

การวิเคราะห์และออกแบบระบบ BBL Portal โดยใช้ UML
(System Analysis and Design of the BBL Portal Using UML)

โดย

นายพรศักดิ์ สุทธธมมัตย์

รหัส 44067495



H002048

อาจารย์ที่ปรึกษา

รศ. บรรจง ปิยะธำรง

วัน เดือน ปี 25 มิ.ค. 2550

เลขทะเบียน 02048

เลขเรียกหนังสือ วท. พ 282 ก 2546

"ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล."

รายงานนี้เป็นส่วนหนึ่งของวิชาการโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาเทคโนโลยีสารสนเทศ
ภาคเรียนที่ 1 ปีการศึกษา 2546
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ชื่อหัวข้อ	การวิเคราะห์และออกแบบระบบ BBL Portal โดยใช้ UML
นักศึกษา	นายพรศักดิ์ สุนทรมาตย์
อาจารย์ที่ปรึกษา	รศ. บรรจง ปิยะธำรง
ระดับการศึกษา	วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2546

บทคัดย่อ

ตามที่ทางฝ่ายโครงการ Public Website ของธนาคารกรุงเทพได้จัดทำเว็บไซต์ เพื่อเป็นสื่อกลางในการติดต่อกับลูกค้าและกลุ่มเป้าหมายผ่านทางเครือข่ายอินเทอร์เน็ตมาในระยะหนึ่งแล้ว นั้น ในลำดับถัดไปเพื่อเป็นการเพิ่มรรถประโยชน์ให้กับเว็บไซต์ จึงได้มีการริเริ่มจัดทำในส่วน ของ “Bangkok Bank Portal (BBL Portal)” เพื่อเป็นการให้บริการกับลูกค้าและกลุ่มเป้าหมายใน รูปแบบที่สามารถจัดการกับลักษณะและรูปแบบข้อมูลที่ต้องการทราบ ทั้งในด้านบทความวิเคราะห์ สินค้าและบริการ อัตราดอกเบี้ย และความเคลื่อนไหวของราคาหุ้นที่สนใจ รวมถึงการรับบริการ ข่าวสารจากทางธนาคารฯ ที่จะจัดส่งให้ทางอีเมลในหัวข้อที่สนใจ นอกจากนั้นยังเป็นการรวบรวม ข้อมูลลูกค้าและกลุ่มเป้าหมาย เพื่อนำไปใช้ประโยชน์ในการตลาดต่อไป

เนื่องจากในปัจจุบันในการวิเคราะห์และออกแบบระบบ ได้มีการนำ UML (Unified Modeling Language) เข้ามาเป็นสื่อกลางทั้งระหว่างผู้วิเคราะห์และออกแบบระบบกับผู้พัฒนาระบบ และยังสามารถใช้ในการนำเสนอแนวคิดของผู้วิเคราะห์และออกแบบระบบกับผู้ใช้ระบบ เพื่อให้มีความเข้าใจที่ถูกต้องตรงกันก่อนจะเริ่มพัฒนาระบบ ดังนั้น จึงเป็นประโยชน์อย่างยิ่งที่จะ ทำการวิเคราะห์และออกแบบระบบ BBL Portal โดยการใช้แนวคิดและรูปแบบของ UML ในการ ทำ Diagram ต่าง ๆ การออกแบบฐานข้อมูล และอาจรวมถึงการทำ Forward Engineering ด้วย Rational Rose เพื่อนำมาใช้ในการทำตัวต้นแบบของระบบ และพัฒนาระบบต่อไป

Title	System Analysis and Design of the BBL Portal using UML
Student	Mr. Pornsak Soontornmat
Advisor	Assoc. Prof. Bunchong Piyatumrong
Level of Study	Master of Science in Information Technology
Major	Information Science
Academic Year	2003

ABSTRACT

As a public web-site division of Bangkok Bank has found web site to be the connection tools for bank clients and bank prospect on the internet for a while. The next step to make their web site more usefulness, they started to product “Bangkok Bank Portal (BBL Portal)” to serve for bank clients the way that all clients are able to manage the category and format of the information no matter what they would like to know such as an article analysis, products and services, interest rates and stock tracker including the interesting topic newsletter by e-mail. Moreover, it is also compile all the information about bank clients and bank prospect to be an useable market research for the future.

Currently, system analyst and design process will use “UML” (Unified Modeling language) to be a communication format between system analyst and system developer and also used for represent the conceptual of system analysis and design to user, and get more them all to the same vision before the development process is started. In consequence, it is so useful to analyse and design BBL Portal system by using “UML” model to develop diagram, database and may include forward engineering by Rational Rose and apply to create the system prototype following by developing the whole system.

กิตติกรรมประกาศ

ในการจัดทำโครงการวิเคราะห์และออกแบบระบบ BBL Portal โดยใช้ UML นี้ ได้รับการสนับสนุนเป็นอย่างดี จากหลายฝ่าย ที่คอยให้คำแนะนำปรึกษา และเสียสละเวลาอันมีค่า จนทำให้การศึกษาโครงการนี้บรรลุผลตามเป้าหมายที่วางไว้ ผู้จัดทำจึงใคร่ขอขอบพระคุณ



พรศักดิ์ สุนทรมาตย์
กันยายน 2546

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญภาพ.....	VIII
บทที่	
1. บทนำ.....	—
1.1 ความเป็นมาและความสำคัญของโครงการ.....	1
1.2 วัตถุประสงค์ของโครงการ.....	2
1.3 ขอบเขตของโครงการ.....	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	2
2. ระบบ BBL Portal.....	
2.1 หลักการของระบบ BBL Portal.....	4
2.2 การศึกษาความเป็นไปได้ (Feasibility)	4
2.3 การกำหนดความต้องการของระบบ (Requirement).....	5
2.4 ขอบเขตการพัฒนาระบบ BBL Portal.....	6
2.5 สถาปัตยกรรมของระบบ.....	6
2.6 เครื่องมือที่ใช้ในการพัฒนาระบบ.....	7
3. การออกแบบเชิงวัตถุ.....	
3.1 การออกแบบเชิงวัตถุคืออะไร.....	8
3.2 คำศัพท์และความหมาย.....	9
3.3 คุณสมบัติที่สำคัญของการออกแบบเชิงวัตถุ.....	10
3.4 Object-oriented Methodology.....	11
3.5 ส่วนประกอบที่สำคัญของ Methodology.....	12

สารบัญ (ต่อ)

หน้า

4. UML และ Waterfall Process Model	
4.1 UML คืออะไร	13
4.2 ประวัติศาสตร์ของ UML	13
4.3 ประเภทของ UML Diagram	14
4.4 ความหมายของแต่ละ Diagram	16
4.5 Waterfall Process Model	20
5. Object-oriented CASE Tool: Rational Rose	
5.1 Rational Rose 2000 Enterprise Edition	25
5.2 Rational Rose J	25
5.3 Rational Rose Oracle8	27
6. การวิเคราะห์และออกแบบระบบ BBL Portal	
6.1 Use Case Diagram	29
6.2 Component Diagram	39
6.3 Class Diagram	40
6.4 Sequence Diagram.....	60
7. การสร้างฐานข้อมูลระบบ BBL Portal	
7.1 การแปลงแบบจำลองเป็นฐานข้อมูล	78
7.2 รายละเอียดข้อมูลที่จัดเก็บในระบบ	78
8. บทสรุป	
8.1 ข้อดีและข้อเสียของการใช้ CASE Tool Ration Rose	91
8.2 สรุปผลการวิเคราะห์และออกแบบระบบงานโดยใช้ CASE Tool	92
8.3 ประโยชน์ที่ได้รับ	93
8.4 ข้อเสนอแนะ	93
บรรณานุกรม	94
ประวัติผู้เขียน	95

สารบัญตาราง

หน้า

ตารางที่

4.1 การเทียบจาก UML ไปเป็นการโปรแกรมเชิงวัตถุ (ภาษา Java).....	23
7.1 ข้อมูลหลักรายละเอียดสมาชิก (Member).....	80
7.2 ข้อมูลหลักกลุ่มข้อมูลข่าวสารที่สมาชิกสนใจ (MemberPreference).....	81
7.3 ข้อมูลหลักอัตราดอกเบี้ยที่สมาชิกสนใจ (MemberInterestRate).....	81
7.4 ข้อมูลหลักอัตราแลกเปลี่ยนที่สมาชิกสนใจ (MemberFXRate).....	81
7.5 ข้อมูลหลักราคาหุ้นที่สมาชิกสนใจ (MemberStock).....	82
7.6 ข้อมูลหลักสินค้าที่สมาชิกสนใจ (MemberProduct).....	82
7.7 ข้อมูลหลักกลุ่มข้อมูลข่าวสาร (Preference).....	82
7.8 ข้อมูลหลักข้อมูลข่าวสาร (Content).....	83
7.9 ข้อมูลหลักรายละเอียดข้อมูลข่าวสาร (ContentDetail).....	83
7.10 ข้อมูลหลักข้อมูลข่าวสารที่สัมพันธ์กับสินค้า (ContentProduct).....	84
7.11 ข้อมูลหลักรายละเอียดคำถามจากสมาชิก (Question).....	84
7.12 ข้อมูลหลักรายละเอียดการ ได้ตอบ (Answer).....	85
7.13 ข้อมูลหลักรายละเอียดเจ้าหน้าที่ (Staff).....	85
7.14 ข้อมูลหลักกลุ่มเจ้าหน้าที่ (StaffGroup).....	85
7.15 ข้อมูลหลักรายละเอียดกลุ่มข้อมูลข่าวสารที่เจ้าหน้าที่จัดการ (StaffPreference).....	86
7.16 ข้อมูลหลักมูลค่าหุ้น (Stock).....	86
7.17 ข้อมูลหลักสินค้า (Product).....	86
7.18 ข้อมูลหลักกลุ่มสินค้า (ProductGroup).....	87
7.19 ข้อมูลหลักชื่อเมือง (City).....	87
7.20 ข้อมูลหลักชื่อประเทศ (Country).....	87
7.21 ข้อมูลหลักสัญชาติ (Nation).....	88
7.22 ข้อมูลหลักอาชีพ (Job).....	88
7.23 ข้อมูลหลักประเภทกิจการ (BusinessType).....	88
7.24 ข้อมูลหลักประวัติการจัดการอัตราดอกเบี้ย อัตราแลกเปลี่ยน และมูลค่าหุ้น (Log).....	88

สารบัญตาราง (ต่อ)

หน้า

ตารางที่

7.25 ข้อมูลหลักการทำงาน (Action).....89

7.26 ข้อมูลหลักบริการ (Service).....89

7.27 ข้อมูลหลักอัตราดอกเบี้ย (InterestRate).....89

7.28 ข้อมูลหลักกลุ่มอัตราดอกเบี้ย (InterestRateGroup).....90

7.29 ข้อมูลหลักอัตราแลกเปลี่ยน (FXRate).....90



สารบัญรูปภาพ

หน้า

รูปที่

3.1 ตัวอย่าง Object ของคนที่ชื่อ “John”.....	9
3.2 ตัวอย่างส่วนประกอบภายใน Object ของคนที่ชื่อ “John”.....	9
3.3 ตัวอย่าง Static Relation ของ Object.....	10
3.4 ตัวอย่าง Dynamic Relation ของ Object.....	10
4.1 แผนภาพแสดง UML Diagram ทั้งหมด 9 Diagram.....	15
4.2 สัญลักษณ์ของ Actor.....	16
4.3 สัญลักษณ์ของ Use Case.....	17
4.4 สัญลักษณ์ของ Communication.....	17
4.5 สัญลักษณ์ของ Relationship.....	17
4.6 สัญลักษณ์ของ Class.....	18
4.7 สัญลักษณ์ของ Association Relationship.....	18
4.8 สัญลักษณ์ของ Aggregation Relationship.....	19
4.9 สัญลักษณ์ของ Depend on Relationship.....	19
4.10 ตัวอย่างความสัมพันธ์แบบ Generalization.....	19
4.11 สัญลักษณ์ที่ใช้ภายใน Component Diagram.....	20
4.12 แผนภาพแสดง Waterfall Model.....	21
6.1 ยูสเคสไดอะแกรมของระบบ BBL Portal.....	31
6.2 Package ภายในทั้งหมด (Logical View).....	40
6.3 Package ภายนอกทั้งหมด (Logical View).....	40
6.4 Class Diagram ภายใน Database Package.....	41
6.5 Class Diagram ในกลุ่ม Member ภายใน Bean Package.....	41
6.6 Class MemberUser ภายใน Bean Package.....	42
6.7 Class MemberFXRate และ MemberInterestRate ภายใน Bean Package.....	43
6.8 Class MemberPreference, MemberStock และ MemberProduct ภายใน Bean Package.....	44
6.9 Class Diagram ในกลุ่ม Preference ภายใน Bean Package.....	44

สารบัญรูปภาพ (ต่อ)

หน้า

รูปที่

6.10 Class Content ภายใน Bean Package.....	45
6.11 Class Preference ภายใน Bean Package.....	45
6.12 Class ContentProduct ภายใน Bean Package.....	46
6.13 Class Diagram ในกลุ่ม QA ภายใน Bean Package.....	46
6.14 Class Diagram ในกลุ่ม Staff ภายใน Bean Package.....	47
6.15 Class Diagram ในกลุ่ม Email Alert and Log ภายใน Bean Package.....	47
6.16 Class FX Rate ภายใน Bean Package.....	48
6.17 Class Action, Log และ Service ภายใน Bean Package.....	48
6.18 Class Interest Rate และ Interest Rate Group ภายใน Bean Package.....	49
6.19 Class Stock ภายใน Bean Package.....	49
6.20 Class Email ภายใน Bean Package.....	50
6.21 Class Diagram ในกลุ่ม General ภายใน Bean Package.....	50
6.22 Class Business Type, Job และ Nation ภายใน Bean Package.....	51
6.23 Class City และ Country ภายใน Bean Package.....	51
6.24 Class Product และ ProductGroup ภายใน Bean Package.....	52
6.25 Class Diagram ภายใน Manager Package.....	52
6.26 Class MemberManager และ StaffManager ภายใน Manager Package.....	53
6.27 Class PreferenceManager และ QAManager ภายใน Manager Package.....	54
6.28 Class NewsletterManager และ EmailManager ภายใน Manager Package.....	54
6.29 Class FXManager และ InterestRateManager ภายใน Manager Package.....	54
6.30 Class StockManager และ LogManager ภายใน Manager Package.....	55
6.31 Class GeneralManager ภายใน Manager Package.....	55
6.32 Class ProductManager ภายใน Manager Package.....	56

สารบัญรูปภาพ (ต่อ)

หน้า

รูปที่

6.33 Class Diagram ในกลุ่ม Member ภายใน Servlet Package.....	56
6.34 Class Diagram ในกลุ่ม Staff ภายใน Servlet Package.....	57
6.35 Class Diagram ในกลุ่ม Preference ภายใน Servlet Package.....	57
6.36 Class Diagram ในกลุ่ม QA ภายใน Servlet Package.....	58
6.37 Class Diagram ในกลุ่ม Alert Log ภายใน Servlet Package.....	58
6.38 Class Diagram ในกลุ่ม General ภายใน Servlet Package.....	59
6.39 Class Diagram ภายใน Form Package.....	60
6.40 Sequence Diagram: Registration.....	60
6.41 Sequence Diagram: Authenticate Member.....	61
6.42 Sequence Diagram: Update Information.....	61
6.43 Sequence Diagram: Update Preference.....	62
6.44 Sequence Diagram: Update Alert Criteria.....	63
6.45 Sequence Diagram: Maintain Question.....	64
6.46 Sequence Diagram: Create New Question.....	64
6.47 Sequence Diagram: Response Answer.....	65
6.48 Sequence Diagram: Authenticate Staff.....	66
6.49 Sequence Diagram: Add Staff.....	66
6.50 Sequence Diagram: Update Staff.....	67
6.51 Sequence Diagram: Delete Staff.....	67
6.52 Sequence Diagram: Add Staff Group.....	68
6.53 Sequence Diagram: Update Staff Group.....	68
6.54 Sequence Diagram: Delete Staff Group.....	69
6.55 Sequence Diagram: Delete Member.....	70
6.56 Sequence Diagram: Update Member.....	70
6.57 Sequence Diagram: Add Preference.....	71

สารบัญรูปภาพ (ต่อ)

หน้า

รูปที่

6.58 Sequence Diagram: Delete Preference.....	71
6.59 Sequence Diagram: Update Preference.....	72
6.60 Sequence Diagram: Add Content.....	72
6.61 Sequence Diagram: Delete Content.....	73
6.62 Sequence Diagram: Update Content.....	73
6.63 Sequence Diagram: Maintain Question and Answer.....	74
6.64 Sequence Diagram: Response Question.....	75
6.65 Sequence Diagram: Stock Transformation.....	75
6.66 Sequence Diagram: Generate Stock Alert.....	76
6.67 Sequence Diagram: Generate Interest Rates Alert.....	76
6.68 Sequence Diagram: Generate FX Rates Alert.....	77
6.69 Sequence Diagram: Generate Newsletter.....	77

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของโครงการ

โครงการวิเคราะห์และออกแบบระบบ BBL Portal โดยใช้ UML เป็นส่วนหนึ่งของโครงการพัฒนาระบบ BBL Portal ของหน่วยงานโครงการ Public Website ของธนาคารกรุงเทพ (โดยมีตัวย่อว่า BBL) ซึ่งเป็นหน่วยงานที่ทำหน้าที่พัฒนาและบริหารเว็บไซต์ในด้านการเผยแพร่ข้อมูล ข่าวสาร ผลิตภัณฑ์ และความรู้เกี่ยวกับธนาคาร โดยมีการปรับปรุงโครงสร้างและรูปแบบเว็บไซต์โดยรวมทั้งหมดในปี พ.ศ. 2544 และต่อมาได้มีการปรับปรุงในเรื่องของโครงสร้างที่เกี่ยวข้องกับระบบงานย่อย และลักษณะการโปรแกรมในปี พ.ศ. 2545 นอกจากนี้ยังได้มีการปรับปรุงเปลี่ยนแปลงรายละเอียดในส่วนต่าง ๆ อย่างต่อเนื่อง เพื่อให้บริการข้อมูลที่มีประโยชน์ ทันสมัย และสามารถใช้งานได้สะดวกยิ่งขึ้นตลอดเวลา

สำหรับโครงการพัฒนาระบบ BBL Portal เป็นโครงการที่ทางหน่วยงานโครงการ Public Website เป็นผู้ริเริ่มดำเนินการ เพื่อเป็นการสร้างความสัมพันธ์กับลูกค้า และผู้เข้าชมให้มีความผูกพันกันมากยิ่งขึ้น โดยจะมีการเปิดให้ลูกค้าและผู้เข้าชมได้ลงทะเบียนเพื่อรับบริการกับทางเว็บไซต์ในรูปแบบเฉพาะ ตามความต้องการของแต่ละบุคคล และการให้บริการข้อมูลข่าวสารที่จะส่งตรงไปยังอีเมลของผู้ที่ต้องการ ตามข้อมูลข่าวสารที่น่าสนใจที่สามารถเลือกรับบริการได้ตลอดเวลา รวมถึงการบริการจากผู้เชี่ยวชาญของทางธนาคาร ที่จะมาบริการให้ความรู้ คำแนะนำ และตอบข้อซักถามจากผู้ลงทะเบียน

การพัฒนาระบบ BBL Portal จะเป็นการร่วมมือกันจากหลายทีมงาน โดยมีกระบวนการพัฒนาระบบอยู่หลายขั้นตอนตามลักษณะของ Waterfall Process Model ซึ่งเป็นรูปแบบวงจรการพัฒนาซอฟต์แวร์ (Software Development Life Cycle Model) ที่หน่วยงานเลือกใช้ โดยจะมีการพัฒนาระบบในรูปแบบของแนวคิดเชิงวัตถุ (Object-Oriented Concept) โดยใช้ UML เป็นสื่อกลางในการพัฒนาระบบ

เนื่องจากการพัฒนาระบบ BBL Portal จัดเป็นระบบที่ค่อนข้างละเอียด ประกอบด้วยระบบงานย่อยหลายระบบ ดังนั้น สำหรับโครงการวิเคราะห์และออกแบบระบบ BBL Portal โดยใช้ UML นี้ จะเป็นการดำเนินการเฉพาะในส่วนของกระบวนการวิเคราะห์ การออกแบบระบบ

โดยจะมีการทำ Forward Engineering สำหรับเป็นโครงสร้างหลักของกระบวนการการพัฒนาระบบในขั้นตอนต่อไป

1.2 วัตถุประสงค์ของโครงการ

- 1.2.1 ศึกษาหลักการของ BBL Portal
- 1.2.2 ศึกษาในหัวข้อ การวิเคราะห์และออกแบบระบบงานด้วย UML
- 1.2.3 ศึกษาการผสมผสานการพัฒนาระบบงานบนเว็บ ด้วยแนวคิดเชิงวัตถุ (Object-oriented Web Application) กับฐานข้อมูลเชิงสัมพันธ์ (Relational Database)
- 1.2.4 ศึกษาวิธีการเทียบเคียง (Mapping) ระหว่างการออกแบบฐานข้อมูล โดยใช้แนวคิดเชิงวัตถุด้วย UML กับฐานข้อมูลเชิงสัมพันธ์
- 1.2.5 สามารถวิเคราะห์และออกแบบระบบงานบนเว็บ โดยใช้ UML
- 1.2.6 สามารถออกแบบฐานข้อมูล และพัฒนาฐานข้อมูลเชิงสัมพันธ์ โดยใช้ CASE Tool

1.3 ขอบเขตของโครงการ

สำหรับโครงการวิเคราะห์และออกแบบระบบ BBL Portal โดยใช้ UML มีขอบเขตของโครงการดังนี้

- 1.3.1 วิเคราะห์และออกแบบระบบงาน ด้วยแนวคิดเชิงวัตถุ โดยสร้าง UML Diagram ต่าง ๆ โดยการใช้โปรแกรม Rational Rose 2000 Enterprise Edition
- 1.3.2 ออกแบบและสร้างฐานข้อมูลเชิงสัมพันธ์ โดยการสร้าง Database Schema ด้วยการใช้ CASE Tool
- 1.3.3 ทำการ Forward Engineering เพื่อเป็นการตรวจสอบการออกแบบ และเป็นโครงสร้างหลักของการพัฒนา

1.4 ประโยชน์ที่คาดว่าจะได้รับ

ในการวิเคราะห์และออกแบบระบบ BBL Portal โดยใช้ UML คาดว่าจะได้รับประโยชน์ดังนี้

1.4.1 นำแนวคิดเชิงวัตถุมาใช้วิเคราะห์และออกแบบระบบ เพื่อให้มีความยืดหยุ่นและตรงความต้องการของผู้ใช้

1.4.2 นำ CASE Tool มาใช้ในการทำงาน เพื่อให้มีความรวดเร็วและเป็นมาตรฐาน ในการสื่อความเข้าใจในการพัฒนาระบบสำหรับทุกขั้นตอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานี้เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4.3 เกิดความคล่องตัวในการประสานงานระหว่างผู้เกี่ยวข้องกับการพัฒนาระบบ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ระบบ BBL Portal

2.1 หลักการของระบบ BBL Portal

ระบบ BBL Portal เป็นระบบสร้างขึ้นเพื่อเป็นการสร้างความสัมพันธ์ที่ดีกับลูกค้าและผู้สนใจที่สมัครเป็นสมาชิก นำเสนอข้อมูลข่าวสารที่ตรงกับความต้องการของสมาชิก เพื่อความพึงพอใจ และเป็นประโยชน์ช่วยสมาชิกในการตัดสินใจเลือกใช้ผลิตภัณฑ์ของทางธนาคาร และยัง สามารถเก็บข้อมูล ลักษณะของสมาชิก เพื่อนำไปใช้ในการวิเคราะห์ ประเมินผล และปรับปรุง ผลิตภัณฑ์ให้ตรงกับความต้องการของสมาชิกมากที่สุด

2.2 การศึกษาความเป็นไปได้ (Feasibility Study)

2.2.1 ความเป็นไปได้ทางเทคนิค (Technical Feasibility) จากระบบเดิมของ Public Website Project ซึ่งเป็นระบบที่ให้บริการเกี่ยวกับข้อมูลทั่วไป ไม่มีการเก็บข้อมูลใดจากลูกค้า ไม่จำเป็นต้องทำการลงทะเบียน รวมถึงระบบไม่จำเป็นต้องทำการตรวจสอบตัวตน (Authentication) ลูกค้า ทำให้ในการพัฒนาระบบ BBL Portal นี้ จำเป็นต้องมีการจัดทำระบบเครือข่ายขึ้นมาใหม่ เพื่อเป็นการแบ่งแยกกระดานการเข้าถึงข้อมูลกันอย่างชัดเจน และเป็นแนวทางในการจัดทำข้อกำหนดเกี่ยวกับด้านความปลอดภัยของเครือข่ายในระดับที่ต่างกัน โดยระบบ BBL Portal นั้นจำเป็นต้องมีความปลอดภัยในระดับที่สูงกว่า เนื่องจากมีการเก็บข้อมูลรายละเอียดส่วนตัวของลูกค้า เพราะฉะนั้นจึงสรุปได้ว่าในเชิงเทคนิคนั้น จำเป็นที่ต้องสร้างระบบเครือข่ายสำหรับระบบ BBL Portal ขึ้นใหม่

2.2.2 ความเป็นไปได้ในการปฏิบัติงาน (Operational Feasibility)

- สมาชิกสามารถเลือกที่รับทราบข้อมูลข่าวสารเฉพาะที่สนใจ ไม่จำเป็นต้องค้นหาข้อมูลนั้นเองทุกครั้งที่ต้องการ โดยทางธนาคารจะเป็นผู้จัดทำทั้งทางเว็บเพจ และทางอีเมล และสามารถถามตอบข้อสงสัยกับเจ้าหน้าที่ของธนาคารได้โดยตรง
- เจ้าหน้าที่ที่ทำหน้าที่ดูแลข้อมูลข่าวสารสามารถที่จะควบคุมดูแลข้อมูลข่าวสารให้ถูกต้อง และทันสมัยตลอดเวลา รวมถึงสามารถเลือกเฉพาะเจาะจงบางข้อมูลข่าวสารให้แสดงขึ้นมาเป็นข้อมูลเด่น หรือเป็นข้อมูลที่ต้องการส่งให้สมาชิกที่สนใจได้ทันที

- เจ้าหน้าที่ที่ทำหน้าที่ดูแลเกี่ยวกับข้อซักถามของสมาชิก สามารถที่จะตอบข้อซักถามของสมาชิก รวมถึงควบคุมจัดการคำถามหรือข้อสงสัยที่ไม่เหมาะสมเพื่อทำการลบออกจากระบบ

- มีการทำงานในแบบอัตโนมัติ เพื่อทำการสร้างข้อมูลข่าวสาร อัตราแลกเปลี่ยน อัตราดอกเบี้ย และราคาหุ้นที่สมาชิกสนใจ และส่งไปให้สมาชิกทางอีเมล และทำการบันทึกการทำงานไว้ทุกครั้ง

- เจ้าหน้าที่ที่ทำหน้าที่ดูแลระบบสามารถที่จะควบคุมดูแลการทำงานของผู้ใช้ระบบทั้งหมด (เจ้าหน้าที่ทุกฝ่ายและสมาชิก) รวมถึงรายงานสรุปเกี่ยวกับการทำงานในส่วนของการสร้างและส่งอีเมลแบบอัตโนมัติ

2.2.3 ความเป็นไปได้ในเชิงเศรษฐศาสตร์ (Economical Feasibility)

- ด้านผลผลิต (Productivity) เป็นการสร้างฐานข้อมูลของสมาชิกผู้ลงทะเบียน ซึ่งก็คือลูกค้าหรือผู้ที่สนใจที่สามารถระบุเฉพาะเจาะจง เพื่อให้บริการได้ตรงกับความต้องการและเกิดประโยชน์กับสมาชิก ทำให้สมาชิกได้รับข้อมูลข่าวสารและเกิดความสนใจที่จะใช้บริการกับธนาคารในที่สุด

- ด้านความแตกต่าง (Differentiation) เป็นระบบใหม่ที่ยังมีคู่แข่งน้อยรายที่ทำและยังไม่มีรายใดที่มีการทำอย่างจริงจัง จึงสามารถนับได้ว่าเป็นธนาคารแรกที่ทำให้ความสำคัญกับลูกค้าและผู้สนใจ และยังเป็นการสร้างความประทับใจเพื่อให้เกิดความผูกพันกับธนาคาร ในการที่จะดำเนินการทำธุรกรรมหรือขอข้อมูลอื่นเพิ่มเติมที่เป็นประโยชน์ในการตัดสินใจเลือกใช้บริการกับทางธนาคารในอนาคตต่อไป

- ด้านการจัดการ (Management) ช่วยในการจัดการข้อมูลรายละเอียดของสมาชิกเพื่อนำไปใช้ในการวิเคราะห์ข้อมูลในด้านการตลาด และนำไปใช้ในการวางแผนและกำหนดเป้าหมายในการดำเนินงาน รวมถึงใช้ประกอบการตัดสินใจและควบคุมให้การดำเนินงานเป็นไปตามนโยบายของธนาคารต่อไป

2.3 การกำหนดความต้องการของระบบ (Requirement)

จากหลักการของระบบ BBL Portal สามารถนำมาสรุปเป็นคุณสมบัติที่สำคัญได้ดังนี้

2.3.1 สามารถมีการลงทะเบียน และเก็บข้อมูลของลูกค้าและผู้สนใจที่ต้องการเป็นสมาชิกของระบบ เพื่อนำไปใช้อ้างอิงในการใช้งานส่วนงานอื่น ๆ ของระบบ

2.3.2 สามารถจัดการเกี่ยวกับลักษณะของเว็บเพจ หรือกลุ่มข้อมูลข่าวสารเฉพาะที่ตรงกับความสนใจของสมาชิกแต่ละคน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.3 สามารถมีส่งข่าวสารให้กับสมาชิกของระบบ เพื่อให้ได้รับทราบความเคลื่อนไหวของผลิตภัณฑ์ต่าง ๆ ของธนาคาร รวมถึงข้อมูลที่น่าสนใจและเป็นประโยชน์สำหรับสมาชิกทาง E-Mail

2.3.4 สามารถมีการส่งข้อความเตือนเกี่ยวกับอัตราดอกเบี้ย และอัตราแลกเปลี่ยนต่าง ๆ ที่มีความเคลื่อนไหวไปในทิศทางที่สมาชิกสนใจทาง E-Mail

2.3.5 สามารถให้ข้อมูลเกี่ยวกับความเคลื่อนไหวของราคาหุ้น โดยมีการดึงข้อมูลมาจากบริษัทที่ให้ข้อมูลเกี่ยวกับหุ้นตามช่วงเวลาที่กำหนด

2.3.6 สามารถมีการโต้ตอบข้อซักถามระหว่างสมาชิก กับผู้เชี่ยวชาญของทางธนาคาร และมีการแจ้งสมาชิกได้ทราบทาง E-Mail เมื่อมีการตอบคำถามจากผู้เชี่ยวชาญ

2.3.7 สามารถจัดการข้อมูลทั้งหมดของระบบ โดยผู้ดูแลระบบ และผู้จัดการข้อมูล

2.4 ขอบเขตการพัฒนา ระบบ BBL Portal

การพัฒนา ระบบ BBL Portal เป็นระบบงานที่จัดทำขึ้นเพื่อให้บริการด้านต่าง ๆ แก่สมาชิก โดยมีขอบเขตและขั้นตอนการทำงานที่สำคัญของระบบดังต่อไปนี้

2.4.1 การรับลงทะเบียน เก็บข้อมูลและความสนใจของสมาชิก และการกรอกรหัสผ่านเพื่อเข้าสู่ระบบ

2.4.2 การปรับเปลี่ยนลักษณะกลุ่มข้อมูลข่าวสารของเว็บเพจที่น่าสนใจสำหรับสมาชิกแต่ละคน

2.4.3 การส่งข่าวสารให้กับสมาชิกทาง E-Mail

2.4.4 การส่งข้อความเตือนเกี่ยวกับอัตราดอกเบี้ย และอัตราแลกเปลี่ยนให้กับสมาชิกทาง E-Mail

2.4.5 การดึงข้อมูลราคาหุ้นจากนายหน้า และแสดงผลตามความต้องการของสมาชิกแต่ละคน

2.4.6 การรับข้อซักถามจากสมาชิก และการแสดงคำตอบของผู้เชี่ยวชาญ

2.4.7 การจัดการข้อมูลทั้งหมดของระบบ

2.5 สถาปัตยกรรมของระบบ

สำหรับ Architecture ของระบบจะเป็นแบบ 2-teirs Client/Server นั่นคือส่วน Presentation จะอยู่ที่ Client ในขณะที่ Business Logic และ Database จะอยู่ที่ Server และเป็นลักษณะของ

WWW-Based Application ที่เรียกว่า Server Side โดยทำการประมวลผลทางด้าน Server และส่งผลลัพธ์ไปที่ Client (WWW Browser)

2.6 เครื่องมือที่ใช้ในการพัฒนาระบบ

ภาษาที่เลือกมาใช้ในการพัฒนาระบบงาน คือ ภาษา Java ซึ่งเป็น Object-oriented Programming Language โดยใช้ JDK (Java Development Kits) รุ่น 1.2 ที่ถูกสนับสนุนโดย UML และใช้ Java Servlet ซึ่งเป็น API อยู่ภายใน Javax Package สนับสนุนการพัฒนาระบบงานแบบ WWW-Based Application ชนิด Server Side และทำการติดต่อกับระบบจัดการฐานข้อมูลเชิงวัตถุสัมพันธ์ (Object Relational Database Management System) คือ Oracle8I Enterprise Edition for Microsoft Windows NT 4.0 ผ่าน Oracle Java Database Connectivity (JDBC) รุ่น 12.01

บทที่ 3

การออกแบบเชิงวัตถุ

3.1 การออกแบบเชิงวัตถุคืออะไร

การพัฒนาระบบสารสนเทศมีการเปลี่ยนแปลงวิธีการตลอดเวลา แนวทางที่จะใช้ในอนาคตนั้นแตกต่างจากวิธีการที่ใช้ในปัจจุบัน โดยไม่สามารถคาดการณ์ได้แน่นอน แต่เราสามารถเลือกแนวทางและเครื่องมือต่าง ๆ ที่จะนำมาใช้ในการพัฒนาระบบเพื่อให้ประสบผลสำเร็จได้ การพัฒนาระบบสารสนเทศหมายถึงการแก้ไขปัญหาในระบบสารสนเทศ ประกอบด้วย การวิเคราะห์ระบบ การสร้างแบบจำลอง การออกแบบระบบ การพัฒนาระบบ การทดสอบ และการบำรุงดูแลรักษา ระบบ

การออกแบบซอฟต์แวร์เชิงโครงสร้าง (Structured Design) กำลังจะกลายเป็นอดีต อันเนื่องมาจากการออกแบบแบบนี้มีลักษณะเป็น Function-Oriented กล่าวคือ เป็นการยัดวิธีการทำงานของแต่ละองค์ประกอบเป็นหลัก ทำให้ระบบซอฟต์แวร์กลายเป็นซอฟต์แวร์เฉพาะองค์ประกอบ ไม่สามารถนำไปใช้กับองค์ประกอบอื่นได้โดยตรง ถึงแม้จะเป็นองค์ประกอบที่มีธุรกิจประเภทเดียวกันก็ตาม นอกจากนี้ ถ้ามีการปรับเปลี่ยนโครงสร้างข้อมูล ก็จะส่งผลกระทบต่อโปรแกรมที่ใช้ข้อมูลเหล่านั้น จึงทำให้ยากต่อการบำรุงรักษาระบบซอฟต์แวร์ และมีค่าใช้จ่ายสูง นอกจากนี้ การสร้างซอฟต์แวร์ด้วยวิธีเดิมยังไม่สามารถที่จะนำโปรแกรมบางส่วนมาใช้ซ้ำได้ เทคโนโลยีเชิงวัตถุได้เข้ามาสร้างความเปลี่ยนแปลงการพัฒนาระบบสารสนเทศในปัจจุบัน

“Object-oriented” เป็นแนวคิด หรือระเบียบวิธีคิด (Paradigm) ของการสร้างหรือพัฒนาระบบงานหนึ่ง ๆ ดังคำพูดที่ว่า “OO is a technique for system modeling” โดยจะมองระบบเป็นกลุ่มของวัตถุ (Object) ที่มีปฏิสัมพันธ์กัน หรือ “a number of object that interact” โดยการรวมเอาข้อมูล (Information) และฟังก์ชันการทำงาน (Operation) เข้าไว้ด้วยกันในวัตถุ และกำหนดวิธีการติดต่อกันระหว่างวัตถุ (Interface) โดยถ้าวัตถุหนึ่งต้องการจะติดต่อกับอีกวัตถุหนึ่ง จะต้องติดต่อผ่าน Interface ที่กำหนดไว้เท่านั้น

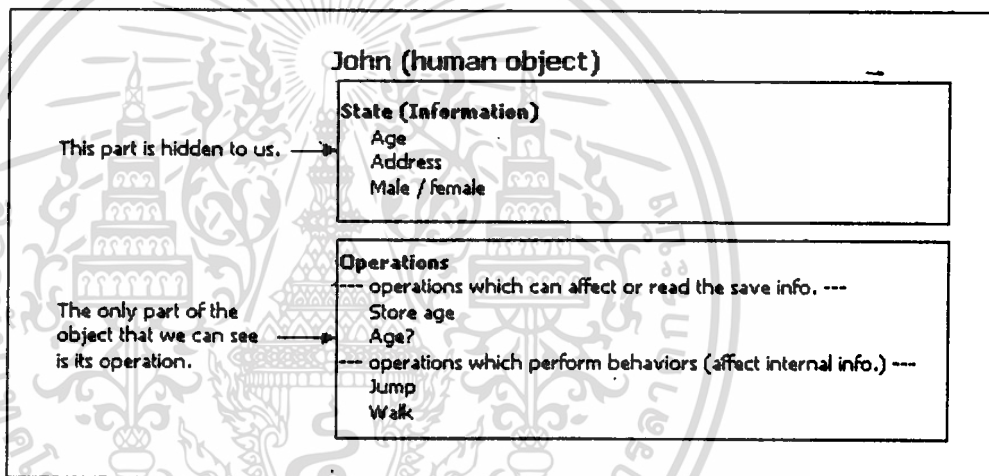
จากหลักการนี้ จะพบว่า การเปลี่ยนแปลงแก้ไข หน้าที่การทำงาน และข้อมูลใด ๆ ในวัตถุหนึ่ง จะมีผลกระทบต่อวัตถุอื่นน้อยมาก ซึ่งจะทำให้การแก้ไขโปรแกรมทำได้สะดวกและรวดเร็วยิ่งขึ้น เป็นการลดต้นทุนในการบำรุงรักษาโปรแกรม นอกจากนี้ยังมีความสามารถในการสืบทอดคุณสมบัติ (Inheritance) จากวัตถุหนึ่งไปอีกวัตถุหนึ่งได้ ทำให้สามารถนำซอฟต์แวร์บางส่วนที่มีอยู่
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เดิมกลับมาใช้ใหม่ได้ (Reusability) โดยผู้พัฒนาไม่จำเป็นต้องทราบว่าซอฟต์แวร์บางส่วนที่นำกลับมาใช้นั้นเขียนขึ้นมาอย่างไร (How does it work?) เพียงแต่ทราบว่าซอฟต์แวร์ส่วนนั้นทำอะไร (What does it work?) และทำการเพิ่มคุณสมบัติอื่น ๆ ที่ผู้พัฒนาต้องการเข้าไป ทำให้การพัฒนาซอฟต์แวร์ใหม่ ๆ ทำได้รวดเร็วขึ้น

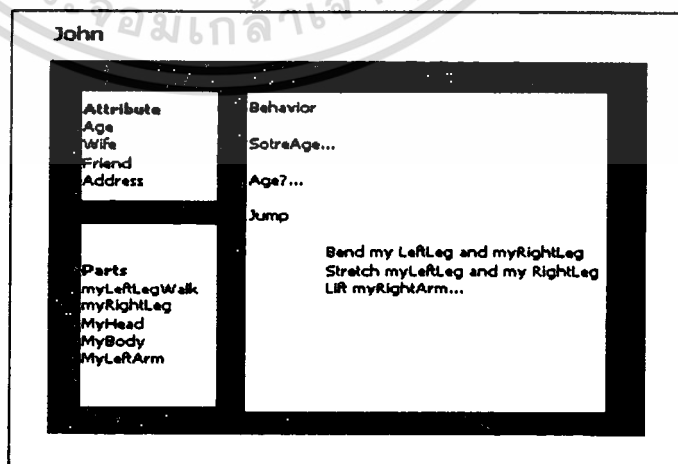
3.2 คำศัพท์และความหมาย

3.2.1 Object คือ วัตถุ หรือสิ่งที่ประกอบด้วย สถานะ (State) หรือข้อมูล (Information) และฟังก์ชันการทำงาน (Operation) หรือพฤติกรรม (Behavior)

ตัวอย่าง



รูปที่ 3.1 ตัวอย่าง Object ของคนที่ชื่อ "John"



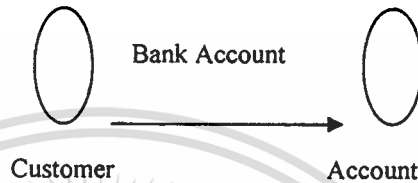
รูปที่ 3.2 ตัวอย่างส่วนประกอบภายใน Object ของคนที่ชื่อ "John"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นเมื่อเรามองเข้าไปใน Object จะประกอบด้วย Attribute, Part และ Behavior ดัง
รูป 3.2

3.2.2 Message passing ความเกี่ยวข้องกันระหว่าง Object แบ่งเป็น 2 ลักษณะ คือ

3.2.2.1 Static Relation เช่น ลูกค้ายเป็นเจ้าของบัญชี



รูปที่ 3.3 ตัวอย่าง Static Relation ของ Object

3.2.2.2 Dynamic Relation เช่น ผู้ฝากเงินเข้าไปในบัญชี



รูปที่ 3.4 ตัวอย่าง Dynamic Relation ของ Object

3.2.3 Class และ Instance สามารถสรุปได้ว่า Class คือ เซตของ Object และ Instance เป็น Object ที่สร้างจาก Class

ตัวอย่าง Class ของคน ประกอบด้วย คนแต่ละคน โดยมี สมชายเป็น Instance ของ Class คน หรือ คนชื่อ “ประคิษฐ์” เป็น Instance ของ Class คน เช่นกัน โดยแต่ละ Instance จะมีความแตกต่างกันระหว่าง Object อื่น ๆ (มีเอกลักษณ์ของ Identity)

3.3 คุณสมบัติที่สำคัญของการออกแบบเชิงวัตถุ

3.3.1 Encapsulation และ Information Hiding

ข้อมูล (Information) และหน้าที่การทำงาน หรือพฤติกรรม (Operation/Behavior) จะถูกจับมัดรวมเข้าด้วยกัน (Encapsulation) ซึ่งจะสนับสนุนให้เกิด การซ่อนข้อมูล (Information Hiding) และสิ่งที่เราจะเห็นได้ของ Object คือ Instance ของมัน

ทำให้การเข้าถึงข้อมูลจะกระทำโดยเรียกใช้ Operation ของ Object ขึ้นมาทำงาน

ดังคำกล่าวที่ว่า “The only way to reaching the information in an object is by using its operation.”

