

ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล.

การพัฒนาพาณิชย์อิเล็กทรอนิกส์สมัยใหม่

ด้วยการใช้ Windows Web Solution

New E-commerce Development Using Windows Web Solution

โดย

นายถนอมจิตต์ เกียรติขจรวิ

รหัส 43067172

อาจารย์ที่ปรึกษา

ดร.โชติพัชร ภรณ์วลัย



H001900

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

ภาคเรียนที่ 1 ปีการศึกษา 2545

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

วัน เดือน ปี.....	19 ส.ค. 2550
เลขทะเบียน.....	01900
เลขเรียกหนังสือ.....	วิท ๓14๖๓ 2545
"ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล."	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในห้องสมุดเท่านั้น ไม่ให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดก็ตาม ห้ามนำไปเผยแพร่หรือทำซ้ำโดยไม่ได้รับอนุญาต และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ	การพัฒนาพาณิชย์อิเล็กทรอนิกส์สมัยใหม่ด้วยการใช้ Windows Web Solution
นักศึกษา	นายถนอมจิตต์ เกียรติขวันะเสวี
อาจารย์ที่ปรึกษา	ผศ.ดร.โชติพัทธ์ ภรณวลัย
ระดับการศึกษา	วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2545

บทคัดย่อ

ในปัจจุบัน ระบบพาณิชย์อิเล็กทรอนิกส์ที่มีใช้กันอย่างแพร่หลายนั้น ยังคงพบกับปัญหาต่างๆ ซึ่งทำให้เกิดความยุ่งยากแก่ผู้พัฒนาระบบ และความสูญเสียทั้งเวลาและค่าใช้จ่ายกับเจ้าของระบบ ดังนั้น การพัฒนาระบบพาณิชย์อิเล็กทรอนิกส์สมัยใหม่ ที่ช่วยจัดการปัญหาเหล่านี้ได้อย่างสะดวกรวดเร็วและประหยัดค่าใช้จ่าย จึงเป็นประโยชน์อย่างมากต่อผู้พัฒนาระบบ เจ้าของกิจการ และธุรกิจ

ในการพัฒนาระบบพาณิชย์อิเล็กทรอนิกส์สมัยใหม่ เราได้มีการนำเทคโนโลยี COM+ และ XML มาใช้ร่วมกับ Windows Web Solution ในการพัฒนาระบบ โดยระบบที่เราจะพัฒนาขึ้นมาเป็นแอปพลิเคชันเว็บแบบ 3-Tier ซึ่งประกอบไปด้วย ฐานข้อมูล COM+ Service (เราจะมีการสร้างคลาสขึ้นมาเก็บไว้ที่ตัว COM+ Application เพื่อความสะดวกในการบำรุงรักษาระบบ และสามารถขยายระบบต่อไปได้ในอนาคต ซึ่งการแก้ปัญหาก็ทุกอย่างจะแก้ไขที่คลาส ทำให้เป็นการง่ายแก่ผู้พัฒนาระบบ) และ Web Application ซึ่งระบบที่เราพัฒนานี้จะสามารถจัดการกับปัญหาต่างๆ ซึ่งเป็นประโยชน์อย่างมากต่อผู้พัฒนาระบบ เจ้าของกิจการและธุรกิจนั้นเป็นอย่างมาก

Title	New E-commerce Development Using Windows Web Solution
Student	Mr. Thanomjit Kiatchavanasawee
Advisor	Asst. Prof. Dr. Chotipat Pornnavalai
Level of Study	Master of Science in Information Technology
Major	Information Science
Academic year	2002

ABSTRACT

Nowadays, the mostly E-commerce system still face of many problems , it lost of time and lost of money investment of the owner of system. So the new invention of E-commerce that can solve these problems. It decreases time and expenses. It is a lot of helpful for developer , the owner and businesses.

The New Development of E-commerce system ,we use the COM+ and XML Technology combine with Windows Web Solution in Development. The Structure of this Web Application use 3- Tier structure. It consist of Database ,COM+ Service (we create Classes and store in COM+ Service ,it is convenient in maintenance and Support future expansion) and Web Application. This System will eliminate all problems, that is a lot of useful for developer, owner and businesses.

กิตติกรรมประกาศ

ขอขอบพระคุณ บิดา มารดา และพี่ชาย ของข้าพเจ้า ที่คอยเป็นกำลังใจ ในการส่งเสริม ชี
ณะแนวทาง การศึกษาการเรียนรู้ รวมทั้งอาจารย์ทุกท่านที่ได้ถ่ายทอดความรู้ เพื่อให้เป็นพื้นฐานใน
การเรียนรู้สิ่งใหม่ต่อไปในอนาคต

ขอขอบพระคุณอาจารย์ที่ปรึกษา ดร. โชติพัทธ์ ภรณ์เวทย์ ซึ่งช่วยในการตรวจอ่านและตั้ง
คำถาม เพื่อให้ข้าพเจ้าได้นำมาปรับปรุงเนื้อหา และรูปแบบการนำเสนอให้สมบูรณ์ยิ่งขึ้น

ข้าพเจ้าขอขอบคุณ นายสามารถ ธีระศิริ กับเพื่อนร่วมงานบริษัท J.V System Co., Ltd และ
ผู้เกี่ยวข้องทุกท่าน ที่ได้คำปรึกษาแนวทางการแก้ไขปัญหาต่างๆ ในการพัฒนาระบบในครั้งนี้

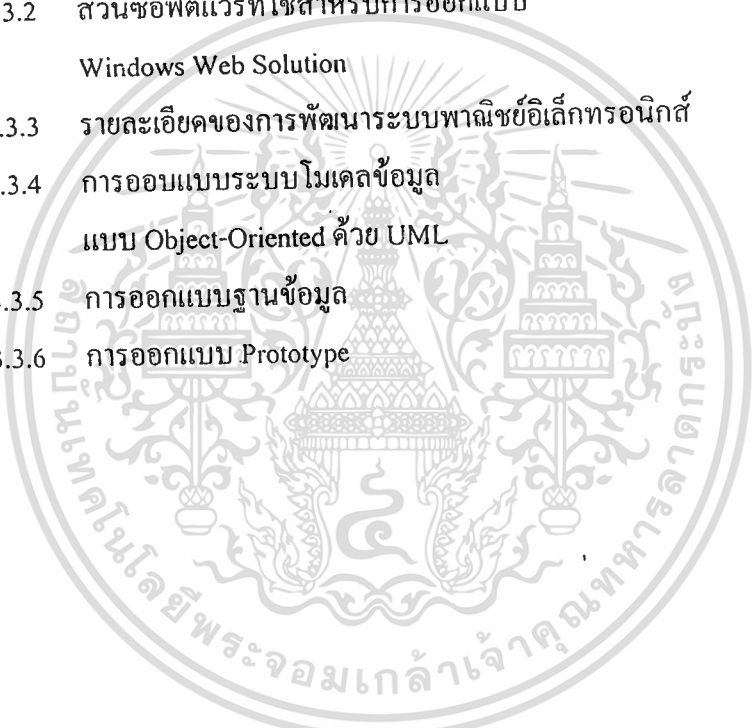
สุดท้ายหากรายงานฉบับนี้มีความผิดพลาดประการใด ต้องขออภัยไว้ ณ ที่นี้ด้วย

นายธนอมจิตต์ เกียรติชวณะเสวี

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VI
สารบัญภาพ	VI
บทที่	
1. บทนำ	1
1.1 ที่มา	1
1.2 เป้าหมายของโครงการ	2
1.3 วัตถุประสงค์ของโครงการ	2
1.4 ขอบเขตของการศึกษา	3
1.5 ขั้นตอนการดำเนินงาน	3
1.6 ประโยชน์ที่คาดว่าจะได้รับ	4
2. ความรู้พื้นฐานเกี่ยวกับ Windows Web Solution กับการประยุกต์การใช้งานเทคโนโลยี COM+	5
2.1 ความเป็นมาของการพัฒนาเว็บแอปพลิเคชัน	5
2.2 การนำเทคโนโลยี Windows web solution มาแก้ไขปัญหาระบบพาณิชย์อิเล็กทรอนิกส์	7
2.3 การศึกษาโครงสร้างการเชื่อมต่อระบบการทำงานของ Windows Web Solution แบบ N-Tier	12
2.4 การศึกษาการทำงานของเทคโนโลยี COM+ ร่วมกับ Windows web Solution	14
2.5 การประยุกต์การใช้งานของเทคโนโลยี COM+ พร้อมการติดตั้ง และการทดสอบการทำงาน	16

3. การพัฒนาระบบพาณิชย์อิเล็กทรอนิกส์สมัยใหม่	22
3.1 ขั้นที่ 1 : Problem Statement	22
3.2 ขั้นที่ 2 : Gathering Requirement	23
3.3 ขั้นที่ 3 : Design	23
3.3.1 การออกแบบสถาปัตยกรรมแบบ N-Tire ของ Windows Web Solution	24
3.3.2 ส่วนซอฟต์แวร์ที่ใช้สำหรับการออกแบบ Windows Web Solution	25
3.3.3 รายละเอียดของการพัฒนาระบบพาณิชย์อิเล็กทรอนิกส์	27
3.3.4 การออกแบบระบบ โมเดลข้อมูล แบบ Object-Oriented ด้วย UML	28
3.3.5 การออกแบบฐานข้อมูล	46
3.3.6 การออกแบบ Prototype	49
4. สรุป	59
บรรณานุกรม	60
ภาคผนวก ก	61
ภาคผนวก ข	68
ประวัติผู้เขียน	73



สารบัญตาราง

ตารางที่

1. ตาราง Venders	47
2. ตาราง Branders	47
3. ตาราง Products	48
4. ตาราง ProductsBranders	48
5. ตาราง Carts	48
6. ตาราง CartsDetails	48
7. ตาราง Orders	49



สารบัญภาพ

ภาพที่	หน้า
1. การใช้ Active Server Pages (ASPs) แบบดั้งเดิม	5
2. การใช้ Active Server Pages (ASPs) ร่วมกับ COM ซึ่งก็คือ A และ B	6
3. การใช้ Active Server Pages (ASPs) ในการเข้าถึงแอปพลิเคชัน COM+	6
4. ตัวอย่างโมเดลการทำงานแบบ 3-Tier	8
5. โครงสร้างพื้นฐานของระบบสถาปัตยกรรมแบบ 3-Tier	9
6. สถาปัตยกรรมของ Windows DNA	11
7. สถาปัตยกรรมของ Universal Data Access	12
8. รูปแบบโมเดลของ ADO 2.5	14
9. COM+ Authorization	15
10. การสร้าง COM+ Application	18
11. Create an Empty Application	18
12. การติดตั้ง Component Object Model (COM)	19
13. การติดตั้ง COM จากไฟล์ DLL	19
14. COM ที่ได้ติดตั้งสำเร็จ	20
15. โปรแกรม Active Directory Security	20
16. การทำงานของ COM+ Application	21
17. องค์ประกอบของการพัฒนาแอปพลิเคชันแบบ 3-Tier หรือ Distributed Application	24
18. ซอฟต์แวร์พื้นฐานในการพัฒนาระบบพาณิชย์อิเล็กทรอนิกส์	25
19. Use Case Diagram	28
20. Class Diagram	29
21. State Diagram ของ Login System	32
22. State Diagram ของ Add Product To Cart	33
23. State Diagram ของ View Cart	34
24. State Diagram ของ Check Out	35

25. Activity Diagram ของ Login System	36
26. Activity Diagram ของ Add Product To Cart	37
27. Activity Diagram ของ View Cart	38
28. Activity Diagram ของ Check Out	39
29. Sequence Diagram ของ Login	40
30. Sequence Diagram ของ Cookie	41
31. Sequence Diagram ของ Search Product	42
32. Sequence Diagram ของ Add Product	43
33. Sequence Diagram ของ Add Quantity	44
34. Sequence Diagram ของ Comfirm Order	45
35. ตัวอย่างการออกแบบฐานข้อมูล	46
36. Site Diagram ของเว็บไซต์ในระบบพาณิชย์อิเล็กทรอนิกส์	49
37. หน้าจอหน้าแรกของเว็บไซต์	50
38. หน้าจอแสดงผลลัพธ์จากการค้นหาข้อมูล	51
39. หน้าจอต้อนรับเพื่อให้ทำการ Login	52
40. หน้าจอรับอินพุต ของการ Login	53
41. หน้าจอแสดงผลการตอบรับหลังจาก Login สำเร็จ	54
42. หน้าจอแสดงรายละเอียดของสินค้า	55
43. หน้าจอแสดงข้อมูลสินค้าที่อยู่ในรถเข็น	56
44. หน้าจอแสดงที่อยู่ของลูกค้า	57
45. หน้าจอแสดงการสิ้นสุดการสั่งซื้อสินค้า	58
46. Component Service ของ DNAECommerce	62
47. การ Run Script ใน SQL Server 2000	63
48. ข้อมูลที่ต้องนำไปติดตั้งใน IIS 5.0	64
49. การเซตค่า Construction String ของ DNABizServ	65
50. การเซตค่า Construction String ของ DNADataServ ที่ติดต่อกับ SQL Server	66
51. การเซตค่า Construction String ของ DNADataServ ที่ติดต่อกับ Active Directory	67

52. การเลือก Export ตัว Server Application	68
53. การสร้าง Server Application	69
54. การเลือก Export ตัว Proxy Application	70
55. การสร้าง Proxy Application	71
56. หลักการในการเชื่อมต่อระบบในการแยกเป็น N-Tier	72



บทที่ 1

บทนำ

1.1 ที่มา

เนื่องจากปัจจุบันเทคโนโลยีระบบเครือข่ายอินเทอร์เน็ต ทำให้ระบบสื่อสารระหว่างองค์กรในหลายด้านเปลี่ยนแปลงไป โดยมีการใช้ระบบคอมพิวเตอร์ในการติดต่อสื่อสารแลกเปลี่ยนกัน หรือมีการดำเนินการร่วมกันในลักษณะคู่ค้าหรือดำเนินธุรกิจแบบพึ่งพาซึ่งกันและกัน เพื่อเพิ่มขีดความสามารถในการทำงาน และลดการดูแลหรือการติดตามการทำงาน โดยมอบหน้าที่ให้ระบบไล่ติดตามงานและดำเนินการบางอย่างแทน

ดังนั้นในการบริหารงานสมัยใหม่ที่มีเทคโนโลยีไอทีเป็นแกนนำจึงมิได้สนใจเฉพาะตนเองหรือองค์กรของตนเองเท่านั้น แต่จะต้องเชื่อมโยงออกไปยังนอกองค์กร มีการสร้างกิจกรรมร่วมกันต่างๆ มากมาย ทำให้ผู้บริหารสมัยใหม่ต้องสร้างแนวคิดในการดำเนินการที่กว้างไกลออกไป และสร้างกิจกรรมต่างๆ ร่วมกับองค์กรอื่น เช่นการสร้างพันธมิตร การสร้างการดำเนินธุรกิจร่วมในลักษณะคู่ค้า เป็นต้น

โดยได้มีการนำบริการบนเว็บมาใช้เพื่อการบริการ ทำให้มีการนำไปประยุกต์ใช้งานในรูปแบบต่างๆ อีกมากมาย โดยเฉพาะการดำเนินการภายในองค์กรแบบอิเล็กทรอนิกส์ และการทำงานระหว่างองค์กร เช่นควบคุมสินค้าคงคลังระหว่างหน่วยงาน การจัดซื้อแบบอิเล็กทรอนิกส์ ซึ่งเราจะเห็นได้อย่างชัดเจนว่า การเชื่อมโยงข้อมูลข่าวสารระหว่างหน่วยงาน เพื่อเข้าสู่การบริการผ่านเว็บในปัจจุบัน จะใช้ HTML อย่างเดียวไม่ได้ การจัดการข้อมูลและวิธีการแลกเปลี่ยนข้อมูลจำเป็นต้องมีมาตรฐานการจัดการบริหารข้อมูลที่ดี เพื่อการแลกเปลี่ยนข่าวสารใหม่ และสิ่งที่กำลังจะมีบทบาทที่สำคัญอย่างใหญ่หลวงคือ XML (eXtensible Mark up Language) ทั้งนี้เพราะ XML นำไปใช้บนเครือข่ายอินเทอร์เน็ตได้ง่าย รวดเร็ว รวมทั้งสนับสนุนการนำไปประยุกต์ใช้งานได้มาก เช่น การวิเคราะห์ข้อมูล การแลกเปลี่ยนข้อมูล ตลอดจนการสร้างมาตรฐานข้อมูล ดังนั้น XML จึงเป็นระบบที่พัฒนาให้ใช้งานบนเว็บได้ดี การเขียนโปรแกรมประมวลผลได้ง่าย ซึ่งเป็นคุณสมบัติของภาษาที่อยู่ในระดับต่ำมี Text เป็นรูปแบบมาตรฐาน ทำให้อ่านเข้าใจด้วยมนุษย์ และเป็นภาษาที่ออกแบบมาให้เข้ากับหลักวิชาการทางวิศวกรรมคอมพิวเตอร์

จากการศึกษาแนวโน้มของการพัฒนาระบบแอปพลิเคชันของเว็บ ที่ผ่านมามีตั้งแต่อดีตจนถึงปัจจุบัน จะเห็นได้ว่าการพัฒนาในเชิงที่ก้าวไปสู่ระบบที่เป็น Distributed มากขึ้น โดยมีแนวโน้มที่จะมีการเชื่อมโยงระบบของเว็บแอปพลิเคชันให้สามารถทำงานร่วมกับระบบงานพื้นฐานที่มีอยู่เดิมอื่นๆ ได้ และสามารถที่จะเชื่อมโยงระบบต่อไปได้ในอนาคต ดังนั้นในการศึกษา และพัฒนาระบบพาณิชย์อิเล็กทรอนิกส์สมัยใหม่ นี้ จึงได้มีการเน้นการศึกษาค้นคว้ารูปแบบสถาปัตยกรรมแบบ N-Tier และการออกแบบระบบแบบ Object-Oriented เพื่อเป็นแนวทางพื้นฐานในการนำเทคโนโลยีที่ได้รับการยอมรับในปัจจุบันมาใช้งาน และนำมาประยุกต์ใช้งานให้มีประสิทธิภาพ เหมาะสมกับระบบพาณิชย์อิเล็กทรอนิกส์ปัจจุบันที่จะต้องมีการพัฒนาต่อไปในอนาคต

1.2 เป้าหมายของโครงการงาน

- 1.2.1 เพื่อเป็นแนวทางการศึกษา การนำระบบ Windows Web Solution มาใช้งาน (ว่าต้องใช้อะไรบ้าง)
- 1.2.2 เพื่อเป็นการเรียนรู้ในการประยุกต์เอาเทคโนโลยีพื้นฐานของ Web Services มาใช้งาน
- 1.2.3 เพื่อฝึกฝนการสร้างซอฟต์แวร์เว็บแอปพลิเคชัน การออกแบบระบบแบบ Object-Oriented และการวิเคราะห์ ในการพัฒนาระบบการทำงานของเว็บแอปพลิเคชัน ให้มีประสิทธิภาพ
- 1.2.4 เพื่อเป็นแนวทางในการออกแบบระบบแบบ N-Tier และนำไปประยุกต์พัฒนาระบบพาณิชย์อิเล็กทรอนิกส์ต่อไป
- 1.2.5 เพื่อเป็นการเสริมสร้างพื้นฐานการเรียนรู้เทคโนโลยีใหม่ๆ เพื่อใช้สำหรับการพัฒนาระบบที่กำลังจะเกิดในอนาคต

1.3 วัตถุประสงค์ของโครงการงาน

- 1.3.1 เพื่อทดสอบนำเอาเทคโนโลยีที่เป็นมาตรฐานที่มีอยู่ใน Windows 2000 Server ที่เรียกว่า COM+ มาใช้งานในการจัดการทรานแซกชัน เพื่อส่งข้อมูลที่เราต้องการผ่านเครือข่ายด้วยเทคโนโลยีของ XML
- 1.3.2 เพื่อทดสอบนำเอาเทคโนโลยีในการรักษาความปลอดภัยของ Active Directory ใน Windows 2000 Server มาใช้งานร่วมกับเทคโนโลยีของเว็บ
- 1.3.3 เพื่อเป็นแนวทางเบื้องต้นในการพัฒนาระบบพาณิชย์อิเล็กทรอนิกส์สมัยใหม่ ให้มีความน่าเชื่อถือ มีความปลอดภัย และสามารถลด หรือ ขยายระบบให้เหมาะสมกับการเติบโตของธุรกิจที่เป็นอยู่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ขอบเขตของการศึกษา

- 1.4.1 ศึกษารูปแบบการพัฒนาระบบแบบ N-Tier ซึ่งช่วยให้การบำรุงรักษาระบบทำได้ง่ายขึ้น (เช่น จากเดิมที่มีการแก้ไขคำสั่ง SQL ใน ASP pages มาแก้ไขที่ Store Procedure แทน หรือ การใช้เวลาในการทำงานในการพัฒนาระบบน้อยลง เป็นต้น)
- 1.4.2 ศึกษาระบบพาณิชย์อิเล็กทรอนิกส์ที่มีการใช้งานอยู่ในปัจจุบัน มาวิเคราะห์และประยุกต์ใช้งานกับระบบพาณิชย์อิเล็กทรอนิกส์สมัยใหม่ที่ได้นำเอาเทคโนโลยี COM+ และ XML มาใช้งาน

เนื่องจากโครงการนี้เน้นการศึกษาสถาปัตยกรรมแบบ N-Tier เพื่อนำมาเป็นแนวทางในการประยุกต์ใช้งานกับระบบพาณิชย์อิเล็กทรอนิกส์ ดังนั้น การนำเสนอรูปแบบของระบบพาณิชย์สมัยใหม่ที่จะนำเสนอในโครงการนี้ จะเป็นโครงสร้างหลักๆ ทัวไปที่ระบบพาณิชย์อิเล็กทรอนิกส์มีการใช้งานกัน ซึ่งก็คือ การค้นหาสินค้า และการรับคำสั่งซื้อสินค้าลงในตะกร้าสินค้า โดยจะมีการนำเสนอในรูปแบบที่ง่ายที่สุด เพื่อเปิดช่องให้ผู้สนใจการพัฒนาแบบ N-Tier สามารถนำไปพัฒนาได้ต่อไป และสามารถออกแบบระบบฐานข้อมูลที่ดีกว่า หรือสามารถเพิ่มเติมระบบเชื่อมต่ออื่นๆ เข้ามาที่ระบบพาณิชย์อิเล็กทรอนิกส์สมัยใหม่ได้ต่อไปในอนาคต

1.5 ขั้นตอนการดำเนินงาน

- 1.5.1 ศึกษาสถาปัตยกรรมการพัฒนาแบบ N-tier เพื่อให้รู้ถึงหลักการทำงาน คุณสมบัติ และมาตรฐานโดยทั่วไป
- 1.5.2 นำเอาสิ่งที่ได้ศึกษาจาก ข้อ 1.5.1 มาวิเคราะห์แนวทางประยุกต์การสร้างระบบพาณิชย์อิเล็กทรอนิกส์สมัยใหม่ให้ใช้งานได้
- 1.5.3 ออกแบบระบบพาณิชย์อิเล็กทรอนิกส์สมัยใหม่ สำหรับเป็นแนวทางในการพัฒนา และประยุกต์งานต่อไป
- 1.5.4 จัดทำโปรแกรมระบบพาณิชย์อิเล็กทรอนิกส์ ตามหลัก Software Development Life Cycle (SDLC) เพื่อเป็นแนวทางในการดำเนินการ ที่เป็นมาตรฐาน

1.6 ประโยชน์ที่คาดว่าจะได้รับ

- 1.6.1 เพื่อเพิ่มความรู้ ความเข้าใจ ในประสบการณ์ในการออกแบบระบบในรูปแบบ Object-Oriented และแนวทางในการพัฒนาเว็บแอปพลิเคชัน
- 1.6.2 เรียนรู้การประยุกต์เอาเทคโนโลยีใหม่ๆ มาใช้งาน ให้สามารถเพิ่มประสิทธิภาพการทำงานให้กับระบบพาณิชย์อิเล็กทรอนิกส์ที่จะพัฒนาขึ้นมาใหม่
- 1.6.3 เพื่อเป็นแนวทางในการพัฒนาระบบพาณิชย์อิเล็กทรอนิกส์ หรือ พัฒนาระบบอื่นๆ แก่ผู้ที่สนใจ ในการนำเอาการออกแบบสถาปัตยกรรมแบบ N-Tier ไปใช้งานต่อไป



