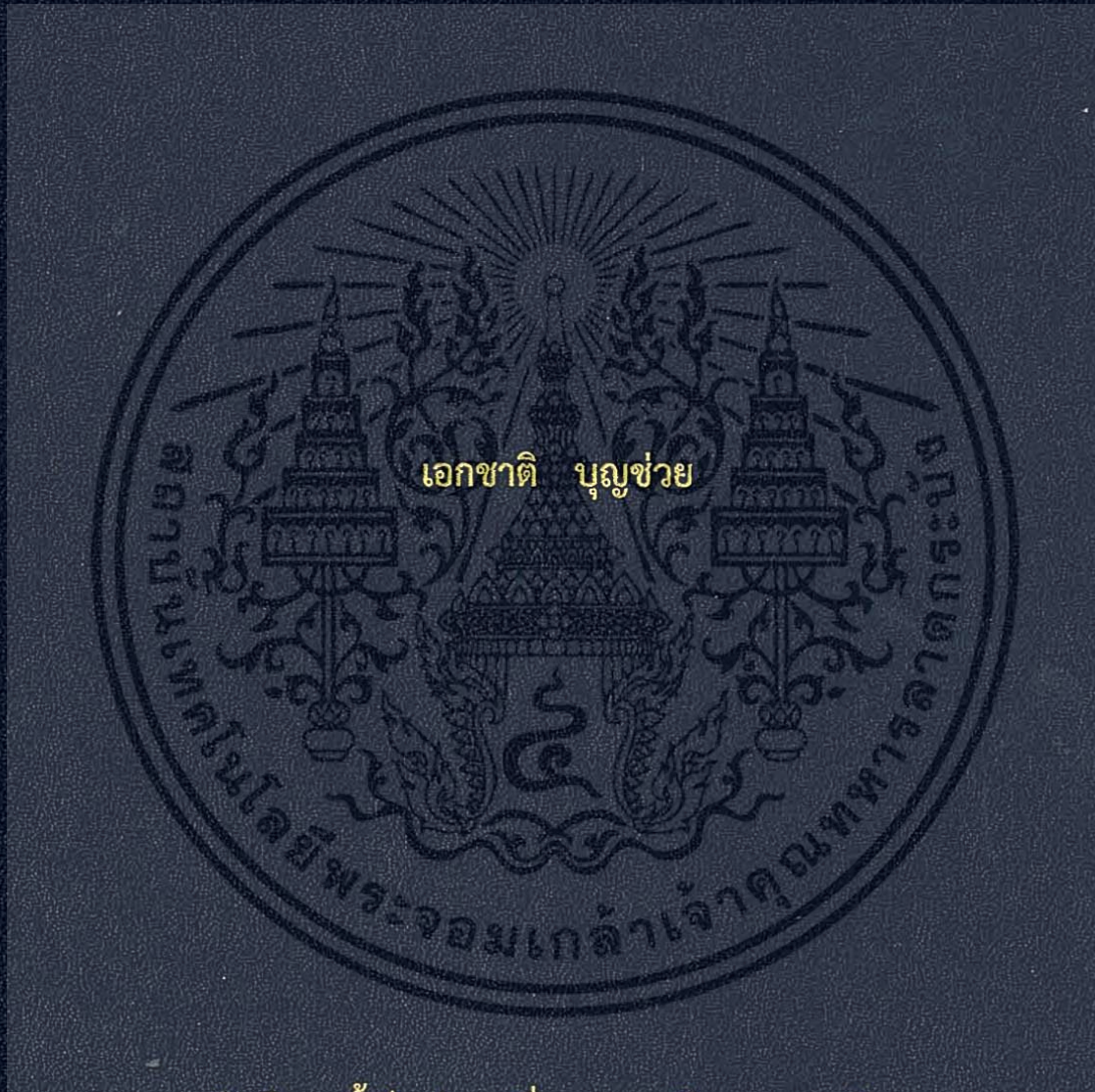


การปรับปรุงระบบบริหารจัดการผลิตภัณฑ์บนเดสก์ท็อป

IMPROVEMENT OF DESKTOP DEPLOYMENT PORTAL



สหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์)
ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2559

การปรับปรุงระบบบริหารจัดการผลิตภัณฑ์บนเดสก์ท็อป

IMPROVEMENT OF DESKTOP DEPLOYMENT PORTAL



สหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์)
ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2559

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

IMPROVEMENT OF DESKTOP DEPLOYMENT PORTAL



A COOPERATIVE EDUCATION SUBMITTED IN PARTIAL
FULFILLMENT OF THE REQUIREMENT 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 2016

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	การปรับปรุงระบบบริหารจัดการการผลิตภัณฑภัณฑ์บนเดสก์ท็อป
ชื่อนักศึกษา	นายเอกชาติ บุญช่วย รหัสนักศึกษา 56050430
ปริญญา	วิทยาศาสตร์บัณฑิต (วิทยาการคอมพิวเตอร์)
ภาควิชา	วิทยาการคอมพิวเตอร์
คณะ	วิทยาศาสตร์
มหาวิทยาลัย	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.)
ปีการศึกษา	2559
อาจารย์ที่ปรึกษา	ดร.ไพรัตน์ ธรเจริญศรี

บทคัดย่อ

โครงการสหกิจศึกษานี้แนะนำเสนอการปรับปรุงและเพิ่มประสิทธิภาพให้กับโปรแกรม Desktop Deployment เพื่อใช้สำหรับจัดการการปล่อยแพ็คเกจข้อมูล, ส่วนอัปเดต และส่วนเสริมต่างๆ ของโปรแกรม Eikon ซึ่งเป็นผลิตภัณฑ์หลักของบริษัทรอยเตอร์ส ซอฟต์แวร์ (ประเทศไทย) จำกัด (มหาชน) โดยทีมผู้พัฒนามีความต้องการให้โปรแกรม Desktop Deployment สามารถรองรับการจัดการแพ็คเกจของโปรแกรม FX Trading ได้เช่นเดียวกับโปรแกรม Eikon รวมไปถึงโปรแกรมอื่นๆที่จะเพิ่มมาในอนาคต เพื่อลดต้นทุน ทรัพยากรและเวลาในการพัฒนา ผู้พัฒนาจึงปรับปรุงโปรแกรมให้สามารถรองรับความต้องการด้านต้นได้ และแก้ไขข้อผิดพลาดบางส่วนที่มีในโปรแกรมด้วย โดยการแก้ไขจะไม่ส่งผลกระทบต่อกระบวนการทำงานเก่าของโปรแกรม โดยในการพัฒนาใช้ภาษา C#, SQL, HTML, JavaScript และ CSS ร่วมกับโปรแกรม Visual Studio 2013 และ Dbeaver ซึ่งโปรแกรม Desktop Deployment ได้ถูกนำไปทดสอบการใช้งานบนเซิร์ฟเวอร์ทดสอบและรอกการอนุมัติ เพื่อปล่อยให้พนักงานนำไปใช้งานต่อไป

คำสำคัญ : ระบบอัปเดต, Eikon, FXT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title	Improvement of Desktop Deployment Portal
Students	Mr. Ekachart Bunchuay Student ID 56050430
Degree	Bachelor of Science (Computer Science)
Department	Computer Science
Faculty	Science
University	King Mongkut's Institute of Technology Ladkrabang (KMITL)
Academic Year	2016
Advisor	Dr. Pairat Thorncharoensri

Abstract

This co-operative project is aimed to improve ability of desktop deployment program. Desktop deployment is used to manage package of Thomson Reuters flagship product name Eikon such as patch, add-on and extension. The company also has several product to serve customer but does not has deployment program. In basic the developer team want desktop deployment program can handle package for product name FX Trading. If it successful this desktop deployment will get develop to support other product of company in the future. Goal of improvement is reduce cost, resource usage and development time. This program developing using several programming language such as C#, SQL, HTML, JavaScript, CSS and use Visual Studio 2013 as main IDE. However this Desktop Deployment program is currently testing on test server and waiting for company approval for release to employee actually use.

Keywords : Update System, Eikon, FXT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปัญหาพิเศษเล่มนี้จะไม่สำเร็จได้เลยหากปราศจากความกรุณาในการให้ออกาสเข้ามาปฏิบัติงานโครงการสหกิจศึกษา ณ บริษัท รอยเตอร์ซอฟต์แวร์ (ประเทศไทย) จำกัด โดยปฏิบัติงานเป็นระยะเวลา 7 เดือน ทำให้ได้รับประสบการณ์การทำงานจริง เรียนรู้ถึงเทคโนโลยีที่ใช้ในการทำงาน และเรียนรู้ข้อเสียของตนเอง เพื่อนำมาพัฒนาปรับปรุงตนเอง ก่อนที่จะก้าวเข้าสู่การทำงานแบบเต็มตัวในอนาคต ขอขอบพระคุณพี่ๆในทีม App server ทุกคน ที่คอยให้คำปรึกษา และอำนวยความสะดวกในการปฏิบัติงานมาโดยตลอด ทำให้การฝึกงานสำเร็จได้ด้วยดี

ขอขอบพระคุณอาจารย์ที่ปรึกษาทุกท่านที่ให้คำแนะนำเกี่ยวกับการปฏิบัติงาน และช่วยแก้ไขปัญหาต่างๆ ระหว่างการทำสหกิจศึกษาจนโครงการสำเร็จลุล่วงไปด้วยดี

ท้ายที่สุด ขอขอบคุณครอบครัว และเพื่อนๆ ที่คอยสนับสนุน ให้คำปรึกษา และให้กำลังใจที่ดีเสมอมา

เอกชาติ บุญช่วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ก
บทคัดย่อภาษาอังกฤษ.....	ข
กิตติกรรมประกาศ.....	ค
สารบัญ.....	ง
สารบัญตาราง.....	ฉ
สารบัญรูป.....	ช
บทที่ 1 บทนำ.....	1
1.1 ที่มาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของงานวิจัย.....	1
1.3 ขอบเขตของงานวิจัย.....	2
1.4 วิธีการดำเนินการโครงการ.....	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	2
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง.....	3
2.1 Web Service.....	3
2.1.1 ความหมายของ Web Service.....	3
2.1.2 องค์ประกอบของ Web Service.....	3
2.1.3 Windows Communication Foundation.....	4
2.2 Web Application.....	6
2.3 Application Programming Interface.....	7
2.4 ภาษาที่ใช้ในการพัฒนา.....	8
2.4.1 ภาษา C#.....	8
2.4.2 ภาษา HTML.....	8
2.4.3 ภาษา CSS.....	9
2.4.4 ภาษา JavaScript.....	10
2.4.5 ภาษา SQL.....	13
2.4.6 ภาษา Gherkin.....	15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
2.5 GIT.....	16
2.6 Rabbit MQ.....	17
2.7 Unit Test.....	17
บทที่ 3 วิธีการดำเนินงานวิจัย.....	19
3.1 ขั้นตอนการศึกษาและรวบรวมข้อมูลของระบบ.....	19
3.2 ขั้นตอนการวิเคราะห์ระบบ.....	19
3.3 ขั้นตอนการออกแบบระบบ.....	20
3.3.1 การออกแบบการทำงานของระบบ.....	20
3.3.2 การออกแบบฐานข้อมูล.....	34
3.4 ขั้นตอนการทดสอบระบบ.....	39
3.4.1 ขอบเขตของการทดสอบระบบส่วนการลงทะเบียนข้อมูล (Register).....	40
3.4.2 ขอบเขตของการทดสอบระบบส่วนการจัดการ Role.....	42
3.4.3 ขอบเขตของการทดสอบระบบส่วนการตรวจสอบ Update.....	43
3.4.4 ขอบเขตของการทดสอบระบบส่วนการสร้างกลุ่ม.....	46
บทที่ 4 ผลการดำเนินงานและการอภิปรายผล	48
4.1 โครงสร้างของระบบ.....	48
4.2 ผลการดำเนินงาน.....	49
4.2.1 ส่วนของผู้ใช้งานทั่วไป.....	49
4.2.2 การเข้าใช้งานส่วน Back Office.....	54
บทที่ 5 สรุปผลการดำเนินงานและข้อเสนอแนะ	57
5.1 สรุปผลการดำเนินงาน.....	57
5.2 สรุปผลการทดสอบระบบ.....	57
5.3 ข้อจำกัดในการพัฒนาระบบ.....	58
5.4 ข้อเสนอแนะในการพัฒนาระบบ.....	58
เอกสารอ้างอิง	59

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
3.1 แสดงรายละเอียดของตาราง Role Member.....	35
3.2 แสดงรายละเอียดของตาราง Installation.....	36
3.3 แสดงรายละเอียดของตาราง Package.....	37
3.4 แสดงรายละเอียดของตาราง User Group Member.....	37
3.5 แสดงรายละเอียดของตาราง Installation Group.....	38
3.6 แสดงรายละเอียดของตาราง Patch.....	38
3.7 แสดงรายละเอียดของตาราง Block Domain.....	38
3.8 แสดงรายละเอียดของตาราง Target Rules.....	39
3.9 แสดงรายละเอียดของตาราง Target Audit Log.....	39
3.10 การทดสอบ Register.....	41
3.11 การทดสอบการจัดการ Role.....	43
3.12 การทดสอบการตรวจสอบ Update.....	44
3.13 การทดสอบการตรวจสอบ Create Group.....	46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 โครงสร้างพื้นฐานของ WCF โดยแบ่งเป็นชั้น (Layer).....	4
2.2 ตัวอย่างการเขียน JavaScript.....	10
2.3 การเขียน JavaScript โดยใช้ jQuery กับการเขียนแบบปกติ.....	11
2.4 เปรียบเทียบหลักการทำงานระหว่างเว็บแอปพลิเคชันทั่วไปและเว็บแอปพลิเคชันที่ใช้ Ajax..	12
2.5 ตัวอย่างการเขียน SQL เบื้องต้น.....	13
2.6 Entity Framework เป็นตัวกลางในการติดต่อระหว่าง Application กับ Database.....	14
2.7 Entity Framework จำลอง Model Entities.....	14
2.8 แสดงตัวอย่างการเขียนภาษา Gherkin.....	15
2.9 โปรแกรม SourceTree ที่ใช้สำหรับจัดการ Git.....	16
2.10 ตัวอย่างการทำงานของ Rabbit MQ	17
2.11 แสดงการทดสอบระดับหน่วยในการพัฒนาแอปพลิเคชัน.....	18
3.1 Use Case Diagram ของระบบ.....	21
3.2 Context Diagram ของระบบ.....	21
3.3 Data Flow Diagram ระดับ 0 ของระบบ.....	23
3.4 Sequence Diagram การเข้าสู่ระบบของพนักงาน.....	24
3.5 Sequence diagram การกำหนดเวอร์ชันและส่วนเสริม.....	25
3.6 Sequence Diagram ฟังก์ชัน Register.....	26
3.7 Sequence Diagram ของฟังก์ชัน Add to group.....	27
3.8 Sequence Diagram ฟังก์ชัน AddToRoles.....	28
3.9 Sequence Diagram ฟังก์ชัน Export.....	29
3.10 Sequence Diagram ฟังก์ชัน Check Update.....	30
3.11 Flow Chart การเข้าใช้งานแอปพลิเคชัน.....	31

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่	หน้า
3.12 Flow Chart การทำงานของฟังก์ชัน Register.....	32
3.13 Flow Chart ฟังก์ชัน Export.....	33
3.14 Entity Relationship ของระบบ.....	34
4.1 โครงสร้างของระบบ.....	48
4.2 หน้าหลักของเว็บแอปพลิเคชัน (1).....	49
4.3 หน้าหลักของเว็บแอปพลิเคชัน (2).....	49
4.4 หน้าต่าง Target Installation.....	50
4.5 การกำหนดเวอร์ชันให้กับ Clients.....	51
4.6 หน้าต่าง Select Installation Groups.....	51
4.7 Toast add to group success.....	52
4.8 หน้า Installation Groups.....	52
4.9 ปุ่ม Export Installation.....	53
4.10 ดาวน์โหลด Export.....	53
4.11 ตัวอย่างไฟล์ Report.....	53
4.12 หน้า Back Office เวอร์ชันเก่า.....	54
4.13 หน้า Back Office เวอร์ชันล่าสุด.....	54
4.14 Menu ของหน้า Back Office เวอร์ชันล่าสุด.....	55
4.15 Dialog การเพิ่ม Package ของโปรแกรม FXT.....	55
4.16 หน้า Report.....	56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของปัญหา

บริษัท รอยเตอร์ส ซอฟต์แวร์ (ประเทศไทย) จำกัด (มหาชน) เป็นองค์กรที่มีการนำเทคโนโลยีทางด้านไอที และการสื่อสารสมัยใหม่มาประยุกต์ใช้กับธุรกิจของตนเองอย่างกว้างขวาง โดยเฉพาะอย่างยิ่งการมีศูนย์พัฒนาซอฟต์แวร์เป็นของตนเอง ทำให้รอยเตอร์สามารถผลิตซอฟต์แวร์คุณภาพสูงเพื่อใช้ภายในองค์กรและจัดจำหน่ายแก่ลูกค้า อีกทั้งยังจัดทำธุรกิจการบริการเผยแพร่ข้อมูลสารสนเทศหรือข้อมูลที่ได้รับการวิเคราะห์และประมวลผลแล้วแก่สาธารณะ โดยมีการแบ่งข้อมูลไว้ทั้งหมด 4 ประเภท ได้แก่ ข่าวและการศึกษา (Intellectual Property and Science) กฎหมาย (Legal) ภาษีและบัญชี (Tax and Accounting) และการเงินและความเสี่ยง (Finance and Risk)

ซึ่งบริษัท รอยเตอร์ส ซอฟต์แวร์ (ประเทศไทย) จำกัด จะทำหน้าที่พัฒนาซอฟต์แวร์สำหรับรองรับข้อมูลทางด้านนี้ วิเคราะห์ ประมวลผล และแสดงผลลัพธ์แก่ลูกค้าในรูปแบบต่างๆผ่านทางหลากหลายโปรแกรม อาทิเช่น โปรแกรมทอมสัน รอยเตอร์ ไอคอน (Thomson Reuters Eikon) โปรแกรมเอฟเอ็กซ์ เทรดดิ้ง (Thomson Reuters FX Trading : FXT) และโปรแกรมช่วยตรวจสอบเครดิต (Thomson Reuters Matching Credit Administrator : MCA) เป็นต้น ซึ่งในแต่ละโปรแกรมจะได้รับการปรับปรุงรุ่นของโปรแกรมให้มีประสิทธิภาพ ลดความผิดพลาด และรักษาความปลอดภัยของข้อมูลอยู่เสมอ อีกทั้งยังมีส่วนเสริม (Add-on) ที่ผู้ใช้สามารถติดตั้งได้ ซึ่งจะแตกต่างกันไปตามเงื่อนไขต่างๆ จึงทำให้การควบคุมมีความยุ่งยาก

ดังนั้นจึงมีระบบเดสก์ท็อปดีพลอยเมนต์พอร์ทัล (Desktop Deployment Portal) ที่คอยควบคุมการให้บริการ โดยที่ในปัจจุบันสามารถควบคุมผลิตภัณฑ์ไอคอนเพียงได้เพียงผลิตภัณฑ์เดียวเท่านั้น จึงทำให้ผลิตภัณฑ์อื่นๆ ยังมีความลำบากในการควบคุมอยู่ จึงได้พัฒนาระบบเดสก์ท็อปดีพลอยเมนต์พอร์ทัลเพิ่มเติม เพื่อให้รองรับกับผลิตภัณฑ์ใหม่ๆ ของบริษัทได้ จะช่วยให้บริษัทสามารถประหยัดทรัพยากร และสามารถดูแลลูกค้าได้ดียิ่งขึ้น

1.2 วัตถุประสงค์ของงานวิจัย

- 1) เพื่อให้ระบบเดสก์ท็อปดีพลอยเมนต์พอร์ทัล รองรับผลิตภัณฑ์ FXT และ MCA รวมไปถึงผลิตภัณฑ์อื่นๆ ในอนาคตของบริษัท
- 2) เพื่อลดต้นทุนในการสร้างระบบใหม่ โดยทำการพัฒนาต่อยอดจากระบบเดิมที่มีอยู่แล้ว และประหยัดทรัพยากรของเซิร์ฟเวอร์
- 3) เพื่ออำนวยความสะดวกให้พนักงานที่อยู่ในทีมพัฒนาซอฟต์แวร์ ที่มีความต้องการติดตาม

ข้อมูลและสถานะของผลิตภัณฑ์ที่ติดตั้งบนเครื่องลูกค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของงานวิจัย

- 1) โปรแกรมสามารถจำแนกผลิตภัณฑ์ตามที่ใช้ต้องการได้อย่างถูกต้อง
- 2) โปรแกรมสามารถจำแนกสิทธิ์การได้รับอนุญาตของผู้ใช้แต่ละคน ในการเข้าควบคุมแต่ละผลิตภัณฑ์ได้อย่างถูกต้อง
- 3) ผู้ใช้สามารถสร้างกลุ่มเพื่อแยกกลุ่มของลูกค้าได้ โดยกลุ่มที่สร้างจะไม่ซ้ำซ้อนกับกลุ่มของผลิตภัณฑ์อื่นๆ
- 4) ผู้ใช้งานที่มีสิทธิ์เป็นผู้ดูแล สามารถกำหนดสิทธิ์ให้กับผู้ใช้อื่นๆได้
- 5) ผู้ใช้งานที่มีสิทธิ์เป็นผู้ดูแล สามารถ เพิ่ม แก้ไข ลบ รายละเอียดของรุ่นและส่วนเสริมของผลิตภัณฑ์ได้

1.4 วิธีการดำเนินการโครงการ

- 1) สอบถามความต้องการของผู้ใช้งานระบบ และวิธีใช้งานระบบ
- 2) ทดลองใช้งานระบบ เพื่อให้มีความเข้าใจมากขึ้น
- 3) ทำการแก้ไขและเพิ่มฟังก์ชันการใช้งานให้ตรงกับความต้องการของผู้ใช้งาน
- 4) ทดสอบการใช้งาน หากเกิดข้อผิดพลาดขึ้นจะต้องทำการแก้ไข และทดสอบการใช้งานจนกว่าจะไม่มีข้อผิดพลาดเกิดขึ้น
- 5) อัปเดตระบบขึ้นเซิร์ฟเวอร์ เพื่อให้ผู้ต้องการใช้งาน

1.5 ประโยชน์ที่คาดว่าจะได้รับ

- 1) สามารถทำให้พนักงานควบคุมจัดการผลิตภัณฑ์ FXT และ MCA ได้ดียิ่งขึ้น
- 2) อำนวยความสะดวกให้พนักงานสามารถให้บริการลูกค้าได้เร็วขึ้น และมีประสิทธิภาพเพิ่มมากขึ้น เพราะเห็นข้อมูลทั้งหมดในระบบเดียว
- 3) องค์กรสามารถประหยัดต้นทุน และลดการใช้ทรัพยากรลงได้ เพราะใช้ระบบเดียวกันทั้งหมดในการควบคุม โดยไม่ต้องสร้างระบบใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 Web Services

2.1.1 ความหมายของ Web Service

Web Services คือระบบซอฟต์แวร์ที่ออกแบบมาเพื่อสนับสนุนการแลกเปลี่ยนข้อมูลระหว่างเครื่องคอมพิวเตอร์ผ่านระบบเครือข่าย โดยที่ภาษาที่ใช้ในการติดต่อสื่อสารระหว่างเครื่องคอมพิวเตอร์ คือ XML หรือ JSON เว็บเซอร์วิสมีอินเทอร์เน็ตเฟสที่ใช้อธิบายรูปแบบข้อมูลที่เครื่องคอมพิวเตอร์ประมวลผลได้ ลักษณะการให้บริการของ Web Services นั้น จะถูกเรียกใช้งานจาก application อื่นๆ ในรูปแบบ RPC (Remote Procedure Call) ซึ่งการให้บริการจะมีเอกสารที่อธิบายคุณสมบัติของบริการกำกับไว้ โดยภาษาที่ถูกใช้เพื่อในการแลกเปลี่ยนคือ XML ทำให้สามารถเรียกใช้ Component ใดๆ ก็ได้ในระบบหรือPlatform ใดๆก็ได้ บน Protocol HTTP ซึ่งเป็น Protocol สำหรับ World Wide Web หรืออินเทอร์เน็ต อันเป็นช่องทางที่ได้รับการยอมรับทั่วโลกในการติดต่อสื่อสารกันระหว่างApplication กับ Application ในปัจจุบัน [1]

2.1.2 องค์ประกอบของ Web Service

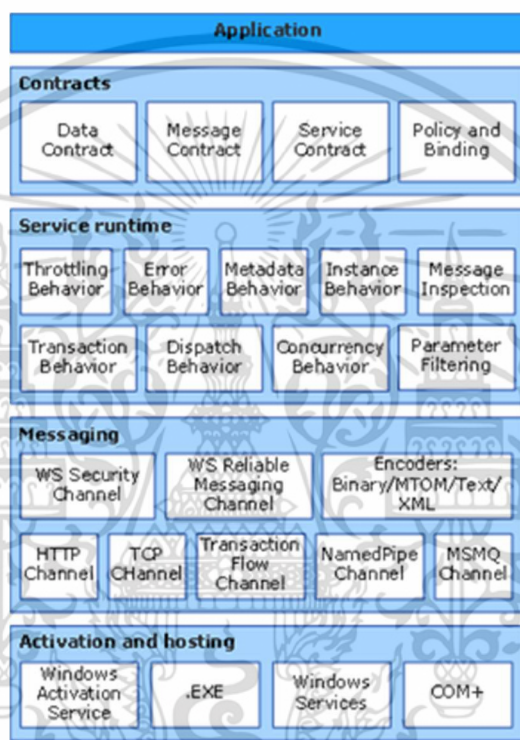
ผู้ให้บริการ (Service Provider) : ผู้ให้บริการจะมีหน้าที่ในการพัฒนาและติดตั้งเว็บเซอร์วิส และเป็นผู้ที่นิยามความหมายของเซอร์วิสและลงทะเบียนเซอร์วิสกับ Service Registry

ผู้ใช้บริการ (Service Requestor) : ผู้ใช้บริการจะเป็นผู้เรียกใช้เว็บเซอร์วิส โดยอาจทำการค้นหาเซอร์วิสจากเซอร์วิสไดเร็กทอรี แล้วทำการเรียกใช้เซอร์วิสจากผู้ให้บริการ

ผู้ลงทะเบียน (Service Registry) : หรืออาจเรียกว่า Service Broker มีหน้าที่ในการรับลงทะเบียนและช่วยในการค้นหาเว็บเซอร์วิส Service Registry จะเก็บรายละเอียดของเว็บเซอร์วิสต่างๆเช่น นิยาม และตำแหน่งของเว็บเซอร์วิส ทำหน้าที่คล้ายกับสมุดโทรศัพท์เพื่อช่วยให้ผู้ใช้บริการสามารถค้นหาเซอร์วิสที่ต้องการได้