เอกสารนี้เป็นเอกสารที่ลงนามและประทับการลงนามเพื่อใช้ในการอ้างอิงเท่านั้น ไม่สามารถนำเอกสารนี้ไปใช้ในการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

“Abstract Data Type (ADT)” คือ Model ที่มี Operation ของมันที่จะกระทบต่อ model

ข้อดี ของการ Encapsulation คือ สามารถลดความซับซ้อนลง เนื่องจากสามารถใช้งานโดยรู้เพียงคุณลักษณะต่าง ๆ เท่านั้น โดยไม่ต้องรู้ถึงโครงสร้างภายใน

3.3.2 Inheritance

การสืบทอดคุณสมบัติทั่ว ๆ ไปที่เหมือนกัน ระหว่าง Class

ตัวอย่าง Class ของผู้ชาย inherits Class ของคน

ข้อดี ของ Inheritance คือ

- สามารถนำนิยามโดยทั่ว ๆ ไปมาใช้ใหม่ (Reuse common description)
- การปรับปรุง แก้ไข ทำเพียงที่เดียวเท่านั้น (Modification can be made in only one place)
- ลดความซ้ำซ้อนสำหรับโปรแกรมย่อยเพื่อให้ง่ายต่อการทำความเข้าใจ (Redundancy can be reduced leading to smaller models that are easier to understand)

“Overriding” คือคุณสมบัติที่สามารถนิยาม Behavior หรือ Information ใหม่ได้ สำหรับ Class ที่สืบทอดมาจาก Super Class

3.3.3 Polymorphism

โดยความหมาย คือ “Having many forms” หมายถึงการส่งข้อความเดียวกันให้กับ Object ที่ต่างกัน จะแสดงพฤติกรรมที่แตกต่างกัน

ตัวอย่าง การบวกของ String และ Integer จะมีสัญลักษณ์เดียวคือ + แต่ทำกับข้อมูลต่างชนิดกัน

ข้อดี ของ Polymorphism คือ สนับสนุนการนำกลับมาใช้ใหม่ (Reuse) และมีการเปลี่ยนแปลงแก้ไขได้

3.4 Object-oriented Methodology

การพัฒนากระบวนวิธีด้วย Object-oriented Methodology นักวิเคราะห์ระบบจะต้องมีแนวคิดหรือระเบียบวิธีคิดในลักษณะเชิงวัตถุที่ได้กล่าวมาแล้ว ตั้งแต่กระบวนการวิเคราะห์ ออกแบบ การเขียนโปรแกรม ไปจนถึงการทดสอบการทำงานของระบบ หรือโปรแกรม (OO Analysis, OO Design, OO Programming และ OO Testing) ยกตัวอย่างเช่น การวิเคราะห์ระบบงาน จะต้องวิเคราะห์ให้ได้ว่าควรมี Class อะไรบ้าง แต่ละ Class มีพฤติกรรมสื่อสารระหว่างกันอย่างไร เกิด

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการเรียนการสอนเท่านั้น ไม่อนุญาตให้เผยแพร่ไปใช้ประโยชน์ทางการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เหตุการณ์อะไรบ้าง เป็นต้น จึงจะได้ออกมาเป็นระบบที่ต้องการ ต่างจากการวิเคราะห์และออกแบบในระบบแบบดั้งเดิม ที่มองแยกกันระหว่างการออกแบบในส่วนของคุณสมบัติ กับส่วนของกระบวนการทำงาน

3.5 ส่วนประกอบที่สำคัญของ Methodology

System Modeling Methodology โดยทั่วไป ประกอบด้วย 2 ส่วน คือ Language และ Process Model โดย Language คือสัญลักษณ์ (Notation) ที่ใช้ในการอธิบาย แสดงความหมายของระบบงาน ส่วน Process หมายถึง การบอกกล่าวว่าจะทำอะไรในขั้นตอนใดของการวิเคราะห์และออกแบบระบบงาน

ดังนั้น Object-oriented Methodology จึงประกอบด้วย 2 ส่วนเช่นกัน โดย Object-oriented Methodology ที่ศึกษาและใช้ในการจัดทำโครงการคือ จะเลือก UML เป็น Language และใช้ Waterfall Process Model เป็นส่วนของการไหลเวียนของงาน (Work Flow) โดยจะกล่าวถึงรายละเอียดของ UML ที่สัมพันธ์กับ Waterfall Process Model ในบทถัดไป

บทที่ 4

UML และ Waterfall Process Model

4.1 UML คืออะไร

UML (Unified Modeling Language) เป็นภาษา หรือรูปแบบจำลอง มาตรฐานที่ใช้ในการจำลองแบบในรายละเอียด (Blue Print) สำหรับการผลิตสร้าง (Construct) ซอฟต์แวร์ เช่นเดียวกับแบบจำลองทางธุรกิจ (Business Modeling) โดย UML จะเป็นการเสนอรูปแบบการปฏิบัติงานจริงในทางวิศวกรรมที่ดี (Best engineering practices) อีกทั้งยังเหมาะกับระบบที่มีขนาดใหญ่และยุ่งยากซับซ้อนอีกด้วย

UML เป็นรูปแบบที่รวมข้อดีของรูปแบบอื่น ๆ เข้าไว้ด้วยกัน คือ

- Data Modeling Concept (Entity Relation Concept)
- Business Modeling (Workflow)
- Object Modeling
- Component Modeling

นอกจากนี้ UML ยังสามารถใช้ได้ตลอดทั้งโครงการ หรือการทำงาน เช่นเดียวกับ SDLC (Software Development Life Cycle)

4.2 ประวัติศาสตร์ของ UML

การกำหนดรูปแบบภาษาการวิเคราะห์และออกแบบเชิงวัตถุ เริ่มปรากฏในระหว่างกลางทศวรรษที่ 1970 ถึงปลายทศวรรษที่ 1980 มีการทดลองด้วยแนวความคิดที่แตกต่างกันอย่างแพร่หลายเกี่ยวกับการวิเคราะห์ และออกแบบเชิงวัตถุ และในช่วงระหว่างปี 1989 – 1994 ได้มีการเพิ่มจำนวนของรูปแบบภาษาจากเดิมซึ่งมีน้อยกว่า 10 รูปแบบไปเป็นมากกว่า 50 รูปแบบ และผู้ใช้จำนวนมากที่ใช้วิธีการทางเชิงวัตถุ (Object-oriented) ก็ยังไม่พบกับวิธีที่ตนเองพอใจได้

จนกระทั่งกลางทศวรรษที่ 1990 ได้เริ่มมีการพัฒนา UML ขึ้นในเดือนตุลาคม 1994 โดย Grady Booch และ Jim Rumbaugh ของ Rational Software Corporation ได้มีการรวมรูปแบบของ Booch และ OMT (Object Modeling Technique) เพื่อศึกษาวิธีการในเชิงวัตถุ หลังจากนั้นในเดือนตุลาคม 1995 ก็มีรูปแบบของ Unified Method 0.8 แต่ก็ไม่ประสบความสำเร็จไปในปี 1995 นั้น Ivar Jacobson ก็ได้นำรูปแบบนั้นมารวมกับวิธี OOSE (Object-oriented Software Engineering) และเอกสารนี้เป็นเอกสารที่ส่งมอบไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สุดท้ายในปี 1996 จึงมี UML 1.0 มาเป็นมาตรฐานโดยการร่วมมือของหลาย ๆ บริษัท เช่น Digital Equipment Corp., I-Logix, Intellicorp, IBM, ICON Computing, MCI Systemhouse, Microsoft, Oracle, Rational Software, TI และ Unisys

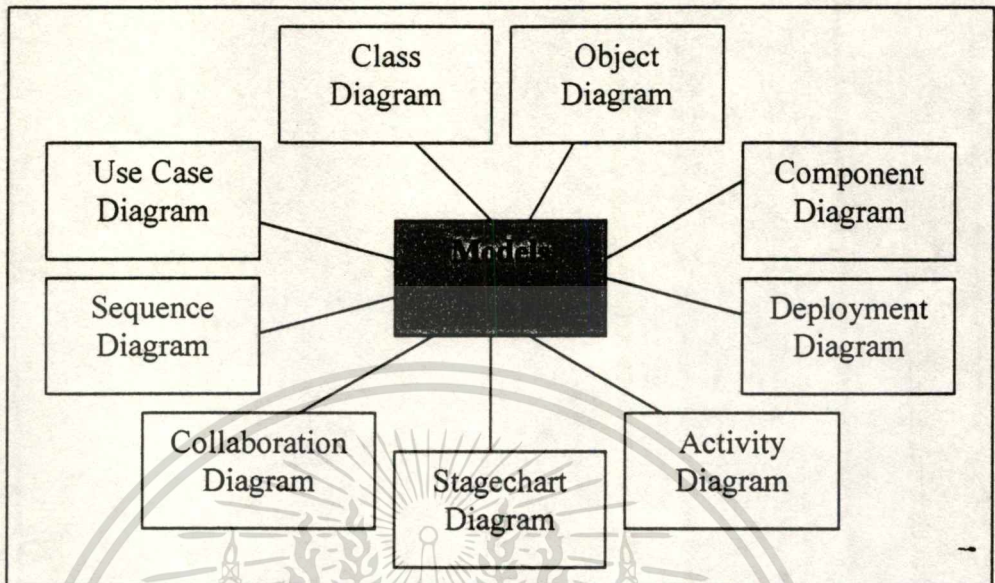
ปัจจุบัน UML ได้รับการรองรับจาก OMG (Object Management Group) ให้เป็นมาตรฐานหนึ่ง และไม่มีใครเป็นเจ้าของ เป็นรูปแบบเปิด กฎเกณฑ์ที่ทำให้ UML ประสบความสำเร็จ คือ

- UML เป็นสิ่งที่ทำแล้วได้ผลในหลาย ๆ วิธีที่เป็นลักษณะรูปแบบทางภาษาที่เคยเสนอมาแล้วในอดีต
- UML เป็นการรวมรูปแบบที่ใช้ได้ในหลาย ๆ ระบบ ขั้นตอนการพัฒนา ระบบ (Development Phase)

หลาย ๆ องค์กรได้มีการสนับสนุนมาตรฐานของ UML ตั้งแต่พื้นฐานรูปแบบทางภาษา (Modeling Language) ซึ่งเป็นวิธีการเชิงวัตถุ UML เป็นที่ยอมรับกันอย่างกว้างขวาง โดย UML 1.0 เป็นเวอร์ชันที่สามารถใช้ได้และมีเสถียรภาพในเดือนมกราคม 1997 เอกสารเกี่ยวกับ UML version 1.0 ก็ได้รับการตอบสนองโดย OMG Analysis & Design Task Force's RFP-1 และในกลางปี 1997 OMG ได้มีการยอมรับมาตรฐานของ UML จึงทำให้หลาย ๆ องค์กร และผู้ผลิตได้ยอมรับมาตรฐาน UML จึงมีการใช้วิธีการสนับสนุนอื่น ๆ ทางด้าน UML มีเครื่องมือที่ช่วยในการพัฒนา ตลอดจน องค์กรที่ช่วยในการฝึกอบรม

4.3 ประเภทของ UML Diagram

UML ได้กำหนด Diagram ทั้งหมด 9 Diagram ดังแสดงในรูปที่ 4.2 เพื่อใช้ในการจำลองแบบระบบงาน โดยการมองในแง่มุมต่าง ๆ (Diagram แต่ละชนิด) เพื่อให้สามารถเข้าใจระบบงานให้มากที่สุด โดยผู้จำลองแบบไม่จำเป็นต้องใช้ทุก Diagram โดยสามารถเลือกใช้ Diagram ที่เหมาะสม



รูปที่ 4.1 แผนภาพแสดง UML Diagram ทั้งหมด 9 Diagram

UML ทั้ง 9 Diagram สามารถจัดแบ่งออกเป็นกลุ่มต่าง ๆ ได้ดังนี้

4.3.1 Functional Model

Functional Model เป็นแบบจำลองที่ใช้ในการแสดงความต้องการของระบบทั้งหมด ช่วยในการอธิบายรายละเอียดหลัก ๆ ภายในวัตถุ แสดงให้เห็นการไหลของข้อมูลในแต่ละการทำงาน โดยจะสนใจเพียงแค่ว่ามีงานอะไรบ้างที่ต้องทำ เครื่องมือที่ใช้ในการแสดงความต้องการของระบบคือ Use Case Diagram ใช้ในการแสดงความต้องการของระบบทั้งหมดในลักษณะที่ผู้ใช้งานสามารถเข้าใจได้ง่าย โดยจะถูกนำไปใช้ต่อไปใน Phase ต่าง ๆ ของการวิเคราะห์และออกแบบระบบ

4.3.2 Object Model

Object Model เป็นแบบจำลองที่ใช้ในการแสดงโครงสร้างของระบบ โดยจะแสดงในรูปของ Class ต่าง ๆ พิจารณาจากความต้องการของระบบ ที่แสดงอยู่ใน Functional model เครื่องมือที่ใช้ในการแสดงโครงสร้างของระบบคือ Component Diagram แสดงถึงโครงสร้างและความสัมพันธ์ขององค์ประกอบต่าง ๆ ของซอฟต์แวร์ ส่วน Class Diagram แสดงถึงโครงสร้างและความสัมพันธ์ของ Class Object Diagram แสดงถึงโครงสร้างและความสัมพันธ์ของ Instance และ Deployment Diagram แสดงถึงความสัมพันธ์ทางกายภาพระหว่างส่วนประกอบของฮาร์ดแวร์และซอฟต์แวร์

4.3.3 Dynamic Model

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Dynamic Model เป็นแบบจำลองที่ใช้ในการแสดงถึงการทำงานระหว่าง Object ต่าง ๆ ตามการส่งข้อความ (Message) หรือเมื่อเหตุการณ์ (Event) ต่าง ๆ ได้เกิดขึ้น Object ในที่นี้ หมายถึง Instance ที่สร้างขึ้นจาก Class ที่ได้ออกแบบไว้ใน Object Model มีคุณสมบัติ และ พฤติกรรมเช่นเดียวกับ Class ต้นแบบ ในการทำงานของระบบจะประกอบขึ้นจากการส่งข้อความ ไปมาระหว่าง Object เหล่านั้น เมื่อมีการทำงานไปเรื่อย ๆ แล้ว Object อาจจะมีการเปลี่ยนสถานะ (State) ไปตามเหตุการณ์ที่เกิดขึ้นได้ เพื่อให้เป็นตามที่กำหนดไว้ใน Functional Model เครื่องมือที่ใช้คือ Sequence Diagram แสดงให้เห็นถึงการแลกเปลี่ยนข่าวสารระหว่าง Object ต่าง ๆ Collaboration Diagram สามารถแสดงถึงความสัมพันธ์ระหว่าง Object ภายใน Activity Diagram แสดงถึงสถานะการทำงานในแต่ละ Action และ State Diagram แสดงวงจรชีวิตของ Object บอกถึง เหตุการณ์ต่าง ๆ มีผลกระทบอะไรเกิดขึ้นบ้าง

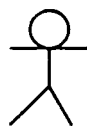
4.4 ความหมายของแต่ละ Diagram

จะอธิบายความหมายและสัญลักษณ์ต่าง ๆ ที่ใช้ใน UML Diagram โดยจะอธิบายเฉพาะ diagram ที่เลือกใช้ในการพัฒนาในโครงการพัฒนาระบบงานนี้ 4 Diagram ได้แก่ Use Case Diagram, Class Diagram, Component Diagram และ Sequence Diagram ดังนี้

4.4.1 Use Case Diagram

Use Case Diagram เป็นแนวความคิดของ Ivar Jacobson ซึ่ง OMG ได้รวมไว้ใน มาตรฐานของ UML ด้วย และมีวิธีการอื่น ๆ นำแนวคิดนี้ไปใช้อย่างกว้างขวาง Use Case Diagram เป็นสิ่งที่ใช้ในการแสดงความต้องการของระบบทั้งหมดในลักษณะที่ผู้ใช้งานสามารถเข้าใจได้ง่าย โดยแสดงความสัมพันธ์ระหว่างผู้ใช้กับระบบต่าง ๆ สัญลักษณ์ได้ดังนี้

- Actor คือสิ่งที่อยู่ภายนอกระบบที่จะพัฒนาขึ้นทั้งหมด ที่ต้องการแลกเปลี่ยน หรือส่งข้อมูลให้กับระบบ โดยที่ Actor อาจจะเป็นคน ระบบ หรือโปรแกรมอื่น ๆ ก็ได้ โดยมากจะเป็นผู้เริ่มทำงานกับ Use Case เช่น ผู้ใช้ระบบ เป็นต้น



รูปที่ 4.2 สัญลักษณ์ของ Actor

- Use Case จะเป็นตัวแทนงานที่เกิดขึ้นในขั้นตอนต่าง ๆ โดยจะใช้สัญลักษณ์เป็นรูปวงรี หรือวงกลม ถ้า Use Case ใดมีกรอบสี่เหลี่ยมสีเทาล้อมรอบแสดงว่ามี Diagram ข่อยที่ใช้อธิบายรายละเอียดของ Use Case ต่อไป



รูปที่ 4.3 สัญลักษณ์ของ Use Case

- Communication เป็นการแสดงความสัมพันธ์หรือการติดต่อสื่อสารกัน (การรับและให้ข้อมูลข่าวสารแก่กันและกัน) ระหว่าง Actor และ Use Case ซึ่งอาจจะเป็นการสื่อสารแบบทางเดียว หรือ 2 ทางก็ได้ แสดงด้วยสัญลักษณ์เส้นที่มีหัวลูกศรสีดำ



รูปที่ 4.4 สัญลักษณ์ของ Communication

- Relationship เป็นการแสดงความสัมพันธ์ระหว่าง Use Case กับ Use Case โดยใช้เส้นที่มีหัวลูกศรสีขาว ความสัมพันธ์ระหว่าง Use Case มีได้ 2 แบบ คือ Extends และ Uses



รูปที่ 4.5 สัญลักษณ์ของ Relationship

Extends เป็นการเพิ่มการทำงานให้กับ Use Case โดยการเรียกใช้จากอีก Use Case หนึ่ง ลูกศรจะออกจาก Use Case ที่ถูกเรียกใช้งานไปยัง Use Case ที่ต้องการเพิ่มเติมการทำงาน และจะมีข้อความ <<extends>> ระบุอยู่ข้างเส้น

Uses เป็นการแสดงให้เห็นถึงการ Inherit หน้าที่การทำงานจาก Use Case หนึ่ง ไปอีก Use Case หนึ่ง ปลายลูกศรจะอยู่ที่ Use Case หลักที่จะถูกถ่ายทอดความสามารถออกไป โดยจะมีข้อความ <<uses>> ปรากฏอยู่ข้างเส้น

4.4.2 Class Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Class Diagram เป็น Model ชนิดหนึ่งที่เป็น Static Model จะเป็นการอธิบายถึง Class และความสัมพันธ์ระหว่าง Class ที่มีโครงสร้างของข้อมูลรวมถึงพฤติกรรมของข้อมูลที่แตกต่างกัน ซึ่ง Class 1 Class สามารถกำหนดทิศทางของการนำไปสร้างโปรแกรมและการสร้าง Class ใน Object-oriented ได้

Class ใช้สัญลักษณ์เป็นรูปสี่เหลี่ยม จะอธิบายถึงกลุ่มของ Object ต่าง ๆ ที่มีคุณลักษณะ (Attribute) หน้าที่การทำงาน (Operation) และความสัมพันธ์ (Relationship) ขึ้นพื้นฐาน ซึ่งในแต่ละ Class ประกอบไปด้วยรายละเอียดพื้นฐาน 3 ส่วน ดังรูป

Class name
Attribute
Operator

รูปที่ 4.6 สัญลักษณ์ของ Class

- Class Name แสดงชื่อของ Class ที่กำหนดในระบบ
- Attribute เป็นการกำหนดคุณลักษณะทั้งหมดที่มีภายใน Class บอกถึงรายละเอียดของข้อมูล (Attribute) ภายใน Class เดียวกัน จะไม่ซ้ำกัน แต่ชื่อข้อมูลใน Class อาจจะไปซ้ำกับชื่อข้อมูลใน Class อื่นได้
- Operation เป็นส่วนที่ใช้อธิบายใน Class นั้นมี Method อะไรบ้าง มีการรับค่าชุดของตัวแปร (Argument) อะไรบ้าง หรือมีการส่งค่าออกไปหรือไม่
- ความสัมพันธ์ระหว่าง Class (Relationship) โดย UML ได้เตรียมความสัมพันธ์ระหว่าง Class ไว้ จะมีความสัมพันธ์อยู่ 4 แบบ ดังนี้
 - Association จะแสดงความสัมพันธ์ระหว่าง Class จะมีได้ทั้งทางเดียวและ 2 ทาง มีสัญลักษณ์ดังนี้

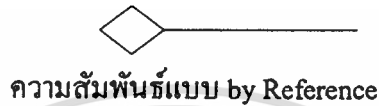
ความสัมพันธ์แบบทางเดียว 

ความสัมพันธ์แบบสองทาง 

รูปที่ 4.7 สัญลักษณ์ของ Association Relationship

- Aggregation เป็นรูปแบบพิเศษของ Association คือ เป็นความสัมพันธ์ระหว่าง Whole และ Parts ของมัน คือ Whole จะประกอบไปด้วย Parts ต่าง ๆ ของมัน ดังนั้น การเอกสารเป็นเอกสารที่ลงนามโดยผู้บริหารหรือการลงนามโดยผู้บริหารอื่น ไม่อนุญาตให้มีการแก้ไขเอกสารฉบับเดิมได้
- ไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คงอยู่ของ Parts จะต้องขึ้นกับ Whole หรือจะเรียกความสัมพันธ์แบบนี้ว่า “Whole-Part” ก็ได้ มีสัญลักษณ์ดังนี้

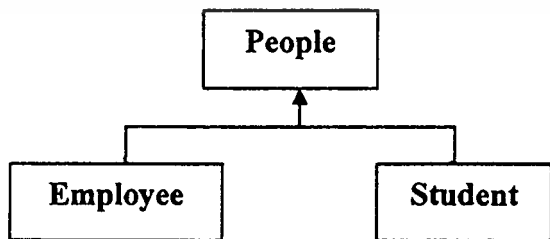


รูปที่ 4.8 สัญลักษณ์ของ Aggregation Relationship

- **Depends on** เป็นรูปแบบความสัมพันธ์แบบหนึ่งที่ใช้แสดงความสัมพันธ์กันระหว่าง Class 2 Class ในแง่ที่ Class หนึ่งเรียกใช้บริการของอีก Class หนึ่ง กล่าวคือ Class ของผู้ขอบริการขึ้นอยู่กับบริการของ Class ของผู้ให้บริการ แต่ไม่มีการขึ้นต่อกันภายในโครงสร้างของ Class มีสัญลักษณ์ดังนี้

รูปที่ 4.9 สัญลักษณ์ของ Depend on Relationship

- **Generalization** ความสัมพันธ์รูปแบบนี้ ใช้แสดงความสัมพันธ์ระหว่าง Class กับ Class ในแง่ที่ Class หนึ่งถ่ายทอดคุณสมบัติและโครงสร้างจากอีก Class หนึ่ง โดยเรียก Class ที่ถูกถ่ายทอดว่า Super Class และเรียก Class ที่ทำการถ่ายทอดว่า Sub Class มีสัญลักษณ์ดังนี้

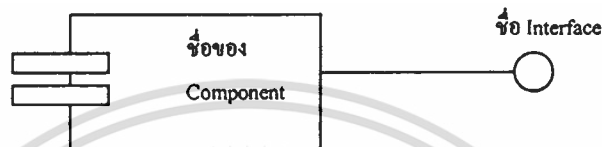


รูปที่ 4.10 ตัวอย่างความสัมพันธ์แบบ Generalization

4.4.3 Component Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็น Diagram ที่แสดงถึงโครงสร้างและความสัมพันธ์ระหว่างองค์ประกอบต่าง ๆ ของซอฟต์แวร์ ซึ่งองค์ประกอบดังกล่าวอาจเป็น Source Program หรือ Library หรือ Executable Program ก็ได้ โดย Component Diagram จะเป็นแบบจำลองที่ประกอบด้วยองค์ประกอบต่าง ๆ ของระบบเชื่อมโยงกันโดยใช้ความสัมพันธ์ในลักษณะที่ขึ้นต่อกันในรูปของเส้นประที่มีหัวลูกศรชี้จาก Component ถูก ไปยัง Component หลัก ดังรูป



รูปที่ 4.11 สัญลักษณ์ที่ใช้ภายใน Component Diagram

4.4.4 Sequence Diagram

เป็น Diagram ที่แสดงให้เห็นถึงการทำงานระหว่าง Object ต่าง ๆ ตามการส่งข้อความ และเมื่อเหตุการณ์ต่าง ๆ ได้เกิดขึ้น ซึ่งผู้เขียนโปรแกรมจะใช้ Diagram ตัวนี้ เป็นตัวช่วย เพื่อที่จะได้เขียนโปรแกรมให้ได้ตามที่ได้ออกแบบไว้

- Object คือ Instance ของ Class ที่ทำหน้าที่รับ-ส่งข้อความ (Message) จาก Object อื่น และทำการตอบสนองตามข้อความนั้น เพื่อให้เกิดการทำงานในขั้นตอนต่าง ๆ ของระบบ โดยจะใช้สัญลักษณ์เป็นเส้นตรงแนวดิ่งและมีชื่อของ Object และ Class กำหนดอยู่ด้านล่าง

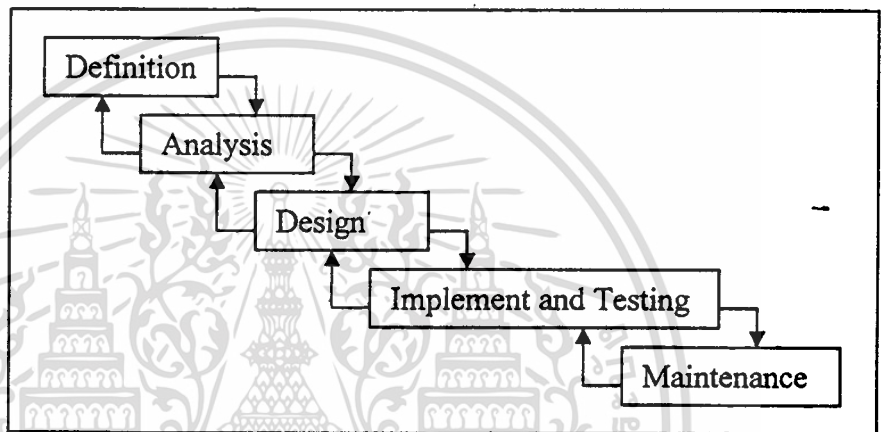
- Message เป็นข้อความที่ส่งไปมาระหว่าง Object ที่ถูกเรียก ซึ่ง Message จะประกอบไปด้วยส่วนประกอบต่าง ๆ ดังต่อไปนี้

- Result Name คือ ตัวแปรที่จะจัดเก็บค่าที่ได้จากการส่งข้อความ (Message)
- Parameters คือ Argument ต่าง ๆ ที่ส่งให้กับ Object เพื่อทำงานตาม Method ที่กำหนดด้วย Message นั้น ๆ
- Return Type คือ ประเภทของข้อมูลที่ส่งกลับมาหลังจากการทำงานของการส่ง Method นั้น ๆ ถ้า Method นั้น กำหนดว่ามีการส่งค่ากลับมา
- Constraint คือ เงื่อนไขการส่ง Message

4.5 Waterfall Process Model

ในการพัฒนาซอฟต์แวร์เป็นเรื่องที่ค่อนข้างซับซ้อน และมีรายละเอียดค่อนข้างมาก จึงได้มีการแบ่งกระบวนการพัฒนาซอฟต์แวร์ออกเป็น Phase ต่าง ๆ เรียกว่า Software Development Life Cycle เอกสารนี้เป็นเอกสารที่ส่งมอบไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Cycle Model ซึ่งจะแบ่งขั้นตอนการพัฒนาออกเป็น Phase ต่าง ๆ ทำให้การปฏิบัติ และติดตามผลของการพัฒนาง่ายขึ้น ซึ่งได้มีการกำหนดมาตรฐาน Software Development Life Cycle ไว้หลาย Model เช่น Waterfall Model, Rapid Prototyping Model, Spiral Model, Concurrent Development Model และ Formal Methods Model เป็นต้น ในที่นี้จะนำ Waterfall Model มาเป็นหลักในการประยุกต์ใช้กับการวิเคราะห์และออกแบบเชิงวัตถุ เพราะ Waterfall Model นั้น เป็น Model ที่นิยมและเข้าใจง่าย



รูปที่ 4.12 แผนภาพแสดง Waterfall Model

Waterfall Model จะแบ่ง Phase การทำงานออกเป็น 5 Phases คือ

1. Definition คือ Phase ของการให้คำนิยามต่าง ๆ ของระบบ
2. Analysis คือ Phase ของการวิเคราะห์ความต้องการ หรือปัญหาของระบบ
3. Design คือ Phase ของการออกแบบเพื่อสนองความต้องการ หรือเพื่อแก้ปัญหาของระบบ
4. Implement and Testing คือ Phase ของการปฏิบัติพัฒนาระบบ รวมถึงการทดสอบระบบ
5. Maintenance คือ Phase ของการบำรุงรักษาระบบ

โดยสามารถอธิบายแต่ละ Phase ได้ดังนี้

4.5.1 Phase 1: Definition

คือ Phase ของการให้คำนิยามต่าง ๆ ของระบบ เป็น Phase ที่สำคัญที่สุด เนื่องจากการศึกษาถึงความต้องการต่าง ๆ ของระบบ ปัญหาของระบบคืออะไร ขอบเขตของระบบงานที่สนใจมีแค่ไหน ความต้องการของผู้ใช้มีอะไรเพิ่มเติม ปรับปรุงบ้าง และมีคำศัพท์อะไรบ้าง จะต้อง

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีการให้คำนิยามของศัพท์เหล่านั้น เพื่อความเข้าใจที่ตรงกัน ถ้าใน Phase นี้ กำหนดคำนิยามของระบบไม่ชัดเจน ก็จะมีผลถึง Phase ต่อ ๆ ไปของการดำเนินงานด้วย ผลลัพธ์ที่ได้จาก Phase นี้เป็น Functional Model คือ Use Case Diagram และพจนานุกรมของข้อมูล (Data Dictionary)

4.5.2 Phase 2: Analysis

คือ Phase ของการวิเคราะห์ความต้องการ หรือปัญหาของระบบ ใน Phase นี้จะเป็นหลักสำคัญในการที่จะศึกษา และออกแบบถึงโครงสร้างของระบบงานตามขอบเขตความต้องการ ขอบเขตของปัญหา รวมถึงข้อจำกัดของระบบ ตาม Use Case Diagram ที่ได้จาก Phase 1 ใน Phase นี้ จะได้ผลลัพธ์เป็น Object Model ผู้วิเคราะห์จะทำการรวบรวม Class ที่สัมพันธ์กันขึ้นมาเป็นหมวดหมู่ เรียกว่า Component เพื่อเป็นการลดความซ้ำซ้อนของระบบ

4.5.3 Phase 3: Design

คือ Phase ของการออกแบบเพื่อสนองความต้องการ หรือเพื่อแก้ปัญหาของระบบ โดยการออกแบบระบบจะเริ่มเมื่อข้อมูลในช่วงของการวิเคราะห์เพียงพอที่จะเริ่มทำการปฏิบัติกับระบบแล้ว ในการออกแบบควรจะพิจารณาถึง Hardware ว่า ระบบเป็น Centralized หรือ Distributed ถ้าเป็นระบบแบบ Distributed มีการจัดการกับระบบความปลอดภัยอย่างไร จะใช้อะไรเป็นตัวติดต่อสื่อสาร ต้องการ Object เพิ่มเติมเพื่อช่วยอำนวยความสะดวกในการติดต่อสื่อสารหรือไม่ เป็นต้น แล้วพิจารณาว่า Software มี Class พิเศษอะไรบ้าง เพื่อสร้างหน้าต่างส่วนที่ติดต่อกับผู้ใช้ (User Interface) ต้องการ Class เพิ่มเติมเพื่อช่วยอำนวยความสะดวกในเรื่องของความสัมพันธ์ระหว่าง Class หรือไม่ ถ้าในระบบต้องการใช้ระบบฐานข้อมูล มี Class พิเศษที่ต้องใช้ในการติดต่อระหว่าง DBMS หรือไม่ ต้องมีการลดความซ้ำซ้อนของข้อมูลหรือไม่ ต้องทำการ Normalization หรือไม่ มีการใช้ Class ของบุคคลภายนอกหรือไม่ การติดต่อระหว่าง Object เป็นแบบพร้อมกัน (Synchronously) หรือเกิดขึ้นไม่พร้อมกัน (Asynchronously) เป็นต้น

ผลลัพธ์ที่ได้จาก Phase นี้ ควรจะได้ Class เพิ่มเติม เช่น Class เกี่ยวกับการติดต่อกับผู้ใช้งาน คือ หน้าต่าง ต่าง ๆ (GUI Classes), Class ที่จัดการและติดต่อกับฐานข้อมูล (Database Classes), การสนทนาระหว่าง Class (Dialogue) และประเภทของ Object ที่ต้องการเป็นแบบถาวร (Persistent Object), ความสัมพันธ์ระหว่าง Object, Transaction ต่าง ๆ ที่สนใจ Dynamic Model จะมี Sequence Diagram และ/หรือ State Diagram

4.5.4 Phase 4: Implement and Testing

คือ Phase ของการปฏิบัติพัฒนาระบบ รวมถึงการทดสอบระบบ เป็น Phase ของการเขียนโปรแกรมและทดสอบให้ได้เป็นไปตามการวิเคราะห์ โดยจะใช้การโปรแกรมเชิงวัตถุ (Object-oriented Programming) ซึ่งการแปลงจากการวิเคราะห์และออกแบบไปเป็นการโปรแกรม

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เชิงวัตถุ ส่วนการ Testing ก็เหมือนเป็นการตรวจสอบคุณภาพของโปรแกรมที่เราทำขึ้นมา ก็มีวิธีการตรวจสอบ Object รองรับ (Object Testing) การแปลงจากการวิเคราะห์และออกแบบไปเป็นการโปรแกรมเชิงวัตถุสามารถทำได้ไม่ยุ่งยาก ดังตารางที่ 4.1 เนื่องจากทั้งการวิเคราะห์และออกแบบเชิงวัตถุ กับการโปรแกรมเชิงวัตถุนั้น สนับสนุนคุณสมบัติของ Object-oriented ได้แก่ Encapsulation, Inheritance และ Polymorphism

ตารางที่ 4.1 การเทียบจาก UML ไปเป็นการโปรแกรมเชิงวัตถุ (ภาษา Java)

การวิเคราะห์และออกแบบระบบเชิงวัตถุ (UML notation)	การโปรแกรมเชิงวัตถุ (ภาษา Java)
Component	เป็น package ในภาษา Java
Class	Class
Attribute	Attribute
Method	Method
Attribute ต่าง ๆ ถ้าเป็น Read-only	เป็น Set Method
Attribute ต่าง ๆ ถ้าเป็น Read-Write	เป็น Get, Set method
Inherit	Sub Class จะต้อง Extends จาก Super Class
ถ้า Component มีการอ้างอิงถึง Module หรือ Component อื่น ๆ	จะต้อง Import Package นั้น ๆ
Public (+)	เป็น Public ใน Java
Private (-)	เป็น Private ใน Java
Protected (#)	เป็น Protected ใน Java
Implementation (?)	คือการกำหนด ณ ขณะ Implement เราอาจจะใช้ Private ในภาษา Java ก็ได้ แล้วแต่ขอบเขตกรณี

4.5.5 Phase 5: Maintenance

คือ Phase ของการบำรุงรักษาระบบ โดยในการ Maintenance สำหรับการพัฒนาระบบด้วยแนวคิดเชิงวัตถุจะทำได้สะดวกกว่า ยกตัวอย่างเช่น ถ้ามีการเปลี่ยนแปลงโครงสร้างของ Class ก็จะทำให้การเปลี่ยนแปลงและ Implement Class นี้ใหม่เท่านั้น Class อื่น ๆ ที่เกี่ยวข้องไม่จำเป็นต้องเปลี่ยนแปลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเห็นว่า ตั้งแต่ Phase 1 ถึง Phase 5 การวิเคราะห์และออกแบบเชิงวัตถุจะใช้เครื่องมือที่มีลักษณะเดียวกัน คือ อ้างอิงถึง Object มองทุกอย่างเป็น Object ทำให้เมื่อมีการเปลี่ยนแปลงแก้ไขสามารถทำได้ง่าย สะดวกต่อการจัดการ การส่งทอดระหว่าง Phase หนึ่งถึงอีก Phase หนึ่งจะไม่มีช่องว่าง เพราะอ้างอิงถึง Object เหมือนกัน และมี Object-oriented CASE Tool ที่สนับสนุนการวิเคราะห์และออกแบบเชิงวัตถุหลายบริษัทแล้วแต่จะเลือกใช้ เช่น Rose จากบริษัท Rational ดังนั้นวิธีการวิเคราะห์และออกแบบระบบเชิงวัตถุจึงเป็นวิธีการที่น่าสนใจและมีแนวโน้มที่ดีในอนาคต



