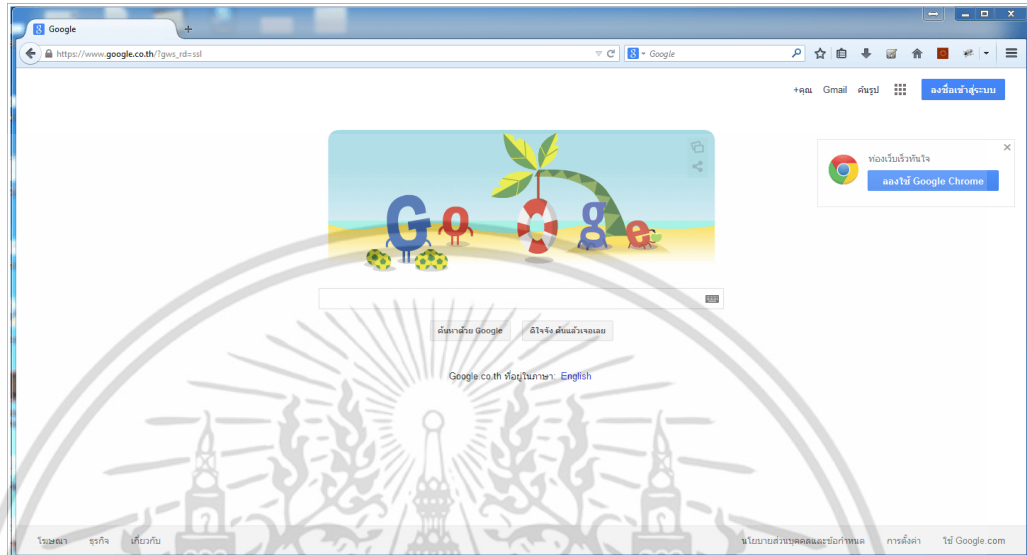


รูปที่ ข.7 การเรียกใช้งาน Selenium IDE (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

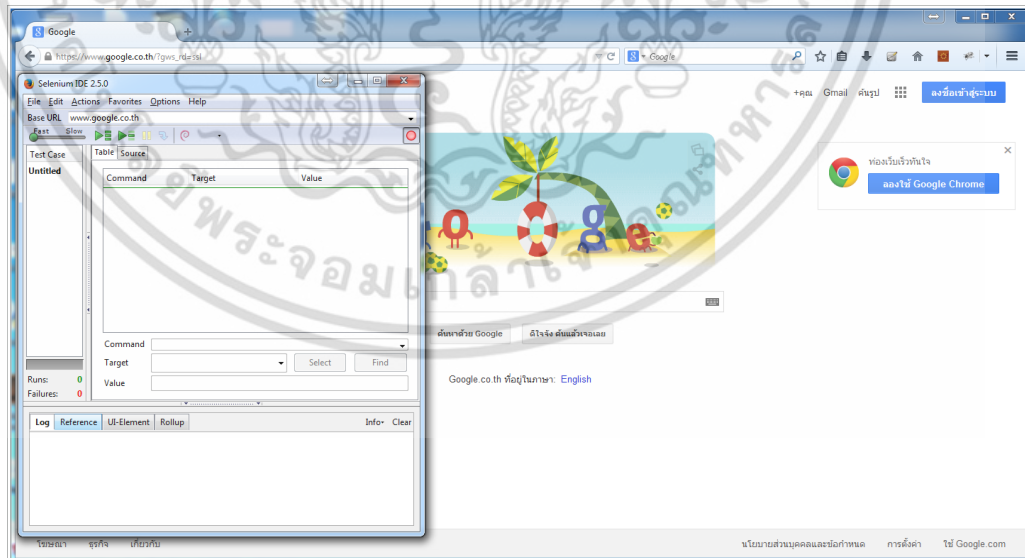
## การใช้งาน Selenium IDE

1. เริ่มจาก เปิด Browser : Fire Fox จากนั้น ใส่ URL : [www.google.co.th](http://www.google.co.th)