2.1.3 Windows Communication Foundation

Windows Communication Foundation (WCF) มีวัตถุประสงค์คือ เพื่อช่วยเพิ่มความสามารถของ ASP.NET Web Service ที่มีข้อจำกัดในด้านเครื่องมือและเทคโนโลยีที่มีความแตกต่าง จะทำให้สื่อสารกันได้ลำบาก ไมโครซอฟต์พัฒนา WCF ขึ้นเพื่อรองรับลักษณะงานประเภท SOA ในกรณีที่มีระบบหรือผู้ใช้ที่มีความหลากหลายในด้านมาตรฐานการสื่อสาร (Protocol) ให้สามารถสื่อสารกันได้เป็นรูปแบบเดียวกัน จะช่วยให้มีประสิทธิภาพการทำงานที่รวดเร็วขึ้นกว่าเดิม สามารถนำ Service ที่พัฒนาไว้นั้นไปใช้งานบน Host ชนิดต่างๆ ได้ [2]



รูปที่ 2.1 โครงสร้างพื้นฐานของ WCF โดยแบ่งเป็นชั้น (Layer)

- Application Layer คือส่วนที่ทำการ implement code ตัวเซอร์วิสไว้ เช่น method, code logic, entity ต่างๆ ที่ใช้ในการทำงานของตัวเซอร์วิส
- Contract Layer เป็นชั้นที่เชื่อมต่อกับ Application Layer ซึ่งนักพัฒนาจะทำงานโดยตรงกับส่วนนี้เพื่อใช้ในการสร้างตัวเซอร์วิสต่างๆ สามารถแบ่งได้ ดังนี้
 - 1) Data Contract เป็นส่วนที่กำหนดโครงสร้างข้อมูลต่างๆ ที่มีรับส่งข้อมูลทั้งเข้าและออกโดย Operation Contract ของ Service Contracts ซึ่งอาจจะประกอบไปด้วย ชนิดข้อมูลพื้นฐาน (Primitive Data Type) เช่น string, int, decimal, float, double, boolean และกลุ่มข้อมูลโครงสร้างที่ซับซ้อน (Complex Data Type) อย่างเช่น Class, Enumeration Data ต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) Message Contract โดยปกติข้อมูลที่ส่งข้ามกันระหว่าง Service และ Client นั้น ถูกกำหนดรูปแบบอยู่ใน SOAP Message แต่บางครั้งเนื่องจากความต้องการในการรองรับการทำงานต่างๆ เช่น ต้องการจัดรูปแบบโครงสร้างข้อมูลใหม่, อยากรู้ Header, การเพิ่มเติม Status เหล่านี้ เราจะต้องอาศัย Message Contract เป็นตัวช่วย

3) Service Contracts ส่วนที่บ่งบอกว่าเซอร์วิสที่เราสร้างขึ้นนั้นมี operation (Method) อะไรไว้ให้บริการบ้าง

4) Policies and Binding ส่วนนี้เป็นตัวกำหนดตั้งค่าเงื่อนไขและวิธีการให้เซอร์วิสมีการติดต่อสื่อสารในรูปแบบใด (Protocol) ใช้ระบบรักษาปลอดภัย (Security) แบบไหน

- Service Runtime ส่วนที่กำหนดพฤติกรรม (Behavior) การทำงานของเซอร์วิส ขณะที่เซอร์วิสทำงานอยู่ มีสามารถแบ่งได้ ดังนี้

1) Throttling Behavior ใช้กำหนดจำนวนของ Message ที่จะเข้ามาทำงานในเซอร์วิส

2) Error Behavior หากมีปัญหาเกิดขึ้นภายในเซอร์วิส ใช้มาช่วยทำการจัดการปัญหาดังกล่าว

3) Metadata Behavior ใช้กำหนดการเปิดเผยข้อมูลโครงสร้าง (Meta Data) ของเซอร์วิสแก่ผู้ใช้งาน

4) Instance Behavior ใช้กำหนดจำนวนของผู้ใช้ที่จะเข้ามาใช้งานในเซอร์วิสโดยพร้อม ๆ กันขณะที่เซอร์วิสทำงานอยู่

5) Transaction Behavior ช่วยในการย้อนกลับ (Rollback) การทำงานหาก Operation เกิดการทำงานที่ผิดพลาดขึ้น

6) Dispatch Behavior เป็นตัวควบคุมว่า Message ต่างๆ นั้นจะเข้าการโปรเซสด้วยสถาปัตยกรรมใดของ WCF

- Messaging Layer เป็นการจัดการ Message ที่ส่งเข้าออกภายในเซอร์วิส ทั้งการกำหนดรูปแบบของ Message โดยการใช้วิธีการแลกเปลี่ยน Message ซึ่งแบ่งได้เป็นสองกลุ่มดังนี้

1) Transport channels ทำการอ่านและเขียน Message จากเน็ตเวิร์กภายในหรือภายนอกองค์กร มี transports บางตัวจำเป็นต้องมีการแปลง (encoder) Message จากหน่วยสตรีมไบท์ที่ใช้รับส่งกันในเน็ตเวิร์กมาอยู่ในรูป XML เช่น HTTP, named pipes, TCP, และ MSMQ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) Protocol channels เป็นการทำงานโปรโตคอลระดับ Message โดยส่วนใหญ่เป็นการเขียนอ่านในส่วนของหัว Message เช่น WS-Security Protocol เป็นการกำหนดใช้ความปลอดภัยในระดับชั้น Message และ WS-Reliability Protocol เป็นการการันตีการส่ง Message ระหว่างกัน

- Activation and Hosting เซอร์วิสที่ได้สร้างไว้เป็นเพียง Library เท่านั้น เมื่อต้องการนำไปรันบน Host เพื่อให้เซอร์วิสสามารถใช้งานได้ ซึ่งมีหลายวิธีดังนี้

1) Windows Activation Service (WAS) มาพร้อมกับ IIS 7.0 จะทำงานเมื่อมีการร้องขอ ร่องรับได้หลายโปรโตคอล HTTP, TCP, Named Pipes, MSMQ

2) Self-Hosting สามารถแปลงให้อยู่ในรูป exe ได้เช่น console app, win app, WPF app เป็นต้น ไม่สามารถทำงานอัตโนมัติได้ ต้องรันโปรแกรมทุกครั้งเพื่อให้ Host ทำงาน

3) Windows Service คล้ายกับ Executable Host แต่จะทำงานอัตโนมัติทุกครั้งที่เราเริ่มระบบ Windows

4) COM+ ส่วนใหญ่ไม่มีคนได้ใช้งาน จะเกี่ยวกับ Windows Component Programming

2.2 Web Application

Web Application เป็นแอปพลิเคชันที่สามารถเรียกใช้งานผ่านบราวเซอร์ต่างๆ โดยที่ผู้ใช้งานไม่ต้องติดตั้งแอปพลิเคชันเหล่านี้ ลงไปในเครื่องคอมพิวเตอร์ Web Application จะถูกติดตั้งไว้ที่ Server คอยให้บริการกับ Client ทำให้สามารถเรียกใช้งานได้ทุกที่ทุกเวลาผ่านระบบอินเทอร์เน็ต [3]

ข้อดีของ Web Application

- เหมาะกับองค์กรขนาดเล็กเพราะมีค่าใช้จ่ายต่ำ
- การใช้งานในองค์กรทำได้ง่าย เพียงแค่มีเว็บเบราว์เซอร์ก็สามารถใช้งานได้
- ข้อมูลจัดเก็บที่เดียว ง่ายต่อการจัดการ และไม่เกิดความซ้ำซ้อน
- ไม่จำเป็นต้องใช้เครื่องคอมพิวเตอร์ที่มีประสิทธิภาพสูงซึ่งมีราคาแพง
- สามารถทำงานได้ทันที โดยไม่ต้องติดตั้งโปรแกรม
- ไม่ต้องมีบุคลากรด้านเทคนิคเป็นของตัวเอง เพราะผู้ให้บริการดูแล Server และการบำรุงรักษาเองทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ส่วนมากใช้ได้หลากหลายระบบปฏิบัติการ เช่น Windows, Linux และ Mac ทำให้องค์กรสามารถเลือกใช้ และลดค่าใช้จ่ายได้
- เชื่อมต่อกับ Web Application ตัวอื่นหรือบริการออนไลน์อื่นๆได้ง่าย

ข้อเสียของ Web Application

- การใช้งานมีได้จำกัด อาจไม่เหมาะกับงานบางประเภทที่ต้องการรูปแบบโปรแกรมที่แตกต่างจากโปรแกรมทั่วไปเช่น โปรแกรมตกแต่งรูป โปรแกรมตัดต่อวิดีโอ
- Web Application ส่วนใหญ่ต้องการอินเทอร์เน็ตในการใช้งานเสมอ แต่บางตัวที่สามารถทำงานแบบออฟไลน์ได้

2.3 Application Programming Interface

Application Programming Interface (API) คือช่องทางการเชื่อมต่อระหว่างเว็บไซต์หนึ่งไปยังอีกเว็บไซต์หนึ่ง หรือเป็นการเชื่อมต่อระหว่างผู้ใช้งานกับ Server หรือจาก Server เชื่อมต่อไปหา Server ซึ่ง API เปรียบได้เป็นภาษาคอมพิวเตอร์ที่ทำให้คอมพิวเตอร์สามารถสื่อสารและแลกเปลี่ยนข้อมูลกันได้อย่างอิสระ

API ทำหน้าที่ช่วยในการเข้าถึงข้อมูลต่างๆ หรือจะเป็นการจัดการข้อมูลต่างๆบนเว็บไซต์ โดยเจ้าของเว็บไซต์ที่มี API สามารถกำหนดขอบเขตในการเข้าถึงบริการต่างๆของทางเว็บไซต์ได้ ประโยชน์ของ API สามารถแบ่งออกมาได้หลายอย่างคือ

- ช่วยในการพัฒนาเว็บไซต์หรือ Application ได้ง่ายและรวดเร็วซึ่ง API จะเป็นตัวช่วยที่นักพัฒนาไม่ต้องเข้าไปแก้ไข Code คำสั่งเลยทำให้สะดวกสบายในการทำงาน
- ช่วยให้นักพัฒนาเว็บไซต์หรือเจ้าของเว็บไซต์สามารถขยายฐานผู้ชมเว็บไซต์ให้มากขึ้น
- ทำให้ผู้ใช้งานเว็บไซต์ต่างๆที่มีการติดตั้ง API ของอีกเว็บไซต์หนึ่ง ไม่ต้องเข้าหน้าเว็บไซต์ที่เป็นเจ้าของ API เกิดสะดวกในการทำงานของผู้ใช้งานเว็บไซต์
- API สามารถรับส่งข้อมูลข้าม Server ได้

ปัจจุบันเว็บไซต์ขนาดใหญ่ หลายเว็บไซต์จะมีการเปิดให้ใช้งาน API ซึ่งสามารถเห็นการใช้งาน API ได้มากขึ้นโดยเฉพาะเว็บไซต์ที่เน้นด้านการติดต่อสื่อสาร Social Network และ E-commerce เช่น บริการของ Amazon มี API ที่เปิดให้ผู้สนใจได้นำสินค้าที่มีขายอยู่ใน Amazon ไปติดไว้ในเว็บไซต์หรือบล็อกของตัวเองได้ และบริการของ Facebook ที่สามารถให้ผู้ผู้ใช้โปรโมตเพจของตนเองผ่านเว็บไซต์หรือบล็อกได้ [4]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 ภาษาที่ใช้ในการพัฒนา

2.4.1 ภาษา C#

C# คือ ภาษาคอมพิวเตอร์ประเภทการเขียนโปรแกรมเชิงวัตถุ (Object - Oriented Programming หรือ OOP) พัฒนาโดยบริษัทไมโครซอฟท์ (Microsoft) โดยมีจุดมุ่งหมายในการรวมความสามารถการคำนวณของ C++ ด้วยการโปรแกรมที่ง่ายกว่าของ Visual Basic โดย C# มีพื้นฐานจาก C++ และมีส่วนการทำงานคล้ายกับภาษา Java

C# ได้รับการออกแบบให้ทำงานร่วมกับ .NET platform ของไมโครซอฟท์ จุดมุ่งหมายคืออำนวยความสะดวกในการแลกเปลี่ยนสารสนเทศและบริการผ่านเว็บ และทำให้ผู้พัฒนาสร้างโปรแกรมประยุกต์ในขนาดเล็ก C# ทำให้โปรแกรมสื่อสารง่ายขึ้นผ่านการใช้ Extensible Markup Language (XML) และ Simple Object Access Protocol (SOAP) ซึ่งยอมให้เข้าถึงวัตถุ (Object) ของโปรแกรมหรือเมธอด โดยปราศจากความต้อการให้ผู้เขียนโปรแกรมเขียนคำสั่งเพิ่มในแต่ละขั้นตอน เนื่องจากผู้เขียนโปรแกรมสามารถสร้างบนคำสั่งที่มีอยู่ แทนที่การคัดลอกซ้ำ C# ภาษา C# ถูกพัฒนาขึ้นโดยเป็นส่วนหนึ่งในการพัฒนาโครงสร้างพื้นฐานของ .NET Framework เป็นการการนำข้อดีของภาษาต่างๆ (เช่นภาษา Delphi, ภาษา C++) มาปรับปรุงเพื่อให้มีความเป็นโปรแกรมเชิงวัตถุมากขึ้น ขณะเดียวกันก็ลดความซับซ้อนในโครงสร้างของภาษาลง ซึ่งเรียบง่ายกว่าภาษา C++ และมีสิ่งที่เกินความจำเป็นน้อยลงเมื่อเทียบกับJava

C# ถูกรับรองจากหน่วยงาน ECMA ซึ่งเป็นหน่วยงานกำหนดมาตรฐานสากลด้านเทคโนโลยีสารสนเทศ และ ISO และปัจจุบันไมโครซอฟท์ยังพัฒนาภาษานี้อย่างต่อเนื่อง [5]

2.4.2 ภาษา HTML

HTML หรือ Hyper Text Markup Language เป็นภาษามาร์กอัพ (Markup) ที่ใช้ในการสร้างเว็บเพจ ถูกพัฒนาและกำหนดมาตรฐานโดยองค์กร World Wide Web Consortium (W3C) สำหรับการสร้างเว็บเพจโดยใช้ภาษา HTML สามารถทำได้โดยใช้โปรแกรม Text Editor หรือเครื่องมือช่วยสร้างเว็บเพจ เช่น Dreamweaver ส่วนการเรียกใช้งานหรือทดสอบการทำงานของเอกสาร HTML จะใช้เบราว์เซอร์ เช่น Chrome, Internet explorer, Firefox เป็นต้น [6]

โครงสร้างของ HTML จะประกอบไปด้วยส่วนของคำสั่ง 2 ส่วน คือ ส่วนที่เป็นส่วนหัว (Head) และส่วนที่เป็นเนื้อหา (Body) โดยมีรูปแบบคำสั่งดังนี้

1) คำสั่งหรือแท็ก (Tag)

แท็กคือคำสั่งที่ใช้ในภาษา HTML อยู่ภายในเครื่องหมายน้อยกว่า (<) และเครื่องหมายมากกว่า (>) แท็กในภาษา HTML แบ่งออกเป็นสองชนิดดังนี้

- แท็กเดี่ยว เป็นแท็กที่ไม่ต้องมีแท็กปิด เช่น <HR>,
 เป็นต้น
- แท็กคู่ เป็นแท็กที่ประกอบด้วยแท็กเปิดและแท็กปิด โดยแท็กปิดจะต้องใส่เครื่องหมายทับ

(/) นำหน้าคำสั่งในแท็กนั้น ๆ เช่น, <p>.....</p> เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์หรือการแจ้งในเพิกการลิขสิทธิ์เท่านั้น มิใช่ข้อมูลเปิดเผยต่อสาธารณะโดยไม่ขออนุญาต
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่งที่สำคัญยกตัวอย่าง เช่น

คำสั่ง <HTML> เป็นคำสั่งเริ่มต้นในการเขียนโปรแกรมและคำสั่ง </HTML> เป็นการสิ้นสุดโปรแกรม คำสั่งนี้จะไม่แสดงผลในโปรแกรมเว็บเบราว์เซอร์ แต่ต้องเขียนเพื่อให้เกิดความเป็นระบบของงาน และเพื่อแจ้งให้เว็บเบราว์เซอร์ทราบว่าเอกสารนี้เป็นเอกสารของภาษา HTML

คำสั่ง <HEAD> คือ ส่วนหัวเป็นส่วนที่ใช้อธิบายข้อมูลเฉพาะของหน้านั้นๆ เช่น ชื่อเรื่องของหน้าเว็บ (Title) ชื่อผู้จัดทำเว็บ (Author) คีย์เวิร์ดสำหรับการค้นหา (Keyword) เป็นต้น

คำสั่ง <BODY> คือส่วนเนื้อหาเป็นส่วนเนื้อหาหลักของหน้าเว็บ ซึ่งการแสดงผลจะต้องใช้แท็กจำนวนมาก ขึ้นอยู่กับลักษณะของข้อมูล เช่น ข้อความ รูปภาพ เสียง วิดีโอ หรือไฟล์ต่างๆ

2) แอททริบิวต์ (Attribute)

แอททริบิวต์เป็นส่วนขยายในแท็ก ใช้สำหรับจัดรูปแบบเพิ่มเติม เช่น ขนาด, สี, ระยะห่าง เป็นต้น ค่าของแอททริบิวต์จะอยู่ในเครื่องหมาย “...” เช่น

<p align="right"> ข้อความในย่อหน้านี้จัดวางในตำแหน่งชิดขวาหน้าจอ</p>

 ไปยังเว็บไซต์ google.com

2.4.3 ภาษา CSS

CSS ย่อมาจาก Cascading Style Sheets เป็นภาษาที่มีรูปแบบการเขียน Syntax ที่เฉพาะและถูกกำหนดมาตรฐานโดย W3C (World Wide Web Consortium) เช่นเดียวกับภาษา HTML และ XHTML ใช้สำหรับตกแต่งเอกสาร HTML/ XHTML ให้มีหน้าตา สี สัน ตัวอักษร เส้นขอบ พื้นหลัง ระยะห่าง ฯลฯ อย่างที่เราต้องการ ด้วยการกำหนดคุณสมบัติให้กับ Element ต่างๆ ของ HTML เช่น <body>, <p>, <h1> เป็นต้น [7]

ประโยชน์ของ CSS มีคร่าวๆ ดังนี้

- การใช้ CSS ในการจัดรูปแบบการแสดงผล จะช่วยลดการใช้ภาษา HTML ในการตกแต่งเว็บเพจ ทำให้ code ใน HTML เหลือเพียงส่วนเนื้อหา ทำให้เข้าใจง่ายขึ้น การแก้ไขเอกสารทำได้ง่ายและรวดเร็ว
- เมื่อ code ภายในเอกสาร HTML ลดลง ทำให้ขนาดไฟล์เล็กลง ดาวน์โหลดได้เร็วขึ้น
- สามารถกำหนดรูปแบบการแสดงผลจากคำสั่ง style sheet ชุดเดียวกัน ให้มีผลกับเอกสาร HTML ทั้งหน้า หรือทุกหน้าได้ ทำให้เวลาแก้ไขหรือปรับปรุงทำได้ง่าย ไม่ต้องตามแก้ที่ tag ต่างๆ ทั่วทั้งเอกสาร
- สามารถควบคุมการแสดงผลให้เหมือนกัน หรือใกล้เคียงกันได้หลายเบราว์เซอร์
- สามารถกำหนดการแสดงผลในรูปแบบที่เหมาะสมกับสื่อชนิดต่างๆ ไม่ว่าจะเป็นการแสดงผลบน

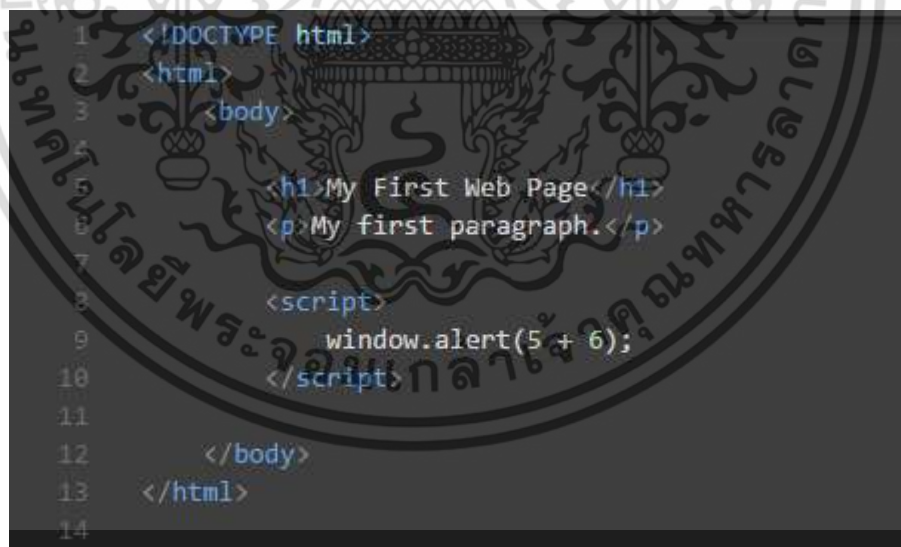
เอกสารนี้เป็นเอกสารหน้าจอบนกระดานเมื่อสั่งพิมพ์, บนมือถือ หรือบน PDA โดยที่เป็นเนื้อหาเดียวกันกับการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.4 ภาษา JavaScript

JavaScript เป็นภาษาสคริปต์ที่มีลักษณะการเขียนเชิงโพรโทไทป์ (Prototype-based Programming) ได้รับการพัฒนาขึ้นโดย Netscape โดยมีวัตถุประสงค์ เพื่อช่วยให้เว็บเพจสามารถแสดงเนื้อหาที่มีการเปลี่ยนแปลงได้ตามเงื่อนไขหรือสภาพแวดล้อมต่างๆ หรือสามารถตอบโต้ตอบกับผู้ใช้งานได้มากขึ้น ทั้งนี้เพราะภาษา HTML ไม่สามารถโต้ตอบกับผู้ใช้งานได้ทันที แต่ใช้แสดงผลเอกสารที่มีเนื้อหาคงที่แน่นอน

การทำงานของ JavaScript เกิดขึ้นบนบราวเซอร์ (Client-Side Script) มีข้อจำกัดคือ ไม่สามารถรับและส่งข้อมูลต่างๆ กับเซิร์ฟเวอร์ได้โดยตรง เช่น การอ่านไฟล์จากเซิร์ฟเวอร์ เพื่อนำมาแสดงบนเว็บเพจ หรือรับข้อมูลจากผู้ใช้งาน เพื่อไปเก็บบนเซิร์ฟเวอร์ เป็นต้น ดังนั้นงานในลักษณะดังกล่าว ต้องอาศัยภาษาฝั่งเซิร์ฟเวอร์ เช่น PHP, Perl, ASP เป็นต้น การเขียน JavaScript ทำได้โดยเขียนให้อยู่ภายในคำสั่ง `<script>...</script>` หรือจะเขียนแยกในอีกไฟล์ต่างหากก็ได้ แล้วเรียกใช้ด้วยแอทริบิวต์ `src` โดยเรียกไปที่ Path ที่เก็บไฟล์ JavaScript ไว้

ปัจจุบันมีการใช้ JavaScript ที่ฝังอยู่ในเว็บเบราว์เซอร์หลายรูปแบบ เช่น ใช้เพื่อสร้างเนื้อหาที่เปลี่ยนแปลงเสมอภายในเว็บเพจ ใช้เพื่อตรวจสอบความถูกต้องของข้อมูลที่ผู้ใช้กรอกก่อนบันทึกลงระบบ ใช้เพื่อเข้าถึงข้อมูลที่อยู่ภายใต้โครงสร้างแบบ DOM เป็นต้น JavaScript มีให้เลือกใช้มากมาย ทั้ง Framework และ Library เช่น Angular, jQuery เป็นต้น [8]



```

1  <!DOCTYPE html>
2  <html>
3  <body>
4
5  <h1>My First Web Page</h1>
6  <p>My first paragraph.</p>
7
8  <script>
9      window.alert(5 + 6);
10 </script>
11
12 </body>
13 </html>
14

```

รูปที่ 2.2 ตัวอย่างการเขียน JavaScript

- ไลบรารี jQuery

jQuery คือไลบรารีของภาษา JavaScript ซึ่งถูกออกแบบมาเพื่อให้การเขียน JavaScript นั้นง่ายขึ้น เพราะสามารถเขียนโค้ดต่างๆ ของ JavaScript ให้สั้นลงได้ jQuery ถูกพัฒนาโดย John

Resig โดยเป็นการพัฒนาแบบโอเพนซอร์ส [9]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณสมบัติหลักของ jQuery มีดังนี้

- การเข้าถึง HTML และ CSS
ส่วนนี้ถือได้ว่าเป็นคุณสมบัติหลักของ jQuery เพราะทำให้ผู้ใช้งานสามารถเข้าถึงอีลีเมนต์ของ HTML และตัวเลือก CSS ต่างๆ ได้ง่ายขึ้น
- การจัดการ HTML และ CSS
คุณสมบัตินี้จะใช้ในการดัดแปลง แก้ไข เพิ่ม หรือลบอีลีเมนต์ต่างๆ สามารถทำได้ง่ายขึ้น
- เหตุการณ์ (Events) ใน JavaScript
สำหรับ JavaScript จะพบปัญหาในเรื่องของเหตุการณ์บ่อยครั้ง เนื่องจากความไม่เข้ากันของเว็บเบราว์เซอร์ แต่เมื่อใช้ jQuery ปัญหาที่เกิดขึ้นจะลดน้อยลงหรือหมดไป เพราะ jQuery จะช่วยตรวจสอบชนิดของเว็บเบราว์เซอร์ให้โดยอัตโนมัติ
- วิวอลเอฟเฟกต์และแอนิเมชัน (Visual Effects and Animation)
วิวอลเอฟเฟกต์และแอนิเมชันเป็นอีกเทคนิคหนึ่งที่กำลังได้รับความนิยมสำหรับการทำเว็บไซต์ในปัจจุบัน
- เอแจ็กซ์ (Ajax)
เอแจ็กซ์ก็เป็นอีกหนึ่งรูปแบบของการเขียนเว็บแอปพลิเคชันในปัจจุบัน โดย jQuery มีฟังก์ชันสำหรับการเรียกใช้งานเอแจ็กซ์ให้ง่ายขึ้นกว่าเดิม
- ส่วนประสานงานผู้ใช้ (User Interface)
ส่วนประสานงานผู้ใช้คือส่วนควบคุมต่างๆ ที่ควรมีหรืออยากให้มี เช่น เมนู (Menu) ปฏิทิน (Calendar) เป็นต้น ทำให้ผู้ใช้งานใช้งานได้สะดวกมากยิ่งขึ้น

jQuery

```
$('body').css('background', '#ccc');
```

JavaScript

```
Function changeBackground(color) {
```

```
    Document.body.style.background = color;
```

```
}
```

```
Onload="changeBackground('red');"
```

