

การพัฒนาระบบงานประยุกต์บนระบบฐานข้อมูลแบบไคลน์เซิร์ฟเวอร์

APPLICATION DEVELOPMENT A CLIENT-SERVER DATABASE ENVIRONMENT

โดย

นายปิยะศักดิ์ เจียตระกูล 35104271

นายสมัชชา โกวิทวงศา 35104448

อาจารย์ที่ปรึกษา

ผศ.ดร.ศุภมิตร จิตตะยโศธร

วัน เดือน ปี 31 ก.ค. 2540

เลขทะเบียน 037066

เลขเรียกหนังสือ T 58159 ป.6ม ก.

ปริญญานิพนธ์สำหรับปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2538

ปริญญานิพนธ์ปีการศึกษา 2538

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การพัฒนาระบบงานประยุกต์บนระบบฐานข้อมูลแบบไคลน์เซิร์ฟเวอร์

ผู้จัดทำ

1. นายปิยะศักดิ์ เจียตระกูล 35104271
2. นายสมัชชา โกวิทวงศา 35104448

อาจารย์ที่ปรึกษา

(ผศ.ดร.ศุภมิตร จิตตะยโศธร)

การพัฒนาระบบงานประยุกต์บนระบบฐานข้อมูลแบบไคลน์เซิร์ฟเวอร์

ปิยะศักดิ์ เจียตระกูล
สมัชชา โกวิทวงศา
ผศ.ดร.ศุภมิตร จิตตะยโศธร อาจารย์ที่ปรึกษา

บทคัดย่อ

ในการพัฒนาระบบงานประยุกต์บนระบบฐานข้อมูลแบบไคลน์เซิร์พเวอร์นั้น สามารถนำไปพัฒนากับระบบงานได้หลายๆด้าน สำหรับในโครงงานชิ้นนี้ได้เลือกใช้พัฒนากับงานบริการยืม-คืน และ งานค้นหาหนังสือของห้องสมุดในคณะวิศวกรรมศาสตร์ โดยในงานเดิมที่มีอยู่นั้น ส่วนใหญ่ยังเป็นระบบเซ็นทรัลไลซ์ ซึ่งโปรแกรมทั้งหมดจะวิ่งอยู่บนเครื่องคอมพิวเตอร์ส่วนกลาง เมื่อนำระบบฐานข้อมูลแบบไคลน์เซิร์พเวอร์มาใช้แล้ว ก็จะช่วยให้ภาระงานที่ตกกับเครื่องคอมพิวเตอร์ส่วนกลางเบาบางลงไป อีกทั้งยังทำให้หน้าจอในส่วนการติดต่อกับผู้ใช้ เป็นแบบกราฟฟิกที่น่าใช้มากขึ้น ในการพัฒนางานของห้องสมุดครั้งนี้ นอกจากจะคงหน้าที่การทำงานของระบบเดิมที่มีอยู่แล้ว ยังมีการเพิ่มเติมงานใหม่ที่คาดว่าจะนำมาใช้ในอนาคต เช่นการจัดเก็บข้อมูลของวารสาร การจัดเก็บข้อมูลโสต-ทัศนศึกษา เป็นต้น อีกทั้งยังมีการปรับเปลี่ยนหน้าที่การทำงานหลายๆอย่างเพื่อให้เหมาะสมกับระบบมากยิ่งขึ้น

APPLICATION DEVELOPMENT A CLIENT-SERVER DATABASE ENVIRONMENT

Piyasak Jeatrakul

Samascha Kowitwongsa

Prof.Ass.Dr.Suppamit Chittayasothorn

1995

Abstract

An Application development client-server database can use to develop many system. In this project, we use it to develop a library system in engineering department such as borrow-return service,search-book service. In the old library system was centralize system that all applications were stored and run in host computer. When we take the client-server database system, it can reduce process that run in host computer and it make a good graphic user interface. In this library development , we use function of old system and add a new fuction that will be use in the future such as store magazine data, store audio-visual aids data and we change many functions for suitable system.

สารบัญ

	หน้า
บทที่ 1 บทนำ	1
บทที่ 2 หลักการที่เกี่ยวข้อง	2
2.1 หลักการของระบบฐานข้อมูลแบบไคลน์เซิร์ฟเวอร์	2
2.2 หลักการของโอดีบีซี	6
2.3 หลักการของเดลไฟ	9
2.4 หลักการของโมเดลเชิงสัมพันธ์	15
2.5 สรุป	25
บทที่ 3 ระบบของห้องสมุด	26
3.1 งานบริการ	27
3.2 งานค้นหา	32
3.3 สรุป	36
บทที่ 4 การพัฒนาระบบ	37
4.1 การออกแบบฐานข้อมูลโดยใช้วิธีไนแอม	37
4.2 ตารางของฐานข้อมูล	47
4.3 คาต้าดิกชันนารีของตารางฐานข้อมูล	51
4.4 คาต้าโฟลว์ไดอะแกรมในระดับตาราง	58
4.5 ออกแบบหน้าจอและรายงานของระบบงานห้องสมุด	74
4.6 การออกแบบโปรแกรม	103
4.7 การพัฒนาแอพพลิเคชั่นโดยใช้เดลไฟ	107
4.8 สรุป	108
บทที่ 5 บทวิจารณ์และสรุป	109
กิตติกรรมประกาศ	
หนังสืออ้างอิง	

สารบัญรูปภาพ

หน้า

บทที่ 2

รูปที่ 2-1 ระบบไคลน์เซิร์ฟเวอร์ 3
รูปที่ 2-2 องค์ประกอบของโอบีดีซี 8
รูปที่ 2-3 หน้าต่างกลุ่มของเดลไฟ 9
รูปที่ 2-4 ภาพบนจอภาพเมื่อเริ่มต้นรันเดลไฟ 10
รูปที่ 2-5 โครงสร้างทางฐานข้อมูลของเดลไฟ 12

บทที่ 3

รูปที่ 3-1 คอนเท็คไดอะแกรมของระบบห้องสมุด 26
รูปที่ 3-2 งานบริการห้องสมุด 27
รูปที่ 3-3 การยืมหนังสือ 28
รูปที่ 3-4 การคืนหนังสือ 29
รูปที่ 3-5 การจอง 30
รูปที่ 3-6 การสร้างรายงาน 31
รูปที่ 3-7 การจัดเก็บข้อมูล 32
รูปที่ 3-8 ระบบการค้นหาในห้องสมุด 33
รูปที่ 3-9 ส่วนรับข้อมูล และ สร้างเอสคิวแอลคิวรี่ ของระบบการค้นหา 34
รูปที่ 3-10 ส่วนค้นหาข้อมูลของระบบการค้นหา 35
รูปที่ 3-11 ส่วนแสดงผลของระบบการค้นหา 36

บทที่ 4

รูปที่ 4-1 แสดงสัญลักษณ์ของชนิดเอนติตีภาควิชา 38
รูปที่ 4-2 แสดงสัญลักษณ์ของชนิดเลเบิลรหัสภาควิชา 39
รูปที่ 4-3 แสดงความสัมพันธ์อ้างอิงแบบหนึ่งต่อหนึ่ง 39
รูปที่ 4-4 แสดงความสัมพันธ์อ้างอิงแบบหนึ่งต่อหนึ่งอย่างย่อ 39
รูปที่ 4-5 แสดงความจริงแบบแมนนี่ทูวัน(many to one) 39
รูปที่ 4-6 แสดงความจริงแบบ แมนนี่ทูแมนนี่(many to many) 39

หน้า

รูปที่ 4-7 แสดงการใช้อินทราแฟคไทค์ยูนิคคอนสแทรน 40
รูปที่ 4-8 แสดงการใช้อินเตอร์แฟคไทค์ยูนิคคอนสแทรน 40
รูปที่ 4-9 แสดงการใช้อีควอลิตี้ คอนสแทรน 41
รูปที่ 4-10 แสดงการใช้เอ็กซ์คลูสชั่น คอนสแทรน 41
รูปที่ 4-11 แสดงการใช้ ซับเซต คอนสแทรน 41
รูปที่ 4-12 แสดงการใช้ ซับไทค์ คอนสแทรน 42
รูปที่ 4-13 แสดงการใช้แมนดาทอรี คอนสแทรน 42
รูปที่ 4-14 แสดงความสัมพันธ์ระหว่างลูกค้ากับเงิน 42
รูปที่ 4-15 เทอนารีแฟคไทค์ 43
รูปที่ 4-16 เนสแฟคไทค์ 43
รูปที่ 4-17 ไนแอมไดอะแกรมในส่วนหนังสือ,การยืม,การจอง 44
รูปที่ 4-18 ไนแอมไดอะแกรมในส่วนผู้ใช้บริการ 45
รูปที่ 4-19 ไนแอมไดอะแกรมในส่วนโสตทัศนศึกษา 46
รูปที่ 4-20 ไนแอมไดอะแกรมในส่วนวารสาร 46
รูปที่ 4-21 ไนแอมไดอะแกรมในส่วนปริญญานิพนธ์ 47
รูปที่ 4-22 คาต้าโฟลว์การยืมหนังสือในระดับตาราง 58
รูปที่ 4-23 คาต้าโฟลว์การยืมหนังสือในระดับตาราง(ต่อ) 59
รูปที่ 4-24 คาต้าโฟลว์การคืนหนังสือในระดับตาราง 60
รูปที่ 4-25 คาต้าโฟลว์การคืนหนังสือในระดับตาราง (ต่อ) 61
รูปที่ 4-26 คาต้าโฟลว์การจองหนังสือในระดับตาราง 62
รูปที่ 4-27 คาต้าโฟลว์การจองหนังสือในระดับตาราง(ต่อ) 63
รูปที่ 4-28 คาต้าโฟลว์การสร้างรายงานในระดับตาราง 64
รูปที่ 4-29 คาต้าโฟลว์การจัดเก็บข้อมูลใหม่ในระดับตาราง 65
รูปที่ 4-30 คาต้าโฟลว์การค้นหาหนังสือตามหัวเรื่อง 66
รูปที่ 4-31 คาต้าโฟลว์การค้นหาหนังสือตามชื่อเรื่อง 66
รูปที่ 4-32 คาต้าโฟลว์การค้นหาหนังสือตามชื่อผู้แต่ง 67
รูปที่ 4-33 คาต้าโฟลว์การค้นหาหนังสือตามปีที่พิมพ์ 67
รูปที่ 4-34 คาต้าโฟลว์การค้นหาโสตฯตามหัวเรื่อง 68
รูปที่ 4-35 คาต้าโฟลว์การค้นหาโสตฯตามชื่อเรื่อง 68

หน้า

รูปที่ 4-36 คาต้าโฟลว์การค้นหาปริญญานิพนธ์ตามชื่อเรื่อง 69
รูปที่ 4-37 คาต้าโฟลว์การค้นหาปริญญานิพนธ์ตามชื่ออาจารย์ที่ปรึกษา 69
รูปที่ 4-38 คาต้าโฟลว์การค้นหาปริญญานิพนธ์ตามชื่อนักศึกษาที่ทำ 70
รูปที่ 4-39 คาต้าโฟลว์การค้นหาปริญญานิพนธ์ตามภาควิชา 70
รูปที่ 4-40 คาต้าโฟลว์การค้นหาปริญญานิพนธ์ตามปีการศึกษา 71
รูปที่ 4-41 คาต้าโฟลว์การค้นหาวารสารตามชื่อวารสาร 71
รูปที่ 4-42 คาต้าโฟลว์การค้นหาวารสารตามชื่อบทความ 72
รูปที่ 4-43 คาต้าโฟลว์การค้นหาวารสารตามชื่อผู้แต่งบทความ 72
รูปที่ 4-44 คาต้าโฟลว์การค้นหาข้อมูลผู้ใช้ 73
รูปที่ 4-45 โครงสร้างของหน้าจองานบริการ 74
รูปที่ 4-46 หน้าจอเมนูหลักของงานบริการ 75
รูปที่ 4-47 หน้าจอการยืม 75
รูปที่ 4-48 หน้าจอการคืน 76
รูปที่ 4-49 หน้าจอแจ้งหนังสือหาย 76
รูปที่ 4-50 หน้าจอการจอง 77
รูปที่ 4-51 หน้าจอยกเลิกการจอง 77
รูปที่ 4-52 หน้าจอจัดเก็บข้อมูล 78
รูปที่ 4-53 หน้าจอจัดเก็บข้อมูลหนังสือ 78
รูปที่ 4-54 หน้าจอจัดเก็บข้อมูลปริญญานิพนธ์ 79
รูปที่ 4-55 หน้าจอจัดเก็บวารสาร 79
รูปที่ 4-56 หน้าจอจัดเก็บบทความ 80
รูปที่ 4-57 หน้าจอจัดเก็บโสตทัศนศึกษา 80
รูปที่ 4-58 หน้าจอจัดเก็บข้อมูลผู้ใช้ 81
รูปที่ 4-59 หน้าจอจัดเก็บข้อมูลนักศึกษา 81
รูปที่ 4-60 หน้าจอจัดเก็บข้อมูลอาจารย์และเจ้าหน้าที่ 82
รูปที่ 4-61 หน้าจอจัดเก็บระเบียบสมาชิก 82
รูปที่ 4-62 หน้าจอจัดเก็บหมดอายุสมาชิก 83
รูปที่ 4-63 หน้าจอรายงาน 83
รูปที่ 4-64 หน้าจอสร้างรายงานสถิติการยืมประจำวัน 84

หน้า

รูปที่ 4-65 หน้าจอสร้างรายงานสถิติการจองประจำวัน 84
รูปที่ 4-66 หน้าจอสร้างรายงานหนังสือค้างส่ง 85
รูปที่ 4-67 หน้าจอสร้างรายงานหนังสือหาย 85
รูปที่ 4-68 หน้าจอสร้างรายงานหนังสือใหม่ 86
รูปที่ 4-69 โครงสร้างของหน้าจอการค้นหา 93
รูปที่ 4-70 หน้าจอเมนูหลักสำหรับผู้ใช้ทั่วไป 94
รูปที่ 4-71 หน้าจอเมนูหลักสำหรับเจ้าหน้าที่ 94
รูปที่ 4-72 หน้าจอค้นหาหนังสือ 95
รูปที่ 4-73 หน้าจอค้นหาปริญญานิพนธ์ 95
รูปที่ 4-74 หน้าจอค้นหาโสตทัศนูปกรณ์ 96
รูปที่ 4-75 หน้าจอค้นหาข้อมูลผู้ใช้ 96
รูปที่ 4-76 หน้าจอแสดงข่าวสารประจำวัน 97
รูปที่ 4-77 หน้าจอแสดงรายชื่อหนังสือใหม่ 97
รูปที่ 4-78 หน้าจอแสดงรายชื่อบุคคลค้างส่งหนังสือ 98
รูปที่ 4-79 หน้าจอสำหรับข้อมูลบางส่วน 98
รูปที่ 4-80 หน้าจอสำหรับข้อมูลทั้งหมดของหนังสือ 99
รูปที่ 4-81 หน้าจอสำหรับข้อมูลทั้งหมดของปริญญานิพนธ์ 99
รูปที่ 4-82 หน้าจอสำหรับข้อมูลทั้งหมดของวารสาร 100
รูปที่ 4-83 หน้าจอแสดงข้อมูลการยืม- คืนหนังสือ 100
รูปที่ 4-84 หน้าจอแสดงข้อมูลการจองหนังสือ 101
รูปที่ 4-85 หน้าจอเปลี่ยนรหัสผ่านใหม่ 101
รูปที่ 4-86 หน้าจอสร้างรายงานประจำวัน 102
รูปที่ 4-87 หน้าจอเลือกรายงานประจำวันจากไฟล์ 102
รูปที่ 4-88 เสตทไดอะแกรมของหน้าจอเมนูหลัก 104
รูปที่ 4-89 สเตทไดอะแกรมของหน้าจอยืม-คืน 106

สารบัญตาราง

	หน้า
ตารางที่ 4-1 ตารางจากไนแอมส่วนหนังสือ,การยืม,การจอง	48
ตารางที่ 4-2 ตารางจากไนแอมส่วนผู้ใช้บริการ	49
ตารางที่ 4-3 ตารางจากไนแอมส่วนโสตทัศนศึกษา	49
ตารางที่ 4-4 ตารางจากไนแอมส่วนวารสาร	50
ตารางที่ 4-5 ตารางจากไนแอมส่วนปริญญานิพนธ์	50
ตารางที่ 4-6 คาต้าดิกชันนารีของตารางฐานข้อมูล	51-57

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

บทที่ 1
บทนำ

ในเนื้อหาของปริญญานิพนธ์เล่มนี้เป็นการพัฒนาระบบงานของห้องสมุดโดยใช้แนวความคิดแบบไคลน์เซิร์ฟเวอร์ สำหรับในการพัฒนาระบบงาน ได้ใช้ผลิตภัณฑ์เดลไฟ (Delphi) มาช่วยในการเขียนโปรแกรม โดยติดต่อกับเอสคิวแอลเซิร์ฟเวอร์ (SQL Server) ผ่านโอดีบีซี (ODBC) ดังนั้น ในเนื้อหาของปริญญานิพนธ์ก็จะเริ่มกล่าวถึงทฤษฎีที่จะนำมาใช้ในการทำงาน ซึ่งก็ได้แก่ ไคลน์เซิร์ฟเวอร์คืออะไร เดลไฟ (Delphi) คืออะไร โอดีบีซีคืออะไร เป็นต้น

ในส่วนของงานห้องสมุดก็มีงานอยู่หลายๆด้านด้วยกัน เช่น งานยืม-คืนหนังสือ งานจัดเก็บข้อมูลงานจองหนังสือ การสร้างรายงานสถิติประจำวัน งานค้นหาข้อมูล เป็นต้น ดังนั้นในส่วนของเนื้อหาถัดมา ก็จะกล่าวถึงระบบงานของห้องสมุดที่มีอยู่ ว่างานทั้งหมดประกอบด้วยอะไรบ้าง แล้วระบบที่เราเสนอให้เป็นอย่างไร เมื่อเรารู้ถึงระบบงานที่มีอยู่แล้ว เนื้อหาถัดมาที่เป็นส่วนสำคัญที่จะกล่าวถึงก็คือ การเริ่มออกแบบระบบ ซึ่งได้แก่ การออกแบบตาราง(Table) การออกแบบหน้าจอ และมาสู่การเขียนโปรแกรม ซึ่งก็คือ ขั้นตอนต่างๆ ในการพัฒนาระบบจนกระทั่งได้ระบบงานที่สมบูรณ์ในที่สุด ในส่วนของเนื้อหาสุดท้ายก็จะกล่าวถึงผลสรุปของโครงงาน

วัตถุประสงค์ของโครงงานนี้ประกอบไปด้วย

1. เพื่อนำวิธีการออกแบบและหลักการต่างๆที่ได้ศึกษา มาประยุกต์ใช้การระบบงานจริงได้
2. เพื่อศึกษาการสร้างโปรแกรมจัดการฐานข้อมูล แบบไคลน์เซิร์ฟเวอร์ ด้วยฐานข้อมูลที่เป็นรีเลชั่นนอลดาต้าเบส(relational database)
3. เพื่อพัฒนาระบบการจัดการห้องสมุดเดิมให้มีประสิทธิภาพมากขึ้น
4. เพื่อเพิ่มความสะดวก ในการค้นหาและการจองหนังสือและโดยระบบติดต่อกับผู้ใช้อย่างง่ายและน่าใช้ (User Friendly) มากที่สุด

บทที่ 2
หลักการที่เกี่ยวข้อง

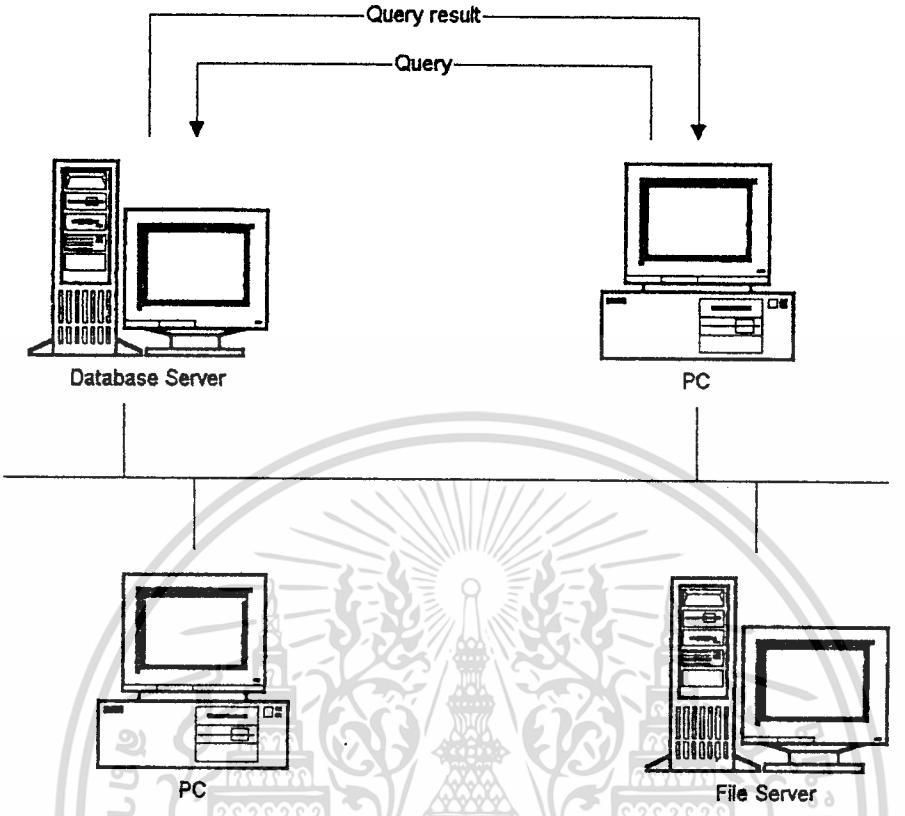
ภายในบทนี้จะกล่าวถึงหลักการและทฤษฎีต่างๆที่เกี่ยวข้องกับโครงงานเพื่อเป็นพื้นฐานของความเข้าใจในเรื่องอื่นๆที่จะกล่าวในบทต่อๆไป

2.1หลักการของระบบฐานข้อมูลแบบไคลน์เซิร์ฟเวอร์

2.1.1 ความหมายของคำว่า ไคลน์เซิร์ฟเวอร์

โดยปกติแล้วเรามักจะมองหรือเรียกระบบแลนว่าเป็นระบบแบบไคลน์เซิร์ฟเวอร์เพราะว่า เวอร์สเตชั่น (Workstation) หรือ ไคลเอ็นท์ จะขอบริการพวก ข้อมูล,ไฟล์โปรแกรม หรือ การพิมพ์ จากเซิร์ฟเวอร์ แต่อย่างไรก็ตาม คำว่า ไคลน์เซิร์ฟเวอร์ ในปัจจุบันนี้ถูกยอมรับโดยแพร่หลายว่าหมายถึงระบบใดๆที่มีการแบ่งการประมวลผลข้อมูลกันระหว่างคอมโพเน็นท์ หรือส่วนประกอบที่แยกกัน

ในรูปแบบที่ง่ายที่สุดของระบบแบบไคลน์เซิร์ฟเวอร์ จะมีการแบ่งการประมวลผลฐานข้อมูลออกกัน ระหว่าง ไคลเอ็นท์พีซี (Client PC) กับ คาต้าเบสเซิร์ฟเวอร์ (Database Server) โดย ไคลน์เอ็นท์พีซีจะวิ่งโปรแกรมประยุกต์ฐานข้อมูล และคาต้าเบสเซอร์พเวอร์จะวิ่งโปรแกรมดีบีเอ็มเอส คาต้าเบสเซอร์เวอร์นี้จะวิ่งอยู่บนไฟล์เซิร์ฟเวอร์ด้วย ไฟล์-เซอร์ฟเวอร์ของระบบแลนก็ยังคงต้องให้บริการการใช้ รีซอร์สร่วมกันเหมือนเดิม โปรแกรมประยุกต์ที่วิ่งอยู่บนไคลเอ็นท์พีซีนี้เราเรียกว่า ฟร้อนท์เอ็นด์ซิสเต็ม (front-end system) ซึ่งจะจัดการกับหน้าจอและการประมวลผลอินพุต เอาท์พุต ส่วนคาต้าเบสเซอร์ฟเวอร์เราจะเรียกว่า แบ็กเอ็นด์ซิสเต็ม (back-end system) ซึ่งจะดูแลการประมวลผลข้อมูลและการติดต่อกับดิสก์ เมื่อผู้ใช้ที่อยู่บนฟร้อนเอ็นด์สร้างรีเควิส(request) ว่าต้องการข้อมูลจากคาต้าเบสเซอร์ฟเวอร์ โปรแกรมแอพลิเคชั่นก็จะส่งรีเควิสผ่านเน็ตเวอร์คไปที่เซอร์ฟเวอร์ คาต้าเบสเซอร์ฟเวอร์ก็จะทำการค้นหาข้อมูลและส่งเฉพาะข้อมูลเหล่านั้นกลับมาตอบคำถามของผู้ใช้ ดังรูปที่ 2-1



รูปที่ 2-1 ระบบไคลน์เซิร์ฟเวอร์

2.1.2 ข้อดีของระบบฐานข้อมูลแบบไคลน์เซิร์ฟเวอร์

ข้อดีอย่างแรกของระบบไคลน์เซิร์ฟเวอร์เกิดจากการแบ่งการประมวลผลระว่างไคลเอ็นท์และเซิร์ฟเวอร์ เพราะการประมวลผลฐานข้อมูลถูกกระทำที่ฝั่งแบ็กเอ็นด์ ดังนั้นจึงทำให้ความเร็วของคีบีเอ็มเอสไม่ถูกยึดติดกับความเร็วของเวอร์คสเตชั่น นอกจากนี้เวอร์คสเตชั่นยังต้องการเพียงให้สามารถวิ่งฟร้อนท์เอ็นด์ ซอร์ฟแวร์ให้ได้เท่านั้นจึงช่วยยืดอายุของเครื่องพีซีรุ่นเก่า เพราะไม่จำเป็นต้องใช้เครื่องที่มีความสามารถสูงๆ

การแบ่งดังกล่าวยังช่วยลดแทรฟฟิกของระบบเครือข่าย เนื่องจากระบบไม่ต้องส่งไฟล์ฐานข้อมูลทั้งหมดกลับไปกลับมา จึงทำให้การติดต่อกันระหว่าง ไคลเอ็นท์และเซอร์ฟเวอร์เป็นไปอย่างรวดเร็วด้วย

นอกเหนือจากเรื่องประสิทธิภาพแล้ว การแบ่งดังกล่าวให้ผู้ใช้ไม่ต้องยึดติดกับระบบหรือแพล็ทฟอร์มใดแบบหนึ่ง ในระบบไคลเอ็นท์เซอร์ฟเวอร์ เวอร์คสเตชั่นหลายๆตัว อาจเป็นคอม-แพททิเบิลพีซี (compatible PC) , แม็คอินทอชส์ (Machintoshes) ,ยูนิกส์ (UNIX) หรือ หลายๆแบบรวมกันก็ได้ นอกจากนี้อาจวิ่งโปรแกรมปฏิบัติการที่ไม่เหมือนกันได้