รูปที่ ข.8 ใส่ URL ที่ต้องการทดสอบ

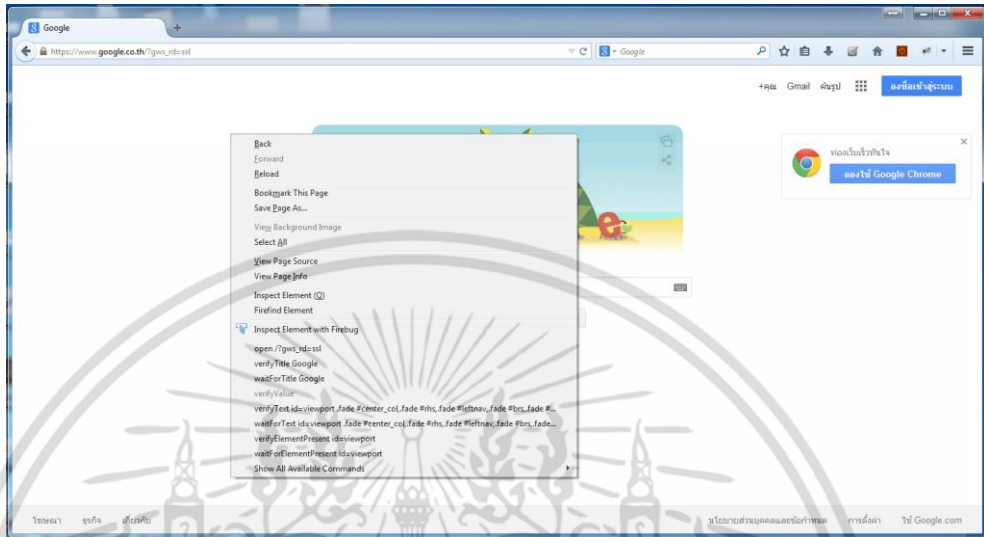
2. เปิด Selenium IDE ขึ้นมา กด ปุ่ม สีแดงๆ มุมขวา ของ Selenium IDE เพื่อบันทึก test scenario ( โดยปกติ ตัว Selenium IDE จะ กด ปุ่ม สีแดง ไว้ให้อยู่แล้ว )



รูปที่ ข.9 เปิดโปรแกรม Selenium IDE

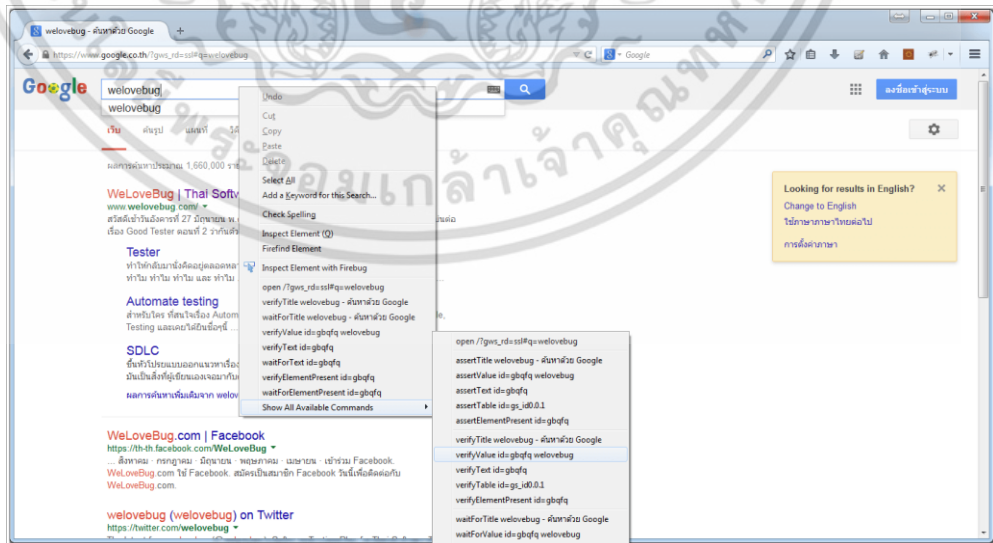
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ใส่ช่อง base URL ของ Selenium IDE ใส่ url : www.google.co.th
4. กลับมาที่ Browser ที่เราเปิด www.google.co.th ที่ังไว้ Click ขวา แล้วจะมีคำว่า Show All Available Command เลือก open



