



รายงานสหกิจศึกษาระดับสมบูรณ

พัฒนาและดูแลระบบธนาคาร SCB
SCB Application Enhancement and Maintenance

นายวีรวัฒน์ บุญลอม

สาขาวิชาวิศวกรรมสารสนเทศ
ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2560



รายงานสหกิจศึกษาฉบับสมบูรณ์

พัฒนาและดูแลระบบธนาคาร SCB

SCB Application Enhancement and Maintenance

นายวีรวัฒน์ บุญล้อม

สาขาวิชาวิศวกรรมสารสนเทศ

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2560

ชื่อโครงการสหกิจศึกษา พัฒนาและดูแลระบบธนาคาร SCB

ชื่อ-สกุล นักศึกษา นาย วีรวัฒน์ บุญล้อม

คณะ วิศวกรรมศาสตร์ ภาควิชา วิศวกรรมคอมพิวเตอร์ สาขาวิชา วิศวกรรมสารสนเทศ

ชื่อ-สกุล อาจารย์นิเทศ รศ.ดร.ชวลิต เบญจางคประเสริฐ

ชื่อ-สกุล ผู้นิเทศงาน คุณ พิมพ์เงิน กมลศิริ

สถานประกอบการ บริษัท ไอพีเอ็ม โซลูชั่นส์ ดิลิเวอรี่

บทคัดย่อ

การศึกษาครั้งนี้มีวัตถุประสงค์เพื่อพัฒนาระบบ SCB Application ให้มีฟังก์ชันที่มากกว่าเดิม สามารถเชื่อมต่อกับบริษัทโอนเงินต่างประเทศ และช่วยสร้างกำไรให้กับ Siam Commercial Bank ซึ่งจะช่วยขยายธุรกิจด้านโอนเงินต่างประเทศให้กับธนาคาร โดยการเพิ่มผู้ให้บริการใหม่เพื่อเป็นทางเลือกในการโอนเงิน และเพิ่มผลิตภัณฑ์ใหม่ในการเปิดบัญชี เพื่อเพิ่มช่องทางต่าง ๆ ให้ลูกค้าได้เลือกมากขึ้น

ผู้วิจัยเลือกพัฒนาซอฟต์แวร์โดยใช้โปรแกรม Microsoft Visual Basic 6.0 และ Java เพราะจากซอฟต์แวร์เดิมใช้โปรแกรมนี้นี้นพัฒนาอยู่ ทำให้ผู้วิจัยต้องทำการศึกษาเพิ่มเติมเนื่องจากเป็นซอฟต์แวร์ที่เก่าแล้ว ส่วนระบบ ฐานข้อมูล ใช้ทั้ง DB2 และ Microsoft SQL Server ในการพัฒนา ส่วนการ Print Report จะใช้ Crystal Report แสดงผล การจัดการควบคุมซอร์ซโค้ดใช้เทคโนโลยีที่เรียกว่า Gitlab

ผลการศึกษาทำให้ผู้วิจัยสามารถออกแบบฐานข้อมูล รายงาน และหน้าจอแสดงผล ตามความเหมาะสม และตรงตามความต้องการของผู้ใช้งานได้ ซึ่งผลงานเป็นที่น่าพอใจ เพราะระบบเป็นไปตามความต้องการของผู้ใช้งาน และสามารถนำไปใช้จริงบนระบบได้

Research Title: SCB Application Enhancement and Maintenance

Student Intern Name: Mr. Wirawat Bunlom

Faculty: Engineering **Department:** Computer Engineering **Major:** Information Engineering

Advisor Name: Assoc.Prof.Dr. Chawalit Benjangkaprasert

Mentor Name: Ms.Pimnguen Kamolsiri

Company: IBM Solution Delivery Co., Ltd.

ABSTRACT

This research aims to develop SCB application system more efficient, can connect with other providers to exchange foreign currencies and help generate profits for Siam Commercial Bank. This will help expand the international money transfer business and add new products to open an account for additional channels to customers.

Researchers choose to develop software using Microsoft Visual Basic and Java because the original software developed in this language so researcher needs to study more because they are old software. The database system uses both DB2 and SQL. The print report uses a crystal report to display and handle the version control using Gitlab.

Research results can help researchers to design database report and GUI as appropriate and exactly user request. The work is satisfactory because the system works as requested by the user and can be used on the production system.

กิตติกรรมประกาศ

การศึกษาค้นคว้า และวิจัยฉบับนี้ สำเร็จได้ด้วยความอนุเคราะห์จากคุณพิมพ์เงิน กมลศิริ ผู้จัดการโครงการ คุณไพลิน คุณกฤตธี รวมถึงพี่ทุกคน ในบริษัทไอบีเอ็ม โซลูชันส์ ดิลิเวอรี จำกัด และธนาคารไทยพาณิชย์ จำกัด (มหาชน) ผู้กรุณา ให้คำปรึกษา คำแนะนำ และความช่วยเหลือตลอดระยะเวลาการทำงาน จนทำให้งานวิจัยชิ้นนี้เสร็จสมบูรณ์ ผู้วิจัยจึงขอขอบพระคุณเป็นอย่างสูงไว้ ณ โอกาสนี้

ขอขอบคุณรศ.ดร.ชวลิต เบญจางคประเสริฐ อาจารย์นิเทศ ผู้กรุณาคอยแนะนำการเตรียมตัว ให้คำปรึกษา และ แนวทางการแก้ปัญหาในการทำงานวิจัยภายใต้โครงการสหกิจศึกษา ทำให้งานวิจัยนี้เป็นไปอย่างราบรื่น

ท้ายที่สุด ขอขอบพระคุณคณาจารย์คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ได้ประสิทธิ์ประสาทวิชาความรู้ ทำให้ผู้วิจัยสามารถนำสิ่งที่ได้รับจากการศึกษานั้นมาใช้ในการศึกษา ค้นคว้า และวิจัยครั้งนี้

วีรวัฒน์ บุญล้อม

สารบัญ

เรื่อง	หน้า
บทคัดย่อ	I
ABSTRACT	II
กิตติกรรมประกาศ	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญรูป.....	VII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญ	1
1.2 วัตถุประสงค์ของการวิจัย	2
1.3 ขอบเขตของการวิจัย	2
1.4 วิธีดำเนินการวิจัย.....	5
1.5 ประโยชน์ที่คาดว่าจะได้รับ	8
บทที่ 2 แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง.....	10
2.1 ทฤษฎีในด้านซอฟต์แวร์คอมพิวเตอร์	10
2.2 ทฤษฎีในด้านธุรกิจ.....	21
บทที่ 3 วิธีดำเนินการวิจัย	24
3.1 โปรเจกต์ RIA	24
3.2 โปรเจกต์ SingleApp.....	27
3.3 โปรเจกต์ Bulk Payroll	31
3.4 โปรเจกต์ Bulk Open Account	33

สารบัญ (ต่อ)

เรื่อง	หน้า
3.5 โปรเจกต์ Bulk Print.....	36
บทที่ 4 ผลการวิจัย	38
4.1 การพัฒนาโปรแกรมให้สอดคล้องกับความต้องการของผู้ใช้	38
4.2 การแก้ปัญหาโปรแกรมเมื่อผู้ใช้งานแจ้งปัญหาเข้ามา	50
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ	52
5.1 สรุปผลการวิจัย	52
5.2 ปัญหาที่พบในระหว่างการดำเนินงานวิจัย.....	52
5.3 แนวทางการแก้ไขปัญหา	53
เอกสารอ้างอิง	54

สารบัญตาราง

ตารางที่	หน้า
ตารางที่ 1.1 แผนการดำเนินการวิจัย.....	5
ตารางที่ 3.1 แผนการพัฒนาโปรเจกต์ RIA.....	25
ตารางที่ 3.2 Business Scenario ของ RIA	26
ตารางที่ 3.3 แผนการพัฒนาโปรเจกต์ SingleApp.....	28
ตารางที่ 3.4 แผนการพัฒนาโปรเจกต์ Bulk Payroll	31
ตารางที่ 3.5 แผนการพัฒนาโปรเจกต์ Bulk Open Account	33
ตารางที่ 3.6 แผนการพัฒนาโปรเจกต์ Bulk Print.....	36

สารบัญรูป

รูปที่	หน้า
รูปที่ 2.1 โลโก้โปรแกรม Visual Basic 6.0	10
รูปที่ 2.2 โลโก้โปรแกรม NetBeans.....	11
รูปที่ 2.3 ตัวอย่างโปรแกรม Crystal Report.....	14
รูปที่ 2.4 ตัวอย่างโปรแกรม GIT ที่ใช้สำหรับทำ Version Control	15
รูปที่ 2.5 ตัวอย่าง Glassfish Server.....	15
รูปที่ 2.6 โลโก้โปรแกรม Microsoft Access 2016	16
รูปที่ 2.7 ตัวอย่างโปรแกรม Microsoft Access 2016	17
รูปที่ 2.8 ตัวอย่างโปรแกรม HeidiSQL	18
รูปที่ 2.9 โลโก้ Microsoft SQL Server.....	18
รูปที่ 2.10 โลโก้ IBM DB2	20
รูปที่ 2.11 ตัวอย่างโปรแกรม Toad for DB2.....	20
รูปที่ 2.12 ตัวอย่างโปรแกรม SoapUI.....	21
รูปที่ 2.13 กระบวนการเปิดบัญชีของธนาคาร	21
รูปที่ 2.14 ตัวอย่างกระบวนการโอนเงินต่างประเทศ ขาเข้า.....	23
รูปที่ 2.15 ตัวอย่างกระบวนการโอนเงินต่างประเทศ ขาออก	23
รูปที่ 3.1 ตัวอย่างเอกสาร Requirement List ของโปรเจกต์ RIA	24
รูปที่ 3.2 กระบวนการทำงานของ Bulk Open Account.....	34
รูปที่ 4.1 ตัวอย่างหน้าจอ RIA Receive	38
รูปที่ 4.2 ตัวอย่างใบเสร็จจ่ายเงินของ RIA.....	39
รูปที่ 4.3 ตัวอย่าง Summary Report	39
รูปที่ 4.4 ตัวอย่าง โปรแกรม SingleApp เมื่อเพิ่มผลิตภัณฑ์ อีซี เซฟวิง-ไม่มีสมุด เข้าไปใน Database.....	40
รูปที่ 4.5 ฟังก์ชันตรวจสอบสถานะ Easy App และ Easy Net เมื่อเลือกผลิตภัณฑ์อีซี-เซฟวิง.....	41
รูปที่ 4.6 ตัวอย่างเงื่อนไขของการเปิดผลิตภัณฑ์อีซี เซฟวิง-ไม่มีสมุด.....	41
รูปที่ 4.7 ฟังก์ชัน Preprinted Form เพิ่มผลิตภัณฑ์เงินฝากพร้อมใช้ หรือ SCB Feel Free Savings.....	42

สารบัญรูป (ต่อ)

รูปที่	หน้า
รูปที่ 4.8 ตัวอย่างผลิตภัณฑ์เงินฝากพร้อมใช้ หรือ SCB Feel Free Savings.....	42
รูปที่ 4.9 ตัวอย่างหน้าจอแสดงผลเมื่อเลือกผลิตภัณฑ์เงินฝากพร้อมใช้ หรือ SCB Feel Free Savings	43
รูปที่ 4.10 เงื่อนไขไม่อนุญาตให้เปิดบัญชีร่วมของผลิตภัณฑ์เงินฝากพร้อมใช้ หรือ SCB Feel Free Savings.	43
รูปที่ 4.11 ตัวอย่างเงื่อนไขในการเปิดผลิตภัณฑ์ใหม่ของโปรแกรม Bulk Payroll	44
รูปที่ 4.12 ตัวอย่างเงื่อนไขผลิตภัณฑ์เงินฝากพร้อมใช้เมื่อทำการเปิดบัญชีถูกต้อง	45
รูปที่ 4.13 ตัวอย่าง โปรแกรม Bulk Detail Input.....	45
รูปที่ 4.14 ตัวอย่างโปรแกรม Bulk Detail Capture ที่ทำการเพิ่มผลิตภัณฑ์แล้ว	46
รูปที่ 4.15 ตัวอย่างผลิตภัณฑ์ เงินฝาก SCB Feel Free ของโปรแกรม Bulk CreateControl	47
รูปที่ 4.16 ตัวอย่างการเพิ่มผลิตภัณฑ์ SCB Feel Free Saving เข้าไปในฟังก์ชัน Print Passbook	47
รูปที่ 4.17 ตัวอย่าง การเพิ่มผลิตภัณฑ์ SCB Feel Free Saving เข้าไปในฟังก์ชัน Print Signature Card.....	48
รูปที่ 4.18 ตัวอย่างฟังก์ชัน Print Passbook ของผลิตภัณฑ์ SCB Feel Free Savings.....	48
รูปที่ 4.19 ตัวอย่างฟังก์ชัน Print Signature Card ของผลิตภัณฑ์ SCB Feel Free Savings	48
รูปที่ 4.20 ตัวอย่างเอกสาร Unit Test.....	49
รูปที่ 4.21 เอกสารขอเปิดการทดสอบระบบบน UAT และใช้จริงบน Production หน้า Implement	50
รูปที่ 4.22 เอกสารขอเปิดการทดสอบระบบบน UAT และใช้จริงบน Production หน้า Roll Back.....	50

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญ

บริษัท ไอปีเอ็ม โซลูชั่นส์ ดิจิเวอร์ จำกัด ก่อตั้งมาตั้งแต่ปี 2528 โดยแรกเริ่มใช้ชื่อบริษัท ไทย ซิสเต็ม คอร์ปอเรชั่น จำกัด ด้วยพนักงานเพียง 50 คน โดยมีจุดมุ่งหมายในการให้บริการลูกค้าทางด้านเอ็ดซอร์สซึ่งที่ได้มาตรฐานครบวงจร ช่วยธุรกิจลดค่าใช้จ่ายในการดำเนินงานด้านไอที ในขณะเดียวกัน ก็ช่วยเพิ่มประสิทธิภาพการทำงาน ผลิตผล รวมทั้งความสามารถในการแข่งขันทางธุรกิจของลูกค้า เมื่อเทียบกับการที่ลูกค้าต้องดำเนินการต่าง ๆ ดังกล่าวเอง ต่อมาบริษัทฯ ได้เปลี่ยนชื่อเป็นบริษัท ไอปีเอ็ม โซลูชั่นส์ ดิจิเวอร์ จำกัด ในปี 2545 และเติบโตมาอย่างรวดเร็ว จนมีพนักงานกว่า 1,000 คน ในปัจจุบัน ให้บริการทางด้านเอ็ดซอร์สซึ่งหลากหลายประเภทแก่ลูกค้าในหลากหลายกลุ่มธุรกิจ ไม่ว่าจะเป็นธุรกิจธนาคาร อุบัติเหตุ บริโภค ค่าปลีก เวชภัณฑ์ ขนส่ง เดินทาง เช่น บริษัทในเครือธนาคารกสิกรไทย จำกัด (มหาชน) บริษัทในเครือไมเนอร์ กรุ๊ป บริษัท บิวตี้ รีเทล จำกัด สายการบินสิงคโปร์ แอร์ไลน์ส ธนาคารธนชาติ จำกัด (มหาชน) ธนาคารทีสโก้ จำกัด (มหาชน) ธนาคารไทยพาณิชย์ เป็นต้น

การเข้าร่วมโครงการสหกิจศึกษากับทางบริษัท ไอปีเอ็ม โซลูชั่นส์ ดิจิเวอร์ จำกัด ในด้านการพัฒนาโปรแกรม (Software Developer) ต้องทำงานด้านบริการเอ็ดซอร์สซึ่งที่ธนาคารไทยพาณิชย์ ในแผนกจัดการความเสี่ยงและสนับสนุน (Risk & Support) ซึ่งจะเป็นแผนกที่ดูแลโปรแกรมที่เกี่ยวกับการเปิดบัญชี และการโอนเงินต่างประเทศ โดยมีหน้าที่พัฒนาโปรแกรมให้สอดคล้องกับความต้องการของผู้ใช้ และแก้ปัญหาโปรแกรมเมื่อผู้ใช้แจ้งปัญหาเข้ามา ทำให้งานวิจัยชิ้นนี้ แบ่งออกเป็น 2 ส่วน คือ

1.1.1 การพัฒนาโปรแกรมให้สอดคล้องกับความต้องการของผู้ใช้

โปรแกรมที่เกี่ยวกับการเปิดบัญชีที่แผนกต้องดูแลมีทั้งหมด 4 โปรแกรม นั่นคือ SingleApp, Bulk Payroll, Bulk Open Account และ Bulk Print ซึ่งถ้ามีการเพิ่มผลิตภัณฑ์ใหม่ ทำให้มีโอกาสที่จะต้องแก้ทั้งหมด 4 โปรแกรม เพื่อให้ทำงานสอดคล้องกันได้อย่างมีประสิทธิภาพ

โปรแกรมที่เกี่ยวกับการโอนเงินต่างประเทศที่ได้รับมอบหมายมี 1 โปรแกรม คือ FEB (Foreign Exchange Booth) มีการพัฒนาเพิ่มฟังก์ชัน และเพิ่ม Provider ในการให้บริการ ทำให้ต้องแก้รายงานที่จะต้องพิมพ์ออกมารายวัน รายสัปดาห์ และรายเดือน

1.1.2 การแก้ปัญหาโปรแกรมเมื่อผู้ใช้แจ้งปัญหาเข้ามา

ปัญหาที่เกิดจากโปรแกรมนั้น อาจเกิดได้ทั้งจากการที่ซอร์ซโค้ด (Source Code) ผิดพลาด และเกิดจากผู้ใช้

1.2 วัตถุประสงค์ของการวิจัย

1.2.1 ออกแบบฐานข้อมูลด้วย Microsoft SQL Server และ DB2 เพื่อให้โปรแกรมทำงานได้ตรงตามความต้องการของผู้ใช้งาน

1.2.2 ศึกษาการเขียนโปรแกรมด้วย Microsoft Visual Basic 6.0 และการทำ Report ด้วย Crystal Report เพื่อเพิ่มประสิทธิภาพในการทำงานของโปรแกรมโอนเงินต่างประเทศ

1.2.3 ศึกษาการทำงานร่วมกันของ Front-End, Back-End และ Web Service เพื่อใช้ในการพัฒนาการทำงานร่วมกันของระบบต่าง ๆ ในธนาคาร

1.2.4 ศึกษาการเขียน Web Application จาก Java, JSP, JavaScript และ JSON เพื่อเพิ่มประสิทธิภาพในการทำงานของระบบเปิดบัญชีธนาคาร และเพื่อเพิ่มตัวเลือกผลิตภัณฑ์ให้ลูกค้าได้เลือกใช้มากขึ้น

1.3 ขอบเขตของการวิจัย

1.3.1 ความสามารถของแอปพลิเคชัน

1.3.1.1 Project RIA

- สร้างหน้าจอสำหรับบริการโอนเงินผ่าน New Provider
- เชื่อมต่อระบบเพื่อ Settlement ระหว่าง SCB กับ Provider
- เชื่อมต่อระบบเพื่อดำเนินธุรกิจเงินโอนต่างประเทศ ทั้งขารับ และขาส่งโดยผ่านช่องทางต่าง ๆ ดังนี้

- ลูกค้ารับเงินสดผ่านเคาน์เตอร์สาขา SCB และผ่านบูธแลกเปลี่ยนฯ SCB
- ลูกค้ารับเงินผ่านบัญชี SCB
- ลูกค้ารับเงินผ่านตู้ ATM SCB
- ลูกค้าส่งเงินผ่านเคาน์เตอร์สาขา SCB และผ่านบูธแลกเปลี่ยนฯ SCB
- ลูกค้าส่งเงินผ่านระบบ SCB Easy Net

1.3.1.2 Project SingleApp

- สร้างผลิตภัณฑ์บัญชีใหม่เพื่อเป็นทางเลือกให้ลูกค้าได้เปิดใช้บริการ ดังนี้
 - EasySaving Account
 - SCB Feel Free Savings Account
 - Education Saving
- ผลิตภัณฑ์ต่าง ๆ จะมีเงื่อนไขในการเปิดบัญชีแตกต่างกันไป
- เปลี่ยน Endpoint และ Port เพื่อรองรับการเชื่อมต่อ Web Service ใหม่ของ ระบบ EasyApp
- เปลี่ยน Term & Condition ของผลิตภัณฑ์เพื่อให้รองรับกับกฎหมายในปัจจุบันตาม requirement
- แก้ปัญหาของ Web Application นี้ เมื่อมีลูกค้าติดต่อส่งเรื่องปัญหาที่เกิดขึ้น
- สร้าง Library ของ Web Application ให้สอดคล้องกับระบบ Salesforce เนื่องจาก SingleApp ทำงานบนระบบ Salesforce

1.3.1.3 Project Bulk Payroll

- สร้างผลิตภัณฑ์บัญชีใหม่เพื่อเป็นทางเลือกให้ลูกค้าได้เปิดใช้บริการ ดังนี้
 - SCB Feel Free Savings Account
 - SCB Feel Free Debit Card

- ผลิตภัณฑ์ต่าง ๆ จะมีเงื่อนไขในการเปิดบัญชีแตกต่างกันไป
- สามารถพิมพ์หน้า First Page ผลิตภัณฑ์ใหม่ให้กับลูกค้าได้