บทที่ 2

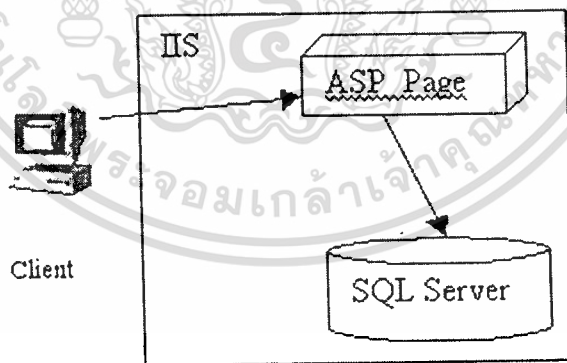
ความรู้พื้นฐานเกี่ยวกับ Windows Web Solution

2.1 ความเป็นมาของการพัฒนาเว็บแอปพลิเคชัน

ตั้งแต่ยุคเริ่มต้นของเทคโนโลยีเว็บนั้น เริ่มมาจาก ยุคที่เป็น Static Web ซึ่งมีการนำเสนอข้อมูลที่มีลักษณะตายตัว แบบ Static จนมาถึงยุคที่เป็น Dynamic Web ซึ่งมีการเพิ่มประสิทธิภาพการทำงานให้มีการประมวลผลและมีการเชื่อมต่อการทำงานกับฐานข้อมูลได้ โดยสามารถอธิบายการเปลี่ยนแปลงของเทคโนโลยีเว็บแบบ Dynamic ได้ดังนี้

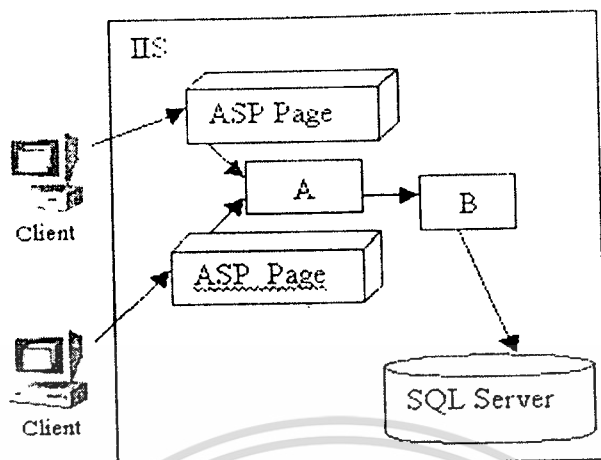
2.1.1 แอปพลิเคชันพื้นฐานกับลोजิกที่ใช้เขียนในเซิร์ฟเวอร์

โดยเริ่มต้นอย่างง่าย ๆ สมมติว่าเราต้องการสร้างแอปพลิเคชันที่มีการร้องขอมาจากเบราว์เซอร์ แล้วสามารถเข้าถึงฐานข้อมูลใน SQL Server ได้ และจะมีการส่งค่าการตอบสนองกลับ เราสามารถทำได้โดยการเขียน ASP Pages ซึ่งก็ถือว่าเป็นประสิทธิภาพที่น่ายอมรับได้ดังแสดงในภาพที่ 2.1 แต่ถ้าเราต้องการจะได้ประสิทธิภาพที่สูงที่สุด เราจะต้องทำการสร้าง ISAPI DLL



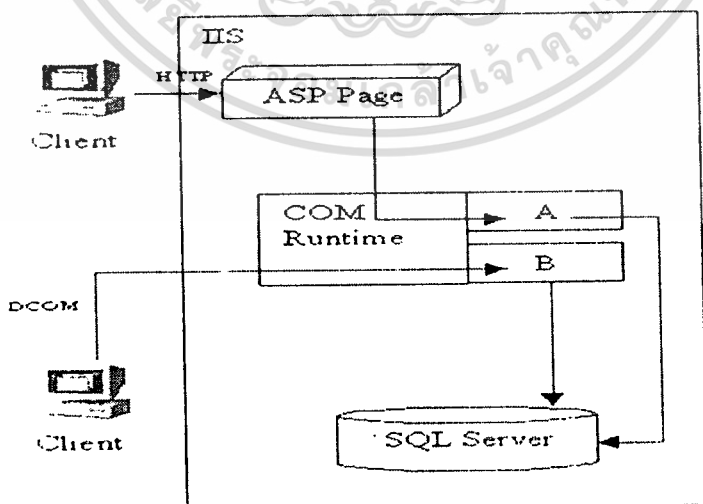
ภาพที่ 2.1 การใช้ Active Server Pages (ASPs) แบบดั้งเดิม

แต่ลोजิกที่ใช้เขียนในเซิร์ฟเวอร์นั้น ASP Pages ไม่มีความยืดหยุ่นที่เพียงพอ เพราะเราจะต้องทำการเขียน โปรแกรมทุกหน้า และมีความยากในการบำรุงรักษา หรือการพัฒนาต่อไป ดังนั้น เราจะใช้วิธีการแก้ปัญหา โดยการทำออปเจกต์ COM แล้วใช้ ASP Pages ทำการเรียกใช้ออปเจกต์ COM และสร้าง User Interface ดังแสดงในภาพที่ 2.2



ภาพที่ 2.2 การใช้ Active Server Pages (ASPs) ร่วมกับ COM ซึ่งก็คือ A และ B

ถ้าระบบที่เราใช้มีการทำงานที่เป็นขั้นตอนที่แน่นอนแบบตรงไปตรงมา การใช้ออปเจกต์ COM ก็จะเป็นการทำงานที่ดี แต่ถ้าแอปพลิเคชันมีความจำเป็นจะต้องมีการกระจายทรานแซกชัน หรือ มีการบริหารจัดการตามสถานะ เราจะใช้ COM+ ในการจัดการกับบริการเหล่านี้ โดยแอปพลิเคชันของ COM+ เป็นการใช้คอมโพเนนต์ COM ซึ่งใช้หน่วยความจำร่วมกับแอปพลิเคชันที่เรียกใช้งาน และในส่วนของไคลเอนต์เพียงร้องขอ เมธอด บน ออปเจกต์ COM แล้วแอปพลิเคชัน COM+ จะดำเนินการให้จาก COM Runtime ที่ซ่อนตัวอยู่ ดังแสดงในภาพที่ 2.3



ภาพที่ 2.3 การใช้ Active Server Pages (ASPs) ในการเข้าถึงแอปพลิเคชัน COM+

ส่วนการประยุกต์ใช้กับเว็บแอปพลิเคชันนั้น ทำให้เราสามารถเรียกใช้งาน เมทอด ที่เราสร้างเก็บไว้ใน Component Services ได้ และที่สำคัญทำให้ง่ายในการดูแลรักษาจากการที่เรามีการออกแบบสถาปัตยกรรมของระบบของเราเป็นแบบ N-Tier

สุดท้ายเป็นการใช้งานความสามารถที่มีในฐานข้อมูล SQL Server 2000 ซึ่งช่วยให้สะดวก รวดเร็วและง่ายในการใช้งาน ทำให้การแลกเปลี่ยนข้อมูลที่ใช้ XML ในการทำงานบนเว็บแอปพลิเคชันสะดวก

2.2 การนำเทคโนโลยี Windows Web solution มาแก้ไขปัญหาระบบพาณิชย์อิเล็กทรอนิกส์

เนื่องจากการพัฒนาระบบแอปพลิเคชันของเว็บในแบบดั้งเดิมนั้น มีลักษณะสถาปัตยกรรมแบบ Client-Server ซึ่งมีข้อจำกัดดังนี้

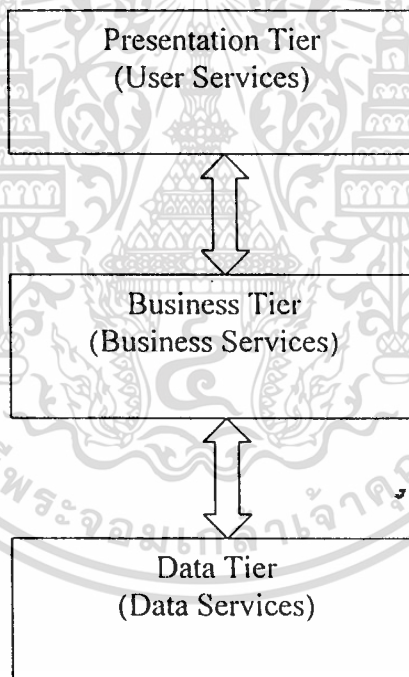
- จำนวนของผู้ใช้ ซึ่งต้องไม่เกินจำนวนที่กำหนดจากที่ได้เซตค่าเองไว้ เช่น จำนวนของผู้ใช้ในการเข้าใช้ฐานข้อมูล หรือผลจากการที่ผู้ใช้มีจำนวนเพิ่มมากขึ้น ส่งผลให้มีการใช้ทรัพยากรของเซิร์ฟเวอร์มากขึ้น ก็จะมีผลทำให้ประสิทธิภาพของระบบลดลง และอาจทำให้ระบบล่มเหลวได้
- ความน่าเชื่อถือ เนื่องจากถ้าหากว่าเซิร์ฟเวอร์มีการปิดเครื่อง อันเนื่องเหตุผลใดก็ตาม เช่น ครบกำหนดการบำรุงรักษาระบบ ก็จะทำให้เครื่อง Client ไม่สามารถเข้ามาใช้งานได้ในช่วงเวลานั้นๆ
- การบำรุงรักษาระบบ หากจำเป็นต้องมีการแก้ไขเปลี่ยนแปลงฟังก์ชันของระบบเพิ่มเติม ก็อาจเป็นการยากที่จะต้องกลับไปแก้ไขโปรแกรมในบางครั้ง

ซึ่งจากข้อจำกัดที่ได้กล่าวมาในบางครั้งก็เป็นสาเหตุให้เว็บไซต์บางที่ไม่สามารถเข้าไปใช้งานได้ อาจส่งผลทำให้บริษัทหรือองค์กรนั้นๆ ขาดรายได้หรือสูญเสียผลประโยชน์ได้ หากเกิดขึ้นบ่อยๆ

ดังนั้นการพัฒนาระบบในลักษณะที่เป็น N-Tier โดยมีนำเทคโนโลยี COM+ มาช่วยในการจัดการบริหารทรานแซกชัน และ XML มาช่วยเพิ่มประสิทธิภาพทางด้านความเร็วในการส่งถ่ายข้อ

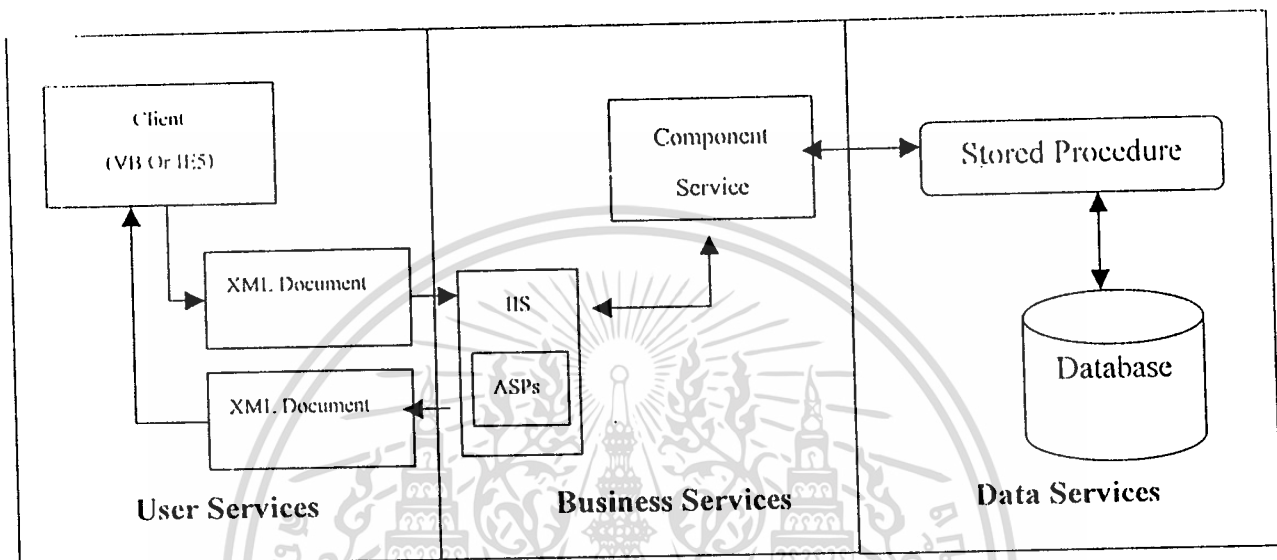
มูลผ่านเครือข่าย หรืออาจจะสามารถนำมาใช้ในการแลกเปลี่ยนข้อมูลระหว่างองค์กรหรือบริษัทก็ได้ และที่สำคัญการออกแบบระบบแบบ N-Tier นั้นช่วยให้การบำรุงรักษาระบบทำได้ง่าย

จากข้อจำกัดของระบบ 2-Tier ที่เป็นแบบ Client-Server ไมโครซอฟท์ได้มีการนำเสนอระบบ Window web solution หรือที่รู้จักกันในชื่อของ Window DNA ซึ่งได้มีการออกแบบสถาปัตยกรรมของแอปพลิเคชันแบบ 3-Tier หรือ N-Tier ก็ได้ โดยรูปแบบของสถาปัตยกรรมแบบ N-Tier นี้ ได้มีการแยกย่อยเพิ่มชั้นการทำงานของ Business Logic ขึ้นมาอีกหลายๆ ชั้น ซึ่งชั้น Business Logic นี้ได้ถูกแยกมาจากทั้ง Client และ ฐานข้อมูล ตามจำนวน Tier ที่ผู้ออกแบบต้องการ ซึ่งถูกเรียกว่า Business Tier หรือบางครั้งเรียกว่า Middle Tier โดยจะยกตัวอย่างโมเดลการทำงานแบบ 3-Tier ดังในภาพที่ 2.4



ภาพที่ 2.4 ตัวอย่างโมเดลการทำงานแบบ 3-Tier

ซึ่งสามารถเปรียบเทียบได้กับระบบสถาปัตยกรรมที่จะทำการออกแบบเพื่อสาธิตการทำงานของระบบได้ดังภาพที่ 2.5



ภาพที่ 2.5 โครงสร้างพื้นฐานของระบบสถาปัตยกรรมแบบ 3-Tier

โดยเราจะพบว่า Business Tier จะถูกแยกมาจากทั้ง Presentation Tier และ Data Tier ซึ่งเป็นตัวกลางในการที่ทำหน้าที่รับผิดชอบสำหรับ Business Rules ที่ได้รับ และสามารถที่จะบริหารจัดการข้อมูลที่ถูกเก็บไว้ใน Data Tier เพื่อส่งผ่านข้อมูลกลับไปให้ Presentation Tier สำหรับแสดงข้อมูลในรูปแบบที่ Client ร้องขอมา

ใน Window Web Solution แบบ 3-Tier เราจะมีกรออกแบบโครงสร้างของระบบสถาปัตยกรรมให้แต่ละ Service ของ Tier มีหน้าที่ในการสนับสนุนการทำงานดังนี้

- Business Service ประกอบด้วย COM components ต่างๆ ซึ่งใช้เป็นกลไกในการจัดการเข้าถึงข้อมูล
- User Services สามารถที่จะเป็นได้ทั้ง Windows Application แบบดั้งเดิม (เช่น Visual Basic Application) หรือ Brower-Based Web Application

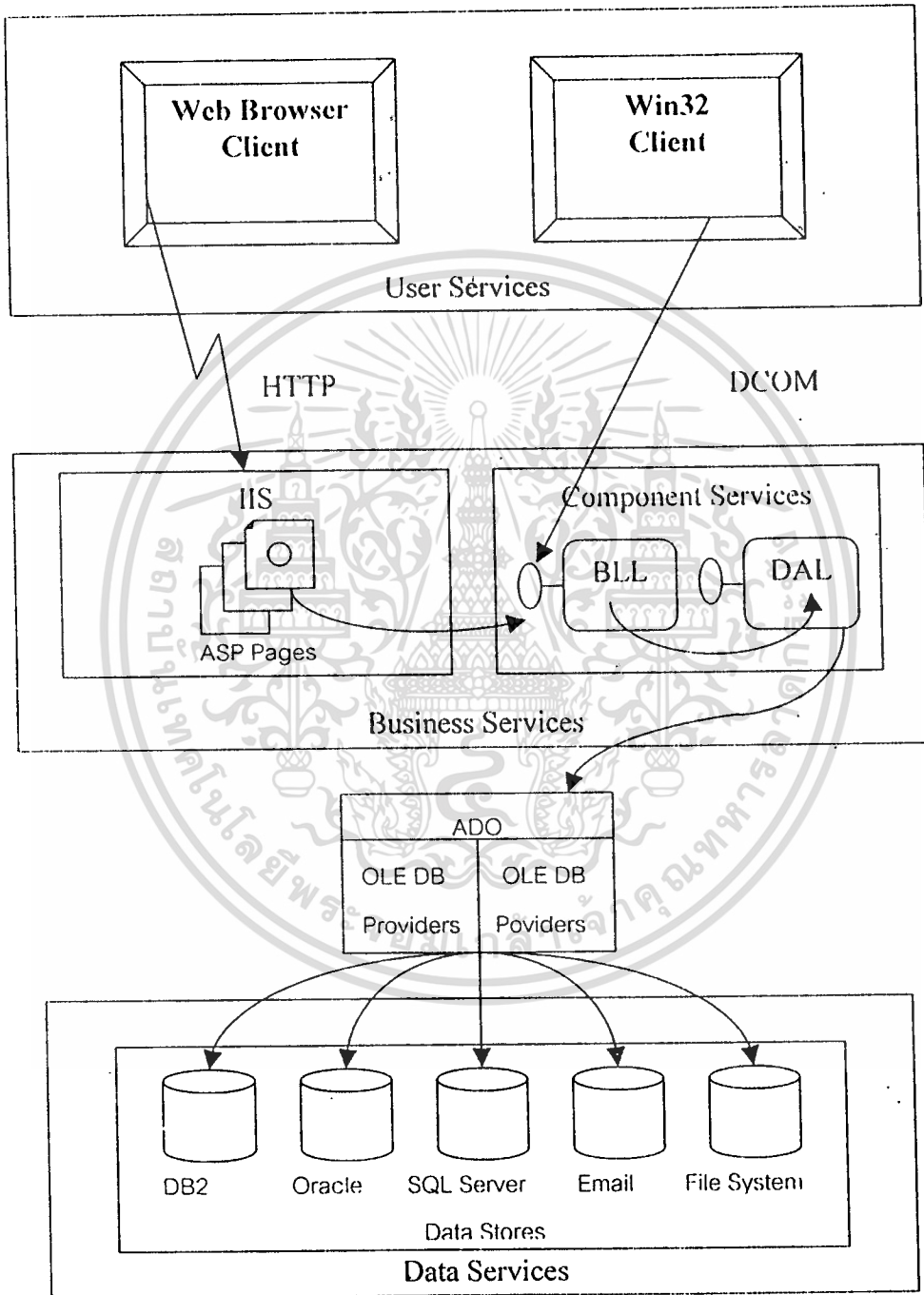
- **Data Services** สามารถเก็บข้อมูลได้ในรูปแบบต่างๆที่หลากหลาย ซึ่งประกอบด้วย Relational Database System เช่น SQL Server หรือ Oracle Database หรือ Mainframe System เช่น ระบบ IBM DB2 เป็นต้น

โดยหน้าที่ความสัมพันธ์ของการทำงานร่วมกันของแต่ละ Tier มีดังนี้

- **Presentation Tier** จะเข้าถึง Business Components ใน Business Tier ได้ โดยตรงผ่าน โพรโตคอล DCOM โดยกรณีของ Win32 Application หรือ โดยทางอ้อมก็จะผ่าน Active Server Pages (ASP) บน Web Server เช่น Internet Information Services (IIS) ด้วยการใ้ โพรโตคอล HTTP ในกรณีของ Web Browser Application
- **Business Tier** สามารถที่จะถูกแบ่งออกได้เป็น Business Logic Layer (BLL) และ Data Access Layer (DAL) ซึ่ง DAL Components ใช้ในการเข้าถึงข้อมูลที่เก็บไว้ใน Data Tier ผ่านทาง Microsoft ActiveX Data Objects (ADO) โดย ADO เป็นส่วนประกอบที่สำคัญมากใน Microsoft Data Access Components (MDAC) ซึ่งเป็นส่วนสำคัญของ Universal Data Access (UDA) ของไมโครซอฟท์ที่สามารถใช้ในการติดต่อกับข้อมูลได้ทุกชนิด อย่างเช่น Relational Database Management Systems หรือ Flat Files และ E-mails เป็นต้น
- **Data Tier** เป็น Tier ที่ทำหน้าที่ในการจัดเก็บข้อมูล หรือเป็นแหล่งข้อมูลต่างๆที่ เราต้องการติดต่อด้วย ซึ่งคอยให้บริการข้อมูลตามการร้องขอของ Business Tier

โดยเราสามารถสรุปสถาปัตยกรรมของ Window Web Solution หรือ Windows DNA ได้

ผังภาพที่ 2.6



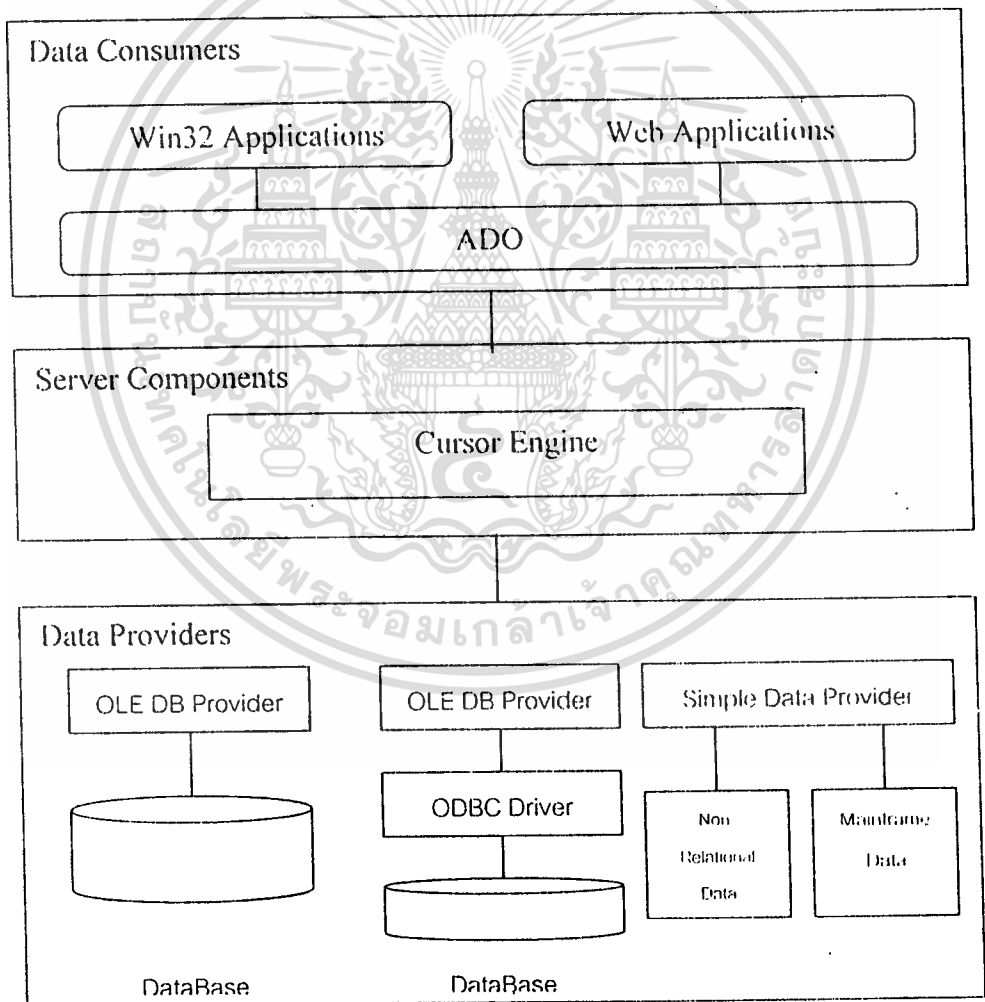
ภาพที่ 2.6 สถาปัตยกรรมของ Windows DNA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 การศึกษาโครงสร้างการเชื่อมต่อระบบการทำงานของ Windows Web Solution แบบ N-Tier
 เมื่อมองจากปัจจุบัน โนโพรเซสเซอร์ได้มีการนำเสนอโครงสร้างของเทคโนโลยี สำหรับบริการ
 เชื่อมโยงระบบและการติดต่อกันแหล่งข้อมูลชนิดต่างๆ ซึ่งสามารถอธิบายได้ดังต่อไปนี้