รูปที่ ข.10 การเลือก Show All Available Command

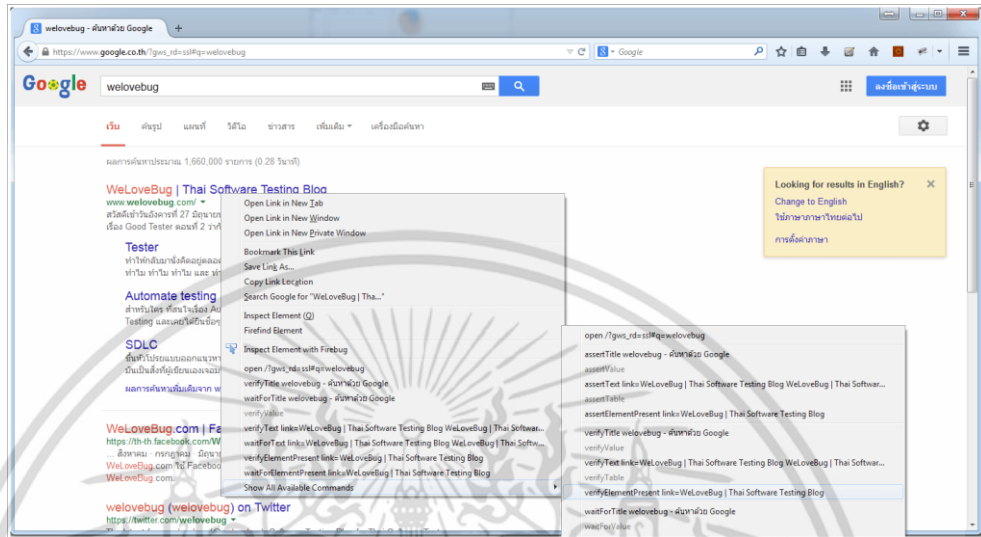
5. พิมพ์ welovebug ลงไปใน ช่องการค้นหา (Command : type Target : id=gbqfq Value : welovebug ) Click ขวา แล้วจะมีคำว่า Show All Available Command เลือก verifyValue : id=gbqfq Value : welovebug พร้อม Click ที่ปุ่มค้นหา



รูปที่ ข.11 การใช้ Command (1)

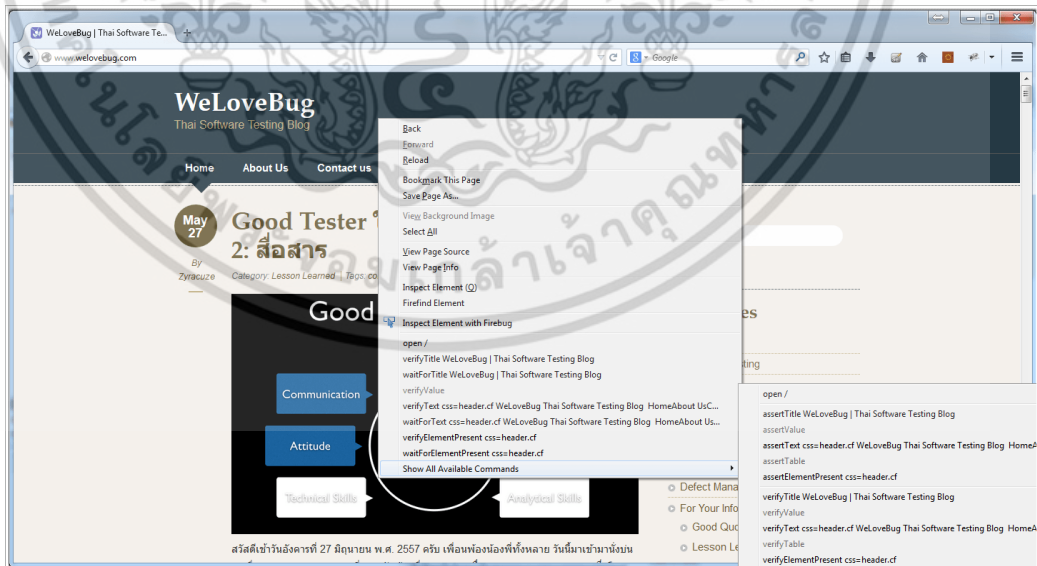
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- จากนั้น จะมี search result ให้เราเลือก welovebug.com Click ขวา แล้วจะมีคำว่า Show All Available Command เลือก Command : verifyElementPresent : Target : link=WeLoveBug | Thai Software Testing Blog