1.3.1.4 Project Bulk Open Account

- สร้างผลิตภัณฑ์บัญชีใหม่เพื่อเป็นทางเลือกให้ลูกค้าได้เปิดใช้บริการ และรองรับการทำงานกับระบบ Core Bank ดังนี้

- SCB Feel Free Savings Account
- SCB Feel Free Debit Card

- ผลิตภัณฑ์ต่าง ๆ จะมีเงื่อนไขในการเปิดบัญชีแตกต่างกันไป
- ออกแบบและพัฒนาให้โปรแกรม Bulk Detail Input Bulk Detail Capture และ Bulk CreateControl ทำงานร่วมกันได้

1.3.1.5 Project Bulk Print

- สร้างฟังก์ชันการพิมพ์สมุดบัญชีและการ์ดลายเซ็นเพื่อรองรับผลิตภัณฑ์ใหม่ ดังนี้

- SCB Feel Free Savings Account
- SCB Feel Free Debit Card

- ผลิตภัณฑ์ต่าง ๆ จะมีเงื่อนไขในการเปิดบัญชีแตกต่างกันไป

1.3.2 ระยะเวลาวิจัย

ใช้เวลาทั้งหมด 16 สัปดาห์

งาน	ส.ค.				ก.ย.				ต.ค.				พ.ย.			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
RIA					←————→											
SingleApp	←————→															
Bulk Payroll	←————→															
Bulk Open Account	←————→								←————→							
Bulk Print	←————→															

ตารางที่ 1.1 แผนการดำเนินการวิจัย

1.4 วิธีดำเนินการวิจัย

ขั้นตอนการดำเนินงานวิจัยที่ได้วางแผนไว้ ได้ถูกแบ่งตามขอบเขตของงานวิจัยได้ ดังนี้

1.4.1 การพัฒนาโปรเจกต์ RIA

1.4.1.1 การวิเคราะห์เอกสารความต้องการของลูกค้าที่ได้รับมา

1.4.1.2 ตรวจสอบผลกระทบของโปรแกรมว่าส่วนที่ต้องพัฒนา กระทบกับส่วนอื่นของโปรแกรมหรือไม่

1.4.1.3 ศึกษากระบวนการทางธุรกิจที่เกี่ยวข้องกับความต้องการของลูกค้า

1.4.1.4 ออกแบบระบบฐานข้อมูลเพื่อเก็บข้อมูลของลูกค้าและดึงมาใช้งาน

1.4.1.5 ออกแบบหน้าจอ ใบบเสร็จ และฟังก์ชันให้ตรงตามที่ผู้ใช้งานต้องการ

1.4.1.6 ประสานงานกับแผนกที่เกี่ยวข้องเพื่อพัฒนาโปรแกรม ตลอดจนโปรแกรมพร้อมใช้งาน

1.4.1.7 ทดสอบโปรแกรมเพื่อหาจุดบกพร่อง และแก้ไข

1.4.1.8 ประสานงานกับผู้ทดสอบโปรแกรม (Tester) ของทีม SIT เพื่อสาธิตวิธีการใช้งานโปรแกรม และให้ผู้ทดสอบโปรแกรมทำการทดสอบการใช้โปรแกรม

1.4.1.9 ประสานงานกับผู้ทดสอบโปรแกรม (Tester) ของทีม UAT ให้ผู้ทดสอบโปรแกรมทำการทดสอบการใช้โปรแกรมขั้นต้นสุดท้ายเพื่อให้แน่ใจว่า ระบบที่พัฒนาพร้อมที่จะใช้งานได้จริง ตรงตามกระบวนการทางธุรกิจ (Business Process) และความต้องการของผู้ใช้งาน

1.4.1.10 โปรแกรมสามารถนำมาใช้งานจริง

1.4.2 การพัฒนาโปรเจกต์ SingleApp

1.4.2.1 การวิเคราะห์เอกสารความต้องการของลูกค้าที่ได้รับมา

1.4.2.2 ตรวจสอบผลกระทบของโปรแกรมว่าส่วนที่ต้องพัฒนา กระทับกับส่วนอื่นของโปรแกรมหรือไม่

1.4.2.3 ศึกษากระบวนการทางธุรกิจที่เกี่ยวข้องกับความต้องการของลูกค้า

1.4.2.4 เพิ่มผลิตภัณฑ์ และฟังก์ชันให้ตรงตามผู้ใช้ที่ต้องการ

1.4.2.5 ประสานงานกับแผนกที่เกี่ยวข้องเพื่อพัฒนาโปรแกรม ตลอดจนโปรแกรมพร้อมใช้งาน

1.4.2.6 ทดสอบโปรแกรมเพื่อหาจุดบกพร่อง และแก้ไข

1.4.2.7 ประสานงานกับผู้ทดสอบโปรแกรม (Tester) ของทีม SIT เพื่อสาธิตวิธีการใช้งานโปรแกรม และให้ผู้ทดสอบโปรแกรมทำการทดสอบการใช้โปรแกรม

1.4.2.8 ประสานงานกับผู้ทดสอบโปรแกรม (Tester) ของทีม UAT ให้ผู้ทดสอบโปรแกรมทำการทดสอบการใช้โปรแกรมขั้นต้นสุดท้ายเพื่อให้แน่ใจว่า ระบบที่พัฒนาพร้อมที่จะใช้งานได้จริง ตรงตามกระบวนการทางธุรกิจ (Business Process) และความต้องการของผู้ใช้งาน

1.4.2.9 โปรแกรมสามารถนำมาใช้งานจริง

1.4.3 การพัฒนาโปรเจกต์ BulkPayroll

1.4.3.1 การวิเคราะห์เอกสารความต้องการของลูกค้าที่ได้รับมา

1.4.3.2 ตรวจสอบผลกระทบของโปรแกรมว่าส่วนที่ต้องพัฒนา กระทับกับส่วนอื่นของโปรแกรมหรือไม่

- 1.4.3.3 ศึกษากระบวนการทางธุรกิจที่เกี่ยวข้องกับความต้องการของลูกค้า
- 1.4.3.4 เพิ่มผลิตภัณฑ์ และฟังก์ชันให้ตรงตามที่ต้องการ
- 1.4.3.5 ประสานงานกับแผนกที่เกี่ยวข้องเพื่อพัฒนาโปรแกรม ตลอดจนโปรแกรมพร้อมใช้งาน
- 1.4.3.6 ทดสอบโปรแกรมเพื่อหาจุดบกพร่อง และแก้ไข
- 1.4.3.7 ประสานงานกับผู้ทดสอบโปรแกรม (Tester) ของทีม SIT เพื่อสาธิตวิธีการใช้งานโปรแกรม และให้ผู้ทดสอบโปรแกรมทำการทดสอบการใช้โปรแกรม
- 1.4.3.8 ประสานงานกับผู้ทดสอบโปรแกรม (Tester) ของทีม UAT ให้ผู้ทดสอบโปรแกรมทำการทดสอบการใช้โปรแกรมขั้นตอนสุดท้ายเพื่อให้แน่ใจว่า ระบบที่พัฒนาพร้อมที่จะใช้งานได้จริง ตรงตามกระบวนการทางธุรกิจ (Business Process) และความต้องการของผู้ใช้งาน
- 1.4.3.9 โปรแกรมสามารถนำมาใช้งานจริง
- 1.4.4 การพัฒนาโปรเจกต์ Bulk Open Account
 - 1.4.4.1 การวิเคราะห์เอกสารความต้องการของลูกค้าที่ได้รับมา
 - 1.4.4.2 ตรวจสอบผลกระทบของโปรแกรมว่าส่วนที่ต้องพัฒนา กระทับกับส่วนอื่นของโปรแกรมหรือไม่
 - 1.4.4.3 ศึกษากระบวนการทางธุรกิจที่เกี่ยวข้องกับความต้องการของลูกค้า
 - 1.4.4.4 เพิ่มผลิตภัณฑ์ และฟังก์ชันให้ตรงตามที่ต้องการ
 - 1.4.4.5 ประสานงานกับแผนกที่เกี่ยวข้องเพื่อพัฒนาโปรแกรม ตลอดจนโปรแกรมพร้อมใช้งาน
 - 1.4.4.6 ทดสอบโปรแกรมเพื่อหาจุดบกพร่อง และแก้ไข
 - 1.4.4.7 ประสานงานกับผู้ทดสอบโปรแกรม (Tester) ของทีม SIT เพื่อสาธิตวิธีการใช้งานโปรแกรม และให้ผู้ทดสอบโปรแกรมทำการทดสอบการใช้โปรแกรม

1.4.4.8 ประสานงานกับผู้ทดสอบโปรแกรม (Tester) ของทีม UAT ให้ผู้ทดสอบโปรแกรมทำการทดสอบการใช้โปรแกรมขั้นสุดท้ายเพื่อให้แน่ใจว่า ระบบที่พัฒนาพร้อมที่จะใช้งานได้จริง ตรงตามกระบวนการทางธุรกิจ (Business Process) และความต้องการของผู้ใช้งาน

1.4.4.9 โปรแกรมสามารถนำมาใช้งานจริง

1.4.5 การพัฒนาโปรเจกต์ Bulk Print

1.4.5.1 การวิเคราะห์เอกสารความต้องการของลูกค้าที่ได้รับมา

1.4.5.2 ตรวจสอบผลกระทบของโปรแกรมว่าส่วนที่ต้องพัฒนา กระทับกับส่วนอื่นของโปรแกรมหรือไม่

1.4.5.3 ศึกษากระบวนการทางธุรกิจที่เกี่ยวข้องกับความต้องการของลูกค้า

1.4.5.4 เพิ่มผลิตภัณฑ์ และฟังก์ชันให้ตรงตามผู้ใช้ที่ต้องการ

1.4.5.5 ประสานงานกับแผนกที่เกี่ยวข้องเพื่อพัฒนาโปรแกรม ตลอดจนโปรแกรมพร้อมใช้งาน

1.4.5.6 ทดสอบโปรแกรมเพื่อหาจุดบกพร่อง และแก้ไข

1.4.5.7 ประสานงานกับผู้ทดสอบโปรแกรม (Tester) ของทีม SIT เพื่อสาธิตวิธีการใช้งานโปรแกรม และให้ผู้ทดสอบโปรแกรมทำการทดสอบการใช้โปรแกรม

1.4.5.8 ประสานงานกับผู้ทดสอบโปรแกรม (Tester) ของทีม UAT ให้ผู้ทดสอบโปรแกรมทำการทดสอบการใช้โปรแกรมขั้นสุดท้ายเพื่อให้แน่ใจว่า ระบบที่พัฒนาพร้อมที่จะใช้งานได้จริง ตรงตามกระบวนการทางธุรกิจ (Business Process) และความต้องการของผู้ใช้งาน

1.4.5.9 โปรแกรมสามารถนำมาใช้งานจริง

1.5 ประโยชน์ที่คาดว่าจะได้รับ

ประโยชน์ที่คาดว่าจะได้รับจากการเข้าร่วมโครงการสหกิจศึกษากับ ทางบริษัท ไอบีเอ็ม โซลูชันส์ ดิลิเวอรี จำกัด จำแนกออกเป็น 2 ส่วน คือ

1.5.1 ประโยชน์ต่อบริษัท

1.5.1.1 ลดการใช้ทรัพยากรบุคคล

1.5.1.2 เพิ่มประสิทธิภาพการทำงานในด้านธุรกิจ

1.5.1.3 ลดเวลาที่ใช้ในการรอคอยในทางธุรกิจ

1.5.1.4 ลดเวลาที่ใช้ในการทำงานในส่วนของเทคโนโลยี

1.5.1.5 ลดความผิดพลาดที่เกิดจากการดำเนินงานของมนุษย์

1.5.1.6 ลดขั้นตอนการดำเนินงาน

1.5.2 ประโยชน์ต่อผู้วิจัย

1.5.2.1 ได้ศึกษาโปรแกรม Visual Basic 6.0 และการใช้ภาษา Java ในด้านต่าง ๆ เพื่อพัฒนาโปรแกรม

1.5.2.2 ได้ทำงานหลัก ทำงานจริงในบริษัท

1.5.2.3 เรียนรู้เกี่ยวกับวัฒนธรรม และสภาพแวดล้อมในการทำงาน

1.5.2.4 เรียนรู้การสื่อสารกับคนจากหลายแผนก และหลายประเทศ ทำให้ทักษะการสื่อสารภาษาอังกฤษพัฒนาขึ้น

1.5.2.5 เรียนรู้และรับมือเมื่อเจออุปสรรค และปัญหาต่าง ๆ

1.5.2.6 สามารถเรียงลำดับความสำคัญในการทำงาน ว่างานไหนควรทำก่อนทำหลัง

1.5.2.7 มีระเบียบวินัยในการทำงานดีขึ้น

1.5.2.8 สามารถนำความรู้และทักษะต่าง ๆ ที่ได้รับมา ปรับใช้กับการทำงานในอนาคตได้

บทที่ 2

แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

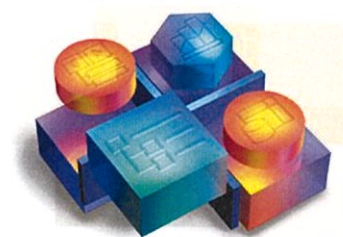
เนื่องจากงานวิจัยชิ้นนี้เกี่ยวข้องกับการเพิ่มประสิทธิภาพ และแก้ไขปัญหาทางธุรกิจ ดังนั้นผู้วิจัยจึงต้องศึกษาทฤษฎีต่าง ๆ ทั้งในด้านกระบวนการทางธุรกิจ หลักการในทางบัญชี กฎหมายด้านการเงิน รวมไปถึง หลักการทำงานของโปรแกรม และวิธีการใช้โปรแกรม เพื่อที่จะนำความรู้เหล่านี้มาประกอบรวมกันเพื่อให้ งานวิจัยสำเร็จลุล่วงไปได้ด้วยดี โดยแนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้องกับงานวิจัยชิ้นนี้ มีดังต่อไปนี้

2.1 ทฤษฎีในด้านซอฟต์แวร์คอมพิวเตอร์

2.1.1 โปรแกรม Visual Basic 6.0 enterprise edition

Visual Basic เป็นภาษารุ่นที่สามในการเขียนโปรแกรมแบบ event-driven programming (การเขียนโปรแกรมที่ขึ้นกับเหตุการณ์) ซึ่งมาพร้อมกับเครื่องมือพัฒนาจาก Microsoft เปิดตัวครั้งแรกในปี 1991 และได้รับการพัฒนาให้ดีขึ้นจนถึงปี 2008 โดย ภาษา Visual Basic นั้นถูกออกแบบมาเพื่อให้ง่ายต่อการเรียนรู้และง่ายต่อการใช้งาน

ภาษา Visual Basic นั้นถูกพัฒนามาจากภาษา Basic ภาษาเขียนโปรแกรมที่เข้าใจง่ายสำหรับผู้เริ่มต้น ซึ่งสนับสนุนการพัฒนาโปรแกรมแบบ rapid application development (RAD) และ graphical user interface (GUI) การเข้าถึงฐานข้อมูล และอื่น ๆ ที่ทำงานภายใต้ .NET Framework เวอร์ชันล่าสุดของ Visual Basic นั้นสนับสนุนการเขียนโปรแกรมเชิงวัตถุอย่างเต็มรูปแบบ



Microsoft
Visual Basic 6.0

รูปที่ 2.1 โลโก้โปรแกรม Visual Basic 6.0

2.1.2 โปรแกรม Netbeans

NetBeans คือ เครื่องมือสำหรับโปรแกรมเมอร์ที่จะใช้พัฒนา Application ด้วยภาษา Java NetBeans นั้นเป็นโปรแกรมประเภท OpenSource software โดยผู้ใช้งานไม่จำเป็นต้องเสียเงิน เพื่อซื้อ มาใช้งาน และยังเปิดเผย Source code ให้ผู้สนใจและนักพัฒนานำไปดัดแปลง แก้ไข ตามกฎของ OpenSource โดยมี Sun Micro System เป็นผู้สนับสนุนโครงการ

ปัจจุบัน NetBeans ได้รับความนิยมมากขึ้น และได้รับการพัฒนาให้มีความสามารถสูงยิ่งขึ้นเรื่อย ๆ จนถึงเวอร์ชันล่าสุด คือ นอกจากจะใช้ในการพัฒนาแอปพลิเคชันด้วยภาษา Java แล้ว ยังสามารถพัฒนาด้วย ภาษาอื่น ๆ ได้อีกหลากหลายโดยติดตั้งโปรแกรมเสริม(Add-on) ได้จาก เว็บไซต์ หรือผ่านตัวอัปเดตเซนเตอร์ (Update Center) ของ NetBeans เช่น ภาษาซี/ซีพลัสพลัส(C/C++) Ruby UML SOA Web Application Java EE Mobility(Java ME) Java Script PHP เป็นต้น

ข้อดีของโปรแกรมนี้คือ โปรแกรม NetBeans นั้นทำงานแยกส่วนต่าง ๆ ออกจากกันเป็น Module จึงทำให้สามารถนำ Module ต่าง ๆ ที่มีผู้ที่ได้พัฒนาต่อเติมมาติดตั้งเพิ่มเติมในภายหลังได้ ใช้งานได้กับ ระบบปฏิบัติการ Window Linux Mac OS X และ Solaris



รูปที่ 2.2 โลโก้โปรแกรม NetBeans

ซึ่งในงานวิจัยนี้ใช้ภาษา Java ในการพัฒนา PC Application และ ใช้ ภาษา Java JavaServer Page และ JavaScript ในการพัฒนา Web Application

2.1.2.1 Java หรือ Java Programming Language เป็นโปรแกรมภาษาที่ถูกพัฒนามาเพื่อรองรับการออกแบบซอฟต์แวร์ที่มีการเชื่อมโยง Internet อีกทั้งยังเป็นโปรแกรมที่สนับสนุนแนวความคิดของการเขียนโปรแกรมเชิงวัตถุ หรือที่รู้จักกันดีที่เรียกว่า OOP (Object-Oriented Programming) โปรแกรมที่เขียนขึ้นถูกสร้างภายในคลาส ดังนั้นคลาสคือที่เก็บเมทอด (Method) หรือพฤติกรรม (Behavior) ซึ่งมีสถานะ

(State) และรูปพรรณ (Identity) ประจำพฤติกรรม (Behavior) โดยมีความสามารถเฉพาะตัวต่างจากโปรแกรมภาษาชั้นสูงอื่น ๆ เช่น C หรือ C++ ในเรื่องของการทำงานข้ามระบบปฏิบัติการ หรือ Platform ได้โดยไม่ต้องมีการ compile ใหม่ ซึ่งโปรแกรมที่ถูกพัฒนาด้วยภาษา Java ถูกแบ่งเป็นสองประเภทหลัก ๆ คือ

1. Java Application – โปรแกรม Java ทั่ว ๆ ไป ที่ทำงานได้ด้วยตัวของมันเอง (Stand Alone Application)
2. Java Applet – โปรแกรม Java ที่ถูกนำไปใช้บน Internet เท่านั้น

2.1.2.2 JavaServer Page เป็นเทคโนโลยีไว้สำหรับพัฒนา Application บนเว็บไซต์ ในรูปแบบของ Server และ Client แสดงผลและโต้ตอบกับ User Interface ผ่าน Web Browser อย่างพวก IE Chrome Firefox และอื่น ๆ โดยอาศัยการทำงานร่วมกับพวก Client Tags เช่นพวก HTML / JavaScript / CSS และพวก jQuery ให้ได้ผลลัพธ์ตามที่ต้องการ ซึ่ง JSP ก็เหมือนกับโปรแกรม ASP PHP และ .Net รูปแบบการทำงานนั้นไม่แตกต่างกัน แต่แตกต่างกันตรงที่ JSP เป็น Subset ของภาษา Java โดยรูปแบบการเขียนนั้นจะใช้รูปแบบคำสั่งและชุด SDK ของ Java และใน JSP จะมีนามสกุลของไฟล์เป็น .jsp แต่การพัฒนา Website หรือ Web Application ด้วยภาษา JSP จะสามารถแยกย่อยได้อีก 3 รูปแบบ ซึ่งข้าพเจ้าได้นำมาใช้ 1 รูปแบบคือ Scriptlets เป็นการเขียน JSP ร่วมกับ HTML ให้มองว่าเหมือนพวก ASP และ PHP แต่จะใช้นามสกุลไฟล์ .jsp ส่วน Code นั้นก็สามารถเขียนร่วมกับ HTML หรือสลับกันไปมาได้

2.1.2.3 JavaScript คือ ภาษาคอมพิวเตอร์สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ตที่กำลังได้รับความนิยมอย่างสูง และเป็น ภาษาสคริปต์เชิงวัตถุ (ที่เรียกกันว่า "สคริปต์" (script)) ซึ่งในการสร้างและพัฒนาเว็บไซต์ (ใช้ร่วมกับ HTML) เพื่อให้เว็บไซต์ของเราดูมีการเคลื่อนไหว สามารถตอบสนองผู้ใช้งานได้มากขึ้น ซึ่งมีวิธีการทำงานในลักษณะ แปลความและดำเนินงานไปที่ละคำสั่ง (interpret) หรือเรียกว่า อ็อบเจกต์โอเรียนเตด (Object Oriented Programming) ที่มีเป้าหมายในการ ออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ต สำหรับผู้เขียนด้วยภาษา HTML สามารถทำงานข้ามแพลตฟอร์มได้ โดยทำงานร่วมกับภาษา HTML และภาษา Java ได้ทั้งทางฝั่งไคลเอนต์ (Client) และ ทางฝั่งเซิร์ฟเวอร์ (Server)