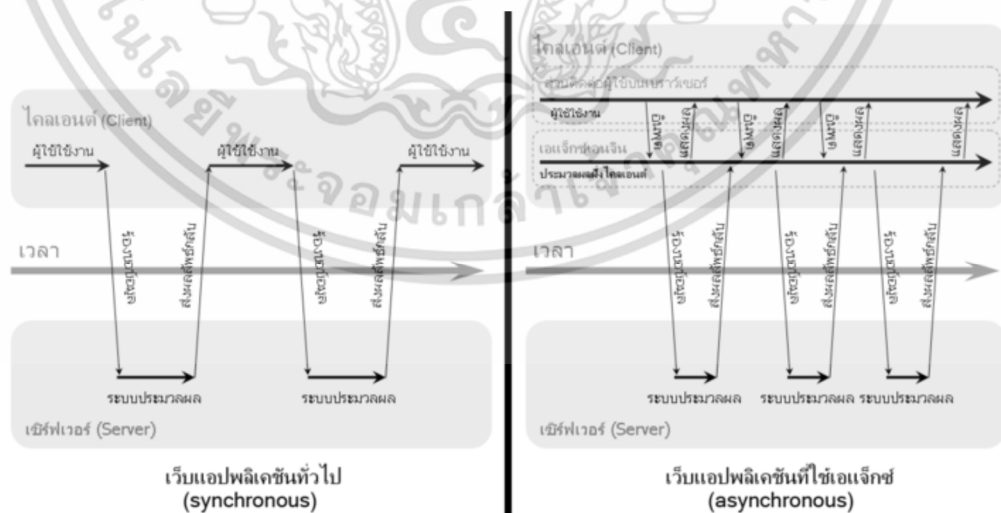
รูปที่ 2.3 การเขียน JavaScript โดยใช้ jQuery กับ การเขียนแบบปกติ
เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับใช้เพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้เผยแพร่ไปยังเว็บไซต์อื่นเป็นการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.3 จะเห็นได้ว่าการเขียนด้วยการใช้ jQuery เข้ามาช่วย จะทำให้ลดบรรทัดในการเขียนลงได้มาก เมื่อเทียบกับการเขียนด้วย JavaScript แบบปกติ และได้ผลลัพธ์ที่เหมือนกัน

- Ajax (Asynchronous JavaScript and XML)

Ajax (Asynchronous JavaScript and XML) เป็นกลุ่มของเทคนิคในการพัฒนาเว็บแอปพลิเคชัน เพื่อให้ความสามารถในด้านการโต้ตอบกับผู้ใช้งานได้ดีขึ้น โดยการรับส่งข้อมูลจะอยู่ในส่วนหลังบ้าน (Backend) ทำให้หน้าเว็บเพจไม่ต้องโหลดใหม่ทั้งหมดทุกครั้งที่มีการเปลี่ยนแปลง ส่งผลให้หน้าเว็บเพจสามารถตอบสนองต่อผู้ใช้งานได้รวดเร็วมากยิ่งขึ้น Ajax ไม่ใช่เทคโนโลยีที่คิดค้นขึ้นมาใหม่ แต่เป็นเทคนิคที่นำเทคโนโลยีหลายอย่างที่มีอยู่แล้วมารวมกันดังต่อไปนี้

- การแสดงผลลัพธ์และรูปแบบข้อมูล ใช้ภาษา HTML และ CSS
 - การแสดงข้อมูลที่มีการเปลี่ยนแปลงให้โต้ตอบกับผู้ใช้งานทันที โดยใช้ ECMAScript เป็นมาตรฐาน เช่น ภาษา JavaScript ในการเข้าถึง เช่น ดอม (Document Object Model: DOM) เป็นต้น
 - การเลือกเปลี่ยนข้อมูลโดยที่ช่วงเวลาไม่สอดคล้องกันกับเซิร์ฟเวอร์ โดยใช้ XHR (XMLHttpRequest)
 - รูปแบบข้อมูลที่ใช้ในการแลกเปลี่ยน เช่น XML, JSON, EBML หรือ plain text
- การทำงานของเว็บแอปพลิเคชันที่ใช้เทคนิค Ajax จะเป็นการทำงานแบบช่วงเวลาไม่ตรงกันหรือการติดต่อสื่อสารแบบไม่ต่อเนื่อง (Asynchronous) โดยเซิร์ฟเวอร์จะทำการส่งผลลัพธ์ไปยังเว็บเพจให้ผู้ใช้งานทันที โดยไม่ต้องรอให้ประมวลผลเสร็จก่อน เว็บเพจของผู้ใช้งานจะทำการดึงข้อมูลในส่วนต่างๆ ภายหลัง [10]



ภาพที่ 2.4 เปรียบเทียบหลักการทำงานระหว่างเว็บแอปพลิเคชันทั่วไปและเว็บแอปพลิเคชันที่ใช้ Ajax

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.5 ภาษา SQL

SQL (Standard relational database Query Language) ถือเป็นภาษามาตรฐานที่ใช้สำหรับระบบฐานข้อมูล เป็นส่วนหนึ่งของระบบฐานข้อมูลแบบรีเลชันเนล ที่ได้รับความนิยมมาก เพราะเป็นภาษาที่เข้าใจง่าย ภาษา SQL แบ่งเป็น 3 ประเภท คือ

- ภาษานิยามข้อมูล (Data Definition Language : DDL) เป็นคำสั่งที่ใช้ในการสร้างฐานข้อมูล กำหนดโครงสร้างข้อมูลว่ามี Attribute ใด ชนิดของข้อมูล รวมทั้งการเปลี่ยนแปลงตาราง และการสร้างดัชนี เช่น คำสั่ง CREATE, DROP, ALTER เป็นต้น
- ภาษาจัดการข้อมูล (Data Manipulation Language : DML) เป็นคำสั่งที่ใช้ในการเรียกใช้ เพิ่ม ลบ และเปลี่ยนแปลงข้อมูลในตาราง เช่น คำสั่ง SELECT, INSERT, UPDATE, DELETE เป็นต้น
- ภาษาควบคุมข้อมูล (Data Control Language : DCL) เป็นคำสั่งที่ใช้ในการกำหนดสิทธิ์การอนุญาต หรือ ยกเลิก การเข้าถึงฐานข้อมูล เพื่อป้องกันความปลอดภัยของฐานข้อมูล เช่น คำสั่ง GRANT, REVOKE เป็นต้น

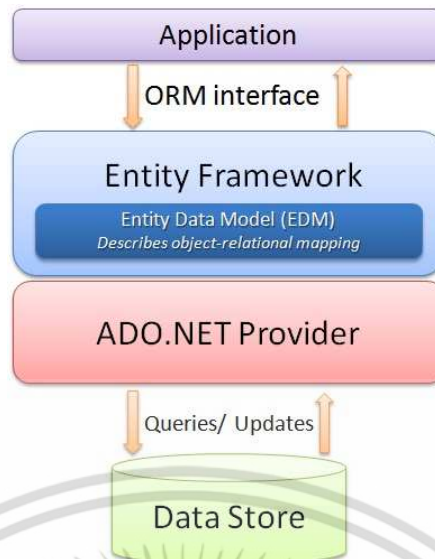
```
SELECT * FROM Customers
WHERE Country='Mexico';
```

รูปที่ 2.5 ตัวอย่างการเขียน SQL เบื้องต้น

จากรูป 2.5 อธิบายการเขียนภาษา SQL ได้คือ ทำการดึงข้อมูลจากฐานข้อมูล เฉพาะคอลัมน์ Country ที่เป็นประเทศเม็กซิโกเท่านั้น [11]

- Entity Framework

Entity Framework คือ Library ที่ถูกสร้างมาเพื่อลดปัญหาของมาตรฐานในการติดต่อกับฐานข้อมูลบน .Net Framework ซึ่งในการเขียนโปรแกรม การออกแบบคลาสเพื่อติดต่อกับฐานข้อมูลมีความสำคัญอย่างมาก ยิ่งโปรแกรมมีความซับซ้อนมากจะทำให้จัดการได้ยาก เนื่องจากนักพัฒนาแต่ละคนอาจจะเรียกใช้งานคลาสแตกต่างกัน มีการเปิด-ปิดฐานข้อมูล เพื่อเขียน SQL Statement ให้ตรงความต้องการ จะเกิดการดึงข้อมูลออกมาใช้งานซ้ำๆ ไม่มีมาตรฐานว่าต้องเรียกใช้ตอนไหน และเรียกใช้อย่างไร Entity Framework จึงถูกออกแบบมาเพื่อแก้ปัญหานี้ โดยจะทำหน้าที่เป็น Entity Data Model อยู่ในชั้น Data Layer เป็นรูปแบบ Object/Relational Mapping (O/RM)



รูปที่ 2.6 Entity Framework ทำหน้าที่เป็นตัวกลางในการติดต่อระหว่าง Application กับ Database

Entity Framework จะสร้าง Layer ทำหน้าที่เป็น Database Model ขึ้นมาเป็นคลาสในโปรเจกต์ โดยจะ Mapping คลาสที่สร้างขึ้นใหม่กับ Table, View และ Procedure จากฐานข้อมูลมาไว้บนโปรเจกต์ ซึ่งสามารถเรียกใช้ผ่านคลาสที่อยู่ในโปรเจกต์ได้ โดยไม่ต้องเขียนคำสั่ง SQL Statement เช่น INSERT, UPDATE หรือ DELETE แล้วส่งไป Execute ที่ Table อีก ทำให้สามารถบรรทัดของการเขียนติดต่อกับฐานข้อมูลลงได้อย่างมาก และลดความสับสนในการแก้ไข SQL statement ลง เพราะ statement ที่ยาวเกินไป อาจทำให้เกิดความผิดพลาดในการเขียนได้



รูปที่ 2.7 Entity Framework จำลอง Model Entities

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Model Entities คือ Object ใหญ่สุดของ Entity Framework ซึ่งเปรียบเสมือนคลาสหลักที่รวบรวม Entity Set ที่ถูกตั้งชื่อให้เหมือนกับชื่อของ Table และ View มาไว้รวมกัน สามารถเพิ่มคุณสมบัติต่างๆ ของ Entity Set ได้ เช่น Relation, Data Type หรืออื่นๆ ภายในหนึ่งโปรเจกต์สามารถมีได้หลาย Model Entities และสามารถเรียกใช้งานได้มากกว่าหนึ่ง Entities ให้ทำงานพร้อมๆ กันได้ [12]

ตัวอย่างการเปรียบเทียบการเขียน SQL Statement แบบปกติกับการเขียนแบบ Entity Framework

- การเขียนแบบปกติ
String sql = SELECT * FROM CUSTOMER ORDER BY CUSTOMER_ID ASC;
- การเขียนแบบ Entity Framework
var sql = db.CUSTOMER.OrderBy(x => (x.CUSTOMER_ID)).ToList();

2.4.6 ภาษา Gherkin

ภาษา Gherkin คือ ภาษาหนึ่งที่ใช้สำหรับการทดสอบซอฟต์แวร์ ซึ่งอธิบายพฤติกรรมของซอฟต์แวร์ หรือ Business Logic โดยเป็นภาษามนุษย์ในรูปแบบประโยคต่างๆ โดยจะบ่งบอกถึงการนำไปพัฒนาในการเขียนโปรแกรม โดย Gherkin มีลักษณะคือ ซึ่งประกอบด้วยคำอธิบายของเมธอดหรือฟังก์ชันที่ทำการทดสอบ ไฟล์ข้อมูลของ Gherkin จะมีชนิดเป็น .feature ภาษา Gherkin มีลักษณะเหมือนกับภาษา Python และ YAML โดย Gherkin เป็นภาษา line-oriented language ใช้ย่อหน้าสำหรับการกำหนดโครงสร้าง และในแต่ละบรรทัดจะจบด้วยคำสั่งและแต่ละบรรทัดจะเริ่มต้นด้วยคำหลักการคอมเมนต์ในแต่ละบรรทัดในไฟล์ใช้เครื่องหมาย # ตัวแบ่งแยกคำจะแบ่งเป็นลักษณะ (Feature) ฉาก (Scenario) และขั้นตอน (Step) เมื่อทำการรันไฟล์ .feature จะทำการประมวลข้อมูลหลังจากคำหลักของแต่ละขั้นตอน

```
Feature: Password management
Scenario: Forgot password
Given a user with email "cukes@cukes.info" exists
When I ask for a password reset
Then an email with a password reset link should be sent
```

รูปที่ 2.8 แสดงตัวอย่างการเขียนภาษา Gherkin

ตัวอย่างการเขียนภาษา Gherkin ดังรูปที่ 2.8 เป็นการเขียนสำหรับการจัดการพาสเวิร์ด ซึ่งเป็นกรณีของการลืมพาสเวิร์ด โดยคำที่เป็นคำหลัก เช่น Feature, Scenario, Given, When, Then ไฟล์ดังกล่าวถูกเก็บในชื่อ password_management.feature [13]

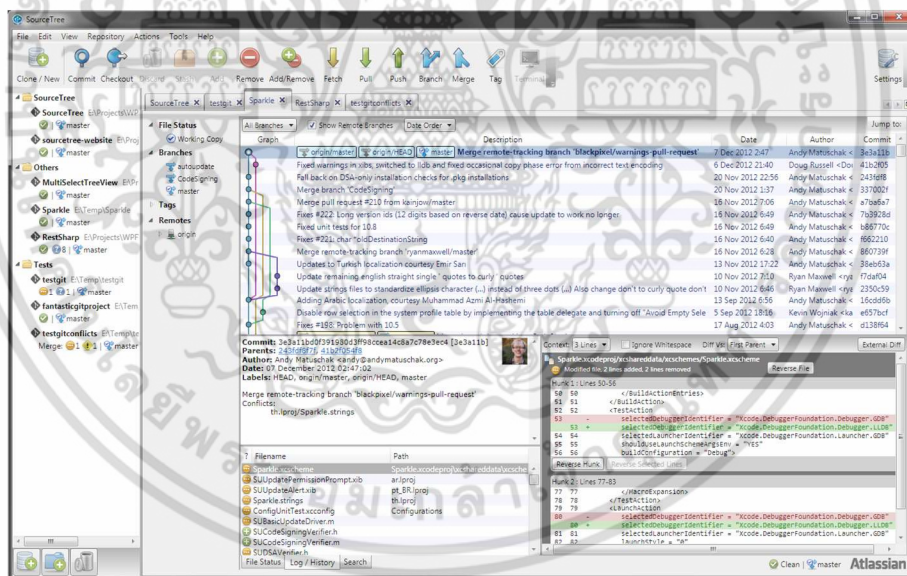
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 GIT

Git คือ Version Control ประเภทหนึ่ง ซึ่งเป็นระบบที่มีหน้าที่ในการจัดเก็บการเปลี่ยนแปลงของไฟล์ในโปรเจกต์ มีการสำรองโค้ดที่เขียนไว้ โดยสามารถที่จะเรียกดู หรือย้อนกลับไปดู Version ต่างๆ ของโปรเจกต์ที่ใดเวลาใดก็ได้ หรือแม้แต่ดูว่าไฟล์นั้นๆใครเป็นคนเพิ่มหรือแก้ไข หรือจะดูว่าไฟล์นั้นๆ ถูกเขียนโดยใครบ้างก็ได้ ทำให้ Git เป็นเครื่องมือที่เหมาะสมอย่างยิ่งสำหรับนักพัฒนา จะช่วยให้การพัฒนาแบบเป็นทีมมีประสิทธิภาพในการพัฒนามากขึ้น โดย Git มีคุณสมบัติครบถ้วนดังนี้

- สามารถ Backup และ Restore ไฟล์ได้
- Synchronization ไฟล์ทำให้ไฟล์นั้นเป็นเวอร์ชันใหม่เสมอไม่ว่าจะมีผู้ใช้งานไฟล์นั้นกี่คนก็ตาม
- สามารถ Track back history ของแต่ละไฟล์ได้
- ใน History นั้นสามารถบอกรายละเอียดได้มากกว่าการจัดเก็บแบบธรรมดา คือสามารถบอกได้ว่าไฟล์นี้ใครเป็นเจ้าของ และใครบ้างที่แก้ไขไฟล์นี้
- ควบคุมจัดการได้ง่ายเมื่อต้องการควบคุมโค้ดที่พัฒนาโดยโปรแกรมเมอร์หลายคนในทีม

ปัจจุบัน Git เป็นสิ่งที่ขาดไม่ได้ในการพัฒนาซอฟต์แวร์ โดยมีบริการให้เลือกใช้มากมาย โดยนักพัฒนาส่วนใหญ่นิยมใช้บริการของ Github และ Bitbucket ในการฝากไฟล์โปรเจกต์ [14]

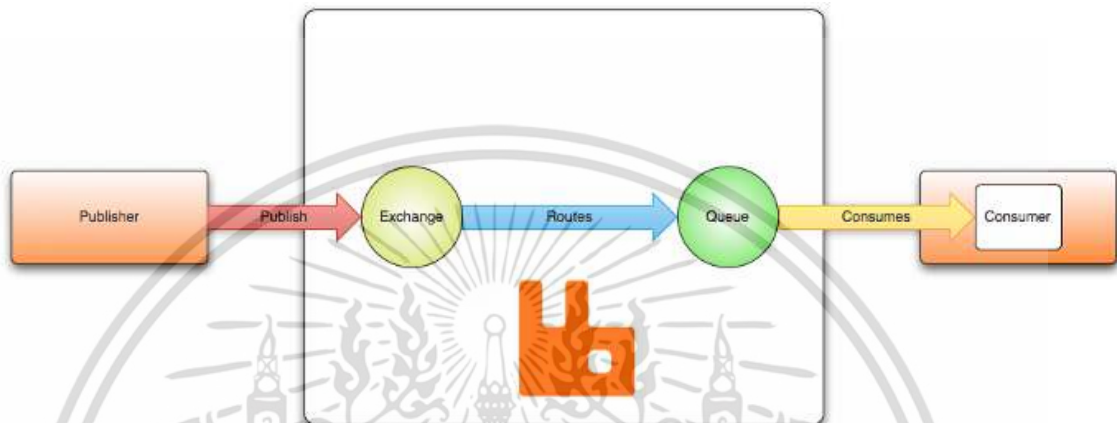


รูปที่ 2.9 โปรแกรม SourceTree ที่ใช้สำหรับจัดการ Git

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 Rabbit MQ

Rabbit MQ คือ แอปพลิเคชันที่พัฒนาด้วยภาษา Erlang เป็นตัวกลางในการสื่อสารหรือการกระจายข้อมูล เป็นการติดต่อสื่อสารกันได้ในระหว่างโปรเซส (Inter Process Communication: IPC) แบบ Message Queue ซึ่งสามารถช่วยให้แอปพลิเคชันต่างๆ สามารถสื่อสารกันได้โดย ผู้ส่งและผู้รับไม่จำเป็นต้องเขียนด้วยภาษาเดียวกัน ข้อมูลที่ส่งมีความปลอดภัยสูง Rabbit MQ รองรับการสื่อสารได้หลายโปรโตคอล เช่น AMQP 0-9-1, AMQP 1.0, MQTT, STOMP เป็นต้น



รูปที่ 2.10 ตัวอย่างการทำงานของ Rabbit MQ

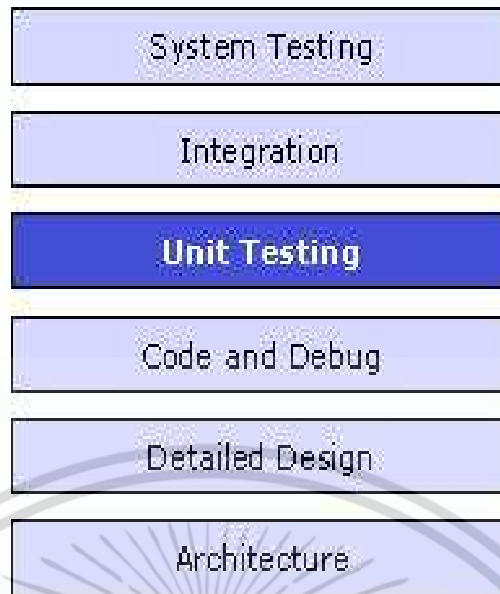
จากรูปที่ 2.10 สามารถอธิบายได้ ดังนี้ Publisher คือ ผู้ที่มีข้อมูลที่ต้องการจะส่งให้กับผู้รับ โดยการส่งข้อมูลจะต้องผ่านตัวกลาง คือ RabbitMQ ซึ่ง RabbitMQ จะเปิดเส้นทาง Routes คอยรับและส่งข้อมูลผ่าน Routes ตัวนั้น ถ้าหากในกรณีที่มีผู้ส่งมีหลายคนและมีจำนวนมาก การส่งข้อมูลเข้ามาพร้อมกันใน 1 วินาที จะมีปริมาณมาก ทำให้ผู้รับต้องทำงานอย่างหนักในการรับข้อมูล RabbitMQ จึงถูกสร้างมาเพื่อช่วยจัดการปัญหานี้ ทำหน้าที่เป็นตัวกลางคอยรับข้อมูล จัดลำดับข้อมูลตามเวลาที่เริ่มส่ง และจะเริ่มส่งให้ผู้รับทีละ 1 ข้อมูลไปเรื่อยๆจนหมด ทำให้ผู้รับสามารถทำงานต่อไปได้ [15]

2.7 Unit Test

Unit Test คือ การทดสอบระดับหน่วย เป็นการทดสอบรูปแบบหนึ่งที่ถูกทำโดยนักพัฒนา โดยใช้กับเมธอดหรือฟังก์ชันการทดสอบ คุณสมบัติ คลาส และองค์ประกอบ การทดสอบระดับหน่วย ทำเพื่อตรวจสอบคุณภาพของโค้ดที่เขียน ทำงานได้ถูกต้องหรือไม่ ส่วนใหญ่มักนำการทดสอบระดับหน่วยเข้าไปใช้กับการพัฒนาด้วย เพื่อเป็นการป้องกันข้อผิดพลาดที่อาจจะเลยไป

การทดสอบระดับหน่วยใช้กับการทดสอบการทำงานที่เป็นส่วนเล็กๆ โดยนักพัฒนาสามารถปรับปรุงแก้ไขโค้ดโดยไม่ต้องกังวลว่าถ้าหากมีการแก้ไขโค้ดบางส่วนขึ้นมาและจะกระทบกับการทำงานส่วนอื่นๆของโปรแกรม การทดสอบระดับหน่วยนี้จะทำการทดสอบโปรแกรมทีละโมดูล เพื่อหาข้อผิดพลาดที่จะเกิดขึ้นภายในการทำงานของแต่ละโมดูล จะทำก็ต่อเมื่อเขียนโค้ดเสร็จแล้ว ซึ่งหากรวมเข้ากับการพัฒนาแอปพลิเคชัน การทดสอบระดับหน่วยจะแสดงดังรูปที่ 2.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 แสดงการทดสอบระดับหน่วยในการพัฒนาแอปพลิเคชัน

จากรูปเห็นได้ว่าหลังจากเขียนโค้ดและหาข้อผิดพลาด (Debug) เรียบร้อยแล้ว จะต้องทำการทดสอบระดับย่อย เพื่อดูว่าแอปพลิเคชันแต่ละส่วนทำงานได้ถูกต้องหรือไม่ แล้วจะทำการรวม ก่อน จะทำการทดสอบแบบโดยรวมในขั้นตอนสุดท้าย [16]

บทที่ 3

วิธีการดำเนินงานวิจัย

3.1 ขั้นตอนการศึกษาและรวบรวมข้อมูลของระบบ

1) ศึกษาปัญหาและความต้องการของระบบ

สอบถามวิธีการใช้งานระบบจากผู้ใช้และทดลองใช้งานด้วยตนเอง เพื่อให้เข้าใจการทำงานของระบบ สอบถามความต้องการของผู้ใช้ที่ต้องการให้พัฒนา และศึกษาเกี่ยวกับเทคโนโลยีต่างๆที่ใช้ในการพัฒนาระบบ เช่น ภาษาที่ใช้พัฒนา ฐานข้อมูลที่ใช้ เป็นต้น

2) ศึกษาขั้นตอนการพัฒนาของระบบ

ศึกษาส่วนประกอบของระบบและขั้นตอนการพัฒนาของระบบ โดยรวบรวมข้อมูลเกี่ยวกับการพัฒนา Web API แอปพลิเคชัน ศึกษาขั้นตอนการทำงานต่างๆ ของระบบจนมีความเข้าใจ

3) ศึกษาข้อมูลและเครื่องมือที่ใช้ในการพัฒนาระบบ

ศึกษาข้อมูลและเครื่องมือที่ใช้ในการพัฒนาระบบ โดยจะใช้โปรแกรม Visual Studio 2013 ในการพัฒนาและใช้ภาษา C# การติดต่อกับฐานข้อมูลจะใช้ Entity Framework ที่ชื่อว่า LINQ ส่วนของ User Interface ถูกพัฒนาด้วยภาษา HTML และใช้ Javascript Library jQuery ในส่วนของฐานข้อมูลจะใช้ Oracle SQL

3.2 ขั้นตอนการวิเคราะห์ระบบ

จากการศึกษาระบบเดสก์ทอป ดิพลอยเมนต์พอร์ทเดิลเดิมนั้น การจะทำให้รองรับผลิตภัณฑ์ได้มากขึ้น ไม่สามารถทำได้ทันที โดยพบปัญหาดังนี้

1) ระบบเดิมนั้นสร้างขึ้นมาเพื่อจัดการกับผลิตภัณฑ์ Eikon โดยเฉพาะ ทำให้ในฐานข้อมูลจะเก็บข้อมูลเครื่อง Clients, ข้อมูล Package, ข้อมูล Patch และตารางอื่นๆ จะมีเฉพาะผลิตภัณฑ์ Eikon เพียงอย่างเดียว ทำให้การดึงข้อมูลไม่จำเป็นต้องคำนึงถึงประเภทของผลิตภัณฑ์

ดังนั้น การเพิ่มข้อมูลของผลิตภัณฑ์อื่นลงในฐานข้อมูล จะเกิดปัญหาในเวลาดึงข้อมูล ข้อมูลที่ถูกดึงออกมาไม่สามารถบอกได้ว่าเป็นของผลิตภัณฑ์ใด ส่งผลให้อาจเกิดข้อผิดพลาดในการให้บริการอัปเดตได้

2) การเข้าใช้งานระบบผ่านหน้าเว็บแอปพลิเคชันแบบเดิม จะทำการตรวจสอบการได้รับอนุญาต (Permission) ว่ามีสิทธิ์การใช้งานแอปพลิเคชันหรือไม่ โดยสิทธิ์ขั้นต่ำสุด คือ ประเภท Read Only ที่สามารถดูข้อมูลได้อย่างเดียว แต่เนื่องจากในระบบใหม่สิทธิ์การเข้าจัดการของแต่ละผลิตภัณฑ์มีความเป็นอิสระต่อกัน ทำให้การเข้าใช้งานระบบใหม่ จำเป็นต้องคำนึงถึงผลิตภัณฑ์ด้วย โดยผู้ใช้จะไม่มีสิทธิ์ดำเนินการใดๆกับผลิตภัณฑ์ที่ไม่ได้รับอนุญาต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ในหน้าเว็บแอปพลิเคชันส่วนของ Back Office บางหัวข้อจะมีปัญหา เพราะไม่มีสิ่งที่จะสามารถแยกข้อมูลของแต่ละผลิตภัณฑ์ออกจากกันได้ ทำให้ตารางข้อมูลที่แสดง ไม่ตรงตามสิทธิ์การใช้งานของพนักงาน