2.3.1 Microsoft Data Access Components (MDAC) และ Universal Data Access (UDA)

MDAC เป็นหัวใจของ UDA ซึ่งเป็นส่วนหนึ่งของ Windows Web Solution โดย UDA นี้
 โนโพรเซสเซอร์ได้ตั้งใจจัดเตรียมไว้สำหรับการเข้าถึงข้อมูลต่างๆ ตลอดทั้งระบบ Enterprise ดังนั้น
 สามารถสรุปสถาปัตยกรรมของ UDA ได้ดังภาพที่ 2.7



ภาพที่ 2.7 สถาปัตยกรรมของ Universal Data Access

จากภาพที่ 2.7 ที่แสดงไว้ข้างต้นจะเห็นได้ว่า ODA ประกอบด้วย 3 ส่วนประกอบหลักคือ

- **Data providers** ใช้ในการนำเสนอข้อมูลที่จัดเก็บ เช่น Database , File Systems ,Email System หรือ Mainframe System
- **Service Components** ส่วนที่ใช้จัดการและนำเสนอผลลัพธ์ของข้อมูลที่ได้อิงจากการใช้ OLE DB
- **Data consumers** ตัวที่ใช้ข้อมูลที่รับจากการจัดเตรียม OLE DB ของ Service components โดยตัวของ ADO เองเป็นตัวที่ใช้จัดการข้อมูลต่อจาก OLE DB ยกตัวอย่าง เช่น Win32 application และ ASP application

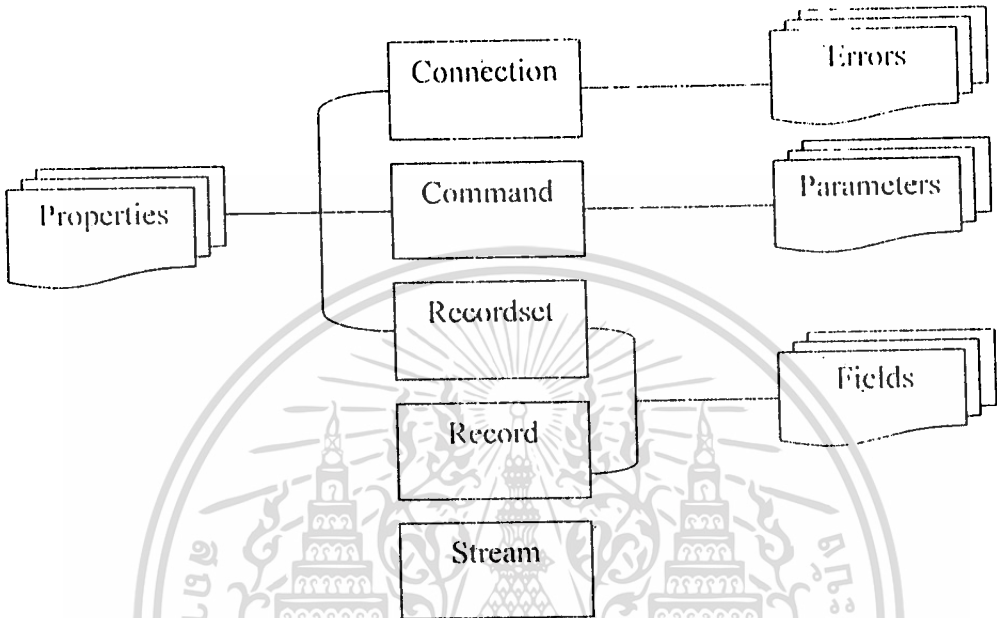
MDAC ถูกจัดเตรียมเพื่ออำนวยความสะดวกเข้าถึงข้อมูลในรูปแบบต่างๆของ ไมโครซอฟท์ โดยตัว MDAC ได้ถูกออกแบบให้มีประสิทธิภาพที่สูงขึ้น และ มีความน่าเชื่อถือมากยิ่งขึ้น

MDAC มีการสนับสนุน 3 เทคโนโลยีหลักๆ คือ

- OLE DB ซึ่งเป็น low-level และมีประสิทธิภาพที่สูงในการติดต่อกับข้อมูลที่หลากหลาย
- ADO ซึ่งเป็น high-level มีการใช้งานที่ง่ายในการติดต่อกับ OLE DB โดย ADO และ OLE DB สามารถทำงานได้เช่นเดียวกันกับข้อมูลได้ทั้งแบบ relational และ nonrelational
- ODBC ซึ่งเป็น low-level อย่างหนึ่ง ที่มีประสิทธิภาพที่สูงซึ่งถูกออกแบบสำหรับการเก็บข้อมูลแบบ relational

2.3.2 ActiveX Data Object (ADO)

ADO เป็น OLE DB แบบ Data Consumers ซึ่งมีรูปแบบโมเดลของ object ดังภาพที่ 2.8



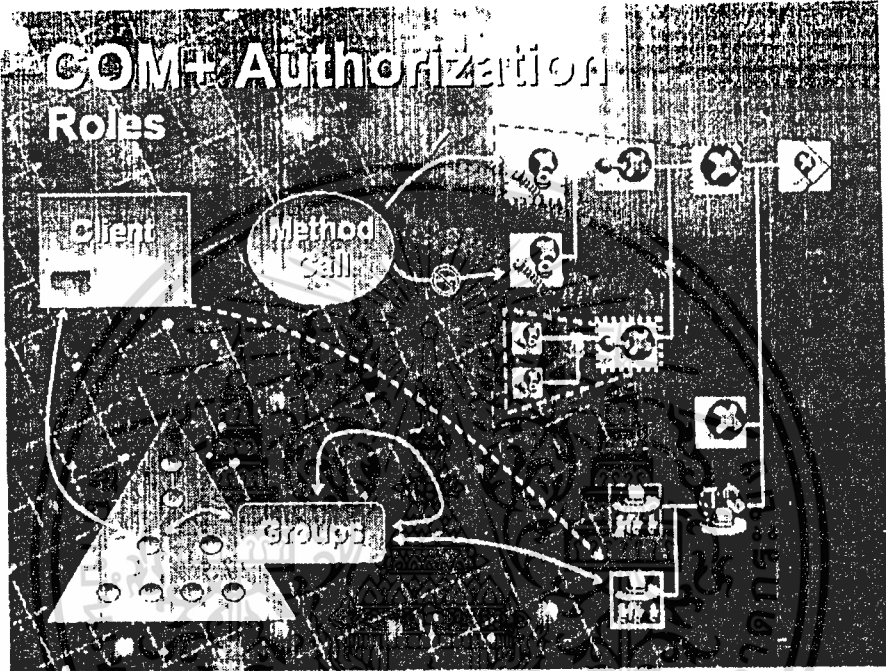
ภาพที่ 2.8 รูปแบบโมเดลของ ADO 2.5

2.4 การศึกษาการทำงานและเทคโนโลยี COM+ ร่วมกับ Windows Web Solution

โดยหัวใจหลักของ Windows Web Solution นั้นก็คือ Component Services นั่นเอง หรือที่เรียกว่า COM+ ซึ่งสามารถสรุปหน้าที่การทำงานที่จะใช้ในโครงงานนี้ได้ดังนี้

- **Distributed transactions** โดยตัว COM+ สามารถที่จะควบคุมบริหารจัดการทรานแซกชันที่เกิดขึ้นทั้งหมดให้มีความถูกต้อง เช่น ถ้าเรามีฐานข้อมูลอยู่หลายที่ซึ่งต้องการจะปรับปรุงข้อมูลให้ถูกต้องครบถ้วนเหมือนกันทั้งหมด สมมติเรามี ฐานข้อมูลอยู่ 2 ที่คือ ที่กรุงเทพฯ กับที่เชียงใหม่ ซึ่งต้องมีการแทรกข้อมูลสัก 100 แถวให้กับฐานข้อมูลทั้ง 2 ที่ ถ้าหากมีเหตุการณ์ในกรณีที่มีการปรับปรุงฐานข้อมูลที่กรุงเทพฯเรียบร้อยแล้ว แต่ที่เชียงใหม่แทรกข้อมูลไปได้แค่ 50 แถวแล้วมีปัญหาทรานแซกชันเกิดขึ้น ไม่ว่าจะด้วยสาเหตุใดก็ตาม ตัว COM+ สามารถที่จะ Rollback ข้อมูลทั้งหมดได้ กลับสู่สภาวะเดิมได้

- Security ตี COM+ สามารถที่จะที่เชื่อมร่วมกับ Active Directory ของ Windows 2000 Server ได้ โดยผ่านการทำงานของโพรโตคอล LDAP โดยที่โครงสร้างของระบบการทำงานดังภาพที่ 2.9



ภาพที่ 2.9 COM+ Authorization

- Object pooling สามารถช่วยลด overhead ที่เกิดจากการสร้าง instances of components ซึ่งใช้ในการปรับปรุงประสิทธิภาพการทำงานให้สูงขึ้น ตัวอย่างเช่น เมื่อ Client มีการเรียกใช้งานตัว object ที่ได้สร้างขึ้นมา ตัว object นั้นจะถูกทำหลาย แต่ในการทำงานของ COM+ นั้น จะมีการเก็บ object ที่เลิกใช้งานนั้นไว้ใน pool ซึ่งดีมากกว่าเมื่อมีการเรียกใช้งาน Client เข้ามาอีกไม่ว่าจะเป็น Client ตัวไหนก็ตามตัว object นั้นก็จะรวมที่จะนำงานใช้งานได้ต่อ แต่ทางฝั่ง Client ก็ยังคงต้องทำการสร้าง instances of components ใหม่เหมือนการเรียกใช้งานปกติ ซึ่งที่เมื่อตัว COM+ จะทำหน้าที่จัดการให้เองตามที่เราได้ทำการเซตค่าเอาไว้

หมายเหตุ เราสามารถเซตค่าจำนวนของ object ที่จัดเก็บไว้ใน pool ได้ และเซตค่าระยะเวลาให้ object อยู่ใน pool ได้

- **Object construction** เป็นการใช้ COM+ Type Library ซึ่งทำให้เราสามารถเลือกการเชื่อมต่อฐานข้อมูลได้ โดยมีวิธีส่งผ่านพารามิเตอร์ Constructor String ของเว็บ COM+ ซึ่งเราไม่ต้องทำการแก้ไขโปรแกรมใหม่ และช่วยให้ประสิทธิภาพการทำงานของเว็บที่รวดเร็วกว่าไฟล์ DSN ด้วย

2.5 การประยุกต์การทำงานของเทคโนโลยี COM+ พร้อมการติดตั้งและการทดสอบการทำงาน

ขอยกตัวอย่างแนวทางในการทำงานของ Component Service หรือที่เรียกว่า COM+ ในลักษณะการรักษาความปลอดภัยของจากการใช้โพรโตคอล LDAP เพื่อใช้ในแนวทางในการแก้ปัญหาการพัฒนาระบบแอปพลิเคชันของเว็บต่อไป โดยจะอธิบายได้คร่าวๆ ดังนี้

2.5.1 วิธีการในการเรียกใช้ Security ของ Active Directory

เราจะต้องทำการ Reference โปรแกรมถึง Active DS Type Library เพื่อใช้สำหรับการเชื่อมระบบเข้ากับ Active Directory

โดยเริ่มแรกเราจะต้องทำการประกาศตัวแปร Object ที่เกี่ยวข้อง ดังนี้

```
Dim np As IADsOpenDSObject
```

```
Dim myUser As ActiveDs.IADsUser
```

```
Dim myGroup As ActiveDs.IADsGroup
```

ทำการ Create ตัวแปร เพื่อใช้ในการค้นหาข้อมูลของระบบ

```
Dim sys As New ADSysInfo
```

ทำการประกาศตัวแปรเพื่อกำหนดข้อมูลชื่อ โดเมนของระบบของเครื่องเซิร์ฟเวอร์นั้น

```
Dim sPath As String
```

```
sPath = Mid(sys.DomainDNSName, 1, InStr(sys.DomainDNSName, ".") - 1)
```

การสร้าง Object ของ LDAP

```
Set np = GetObject("LDAP:")
```

ทำการใช้ Object ของ LDAP ในการส่งพารามิเตอร์ต้นฐานของ LDAP ไปที่การดึง
 ออกแสดงการใช้งาน

```
Set myUser = np.OpenDSObject("LDAP://CN " & พารามิเตอร์ Username & "." &
    "CN Users,DC " & Trim(Path) & ".DC com", พารามิเตอร์ Username, พารามิเตอร์
    Password, ADS_SECURE_AUTHENTICATION)
```

ทำการตรวจสอบข้อมูลของ User ว่ามีอยู่ใน Group ดังกล่าวหรือไม่

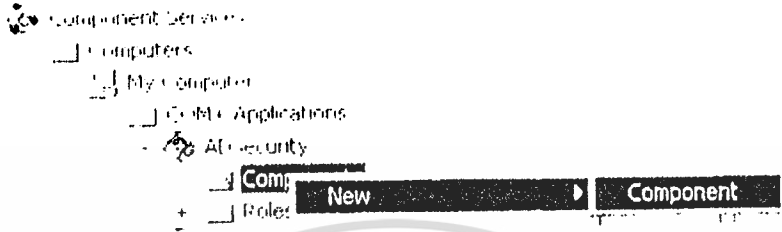
```
For Each myGroup In myUser.Groups
    If "พารามิเตอร์ของ Group" = Mid(myGroup.Name, 4, Len(myGroup.Name) - 3) Then
        VerifyLogOn = True
        Exit Function
    End If
Next
```

2.5.2 ขั้นตอนการติดตั้งการทำงานของ COM+

แหล่งเอกสารการสร้าง Class ที่ใช้สำหรับการเชื่อมต่อกับ Active Directory แล้ว เราจะทำการ
 ครอบหุ้มไฟล์เป็นไฟล์ DLL แล้วทำการติดตั้งลงในตัว COM+ ซึ่งมีขั้นตอนดังต่อไปนี้

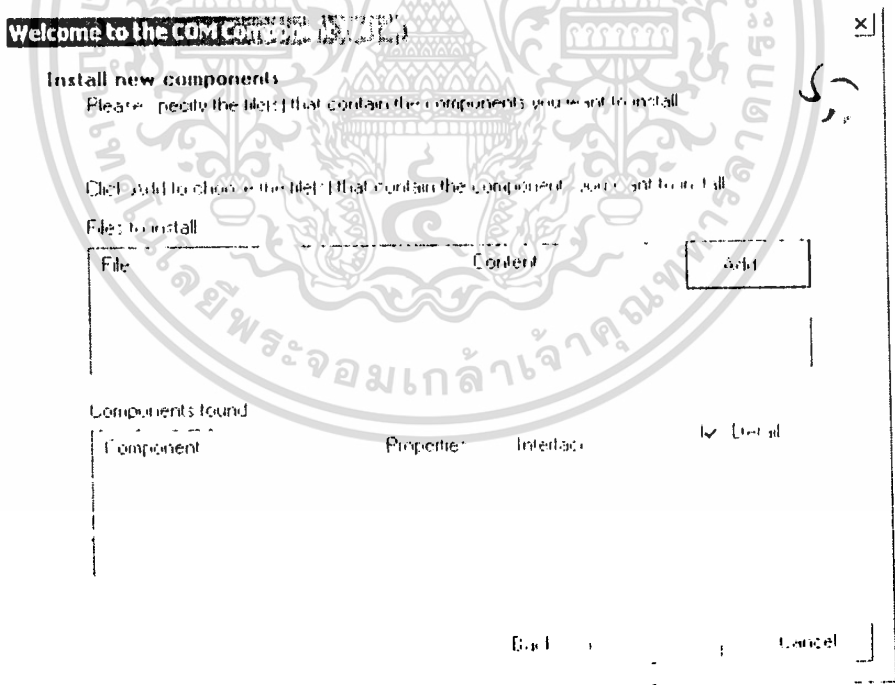
- ไปที่ Start >Program > Administrative Tools >Component Services
- ไปที่ Component Service >My Computer
- คลิกขวาที่ COM+ Application แล้วเลือก New >Application ดังภาพที่ 2.10

- กลับมาที่ขั้นถัดไปการเพิ่ม Component Object Model (COM) เป็น COM Application ที่เคยเรียนรู้ไว้ดูภาพที่ 2.12



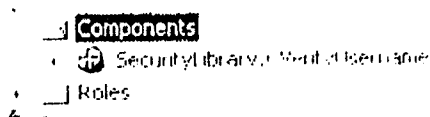
ภาพที่ 2.12 การติดตั้ง Component Object Model (COM)

- คลิก Install new component(s) แล้ว คลิก Add เพื่อทำการเลือกไฟล์ DLL ที่ได้ทำการสำรองไว้ ดังภาพที่ 2.13



ภาพที่ 2.13 การติดตั้ง COM จากไฟล์ DLL

- เมื่อติดตั้งได้เสร็จจะปรากฏหน้าต่างดัง ภาพที่ 2.14



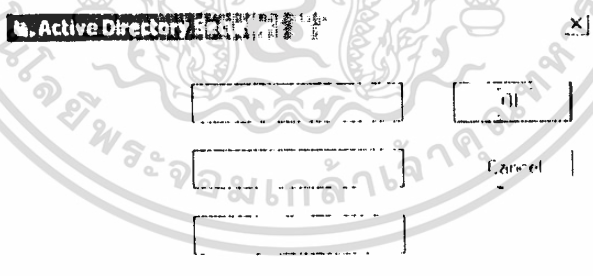
ภาพที่ 2.14 COM ที่ได้ติดตั้งเสร็จ

2.5.3 ขั้นตอนการสร้างการรักษาความปลอดภัยของ Active Directory

- ไปที่ Start -> Program -> Administrative Tools -> Active Directory Users and Computers
- ทำการสร้าง Groups ขึ้นมา
- ทำการสร้าง Username ขึ้นมาแล้วทำการ Add ให้เข้าไปอยู่ใน Groups ที่ได้สร้างเอาไว้

2.5.4 ขั้นตอนการทดสอบการทำงานของ COM+ Application กับ Active Directory

- ทำการ Run โปรแกรมที่ได้สร้างไว้ ชื่อ Active Directory Security เพื่อทำการทดสอบการทำงาน ดังภาพที่ 2.15



ภาพที่ 2.15 โปรแกรม Active Directory Security

บทที่ 3

การพัฒนาระบบพาณิชย์อิเล็กทรอนิกส์สมัยใหม่ด้วย Windows Web Solution

จากการศึกษาแนวโน้มของการพัฒนาระบบแอปพลิเคชันของเว็บ ที่ถนัดมาตั้งแต่อดีตจนถึงปัจจุบัน จะเห็นได้ว่าการพัฒนาในเชิงที่ก้าวไปสู่ระบบที่เป็น Distributed มากขึ้น โดยมีแนวโน้มที่จะมีการเชื่อมโยงระหว่างระบบของเว็บแอปพลิเคชันให้สามารถทำงานร่วมกับระบบบนพีซีอื่น ๆ ที่มีอยู่มาให้ และสามารถที่จะเชื่อมโยงระบบต่อไปได้ในอนาคต โดยสถาปัตยกรรมของระบบที่จะออกมามี จะเป็นในลักษณะของ N-Tier ซึ่งยึดหลักการพัฒนาระบบซอฟต์แวร์ตามแบบ Object-Oriented เพื่อเป็นแนวทางในการดำเนินการที่เป็นมาตรฐาน ซึ่งมีขั้นตอนดังนี้

3.1 ชั้นที่ 1 : Problem Statement

เบื้องต้นการพัฒนาเว็บแอปพลิเคชันของเว็บในแบบดั้งเดิมนั้น มีลักษณะสถาปัตยกรรม Client-Server ซึ่งมีข้อจำกัดดังนี้

- จำนวนของผู้ใช้ ซึ่งต้องไม่เกินจำนวนที่กำหนดเกณฑ์ได้เซตค่าคงไว้ เช่น จำนวนของผู้ใช้ในการเข้าใช้ฐานข้อมูล หรือผลจากการที่ผู้ใช้มีจำนวนเพิ่มขึ้น ส่งผลให้มีการใช้ทรัพยากรของเซิร์ฟเวอร์มากขึ้น ก็จะมีผลทำให้ประสิทธิภาพของระบบลดลง และอาจทำให้ระบบล่มเหลวได้
- ความเร็วเชื่อถือ เนื่องจากถ้าหากว่าเซิร์ฟเวอร์มีการอัปเดตเครื่อง อันเนื่องมาเหตุผลใดก็ตาม เช่น ครบกำหนดการบำรุงรักษาระบบ ก็จะทำให้เครื่อง Client ไม่สามารถเข้ามาใช้งานได้ในช่วงเวลานั้นๆ
- การบำรุงรักษาระบบ หากจำเป็นต้องมีการแก้ไขเปลี่ยนแปลงฟังก์ชันของระบบเพิ่มเติม ก็อาจเป็นการยากที่จะต้องกลับไปแก้ไข โปรแกรมในบางครั้ง

ซึ่งจากข้อจำกัดที่ได้กล่าวมาในบางครั้งก็เป็นสาเหตุให้เว็บไซต์บางที่ไม่สามารถเข้าไปใช้งานได้ อาจส่งผลทำให้บริษัทหรือองค์กรนั้นๆ ขาดรายได้หรือสูญเสียผลประโยชน์ได้ หากเกิดขึ้นบ่อยๆ

ดังนั้นการพัฒนาระบบในลักษณะที่เป็น N-Tier โดยที่นักเทคโนโลยี (COM) มาช่วยในการจัดการระบบสารสนเทศ และ XMI มาช่วยเพิ่มประสิทธิภาพทางด้านระบบช่วยในการส่งถ่ายข้อมูลแบบเสมือน หรืออาจจะสามารถนำมาใช้ในการแลกเปลี่ยนข้อมูลระบบกับองค์กรหรือบริษัทก็ได้ ขณะที่นักการออกแบบระบบแบบ N-Tier นั้นช่วยให้เราสามารถรักษาระบบทำได้ง่าย

3.2 ขั้นที่ 2 : Gathering Requirement

การพัฒนาระบบพาณิชย์อิเล็กทรอนิกส์สมัยใหม่แบบ Window Web Solution นั้น เมื่อเสร็จสิ้นโครงการพัฒนาระบบจะมีความสามารถในการดำเนินงานดังนี้

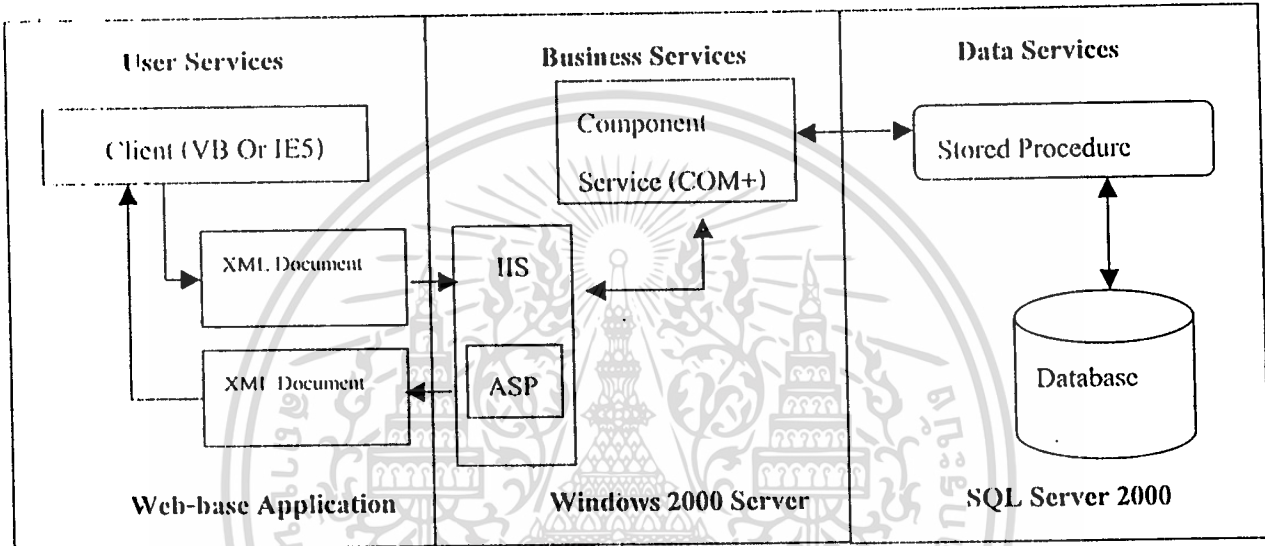
- สามารถบริหารจัดการระบบสารสนเทศที่มีการเชื่อมต่อฐานข้อมูลได้อย่างมีประสิทธิภาพมากขึ้น (เช่น สามารถเรียกใช้งานฐานข้อมูลได้มากกว่า 1 ฐานข้อมูลเป็นต้น)
- สามารถรองรับ Transparency ต่างๆของระบบ Distributed ได้ เช่น Location Transparency
- สามารถส่งผ่านข้อมูลในรูปแบบ XMI
- มีระบบการรักษาความปลอดภัยที่สูงขึ้น
- มีความน่าเชื่อถือ
- สามารถบำรุงรักษาได้ง่ายมากขึ้น