2.1.2.4 JSON (JavaScript Object Notation) คือ รูปแบบของข้อมูลที่ใช้สำหรับแลกเปลี่ยนข้อมูลที่มีขนาดเล็ก ซึ่งคนสามารถทำความเข้าใจได้ง่าย และสามารถถูกสร้างและอ่านโดยเครื่องได้ง่าย มันถูกกำหนดภายใต้ภาษา JavaScript (JavaScript Programming Language, Standard ECMA-262 3rd Edition – December 1999.) JSON เป็นรูปแบบข้อมูลตัวอักษรที่มีความเป็นอิสระอย่างสมบูรณ์ แต่จะมี

หลักการการเขียนที่คุ้นเคยกับนักเขียนโปรแกรมภาษาต่าง ๆ ได้ ไม่ว่าจะเป็น ภาษา C, C++, C#, Java, Javascript, Perl, Python และอื่น ๆ คุณสมบัติเหล่านี้ทำให้ JSON เป็นภาษาแลกเปลี่ยนข้อมูลที่มีสมบูรณ์แบบ

ปัจจุบัน JSON นิยมใช้ในเว็บแอปพลิเคชัน โดยเฉพาะ AJAX โดย JSON เป็นฟอร์แมตทางเลือกในการส่งข้อมูล นอกเหนือไปจาก XML ซึ่งนิยมใช้กันอยู่แต่เดิม สาเหตุที่ JSON เริ่มได้รับความนิยมเป็นเพราะกระชับและเข้าใจง่ายกว่า XML

2.1.2.5 AJAX ย่อมาจาก Asynchronous JavaScript and XML เป็นการนำเทคโนโลยีต่าง ๆ มาทำงานร่วมกัน เพื่อให้การใช้งานเว็บเพจประหยัดและสะดวกขึ้น

โดยปกติเมื่อ ผู้ใช้เข้าเว็บหรือร้องขอข้อมูลจากทาง Server จะต้องรอ Server ประมวลผลเสร็จก่อนจึงจะส่งกลับมาแสดงผลเว็บเพจของผู้ใช้ จึงเกิดการ Refresh ใหม่ทุกครั้งทีร้องขอข้อมูล จึงมีการนำ AJAX มาช่วยในการทำงานให้รวดเร็วขึ้นโดย AJAX สามารถลดการ Refresh ทำให้การแสดงผลนุ่มนวลขึ้น

โดยหัวใจสำคัญในการทำงานของ AJAX คือ XMLHttpRequest Object ซึ่งทำหน้าที่ขอข้อมูลทางฝั่ง Server โดยข้อมูลที่ขอนั้นเป็นข้อมูลเล็ก ๆ และใช้ JavaScript นำมาแสดงผลบนหน้าจอเดิม โดยไม่ต้อง Refresh หน้าจอใหม่

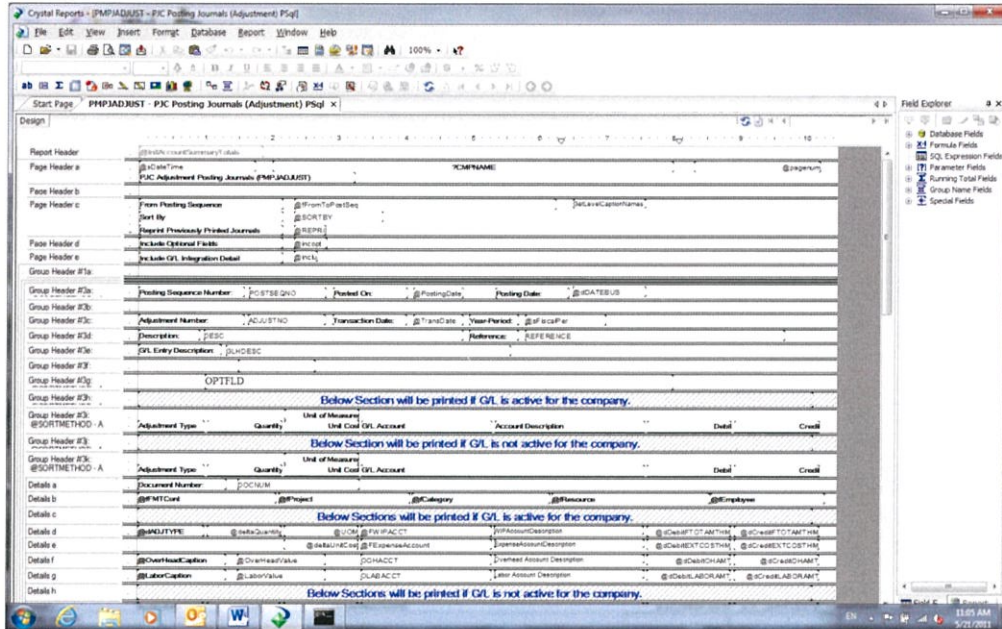
2.1.3 โปรแกรม Crystal Report

Crystal Report คือ เครื่องมือที่ใช้ในการพัฒนารายงาน ซึ่งสามารถ แสดงรายงานได้หลากหลายรูปแบบ ทั้งแบบ รายงานธรรมดา แบบ Cross Tab และแบบอื่น ๆ ซึ่งมีเครื่องมือที่ออกแบบมาให้ง่ายต่อการใช้งาน และการติดต่อกับฐานข้อมูลก็สามารถทำได้หลากหลาย เช่น Microsoft SQL Server, Microsoft Access Excel XML ADO.Net ตลอดจนสามารถนำข้อมูลจาก Viewer ของเครื่องมาดูก็ทำได้ ซึ่งให้ความสามารถที่หลากหลาย และการ View ก็สามารถใช้ได้หลากหลาย เช่น การ View ผ่านตัวโปรแกรมเอง การ View ผ่านโปรแกรมที่เป็น โปรแกรมประยุกต์ที่ Software House ต่าง ๆ ผลิตขึ้นมา หรือแม้กระทั่งดูบนเว็บ ซึ่งจากความสามารถที่หลากหลายดังกล่าวจึงเป็นที่นิยมใช้งานในเชิงพาณิชย์ ซึ่งโครงสร้างของ Crystal Report จะประกอบไปด้วย

2.1.3.1 พื้นที่ต่าง ๆ ในการทำงาน (Section)

2.1.3.2 ส่วนที่ใช้ในการแสดงผลข้อมูล (Fields)

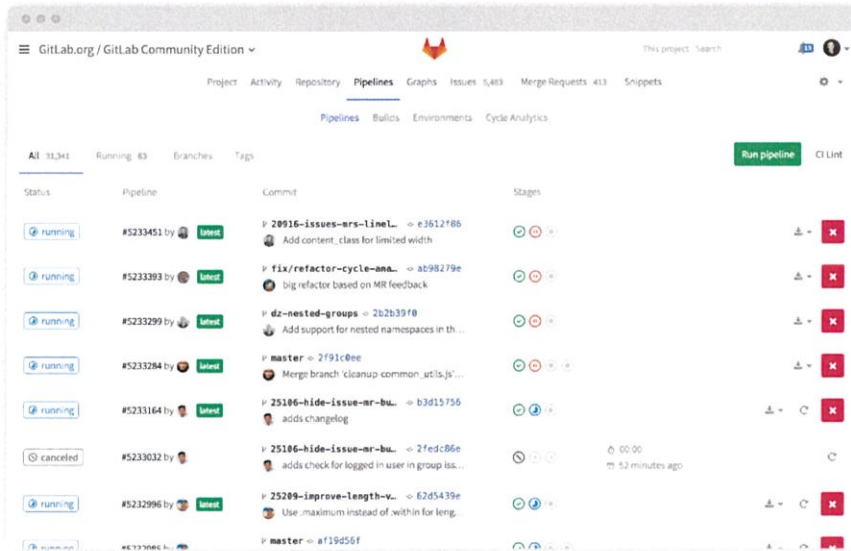
- 2.1.3.3 การสร้างสูตรและการใช้งานสูตร (Function)
- 2.1.3.4 การจัดกลุ่ม (Group)
- 2.1.3.5 การแสดงกราฟ (Chart)
- 2.1.3.6 การสร้างรายงานย่อย (SubReport)



รูปที่ 2.3 ตัวอย่างโปรแกรม Crystal Report

2.1.4 Gitlab

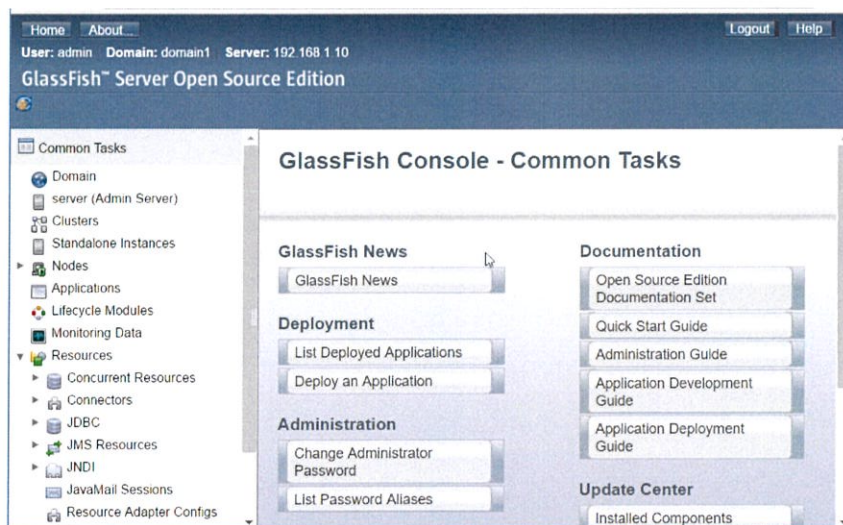
Git คือ Version Control ตัวหนึ่ง ซึ่งเป็นระบบที่มีหน้าที่ในการจัดเก็บการเปลี่ยนแปลงของไฟล์ในโปรเจกต์นี้ มีการ backup code ให้ สามารถที่จะเรียกดูหรือย้อนกลับไปดูเวอร์ชันต่าง ๆ ของโปรเจกต์ที่ได้เวลาใดก็ได้ หรือแม้แต่ดูว่าไฟล์นั้น ๆ ใครเป็นคนเพิ่มหรือแก้ไข หรือว่าจะดูว่าไฟล์นั้น ๆ ถูกเขียนโดยใครบ้างก็สามารถทำได้ ฉะนั้น Version Control ก็เหมาะอย่างยิ่งสำหรับนักพัฒนาไม่ว่าจะเป็นคนเดียวโดยเฉพาะอย่างยิ่งจะมีประสิทธิภาพมากหากเป็นการพัฒนาเป็นทีม ซึ่งที่ทีมผู้วิจัยใช้จะเป็น Gitlab ที่ทางทีม IT ของ SCB จัดการให้ จุดเด่นของ Gitlab คือ สร้าง Private Project รองรับการทำงานเป็นทีมได้ โดยสามารถเพิ่มผู้ใช้ กำหนดสิทธิการใช้งาน และกำหนดสมาชิกให้แต่ละโครงการได้



รูปที่ 2.4 ตัวอย่างโปรแกรม GIT ที่ใช้สำหรับทำ Version Control

2.1.5 Glassfish Server

Glassfish Server คือ Application Server ที่ใช้สร้าง Web Server เพื่อที่จะทำให้ Web Application ทำงานได้ โดยใช้ผ่านทาง link Glassfish และ Username , Password ที่ได้ตั้งค่าไว้ ซึ่งโปรเจกต์นี้ได้ใช้ Glassfish Server ในการลง SingleApp (Web Application)



รูปที่ 2.5 ตัวอย่าง Glassfish Server

2.1.6 Microsoft Access 2016

Microsoft Access เป็นโปรแกรมในการจัดการระบบฐานข้อมูล (Database Management System) ที่สามารถจัดเก็บข้อมูลจำนวนมาก ๆ เข้ามาและยังสามารถพัฒนาแบบฟอร์ม (Form) และสร้างรายงาน (Report) ได้อีกด้วย โดยสามารถใช้พัฒนาระบบงานง่าย ๆ จนถึงซับซ้อน ได้ และยังสามารถใช้งานพร้อมกันหลาย ๆ คน ได้ โดยมีส่วนประกอบที่สำคัญของ Microsoft Access ดังนี้

1. ตาราง (Table) คือ ตารางจัดเก็บข้อมูล โดยจำเป็นต้องออกแบบ เพื่อรองรับกับการทำงาน
2. คิวรี (Query) คือ แบบสอบถาม เพื่อสืบค้นข้อมูลจากฐานข้อมูล
3. ฟอร์ม (Form) คือ แบบฟอร์ม เพื่อใช้เป็นหน้าจอในการ กรอกข้อมูล หรือ แสดงผลข้อมูล
4. รายงาน (Report) คือ รายงาน ที่เป็นผลลัพธ์ เพื่อใช้แสดงผล และพิมพ์ออกมาทางเครื่องพิมพ์
5. มาโคร (Macro) คือ ชุดคำสั่งอัตโนมัติที่ใช้สั่งงานให้ทำงานตามที่ต้องการ
6. โมดูล (Module) คือ ชุดคำสั่งขั้นสูงที่จะให้นักพัฒนาระบบสามารถปรับแต่ง สั่งงาน ให้ทำงานตามที่ต้องการได้



รูปที่ 2.6 โลโก้โปรแกรม Microsoft Access 2016

Microsoft Access ถูกนำไปใช้งานในระบบฐานข้อมูลได้ในหลากหลายธุรกิจ รองรับการทำงานพร้อม ๆ กัน (Concurrent Usage) รองรับการพัฒนาปรับแต่ง และนำไปใช้กับระบบอื่น ๆ ได้ สามารถ Import/Export Data ไปยังระบบต่าง ๆ เช่น Excel, SQL Server, Text File เป็นต้น ซึ่งจะมีนามสกุล .accdb หรือ .mdb

ลักษณะงานเหมาะกับ Microsoft Access

- งานด้านลูกค้าสัมพันธ์ (Customer Management)
- งานระบบ เช่า / ยืม-คืน สินค้า (Rental System)
- งานติดตามใบสั่งซื้อ (Order Tracking)
- งานระบบ ซื้อ/ขาย สินค้า (Order and Purchase System)

- งานติดตามงานในองค์กร (Task Tracking)
- งานบันทึกสินค้าคงคลัง และจัดการสินทรัพย์ (Inventory and Asset Tracking)

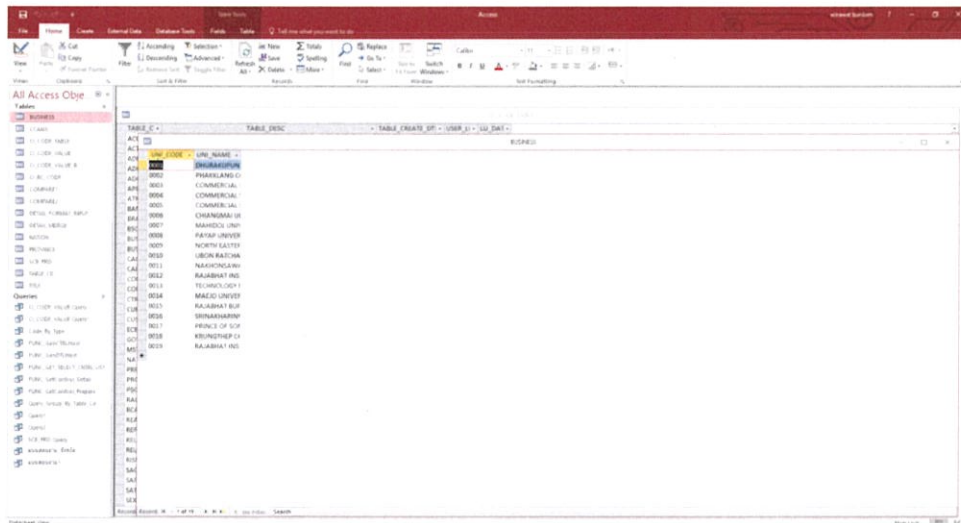
ข้อจำกัดของ Microsoft Access

- การใช้งานหลายคนจำเป็นต้องแชร์ข้อมูล (Sharing) ซึ่งจะต้องกำหนดสิทธิ์ในการเข้าถึงให้เหมาะสม

- รองรับขนาดไฟล์สูงสุด คือ 2 GB.

- หากไฟล์เสีย อาจจะทำให้ทุกเครื่องใช้งานไม่ได้ ดังนั้นจำเป็นต้อง Backup และดูแลรักษาให้ถูก

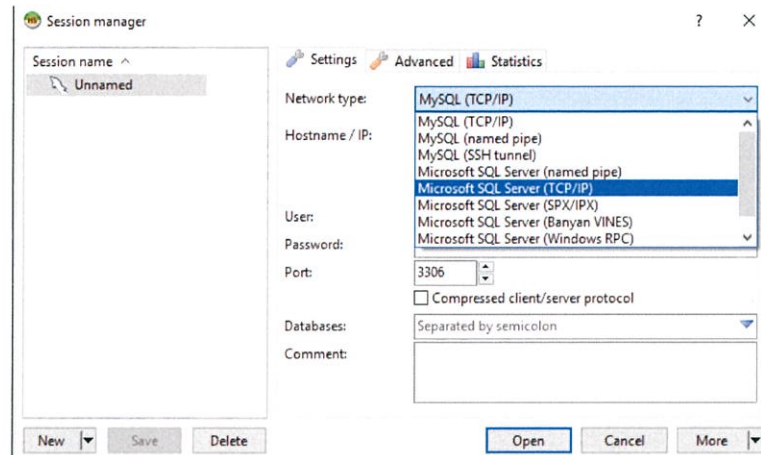
วิธี



รูปที่ 2.7 ตัวอย่างโปรแกรม Microsoft Access 2016

2.1.7 HeidiSQL

HeidiSQL เป็นโปรแกรม ขนาดเล็ก ที่ใช้ติดต่อกับ SQL Server เพื่อบริหารและจัดการฐานข้อมูล MySQL/MSSQL ทั้งที่อยู่ใน Localhost และ บน Web Hosting สามารถ Create Table, Create View, Create Stored Procedure, Create Trigger, Create scheduled event, Run SQL queries และอื่น ๆ อีกมากมาย ใช้งานง่ายและสะดวก ที่สำคัญที่สุดคือ เป็น Freeware



รูปที่ 2.8 ตัวอย่างโปรแกรม HeidiSQL

จากรูปผู้วิจัยได้ใช้โปรแกรม HeidiSQL เพื่อพัฒนาโปรแกรมทั้ง FEB SingleApp และ Bulk Payroll โดยการใช้การเชื่อมต่อแบบ Microsoft SQL Server (TCP/IP) ทั้ง 3 โปรแกรม

SQL Server หรือ Microsoft SQL Server คือระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management System หรือ RDBMS) ผลิตโดยบริษัท Microsoft เป็นระบบฐานข้อมูลแบบ Client/Server และรันอยู่บน Window NT ซึ่งใช้ภาษา T-SQL ในการดึงเรียกข้อมูล ด้วยเหตุที่ข้อมูลส่วนใหญ่ทั่วโลกเก็บไว้ในเครื่องที่ใช้ Microsoft Windows เป็น Operating System จึงทำให้เป็นการง่ายต่อ Microsoft SQL ที่จะนำ ข้อมูลที่อยู่ในรูป Windows Based มาเก็บและประมวลผล และ ประกอบกับการที่ ราคาถูกและหาง่าย จึงเป็นปัจจัยหลักที่ทำให้ Microsoft SQL จึงเป็นระบบฐานข้อมูลที่มีมักจะถูกเลือกใช้



รูปที่ 2.9 โลโก้ Microsoft SQL Server

2.1.8 Toad for DB2

DB2 เป็นผลิตภัณฑ์ในกลุ่ม ระบบจัดการฐานข้อมูล (RDBMS) ของ ไอบีเอ็ม ปัจจุบันถูกเรียกว่า ไอบีเอ็ม DB2 Data Server ซึ่งเป็นผลิตภัณฑ์ในกลุ่ม ไอบีเอ็ม Information Management Software

ไอบีเอ็ม DB2 Data Server แบ่งเป็นหลายแบบตามความเหมาะสมในการใช้งาน DB2 สามารถทำงานกับอุปกรณ์ต่าง ๆ ตั้งแต่เครื่อง PC จนกระทั่งถึงเมนเฟรมคอมพิวเตอร์ นอกเหนือจาก DB2 แล้ว ไอบีเอ็ม ยังมีระบบจัดการฐานข้อมูล อีกตัวอื่น เช่น Informix ซึ่งถูก ไอบีเอ็ม ซื้อมาเมื่อปี 2001.