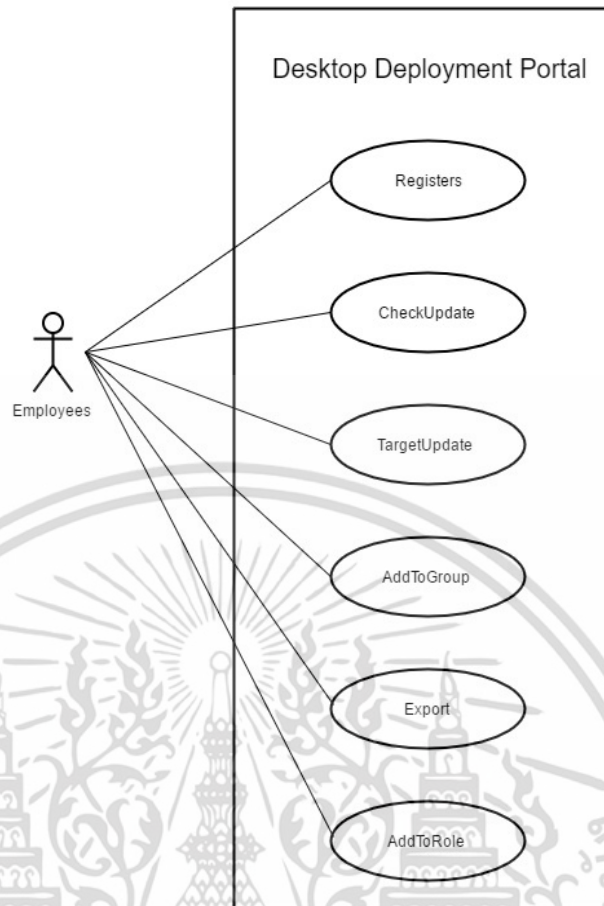
3.3 ขั้นตอนการออกแบบระบบ

3.3.1 การออกแบบการทำงานของระบบ

1) แผนภาพแสดงการทำงานของผู้ใช้ระบบ (Use Case Diagram)

โปรแกรม Desktop Deployment มีฟังก์ชันการใช้งานหลัก แบ่งได้ 6 ฟังก์ชัน ดังนี้

- ฟังก์ชัน Register คือ ฟังก์ชันที่ทำการบันทึกข้อมูลของผู้ใช้งานที่ติดตั้งผลิตภัณฑ์แล้ว โดยจะนำข้อมูลเหล่านั้นมาใช้ในการตัดสินใจแก่พนักงาน ในการอัปเดตเวอร์ชันและส่วนเสริม ข้อมูลที่เก็บเข้ามา ประกอบไปด้วย InstallationGuid, MachineUUID, MacAddress, Version, LoginHistory เป็นต้น
- ฟังก์ชัน CheckUpdate คือ ฟังก์ชันที่ผู้ใช้งานทำการตรวจสอบผลิตภัณฑ์ที่ติดตั้งว่าเป็นรุ่นล่าสุดหรือไม่ หรือมีส่วนเสริมใดที่สามารถดาวน์โหลดได้ ซึ่งการปล่อยแพ็คเกจจะแตกต่างออกไปตามประเทศหรือกฎหมายของประเทศนั้นๆ
- ฟังก์ชัน TargetUpdate คือ ฟังก์ชันที่พนักงานทำการกำหนดเวอร์ชันและส่วนเสริมที่จะปล่อยให้ลูกค้า โดยสามารถกำหนดแบบรายบุคคล หรือปล่อยให้แก่ลูกค้าทั้งหมดก็ได้ เมื่อเวอร์ชันที่ปล่อยไปเกิดปัญหาขึ้น สามารถดาวน์โหลดได้ด้วยการกำหนดเวอร์ชันที่ต่ำกว่า
- ฟังก์ชัน AddToGroup คือ ฟังก์ชันที่สามารถจัดกลุ่มลูกค้าให้อยู่ในกลุ่มที่ต้องการได้ โดยกลุ่มนั้น สามารถเลือกได้ว่าจะดำเนินการใด เช่น กลุ่มผู้ทดลองใช้, กลุ่มที่ต้องการส่วนเสริมแบบพิเศษ เป็นต้น
- ฟังก์ชัน Export คือ ฟังก์ชันที่สามารถดาวน์โหลดข้อมูลต่างๆ เช่น ข้อมูลเครื่องลูกค้า, ข้อมูล Package, ข้อมูล Patch, ข้อมูลกลุ่ม เป็นต้น โดยจะดาวน์โหลดเป็นไฟล์นามสกุล csv
- ฟังก์ชัน AddToRole คือ ฟังก์ชันที่ผู้ใช้งานที่เป็นผู้ดูแลระบบ (Super Admin) กำหนดให้ผู้ใช้รายอื่นมีสิทธิ์การใช้งานโปรแกรมแตกต่างกันไป โดยฟังก์ชันนี้มีสิทธิ์เฉพาะผู้ดูแลระบบเท่านั้น ซึ่งแต่ละโปรแกรมต้องมีผู้ที่มีสิทธิ์อย่างน้อย 1 คน

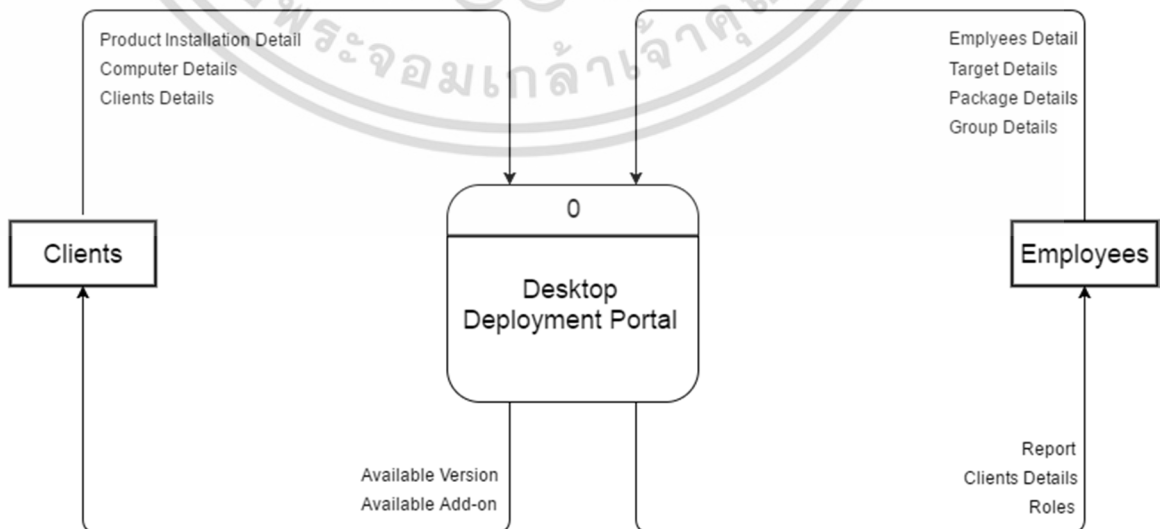


รูปที่ 3.1 Use Case Diagram ของระบบ

2) แผนผังบริบท (Context Diagram)

จากขั้นตอนการวิเคราะห์ระบบข้างต้น สามารถเขียน Context Diagram ของระบบคร่าวๆ

ได้ดังนี้



รูปที่ 3.2 Context Diagram ของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเชิงวิชาการเท่านั้น เมื่อผู้ผู้ใดให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

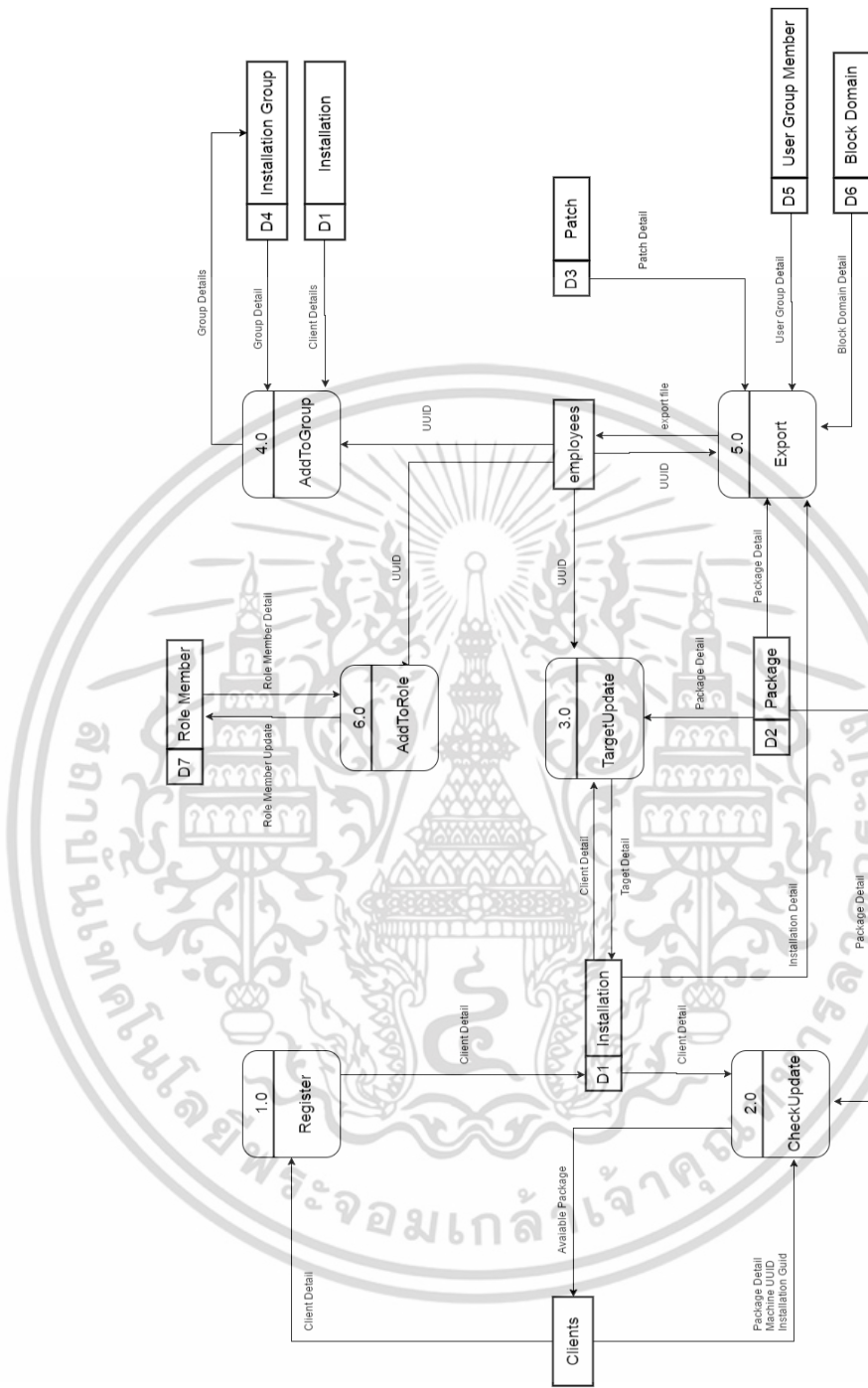
จากรูปที่ 3.2 อธิบายได้คือ ระบบเดสก์ท็อป ดีพลอยเมนท์พอทัล เป็นระบบที่คอยจัดการกับผลิตภัณฑ์ของบริษัท จะมีผู้มีส่วนร่วม 2 กลุ่ม คือ พนักงาน(Employees) และลูกค้า (Clients)

- พนักงานเมื่อจะเข้าใช้งานโปรแกรมต้องใช้รหัสประจำตัว (UUID) ในการเข้าใช้งาน โดยพนักงานนั้น สามารถจัดการกับข้อมูลของ Package, Patch, Group ต่างๆได้ และสามารถกำหนดเวอร์ชันเป้าหมาย และส่วนเสริมให้กับเครื่องลูกค้าได้
- ลูกค้าเมื่อเข้าใช้งานผลิตภัณฑ์ ผลิตภัณฑ์จะส่งข้อมูลของลูกค้า เช่น รหัสผู้ใช้งาน, รหัสการติดตั้ง, ข้อมูลเครื่อง เป็นต้น และเมื่อพนักงานทำการกำหนดเวอร์ชัน เครื่องลูกค้าจะได้รับการอัปเดตผ่านทางผลิตภัณฑ์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) แผนภาพกระแสข้อมูล (Data Flow Diagram : DFD)



รูปที่ 3.3 Data Flow Diagram ระดับ 0 ของระบบ

จากรูปที่ 3.3 เป็น Dataflow Diagram level 0 โดยแบ่งออกเป็น 6 ส่วนหลัก ได้ดังนี้

- ฟังก์ชัน Register คือ ฟังก์ชันที่ทำการบันทึกข้อมูลของลูกค้าที่ติดตั้งผลิตภัณฑ์แล้ว โดยจะนำข้อมูลเหล่านั้นมาใช้ในการตัดสินใจแก่พนักงาน ในการอัพเดทเวอร์ชันและส่วนเสริม

- ฟังก์ชัน CheckUpdate คือ ฟังก์ชันที่ลูกค้าทำการตรวจสอบผลิตภัณฑ์ที่ติดตั้งว่าเป็นรุ่นล่าสุดหรือไม่ หรือมีส่วนเสริมใดปล่อยให้ดาวน์โหลดได้บ้าง

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ฟังก์ชัน TargetUpdate คือ ฟังก์ชันที่พนักงานทำการกำหนดเวอร์ชันและส่วนเสริมที่จะปล่อยให้กับลูกค้า โดยสามารถกำหนดแบบรายบุคคล หรือปล่อยให้แก่ลูกค้าทั้งหมดก็ได้ เมื่อเวอร์ชันที่ปล่อยไป มีปัญหา สามารถดาวน์โหลดได้ด้วยการกำหนดเวอร์ชันที่ต่ำกว่า

- ฟังก์ชัน AddToGroup คือ ฟังก์ชันที่สามารถจัดกลุ่มลูกค้าให้อยู่ในกลุ่มที่ต้องการได้ โดยกลุ่มนั้น สามารถเลือกได้ว่าจะดำเนินการใด เช่น กลุ่มผู้ทดลองใช้ เป็นต้น

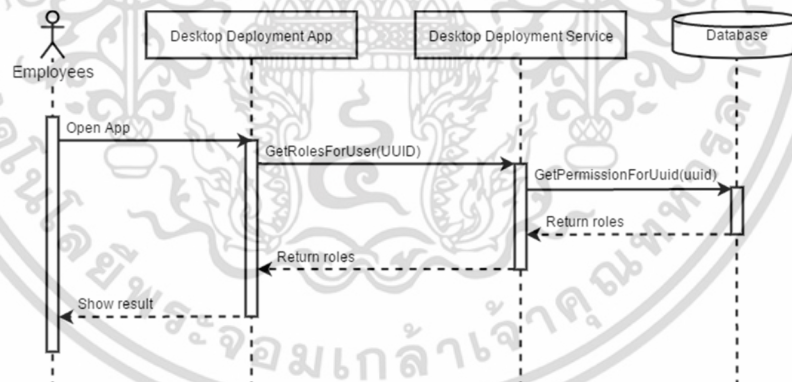
- ฟังก์ชัน Export คือ ฟังก์ชันที่สามารถดาวน์โหลดข้อมูลต่างๆ เช่น ข้อมูลเครื่องลูกค้า, ข้อมูล Package, ข้อมูล Patch, ข้อมูลกลุ่ม เป็นต้น

- ฟังก์ชัน AddToRole คือ ฟังก์ชันที่ผู้ใช้งานที่เป็นผู้ดูแลระบบ (Super Admin) กำหนดให้ผู้ใช้รายอื่นมีสิทธิ์การใช้งานต่างกันไป โดยฟังก์ชันนี้มีสิทธิ์เฉพาะผู้ดูแลระบบเท่านั้น

4) แผนภาพแสดงลำดับเหตุการณ์ (Sequence Diagram)

4.1) การเข้าใช้งานระบบของพนักงาน

เมื่อพนักงานทำการเปิดแอปพลิเคชันบนเว็บเบราว์เซอร์ขึ้นมา ตัวแอปพลิเคชันจะทำการตรวจสอบสิทธิ์การใช้งาน ด้วยการนำรหัสประจำตัวพนักงาน (UUID) ที่ได้จากการเข้าสู่ระบบเซิร์ฟเวอร์ของพนักงาน ส่งไปที่ Service และ Service จะทำการค้นหา Roles ของพนักงานในฐานข้อมูล จากนั้นจะส่งผลลัพธ์กลับไปแสดงผล ดังรูปที่ 3.4



รูปที่ 3.4 Sequence Diagram การเข้าสู่ระบบของพนักงาน

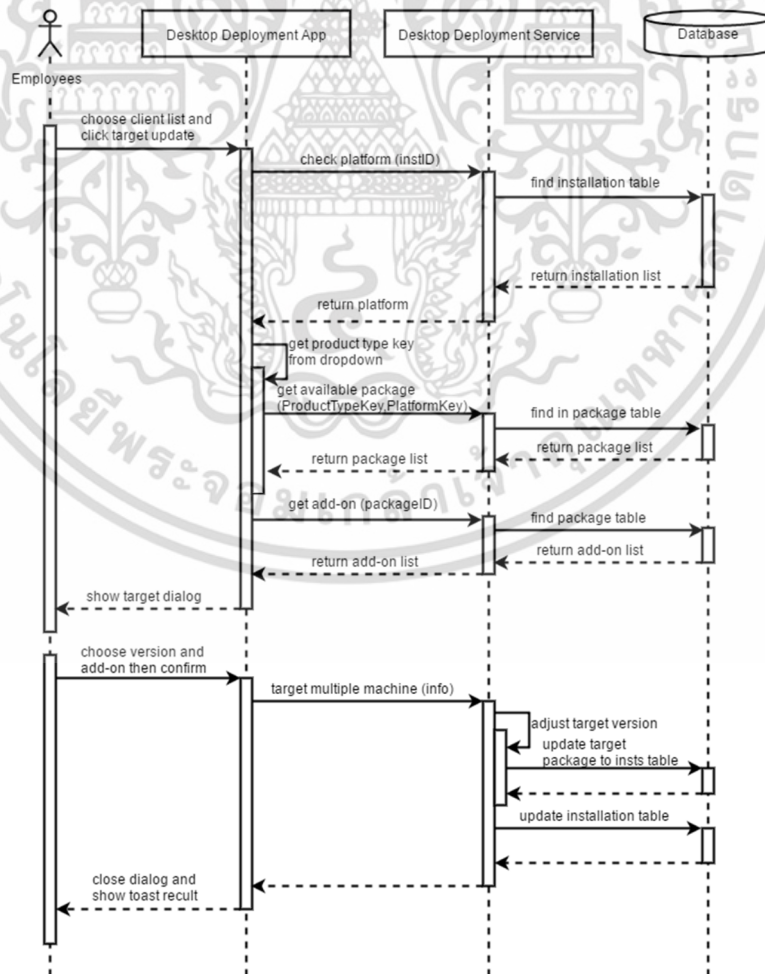
4.2) การกำหนดเวอร์ชันและส่วนเสริมให้ลูกค้า (Target Update)

ในฟังก์ชันนี้มีการทำงาน คือ พนักงานจะเลือกรายชื่อของเครื่อง Clients ที่แสดงในตารางในหน้าแรกของเว็บแอปพลิเคชัน โดยจะแสดงเป็นหน้าต่าง Dialog ขึ้นมา มีรายละเอียดคือ ระบบจะนำ Installation ID ของรายการที่เลือก ไปทำการตรวจสอบแพลตฟอร์มว่าเป็นแบบ 32 บิตหรือ 64 บิต โดยการเข้าไปค้นหาที่ตาราง Installation เมื่อได้แพลตฟอร์มเรียบร้อยแล้ว จะรับตัวแปรรหัสผลิตภัณฑ์ จากผลิตภัณฑ์ล่าสุดที่ถูกเลือกในคราวนี้ ซึ่งผู้ใช้ไม่สามารถเปลี่ยนผลิตภัณฑ์ได้ ขณะกล่อง Dialog ถูกเปิดอยู่ โดยระบบจะนำแพลตฟอร์มและรหัสผลิตภัณฑ์ที่ได้ ไปตรวจสอบเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Package ที่มีคุณสมบัติตรงกับที่เครื่อง Client สามารถติดตั้งได้ และเมื่อได้ข้อมูล Package มาแล้ว จะนำ Package นั้นไปค้นหาส่วนเสริม โดย Add-on ของแต่ละ Package จะแตกต่างกันออกไป หรือ บาง Package สามารถใช้ Add-on เดียวกันได้ จากนั้นจะนำข้อมูลที่ได้มา ไปแสดงผลที่ตัว Dialog

เมื่อผู้ใช้เลือก Package และ Add-on ที่ต้องการได้แล้ว จะทำการกดปุ่ม Target Update ตัวเว็บแอปพลิเคชันจะทำการส่งข้อมูลเข้าฟังก์ชัน TargetMultipleMachine ที่ Service โดย Service จะทำการตรวจสอบข้อมูลว่า Package แต่ละตัว มีอัปเดตใหม่หรือไม่ แล้วจึงทำการบันทึก ข้อมูลลงที่ตาราง Installation ถ้าการกำหนดรุ่นสำเร็จ หน้าต่าง Dialog จะปิดลงโดยอัตโนมัติ แต่ หากเกิดข้อผิดพลาดขึ้น จะมีข้อความแจ้งเตือนที่บริเวณด้านบนของจอภาพ

ดังนั้น เมื่อพนักงานเข้าสู่หน้าเว็บ ระบบจะทำการตรวจสอบสิทธิ์โดยนำรหัสพนักงาน ไป ค้นหาในฐานข้อมูล เมื่อพบ Roles ในผลิตภัณฑ์ใดๆ ก็สามารถเข้าใช้งานระบบ โดยมีสิทธิ์เฉพาะ ผลิตภัณฑ์นั้นๆ ได้ และในส่วนของผู้ดูแลระบบสามารถเข้าไปจัดการฐานข้อมูลผ่านเว็บแอปพลิเคชัน ได้ โดยเข้าผ่านทางปุ่ม Back Office และปุ่มนี้จะแสดงให้เห็นเฉพาะในหน้าของผลิตภัณฑ์ที่มี Roles เป็น Super admin เท่านั้น โดยผู้ที่มี Roles ที่ไม่ใช่ Read only สามารถกำหนดรุ่นและส่วนเสริม, จัดกลุ่มลูกค้า, ลบข้อมูล, และออกรายงานของผลิตภัณฑ์นั้นๆได้

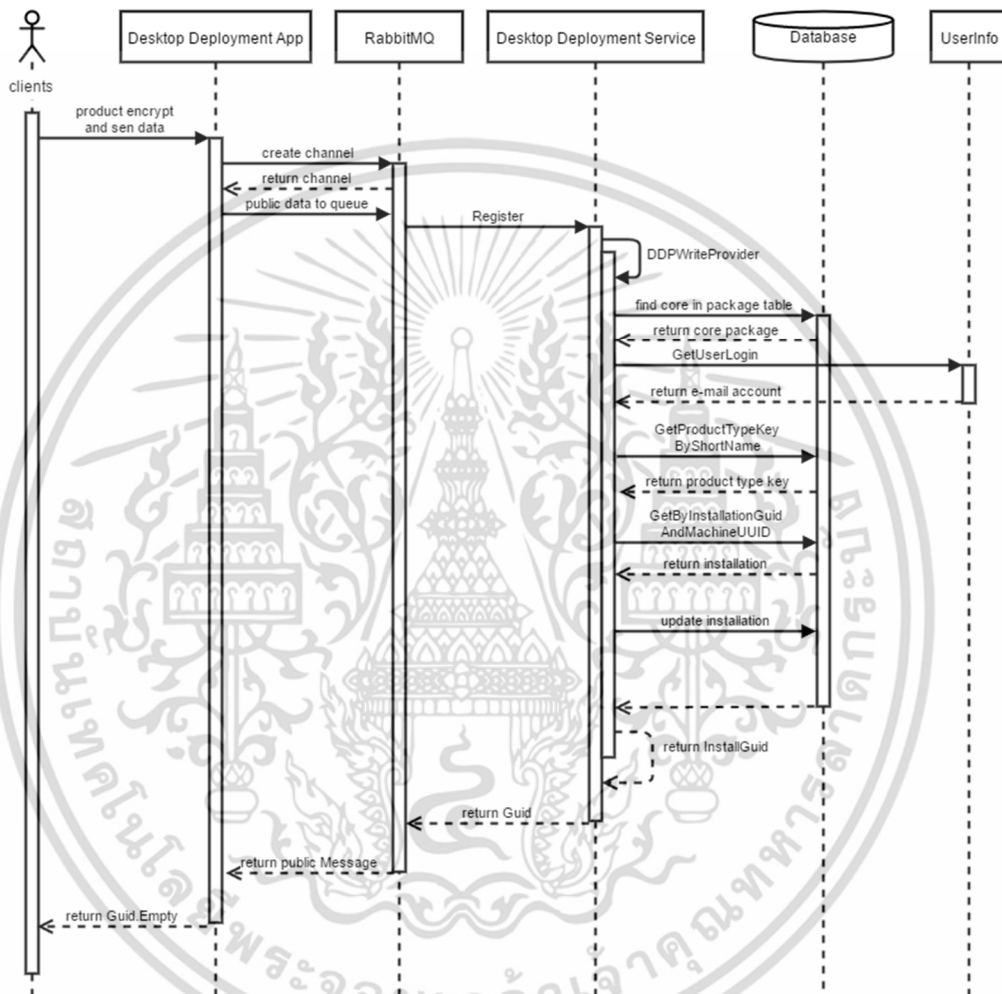


รูปที่ 3.5 Sequence diagram การกำหนดเวอร์ชันและส่วนเสริม

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ ไม่ว่าการณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3) การลงทะเบียนของผลิตภัณฑ์ (Register)

การลงทะเบียนของผลิตภัณฑ์ คือ การนำข้อมูลของเครื่องที่ติดตั้งโปรแกรมของบริษัท รอยเตอร์ส เช่น Eikon, FXT และ MCA เป็นต้น ประกอบไปด้วย เครื่องลูกข่าย (clients), App, RabbitMQ สำหรับจัดการ Message ของเครื่องลูกข่าย, Service ที่ให้บริการ, ฐานข้อมูล และ เซอร์วิสชื่อ UserInfo