3.3 ขั้นที่ 3 : Design

ผู้ศึกษาได้ทำการออกแบบระบบ เพื่อการพัฒนาโปรแกรมนำร่อง โดยกำหนดให้โปรแกรมที่พัฒนาได้นี้ มีความสามารถในการทำงานตามข้อ 4.2 ทั้งนี้โปรแกรมนำร่องนี้ จะสอดคล้องขั้นตอนการดำเนินงานของระบบพาณิชย์อิเล็กทรอนิกส์พื้นฐานในการค้นหาสินค้า และกรรับคำสั่งซื้อสินค้าลงในตะกร้าสินค้า โดยการออกแบบระบบการทำงานจะแบ่งออกได้ดังนี้

3.3.1 การออกแบบสถาปัตยกรรมแบบ N-Tier ของ Windows Web Solution

ขงข้อจำกัดของเว็บแอปพลิเคชันแบบ Client-Server ซึ่งไม่สามารถรองรับการทำงานของระบบที่ซับซ้อนมากๆ ได้ ดังนั้นจึงทำให้ต้องมีการพัฒนาระบบที่เป็นแพลตฟอร์มขึ้นมาใช้ในการพัฒนาแอปพลิเคชันแบบ 3-Tier หรือ Distributed Application โดยมีองค์ประกอบดังภาพที่ 3.1



ภาพที่ 3.1 องค์ประกอบของการพัฒนาแอปพลิเคชันแบบ 3-Tier หรือ Distributed Application

โดยระบบ Windows Web Solution ที่เราได้ออกแบบนี้ เราจะแบ่งการทำงานออกได้เป็น 3 ส่วนหลักๆ ก็คือ

- **User Services** ซึ่งทำหน้าที่ในการเชื่อมต่อการทำงาน และส่งพารามิเตอร์ กับ Business Services ด้วยโพรโตคอล HTTP แล้วนำผลลัพธ์ที่ได้มาแสดงให้กับฝั่ง Client ผ่านทาง ASP สำหรับการพัฒนาระบบเว็บแอปพลิเคชัน แต่ถ้าหากเป็นการพัฒนาระบบแบบ Windows Application เช่น VB เราจะมีการใช้โพรโตคอล DCOM ในการเชื่อมต่อการทำงานกับ Component Service ของ Business Services โดยตรงไม่ต้องผ่าน ASP
- **Business Service** ซึ่งทำหน้าที่ในการเชื่อมต่อการทำงานระหว่าง User Services กับ Data Services ซึ่งสามารถที่จะแบ่งการทำงานเป็นส่วนย่อยๆ ลงไปได้ถึง ซึ่งได้แก่
 - **Transport Layer** ซึ่งเป็นชั้นที่ทำหน้าที่ให้บริการ ในการสับแก๊งข้อมูล โดยอาจจะอยู่ในรูปแบบของ XML หรือ Recordset เป็นต้น

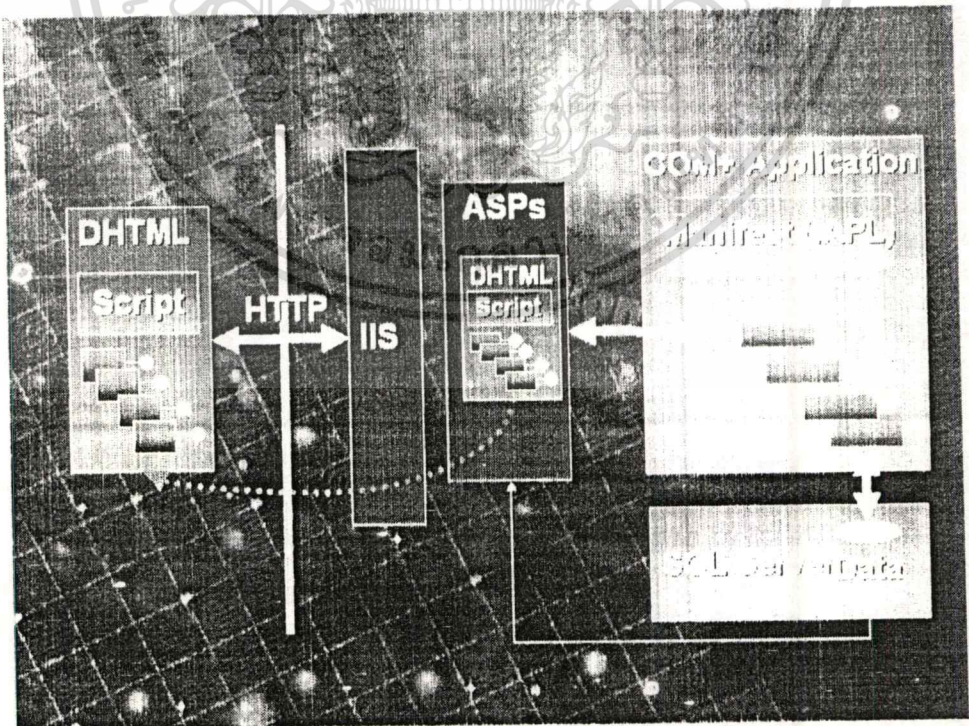
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Business Logic Layer(BLL)** ซึ่งเป็นชั้นที่ที่แทนที่ไ้กับกระบวนการทำงานต่างๆที่ได้รับมา Gathering Requirement หรือต้นแบบได้
- **Data Access Layer (DAL)** ซึ่งเป็นชั้นที่ที่แทนที่ไ้กับกระบวนการเชื่อมข้อมูลกับฐานข้อมูลผ่านทาง ADO
- **Data Service** ทำหน้าที่ในการบริหารจัดการข้อมูลผ่านทาง Stored Procedure ซึ่งอาจจะเรียกได้ว่าเป็น process การทำงานอย่างหนึ่งที่ได้ฝังคำสั่ง Sql command ไว้

3.3.2 ส่วนซอฟต์แวร์ที่ใช้สำหรับการออกแบบ Windows Web Solution

ในการพัฒนาระบบพาณิชย์อิเล็กทรอนิกส์สมัยใหม่นี้ จะเลือกใช้โปรแกรม VB เป็นตัวหลักในการพัฒนาระบบในส่วนของ Component Services หรือที่เรียกว่า COM+ เพื่อจะได้ง่ายในการใช้เป็นแนวทางสำหรับการศึกษาพัฒนาระบบอื่นๆ โดยเราจะใช้ตัวโปรแกรม VB เป็นตัวช่วยสร้าง DDL ซึ่งเป็น Component Object Model มาใช้งานร่วมกับ Component Services

ส่วนซอฟต์แวร์อื่นที่ต้องใช้ในการพัฒนาระบบพาณิชย์อิเล็กทรอนิกส์สมัยใหม่นี้ สามารถดูได้ดังภาพที่ 3.2



ภาพที่ 3.2 ซอฟต์แวร์พื้นฐานในการพัฒนาระบบพาณิชย์อิเล็กทรอนิกส์
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากภาพที่ 3.2 ข้างต้น จะเห็นได้ว่า เราสามารถประยุกต์โครงสร้างระบบที่โดยกรรมให้ลดแรงกดดันได้ ซึ่งทำให้ระบบของเรามีความยืดหยุ่นได้ดั่งที่เราต้องการ กล่าวคือ

- แบบที่ 1 เป็นลักษณะที่แบ่งการทำงานออกเป็นเพียง 2 ส่วน
 - ฝ่าย Client
 - โปรแกรม Web-based Application เช่น Internet Explorer เป็นต้น
 - ระบบปฏิบัติการ เช่น Windows98 ,Window Me, Linux ก็ได้
 - ฝ่าย Server
 - ระบบปฏิบัติการ Windows 2000 Server
 - โปรแกรม IIS 5 ขึ้นไป หรือ อื่นๆก็ได้ ที่รองรับการดำเนินงานของ ASP
 - โปรแกรม Component Services
 - โปรแกรม SQL Server 2000
- แบบที่ 2 เป็นลักษณะที่แบ่งการทำงานออกเป็น 3 ส่วน
 - ส่วน Client
 - โปรแกรม Web-based Application เช่น Internet Explorer เป็นต้น
 - ระบบปฏิบัติการ เช่น Windows98 ,Window Me, Linux ก็ได้
 - ส่วน Server ของ Business services
 - ระบบปฏิบัติการ Windows 2000 Server
 - โปรแกรม IIS 5 ขึ้นไป หรือ อื่นๆก็ได้ ที่รองรับการดำเนินงานของ ASP
 - โปรแกรม Component Services

หมายเหตุ เราสามารถแยกเป็น Server ย่อยๆ ได้อีกก็ได้
 - ส่วน Server ของ Data Service
 - ระบบปฏิบัติการ Windows 2000 Server
 - โปรแกรม SQL Server 2000

โดยเราสามารถเปรียบเทียบข้อดี-ข้อด้อยของแบบที่ 1 และ 2 ได้ดังนี้

- แบบที่ 1 ใช้ต้นทุนในการดำเนินการในระยะเริ่มต้นน้อยกว่า แบบที่ 2 ซึ่งใช้จำนวน Server มากกว่า
- แบบที่ 1 ประสิทธิภาพค่อนข้างต่ำ เมื่อเทียบกับ แบบที่ 2 ในกรณีที่มีการโหลดใช้งานพร้อมกันทั้งในส่วนที่เป็น Business Service และ Data Service พร้อมกัน

3.3.3 รายละเอียดของการพัฒนาระบบพาณิชย์อิเล็กทรอนิกส์

ระบบพาณิชย์อิเล็กทรอนิกส์นี้ เราจะมีการแบ่งกลุ่มลูกค้าเป็น 2 กลุ่ม คือ กลุ่มที่เป็นสมาชิก และ กลุ่มลูกค้าทั่วไป โดยกลุ่มที่เป็นสมาชิก จะต้องทำการ Login เข้าสู่ระบบก่อนเสมอ จึงจะสามารถสั่งซื้อสินค้าได้ โดยระบบพาณิชย์อิเล็กทรอนิกส์ที่เราพัฒนานี้จะใช้ระบบการรักษาความปลอดภัยของ Active Directory ของระบบปฏิบัติการ Windows 2000 Server แต่ถ้าหากว่ายังไม่ได้มีการสมัครเป็นสมาชิก ก็จะสามารถทำการค้นหาข้อมูลสินค้าได้ตามกลุ่มของสินค้า ชื่อสินค้า และ ยี่ห้อสินค้า แต่จะไม่สามารถสั่งซื้อสินค้าได้

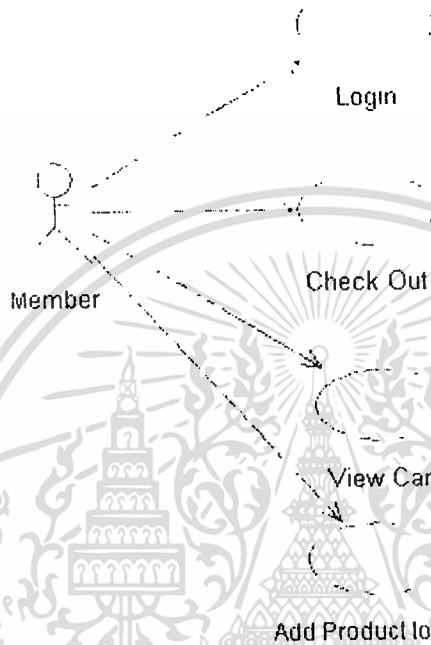
โดยแนวคิดในการพัฒนาระบบพาณิชย์อิเล็กทรอนิกส์นี้ เราจะมีการเขียนคลาสขึ้นมาเพื่อใช้เชื่อมการทำงานของตัว COM+ Application กับ Active Directory และ ฐานข้อมูล โททเว จะแบ่งคลาสที่เราพัฒนาออกเป็น 3 ส่วน คือ

- ส่วนที่ทำหน้าที่เชื่อมต่อกับ Active Directory เพื่อทำหน้าที่ในเรื่องของความปลอดภัย
- ส่วนที่ทำหน้าที่เชื่อมต่อกับฐานข้อมูล เพื่อใช้ในการเรียกใช้งาน Stored Procedures ในฐานข้อมูลที่เราได้เขียน Script คำสั่ง SQL เอาไว้
- ส่วนที่ทำหน้าที่ในการแปลงข้อมูลให้อยู่ในรูปแบบของ XML และดึง Script ของ XSL ที่เราสร้างเอาไว้มาจัดรูปแบบในการนำเสนอแบบ HTML

โดยคลาสที่เราพัฒนานี้จะถูกเก็บไว้ในตัว COM+ Application เพื่อช่วยให้ระบบที่เราพัฒนาสามารถช่วยกระจายโหลดที่จะเกิดขึ้นในอนาคตได้ ซึ่งสามารถดูวิธีแยกระบบได้ที่ ภาคผนวก ข

3.3.4 การออกแบบระบบโมเดลข้อมูลแบบ Object-Oriented ด้วย UML.

3.3.4.1 Use Case Diagram

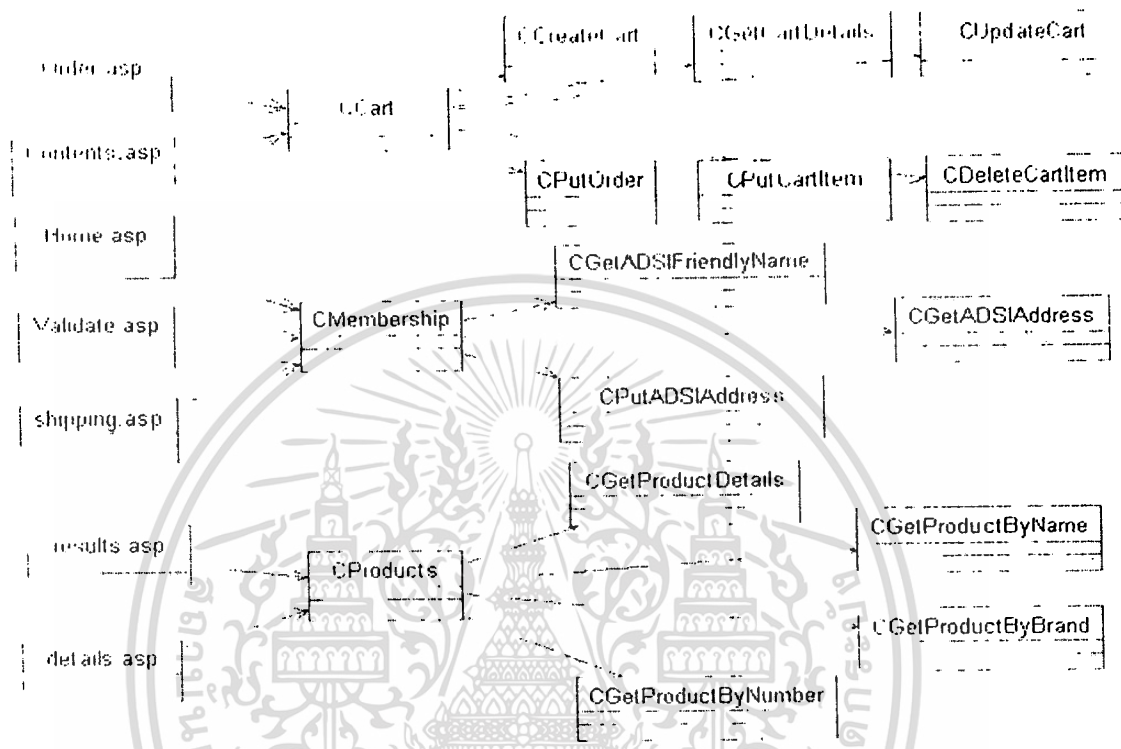


ภาพที่ 3.3 Use Case Diagram

โดยเราจะแบ่งการทำงานของระบบพาณิชย์อิเล็กทรอนิกส์ออกเป็น 4 Use Case คือ

- Login ใช้ในการตรวจสอบสิทธิ์ในการเข้าใช้ระบบพาณิชย์อิเล็กทรอนิกส์
- Check Out ใช้ในการตรวจสอบความถูกต้องของการสั่งซื้อสินค้า ก่อนที่จะสิ้นสุดกระบวนการสั่งซื้ออย่างสมบูรณ์
- View Cart ใช้ในการดูข้อมูลสินค้าที่อยู่ในรถเข็น ซึ่งสามารถเปลี่ยนแปลงจำนวนของสินค้าได้ ถ้าต้องการสินค้ามากกว่า 1 ชิ้น
- Add Product to Cart ใช้ในการเพิ่มสินค้าลงในรถเข็น

3.3.4.2 Class Diagram



ภาพที่ 3.4 Class Diagram

ส่วนนี้เป็นการออกแบบการทำงานของระบบพาณิชย์อิเล็กทรอนิกส์ ในส่วนของระบบที่ที่
 อกเพิ่มเติมการทำงานกับฐานข้อมูล โดยเราจะมีการออกแบบระบบที่ในคลเซ เพื่อใช้ในการเรียก
 ใช้ Stored Procedure ของ SQL Server 2000 หากเราพิจารณาจากภาพที่ 4.4 เราสามารถแยกระบบ
 ออกได้เป็น 3 ส่วนได้ ดังนี้

- User Service ซึ่งได้แก่
 - Order.asp ใช้ในการ Confirm การสั่งซื้อสินค้าก่อนสิ้นสุดกระบวนการรับ Order
 - Contents.asp ใช้ในการแสดงข้อมูลของสินค้าที่อยู่ในรถเข็น ที่ซึ่งสมาชิกได้ทำการใส่ข้อมูลไว้
 - Home.asp ใช้ในการรับข้อมูล UserName และ Password เพื่อใช้ในการตรวจสอบสิทธิ์

- shipping.asp ใช้ในการระบุข้อมูลสถานที่ที่รับส่งพัสดุ
- results.asp ใช้ในการแสดงผลข้อมูลสินค้า จากกรค้นแบบค้นแล้ว
- detail.asp ใช้ในการแสดงผลรายละเอียดของข้อมูลสินค้า
- Business Service ซึ่งได้แก่
 - CCart โดยจะทำหน้าเรียกใช้งาน CCreateCart ,CGetCartDetails, CUpdateCart ,CPutOrder,CPutCartItem และ CDeleteCartItem
 - CMembership โดยจะทำหน้าเรียกใช้งาน CGetADSIFriendlyName, CGetADSIAddress และ CPutADSIAddress
 - CProducts โดยจะทำหน้าเรียกใช้งาน CGetProductDetails , CgetProductByName, CGetProductByBrand และ CGetProductByNumber
- Data Service ซึ่งได้แก่
 - CCreateCart โดยจะทำหน้าเรียกใช้งาน Stored Procedure ชื่อ DNACreateCart
 - CGetCartDetails โดยจะทำหน้าเรียกใช้งาน Stored Procedure ชื่อ DNAGetCartDetails
 - CUpdateCart โดยจะทำหน้าเรียกใช้งาน Stored Procedure ชื่อ DNADeleteCartItem และ DNAUpdateQuantity
 - CPutOrder โดยจะทำหน้าเรียกใช้งาน Stored Procedure ชื่อ DNAPutOrder
 - CPutCartItem โดยจะทำหน้าเรียกใช้งาน Stored Procedure ชื่อ DNAPutCartItem
 - CDeleteCartItem โดยจะทำหน้าเรียกใช้งาน Stored Procedure ชื่อ DNADeleteCartItem
 - CGetADSIFriendlyName โดยจะทำหน้าเรียกใช้งาน Active Directory ในการตรวจสอบความถูกต้อง
 - CGetADSIAddress โดยจะทำหน้าที่เรียกใช้งาน Active Directory ในการเลือกข้อมูลของสมาชิกจาก Active Directory
 - CPutADSIAddress โดยจะทำหน้าที่เรียกใช้งาน Active Directory ในการแก้ไขข้อมูลของสมาชิก

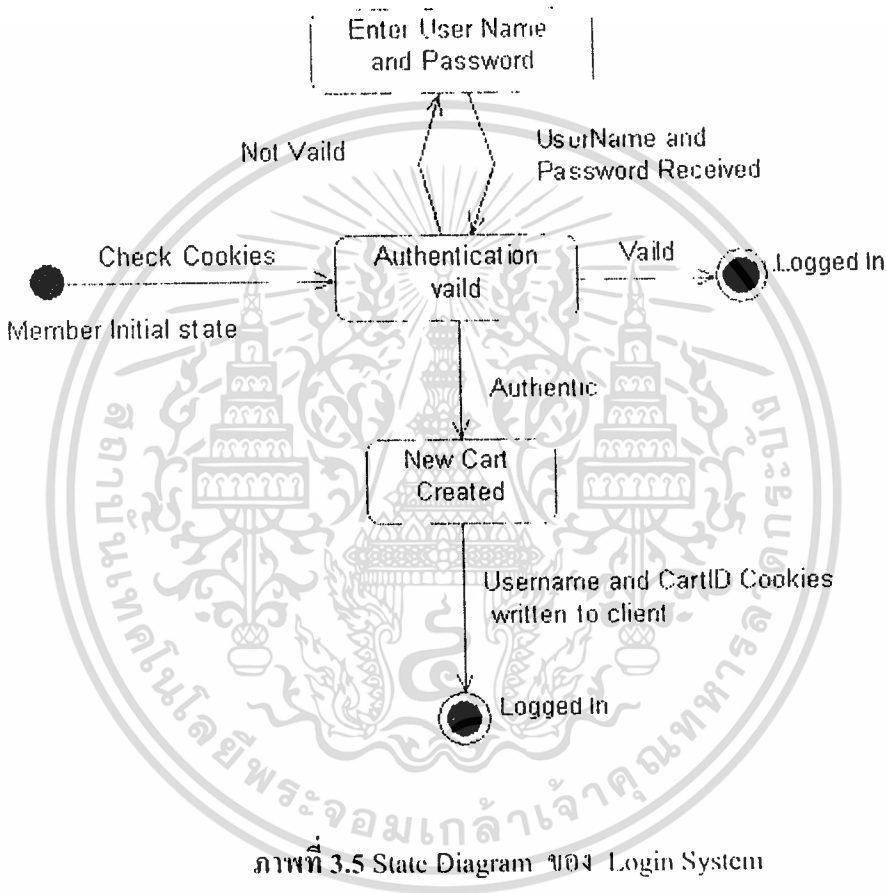
- o) CGetProductDetails โดยจะทำการเรียกใช้งาน Stored Procedure ชื่อ DNAGetProductsDetails
- o) CGetProductByName โดยจะทำการเรียกใช้งาน Stored Procedure ชื่อ DNAGetProductsByProductName
- o) CGetProductByBrand โดยจะทำการเรียกใช้งาน Stored Procedure ชื่อ DNAGetProductsByBrandName
- o) CGetProductByNumber โดยจะทำการเรียกใช้งาน Stored Procedure ชื่อ DNAGetProductsByVenderName



3.3.4.3 State Diagram

นอกจากแผนภาพ Use Case Diagram เราสามารถแสดงถึงสภาวะระบบต่อๆ Use Case ที่เราเปิด แผนภาพออกมาเป็นได้ดังนี้