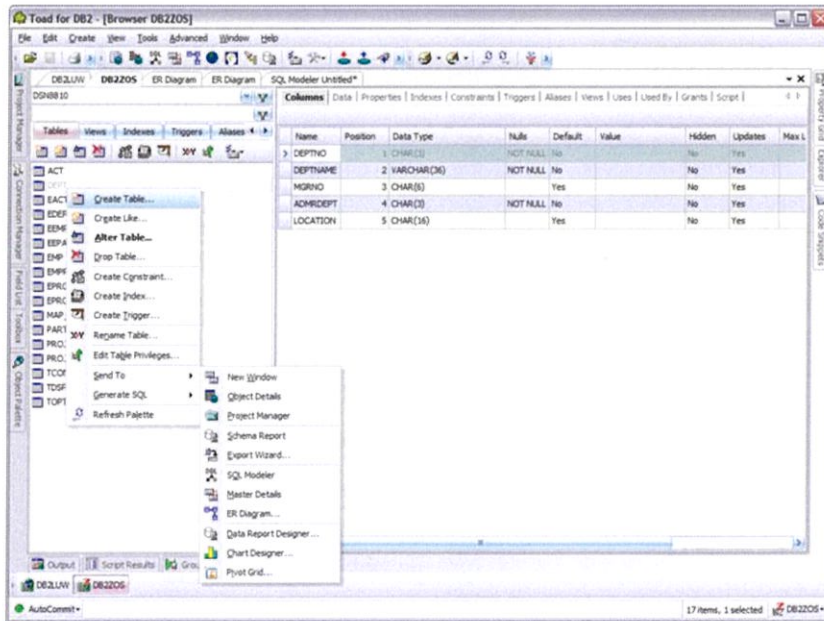
DB2 มีประวัติยาวนานและถูกมองว่าเป็นระบบจัดการฐานข้อมูลแรกที่ใช้ภาษา SQL (SQL คิดค้นโดย ไอบีเอ็ม) ชื่อ "DB2" ถูกใช้ครั้งแรกเมื่อค.ศ. 1983 เมื่อ ไอบีเอ็ม ส่ง DB2 ซึ่งทำงานกับระบบปฏิบัติการ MVS บนเครื่องเมนเฟรมคอมพิวเตอร์ แต่ก่อนหน้านี้ ไอบีเอ็ม เคยออกผลิตภัณฑ์ที่ชื่อว่า SQL/DS ซึ่งเป็นระบบจัดการฐานข้อมูลเหมือนกันแต่ทำงานบนระบบปฏิบัติการ VM บนเครื่องเมนเฟรมคอมพิวเตอร์มาก่อนแล้ว สำหรับระบบจัดการฐานข้อมูลที่เป็นตัวต้นแบบนั้นถูกพัฒนามาตั้งแต่ยุค 1970s ซึ่งถูกเรียกว่า System R และถูกนำมาใช้งานร่วมกับเครื่อง System 38 ในยุคปี ค.ศ. 1968 Dr. E.F. Codd (ซึ่งทำงานอยู่ที่ ไอบีเอ็ม) ได้นำเสนอโมเดลเชิงสัมพันธ์ ซึ่งมีหลักการอยู่บนพื้นฐาน ทางคณิตศาสตร์ เพื่อแก้ปัญหาของโมเดลฐานข้อมูลแบบเดิม ไอบีเอ็ม ก็ได้นำแนวคิด ของ Dr. E.F. Codd ไปสร้างระบบฐานข้อมูลเชิงสัมพันธ์ที่ชื่อว่า System R ขึ้น และพัฒนาต่อเนื่องมาเรื่อย จนเป็น DB2 รวมถึงการพัฒนาภาษา SEQUEL เพื่อใช้สำหรับเรียกดูและจัดการกับข้อมูลในฐานข้อมูลเชิงสัมพันธ์ ในช่วงปีต่อมาคือ 1976-1977 ได้มีการปรับปรุงเวอร์ชันใหม่จาก SEQUEL เป็น SEQUEL/2 และภายหลังก็ได้เปลี่ยนแปลงชื่อเป็น SQL

เริ่มแรกนั้น DB2 ถูกพัฒนามาเพื่อใช้งานกับเครื่อง ไอบีเอ็ม เมนเฟรม หลังจากนั้น (1990s) ได้มีการพัฒนาต่อเนื่องทำให้สามารถใช้งานกับแพลตฟอร์มอื่น ๆ ได้ทั้ง MS. Windows Linux ต่าง ๆ ไอบีเอ็ม AIX Sun Solaris HP-UX ไอบีเอ็ม i/OS แต่ภาษาที่ใช้ในการพัฒนา DB2 บนเครื่องเมนเฟรมกับแพลตฟอร์มอื่นนั้นแตกต่างกัน DB2 สำหรับเครื่องเมนเฟรมนั้นถูกพัฒนาโดยภาษา PL/S แต่ DB2 LUW (Linux/Unix/Windows) นั้นถูกพัฒนาโดยภาษา C++ แต่โดยรวมแล้ว DB2 จะมีการแชร์ฟังก์ชันและ Common Architecture ร่วมกัน กลางยุค 1990s ไอบีเอ็ม ได้มีการพัฒนา DB2 Parallel Edition ขึ้นภายใต้แนวความคิด Shared Nothing Architecture เพื่อรองรับการขยายตัวของฐานข้อมูลขนาดใหญ่ โดยสามารถทำการกระจายข้อมูลต่าง ๆ ไปยังเครื่อง DB2 Server หลาย ๆ เครื่องได้ หลังจากนั้นได้มีการพัฒนาเรื่อยมาและมีการเปลี่ยนชื่อเป็น DB2 Extended Enterprise Edition (DB2 EEE) และในปัจจุบัน ไอบีเอ็ม เรียกความสามารถในส่วนนี้ว่า Data Partitioning Feature (DPF) ซึ่งเป็นส่วนเสริม (option) ซึ่งต้องใช้งานร่วมกับ DB2 Enterprise Edition ปี 2006 ทาง ไอบีเอ็ม ได้มีการพัฒนาเพิ่มเติมในส่วนของ XML ทำให้ DB2 สามารถจัดเก็บข้อมูล XML ในลักษณะ "natively" โดยมีการพัฒนา Database Engine เพิ่มเติมสำหรับจัดการข้อมูล XML โดยเฉพาะเรียกว่า (PureXML) ซึ่งในโหมดนี้ XML data จะถูกจัดเก็บในลักษณะที่เป็น XML จริง ๆ ไม่ใช่ CLOB (Character Large Object) หรือ relation data ทำให้มีประสิทธิภาพในการเข้าถึงดีขึ้น



รูปที่ 2.10 โลโก้ IBM DB2

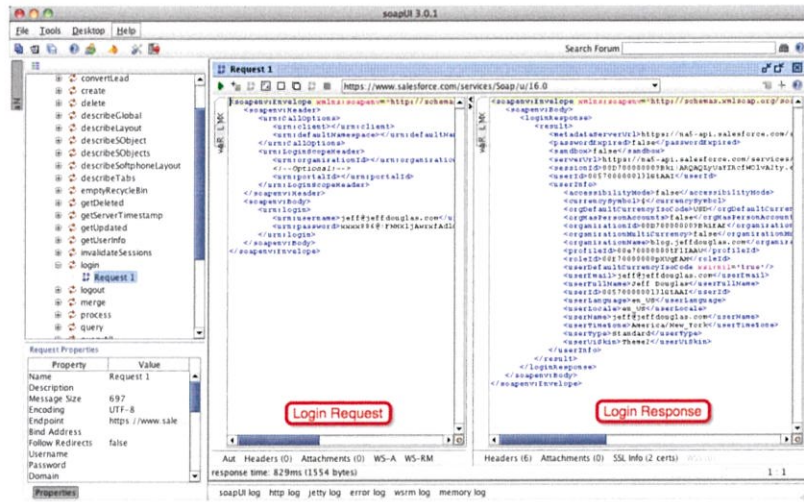
การบริหารจัดการ ไอบีเอ็ม DB2 นั้นสามารถทำได้ผ่าน 2 ช่องทางคือ command-line หรือ Graphic User Interface (GUI) ที่เรียกว่า DB2 Control Center ซึ่งสามารถใช้บริหารจัดการ DB2 ได้ทุกเอ็ดิชั่นและเนื่องจาก Control Center นั้นถูกพัฒนาโดย JAVA จึงทำให้สามารถทำงานได้หลายแพลตฟอร์ม DB2 สนับสนุนการใช้คำสั่งทั้ง SQL และ XQuery ผู้พัฒนาแอปพลิเคชันสามารถพัฒนาโปรแกรมผ่าน APIs ต่าง ๆ ดังนี้ API for .NET CLI, Java, Python, Perl, PHP, Ruby, C++, C, REXX, PL/I, COBOL, RPG, FORTRAN, และภาษาอื่น ๆ การพัฒนา DB2 แอปพลิเคชันนั้น สามารถพัฒนาผ่าน Eclipse หรือ Visual Studio .NET และอื่น ๆ ได้



รูปที่ 2.11 ตัวอย่างโปรแกรม Toad for DB2

2.1.8 SoapUI

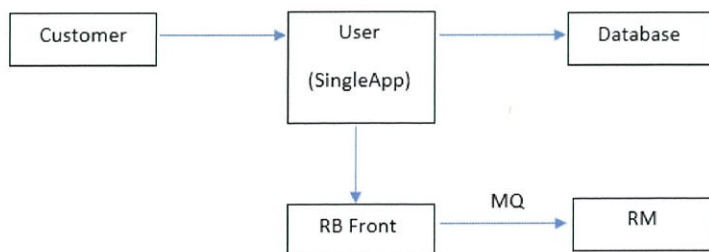
SoapUI คือ Software Testing tool ของ Service Oriented Architecture (SOA) ที่มีไว้เพื่อเหล่าผู้พัฒนา Software (Software Developer) โดย SoapUI นั้นเป็น โปรแกรมในกลุ่ม Open Source ที่เผยแพร่ภายใต้ข้อตกลงของ GNU Lesser General Public License. และเจ้าตัว soapUI นี้ยังเป็นเครื่องมือทดสอบใช้เรียกเว็บเซอร์วิส โดยไม่จำเป็นต้องเขียนโปรแกรมเพื่อเรียก ลดความยุ่งยาก และเพิ่มความรวดเร็วในการทดสอบเว็บเซอร์วิส โดย soapUI นั้นเป็นเครื่องมือสำหรับดูเอกสาร wsdl และสั่ง (invoke) method บน webservice



รูปที่ 2.12 ตัวอย่างโปรแกรม SoapUI

2.2 ทฤษฎีในด้านธุรกิจ

2.2.1 กระบวนการเปิดบัญชีของธนาคาร

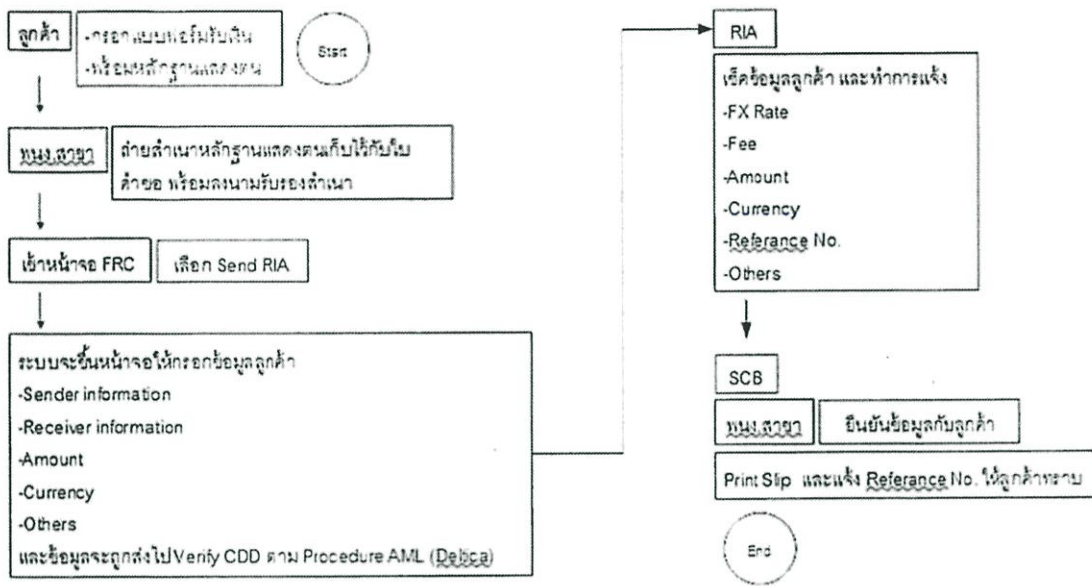


รูปที่ 2.13 กระบวนการเปิดบัญชีของธนาคาร

จากรูปที่ 2.13 ถ้าลูกค้าต้องการเปิดบัญชีที่สาขา พนักงานที่สาขาจะทำรายการเปิดบัญชีผ่าน SingleApp โดยจะเก็บข้อมูลต่าง ๆ ตามเงื่อนไขของบัญชีที่ลูกค้าต้องการเปิด และข้อมูลจะถูกบันทึกลงฐานข้อมูลของ SingleApp และส่งไปยังระบบ RB Front เพื่อทำหน้าที่ Confirm ข้อมูลกับระบบ Core Bank จากนั้นจะส่งต่อไปยังระบบ RM เพื่อเก็บข้อมูลในขั้นตอนสุดท้ายโดยผ่าน Web Service MQ

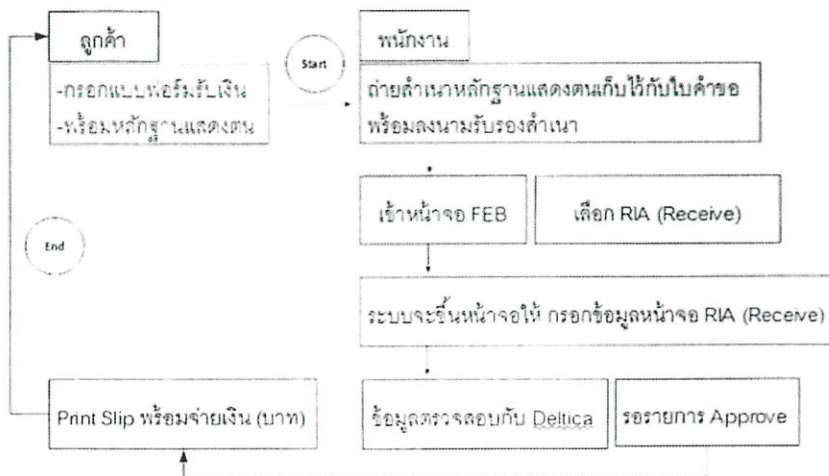
2.2.2 กระบวนการโอนเงินต่างประเทศ

กระบวนการโอนเงินต่างประเทศจะเริ่มจากลูกค้ากรอกแบบฟอร์มรับเงินพร้อมหลักฐานแสดงตนให้กับพนักงานสาขาเพื่อยืนยันตน พนักงานสาขาจะถ่ายสำเนาหลักฐานแสดงตนเก็บไว้กับใบคำขอรับเงิน พร้อมลงนามรับรองสำเนา จากนั้นพนักงานสาขาจะเข้าสู่หน้าจอ FRC เลือก ผู้ให้บริการที่ต้องการใช้งาน จากนั้นระบบจะขึ้นหน้าจอให้กรอกข้อมูลลูกค้าทั้งคนรับ คนส่ง และจำนวนเงินที่ต้องการโอน ข้อมูลดังกล่าวจะถูกส่งไปตรวจสอบกับระบบตรวจสอบรายชื่อผู้ติด Blacklist (Detica) เมื่อตรวจสอบเสร็จแล้วผ่านจะส่งข้อมูลไปยัง RIA เพื่อเช็คข้อมูลลูกค้า และทำการแจ้งข้อมูลที่จำเป็นต้องนำกลับมาทาง SCB จากนั้นพนักงานสาขาจะทำการยืนยันข้อมูลกับลูกค้า และ Print Slip ตามด้วยแจ้ง Reference No. ให้ลูกค้าทราบ



รูปที่ 2.14 ตัวอย่างกระบวนการโอนเงินต่างประเทศ ขาเข้า

เมื่อลูกค้าได้ Reference No. แล้ว จะส่งให้กับผู้รับเงิน จากนั้นผู้รับเงินจะนำ Reference No. มาแสดงกับพนักงานพร้อมหลักฐานแสดงตน พนักงานจะถ่ายสำเนาหลักฐานแสดงตนเก็บไว้กับใบคำขอพร้อมลงนามรับรองสำเนา จากนั้นพนักงานจะเข้าโปรแกรม FEB (Foreign Exchange Booth) แล้วเลือก RIA Receive ทำการกรอกข้อมูล Reference No. ของลูกค้า จากนั้นจะแสดงข้อมูลให้ลูกค้าตรวจสอบความถูกต้อง เมื่อถูกต้องแล้วจะทำการ Print Slip พร้อมจ่ายเงิน(บาท) ให้กับลูกค้า เป็นอันสิ้นสุดโปรแกรม



รูปที่ 2.15 ตัวอย่างกระบวนการโอนเงินต่างประเทศ ขาออก

บทที่ 3 วิธีดำเนินการวิจัย

งานวิจัยชิ้นนี้ เป็นงานวิจัยเชิงพัฒนา โดยมีการพัฒนาระบบที่สนับสนุนการขับเคลื่อนธุรกิจให้มีประสิทธิภาพมากยิ่งขึ้น และมีการแก้ไขปัญหาที่เกิดขึ้นภายในระบบ เพื่อเพิ่มความพึงพอใจแก่ผู้ใช้ โดยงานวิจัยชิ้นนี้ถูกแบ่งออกเป็น 3 ส่วน คือ การพัฒนาโปรแกรม การทดสอบ และการนำไปใช้จริง ซึ่งบางโปรเจกต์ผู้วิจัยได้รับมอบหมายให้แก้ปัญหาโปรแกรมเมื่อผู้ใช้งานแจ้งปัญหาเข้ามาด้วย โดยวิธีการดำเนินงานจะแบ่งออกเป็นแต่ละโปรเจกต์ ดังนี้

3.1 โปรเจกต์ RIA

3.1.1 การพัฒนา (Development)

Development เป็นส่วนการพัฒนาโปรแกรม FEB (Foreign Exchange Booth) ซึ่งใช้ Visual Basic 6.0 ในการพัฒนา เริ่มแรกจะมีเอกสารที่แสดงความต้องการต่าง ๆ ของผู้ใช้งาน (Requirement List) มาจากผู้ใช้งานว่าต้องการแก้ไขโปรแกรมไปในทางไหน จากนั้นผู้วิจัยจะทำการพิจารณาและตีความเพื่อหาวิธีการแก้ไขโปรแกรมให้ตรงตามความต้องการของผู้ใช้งานและไม่กระทบส่วนต่าง ๆ ของโปรแกรมเดิม

Module	Module Description	Req.#	Status	Req. Description	Validation	Validation Required	Validation Completed	Error ID
1	เพิ่มเมนูใหม่สำหรับค้นหา New Provider		Waiting screen	Create screen to support search/submit to Cash				
2	เพิ่มเมนูใหม่สำหรับค้นหา New Provider		Waiting screen	Create screen to support search/submit to Cash				
3	เพิ่มเมนูใหม่สำหรับค้นหา New Provider		Waiting screen	Create screen to support search/submit to Cash				
4	เพิ่มเมนูใหม่สำหรับค้นหา New Provider		Waiting screen	Create screen to support search/submit to Cash				
5	เพิ่มเมนูใหม่สำหรับค้นหา New Provider		Waiting screen	Create screen to support search/submit to Cash				
6	เพิ่มเมนูใหม่สำหรับค้นหา New Provider		Waiting screen	Create screen to support search/submit to Cash				

รูปที่ 3.1 ตัวอย่างเอกสาร Requirement List ของโปรเจกต์ RIA

งาน / ระยะเวลา	ส.ค.				ก.ย.				ต.ค.				พ.ย.			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
ประชุมงาน					■								■			
วางแผน/ออกแบบ					■								■			
พัฒนาโปรแกรม						■	■	■	■	■	■	■		■	■	■
ทดสอบการใช้งาน									■	■					■	■
นำไปใช้งานจริง																■

ตารางที่ 3.1 แผนการพัฒนาโปรเจกต์ RIA

3.1.1.1 ความต้องการของผู้ใช้งาน

- สร้างหน้าจอสําหรับบริการโอนเงินผ่าน New Provider
- เชื่อมต่อระบบเพื่อ Settlement ระหว่าง SCB กับ Provider
- เชื่อมต่อระบบเพื่อดําเนินธุรกิจเงินโอนต่างประเทศ ทั้งขารับ และขาส่งโดยผ่านช่องทางต่าง ๆ ดังนี้
 - ลูกค้ํารับเงินสดผ่านเคาน์เตอร์สาขา SCB และผ่านบูธแลกเปลี่ยนฯ SCB
 - ลูกค้ํารับเงินผ่านบัญชี SCB
 - ลูกค้ํารับเงินผ่านตู้ ATM SCB
 - ลูกค้ําส่งเงินผ่านเคาน์เตอร์สาขา SCB และผ่านบูธแลกเปลี่ยนฯ SCB
 - ลูกค้ําส่งเงินผ่านระบบ SCB Easy Net

No.	Business Scenario	Expected Results
1	ลูกค้าโอนเงินผ่านระบบ RIA ที่สาขา โดยใช้เงินสด จากไทยไปต่างประเทศ	ลูกค้าในต่างประเทศได้รับเงิน
2	ลูกค้าโอนเงินผ่านระบบ RIA ที่สาขา โดยใช้ ATM ไทยไปต่างประเทศ	ลูกค้าในต่างประเทศได้รับเงิน
3	ลูกค้ารับเงินโอนผ่านระบบ RIA ที่สาขา เป็นเงินสด	ลูกค้าได้รับเงิน
4	ลูกค้ารับเงินโอนผ่านระบบ RIA เข้าบัญชี SCB	ลูกค้าได้รับเงิน
5	ลูกค้ารับเงินโอนผ่านระบบ RIA ที่บูธฯ เป็นเงินสด	ลูกค้าได้รับเงิน