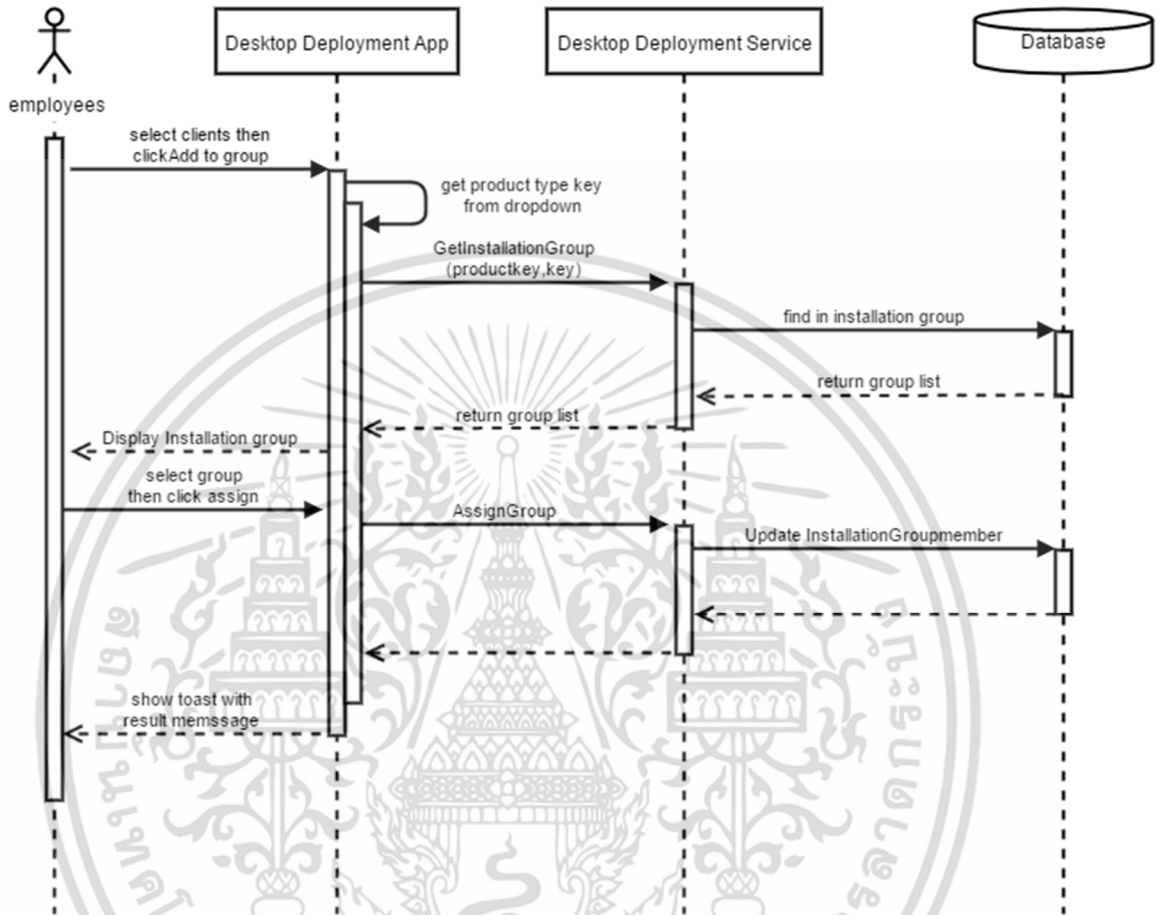
รูปที่ 3.6 Sequence Diagram ฟังก์ชัน Register

จากรูป 3.6 สามารถอธิบายได้ดังนี้ ในการบันทึกข้อมูลของเครื่อง Clients ลงที่ฐานข้อมูล เพื่อสำหรับการตรวจสอบการอัปเดต หรือติดตามเครื่องลูกค้า การ Register จะทำโดยผลิตภัณฑ์จะส่งข้อมูลที่เข้ารหัสไปที่เว็บแอปพลิเคชัน เว็บแอปพลิเคชันจะส่งต่อไปที่ RabbitMQ โดย RabbitMQ จะทำหน้าที่ในการบริหารจัดการ Register ไปยัง DDP เนื่องจาก มีผู้ใช้งานผลิตภัณฑ์อยู่มากมายทั่วโลก ทำให้ต้องมีการจัดลำดับ Queue ที่จะเข้าไปทำงานใน Service เพื่อไม่ให้ Service ทำงานหนักเกินไป เมื่อข้อมูลถูกส่งไปที่ service แล้ว จะทำการถอดรหัสข้อมูล และตรวจสอบว่าข้อมูลที่เข้ามาเป็นข้อมูลของผลิตภัณฑ์ใด ก่อนจะส่งแยกไปตามผลิตภัณฑ์นั้นๆ เพื่อบันทึกลงในตาราง Installation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4) การจัดกลุ่มลูกค้า (Add to group)

เป็นฟังก์ชันสำหรับจัดกลุ่มเครื่องที่ติดตั้งโปรแกรมอยู่ โดยกลุ่มที่สร้างและรายชื่อสมาชิกในกลุ่มจะถูกจัดเก็บในฐานข้อมูล



รูปที่ 3.7 Sequence Diagram ของฟังก์ชัน Add to group

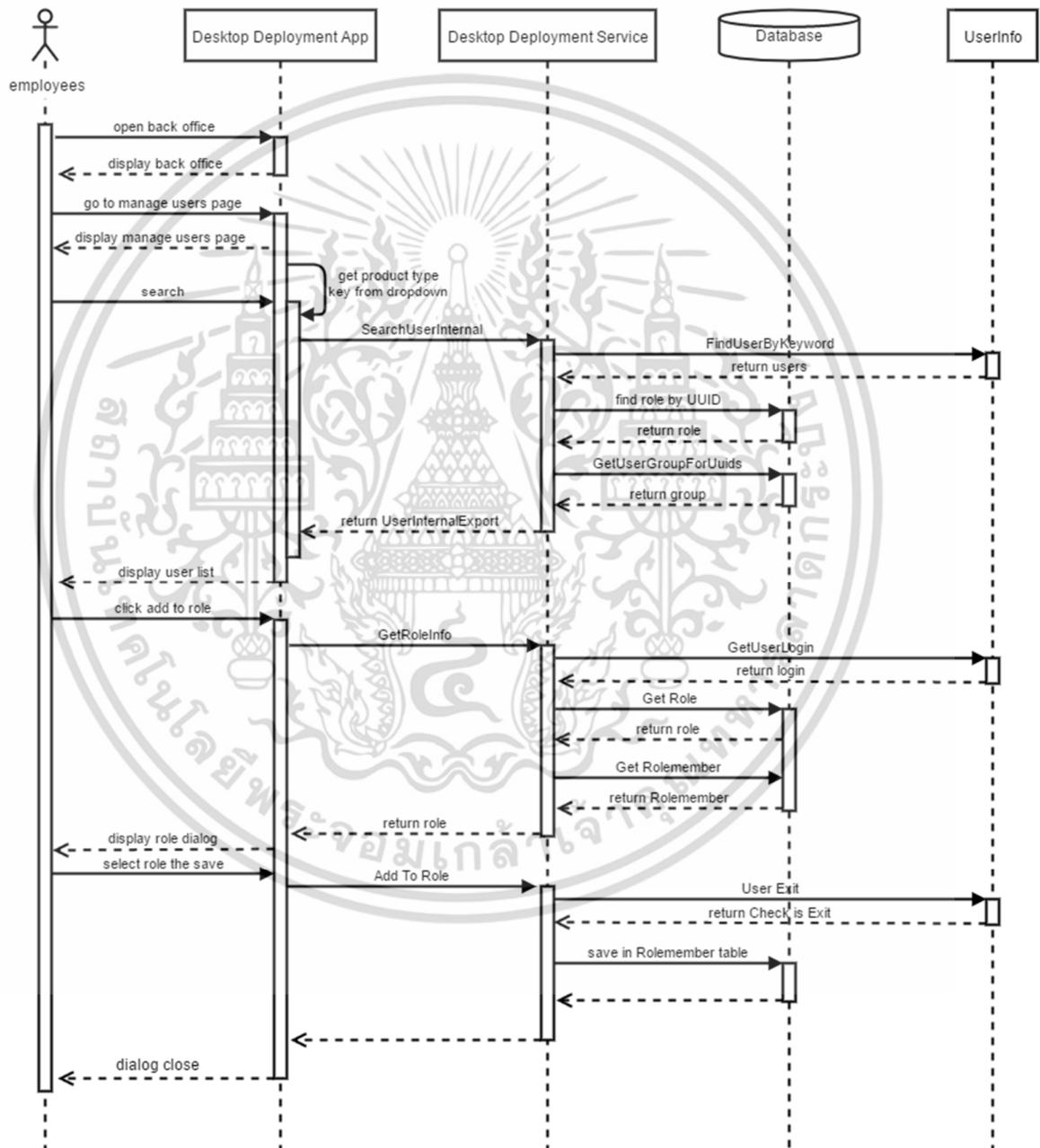
จากรูปที่ 3.7 แผนภาพลำดับเหตุการณ์อธิบายได้ว่า เมื่อพนักงานทำการเลือกเครื่อง Client ได้เรียบร้อยแล้ว แล้วจึงทำการกดปุ่ม Add to group หลังจากกดปุ่มแล้ว ตัวเว็บแอปพลิเคชันจะเริ่มดึงข้อมูลเพื่อสร้างหน้าต่าง โดยจะเก็บค่าของผลิตภัณฑ์ในขณะนั้นจากตารางก่อนจะส่งไปคำนวณที่ Service ในฟังก์ชัน GetInstallationGroup ฟังก์ชันนี้จะนำรหัสผลิตภัณฑ์ที่ส่งมาไปค้นหาในฐานข้อมูล ก่อนจะนำกลุ่มที่ได้ ส่งกลับมาให้เว็บแอปพลิเคชันแสดงผลเป็นหน้าต่างให้กับพนักงานเลือก

เมื่อพนักงานเลือกและกด Assign แล้ว ตัวเว็บแอปพลิเคชัน จะทำการส่งข้อมูลรายชื่อ Clients, กลุ่มที่พนักงานเลือกและ รหัสประจำตัวพนักงาน เพื่อไปจัดเก็บลงในตาราง Installation โดยมีการตั้ง Foreign Key สองตัว คือ InstKey กับ InstGroupKey ออกมาสร้างตารางใหม่ชื่อว่า InstallationGroupMember เมื่อทำการบันทึกเสร็จแล้วจะแสดง Toast ขึ้นแจ้งเตือนผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5) การจัดการ Roles ของผู้ดูแลระบบ (Add to roles)

ฟังก์ชัน AddToRoles จะดำเนินการได้เฉพาะผู้มี Roles เป็น Super Admin เท่านั้น โดยพนักงานจะเข้าใช้งานแอปพลิเคชัน และเข้าสู่หน้า Back Office และไปที่ Manage Users จึงทำการค้นหาชื่อของผู้ที่ต้องการจะเพิ่ม Roles ตัว Service จะนำชื่อที่ค้นหา ไปค้นหาที่ UserInfo ซึ่งเป็น Service หนึ่งที่ทำให้บริการข้อมูลของผู้ใช้ โดยเมื่อ UserInfo คืนค่ารายชื่อที่หาพบ จะถูกส่งไปที่ตัวเว็บแอปพลิเคชัน เพื่อให้พนักงานทำการเพิ่ม Roles ได้ จะเก็บข้อมูลที่ตาราง RoleMember

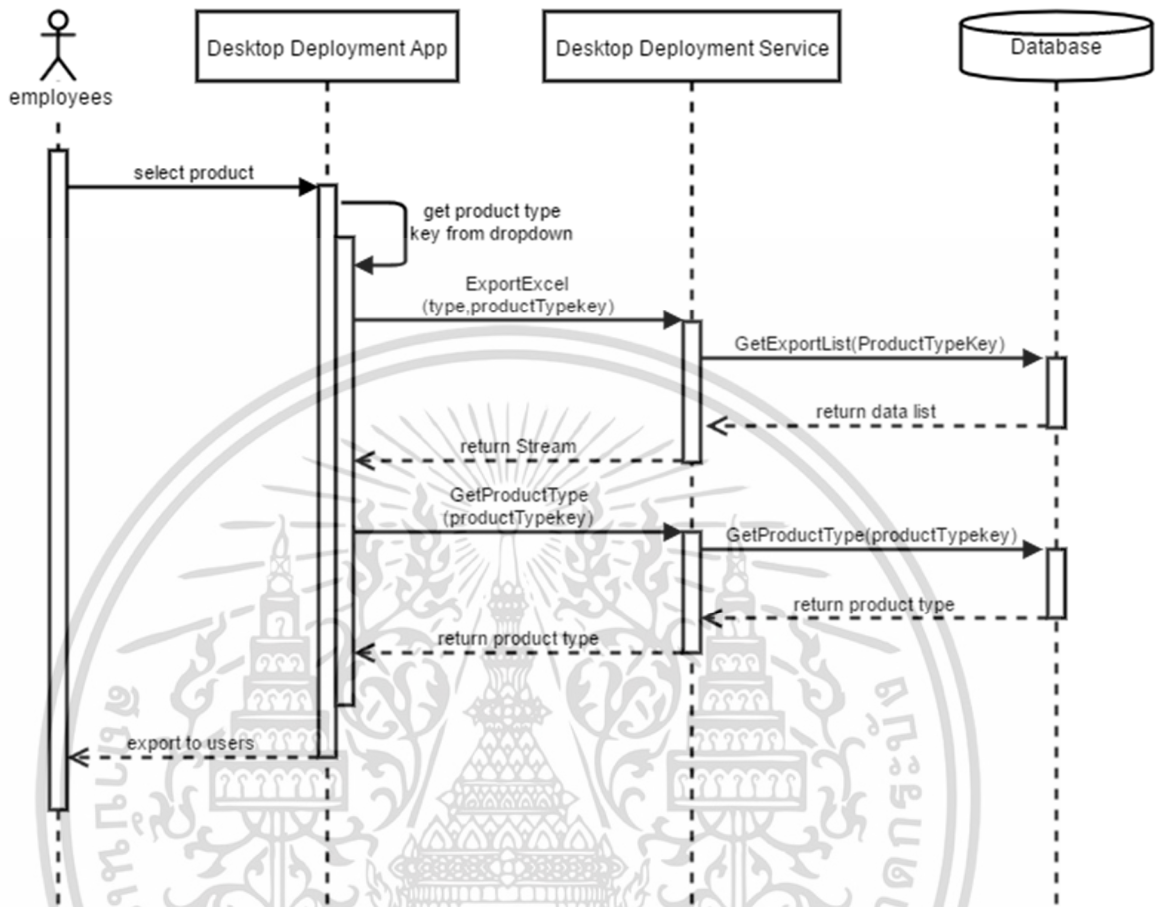


รูปที่ 3.8 Sequence Diagram ฟังก์ชัน AddToRoles

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6) การออกรายงานในส่วนต่างๆ ของระบบ Export Excel

การออกรายงาน เป็นส่วนที่ผู้ใช้สามารถดาวน์โหลดข้อมูลต่างๆเป็นไฟล์นามสกุล csv

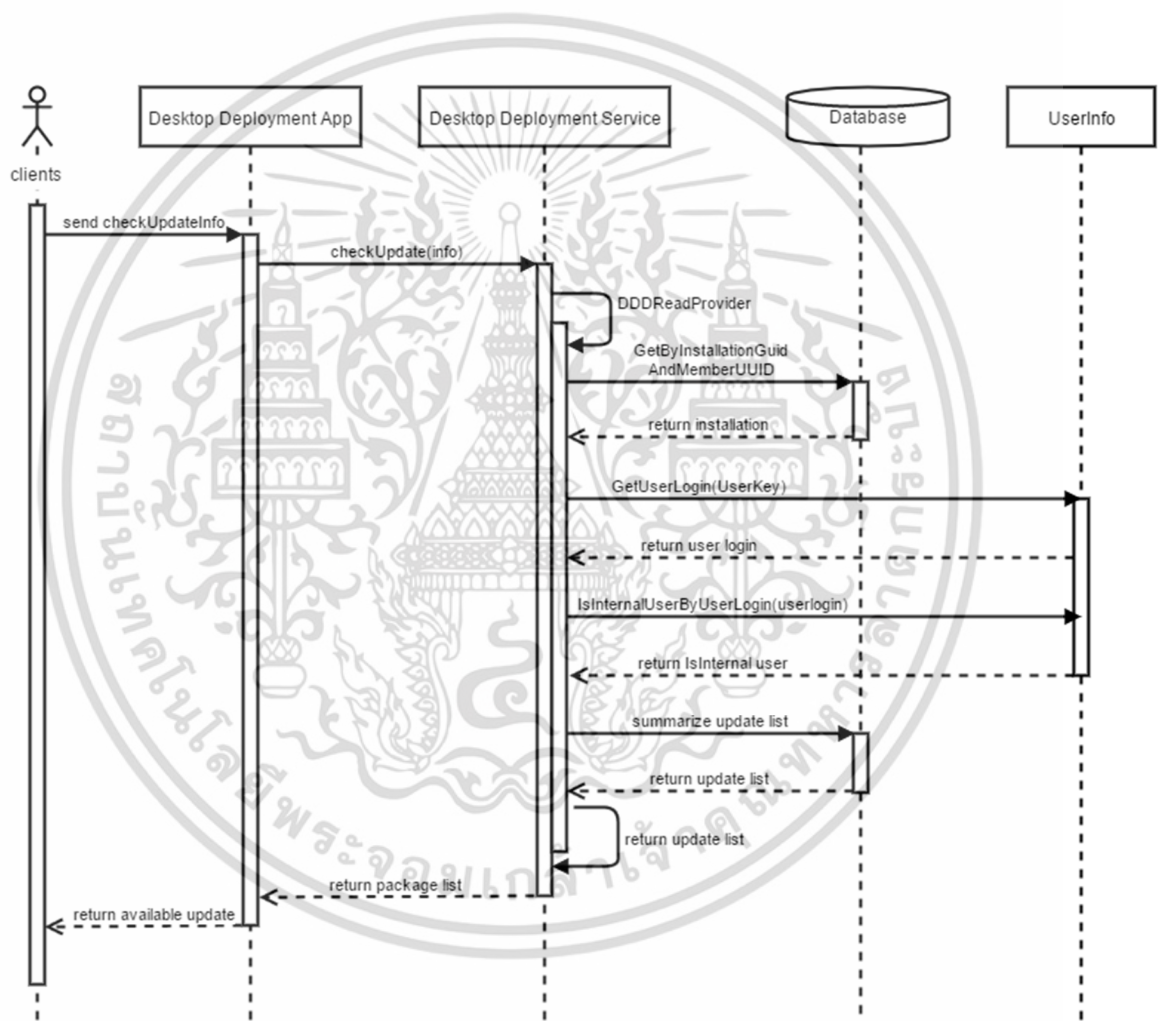


รูปที่ 3.9 Sequence Diagram ฟังก์ชัน Export

ฟังก์ชัน Export มีขั้นตอนการดำเนินงานคือ พนักงานที่มี Roles สูงกว่า Read only จะดำเนินการออกรายงาน โดยการออกรายงานนั้น สามารถออกได้หลายตาราง เช่น ตาราง installation ในหน้าแรก, ตาราง Package, ตาราง Patch เป็นต้น โดยผู้ใช้งานจะเลือกผลิตภัณฑ์ที่ตรงปกตามก่อนกดปุ่ม Export ตัวเว็บแอปพลิเคชันจะเรียกใช้ฟังก์ชัน Export Excel ใน Service โดยจะตรวจสอบว่าประเภทข้อมูลที่พนักงานต้องการนั้นเป็นประเภทใด จึงทำการส่งข้อมูลมาเป็น Stream เพื่อเขียนลงไฟล์ และตั้งชื่อไฟล์ตามข้อมูลที่พนักงานใช้งาน

4.7) การตรวจสอบการอัปเดต Check Update

ฟังก์ชัน Check Update มีหลักการทำงานดังนี้ ลูกค้านักใช้งานผลิตภัณฑ์ได้ทำการตรวจสอบการอัปเดต โดยผลิตภัณฑ์จะติดต่อมาที่ตัวเว็บแอปพลิเคชัน โดยส่งข้อมูลของผลิตภัณฑ์ที่ติดตั้งมาด้วย จากนั้นเว็บแอปพลิเคชันจะเรียกฟังก์ชัน CheckUpdate ที่ตัว Service โดยในฟังก์ชัน CheckUpdate นั้นจะทำการตรวจสอบข้อมูลที่ตัวแอปพลิเคชันส่งมา เมื่อตรวจสอบเสร็จแล้วไปทำงานต่อที่ DDPReadProvider จะตรวจสอบว่า Clients นั้นเป็นเวอร์ชันล่าสุดหรือไม่ หรือมีส่วนเสริมใดบ้างที่สามารถติดตั้งได้ เมื่อตรวจสอบเสร็จแล้ว จะส่งค่าที่ได้คืนให้กับตัวแอปพลิเคชัน ก่อนจะส่งให้กับลูกค้าทราบในขั้นสุดท้าย ซึ่งลูกค้าสามารถเลือกได้ว่าจะติดตั้งหรือไม่



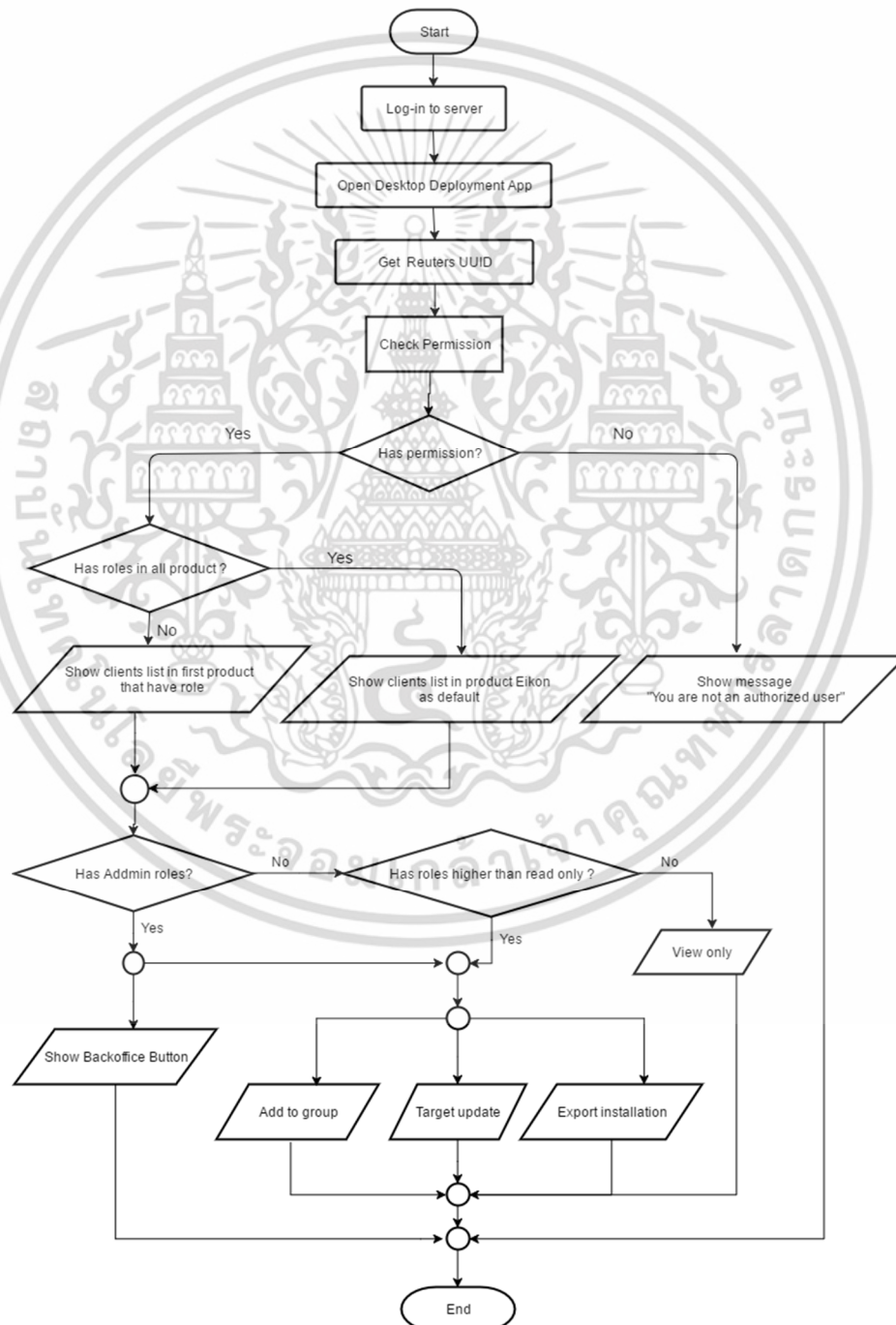
รูปที่ 3.10 Sequence Diagram ฟังก์ชัน Check Update

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) ผังงาน (Flowchart Diagram)

5.1) การเข้าใช้งานระบบของพนักงาน

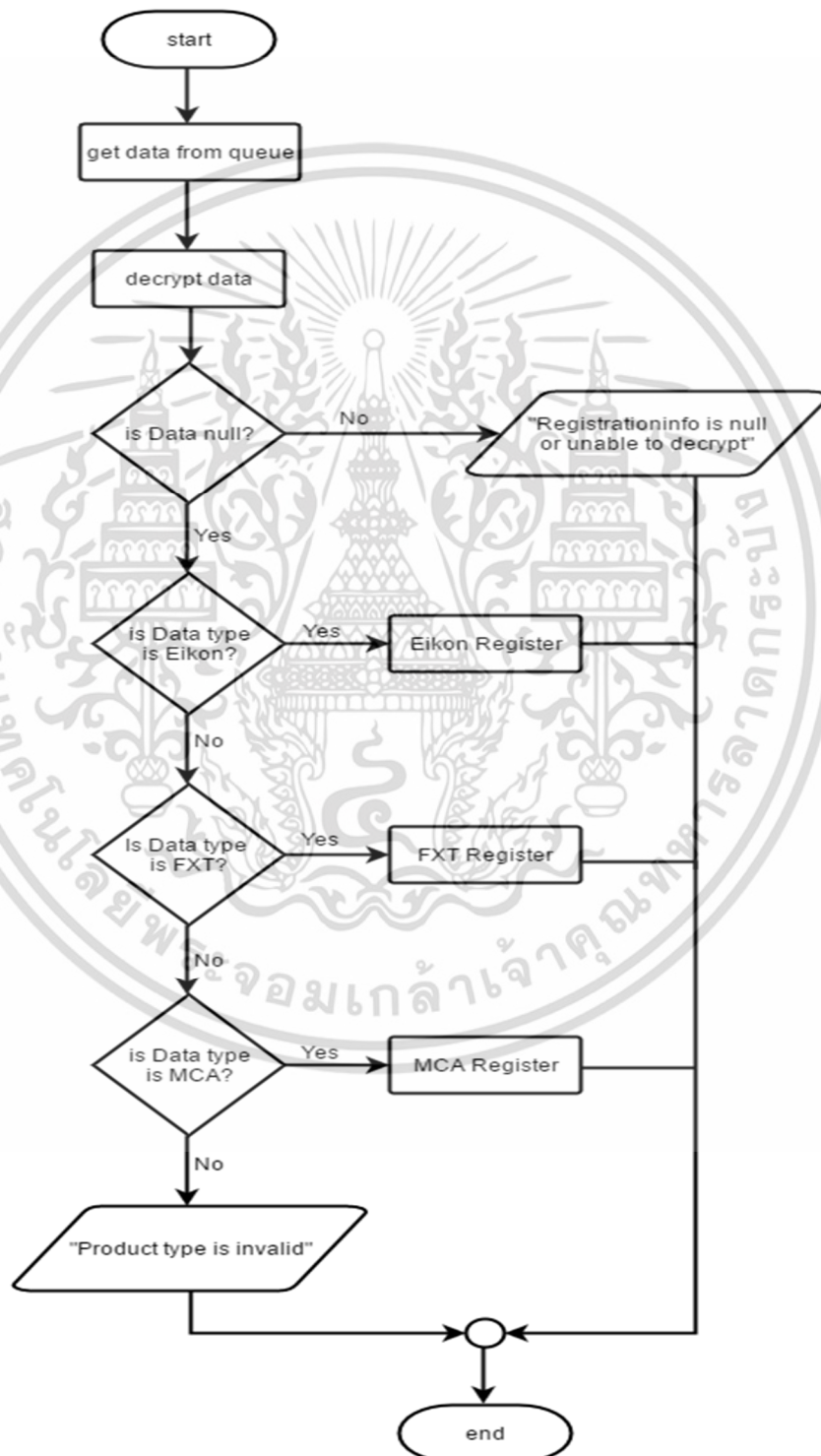
การเข้าใช้ระบบของพนักงานจะต้องทำการเข้าสู่ระบบด้วยบัญชีของพนักงาน จากนั้นตัวเว็บแอปพลิเคชันจะนำบัญชีไปค้นหา UUID ผ่านเซอวิสจาก เพื่อตรวจสอบสิทธิ์ในการเข้าดูข้อมูล โดยต้องมีสิทธิ์ในการเข้าดูอย่างน้อย 1 โปรแกรม ถ้าไม่มีจะไม่สามารถเข้าใช้งานเว็บแอปพลิเคชันนี้ได้ โดยแต่ละสิทธิ์ในการใช้งานจะไม่เท่ากัน ถ้ามีสิทธิ์เป็น Admin จะสามารถเข้าจัดการในส่วน Back Office ได้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่ควรเผยแพร่ไปใช้ประโยชน์ด้านการค้า
 รูปที่ 3.11 Flow Chart การเข้าใช้งานแอปพลิเคชัน
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2) การลงทะเบียนของผลิตภัณฑ์ (Register)