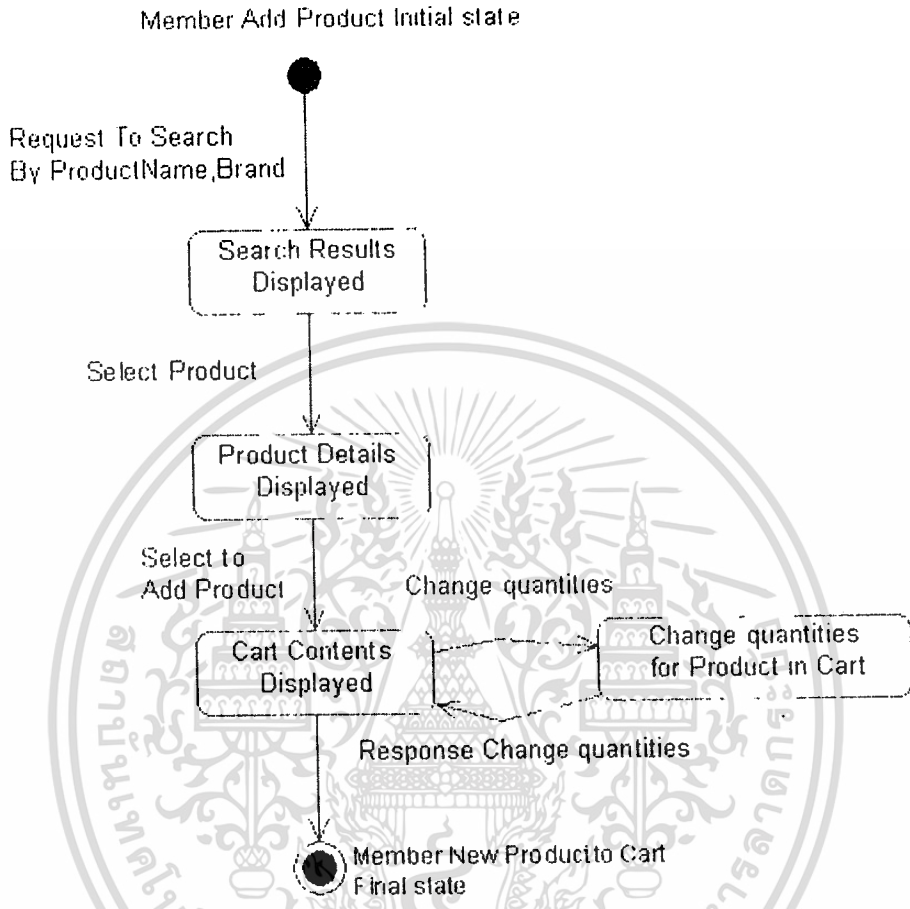
- Login System



ภาพที่ 3.5 State Diagram ของ Login System

เมื่อมีการเข้าใช้งานระบบ ระบบจะทำการตรวจสอบ Cookies ของเครื่อง Client นั้นว่ามี
 สกีนถูกแปลงตามเงื่อนไขที่ระบบกำหนดไว้หรือไม่ โดยเราจะมีการกำหนดไว้ Cookies มีอายุการใช้
 รมเป็นไปเพียง 1 วัน หลังจากที่มีการ Loginเข้าใช้งานระบบ และข้อมูลใน Cookies นี้จะถูกทำลาย
 หมดไปเมื่อมีการยืนยันการสั่งซื้อสินค้า แต่ถ้าหากว่าข้อมูลใน Cookies ไม่มีหมดอายุก็ต้อง สมาชิก
 จะต้องทำการ Login เข้าระบบไปม่อีกครั้ง และจะได้รับ หมายเลขรถเล่นใหม่ เพื่อใช้ในการเก็บราย
 ละเอียดสมาชิกแต่ละคนได้ทำการสั่งซื้อ

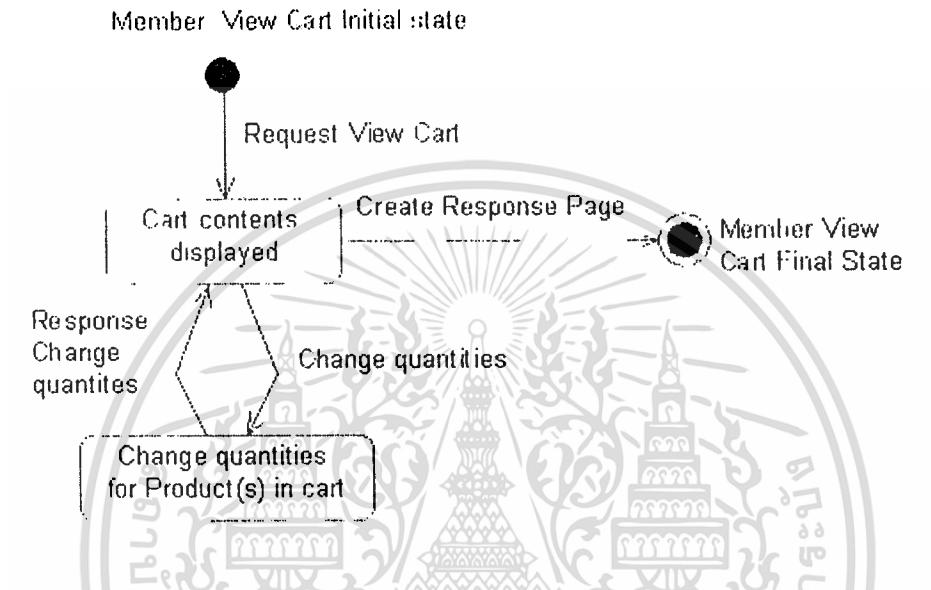
- Add Product To Cart



ภาพที่ 3.6 State Diagram ของ Add Product To Cart

สมาชิกที่ทำการ Login เข้าสู่ระบบแล้วจะสามารถ ทำการค้นหา เช็คนุมูลสินค้า เพื่อเลือกกรวย
 กรวยสินค้าที่จะทำการสั่งซื้อได้ และหากต้องการเพิ่มจำนวนสินค้าก็สามารถทำการเปลี่ยนแปลง
 จำนวนสินค้าที่ต้องการสั่งซื้อได้

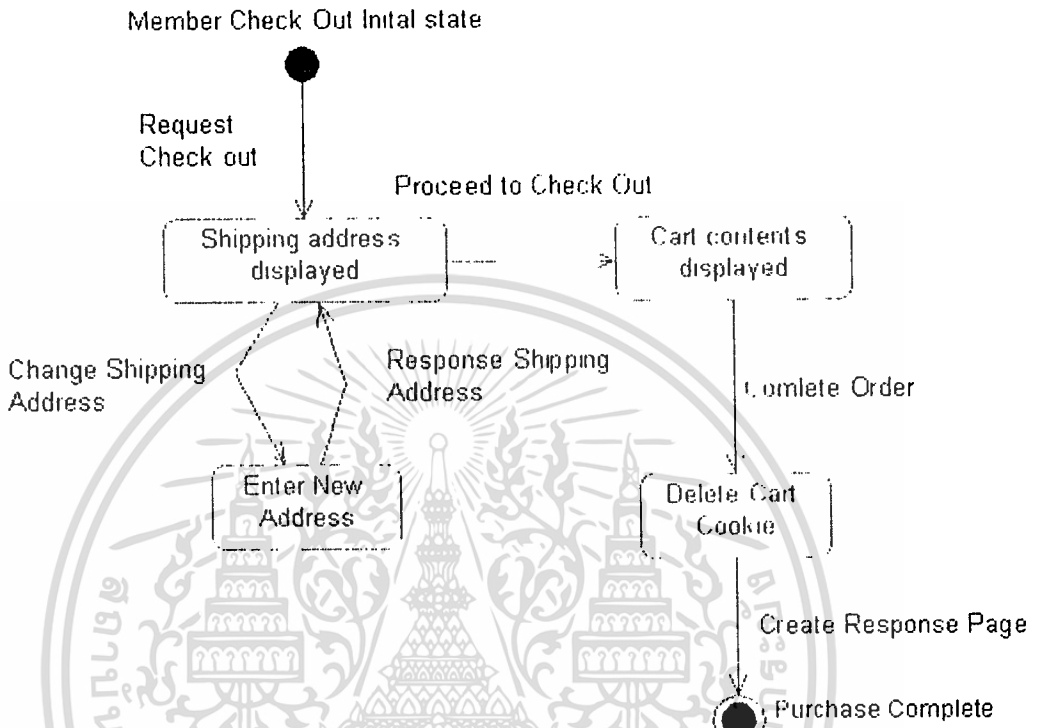
- View Cart



ภาพที่ 3.7 State Diagram ของ View Cart

หลังจากที่สมาชิก Login เข้าสู่ระบบและมีการสั่งซื้อสินค้า สมาชิกสามารถทำการตรวจสอบรายการสั่งซื้อสินค้าและจำนวนสินค้าที่ได้ทำการสั่งซื้อไว้ที่อยู่ในหมายเหตุรถเงินที่สมาชิกได้ร่ำตั้งแต่ Login เข้าสู่ระบบได้ตลอด และหากต้องการแก้ไขเปลี่ยนแปลงจำนวนสินค้าก็สามารถเปลี่ยนแปลงได้

- Check Out



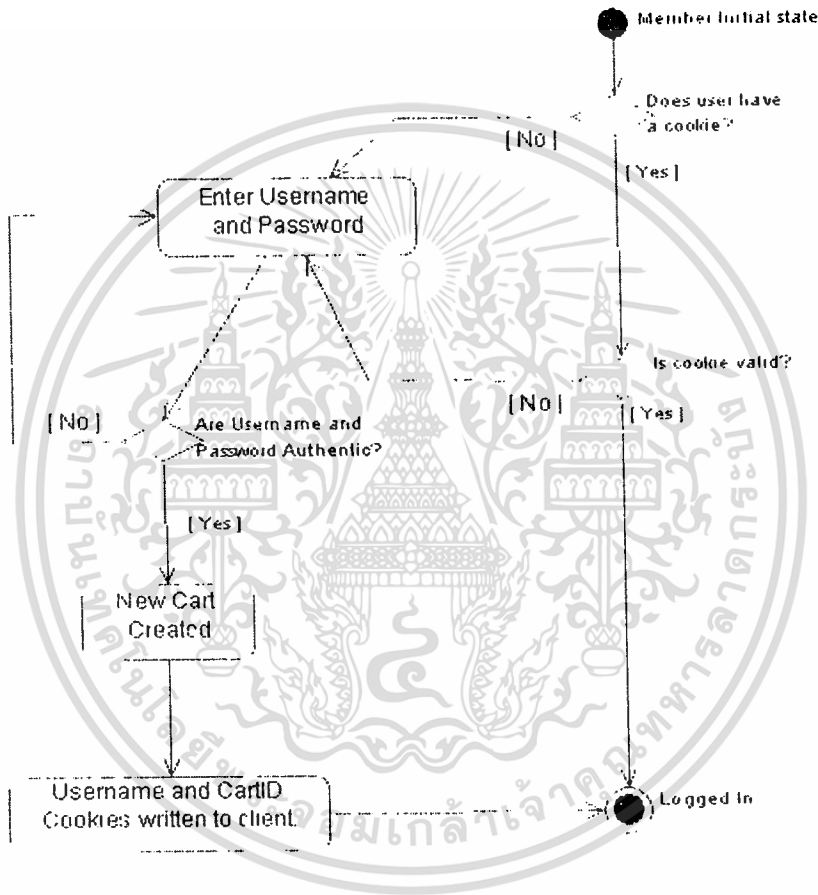
ภาพที่ 3.8 State Diagram ของ Check Out

ขั้นตอนสุดท้าย เมื่อสมาชิกได้ทำการเลือกกรณการสินค้าที่ต้องการเรียบร้อยแล้ว และเมื่อการแสดงผลจากระบบ สมาชิกจะต้องทำการตรวจสอบความถูกต้องของข้อมูลที่สมาชิกต้องจะรับเลยค่า หากสมาชิกต้องการเปลี่ยนแปลงแก้ไขก็สามารถที่จะแก้ไขจนพอใจได้ และระบบจะทำการแสดงข้อมูลรายการสินค้าที่สมาชิกได้ทำการสั่งซื้อทั้งหมดอีกครั้ง เพื่อยืนยันความถูกต้อง และหลังจากที่สมาชิกได้ทำการยืนยันความถูกต้องแล้วระบบจะทำการลบข้อมูลใน Cookie และถ้าหากว่าสมาชิกต้องการสั่งซื้อสินค้าใหม่ก็จะต้องทำการ Login เข้าสู่ระบบใหม่

3.3.4.4 Activity Diagram

ภาพที่ 3.9 เป็น Use Case Diagram ที่ใช้วิธีการออกแบบแบบคอมโพสิชันที่เริ่มแรกของระบบ โดยแสดงกิจกรรมหลักๆ ได้ดังนี้

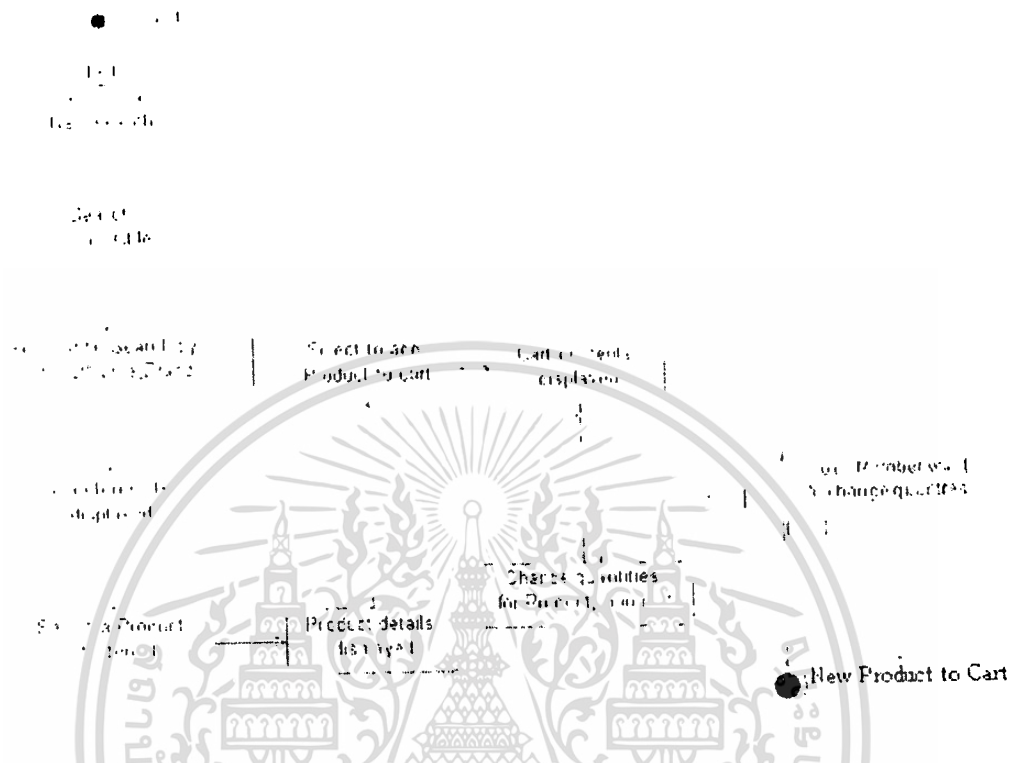
- Login System



ภาพที่ 3.9 Activity Diagram ของ Login System

ถ้ามีผู้เยี่ยมชมเข้ามาที่ Web Site ระบบจะทำการตรวจสอบการทำงานโดย ถ้าเป็น anonymous จะสามารถใช้ระบบได้เพียงค้นหาข้อมูล และดูว่ามีส่วนไหนที่เข้าถึงได้เพียงอย่างเดียวเท่านั้น แต่ถ้าหากเป็น สมาชิกจะต้องทำการ Login ระบบก่อนจึงจะมีสิทธิ์ในการสั่งซื้อสินค้า และดูรายละเอียดของสินค้าได้ โดยระบบจะทำการสร้างรถเข็นให้ หลังจากที่ตรวจสอบ UserName และ Password จาก Active Directory เรียบร้อยแล้วว่าถูกต้อง ซึ่งจะถูกเก็บอยู่ในรูปของ Cookies บนระบบเครื่องของผู้ใช้แล้ว สมาชิกมีการเข้ามาทำการสั่งซื้อสินค้าต่อ ข้อมูลของสินค้าที่เคยสั่งซื้อไป ก็ถูกเก็บไว้ที่เครื่องของผู้ใช้ การใช้บริการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

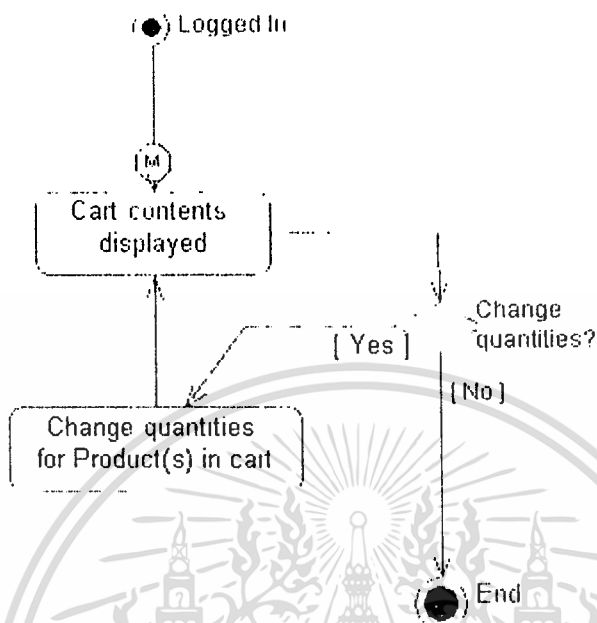
- Add Product To Cart



ภาพที่ 3.10 Activity Diagram ของ Add Product To Cart

หลังจากที่สมาชิก Login เข้ามาในระบบแล้วนั้น จะสามารถดูรายละเอียดของสินค้า และสั่งซื้อสินค้าได้ โดยจะมีการเก็บข้อมูลไว้ในรถเข็นสินค้า ซึ่งระบบได้มีการสร้างไว้ให้หลังจากที่มีการ Login เข้าสู่ระบบ และหากว่าสมาชิกต้องการสินค้ามากกว่า 1 ชิ้น ก็สามารถทำการแก้ไขคำสั่งซื้อที่เปลี่ยนแปลงกันได้ แต่ถ้าหากต้องการยกเลิกสินค้านั้นก็สามารถทำได้ โดยระบุจำนวนสินค้าที่ต้องสั่งเข้าไปใน สุ่มซ์ แล้วทำการบันทึกข้อมูล

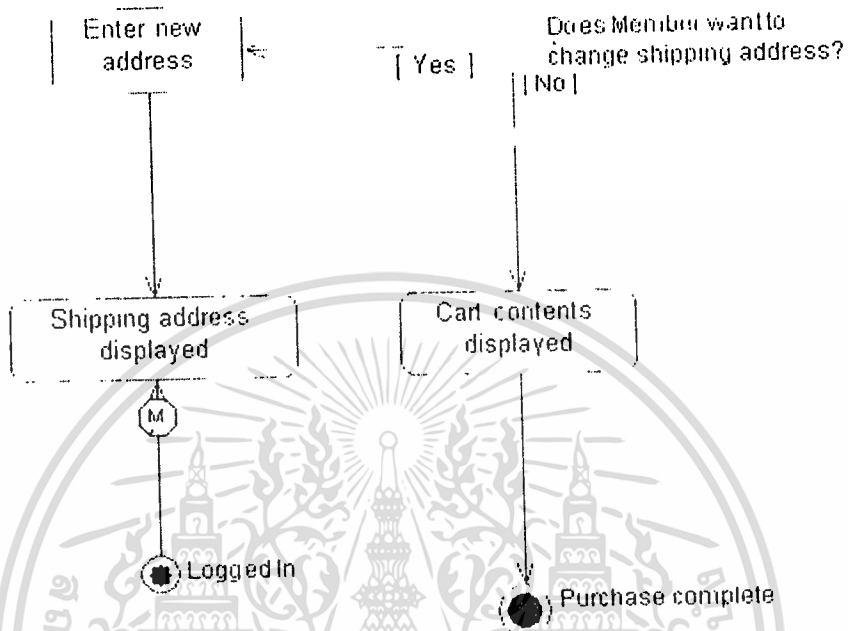
- View Cart



ภาพที่ 3.11 Activity Diagram 1113 View Cart

ถ้าหากต้องกรณข้อมูลการสั่งซื้อว่าสั่งซื้ออะไรไปบ้างเท่าไรก็สามารถที่จะย้อนกลับมาดูได้
 เรายังสามารถแก้ไขยอดการสั่งซื้อได้ แต่ถ้าหากว่าระหว่างที่กำลังเลือกซื้อสินค้าอยู่ แล้วเกิดปัญหา
 ใดๆ ขึ้นมาแล้วทำให้สมาชิกต้องเข้ามาสั่งซื้อสินค้าให้ ระบบจะทำการเก็บข้อมูลก่อนหน้านี้ที่
 สมาชิกได้ทำการสั่งซื้อไว้ ถ้าขึ้นมาให้ ทำให้สมาชิกไม่ต้องทำการเลือกซื้อสินค้าใหม่ตั้งแต่ต้น

- Check Out



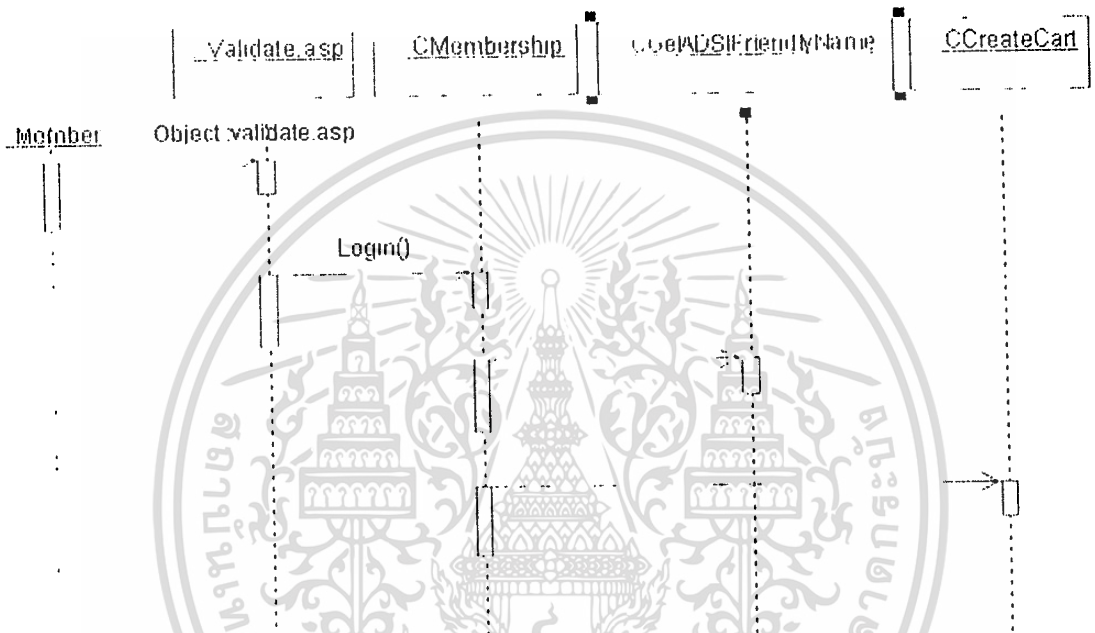
ภาพที่ 3.12 Activity Diagram หน้า Check Out

สุดท้ายถ้าหากลูกค้าได้ทำการเลือกซื้อสินค้าเรียบร้อยแล้ว ระบบจะตามส่งข้อมูลชื่อที่อยู่แบบสมาชิกให้ทำการตรวจสอบ ลึกครั้งเพื่อทำการยืนยันชื่อที่อยู่ที่ถูกตั้ง (มีไว้สำหรับกรณีที่ระบบออกแจ้งการเปลี่ยนแปลงที่อยู่ใหม่) และระบบจะให้สมาชิกตรวจสอบข้อมูลการสั่งซื้อสินค้าที่ระบบได้เลือกชื่อไปอีกครั้ง เพื่อให้สมาชิกทึยืนยันความถูกต้องครั้งสุดท้าย ก่อนที่จะนำข้อมูลไปเก็บแบบเก็บไว้ในฐานข้อมูลการรับตัวสั่งซื้อ

3.3.4.5 Sequence Diagram

ภาพแสดงขั้นตอนการขึ้นตอนการที่เงินเป็นส่วนการเรียกใช้บุคคลที่หนึ่งแล้วจึงขึ้นมาเพื่อให้ระบบที่บนเดมกับขั้นตอนต่างๆ ได้ดังนี้