ตารางที่ 3.2 Business Scenario ของ RIA

ซึ่งผู้วิจัยได้ทำความเข้าใจเกี่ยวกับความต้องการของผู้ใช้งานได้ว่า ผู้ใช้งานต้องการเพิ่มหน้า RIA Receive เพื่อตอบสนองต่อ Provider ใหม่ รวมถึงเก็บข้อมูลลูกค้าลงฐานข้อมูล สามารถทำงานร่วมกับโปรแกรมอื่นได้อย่างสอดคล้องกัน และที่สำคัญคือไม่มีความผิดพลาดในการทำรายการจ่ายเงินให้กับลูกค้า

3.1.1.2 วางแผนและออกแบบการพัฒนาโปรแกรม

จากความต้องการของผู้ใช้งานทำให้ต้องออกแบบ Form ขึ้นมาใหม่ รวมถึงออกแบบ Database table ใหม่ในการเก็บข้อมูลอีก 3 table ด้วย ซึ่งในส่วนของการออกแบบฐานข้อมูล จะแบ่งตามการทำงานคือ ที่ Local จะใช้ Microsoft SQL Server (TCP/IP) เก็บ 2 table และ Host ใช้ DB2 เก็บ 1 table ส่วน field ต่าง ๆ จะทำการสร้างตามที่คุณต้องการให้เก็บข้อมูล นอกจากนี้ยังมีการพัฒนารายงานที่ทำมาจาก Crystal Report โดยจะ query ข้อมูลจาก Database ที่สร้างขึ้นเพิ่มเข้ามาในรายงานเพื่อแสดงข้อมูลแก่ผู้ใช้งาน เมื่อพัฒนาโปรแกรมเสร็จแล้วจะต้องทำ Unit Test ซึ่งคือการทดสอบด้วยตัวของผู้เขียนโปรแกรมเอง ว่าโปรแกรมทำงานถูกต้องตามความต้องการของผู้ใช้งานแล้วหรือไม่ เมื่อทำงานถูกต้องแล้วจึงจะส่งให้ทีมทดสอบทำการทดสอบโปรแกรม

3.1.2 การทดสอบ (Testing)

Testing จะเกิดเมื่อมีการเปลี่ยนแปลง หรือพัฒนาโปรแกรมจากส่วนของ Development เสร็จเรียบร้อยแล้ว จะส่งขึ้นมายังทีม System Integration Test (SIT) เพื่อทดสอบว่าระบบต่าง ๆ ทำงานได้ตรงตามความต้องการของผู้ใช้หรือไม่ เมื่อทีม SIT ทำการทดสอบเสร็จแล้ว ทีม Development จะต้องทำการเปิด Change เพื่อขออนุญาตติดตั้ง (Deploy) ไปยังระบบ User Acceptance Testing (UAT) เพื่อให้ทางทีม UAT

ทำการทดสอบโปรแกรม ซึ่งถ้าตรวจสอบแล้วพบว่ามีปัญหา หรือมีสิ่งที่ต้องแก้ไข จะถูกส่งกลับไปยังส่วนของ Development อีกครั้ง เพื่อทำการแก้ไข แต่ถ้าไม่มีสิ่งที่ต้องแก้ไข จึงจะส่งไปยังส่วนถัดไป

3.1.3 การนำไปใช้งานจริง (Production)

Production เป็นส่วนที่เมื่อการพัฒนาโปรแกรม หรือการเปลี่ยนแปลงทุกอย่างเสร็จสิ้น โดยผ่านการทดสอบจากทีมทดสอบ ทั้งทีม SIT และ UAT แล้ว จะทำการเปิด Change เพื่อขออนุญาตติดตั้งไปที่ส่วนของ Production เพื่อทำการใช้งานจริงในระบบ เนื่องจากเป็นขั้นตอน UAT และ Production จึงต้องมี QA (Quality Assurance) หรือ ผู้ที่รับประกันคุณภาพงาน อนุมัติเพื่อที่จะทำการติดตั้งโปรแกรมเข้าระบบ ซึ่งถ้ายังเป็น Production แล้วจะต้องผ่านฝ่าย IT Security ด้วยเพื่อทำการตรวจสอบว่าถูกต้องหรือมีช่องโหว่หรือไม่ ถ้ามีช่องโหว่ โปรแกรมจะไม่ถูกอนุมัติเพื่อขึ้น Production แต่ถ้าผ่านการทดสอบแล้วจึงจะสามารถขออนุญาตติดตั้งโปรแกรมเพื่อขึ้น Production ได้

3.2 โปรเจกต์ SingleApp

3.2.1 การพัฒนา (Development)

3.2.1.1 ความต้องการของผู้ใช้งาน

- สร้างผลิตภัณฑ์บัญชีใหม่เพื่อเป็นทางเลือกให้ลูกค้าได้เปิดใช้บริการ ดังนี้
 - EasySaving Account
 - SCB Feel Free Savings Account
 - Education Saving
- ผลิตภัณฑ์ต่าง ๆ จะมีเงื่อนไขในการเปิดบัญชีแตกต่างกันไป
- เปลี่ยน Endpoint และ Port เพื่อรองรับการเชื่อมต่อ Web Service ใหม่ของ ระบบ EasyApp
- เปลี่ยน Term & Condition ของผลิตภัณฑ์เก่าให้รองรับกับกฎหมายในปัจจุบันตาม requirement
- แก้ปัญหาของ Web Application นี้ เมื่อมีลูกค้าติดต่อส่งเรื่องปัญหาที่เกิดขึ้น

- สร้าง Library ของ Web Application ให้สอดคล้องกับระบบ Salesforce เนื่องจาก SingleApp ทำงานบนระบบ Salesforce

งาน \ ระยะเวลา	ส.ค.				ก.ย.				ต.ค.				พ.ย.			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
ประชุมงาน	■						■					■				
วางแผน/ออกแบบ	■						■					■				
พัฒนาโปรแกรม	■	■	■					■	■	■				■	■	
ทดสอบการใช้งาน		■	■	■	■	■			■	■				■	■	■
นำไปใช้งานจริง						■						■				■

ตารางที่ 3.3 แผนการพัฒนาโปรแกรม SingleApp

3.2.1.2 วางแผนและออกแบบการพัฒนาโปรแกรม

Development เป็นส่วนการพัฒนาโปรแกรม SingleApp ซึ่งเป็น Web Application พัฒนาโดยโปรแกรม NetBeans และใช้ Java, JavaScript, JSON, JSP และ AJAX ในการพัฒนาความต้องการของผู้ใช้งานคือ เพิ่มผลิตภัณฑ์ใหม่เข้าไปที่ Web Application นี้ เพื่อตอบสนองความต้องการของผู้ใช้งานจึงต้องเพิ่มข้อมูลผลิตภัณฑ์ใหม่เข้าไปในฐานข้อมูล ซึ่งจะเป็น Microsoft SQL Server (TCP/IP) จากนั้นทำการเขียนโปรแกรมตามความต้องการของผู้ใช้ โดยหน้า UX/UI และฟังก์ชัน จะต้องตรงตามเอกสาร

เมื่อมีการพัฒนาของโปรแกรมอื่น ๆ แล้วมีผลกระทบต่อโปรแกรม SingleApp เช่น มีการเปลี่ยน Endpoint ของ Web Service ที่ SingleApp ใช้อยู่ จะต้องทำการสร้าง Library .jar ขึ้นใหม่ โดยใช้ Endpoint ที่เปลี่ยนแล้ว เป็นต้น

เมื่อพัฒนาโปรแกรมเสร็จแล้วต้องการทดสอบต้องเพื่อขออนุญาตติดตั้ง ผ่าน Glassfish Server เท่านั้นเพราะ Run ผ่านเครื่องโดยตรงไม่ได้ เนื่องจากจะติดปัญหาเรื่อง Database Web Service และสิทธิของผู้ใช้งาน จากนั้นจะต้องทำ Unit Test ซึ่งคือการทดสอบด้วยตัวของผู้เขียนโปรแกรมเอง ว่าโปรแกรมทำงานถูกต้องตามความต้องการของผู้ใช้งานแล้วหรือไม่ เมื่อทำงานถูกต้องแล้วทำการนำซอร์ซโค้ดลง Gitlab จากนั้นจึงจะส่งให้ทีมทดสอบทำการทดสอบโปรแกรม

3.2.2 การทดสอบ (Testing)

Testing จะเกิดเมื่อมีการเปลี่ยนแปลง หรือพัฒนาโปรแกรมจากส่วนของ Development เสร็จเรียบร้อยแล้ว จะส่งขึ้นมายังทีม System Integration Test (SIT) เพื่อทดสอบว่าระบบต่าง ๆ ทำงานได้ตรงตามความต้องการของผู้ใช้หรือไม่ เมื่อทีม SIT ทำการทดสอบเสร็จแล้ว ทีม Development จะต้องทำการเปิด Change เพื่อขออนุญาตติดตั้งไปยังบนระบบ User Acceptance Testing (UAT) เพื่อให้ทางทีม UAT ทำการทดสอบโปรแกรม ซึ่งถ้าตรวจสอบแล้วพบว่ามีปัญหา หรือมีสิ่งที่จะต้องแก้ไข จะถูกส่งกลับไปยังส่วนของ Development อีกครั้ง เพื่อทำการแก้ไข แต่ถ้าไม่มีสิ่งที่จะต้องแก้ไข จึงจะส่งไปยังส่วนถัดไป

3.2.3 การนำไปใช้งานจริง (Production)

Production เป็นส่วนที่เมื่อการพัฒนาโปรแกรม หรือการเปลี่ยนแปลงทุกอย่างเสร็จสิ้น โดยผ่านการทดสอบจากทีมทดสอบ ทั้งทีม SIT และ UAT แล้ว จะทำการเปิด Change เพื่อขออนุญาตติดตั้งไปที่ส่วนของ Production เพื่อทำการใช้งานจริงในระบบ เนื่องจากเป็นขั้นตอน UAT และ Production จึงต้องมี QA (Quality Assurance) หรือ ผู้ที่รับประกันคุณภาพงาน อนุมัติเพื่อที่จะทำการติดตั้งเข้าระบบ ซึ่งถ้ายังเป็น Production แล้วจะต้องผ่านฝ่าย IT Security ด้วยเพื่อทำการตรวจสอบว่าถูกต้องหรือมีช่องโหว่หรือไม่ ถ้ามีช่องโหว่ โปรแกรมจะไม่ถูกอนุมัติเพื่อขึ้น Production แต่ถ้าผ่านการทดสอบแล้วจึงจะสามารถขออนุญาตติดตั้งโปรแกรมเพื่อขึ้น Production ได้

3.2.4 การแก้ไขปัญหาที่เกิดขึ้นภายในระบบ

ผู้วิจัยได้รับมอบหมายให้ดูแลระบบ SingleApp เป็นส่วนใหญ่ เมื่อมีปัญหาเกิดขึ้นเกี่ยวกับโปรแกรมที่ทำงานผิดพลาด ผู้ใช้งานจะติดต่อผ่านทาง Helpdesk เพื่อประสานงานมายังผู้วิจัย จากนั้นผู้ใช้งานจะต้องทำการเปิด คำร้องขอแก้ปัญหาโปรแกรม (Incident) มายังทีมพัฒนา เพื่ออธิบายรายละเอียด เวลา และรูปภาพของปัญหาที่เกิดขึ้น เมื่อวินิจฉัยปัญหาแล้วผิดจากตัวผู้ใช้งานเองที่ไม่ทำตามคู่มือ จึงติดต่อกลับหาผู้ใช้งานเพื่อทำการแก้ไขให้ถูกต้อง แต่ถ้าผิดที่ตัวโปรแกรม แล้ววินิจฉัยว่าต้องแก้ตัวโปรแกรม จะต้องเปลี่ยน Incident นี้ให้เป็น คำร้องขอแก้โปรแกรม (Problem) ซึ่งต้องรอขึ้นตาม Release หรือถ้าเป็นปัญหาที่เร่งด่วนก็สามารถขอขึ้นนอก Release แบบเร่งด่วนได้ โดยระบบ SingleApp ที่ผู้วิจัยได้ทำการแก้ไขปัญหาที่เกิดขึ้นจะแบ่งออกหัวข้อย่อยตามปัญหาที่ผู้ใช้งานแจ้งเข้ามา ดังนี้

- บัญชีเงินฝากสมาร์ทคิดส์ ไม่สามารถเปิดให้บุคคลอายุต่ำกว่า 1 ปีได้ ซึ่งตามจริงจะกำหนดให้บุคคลธรรมดาที่มีอายุไม่เกิน 16 ปี สามารถเปิดใช้บริการได้ ในกรณีนี้ผู้วิจัยได้วินิจฉัยและลอง

ทดสอบแล้ว พบว่าผิดที่โปรแกรม ในฟังก์ชันคำนวณอายุ จึงต้องทำการแก้ไขโปรแกรมให้ถูกต้อง แล้วขออนุญาตติดตั้งโปรแกรมขึ้น Production ตาม Release ให้เร็วที่สุด

- ระบบ Foreign Account Tax Compliance Act (FATCA) บน SingleApp ทำงานผิดพลาด และไม่สอดคล้องกับระบบอื่น ๆ ผู้วิจัยจึงทำการวินิจฉัยและทดสอบ พบว่า ผิดที่โปรแกรมจริง ซึ่งเป็นส่วนที่ส่งผลกระทบจากการแก้ไขโปรแกรมใน UR อื่น ทำให้ต้องแก้ไขโปรแกรม และขออนุญาตเพื่อติดตั้งโปรแกรมขึ้น Production ใหม่ให้เร็วที่สุด

- ลูกค้าไม่สามารถทำรายการเปิดบัญชีได้ ปัญหานี้เกิดขึ้นเป็นประจำ เนื่องจากตัว SingleApp เป็น Web Application ซึ่งจะไม่ทำงานเมื่อพบ อักขระพิเศษ เพราะใช้ Javascript ในการพัฒนาโปรแกรม ผู้วิจัยจึงทำการตรวจสอบ ข้อมูลของลูกค้าว่ามีอักขระพิเศษปรากฏอยู่หรือไม่ ซึ่งแทบทุกกรณีจะมีอักขระพิเศษปรากฏอยู่ จึงทำการแก้ไขข้อมูลลูกค้าโดยตรง ด้วยการลบอักขระพิเศษให้ลูกค้า และทำการแจ้งว่ามีอักขระพิเศษปะปนอยู่ในข้อมูลของลูกค้า นอกจากนั้นผู้วิจัยได้ทำการแก้ไขโปรแกรมเพื่อให้ผู้ใช้งานไม่สามารถป้อนอักขระพิเศษ เข้าไปในข้อมูลของลูกค้าได้ ทำให้ลดโอกาสที่จะเกิดปัญหานี้ขึ้นอีก แต่จะพบปัญหานี้ในลูกค้ารายเก่าที่มีอักขระพิเศษอยู่ในข้อมูลส่วนตัวอยู่แล้ว เมื่อมาทำรายการผ่าน SingleApp ก็จะทำให้เกิดเหตุการณ์เดียวกัน จึงต้องทำการแก้ไขข้อมูลลูกค้าโดยตรงตามที่ได้กล่าวมา

- ในการเปิดบัญชีแบบไม่มีสมุด ทำรายการเปิดผลิตภัณฑ์ Easy Net ไม่ต้องทำการกรอกข้อมูล สามารถทำรายการต่อได้ ตามจริงจะไม่สามารถทำรายการต่อได้เพราะจะต้องกรอกข้อมูลของ Easy Net ให้เรียบร้อยก่อน ผู้วิจัยจึงทำการตรวจสอบ พบว่าโปรแกรมได้ผิดจริง เพราะมีการเขียนให้ข้ามการตรวจสอบ Mandatory field นั้น และปรึกษากับทีมถึงเหตุผลที่ทำให้การเขียนข้ามนี้ จึงพบว่าเป็นการเขียนทดสอบโปรแกรมของทีมพัฒนาเก่า ผู้วิจัยจึงทำการแก้ไขโปรแกรมเพื่อให้ทำงานได้ตามปกติเดิม

- ไม่ปรากฏ Username และชื่อผู้ใช้งาน ให้ผู้ใช้งานทำรายการในหน้ายืนยันตัวตนและหน้าทำรายการเปิดบัญชี จากการตรวจสอบของผู้วิจัยพบว่าปัญหานี้เกิดจาก Endpoint ของ Web Service ของระบบ Salesforce ไม่ตรงกับของทางระบบ SingleApp ที่เรียกไป จึงต้องทำการแก้ไขไฟล์ wsdl ให้มี Endpoint เหมือนกัน เพื่อสร้าง Library ที่ถูกต้องขึ้นในการ Compile และ Build ไฟล์ .war เพื่อทำการขออนุญาตติดตั้งโปรแกรมลงระบบใหม่

- ระบบ SingleApp อนุญาตให้พิมพ์สระติดกันได้ ส่งผลให้ผู้ใช้งานทำการป้อนข้อมูลลูกค้าผิดได้ ผู้วิจัยจึงทำการแก้ไขโปรแกรม และทดสอบ เพื่อแก้ไขให้ผู้ใช้งานสามารถพิมพ์สระได้ถูกต้อง

- ในช่องสำหรับกรอกเอกสารประจำตัว Passport และเลขประจำตัวต่างด้าว อนุญาตให้
 ผู้ใช้งานป้อนค่าภาษาไทยและอักขระพิเศษเข้าไปได้ ทำให้ข้อมูลของลูกค้ามีสิทธิที่จะผิดพลาด ผู้วิจัยได้
 ตรวจสอบโปรแกรมพบว่าตัวโปรแกรมผิดพลาด จึงทำการแก้ไขโปรแกรมให้มีการเช็คอักขระพิเศษ และห้าม
 กรอกภาษาไทยเข้าไปในช่องนั้น

3.3 โปรเจกต์ Bulk Payroll

3.3.1 การพัฒนา (Development)

3.3.1.1 ความต้องการของผู้ใช้งาน

- สร้างผลิตภัณฑ์บัญชีใหม่เพื่อเป็นทางเลือกให้ลูกค้าได้เปิดใช้บริการ ดังนี้
 - SCB Feel Free Savings Account
 - SCB Feel Free Debit Card
- ผลิตภัณฑ์ต่าง ๆ จะมีเงื่อนไขในการเปิดบัญชีแตกต่างกันไป
- สามารถพิมพ์หน้า First Page ผลิตภัณฑ์ใหม่ให้กับลูกค้าได้

ระยะเวลา งาน	ส.ค.				ก.ย.				ต.ค.				พ.ย.			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
ประชุมงาน	■				■											
วางแผน/ออกแบบ		■														
พัฒนาโปรแกรม			■		■											
ทดสอบการใช้งาน			■	■		■	■	■								
นำไปใช้งานจริง									■							

ตารางที่ 3.4 แผนการพัฒนาโปรเจกต์ Bulk Payroll

3.3.1.2 วางแผนและออกแบบการพัฒนาโปรแกรม

Development เป็นส่วนการพัฒนาโปรแกรม Bulk Payroll ซึ่งใช้โปรแกรม NetBeans ในการพัฒนา โดยใช้ภาษา Java ความต้องการของผู้ใช้งานคือ ต้องการเพิ่มผลิตภัณฑ์ใหม่เข้าไปโดยใช้ชื่อว่า SCB Feel Free Savings Account และ SCB Feel Free Debit Card โดยมีเงื่อนไขต่าง ๆ ตามความต้องการของผู้ใช้ที่ลูกค้าจะต้องทำตาม ในส่วนของการพัฒนาโปรแกรมจะมีการแก้ไขไฟล์ .properties เพื่อเพิ่มผลิตภัณฑ์ด้วย จากนั้นระบบฐานข้อมูลจะใช้ Microsoft SQL Server (TCP/IP) ก็ต้องเพิ่มผลิตภัณฑ์ใหม่เข้าไป และมีการสร้าง First Page ที่จะพิมพ์ให้ลูกค้าขณะทำการเปิดบัญชีเสร็จแล้ว จากนั้นโปรแกรม Bulk Payroll จะให้ไฟล์ที่ต้องส่งไป Core Bank เพื่อทำรายการต่อ เมื่อพัฒนาโปรแกรมเสร็จแล้วจะต้องทำ Unit Test ซึ่งคือการทดสอบด้วยตัวของผู้เขียนโปรแกรมเอง ว่าโปรแกรมทำงานถูกต้องตามความต้องการของผู้ใช้งานแล้วหรือไม่ เมื่อทำงานถูกต้องแล้วจึงจะส่งให้ทีมทดสอบทำการทดสอบโปรแกรม

3.3.2 การทดสอบ (Testing)