ข้อดีอีกอย่างหนึ่งของไคลน์เอ็นท์เซอร์ฟเวอร์ก็คือการสามารถรักษาความถูกต้องของข้อมูลได้ เนื่องจากปัจจุบันนี้ คาต้าเบสเซอร์ฟเวอร์จะวิ่งดีบีเอ็มเอสที่ใช้ คาต้าโมเดลแบบรีเลชั่นนอล และผู้ใช้จะถูกกั้นไม่ให้สามารถเข้าถึงข้อมูลได้โดยตรง โดยไม่ผ่านดีบีเอ็มเอส

นอกจากนี้ดีบีเอ็มเอส ก็สามารถใช้บริการอื่นๆที่ช่วยรักษาความปลอดภัยของข้อมูลได้ดีด้วย ไม่ว่าจะเป็นการเข้ารหัสข้อมูล การทำเรียลไทม์แบ็กค์อัพ (realtime backup) ข้อมูลลงสู่เทป เป็นต้น

2.1.3 ข้อเสียของระบบฐานข้อมูลแบบไคลน์เซิร์ฟเวอร์

ข้อเสียหลักๆของระบบไคลเอ็นท์เซอร์ฟเวอร์ก็อยู่ที่ค่าใช้จ่ายที่เพิ่มขึ้นในการจัดการระบบ และการหาบุคคลากรที่มาคอยดูแลคาต้าเบสเซิร์ฟเวอร์ บนระบบเครือข่ายขนาดเล็ก ผู้ควบคุมเครือข่าย (network administrator) อาจเป็นผู้ดูแลคาต้าเบสเซิร์ฟเวอร์เองเลยก็ได้ แต่อย่างไรก็ตามเมื่อมีจำนวนผู้ใช้มากขึ้นหรือตัวคาต้าเบสเอง โตขึ้นเรื่อยๆ ก็ควรจะต้องมีผู้มาดูแลเฉพาะ

นอกจากนี้ยังต้องมีค่าใช้จ่ายเพิ่มเติมในด้ านฮาร์ดแวร์ ด้วยคือ ถึงแม้ว่าผู้ขายดีบีเอ็มเอสแบบไคลเอ็นท์เซอร์ฟเวอร์ จะอ้างว่ามันสามารถวิ่งอยู่บนฮาร์ดแวร์ตัวเดียวกันกับไฟล์เซอร์ฟเวอร์ได้เลยก็ตาม คาต้าเบสเซอร์ฟเวอร์ ก็ควรจะถูกวิ่งบนเครื่องที่อุทิศเพื่องานนี้โดยเฉพาะเพื่อสร้างความมั่นใจในเรื่องประสิทธิภาพและความถูกต้องและปลอดภัยของข้อมูล

ราคาของซอร์ฟแวร์ไคลเอ็นท์เซอร์ฟเวอร์ก็มักจะมีราคาสูงกว่าโปรแกรมโดยทั่วไป และจะต้องมีค่าใช้จ่ายในด้านการฝึกอบรมโปรแกรมเมอร์(Programmer) เพื่อทำงานกับระบบใหม่อีกด้วย

นอกจากเรื่องต่างๆที่ได้กล่าวมาแล้ว ความซับซ้อนของระบบไคลเอ็นท์เซอร์ฟเวอร์ก็เป็นเรื่องหนึ่งที่ต้องคำนึงถึงด้วย เมื่อระบบไคลเอ็นท์เซอร์ฟเวอร์มีองค์ประกอบมาก จึงทำให้เป็นการยากมากขึ้นในการตรวจหาข้อผิดพลาดที่อาจเกิดขึ้นและการติดตั้งระบบแบบไคลเอ็นท์เซอร์ฟเวอร์ก็ยากกว่าด้วยเหตุผลในทำนองเดียวกัน

2.1.4 แพล็ทฟอร์ม(Platforms) ของระบบฐานข้อมูลไคลน์เซิร์ฟเวอร์

แพล็ทฟอร์มนี้หมายถึงการประกอบกันของฮาร์ดแวร์และซอฟแวร์ที่ดีบีเอ็มเอสแบบไคลเอ็นท์เซอร์ฟเวอร์ใช้วิ่งอยู่ เราสามารถแบ่งแพล็ทฟอร์มของระบบออกได้เป็น 4 แบบ คือ พีซี,ยูนิกส์เวอร์ค-สตชั่น ,มินิคอมพิวเตอร์(minicomputer),เมนเฟรม แม้ว่าระบบไคลเอ็นท์เซอร์ฟเวอร์มักจะใช้กับแพล็ทฟอร์มแบบพีซี แต่ว่าแพล็ทฟอร์มทั้ง 4 ก็มีข้อดีข้อเสีย แตกต่างกันไป

ในขณะที่ฮาร์ดแวร์ของระบบมีให้เลือกใช้มากมายตามคุณสมบัติและความสามารถ แต่มีคุณสมบัติที่ต้องมีเหมือนๆ กันของระบบปฏิบัติการที่ใช้สำหรับระบบเหล่านี้ ระบบปฏิบัติการหรือที่เรียก

ว่า โอเอส(OS) นี้เป็นซอฟแวร์สำคัญของระบบ โดยทำหน้าที่เป็นตัวประสานฮาร์ดแวร์และโปรแกรมประยุกต์เข้าด้วยกัน เพื่อให้สามารถปฏิบัติการตามที่ต้องการได้ โดยปกติแล้วแอพพลิเคชั่นมักจะถูกเขียนขึ้นมาเฉพาะสำหรับโอเอสแต่ละตัว สำหรับโอเอสที่เป็นที่รู้จักกันดีก็ได้แก่ MS/PC-DOS,OS/2, ยูนิกส์ยี่ห้อต่างๆ,VMS ของบริษัทเด็ค(DEC) และ MVS/XA ที่วิ่งอยู่บนเครื่องเมนเฟรมของ ไอบีเอ็ม(IBM) เป็นต้น

คุณสมบัติหลักที่โอเอสต้องมีเพื่อรองรับดีบีเอ็มเอสแบบไคลเอ็นท์เซอร์ฟเวอร์ก็คือ มัลติทาส-กิ้ง (Multitasking) มัลติทาสกิ้งคือการที่มีโปรแกรมประยุกต์สามารถวิ่งบนระบบไปพร้อมๆกันนั่นเอง มัลติทาสกิ้งจะทำให้ดีบีเอ็มเอสสามารถจัดการกับคิวรี่ของผู้ใช้หลายๆคนได้พร้อมๆกัน โดยใช้การแบ่งเวลาที่ใช้ในการประมวลผลของซีพียู (CPU) ให้กับแต่ละโพรเซส(Process)

นอกจากนี้โอเอสอาจจะมีความสามารถที่เรียกว่า มัลติยูสเซอร์ (Multiuser) ด้วย คือสามารถรองรับผู้ใช้ที่ทำงานอยู่บนระบบหลายๆคนพร้อมกันได้ แต่ความสามารถทางมัลติยูสเซอร์ไม่ได้เป็นข้อดีหรือข้อเสียพิเศษใดๆสำหรับดีบีเอ็มเอสแบบไคลเอ็นท์เซอร์ฟเวอร์ โอเอสจะมีความสามารถนี้หรือไม่ก็ได้

2.1.5 ฮาร์ดแวร์และซอร์ฟแวร์ของระบบเครือข่าย

ไคลน์เอ็นท์ติดต่อกับเซอร์ฟเวอร์โดยผ่ายเครือข่ายที่เป็นการรวมกันของฮาร์ดแวร์และซอฟแวร์ โดยเครื่องพีซี,เวอร์คสเตชั่นและเซอร์ฟเวอร์จะมีเน็ตเวอร์คอินเตอร์เฟสการ์ด(network interface card) หรือแผ่นวงจรที่ใช้ในการเชื่อมต่อกับเครือข่ายติดตั้งอยู่กับตัวเครื่อง

โดยปกติแล้วถ้าเครือข่ายของเครื่องคอมพิวเตอร์อยู่ภายในอาคารเดียวกันเราจะเรียกว่า แลน หรือ โลคอลแอเรียเน็ตเวอร์ค (local area network) แต่ถ้าระบบเครือข่ายนั้นถูกขยายให้มีการติดต่อกันกว้างขึ้นไปเราก็จะเรียกมันว่าเป็นแวน (WAN) หรือ ไวด์แอเรียเน็ตเวอร์ค(wide area network)

ในปัจจุบันนี้มีโทโพโลจี(topology)ของระบบแลนหรือลักษณะการเชื่องต่อของระบบแลนใช้กันอย่างแพร่หลายอยู่สามแบบด้วยกันคือ อีเทอร์เน็ท(Ethernet),อาร์คเน็ท(ARCnet),และ โทเค็นริง (Token Ring) โดยปกติ ระบบที่เป็นอีเทอร์เน็ทจะวิ่งที่ความเร็วประมาณ 10 เมกะบิทต่อวินาที (Mbps) และใช้สายเคเบิลแบบ ทวิสท์แพร์(twisted-pair,TP) ต่อเข้ากับฮับ(hub) ในรูปแบบสตาร์(Star)

สำหรับโทเค็นริงจะวิ่งที่ความเร็ว 4 หรือ 16 Mps เครือข่ายที่เชื่อมต่อกันแบบโทเคนริงจะดูคล้ายกับการต่อแบบสตาร์ที่ใช้ฮับของอีเทอร์เน็ท

อาร์คเน็ท เป็นเครือข่ายแบบโทเค็นบัส(token-bus network) ที่วิ่งด้วยความเร็ว 2 Mps โดยใช้แอ็กทีฟ(active) และ แพสซีฟ (passive) ฮับต่อกับสายเคเบิลแบบโคเอ็กเชียล

เมื่อพีซีต่อเข้ากับเครือข่าย เราจะต้องทำการติดตั้ง ซอฟแวร์ไดร์เวอร์ (Sofeware driver) เพื่อทำให้พีซีเครื่องนั้นสามารถติดต่อผ่านสายเคเบิลของเครือข่ายได้ นอกจากนี้อาจต้องติดตั้งไดร์เวอร์อื่นๆ เข้าไปด้วยเพื่อให้เครื่องสามารถเข้าถึงข้อมูลหรือไฟล์ต่างๆบนไฟล์เซอร์ฟเวอร์ ส่งงานไปพิมพ์ที่เครื่องพิมพ์ที่ใช้ร่วมกัน,หรือติดต่อกับคาต้าเบสเซอร์ฟเวอร์ได้

2.1.6 เน็ตเวอร์คโพรโตคอล (Network Protocols)

โพรโตคอลคือมาตรฐานที่ใช้ในการติดต่อสื่อสารกันระหว่างเครื่องคอมพิวเตอร์ผ่านระบบเครือข่าย มีเน็ตเวอร์คโพรโตคอลที่เป็นของเฉพาะสำหรับแต่ละผู้ผลิต อยู่มากมายในปัจจุบัน แต่มีเพียงไม่กี่อันที่ถูกออกแบบมาเพื่อระบบฐานข้อมูลแบบไคลน์เซิร์ฟเวอร์ ตัวอย่างโพรโตคอลที่ใช้กันอยู่ทั่วไปในการสื่อสารบนระบบเครือข่ายได้แก่ โพรโตคอล IPX/SPX ใน NetWare ของบริษัทโนเวล (Novell) โพรโตคอล NetBIOS ใน LAN Server ของไอบีเอ็ม และใน LAN Manager ของบริษัทไมโครซอฟท์ (Microsoft) โพรโตคอล DECNet ใน DEC-based LAN เป็นต้น

โพรโตคอลที่ใช้ในการสื่อสารข้ามแพล็ทฟอร์มมีชื่อว่า TCP/IP ซึ่งย่อมาจาก Transmission Control/Internet Protocol แต่เดิม TCP/IP ถูกพัฒนามาเป็นเน็ตเวอร์คโพรโตคอลสำหรับยูนิกส์ ปัจจุบัน TCP/IP มีใช้ทุกๆแพล็ทฟอร์มและทุกระบบปฏิบัติการและได้รับความนิยมอย่างแพร่หลาย ในการนำมาใช้ในการเชื่อมต่อพีซี,เวอร์คสเตชั่น,มินิคอมพิวเตอร์ และเมนเฟรมเข้าด้วยกัน

ถึงแม้ว่าโพรโตคอลที่กล่าวมาแล้วข้างต้นจะเพียงพอที่จะใช้ในการสื่อสารบนแลนแล้วก็ตาม แต่โพรโตคอลเหล่านี้ไม่เพียงพอสำหรับไคลน์เซิร์ฟเวอร์เลยเพราะต้องการโพรโตคอลที่มีระดับสูงกว่านั้น ในปัจจุบันดีบีเอ็มเอสของแต่ละบริษัทจึงต้องกำหนดโพรโตคอลเฉพาะของเขาเองขึ้นมาเพื่อทำหน้าที่นี้เช่น โพรโตคอลที่มีชื่อว่า Pipes เป็นโพรโตคอลที่ใช้สำหรับคาต้าเบสเซอร์ฟเวอร์ที่วิ่งอยู่บน OS/2 เป็นต้น

2.2 หลักการของโอดีบีซี (ODBC)

2.2.1 ความหมายของคำว่าโอดีบีซี

Open Database Connectivity คือวิธีการติดต่อและเข้าถึงจากแอพพลิเคชั่นสู่ระบบบริหารฐานข้อมูล(DBMS) โดยใช้ภาษาเอสคิวแอลเป็นมาตราฐานการเข้าถึงข้อมูล ความสามารถในการเชื่อมต่อแบบนี้ทำให้แอพพลิเคชั่นสามารถเข้าถึงฐานข้อมูลได้หลายรูปแบบ ซึ่งทำให้ผู้พัฒนาโปรแกรมสามารถพัฒนาโปรแกรมไปได้โดยไม่ต้องทำการระบุชนิดของดีบีเอ็มเอส

แต่เดิมนั้นการพัฒนาโปรแกรมประยุกต์ที่ใช้งานเกี่ยวกับฐานข้อมูล การเข้าใช้ฐานข้อมูล โปรแกรมเหล่านี้จะทำผ่านการเรียกใช้เอ็มเบทเด็ด เอสคิวแอล (Embedded SQL) ซึ่งในขณะนั้นวิถีทางแบบนี้ก็ดูจะไปได้ทีเดียว เพราะว่าตัวโปรแกรมสามารถทำการเปลี่ยนรูปแบบของระบบไม่ว่าจะเป็นทางด้านฮาร์ดแวร์ หรือ ซอฟท์แวร์ได้หลายรูปแบบ รวมทั้งระบบปฏิบัติการด้วย (โดยการคอมไพล์ใหม่ทุกครั้งที่มีการย้ายระบบ)

อย่างไรก็ตามในการพัฒนาโปรแกรมในระบบที่มีความแตกต่างกัน เช่น การเรียกใช้ข้อมูลของออราเคิลจาก ไมโครซอฟท์ เอ็กซ์เซล(Microsoft Excel) วิธีการเข้าถึงข้อมูลแบบเดิมนั้นจะต้องทำการพรีคอมไพล์โค้ดของเอ็กซ์เซล และออราเคิลโดยใช้ IBM Precompiler และ Oracle Precompilor ตามลำดับ ซึ่งจะเห็นว่าเป็นการยุ่งยากมากทีเดียว

วิธีการต่อเชื่อมแบบโอดีบีซี จะให้ความสะดวกในการติดต่อข้อมูลมากกว่าวิธีการดั้งเดิม โดยการกำหนดมาตรฐานการต่อเชื่อมของข้อมูล(Data protocol, DBMS capability) และ แนวทางนี้ได้ทำให้เกิดความคิดที่จะสร้างไดร์เวอร์การติดต่อกับฐานข้อมูลขึ้นมา(DLL)

2.2.2 ข้อดีของการติดต่อโดยใช้โอดีบีซี

- ฟังก์ชั่นของโอดีบีซี อนุญาติให้ แอพพลิเคชั่นติดต่อกับดีบีเอ็มเอส ได้โดยสะดวก (การทำคำสั่งเอสคิวแอลและการรับผลลัพธ์)
- ใช้ภาษาเอสคิวแอลตามมาตราฐาน SQL CAE, X/Open และ SQL Access Grop (SAG)
- มีการกำหนด การส่งกลับรหัสความผิดพลาด (Error Code) เป็นมาตรฐานเดียวกัน
- เป็นวิธีการมาตรฐานในการติดต่อกับดีบีเอ็มเอส
- มีการกำหนดชนิดของข้อมูล(Data Type) เป็นมาตรฐาน
- ชุดคำสั่งเอสคิวแอล สามารถกำหนดได้แม้ในขณะรัน
- สามารถเขียนโปรแกรมชุดเดียวแต่สามารถเข้าใช้ดีบีเอ็มเอสได้หลายตัว
- ตัวโปรแกรมไม่ต้องรับผิดชอบในการดูแลการติดต่อข้อมูลกับดีบีเอ็มเอส
- ค่าข้อมูลสามารถถูกส่งหรือรับได้ในรูปแบบที่สะดวกขึ้น

2.2.3 องค์ประกอบของโอดีบีซี

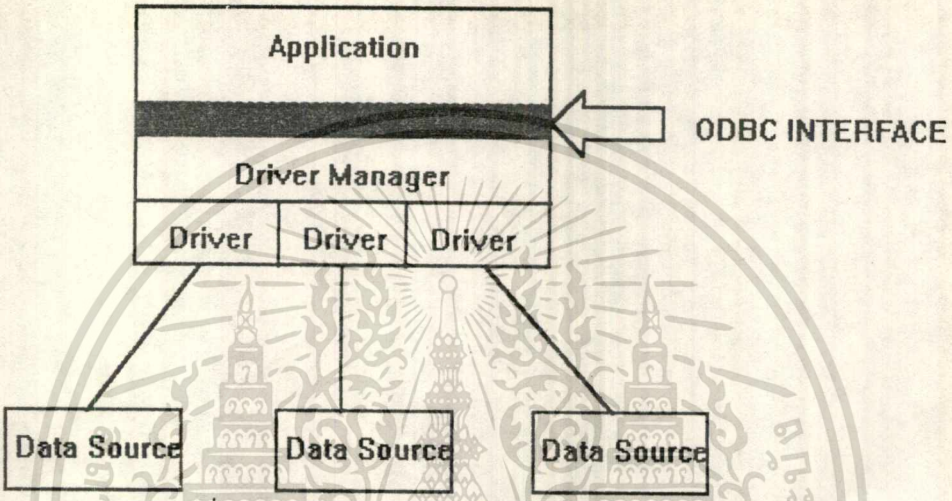
สถาปัตยกรรมของโอดีบีซีประกอบด้วย 4ส่วนสำคัญ

1.แอพพลิเคชั่น ทำหน้าที่ประมวลผลและเรียกใช้ฟังก์ชั่นของโอดีบีซี ตามคำสั่งภาษาเอสคิวแอล พร้อมทั้งทำการรับผลลัพธ์ด้วย

2.ตัวจัดการไดร์เวอร์(Driver Manager) ทำหน้าที่ load driver เชื่อมต่อกับแหล่งข้อมูล

3. ไดร์เวอร์(Driver)ทำหน้าที่ประมวลผลการเรียกใช้ฟังก์ชั่นของโอดีบีซีส่งคำสั่งเอสคิว-แอลไปสู่แหล่งข้อมูลที่ถูกต้องและทำการส่งผลลัพธ์กลับให้แอพพลิเคชั่นและในบางครั้งไดร์เวอร์จะทำหน้าที่แปลงคำสั่งที่ส่งมาให้อยู่ในรูปที่สนับสนุนโดยดีบีเอ็มเอส แต่ละชนิดอีกด้วย

4.แหล่งข้อมูล(Data Source)เป็นแหล่งข้อมูลที่ผู้ใช้ต้องการเข้าถึง



รูปที่ 2-2 องค์ประกอบของโอบีดีซี

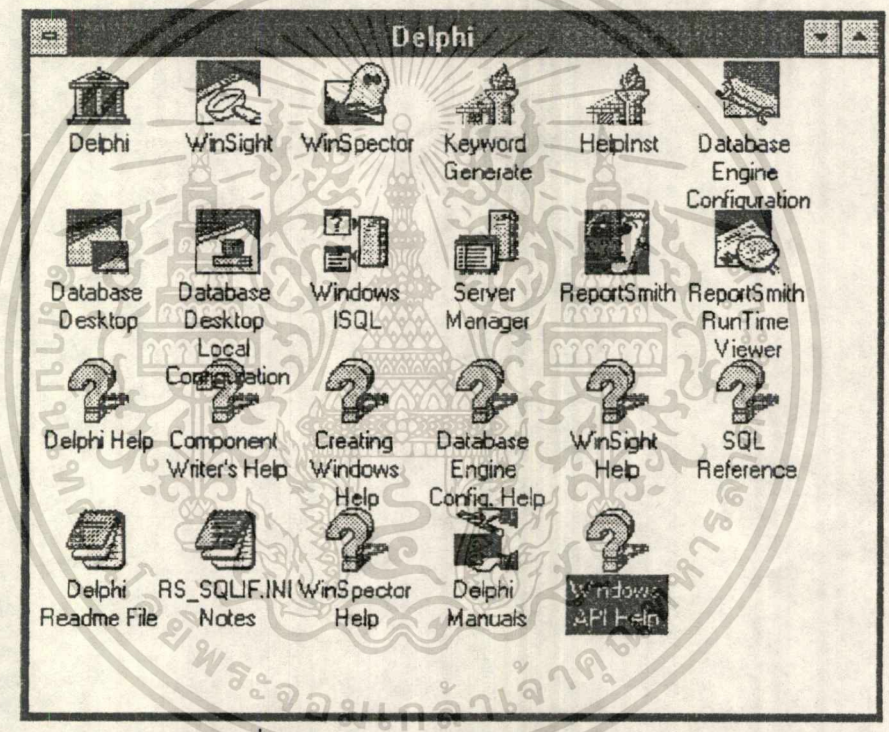
ตัวโปรแกรมจะเรียกใช้การเชื่อมโอบีดีซี ในการทำงานต่อไปนี้

1. ร้องขอการต่อเชื่อมกับแหล่งข้อมูล
2. ส่งคำสั่งเอสคิวแอลสู่แหล่งข้อมูล
3. กำหนดพื้นที่การจัดเก็บและรูปแบบของข้อมูล ที่เป็นผลลัพธ์จากเอสคิวแอล รีเควสซ์ (SQL request)
4. ร้องขอผลลัพธ์
5. ประมวลผลและจัดการกับข้อผิดพลาด
6. รายงานผลให้กับผู้ใช้ (ถ้าจำเป็น)
7. ร้องขอการคอมมิท (Commit) หรือ โรลแบ็ค (Roll back) สำหรับควบคุมการประมวลผลทรานแอคชั่น (Tranaction)
8. ยกเลิกการติดต่อกับแหล่งข้อมูล

2.3 หลักการของเดลไฟ (DELPHI)

เดลไฟ ก็คือ วิชวลปาสคาลเพื่อการโปรแกรมบนวินโดวส์ที่ง่าย และรวดเร็ว โดย เดลไฟจะคอมพายล์โปรแกรมเป็นไฟล์ .EXE อย่างแท้จริง ไม่มีพีโค้ด (p-code) ซึ่งจะต้องแปลในแบบอิน-เตอร์พรีต(interprete) อีกในขณะรัน จึงรันได้เร็วกว่า 10-20 เท่า

2.3.1 โปรแกรมในชุดของเดลไฟ



รูปที่ 2-3 หน้าต่างกลุ่มของเดลไฟ

เมื่อติดตั้งเดลไฟจะเปิดหน้าต่างกลุ่ม (group window) ให้ ดังแสดงในรูป ซึ่งประกอบด้วยไอคอนโปรแกรม มีทั้งโปรแกรมหลักและโปรแกรมสนับสนุนซึ่งต่างก็เป็นอิสระต่อกัน และไอคอนข้อมูลของระบบช่วย (Help) (ที่แสดงเป็นรูปเครื่องหมาย?) แบ่งได้เป็น 8 กลุ่มคือ

1. Delphi คือตัวเดลไฟเป็นโปรแกรมหลัก
2. Winsight และ WinSpector เพื่อช่วยในการตรวจแก้โปรแกรม
3. Keyword Generate และ HelpInst เพื่อสร้างไฟล์ Help ของเรา
4. Database Engine Configuration เพื่อปรับแต่งระบบ BDE (Borland Database Engine) เช่น ให้แสดงผลภาษาไทย

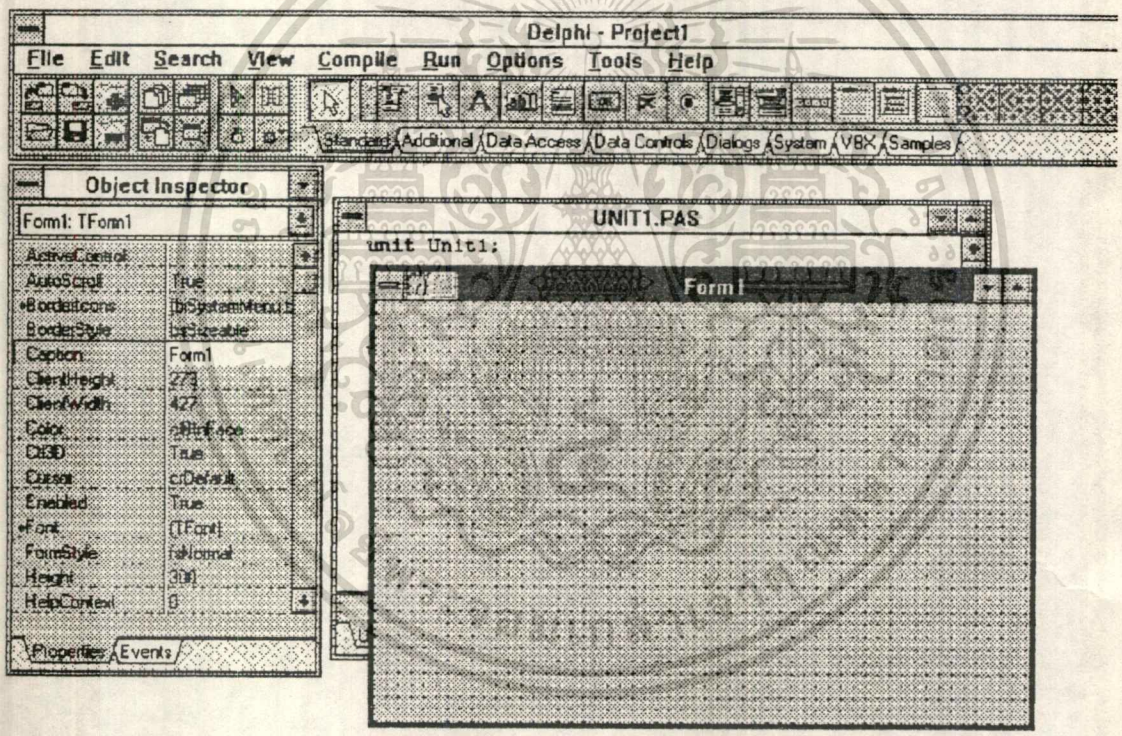
5. Database Desktop และ Database Desktop Local Configuration เป็นฐานข้อมูลที่เป็นโลคอล

6. Server Manager และ Windows ISQL เป็นเซิร์พเวอร์(server) และ ไคลน์(client) ของอินเตอร์เบส(InterBase)

7. ReportSmith และ ReportSmith RunTime Viewer เพื่อการจัดทำรายงาน

8. Delphi Manuals เป็นหนังสือคู่มืออิเล็กทรอนิกส์

2.3.2 หน้าตาของเดลไฟ



รูปที่ 2-4 ภาพบนจอภาพเมื่อเริ่มต้นรันเดลไฟ

เมื่อเริ่มต้นรันเดลไฟจะปรากฏภาพบนจอภาพดังรูปข้างต้น ซึ่งประกอบด้วยหน้าต่าง 4บาน ด้วยกัน คือ