รูปที่ ข.12 การใช้ Command (2)

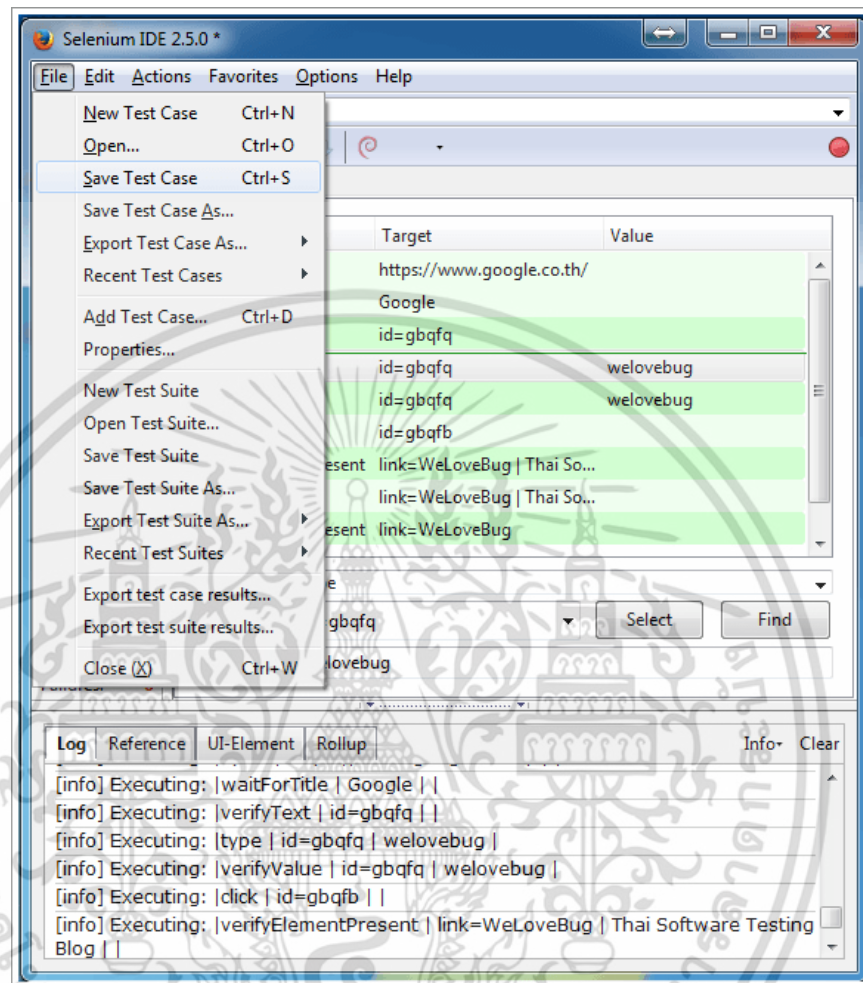
- Click เพื่อเข้าสู่ welovebug.com Click ขวา แล้วจะมีคำว่า Show All Available Command เลือก Command : verifyElementPresent : Target link=WeLoveBug



รูปที่ ข.13 การใช้ Command (3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

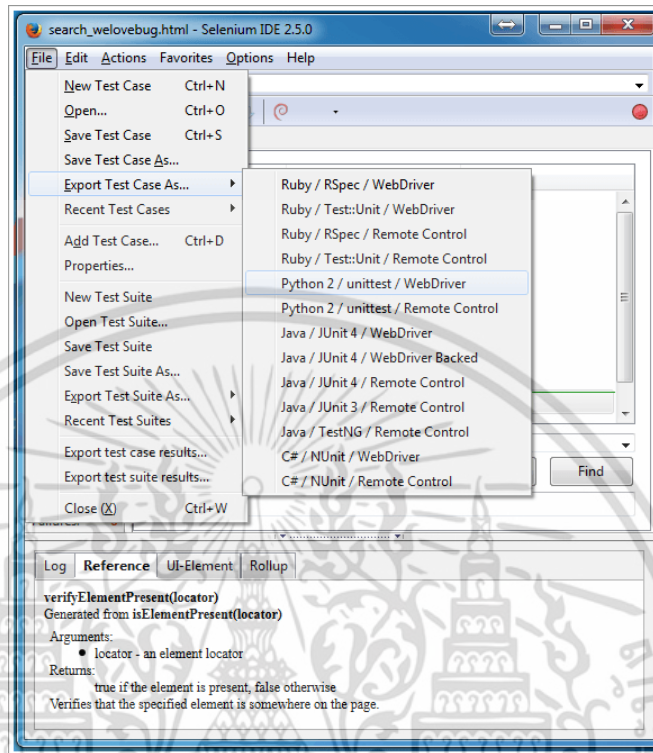
จากนั้น สั่ง Save Test Case เพื่อ Save Test Case นี้ในรูปแบบ HTML เพื่อใช้ในการ Run ซ้ำ ภายหลัง กับ Selenium IDE Record