Testing จะเกิดเมื่อมีการเปลี่ยนแปลง หรือพัฒนาโปรแกรมจากส่วนของ Development เสร็จเรียบร้อยแล้ว จะส่งขึ้นมายังทีม System Integration Test (SIT) เพื่อทดสอบว่าระบบต่าง ๆ ทำงานได้ตรงตามความต้องการของผู้ใช้หรือไม่ เมื่อทีม SIT ทำการทดสอบเสร็จแล้ว ทีม Development จะต้องทำการเปิด Change เพื่อขออนุญาตติดตั้งไปยังบนระบบ User Acceptance Testing (UAT) เพื่อให้ทางทีม UAT ทำการทดสอบโปรแกรม ซึ่งถ้าตรวจสอบแล้วพบว่ามีปัญหา หรือมีสิ่งที่จะต้องแก้ไข จะถูกส่งกลับไปยังส่วนของ Development อีกครั้ง เพื่อทำการแก้ไข แต่ถ้าไม่มีสิ่งที่จะต้องแก้ไข จึงจะส่งไปยังส่วนถัดไป

3.3.3 การนำไปใช้งานจริง (Production)

Production เป็นส่วนที่เมื่อการพัฒนาโปรแกรม หรือการเปลี่ยนแปลงทุกอย่างเสร็จสิ้น โดยผ่านการทดสอบจากทีมทดสอบ ทั้งทีม SIT และ UAT แล้ว จะทำการเปิด Change เพื่อขออนุญาตติดตั้งไปที่ส่วนของ Production เพื่อทำการใช้งานจริงในระบบ เนื่องจากเป็นขั้นตอน UAT และ Production จึงต้องมี QA (Quality Assurance) หรือ ผู้ที่รับประกันคุณภาพงาน อนุมัติเพื่อที่จะทำการติดตั้งเข้าระบบ ซึ่งถ้ายังเป็น Production แล้วจะต้องผ่านฝ่าย IT Security ด้วยเพื่อทำการตรวจสอบว่าถูกต้องหรือมีช่องโหว่หรือไม่ ถ้ามีช่องโหว่ โปรแกรมจะไม่ถูกอนุมัติเพื่อขึ้น Production แต่ถ้าผ่านการทดสอบแล้วจึงจะสามารถขออนุญาตติดตั้งโปรแกรมเพื่อขึ้น Production ได้

3.4 โครงการ Bulk Open Account

3.4.1 การพัฒนา (Development)

3.4.1.1 ความต้องการของผู้ใช้งาน

- สร้างผลิตภัณฑ์บัญชีใหม่เพื่อเป็นทางเลือกให้ลูกค้าได้เปิดใช้บริการ และรองรับการทำงานกับระบบ Core Bank ดังนี้

- SCB Feel Free Savings Account
- SCB Feel Free Debit Card

- ผลิตภัณฑ์ต่าง ๆ จะมีเงื่อนไขในการเปิดบัญชีแตกต่างกันไป

- ออกแบบและพัฒนาให้โปรแกรม Bulk Detail Input , Bulk Detail Capture และ Bulk CreateControl ทำงานร่วมกันได้

ระยะเวลา งาน	ส.ค.				ก.ย.				ต.ค.				พ.ย.			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
ประชุมงาน			■						■							
วางแผน/ออกแบบ			■						■							
พัฒนาโปรแกรม				■	■	■	■	■	■	■						
ทดสอบการใช้งาน					■	■	■	■	■	■	■	■				
นำไปใช้งานจริง													■			

ตารางที่ 3.5 แผนการพัฒนาโครงการ Bulk Open Account

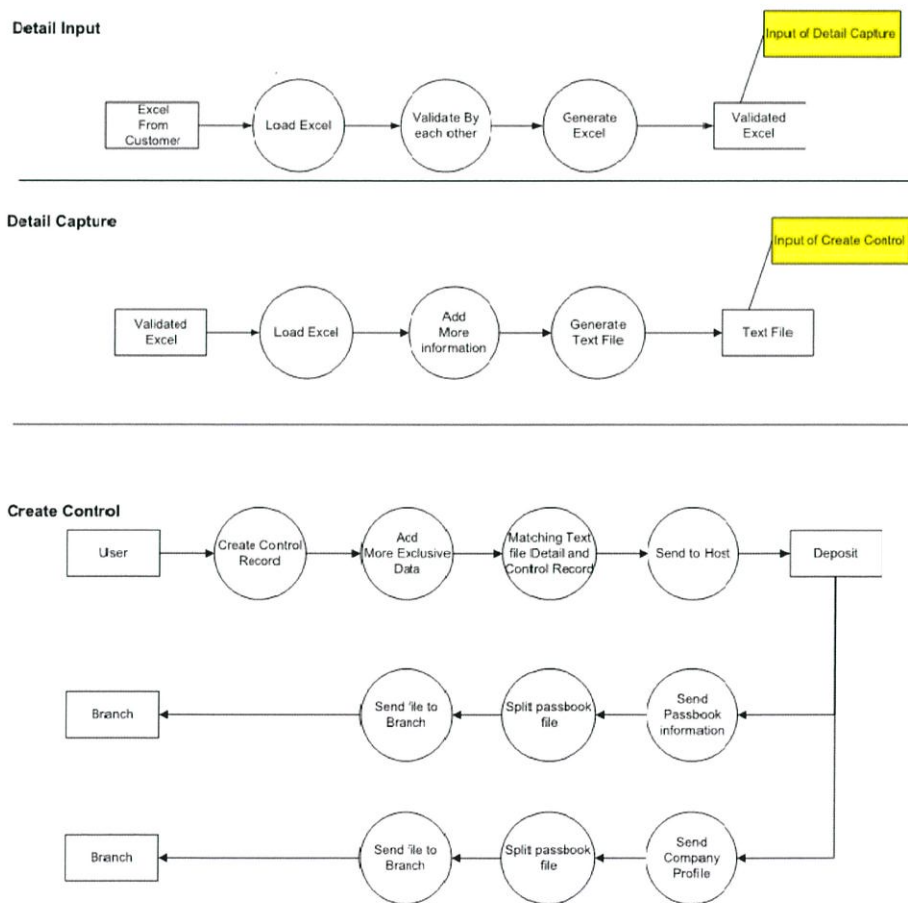
3.4.1.2 วางแผนและออกแบบการพัฒนาโปรแกรม

Development เป็นส่วนการพัฒนาโปรแกรม Bulk Open Account ซึ่งใช้โปรแกรม Visual Basic 6.0 ในการพัฒนา ความต้องการของผู้ใช้งานคือ ต้องการเพิ่มผลิตภัณฑ์ใหม่เข้าไปโดยใช้ชื่อว่า SCB Feel Free Savings Account และ SCB Feel Free Debit Card โดยมีเงื่อนไขต่าง ๆ ตามความต้องการของผู้ใช้ที่ลูกค้าจะต้องทำตาม โดยโปรแกรม Bulk Open Account จะแบ่งย่อยออกเป็น 3 โปรแกรม คือ

1. Bulk Detail Input เมื่อมีการเพิ่มผลิตภัณฑ์ จึงต้องเพิ่มข้อมูลจำพวกรหัสบัญชี ATM เข้าไปในฐานข้อมูลที่เป็น Microsoft Access ของโปรแกรมนี้ด้วย เพื่อให้โปรแกรมทำงานได้ต่อเนื่องอย่างมีประสิทธิภาพ แต่ไม่ต้องทำการแก้โปรแกรมแต่อย่างใด

2. Bulk Detail Capture ทำการเพิ่มข้อมูลจำพวกรหัสบัญชี ATM เข้าไปในฐานข้อมูลที่เป็น Microsoft Access ของโปรแกรมนี้ด้วย เพื่อให้โปรแกรมทำงานได้ต่อเนื่องอย่างมีประสิทธิภาพ แต่ไม่ต้องทำการแก้โปรแกรมแต่อย่างใด

3. Bulk Create Control พัฒนาโดยจะเพิ่มข้อมูลบัตรเดบิตชนิดใหม่ และผลิตภัณฑ์ชนิดใหม่ เข้าไปในฐานข้อมูลที่เป็น Microsoft Access ของโปรแกรมนี้ จากนั้นทำการเขียนโปรแกรมเพิ่มใน Visual Basic 6.0 โดยใช้ภาษา Visual Basic เพื่อทำการเพิ่มผลิตภัณฑ์ใหม่ และเงื่อนไขต่าง ๆ ในการเปิดผลิตภัณฑ์นั้นเพื่อให้โปรแกรมทำงานได้ต่อเนื่องอย่างมีประสิทธิภาพ



รูปที่ 3.2 กระบวนการทำงานของ Bulk Open Account

กระบวนการทำงานของโปรแกรม Bulk Open Account ที่แบ่งย่อยออกมา 3 โปรแกรมนั้น จะทำงานร่วมกันโดย โปรแกรม Bulk Detail Input รับไฟล์ Excel จากโปรแกรม Bulk Payroll เพื่อทำการตรวจสอบข้อมูลของลูกค้า และสร้าง Excel ใหม่ที่ข้อมูลถูกต้องตามฟอร์มเมท จากนั้นนำไฟล์ Excel ที่ได้มาทำการตรวจสอบ และเพิ่มข้อมูลลูกค้าอีกครั้งที่โปรแกรม Bulk Detail Capture เพื่อสร้างไฟล์ Text และสุดท้ายนำไฟล์ Text ไปเข้าโปรแกรม Bulk Create Control เพื่อทำการเพิ่ม Exclusive Data และบันทึกข้อมูลเพื่อส่งไปยัง Host และทำการส่งต่อไปยัง Core Bank เพื่อทำรายการต่อไป

เมื่อพัฒนาโปรแกรมเสร็จแล้วจะต้องทำ Unit Test ซึ่งคือการทดสอบด้วยตัวของผู้เขียนโปรแกรมเอง ว่าโปรแกรมทำงานถูกต้องตามความต้องการของผู้ใช้งานแล้วหรือไม่ เมื่อทำงานถูกต้องแล้วจึงจะส่งให้ทีมทดสอบทำการทดสอบโปรแกรม

3.4.2 การทดสอบ (Testing)

Testing จะเกิดเมื่อมีการเปลี่ยนแปลง หรือพัฒนาโปรแกรมจากส่วนของ Development เสร็จเรียบร้อยแล้ว จะส่งขึ้นมายังทีม System Integration Test (SIT) เพื่อทดสอบว่าระบบต่าง ๆ ทำงานได้ตรงตามความต้องการของผู้ใช้หรือไม่ เมื่อทีม SIT ทำการทดสอบเสร็จแล้ว ทีม Development จะต้องทำการเปิด Change เพื่อขอ Deploy ไปยังบนระบบ User Acceptance Testing (UAT) เพื่อให้ทางทีม UAT ทำการทดสอบโปรแกรม ซึ่งถ้าตรวจสอบแล้วพบว่ามีปัญหา หรือมีสิ่งที่จะต้องแก้ไข จะถูกส่งกลับไปยังส่วนของ Development อีกครั้ง เพื่อทำการแก้ไข แต่ถ้าไม่มีสิ่งที่จะต้องแก้ไข จึงจะส่งไปยังส่วนถัดไป

3.4.3 การนำไปใช้งานจริง (Production)

Production เป็นส่วนที่เมื่อการพัฒนาโปรแกรม หรือการเปลี่ยนแปลงทุกอย่างเสร็จสิ้น โดยผ่านการทดสอบจากทีมทดสอบ ทั้งทีม SIT และ UAT แล้ว จะทำการเปิด Change เพื่อขออนุญาตติดตั้งไปที่ส่วนของ Production เพื่อทำการใช้งานจริงในระบบ เนื่องจากเป็นขั้นตอน UAT และ Production จึงต้องมี QA (Quality Assurance) หรือ ผู้ที่รับประกันคุณภาพงาน อนุมัติเพื่อที่จะทำการติดตั้งเข้าระบบ ซึ่งถ้ายังเป็น Production แล้วจะต้องผ่านฝ่าย IT Security ด้วยเพื่อทำการตรวจสอบว่าถูกต้องหรือมีช่องโหว่หรือไม่ ถ้ามีช่องโหว่ โปรแกรมจะไม่ถูกอนุมัติเพื่อขึ้น Production แต่ถ้าผ่านการทดสอบแล้วจึงจะสามารถขออนุญาตติดตั้งโปรแกรมเพื่อขึ้น Production ได้

3.5 โปรเจกต์ Bulk Print

3.5.1 การพัฒนา (Development)

3.5.1.1 ความต้องการของผู้ใช้งาน

- สร้างฟังก์ชันการพิมพ์สมุดบัญชีและการ์ดลายเซ็นเพื่อรองรับผลิตภัณฑ์ใหม่ ดังนี้

- SCB Feel Free Savings Account
- SCB Feel Free Debit Card

- ผลิตภัณฑ์ต่าง ๆ จะมีเงื่อนไขในการเปิดบัญชีแตกต่างกันไป

ระยะเวลา งาน	ส.ค.				ก.ย.				ต.ค.				พ.ย.			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
ประชุมงาน		■			■											
วางแผน/ออกแบบ		■			■											
พัฒนาโปรแกรม		■	■	■		■										
ทดสอบการใช้งาน			■	■		■	■	■								
นำไปใช้งานจริง									■							

ตารางที่ 3.6 แผนการพัฒนาโปรเจกต์ Bulk Print

3.5.1.2 วางแผนและออกแบบการพัฒนาโปรแกรม

Development เป็นส่วนการพัฒนาโปรแกรม Bulk Print ซึ่งใช้โปรแกรม Visual Basic 6.0 ในการพัฒนา ความต้องการของผู้ใช้งานคือ ต้องการพิมพ์สมุดบัญชีและการ์ดลายเซ็นผลิตภัณฑ์ใหม่เข้าไปโดยใช้ชื่อว่า SCB Feel Free Savings Account จากนั้นทำการเพิ่มข้อมูลในฐานข้อมูล ที่ Microsoft Access เมื่อพัฒนาโปรแกรมเสร็จแล้วจะต้องทำ Unit Test ซึ่งคือการทดสอบด้วยตัวของผู้เขียนโปรแกรมเอง ว่าโปรแกรมทำงานถูกต้องตามความต้องการของผู้ใช้งานแล้วหรือไม่ เมื่อทำงานถูกต้องแล้วจึงจะส่งให้ทีมทดสอบทำการทดสอบโปรแกรม

3.5.2 การทดสอบ (Testing)

Testing จะเกิดเมื่อมีการเปลี่ยนแปลง หรือพัฒนาโปรแกรมจากส่วนของ Development เสร็จเรียบร้อยแล้ว จะส่งขึ้นมายังทีม System Integration Test (SIT) เพื่อทดสอบว่าระบบต่าง ๆ ทำงานได้ตรงตามความต้องการของผู้ใช้หรือไม่ เมื่อทีม SIT ทำการทดสอบเสร็จแล้ว ทีม Development จะต้องทำการเปิด Change เพื่อขออนุญาตติดตั้งไปยังบนระบบ User Acceptance Testing (UAT) เพื่อให้ทางทีม UAT ทำการทดสอบโปรแกรม ซึ่งถ้าตรวจสอบแล้วพบว่ามีปัญหา หรือมีสิ่งที่จะต้องแก้ไข จะถูกส่งกลับไปยังส่วนของ Development อีกครั้ง เพื่อทำการแก้ไข แต่ถ้าไม่มีสิ่งที่จะต้องแก้ไข จึงจะส่งไปยังส่วนถัดไป

3.2.3 การนำไปใช้งานจริง (Production)

Production เป็นส่วนที่เมื่อการพัฒนาโปรแกรม หรือการเปลี่ยนแปลงทุกอย่างเสร็จสิ้น โดยผ่านการทดสอบจากทีมทดสอบ ทั้งทีม SIT และ UAT แล้ว จะทำการเปิด Change เพื่อขออนุญาตติดตั้งไปที่ส่วนของ Production เพื่อทำการใช้งานจริงในระบบ เนื่องจากเป็นขั้นตอน UAT และ Production จึงต้องมี QA (Quality Assurance) หรือ ผู้ที่รับประกันคุณภาพงาน อนุมัติเพื่อที่จะทำการติดตั้งเข้าระบบ ซึ่งถ้ายังเป็น Production แล้วจะต้องผ่านฝ่าย IT Security ด้วยเพื่อทำการตรวจสอบว่าถูกต้องหรือมีช่องโหว่หรือไม่ ถ้ามีช่องโหว่ โปรแกรมจะไม่ถูกอนุมัติเพื่อขึ้น Production แต่ถ้าผ่านการทดสอบแล้วจึงจะสามารถขออนุญาตติดตั้งโปรแกรมเพื่อขึ้น Production ได้

บทที่ 4 ผลการวิจัย

ในส่วนของผลการวิจัยนั้นผู้วิจัยได้แบ่งผลการวิจัยออกเป็นสองส่วน คือ การพัฒนาโปรแกรมให้สอดคล้องกับความต้องการของผู้ใช้ และการแก้ปัญหาโปรแกรมเมื่อผู้ใช้งานเจอปัญหาเข้ามา

4.1 การพัฒนาโปรแกรมให้สอดคล้องกับความต้องการของผู้ใช้

การพัฒนาโปรแกรมในแต่ละครั้ง จะขึ้นอยู่กับความต้องการของผู้ใช้งาน โดยจะมีการประชุมในการแจกแจงปัญหา หรือการเพิ่มผลกำไรให้กับบริษัท เมื่อประชุมเสร็จแล้วจะมีเอกสาร User Request มาว่าต้องการปรับเปลี่ยน หรือเพิ่มเติมโปรแกรมในส่วนไหนบ้าง ซึ่งในงานวิจัยชิ้นนี้จะแบ่งผลการวิจัยออกเป็นอีก 5 โปรเจกต์ ดังนี้

4.1.1 โปรเจกต์ RIA

มีการเพิ่มหน้าจอแสดงผลสำหรับ Provider ใหม่ หรือ RIA Receive เพื่อบริการลูกค้าเมื่อมีการโอนเงินต่างประเทศ โดยช่องต่าง ๆ จะแสดงเมื่อมีการส่งค่า Reference Number ผ่าน Web Service FRC ไปยัง Provider RIA เพื่อให้ RIA ส่งข้อมูลต่าง ๆ ของลูกค้ากลับมาแสดงบนหน้าจอ RIA Receive

The screenshot shows a software window titled "RIA Receive". It contains several sections for data entry:

- RIA Information:** Includes a "Reference Number" field and an "Inquiry" button.
- Receiver Information:** Contains fields for First Name, Last Name, District, Address, Sub District, State (dropdown), Tel, City, ID State (dropdown), Zipcode, Country (dropdown), Occupation (dropdown), ID Card No. (radio), Passport No. (radio), Alien No. (radio), ID Country (dropdown), Date of Birth, Company's Name/Address, and Country of Birth (dropdown).
- Sender Information:** Contains fields for First Name, Last Name, Tel, Address, City, State, Zipcode, and Country.
- Transaction Details:** Includes fields for Test Question, Test Answer, Message Field, Transaction Status, Send Date, Reference No., Recieve Currency, Recieve Amount, Originating Country, Original Send Currency, Original Exchange Rate, Original Send Amount, Original Send Currency, Fee Code, Fee Currency, Fee Amount, Fee Rate, and Fee Bahk.

At the bottom right, there are "Receive" and "Exit" buttons.