1. หน้าต่างหลักอยู่บนสุด ประกอบด้วย

-ไตเติลบาร์และเมนูบาร์(ตามปกติ)

-สปีดบาร์ (Speed bar) ได้เมนูบาร์ ตั้งแต่ File ถึง View คือสปีดบาร์ มีปุ่มต่างๆแทนรายการในเมนู

-กล่องอุปกรณ์ (Component palette) ใต้เมนูบาร์ทางด้านขวาตั้งแต่รายการ Compile เป็นต้นไป คือกล่องอุปกรณ์ แบ่งออกเป็นหลายชั้นหรือหลายหน้าตามที่ได้ทำที่คั่นหน้า ไว้เป็น Standard ถึง Sample ซึ่งเมื่อเลือกที่คั่นหน้าใดจะแสดงอุปกรณ์คือคอมโปแนนต์ของหน้านั้นให้เลือกต่อ

2. หน้าต่างออฟเจคอินสเปคเตอร์ (Object Inspector) อยู่ทางด้านซ้าย แบ่งออกเป็น 2 หน้า คือ พรอพเพอร์ตี้(Properties) เพื่อให้ค่าพรอพเพอร์ตี้ และ อีเว้นท์ (Event) เพื่อเลือกเหตุในการป้อนโปรแกรม

3. แบบฟอร์มหรือฟอร์ม คือหน้าต่าง Form1 ในรูป เพื่อการกำหนดคอมโปเนนต์

4. หน้าต่างเอดิเตอร์ คือหน้าต่าง UNIT1.PAS ตามรูป เพื่อการป้อนโปรแกรม

2.3.3 ข้อดีของเดลไฟ

1. เป็นการเขียนโปรแกรมในแบบโอโอพีอย่างแท้จริง นั่นคือเราสามารถนำโปรแกรมเก่ามาใช้ได้อีก

2. มีคอมโปเนนต์ (component) ที่จะใช้ในโปรแกรมให้เลือกมากกว่า 75 ตัว ตั้งแต่คอนโทรลธรรมดา เช่น ปุ่มควบคุมและปุ่มวิทยุ ระดับกลางเช่น นาฬิกาและกรอบรายการไฟล์ และที่ซับซ้อนได้แก่คอมโปเนนต์ทางด้านฐานข้อมูล

และเพราะเป็นโอโอพีจึงทำให้สามารถสร้างผู้สืบสกุลจากคอมโปเนนต์เหล่านี้ให้สามารถทำงานตามที่เราต้องการได้

3. ใช้ภาษาปาสคาลในการสร้างคอมโปเนนต์และให้เราเขียนโปรแกรม ซึ่งสามารถที่จะติดต่อกับหรือใช้กับ โอแอลอี 2.0 (OLE 2.0), ดีดีอี (DDE), วีบีเอ็กซ์ (VBX) และ โอดีบีซี ได้

4. ให้เราสามารถเขียนโปรแกรมติดต่อกับฐานข้อมูล แยกได้เป็น 3 ทางคือ

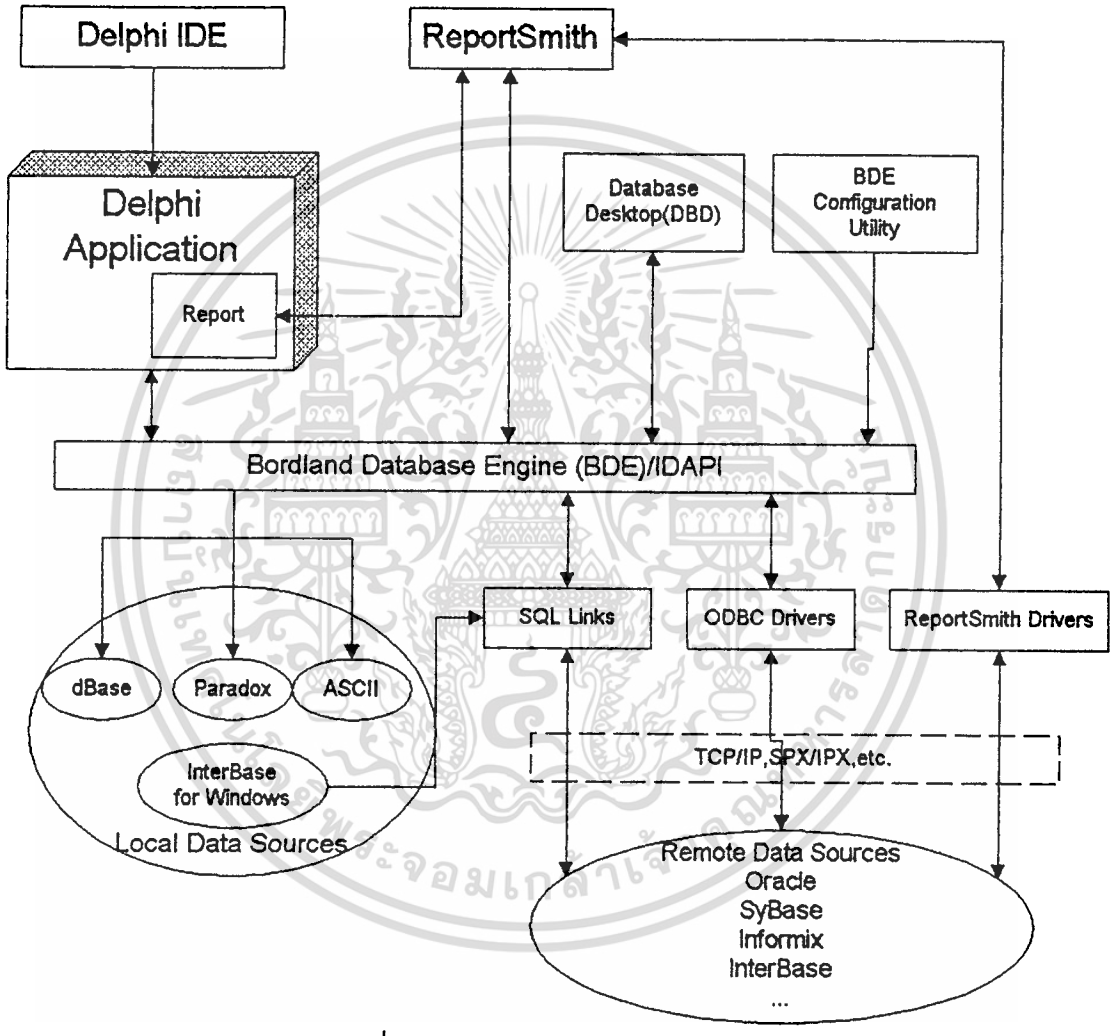
4.1 ติดต่อโดยตรงกับฐานข้อมูลภายใน คือ ไฟล์ของ ดีเบส (dBase) ,พาราด็อกซ์ (Paradox) และ อินเตอร์เบส (InterBase)

4.2 ผ่านทาง เอสคิวแอลลิ้งค์ (SQL Links) เพื่อการติดต่อกับระบบฐานข้อมูลภายนอกคือ ออราเคิล (Oracle),ไซเบส(Sybase),อินฟอร์มิกซ์(Informix),ไมโครซอฟท์เอสคิวแอลเซิร์ฟเวอร์ (Microsoft SQL Server) และ อินเตอร์เบส ที่เรียกว่า ไคลน์เซิร์ฟเวอร์

4.3 ผ่านทางโอดีบีซี ไดร์เวอร์ (ODBC Drivers) เพื่อการติดต่อกับฐานข้อมูลภายนอกคือ บีทรีพ (Btrieve) ,ไมโครซอฟแอกเซส (Microsoft Access) และ ดีบีทู (DB2) เป็นต้น ซึ่งก็เป็นแบบไคลน์เซิร์ฟเวอร์ เหมือนกัน

ในการพัฒนาแอพพลิเคชั่นทางดาต้าเบสนั้น เดลไฟจะใช้คอมโพเน้นติดต่อกับบอร์คแลนด์ดาตาเบส เอ็นจิ้น (Borland Database Engine:BDE) เพื่อทำการติดต่อกับฐานข้อมูลอีกทีนึง ในรูป

ที่ 2-3 จะแสดงความสัมพันธ์ของเดลไฟกับบีดีอีและแหล่งข้อมูล ซึ่งลักษณะดังกล่าวทำให้เราสามารถสร้างแอพพลิเคชั่นทางฐานข้อมูลโดยให้การติดต่อกับพาราดอกซ์และดีเบสเทบิล และ โลคอล อินเตอร์เบส เซิร์ฟเวอร์ โดยผ่าน บีดีอี



รูปที่ 2-5 โครงสร้างทางฐานข้อมูลของเดลไฟ

(Delphi database architecture)

2.3.4 การสร้างไคลน์เซิร์ฟเวอร์แอพพลิเคชั่น

เดลไฟสามารถพัฒนาแอพพลิเคชั่นที่ทำงานแบบรีโมตเซิร์ฟเวอร์ (Remote Server) ได้แก่พวก ออราเคิล ไซเบส อิมฟอร์มิกส์ และ อินเตอร์เบส อีกทั้งยังทำงานแบบโลคอล ได้แก่ฐานข้อมูลของ พาราดอกซ์ ดีเบส ซึ่งในการทำงานแบบรีโมตเซิร์ฟเวอร์ เครื่องไคลน์ และเซิร์ฟเวอร์จะต้องติดต่อกันผ่านเน็ตเวอร์ค

ในการพัฒนาแอพพลิเคชั่นแบบไคลน์เซิร์ฟเวอร์มีหลายเรื่องด้วยกันที่ต้องพิจารณา อันได้แก่

พอร์ตเทบิลิตี้ (Portability) กับ ออฟติไมเซชั่น (Optimization) เช่น แอพพลิเคชั่นนั้นใช้เอสคิวแอล ที่เป็นของเฉพาะเซิร์พเวอร์นั้นหรือเปล่า และมีวิธีใดบ้างที่ฐานข้อมูลสามารถถูกออฟติไมซ์โดยเซิร์พเวอร์

การเชื่อมต่อ(Connectivity) คือมีโปรโตคอลอะไรบ้างที่แอพพลิเคชั่นต้องใช้

ทรานแอ็คชั่น(Tranactions) คือมีชนิดของทรานเอ็คชั่นคอนโทรลอะไรบ้างที่แอพพลิเคชั่นต้องการ

2.3.4.1 พอร์ตเทบิลิตี้ กับออฟติไมเซชั่น

ในระบบไคลน์เซิร์ฟเวอร์นั้น ฐานข้อมูลจะรันบนเซิร์ฟเวอร์ และแอพพลิเคชั่นจะรันบนไคลน์ ดังนั้น พอร์ตเทบิลิตี้จึงหมายถึง แอพพลิเคชั่นนั้นสามารถที่จะไปรันเซิร์ฟเวอร์อื่นๆ ได้โดยง่าย โดยไม่ยึดติดกับเซิร์ฟเวอร์ใดเซิร์ฟเวอร์หนึ่งโดยเฉพาะ ส่วน ออฟติไมเซชั่น นั้นหมายถึงคุณสมบัติพิเศษของเซิร์ฟเวอร์ที่แอพพลิเคชั่นนั้นนำไปใช้ โดยจะทำให้ยึดติดกับเซิร์ฟเวอร์นั้นๆ แต่ก็จะทำให้แอพพลิเคชั่นนั้น มีประสิทธิภาพมากยิ่งขึ้น

ในเรื่องพอร์ตเทบิลิตี้ของไคลน์นั้นไม่เป็นปัญหาของเดลไฟเพราะ เดลไฟสามารถรันได้ทั้ง 16 บิต และ 32 บิต วินโดว์แพลตฟอร์ม แต่อย่างไรก็ตามในเรื่องพอร์ตเทบิลิตี้ของเซิร์ฟเวอร์ และพอร์ตเทบิลิตี้ของการสื่อสาร(Communication) จะต้องนำมาพิจารณา

เราอาจออกแบบให้แอพพลิเคชั่นสามารถติดต่อกับเซิร์ฟเวอร์หลายๆชนิด เพราะแต่ละกลุ่มของผู้ใช้อาจมีความต้องการใช้ไม่เหมือนกัน ซึ่งการออกแบบระบบแบบไคลน์เซิร์ฟเวอร์นั้นจะเกิดข้อได้เปรียบ-เสียเปรียบ ระหว่างพอร์ตเทบิลิตี้ กับออฟติไมเซชั่น เพราะว่าหากเราใช้คุณสมบัติโดยตรงนั้นแม้จะเพิ่มประสิทธิภาพของแอพพลิเคชั่น แต่ก็จะลดพอร์ตเทบิลิตี้ลง

ในเดลไฟแอพพลิเคชั่นหากนำ Ttable คอมโพเนนต์มาใช้ก็จะพอร์ตเทบิลิตี้อย่างสมบูรณ์แต่หากใช้ Tquery คอมโพเนนต์มาใช้ มันจะสามารถเป็นพอร์ตเทบิลิตี้ได้ก็ต่อเมื่อใช้ไวยากรณ์ของเอสคิวแอลที่เป็นมาตรฐานของ ANSI เท่านั้น

2.3.4.2 การเชื่อมต่อ

ในแอพพลิเคชั่นของไคลน์ที่ใช้เดลไฟนั้นสามารถใช้เน็ตเวอร์คโปรโตคอลที่เหมาะกับเซิร์ฟเวอร์ได้หลายๆแบบ (เช่น TCP/IP ,โนเวล SPX/IPX) ตราบใดที่ทั้งเซิร์ฟเวอร์และไคลน์ มีซอร์ฟแวร์สำหรับการสื่อสารที่ถูกต้อง และจะต้องทำการคอนฟิค เอสคิวแอลลิ้งค์ไดร์เวอร์ (SQL Link driver) ของโปรโตคอลที่ต้องการเสียก่อน

2.3.4.3 การควบคุมทรานแอ็คชั่น

ฐานข้อมูลของเซิร์ฟเวอร์มีการจัดการรีเควสในรูป หน่วยของงานที่เรียกว่าทรานแอ็คชั่น ดังนั้น ทรานแอ็คชั่นก็คือ กลุ่มของแอสคิวแอล สเตทเมนต์ที่จะกระทำให้สำเร็จทั้งหมดก่อนที่จะตกลงในการเปลี่ยนแปลงฐานข้อมูล หากมีสเตทเมนต์ใดเฟล(fail) ก็จะเฟลทั้งหมด ไม่มีการเปลี่ยนแปลงฐานข้อมูลใดๆของทรานแอ็คชั่นนั้น

ทรานแอ็คชั่นเป็นการประกันความน่าเชื่อถือในความถูกต้องของฐานข้อมูล เช่นเมื่อฮาร์ดแวร์เสียหาย หรือ เมื่อมีการใช้ข้อมูลร่วมกันของผู้ใช้หลายๆคน

ในเอสคิวแอลนั้นทรานแอ็คชั่นสามารถใช้คำสั่งในการตกลงหรือปฏิเสธอย่างชัดเจนซึ่งก็คือ คอมมิท(Commit) เพื่อตกลงในการเปลี่ยนแปลงของข้อมูล และ โรลแบ็ค(Rollback) ในการปฏิเสธการเปลี่ยนแปลงข้อมูลทั้งหมดที่เกิดขึ้นในทรานแอ็คชั่นนั้น

ในเดลไฟแอพพลิเคชั่นสามารถควบคุมทรานแอ็คชั่นได้ 2 แบบหลักๆคือ

1. แบบโดยนัย (Implicitly) เดลไฟจะเริ่มและคอมมิททรานแอ็คชั่นที่ต้องการอย่างอัตโนมัติเมื่อแอพพลิเคชั่นนั้นเรียกใช้โพสต์เมตอท (Post method)และเพิ่มเรคคอทด์ (AppendRecord) เป็นต้น โดยแต่ละคำสั่งเขียนจะคอมมิททุกครั้งโดยเป็นแบบบรรทัดต่อบรรทัด ซึ่งจะเป็นการลดปัญหาการขัดแย้งกันเรื่องการเปลี่ยนแปลงข้อมูล(Update Conflicts) แต่มันจะเพิ่มงานให้กับเน็ตเวอร์คแทรฟฟิค (Network Traffic)

ในการใช้แบบโดยนัยมาควบคุมทรานแอ็คชั่นจะต้องเซต SQLPASSTHRUMODE ในบีดีอีคอนฟิคกูเรชั่น เป็นแชร์ออโตคอมมิท (SHARED AUTOCOMMIT) ซึ่งจะเริ่มและคอมมิทของการเปลี่ยนแปลงแต่ละบรรทัด

ในการควบคุมทรานแอ็คชั่นแบบโดยนัยนี้จะทำงานอย่างอัตโนมัติดังนั้นจึงขาดความยืดหยุ่น ถ้าต้องการทรานแอ็คชั่นที่เป็นมัลติ-โรล (multi-row transaction) จะต้องใช้วิธีถัดไป

2. แบบชัดเจน(Explicitly) ได้แก่

-การใช้เริ่มทรานแอ็คชั่น(Start Tranaction),คอมมิทและโรลแบ็คเมตอท ของ Tdatabase คอมโพเน้นต์ และจะทำงานตามระดับทรานแอ็คชั่นไอโซเลชั่น(Transaction Isolation level) ที่ระบุที่ TransIsolation พรอพเพอร์ตีส์ของ Tdatabase ในกรณีนี้จะต้อง เซต SQLPASSTHRUMODE ในบีดีอีคอนฟิคกูเรชั่น เป็นแชร์โนออโตคอมมิท (SHARED NOAUTOCOMMIT)

-การใช้พาสทรูเอสคิวแอล (Passthrough SQL) ใน Tquery คอมโพเน้นต์ เป็นใช้เอสคิวแอลควบคุมทรานแอ็คชั่น ที่เป็นลักษณะเฉพาะของเซิร์พเวอร์นั้นๆ ในกรณีนี้จะต้องเซต SQLPASSTHRUMODE เป็น NOT SHARED

●ระดับทรานแอคชั่นไอโซเลชั่น

เป็นการกำหนดความสามารถในการเข้าถึงข้อมูลชุดเดียวกัน โดยทั้งทรานแอ็คชั่นแบบโดยนัยและแบบชัดเจนจะมีการกำหนด TransIsolation พรอพเพอตีส์ของTdatabase คอมโพเนนต์ ซึ่งสามารถแบ่งระดับได้ดังต่อไปนี้

tiDirtyRead : คือทรานแอ็คชั่นสามารถ อ่านข้อมูลที่มีการเปลี่ยนแปลงและยังไม่คอมมิทได้ สำหรับแบบนี้เป็นระดับไอโซเลชั่นต่ำสุด

tiReadCommited : คือทรานแอ็คชั่นสามารถอ่านข้อมูลที่คอมมิทแล้วเท่านั้น (เป็นค่าดีฟอลต์)

tiRepeatableRead : คือทรานแอ็คชั่นไม่สามารถอ่านข้อมูลได้เดิมได้อีกหากมีทรานแอ็คชั่นอื่นไปแก้ไขข้อมูล นั่นเป็นการรับประกันว่าทรานแอ็คชั่นจะสามารถอ่านเรคคอร์ดได้ครั้งเดียวและมันจะต้องไม่เปลี่ยนแปลงหากต้องการอ่านอีกครั้ง สำหรับแบบนี้ เป็นระดับไอโซเลชั่นสูงสุด

2.4 หลักการของโมเดลเชิงสัมพันธ์ (Relational model)

โมเดลเชิงสัมพันธ์เป็นโมเดลที่ใช้ในการอธิบายความสัมพันธ์ของข้อมูลที่ถูกเก็บด้วยระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (Relational DataBase Management System : RDBMS) ซึ่งเป็นผลงานของ ดร. คอดด์ (Codd) ที่ได้เสนอผลงานวิจัยให้ชาวโลกรู้จักในปี พ.ศ. 2513 โดยมีบรรดานักวิชาการทางคอมพิวเตอร์ก็ได้ให้ความสนใจและทุ่มเททำการวิจัยเกี่ยวกับโมเดลนี้มากมาย จนในปัจจุบันนี้ได้แพร่หลายไปมาก มีการนำไปใช้งานกับเครื่องระดับตั้งแต่เมนเฟรมลงไปจนถึงเครื่องระดับไมโครด้วย และก็เป็นที่ยอมรับกันแล้วว่า บรรดาผู้ใช้ระบบฐานข้อมูล (โดยเฉพาะผู้ที่ทำงานด้วยเครื่องระดับมินิและระดับไมโคร) จะมีความคุ้นเคยกับโมเดลเชิงสัมพันธ์นี้มากกว่าอีก 2 โมเดล คือ โมเดลเชิงแตกสาขา (Hierarchical model) และโมเดลเชิงโครงข่าย (Network model) ที่มีมาก่อนหน้านี้

นอกเหนือจากความแพร่หลายของโมเดลเชิงสัมพันธ์นี้แล้ว ข้อดีของโมเดลเชิงสัมพันธ์ที่มีมากกว่าอีก 2 โมเดล ดังนี้

1. โมเดลเชิงสัมพันธ์เป็นโมเดลที่สามารถสร้างความเข้าใจได้ง่ายกว่า เพราะภาพลักษณ์ของข้อมูลที่เก็บโดยโมเดลเชิงสัมพันธ์จะมาจากมุมมองของผู้ใช้ ซึ่งจะมีความซับซ้อนน้อยกว่าภาพลักษณ์ของข้อมูลที่เก็บโดยอีก 2 โมเดล
2. ระบบส่วนใหญ่ที่ใช้โมเดลเชิงสัมพันธ์นี้มักจะมีเครื่องมือที่ช่วยให้ผู้ใช้สามารถจัดการกับข้อมูลที่เก็บอยู่ได้ง่ายกว่าข้อมูลที่จัดเก็บด้วยโมเดลแบบอื่น
3. โมเดลเชิงสัมพันธ์นี้มีเครื่องมือที่ช่วยให้ผู้ใช้สามารถค้นพบปัญหาที่เกิดขึ้นในการออกแบบระบบฐานข้อมูลได้โดยง่าย และยังง่ายในการแก้ไขการออกแบบที่ผิดพลาดนั้นด้วย

4. โมเดลเชิงสัมพันธ์เป็นโมเดลที่มีความสอดคล้องกับหลักการของฐานข้อมูล ผู้ใช้ไม่ต้องพะวงกับรายละเอียดของการจัดเก็บข้อมูลเหมือนกับการจัดข้อมูลของโมเดลอื่น
5. ภาษาที่ใช้ในการจัดการกับข้อมูลที่จัดเก็บด้วยระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (ภาษา SQL : Structure Query Language) เป็นภาษาแบบ set oriented ซึ่งจะต่างกับภาษาที่ใช้ในการจัดการกับข้อมูลที่จัดเก็บด้วยระบบจัดการฐานข้อมูลของโมเดลอื่นที่เป็นภาษาแบบ record-at-a-time

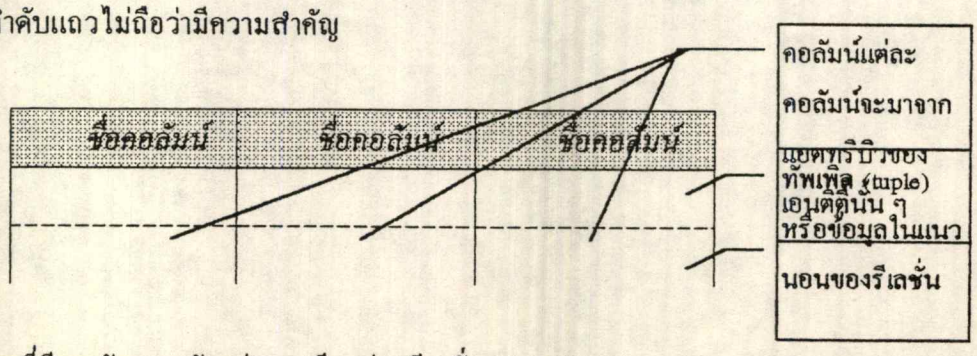
แม้ว่าโมเดลเชิงสัมพันธ์จะมีข้อดีหลายประการดังได้กล่าวไปแล้ว แต่ในปัจจุบันก็ยังมีจุดอ่อนที่มีการอ้างอิงถึงเสมอ คือ ระบบจัดการฐานข้อมูลแบบโมเดลเชิงสัมพันธ์นี้มักจะมีประสิทธิภาพในการใช้งานสู้อีก 2 โมเดลไม่ได้ โดยเฉพาะในการประยุกต์ใช้งานขององค์กรขนาดใหญ่ จุดอ่อนนี้ก็ได้มีการแย้งกลับมาในแง่ที่ว่า โมเดลเชิงสัมพันธ์เป็นโมเดลที่มีอายุการพัฒนาน้อยกว่าอีก 2 โมเดล จึงเป็นไปได้ว่า การพัฒนาที่ผ่านมาของโมเดลเชิงสัมพันธ์ก็ยังมีจำนวนระดับขั้นที่ได้พัฒนาไปแล้วน้อยกว่าอีก 2 โมเดล ดังนั้นหากต้องการเปรียบเทียบการทำงานระหว่างโมเดลเชิงสัมพันธ์กับโมเดลอื่นก็ควรที่จะทำการเปรียบเทียบที่ระดับจำนวนขั้นการพัฒนาที่เท่ากันจึงจะสมเหตุสมผล

2.4.1 ศัพท์เฉพาะของโมเดลเชิงสัมพันธ์

ในหัวข้อนี้จะกล่าวถึงโมเดลเชิงสัมพันธ์ โดยกำหนดนิยามและกล่าวถึงคำศัพท์ต่าง ๆ ที่เกี่ยวข้องกับโมเดลนี้

จากการที่ข้อมูลที่เก็บด้วยโมเดลเชิงสัมพันธ์จะถูกเก็บไว้ในตารางที่จะถูกเรียกว่า*"รีเลชั่น"* โดยที่รีเลชั่นทุกรีเลชั่นจะอยู่ในรูปของตาราง แต่ตารางบางตารางอาจไม่เป็นรีเลชั่นก็ได้ ดังนี้ตารางที่มีลักษณะเป็นรีเลชั่นจะต้องมีคุณลักษณะดังนี้

1. แต่ละช่องของตารางจะบรรจุข้อมูลได้เพียงค่าเดียว
2. ชื่อหัวข้อในแต่ละคอลัมน์มีความแตกต่างกัน อันเป็นชื่อของแอตทริบิวของเอนติตี้
3. ค่าข้อมูลที่อยู่ในแต่ละคอลัมน์ คือ ค่าของแอตทริบิวตามที่ระบุหัวข้อไว้ที่หัวของคอลัมน์นั้น ๆ
4. การเรียงลำดับคอลัมน์ไม่ถือว่ามีความสำคัญ
5. ข้อมูลแต่ละแถวจะต้องแตกต่างกัน
6. การเรียงลำดับแถวไม่ถือว่ามีความสำคัญ



ตารางที่มีคุณลักษณะดังกล่าวจะเรียกว่า *รีเลชั่น*

ดังนั้น เราจะได้นิยามของ*ฐานข้อมูลเชิงสัมพันธ์* คือ ฐานข้อมูลที่เกิดจากการรวบรวมรีเลชั่นต่าง ๆ ที่มีความสัมพันธ์ (relationship) ระหว่างกัน