การลงทะเบียนของผลิตภัณฑ์เป็นการรับค่าที่ส่งมาจากเครื่องลูกข่าย โดยจะเซอร์วิสจะรับค่าจาก Rabbit MQ โดยข้อมูลทั้งหมดจะอยู่ในคิว เพื่อรอการอ่านค่า โดยถ้าข้อมูลที่ส่งมาเป็น Package เปล่า จะส่ง Message กลับไป แต่ถ้า Package ส่งมาอย่างถูกต้อง จะถูกถอดรหัสเพื่อเข้าไปลงทะเบียนแยกตามผลิตภัณฑ์นั้นๆต่อไป

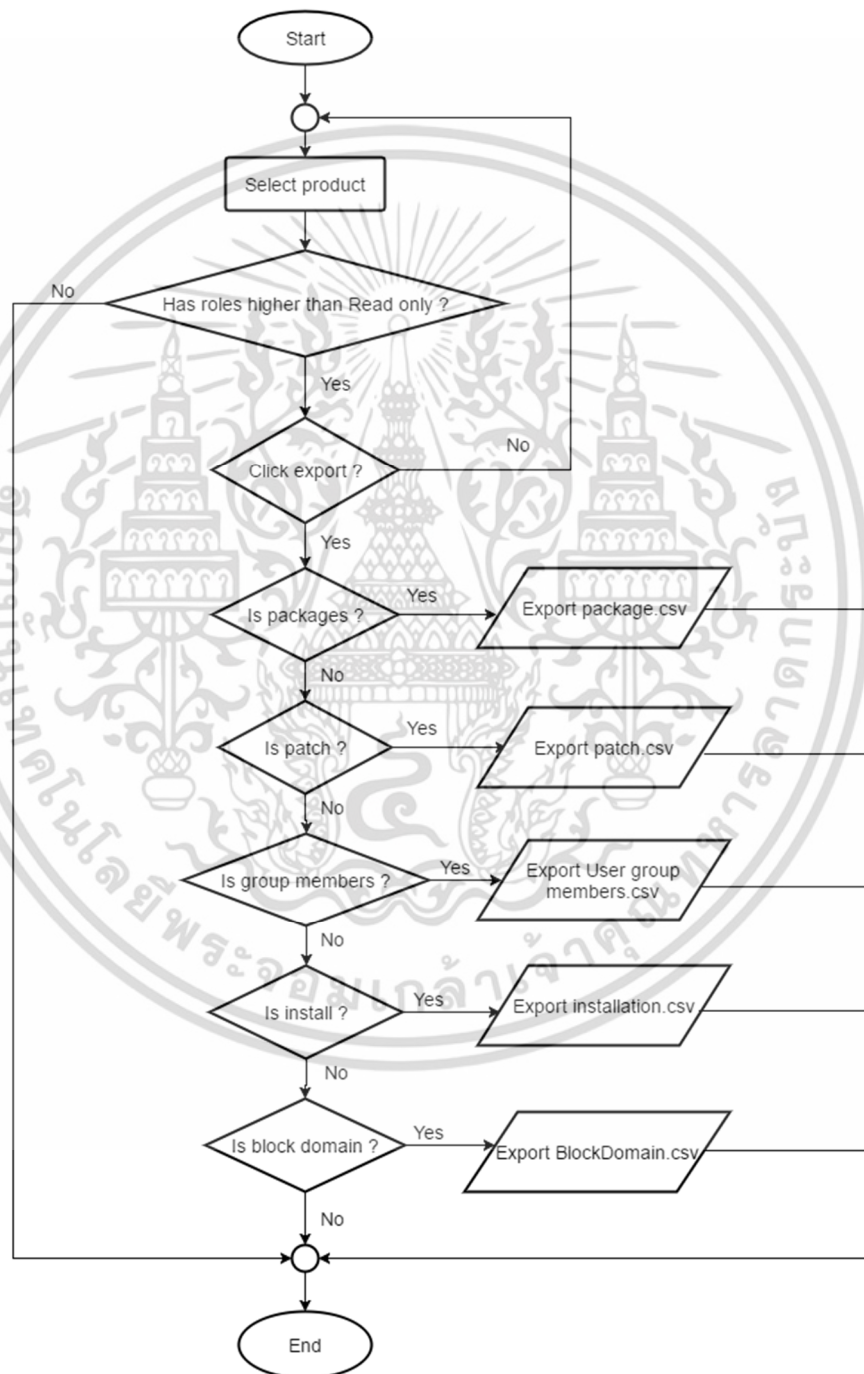


รูปที่ 3.12 Flow Chart การทำงานของฟังก์ชัน Register

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้ดูแลระบบได้แก้ไขเงื่อนไขการดำเนินการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3) การออกรายงานในส่วนต่างๆ ของระบบ Export Excel

ระบบจะทำการตรวจสอบว่าข้อมูลที่พนักงานต้องการดาวน์โหลดเป็นข้อมูลประเภทใด เช่น Package, Patch, Group Members, Installation หรือ Block Domain เมื่อดาวน์โหลดแล้ว ระบบจะนำชื่อผลิตภัณฑ์มาตั้งให้กับชื่อไฟล์ด้วย เช่น EIKON Package, FXT Patch เป็นต้น เพื่อป้องกันไม่ให้ผู้ใช้สับสนชื่อไฟล์ อธิบายได้ดังรูปที่ 3.13

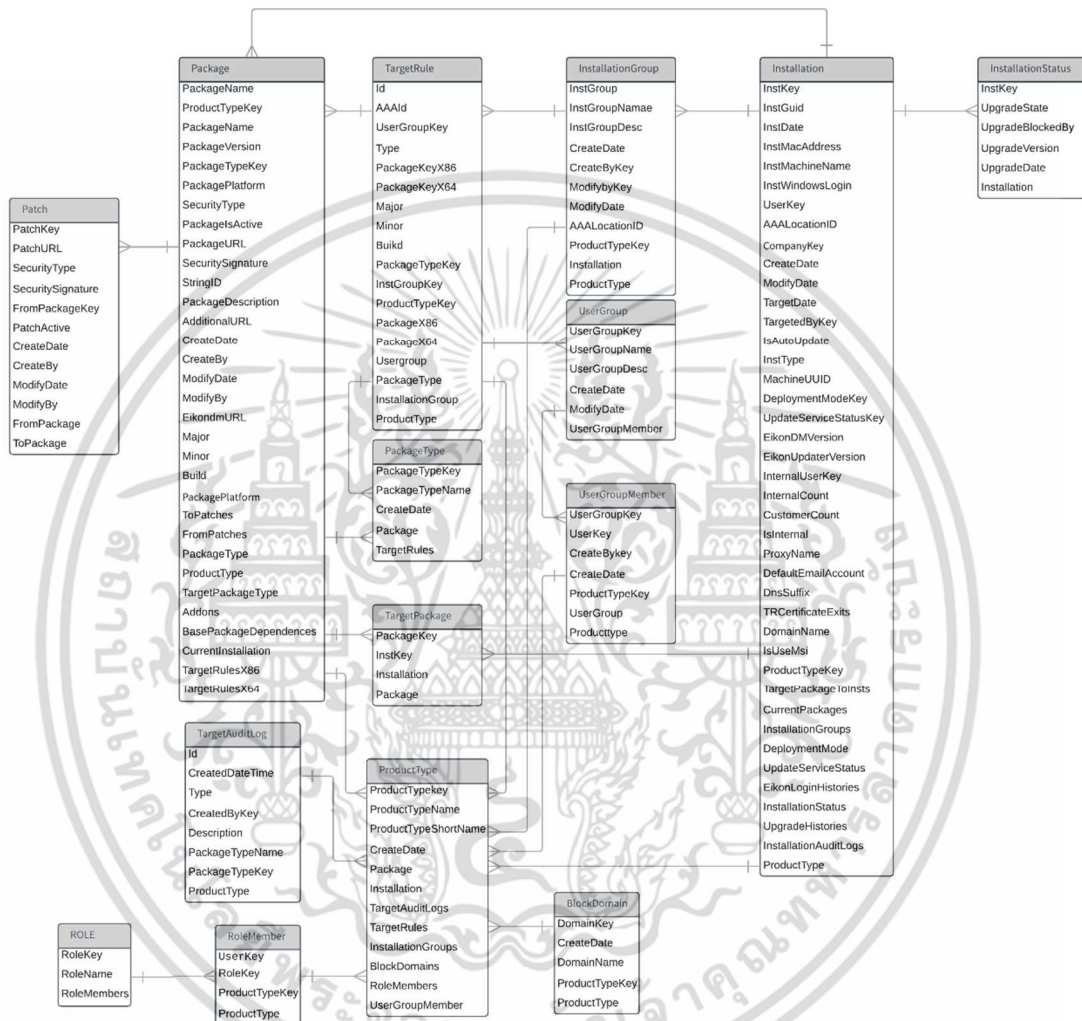


รูปที่ 3.13 Flow Chart ฟังก์ชัน Export

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 การออกแบบฐานข้อมูล

แผนภาพแสดงความสัมพันธ์ระหว่างเอนทิตี (ER Diagram) คือ แบบจำลองความสัมพันธ์ของข้อมูลเพื่อการนำเสนอรายละเอียดต่างๆ ที่เกี่ยวข้องในด้านของความสัมพันธ์ระหว่างข้อมูลในฐานข้อมูลและในการออกแบบ โดยแสดงได้ดังภาพที่ 3.14



รูปที่ 3.14 Entity Relationship ของระบบ

จากรูปที่ 3.14 แผนภาพ Entity Relationship Model (E-R Diagram) ของการพัฒนา ระบบเดสก์ท็อปดีพลอยเมนต์พอยท์ สำหรับควบคุมและติดตามกระบวนการจัดการเกี่ยวกับโปรแกรม ของบริษัท สามารถนำมาแสดงไฟล์ต่างๆในฐานข้อมูล เพื่อแสดงการเก็บรายละเอียดในแต่ละตาราง ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 1) ตาราง RoleMember ใช้สำหรับเก็บ Roles ของผู้ใช้งาน โดย Roles ของแต่ละผลิตภัณฑ์ จะมีความเป็นอิสระต่อกัน คือ Roles ในผลิตภัณฑ์อื่น ไม่สามารถใช้ในอีกผลิตภัณฑ์หนึ่งได้ ตารางนี้มี Primary Key ร่วมกัน 3 คอลัมน์คือ UserKey, RoleKey และ ProductTypeKey อธิบายได้ในตารางที่ 3.1
- 2) ตาราง Installation สำหรับเก็บข้อมูลของ Clients เพื่อใช้สำหรับการ Register, Check Update, Add To Group เป็นต้น เป็นตารางที่มีความสำคัญที่สุดในระบบ โดยตารางนี้มี InstKey เป็น Primary Key อธิบายได้ในตารางที่ 3.2
- 3) ตาราง Package สำหรับเก็บข้อมูล Package ของผลิตภัณฑ์ทั้งสาม จะกำหนดรุ่นและส่วนเสริมได้ต้องใช้ตารางนี้ในการเลือก Package โดยมี Primary Key คือ PackageKey อธิบายได้ในตารางที่ 3.3
- 4) ตาราง User Group Member สำหรับเก็บตำแหน่งของพนักงานที่มีส่วนในการปล่อยผลิตภัณฑ์ให้ใช้งานภายในบริษัท มี Primary Key คือ UserGroupKey, UserKey, ProductTypeKey อธิบายได้ในตารางที่ 3.4
- 5) ตาราง Installation Group สำหรับเก็บข้อมูลกลุ่มที่สร้างในผลิตภัณฑ์นั้นๆ มี Primary Key คือ InstGroupKey กับ ProductTypeKey อธิบายได้ในตารางที่ 3.5
- 6) ตาราง Patch สำหรับเก็บการอัปเดตระหว่างเวอร์ชันของผลิตภัณฑ์ หรือการอัปเดตของส่วนเสริม เป็นต้น มี Primary Key คือ PatchKey อธิบายได้ในตารางที่ 3.6
- 7) ตาราง Block Domain สำหรับบันทึกโดเมนที่ถูกจำกัด ไม่ให้ได้รับการอัปเดต มี Primary Key คือ DomainKey กับ ProductTypeKey อธิบายได้ในตารางที่ 3.7
- 8) ตาราง Target Rules ใช้สำหรับกำหนดรุ่นให้ Clients ส่วนใหญ่จะถูกใช้กับพนักงานภายใน มี Primary Key คือ ID อธิบายได้ในตารางที่ 3.8
- 9) ตาราง Target Audit Log ใช้สำหรับบันทึกข้อมูลการกำหนดรุ่นว่า ใครเป็นผู้ดำเนินการ ทำเมื่อไหร่ เป็นต้น มี Primary Key คือ ID อธิบายได้ในตารางที่ 3.9

ตาราง 3.1 แสดงรายละเอียดของตาราง Role Member

ลำดับที่	ชื่อคอลัมน์	ความหมาย	ประเภท	คีย์
1	UserKey	รหัสพนักงาน	VARCHAR2 (50 BYTE)	PK
2	RoleKey	รหัสตำแหน่ง	NUMBER (3,0)	PK, FK
3	ProductTypeKey	รหัสผลิตภัณฑ์	NUMBER (3,0)	PK, FK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.2 แสดงรายละเอียดของตาราง Installation

ลำดับที่	ชื่อคอลัมน์	ความหมาย	ประเภท	คีย์
1	InstKey	รหัสการติดตั้ง	NUMBER (10,0)	PK
2	InstGuid	รหัส GUID	RAW	
3	MachineUUID	รหัสเครื่องที่ติดตั้ง	RAW	
4	InstDate	วันที่ติดตั้ง	DATE	
5	InstMacAddress	แมคแอดเดรส	VARCHAR2 (20 BYTE)	
6	InstMachinename	ชื่อเครื่องที่ติดตั้ง	VARCHAR2 (50 BYTE)	
7	InstWindowsLogin	ชื่อที่เข้าสู่วินโดว	NVARCHAR2 (100 CHAR)	
8	UserKey	รหัสผู้ใช้งาน	VARCHAR2 (50 BYTE)	
9	AAALocationID	รหัสสถานที่	VARCHAR2 (50 BYTE)	
10	CompanyKey	รหัสบริษัท	NUMBER (10,0)	
11	TargetDate	วันที่กำหนดรุ่น	DATE	
12	TargetedByKey	รหัสกำหนดรุ่น	VARCHAR2 (50 BYTE)	
13	IsAutoUpdate	อัปเดตแบบอัตโนมัติ	NUMBER (1,0)	
14	InstType	ประเภทที่ติดตั้ง	VARCHAR2 (20 BYTE)	
15	UpdateServiceStatusKey	รหัสการอัปเดต	NUMBER (3,0)	FK
16	DeploymentModeKey	รหัสการปล่อยผลิตภัณฑ์	NUMBER (3,0)	FK
17	EikonDMVersion	เวอร์ชันล่าสุด	VARCHAR2 (15 BYTE)	
18	EikonUpdaterVersion	เวอร์ชันของตัวอัปเดต	VARCHAR2 (15 BYTE)	
19	CreateDate	วันที่บันทึกข้อมูล	DATE	
20	ModifyDate	วันที่แก้ไขข้อมูล	DATE	
21	InternalUserKey	รหัสพนักงานภายใน	VARCHAR2 (50 BYTE)	
22	CustomerCount	จำนวนลูกค้า	NUMBER (10,0)	
23	InternalCount	จำนวนพนักงาน	NUMBER (10,0)	
24	IsInternal	พนักงานภายใน	NUMBER (1,0)	
25	TRCertificateExists	มีใบรับรอง	NUMBER (1,0)	
26	ProxyName	ชื่อพร็อกซี	VARCHAR2 (70 BYTE)	
27	DefaultEmailAccount	อีเมล	VARCHAR2 (70 BYTE)	
28	DnsSuffix	ชื่อโดเมนที่เชื่อมต่อ	VARCHAR2 (250 BYTE)	
29	DomainName	ชื่อโดเมน	VARCHAR2 (50 BYTE)	
30	IsUseMsi	ติดตั้งด้วยตัวติดตั้ง	NUMBER (1,0)	
31	ProductTypeKey	รหัสผลิตภัณฑ์	NUMBER (3,0)	FK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 3.3 แสดงรายละเอียดของตาราง Package

ลำดับที่	ชื่อคอลัมน์	ความหมาย	ประเภท	คีย์
1	PackageKey	รหัสแพคเกจ	NUMBER (10,0)	PK
2	ProductTypeKey	รหัสผลิตภัณฑ์	NUMBER (3,0)	FK
3	PackageName	ชื่อแพคเกจ	VARCHAR2 (255 BYTE)	
4	PackageVersion	รุ่นแพคเกจ	VARCHAR2 (15 BYTE)	
5	PackageTypeKey	รหัสประเภทแพคเกจ	NUMBER (3,0)	FK
6	PackagePlatformKey	รหัสแพลตฟอร์ม	NUMBER (3,0)	FK
7	SecurityType	ประเภทของการเข้ารหัส	VARCHAR2 (50 BYTE)	
8	PackagesActive	แพคเกจใช้งานได้	NUMBER (1,0)	
9	PackageURL	URL ของแพคเกจ	VARCHAR2 (350 BYTE)	
10	SecuritySignature	ลายเซ็นการเข้ารหัส	VARCHAR2 (50 BYTE)	
11	StringID	ชื่อผลิตภัณฑ์แบบย่อ	VARCHAR2 (50 BYTE)	
12	PackageDescription	รายละเอียดแพคเกจ	VARCHAR2 (500 BYTE)	
13	AdditionalURL	URL เพิ่มเติม	VARCHAR2 (500 BYTE)	
14	EikondmURL	URL ของไอคอนอัปเดต	VARCHAR2 (350 BYTE)	
15	CreateDate	วันที่บันทึก	DATE	
16	CreateBy	พนักงานที่บันทึก	VARCHAR2 (50 BYTE)	
17	ModifyDate	วันที่แก้ไข	DATE	
18	ModifyBy	พนักงานที่แก้ไข	VARCHAR2 (50 BYTE)	
19	Major	เลขตำแหน่งที่ 1 ในเวอร์ชัน	NUMBER (3,0)	
20	Minor	เลขตำแหน่งที่ 2 ในเวอร์ชัน	NUMBER (3,0)	
21	Build	เลขตำแหน่งที่ 3 ในเวอร์ชัน	NUMBER (10,0)	

ตาราง 3.4 แสดงรายละเอียดของตาราง User Group Member

ลำดับที่	ชื่อคอลัมน์	ความหมาย	ประเภท	คีย์
1	UserGroupKey	รหัสกลุ่ม	NUMBER (10,0)	PK
2	UserKey	รหัสประจำตัว	VARCHAR2 (50 BYTE)	PK, FK
3	CreateByKey	รหัสประจำตัวผู้สร้าง	VARCHAR2 (50 BYTE)	
4	CreateDate	วันที่บันทึก	DATE	
5	ProductTypeKey	รหัสผลิตภัณฑ์	NUMBER (3,0)	PK, FK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 3.5 แสดงรายละเอียดของตาราง Installation Group

ลำดับที่	ชื่อคอลัมน์	ความหมาย	ประเภท	คีย์
1	InstGroupKey	รหัสกลุ่ม	NUMBER (10,0)	PK
2	InstGroupName	ชื่อกุ่ม	VARCHAR2 (50 BYTE)	
3	InstGroupDesc	รายละเอียดของกลุ่ม	VARCHAR2 (500 BYTE)	
4	CreateDate	วันที่สร้าง	DATE	
5	CreateByKey	รหัสพนักงานที่สร้าง	VARCHAR2 (50 BYTE)	
6	ModifyByKey	รหัสพนักงานที่แก้ไข	VARCHAR2 (50 BYTE)	
7	ModifyDate	วันที่แก้ไข	DATE	
8	AAALocationID	รหัสสถานที่	VARCHAR2 (50 BYTE)	
9	ProductTypeKey	รหัสผลิตภัณฑ์	NUMBER (3,0)	FK

ตาราง 3.6 แสดงรายละเอียดของตาราง Patch

ลำดับที่	ชื่อคอลัมน์	ความหมาย	ประเภท	คีย์
1	PatchKey	รหัส Patch	NUMBER (10,0)	PK
2	PatchURL	URL ของ Patch	VARCHAR2 (350 BYTE)	
3	SecurityType	ประเภทความปลอดภัย	VARCHAR2 (50 BYTE)	
4	SecuritySignature	ลายเซ็นความปลอดภัย	VARCHAR2 (50 BYTE)	
5	FromPackageKey	รหัส Package ต้นต้น	NUMBER (10,0)	FK
6	ToPackageKey	รหัส Package ที่จะอัปเดต	NUMBER (10,0)	FK
7	PatchIsActive	Patch ใช้งานได้	NUMBER (1,0)	
8	CreateDate	วันที่สร้าง	DATE	
9	CreateBy	บุคคลที่สร้าง	VARCHAR2 (50 BYTE)	
10	ModifyDate	วันที่แก้ไข	DATE	
11	ModifyBy	บุคคลที่แก้ไข	VARCHAR2 (50 BYTE)	

ตาราง 3.7 แสดงรายละเอียดของตาราง Block Domain

ลำดับที่	ชื่อคอลัมน์	ความหมาย	ประเภท	คีย์
1	DomainKey	รหัสโดเมน	NUMBER (10,0)	PK
2	CreateDate	วันที่บันทึก	DATE	
3	DomainName	ชื่อโดเมน	VARCHAR2 (50 BYTE)	
4	ProductTypeKey	รหัสผลิตภัณฑ์	NUMBER (3,0)	FK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 3.8 แสดงรายละเอียดของตาราง Target Rules

ลำดับที่	ชื่อคอลัมน์	ความหมาย	ประเภท	คีย์
1	Id	รหัส	NUMBER (10,0)	PK
2	AAId	รหัส AAA	VARCHAR2 (50 BYTE)	
3	UserGroupKey	รหัสกลุ่ม	NUMBER (10,0)	FK
4	Type	ประเภท	VARCHAR2 (2 BYTE)	
5	PackageKeyX86	รหัส Package 32 บิต	NUMBER (10,0)	FK
6	PackageKeyX64	รหัส Package 64 บิต	NUMBER (10,0)	FK
7	Major	เลขตำแหน่งที่ 1 ในเวอร์ชัน	NUMBER (3,0)	
8	Minor	เลขตำแหน่งที่ 2 ในเวอร์ชัน	NUMBER (3,0)	
9	Build	เลขตำแหน่งที่ 3 ในเวอร์ชัน	NUMBER (10,0)	
10	PackageTypeKey	รหัสประเภท Package	NUMBER (3,0)	FK
11	InstGroupKey	รหัสกลุ่ม	NUMBER (10,0)	FK
12	ProductTypeKey	รหัสผลิตภัณฑ์	NUMBER (3,0)	FK

ตาราง 3.9 แสดงรายละเอียดของตาราง Target Audit Log

ลำดับที่	ชื่อคอลัมน์	ความหมาย	ประเภท	คีย์
1	Id	ไอดี	NUMBER (10,0)	PK
2	CreatedDateTime	วันที่สร้าง	DATE	
3	Type	ประเภท	VARCHAR2 (25 BYTE)	
4	CreatedByKey	รหัสประจำตัวผู้สร้าง	VARCHAR2 (50 BYTE)	
5	Description	รายละเอียด	VARCHAR2 (255 BYTE)	
6	PackageTypeName	ชื่อประเภท Package	VARCHAR2 (255 BYTE)	
7	ProductTypeKey	รหัสผลิตภัณฑ์	NUMBER (3,0)	FK

3.4 ขั้นตอนการทดสอบระบบ

สำหรับการทดสอบระบบนั้น ผู้จัดทำได้ดำเนินการโดยแบ่งเป็นส่วนๆ ได้ดังนี้

- 1) ทดลองใช้งานระบบจริงๆ โดยการจำลองข้อมูลของผลิตภัณฑ์อื่นขึ้นมา
- 2) ทดสอบการ Register ของผลิตภัณฑ์ FXT และ MCA
- 3) ทดสอบการ Target Update ของผลิตภัณฑ์ FXT และ MCA
- 4) ทดสอบการเพิ่มกลุ่มของผลิตภัณฑ์ FXT และ MCA
- 5) ทดสอบใช้งานในส่วน Back Office สำหรับการรองรับ FXT และ MCA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.1 ขอบเขตของการทดสอบระบบส่วนการลงทะเบียนข้อมูล (Register)

ระบบเดสทอปดีพลอยเมนต์พ็อตล จะมีส่วนที่เปิดรับข้อมูลของเครื่องที่ติดตั้งโปรแกรม Eikon และโปรแกรม FXT โดยข้อมูลจะถูกส่งเข้ามาที่ระบบ และระบบจะทำการบันทึกลงฐานข้อมูล เพื่อสำหรับใช้งานต่อไป ข้อมูลที่เก็บเข้ามา ประกอบไปด้วย InstallationGuid, MachineUUID, MacAddress, Version, LoginHistory เป็นต้น การทดสอบระบบส่วนการลงทะเบียนสามารถเขียนเป็นภาษา Gherkin ได้ดังตัวอย่างนี้

Scenario Outline: Register Unexisting case, every field is stored correctly

When I register with these arguments

Field	Value
InstallationGuid	00000001-0000-0000-0000-000000000000
MachineUUID	00000001-0000-0000-0000-000000000000
MacAddress	11-22-33-44-55-66
MachineName	Moozz-PC
WindowsLogin	Admin
InstallType	MACHINE
EikonUserName	live_user@other.com
DeploymentMode	CUSTOMER_MANAGED
EikondmVersion	1.0.0
UpdaterVersion	1.2.0
TRCertificateExists	false
ProxyName	Proxy
DefaultEmailAccount	DefaultEmailAccount
DnsSuffix	DnsSuffix
DomainName	DomainName
Type	<Product>