รูปที่ ข.14 การสั่ง Save Test Case

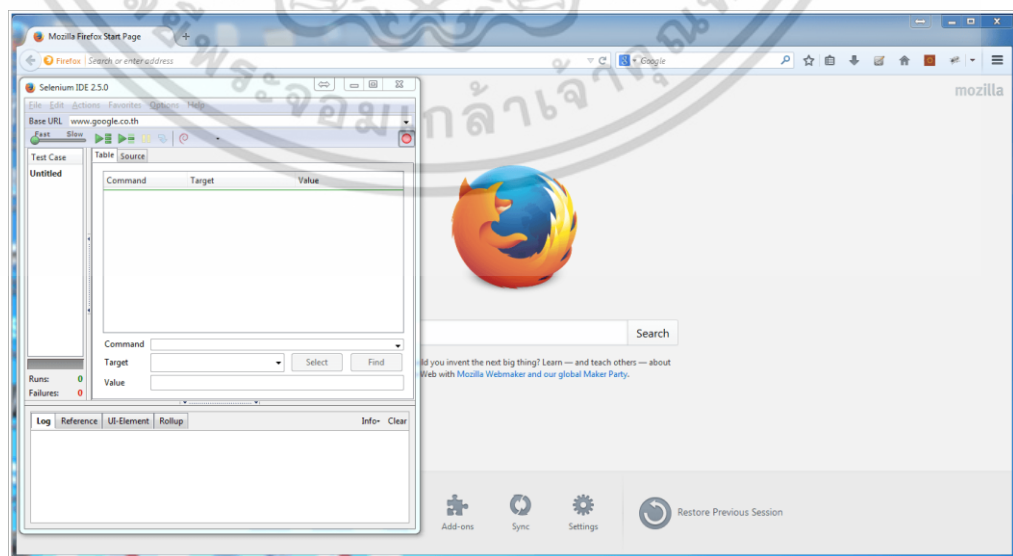
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. หรือจะสั่ง Export Test Case As : ตามด้วย ภาษา ที่เราต้องการ เช่น Python เพื่อนำไปใช้ Run กับ Selenium WebDriver ภายหลัง