เราจะเรียกข้อมูลแต่ละแถวในแนวนอนของรีเลชั่นว่า *ทัพเพิล (tuple)* และเรียกข้อมูลแต่ละแถวในแนวตั้งหรือแนวคอลัมน์ว่า *แอตทริบิว (attribute)* โดยที่คำว่า *คีย์ (key)* จะหมายถึงข้อมูลที่เกิดจากแอตทริบิว 1 ตัวหรือหลายตัวก็ได้ แต่ละรีเลชั่นจะต้องมีสิ่งที่เรียกว่า *คีย์หลัก (primary key)* คือ ข้อมูลของแอตทริบิว 1 ตัวหรือมากกว่า 1 ตัวก็ได้ที่สามารถใช้เป็นตัวเจาะจงบอกเราได้ว่ากำลังอ้างอิงถึงข้อมูลทัพเพิลใด ส่วนคีย์ที่เป็นแอตทริบิวของรีเลชั่นอื่นที่ซ้ำกับแอตทริบิวที่เป็นคีย์หลักของรีเลชั่นหนึ่งจะเรียกว่า *คีย์นอก (foreign key)* (เช่น รีเลชั่น A มีแอตทริบิวรหัสประจำตัวเป็นคีย์หลัก แล้วในรีเลชั่น B มีแอตทริบิวรหัสประจำตัวเช่นเดียวกับ A เราจะเรียก แอตทริบิวดังกล่าวของรีเลชั่น B ว่าเป็นคีย์นอกของคีย์หลักของรีเลชั่น A) ในกรณีที่รีเลชั่นมีแอตทริบิวหรือกลุ่มแอตทริบิวที่มีคุณสมบัติเป็นคีย์หลักได้อยู่หลายแอตทริบิว เราจะเรียกแอตทริบิวหรือกลุ่มแอตทริบิวที่มิได้ถูกเลือกให้เป็นคีย์หลักว่า *คีย์คู่แข่ง (candedate key)* หรือ *คีย์สำรอง (alternate key)* และแอตทริบิวอื่น ๆ ที่เหลือที่มิได้เป็นคีย์หลักและไม่ได้เป็นส่วนใดส่วนหนึ่งของคีย์หลักก็จะถูกเรียกว่า *นันคีย์ (nonkey attribute)*

คำว่า *โดเมน (domain)* จะหมายถึง กรอบของค่าต่าง ๆ ที่เป็นไปได้ เช่น โดเมนของแอตทริบิววันที่ ก็จะหมายถึงค่าของวันที่ที่เป็นไปได้ คือ มีค่าเท่ากับ 1 ถึง 31 ในเดือนที่ลงท้ายด้วยคำว่า "คม", มีค่าเท่ากับ 1 ถึง 30 ในเดือนที่ลงท้ายด้วยคำว่า "ยน" และในเดือนกุมภาพันธ์อาจมีค่าเท่ากับ 1 ถึง 28 หรือ 29 ก็ได้ แต่ในการเก็บค่าข้อมูลลงในรีเลชั่นนั้น บางกรณีที่เรามีการกำหนดโดเมนให้กับแอตทริบิวแล้ว แต่ข้อมูลที่จะถูกเก็บเข้าไปอาจถูกบรรจุเข้าไปในภายหลัง ลักษณะนี้จะคำให้เกิด *ค่าว่าง (Null value)* ขึ้นชั่วขณะก่อนที่จะมีการบรรจุค่าข้อมูลที่อยู่ในโดเมนที่กำหนดไว้เข้าไป ดังนั้น คำว่า "ค่าว่าง" จึงหมายถึงค่าที่ยังมิทราบชัดว่าแอตทริบิวนั้นจะมีค่าเป็นค่าใด หรือ ค่าของข้อมูลที่ไม่อยู่ในโดเมนที่กำหนด โดยมีข้อบังคับไว้ว่า แอตทริบิวที่ทำหน้าที่เป็นคีย์หลักของรีเลชั่นจะมีค่าข้อมูลเป็นค่าว่างไม่ได้เสมอ เพราะจะทำให้การเข้าถึงข้อมูลในทัพเพิล (tuple) นั้นกระทำไม่ได้

เมื่อมีการจัดเก็บข้อมูลในฐานข้อมูลใด ๆ แล้ว ข้อมูลจะถูกแยกออกเป็นกลุ่มของข้อมูลเป็นชุดที่ประกอบด้วยแอตทริบิวต่าง ๆ ที่มีความสัมพันธ์กัน เช่น การเก็บข้อมูลของบุคลากรในโรงเรียนก็อาจแยกเก็บเป็นกลุ่มข้อมูลของนักเรียน, กลุ่มข้อมูลของครูอาจารย์, และกลุ่มข้อมูลของนักการภารโรง เป็นต้น กลุ่มข้อมูลแต่ละกลุ่มนี้จะเรียกว่า *เอนติตี้ (entity)* ซึ่งแต่ละเอนติตี้จะประกอบไปด้วยแอตทริบิวต่าง ๆ ที่มีความสัมพันธ์กัน เช่น เอนติตี้ของนักเรียนก็จะประกอบไปด้วยชื่อ, นามสกุล, ที่อยู่, ชั้นเรียน เป็นต้น

จากการแยกจัดเก็บข้อมูลออกเป็นเอนติตี้นี้ แต่ละเอนติตี้ก็จะมีความสัมพันธ์กัน ความสัมพันธ์ระหว่างเอนติตี้สามารถแบ่งออกเป็น 3 ชนิด คือ

- ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (one to one)

- ความสัมพันธ์แบบหนึ่งต่อกลุ่ม (one to many)
- ความสัมพันธ์แบบกลุ่มต่อกลุ่ม (many to many)

ความสัมพันธ์แบบหนึ่งต่อหนึ่ง

ความสัมพันธ์แบบหนึ่งต่อหนึ่งระหว่างเอนติตี้ก็หมายความว่า เมื่อเอนติตี้หนึ่งมีข้อมูลของคีย์หลักค่าหนึ่ง ค่าข้อมูลดังกล่าวก็จะมีความสัมพันธ์กับค่าข้อมูลของคีย์หลักของอีกเอนติตี้หนึ่งเพียงค่าเดียวเท่านั้น เช่น หากเรากำหนดให้ความสัมพันธ์ระหว่างเอนติตี้นักเรียนกับเอนติตี้ผู้ปกครองเป็นแบบหนึ่งต่อหนึ่งแล้วก็จะหมายความว่า การที่เราอ้างถึงนักเรียนคนใดคนหนึ่งก็จะสามารถอ้างถึงผู้ปกครองได้เพียงคนเดียวเท่านั้น และในการตรงกันข้ามก็ต้องเป็นจริงด้วย คือ เมื่อเราอ้างถึงผู้ปกครองคนใดคนหนึ่งแล้วก็จะสามารถอ้างอิงถึงนักเรียนได้เพียงคนเดียวเท่านั้น

ชื่อนักเรียน	ชื่อผู้ปกครอง
A	a
B	b
C	c

ความสัมพันธ์แบบหนึ่งต่อกลุ่ม

ความสัมพันธ์แบบหนึ่งต่อกลุ่มระหว่างเอนติตี้ก็หมายความว่า เมื่อเอนติตี้หนึ่งมีข้อมูลของคีย์หลักค่าหนึ่ง ค่าข้อมูลดังกล่าวก็จะมีความสัมพันธ์กับค่าข้อมูลของคีย์หลักของอีกเอนติตี้หนึ่งได้หลายค่า เช่น หากเรากำหนดให้ความสัมพันธ์ระหว่างเอนติตี้นักเรียนกับเอนติตี้ผู้ปกครองเป็นแบบหนึ่งต่อกลุ่มแล้วก็จะหมายความว่า การที่เราอ้างถึงนักเรียนคนใดคนหนึ่งก็จะสามารถอ้างถึงผู้ปกครองได้หลายคนเดียว และในการตรงกันข้ามก็จะมีความหมายว่า เมื่อเราอ้างถึงผู้ปกครองคนใดคนหนึ่งแล้วก็จะสามารถอ้างอิงถึงนักเรียนได้เพียงคนเดียวเท่านั้น แต่ผู้ปกครองที่เราอ้างถึงเป็นคนละคนกันก็อาจจะอ้างอิงถึงนักเรียนคนเดียวกันก็เป็นได้

ชื่อนักเรียน	ชื่อผู้ปกครอง
A	a
B	a
C	c

ความสัมพันธ์แบบกลุ่มต่อกลุ่ม

ความสัมพันธ์แบบกลุ่มต่อกลุ่มระหว่างเอนติตี้ก็หมายความว่า ค่าข้อมูลของคีย์หลักของเอนติตี้หนึ่งที่ต่างกันอาจอ้างอิงถึงค่าข้อมูลของคีย์หลักของอีกเอนติตี้หนึ่งได้ค่าเดียวหรือหลายค่าก็ได้ เช่น หากเรากำหนดให้ความสัมพันธ์ระหว่างเอนติตี้นักเรียนกับเอนติตี้ผู้ปกครองเป็นแบบกลุ่มต่อกลุ่มแล้วก็จะหมายความว่าการที่เราอ้างถึงนักเรียนคนหนึ่งหรือหลายคนก็จะสามารถอ้างอิงถึงผู้ปกครองคนเดียวกันก็ได้ และในทางกลับกัน การที่เราอ้างถึงผู้ปกครองคนหนึ่งหรือหลายคนก็จะสามารถอ้างอิงถึงนักเรียนคนเดียวกันก็ได้

ชื่อนักเรียน	*ชื่อผู้ปกครอง*
A	a
B	a
C	c
C	d

เมื่อเราได้แอตทริบิวต่าง ๆ มาประกอบกันเป็นเอนติตี้ และสามารถกำหนดความสัมพันธ์ระหว่างเอนติตี้ได้แล้ว ขั้นตอนต่อไปก็คือการเขียนภาพแสดงความสัมพันธ์ระหว่างเอนติตี้ต่าง ๆ ด้วยโมเดล ER (ER-model)

นอกจากนี้ เอนติตี้ยังมีเอนติตี้บางประเภทที่การอ้างถึงตัวมันได้อย่างสมบูรณ์นั้นจะต้องอ้างถึงเอนติตี้อื่นเสมอ เช่น หากระบบฐานข้อมูลของบุคลากรในโรงเรียนมีเอนติตี้ของลูกนักการภารโรงด้วยแล้ว การอ้างอิงถึงเอนติตี้นี้ได้จะต้องอ้างถึงเอนติตี้นักการภารโรงด้วยเสมอ เอนติตี้ประเภทเราเรียกว่า *เอนติตี้ชนิดอ่อน (Weak entity)*

2.4.2 โมเดล ER (ER-model)

โมเดล ER (ER-model : Entiry Relationship model) เป็นแผนภาพที่ถูกออกแบบมาเพื่อแสดงความสัมพันธ์ระหว่างเอนติตี้ต่าง ๆ ในรูปแบบที่เป็นรูปธรรมมากขึ้น เมื่อมีการใช้โมเดล ER แสดงความสัมพันธ์ระหว่างเอนติตี้แล้วก็จะไม่ต้องมีคำอธิบายความสัมพันธ์ใด ๆ อีก เพราะโมเดล ER ประกอบไปด้วยสัญลักษณ์ต่าง ๆ ที่แสดงถึงคุณลักษณะของเอนติตี้และแอตทริบิวได้ในตัวเองแล้ว

หลังจากที่เราสามารถหาความสัมพันธ์ระหว่างข้อมูลต่าง ๆ โดยแสดงความสัมพันธ์ของข้อมูลเหล่านั้นด้วยโมเดล ER แล้ว ขั้นตอนต่อไปนี้คือการเปลี่ยนความสัมพันธ์ของข้อมูลที่อยู่บนโมเดล ER ให้อยู่ในรูปของรีเลชั่น ซึ่งการเปลี่ยนความสัมพันธ์ของข้อมูลที่อยู่บนโมเดล ER ให้อยู่ในรูปของรีเลชั่นนั้นจะมีขั้นตอนในการเปลี่ยนอยู่ 7 ขั้นตอน หลังจากนั้นก็ต้องทำการปรับปรุงรีเลชั่นที่ได้ให้มีความซ้ำซ้อนของความสัมพันธ์ระหว่างข้อมูลหมดไป(หรือเหลือน้อยที่สุดเพื่อความเหมาะสมในการ

037066

นำไปใช้งานจริง) ซึ่งการปรับปรุงดังกล่าวก็จะมีทฤษฎีที่จะต้องอ้างถึงคือ กฎของความคงสภาพ (Integrity rule), ฟังก์ชันการขึ้นต่อกัน(Functional dependency) และทฤษฎีที่สำคัญที่สุดก็คือ ทฤษฎีการนอร์มัลไลซ์ (Normalization) ดังจะกล่าวต่อไปนี้

2.4.2.1 การแปลงความสัมพันธ์ของข้อมูลจากโมเดล ER ไปสู่ในรูปของรีเลชั่น

การแปลงความสัมพันธ์ของข้อมูลจากโมเดล ER ไปสู่ในรูปของรีเลชั่นมีขั้นตอนดังนี้

1. สำหรับแต่ละเอนติตี้ที่ไม่ใช่เอนติตี้แบบอ่อนของโมเดล ER เราจะสร้างเป็นรีเลชั่นโดยมีทุกแอตทริบิวที่เป็นแอตทริบิวธรรมดา (simple attribute) มาประกอบกัน แล้วทำการเลือกแอตทริบิวใดแอตทริบิวหนึ่งหรือกลุ่มของแอตทริบิวมาทำหน้าที่เป็นคีย์หลักของรีเลชั่น
2. สำหรับแต่ละเอนติตี้แบบอ่อน เราจะสร้างรีเลชั่นที่เกิดจากการรวมกันของแอตทริบิวธรรมดาของเอนติตี้นั้น โดยที่รีเลชั่นนี้จะมีคีย์หลักคือคีย์รวม (combine key)ที่เกิดจากการรวมกันของคีย์หลักของเอนติตี้แบบอ่อน(partial key)กับคีย์หลักของเอนติตี้ที่มันต้องอ้างถึง(ในกรณีนี้คีย์นี้จะเรียกว่าเป็นคีย์นอก)
3. สำหรับความสัมพันธ์ระหว่างเอนติตี้แบบหนึ่งต่อหนึ่ง เราจะสร้างรีเลชั่นจากความสัมพันธ์ดังกล่าวได้ 2 ลักษณะ คือ เลือกคีย์หลักของเอนติตี้ใดเอนติตี้หนึ่งมาเป็นคีย์หลักของรีเลชั่นนี้ แล้วให้คีย์หลักของอีกเอนติตี้หนึ่งมาเป็นคีย์นอกของรีเลชั่นนี้ โดยถ้าความสัมพันธ์นี้มีแอตทริบิวก็ให้นำแอตทริบิวเหล่านั้นมารวมอยู่ในรีเลชั่นนี้ด้วย
4. สำหรับความสัมพันธ์ระหว่างเอนติตี้แบบหนึ่งต่อกลุ่ม เราจะสร้างรีเลชั่นจากความสัมพันธ์ดังกล่าวโดยนำเอาคีย์หลักของเอนติตี้ฝั่งที่มีความสัมพันธ์แบบกลุ่มมาเป็นคีย์หลักของรีเลชั่นนี้ แล้วให้นำเอาคีย์หลักของเอนติตี้ฝั่งที่มีความสัมพันธ์แบบหนึ่งมาเป็นคีย์นอกของรีเลชั่นนี้ โดยถ้าความสัมพันธ์นี้มีแอตทริบิวก็ให้นำเอาแอตทริบิวเหล่านั้นมารวมอยู่ในรีเลชั่นนี้ด้วย
5. สำหรับความสัมพันธ์ระหว่างเอนติตี้แบบกลุ่มต่อกลุ่ม เราจะสร้างรีเลชั่นจากความสัมพันธ์นี้โดยนำเอาคีย์หลักของทั้งสองเอนติตี้มาประกอบกันเป็นคีย์หลักของรีเลชั่นนี้ โดยถ้าความสัมพันธ์นี้มีแอตทริบิวก็ให้นำเอาแอตทริบิวเหล่านั้นมารวมอยู่ในรีเลชั่นนี้ด้วย
6. สำหรับเอนติตี้ใดที่มีแอตทริบิวที่มีค่าข้อมูลแบบหลายค่า (multivalue attribute หรือ repeating group) ก็ให้สร้างรีเลชั่นใหม่โดยมีคีย์หลักของเอนติตี้นั้นรวมกับแอตทริบิวดังกล่าวเป็นคีย์หลักของรีเลชั่นนี้
7. สำหรับความสัมพันธ์ระหว่างเอนติตี้ที่เกิดจากเอนติตี้มากกว่า 2 เอนติตี้ ให้สร้างรีเลชั่นของความสัมพันธ์นี้โดยนำคีย์หลักของทุกเอนติตี้มาประกอบกันเป็นคีย์หลักของเอนติตี้นี้ โดยถ้าความสัมพันธ์นี้มีแอตทริบิว ก็ให้นำแอตทริบิวเหล่านั้นมารวมอยู่ในรีเลชั่นนี้ด้วย

เมื่อทำการเปลี่ยนความสัมพันธ์ระหว่างข้อมูลที่อยู่ในรูปของโมเดล ER มาอยู่ในรูปของรีเลชั่นแล้ว เราจะได้รีเลชั่นที่อยู่ในรูปแบบนอร์มัลระดับที่ 1 ซึ่งจะต้องมีการทำการนอร์มัลไลซ์ต่อไปเสียก่อนจึงจะสามารถนำรีเลชั่นเหล่านี้ไปใช้งานในระบบจัดการฐานข้อมูลเชิงสัมพันธ์ได้

2.4.3 ทฤษฎีของระบบฐานข้อมูลเชิงสัมพันธ์

2.4.3.1 กฎของความคงสภาพ (Integrity rule)

กฎของความคงสภาพ (Integrity rule) ของโมเดลเชิงสัมพันธ์ (Relational model) เป็นทฤษฎีที่ช่วยยืนยันความถูกต้องของความสัมพันธ์ระหว่างข้อมูลว่า รีเลชั่นใดที่เป็นไปตามกฎของความคงสภาพนี้แล้วย่อมจะมีความสัมพันธ์ระหว่างข้อมูลอย่างถูกต้องอยู่ตลอดเวลา ไม่ว่ารีเลชั่นนั้นจะมีการเปลี่ยนแปลงแก้ไขข้อมูลไปในรูปแบบใดก็ตาม

กฎของความคงสภาพมีความหมายอยู่ 2 ลักษณะ คือ *กฎความคงสภาพของ entiry (entity integrity rule)* และ *กฎความคงสภาพของการอ้างอิง (referential integriry rule)* ดังอธิบายได้ดังนี้

1. **กฎความคงสภาพของเอนติตี้** กล่าวว่า

"แอตทริบิวทุกตัวที่เป็นส่วนของคีย์หลักจะไม่อนุญาตให้มีค่าว่าง"

หมายความว่า คีย์หลักของทุกรีเลชั่นจะไม่สามารถเก็บค่าข้อมูลที่เป็นค่าว่างได้ เหตุผลของข้อกำหนดนี้ก็คือ เพื่อให้การเข้าถึงข้อมูลในทัพเพิลใด ๆ ของรีเลชั่นมีความเป็นไปได้เสมอ เพราะถ้าคีย์หลักของทัพเพิลใดมีค่าข้อมูลเป็นค่าว่างแล้ว ก็จะส่งผลให้การเข้าถึงข้อมูลในทัพเพิลนั้นไม่สามารถกระทำได้อย่างแน่นอน

2. **กฎความคงสภาพของการอ้างอิง** กล่าวว่า

"ถ้าเรามีรีเลชั่น R2 ซึ่งมี FK เป็นคีย์นอกที่อ้างอิงถึงคีย์หลัก PK ใน รีเลชั่น R1 สำหรับทุกค่าของ FK ใน R2 จะต้อง

ก. มีค่าเท่ากับค่า PK ในทัพเพิลใดทัพเพิลหนึ่งในรีเลชั่น R1

หรือ ข. มีค่าของแอตทริบิวทุกตัวใน FK เป็นค่าว่าง"

หมายความว่า แอตทริบิวใด ๆ ที่เป็นคีย์หลักของรีเลชั่นหนึ่ง เมื่อมีการนำแอตทริบิวนั้นไปเป็นคีย์นอกของอีกรีเลชั่นหนึ่ง การเป็นคีย์นอกของแอตทริบิวนั้นจะต้องมีโดเมนเป็นโดเมนเดียวกันกับแอตทริบิวที่เป็นคีย์หลัก ทั้งนี้ ก็เพื่อให้การนำรีเลชั่นมาใช้งานร่วมกัน (การนำรีเลชั่นมา join กัน) กระทำได้อย่างถูกต้อง คือ ทุกแอตทริบิวที่เป็นคีย์นอกจะต้องมีข้อมูลซ้ำกับข้อมูลของแอตทริบิวที่เป็นคีย์หลักอย่างแน่นอน แต่อาจมีบางค่าข้อมูลของแอตทริบิวที่เป็นคีย์หลักเป็นข้อมูลไม่อยู่ในโดเมนของแอตทริบิวที่เป็นคีย์นอกก็ได้ นั่นคือ โดเมนของคีย์นอกจะต้องเล็กกว่าหรือเท่ากับโดเมนของคีย์หลักเสมอ

รีเลชั่น R1

คีย์หลักของ R	*คีย์อื่น ๆของ R*
A	1
B	2
C	3
D	4

รีเลชั่น R2

คีย์หลักของ R	*คีย์นอกของ R*
a	A
b	B
c	B
d	C

2.4.3.2 ฟังก์ชันการขึ้นต่อกัน (Functional dependency)

ฟังก์ชันการขึ้นต่อกันเป็นข้อกำหนดที่ช่วยให้เราเห็นถึงความสัมพันธ์ของแอตทริบิวต่าง ๆ ที่อยู่ในรีเลชั่น ทั้งเพราะแอตทริบิวต่าง ๆ ที่อยู่ในเอนติตี้เดียวกันก็เป็นไปได้ที่แอตทริบิวเหล่านั้นจะมีความสัมพันธ์กันเองโดยที่ความสัมพันธ์นี้อาจเกี่ยวข้องหรือไม่เกี่ยวกับความสัมพันธ์ที่มันมีต่อคีย์หลักของเอนติตี้นั้นก็เป็นได้ ซึ่งการที่แอตทริบิวเหล่านั้นมีความสัมพันธ์กันเองจะเป็นสิ่งที่เราต้องพิจารณาแยกออกเป็นรีเลชั่นย่อย ๆ เพราะแอตทริบิวของแต่ละรีเลชั่นก็ควรจะมีความสัมพันธ์กับคีย์หลักของรีเลชั่นของตนเองเท่านั้น

กำหนดรีเลชั่น R ถ้ามีแอตทริบิว Y ของ R เป็นฟังก์ชันที่ขึ้นต่อแอตทริบิว X ของรีเลชั่น เราสามารถเขียนแทนได้ด้วยสัญลักษณ์

R.X - - -> R.Y

อ่านว่า R.X มีฟังก์ชันการขึ้นอยู่กับ R.Y

หรือ R.X มีฟังก์ชันในการเลือก R.Y

หรือ R.Y ขึ้นอยู่กับ R.X

นิยาม R.X มีฟังก์ชันการขึ้นอยู่กับ R.Y ก็ต่อเมื่อ ทุกค่าข้อมูลของแอตทริบิว X ใน R จะมีค่าข้อมูลของแอตทริบิว Y ใน R ได้เพียงค่าเดียวเสมอ โดยที่แอตทริบิว X และ Y อาจจะเป็นคีย์แบบรวม (composite key) ก็ได้

รีเลชั่น R

X	Y
a	1
b	2
a	1
b	1

นิยาม R.X มีฟังก์ชันการขึ้นอยู่กับ R.Y อย่างเต็มที่ (R.Y fully functionally dependent on R.X) ก็ต่อเมื่อ R.Y มีฟังก์ชันการขึ้นอยู่กับ R.X และไม่ขึ้นอยู่กับข้อมูลเพียงบางส่วนของ R.X โดยที่แอตทริบิว X และ Y อาจจะเป็นคีย์แบบรวมก็ได้

รีเลชั่น R

X		Y
A	a	1
B	b	2
A	a	1
B	c	3

2.4.3.3 การนอร์มัลไลซ์ (Normalization)

การนอร์มัลไลซ์เป็นการออกแบบฐานข้อมูลแบบที่เป็นมาตรฐานที่สุด ออกแบบโดย Codd โดยมีจุดประสงค์ของการออกแบบก็เพื่อลดความซ้ำซ้อนของความสัมพันธ์ของข้อมูลให้เหลือน้อยที่สุด (minimum redundancy) ซึ่งตามมาตรฐานปกติจะมีอยู่ 3 ระดับ คือ

1NF (First Normal Form)

2NF (Second Normal Form)

3Nf (Third Normal Form)

โดยที่รีเลชันใดที่ยังไม่สอดคล้องตามรูปแบบนอร์มัล (Normal Form) ทั้งสามก็จะต้องมีการแยกรีเลชันนั้น ๆ ออกเป็นรีเลชันย่อย ๆ ต่อไปอีก (decomposition method)

ต่อมาได้มีการออกแบบเพิ่มเติมขึ้นอีก 2 ระดับ คือ

4NF (Forth Normal Form)

5NF (Fifth Normal Form)

หากรีเลชันใดมีมาตรฐานถึงรูปแบบนอร์มัลระดับที่ 5 (5NF) แล้วก็จะมั่นใจได้ว่า รีเลชันนั้นจะไม่มีความซ้ำซ้อนของความสัมพันธ์ของข้อมูลอย่างแน่นอน

นอกจากนี้ยังมีการออกแบบรูปแบบนอร์มัลเพิ่มเติมระหว่างรูปแบบนอร์มัลที่ 3 (3NF) และรูปแบบนอร์มัลที่ 4 (4NF) โดย Boyce และ Codd ซึ่งมีชื่อว่า Boyce Codd Normal Form (BCNF) อีกด้วย

รูปแบบนอร์มัลระดับที่1

การปรับรีเลชั่นให้อยู่ในรูปแบบนอร์มัลระดับที่ 1 คือ การปรับจากรีเลชั่นที่ไม่นอร์มัล (unnormalized relation) เช่น รีเลชั่นที่มีข้อมูลของแอตทริบิวบางช่องมีมากกว่า 1 ค่า (มีแอตทริบิวที่มีข้อมูลเป็น repeating group)

นิยาม รีเลชั่นจะอยู่ในรูปแบบนอร์มัลระดับที่ 1 (1NF) ก็ต่อเมื่อโดเมนของแต่ละแอตทริบิวประกอบด้วยข้อมูลที่เป็นหน่วยย่อยที่สุด (A relation is in first normal form (1NF) if and only if all underlying simple domains contain atomic values only.)