Then These properties of the installation 00000001-0000-0000-0000-000000000000 on machine 00000001-0000-0000-0000-000000000000 should be

Field	Value
MacAddress	11-22-33-44-55-66
MachineName	Moozz-PC
WindowsLogin	Admin
Type	MACHINE
UserLogin	live_user@other.com
DeploymentMode	CUSTOMER_MANAGED
EikonDMVersion	1.0.0
EikonUpdaterVersion	1.2.0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TRCertificateExists	false	
ProxyName	Proxy	
DefaultEmailAccount	live_user@other.com	
DnsSuffix	DnsSuffix	
DomainName	DomainName	

Examples:

Product	
Eikon	
FXT	

จากภาษา Gherkin อธิบายได้ คือ เมื่อมีการลงทะเบียนข้อมูลผลิตภัณฑ์ที่ไม่มีในฐานข้อมูล จะเก็บข้อมูลใหม่เหล่านั้นลงฐานข้อมูลได้อย่างถูกต้อง เช่น เมื่อมีการเก็บข้อมูลของผู้ใช้ที่มี InstallationGuid คือ 00000001-0000-0000-0000-000000000000 เมื่อบันทึกลงฐานข้อมูลและตรวจสอบจะต้องมีข้อมูลตามที่เก็บได้ เป็นต้น สามารถเขียนเป็นตารางโดยรวมได้ดังนี้

ตาราง 3.10 การทดสอบ Register

ลำดับที่	เงื่อนไขการทดสอบ	ผลลัพธ์ที่คาดหวัง	ผล
1	Register Unexisting case, every field is stored correctly	Every field is stored correctly	ผ่าน
2	Register Unexisting case, InstallType and UpdateServiceEnable affects UpdateServiceStatus	UpdateServiceStatus is set correctly	ผ่าน
3	Register Existing case, Current packages are changed correctly	packages changed correctly	ผ่าน
4	Register with blocked domains, every field is stored correctly	Every field is stored correctly	ผ่าน
5	Register unexisting case, register with non-internal login. Customer count must be one.	Customer count is one	ผ่าน
6	Register unexisting case, register with internal login. internal count must be one.	internal count is one.	ผ่าน
7	Register existing case, register with non-internal login. customer count must be increased by one.	customer count is increased by one.	ผ่าน
8	Register existing case, register with internal login. internal count must be increased by one.	internal count is increased by one.	ผ่าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 3.10 การทดสอบ Register (ต่อ)

9	Register unexisting case, login history properties must be stored correctly	Login history properties is stored correctly	ผ่าน
10	Register existing case, Login history properties must be stored correctly	Login history properties is stored correctly	ผ่าน
11	Register existing case, Register twice, login history properties must be stored correctly	Login history properties is stored correctly	ผ่าน
12	Register existing case, Register more than 100 times, the first login history must be removed	First login history is removed correctly	ผ่าน

3.4.2 ขอบเขตของการทดสอบระบบส่วนการจัดการ Role

ระบบเดสทอปที่ปลอดภัยเหมาะที่พอหัล สามารถอนุญาตให้ผู้ใช้ที่มี Role สูงสุดสามารถกำหนด Role ให้กับผู้ใช้คนอื่นได้ โดยผู้ใช้ที่มี Role ต่างกัน จะมีความสามารถในการใช้งานโปรแกรมต่างกันด้วย เช่น ผู้ใช้ประเภท Read Only สามารถอ่านข้อมูลได้อย่างเดียว แต่ผู้ใช้ประเภท Write Access สามารถเขียนแก้ไขข้อมูลได้ เป็นต้น สามารถเขียนเป็นภาษา Gherkin ได้ดังตัวอย่างนี้

Scenario Outline: Add a user to roles

When I set roles of the user test@db.com to WRITE_ACCESS, READ_ONLY in product <ProductType>

Then the role of user test@db.com must be WRITE_ACCESS in product <ProductType>

And the role of user test@db.com must be READ_ONLY in product <ProductType>

Examples:

ProductType	
Eikon	
FXT	

จากภาษา Gherkin อธิบายได้ คือ เมื่อทำการกำหนด user ที่ใช้อีเมล test@db.com ให้เป็น WRITE_ACCESS และ READ_ONLY ผลลัพธ์ที่ได้คือ ต้องมี Roles ตามที่กำหนดอย่างถูกต้องในฐานข้อมูล สามารถเขียนเป็นตารางโดยรวมได้ดังนี้

ตาราง 3.11 การทดสอบการจัดการ Role

ลำดับที่	เงื่อนไขการทดสอบ	ผลลัพธ์ที่คาดหวัง	ผล
1	Add a user to roles	Users' roles is set correctly	ผ่าน
2	Add a user to roles, some are removed	Users' roles is change correctly	ผ่าน
3	Add a user to user groups	Users' groups is set correctly	ผ่าน
4	Add a user to user groups, some are removed	Users' groups is change correctly	ผ่าน

3.4.3 ขอบเขตของการทดสอบระบบส่วนการตรวจสอบ Update

สำหรับการอัปเดตโปรแกรม Eikon และโปรแกรม FXT สามารถทำได้ผ่านระบบทดสอบที่ออปติไมซ์เพื่อทดสอบ โดยจะใช้ข้อมูลที่เก็บได้จากการ Register มาประมวลผล และทำการปล่อย Patch หรือ Package ที่เหมาะสมกับผู้ใช้นั้นๆ สามารถเขียนเป็นภาษา Gherkin ได้ดังตัวอย่างนี้

Scenario Outline: Check update, check all fields

Given Everyone is targeted to product <ProductType> package version 4.0 for package type 1

And Everyone is targeted to product <ProductType> package version 4.1 for package type 1

And These Installations are available

InstGuid : 00000001-0000-0000-0000-000000000000

MachineUUID : 00000001-0000-0000-0000-000000000000

CurrentPackageKeys : <CurrentPackageKeys>

ProductTypeKey : <ProductTypeKey>

When I check for updates with these arguments

InstallationGuid : 00000001-0000-0000-0000-000000000000

MachineUUID : 00000001-0000-0000-0000-000000000000

Patchable : true

PackagePlatform : x86

Version : 4.0

ProductType : <ProductTypeKey>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Then I should get this package info list

```
| Key | IsPatch | Name | Url | InternetUrl | PackageType | StringId | Checksum |
| <Key> | false | <Name> | <Url> | <InternetUrl> | Core | <StringId> | s2 |
```

Examples:

```
ProductType      : Eikon, Fxt   CurrentPackageKeys : 1, 3
ProductTypeKey   : 1, 3       Key                  : 2, 4
Name             : Eikon_x86_4_1.eik, FXT_x86_4_1.eik
Url              : http://172.20.41.185/Eikon_x86_4_1.eik, http://172.20.41.185/FXT_x86_4_1.eik
InternetUrl      : http://172.20.41.186/Eikon_x86_4_1.eik, http://172.20.41.186/FXT_x86_4_1.eik
StringId         : EIKON4.1, FXT4.1
```

จากภาษา Gherkin อธิบายได้ คือ เมื่อมีการ Check update ผลลัพธ์ที่ จะตรวจสอบ แพ็คเก็จล่าสุดที่ปล่อยให้ดาวน์โหลด โดยแต่ละโปรแกรมเมื่อทำการตรวจสอบสามารถได้แพ็คเก็จที่ ถูกต้อง เช่น ให้โปรแกรมทั้งสองมีแพ็คเก็จปัจจุบันที่ 4.0 เมื่ออัปเดตจะได้เวอร์ชัน 4.1 โปรแกรม Eikon จะได้แพ็คเก็จ Eikon_x86_4_1.eik มี URL คือ http://172.20.41.185/Eikon_x86_4_1.eik ส่วนโปรแกรม FXT จะได้แพ็คเก็จ FXT_x86_4_1.eik และมี URL คือ http://172.20.41.185/FXT_x86_4_1.eik เป็นต้น สามารถเขียนตารางทดสอบได้ดังนี้

ตาราง 3.12 การทดสอบการตรวจสอบ Update

ลำดับที่	เงื่อนไขการทดสอบ	ผลลัพธ์ที่คาดหวัง	ผล
1	Check update, check all fields	User get update correctly	ผ่าน
2	Check update, check all fields for patch	User get path correctly	ผ่าน
3	Check update, block account should get No Update	block account get No Update	ผ่าน
4	Check update new core and new addons	User get update correctly	ผ่าน
5	Check update new core and new addons (some addons is not available on new core)	User get update correctly	ผ่าน
6	Check update new core and new addons (user didn't install some addons)	User get addons update correctly	ผ่าน
7	Check update core version mismatch	User get core update correctly	ผ่าน
8	Check update addons version mismatch	User get addons update correctly	ผ่าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 3.12 การทดสอบการตรวจสอบ Update (ต่อ)

ลำดับที่	เงื่อนไขการทดสอบ	ผลลัพธ์ที่คาดหวัง	ผล
9	CheckUpdate Empty Location, Legal Entity or Ultimate Parent, it should get No Update	Legal Entity or Ultimate Parent get No Update	ผ่าน
10	CheckUpdate Empty Location but the installation group is targeted to an update, it should get the update	Empty Location get the update	ผ่าน
11	CheckUpdate Empty Location but the user is targeted to an update, it should get the update	Empty Location get the update	ผ่าน
12	CheckUpdate Empty Location but the user group is targeted to an update, it should get the update	Empty Location get the update	ผ่าน
14	Check update, Patch is returned	Patch is returned correctly	ผ่าน
15	Check update, Patch is not returned	Patch is not returned	ผ่าน
16	Check update, Patch : addon get full package because core is updated	Addon get full package	ผ่าน
17	Check update, Patch : addon get patch because core is not updated	Addon get patch	ผ่าน
19	CheckUpdate with unknown package registered return Up-to-date	Package is Up-to-date	ผ่าน
21	CheckUpdate with unknown package registered return Up-to-date when installation is set to no updates	Package is Up-to-date	ผ่าน
24	CheckUpdate for each visibility when installation is targeted	User get update correctly	ผ่าน
25	Check update, data stream addon on Per User should not be returned	User not get update	ผ่าน
26	Check update, data stream addon on Per Machine should be returned	Addon is returned	ผ่าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.4 ขอบเขตของการทดสอบระบบส่วนการสร้างกลุ่ม

เป็นการทดสอบสร้างกลุ่มในแต่ละผลิตภัณฑ์ เพื่อสามารถแยกผู้ใช้งานออกเป็นหลายๆส่วน เพื่อใช้สำหรับทดสอบการปล่อย Package หรือสำหรับผู้ที่ใช้ที่ต้องการสิทธิพิเศษ โดยชื่อกลุ่มนั้นต้องไม่เป็นช่องว่าง มีขนาดไม่เกิน 50 ตัวอักษร และห้ามตั้งชื่อกลุ่มเหมือนกัน

Scenario Outline: Create a group

When I create an installation group "<Group>" with the description "Just want to test" in product <ProductType>

Then There must be a "<Group>" with the description "Just want to test" in the installation group list of product <ProductType>

Examples: | ProductType | Group |
 | Eikon | My Group Name |
 | FXT | My Group Name2 |

จากภาษา Gherkin อธิบายได้ คือ เมื่อทำการสร้างกลุ่มสำหรับโปรแกรม Eikon และ FXT มีชื่อว่า My Group Name และ My Group Name 2 พร้อมใส่รายละเอียดว่า “Just want to test” จะต้องพบข้อมูลกลุ่มทั้ง 2 บนฐานข้อมูลอย่างถูกต้อง การทดสอบทั้งหมดแสดงได้ดังตารางนี้

ตาราง 3.13 การทดสอบการตรวจสอบ Create Group

ลำดับที่	เงื่อนไขการทดสอบ	ผลลัพธ์ที่คาดหวัง	ผล
1	Create a group	Group is created correctly	ผ่าน
2	Create a group, name is empty	Show message The group name must contain at least 1 character.	ผ่าน
3	Create a group, duplicated group name	Show error with message The same group name already exists.	ผ่าน
4	Create a group, name is too long	Show error with message Maximum length for group name is 50	ผ่าน
5	Edit a group name	Group name is changed correctly	ผ่าน
6	Edit a group, name is empty	Show message The group name must contain at least 1 character.	ผ่าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 3.13 การทดสอบการตรวจสอบ Create Group (ต่อ)

ลำดับที่	เงื่อนไขการทดสอบ	ผลลัพธ์ที่คาดหวัง	ผล
7	Edit a group, duplicated group name	Show error with message The same group name already exists.	ผ่าน
8	Edit a group, name is too long	Show error with message Maximum length for group name is 50	ผ่าน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

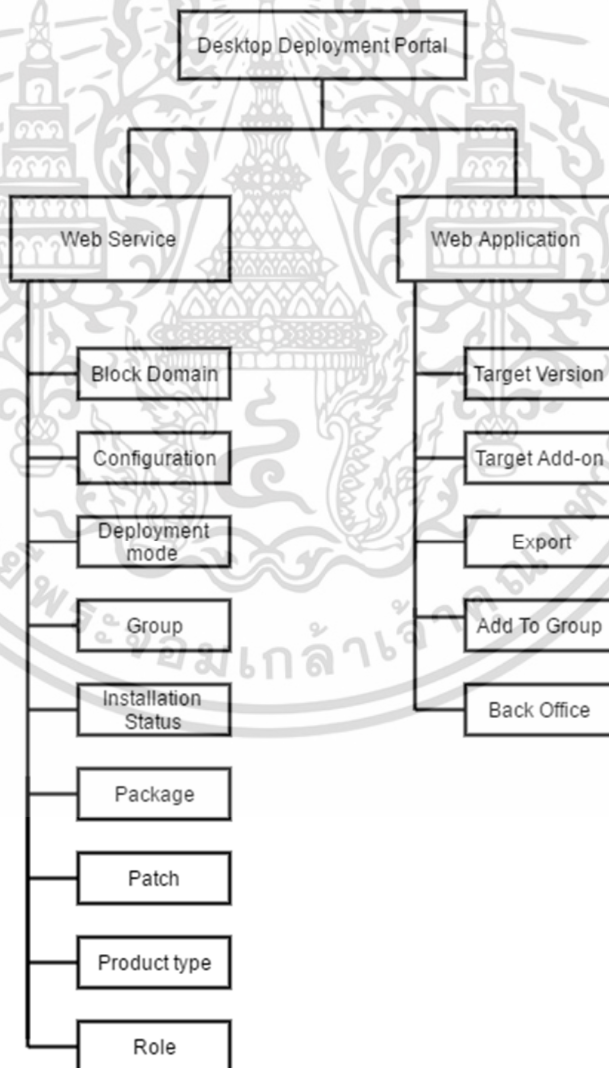
บทที่ 4

ผลการดำเนินงานและการอภิปรายผล

4.1 โครงสร้างของระบบ

ระบบเดสก์ทอปดีพลอยเมนต์ พอร์ตัล ประกอบด้วยโครงสร้าง ดังรูปที่ 4.1 แบ่งเป็น 2 ส่วนหลัก คือ Web Service และ Web Application

ในส่วนของ Web Service ประกอบด้วย block Domain โดเมนบางประเภทไม่สามารถเข้าใช้งานได้ Configuration คือ ส่วนตั้งค่าของผู้ใช้ Deployment mode คือ ส่วนของการเลือกวิธีการปล่อย Package Group คือ ส่วนของการสร้างกลุ่มแยกตามผลิตภัณฑ์ เพื่อเป้าหมายบางประการ Installation Status คือสถานะของการติดตั้งโปรแกรม Package, Patch, Product type คือข้อมูลของโปรแกรมต่างๆ และ Role คือ สิทธิการเข้าใช้ระบบของผู้ใช้



รูปที่ 4.1 โครงสร้างของระบบ

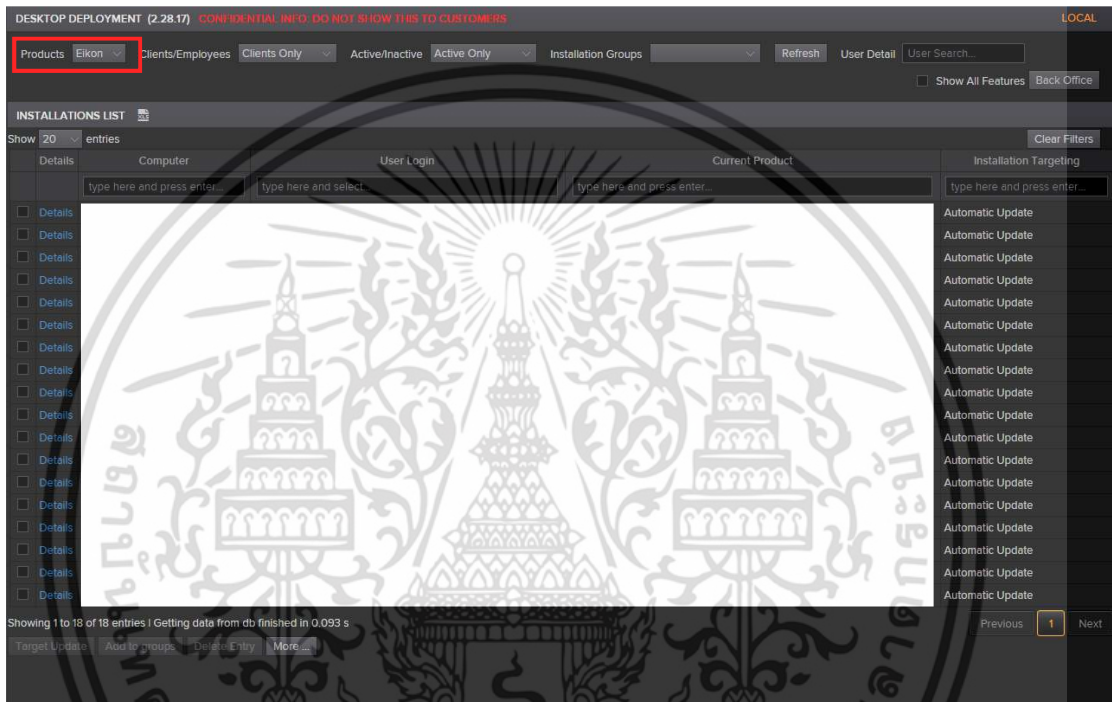
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ผลการดำเนินงาน

4.2.1 ส่วนของผู้ใช้งานทั่วไป

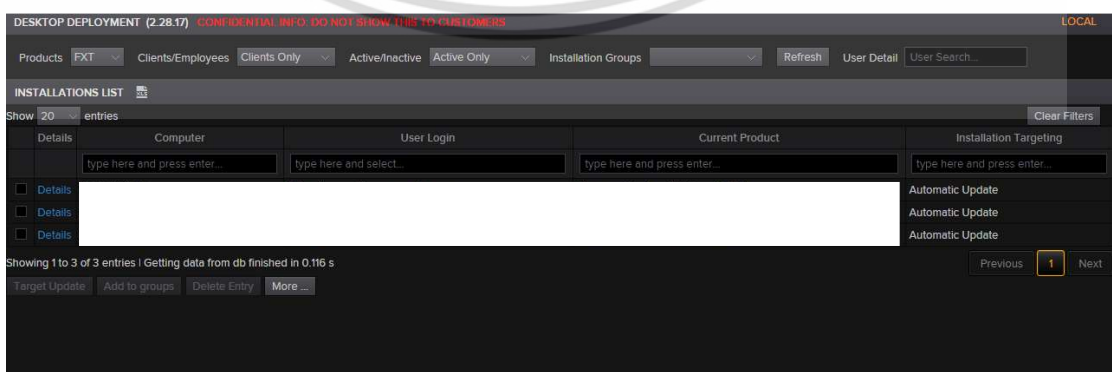
- หน้าแรกของโปรแกรม

เมื่อผู้ใช้งาน (Employee) ทำการเข้าสู่ระบบในเซิร์ฟเวอร์ และได้รับการตรวจสอบสิทธิ์การใช้งานว่า สามารถใช้งานแอปพลิเคชันได้ จะเข้ามาพบกับหน้าแรก ดังรูป 4.2 โดยได้เพิ่ม Dropdown ที่เป็นรายการของผลิตภัณฑ์ที่ผู้ใช้มีสิทธิ์ดู สามารถเลือกเปลี่ยนรายการผลิตภัณฑ์ได้



รูปที่ 4.2 หน้าหลักของเว็บแอปพลิเคชัน (1)

เมื่อผู้ใช้เปลี่ยน Dropdown จะเห็นว่ารายการของผู้ใช้เปลี่ยนไปจากผลิตภัณฑ์ Eikon เป็นผลิตภัณฑ์ FXT และปุ่ม Back Office ได้หายไป เนื่องจากผู้ใช้งานไม่มีสิทธิ์เป็น Admin ในผลิตภัณฑ์ FXT

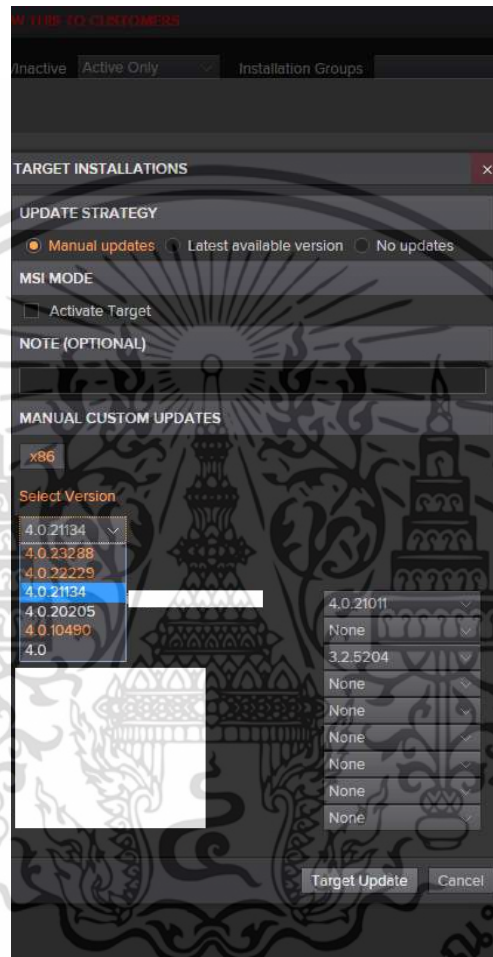


รูปที่ 4.3 หน้าหลักของเว็บแอปพลิเคชัน (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ส่วนการกำหนดรุ่นและส่วนเสริม

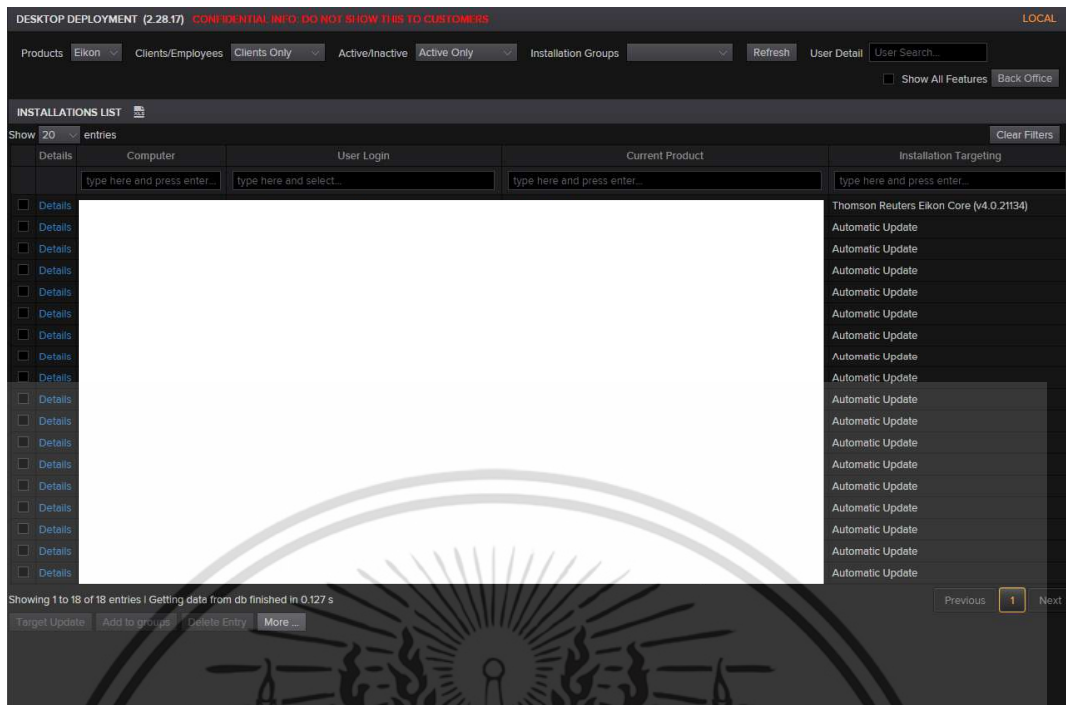
เมื่อผู้ใช้ทำการเลือกเครื่อง Clients ที่ต้องการได้แล้ว ทำการกดที่ Target Update จะพบหน้าต่าง Target Installation ดังรูป ผู้ใช้สามารถเลือกรุ่นและส่วนเสริมได้จากรายการที่มีให้ โดยทั้งรุ่นและส่วนเสริมเหล่านั้นได้ทำการกรองมาเรียบร้อยแล้ว ทำให้ผู้ใช้ไม่ต้องกังวลว่าจะทำการกำหนดผิด โดยเมื่อผู้ใช้เลือกได้แล้ว จึงกดปุ่ม Target Update เพื่อเป็นการยืนยัน



รูปที่ 4.4 หน้าต่าง Target Installation

เมื่อกดปุ่ม Target Update ระบบจะทำการเปลี่ยนค่าในคอลัมน์ Installation Targeting เป็น Package ที่ได้เลือกไว้ ดังรูป 4.5 โดยเมื่อเครื่องลูกข่ายทำการอัปเดตโปรแกรม ก็จะได้ Package ที่ได้เลือกไว้นั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

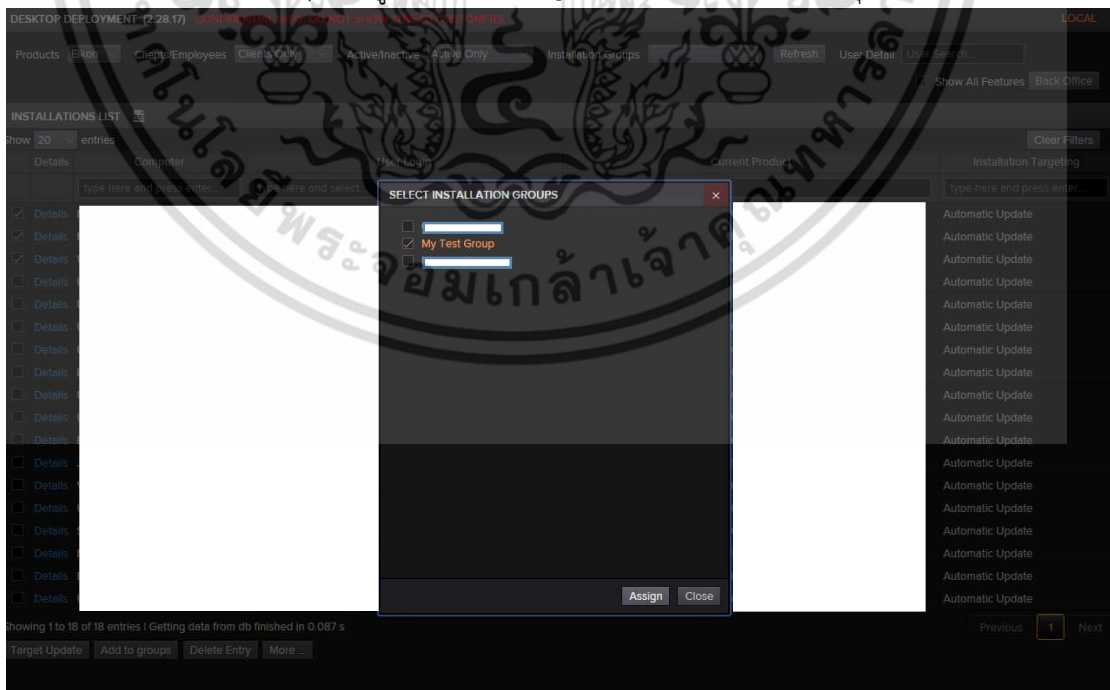


รูปที่ 4.5 การกำหนดเวอร์ชันให้กับ Clients

- การจัดกลุ่มของ Clients

ผู้ใช้เลือกรายชื่อเครื่อง Clients ที่ต้องการ และกดที่ปุ่ม Add to groups จะปรากฏหน้าต่าง