บทที่ 5

Object-oriented CASE Tool: Rational Rose

5.1 Rational Rose 2000 Enterprise Edition

Rational Rose 2000 Enterprise Edition เป็น Visual Modeling Tool และ Object-oriented CASE Tool ที่สามารถสร้าง UML Diagram ทั้งหมด เพื่อใช้ในการจำลองระบบงานได้ ด้วยการวาด Diagram ต่าง ๆ โดยมีตัวจัดการ Diagram และกำหนดคุณสมบัติต่าง ๆ ของ Diagram ที่จำเป็น การเชื่อมโยงกันระหว่าง Diagram และการจัดการเอกสารจากแบบจำลองที่ได้สร้างขึ้น รวมทั้งมีตัวช่วยอัตโนมัติในการสร้าง Class และ Add-in ต่าง ๆ เพิ่มเติม เช่น Rational Rose J และ Rational Rose Oracle 8 ที่จะกล่าวถึงต่อไป

5.2 Rational Rose J

Rational Rose J เป็นส่วนเพิ่มเติมของ Rational Rose ที่มีคุณสมบัติดังต่อไปนี้

1. สนับสนุนวิวัฒนาการจากขั้นวิเคราะห์ไปจนถึงการออกแบบ
2. สนับสนุนการทำงานเป็นทีม โดยการทำ Configuration Management และ Version Control System
3. สามารถสร้าง Source Code ภาษา Java จาก Model
4. สามารถเพิ่มเติม Class Path ภายใน Model
5. สนับสนุนไฟล์ชนิด .zip, .jar และ .cab ใน Class Path
6. สามารถลากและวางไฟล์ชนิด .java, .class, .zip, .jar และ .cab เข้าไปใน Model โดยการทำ Reverse Engineering อย่างรวดเร็ว
7. ขยายไฟล์แบบอัตโนมัติสำหรับไฟล์ชนิด .zip, .jar และ .cab เมื่อขณะทำการ Reverse Engineering
8. สามารถออกแบบจำลอง ทำให้เห็นภาพของการสร้างโปรแกรมของภาษา Java ได้แก่ Package, Class, Interfaces, Imports, Inheritance, Fields, Methods และ Modifiers
9. สามารถควบคุมคุณสมบัติต่าง ๆ อย่างอัตโนมัติ ได้แก่ Finalizers, และ Static Initializers, และสามารถกำหนดชื่อ Field ได้เอง
10. สนับสนุนการ Application Partitioning และ Distribution

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. การ Mapping อย่างฉลาดจาก Model ไปสู่ Code
12. สนับสนุน Default Package ต่าง ๆ
13. ปรับปรุงการสนับสนุน Class ภายในและ Class อื่น ๆ ที่ไม่อยู่ในระบบ
14. การสร้าง Bean (Get/Set)
15. โปรแกรม Editor สามารถแสดงสีได้
16. สนับสนุน Framework ขนาดใหญ่ ประกอบไปด้วย Component ให้เลือกใช้งาน อาทิ
 - JDK Class Library
 - Java Database Connectivity (JDBC)
 - Java Generic Library (JGL)
 - Application Foundation Classes (AFC)
 - Windows Foundation Classes
 - Enterprise Java Framework (JNI, EJB, JTS)

5.2.1 ขั้นตอนการทำ Forward Engineering ด้วย Rational Rose J

Forward Engineering คือ กระบวนการสร้างโปรแกรมภาษา Java จาก Class หรือ Package หรือ Component ใน Rational Rose Model

ขั้นแรก จะต้องทำการกำหนด Class Path ให้ชี้ไปยัง Directory หรือไฟล์ ชนิด .jar, .cab, .zip ที่ต้องการใช้งาน โดยเพิ่มเติมในส่วนของ Directory ที่เพิ่มไปจาก Class Path Environment Variable ที่ถูกกำหนดไว้ก่อนแล้ว

ขั้นที่ 2 ให้ทำการสร้าง Class Diagram ที่ต้องการ ดังเช่นตัวอย่างต่อไปนี้ สร้าง Class “Preference”

ขั้นที่ 3 เมื่อทำการเลือก Class ที่ต้องการแล้ว เลือก Tools > Java > Generate และสามารถดูผลลัพธ์ได้โดยเลือก Tools > Java > Browse Java Source

เมื่อทำตามขั้นตอนแล้ว จะได้ผลลัพธ์ดังเป็น Source Code เป็นไฟล์ .java และถูกบันทึกอยู่ใน Directory ที่กำหนดไว้ใน Class Path

5.2.2 ขั้นตอนการทำ Reverse Engineering ด้วย Rational Rose J

Reverse Engineering คือ กระบวนการวิเคราะห์ Source Code ภาษา Java แล้วแปลงกลับไปเป็น Class หรือ Component Model ใน Rational Rose Model

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนแรก ทำเช่นเดียวกับการทำ Forward Engineering คือ ต้องมีการกำหนด Class Path ให้ชี้ไปยัง Directory หรือไฟล์ชนิด .jar, .cab, .zip ที่ต้องการใช้งาน โดยเพิ่มเติมในส่วน ของ Directory ที่เพิ่มไปจาก Class Path Environment Variable ที่ถูกกำหนดไว้ก่อนแล้ว

ขั้นตอนที่ 2 เลือกไฟล์ .jar, .class ที่ต้องการ โดยเข้าที่เมนู Tools > Java > Reverse Engineering Java

ขั้นตอนสุดท้าย หลังจากได้เลือกไฟล์เรียบร้อยแล้ว ให้กดปุ่ม Reverse โปรแกรม จะทำการ Reverse ออกมาเป็น Class Diagram ที่ต้องการ

5.3 Rational Rose Oracle8

ในทำนองเดียวกัน Rational Rose Oracle8 คือส่วนเพิ่มเติมของ Rational Rose ที่มี คุณสมบัติ ดังต่อไปนี้

1. สามารถสร้าง Object Models จาก Oracle8 Relational Schema และขยายความสามารถ ไปสู่มุมมองเชิงวัตถุของ Oracle8
2. สามารถทำให้มองเห็นภาพ หรือจำลอง Relational Database ที่มีอยู่เดิม และอำนวยความสะดวกในการค้นหา และประกอบเข้าด้วยกันกับ Business Object ซึ่งจะช่วยลด การลงทุนสำหรับการปรับปรุง Relational Database
3. สามารถทำให้ผู้พัฒนาระบบสร้างแบบจำลองทางธุรกิจในรูปแบบของภาษาที่ใช้ในการ พัฒนาโปรแกรมและฐานข้อมูล รวมทั้งตัวช่วยอัตโนมัติที่นำทางในการสร้าง Object Model ที่พัฒนาจาก Relational ไปสู่ Object-relational โดยสามารถสร้าง Relational Tables, Triggers, Object Types, Object Views, VARRAYs, Nested Tables และ คุณสมบัติเชิงวัตถุต่าง ๆ ใน Oracle8

5.3.1 การทำ database normalization

การจำลองแบบของข้อมูลสำหรับ Rational Rose Oracle8 ยังจำเป็นต้องผ่าน กระบวนการออกแบบฐานข้อมูล และการทำ Normalize ข้อมูลจริงเสียก่อน จากนั้นจึงจะนำตารางที่ ได้มาจากการ Normalize มาสร้างเป็น Class Diagram รวมไว้ใน Database Package แล้วจึงจะนำไป สร้างเป็น Database Schema ด้วย Rational Rose Oracle8 ซึ่งจะกล่าวถึงในหัวข้อต่อไป

5.3.2 ขั้นตอนการสร้าง database schema ของระบบด้วย Rational Rose Oracle8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนแรก จาก Class Diagram ใน Database Package เลือกเมนู Tools > Oracle8

> Schema Generation

ขั้นตอนที่ 2 เมื่อได้ผลลัพธ์แล้วให้ทำการกดปุ่ม Generate จะได้ Source Code (DDL)

ในกรณีต้องการให้มีการ Create Database ด้วย Rational Rose สามารถทำได้โดยการกดปุ่ม Execute โดยระบบจะต้องมีการติดตั้ง DBMS เสร็จสมบูรณ์แล้ว แต่ในกรณีที่ไม่ต้องการสามารถใช้วิธี copy Source Code เก็บไว้เพื่อนำไปใช้ Create Database ในภายหลัง ผลลัพธ์ที่ได้จาก Class Diagram ของระบบ คือ Source Code (DDL: Database Definition Language)



บทที่ 6

การวิเคราะห์และออกแบบระบบ BBL Portal

6.1 Use Case Diagram

6.1.1 การกำหนดแอกเตอร์

จากการวิเคราะห์ระบบ จะค้นหาแอกเตอร์ของระบบก่อน จากการศึกษาพบว่า ผู้ใช้ระบบคือ เจ้าหน้าที่ดูแลระบบ เจ้าหน้าที่จัดการเรื่องข้อมูล เจ้าหน้าที่จัดการเรื่องถาม-ตอบ สมาชิก และระบบงานหลัก ระบบงานภายนอกได้แก่ ระบบ Public Web Site (PWS) ระบบนายหน้าการซื้อขายหุ้น

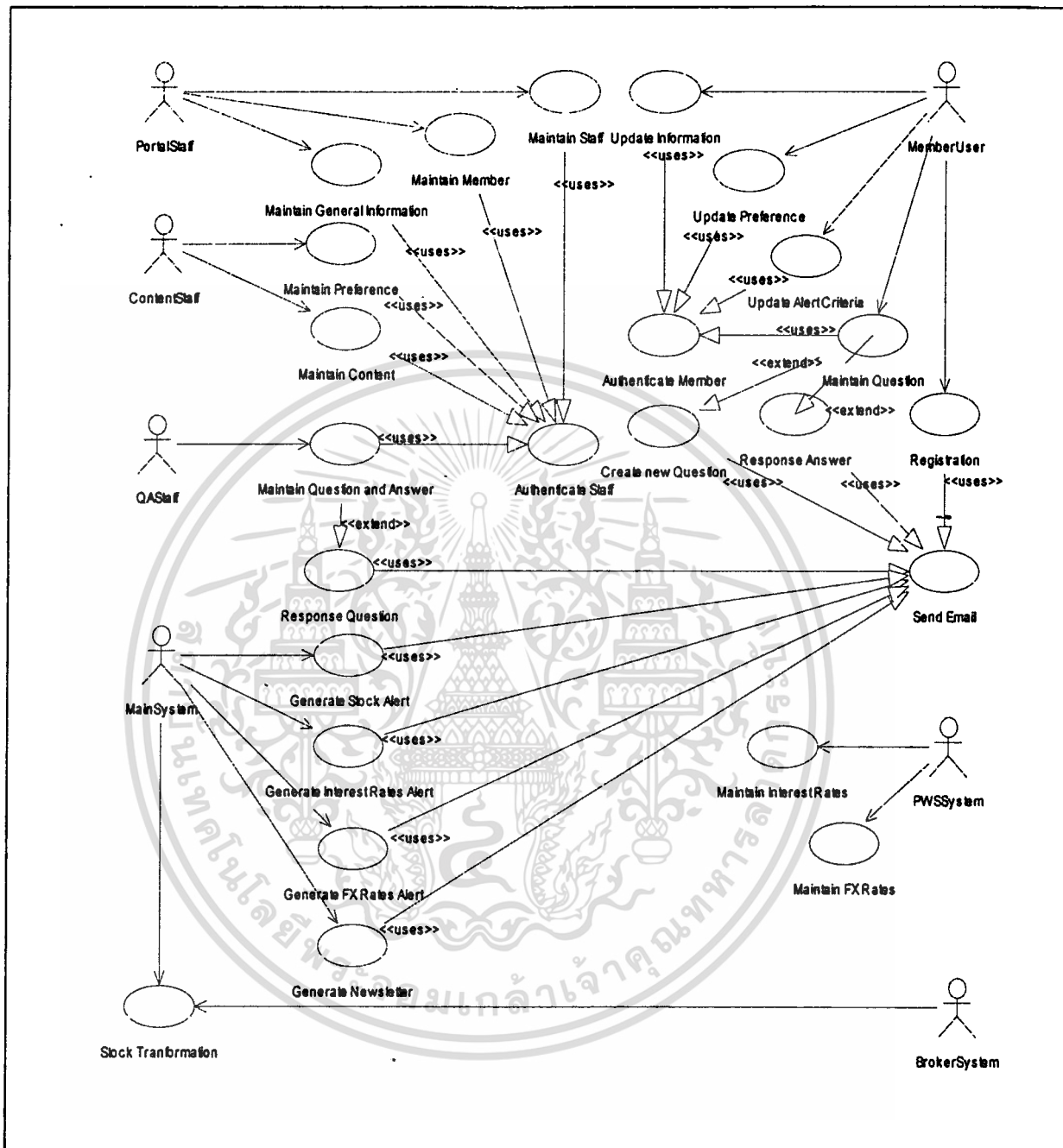
- เจ้าหน้าที่ดูแลระบบ (Portal Staff) ทำหน้าที่จัดการข้อมูลเจ้าหน้าที่ที่จะเข้าใช้ระบบ ข้อมูลสมาชิก และข้อมูลประกอบทั่วไปของระบบ
- เจ้าหน้าที่จัดการเรื่องข้อมูล (ContentStaff) ทำหน้าที่จัดการทางด้านข้อมูลเพื่อนำมาเผยแพร่ในระบบ เป็นข้อมูลข่าวสารหรือเรื่องราวต่าง ๆ ที่น่าสนใจเพื่อเป็นประโยชน์ต่อสมาชิก
- เจ้าหน้าที่จัดการเรื่องถาม-ตอบ (QAStaff) ทำหน้าที่ดูแลข้อมูลที่มีการถาม-ตอบจากสมาชิกมายังระบบ
- ระบบหลัก (MainSystem) ทำหน้าที่จัดการการทำงานอัตโนมัติทั้งหมดของระบบ เช่น การดึงข้อมูลหุ้นจากระบบนายหน้าซื้อขายหุ้น รวมถึงการสร้างและส่งอีเมลไปยังสมาชิกตามกลุ่มข้อมูลข่าวสาร และอัตราต่าง ๆ ที่สมาชิกสนใจ
- สมาชิก (Member) เป็นผู้ที่เข้ามาเพื่อตรวจสอบข้อมูลในด้านต่าง ๆ ที่สนใจ รวมถึงเป็นการศึกษาค้นคว้าข้อมูลเกี่ยวกับธนาคาร และยังอาจจะเป็นลูกค้าในอนาคต
- ระบบ PWS (PWS System) เป็นระบบที่มีอยู่เดิม ทำหน้าที่เผยแพร่ข้อมูลข่าวสารแก่สาธารณชน และบุคคลทั่วไป โดยระบบได้มีการขอใช้ข้อมูลอัตราดอกเบี้ย และอัตราแลกเปลี่ยนร่วมกับระบบ PWS

- ระบบนายหน้าซื้อขายหุ้น (BrokerSystem) เป็นระบบภายนอกธนาคารที่ได้มีการทำธุรกิจเกี่ยวกับการซื้อขายหุ้น โดยระบบได้มีการขอดึงข้อมูลเกี่ยวกับราคาหุ้น เพื่อนำมาใช้ในระบบ

6.1.2 การสร้างยูสเคสของระบบ

ต่อมาจะเป็นขั้นตอนการสร้างยูสเคสโคแอมแกรมแสดงการติดต่อระหว่างผู้ใช้กับระบบ โดยแสดงถึงการทำงานของผู้ใช้และหน้าที่ของระบบ เช่น สมาชิกทำการลงทะเบียนขอเข้าใช้ระบบ และระบบได้มีการส่งอีเมลล์เพื่อทำการยืนยันการลงทะเบียน หรือสมาชิกเข้ามาปรับปรุงเงื่อนไขการดูราคาหุ้นที่สนใจ โดยจะต้องผ่านการตรวจสอบจากรหัสผู้ใช้ และรหัสผ่านจากระบบก่อน เป็นต้น ดังแสดงในรูปที่ 6.1





รูปที่ 6.1 ยูสเคสไดอะแกรมของระบบ BBL Portal

จากยูสเคสไดอะแกรมที่ได้จากการศึกษาความต้องการของระบบ และทำซ้ำหลาย ๆ รอบเพื่อให้มีความถูกต้อง มีรายละเอียดแต่ละยูสเคสดังต่อไปนี้

- 1) ชื่อยูสเคส : Registration คือ ยูสเคสที่อธิบายเกี่ยวกับการลงทะเบียนของสมาชิก
แอกเตอร์ : Member

รายละเอียด :

เอกสารนี้เป็นเอกสารที่ส่งไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. สมาชิกเลือกหัวข้อการลงทะเบียน
 2. สมาชิกกรอกรายละเอียดทั่วไป
 3. สมาชิกเลือกข้อมูลข่าวสารที่สนใจ (อย่างน้อย 3 กลุ่ม) อัตราดอกเบี้ย อัตราแลกเปลี่ยน หน้าที่สนใจ
 4. ระบบตรวจสอบความครบถ้วนถูกต้องของข้อมูล
 5. ระบบส่งอีเมลยืนยันการลงทะเบียน
 6. สมาชิกได้รับอีเมลยืนยันการลงทะเบียน และอนุญาตให้เข้าใช้ระบบได้ด้วยรหัสชื่อและรหัสผ่านที่กรอกไว้
- 2) ชื่อยูสเคส : Authenticate Member คือ ยูสเคสที่อธิบายเกี่ยวกับการตรวจสอบตัวตนของสมาชิก
- แอดเดอร์ : Member
- รายละเอียด :
1. สมาชิกเข้ามากรอกรหัสชื่อและรหัสผ่านเพื่อเข้าสู่ระบบ
 2. ระบบจะตรวจสอบความถูกต้องของรหัสชื่อและรหัสผ่านเพื่อยืนยันสมาชิก และเลือกข้อมูลสมาชิกมาทำการแสดงผลที่หน้าจอหลักของสมาชิก
- 3) ชื่อยูสเคส : Update Member Information คือ ยูสเคสที่อธิบายการปรับปรุงข้อมูลของสมาชิก
- แอดเดอร์ : Member
- รายละเอียด :
1. สมาชิกต้องทำการตรวจสอบตัวตนก่อน จากนั้นเลือกหัวข้อปรับปรุงข้อมูลสมาชิก
 2. ระบบจะดึงข้อมูลสมาชิกแล้วนำมาแสดงบนฟอร์มเพื่อเตรียมให้ปรับปรุงข้อมูล
 3. สมาชิกทำการปรับปรุงข้อมูลบนฟอร์ม แล้วยืนยันความถูกต้อง
 4. ระบบทำการบันทึกข้อมูลที่ปรับปรุงใหม่เข้าสู่ระบบ
- 4) ชื่อยูสเคส : Update Member Preference คือ ยูสเคสที่อธิบายการปรับปรุงกลุ่มข้อมูลข่าวสารที่น่าสนใจ ในการแสดงบนหน้าจอหลักของสมาชิก
- แอดเดอร์ : Member
- รายละเอียด :
1. สมาชิกต้องทำการตรวจสอบตัวตนก่อน จากนั้นเลือกหัวข้อปรับปรุงข้อมูลข่าวสารที่สนใจ

2. ระบบจะดึงข้อมูลสมาชิกเกี่ยวกับข้อมูลข่าวสารที่สนใจ แล้วนำมาแสดงบนฟอรัมเพื่อเตรียมให้ปรับปรุงข้อมูล
 3. สมาชิกทำการปรับปรุงข้อมูลข้อมูลข่าวสารที่สนใจบนฟอรัม แล้วยืนยันความถูกต้อง
 4. ระบบทำการบันทึกข้อมูลที่ปรับปรุงใหม่เข้าสู่ระบบ
- 5) ชื่อยูสเคส : Update Member Alert Criteria คือ ยูสเคสที่อธิบายการปรับปรุงข้อมูลข่าวสาร อัตราดอกเบี้ย อัตราแลกเปลี่ยน และมูลค่าหุ้นที่น่าสนใจ

แอกเตอร์ : Member

รายละเอียด :

1. สมาชิกต้องทำการตรวจสอบตัวตนก่อน จากนั้นเลือกหัวข้อปรับปรุงอัตราดอกเบี้ย อัตราแลกเปลี่ยน และหุ้นที่น่าสนใจ
 2. ระบบจะดึงข้อมูลสมาชิกเกี่ยวกับอัตราดอกเบี้ย อัตราแลกเปลี่ยน และหุ้นที่น่าสนใจ แล้วนำมาแสดงบนฟอรัมเพื่อเตรียมให้ปรับปรุงข้อมูล
 3. สมาชิกทำการปรับปรุงข้อมูลอัตราดอกเบี้ย อัตราแลกเปลี่ยน และหุ้นที่น่าสนใจบนฟอรัม แล้วยืนยันความถูกต้อง
 4. ระบบทำการบันทึกข้อมูลที่ปรับปรุงใหม่เข้าสู่ระบบ
- 6) ชื่อยูสเคส : Maintain Member Question คือ ยูสเคสที่อธิบายการจัดการเรื่องคำถามของสมาชิก

แอกเตอร์ : Member

รายละเอียด :

1. สมาชิกต้องทำการตรวจสอบตัวตนก่อน จากนั้นเลือกหัวข้อถาม-ตอบ
 2. ระบบจะแสดงรายการถาม-ตอบของสมาชิกมาแสดง เพื่อเตรียมให้เลือกเพื่ออ่านหรือแจ้งข้อมูลเพิ่มเติม หรือเพิ่มคำถามใหม่
 3. ในกรณีที่สมาชิกต้องการลบคำถามที่ไม่ต้องการ สามารถเลือกลบจากรายการ
- 7) ชื่อยูสเคส : Create new Question คือ ยูสเคสที่อธิบายการสร้างคำถามใหม่ของสมาชิก

แอกเตอร์ : Member

รายละเอียด :

1. สมาชิกต้องทำการตรวจสอบตัวตนก่อน จากนั้นเลือกหัวข้อถาม-ตอบ และเลือกหัวข้อเพิ่มคำถามใหม่ จากหน้าจอแสดงรายการถาม-ตอบของสมาชิก
2. สมาชิกกรอกข้อมูลเกี่ยวกับคำถามตามฟอรัมที่กำหนด แล้วทำการยืนยัน

3. ระบบจะทำการตรวจสอบความถูกต้องก่อนบันทึกเข้าสู่ระบบ
 4. ระบบจะทำการส่งอีเมลไปยัง เจ้าหน้าที่จัดการเรื่องถามตอบ (QAStaff)
- 8) ชื่อยูสเคส : Response Staff Answer คือ ยูสเคสที่อธิบายการสอบถามเพิ่มเติม หรือแจ้งข้อมูลเพิ่มเติมจากคำถามเดิมของสมาชิก

แอกเตอร์ : Member

รายละเอียด :

1. สมาชิกต้องทำการตรวจสอบตัวตนก่อน จากนั้นเลือกหัวข้อถาม-ตอบ และเลือกคำถามที่ต้องการจากหน้าจอแสดงรายการถาม-ตอบ
2. สมาชิกกรอกข้อมูลเพิ่มเติมหรือคำถามต่อเนื่องตามฟอร์มที่กำหนด แล้วทำการยืนยัน
3. ระบบจะทำการตรวจสอบความถูกต้องก่อนบันทึกเข้าสู่ระบบ
4. ระบบจะทำการส่งอีเมลไปยัง เจ้าหน้าที่จัดการเรื่องถามตอบ (QAStaff) ทุกครั้งที่มีการสอบถาม หรือแจ้งข้อมูลเพิ่มเติม

- 9) ชื่อยูสเคส : Authenticate Staff คือ ยูสเคสที่อธิบายการตรวจสอบตัวตนของเจ้าหน้าที่
แอกเตอร์ : -

รายละเอียด :

1. เจ้าหน้าที่เข้ามากรอกรหัสชื่อและรหัสผ่านเพื่อเข้าสู่ระบบ
 2. ระบบจะตรวจสอบความถูกต้องของรหัสชื่อและรหัสผ่านเพื่อยืนยันเจ้าหน้าที่ และเลือกข้อมูลสมาชิกมาทำการแสดงผลที่หน้าจอหลักของสมาชิก
- 10) ชื่อยูสเคส : Maintain Staff Information คือ ยูสเคสที่อธิบายการจัดการข้อมูลเจ้าหน้าที่
แอกเตอร์ : PortalStaff
- รายละเอียด :
1. เจ้าหน้าที่ดูแลระบบ (PortalStaff) ต้องทำการตรวจสอบตัวตนก่อน แล้วจึงเลือกหัวข้อการจัดการข้อมูลเจ้าหน้าที่
 2. ระบบจะแสดงรายการเจ้าหน้าที่ และสามารถกำหนดเงื่อนไขในการแสดงรายการ
 3. ในกรณีต้องการลบข้อมูลเจ้าหน้าที่ ต้องเลือกเจ้าหน้าที่ที่ต้องการ แล้วระบบจะแสดงรายละเอียดของเจ้าหน้าที่ที่ต้องการขึ้นมาก่อน แล้วจึงยืนยันการลบ จากนั้นระบบจะทำการลบข้อมูลออกจากระบบ

4. ในกรณีต้องการเพิ่มข้อมูลเจ้าหน้าที่ สามารถเลือกหัวข้อเพิ่มเจ้าหน้าที่ แล้วกรอกข้อมูลตามฟอร์ม แล้วทำการยืนยันข้อมูล จากนั้นระบบจะทำการตรวจสอบความถูกต้องของข้อมูลก่อน แล้วจึงบันทึกสู่ระบบ
 5. ในกรณีต้องการแก้ไขข้อมูลเจ้าหน้าที่ ต้องเลือกเจ้าหน้าที่ที่ต้องการ แล้วกรอกข้อมูลตามฟอร์ม แล้วทำการยืนยันข้อมูล จากนั้นระบบจะทำการตรวจสอบความถูกต้องของข้อมูลก่อน แล้วจึงบันทึกสู่ระบบ
- 11) ชื่อยูสเคส : Maintain Member Information คือ ยูสเคสที่อธิบายการจัดการข้อมูลของสมาชิก
- แอดเดอร์ : PortalStaff
- รายละเอียด :
1. เจ้าหน้าที่ดูแลระบบ (PortalStaff) ต้องทำการตรวจสอบตัวตนก่อน แล้วจึงเลือกหัวข้อการจัดการข้อมูลสมาชิก
 2. ระบบจะแสดงรายการสมาชิก และสามารถกำหนดเงื่อนไขในการแสดงรายการ
 3. ในกรณีต้องการลบข้อมูลสมาชิก ต้องเลือกสมาชิกที่ต้องการ แล้วระบบจะแสดงรายละเอียดของสมาชิกที่ต้องการขึ้นมาก่อน แล้วจึงยืนยันการลบ จากนั้นระบบจะทำการลบข้อมูลออกจากระบบ
- 12) ชื่อยูสเคส : Maintain General Information คือ ยูสเคสที่อธิบายการจัดการข้อมูลทั่วไป เช่น ข้อมูลเมือง ข้อมูลประเทศ ข้อมูลอาชีพ เป็นต้น
- แอดเดอร์ : PortalStaff
- รายละเอียด :
1. เจ้าหน้าที่ดูแลระบบ (PortalStaff) ต้องทำการตรวจสอบตัวตนก่อน แล้วจึงเลือกหัวข้อการจัดการข้อมูลที่ต้องการ
 2. ระบบจะแสดงรายการข้อมูลทั่วไปที่ต้องการ และสามารถกำหนดเงื่อนไขในการแสดงรายการ
 3. ในกรณีต้องการลบข้อมูลทั่วไป ต้องเลือกข้อมูลทั่วไปที่ต้องการ แล้วระบบจะแสดงรายละเอียดของข้อมูลทั่วไปที่ต้องการขึ้นมาก่อน แล้วจึงยืนยันการลบ จากนั้นระบบจะทำการลบข้อมูลออกจากระบบ แต่ในกรณีที่มิข้อมูลอื่นอ้างอิงข้อมูลที่ต้องการลบนั้น ระบบจะยังไม่ทำการลบ ต้องไปทำการลบข้อมูลที่อ้างอิงนั้นก่อน

4. ในกรณีต้องการเพิ่มข้อมูลทั่วไป สามารถเลือกหัวข้อเพิ่มข้อมูลทั่วไป แล้วกรอกข้อมูลตามฟอร์ม แล้วทำการยืนยันข้อมูล จากนั้นระบบจะทำการตรวจสอบความถูกต้องของข้อมูลก่อน แล้วจึงบันทึกสู่ระบบ
5. ในกรณีต้องการแก้ไขข้อมูลทั่วไป ต้องเลือกข้อมูลทั่วไปที่ต้องการ แล้วกรอกข้อมูลตามฟอร์ม แล้วทำการยืนยันข้อมูล จากนั้นระบบจะทำการตรวจสอบความถูกต้องของข้อมูลก่อน แล้วจึงบันทึกสู่ระบบ

13) ชื่อยูสเคส : Maintain Preference คือ ยูสเคสที่อธิบายการจัดการกลุ่มข้อมูลข่าวสาร

แอดเดอร์ : ContentStaff

รายละเอียด :

1. เจ้าหน้าที่ดูแลระบบ (ContentStaff) ต้องทำการตรวจสอบตัวตนก่อน แล้วจึงเลือกหัวข้อการจัดการข้อมูลกลุ่มข้อมูลข่าวสาร
2. ระบบจะแสดงรายการกลุ่มข้อมูลข่าวสาร และสามารถกำหนดเงื่อนไขในการแสดงรายการ
3. ในกรณีต้องการลบข้อมูลกลุ่มข้อมูลข่าวสาร ต้องเลือกกลุ่มข้อมูลข่าวสารที่ต้องการ แล้วระบบจะแสดงรายละเอียดของกลุ่มข้อมูลข่าวสารต้องการขึ้นมาก่อน แล้วจึงยืนยันการลบ จากนั้นระบบจะทำการลบข้อมูลออกจากระบบ แต่ในกรณีที่ข้อมูลอื่นอ้างอิงข้อมูลที่ต้องการลบนั้น ระบบจะยังไม่ทำการลบ ต้องไปทำการลบข้อมูลที่อ้างอิงนั้นก่อน
4. ในกรณีต้องการเพิ่มข้อมูลกลุ่มข้อมูลข่าวสาร สามารถเลือกหัวข้อเพิ่มกลุ่มข้อมูลข่าวสาร แล้วกรอกข้อมูลตามฟอร์ม แล้วทำการยืนยันข้อมูล จากนั้นระบบจะทำการตรวจสอบความถูกต้องของข้อมูลก่อน แล้วจึงบันทึกสู่ระบบ
5. ในกรณีต้องการแก้ไขข้อมูลกลุ่มข้อมูลข่าวสาร ต้องเลือกกลุ่มข้อมูลข่าวสารที่ต้องการ แล้วกรอกข้อมูลตามฟอร์ม แล้วทำการยืนยันข้อมูล จากนั้นระบบจะทำการตรวจสอบความถูกต้องของข้อมูลก่อน แล้วจึงบันทึกสู่ระบบ

14) ชื่อยูสเคส : Maintain Content คือ ยูสเคสที่อธิบายการจัดการข้อมูลข่าวสารที่น่าสนใจ

แอดเดอร์ : ContentStaff

รายละเอียด :

1. เจ้าหน้าที่ดูแลระบบ (ContentStaff) ต้องทำการตรวจสอบตัวตนก่อน แล้วจึงเลือกหัวข้อการจัดการข้อมูลกลุ่มข้อมูลข่าวสาร

2. ระบบจะแสดงรายการข้อมูลข่าวสาร และสามารถกำหนดเงื่อนไขในการแสดงรายการ
 3. ในกรณีต้องการลบข้อมูลข่าวสาร ต้องเลือกข้อมูลข่าวสารที่ต้องการ แล้วระบบจะแสดงรายละเอียดของข้อมูลข่าวสารที่ต้องการขึ้นมาก่อน แล้วจึงยืนยันการลบ จากนั้นระบบจะทำการลบข้อมูลออกจากระบบ แต่ในกรณีที่มีข้อมูลอื่นอ้างอิงข้อมูลที่ต้องการลบนั้น ระบบจะยังไม่ทำการลบ ต้องไปทำการลบข้อมูลที่อ้างอิงนั้นก่อน
 4. ในกรณีต้องการเพิ่มข้อมูลข่าวสาร สามารถเลือกหัวข้อเพิ่มข้อมูลข่าวสาร แล้วกรอกข้อมูลตามฟอร์ม แล้วทำการยืนยันข้อมูล จากนั้นระบบจะทำการตรวจสอบความถูกต้องของข้อมูลก่อน แล้วจึงบันทึกสู่ระบบ
 5. ในกรณีต้องการแก้ไขข้อมูลข่าวสาร ต้องเลือกข้อมูลข่าวสารที่ต้องการ แล้วกรอกข้อมูลตามฟอร์ม แล้วทำการยืนยันข้อมูล จากนั้นระบบจะทำการตรวจสอบความถูกต้องของข้อมูลก่อน แล้วจึงบันทึกสู่ระบบ
 6. ในการเพิ่มเติมหรือแก้ไขข้อมูลข่าวสารนั้น ระบบจะเตรียมกลุ่มของย่อหน้าว่าง (ข้อความ รูป ตำแหน่งรูป) ให้ 5 กลุ่มย่อหน้า ต่อจากกลุ่มข้อความเดิมเสมอ
- 15) ชื่อยูสเคส : Maintain QA คือ ยูสเคสที่อธิบายการจัดการข้อมูลคำถามของเจ้าหน้าที่
 แอคเตอร์ : QAStaff
 รายละเอียด :
1. เจ้าหน้าที่ดูแลระบบ (QAStaff) ต้องทำการตรวจสอบตัวตนก่อน แล้วจึงเลือกหัวข้อการจัดการข้อมูลถาม-ตอบ
 2. ในกรณีต้องการลบข้อมูลถาม-ตอบที่ไม่เหมาะสม หรือไม่ต้องการ ต้องเลือกคำถามที่ต้องการ แล้วระบบจะแสดงรายละเอียดของข้อมูลถาม-ตอบที่ต้องการขึ้นมาก่อน แล้วจึงยืนยันการลบ จากนั้นระบบจะทำการลบข้อมูลออกจากระบบ
- 16) ชื่อยูสเคส : Response Member QA คือ ยูสเคสที่อธิบายการตอบคำถามของเจ้าหน้าที่
 แอคเตอร์ : QAStaff
 รายละเอียด :
1. เจ้าหน้าที่ดูแลระบบ (QAStaff) ต้องทำการตรวจสอบตัวตนก่อน แล้วจึงเลือกหัวข้อการจัดการข้อมูลถาม-ตอบ จากนั้นเลือกคำถามที่ต้องการ แล้วระบบจะแสดงรายละเอียดของข้อมูลถาม-ตอบที่ต้องการขึ้นมาก่อน

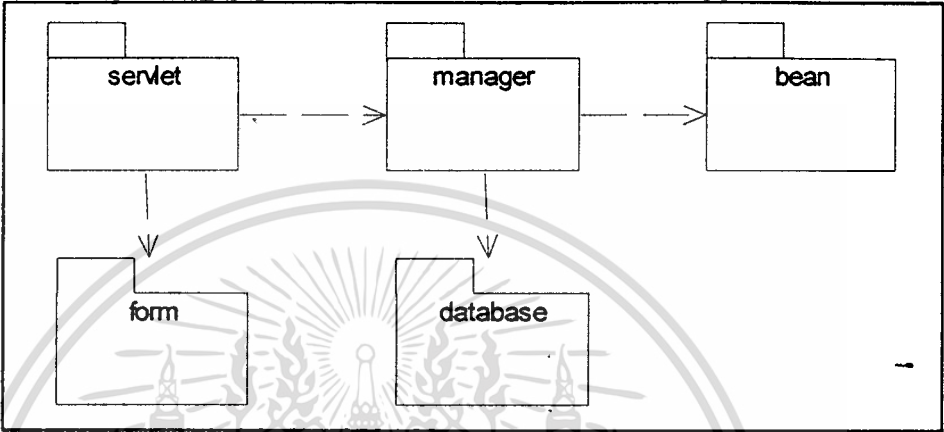
2. เจ้าหน้าที่ทำการกรอกรวบรวมรายละเอียดยื่นมา พร้อมรายละเอียด และข้อมูลถาม-ตอบก่อนหน้านั้น แล้วจึงยืนยันข้อมูล จากนั้นระบบจะทำการตรวจสอบความถูกต้องก่อนบันทึกข้อมูล
 3. ระบบจะทำการส่งอีเมลแจ้งเตือนไปยังสมาชิกเจ้าของคำถามทุกครั้งที่มีการตอบคำถาม หรือแจ้งข้อมูลเพิ่มเติม
- 17) ชื่อยูสเคส : Stock Transformation คือ ยูสเคสที่อธิบายการดึงข้อมูลหุ้นจากระบบหน้าซื้อขายหุ้น
- แอกเตอร์ : MainSystem
- รายละเอียด :
1. ระบบจะทำการดึงข้อมูลหุ้นจากระบบหน้าซื้อขายหุ้น ทุก 5 นาที ตามเวลาที่ตลาดหุ้นเปิดทำการ (ทุกวันวันหยุด) โดยอัตโนมัติ เพื่อเตรียมมาใช้ในระบบต่อไป
- 18) ชื่อยูสเคส : Generate Stock Alert คือ ยูสเคสที่อธิบายการสร้างอีเมลเกี่ยวกับราคาหุ้นตามเงื่อนไขของสมาชิก และทำการส่งไปให้สมาชิก
- แอกเตอร์ : MainSystem
- รายละเอียด :
1. ระบบจะทำการสร้างข้อมูลอีเมลทุกครั้งที่มีการเปลี่ยนแปลงข้อมูลหุ้น (โดยการตั้ง Trigger) โดยอิงตามเงื่อนไขที่สมาชิกเลือกไว้ จากนั้นจะทำการส่งอีเมลแจ้งเตือนไปยังสมาชิกในกรณีโดยตรงตามเงื่อนไข
- 19) ชื่อยูสเคส : Generate Interest Rate Alert คือ ยูสเคสที่อธิบายการสร้างอีเมลเกี่ยวกับอัตราดอกเบี้ยตามเงื่อนไขของสมาชิก และทำการส่งไปให้สมาชิก
- แอกเตอร์ : MainSystem
- รายละเอียด :
1. ระบบจะทำการสร้างข้อมูลอีเมลทุกครั้งที่มีการเปลี่ยนแปลงข้อมูลอัตราดอกเบี้ย (โดยการตั้ง Trigger) โดยอิงตามเงื่อนไขที่สมาชิกเลือกไว้ จากนั้นจะทำการส่งอีเมลแจ้งเตือนไปยังสมาชิกในกรณีโดยตรงตามเงื่อนไข
- 20) ชื่อยูสเคส : Generate FX Rate Alert คือ ยูสเคสที่อธิบายการสร้างอีเมลเกี่ยวกับอัตราแลกเปลี่ยนตามเงื่อนไขของสมาชิก และทำการส่งไปให้สมาชิก
- แอกเตอร์ : MainSystem
- รายละเอียด :