รูปที่ ข.15 การสั่ง Export Test Case

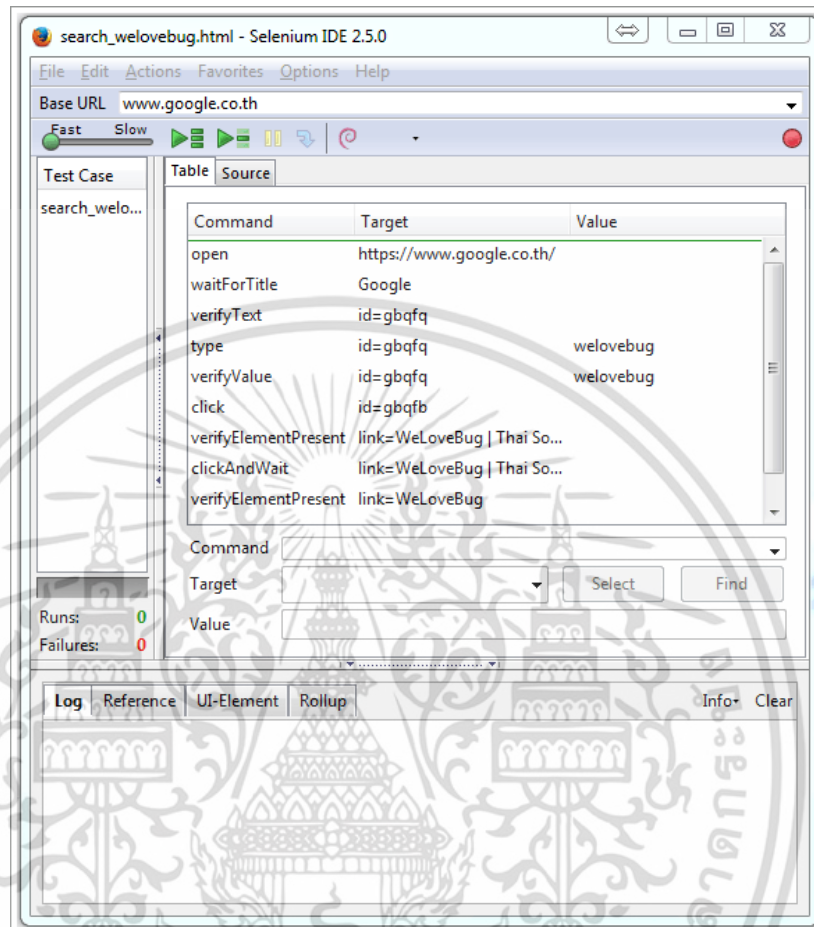
9. การ Run Test Script บน Selenium IDE record เปิด web browser : Fire Fox ทิ้งไว้ และเปิด Selenium IDE Record



รูปที่ ข.16 การ Run Test Script บน Selenium IDE (1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

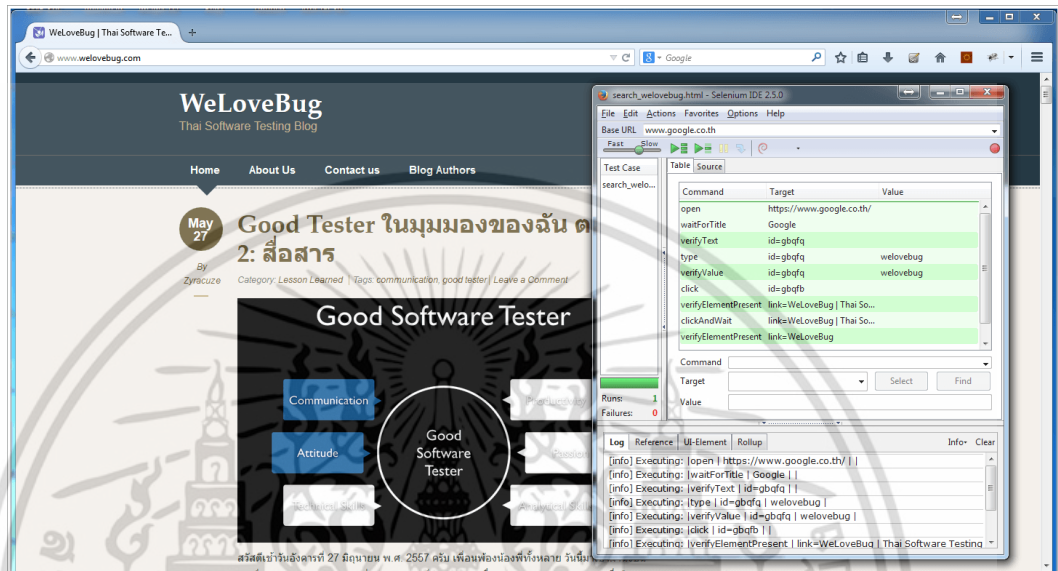
10. ไปที่ Selenium IDE Record เลือก Open จากนั้น เลือก Script ที่บันทึกไว้ใน HTML File



รูปที่ ข.17 การ Run Test Script บน Selenium IDE (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. จากนั้นกลับมาที่ Selenium IDE Record สั่ง Play Current Test Case เลือก ความเร็ว เป็น slow Selenium IDE Record จะทำการ Run Test Script ตามที่เราบันทึกไว้ ถ้าหาก Testcase ทำงานถูกต้อง Testcase นั้นจะแสดงเป็นสีเขียว หากมีข้อผิดพลาดจะแสดงด้วยสีแดง



รูปที่ ข.18 การ Run Test Script บน Selenium IDE (3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้