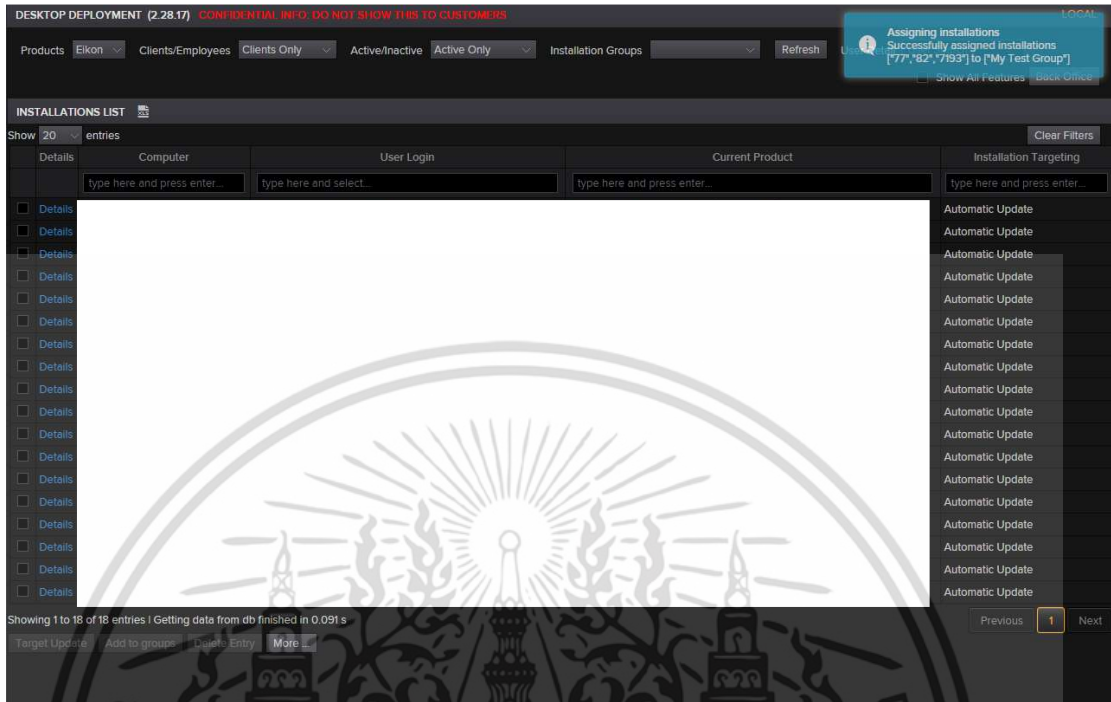
Select Installation Groups ดังรูป 4.6 กด Assign เพื่อยืนยันการเพิ่มเข้ากลุ่ม



รูปที่ 4.6 หน้าต่าง Select Installation Groups

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อกด Assign แล้วจะมี ข้อความแสดงผลการเพิ่ม client เข้ากลุ่ม ดังรูป 4.7 ถ้าการเพิ่ม
 เข้ากลุ่มสำเร็จจะปรากฏเป็นสีเขียว ถ้าไม่สำเร็จจะเป็นสีแดง



รูปที่ 4.7 Toast add to group success

เมื่อต้องการเข้าไปดูกลุ่ม สามารถดูได้โดยการเลือกที่หัวข้อ Installation Groups แล้วเลือก
 กลุ่มที่ต้องการ จะเห็นว่ารายชื่อที่เลือกไว้ ได้เข้ามาในกลุ่มเป็นที่เรียบร้อยแล้ว

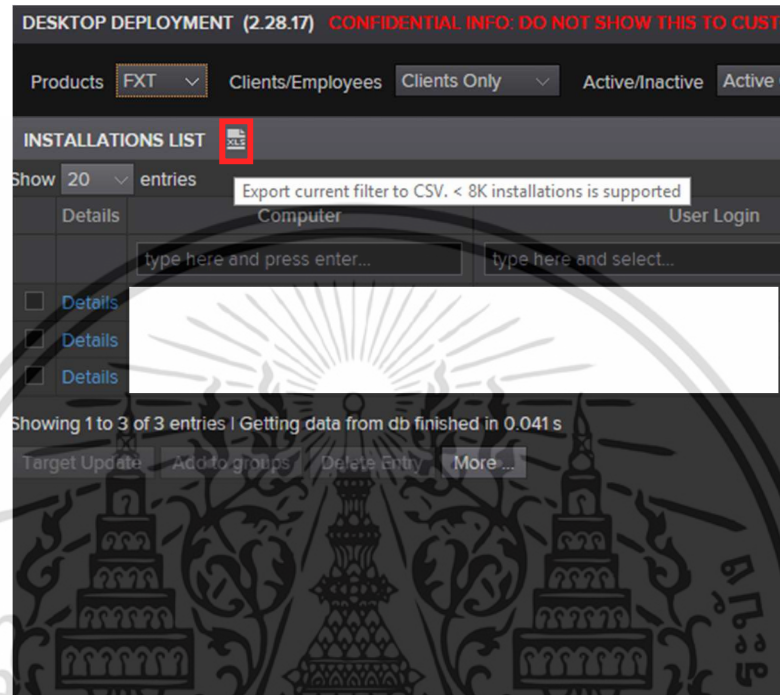


รูปที่ 4.8 หน้า Installation Groups

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การออกรายงาน

ผู้ใช้สามารถออกรายงานได้ตามสิทธิ์ที่ผู้ใช้งานมี โดยผู้ใช้ที่ไม่ใช่สิทธิ์แบบ READ_ONLY สามารถดาวน์โหลดรายงานได้ โดยสามารถทำได้โดยกดที่ปุ่มที่อยู่ด้านหลังหัวข้อ Installation List ดังรูป 4.9



รูปที่ 4.9 ปุ่ม Export Installation

เมื่อดาวน์โหลดรายงาน จะได้ไฟล์นามสกุล csv ออกมา โดยถ้าข้อมูลมีปริมาณมาก การดาวน์โหลดจะยิ่งใช้เวลานานตามไปด้วย



รูปที่ 4.10 ดาวน์โหลด Export

จากรูป 4.10 เมื่อระบบดาวน์โหลดไฟล์เสร็จแล้ว สามารถเปิดได้ด้วยโปรแกรม Microsoft Excel จะได้ข้อมูลดังรูป ซึ่งการออกรายงานแต่ละครั้ง ผู้ใช้สามารถกำหนดขอบเขตของข้อมูลได้ เช่น การเข้ากลุ่ม เป็นต้น

Export Date	Rows Count				
เส. 16. 18 2559 1:15 AM	3				
Installation Id	Computer	User Login	UUID	Current Product	Installation Targeting
					Automatic Update
					Automatic Update
					Automatic Update

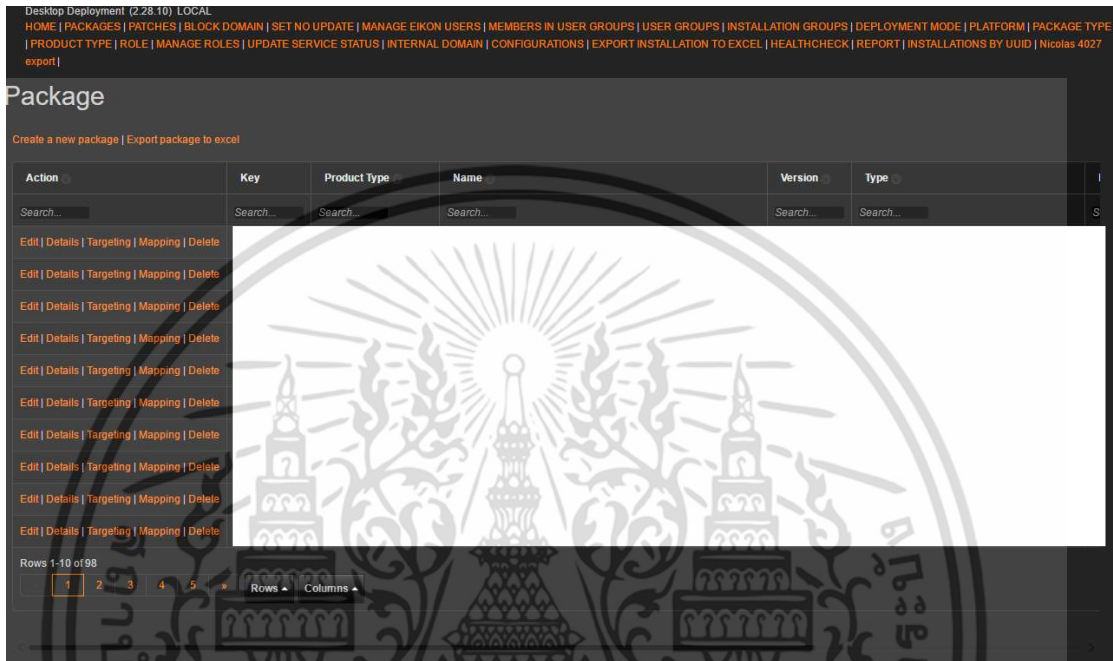
รูปที่ 4.11 ตัวอย่างไฟล์ Report

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2 การเข้าใช้งานส่วน Back Office

การเข้าใช้งานส่วน Back Office แบบเก่านั้น ในเวอร์ชันเก่าส่วนของเมนูจะดูแออัด และไม่สามารถเลือกผลิตภัณฑ์ได้เพราะรองรับแค่ผลิตภัณฑ์ Eikon โดยได้ทำการจัดระเบียบเมนูให้ดูสะอาดมากขึ้น ด้วยการย้ายจากปุ่มเป็นดรอปดาวน์เมนู และเพิ่มดรอปดาวน์สำหรับผลิตภัณฑ์เข้ามาดังรูป

4.13



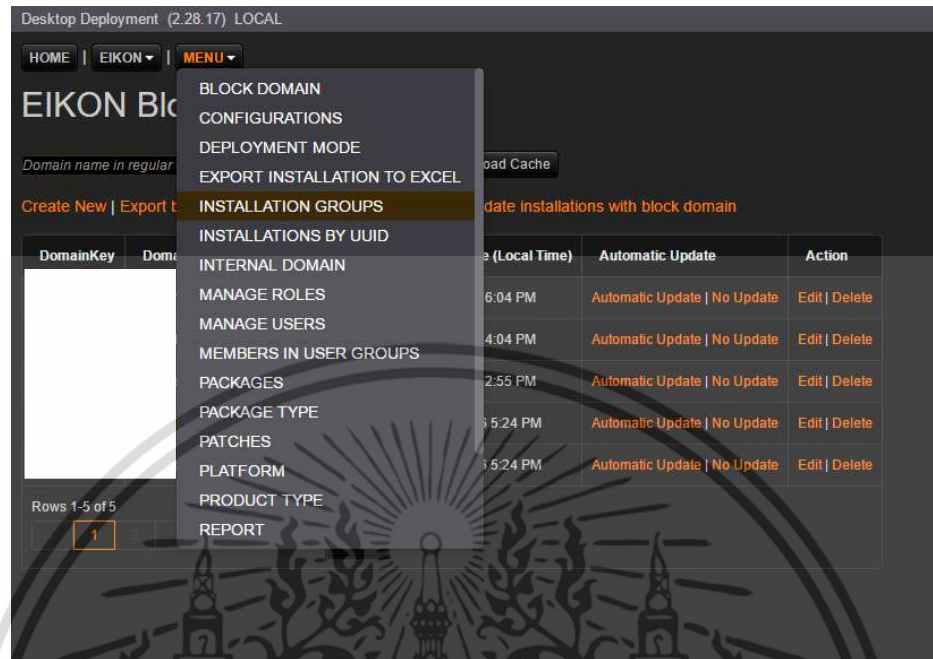
รูปที่ 4.12 หน้า Back Office เวอร์ชันเก่า



รูปที่ 4.13 หน้า Back Office เวอร์ชันล่าสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

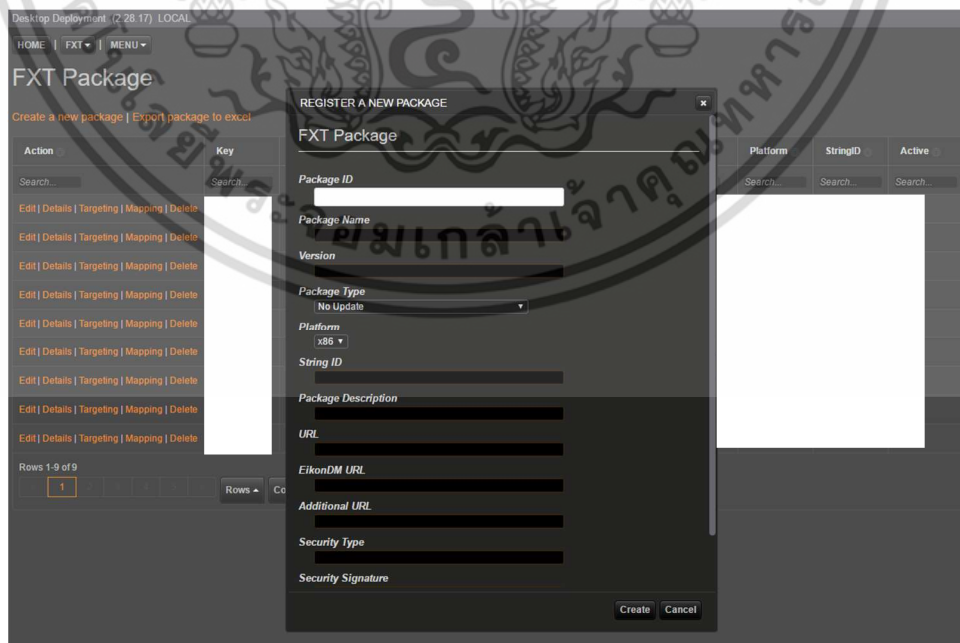
การเปลี่ยนหมวดในหน้า Back Office จะเป็น Drop down menu เพื่อคัดเลือก Menu ที่ต้องการ โดยผู้ที่จะเข้าใช้งานหน้า Back Office ได้นั้นต้องเป็นผู้ใช้ที่มีสิทธิ์เป็น Admin เท่านั้น



รูปที่ 4.14 Menu ของหน้า Back Office เวอร์ชันล่าสุด

- การเพิ่มข้อมูล Package

เป็นส่วนใช้เพิ่ม, ลบ และแก้ไข Package ของโปรแกรมต่างๆ จากรูปเป็นของโปรแกรม FXT

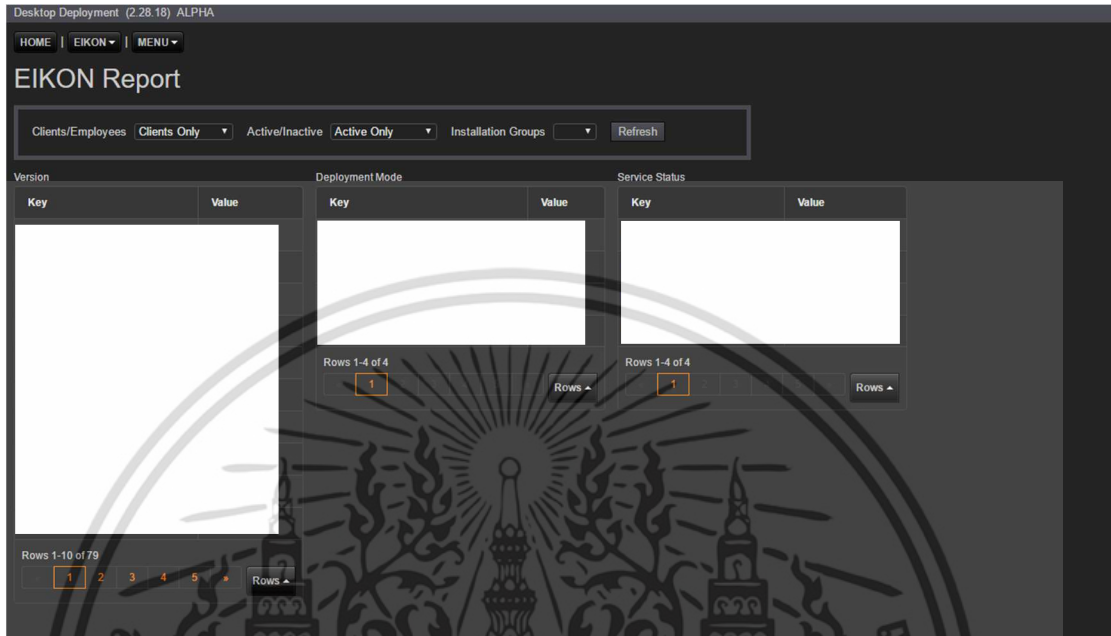


รูปที่ 4.15 Dialog การเพิ่ม Package ของโปรแกรม FXT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- หัวข้อ Report

ได้ทำการปรับปรุงใหม่ โดยเพิ่มความสามารถให้ดูข้อมูลได้เจาะจงมากยิ่งขึ้น โดยเป็นสรุปของ Package ที่มีทั้งหมดของระบบ จะแสดงจำนวนผู้ใช้ที่ติดตั้ง Package นั้นๆ



รูปที่ 4.16 หน้า Report

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการดำเนินงานและข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

- 1) ระบบเดสก์ท็อปดีฟลอยเม้นท์พอร์ทลสามารถแสดงข้อมูลรายละเอียดของเครื่องลูกข่ายที่ติดตั้งซอฟต์แวร์ผลิตภัณฑ์ทั้งสามได้ถูกต้อง โดยข้อมูลจะถูกกรองและแสดงผลตามสิทธิ์การจัดการของผู้ดูแลได้อย่างถูกต้อง
- 2) ผู้ใช้งานโปรแกรมสามารถกำหนดรุ่นและแพ็คเกจส่วนเสริมต่างๆ ให้กับเครื่องลูกข่ายได้ และรายการแพ็คเกจของแต่ละผลิตภัณฑ์ไม่ซ้ำซ้อนกัน
- 3) ผู้ใช้งานที่เป็นผู้ดูแลระบบสามารถเข้าไปเพิ่ม แก้ไข ลบข้อมูลต่างๆ ของผลิตภัณฑ์ได้ในส่วน Back Office และสามารถกำหนดสิทธิ์ให้กับผู้ใช้อื่นได้ และเมื่อเกิดข้อผิดพลาดระบบสามารถแสดงรายละเอียดแก่ผู้ใช้ได้ถูกต้อง
- 4) ผู้ใช้งานสามารถออกรายงานของเครื่องลูกข่ายได้ถูกต้องตามประเภทของผลิตภัณฑ์และหัวข้อที่ต้องการ และสามารถจัดกลุ่มของเครื่องลูกข่ายได้

5.2 สรุปผลการทดสอบระบบ

จากการนำระบบไปใช้งานจริงพบว่า ระบบเดสก์ท็อปดีฟลอยเม้นท์พอร์ทลมีความสามารถดังนี้

- 1) หน้าหลักของระบบ สามารถแสดงข้อมูลลูกค้าแยกตามผลิตภัณฑ์ได้ถูกต้อง
- 2) ระบบสามารถจำแนกสิทธิ์การใช้งานระบบของพนักงาน แยกตามผลิตภัณฑ์ได้อย่างถูกต้อง
- 3) พนักงานสามารถกำหนดเวอร์ชันและส่วนเสริมของผลิตภัณฑ์ FXT และ MCA ได้ เช่นเดียวกับผลิตภัณฑ์ Eikon โดยเครื่องที่ทดสอบได้รับเวอร์ชันผลิตภัณฑ์อย่างถูกต้อง
- 4) พนักงานที่เป็นผู้ดูแล สามารถเข้าใช้งานในส่วน Back office ได้ สามารถจัดการแก้ไขข้อมูลต่างๆ ได้ เช่น การให้สิทธิ์พนักงานคนอื่น, การแก้ไขแพ็คเกจของผลิตภัณฑ์นั้นๆ ได้, สามารถดูสถิติของผู้ใช้ได้ เป็นต้น
- 5) พนักงานสามารถสร้างกลุ่มเพื่อแยกกลุ่มลูกค้าได้ โดยกลุ่มที่สร้างจะไม่ส่งผลกระทบต่อกลุ่มของผลิตภัณฑ์อื่น
- 6) ผู้ใช้งานสามารถออกรายงานแยกตามผลิตภัณฑ์ในส่วนที่ต้องการได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 ข้อจำกัดในการพัฒนาระบบ

- 1) ระบบเมื่อทำการทดสอบบนเครื่องส่วนตัว สามารถทำงานได้อย่างรวดเร็ว แต่เมื่อนำขึ้นไปใช้งานจริงบนเครื่องเซิร์ฟเวอร์จะพบว่า ในการร้องขอแต่ละครั้งจะใช้เวลาานาน เพราะเครื่องเซิร์ฟเวอร์มีเซอริวิตีทำงานอยู่มาก ทำให้การตอบสนองต้องใช้เวลา และบางครั้งอาจหมดเวลาการเชื่อมต่อก่อน
- 2) ระบบนี้ถูกพัฒนาให้ใช้งานบนบราวเซอร์ Chrome เป็นหลัก ทำให้การแสดงผลบนบราวเซอร์อื่น อาจเกิดความผิดพลาดขึ้นได้

5.4 ข้อเสนอแนะในการพัฒนาระบบ

เนื่องจากระบบนี้เป็นระบบให้บริการอัปเดตและปล่อยส่วนเสริมให้แก่ผลิตภัณฑ์ของลูกค้า ในฐานะของผู้พัฒนาได้มีข้อเสนอแนะ ดังนี้

- 1) สามารถบอกความคืบหน้าในการดาวน์โหลดรายงาน อาจจะเป็นแถบสีหรือเป็นเปอร์เซ็นต์ เพื่อบอกให้ผู้ใช้ทราบ
- 2) สามารถจำแนกลูกค้าตามประเทศ หรือทวีปได้ เพื่อให้การให้บริการอัปเดตสามารถเจาะจงได้ละเอียดมากขึ้น
- 3) สามารถแจ้งเตือนลูกค้าผ่านทางอีเมลได้ว่า มีเวอร์ชันที่ปล่อยให้อัปเดตได้ ในกรณีที่ลูกค้าไม่ได้เข้ามาใช้งานผลิตภัณฑ์เป็นระยะเวลาานาน

เอกสารอ้างอิง

- [1] Once More With Feeling. 2013. **Web Service**. [Online]. Available :
<http://plainsofeternity.blogspot.com/2013/07/5-web-services.html>.
 เข้าถึงเมื่อวันที่ 29 มี.ค. 2559.
- [2] Erik Reitan. 2016. **Windows Communication Foundation**. [Online]. Available :
<https://msdn.microsoft.com/en-us/library/ms731082%28v=vs.110%29.aspx>.
 เข้าถึงเมื่อวันที่ 29 มี.ค. 2559.
- [3] RiderOOO. 2016. **Web Application**. [Online]. Available :
<http://aicomputer.co.th/sArticle/002-what-is-Web-Application.aspx>.
 เข้าถึงเมื่อวันที่ 29 มี.ค. 2559
- [4] Vangie Beal. 2015. **Application Programming Interface**. [Online]. Available :
<http://www.webopedia.com/TERM/A/API.html>. เข้าถึงเมื่อวันที่ 29 มี.ค. 2559
- [5] Bill Wagner. 2014. **C# Programming Guide**. [Online]. Available :
<https://msdn.microsoft.com/en-us/library/67ef8sbd.aspx>.
 เข้าถึงเมื่อวันที่ 11 มี.ค. 2559
- [6] W3Schools. 2010. **HTML(5) Tutorial**. [Online]. Available :
<http://www.w3schools.com/html>. เข้าถึงเมื่อวันที่ 4 มี.ค. 2559
- [7] Anonymous. 2012. **CSS Tutorial**. [Online]. Available :
<http://www.w3schools.com/css>. เข้าถึงเมื่อวันที่ 30 มี.ค. 2559
- [8] Anonymous. 2010. **JavaScript Tutorial**. [Online]. Available :
<http://www.w3schools.com/js>. เข้าถึงเมื่อวันที่ 4 มี.ค. 2559
- [9] Anonymous. 2012. **jQuery Tutorial**. [Online]. Available :
<http://www.w3schools.com/jquery>. เข้าถึงเมื่อวันที่ 4 มี.ค. 2559
- [10] Anonymous. 2012. **AJAX Introduction**. [Online]. Available :
https://www.w3schools.com/xml/ajax_intro.asp. เข้าถึงเมื่อวันที่ 5 มี.ค. 2559
- [11] Anonymous. 2011. **SQL Tutorial**. [Online]. Available :
<https://www.w3schools.com/SQL/default.asp>. เข้าถึงเมื่อวันที่ 4 มี.ค. 2559

- [12] ThaiCreate.Com Team. 2015. **Entity Framework**. [Online]. Available : <http://www.thaicreate.com/tutorial/entity-framework-introduction.html>. เข้าถึงเมื่อวันที่ 30 มี.ค. 2559
- [13] Anonymous. 2013. **writing features - gherkin language**. [Online]. Available : <http://docs.behat.org/en/v2.5/guides/1.gherkin.html>. เข้าถึงเมื่อวันที่ 17 พ.ค. 2559
- [14] Anonymous. 2011. **เริ่มต้นใช้งาน - Git ขั้นพื้นฐาน**. [Online]. Available : <https://git-scm.com/book/th/v1/เริ่มต้นใช้งาน-Git-ขั้นพื้นฐาน>. เข้าถึงเมื่อวันที่ 17 เม.ย. 2559
- [15] Lovisa Johanson. 2015. **RabbitMQ for beginners - What is RabbitMQ?**. [Online]. Available : <https://www.cloudamqp.com/blog/2015-05-18-part1-rabbitmq-for-beginners-what-is-rabbitmq.html>. เข้าถึงเมื่อวันที่ 17 เม.ย. 2559
- [16] Anonymous. 2014. **Unit Test Getting Start**. [Online]. Available : <http://www.specflow.org/getting-started>. เข้าถึงเมื่อวันที่ 16 พ.ค. 2559

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้