1. ระบบจะทำการสร้างข้อมูลอีเมลทุกครั้งที่มีการเปลี่ยนแปลงข้อมูลอัตราแลกเปลี่ยน (โดยการตั้ง Trigger) โดยอิงตามเงื่อนไขที่สมาชิกเลือกไว้ จากนั้นจะทำการส่งอีเมลแจ้งเตือนไปยังสมาชิกในกรณีตรงตามเงื่อนไข
- 21) ชื่อยูสเคส : Generate Newsletter คือ ยูสเคสที่อธิบายการสร้างอีเมลเกี่ยวกับข้อมูลข่าวสารตามเงื่อนไขของสมาชิก และทำการส่งไปให้สมาชิก
- แอกเตอร์ : MainSystem
- รายละเอียด :
1. ระบบจะทำการสร้างข้อมูลอีเมลวันละ 1 ครั้ง ในช่วง 1.00 – 6.00 โดยอิงตามเงื่อนไขที่สมาชิกเลือกไว้ จากนั้นจะทำการส่งอีเมลแจ้งเตือนไปยังสมาชิกในกรณีตรงตามเงื่อนไข
- 22) ชื่อยูสเคส : Send Email คือ ยูสเคสที่อธิบายเกี่ยวกับการส่งอีเมล
- แอกเตอร์ : -
- รายละเอียด :
1. เป็นโปรแกรมย่อยสำหรับส่งอีเมล
 2. ส่วนงานย่อยที่มีการส่งอีเมล จะจัดเตรียมข้อมูลตามข้อตกลงเกี่ยวกับรูปแบบข้อมูลที่ใช้ในการส่งอีเมล แล้วส่งข้อมูลนั้นมายังโปรแกรมย่อยนี้ จากนั้นจึงทำการส่งอีเมลออกไปตามข้อมูลที่จัดเตรียมมา
- 23) ชื่อยูสเคส : Maintain Interest Rates คือ ยูสเคสที่อธิบายการจัดการข้อมูลอัตราดอกเบี้ยโดยระบบ PWS
- แอกเตอร์ : PWS System
- รายละเอียด :
- เป็นการทำงานภายในระบบ PWS
- 24) ชื่อยูสเคส : Maintain FX Rates คือ ยูสเคสที่อธิบายการจัดการข้อมูลอัตราแลกเปลี่ยนโดยระบบ PWS
- แอกเตอร์ : PWSSystem
- เป็นการทำงานภายในระบบ PWS

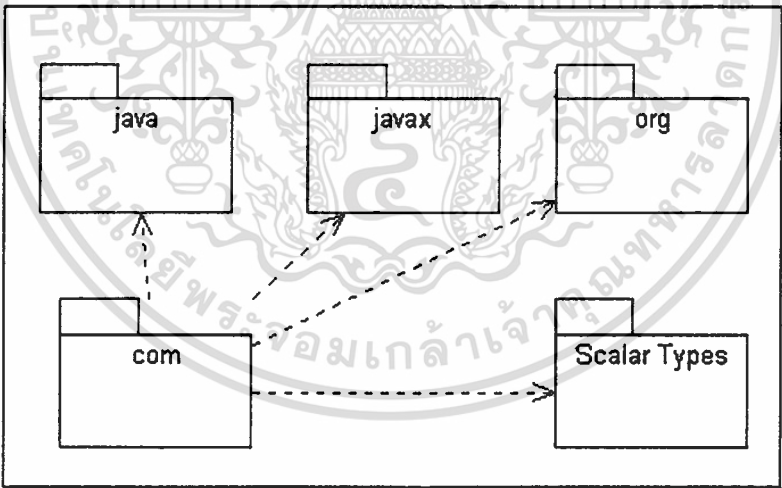
6.2 Component Diagram

เมื่อผ่าน Phase ของการวิเคราะห์ (Analysis) ทำให้สามารถออกแบบ Class Diagram โดย Packages ภายในที่จะต้องสร้างขึ้นเอง มีด้วยกัน 5 Packages ได้แก่ Database, Bean, Manager, เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ในการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Servlet, Form ส่วน Package ภายนอกที่นำมาใช้ได้แก่ Java และ Javax ซึ่งมี API ต่าง ไว้ให้ใช้งาน และ Scalar Type ซึ่งมีประเภทข้อมูลที่ใช้ใน Oracle ให้เลือกใช้ ดังรูปที่ 6.2 และ 6.3



รูปที่ 6.2 Package ภายในทั้งหมด (Logical View)



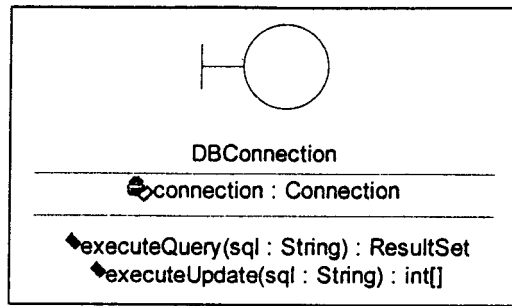
รูปที่ 6.3 Package ภายนอกทั้งหมด (Logical View)

6.3 Class Diagram

เมื่อผ่านขั้นวิเคราะห์ (Analysis เข้าสู่ขั้นออกแบบ (Design) ทำให้สามารถออกแบบ Class Diagram สำหรับอธิบายรายละเอียดของ Class ภายใน โปรแกรมที่ต้องสร้างขึ้น ได้แก่ Database Package, Bean Package, Manager Package, Servlet Package, Form Package

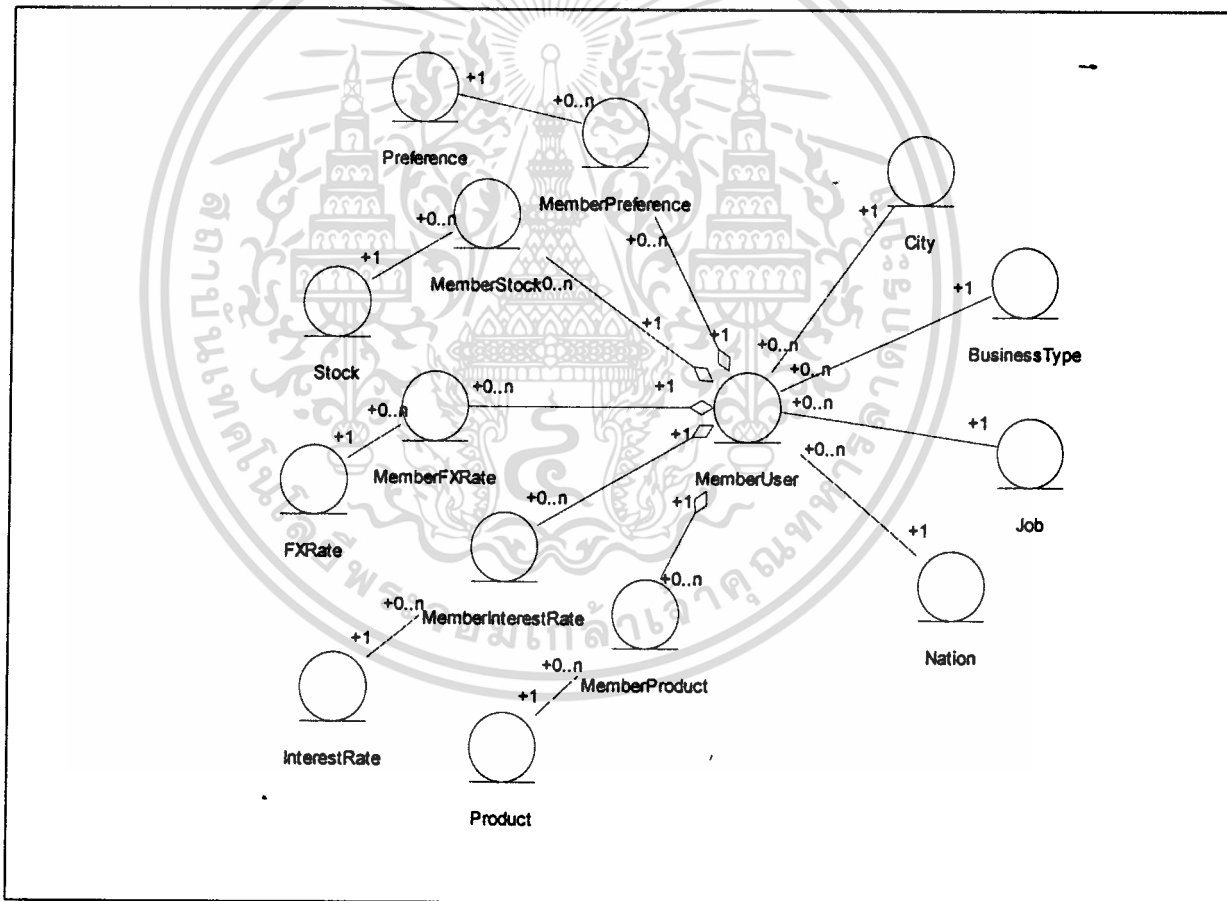
6.3.1 Class Diagram ภายใน Database Package

เอกสารนี้เป็นเอกสารที่ประกอบด้วย 1 Class ดังรูปที่ 6.4 ศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

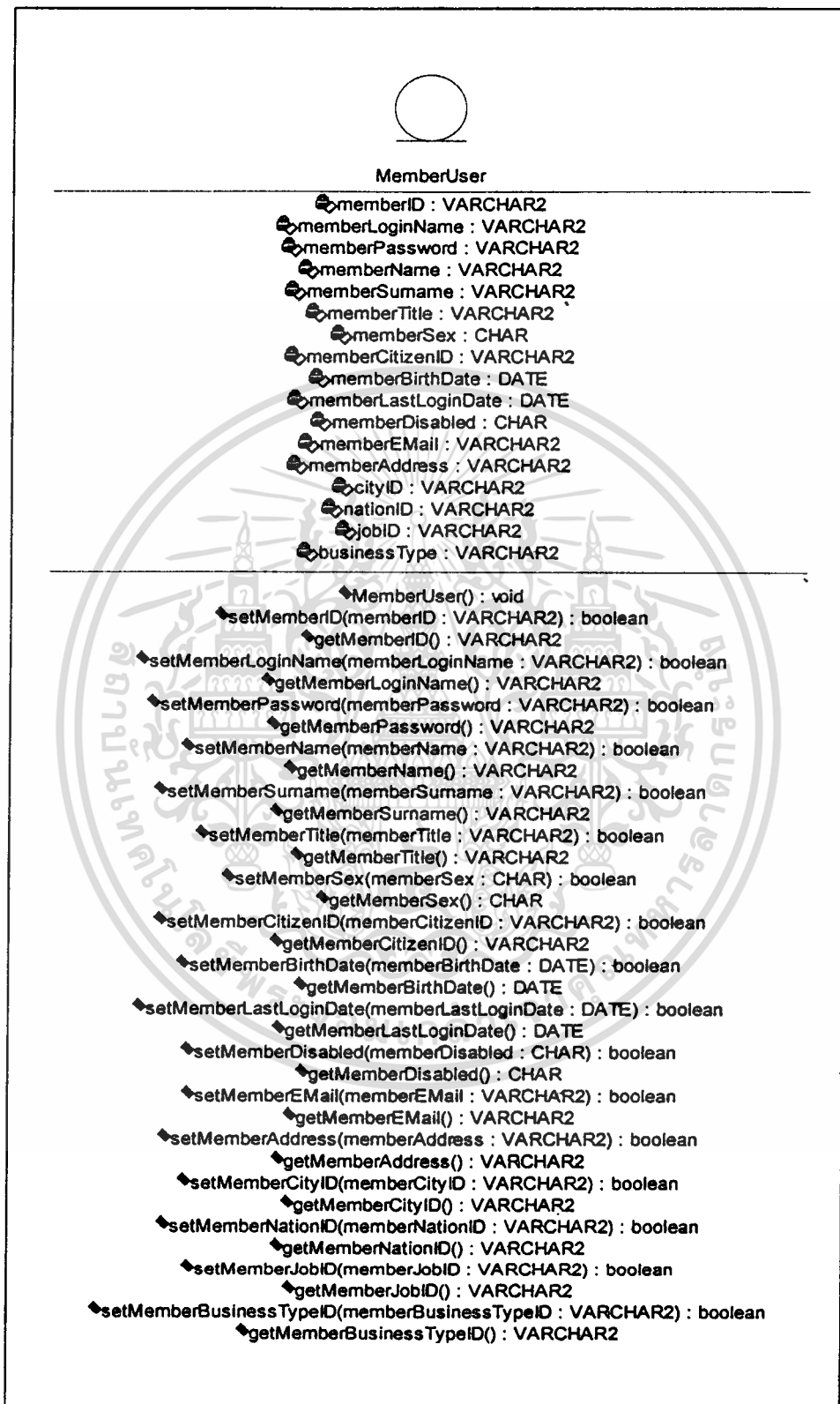


รูปที่ 6.4 Class Diagram ภายใน Database Package

6.3.2 Class Diagram ภายใน Bean Package

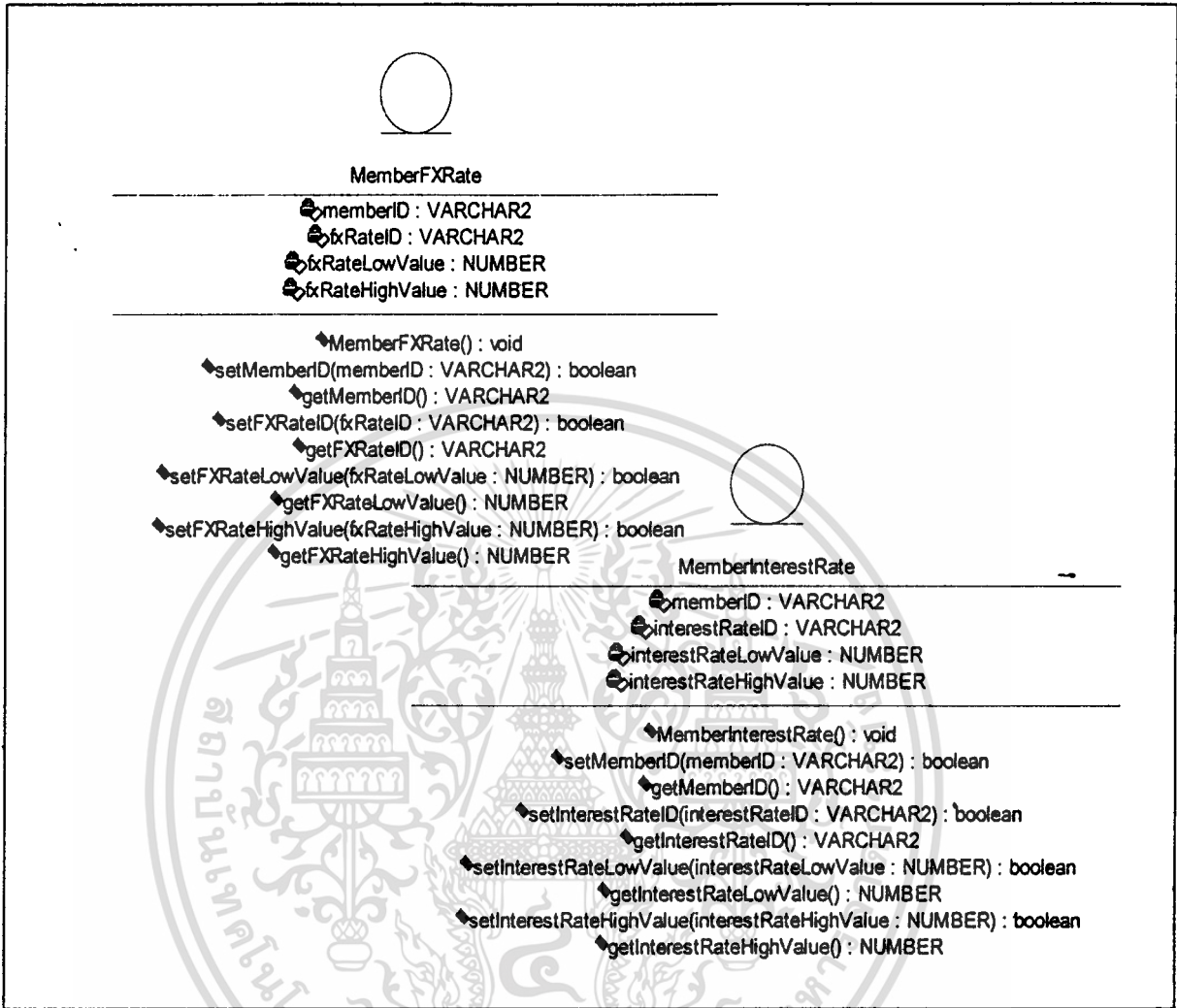


รูปที่ 6.5 Class Diagram ในกลุ่ม Member ภายใน Bean Package

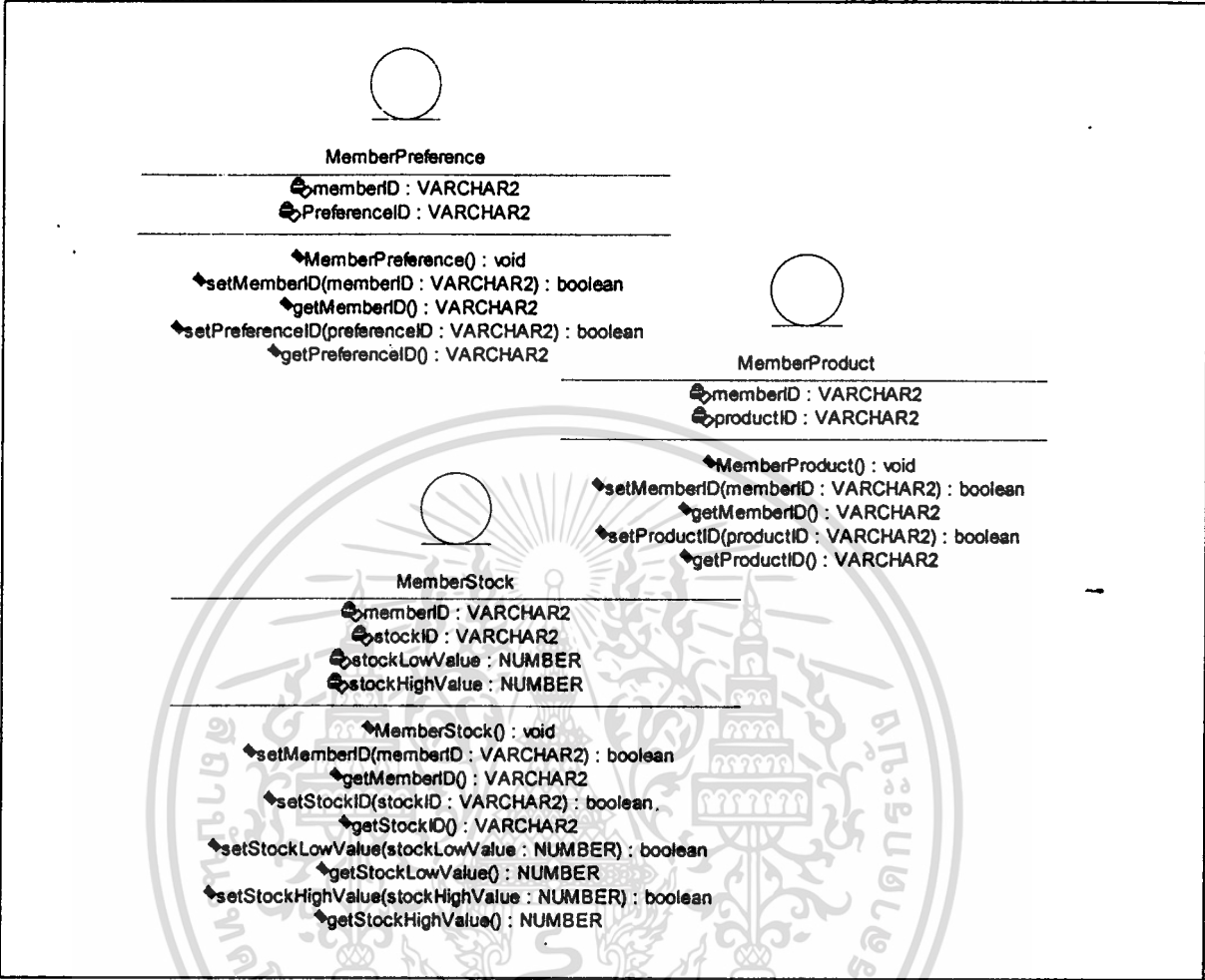


รูปที่ 6.6 Class MemberUser ภายใน Bean Package

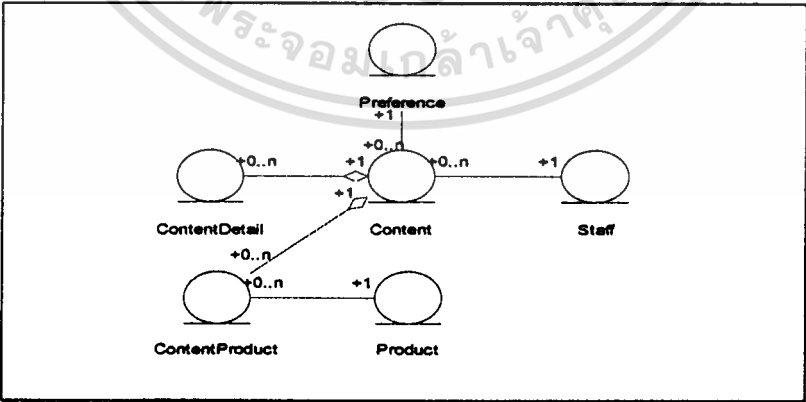
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



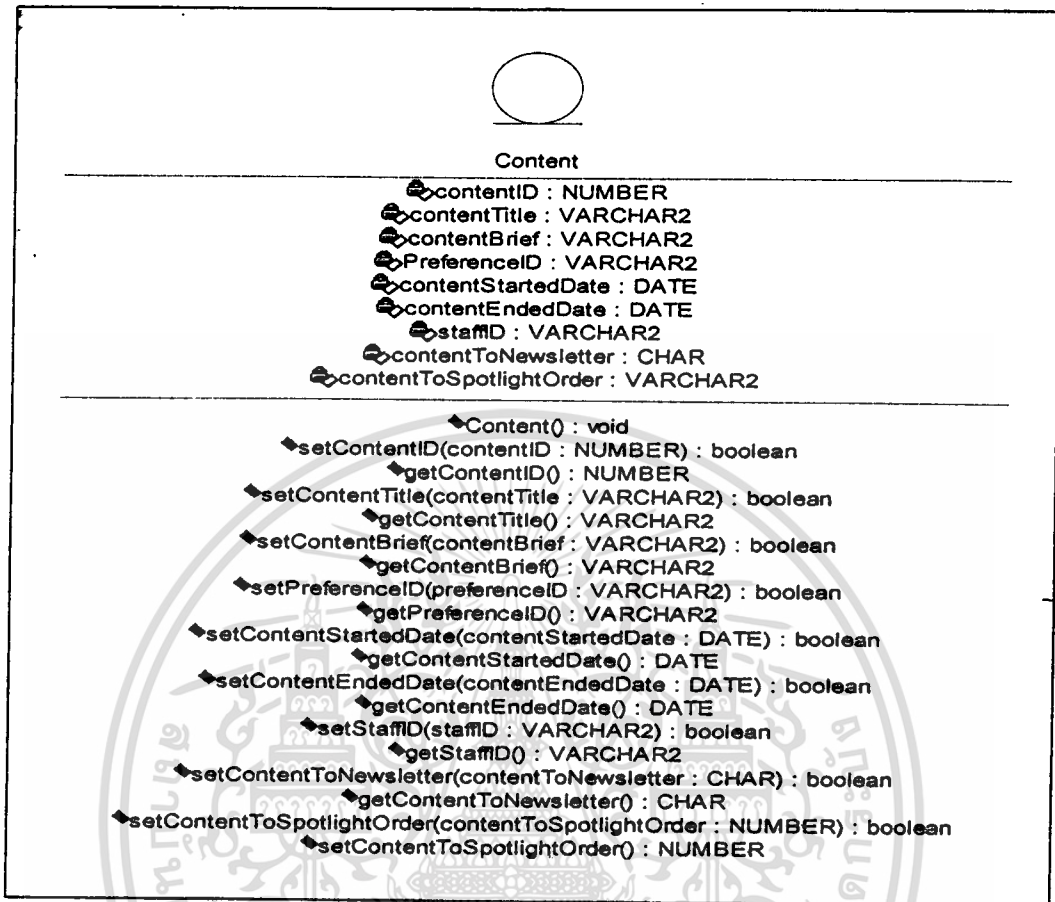
รูปที่ 6.7 Class MemberFXRate และ MemberInterestRate ภายใน Bean Package



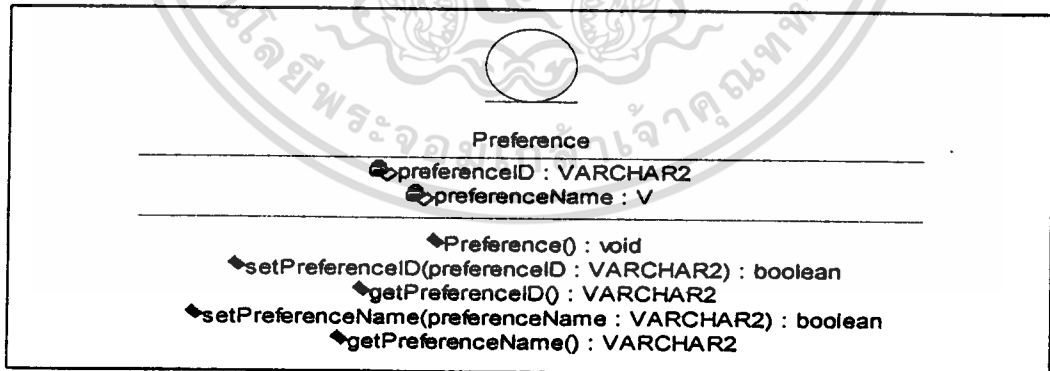
รูปที่ 6.8 Class MemberPreference, MemberStock และ MemberProduct ภายใน Bean Package



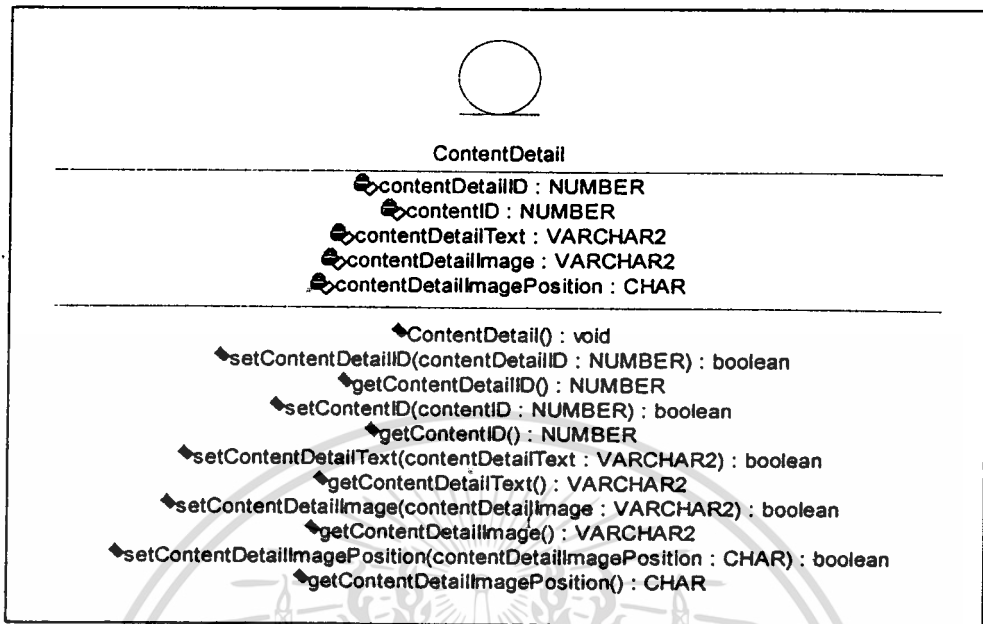
รูปที่ 6.9 Class Diagram ในกลุ่ม Preference ภายใน Bean Package



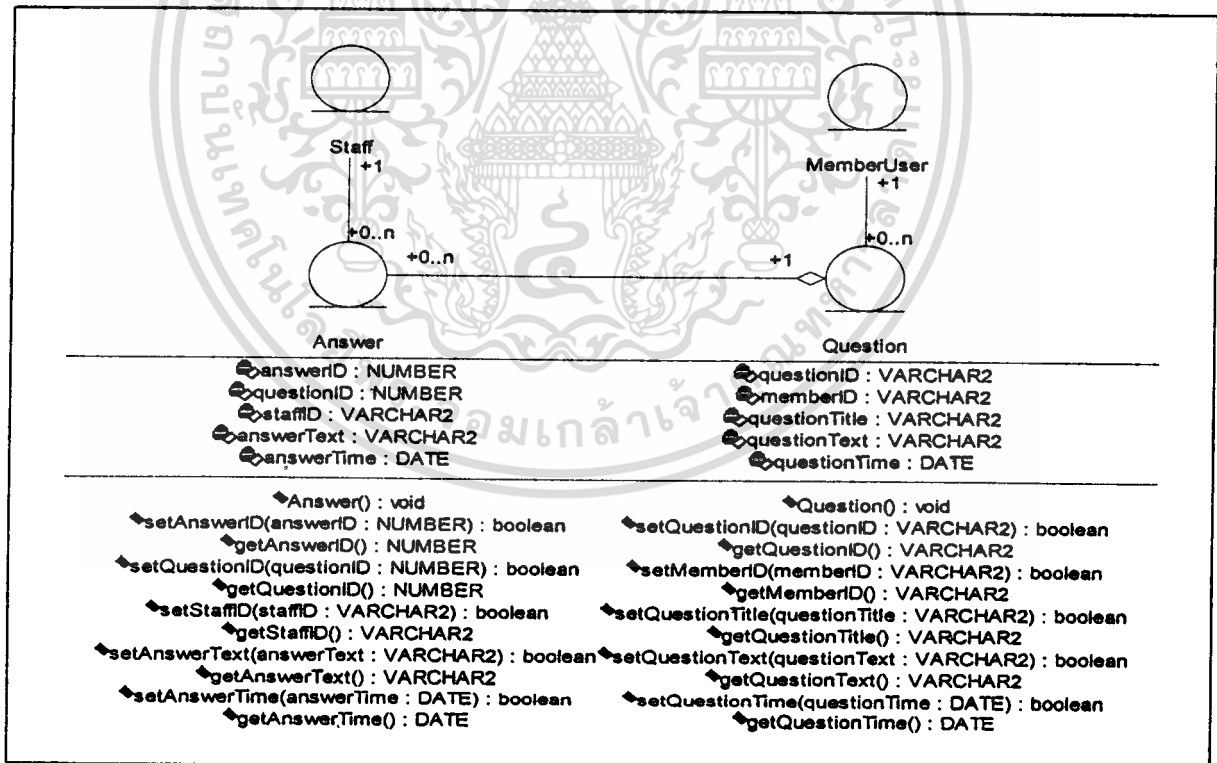
รูปที่ 6.10 Class Content ภายใน Bean Package



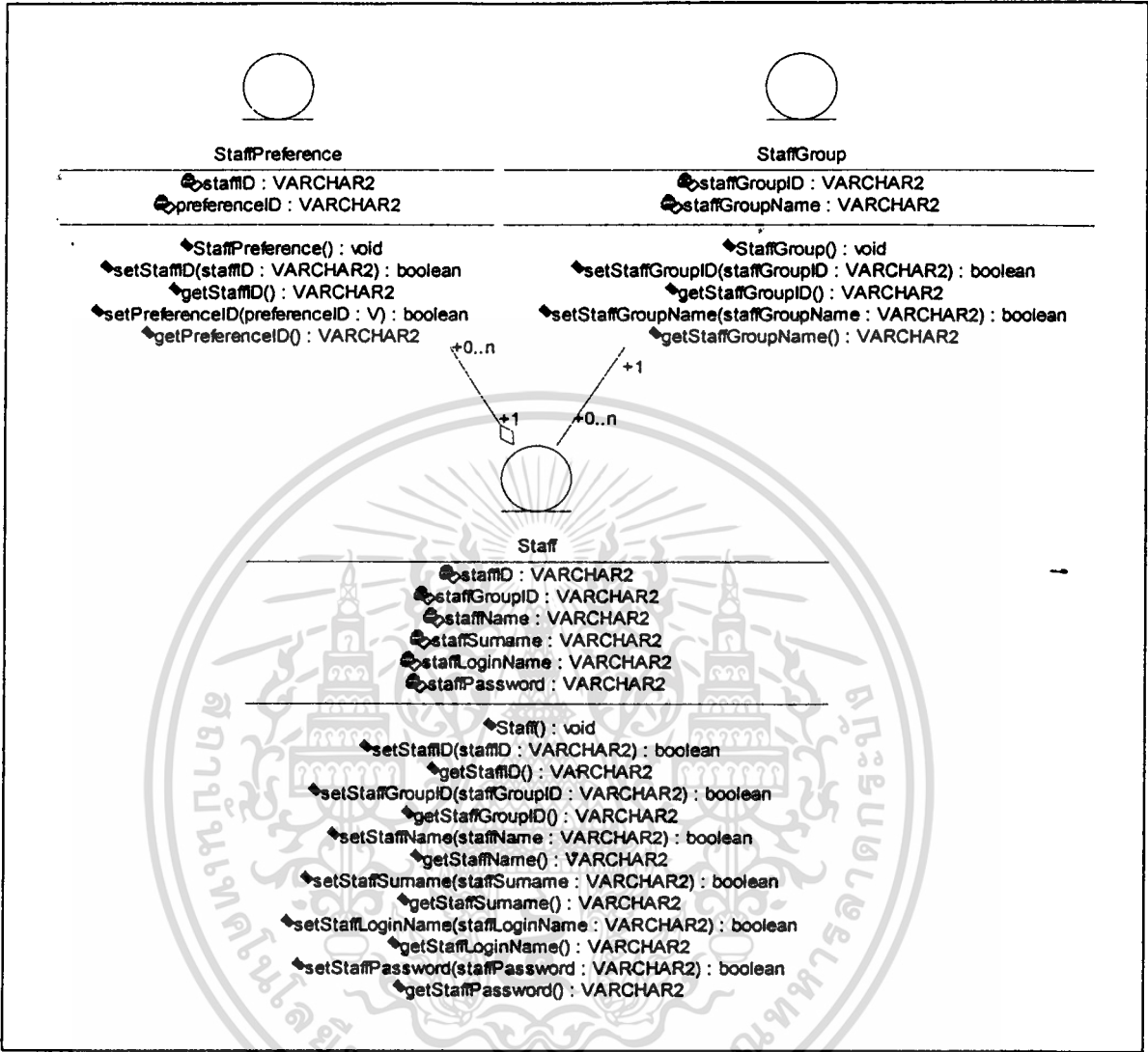
รูปที่ 6.11 Class Preference ภายใน Bean Package



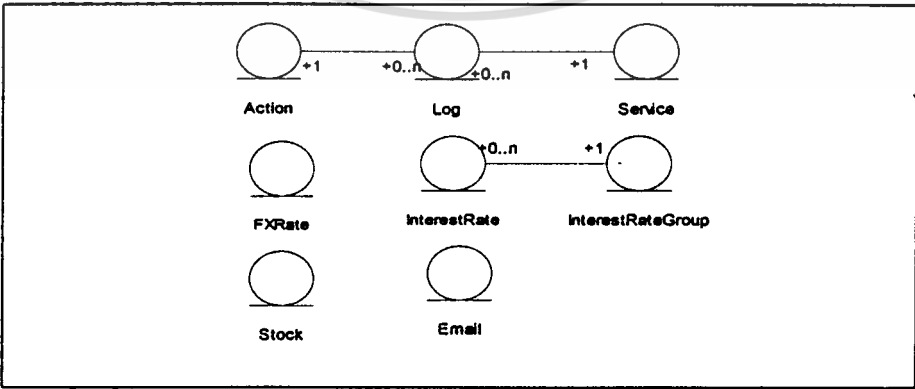
รูปที่ 6.12 Class ContentProduct ภายใน Bean Package



รูปที่ 6.13 Class Diagram ในกลุ่ม QA ภายใน Bean Package

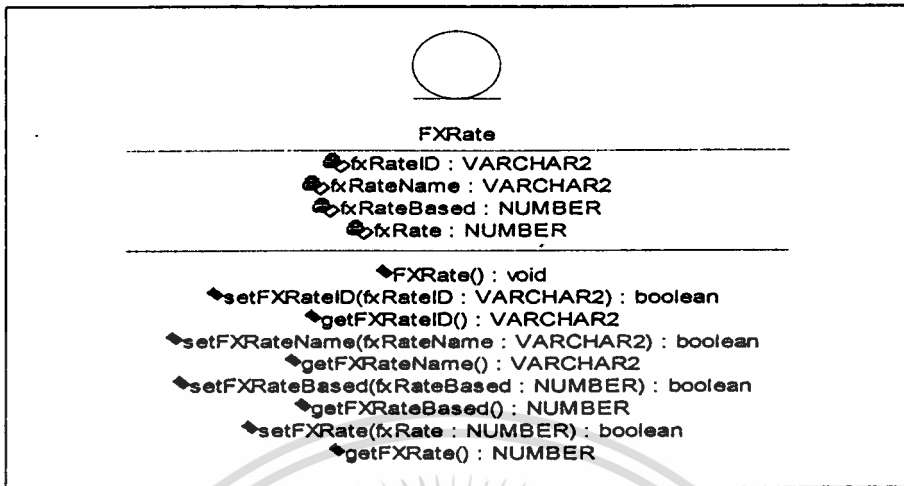


รูปที่ 6.14 Class Diagram ในกลุ่ม Staff ภายใน Bean Package

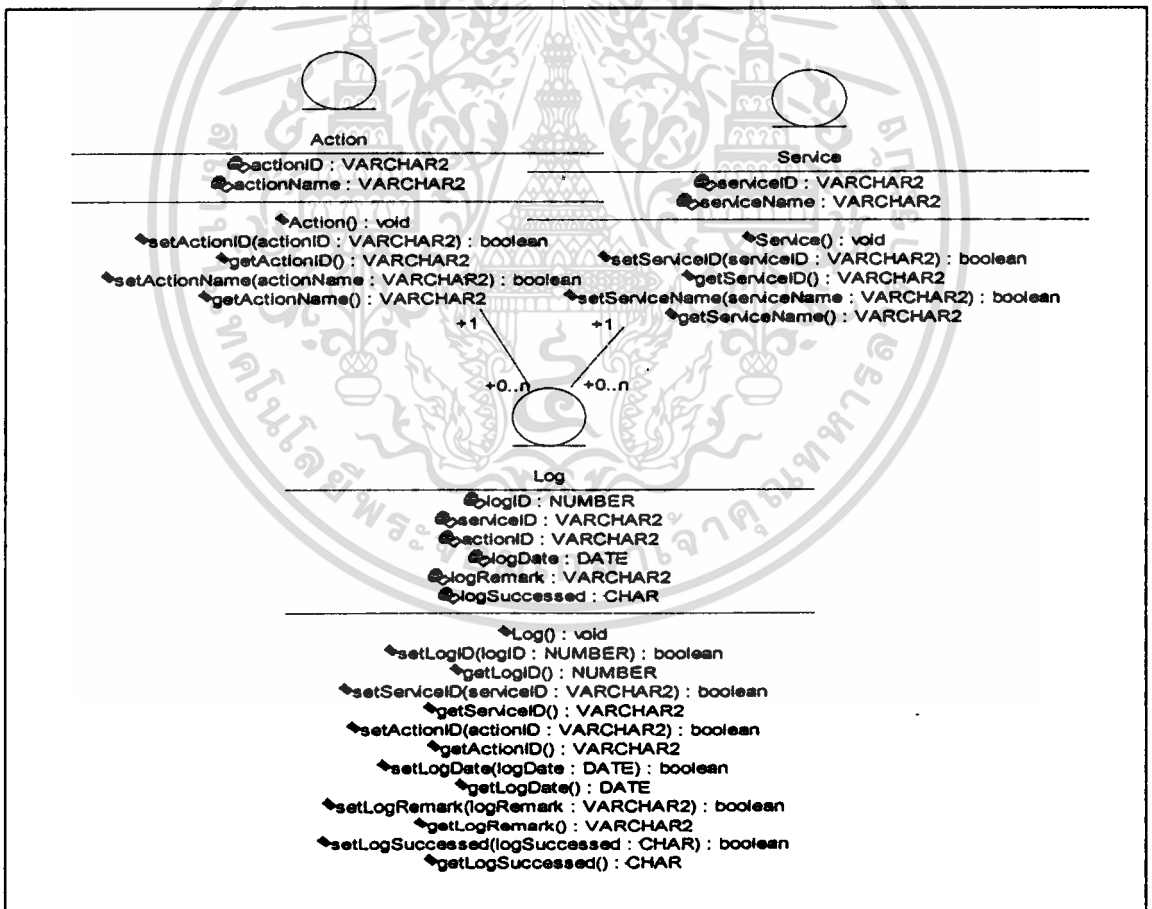


รูปที่ 6.15 Class Diagram ในกลุ่ม Email Alert and Log ภายใน Bean Package

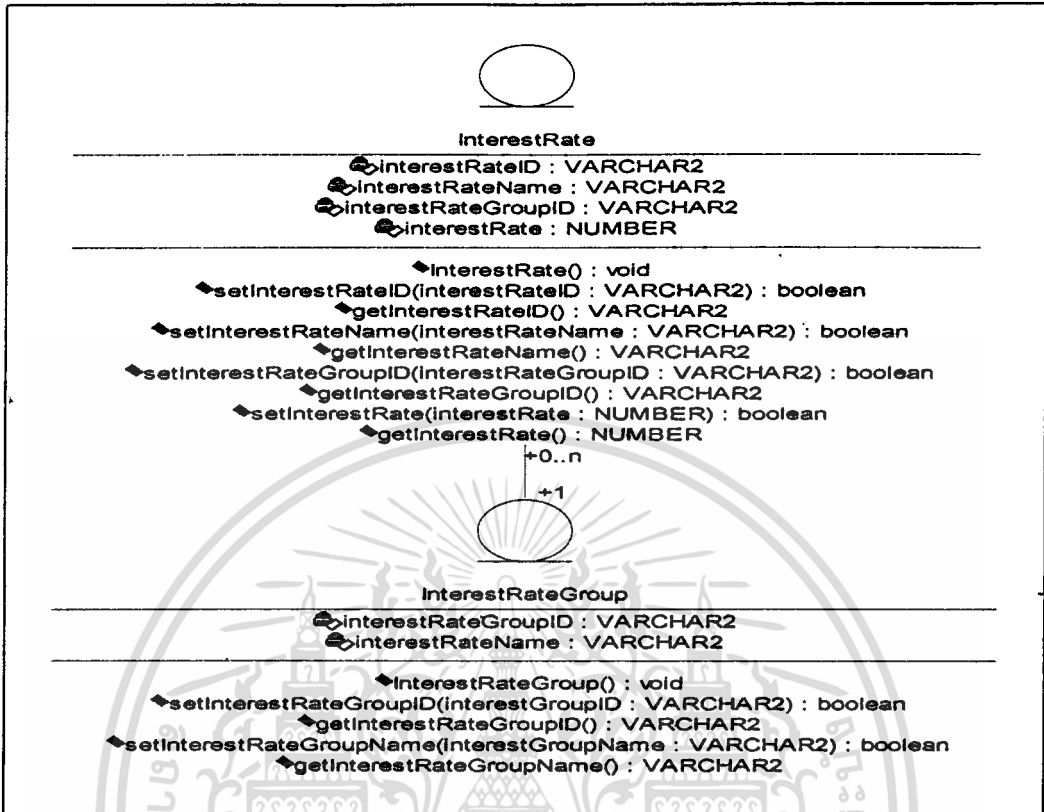
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