สิ่งที่ได้จากการที่รีเลชั่นอยู่ในรูปแบบนอร์มัลระดับที่ 1 ก็คือ รีเลชั่นยังคงมีความซ้ำซ้อนของความสัมพันธ์ระหว่างข้อมูลอยู่มากมาย เพราะนิยามของรูปแบบนอร์มัลระดับที่ 1 นี้กำหนดเพียงเฉพาะว่า แต่ละแอตทริบิวของรีเลชั่นจะมีโดเมนที่มีสมาชิกเป็นหน่วยที่เล็กที่สุดเท่านั้น มิได้เป็นการลดความซ้ำซ้อนของความสัมพันธ์ระหว่างข้อมูลแต่ประการใด

รูปแบบนอร์มัลระดับที่ 2

นิยาม รีเลชั่นจะอยู่ในรูปแบบนอร์มัลระดับที่ 2 (2NF) ก็ต่อเมื่อรีเลชั่นนั้นอยู่ในรูปแบบนอร์มัลระดับที่ 1 แล้ว และทุกแอตทริบิวที่ไม่เป็นส่วนใดส่วนหนึ่งของคีย์หลัก (non-key attribute) จะต้องขึ้นอยู่กับคีย์หลักของรีเลชั่นนั้นอย่างเต็มที่ (A relation is in 2NF if and only if it is in 1NF and every nonkey attribute fully depends on the primary key)

สิ่งที่ได้จากการที่รีเลชั่นอยู่ในรูปแบบนอร์มัลระดับที่ 2 คือ ข้อมูลของบางแอตทริบิวที่ไม่ใช่คีย์หลักอาจมีความสัมพันธ์กันเองโดยที่ไม่มีความสัมพันธ์กับคีย์หลักเลย ซึ่งความสัมพันธ์ดังกล่าวนี้ถือว่าเป็นความซ้ำซ้อนประการหนึ่งของรีเลชั่นนั้น ๆ ที่จะต้องทำการลดด้วยรูปแบบนอร์มัลในระดับต่อไป

รูปแบบนอร์มัลระดับที่ 3

นิยาม รีเลชั่นจะอยู่ในรูปแบบนอร์มัลระดับที่ 3 (3NF) ก็ต่อเมื่อรีเลชั่นนั้นอยู่ในรูปแบบนอร์มัลระดับที่ 2แล้ว และทุกแอตทริบิวที่ไม่เป็นส่วนใดส่วนหนึ่งของคีย์หลักจะต้องไม่เป็นฟังก์ชันที่ขึ้นต่อกันเอง (A relation is in third normal form if and only if it is in 2NF and every nonkey attribute is nontransitively dependent on the primary key)

โดยปกติแล้ว สิ่งที่ได้จากการที่รีเลชั่นอยู่ในรูปแบบนอร์มัลระดับที่ 3 คือ รีเลชั่นจะไม่มีความซ้ำซ้อนอีกต่อไปโดยที่จะสอดคล้องกับรูปแบบนอร์มัลระดับที่ 4 และ 5 ด้วย แต่ก็มีรีเลชั่นบางลักษณะที่จะต้องทำให้อยู่ในรูปแบบนอร์มัลระดับที่ 4 และ 5 ต่อไปเอง รีเลชั่นดังกล่าวจะมีลักษณะดังต่อไปนี้

1.) เป็นรีเลชั่นที่มีหลายคีย์คู่แข่ง และ

2.) เป็นคีย์คู่แข่งที่เกิดจากการรวมกันของคีย์ย่อย ๆ (candidate key เป็น combine key) และ

3.) เป็นคีย์คู่แข่งที่มีการเหลื่อมซ้อนกัน (overlap กัน)

เนื่องจากรีเลชั่น ณ ที่นี้ไม่มีลักษณะดังกล่าวจึงไม่ขอกล่าวถึงรูปแบบนอร์มัลแบบ BCNF, รูปแบบนอร์มัลระดับที่ 4 และ 5

2.5 สรุป

ในบทนี้ได้กล่าวถึงทฤษฎีที่จะนำไปใช้ในบทต่อๆไป ประกอบด้วยเนื้อหา 3 เรื่องใหญ่ๆด้วยกันได้แก่ ไคลน์เซิร์ฟเวอร์คืออะไร โอดีบีซีคืออะไร เดลไฟคืออะไร และท้ายสุดได้กล่าวถึงหลักการโมเดลเชิงสัมพันธ์

ไคลน์เซิร์ฟเวอร์นั้นหมายถึง ระบบใดๆที่มีการแบ่งการประมวลผลข้อมูลกันระหว่างคอมโพเนนต์หรือส่วนประกอบที่แยกกัน โดยทางไคลน์พีซีจะวิ่งโปรแกรมประยุกต์ (front-end system) ส่วนดาต้าเบสเซิร์ฟเวอร์จะวิ่งโปรแกรมดีบีเอ็มเอส(back-end system) ในการติดต่อสื่อสารกัน ฝั่งไคลน์จะส่งคิวรี่ที่ต้องการไปยังเซิร์ฟเวอร์ จากนั้นเซิร์ฟเวอร์จะค้นหาข้อมูลและส่งเฉพาะข้อมูลนั้นกลับมายังฝั่งไคลน์

หลังจากที่ได้กล่าวถึงความหมายแล้วในบทนี้ก็ได้กล่าวถึงข้อดี-ข้อเสีย ของระบบไคลน์เซิร์ฟเวอร์ และแพล็ทฟอร์ม, ฮาร์ดแวร์, ซอร์ฟแวร์ และเน็ตเวอร์คโปรโตคอล ของระบบฐานข้อมูลไคลน์เซิร์ฟเวอร์

โอดีบีซี คือวิธีที่ทำให้แอพพลิเคชั่นสามารถติดต่อกับเซิร์ฟเวอร์ได้หลายชนิดโดยใช้ภาษาเอสคิวแอลมาตรฐาน โอดีบีซีมีส่วนประกอบสำคัญอยู่ 4 ส่วนด้วยกัน ได้แก่ แอพพลิเคชั่น,ตัวจัดการไดร์เวอร์, ไดร์เวอร์ และแหล่งข้อมูล

นอกจากได้กล่าวถึงความหมายและส่วนประกอบของโอบีดีซีแล้ว ในบทนี้เรายังได้กล่าวถึงข้อดีของการติดต่อโดยใช้โอบีดีซี และการทำงานเมื่อเรียกใช้โอบีดีซี

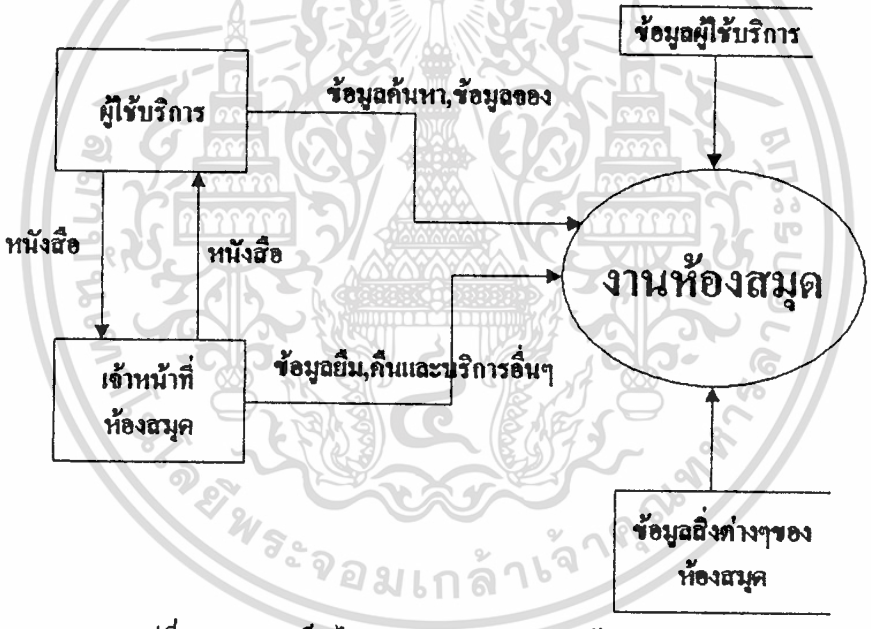
เดลไฟ คือวิชวลปาสคาลเพื่อการโปรแกรมบนวินโดวส์นั่นเอง

ในบทนี้เราได้กล่าวถึงส่วนประกอบของโปรแกรมในชุดเดลไฟ,หน้าตาของเดลไฟ,ข้อดีของเดลไฟ และการนำเดลไฟมาใช้ในไคลน์เซิร์ฟเวอร์แอพพลิเคชั่น โดยได้กล่าวถึงเรื่องพอร์ตเทบิลิตี้, ออปติไมเซชั่น,การเชื่อมต่อ และทรานแอ็คชั่นของเดลไฟ

ในส่วนสุดท้ายเราได้กล่าวถึงหลักการโมเดลเชิงสัมพันธ์โดยได้เริ่มกล่าวถึงนิยามของฐานข้อมูลเชิงสัมพันธ์ ต่อมาได้กล่าวถึงโมเดล ER และการแปลงความสัมพันธ์ของข้อมูลจากโมเดลER ไปสู่ในรูปของรีเลชั่น และส่วนสุดท้ายได้กล่าวถึง ทฤษฎีของระบบฐานข้อมูลเชิงสัมพันธ์ และการนอร์มัลไลซ์

บทที่ 3
ระบบของห้องสมุด

ในบทนี้เป็นขั้นแรกสุดของการทำงานเพื่อจะสร้างโปรแกรมไปประยุกต์กับระบบใดๆระบบหนึ่ง คือต้องเข้าไปเรียนรู้ระบบเดิมให้เข้าใจอย่างลึกซึ้งก่อน การศึกษาระบบของห้องสมุดว่าประกอบด้วยการทำงานอย่างไรบ้าง สามารถแทนการทำงานต่างๆ ด้วยดาต้าโฟลว์ไดอะแกรม (Dataflow Diagram) โดยจะเริ่มจาก ดาต้าโฟลว์ระดับบนสุด หรือเรียกว่าคอนเท็คไดอะแกรม (Context Diagram) แล้วจึงแตกรายละเอียดย่อยลงไปเพื่อให้ทราบข้อมูลในระบบทั้งหมดอย่างสมบูรณ์



รูปที่ 3-1 คอนเท็คไดอะแกรมของระบบห้องสมุด

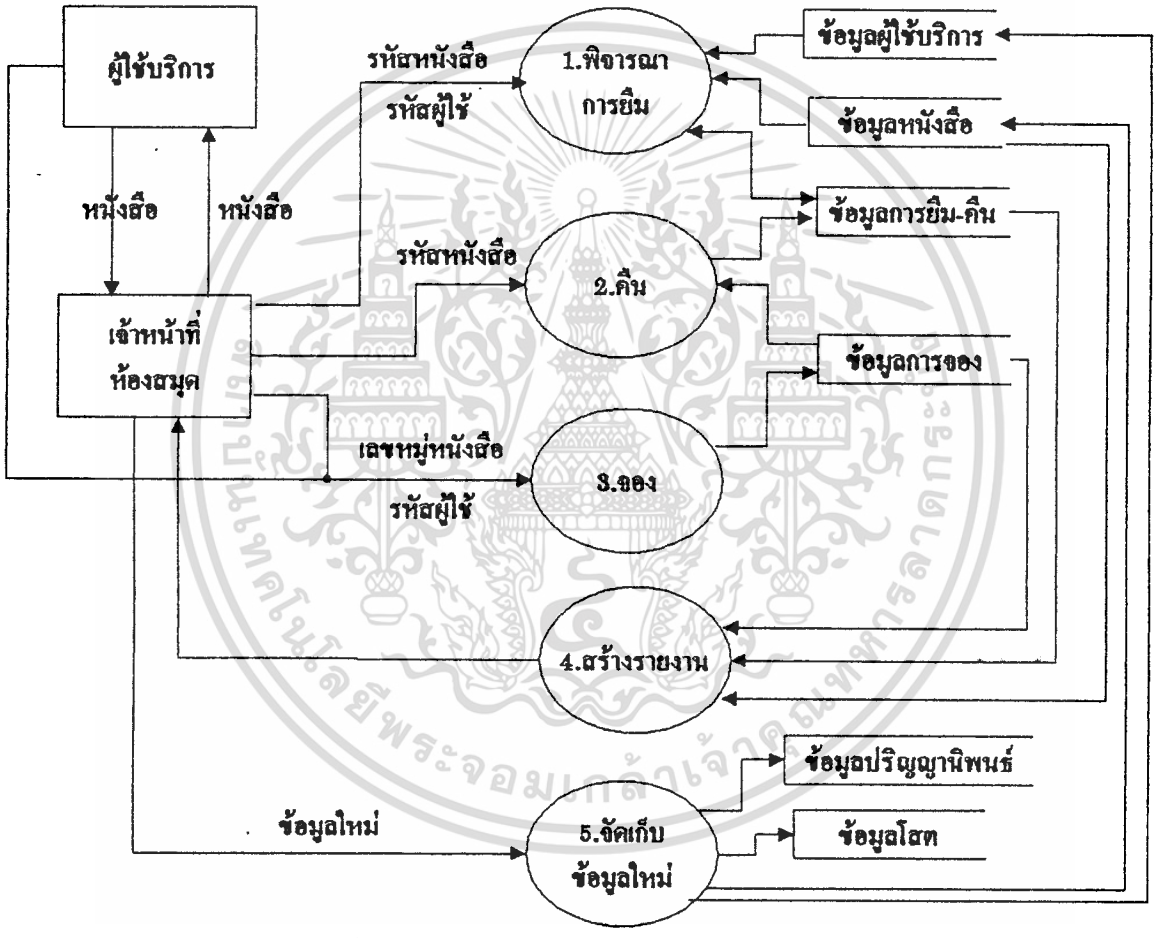
จากคอนเทคไดอะแกรมจะเห็นว่างานของระบบห้องสมุดจะมีงานหลักๆอยู่ 2 งานด้วยกันคือ งานบริการและงานค้นหา งานบริการก็เป็นงานทางด้านบรรณารักษ์ได้แก่งานบริการยืม-คืนหนังสือ งานจัดเก็บข้อมูลหนังสือใหม่ เป็นต้น ส่วนงานค้นหาก็เป็นงานทางด้านผู้ใช้บริการห้องสมุด เช่น งานค้นหาหนังสือตามชื่อเรื่อง ค้นหาหนังสือตามชื่อผู้แต่ง เป็นต้น

ในช่วงต่อไปนี้จะขอกล่าวในรายละเอียดในด้านงานบริการก่อนโดยจะแตกดาต้าโฟลว์ให้ลึกมากที่สุด จากนั้นค่อยมาดูรายละเอียดในส่วนของงานค้นหาต่อไป

3.1 งานบริการ

จากรูปจะเห็นว่างานบริการของห้องสมุดนั้นมี 5 งานใหญ่ๆ คือ

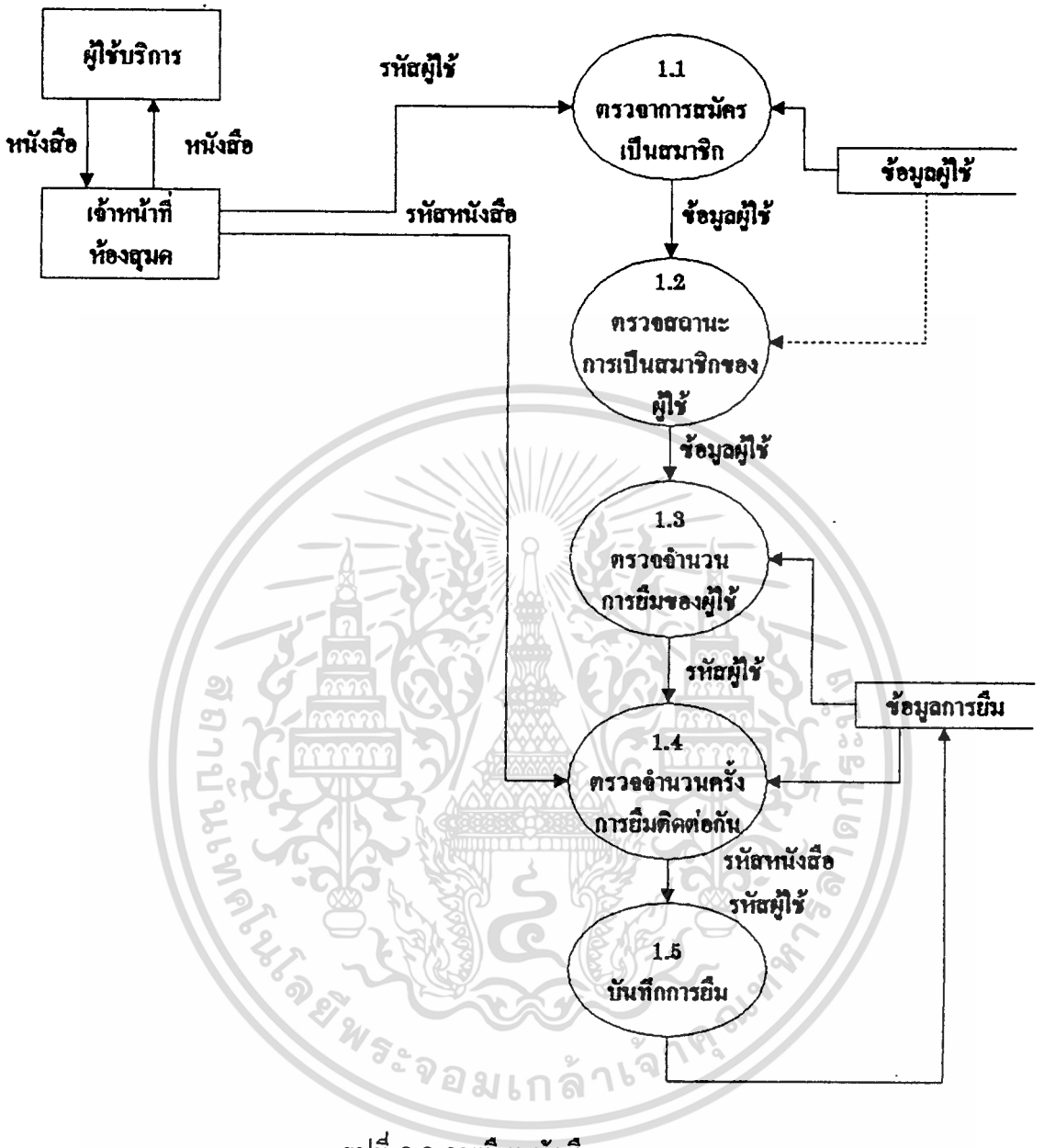
1. การยืมหนังสือ 2. การคืนหนังสือ
3. การจองหนังสือ 4. การสร้างรายงานสถิติต่างๆต่อบรรณารักษ์
5. การจัดเก็บข้อมูล



รูปที่ 3-2 งานบริการห้องสมุด

3.1.1 การยืมหนังสือ

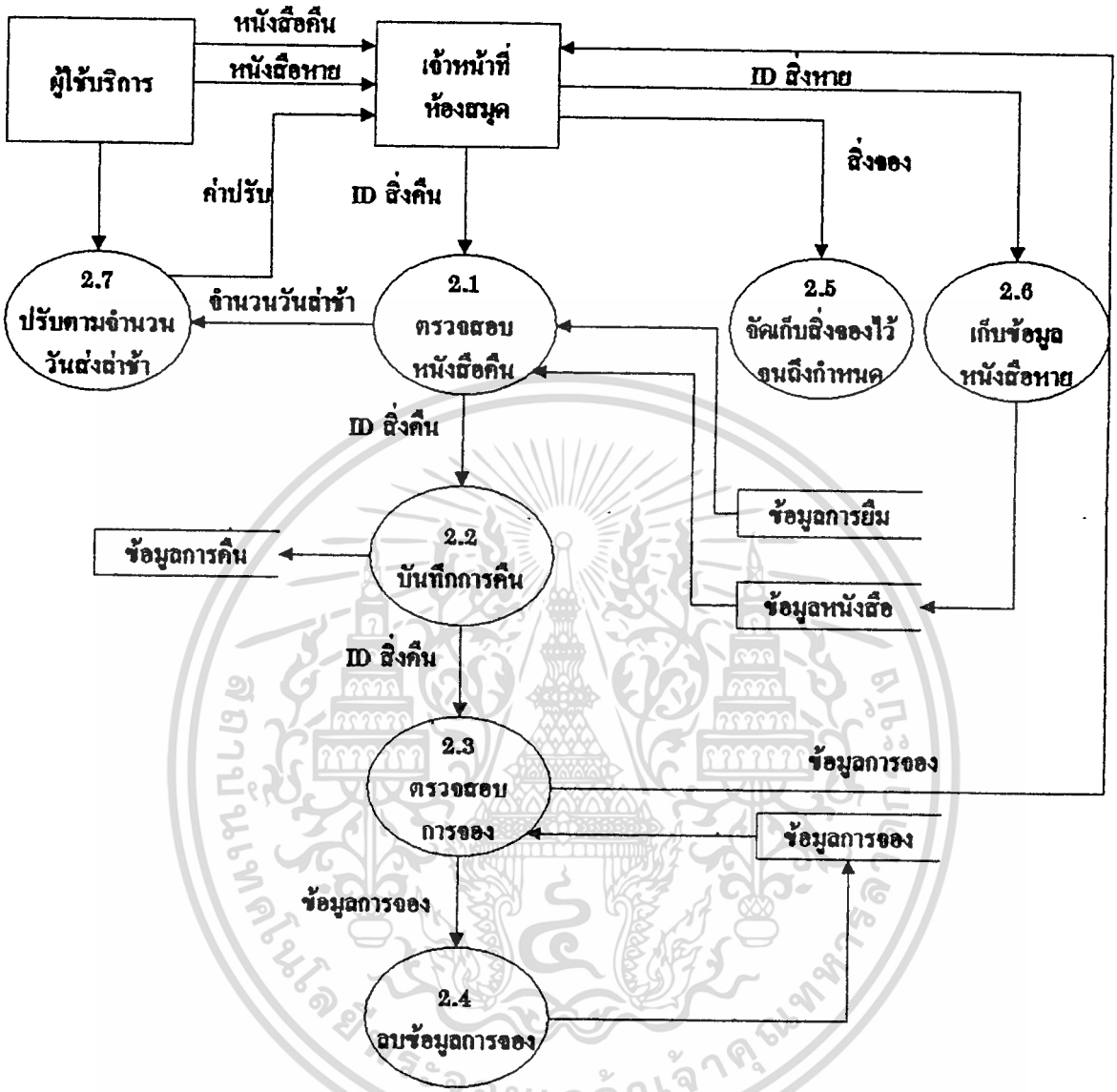
ในการยืมหนังสือนั้นต้องมีการตรวจดูว่า ผู้ใช้บริการเป็นสมาชิกหรือไม่ ถ้ายังต้องทำการเก็บรายละเอียดผู้ใช้ใหม่เสียก่อน ต้องตรวจสถานะของผู้ใช้ว่าหมดอายุสมาชิกหรือยัง ตรวจดูจำนวนสูงสุดที่ให้ยืมหนังสือได้ และต้องตรวจด้วยว่าได้ยืมหนังสือเล่มเดียวกันติดต่อกันเกิน 2 ครั้งหรือไม่ ซึ่งระเบียบกำหนดไว้ว่า ไม่ให้ยืมหนังสือติดต่อกันเกิน 2 ครั้ง ดังรูป 3-3



รูปที่ 3-3 การยืมหนังสือ

3.1.2 การคืนหนังสือ

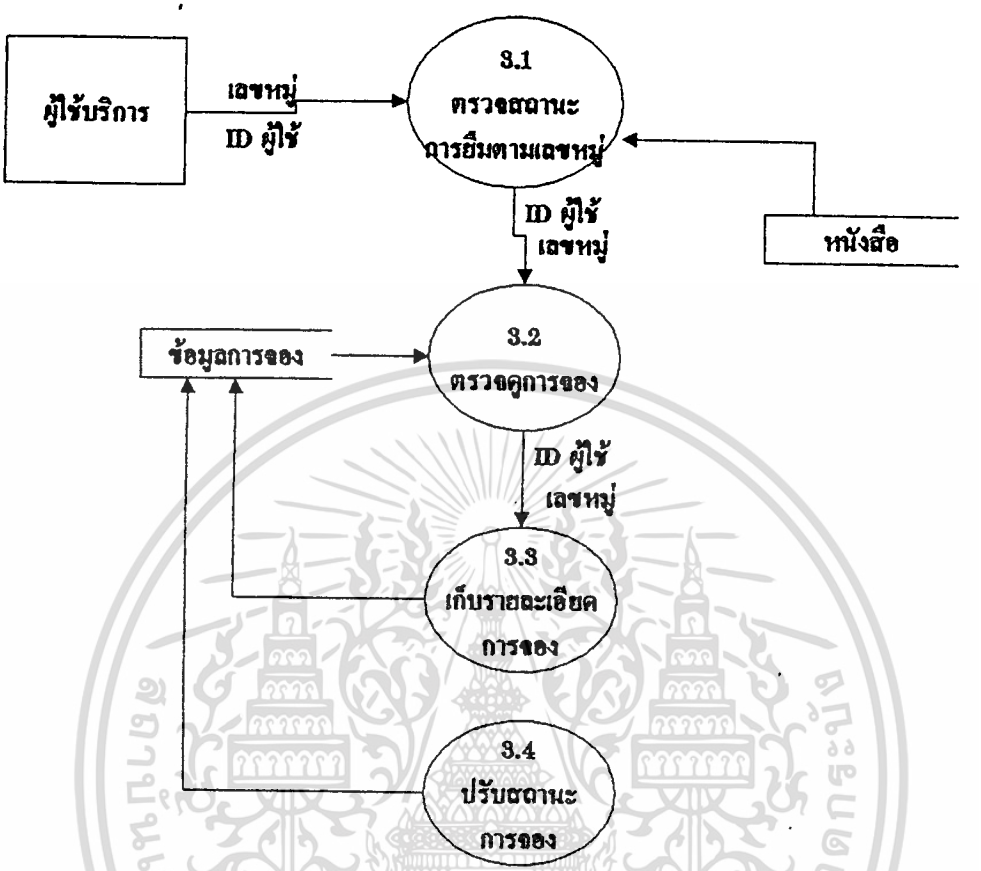
ในการคืนหนังสือจะต้องตรวจสอบว่า หนังสือที่คืนนั้นคืนเกินกำหนด หรือไม่ หากเกินก็ปรับตามจำนวนวัน และจะต้องตรวจด้วยว่ามีคนจองหนังสือเล่มนี้ต่อหรือไม่ ถ้ามีการจอง เจ้าหน้าที่ก็จะทำการเก็บหนังสือไว้ 3 วัน เพื่อรอผู้จองมารับหนังสือ และหากมีการแจ้งหนังสือหายก็ต้องบันทึกรายการหนังสือหาย เพื่อนำไปสร้างรายงานต่อไป ดังรูปที่ 3-4