รูปที่ 4.1 ตัวอย่างหน้าจอ RIA Receive

จากนั้นเมื่อทำรายการจ่ายเงินเสร็จเรียบร้อยแล้ว ระบบจะทำการพิมพ์ใบเสร็จรับเงินให้กับผู้รับเงิน และบันทึกข้อมูลลงฐานข้อมูล เป็นอันเสร็จสิ้นโปรแกรม



รูปที่ 4.2 ตัวอย่างใบเสร็จจ่ายเงินของ RIA

REPORT NO. DFEB0098 THE SIAM COMMERCIAL BANK PCL PAGE : Page 7 of 6
REPORT TO : SUMMARY REPORT - OFFLINE RUN DATE : 01-08-2017 TIME: 16:37:52
AS AT 01-08-2017 EXPIRE DATE : 01-08-2027

BILL REF. NO.	CUSTOMER NAME	PROD BR/ECC/AUTH NO./MG REF. CHQ./CARD NO.	ENT. DATE	ENT. TIME	APP. DATE	APP. TIME	STAT	RATE	BAHT	FEE TYPE	FEE AMT	USER ID NET BAHT
01001	DAFAQE ERTWERT	0137316099901001	09/09/2559	19:32:43	01/01/0001	01:01:01	AUD	1,000.00	30,740.0000	CT	30,740.00	27991
TOTAL OF BILL - PAID												30,740.00
01002	CUSTOMER NAME	0137316099901002	09/09/2559	19:37:37	01/01/0001	01:01:01	AED	4,000.00	8,850.0000	CT	35,400.00	27991
TOTAL OF BILL - PAID												35,400.00
01003	CUSTOMER NAME	0137316099901003	09/09/2559	19:41:41	01/01/0001	01:01:01	JPY	20,000.00	6,377.0000	CT	7,554.00	27991
TOTAL OF BILL - PAID												7,554.00
01004	CUSTOMER NAME	0137316099901004	09/09/2559	19:53:41	01/01/0001	01:01:01	CAD	100.00	32,830.0000	CT	3,283.00	27991
TOTAL OF BILL - PAID												3,283.00

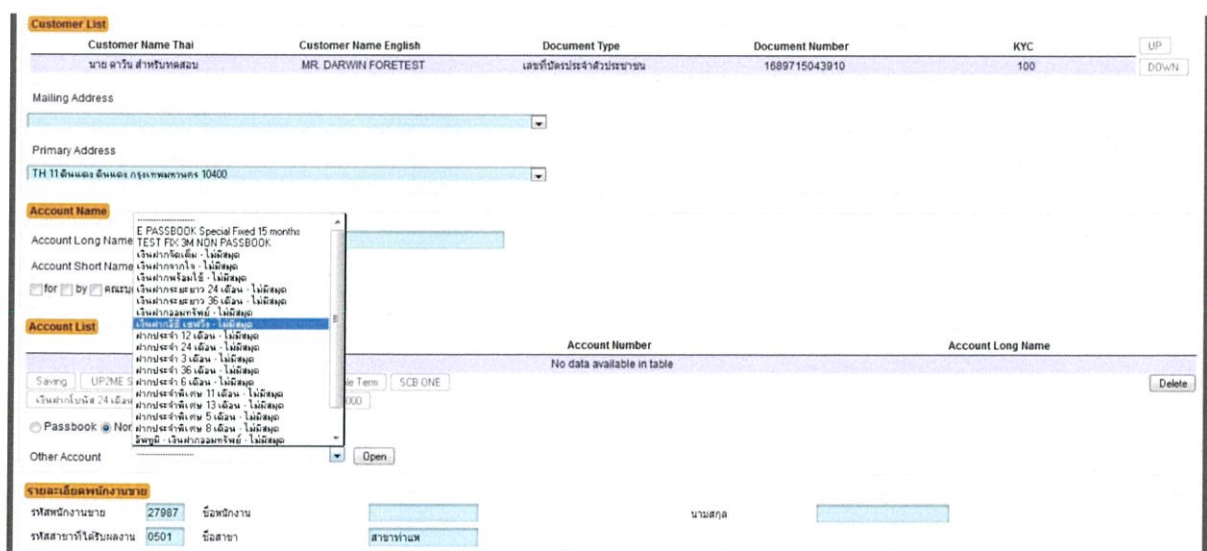
STOCK OF T/CHOS IN CASHIER :-											
CUR	BRAND	DENOM	FROM	SERIAL NO.	TO	QTY	AMOUNT	TOTAL			
TOTAL BY BRAND											
TOTAL BY CURRENCY											

รูปที่ 4.3 ตัวอย่าง Summary Report

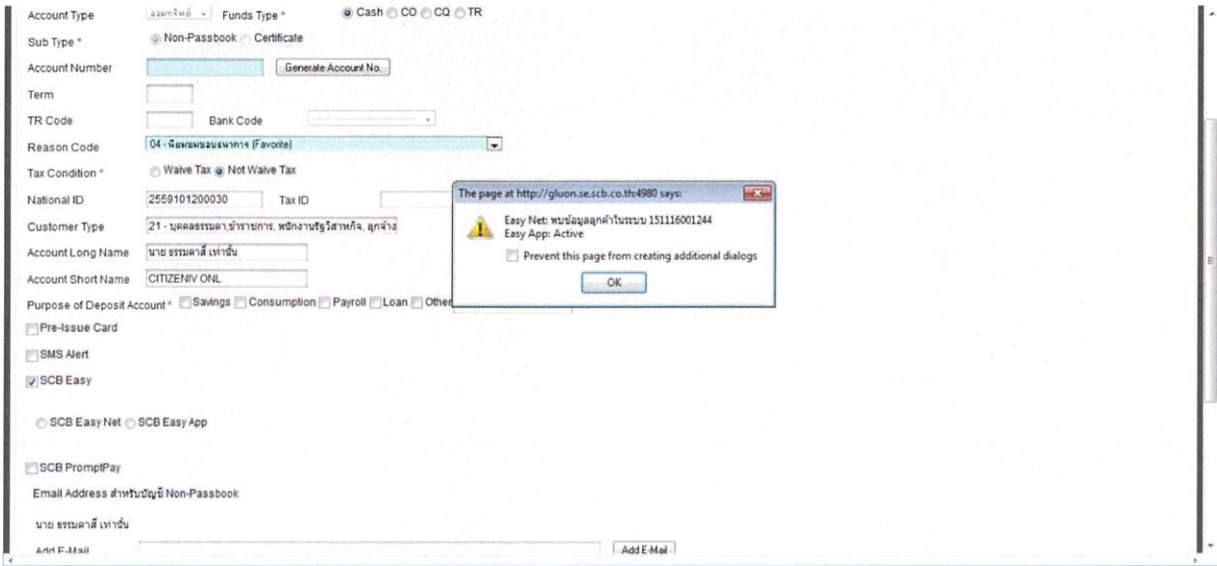
4.1.2 โปรเจกต์ SingleApp

โปรแกรมจะมีผลิตภัณฑ์เพิ่มหลัก ๆ 3 ผลิตภัณฑ์ คือ เงินฝากอีซีเซฟวิ้ง เงินฝาก SCB FeelFree และ เงินฝากเพื่อการศึกษา แต่จะแบ่งย่อยได้ ดังนี้

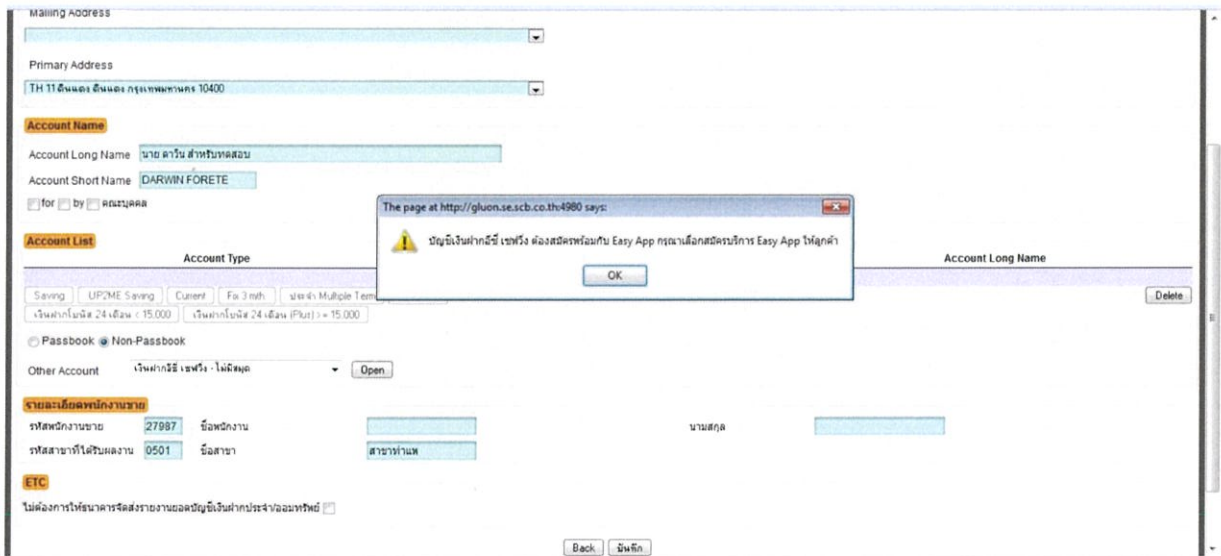
- เงินฝากอีซีเซฟวิ้ง - ไม่มีสมุด
- เงินฝาก SCB Feel Free
- เงินฝาก SCB Feel Free – ไม่มีสมุด
- เงินฝากเพื่อการศึกษา 24 เดือน
- เงินฝากเพื่อการศึกษา 24 เดือน – ไม่มีสมุด
- เงินฝากเพื่อการศึกษา 36 เดือน
- เงินฝากเพื่อการศึกษา 36 เดือน – ไม่มีสมุด



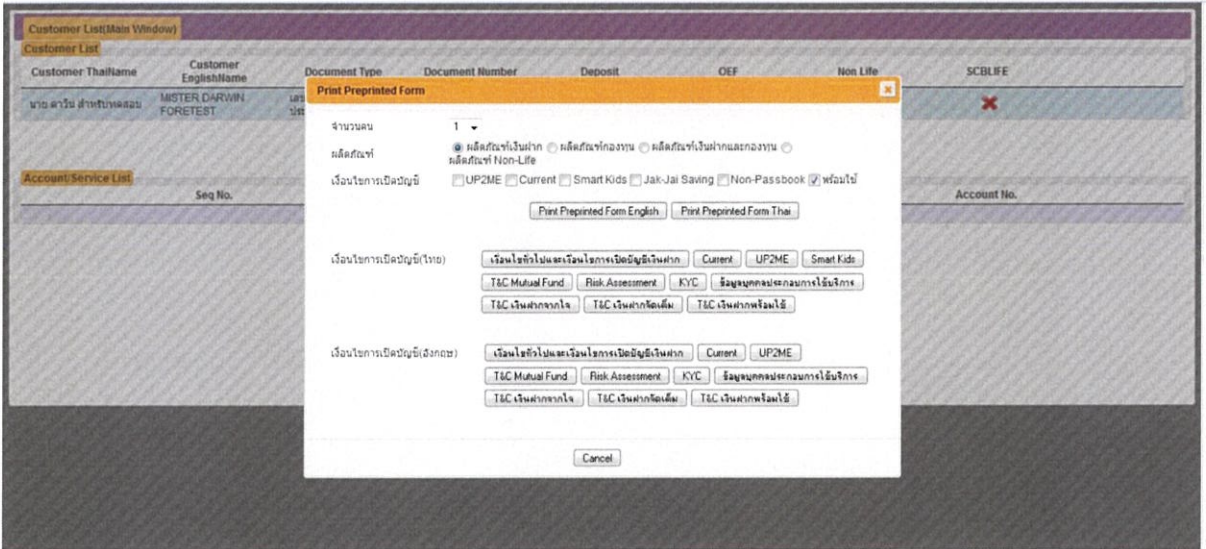
รูปที่ 4.4 ตัวอย่าง โปรแกรม SingleApp เมื่อเพิ่มผลิตภัณฑ์ อีซี เซฟวิ้ง-ไม่มีสมุด เข้าไปใน Database



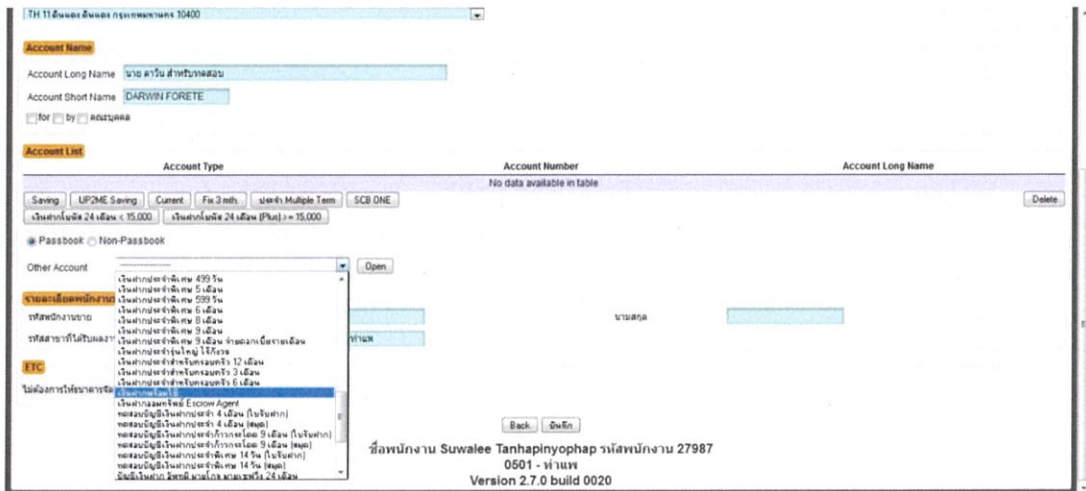
รูปที่ 4.5 ฟังก์ชันตรวจสอบสถานะ Easy App และ Easy Net เมื่อเลือกผลิตภัณฑ์อีซี-เซฟวิ่ง



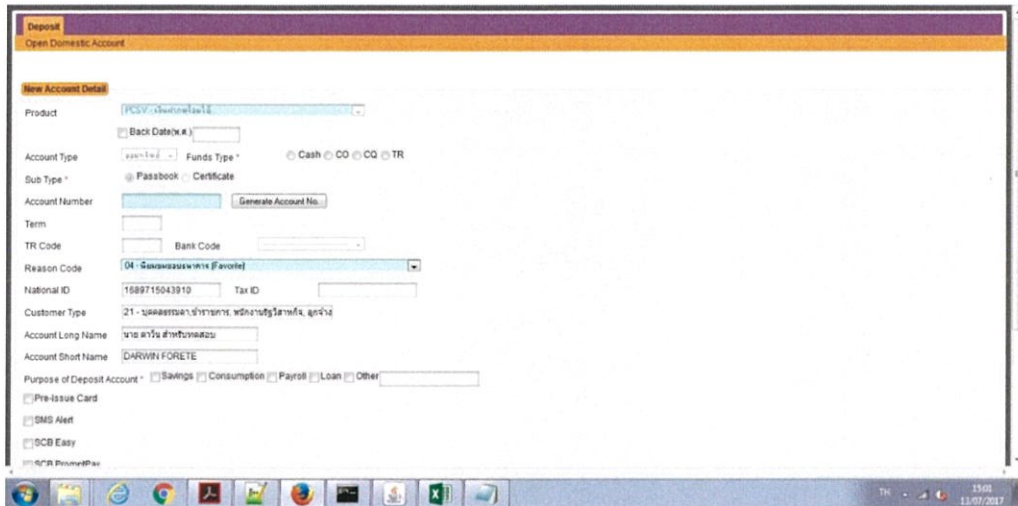
รูปที่ 4.6 ตัวอย่างเงื่อนไขของการเปิดผลิตภัณฑ์อีซี เซฟวิ่ง-ไม่มีสมุด



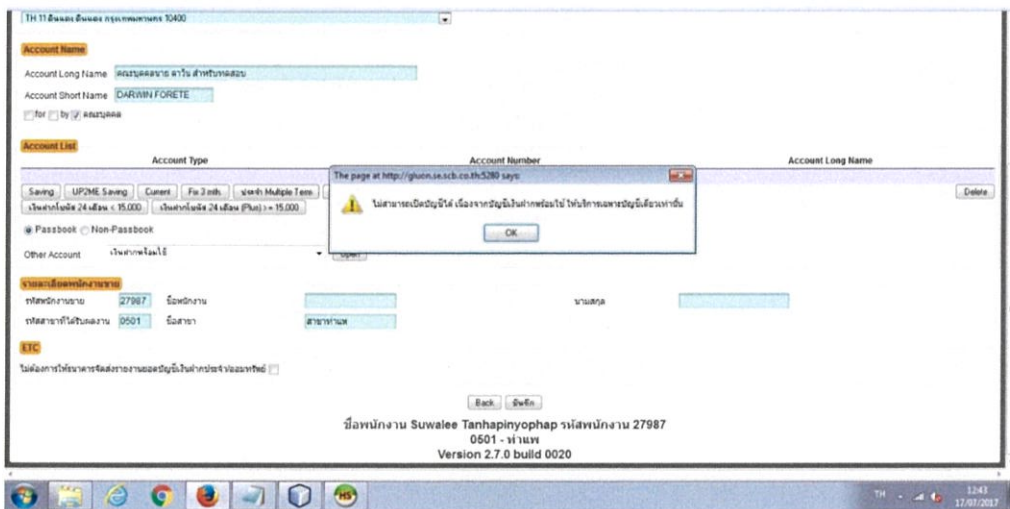
รูปที่ 4.7 ฟังก์ชัน Preprinted Form เพิ่มผลิตภัณฑ์เงินฝากพร้อมใช้ หรือ SCB Feel Free Savings



รูปที่ 4.8 ตัวอย่างผลิตภัณฑ์เงินฝากพร้อมใช้ หรือ SCB Feel Free Savings



รูปที่ 4.9 ตัวอย่างหน้าจอแสดงผลเมื่อเลือกผลิตภัณฑ์เงินฝากพร้อมใช้ หรือ SCB Feel Free Savings



รูปที่ 4.10 เงื่อนไขไม่อนุญาตให้เปิดบัญชีร่วมของผลิตภัณฑ์เงินฝากพร้อมใช้ หรือ SCB Feel Free Savings

นอกจากนั้นแต่ละผลิตภัณฑ์ จะมีระยะเวลาและเงื่อนไขในการเปิดหรือสมัครใช้บัญชี โดยแต่ละผลิตภัณฑ์จะมีเงื่อนไข ดังนี้

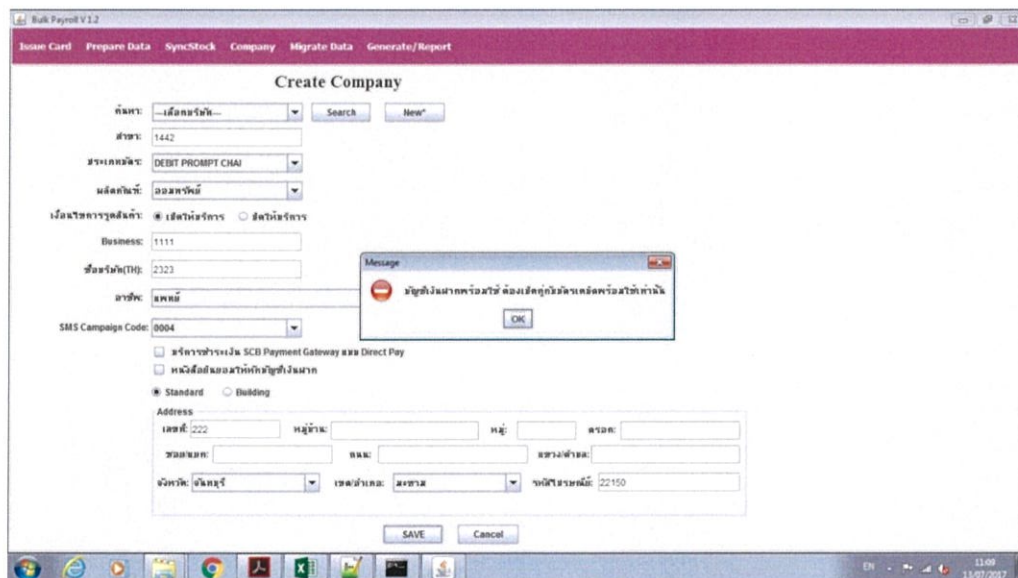
- เงินฝากอีซีเซฟวิ่ง – ไม่มีสมุด มีเงื่อนไขที่สำคัญ คือ จะต้องมีการสมัครบริการ Easy App ด้วย หรือเป็นผู้ที่สมัครบริการ Easy App อยู่แล้ว
- เงินฝาก SCB Feel Free มีเงื่อนไขที่สำคัญ คือ จะต้องสมัครพร้อมกับ SCB Feel Free Debit Card

- เงินฝากเพื่อการศึกษา 24 เดือน มีเงื่อนไขที่สำคัญ คือ เปิดให้บุคคลที่มีอายุอยู่ในช่วง 7-18 ปี ฝากในระยะเวลา 24 เดือน

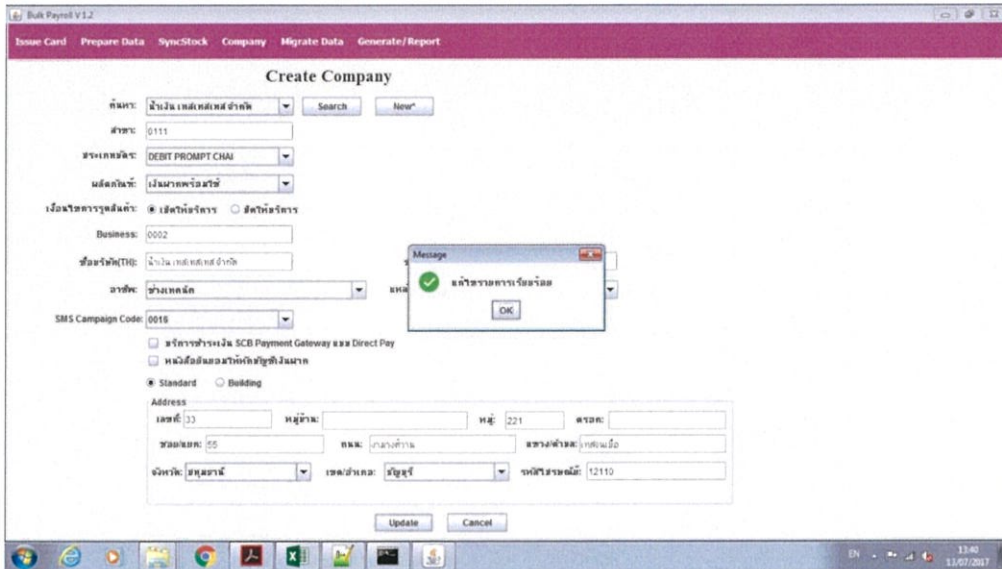
- เงินฝากเพื่อการศึกษา 36 เดือน มีเงื่อนไขที่สำคัญ คือ เปิดให้บุคคลที่มีอายุอยู่ในช่วง 7-18 ปี ฝากในระยะเวลา 36 เดือน

4.1.3 โปรเจกต์ Bulk Payroll

มีผลิตภัณฑ์ SCB Feel Free Savings และ Debit SCB Feel Free ให้ลูกค้าได้เลือกใช้ โดยจะมีเงื่อนไขในการเปิดผลิตภัณฑ์ คือต้องเปิด SCB Feel Free Savings Account พร้อมกับ Debit SCB Feel Free ระบบจึงจะยอมให้เปิดใช้บริการได้