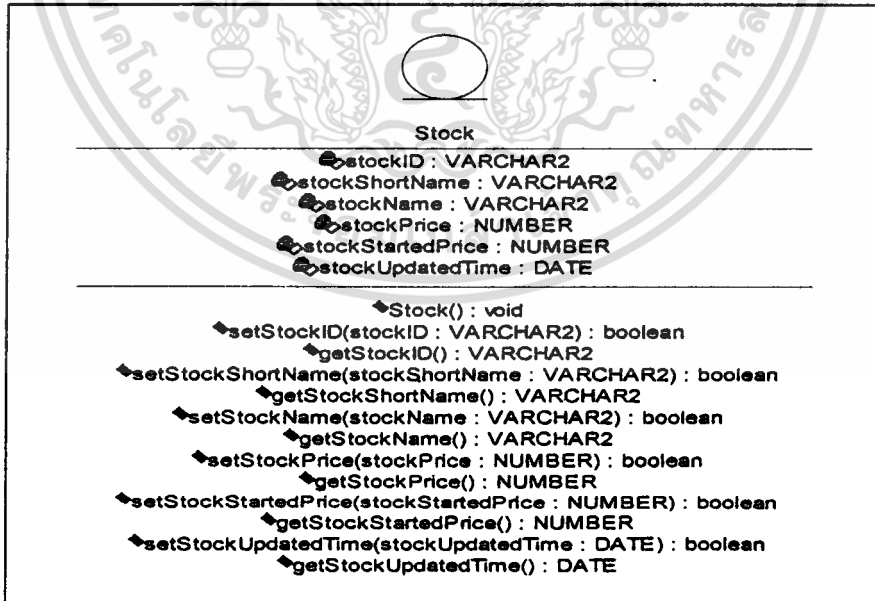
รูปที่ 6.16 Class FX Rate ภายใน Bean Package



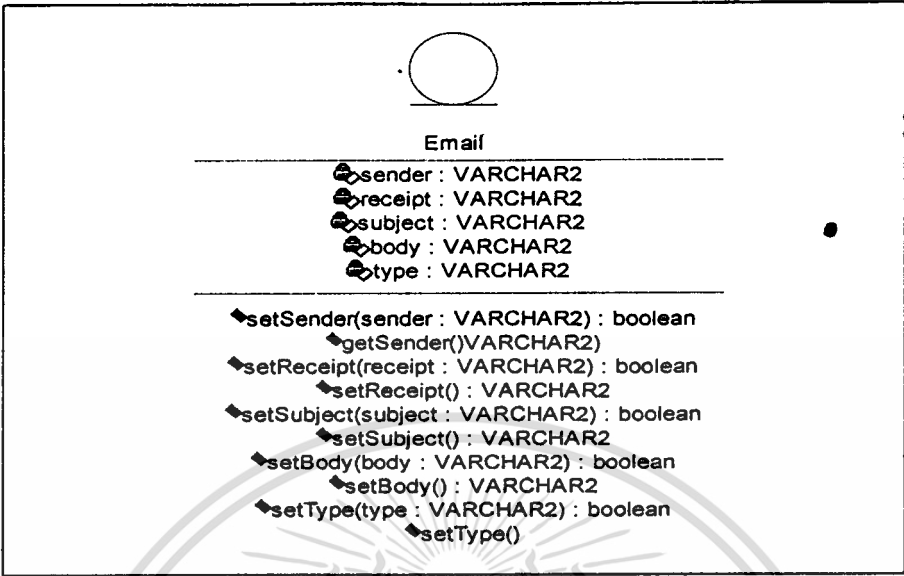
รูปที่ 6.17 Class Action, Log และ Service ภายใน Bean Package



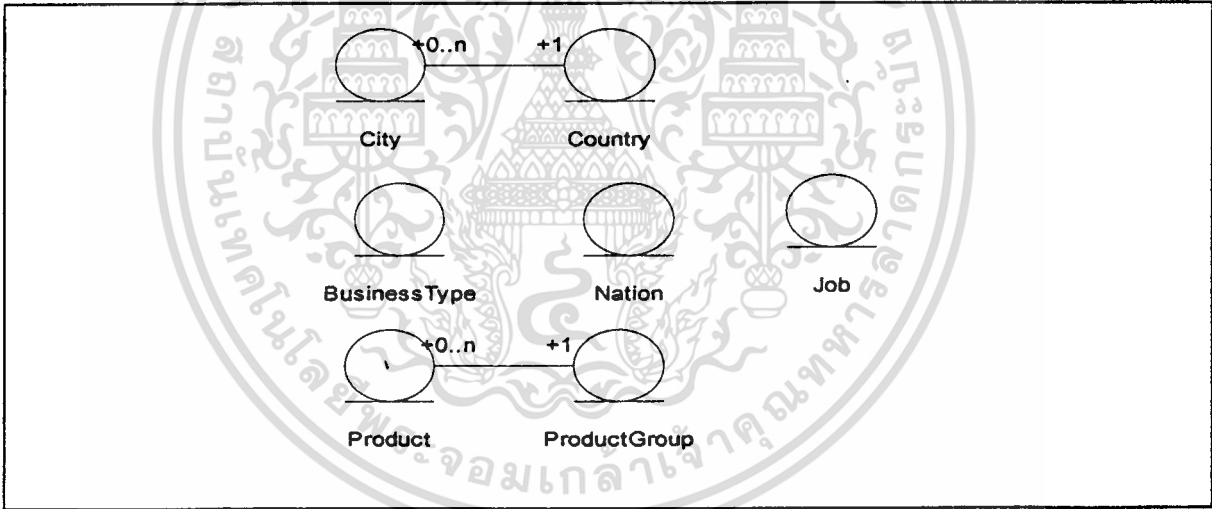
รูปที่ 6.18 Class Interest Rate และ Interest Rate Group ภายใน Bean Package



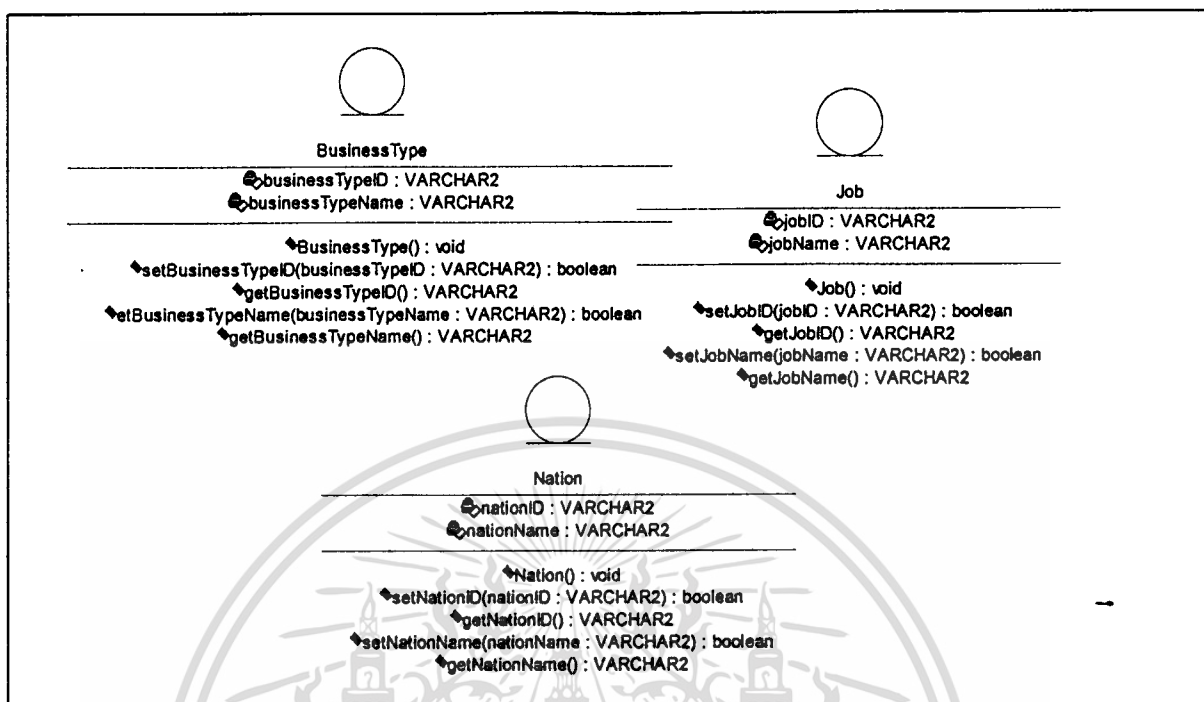
รูปที่ 6.19 Class Stock ภายใน Bean Package



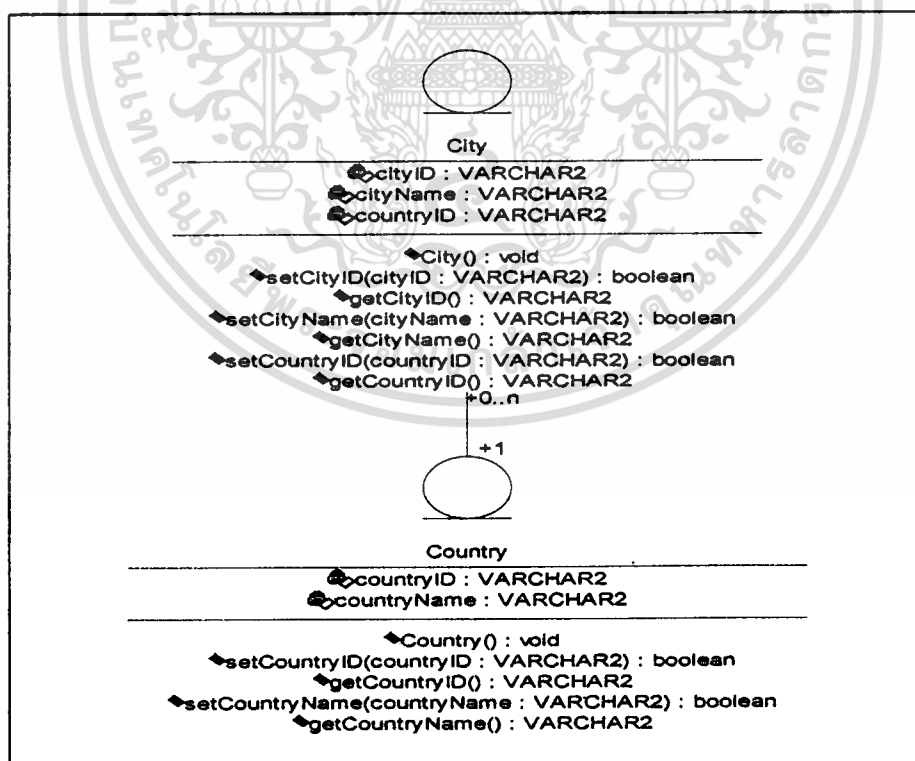
รูปที่ 6.20 Class Email ภายใน Bean Package



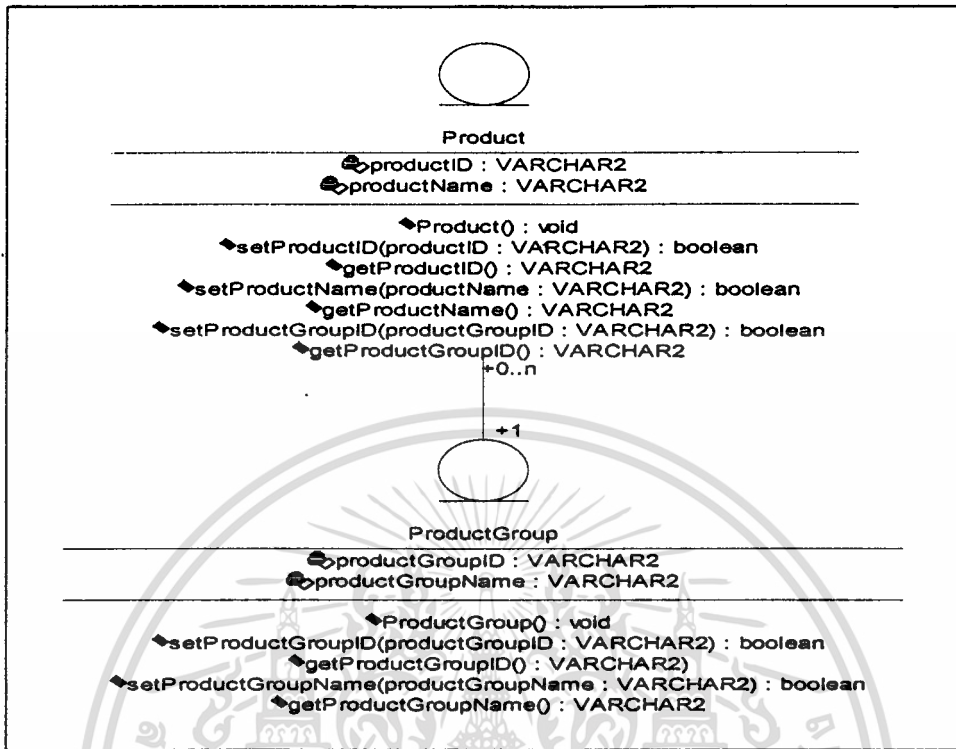
รูปที่ 6.21 Class Diagram ในกลุ่ม General ภายใน Bean Package



รูปที่ 6.22 Class Business Type, Job และ Nation ภายใน Bean Package

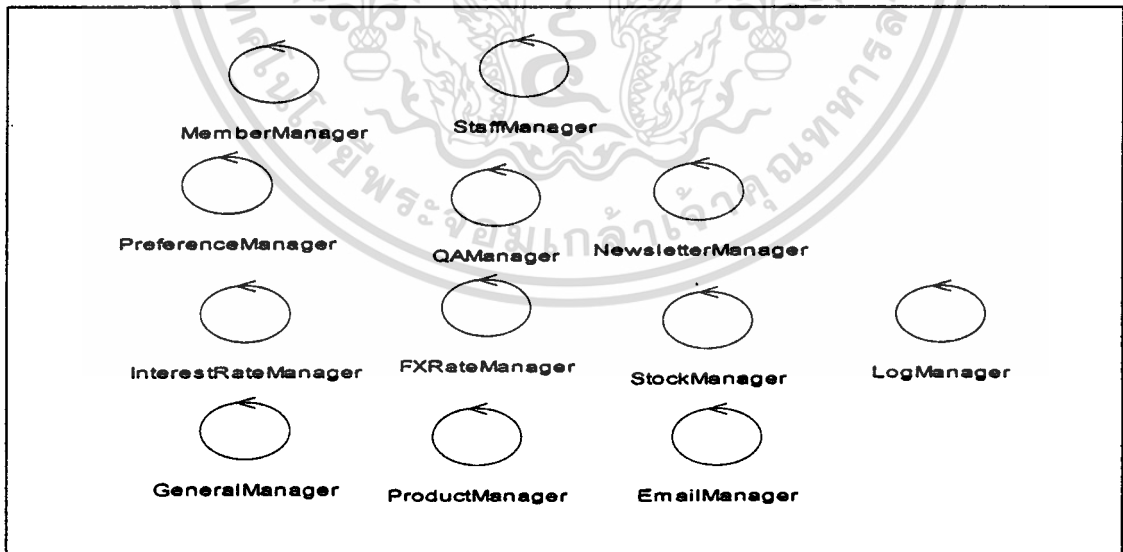


รูปที่ 6.23 Class City และ Country ภายใน Bean Package

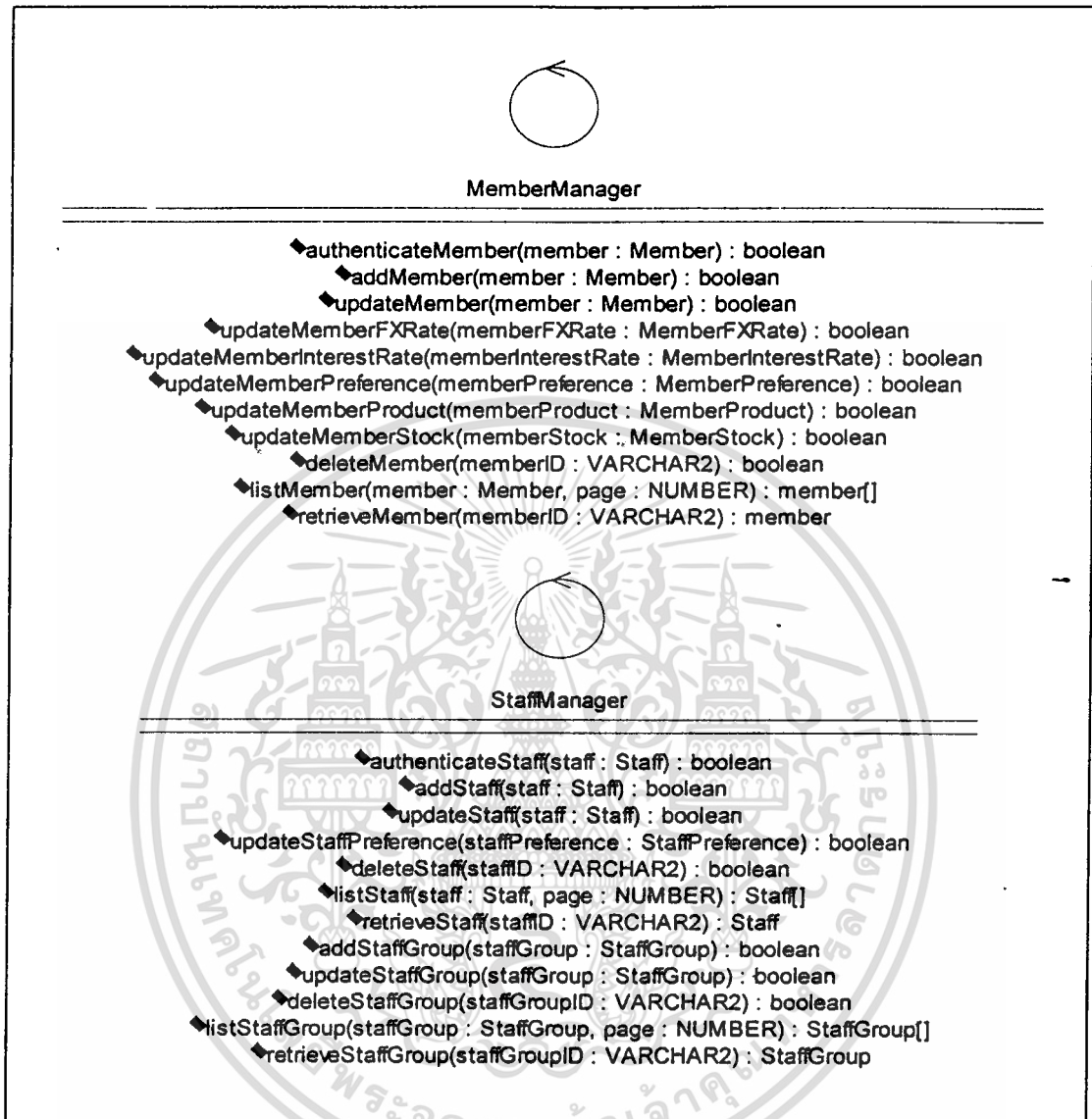


รูปที่ 6.24 Class Product และ ProductGroup ภายใน Bean Package

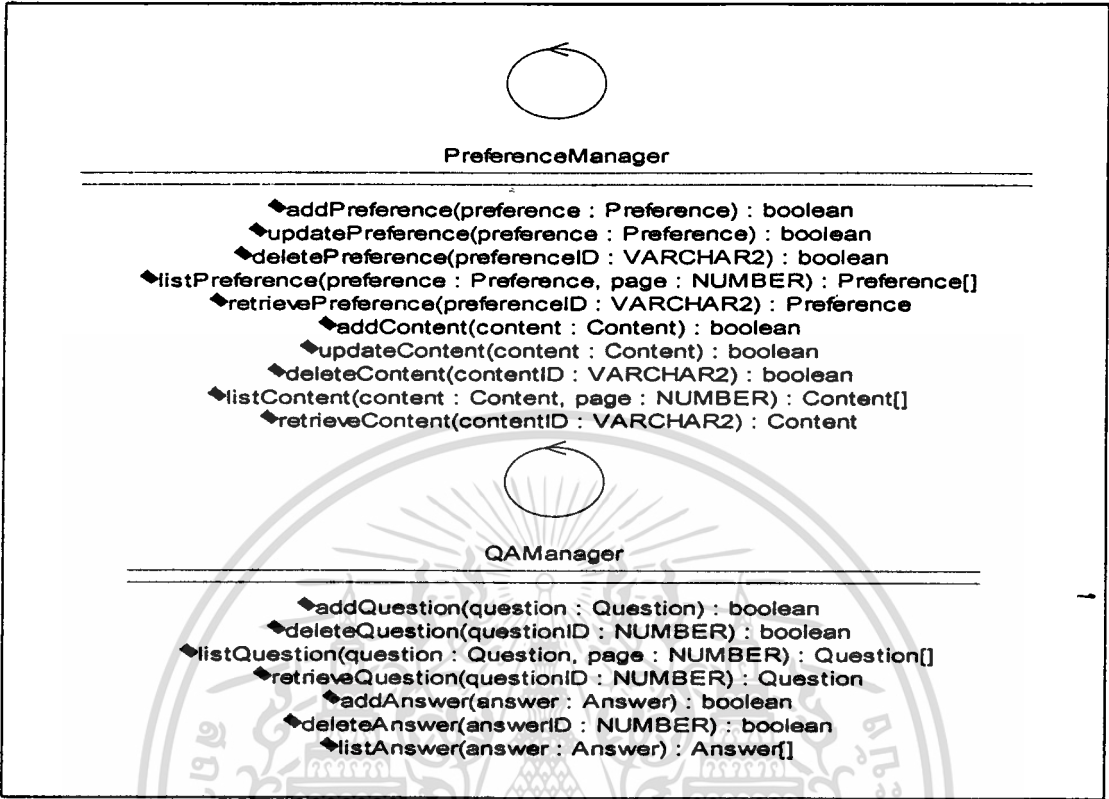
6.3.3 Class Diagram ภายใน Manager Package



รูปที่ 6.25 Class Diagram ภายใน Manager Package



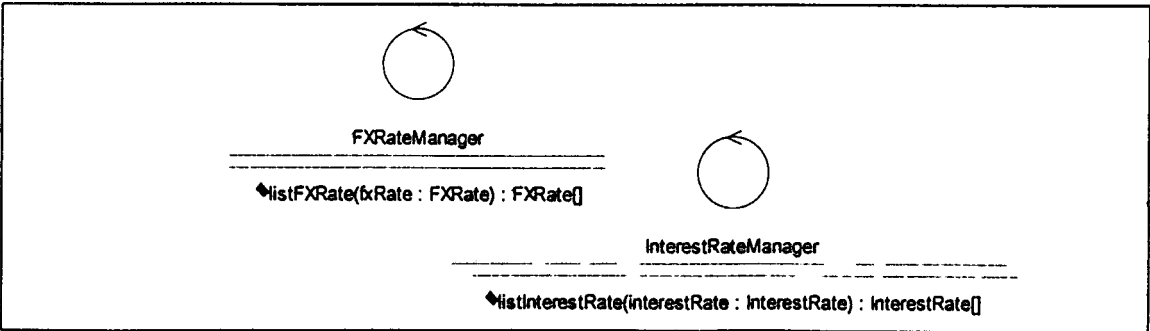
รูปที่ 6.26 Class MemberManager และ StaffManager ภายใน Manager Package



รูปที่ 6.27 Class PreferenceManager และ QAManager ภายใน Manager Package

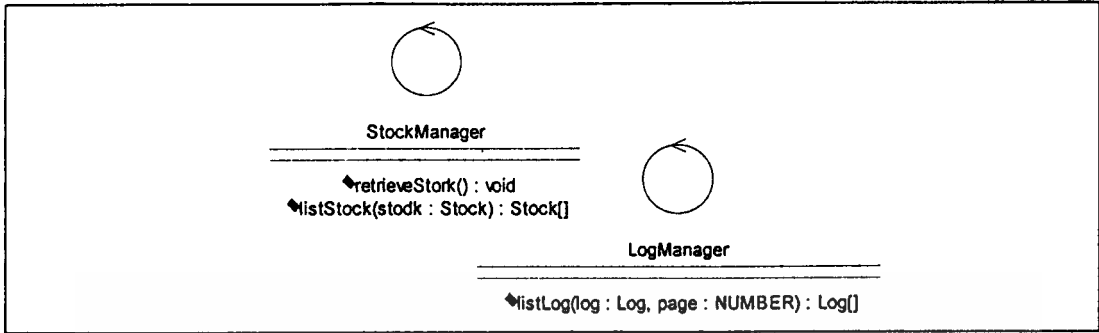


รูปที่ 6.28 Class NewsletterManager และ EmailManager ภายใน Manager Package



รูปที่ 6.29 Class FXManager และ InterestRateManager ภายใน Manager Package

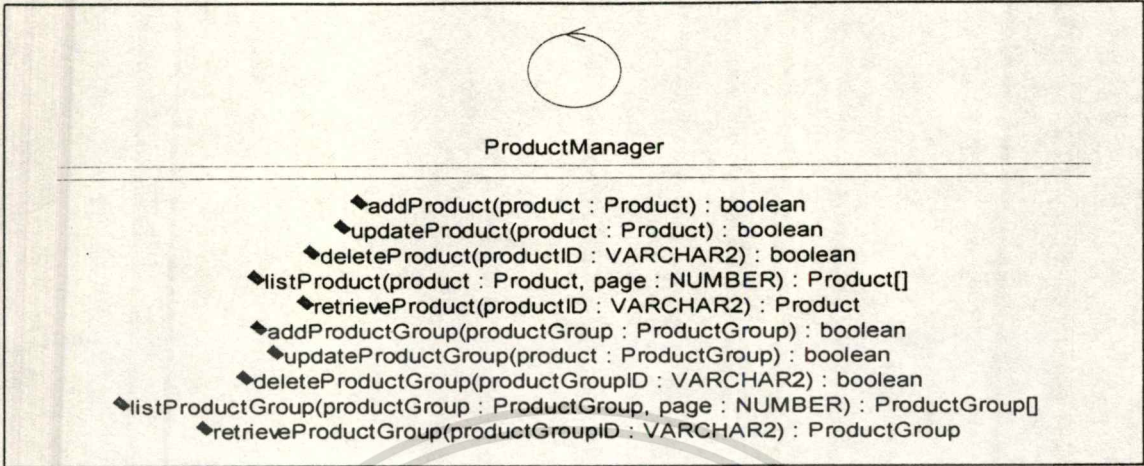
เอกสารนี้เป็นเอกสารตัวอย่าง ไม่สามารถนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.30 Class StockManager และ LogManager ภายใน Manager Package

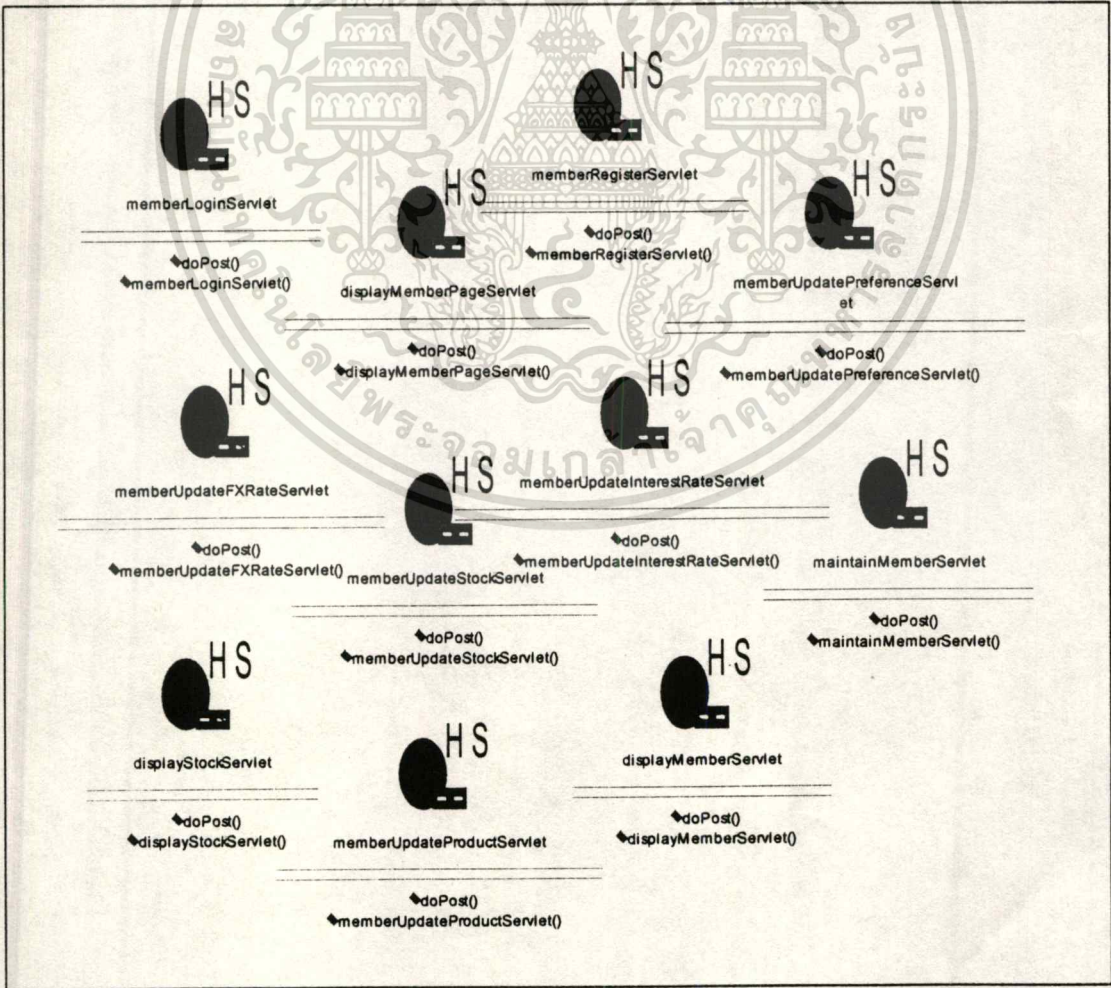


รูปที่ 6.31 Class GeneralManager ภายใน Manager Package



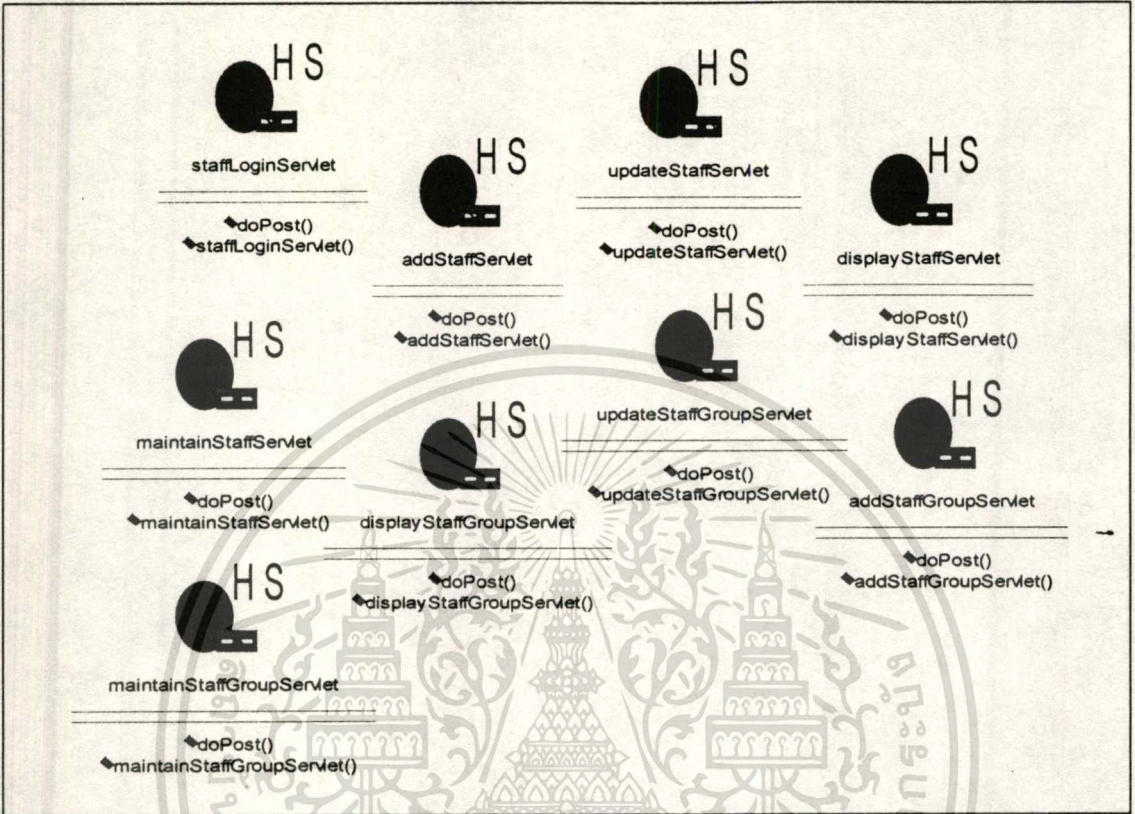
รูปที่ 6.32 Class ProductManager ภายใน Manager Package