รูปที่ 3-4 การคืนหนังสือ

3.1.3 การจองหนังสือ

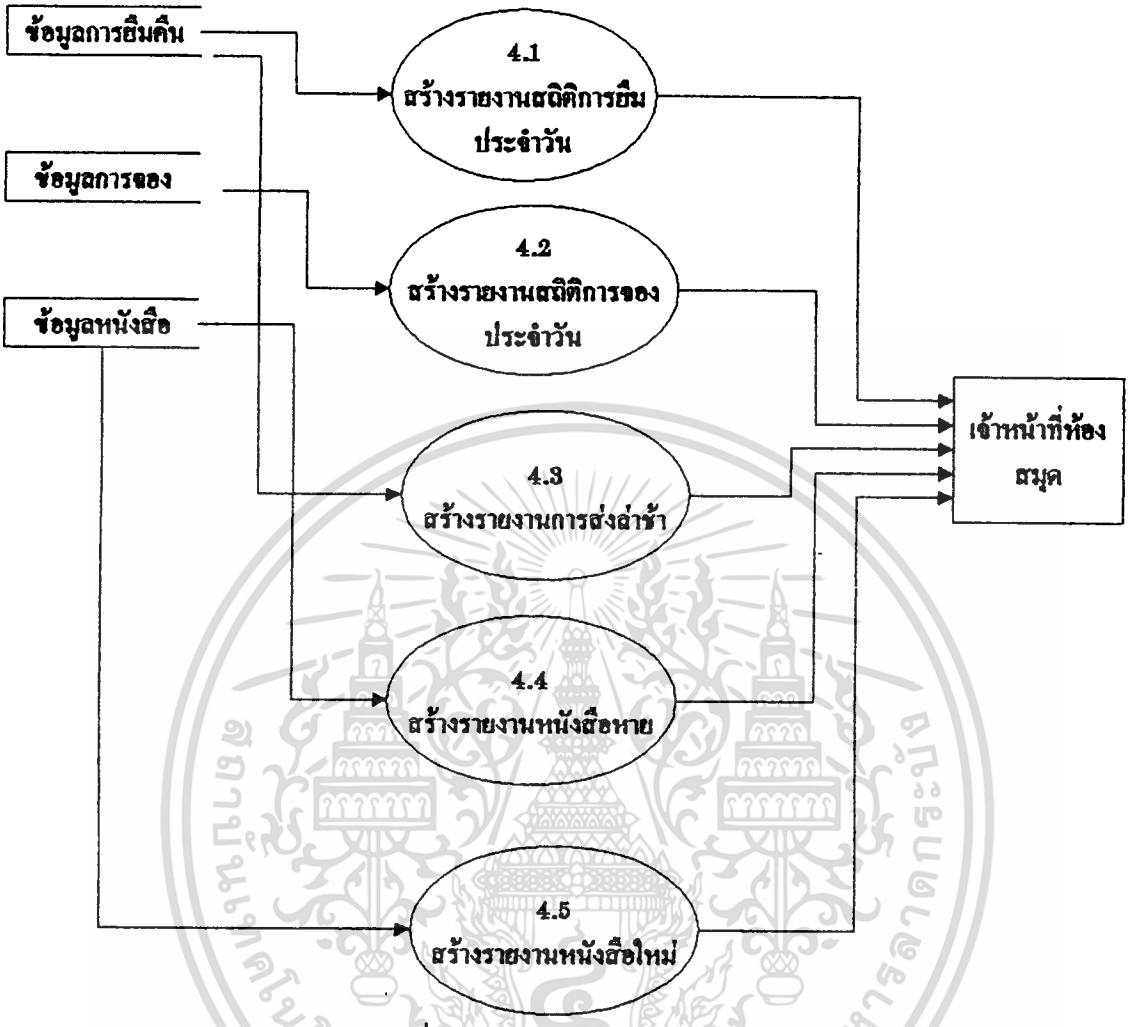
ในงานจองต้องมีการตรวจด้วยว่า มีการจองเกินระเบียบห้องสมุดหรือไม่ และตรวจด้วยว่า จองเลขหมู่ซ้ำกัน 2 ครั้ง หรือไม่ อีกทั้งต้องทำการปรับการจอง หากใน 3 วัน ไม่มีผู้มารับหนังสือก็นำหนังสือกลับสู่ห้องสมุดตามปกติ หรืออาจเก็บไว้ให้คนจองคนถัดไป หากหนังสือเล่มนี้มีการจอง 2 คนขึ้นไป ดังนั้นในการจองจะต้องเก็บข้อมูล ทั้งวันจอง และเวลาจองไว้ด้วย โดยผู้ที่จองก่อนจะต้องได้รับหนังสือก่อน ดังแสดงในรูป 3-5



รูปที่ 3-5 การจอง

3.1.4 การสร้างรายงาน

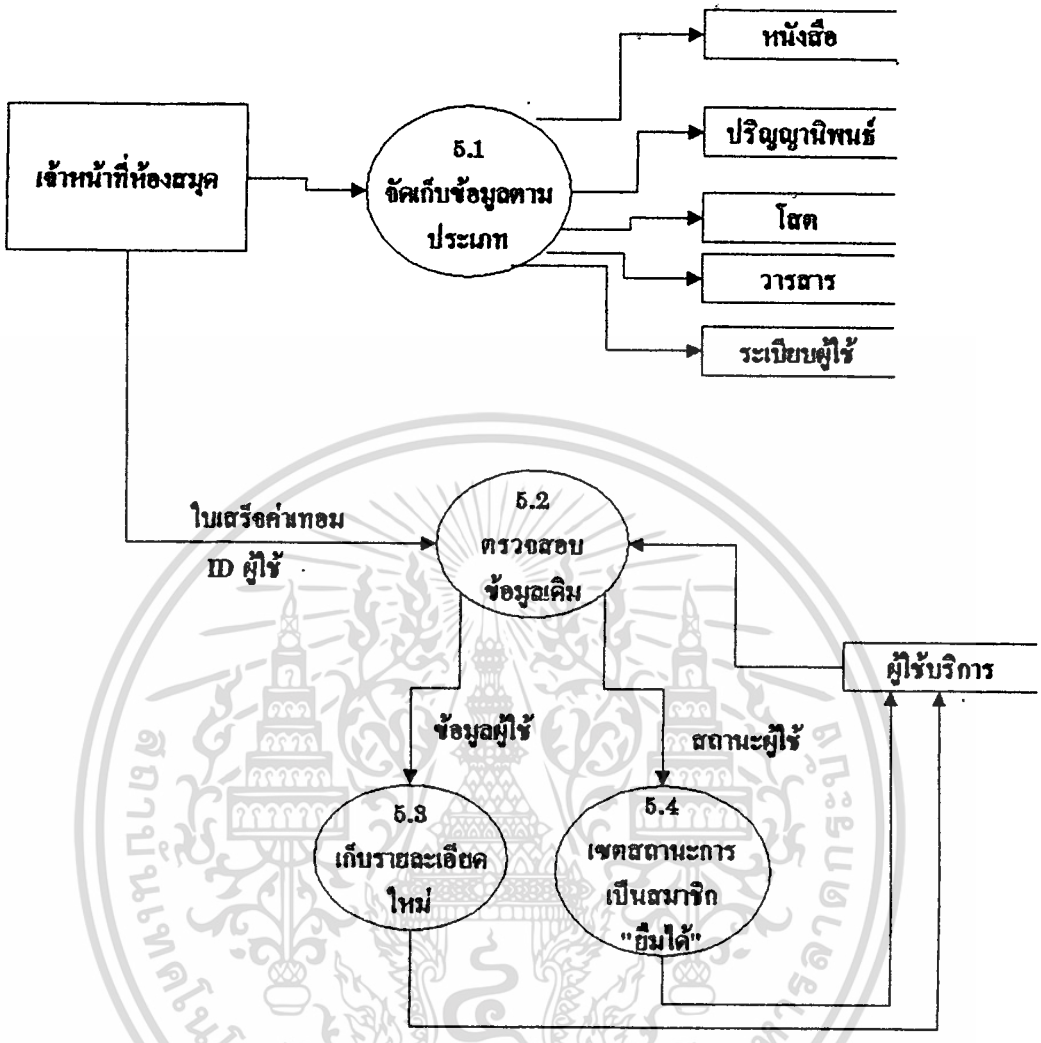
จะต้องมีการสร้างรายงานสถิติการยืมประจำวัน สถิติการจอง รายงานหนังสือหาย รายงานหนังสือใหม่ และรายงานหนังสือที่ค้างส่ง โดยหากผู้ค้างส่งเป็นอาจารย์หรือเจ้าหน้าที่ ก็จะต้องมีใบทวงถามหนังสือของแต่ละท่าน ดังแสดงในรูป 3-6



รูปที่ 3-6 การสร้างรายงาน

3.1.5 การจัดเก็บข้อมูล

ในการเก็บข้อมูล จะต้องมีการเก็บข้อมูลหลายๆประเภท คือ ข้อมูลหนังสือใหม่ ข้อมูลปริญญานิพนธ์ โสตทัศนศึกษา วารสาร กฎระเบียบผู้ใช้ และเก็บข้อมูลผู้ใช้หม่ และเก็บข้อมูลผู้ใช้ที่ต่ออายุสมาชิก ดังนั้นจึงต้องมีงานที่จัดการเปลี่ยนสถานะให้สมาชิกหมดอายุเมื่อหมดเทอมการศึกษา ดังแสดงในรูปที่ 3-7



รูปที่ 3-7 การจัดเก็บข้อมูล

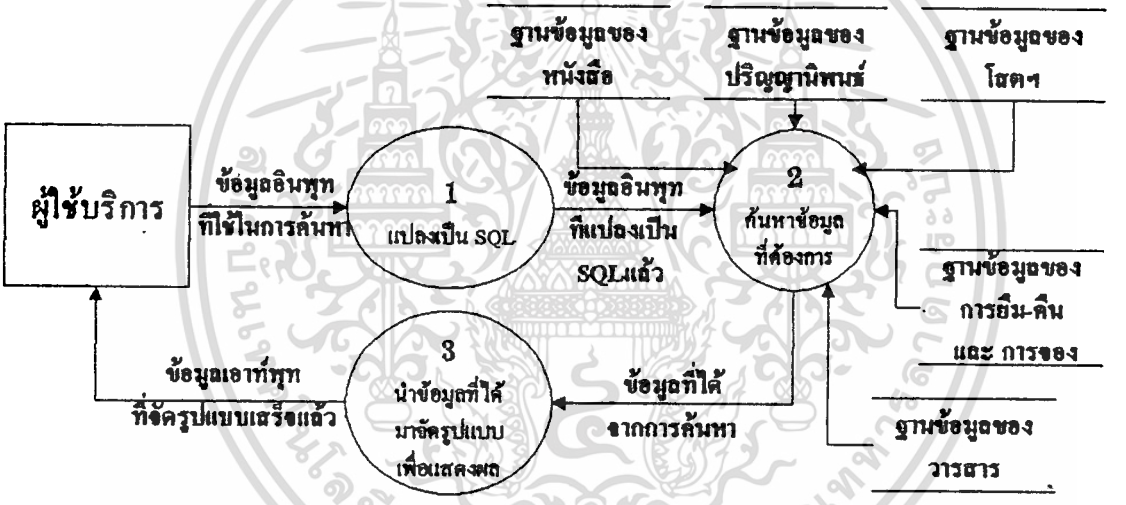
3.2 งานค้นหา

จากระบบงานดั้งเดิมที่ใช้กันอยู่ในปัจจุบัน จะมีการค้นหาหนังสือและปริญญานิพนธ์เท่านั้น ซึ่งในส่วน การค้นหาหนังสือและปริญญานิพนธ์เราจะใช้ระบบงานเดิมมาปรับปรุงให้มีประสิทธิภาพมากขึ้น เพื่อประโยชน์คือ ช่วยให้ผู้ใช้เรียนรู้และใช้ระบบได้สะดวกยิ่งขึ้น เพราะผู้ใช้มีความคุ้นเคยกับระบบงานเดิมอยู่แล้ว

ต่อมาเนื่องจากทางห้องสมุดต้องการเพิ่มบริการในส่วนการค้นหาเพิ่มขึ้นจากบริการเดิมที่มีอยู่นั้นคือ ทางห้องสมุดต้องการให้มีบริการการค้นหา โสตทัศนูปกรณ์ และวารสารเพิ่มเพราะทางห้องสมุดได้มีการเก็บข้อมูลของ โสตทัศนูปกรณ์ (ในที่นี้หมายถึง วีดีโอ เท่านั้น) และวารสารไว้แล้ว ทำให้ต้องมีการสำรวจหาความต้องการทั้งของผู้ใช้บริการทั่วไปและเจ้าหน้าที่ และจากการสำรวจทำให้ทราบความต้องการต่างๆ เพิ่มขึ้นและได้เพิ่มการค้นหาข้อมูลของผู้ใช้ ซึ่งจะกล่าวถึงรายละเอียดของการค้นหาแต่ละส่วนออกเป็นข้อๆ โดยสามารถแบ่งได้ 5 หัวข้อดังนี้

1. การค้นหาหนังสือ
2. การค้นคว้าวิทยานิพนธ์
3. การค้นหาวารสาร
4. การค้นหาโสตทัศนูปกรณ์ (วีดีโอ เท่านั้น)
5. การค้นหาข้อมูลของผู้ใช้ แบ่งเป็น
   5.1 ข้อมูลเกี่ยวกับการยืม-คืน หนังสือ
   5.2 ข้อมูลเกี่ยวกับการจอง หนังสือ

ดาต้าโฟลว์ ข้างล่างนี้จะแสดงระบบการค้นหาทั้งหมดของห้องสมุด



รูปที่ 3-8 ระบบการค้นหาในห้องสมุด

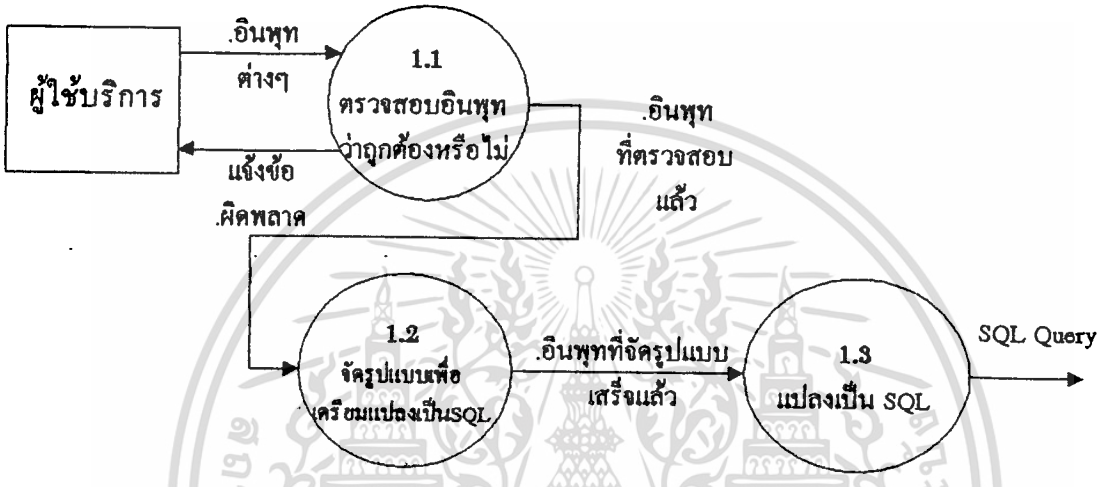
จากรูปสามารถแบ่งออกเป็นส่วนย่อยๆ 3 ส่วน คือ

1. ส่วนรับข้อมูล และ สร้าง เอสคิวแอล คิวรี่
2. ส่วนค้นหาข้อมูล
3. ส่วนแสดงผล

3.2.1 ส่วนรับข้อมูล และสร้างเอสคิวแอลคิวรี่

ในส่วนนี้จะรับข้อมูลที่ผู้ใช้ต้องการค้นหาเข้ามา เพื่อสร้าง เอสคิวแอลคิวรี่ ให้เหมาะสม และนำไปใช้ในส่วนที่ 2 ต่อไป

สามารถแสดงเป็น ดาต้าโฟล์ว ได้ดังนี้



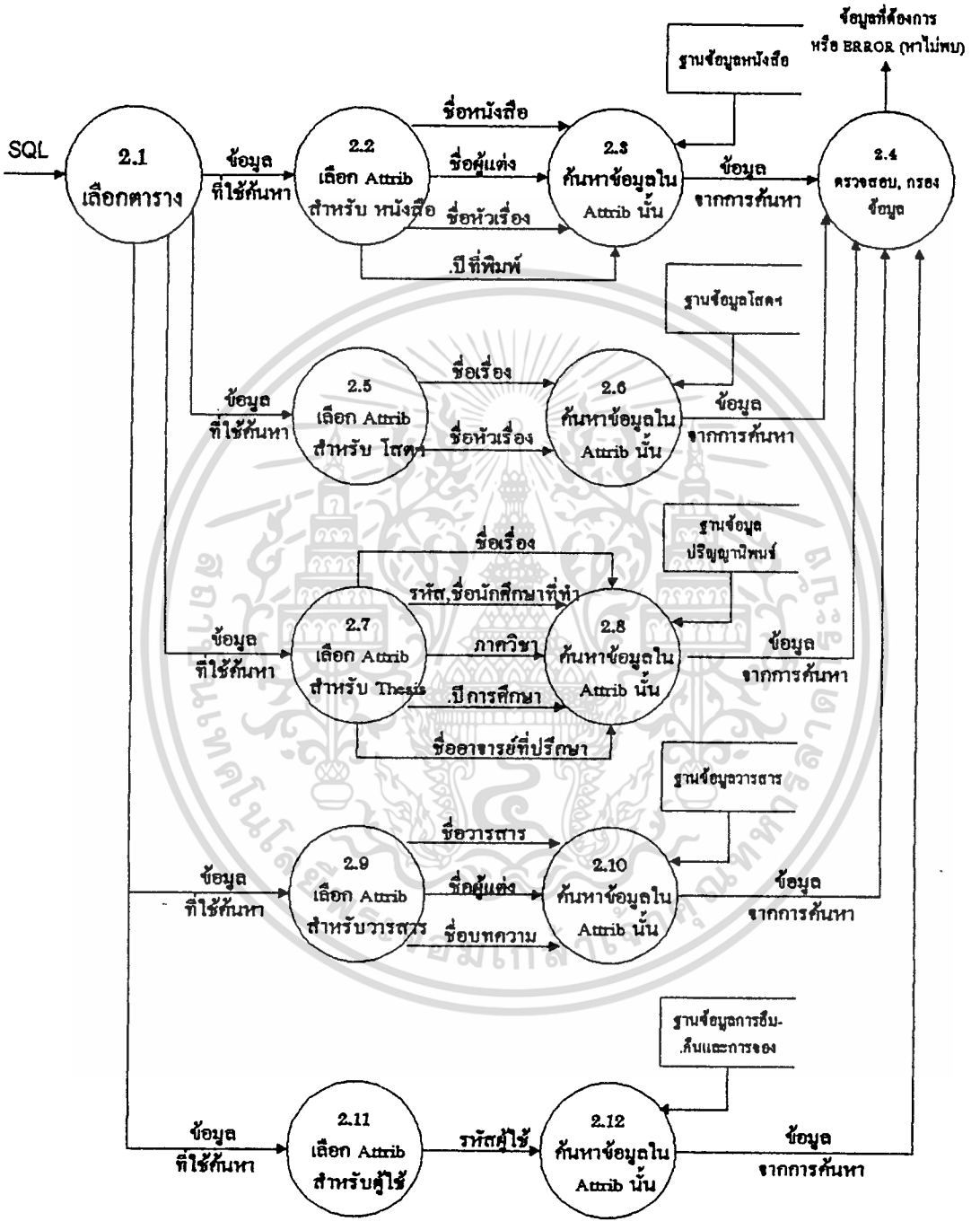
รูปที่ 3-9 ส่วนรับข้อมูล และ สร้างเอสคิวแอลคิวรี่ของระบบการค้นหา

3.2.2 ส่วนค้นหาข้อมูล

จากที่กล่าวมาข้างต้นว่า การค้นหาของระบบห้องสมุดจะแบ่งออกเป็น 5 หัวข้อหลักๆ คือ

1. การค้นหาหนังสือ
2. การค้นหาวิทยานิพนธ์
3. การค้นหาวารสาร
4. การค้นหาโสตทัศนูปกรณ์ (วีดีโอ เท่านั้น)
5. การค้นหาข้อมูลของผู้ใช้ แบ่งเป็น
   - 5.1 ข้อมูลเกี่ยวกับการยืม-คืน หนังสือ
   - 5.2 ข้อมูลเกี่ยวกับการจอง หนังสือ

ในการค้นหาจะใช้ เอสคิวแอลคิวรี่ จากส่วนที่ 1 มาค้นหา สามารถแสดงเป็นดาต้าโฟล์ว ได้ดังนี้

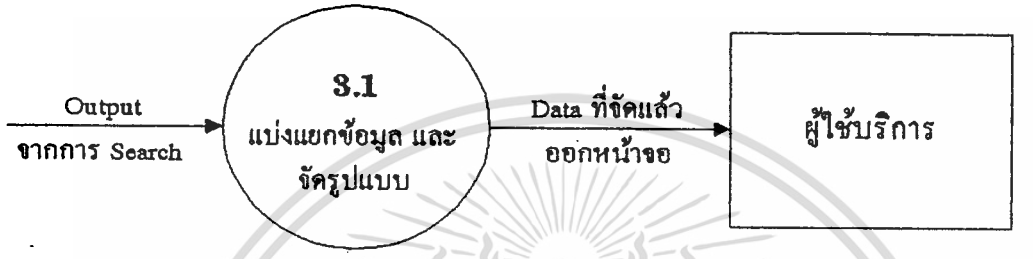


รูปที่ 3-10 ส่วนค้นหาข้อมูลของระบบการค้นหา

3.2.3 ส่วนแสดงผล

ในส่วนนี้จะรับข้อมูลที่ค้นหาได้ทั้งหมดจากส่วนที่ 2 มาจัดรูปแบบ เพื่อแสดงผลออกทางจอภาพตามที่ผู้ใช้ต้องการ

สามารถเขียน คาต้าโฟล์ว ได้ดังนี้



รูปที่ 3-11 ส่วนแสดงผลของระบบการค้นหา

3.3 สรุป

ในบทนี้ได้กล่าวถึงระบบงานของห้องสมุดว่ามีขั้นตอนการทำงานอะไรบ้าง ซึ่งจะเห็นว่างานห้องสมุดนั้นมี 2 งานหลักคือ

1. งานบริการ ซึ่งประกอบด้วยงานย่อยอีก 5 งาน ได้แก่ การยืมหนังสือ การคืนหนังสือ การจองหนังสือ การสร้างรายงาน และการจัดเก็บข้อมูล

2. งานค้นหา ก็ประกอบด้วยงานย่อยอีก 5 งานเช่นกันได้แก่ การค้นหาหนังสือ การค้นหาวิทยานิพนธ์ การค้นหาวารสาร การค้นหาโสตทัศนูปกรณ์ (วีดีโอ เท่านั้น) และการค้นหาข้อมูลของผู้ใช้ โดยการค้นหาข้อมูลของผู้ใช้ยังแบ่งเป็น การค้นหาข้อมูลเกี่ยวกับการยืม-คืน ว่าผู้ใช้ได้ยืมหนังสืออะไรไปบ้าง และ การค้นหาข้อมูลเกี่ยวกับการจองหนังสือ ว่าตอนนี้หนังสือที่ตัวเองจองพร้อมที่จะรับได้แล้วหรือยัง

บทที่ 4
การพัฒนาระบบ

หลังจากในบทก่อน ได้เริ่มศึกษาระบบงานของห้องสมุดว่ามีการดำเนินงานอย่างไรบ้างจนครบถ้วนสมบูรณ์แล้ว ในการพัฒนาระบบงานต่อไปก็เป็นขั้นของการออกแบบและเขียนโปรแกรม ซึ่งมีลำดับ ขั้นตอนดังต่อไปนี้

1. เริ่มออกแบบในส่วนของฐานข้อมูลคือ เริ่มเขียนไนแอมไดอะแกรม (Niam Diagram) โดยนำเอาเอ็นติตี้ (entity) ต่างๆที่เกี่ยวข้องมาสร้างความสัมพันธ์เป็นไนแอมไดอะแกรม
2. เมื่อเขียนไนแอมไดอะแกรมเสร็จ ก็ทำการแปลงไนแอมไดอะแกรมให้เป็น ตาราง (Table) ของฐานข้อมูล โดยส่วนตารางต่างๆ จะระบุถึงชนิดของข้อมูล (Data type) ของแต่ละเอ็นติตี้
3. จากนั้นจึงได้กลับมาเขียนดาต้าโฟลว์ไดอะแกรม ให้ลงลึกถึงระดับตาราง
4. เริ่มออกแบบหน้าจอที่จะติดต่อกับผู้ใช้และรายงานต่างๆที่ต้องพิมพ์ให้ผู้ใช้
5. ออกแบบเอสคิวแอล (SQL) ที่จะนำมาใช้ โดยจะออกแบบตามดาต้าโฟว์ และ หน้าจอ
6. ออกแบบโปรแกรมโดยเขียนในรูปเสตทไดอะแกรม และ ซูโดโค้ด (psudo code)
7. เริ่มเขียนโปรแกรมตามที่ได้ออกแบบมา
8. นำโปรแกรมที่พอที่จะทำงานได้ไปให้ผู้ใช้ตรวจสอบ
9. แก้ไขข้อผิดพลาดหรือความต้องการให้ตรงกับผู้ใช้
10. ตรวจสอบโปรแกรมให้มีข้อผิดพลาดในการทำงานน้อยที่สุด
11. เขียนคู่มือการใช้โปรแกรม

4.1 การออกแบบฐานข้อมูลโดยใช้วิธีไนแอม

การออกแบบฐานข้อมูลโดยใช้วิธีไนแอม (NIAM : Nijssen's Information Analysis Methodology) เป็นวิธีการในการออกแบบฐานข้อมูล โดยการแสดงความสัมพันธ์และข้อจำกัดต่างๆ ของข้อมูลด้วยแบบจำลองข้อมูลที่ประกอบไปด้วยสัญญลักษณ์ต่างๆ และเป็นวิธีที่มีอัลกอริธึมที่สามารถออกแบบฐานข้อมูลแบบรีเลชันแนลที่อยู่ในรูปนอมัลฟอร์มที่ 5 (Fifth Normal Form) ได้โดยตรง กล่าวคือข้อมูลจะต้องไม่มีการเก็บซ้ำซ้อน ซึ่งอาจจะทำให้เกิดความขัดแย้งของข้อมูลอันเนื่องมาจากการเปลี่ยนแปลงข้อมูลเฉพาะที่

ไนแอมมีขั้นตอนในการออกแบบอยู่ 9 ขั้นตอน (CSDP 9 steps : Conceptual Schema Design Procedure)

ขั้น 1 : กำหนดขอบเขตของงาน (Universe of Discourse : UOD) และความจริงที่เกิดขึ้นภายในขอบเขตของงานที่กำหนดไว้

ขั้น 2 : วาด คอนเส็ปชวลสคีมาไดอะแกรม (Conceptual Schema Diagram) โดยคร่าวๆจากความจริงในขอบเขตของงาน

ขั้น 3 : จัดรูปสคีมาให้เป็นระเบียบและหาชนิดความจริงที่ได้รับข้อมูลมาจากชนิดความจริงอื่น

ขั้น 4 : เติมสัญญลักษณ์แสดง ยูนิคคอนสแทรน (Uniqueness constraints)

ขั้น 5 : ตรวจสอบความถูกต้องของชนิดความจริง

ขั้น 6 : เติมสัญลักษณ์แสดง เล็กซิคอล(Lexical),แมนดาทอรีโรล(Mandatory Role),ซับไทค์คอนสแทรน(Subtype constraints)

ขั้น 7 : ตรวจสอบยูนิคไอเด็นติไฟล์เออร์ (Unique Identifier) ของแต่ละชนิดเอนติตี

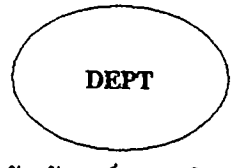
ขั้น 8 : เติมสัญญลักษณ์แสดง อิคอลลิตี้(Equality), เอ็กคลูชชั่น(Exclusion), ซับเซ็ตคอนสแทรน (Subset constraints)