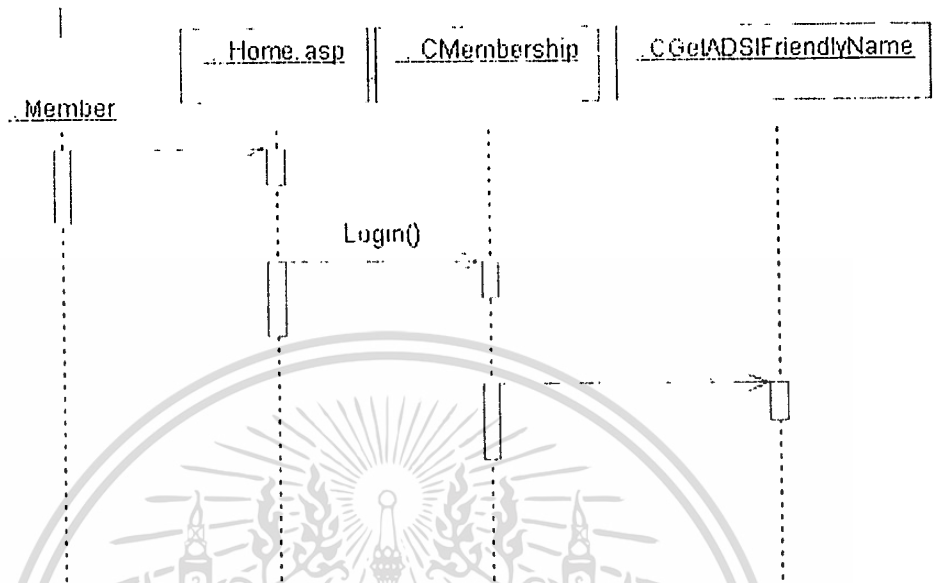
- Login



ภาพที่ 3.13 Sequence Diagram ของ Login

เมื่อสมาชิกทำการ Login เข้าระบบ Validate.asp จะทำการเรียกใช้โปรแกรมชอต Login ของ CMembership แล้วไปขอชอต Login จะทำการเรียกใช้ CGetADSIFriendlyName ซึ่งทำหน้าที่ติดต่อ Active Directory ในการตรวจสอบความถูกต้องของ UserName และ Password ถ้าหากตรวจสอบแล้วถูกต้องก็จะทำการเรียก CCreateCart เพื่อสร้างหมายเลขของรถเล่นส่วนตัว เพื่อนำข้อมูลดังกล่าวเก็บไว้ใน Cookie และ CMembership ก็จะโหลด member.xml หรือ anonymous.xml เพื่อใช้ในการดึงข้อมูลเอกสารของ XML ส่งกลับมาจาก CGetADSIFriendlyName

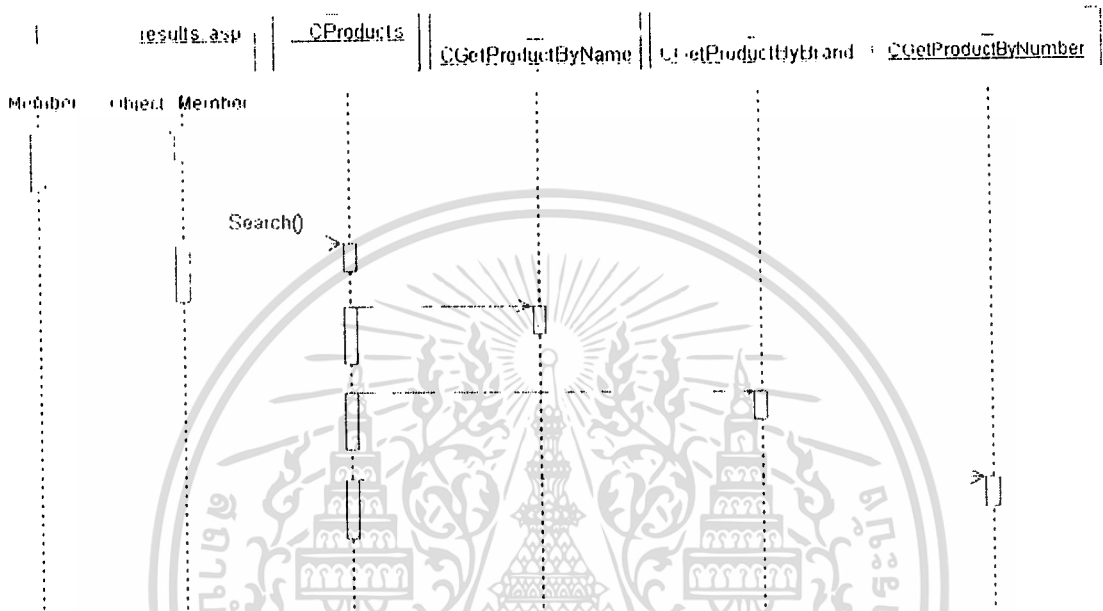
- Cookie



ภาพที่ 3.14 Sequence Diagram ของ Cookie

เมื่อมีผู้เข้ามาเยี่ยมชมเว็บไซต์ และต้องการเข้าใช้งานบางอย่าง ที่ต้องมีการ Login เข้าระบบ ก่อนถึงจะมีการเข้าใช้งานได้ ระบบการทำงานของส่วนนี้ จะทำการตรวจสอบ Cookie ของเครื่อง โคลงแลนค์ ให้อย่างอัตโนมัติ ถ้าเคย Login มาแล้วก็จะสามารถเข้าใช้ระบบได้ทันที โดยเราจะมีการ ตั้ง หมายเหตุภายในของ Cookie เอาไว้ด้วยตามความเหมาะสม โดยระบบจะมีการตรวจสอบความถูกต้องของ Cookie กับ ขั้นตอนในการ Login ว่าจะไม่มีส่วนของการ โหลดเอกสาร XSS. เข้ามาเกี่ยวข้อง

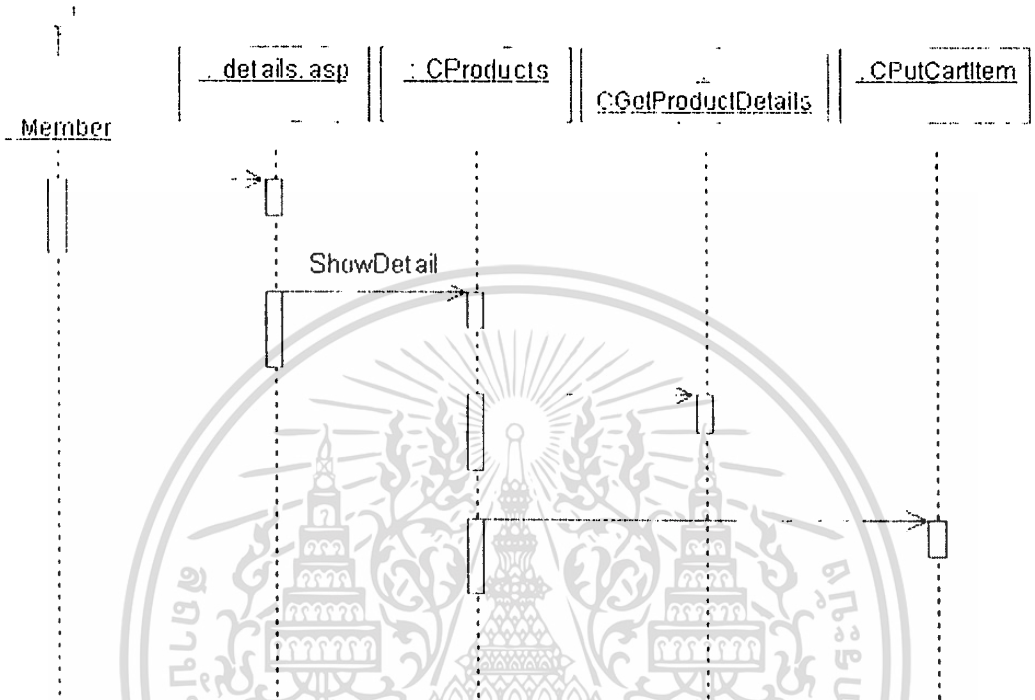
- Search Product



ภาพที่ 3.15 Sequence Diagram ของ Search Product

เมื่อมีผู้เยี่ยมชมเว็บไซต์ ทำการค้นหาข้อมูล ระบบก็จะทำการเรียกใช้ results.asp เพื่อทำการเรียกใช้เมธอด Search จาก CProducts โดย CProducts ก็จะทำการตรวจสอบว่าจะเรียกใช้คลาส CGetProductByName , CGetProductByBrand หรือ CGetProductByNumber ในการค้นหาข้อมูล และ CProducts ก็จะโหลด results.xsl เพื่อใช้ในการจัดรูปแบบเอกสารแบบ XML ส่งกลับมาจาก CGetProductByName , CGetProductByBrand หรือ CGetProductByNumber

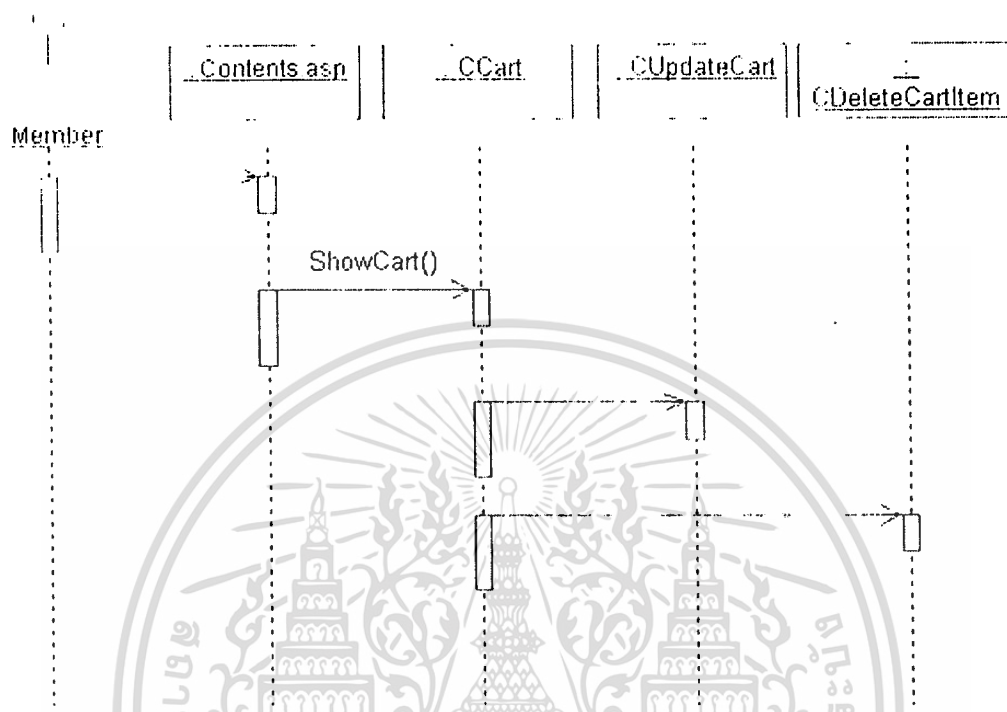
- Add Product



ภาพที่ 3.16 Sequence Diagram ของ Add Product

เมื่อสมาชิกทำการเลือกรายละเอียดของสินค้า ระบบก็จะทำการเรียกใช้ details.asp เพื่อทำการเรียกใช้เมธอด ShowDetail จาก CProducts โดย CProducts ก็จะทำการเรียกใช้คลาส CGetProductDetails ในการดึงข้อมูลรายละเอียดของสินค้า และ CProducts ก็จะโหลด details.xsl เพื่อใช้ในการจัดรูปแบบเอกสารของ XML ส่งกลับมาจาก CGetProductDetails และถ้าหากสมาชิกมีการสั่งซื้อสินค้าลงในรถเข็น CProducts ก็จะทำการเรียกใช้คลาส CPutCartItem เพื่อเก็บข้อมูลสินค้าลงในตาราง CartDetails

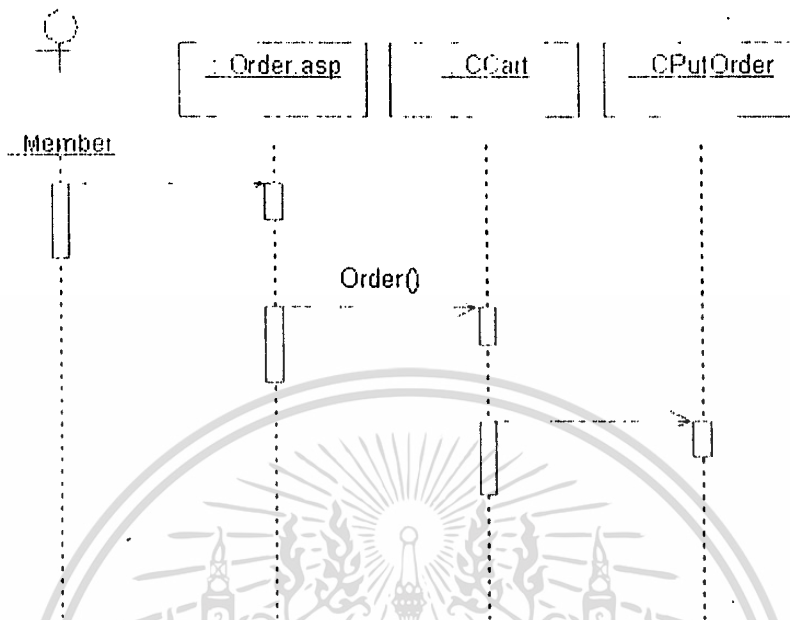
- Add Quantity



ภาพที่ 3.17 Sequence Diagram ของ Add Quantity

เมื่อสมาชิกต้องการเรียกดูข้อมูลในรถเข็นที่ตนเองได้สั่งซื้อไว้ก็สามารถเรียกดูได้ โดยระบบก็จะทำการเรียกใช้ Contents.asp เพื่อทำการเรียกใช้เมธอด ShowCart จาก CCart โดย CCart ก็จะโหลด cart.xml เพื่อใช้ในการจัดรูปแบบเอกสารของ XML ส่งกลับมา และถ้าหากสมาชิกเกิดสั่งซื้อจำนวนสินค้าเพิ่ม CCart ก็จะทำการเรียกใช้คลาส CUpdateCart แต่ถ้าต้องการยกเลิกสิทธิ์ CCart ก็จะทำการเรียกใช้คลาส CDeleteCartItem

- Confirm Order

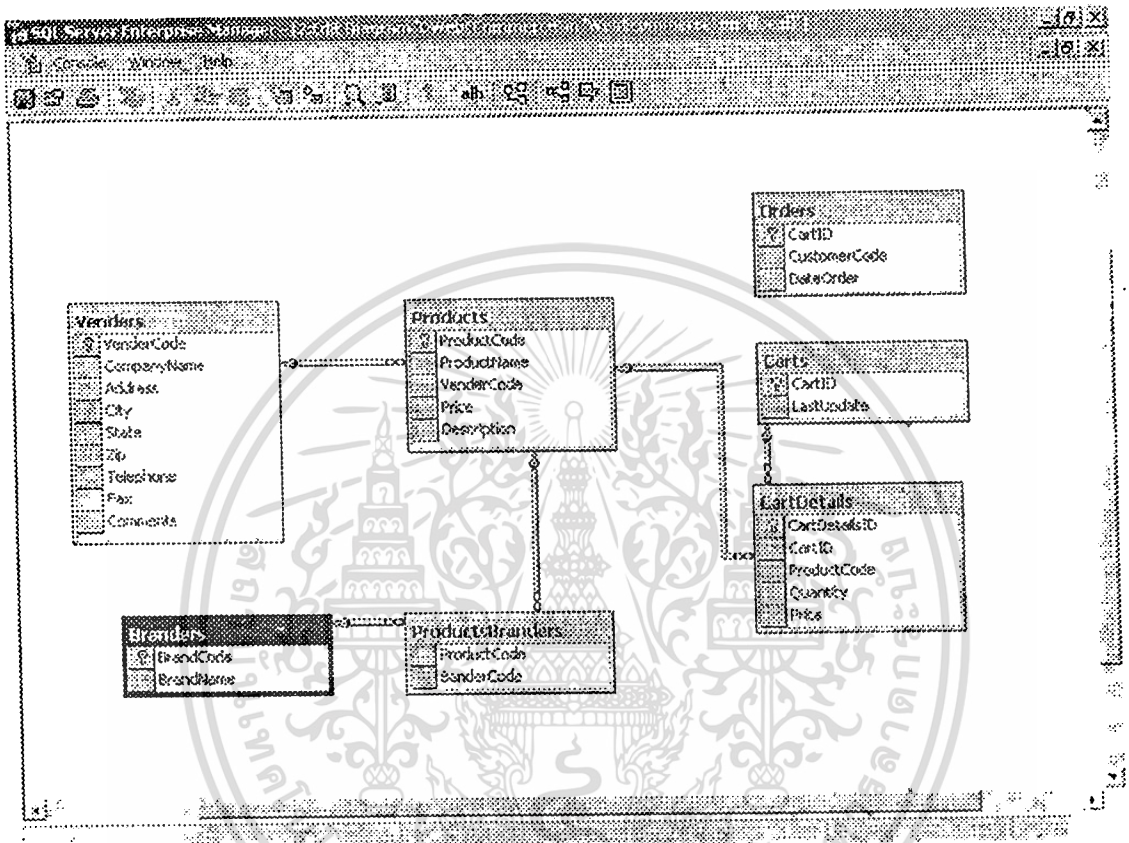


ภาพที่ 3.18 Sequence Diagram ของ Confirm Order

เมื่อสมาชิกต้องการยืนยันการสั่งซื้อสินค้าในรอนเงิน โดยระบบก็จะทำการเรียกใช้ Order.asp เพื่อทำการเรียกใช้เมธอด Order จาก CCart โดย CCart ก็จะโหลด Checkout.xml เพื่อใช้ไปตรวจสอบแบบเอกสารเอง XML ส่งกลับมา และถ้าหากสมาชิกมีการยืนยันจำนวนสินค้าที่ได้ทำการสั่งซื้อแล้วว่าถูกต้อง CCart ก็จะทำการเรียกใช้คลาส CPutOrder โดย CCart ก็จะโหลด Order.xml เพื่อใช้ในการจัดรูปแบบเอกสารของ XML ส่งกลับมา

3.3.5 การออกแบบฐานข้อมูล

เราสามารถออกแบบการทำงานของฐานข้อมูลได้ดังนี้



ภาพที่ 3.19 ตัวอย่างการออกแบบฐานข้อมูล

โดยฐานข้อมูลของเราจะประกอบด้วย ตาราง (Table) ดังต่อไปนี้

- ตาราง Vendors ซึ่งใช้ในการเก็บข้อมูลของคนขายสินค้า
- ตาราง Brands ซึ่งใช้เก็บข้อมูลของยี่ห้อสินค้า
- ตาราง Products ซึ่งใช้เก็บข้อมูลรายละเอียดของสินค้า ราคา และคนขายสินค้า
- ตาราง ProductsBrands ซึ่งใช้เก็บข้อมูลสินค้าที่มีอยู่ เป็นของยี่ห้ออะไรบ้าง
- ตาราง Carts ซึ่งใช้ในการทำการเก็บข้อมูลการสร้างหมายเลขของรถเข็นสินค้า หลังจากที่สมาชิก มีการ Login เข้าสู่ระบบ
- ตาราง CartDetails ทำการเก็บข้อมูลของสินค้านั้นระหว่างที่สมาชิกทำการเลือกซื้อสินค้าลงในรถเข็น

- ตาราง Orders ที่เก็บมีข้อมูลลูกค้าและระยะเวลาของชิ้นงานที่สมาชิกทำการยื่นขออนุญาตสั่งซื้อสินค้าทั้งหมดที่อยู่ในระบบ

โดยตารางดังกล่าวข้างต้น ประกอบด้วยข้อมูลดังต่อไปนี้

ตารางที่ 3.1 ตาราง Venders

Field Name	Data Type	Length	Description	Key
VenderCode	Varchar	10	รหัสผู้ขาย	Primary Key
CompanyName	Varchar	50	ชื่อบริษัท	
Address	Varchar	50	ที่อยู่	
City	Varchar	30	เมือง	
State/Province	Varchar	30	รัฐ/จังหวัด	
Zip	Varchar	10	รหัสไปรษณีย์	
Telephone	Varchar	30	เบอร์โทรศัพท์	
Fax	Varchar	30	แฟกซ์	
Comments	Varchar	50	หมายเหตุ	

ตารางที่ 3.2 ตาราง Branders

Field Name	Data Type	Length	Description	Key
BrandCode	Varchar	10	รหัสยี่ห้อ	Primary Key
BrandName	Varchar	50	ชื่อยี่ห้อ	

ตารางที่ 3.3 ตาราง Products

Field Name	Data Type	Length	Description	Key
ProductCode	Varchar	10	รหัสสินค้า	Primary Key
ProductName	Varchar	50	ชื่อสินค้า	
VenderCode	Varchar	10	รหัสผู้ขาย	
Price	Money	8	ราคา	
Description	Varchar	500	ลักษณะสินค้า	

ตารางที่ 3.4 ตาราง ProductsBranders

Field Name	Data Type	Length	Description	Key
ProductCode	Varchar	10	รหัสสินค้า	Foreign Key
BanderCode	Varchar	10	รหัสยี่ห้อ	Foreign Key

ตารางที่ 3.5 ตาราง Carts

Field Name	Data Type	Length	Description	Key
CartID	Int	4	หมายเลขรถเข็น	Primary Key
LastUpdate	datetime	8	วันที่สร้าง	

ตารางที่ 3.6 ตาราง CartsDetails

Field Name	Data Type	Length	Description	Key
CartDetailsID	Int	4	ลำดับรายการสินค้า	Primary Key
CartID	Int	4	หมายเลขรถเข็น	
ProductCode	Varchar	50	รหัสสินค้า	
Quantity	Int	4	จำนวนสินค้า	
Price	Money	8	ราคาสินค้า	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรณีใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านอื่นๆ

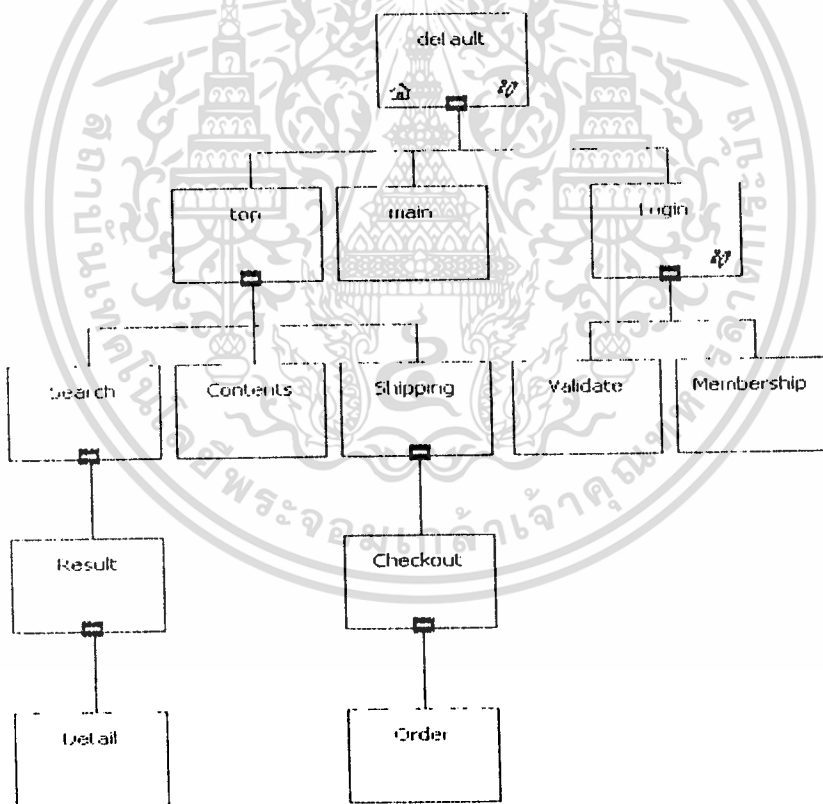
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.7 ตาราง Orders

Field Name	Data Type	Length	Description	Key
CartID	Int	4	หมายเลขรถเข็น	Primary Key
CustomerCode	Varchar	10	รหัสลูกค้า	
DateOrder	Datetime	8	วันที่รับคำสั่งซื้อ	