6.3.4 Class Diagram ภายใน Servlet Package

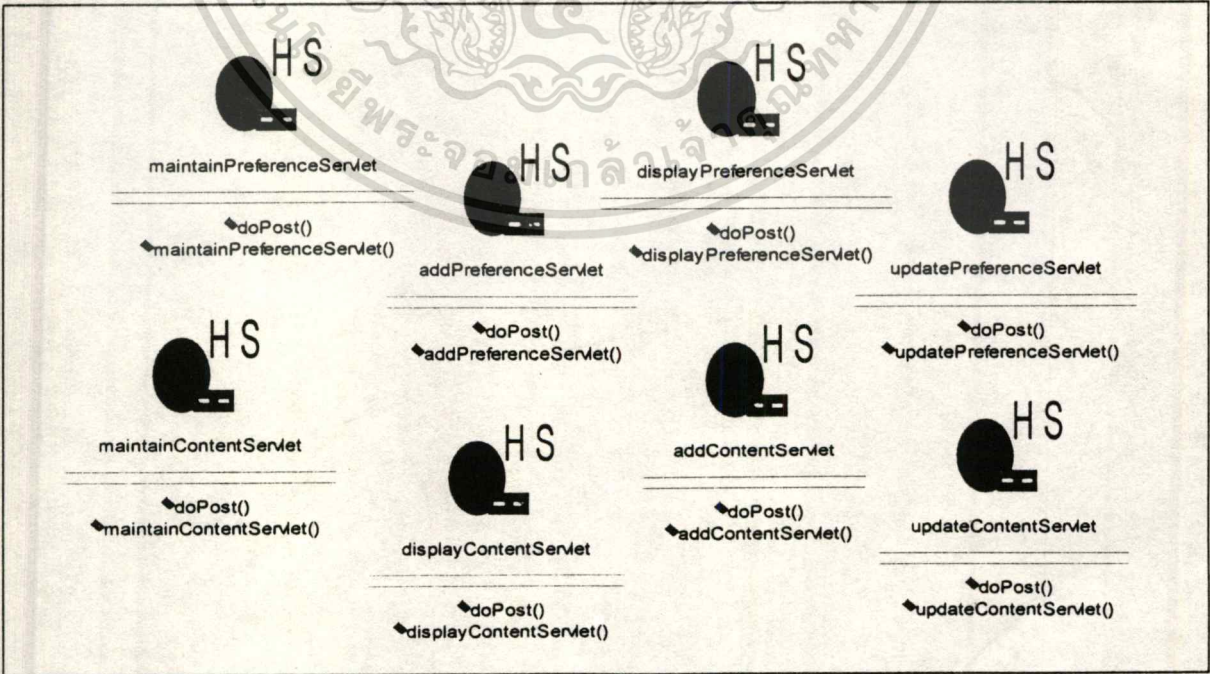


รูปที่ 6.33 Class Diagram ในกลุ่ม Member ภายใน Servlet Package

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

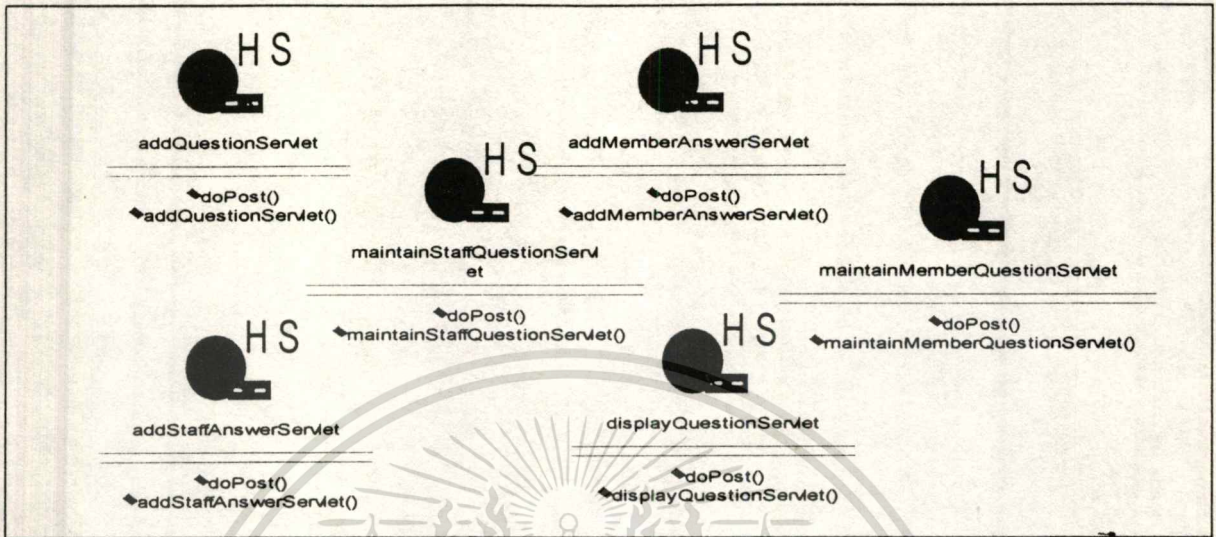


รูปที่ 6.34 Class Diagram ในกลุ่ม Staff ภายใน Servlet Package

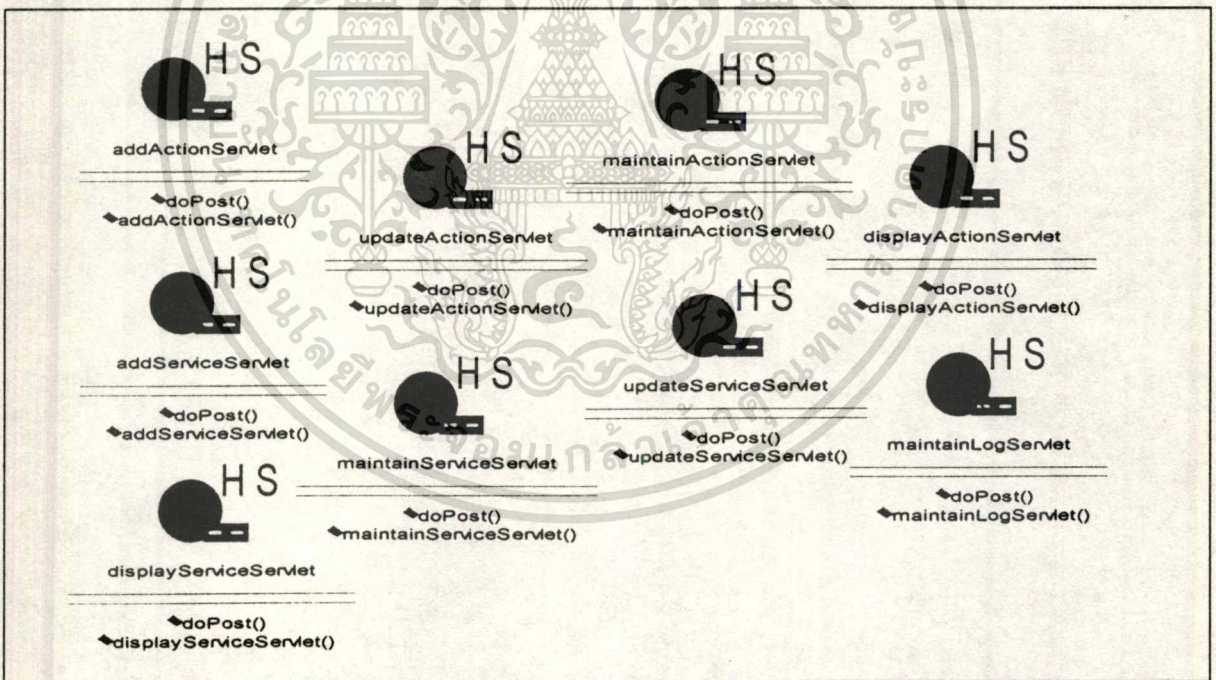


รูปที่ 6.35 Class Diagram ในกลุ่ม Preference ภายใน Servlet Package

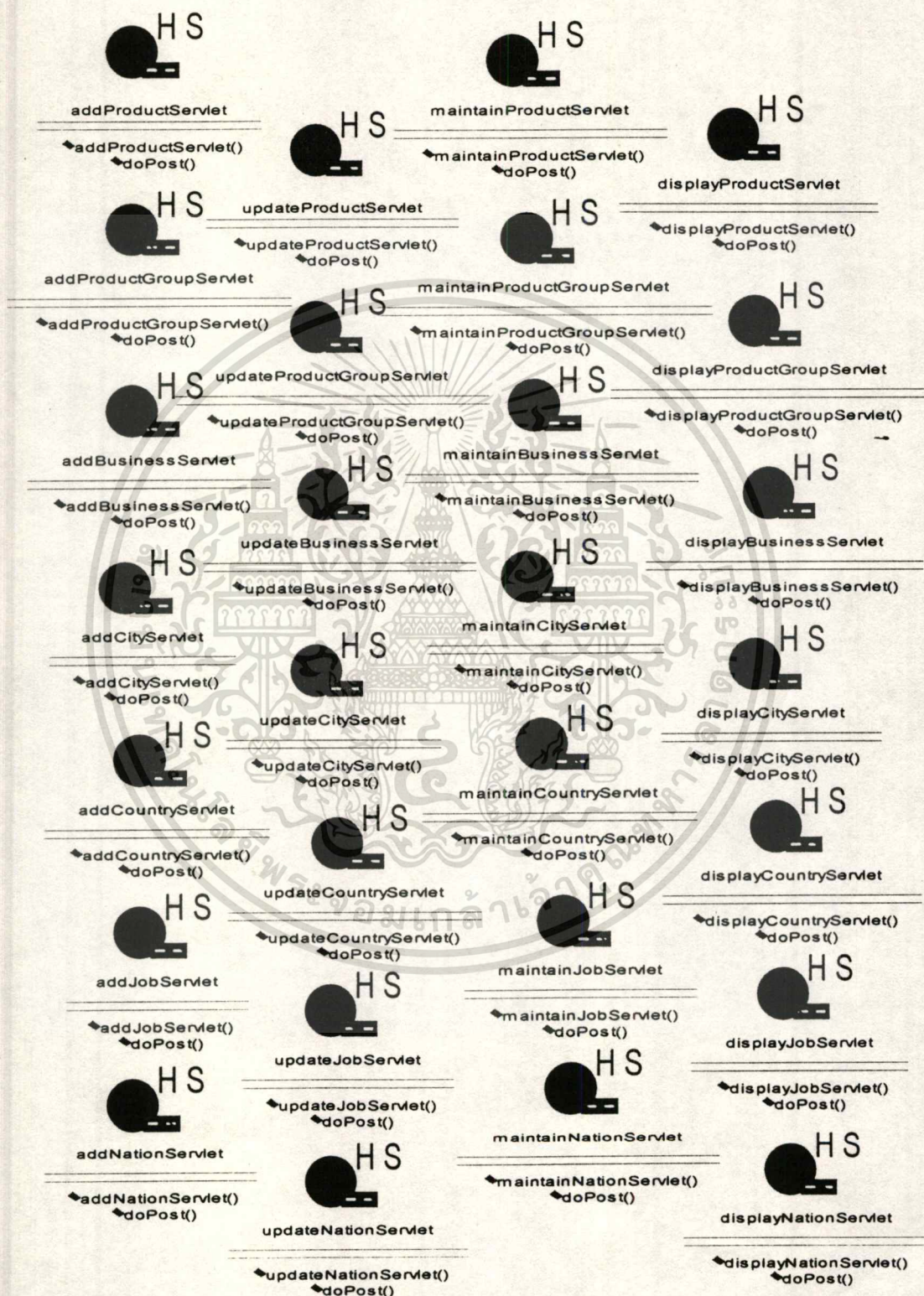
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.36 Class Diagram ในกลุ่ม QA ภายใน Servlet Package



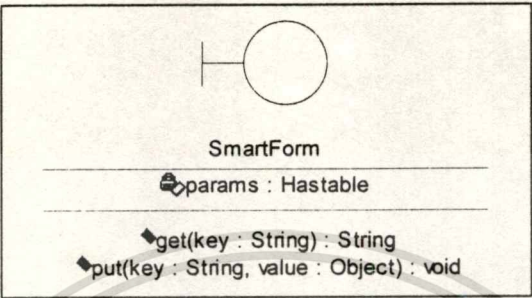
รูปที่ 6.37 Class Diagram ในกลุ่ม Alert Log ภายใน Servlet Package



เอกสารนี้เป็นเอกสารรูปที่ 6.38 Class Diagram ในกลุ่ม General ภายใน Servlet Package ซึ่งประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

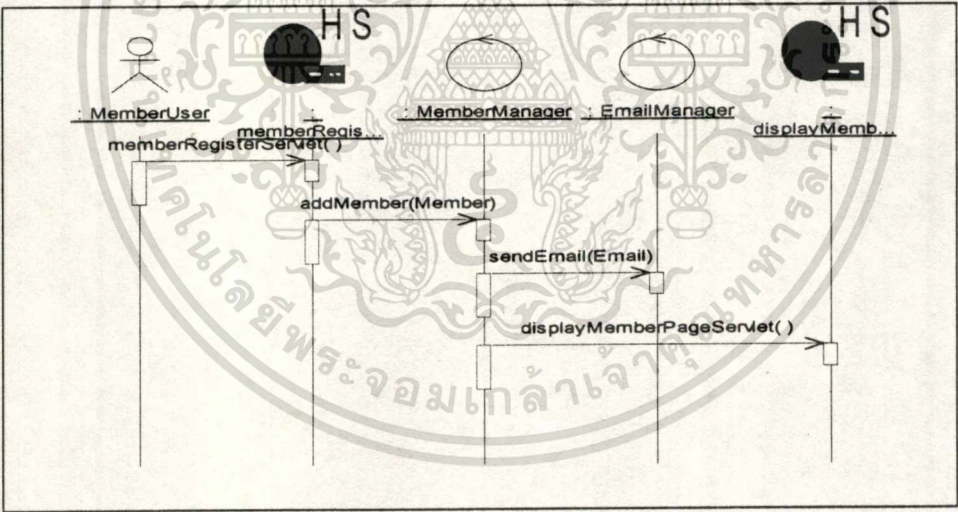
6.3.5 Class Diagram ภายใน Form Package



รูปที่ 6.39 Class Diagram ภายใน Form Package

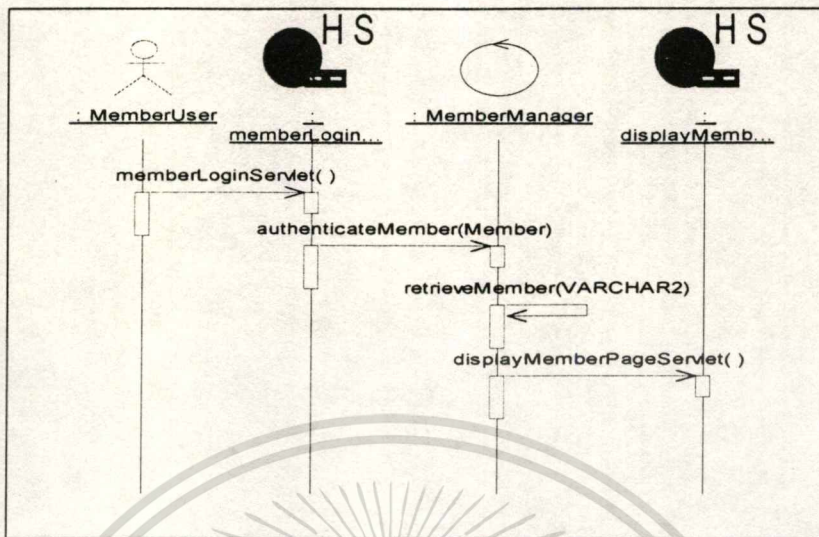
6.4 Sequence Diagram

1) ชื่อชุดเคส : Registration



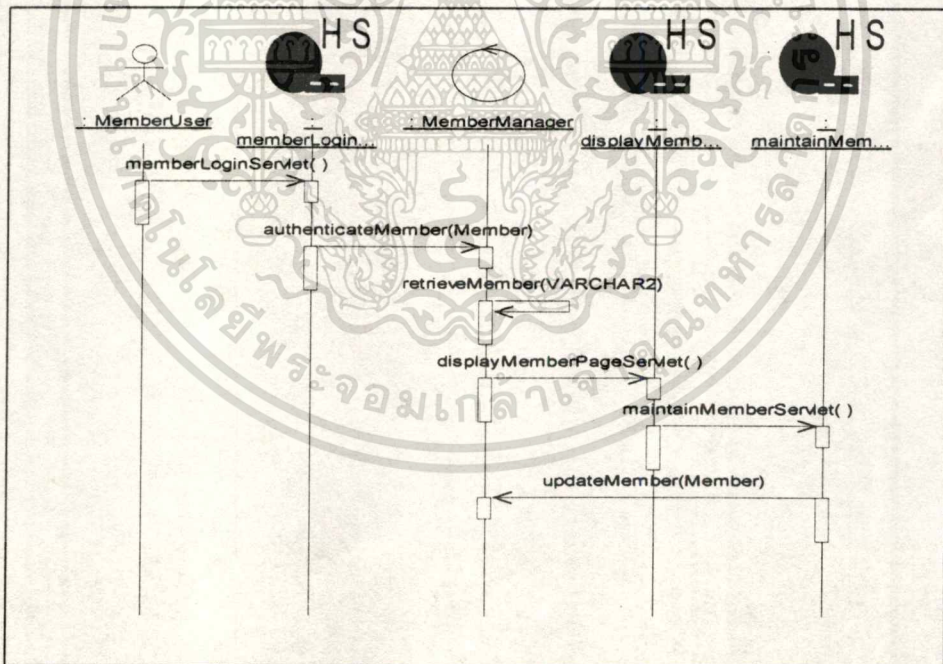
รูปที่ 6.40 Sequence Diagram: Registration

2) ชื่อชุดเคส : Authenticate Member



รูปที่ 6.41 Sequence Diagram: Authenticate Member

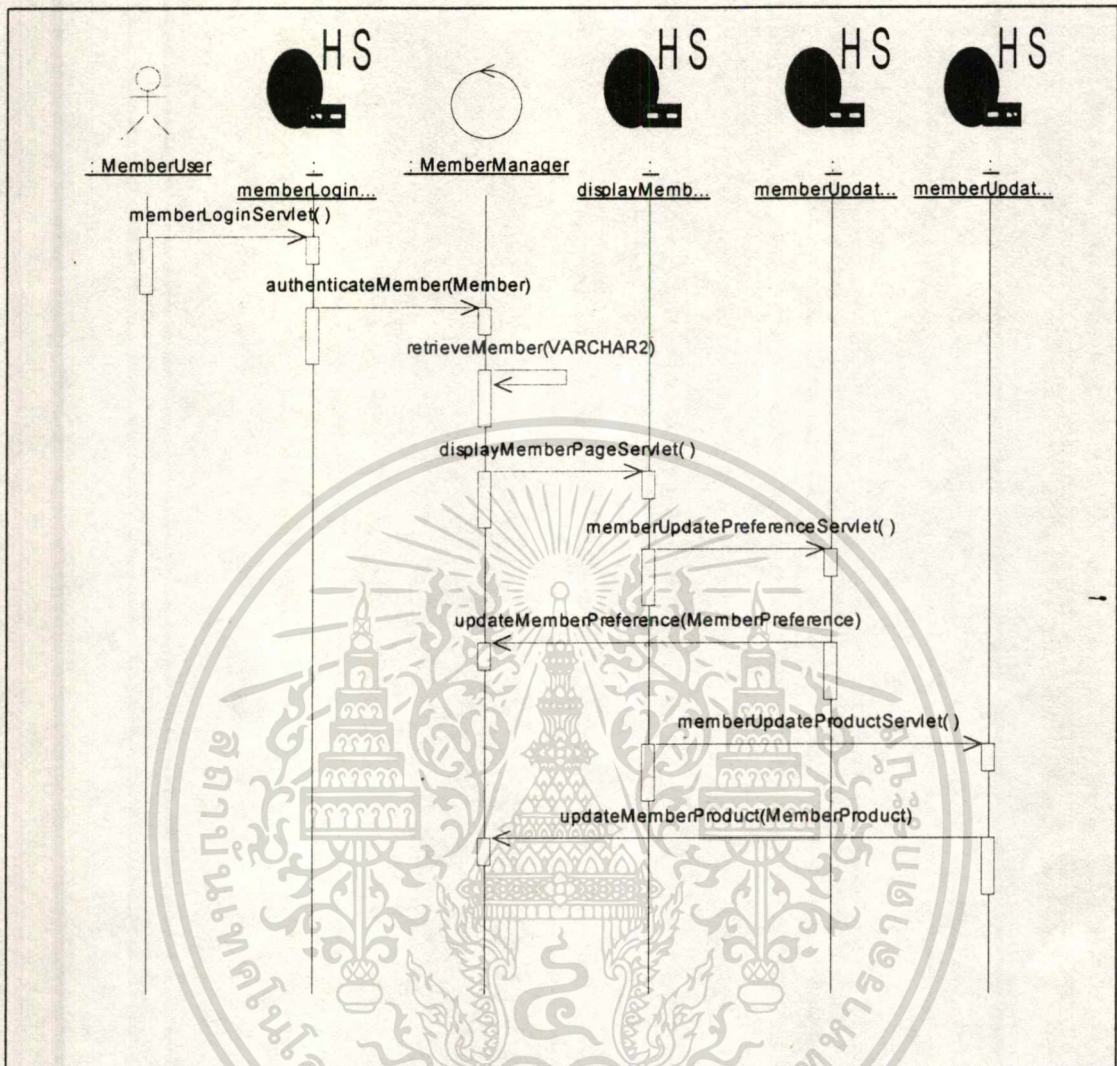
3) ข้อบกพร่อง : Update Member Information



รูปที่ 6.42 Sequence Diagram: Update Information

4) ข้อบกพร่อง : Update Member Preference

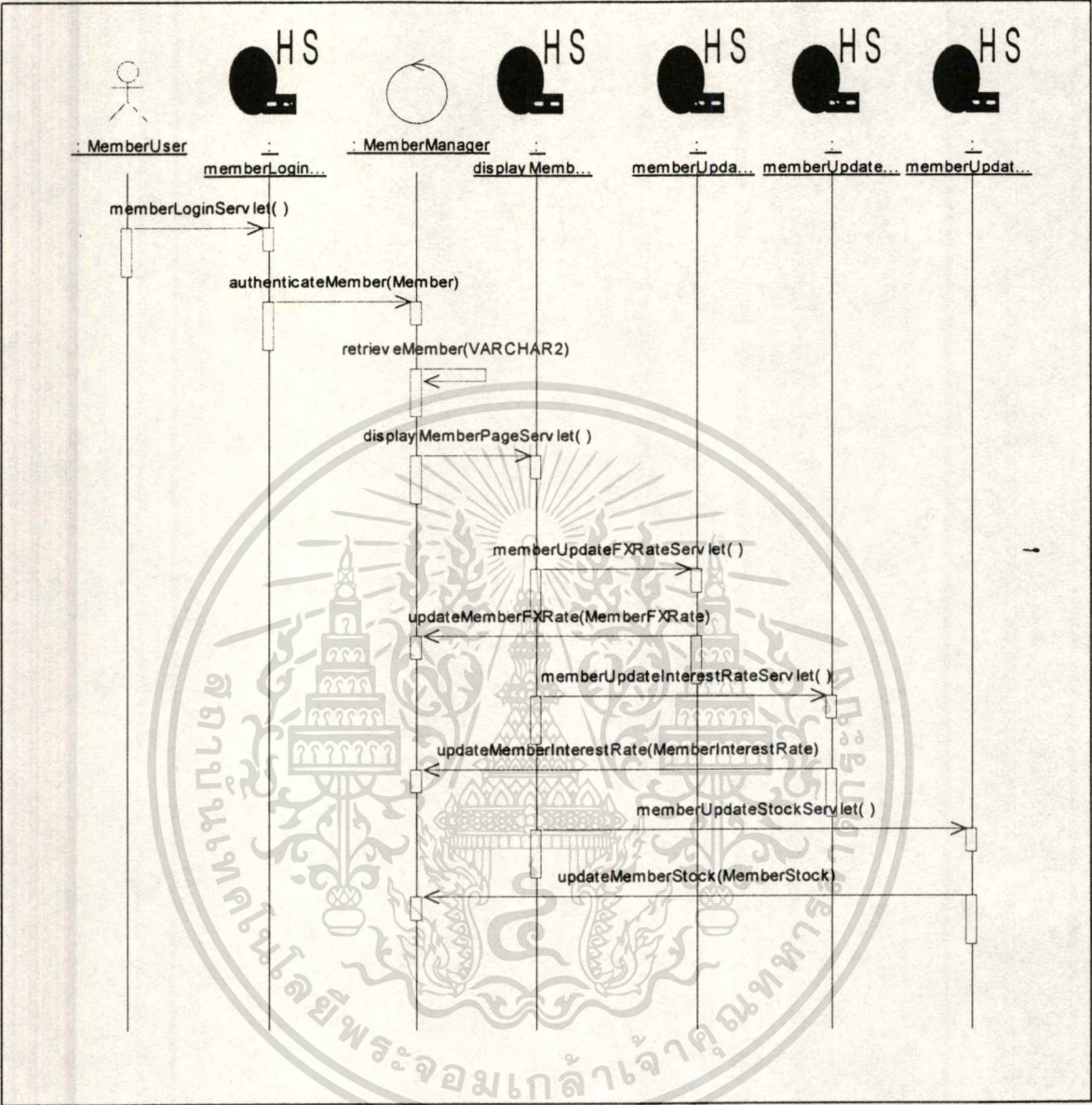
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.43 Sequence Diagram: Update Preference

5) ข้อยุติทดสอบ : Update Member Alert Criteria

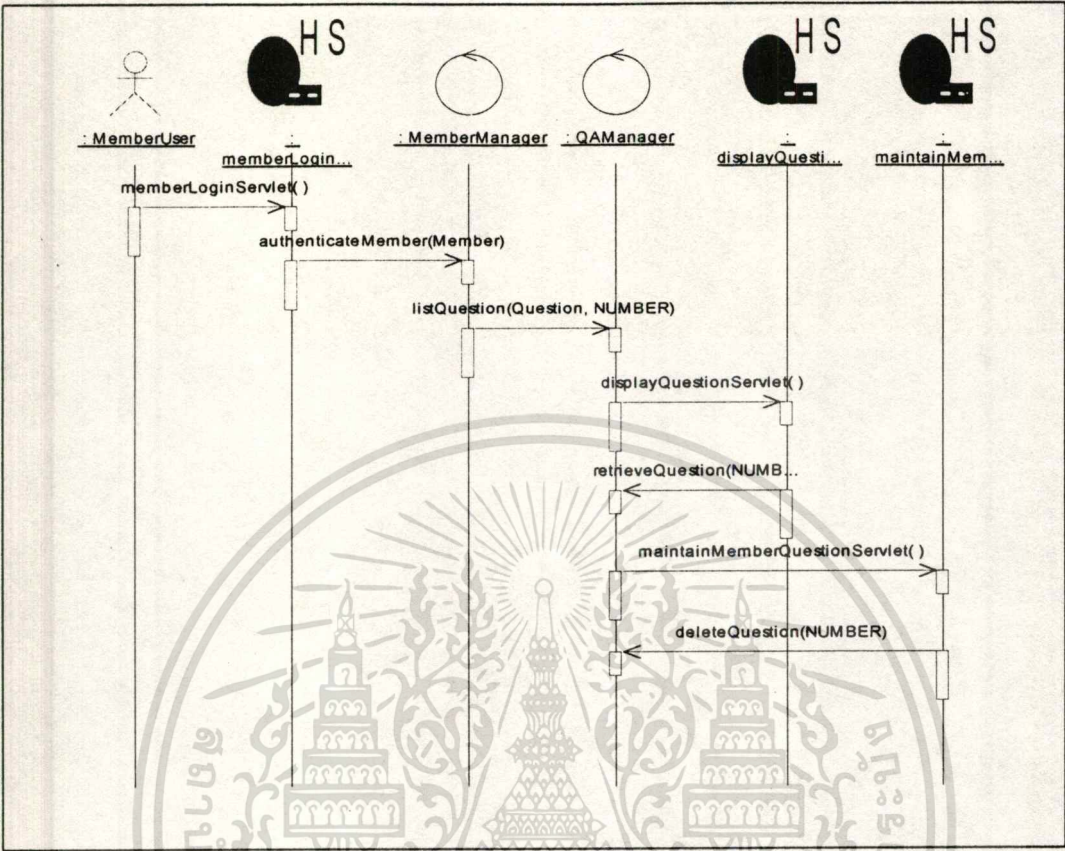
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.44 Sequence Diagram: Update Alert Criteria

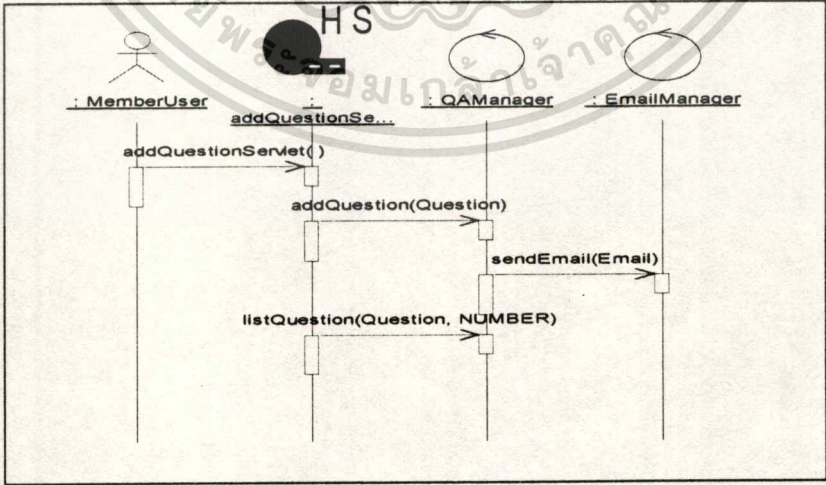
6) ชื่อชุดทดสอบ : Maintain Member Question

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.45 Sequence Diagram: Maintain Question

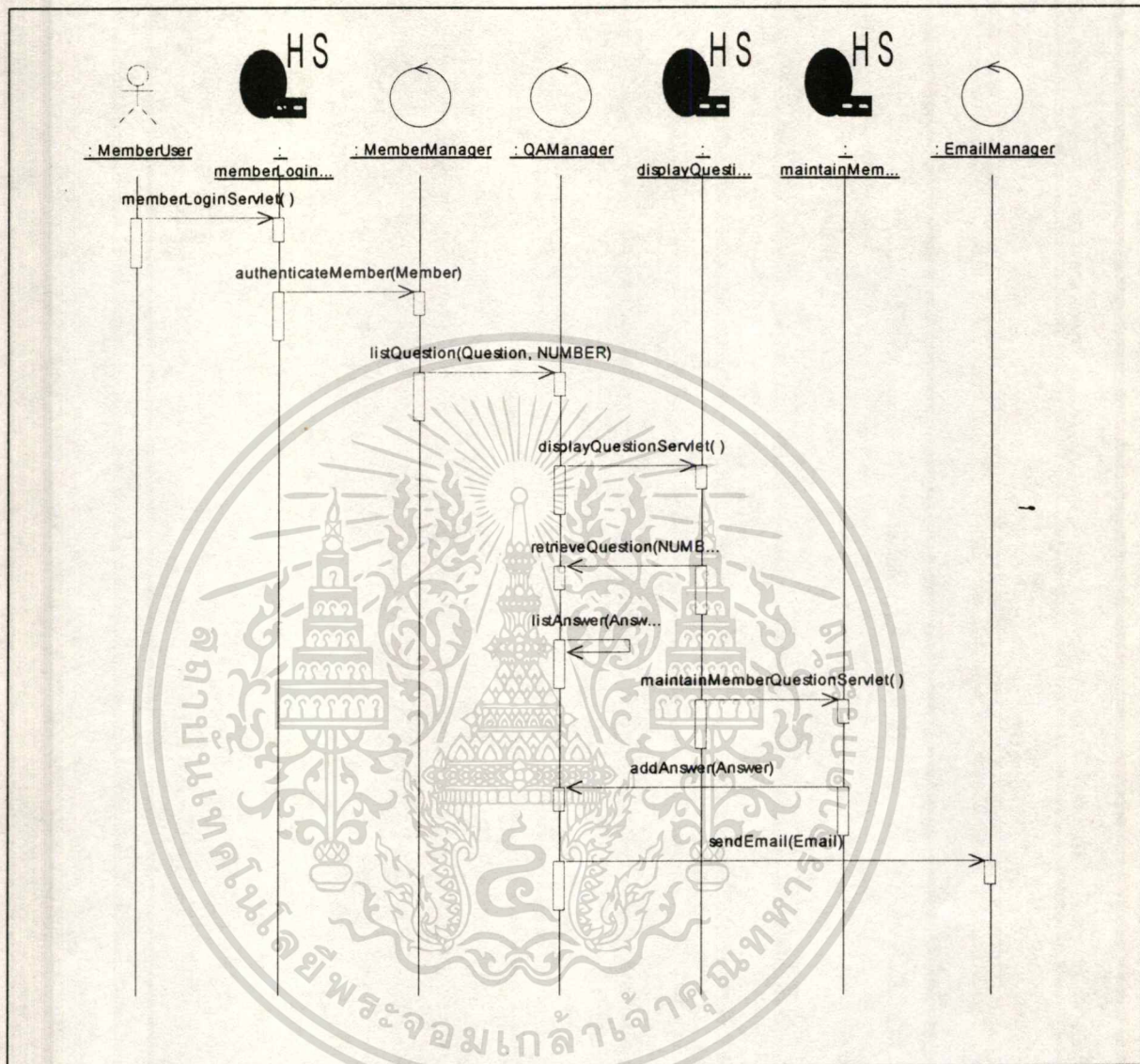
7) ข้อผิดพลาด : Create new Question



รูปที่ 6.46 Sequence Diagram: Create New Question

8) ข้อผิดพลาด : Response Staff Answer

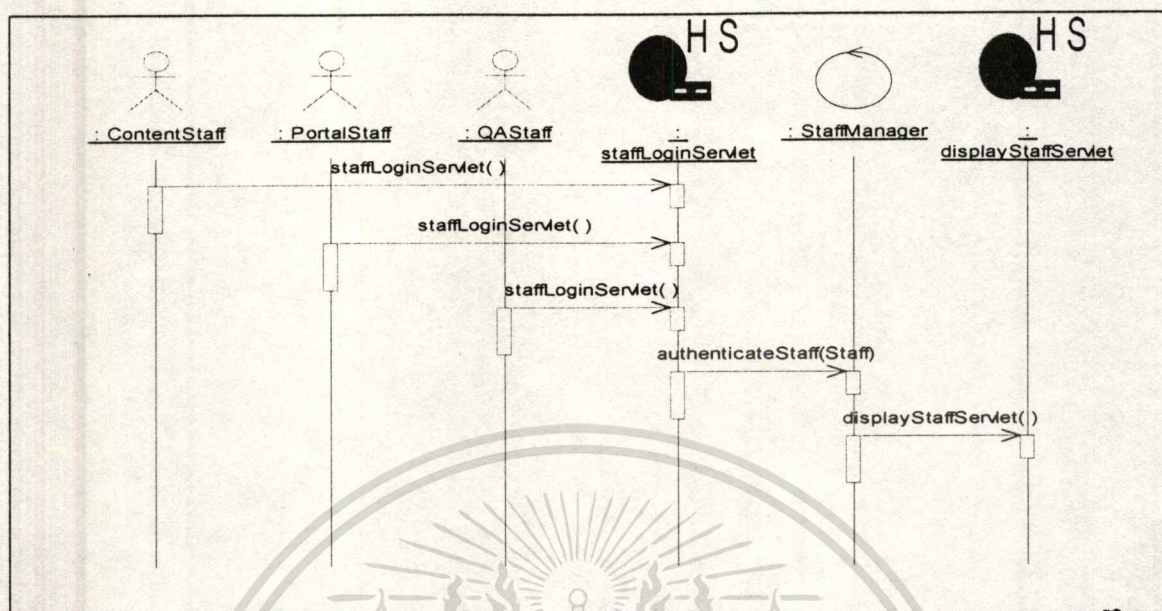
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.47 Sequence Diagram: Response Answer

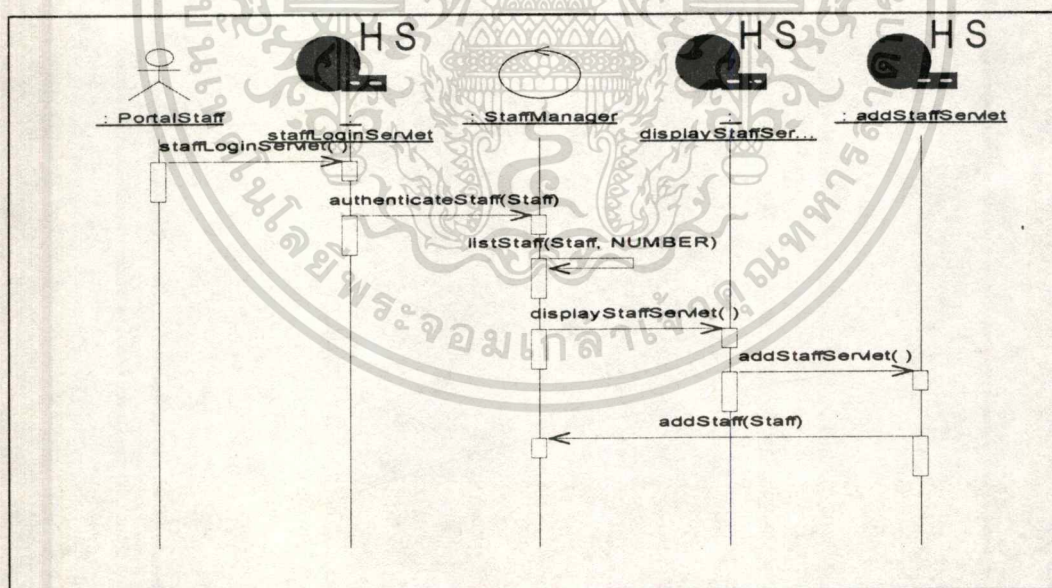
9) ชื่อชุดทดสอบ : Authenticate Staff

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

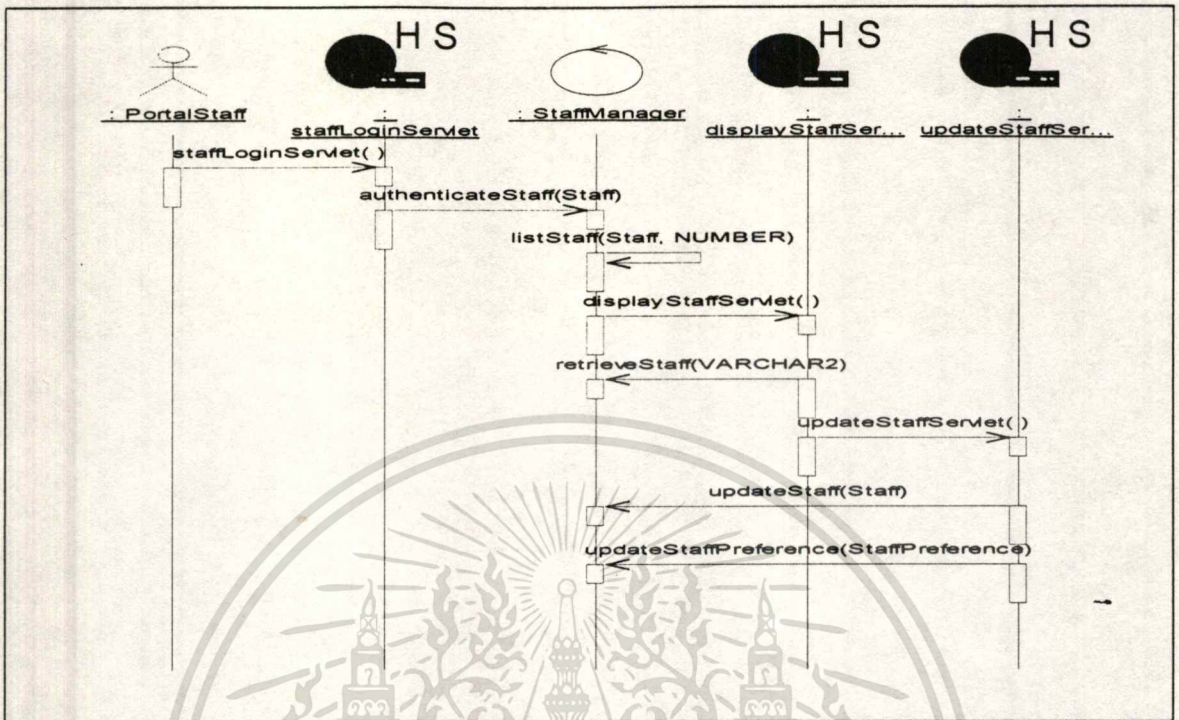


รูปที่ 6.48 Sequence Diagram: Authenticate Staff

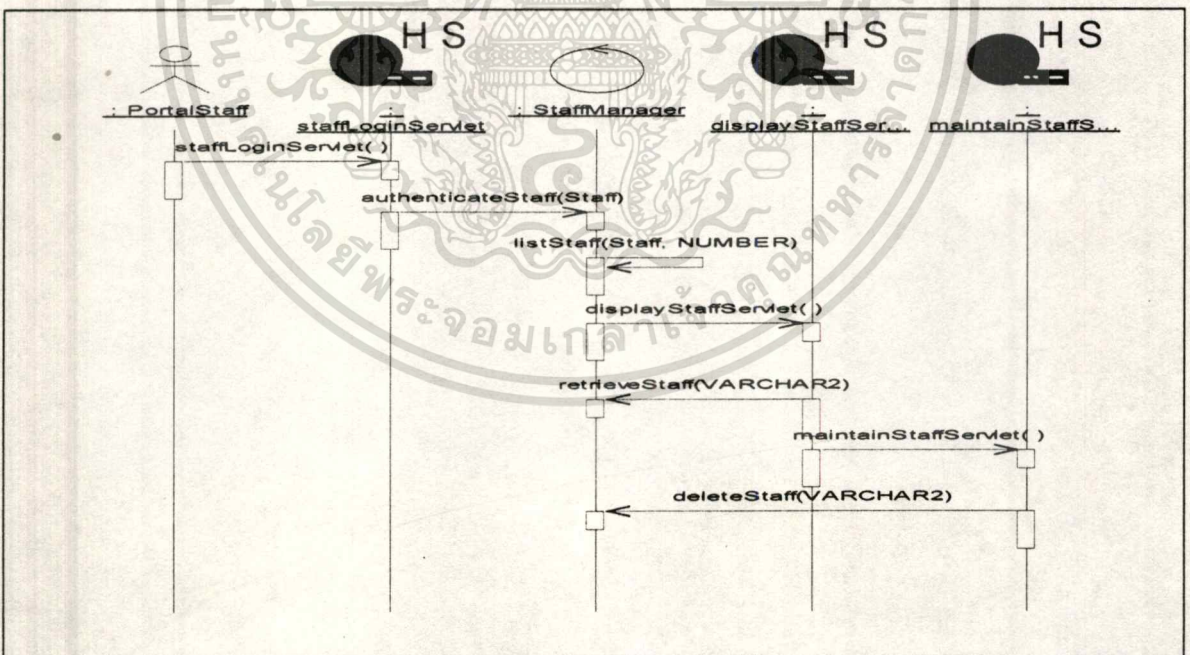
10) ซี่อยูสเคส : Maintain Staff Information