รูปที่ 4.11 ตัวอย่างเงื่อนไขในการเปิดผลิตภัณฑ์ใหม่ของโปรแกรม Bulk Payroll

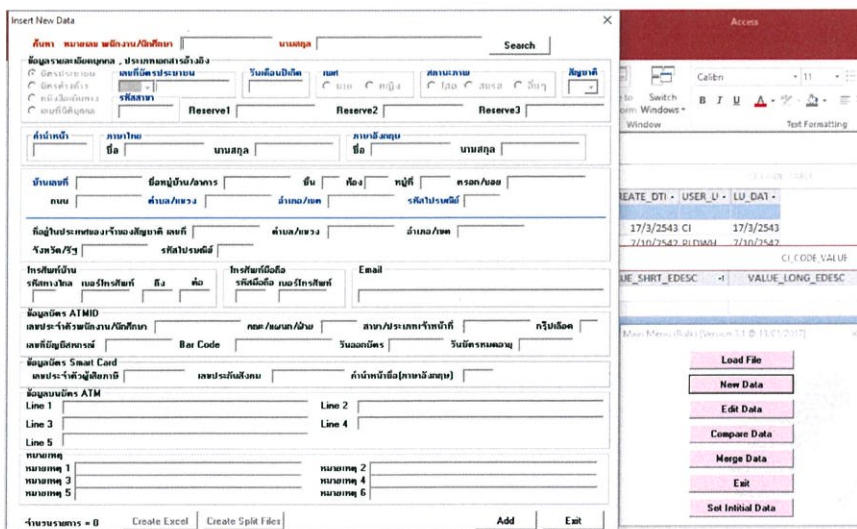


รูปที่ 4.12 ตัวอย่างเงื่อนไขผลิตภัณฑ์เงินฝากพร้อมใช้เมื่อทำการเปิดบัญชีถูกต้อง

4.1.4 โปรแกรม Bulk Open Account

4.1.4.1 โปรแกรม Bulk Detail Input

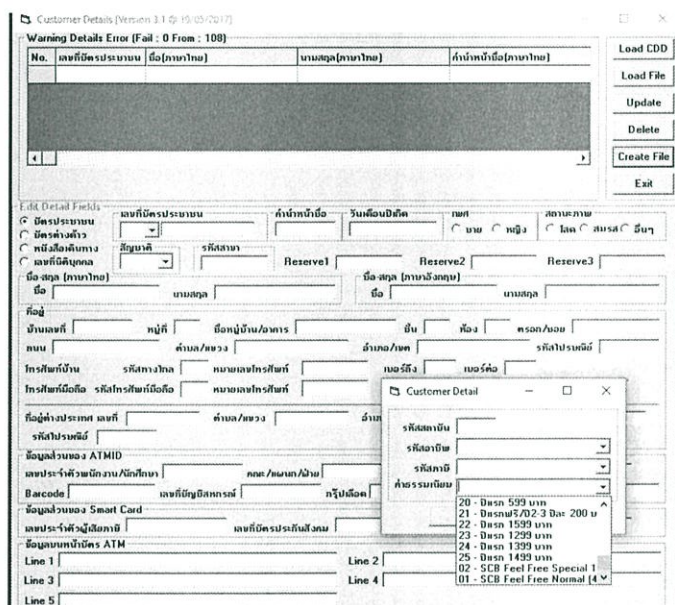
มีการเพิ่มผลิตภัณฑ์ SCB Feel Free Savings ให้ลูกค้าได้เลือกใช้งาน และสามารถทำงานร่วมกับระบบอื่นได้อย่างสอดคล้อง



รูปที่ 4.13 ตัวอย่าง โปรแกรม Bulk Detail Input

4.1.4.2 โปรแกรม Bulk Detail Capture

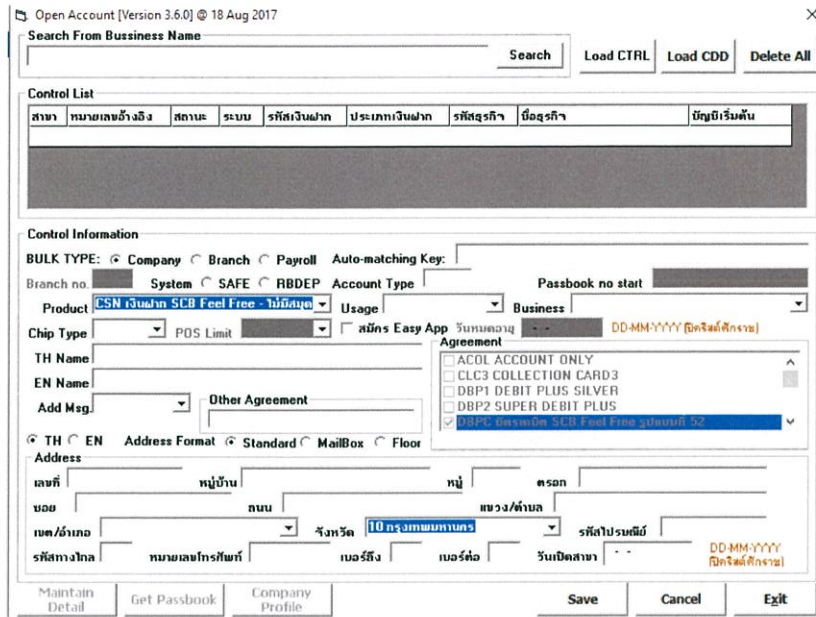
มีการเพิ่มผลิตภัณฑ์ SCB Feel Free Savings ให้ลูกค้าได้เลือกใช้งาน โดยที่ช่องค่าธรรมเนียมจะมี SCB Feel Free Special และ SCB Feel Free Normal ให้ลูกค้าเลือก และสามารถทำงานร่วมกับระบบอื่นได้อย่างสอดคล้อง



รูปที่ 4.14 ตัวอย่างโปรแกรม Bulk Detail Capture ที่ทำการเพิ่มผลิตภัณฑ์แล้ว

4.1.4.2 โปรแกรม Bulk CreateControl

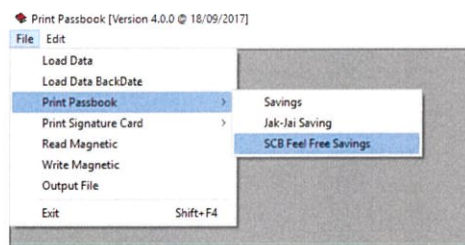
มีผลิตภัณฑ์ SCB Feel Free Savings และ Debit SCB Feel Free ให้ลูกค้าได้เลือกใช้ โดยจะมีเงื่อนไขในการเปิดผลิตภัณฑ์ คือต้องเปิด SCB Feel Free Savings Account พร้อมกับ Debit SCB Feel Free ระบบจึงจะยอมให้เปิดใช้บริการได้ ซึ่ง SCB Feel Free Savings Account จะแบ่งออกเป็นแบบมีสมุดกับไม่มีสมุด แต่ทั้ง 2 อย่างจะต้องเปิดคู่กับ SCB Feel Free Debit Card เท่านั้น



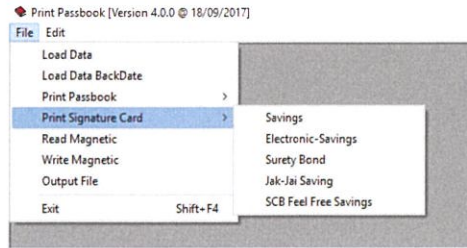
รูปที่ 4.15 ตัวอย่างผลิตภัณฑ์ เงินฝาก SCB Feel Free ของโปรแกรม Bulk CreateControl

4.1.5 โปรแกรม Bulk Print

โปรแกรม Bulk Print มีผลิตภัณฑ์ SCB Feel Free Savings เพิ่มในฟังก์ชัน Print Passbook และ Print Signature Card



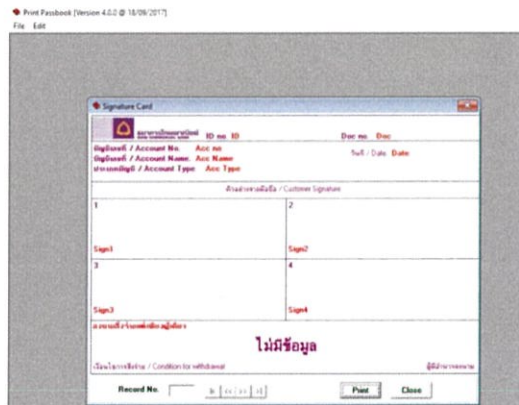
รูปที่ 4.16 ตัวอย่างการเพิ่มผลิตภัณฑ์ SCB Feel Free Saving เข้าไปในฟังก์ชัน Print Passbook



รูปที่ 4.17 ตัวอย่าง การเพิ่มผลิตภัณฑ์ SCB Feel Free Saving เข้าไปในฟังก์ชัน Print Signature Card



รูปที่ 4.18 ตัวอย่างฟังก์ชัน Print Passbook ของผลิตภัณฑ์ SCB Feel Free Savings

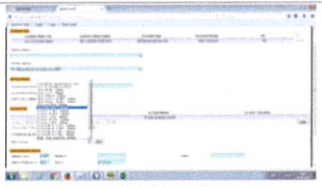


รูปที่ 4.19 ตัวอย่างฟังก์ชัน Print Signature Card ของผลิตภัณฑ์ SCB Feel Free Savings

Unit Test คือ การทดสอบโปรแกรมเมื่อทำการเขียนโปรแกรมเสร็จแล้ว โดยจะแสดงรายละเอียดต่าง ๆ ดังรูป ชิตแรก จะเป็น UR ที่เราทำการแก้ไขชิตที่สอง เป็นข้อมูลต่าง ๆ ของผู้จัดทำ และผู้รับรองผล ชิตที่สามจะเป็น Data Test เพื่อนำมาใช้ในการทดสอบ ชิตที่สี่ คือ รายละเอียดการทดสอบ จะแบ่งเป็น

- Project คือ ชื่อโปรเจกต์
- Test Case No. คือ ชื่อย่อสำหรับการทดสอบในแต่ละกรณี
- Test Case Name คือ ชื่อกรณีในการทดสอบ

- Test Case Objective / focus area คือ จุดที่สนใจในการทดสอบ
- Prerequisites คือ ข้อกำหนดเบื้องต้นในการทดสอบ ซึ่งจะบอก Environment ที่ใช้ทดสอบ และการจัดเตรียมข้อมูลในการทดสอบ
- Test Step คือ ลำดับขั้นตอนในการทดสอบ
- Action คือ รายละเอียดขั้นตอนในการทดสอบ
- Expected Result คือ ผลลัพธ์ที่คาดหวังจากการทดสอบ
- Actual Result คือ ผลลัพธ์ที่ได้จากการทำตามขั้นตอนจริง
- Pass or Fail คือ ผลลัพธ์แสดงว่าโปรแกรมที่เขียนขึ้นตรงตาม UR หรือไม่ ถ้าตรง คือ Pass แต่ถ้าไม่ตรง คือ Fail
- Defect No. คือ เลขที่ทางทีมทดสอบสร้างขึ้น เมื่อตรวจพบว่าโปรแกรมไม่ตรงตาม UR

Test Step	Action	Expected Result	Actual Result	Pass or Fail	Defect No.
1	Enter to http //				
2	Search Customer > Name				
3	Click Sale > Deposit > Choose Non Passbook > Other Account disdown	Show new product ใหม่ใหม่ใหม่ ใหม่ใหม่		PASSED	

รูปที่ 4.20 ตัวอย่างเอกสาร Unit Test

เอกสาร Implement Playbook คือ เอกสารที่ใช้ตอนเปิด Change เพื่อทำการทดสอบบนระบบ UAT และ Production จัดทำเพื่อให้ผู้ที่มีหน้าที่ลงโปรแกรมอ่านและทำตามขั้นตอน รวมถึงปฏิบัติตามเวลาอย่างเคร่งครัด เพื่อให้ไม่เกิดปัญหาในภายหลัง แต่ถ้าผู้ที่มีหน้าที่ Implement สงสัยหรือมีปัญหาที่ขั้นตอนไหนสามารถติดต่อผู้ที่เกี่ยวข้องเพื่อทำการสอบถามได้

URPB no. (Mandatory)	Application (Mandatory)	Task no. (Mandatory)	Prerequisite (Optional)	Task name (Mandatory)	Responsible name (Mandatory)	Start date (Mandatory)	Start time (Mandatory)	Finish date (Mandatory)	Finish time (Mandatory)	Remark (Optional)	Status (Optional)	Application (Optional in case more than 1 App in Playbook)
Implement Playbook												
Application: Implement												
URPB List:												
UR	Single App	1		Rollback Activity Single App: Request change request in ECCD	Name (Tel No.)	11/10/2017	09:00	11/10/2017	17:00			Single App
Rollback Activity												
		2		Migrate user file Single App: Backup the existing version both App & Database 1 Backup war file ผู้รวม 2 Deploy war file (app\war folder\war) 3 Deploy batch file to Server	Name (Tel No.)	11/10/2017	21:00	11/10/2017	21:30			Single App
		3		Restart Ova/ssh	Name (Tel No.)	11/10/2017	21:31	11/10/2017	22:00			
Checkpoint												
		4		Verify Application ตรวจสอบเข้าใช้งานบน OVA ที่นำส่งจาก ผู้รวม/DevOps Version	Name (Tel No.)	11/10/2017	22:00	11/10/2017	22:30			

รูปที่ 4.21 เอกสารขอเปิดการทดสอบระบบบน UAT และใช้จริงบน Production หน้า Implement

URPB no. (Mandatory)	Application (Mandatory)	Task no. (Mandatory)	Prerequisite (Optional)	Task name (Mandatory)	Responsible name (Mandatory)	Start date (Mandatory)	Start time (Mandatory)	Finish date (Mandatory)	Finish time (Mandatory)	Remark (Optional)	Status (Optional)	Application (Optional in case more than 1 App in f
Roll Back Playbook												
Application: Implement												
URPB List:												
UR	Single App	1		Rollback Activity Rollback Single App Program - Copy previous version to Production 1 Verify War File ส่งให้ DevOps ตรวจสอบ 2 Deploy war file จากข้อ 1 Deploy to Server	Name (Tel No.)							
Checkpoint												
		2		Verify Application ตรวจสอบเข้าใช้งานบน OVA ที่นำส่งจาก ผู้รวม/DevOps Version	Name (Tel No.)							

รูปที่ 4.22 เอกสารขอเปิดการทดสอบระบบบน UAT และใช้จริงบน Production หน้า Roll Back

4.2 การแก้ปัญหาโปรแกรมเมื่อผู้ใช้งานแจ้งปัญหาเข้ามา

เมื่อทำการวินิจฉัยปัญหาที่ผู้ใช้งานแจ้งเข้ามาแล้ว จะมีอยู่ 2 กรณี คือ

4.2.1 ไม่จำเป็นต้องแก้โปรแกรม

เมื่อผู้วิจัยดำเนินการมาจนถึงขั้นตอนที่แก้ไขปัญหาเสร็จแล้ว จะต้องทำการติดต่อประสานงานถึงผู้ใช้งานที่แจ้งปัญหาเข้ามา เพื่อให้ผู้ใช้งานลองทดสอบว่าระบบทำงานตามปกติแล้วหรือไม่ ถ้าทำงานได้ตามปกติแล้ว ผู้วิจัยจะต้องทำการปิด ใบคำร้องแจ้งโปรแกรมมีปัญหา และแจ้งต่อ Helpdesk เพื่อดำเนินการต่อไป

4.2.2 จำเป็นต้องแก้โปรแกรม

เมื่อผู้วิจัยดำเนินการแก้โปรแกรมจนถึงขั้นตอนที่ทำ Unit Test เสร็จแล้ว จะต้องทำเอกสาร Factsheet ต่อ ซึ่งเอกสาร Factsheet คือ เอกสารที่อธิบายรายละเอียดของปัญหานั้น ๆ สาเหตุ วิธีการแก้ไข และผู้ที่ดำเนินการแก้ไข จากนั้นจะถูกรับรองโดยผู้จัดการโครงการ จากนั้นทำการนำโปรแกรมขึ้นระบบเพื่อทำการทดสอบทั้งบนระบบ SIT และ UAT เมื่อทดสอบผ่านแล้วจึงจะนำขึ้นไปใช้จริงที่ Production

บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

ผู้วิจัยได้ดำเนินงานวิจัยเพื่อเพิ่มประสิทธิภาพของโปรแกรมต่าง ๆ ในระบบธนาคาร โดยแบ่งออกเป็น 5 โปรแกรม ซึ่งโปรแกรม RIA เป็นการเพิ่มผู้ให้บริการด้านการโอนเงินต่างประเทศเพื่อเพิ่มประสิทธิภาพให้กับโปรแกรมโอนเงินต่างประเทศของธนาคารไทยพาณิชย์ ส่วนอีก 4 โปรแกรม คือ SingleApp Bulk Payroll Bulk Open Account และ Bulk Print เป็นการเพิ่มผลิตภัณฑ์ใหม่เข้าไปในโปรแกรมเพื่อเป็นการเพิ่มประสิทธิภาพในการทำงานของระบบและช่วยเพิ่มผลกำไรให้กับธนาคารไทยพาณิชย์

เมื่อการดำเนินงานวิจัยเสร็จสิ้น และทำการวัดผลแล้ว พบว่าประสิทธิภาพในการทำงานของระบบการเปิดบัญชี และระบบโอนเงินต่างประเทศเพิ่มขึ้น โดยมีผลิตภัณฑ์ใหม่ให้ลูกค้าเลือกใช้บริการ และมีผู้ให้บริการโอนเงินต่างประเทศเจ้าใหม่ เป็นทางเลือกให้ลูกค้าใช้บริการ ทำให้เกิดกำไรให้กับธนาคารมากขึ้น รวมไปถึงมีการแก้ปัญหาที่มีอยู่เดิมแล้วของโปรแกรมให้ทำงานถูกต้องอย่างที่ควรเป็น ทำให้ไม่พบข้อบกพร่องของโปรแกรมที่ส่งผลเสียต่อระบบธุรกิจ ซึ่งการพัฒนาเหล่านี้นำไปสู่เป้าหมายที่ได้วางไว้ คือการเพิ่มประสิทธิภาพให้กับระบบการเปิดบัญชีของธนาคาร และระบบโอนเงินต่างประเทศ รวมถึงมีการลดการใช้ทรัพยากร บุคคล เวลา และค่าใช้จ่ายในด้านต่าง ๆ

5.2 ปัญหาที่พบในระหว่างการดำเนินงานวิจัย

5.2.1 ในการทดสอบโปรแกรมต่าง ๆ ต้องมีการทดสอบให้ครอบคลุมทุกกรณีทางธุรกิจ เพื่อป้องกันการเกิดข้อบกพร่องของโปรแกรม

5.2.2 ในการดำเนินงาน จะประกอบไปด้วยผู้เกี่ยวข้องหลายส่วน จึงทำให้วางแผนการดำเนินงานได้ยาก รวมถึงต้องมีการรอคอยในการทำงาน เพราะแต่ละแผนกมีงานที่รับผิดชอบหลายงาน

5.2.3 เมื่อผู้ใช้งานแจ้งปัญหาเข้ามา บางโปรแกรมไม่สามารถดู Log Production เองได้ จึงต้องแจ้ง SP ให้ส่ง Log Production มาให้เพื่อทำการ Investigate และแก้ปัญหาต่อไป

5.3 แนวทางการแก้ไข้ปัญหา

5.3.1 ในการทดสอบโปรแกรมเพื่อให้ครอบคลุมทุกกรณีทางธุรกิจนั้น ต้องมีการพิจารณาร่วมกับผู้ใช้ เนื่องจากผู้ใช้มีความรู้ในด้านธุรกิจจึงต้องมีการขอคำปรึกษาาร่วมด้วย

5.3.2 วางแผนกำหนดการดำเนินงานอย่างชัดเจน และต้องมีการวางแผนเพื่อเวลาเอาไว้ก่อน ถึงกำหนดการจริง เพื่อป้องกันการเกิดความล่าช้าในการดำเนินงาน

5.3.3 ทำการขอสิทธิในการดู Log Production เนื่องจากผู้ใช้งานแจ้งปัญหาเข้ามาบ่อย

เอกสารอ้างอิง

1. Watsan Homsin. 2015. **Visual Basic**. สืบค้นเมื่อ 20 กันยายน 2560, จาก
<http://marcuscode.com/lang/visual-basic>
2. ธนาคารแห่งประเทศไทย. 2015. **กฎหมายธนาคารแห่งประเทศไทย**. สืบค้นเมื่อ 8 กันยายน 2560, จาก
<https://www.bot.or.th/Thai/AboutBOT/LawAndRegulations/Pages/default.aspx>
3. John Resig. (2007). **Pro JavaScript Techniques**. 1st ed. United States of America: Apress.
4. สุดา เจริญมนตรี. (2013). **คู่มือเรียนเขียนโปรแกรมภาษา Java**. พิมพ์ครั้งที่ 2. กรุงเทพฯ: อดิษฐ์
พรีเมียร์.
5. Angela Stringfellow. 2017. **Web Application Architecture**. สืบค้นเมื่อ 7 กันยายน 2560, จาก
<https://stackify.com/web-application-architecture>
6. IBM. 2017. **IBM Db2 Database**. สืบค้นเมื่อ 27 กันยายน 2560, จาก
<https://www.ibm.com/analytics/us/en/db2>