3.3.6 การออกแบบ Prototype

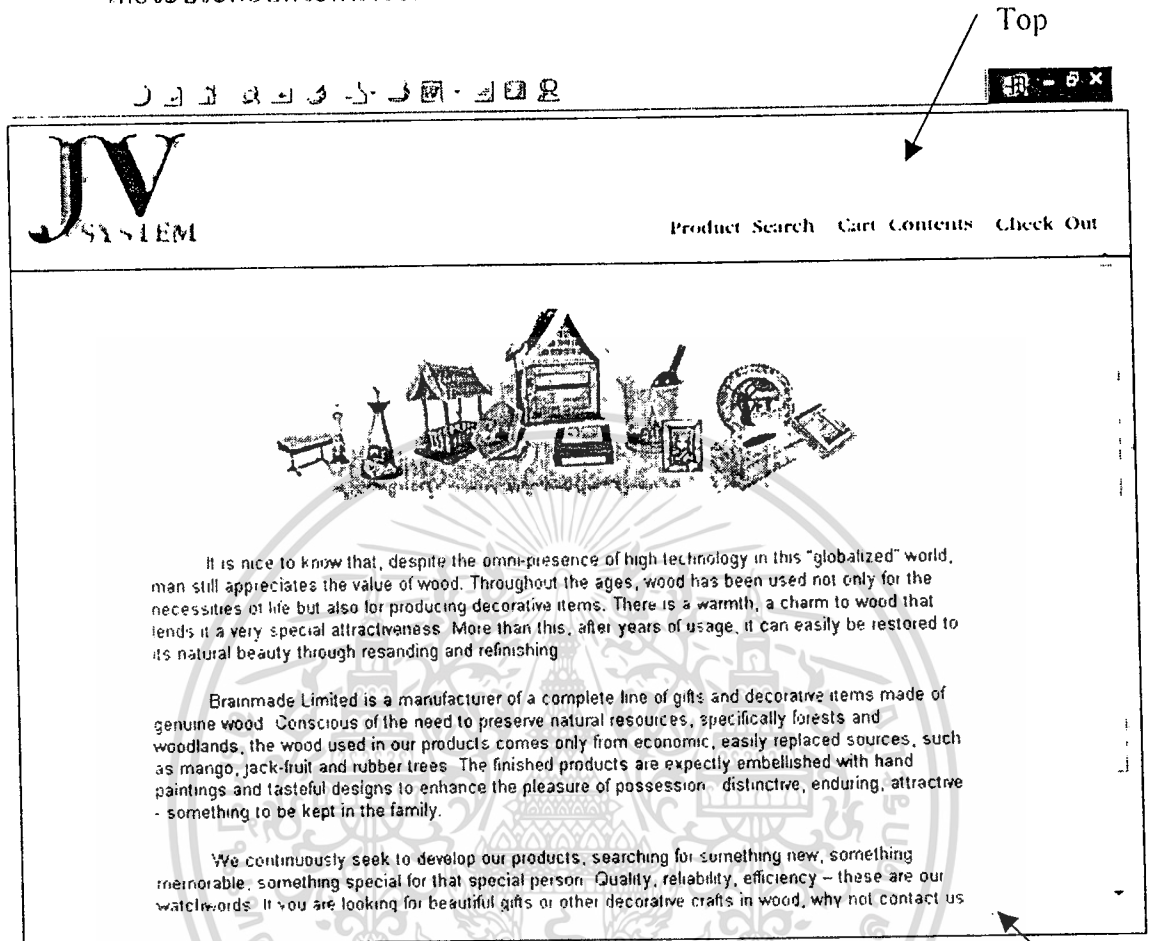
โดยโครงสร้างของเส้นทางการทำงาน ในการออกแบบเว็บไซต์นั้นมี ดังนี้



ภาพที่ 3.20 Site Diagram ของเว็บไซต์ในระบอบพาณิชย์อิเล็กทรอนิกส์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


โดยจะประกอบด้วยหน้าจอการทำงานของโปรแกรมดังนี้



ภาพที่ 3.21 หน้าจอหน้าแรกของเว็บไซต์

หน้าจอแรกของเว็บไซต์นี้ ก็คือหน้าจอ default ซึ่งประกอบไปด้วยเว็บเพจ ดังต่อไปนี้

- Top ซึ่งเป็นเพจที่ทำหน้าที่ในการแสดง Logo และ เมนู ที่อยู่ด้านบนสุดของเพจดังรูปข้างต้น ซึ่งก็คือ Jvsystem ,Product Search ,Cart Contents และ Check Out
- Main ซึ่งเป็นเพจที่แนะนำบริษัท


Cart Contents Check Out

Search [Products] Name | Search

Product Code	Product Name	Vender Company
	clock	brainmade
	lamps	brainmade
	mail boxes	brainmade
	letter holders	brainmade
UJF	mirrors	brainmade
UJG	picture and photo frames	brainmade
	tissue holders	brainmade
UJH	key rings	brainmade
UJI	namecard holders	brainmade
	wine coolers	brainmade

XML+Result.xml

ภาพที่ 3.22 หน้าจอแสดงผลลัพธ์จากการค้นหาข้อมูล

หน้าจอนี้ เป็นหน้าจอที่เกิดจากเรียกใช้ เมนู Product Search ในภาพที่ 3.21 ซึ่งจะทำการเรียกใช้ เพจ Search และเมื่อมีการค้นหาข้อมูล ระบบจะทำการเรียกใช้งานเพจ Result เพื่อแสดงผลลัพธ์ของสินค้าที่ต้องการค้นหาออกมา โดยผลลัพธ์ที่ได้แสดงอยู่ที่หน้าจอนี้ เกิดจากรูปแบบของข้อมูลที่เป็น XML ซึ่งทำงานร่วมกับ Rresult.xml ซึ่งแสดงผลออกมาในรูปของตาราง ดังรูป 3.22



Product Search Cart Contents

UserName :

Password :

ภาพที่ 3.24 หน้าจอรับอินพุต ของการ Login

จากรูปที่ 3.23 เมื่อสมาชิกทำการเลือก Login เข้าสู่ระบบ ระบบจะทำการเรียกใช้เพจ Login ขึ้นมาใช้สมาชิกทำการป้อนข้อมูล UserName และ Password ดังในภาพที่ 3.24 เมื่อ มีการกดปุ่ม Submit ระบบจะทำการดึงเพจ Validate เพื่อทำการตรวจสอบความถูกต้องกับ Active Directory ของ Windows 2000 Servers



XML+Member.xml

Product Search Cart Contents Check Out

Welcome, Noi Kiatchavanasawee

If you are not Noi Kiatchavanasawee, then please



ภาพที่ 3.25 หน้าจอแสดงผลการตอบรับหลังจาก Login สำเร็จ

ถ้าหาก UserName และ Password ของสมาชิกถูกต้อง ระบบจะทำการแสดงผลพัช์ดังที่แสดงในภาพที่ 3.25 โดยระบบจะทำการดึงข้อมูลของชื่อที่ใช้ UserName ที่ได้ทำการ Login เข้าสู่ระบบจากตัว Active Directory ของ Windows 2000 Server นั้นมาให้ดูว่าถูกต้องหรือไม่ ซึ่งผลพัช์ที่ได้แสดงอยู่ที่หน้าจอนี้ เกิดจากรูปแบบของข้อมูลที่เป็น XML ซึ่งทำงานร่วมกับ Member.xml



XML+Details.xml

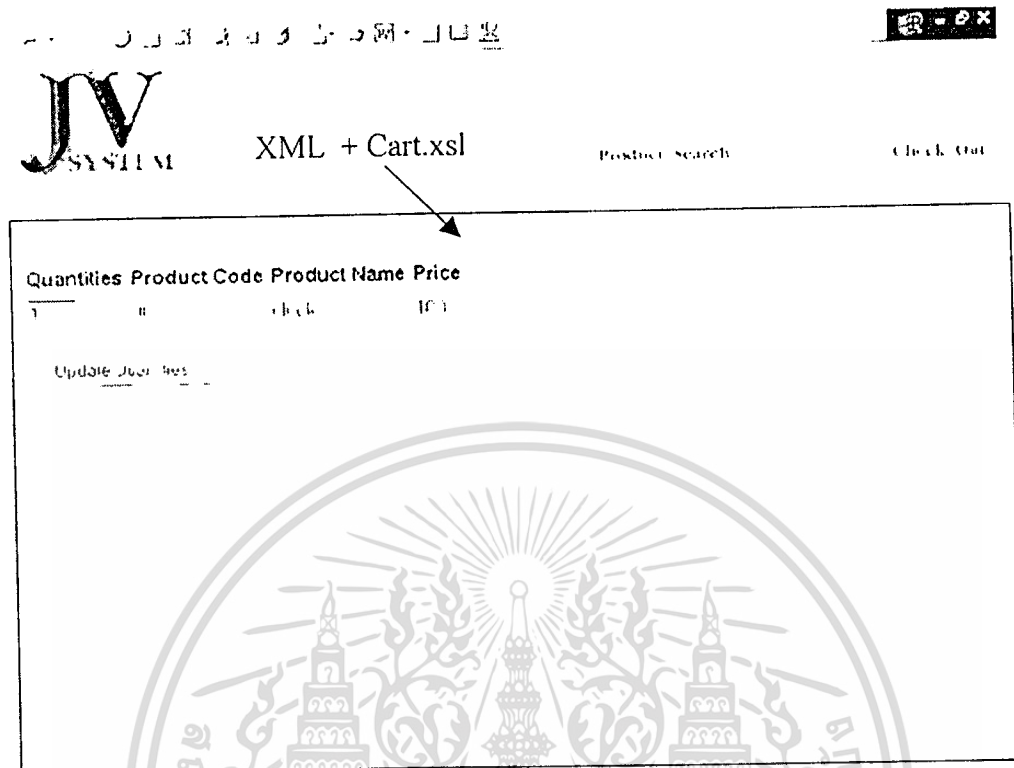
Cart Contents Check Out

Product Code	Product Name	Product Description	Price
001	clock	A wide selection of round wooden clocks made of several types of typical Thai wood hand-painted in acrylic or oil	100

Add to Cart

ภาพที่ 3.26 หน้าจอแสดงรายละเอียดของสินค้า

หลังจากที่ สมาชิกได้ทำการ Login เข้าสู่ระบบแล้วสมาชิกก็จะสามารถดูรายละเอียดของสินค้านั้นได้ต่อจากภาพที่ 3.22 พร้อมทั้งราคา โดยระบบจะทำการเรียกใช้เพจ Detail เพื่อใช้ในการดึงข้อมูลของรหัสสินค้านั้น ซึ่งผลลัพธ์ที่ได้แสดงอยู่ที่หน้าจอนี้ เกิดจากรูปแบบของข้อมูลที่เป็น XML ซึ่งทำงานร่วมกับ Details.xml และถ้าหากสมาชิกต้องการสั่งซื้อสินค้าก็สามารถกด Add to Cart เพื่อเพิ่มรายการการสั่งซื้อสินค้าลงในหมายเลขรถเข็นที่ได้รับจากการ Login เข้าสู่ระบบ



ภาพที่ 3.27 หน้าจอแสดงข้อมูลสินค้าที่อยู่ในรถเข็น

หลังจากที่มีการเพิ่มรายการสินค้าลงในหมายเลขรถเข็น จากภาพที่ 3.26 ระบบก็จะทำการเรียกใช้เพจ Contents แสดงหน้าจอของรายการสินค้าที่อยู่ในหมายเลขรถเข็นนั้นทั้งหมด โดยสมาชิกสามารถทำการแก้ไขจำนวนรายการสินค้าที่ต้องการได้ ซึ่งผลลัพธ์ที่ได้แสดงอยู่ที่หน้าจอนี้ เกิดจากรูปแบบของข้อมูลที่เป็น XML ซึ่งทำงานร่วมกับ Cart.xml

Product Search - Cart Contents



Product Search - Cart Contents

Company	Thailand
Address	951/22 Plo
City	Bangkok
State	10
Zip	10240
Update Address	

ภาพที่ 3.28 หน้าจอแสดงที่อยู่ของลูกค้า

สุดท้าย หากสมาชิกต้องการออกจากระบบและต้องการยืนยันการสั่งซื้อสินค้า ระบบจะทำการเรียกใช้เพจ Shipping เพื่อแสดงข้อมูลสถานที่ส่งสินค้าให้กับสมาชิกดู ซึ่งผลลัพธ์ที่ได้แสดงอยู่ที่หน้าจอนี้ เกิดจากรูปแบบของข้อมูลที่เป็น XML ซึ่งทำงานร่วมกับ Address.xml โดยมีไว้ในกรณีที่สมาชิกอาจมีความต้องการ เปลี่ยนแปลงสถานที่ส่งสินค้าใหม่ และถ้าต้องการดำเนินการยืนยันการสั่งซื้อต่อก็ ให้ทำการคลิกที่ลิงก์ Proceed to CheckOut ก็จะแสดงหน้าจอดังภาพที่ 3.29

บทที่ 4

สรุป

จากการศึกษาการใช้งาน COM+ Application และการทำงานของระบบการรักษาความปลอดภัย ร่วมกับ Windows Web Solution นั้น ซึ่งพบว่าระบบมีความยืดหยุ่นในการทำงานสามารถเลือกเชื่อมโยงการทำงานกับระบบฐานข้อมูลอื่น ๆ ที่ต้องการได้ อย่างเช่น Oracle หรือ SQL Server หรือแม้กระทั่งการเชื่อมต่อกับตัว Active Directory ของตัว Windows 2000 Server เพื่อใช้ความสามารถของการรักษาความปลอดภัย

โดยรูปแบบของ COM Component ที่พัฒนาในระบบนี้ จะเป็นการมุ่งเน้นให้เห็นถึงการนำไปใช้งานร่วมกับตัว COM+ Application ในลักษณะของการให้บริการในการเชื่อมต่อระบบ โดยเราสามารถเลือกระบบที่จะทำการเชื่อมต่อได้ โดยการเซตค่าของ Construction String ในถูกต้อง ซึ่งสามารถดูวิธีเซต และรูปแบบการเซตค่าได้ที่ ภาคผนวก ก และที่สำคัญที่เรามีการพัฒนาด้วยการใช้เทคโนโลยี COM+ นั้น เพราะตัว COM+ จะเป็นตัวช่วยให้ง่ายในการพัฒนาระบบให้แยกออกเป็น N-Tier ได้ ซึ่งสามารถดูวิธีการเชื่อมต่อในการเรียกใช้งานของ COM+ ข้ามเครื่องได้ที่ ภาคผนวก ข

บรรณานุกรม

สุวัฒนา สุขสมจินตน์ ,ผู้แปล 2545. คัมภีร์การใช้งาน XML ฉบับสมบูรณ์. กรุงเทพฯ : ซีเอ็ดยูเคชั่น .

A. Russell Jones. 1999. **Better Web Data Access** : [Online]. Available: <http://www.vbpi.com>

Dan Fox. 1999. **Directory-Enable Apps With ADSI** : [Online]. Available:

<http://www.vbpi.com>

Microsoft COM+ site : [Online]. Available: <http://microsoft.com/com/tech/complus.asp>

Microsoft Windows DNA site : [Online]. Available: <http://www.microsoft.com/dna>

Microsoft MDAC site : [Online]. Available: <http://www.microsoft.com/data>



ภาคผนวก ก การติดตั้งระบบ

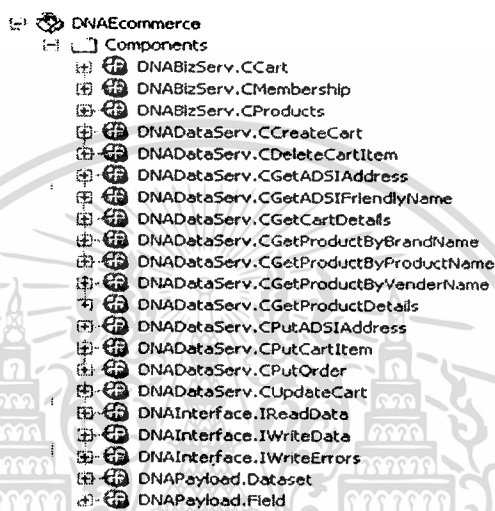
หากมีความต้องการนำระบบนี้ไปทำการติดตั้งใช้งาน สิ่งที่ต้องจัดเตรียม และเซตเพิ่มเติมมีดังนี้

1. ในเรื่องของ ซอร์แวร์ที่จะต้องจัดเตรียม

- Windows 2000 Server โดยต้องทำการเซตชื่อเครื่องเป็น DomainMe และชื่อ Domain เป็น WebDomain แต่ถ้าหากเครื่อง Server ของท่าน ไม่ได้เซตตามนี้ ท่านจะต้องเปลี่ยนแปลงค่า Construction String ของตัว Component Service ใหม่ คู่มือการเซตได้ในข้อ 3
- Component Service ซึ่งสามารถดูรายละเอียดเพิ่มเติมได้ที่บทที่ 2
- Internet Service Manager หรือ ตัว IIS 5.0 ขึ้นไป
- MDAC 2.7 ซึ่งใช้ในการเข้าถึงข้อมูลประเภทต่างๆ สามารถดูรายละเอียดเพิ่มเติมได้ที่ บทที่ 2

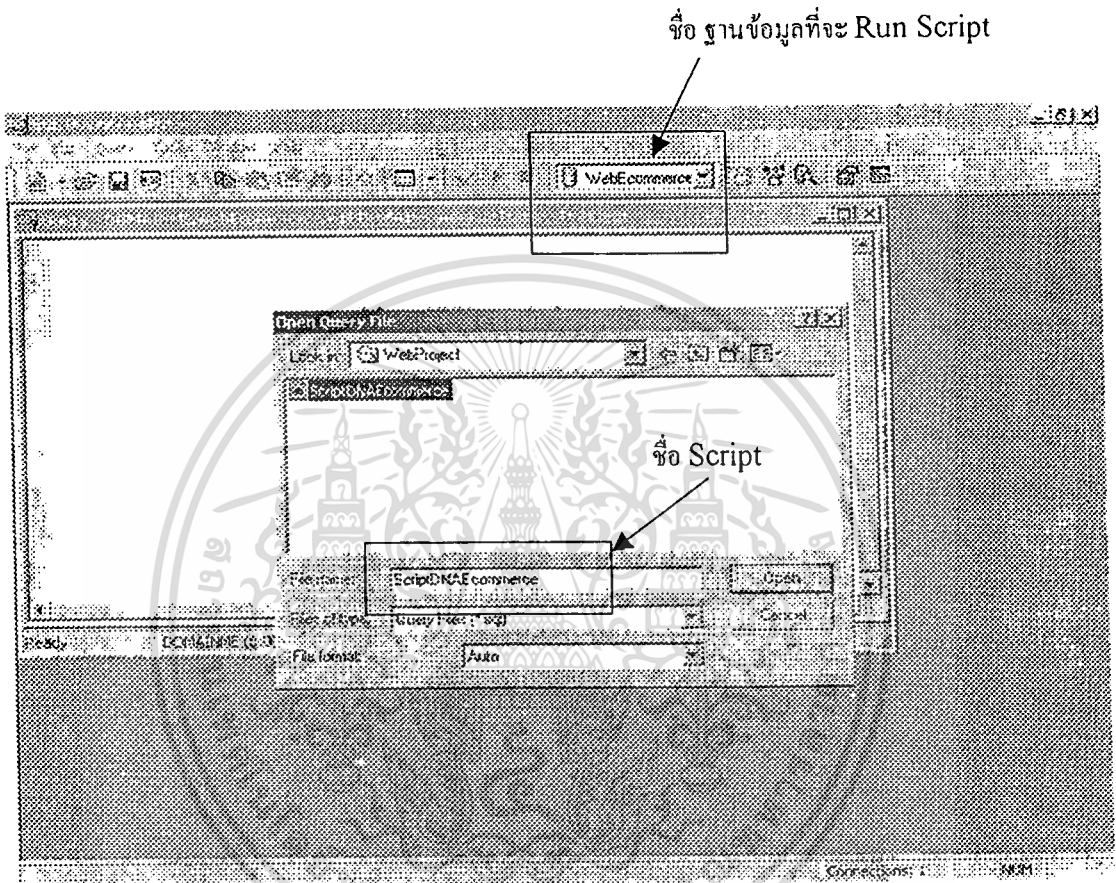
1. ขั้นตอนการติดตั้งโปรแกรม COM+ Server Application

- a. เราได้มีการทำตัวติดตั้งระบบไว้ชื่อ SevDNAECommcrc ซึ่งตัวติดตั้งนี้จะทำการติดตั้งตัว Component Service ที่ชื่อว่า DNAECommerce ให้อย่างอัตโนมัติ ดังภาพที่ ก.1



ภาพที่ ก.1 Component Service ของ DNAECommerce

- b. ทำการสร้างฐานข้อมูลชื่อ WebEcommerce แล้วทำการ Run Script ที่ชื่อ ScriptDNAEcommerce.sql ที่ SQL Query Analyzer ในตัวของ SQL Server 2000 ดังภาพที่ ก.2



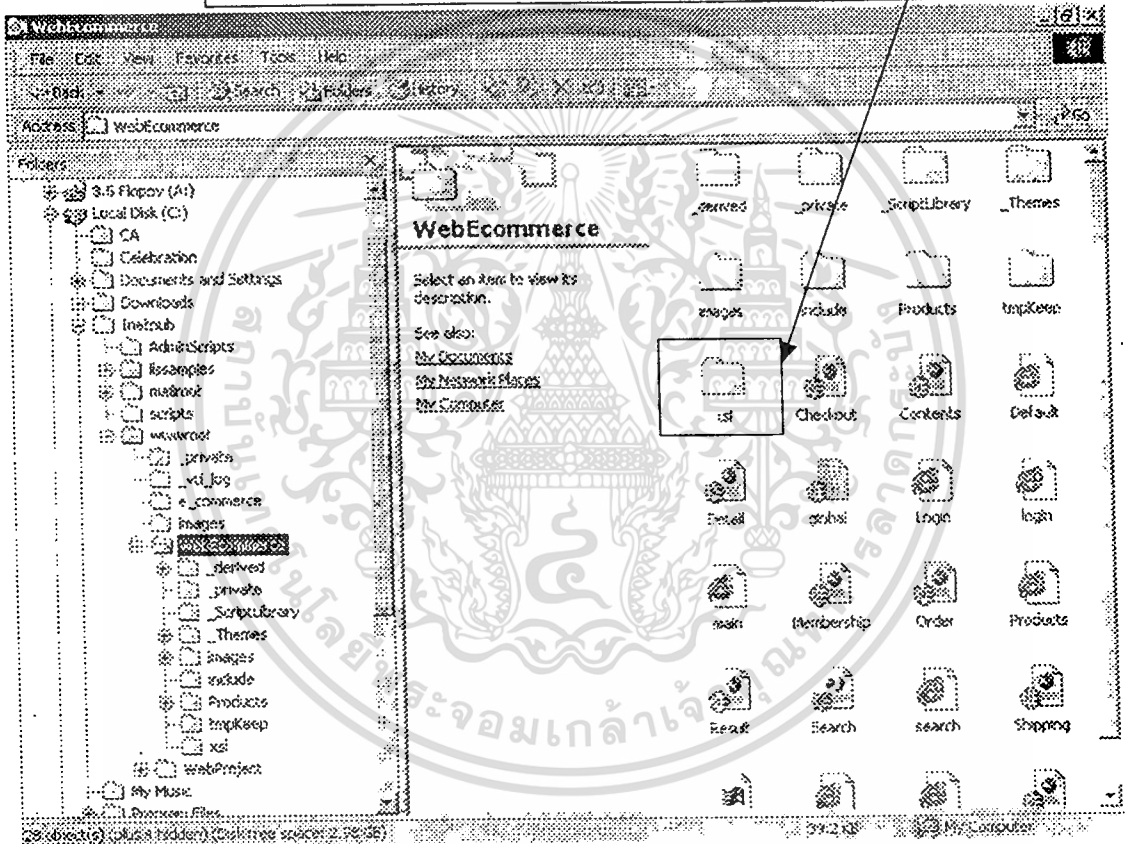
ภาพที่ ก.2 การ Run Script ใน SQL Server 2000

- c. ทำการติดตั้ง เพจ ของเว็บลงไปใน IIS 5.0 โดย Copy ไฟล์ทั้งหมดที่อยู่ใน WebEcommerce_Local ลงในระบบของท่านตาม Path directory ดังนี้

C:\Inetpub\wwwroot\WebEcommerce\...