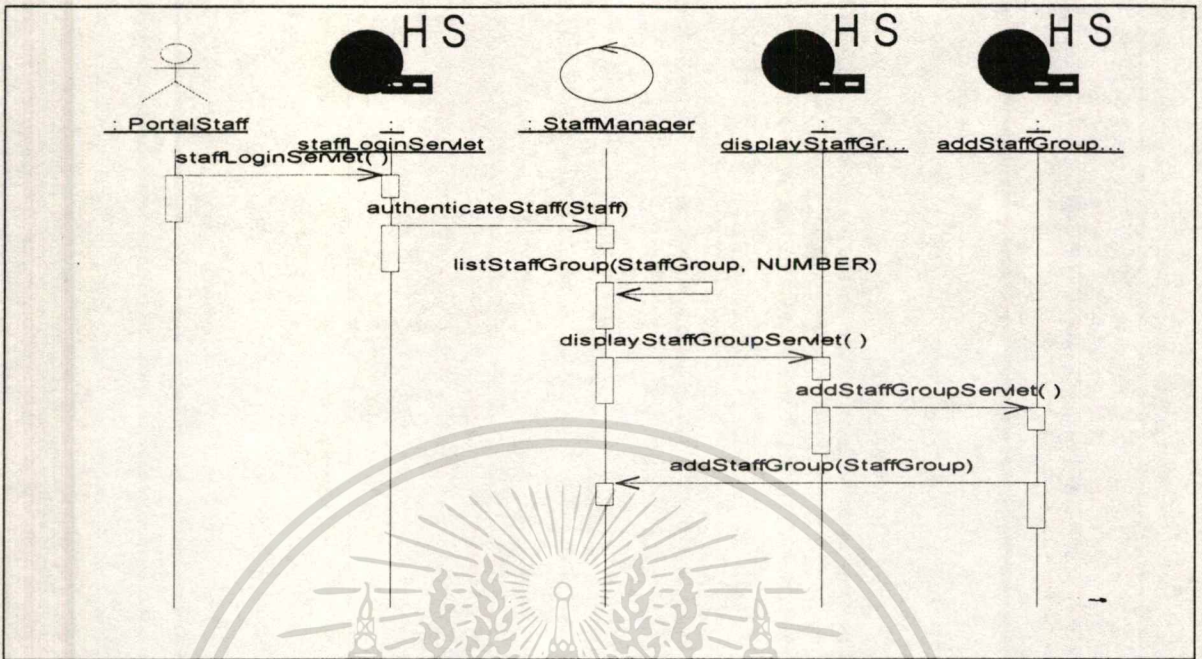
รูปที่ 6.49 Sequence Diagram: Add Staff



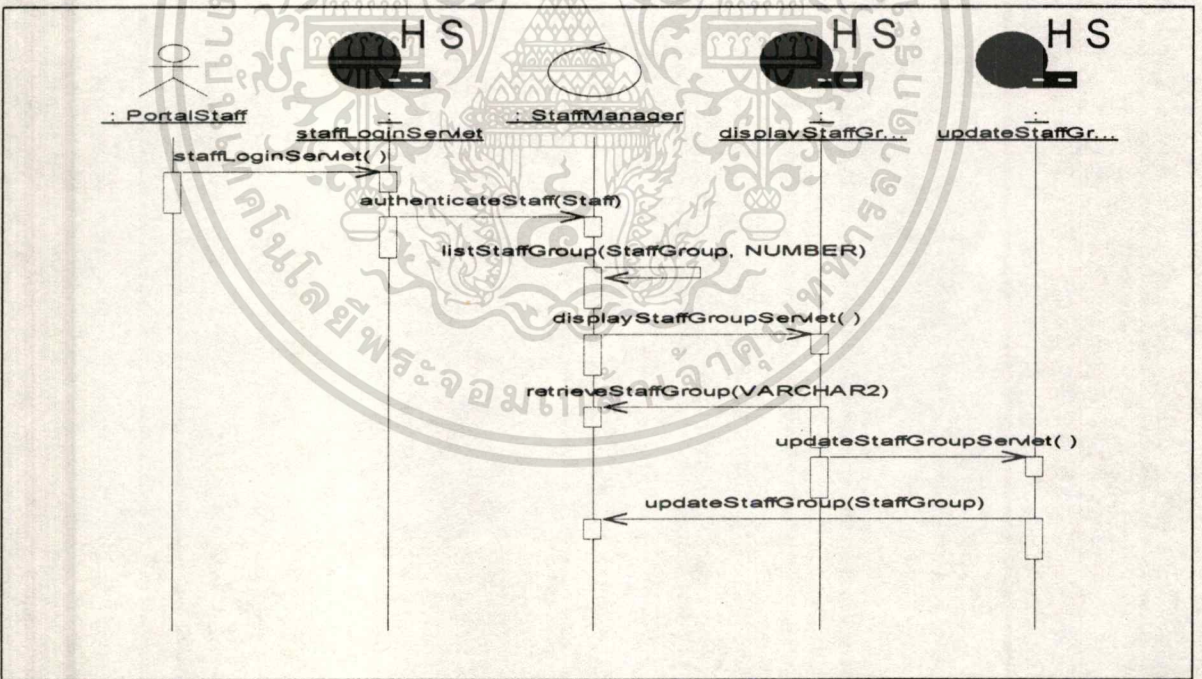
รูปที่ 6.50 Sequence Diagram: Update Staff



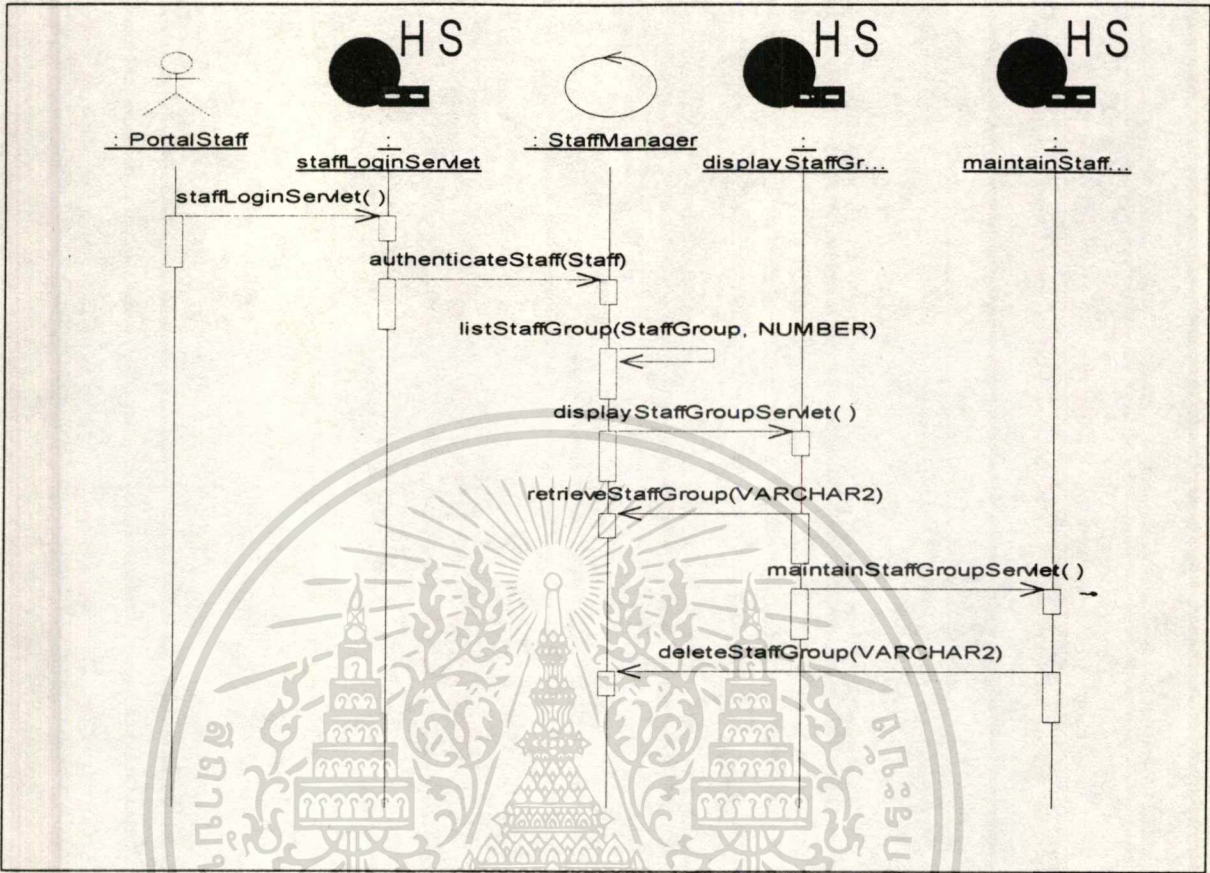
รูปที่ 6.51 Sequence Diagram: Delete Staff



รูปที่ 6.52 Sequence Diagram: Add Staff Group

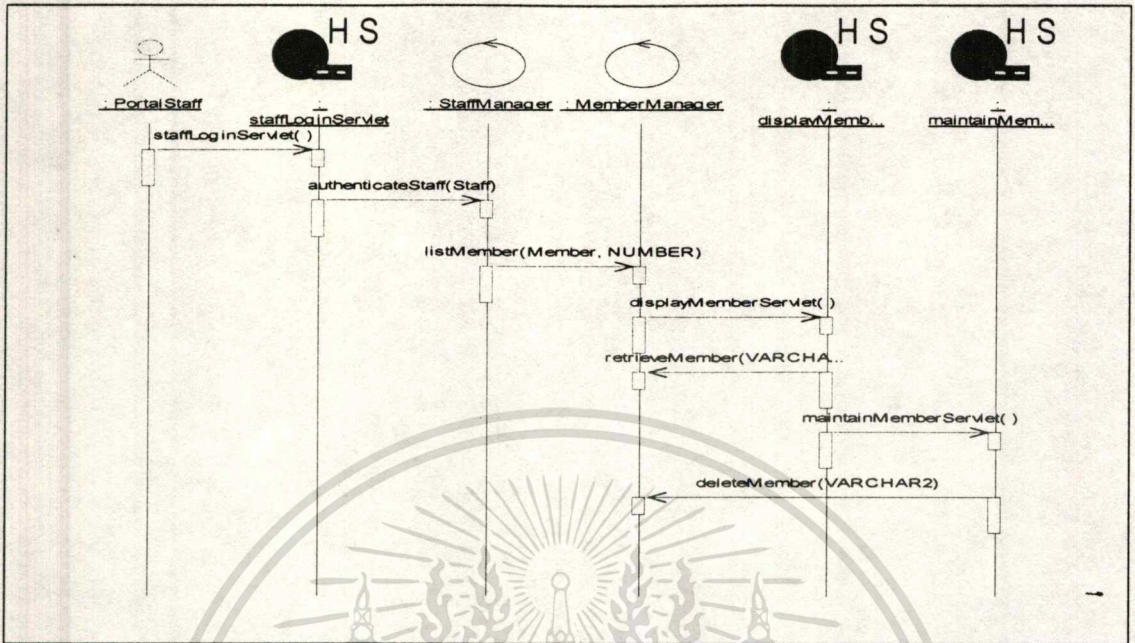


รูปที่ 6.53 Sequence Diagram: Update Staff Group

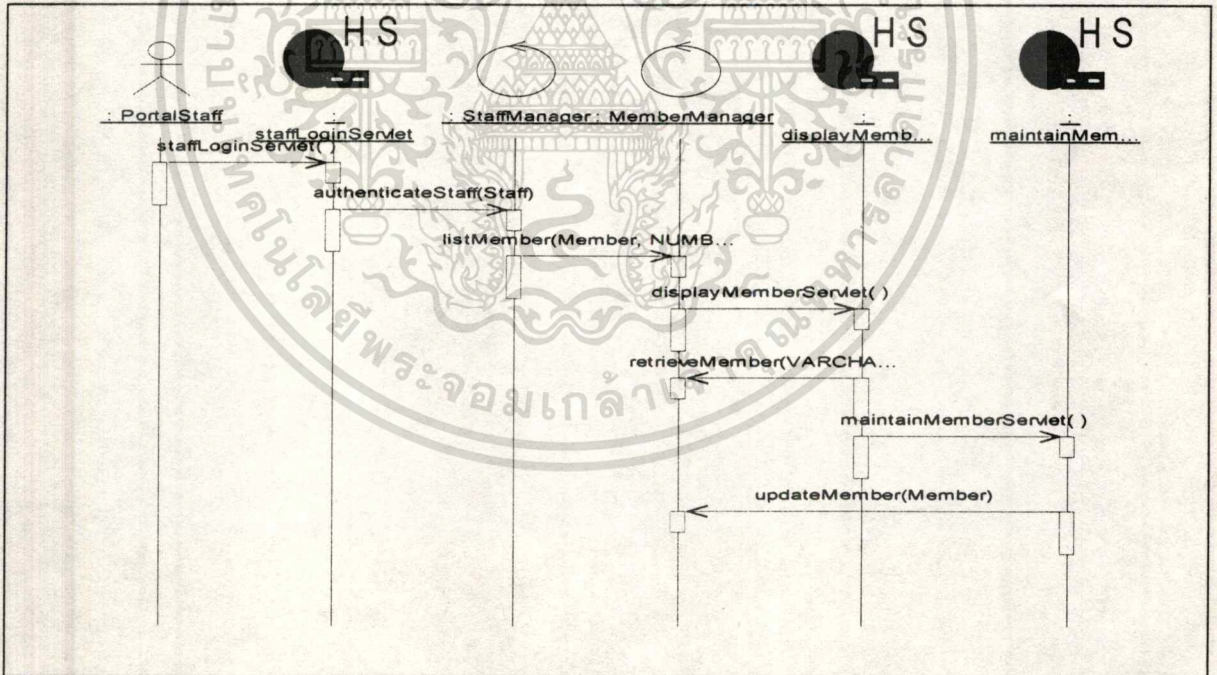


รูปที่ 6.54 Sequence Diagram: Delete Staff Group

11) ชื่อชุดเคส : Maintain Member Information



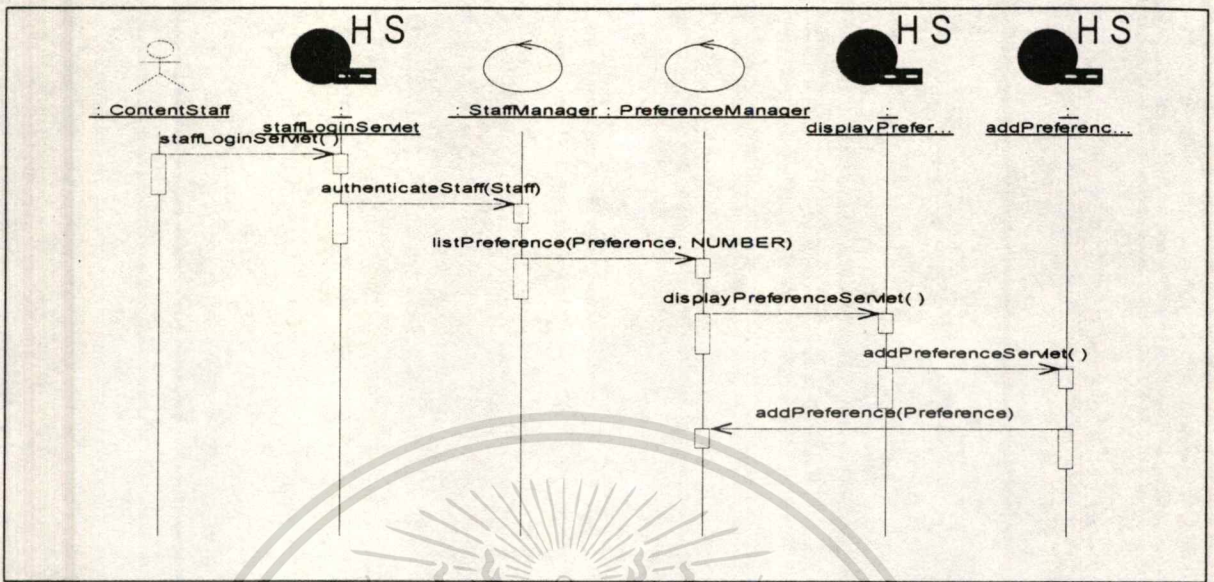
รูปที่ 6.55 Sequence Diagram: Delete Member



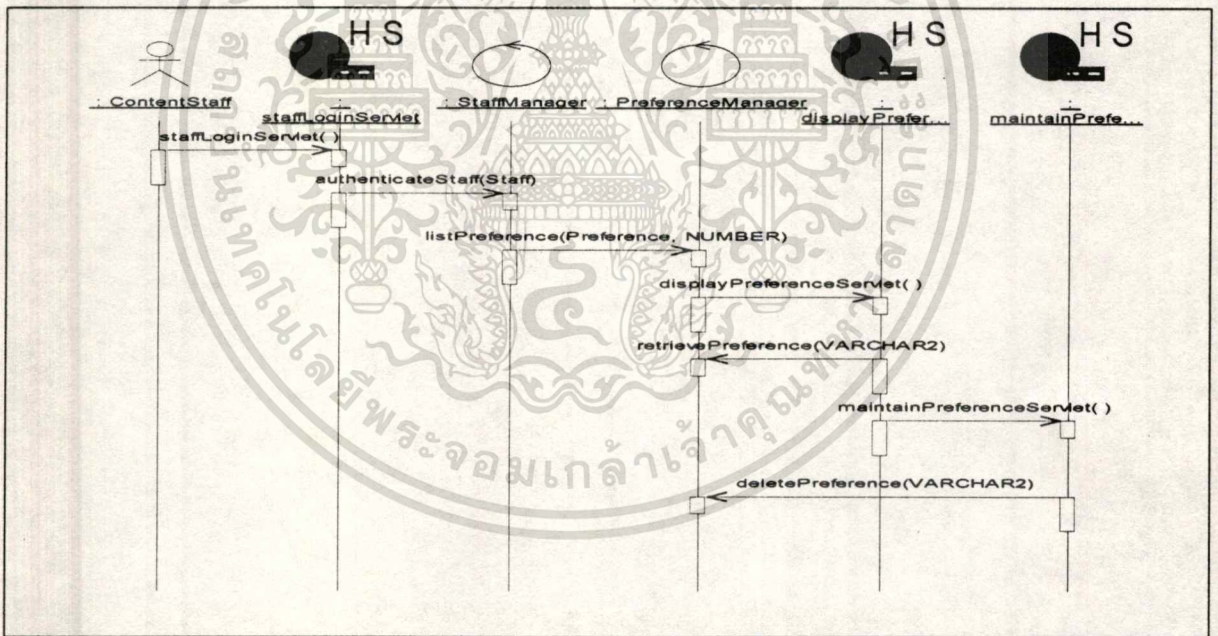
รูปที่ 6.56 Sequence Diagram: Update Member

12) ข้อมูลทดสอบ : Maintain Preference

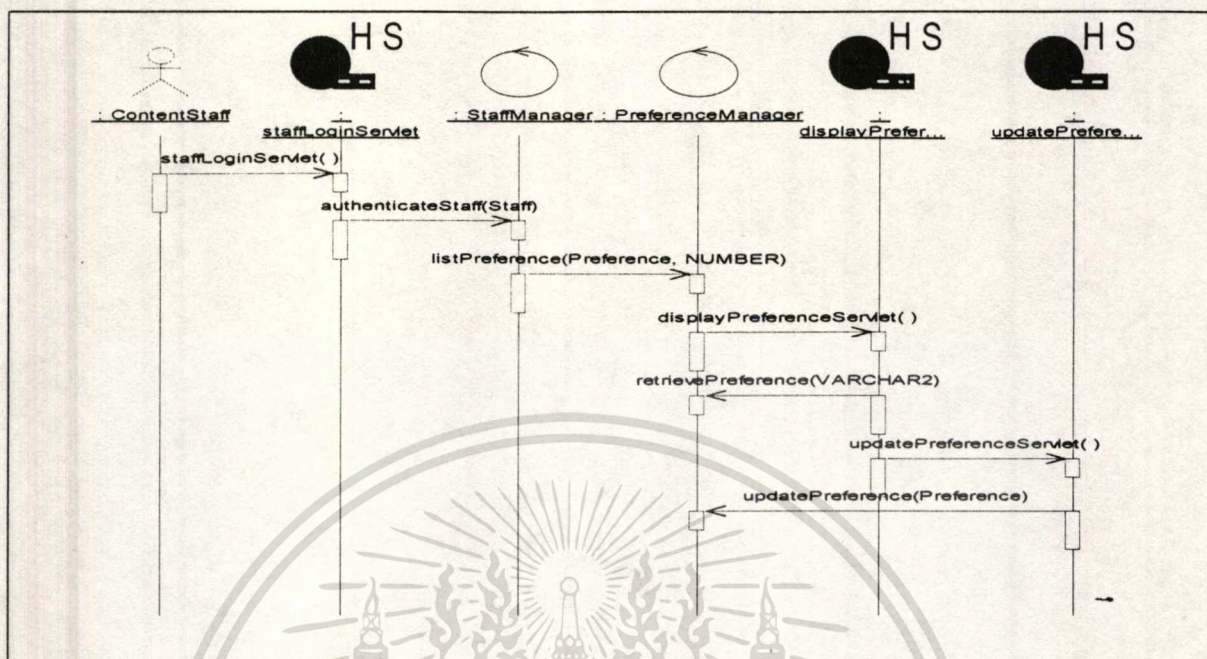
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.57 Sequence Diagram: Add Preference

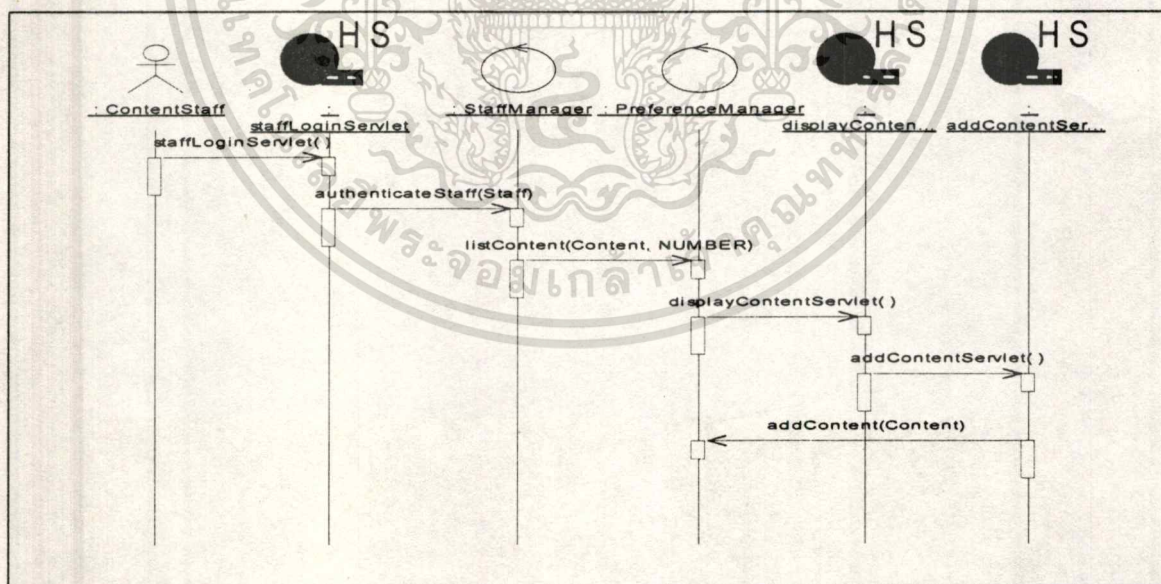


รูปที่ 6.58 Sequence Diagram: Delete Preference



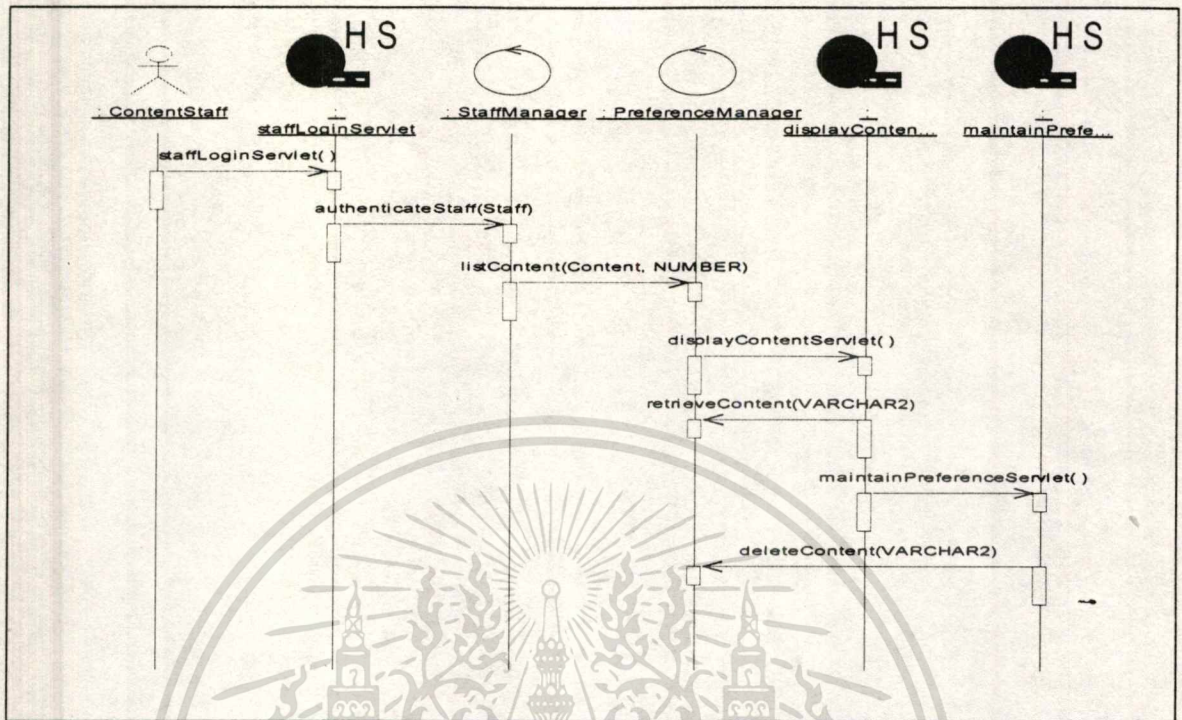
รูปที่ 6.59 Sequence Diagram: Update Preference

13) ข้อประสงค์ : Maintain Content

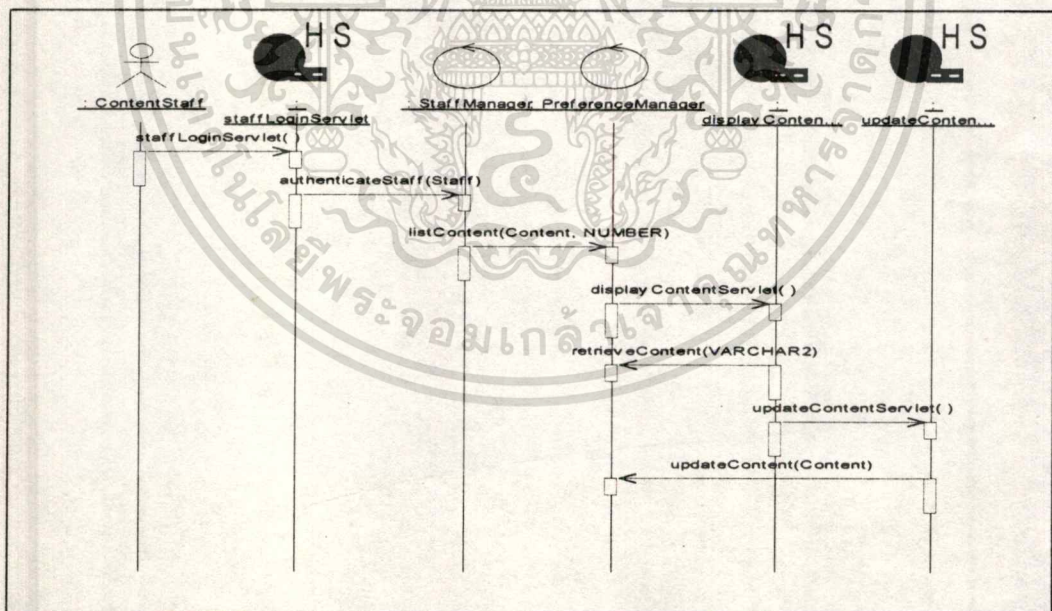


รูปที่ 6.60 Sequence Diagram: Add Content

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



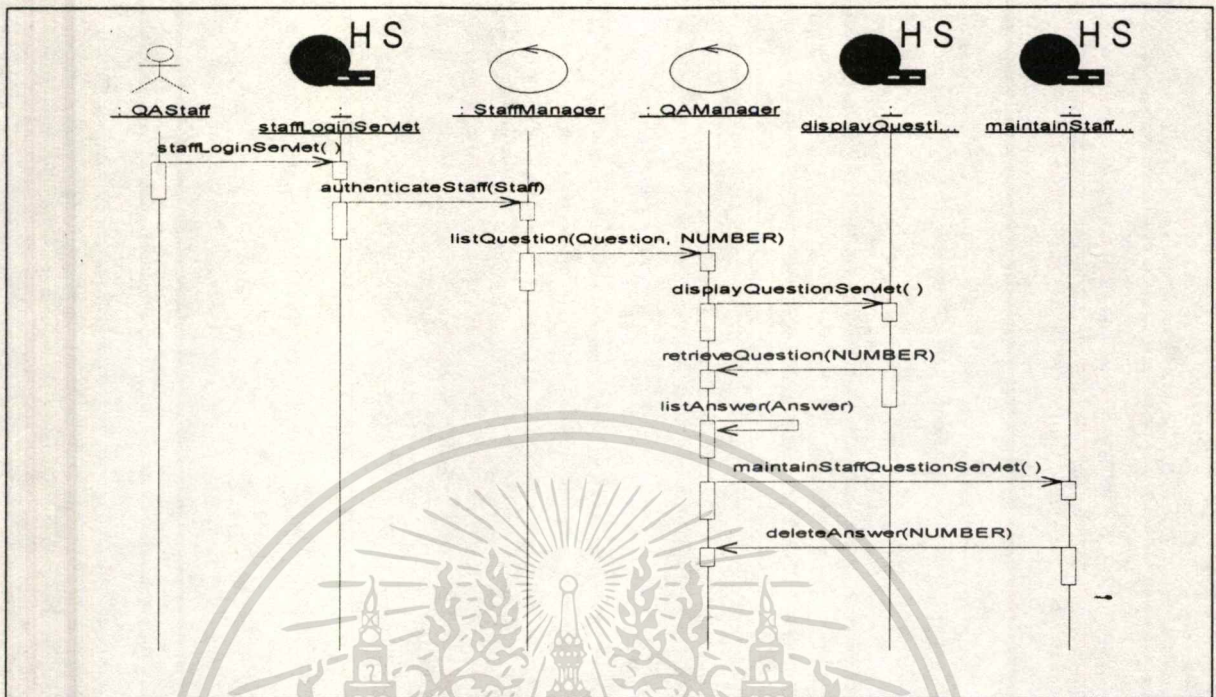
รูปที่ 6.61 Sequence Diagram: Delete Content



รูปที่ 6.62 Sequence Diagram: Update Content

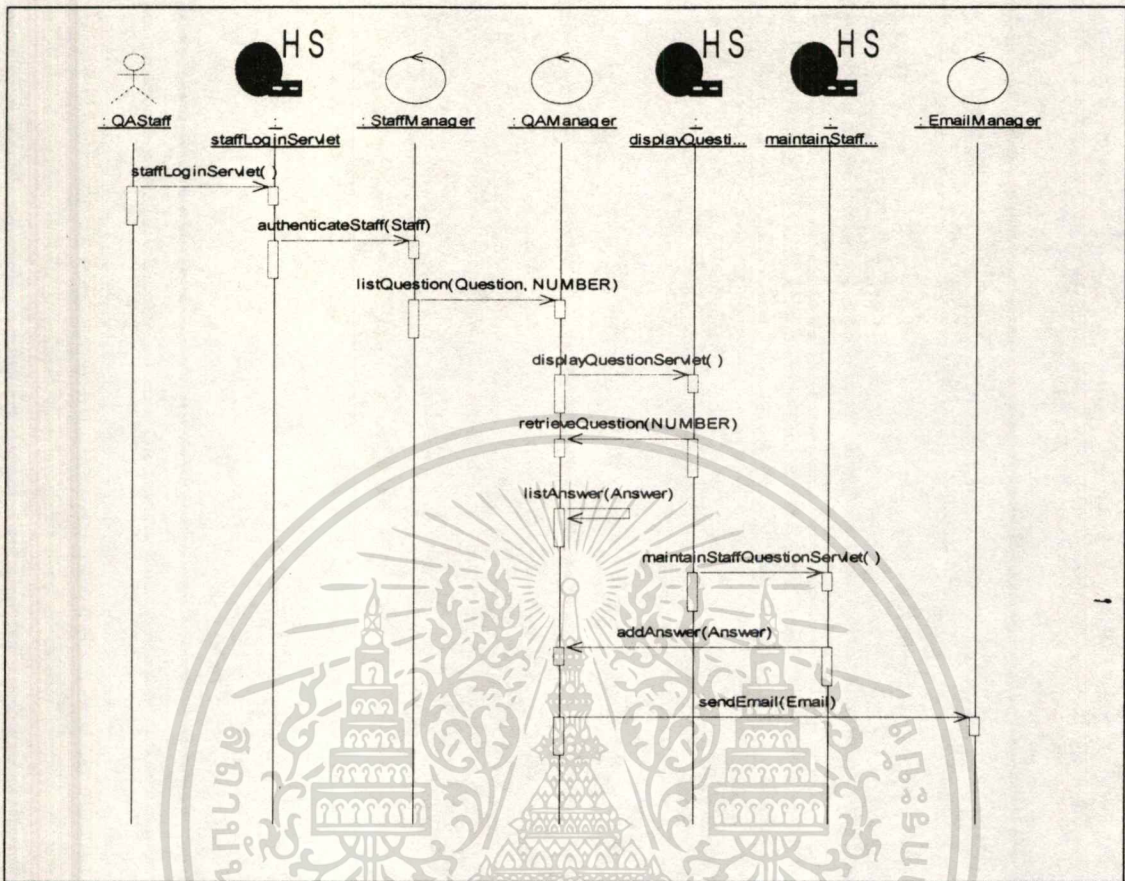
14) ชื่อชุดทดสอบ : Maintain QA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



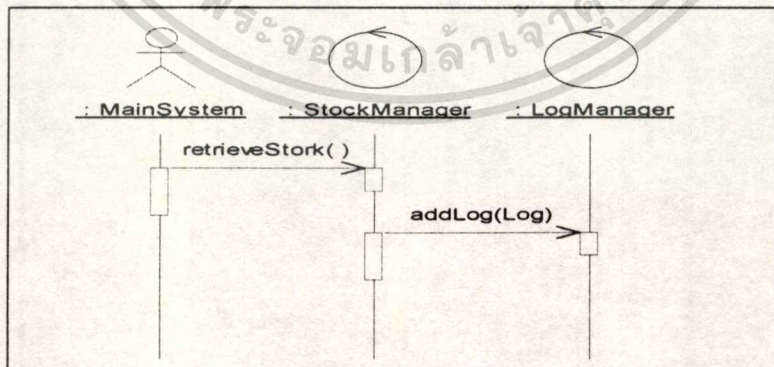
รูปที่ 6.63 Sequence Diagram: Maintain Question and Answer

15) ข้อผิดพลาด : Response Member QA



รูปที่ 6.64 Sequence Diagram: Response Question

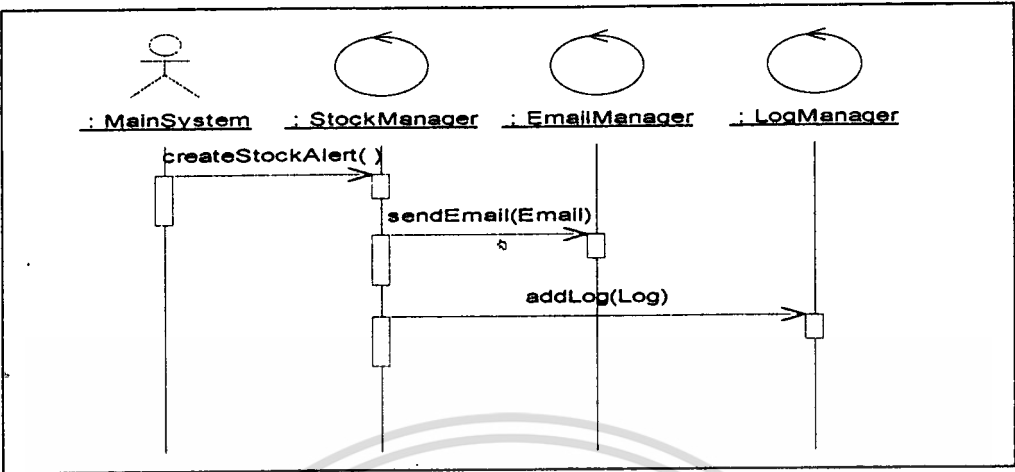
16) ชื่อชุดทดสอบ : Stock Transformation



รูปที่ 6.65 Sequence Diagram: Stock Transformation

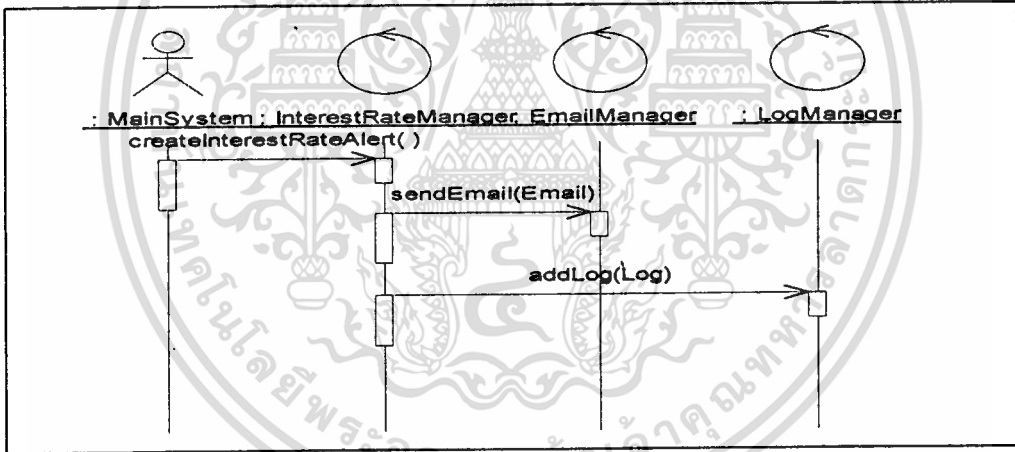
17) ชื่อชุดทดสอบ : Generate Stock Alert

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่จำกัดสิทธิ์ในสิ่งอื่นใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