ขั้น 9 : ตรวจสอบความสมบูรณ์ของคอนเส็ปชวลสคีมา ว่าต้องสอดคล้องกับตัวอย่างข้อมูลไม่มีความซ้ำซ้อนของข้อมูล

4.1.1 ส่วนประกอบพื้นฐานของไนแอม

1. ชนิดเอนติตี้ (Entity Type)
2. ชนิดเลเบิล (Lable Type)
3. ชนิดความจริง (Fact Type)
4. ชนิดอ้างอิง (Reference Type)
5. ข้อจำกัดเพื่อความถูกต้องของข้อมูล (Integrity Contraints)

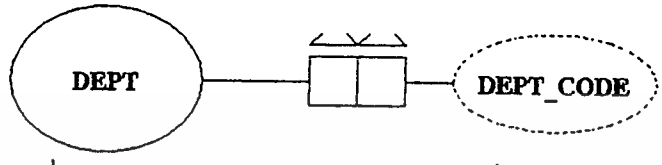
สัญญลักษณ์และตัวอย่างการใช้ส่วนประกอบพื้นฐานของแบบจำลองไนแอมแสดงไว้ดังรูป



รูปที่ 4-1 แสดงสัญลักษณ์ของชนิดเอนติตีภาควิชา



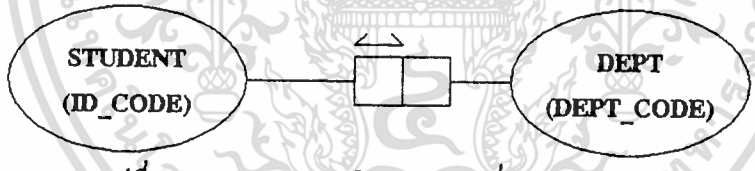
รูปที่ 4-2 แสดงสัญลักษณ์ของชนิดเลเบิลรหัสภาควิชา



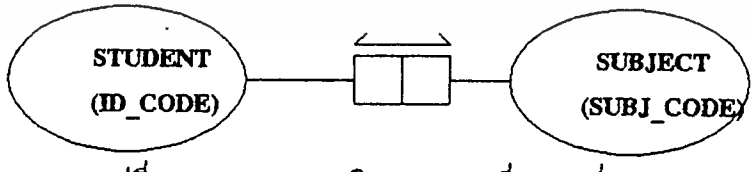
รูปที่ 4-3 แสดงความสัมพันธ์อ้างอิงแบบหนึ่งต่อหนึ่ง (one to one) หมายความว่า ภาควิชาใดๆ จะมีรหัสภาควิชาเพียงรหัสเดียวเท่านั้น และไม่ซ้ำกับภาควิชาอื่น



รูปที่ 4-4 แสดงความสัมพันธ์อ้างอิงแบบหนึ่งต่อหนึ่งอย่างย่อ



รูปที่ 4-5 แสดงความจริงแบบแมนนี่ทูวัน(many to one) หมายความว่านักศึกษาหนึ่งคนจะสังกัดภาควิชาได้เพียงภาควิชาเดียว แต่ภาควิชาใดๆ สามารถมีนักศึกษาในสังกัดได้มากกว่าหนึ่งคน



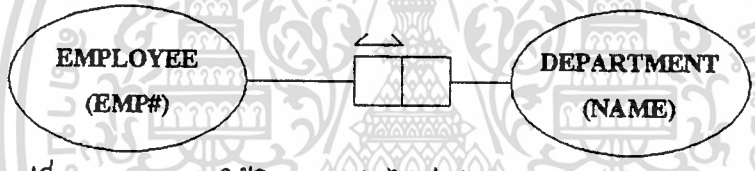
รูปที่ 4-6 แสดงความจริงแบบ แมนนี่ทูแมนนี่(many to many) หมายความว่านักศึกษาหนึ่งคน สามารถลงทะเบียนได้หลายวิชาและแต่ละวิชาที่เปิดสอนสามารถรับจำนวนนักศึกษาได้มากกว่าหนึ่งคน แต่นักศึกษาที่ลงทะเบียนเรียนวิชาใดๆแล้ว จะลงทะเบียนซ้ำวิชาเดิมไม่ได้

ชนิดเอนติตีเป็นเซ็ต (set) ซึ่งมีสมาชิกเป็นตัวอย่างเอนติตี (Entity Instance) เช่น ภาควิชา A (ภาควิชาโทรคมนาคม) ภาควิชา B (ภาควิชาวิศวกรรมคอมพิวเตอร์) เป็นตัวอย่างเอนติตีของชนิดเอนติตีภาควิชา

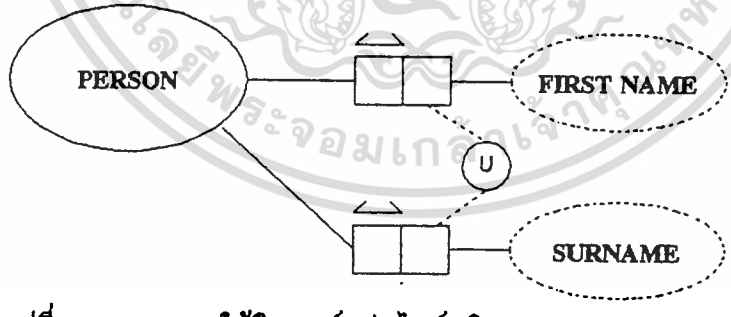
เครื่องหมายความสัมพันธ์ที่เป็นส่วนเชื่อมโยงระหว่างชนิดเอนติตี และชนิดเอนติตีหรือชนิดเลเบิลนั้น เรียกว่า บทบาท (role) จะเขียนความหมายของบทบาทนั้นไว้ภายในหรือข้างๆ สัญญลักษณ์ของมัน

การแปลงข้อมูลที่วิเคราะห์มาให้อยู่ในรูปแบบจำลอง ก่อนอื่นต้องนำข้อมูลมากำหนดเป็นชนิดเอนติตีและเลเบิลให้เรียบร้อยเสียก่อน จึงนำชนิดเอนติตีที่ได้มาเขียนเป็นประโยคความจริงมูลฐาน (Elementary Fact) แล้วเอาความจริงทั้งหมดที่ได้มาเขียนเป็นแบบจำลอง และเติมข้อจำกัดต่างๆลงไปตามความเป็นจริงในขอบเขตของงาน

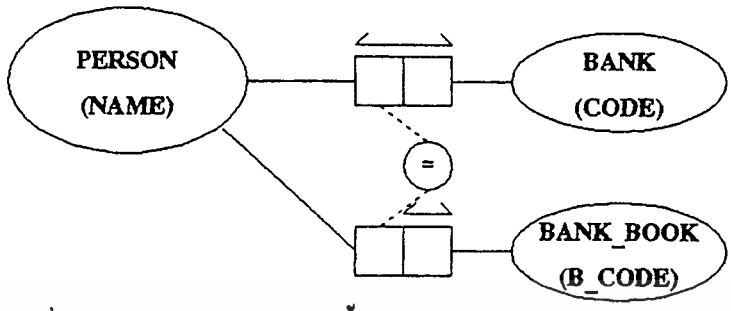
ตัวอย่างการใช้ข้อจำกัดต่างๆแสดงได้ดังรูปต่อไปนี้



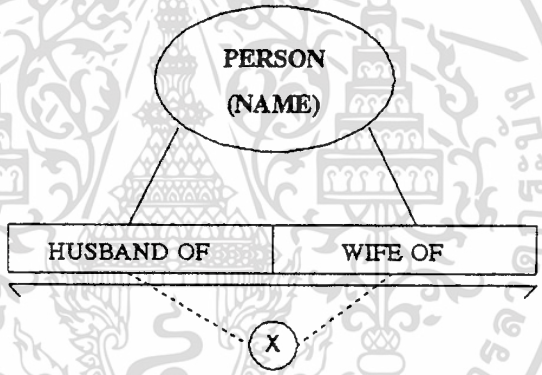
รูปที่ 4-7 แสดงการใช้อินทราแฟคไทด์ยูนิคคอนสแทรน(intra fact type uniqueness constraint) หมายความว่าลูกจ้างหนึ่งคนจะมีที่ทำงานได้ที่เดียวเท่านั้น



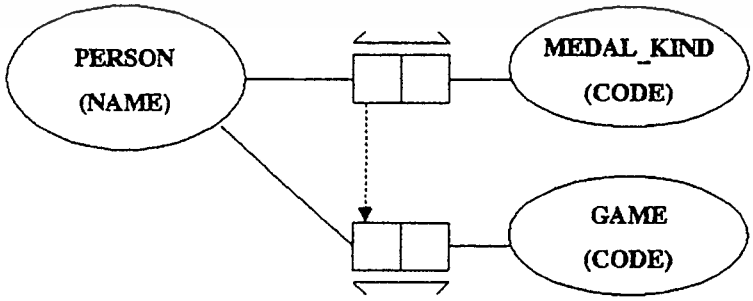
รูปที่ 4-8 แสดงการใช้อินเตอร์แฟคไทด์ยูนิคคอนสแทรน(inter fact type uniqueness constraint) หมายความว่า บุคคลหนึ่งจะมีชื่อ 1 ชื่อ นามสกุล 1 นามสกุล ชื่อของบางคนอาจจะซ้ำกัน และนามสกุลของบางคนอาจจะซ้ำกัน แต่ชื่อรวมกับนามสกุลจะต้องไม่ซ้ำกัน



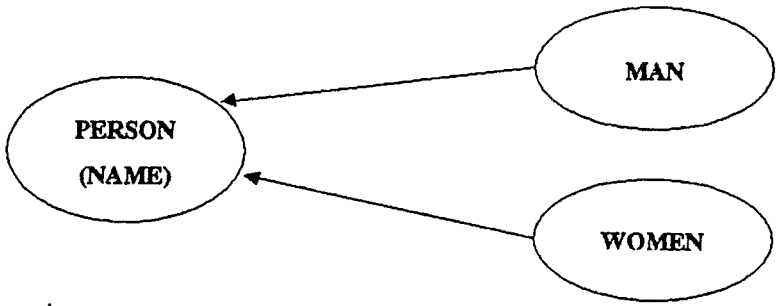
รูปที่ 4-9 แสดงการใช้อีควอลิตี้ คอนสแทรน(equality constrain)
หมายความว่า ถ้าบุคคลหนึ่งเป็นลูกค้าของธนาคารใดแล้ว บุคคลนั้นต้องมีสมุดบัญชีของธนาคารนั้นด้วย ในทางกลับกัน ถ้าบุคคลใดมีสมุดบัญชีของธนาคารใดแล้ว ก็ต้องเป็นลูกค้าของธนาคารนั้นด้วย



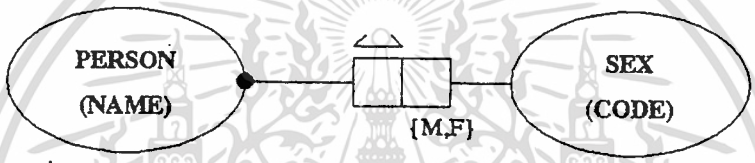
รูปที่ 4-10 แสดงการใช้เอ็กซ์คลูสชั่น คอนสแทรน (exclusion constraint)
หมายความว่า บุคคลใดเป็นภรรยาของอีกบุคคลหนึ่งแล้ว บุคคลนั้นต้องไม่เป็นสามีของบุคคลใดๆ ในทางกลับกัน บุคคลที่เป็นสามีของบุคคลหนึ่งแล้ว จะต้องไม่เป็นภรรยาของบุคคลใดด้วย



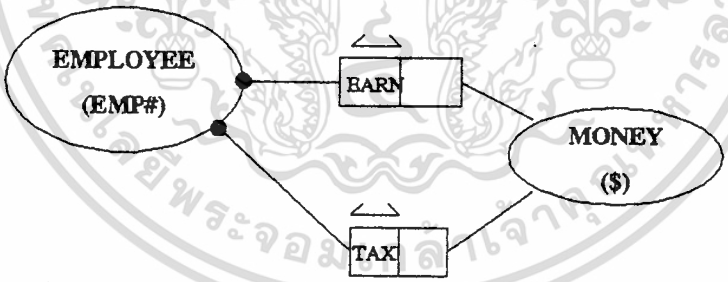
รูปที่ 4-11 แสดงการใช้ ซับเซต คอนสแทรน (subset constraint)
หมายความว่า บุคคลที่ชนะการแข่งขันกีฬาทุกคนจะต้องเป็นบุคคลที่เล่นกีฬา แต่บุคคลที่เล่นกีฬาไม่จำเป็นต้องชนะการแข่งขันกีฬาทุกคน



รูปที่ 4-12 แสดงการใช้ ซับไทค์ คอนสแทรน(subtype constraint)
หมายความว่า ตัวอย่างเอนติตีทุกตัวของชนิดเอนติตีผู้ชาย และชนิดเอนติตีผู้หญิง ต่างก็เป็นสมาชิกของ ชนิดเอนติตีบุคคล

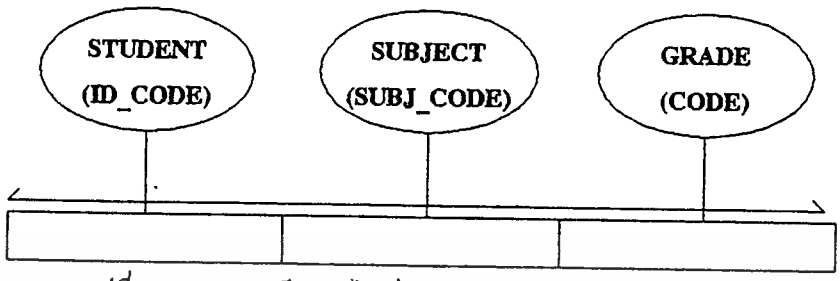


รูปที่ 4-13 แสดงการใช้แมนดาทอรี คอนสแทรน(mandatory constraint), เลคซิคอล คอนสแทรน (lexical constraint) หมายความว่าบุคคลทุกคนต้องมีเพศ และสมาชิกของชนิดเอนติตีเพศมีเพียง M (Male) และ F (Female) เท่านั้น



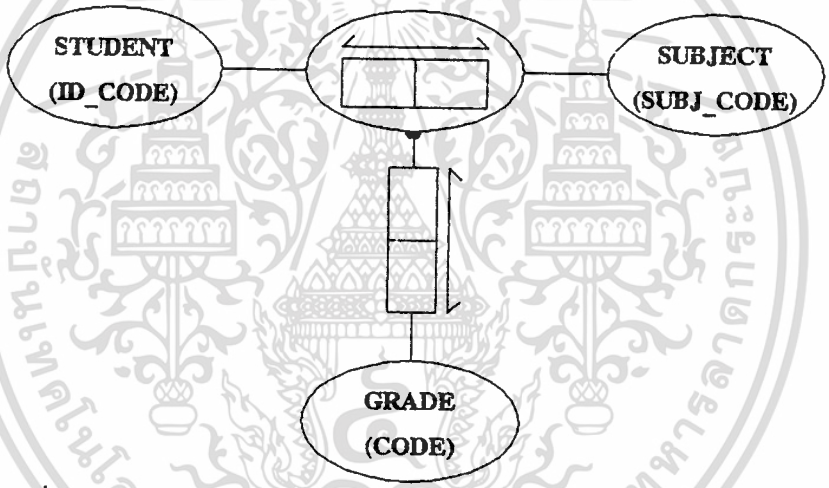
รูปที่ 4-14 แสดงความสัมพันธ์ระหว่างลูกจ้างกับเงิน
โดยหมายความว่าลูกจ้างทุกคนต้องมีรายได้ และทุกคนต้องเสียภาษี

นอกจากตัวอย่างจริงแบบไบนารีแฟคไทค์ (Binary Fact Type) ที่แสดงไว้ข้างต้นแล้วยังมีตัวอย่างชนิดอื่นอีกเช่น



รูปที่ 4-15 เทอนารีแฟคไทด์ (Ternary Fact Type)

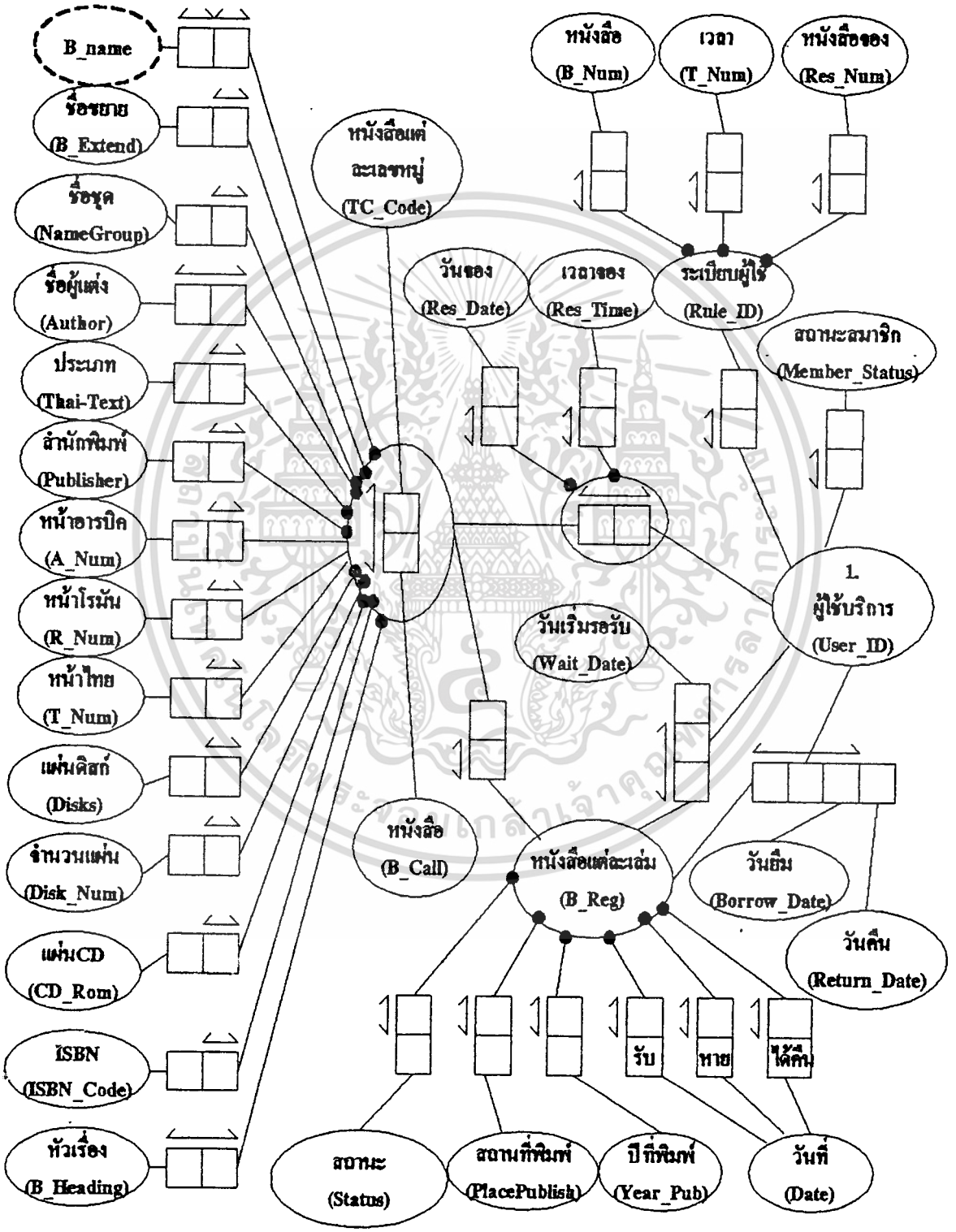
หมายความว่า ข้อมูลการเรียนของนักศึกษาทุกคนจะต้องมีทั้งรหัสวิชาและเกรด



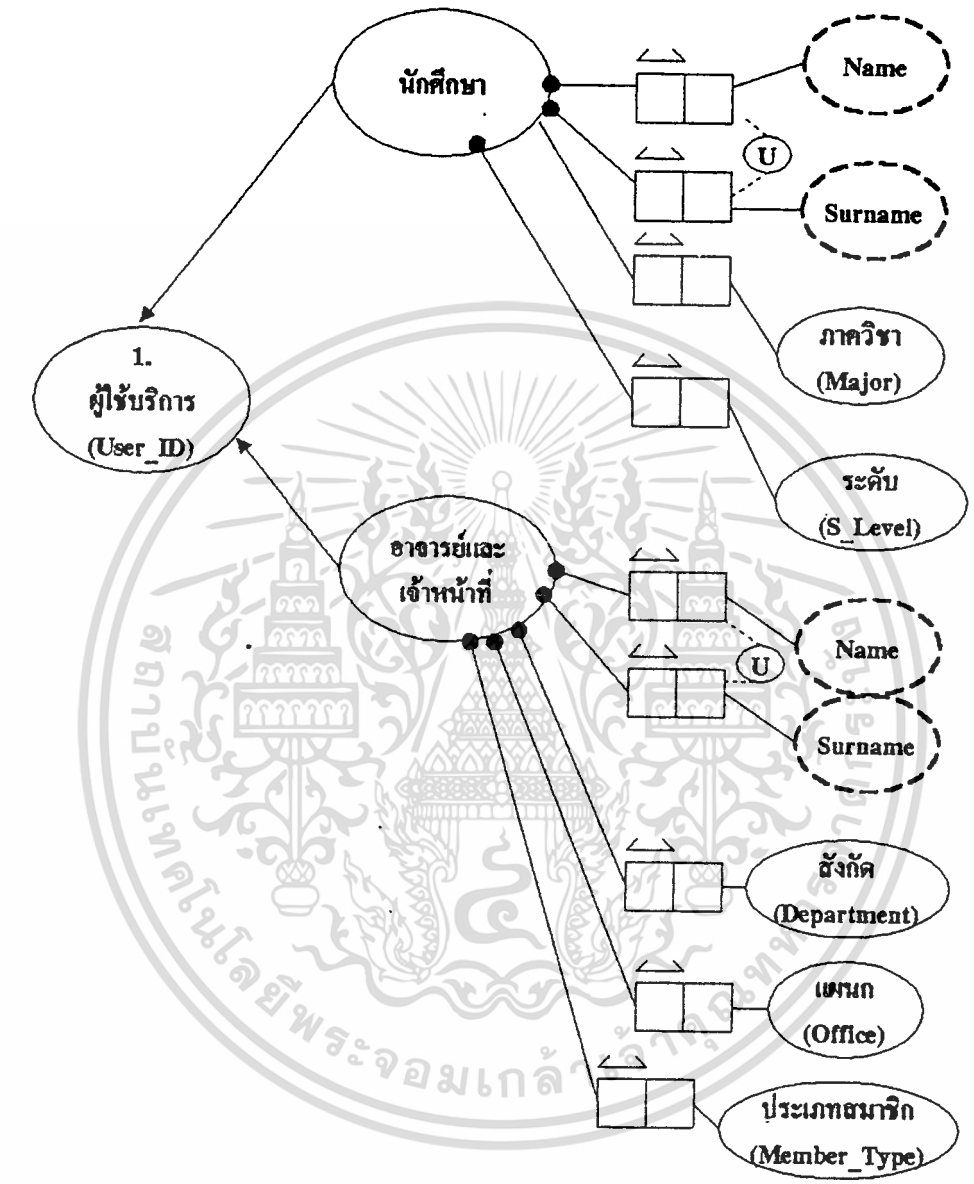
รูปที่ 4-16 เนสแฟคไทด์ (Nested Fact Type) มีความหมายเหมือนรูปที่แล้ว

4.12 ไนแอมไดอะแกรมของระบบห้องสมุด

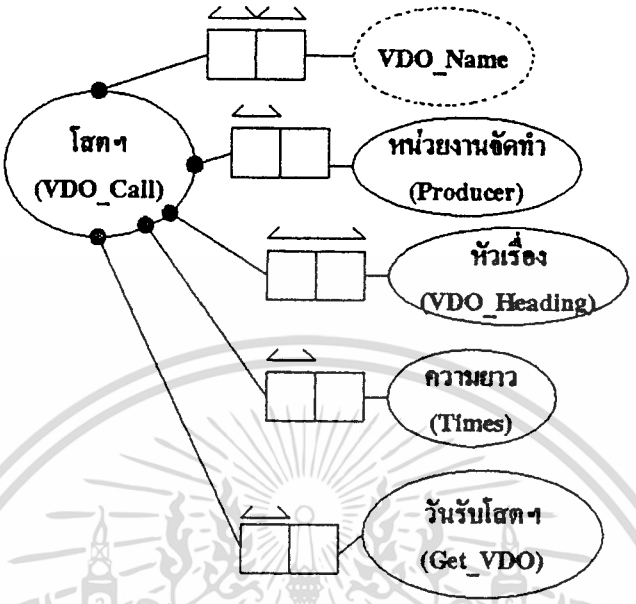
ต่อจากนี้ไปจะเป็นไนแอมของระบบห้องสมุด



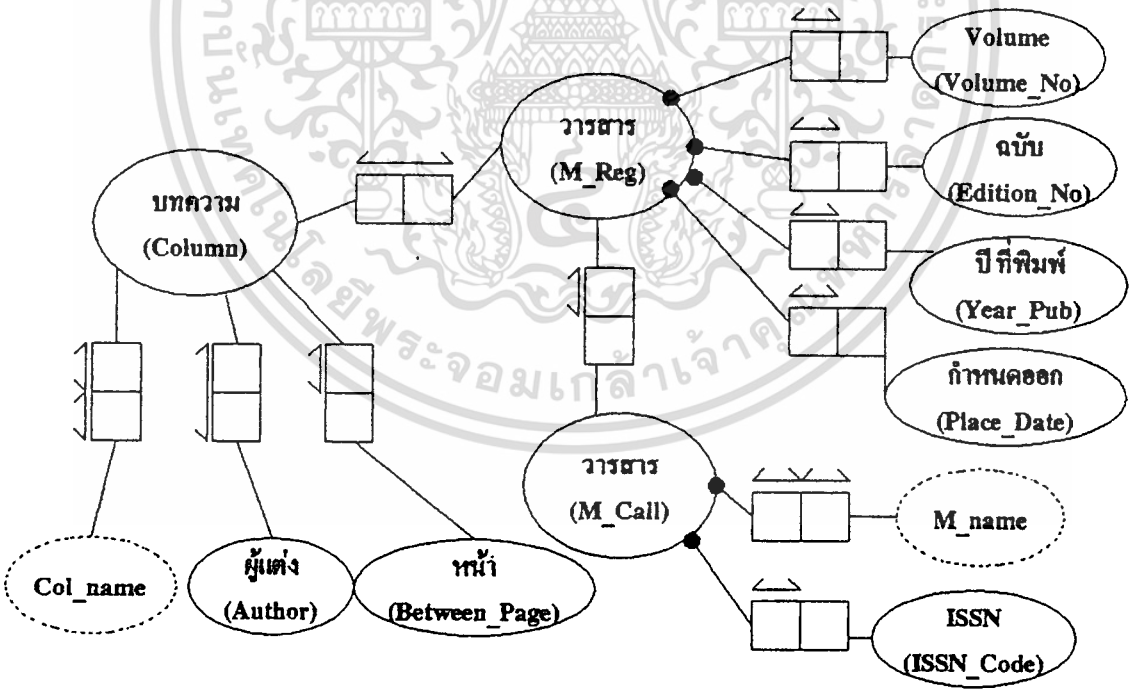
รูปที่ 4-17 ไนแอมไดอะแกรมในส่วนหนังสือ,การยืม,การจอง