แต่ถ้าหากไม่ได้เซต Path directory ตามนี้ท่านจะต้องทำการแก้ไข Construction String ในตัว Component Service ใหม่ เพื่อใช้ในการอ้างอิง Path directory ที่เก็บรูปแบบของ XSL เอาไว้

C:\Inetpub\wwwroot\WebEcommerce ที่ใช้อ้างอิง XSL ในการติดตั้ง

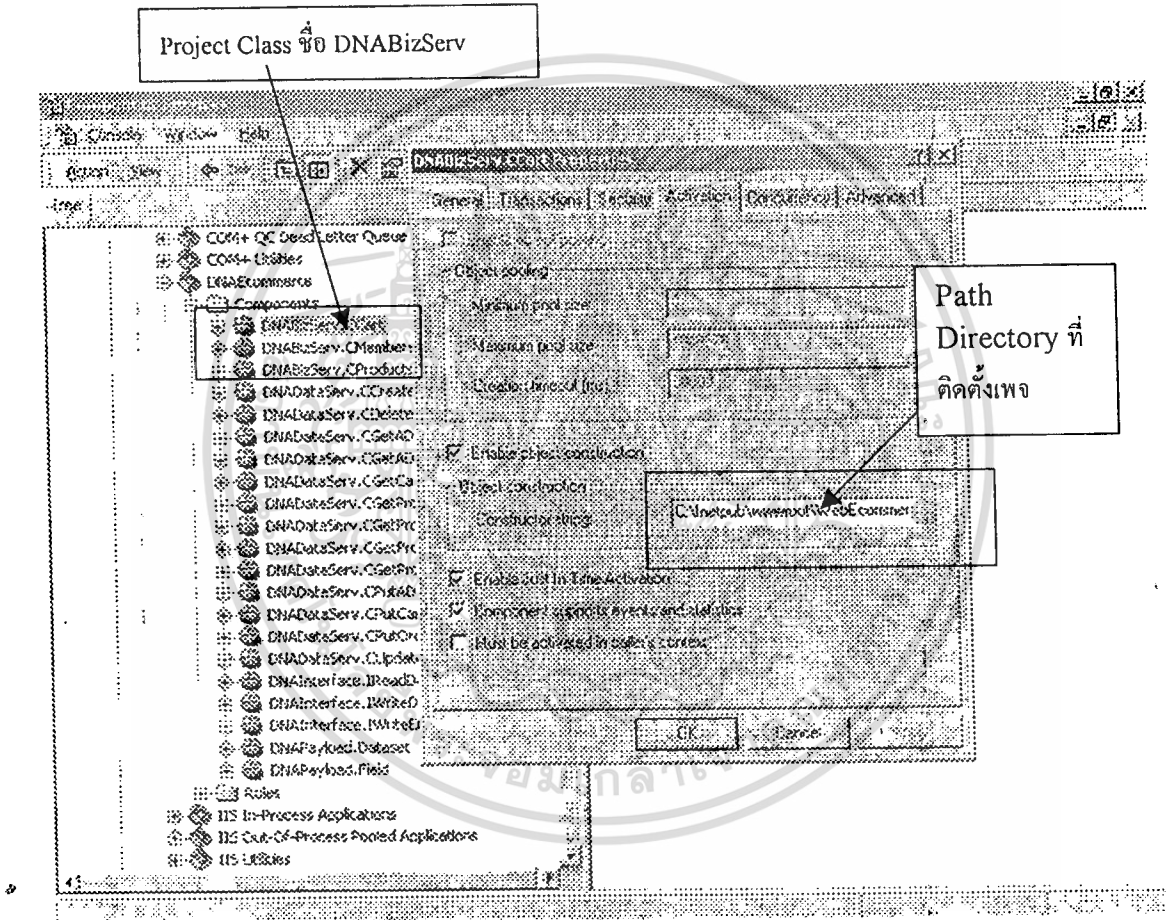


ภาพที่ ก.3 ข้อมูลที่ต้องนำไปติดตั้งใน IIS 5.0

2. วิธีการเซตค่า Construction String

โดยเราจะแบ่งการ เซตค่า Construction String เป็น 3 ส่วนคือ

- Project Class ที่ชื่อ DNABizServ เราจะต้องเซตค่าให้ตรงกับ Path Directory ที่เราได้ทำการติดตั้ง เพจ เอาไว้ ดังภาพที่ ก.4

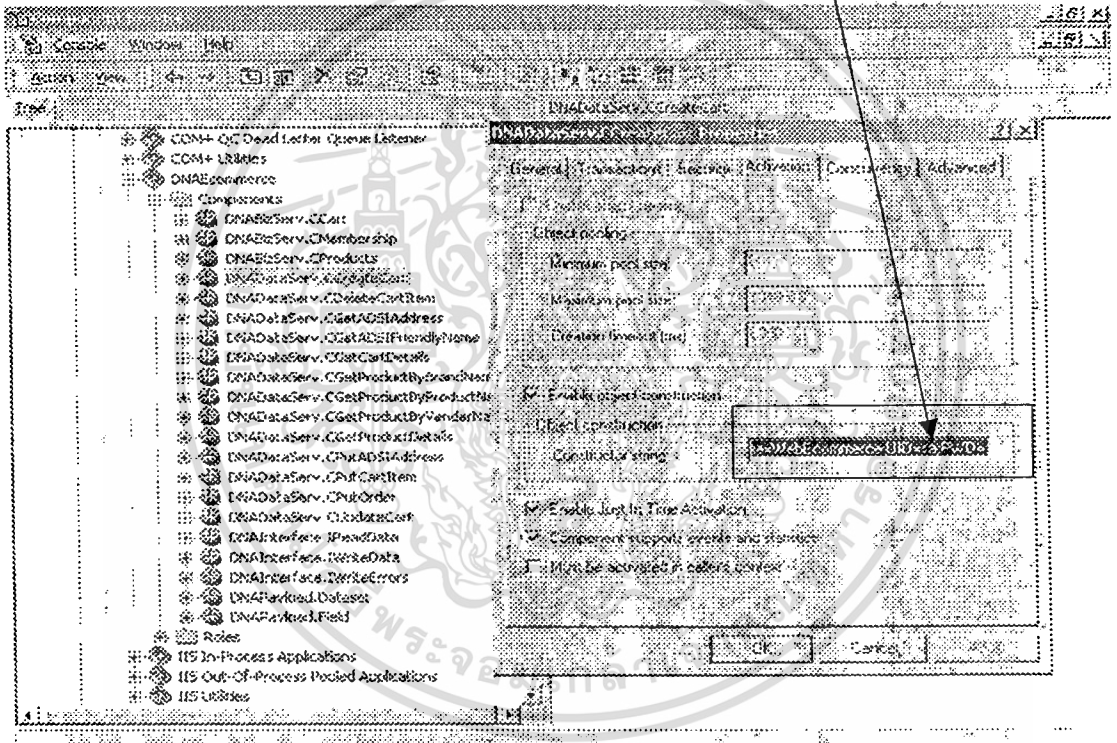


ภาพที่ ก.4 การเซตค่า Construction String ของ DNABizServ

b. Project Class ที่ชื่อ DNADDataServ เราจะต้องเซตค่าไว้ตรงกับ Path ที่เราเรียกใช้งาน ซึ่งเราจะแบ่งเป็น 2 ส่วนคือ

- ส่วนที่เรียกใช้งานกับฐานข้อมูล SQL Server 2000 ซึ่งเราจะต้องเซต Path ให้ถูกต้อง ดังภาพที่ ก.5 โดยPath เดิมที่เซตเอาไว้คือ

```
Provider=MSDataShape;DataProvider=SQLOLEDB;
Server=(local);Database=WebEcommerce
```



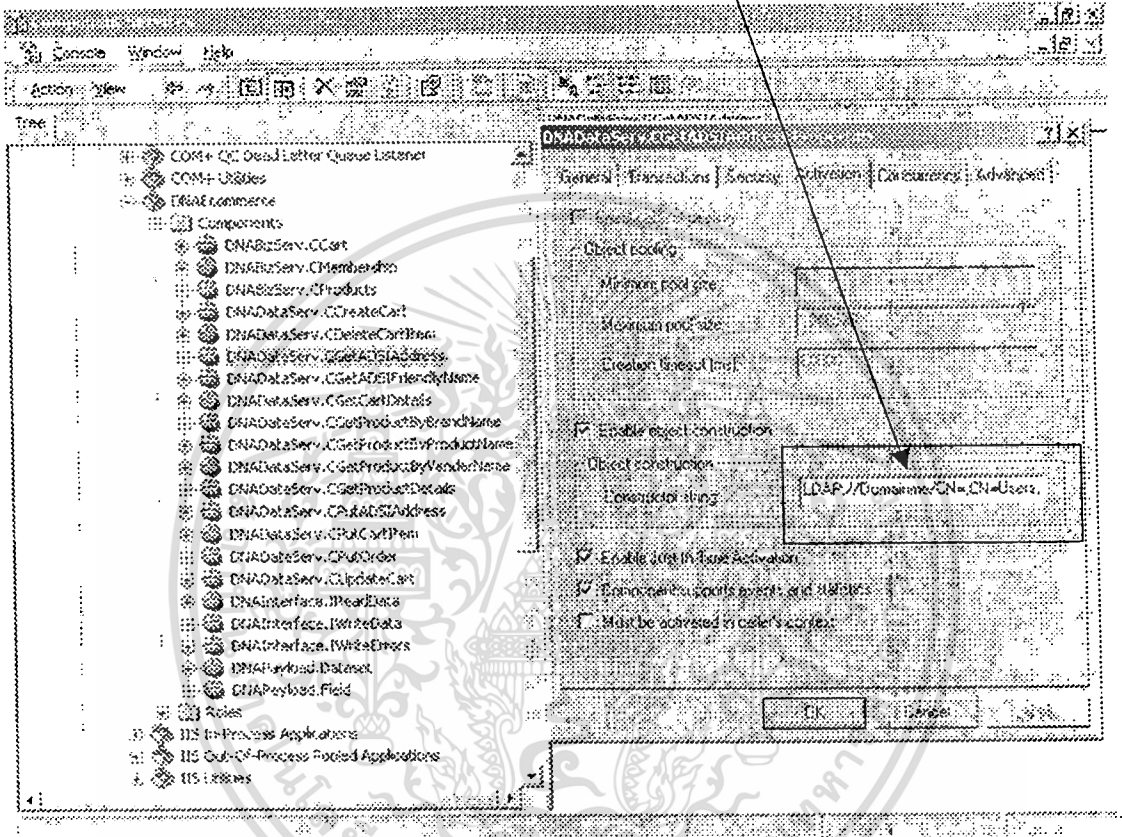
ภาพที่ ก.5 การเซตค่า Construction String ของ DNADDataServ ที่ติดต่อกับ SQL Server

หาก มีการเปลี่ยนแปลง ชื่อฐานข้อมูล หรือชื่อ Server ของฐานข้อมูล ก็จะต้องทำการแก้ไข Path ให้ถูกต้องตามรูปแบบต่อไปนี้

```
Provider=MSDataShape;DataProvider=SQLOLEDB;
Server=ชื่อ Server ของฐานข้อมูล;Database=ชื่อฐานข้อมูล
```

- ส่วนที่เรียกใช้งานกับ Active Directory ซึ่งเราจะต้องเซต Path ให้ถูกต้อง ดังภาพที่ ก.6 โดย Path เดิมที่เซตเอาไว้คือ

LDAP://Domainmc/CN=,CN=Users,DC=WebDomains,DC=com



ภาพที่ ก.6 การเซตค่า Construction String ของ DNADDataServ ที่ติดต่อกับ Active Directory

หาก มีการเปลี่ยนแปลง ชื่อเครื่องของ Server หรือชื่อ Domain ของ Server ก็จะต้องทำการแก้ไข Path ให้ถูกต้องตามรูปแบบต่อไปนี้

LDAP://ชื่อของเครื่อง/CN=,CN=Users,DC=ชื่อของ Domain ,DC=com

สามารถศึกษาเพิ่มเติมได้ที่บทที่ 2

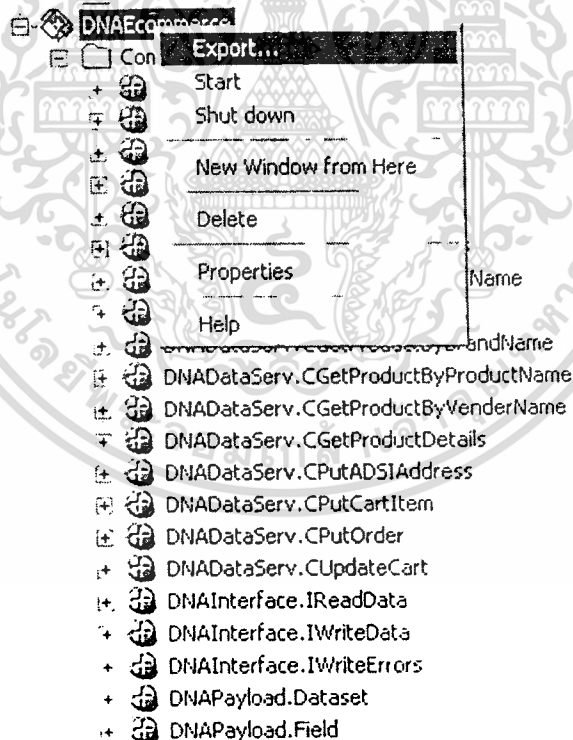
ภาคผนวก ข

การสร้างระบบเชื่อมต่อระหว่าง Server ด้วย COM+ Application

หากเราต้องแยกระบบที่พัฒนาที่ด้วยตัว COM+ Application ออกเป็น Tier ย่อยๆ ในขนาด เราสามารถทำได้โดย การสร้าง Server Application และ Proxy Application จากตัว COM+ Application ที่เราได้ทำการพัฒนา ซึ่งมีขั้นตอนการสร้างดังนี้

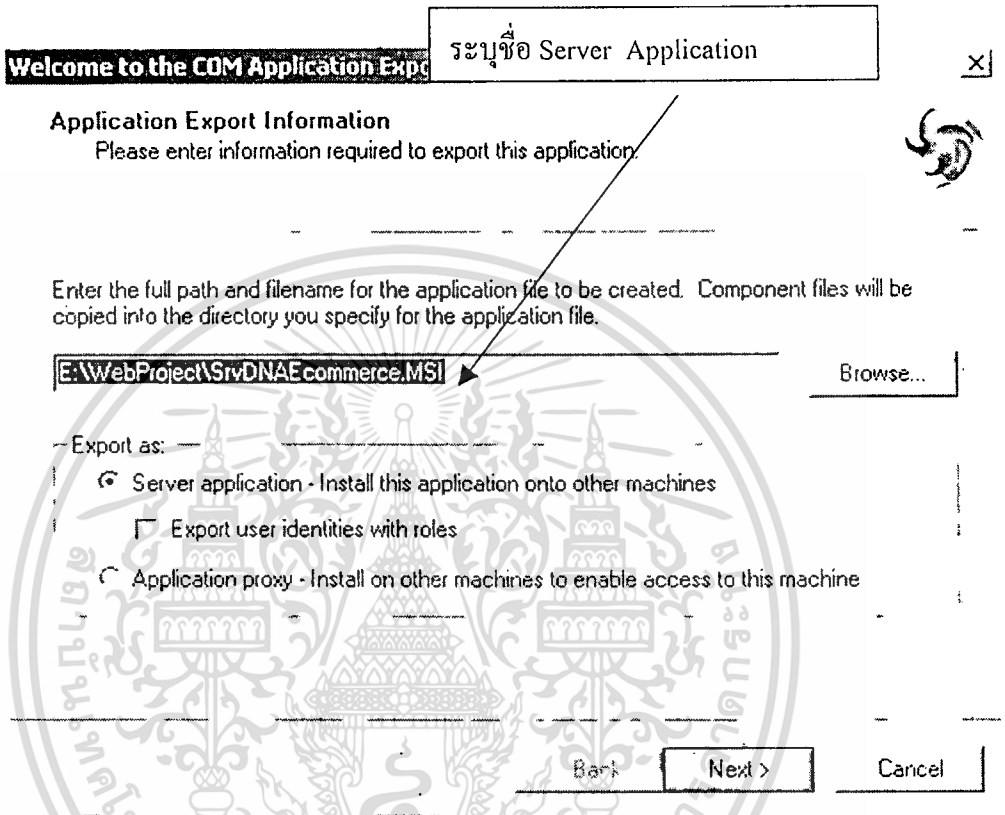
- วิธีการสร้าง Server Application

- เปิดโปรแกรม Component Service แล้วคลิกขวาที่ DNAEcommerce > Export ดังภาพที่ ข.1



ภาพที่ ข.1 การเลือก Export ตัว Server Application

- ทำการเลือก Exports as : Server Application และระบุชื่อ Server Application ที่ต้องการสร้าง เพื่อใช้ในการนำไปติดตั้งที่เครื่อง Server ตัวอื่น ดังภาพที่ ข.2

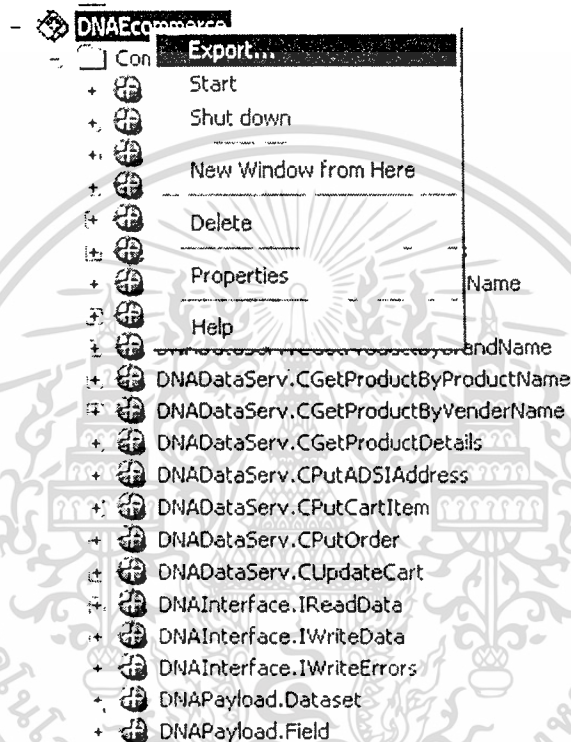


ภาพที่ ข.2 การสร้าง Server Application

- หลังจากนั้นก็นำ ไฟล์ที่ได้ไปติดตั้งที่ เครื่องเซิร์ฟเวอร์ที่ต้อง ซึ่งสามารถดูรายละเอียดการเซตค่าได้ที่ ภาคผนวก ก

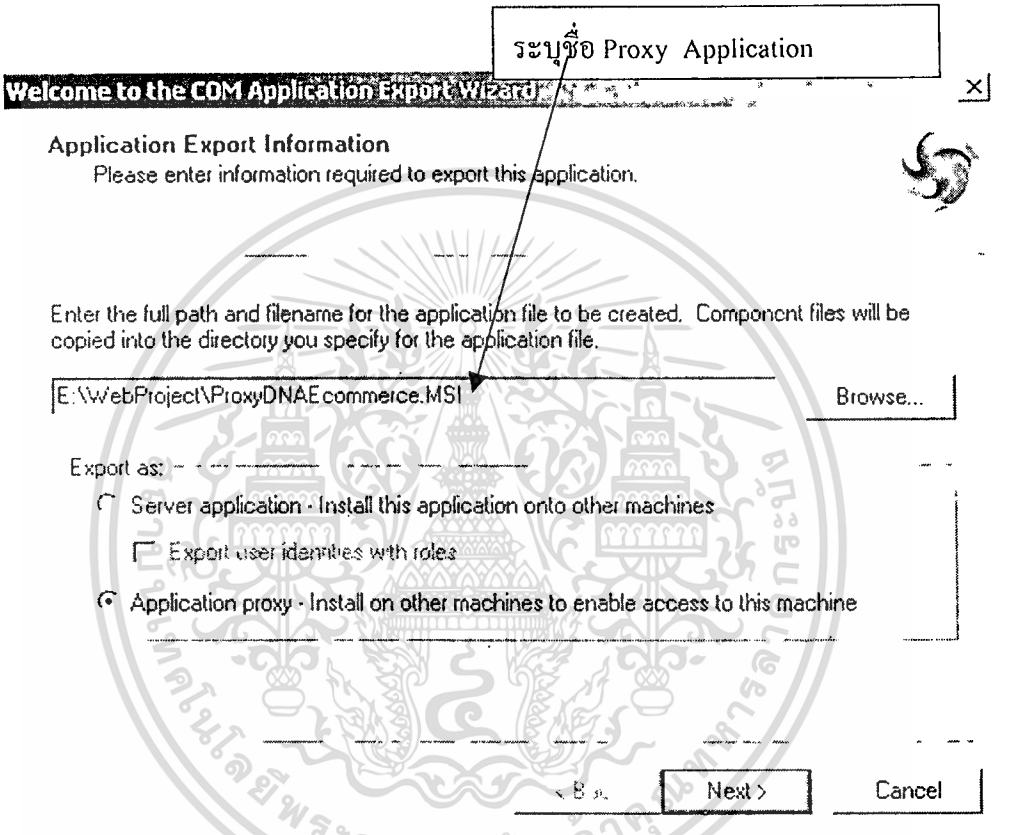
- วิธีการสร้าง Proxy Application

- ให้ทำการสร้าง Proxy Application ที่เครื่อง Server ที่ต้องการให้ระบบทำการเชื่อมต่อด้วยเท่านั้น โดยเปิดโปรแกรม Component Service แล้วคลิกขวาที่ DNAEcommerce > Export



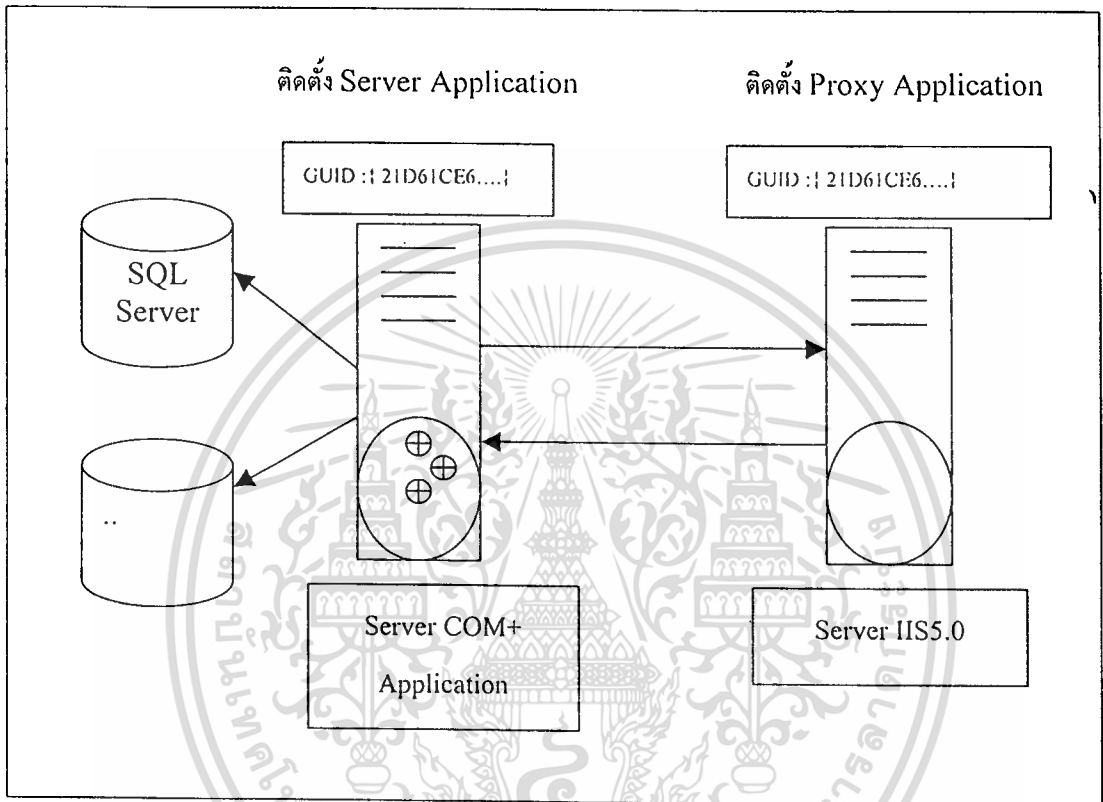
ภาพที่ ข.3 การเลือก Export ตัว Proxy Application

- ทำการเลือก Proxy Application และระบุชื่อ Proxy Application ที่ต้องการสร้าง เพื่อใช้ในการนำไปติดตั้งที่เครื่อง Server หรือ Client ตัวที่ต้องการเข้ามาร้องขอใช้บริการจากตัว Server Application ที่เราได้ทำการ Export Proxy Application ออกมา



ภาพที่ ข.4 การสร้าง Proxy Application

ดังนั้นระบบจะทำการเชื่อมต่อกันได้ ก็ให้ดูที่เลข GUID ที่อยู่ใน C:\Program File \COMPlus Application\... จากการที่เราได้ทำการติดตั้ง Server Application ที่ตัว Server ที่คอยให้บริการและ Proxy Application ที่ตัว Server ที่จะทำการร้องขอ ดังภาพที่ ข.5



ภาพที่ ข.5 หลักการในการเชื่อมต่อระบบในการแยกเป็น 3-Tier

ประวัติผู้เขียน

ผู้เขียน

นายถนอมจิตต์ เกียรติหวนะเสวี

วันเดือนปีเกิด

21 เมษายน พ.ศ. 2520

การศึกษา

2532-2538

มัธยมศึกษา

โรงเรียนกุ๊กแก้ววิทยาลัย

2538-2542

วิศวกรรมศาสตรบัณฑิต (ไฟฟ้า)

มหาวิทยาลัยสงขลานครินทร์ วิทยาเขตหาดใหญ่

การทำงาน :

2544-ปัจจุบัน

บริษัท เจ.วี.ซิสเต็ม จำกัด