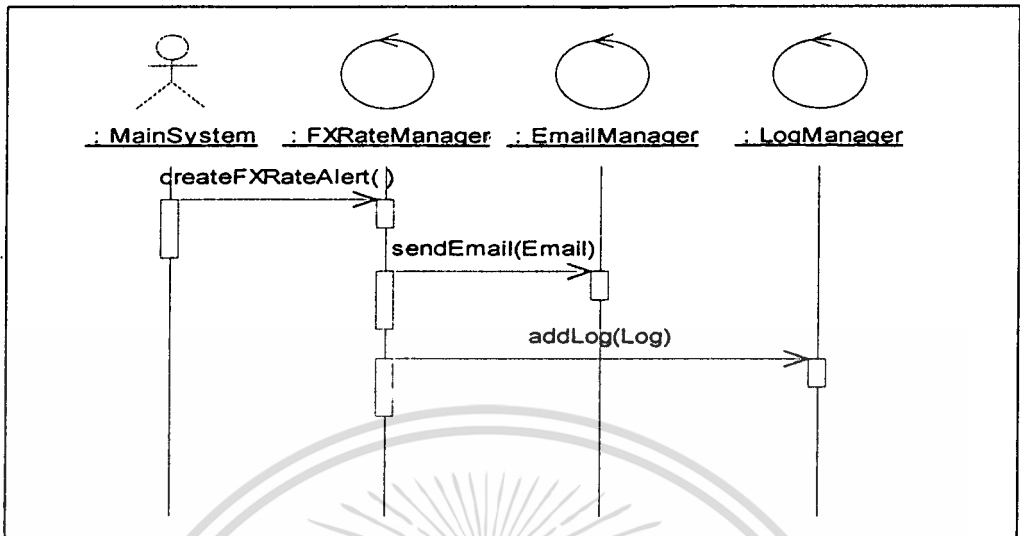
รูปที่ 6.66 Sequence Diagram: Generate Stock Alert

18) ข้อบกพร่อง : Generate Interest Rate Alert



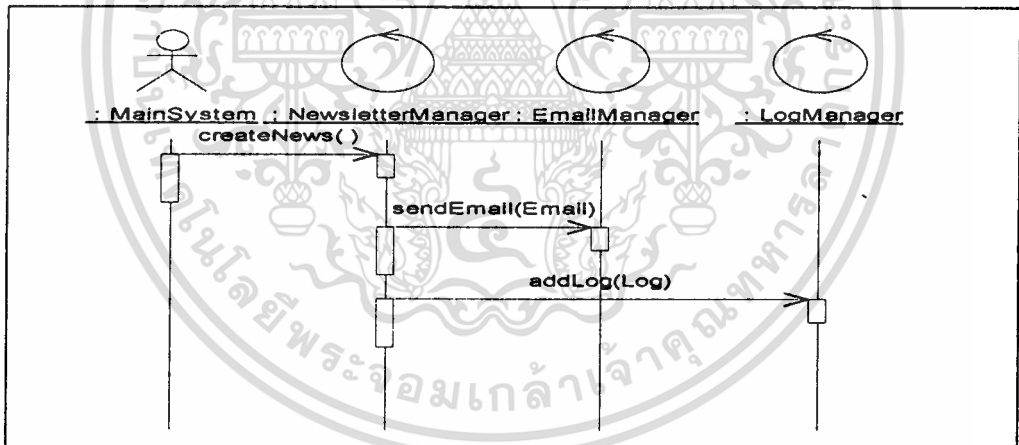
รูปที่ 6.67 Sequence Diagram: Generate Interest Rates Alert

19) ข้อบกพร่อง : Generate FX Rate Alert



รูปที่ 6.68 Sequence Diagram: Generate FX Rates Alert

20) ข้อผิดพลาด : Generate Newsletter



รูปที่ 6.69 Sequence Diagram: Generate Newsletter

บทที่ 7

การสร้างฐานข้อมูลระบบ BBL Portal

หลังจากที่รวบรวมความต้องการต่าง ๆ และวิเคราะห์ระบบ BBL Portal ในลำดับต่อมาจะเป็นการนำข้อมูลความต้องการเหล่านั้นมาพิจารณาเพื่อออกแบบฐานข้อมูลและเว็บแอปพลิเคชันต่อไป

7.1 การแปลงแบบจำลองเป็นฐานข้อมูล

จากการออกแบบระบบด้วยเทคโนโลยีเชิงวัตถุนั้น เว็บแอปพลิเคชันจะต้องติดต่อกับข้อมูลที่อยู่ในฐานข้อมูลเชิงสัมพันธ์ ดังนั้นการพัฒนาระบบเชิงวัตถุไม่สามารถนำมาจับคู่กับข้อมูลที่เป็นฐานข้อมูลได้ แต่มีวิธีแก้ไขปัญหาคือ มีเครื่องมือในการจับคู่ระหว่างออบเจกต์และข้อมูลเชิงสัมพันธ์ โดยใช้วิธี Reverse Engineering ซึ่งเป็นการสร้างออบเจกต์โมเดลจากฐานข้อมูลเชิงสัมพันธ์ และวิธี Forward Engineering โดยสร้างความสัมพันธ์

จากออบเจกต์โมเดล เครื่องมือที่ใช้สำหรับการจับคู่กันระหว่างออบเจกต์และข้อมูลเชิงสัมพันธ์มีขั้นตอนหลักคือ กำหนดความสัมพันธ์ระหว่างโครงสร้างของตารางในฐานข้อมูลเชิงสัมพันธ์ แล้วแทนด้วยคลาสในออบเจกต์โมเดล

ฐานข้อมูลเชิงสัมพันธ์ประกอบด้วยตารางที่มีแถวและคอลัมน์ซึ่งในออบเจกต์โมเดลเปรียบคลาสดูได้กับตาราง สำหรับแถวในตารางประกอบด้วยข้อมูลที่เปรียบเสมือนอินสแตนซ์ (Instance) ของคลาส คอลัมน์เปรียบได้กับแอตทริบิวต์ (Attribute) และ Stored Procedure ก็เปรียบได้กับเมธอด (Method) จึงได้มีการแปลงเป็นแบบจำลองที่สร้างขึ้นให้อยู่ในรูปแบบฐานข้อมูลเชิงสัมพันธ์ โดยใช้ CASE Tool Rational Rose เพื่อทำ Forward Engineering โดยใช้การสร้างไฟล์ Data Definition Language (DDL) แล้วนำไฟล์ที่ได้มาสร้างเป็นตารางในฐานข้อมูล Oracle

7.2 รายละเอียดข้อมูลที่จัดเก็บในระบบ

7.2.1 ตารางที่อยู่ในความรับผิดชอบของระบบ ประกอบด้วยตารางที่เกี่ยวข้องกับสมาชิก ผู้ดูแลระบบ และตารางที่เก็บข้อมูลของการทำงานในแต่ละหน้าที่ ดังต่อไปนี้

1. ข้อมูลเกี่ยวกับสมาชิก

- ข้อมูลหลักรายละเอียดสมาชิก (Member)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ข้อมูลหลักกลุ่มข้อมูลข่าวสารที่สมาชิกสนใจ (MemberPreference)
- ข้อมูลหลักอัตราดอกเบี้ยที่สมาชิกสนใจ (MemberInterestRate)
- ข้อมูลหลักอัตราแลกเปลี่ยนที่สมาชิกสนใจ (MemberFXRate)
- ข้อมูลหลักราคาหุ้นที่สมาชิกสนใจ (MemberStock)
- ข้อมูลหลักสินค้าที่สมาชิกสนใจ (MemberProduct)

2. ข้อมูลเกี่ยวกับข้อมูลข่าวสารในเว็บไซต์

- ข้อมูลหลักกลุ่มข้อมูลข่าวสาร (Preference)
- ข้อมูลหลักข้อมูลข่าวสาร (Content)
- ข้อมูลหลักรายละเอียดข้อมูลข่าวสาร (ContentDetail)
- ข้อมูลหลักข้อมูลข่าวสารที่สัมพันธ์กับสินค้า (ContentProduct)

3. ข้อมูลเกี่ยวกับการสอบถามผู้เชี่ยวชาญ

- ข้อมูลหลักรายละเอียดคำถามจากสมาชิก (Question)
- ข้อมูลหลักรายละเอียดการได้ตอบ (Answer)

4. ข้อมูลเกี่ยวกับผู้ดูแลระบบและผู้จัดการข้อมูล

- ข้อมูลหลักรายละเอียดเจ้าหน้าที่ (Staff)
- ข้อมูลหลักกลุ่มเจ้าหน้าที่ (StaffGroup)
- ข้อมูลหลักรายละเอียดกลุ่มข้อมูลข่าวสารที่เจ้าหน้าที่จัดการ (StaffPreference)

5. ข้อมูลเกี่ยวกับมูลค่าหุ้น

- ข้อมูลหลักมูลค่าหุ้น (Stock)

6. ข้อมูลทั่วไปของระบบ

- ข้อมูลหลักสินค้า (Product)
- ข้อมูลหลักกลุ่มสินค้า (ProductGroup)
- ข้อมูลหลักชื่อเมือง (City)
- ข้อมูลหลักชื่อประเทศ (Country)
- ข้อมูลหลักสัญชาติ (Nation)
- ข้อมูลหลักอาชีพ (Job)
- ข้อมูลหลักประเภทกิจการ (BusinessType)
- ข้อมูลหลักประวัติการจัดการอัตราดอกเบี้ย อัตราแลกเปลี่ยน และมูลค่าหุ้น (Log)

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์และใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ข้อมูลหลักบริการ (Service)

ที่ได้มีการขอใช้งานร่วมกัน

7.2.2 ตารางที่อยู่ในความรับผิดชอบของระบบอื่น
ประกอบด้วยตารางที่เกี่ยวกับข้อมูลหลักต่าง ๆ ดังต่อไปนี้

- 1. ข้อมูลหลักอัตราดอกเบี้ย (InterestRate)
- 2. ข้อมูลหลักกลุ่มอัตราดอกเบี้ย (InterestRateGroup)
- 3. ข้อมูลหลักอัตราแลกเปลี่ยน (FXRate)

ซึ่งมีรายละเอียดของแต่ละตารางดังตารางที่ 7.1 -7.29

ตารางที่ 7.1 ข้อมูลหลักรายละเอียดสมาชิก (Member)

Table Name: Member					
Description: ข้อมูลหลักรายละเอียดสมาชิก					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
MemberID	รหัสสมาชิก	VARCHAR2	15	PK	
MemberLoginName	ชื่อเข้าสู่ระบบ	VARCHAR2	15		
MemberPassword	รหัสผ่านเข้าสู่ระบบ	VARCHAR2	15		
MemberName	ชื่อสมาชิก	VARCHAR2	50		
MemberSurname	ชื่อสกุลสมาชิก	VARCHAR2	50		
MemberTitle	คำนำหน้าสมาชิก	VARCHAR2	10		
MemberSex	เพศสมาชิก	CHAR	1		
MemberCitizenID	เลขที่บัตรประจำตัว ประชาชนสมาชิก	VARCHAR2	13		
MemberBirthDate	วันเกิดสมาชิก	DATE			
MemberLastLoginDate	วันที่เข้าสู่ระบบครั้ง ล่าสุด	DATE			
MemberDisable	สถานะสมาชิก	CHAR	1		
MemberEmail	อีเมลสมาชิก	VARCHAR2	50		
MemberAddress	ที่อยู่สมาชิก	VARCHAR2	200		

CityID	รหัสเมือง	VARCHAR2	2	FK	City
NationID	รหัสสัญชาติ	VARCHAR2	2	FK	Nation
JobID	รหัสอาชีพ	VARCHAR2	2	FK	Job
BusinessTypeID	รหัสประเภทกิจการ	VARCHAR2	2	FK	BusinessType

ตารางที่ 7.2 ข้อมูลหลักกลุ่มข้อมูลข่าวสารที่สมาชิกสนใจ (MemberPreference)

Table Name: MemberPreference					
Description: ข้อมูลหลักกลุ่มข้อมูลข่าวสารที่สมาชิกสนใจ					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
MemberID	รหัสสมาชิก	VARCHAR2	15	PK,FK	Member
PreferenceID	รหัสกลุ่มข้อมูลข่าวสาร	VARCHAR2	2	PK,FK	Preference

ตารางที่ 7.3 ข้อมูลหลักอัตราดอกเบี้ยที่สมาชิกสนใจ (MemberInterestRate)

Table Name: MemberInterestRate					
Description: ข้อมูลหลักอัตราดอกเบี้ยที่สมาชิกสนใจ					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
MemberID	รหัสสมาชิก	VARCHAR2	15	PK,FK	Member
InterestRateID	รหัสอัตราแลกเปลี่ยน	VARCHAR2	2	PK,FK	InterestRate
InterestRateLowValue	อัตราดอกเบี้ยต่ำสุดที่สนใจ	NUMBER	2,2		
InterestRateHighValue	อัตราดอกเบี้ยสูงสุดที่สนใจ	NUMBER	2,2		

ตารางที่ 7.4 ข้อมูลหลักอัตราแลกเปลี่ยนที่สมาชิกสนใจ (MemberFXRate)

Table Name: MemberFXRate					
Description: ข้อมูลหลักอัตราแลกเปลี่ยนที่สมาชิกสนใจ					
Attribute Name	Content	Type	Length	Key	Foreign Key

					Referenced Table
MemberID	รหัสสมาชิก	VARCHAR2	15	PK,FK	Member
FXRateID	รหัสอัตราแลกเปลี่ยน	VARCHAR2	2	PK,FK	FXRate
FXRateLowValue	อัตราแลกเปลี่ยนต่ำสุดที่สนใจ	NUMBER	3		
FXRateHighValue	อัตราแลกเปลี่ยนสูงสุดที่สนใจ	NUMBER	3		

ตารางที่ 7.5 ข้อมูลหลักราคาหุ้นที่สมาชิกสนใจ (MemberStock)

Table Name: MemberStock					
Description: ข้อมูลหลักราคาหุ้นที่สมาชิกสนใจ					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
MemberID	รหัสสมาชิก	VARCHAR2	15	PK,FK	Member
StockID	รหัสหุ้น	VARCHAR2	2	PK,FK	Stock
StockLowPrice	ราคาหุ้นต่ำสุดที่สนใจ	NUMBER	5,2		
StockHighPrice	ราคาหุ้นสูงสุดที่สนใจ	NUMBER	5,2		

ตารางที่ 7.6 ข้อมูลหลักสินค้าที่สมาชิกสนใจ (MemberProduct)

Table Name: MemberProduct					
Description: ข้อมูลหลักสินค้าที่สมาชิกสนใจ					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
MemberID	รหัสสมาชิก	VARCHAR2	15	PK,FK	Member
ProductID	รหัสสินค้า	VARCHAR2	2	PK,FK	Product

ตารางที่ 7.7 ข้อมูลหลักกลุ่มข้อมูลข่าวสาร (Preference)

Table Name: Preference					
Description: ข้อมูลหลักกลุ่มข้อมูลข่าวสาร					

Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
PreferenceID	รหัสกลุ่มข้อมูลข่าวสาร	VARCHAR2	2	PK	
PreferenceName	ชื่อกลุ่มข้อมูลข่าวสาร	VARCHAR2	50		

ตารางที่ 7.8 ข้อมูลหลักข้อมูลข่าวสาร (Content)

Table Name: Content					
Description: ข้อมูลหลักข้อมูลข่าวสาร					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
ContentID	รหัสข้อมูลข่าวสาร	NUMBER	10	PK	
ContentTitle	สรุปข้อมูลข่าวสาร	VARCHAR2	100		
ContentBrief	เกริ่นนำข้อมูล ข่าวสาร	VARCHAR2	500		
PreferenceID	กลุ่มข้อมูลข่าวสาร	VARCHAR2	2	FK	Preference
ContentStartDate	วันที่เริ่มเผยแพร่	DATE			
ContentEndDate	วันที่หยุดเผยแพร่	DATE			
StaffID	รหัสเจ้าหน้าที่	VARCHAR2	10	FK	Staff
ContentToNewsletter	ส่งเป็นอีเมลหรือไม่	CHAR	1		
ContentToSpotlightOrder	จัดเป็นข้อมูล ข่าวสารเด่นลำดับที่ เท่าไร	NUMBER	2		

ตารางที่ 7.9 ข้อมูลหลักรายละเอียดข้อมูลข่าวสาร (ContentDetail)

Table Name: ContentDetail					
Description: ข้อมูลหลักรายละเอียดข้อมูลข่าวสาร					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
ContentDetailID	รหัสรายละเอียดข้อมูล	NUMBER	15	PK	

	ข่าวสาร				
ContentID	รหัสข้อมูลข่าวสาร	NUMBER	10	FK	Content
ContentOrder	ลำดับที่รายละเอียด	NUMBER	2		
ContentDetailText	ข้อความ	VARCHAR 2	500		
ContentDetailImage	รูปภาพประกอบ	VARCHAR 2	50		
ContentDetailImagePosition	ตำแหน่งรูปภาพประกอบ	CHAR	1		

ตารางที่ 7.10 ข้อมูลหลักข้อมูลข่าวสารที่สัมพันธ์กับสินค้า (ContentProduct)

Table Name: ContentProduct					
Description: ข้อมูลหลักข้อมูลข่าวสารที่สัมพันธ์กับสินค้า					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
ContentID	รหัสข้อมูลข่าวสาร	NUMBER	10	PK,FK	Content
ProductID	รหัสสินค้า	VARCHAR2	2	PK,FK	Product

ตารางที่ 7.11 ข้อมูลหลักรายละเอียดคำถามจากสมาชิก (Question)

Table Name: Question					
Description: ข้อมูลหลักรายละเอียดคำถามจากสมาชิก					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
QuestionID	รหัสคำถาม	NUMBER	10	PK	
MemberID	รหัสสมาชิก	VARCHAR2	15	FK	Member
QuestionTitle	สรุปคำถาม	VARCHAR2	100		
QuestionText	ข้อความคำถาม	VARCHAR2	200		
QuestionTime	เวลาถาม	DATE			

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 7.12 ข้อมูลหลักรายละเอียดการ ได้ตอบ (Answer)

Table Name: Answer					
Description: ข้อมูลหลักรายละเอียดการ ได้ตอบ					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
AnswerID	รหัสคำตอบ	NUMBER	15	PK	
QuestionID	รหัสคำถาม	NUMBER	10	FK	Question
StaffID	รหัสเจ้าหน้าที่	VARCHAR2	10	FK	Staff
AnswerText	ข้อความ	VARCHAR2	200		
AnswerTime	เวลาที่ตอบ	DATE			

ตารางที่ 7.13 ข้อมูลหลักรายละเอียดเจ้าหน้าที่ (Staff)

Table Name: Staff					
Description: ข้อมูลหลักรายละเอียดเจ้าหน้าที่					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
StaffID	รหัสเจ้าหน้าที่	VARCHAR2	10	PK	
StaffGroupID	รหัสกลุ่มเจ้าหน้าที่	VARCHAR2	2	FK	StaffGroup
StaffName	ชื่อเจ้าหน้าที่	VARCHAR2	50		
StaffSurname	นามสกุลเจ้าหน้าที่	VARCHAR2	50		
StaffLoginName	ชื่อเข้าสู่ระบบ	VARCHAR2	15		
StaffPassword	รหัสผ่านเข้าสู่ระบบ	VARCHAR2	15		

ตารางที่ 7.14 ข้อมูลหลักกลุ่มเจ้าหน้าที่ (StaffGroup)

Table Name: StaffGroup					
Description: ข้อมูลหลักกลุ่มเจ้าหน้าที่					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
StaffGroupID	รหัสกลุ่มเจ้าหน้าที่	VARCHAR2	2	PK	

StaffGroupName	ชื่อกลุ่มเจ้าหน้าที่	VARCHAR2	50		
----------------	----------------------	----------	----	--	--

ตารางที่ 7.15 ข้อมูลหลักรายละเอียดกลุ่มข้อมูลข่าวสารที่เจ้าหน้าที่จัดการ (StaffPreference)

Table Name: StaffPreference					
Description: ข้อมูลหลักรายละเอียดกลุ่มข้อมูลข่าวสารที่เจ้าหน้าที่จัดการ					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
StaffID	รหัสเจ้าหน้าที่	VARCHAR2	10	PK,FK	Staff
PreferenceID	รหัสกลุ่มข้อมูล ข่าวสาร	VARCHAR2	2	PK,FK	Preference

ตารางที่ 7.16 ข้อมูลหลักมูลค่าหุ้น (Stock)

Table Name: Stock					
Description: ข้อมูลหลักมูลค่าหุ้น					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
StockID	รหัสหุ้น	VARCHAR2	2	PK	
StockShortName	ชื่อย่อของหุ้น	VARCHAR2	15		
StockName	ชื่อเต็มของหุ้น	VARCHAR2	50		
StockPrice	ราคาหุ้น	NUMBER	5,2		
StockStartedPrice	ราคาหุ้นตอนเปิดวัน	NUMBER	5,2		
StockUpdatedTime	เวลาที่ปรับปรุงราคา	DATE			

ตารางที่ 7.17 ข้อมูลหลักสินค้า (Product)

Table Name: Product					
Description: ข้อมูลหลักสินค้า					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
ProductID	รหัสสินค้า	VARCHAR2	2	PK	

ProductName	ชื่อสินค้า	VARCHAR2	50		
ProductGroupID	รหัสกลุ่มสินค้า	VARCHAR2	2	FK	ProductGroup

ตารางที่ 7.18 ข้อมูลหลักกลุ่มสินค้า (ProductGroup)

Table Name: ProductGroup					
Description: ข้อมูลหลักกลุ่มสินค้า					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
ProductGroupID	รหัสกลุ่มสินค้า	VARCHAR2	2	PK	
ProductGroupName	ชื่อกลุ่มสินค้า	VARCHAR2	50		

ตารางที่ 7.19 ข้อมูลหลักชื่อเมือง (City)

Table Name: City					
Description: ข้อมูลหลักชื่อเมือง					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
CityID	รหัสเมือง	VARCHAR2	2	PK	
CityName	ชื่อเมือง	VARCHAR2	50		
CountryID	รหัสประเทศ	VARCHAR2	2	FK	Country

ตารางที่ 7.20 ข้อมูลหลักชื่อประเทศ (Country)

Table Name: Country					
Description: ข้อมูลหลักชื่อประเทศ					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
CountryID	รหัสประเทศ	VARCHAR2	2	PK	
CountryName	ชื่อประเทศ	VARCHAR2	50		

ตารางที่ 7.21 ข้อมูลหลักสัญชาติ (Nation)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table Name: Nation					
Description: ข้อมูลหลักสัญชาติ					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
NationID	รหัสสัญชาติ	VARCHAR2	2	PK	
NationName	ชื่อสัญชาติ	VARCHAR2	50		

ตารางที่ 7.22 ข้อมูลหลักอาชีพ (Job)

Table Name: Job					
Description: ข้อมูลหลักอาชีพ					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
JobID	รหัสอาชีพ	VARCHAR2	2	PK	
JobName	ชื่ออาชีพ	VARCHAR2	50		

ตารางที่ 7.23 ข้อมูลหลักประเภทกิจการ (BusinessType)

Table Name: BusinessType					
Description: ข้อมูลหลักประเภทกิจการ					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
BusinessTypeID	รหัสประเภทกิจการ	VARCHAR2	2	PK	
BusinessTypeName	ชื่อประเภทกิจการ	VARCHAR2	50		

ตารางที่ 7.24 ข้อมูลหลักประวัติการจัดการอัตราดอกเบี้ย อัตราแลกเปลี่ยน และมูลค่าหุ้น (Log)

Table Name: Log					
Description: ข้อมูลหลักประวัติการจัดการอัตราดอกเบี้ย อัตราแลกเปลี่ยน และมูลค่าหุ้น					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
LogID	รหัสการทำงาน	NUMBER	20	PK	

ServiceID	รหัสบริการ	VARCHAR2	2	FK	Service
ActionID	รหัสการทำงาน	VARCHAR2	2	FK	Action
LogDate	วันที่ทำงาน	DATE			
LogRemark	หมายเหตุ	VARCHAR2	200		
LogSucceeded	สถานะการทำงาน	CHAR	1		

ตารางที่ 7.25 ข้อมูลหลักการทำงาน (Action)

Table Name: Action					
Description: ข้อมูลหลักการทำงาน					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
ActionID	รหัสการทำงาน	VARCHAR2	2	PK	
ActionName	ชื่อการทำงาน	VARCHAR2	50		

ตารางที่ 7.26 ข้อมูลหลักบริการ (Service)

Table Name: Service					
Description: ข้อมูลหลักบริการ					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
ServiceID	รหัสบริการ	VARCHAR2	2	PK	
ServiceName	ชื่อบริการ	VARCHAR2	50		

ตารางที่ 7.27 ข้อมูลหลักอัตราดอกเบี้ย (InterestRate)

Table Name: InterestRate					
Description: ข้อมูลหลักอัตราดอกเบี้ย					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
InterestRateID	รหัสอัตราดอกเบี้ย	VARCHAR2	2	PK	
InterestRateName	ชื่ออัตราดอกเบี้ย	VARCHAR2	50		

InterestRate	อัตราดอกเบี้ย	NUMBER	2,2		
InterestRateGroupID	รหัสกลุ่มอัตราดอกเบี้ย	VARCHAR2	2	FK	InterestRateGroup

ตารางที่ 7.28 ข้อมูลหลักกลุ่มอัตราดอกเบี้ย (InterestRateGroup)

Table Name: InterestRateGroup					
Description: ข้อมูลหลักกลุ่มอัตราดอกเบี้ย					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
InterestRateGroupID	รหัสกลุ่มอัตราแลกเปลี่ยน	VARCHAR2	2	PK	--
InterestRateGroupName	ชื่อกลุ่มอัตราแลกเปลี่ยน	VARCHAR2	50		

ตารางที่ 7.29 ข้อมูลหลักอัตราแลกเปลี่ยน (FXRate)

Table Name: FXRate					
Description: ข้อมูลหลักอัตราแลกเปลี่ยน					
Attribute Name	Content	Type	Length	Key	Foreign Key Referenced Table
FXRateID	รหัสอัตราแลกเปลี่ยน	VARCHAR2	2	PK	
FXRateName	ชื่ออัตราแลกเปลี่ยน	VARCHAR2	50		
FXRateBased	อัตราพื้นฐานเปรียบเทียบ	NUMBER	5,2		
FXRate	อัตราแลกเปลี่ยน	NUMBER	5,2		

บทที่ 8

บทสรุป

จากการศึกษาการวิเคราะห์และออกแบบระบบ BBL Portal โดยใช้ UML นั้น มีขั้นตอนการทำงานหลายขั้นตอน ตั้งแต่การกำหนดวัตถุประสงค์ของการพัฒนาระบบ การวิเคราะห์และการออกแบบระบบโดยใช้ CASE Tool ระบบจัดการฐานข้อมูล สามารถสรุปผลการศึกษาได้ดังต่อไปนี้

8.1 ข้อดีและข้อเสียของการใช้ CASE Tool Rational Rose

1) ข้อดี

- แสดงแบบจำลองระบบ ช่วยให้ผู้ใช้วิเคราะห์ระบบได้เห็นแบบจำลองของระบบที่ได้ทำการวิเคราะห์และออกแบบได้ทันที ทำให้สามารถเห็นจุดดี จุดไม่ดีของระบบนั้นในแง่มุมต่าง ๆ ตามแต่ลักษณะของแต่ละ Diagram ที่สร้างขึ้น และสามารถปรับปรุงแก้ไข หรือเพิ่มเติมได้ทันที และสามารถนำแบบจำลองที่สร้างขึ้นนั้น ไปใช้เป็นส่วนกลางในการนำเสนอผลการวิเคราะห์และออกแบบระบบต่อผู้ใช้งานและผู้พัฒนาระบบ ให้เข้าใจถึงความคิด ความต้องการของผู้วิเคราะห์ระบบได้อย่างมีประสิทธิภาพ

- ตรวจสอบความถูกต้อง มีการทำงานที่สำคัญอีกเรื่องหนึ่งคือ ช่วยในการตรวจสอบแบบจำลองที่ผู้ใช้วิเคราะห์ระบบได้จัดทำขึ้น เพื่อทราบถึงความถูกต้องของแต่ละ Diagram รวมถึงความสัมพันธ์ของแต่ละวัตถุในแต่ละ Diagram ของแบบจำลองว่าจะสามารถนำไปใช้ในการพัฒนาระบบจริงได้หรือไม่ ช่วยให้ผู้ใช้วิเคราะห์ระบบทราบถึงความเป็นไปได้ในการพัฒนาของระบบที่ได้ทำการวิเคราะห์และออกแบบ และปรับปรุงได้อย่างทันทั่วทั้งที่

- จัดทำโครงสร้างระบบ ช่วยในการสร้างโครงสร้างของระบบ โดยมีการ Generate Source Code เพื่อเป็นโครงสร้างของระบบ ที่จะนำไปพัฒนาในส่วนรายละเอียดของระบบให้เป็นไปในแนวทางเดียวกัน รวมถึง DDL ที่ใช้ในการสร้าง Database Schema ที่ใช้ประกอบระบบ

- บำรุงรักษาระบบ หลังจากได้มีการวิเคราะห์และออกแบบระบบจนสมบูรณ์แล้วนั้น เมื่อมีการนำไปพัฒนาระบบจริง อาจจะยังมีข้อบกพร่อง หรือปัญหาที่คาดไม่ถึง ทำให้ต้องมีการปรับเปลี่ยนโครงสร้างของระบบในการพัฒนานั้น เพื่อเป็นการบำรุงรักษาระบบให้มีความถูกต้องและทันสมัยอยู่เสมอ จึงควรมีการ Reverse Engineer ระบบที่ได้มีการพัฒนานั้น กลับมาปรับปรุงแบบจำลองอีกครั้ง เพื่อใช้เป็นส่วนกลางในการพัฒนาระบบต่อไป

- เอกสารประกอบระบบ ช่วยในการสร้างเอกสารประกอบระบบจากที่ได้มีการใส่ข้อมูลอ้างอิงไว้ในแบบจำลอง เพื่อใช้เป็นเอกสารประกอบระบบต่อไป
- มาตรฐาน UML ช่วยให้นักวิเคราะห์ระบบสร้างแบบจำลองได้ถูกต้องตามมาตรฐาน UML

2). ข้อเสีย

- ขั้นตอนการพัฒนากระบวน อาจทำให้ขั้นตอนของการวิเคราะห์ระบบถูกลดความสำคัญลง เนื่องจากนักวิเคราะห์ระบบสามารถใช้ Tool ในการสร้างแบบจำลองระบบขึ้นมาได้ทันที และสามารถปรับปรุงเปลี่ยนแปลงได้ตามต้องการ โดยไม่จำเป็นต้องมีการวิเคราะห์ระบบอย่างสมบูรณ์นัก
- พัฒนาการของผู้วิเคราะห์ระบบ จากการที่มีการทำงานที่ช่วยอำนวยความสะดวกให้กับนักวิเคราะห์ระบบอย่างมากมาย ซึ่งช่วยให้ทำงานง่ายและรวดเร็วขึ้น รวมถึงคุณภาพของระบบที่วิเคราะห์ออกแบบจะค่อนข้างสมบูรณ์และถูกต้องตามมาตรฐานมากขึ้น แต่ในทางกลับกันอาจจะทำให้นักวิเคราะห์ระบบละเลยในส่วนของรายละเอียดซึ่งช่วยเสริมสร้างพัฒนาการในการวิเคราะห์และออกแบบ เนื่องจากยึดติดกับ Tool มากเกินไป และความละเอียดรอบคอบในการวิเคราะห์ระบบอาจจะถูกมองข้ามไป เนื่องจากสามารถตรวจสอบและแก้ไขได้ทันทีจากความสามารถของ Tool
- การนำไปใช้ร่วมกับ CASE Tool อื่น มีความสามารถค่อนข้างจำกัดในการนำไปใช้ร่วมกับ Tool อื่น เนื่องจากแบบจำลองที่สร้างขึ้น จะมีข้อมูลและลักษณะเฉพาะที่ถูกกำหนดโดย Tool ทำให้ Tool อื่นไม่สามารถที่จะเข้าใจ และนำไปใช้ต่อได้ ต้องใช้กับ Tool นี้เท่านั้น
- เครื่องมือและอุปกรณ์ที่ใช้ประกอบซอฟต์แวร์ จำเป็นที่ต้องใช้อุปกรณ์ที่มีประสิทธิภาพค่อนข้างสูงประกอบการใช้ซอฟต์แวร์ เนื่องจากเป็นซอฟต์แวร์ที่มีความสามารถสูง และมีการประมวลผลที่ซับซ้อน

8.2 สรุปผลการวิเคราะห์และออกแบบระบบงานโดยใช้ CASE Tool

1) ในการทำงานโดยใช้ CASE Tool ช่วยทำให้มีเครื่องมือช่วยในการสร้างเอกสารของระบบ ซึ่งทำให้การค้นหาเอกสารของการวิเคราะห์และออกแบบระบบทำได้ง่าย

2). CASE Tool Rational Rose 2000 อำนวยความสะดวกสำหรับผู้ที่ต้องเข้ามาศึกษาระบบใหม่ โดยไม่จำเป็นต้องรู้รายละเอียดของระบบงานมาก่อน และสามารถศึกษาระบบเดิมเพื่อพัฒนาต่อไปได้ โดยเริ่มจากการศึกษา Use Case Diagram ของระบบ ซึ่งทำให้มองเห็นภาพรวมของระบบได้ง่าย

ขึ้น
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) การออกแบบโดยใช้ Rational Rose 2000 ซึ่งมีเครื่องมือช่วยในการ Generate Code จาก Class Diagram เป็นภาษาต่าง ๆ ได้โดยไม่ต้องเริ่มเขียนโปรแกรมใหม่ ซึ่งทำให้ใช้เวลาการพัฒนาโปรแกรมสั้นลง

4) การพัฒนาระบบโดยใช้เทคโนโลยีเชิงวัตถุนี้ ได้นำทฤษฎีที่ศึกษามาใช้ในการพัฒนาระบบงานตามขั้นตอนต่าง ๆ ช่วยให้การศึกษาก้าวไปอย่างมีขั้นตอนและปรับปรุงขั้นตอนการทำงานในส่วนต่าง ๆ ที่เกี่ยวข้องกันได้ง่าย และสมบูรณ์ขึ้น

8.3 ประโยชน์ที่ได้รับ

ระบบ BBL Portal เป็นระบบที่ค่อนข้างมีความซับซ้อน และมีส่วนการทำงานที่หลากหลายรวมอยู่ด้วยกันเพื่อจัดการกับการทำงานในด้านต่าง ๆ จึงจำเป็นต้องมีความเข้าใจในระบบเป็นอย่างดี เพื่อให้การดำเนินการพัฒนาระบบเป็นไปอย่างมีประสิทธิภาพ และจากการศึกษาทำให้ทราบถึงวิธีการทำงานของระบบ BBL Portal ที่มีทั้งจุดเด่นและจุดด้อยรวมกันอยู่ แล้วนำจุดด้อยและปัญหาต่าง ๆ เหล่านั้นมาปรับปรุงโดยการวิเคราะห์และออกแบบระบบเพื่อขจัดหรือบรรเทาให้เจือจางลง นอกจากนั้นยังได้มีโอกาสเรียนรู้การใช้งาน และการทำงานของ CASE Tool Rational Rose 2000 ซึ่งใช้ในการวิเคราะห์และออกแบบระบบนี้ด้วย

8.4 ข้อเสนอแนะ

ในการนำเทคโนโลยีเชิงวัตถุมาใช้นั้น ยังเป็นสิ่งที่ไม่คุ้นเคยกันเท่าที่ควรสำหรับนักพัฒนาระบบโดยทั่วไป ดังนั้น หากนักพัฒนาระบบที่ประสบปัญหาการออกแบบระบบบางอย่าง อันเนื่องจากหลาย ๆ สาเหตุ เช่น การเปลี่ยนแปลงความต้องการของผู้ใช้ที่มีอยู่ตลอดเวลา การเขียนโปรแกรมแล้ว แต่ไม่สามารถนำไปใช้กับงานที่คล้ายคลึงกันได้ ต้องทำการเขียนโปรแกรมใหม่ทั้งหมดอยู่เสมอ เป็นต้น คุณสมบัติของเทคโนโลยีเชิงวัตถุจึงน่าจะมีประโยชน์ และน่าจะเป็นอีกตัวเลือกหนึ่งที่เหมาะสมในการนำไปพัฒนาและปรับปรุงระบบที่มีอยู่เดิมให้มีประสิทธิภาพยิ่งขึ้น

บรรณานุกรม

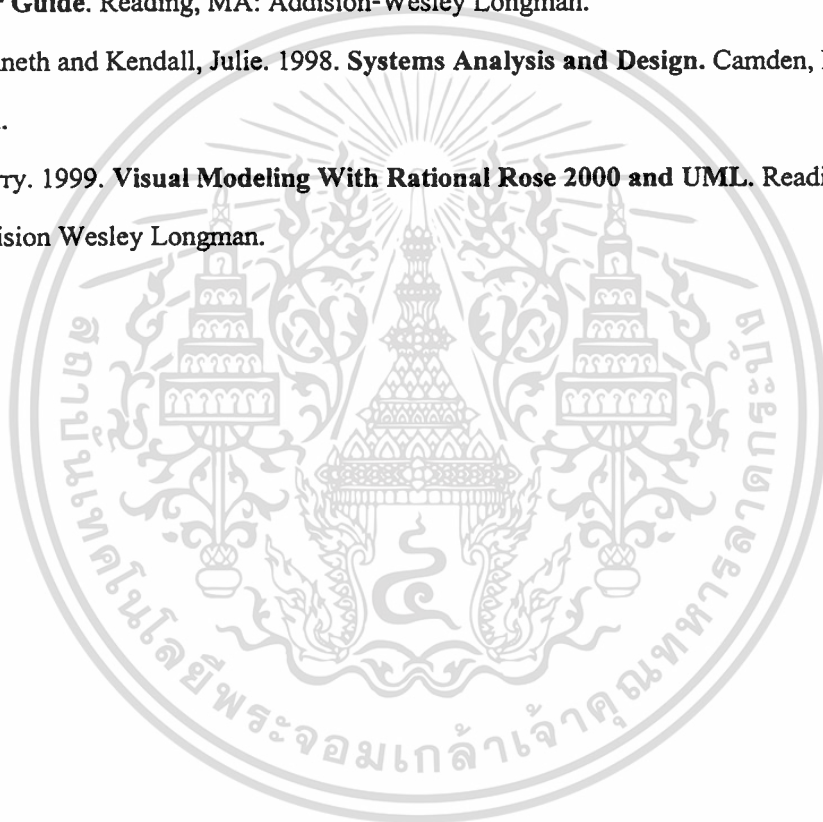
กิตติ ภักดีวัฒนกุล และ กิตติพงษ์ กลมกล่อม. 2544. UML วิเคราะห์และออกแบบระบบเชิงวัตถุ.
กรุงเทพฯ: ไทยเจริญการพิมพ์.

Bahrami, Ali. 1999. **Object Oriented Systems Development**. Singapore: Irwin/McGraw-Hill.

Booch, Grady, Jacobson, Ivar and Rumbaugh, James. 1998. **The Unified Modeling Language User Guide**. Reading, MA: Addison-Wesley Longman.

Kendall, Kenneth and Kendall, Julie. 1998. **Systems Analysis and Design**. Camden, NJ: Prentice-Hall.

Quatrani, Terry. 1999. **Visual Modeling With Rational Rose 2000 and UML**. Reading, MA: Addison Wesley Longman.



ประวัติผู้เขียน

ชื่อผู้เขียน	นายพรศักดิ์ สุนทรมาตย์
สถานที่เกิด	กรุงเทพมหานคร
วุฒิการศึกษา	บริหารธุรกิจบัณฑิต คณะบริหารธุรกิจ สาขาวิชาระบบสารสนเทศ สถาบันเทคโนโลยีราชมงคล
เม.ย. 2540 – มี.ย. 2543	Programmer บริษัทเครือเจริญโภคภัณฑ์อาหารสัตว์ จำกัด (มหาชน)
มี.ย. 2543 – มี.ก. 2545	Senior Programmer บริษัทเอชเอ็นเคียร์คอทคอม จำกัด
เม.ย. 2545 – ต.ค. 2545	Application Developer (IBM: Sub-Contractor) บริษัทเซฟตี้ฟอร์ช จำกัด
พ.ย. 2545 – ปัจจุบัน	System Analyst บริษัท ธนาคารกรุงเทพ จำกัด (มหาชน)