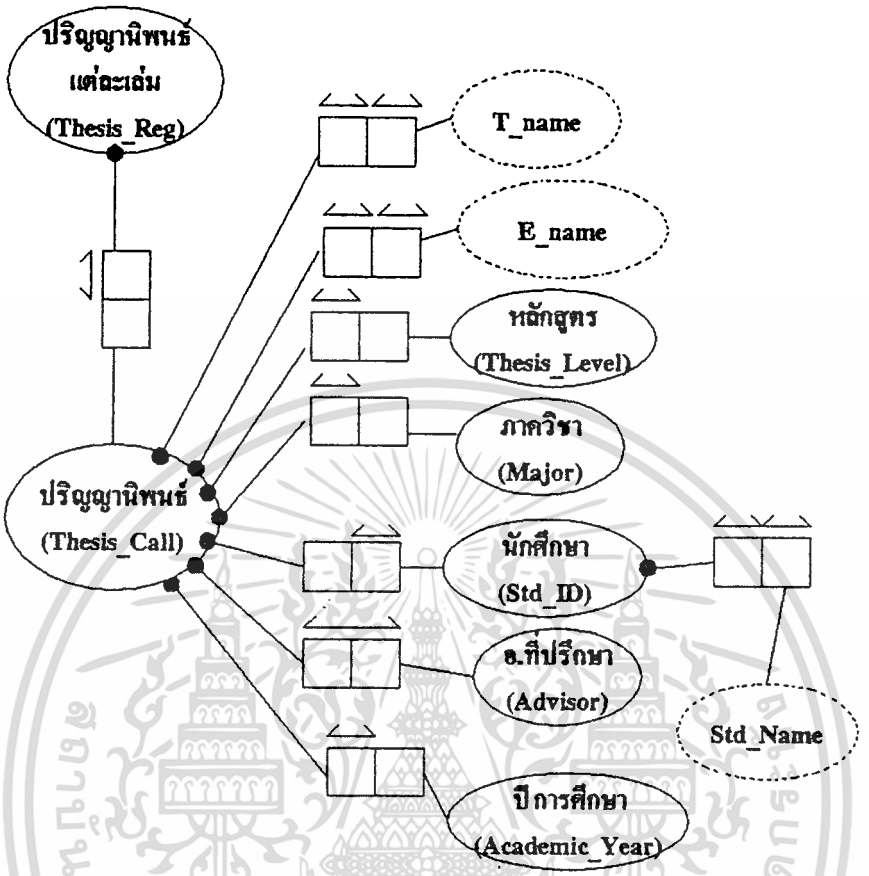
รูปที่ 4-18 ไนแอมไดอะแกรมในส่วนผู้ใช้บริการ



รูปที่ 4-19 ไนแอมไดอะแกรมในส่วนโสตทัศนศึกษา



รูปที่ 4-20 ไนแอมไดอะแกรมในส่วนวารสาร



รูปที่ 4-21 ไนแอมไดอะแกรมในส่วนปริญญานิพนธ์

ต่อจากนี้จะเป็นตารางต่างๆที่แปลงมาจากไนแอมไดอะแกรม และดาต้าดิกชั่นนารีของตาราง (data dictionary for table) ซึ่งบอกถึงชนิดของข้อมูลในตารางและตัวอย่างข้อมูลที่นำมาใช้

4.2 ตารางของฐานข้อมูล

ในการจัดกลุ่มความจริงในคอนเส็ปชวลสคีมา(Conceptual Schema) ให้เป็น รีเลชั่นนอลดาต้าเบสสคีมา(Relational Database Schema) หรือการแปลงไนแอมเป็นตารางฐานข้อมูล มีหลักการในการปฏิบัติดังต่อไปนี้

1. สร้าง 1 รีเลชันหรือ 1 ตาราง สำหรับชนิดความจริงแบบไบนารีซึ่งมีความสัมพันธ์แบบ เมนี่ทูเมนนี่ (many to many) โดยที่ยูนิคไอเด็นทิไฟเออร์ (Unique Identifier) ของชนิดเอนติตีที่เกี่ยวข้องทั้งหมดเป็นไพรมารี่คีย์ (Primary Key)

2. สร้าง 1 รีเลชั่นสำหรับแต่ละชนิดความจริงแบบเอ็นอาเรย์(n-ary) โดยที่ยูนิคไอเด็นทิไฟเออร์ของชนิดเอนติตีซึ่งมีโรล(Role) ถูกบังคับด้วยยูนิคคอนสแทรนเดียวกันเป็นเคนดิเดทคีย์(Candidate key)

3. พิจารณาชนิดเอนติตีที่เกี่ยวข้องกับความจริงแบบไบนารีที่มีความสัมพันธ์เป็นแบบวันทูวัน หรือ แมนนี่ทูวัน โดยที่โรลของชนิดเอนติตีเหล่านั้นถูกบังคับด้วย ยูนิค คอนสเทรน ให้สร้างรีเลชั่นโดยมียูนิค ไอเด็นทไฟเออร์ ของชนิดเอนติตีเหล่านั้นเป็นแคนดิเดทคีย์

BOOK

B_CALL	TC_CODE	B_NAME	B_EXTEND	NAMEGROUP	THAI_TEXT	PUBLISHER
Var Char 40	Char 1	Var Char 100	Var Char 100	Var Char 100	Char 1	Var Char 50

A_NUM	R_NUM	T_NUM	DISKS	DISK_NUM	CD_ROM	ISBN
Integer	Integer	Integer	Char 1	Integer	Char 1	Char 15

BOOK_A

B_CALL	TC_CODE	AUTHOR
Var Char 40	Char 1	Var Char 50

BOOK_H

B_CALL	TC_Code	B_HEADING
Var Char 40	Char 1	Var Char 100

EACH_B

B_REG	B_CALL	TC_CODE	B_STATUS	GET_BOOK
Char 6	Var Char 40	Char 1	Char 1	Date

LOSE_BOOK	GET_RETURN	PLACEPUBLISH	YEAR_PUB
Date	Date	Var Char 30	Char 4

BORROW_R

USER_ID	B_REG	BORROW_DATE	RETURN_DATE
Char 8	Char 6	Date	Date

RES

USER_ID	B_CALL	TC_Code	RES_DATE	RES_TIME
Char 8	Var Char 40	Char 1	Date	FLOAT

WAIT_B

USER_ID	B_REG	WAIT_DATE
Char 8	Char 6	Date

USR_RULE

RULE_ID	B_NUM	T_NUM	RES_NUM
Integer	Integer	Integer	Integer

ตารางที่ 4-1 ตารางจากไนแอมส่วนหนังสือ,การยืม,การจอง

STUDENT

S_ID	NAME	SURNAME	MAJOR	S_LEVEL
Char 8	Char 20	Char 20	Var Char 30	Char 3

MEMBER_STATUS	RULE_ID
char 1	Integer

TEACHER

TEACH_OFFIC_ID	NAME	SURNAME	DEPARTMENT	OFFICE
Char 6	Char 20	Char 20	Var Char 30	Var Char 30

MEMBER_TYPE	RULE_ID	MEMBER_STATUS
Char 1	Integer	Char 1

ตารางที่ 4-2 ตารางจากไนแอมส่วนผู้ใช้บริการ

VDO

VDO_CALL	VDO_NAME	PRODUCER	GET_VDO	TIMES
Char 6	Var Char 50	Var Char 50	Date	Integer
				CAT_VDO
				Char 1

VDO_H

VDO_CALL	VDO_HEAD
Char 6	Var Char 50

ตารางที่ 4-3 ตารางจากไนแอมส่วนโสตทัศนศึกษา

MAGAZINE

M_CALL	M_NAME	ISSN
Char 10	Var Char 100	Char 10

EACH_M

M_REG	VOLUME	EDITION_NO	PLACE_DATE	YEAR_PUB
LongInt	Integer	Integer	Char 10	Integer

M_CALL
Char 10

M_COLUMN

M_REG	COLUMN_ID
LongInt	LongInt

COLUMN_A

COLUMN_ID	AUTHOR
LongInt	Char 40

EACH_C

COLUMN_ID	COL_NAME	BETWEEN_PAGE
LongInt	Var Char 150	Char 15

ตารางที่ 4-4 ตารางจากไนแอมส่วนวารสาร

THESIS

THESIS_CALL	T_NAME	E_NAME	MAJOR
Char 15	Var Char 100	Var Char 100	Var Char 20

THESIS_LEVEL	ACADEMIC_YEAR
Char 1	Char4

EACH_T

THESIS_REG	THESIS_CALL
Char 10	Char 15

ADVISORS

THESIS_CALL	ADVISOR
Char 15	Char 40

THESIS_S

STD_ID	STUDENT	THESIS_CALL
Char 8	Char 40	Char 15

ตารางที่ 4-5 ตารางจากไนแอมส่วนปริญญานิพนธ์

4.3 คาต้าดิกชันนารีของตารางฐานข้อมูล (Data dictionary for tables)

BOOK			
Label Type	Type	Sample Value	Description
B_CALL	Var Char 40	QA76.758.s657, QA76.9.D3K67 005.74-dc20	เลขหมู่หนังสือ เป็น Null ไม่ได้
TC_CODE	Char 1	1,2,3	เป็นCodeที่ใช้แยกหนังสือ ต่างชื่อกันแต่ใช้เลขหมู่ เดียวกัน เป็น Null ไม่ได้
B_NAME	Var Char 100	DELPHI, Object Lifecycles Modeling	ชื่อหนังสือ เป็น Null ไม่ได้
B_EXTEND	Var Char 100	Teach Yourself In 21 Day, The World in State, Programmer's Reference and SDK Guide	ชื่อขยาย เป็น Null ได้
NAMEGROUP	Var Char 100		ชื่อชุด เป็น Null ไม่ได้
THAI_TEXT	Char 1	'T','X'	ประเภทหนังสือ 'T' หมายถึง หนังสือไทย 'X' หมายถึง Text เป็นNull ไม่ได้
PUBLISHER	Var Char 50	McGRAW-HILL, ฟิสิกส์เซ็นเตอร์	สำนักพิมพ์ เป็น Null ได้
A_NUM	Integer	200,800	จำนวนหน้าอารบิค เป็น Null ได้
R_NUM	Integer	20,30	จำนวนหน้าโรมัน เป็น Null ได้
T_NUM	Integer	10,200	จำนวนหน้าไทย เป็น Null ได้
DISKS	Char 1	Y,N	หนังสือมีแผ่นดิสก์หรือไม่ 'Y' คือ มี 'N' คือ ไม่มี เป็น Null ไม่ได้
DISK_NUM	Integer	1,2,3	จำนวนแผ่นดิสก์ เป็น Null ได้

ISBN	Char 15	0-672-30470-8, 0-07-065161-2	International Standard Book Number เป็น Null ได้
CD_ROM	Char 1	'Y','N'	หนังสือมีแผ่นซีดีหรือไม่ 'Y' คือ มี 'N' คือ ไม่มี เป็น Null ไม่ได้

BOOK_A			
AUTHOR	Var Char 50	Andrew Wozniewicz ครรชิต มาลัยวงศ์	ชื่อผู้แต่ง เป็น Null ไม่ได้

BOOK_H			
B_HEADING	Var Char 100	CODING THEORY C(COMPUTER PROGRAM LANGUAGE)	หัวเรื่องหนังสือ เป็น Null ไม่ได้

EACH_B			
B_REG	Char 6	009149	รหัสหนังสือ เป็น Null ไม่ได้
B_STATUS	Char 1	'B','I','R','L'	สถานะหนังสือ 'B' =ยืม, 'I'=คืน, 'R'=จอง, 'L'=หาย เป็น Null ไม่ได้
GET_BOOK	Date		วันรับหนังสือเข้าห้องสมุด เป็น Null ได้
LOSE_BOOK	Date		วันที่หนังสือหาย เป็น Null ได้
GET_RETURN	Date		วันที่ได้หนังสือคืน เป็น Null ได้
PLACEPUBLISH	Var Char 30	England,NewYork	สถานที่พิมพ์ เป็น Null ได้
YEAR_PUB	Char 4	2533,1980	ปีที่พิมพ์ เป็น Null ได้

BORROW_R			
USER_ID	Char 8	35104271,35104448	รหัสผู้ใช้บริการ เป็น Null ไม่ได้
BORROW_DATE	Date		วันยืม เป็น Null ไม่ได้
RETURN_DATE	Date		วันคืน เป็น Null ไม่ได้

WAIT_B			
WAIT_DATE	Date		วันเริ่มรอการรับหนังสือจอง เป็น Null ไม่ได้

RES			
RES_DATE	Date		วันจอง เป็น Null ไม่ได้
Res_TIME	Float		เวลาที่จอง เป็น Null ไม่ได้

TEACHER			
TEACH_OFFIC_ID	Char6	010066	รหัสอาจารย์,เจ้าหน้าที่ เป็น Null ไม่ได้
NAME	Char 20	สมศักดิ์,ประทีป	ชื่ออาจารย์,เจ้าหน้าที่ เป็น Null ไม่ได้
SURNAME	Char 20	มิตะถา,บัญญัตินพรัตน์	นามสกุล เป็น Null ไม่ได้
MEMBER_TYPE	Char 1	'A', 'B', 'C' , 'D'	'A'=อาจารย์ในคณะวิศวกรรมศาตร์ 'B'=อาจารย์นอกคณะ 'C'=เจ้าหน้าที่ในคณะ 'D'=เจ้าหน้าที่นอกคณะ
DEPARTMENT	Var Char 30	อิเลคทรอนิค	ภาควิชา เป็น Null ได้
OFFICE	Var Char 30	กิจกรรมนักศึกษา	แผนก เป็น Null ได้
MEMBER_STATUS	Char 1	'Y', 'N'	สถานะการเป็นสมาชิก 'Y'=ยืมได้ ,'N'=หมดอายุ

Student			
NAME	Char 20		ชื่อ เป็น Null ไม่ได้
SURNAME	Char 20		สกุล เป็น Null ไม่ได้
MAJOR	Var Char 30	คอมพิวเตอร์,วัดคุม	ภาค เป็น Null ได้
S_LEVEL	Char 3	'B' , 'MTP', 'MTM', 'MSP', 'MSM', 'PT' , 'PS'	ระดับ 'B' =ปริญญาตรี 'MTP'=ปริญญาโท ทดลองวิจัยไฟฟ้า 'MTM'=ปริญญาโท ทดลองวิจัยเครื่องกล 'MSP'=โทสามัญไฟฟ้า 'MSM'=โทสามัญ เครื่องกล 'PT'=เอกทดลองวิจัย 'PS'=เอกสามัญ เป็น Null ไม่ได้
USR_RULE			
RULE_ID	Integer	1,2,3	รหัสระเบียบห้องสมุด เป็น Null ไม่ได้
B_NUM	Integer	3,5,7	จำนวนเล่มที่ยืมได้ เป็น Null ไม่ได้
T_NUM	Integer	7,30	จำนวนวันที่ยืมได้ เป็น Null ไม่ได้
RES_NUM	Integer	1,2,3	จำนวนจอง เป็น Null ไม่ได้
VDO			
VDO_CALL	Char 6	VC 0084 , VC001-8	เลขหมู่ VDO เป็น Null ไม่ได้
VDO_NAME	Var Char 50	การประหยัดพลังงานในโรง งานอุตสาหกรรม,Follow Me	ชื่อ VDO เป็น Null ไม่ได้
PRODUCER	Var Char 50	สภาอุตสาหกรรมแห่ง ประเทศไทย,BBC English	ผู้จัดทำ เป็น Null ได้

Get_VDO	Date		วันรับVDO เป็น Null ได้
TIMES	Integer	120,90	ความยาว(นาที) เป็น Null ได้
CAT_VDO	Char 1	'T', 'E','I' ,'O'	พากย์ภาษา 'T'=ไทย , 'E'=อังกฤษ 'I'=ไทย+อังกฤษ 'O'=อื่นๆ เป็น Null ไม่ได้

VDO_H			
VDO_HEAD	Var Char 50	สารคดี,บันเทิง,ประวัติศาสตร์	หัวเรื่อง เป็น Null ไม่ได้

THESIS			
THESIS_CALL	Char 15	T 36254 พ4 T 2801 ก6	เลขหมู่ปริญญานิพนธ์ เป็น Null ไม่ได้
T_NAME	Var Char 100	การพัฒนาระบบงานประยุกต์บนระบบข้อมูลแบบไคลเอ็นท์เซิร์ฟเวอร์	ชื่อภาษาไทย เป็น Null ไม่ได้
E_NAME	Var Char 100	CLIENT-SERVER - DATABASE - APPLICATION - DEVELOPMENT	ชื่อภาษาอังกฤษ เป็น Null ไม่ได้
MAJOR	Var Char 20	คอมพิวเตอร์,วัดคุม,ไฟฟ้า	ภาควิชา เป็น Null ไม่ได้
THESIS_LEVEL	Char 1	'B' , 'M' , 'P'	ระดับ เป็น Null ไม่ได้
ACADEMIC_YEAR	Char 4	2537 , 2536	ปีการศึกษา เป็น Null ไม่ได้

EACH_T			
THESIS_REG	Char 10	017090 c.1 017091 c.2	เลขทะเบียนปริญญานิพนธ์ เป็น Null ไม่ได้

THESIS_S			
STD_ID	Char 8	35104271	รหัสนักศึกษา เป็น Null ไม่ได้
STUDENT	Char 40	สมชาย ยอดเมือง วีระชัย เจริญอุดมศักดิ์	ชื่อนักศึกษาปริญญา นิพนธ์ เป็น Null ไม่ได้

ADVISORS			
ADVISOR	Char 40	สมศักดิ์ มิตะถา	ชื่ออาจารย์ที่ปรึกษา เป็น Null ไม่ได้

MAGAZINE			
M_CALL	Char 10	103.1/2 , 166.1/15	เลขหมู่วารสาร เป็น Null ไม่ได้
M_NAME	Var Char 100	SOLID-STATE CIRCUIT, AUTOMATIC CONTROL	ชื่อวารสาร เป็น Null ไม่ได้
ISSN	Char 10	0018-9200 0125-7781	เป็น Null ได้

EACH_M			
M_REG	LongInt	1264 , 2429	เลขทะเบียนวารสาร เป็น Null ไม่ได้
VOLUME	Integer	25 , 38	เป็น Null ได้
EDITION_NO	Integer	4, 11	ฉบับที่ เป็น Null ได้
PLACE_DATE	Char 10	AUG-DEC สค.-ธช.	กำหนดออกวารสาร เป็น Null ได้
YEAR_PUB	Integer	1993 ,2530	ปีพิมพ์ เป็น Null ได้

M_COLUMN			
COLUMN_ID	LongInt	1,100,1025	รหัสบทความ เป็น Null ไม่ได้

COLUMN_A			
AUTHOR	Char 40	T.Takeshima D.Senderowicz	ผู้แต่งบทความ เป็น Null ไม่ได้

EACH_C			
COL_NAME	Var Char 150	A 55-ns 16-Mb DRAM with Built-in Self-Test Function Using Microprogram ROM , An Analytical Maximum Toggle Frequency Expression and its Application to Optimizing High-Speed ECL Frequency Dividers	ชื่อบทความ เป็น Null ไม่ได้
BETWEEN_PAGE	Char 15	903-911 , 1005-1007	เลขหน้าของบทความ เป็น Null ได้

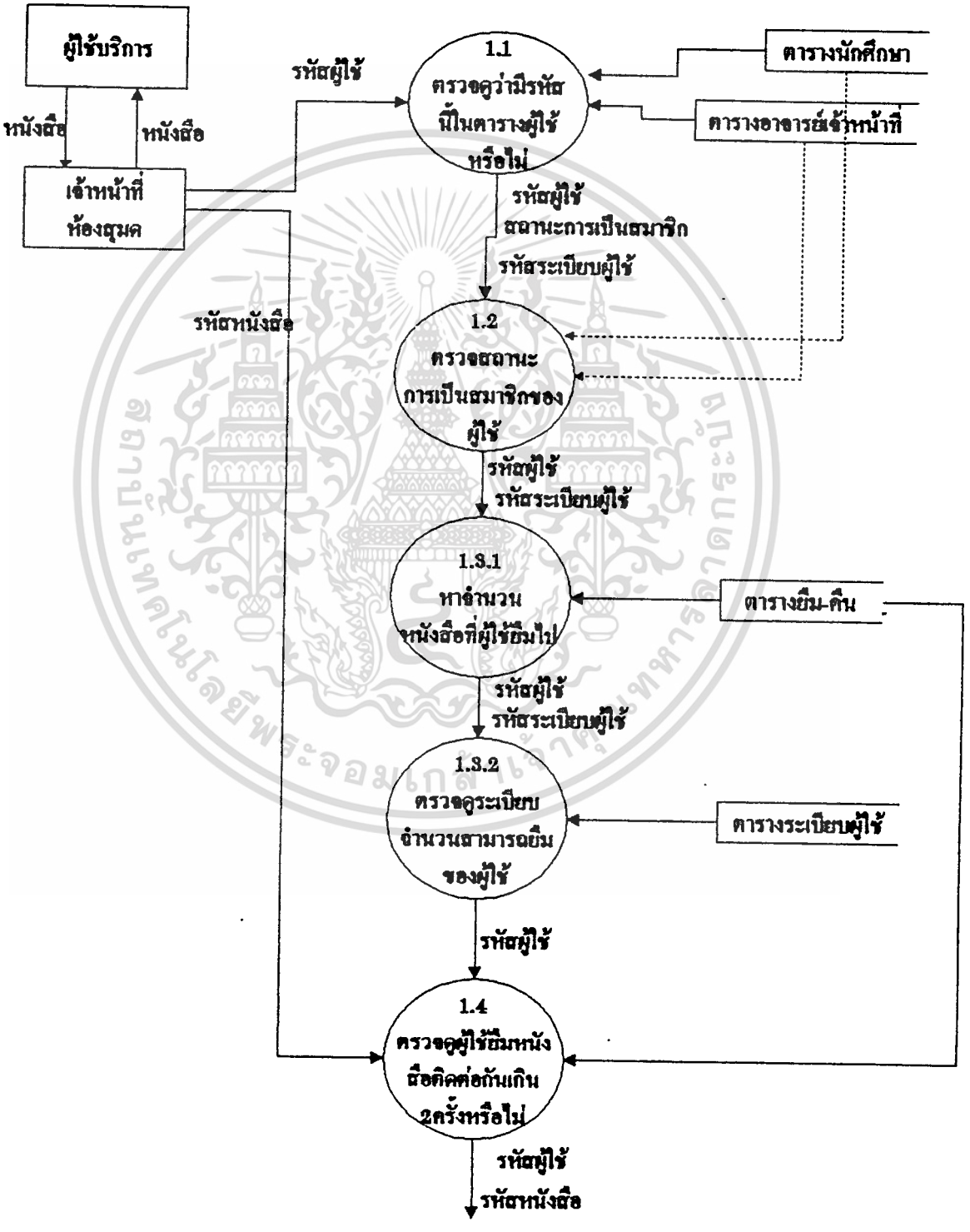
ตารางที่ 4-6 คาต้าดิกชันนารีของตารางฐานข้อมูล

เมื่อได้ออกแบบตารางเพื่อจัดเก็บข้อมูลเสร็จ จากนี้ไปก็จะเป็นโฟลว์ชาร์ตที่ทำต่อจากบทที่ 3 โดยจะแตกโฟลว์ชาร์ตให้ลึกจนถึงระดับตาราง

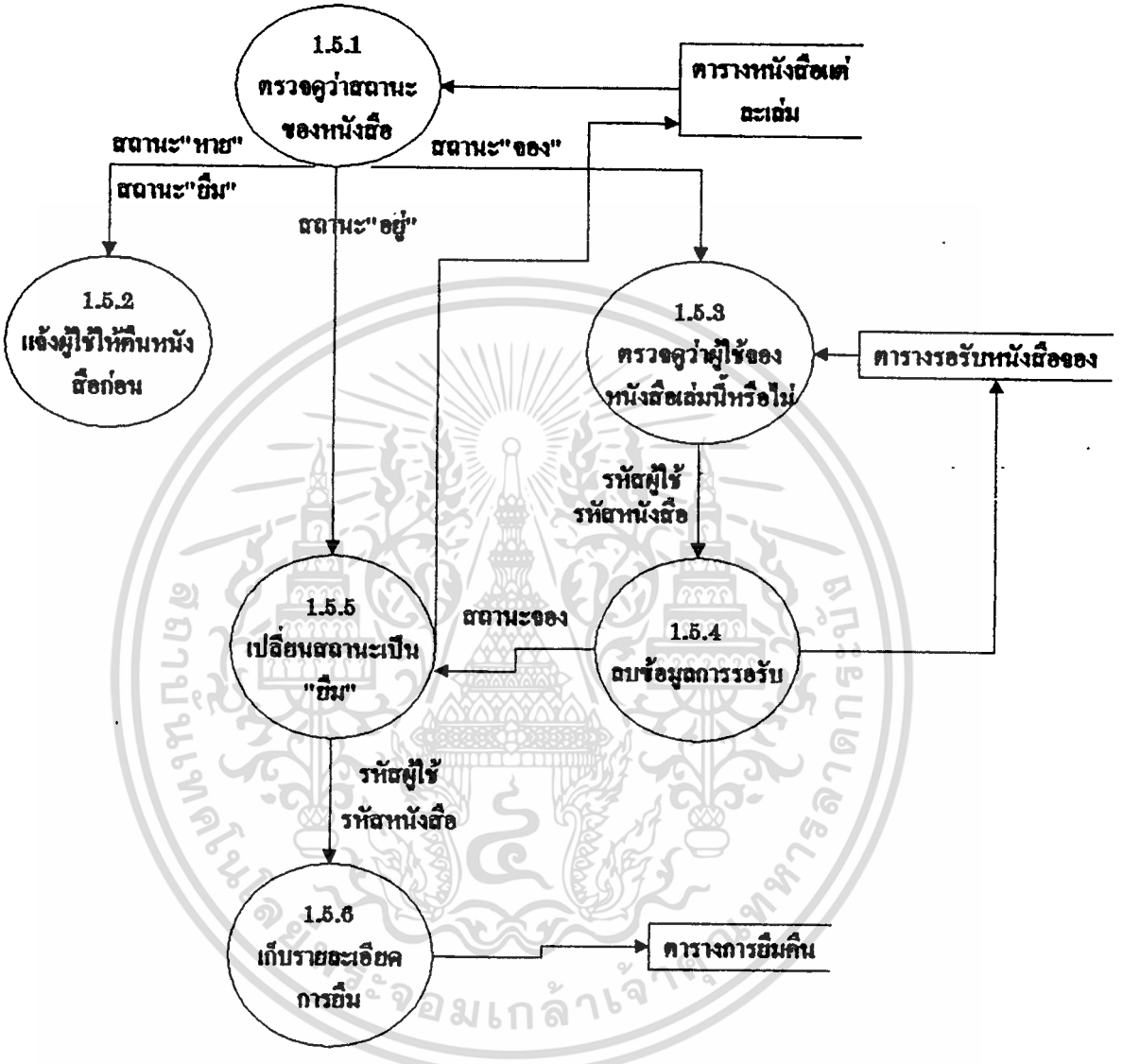
4.4 คาต้าโฟลว์ไดอะแกรมในระดับตาราง

4.4.1 งานบริการ

4.4.1.1 การยืมหนังสือ

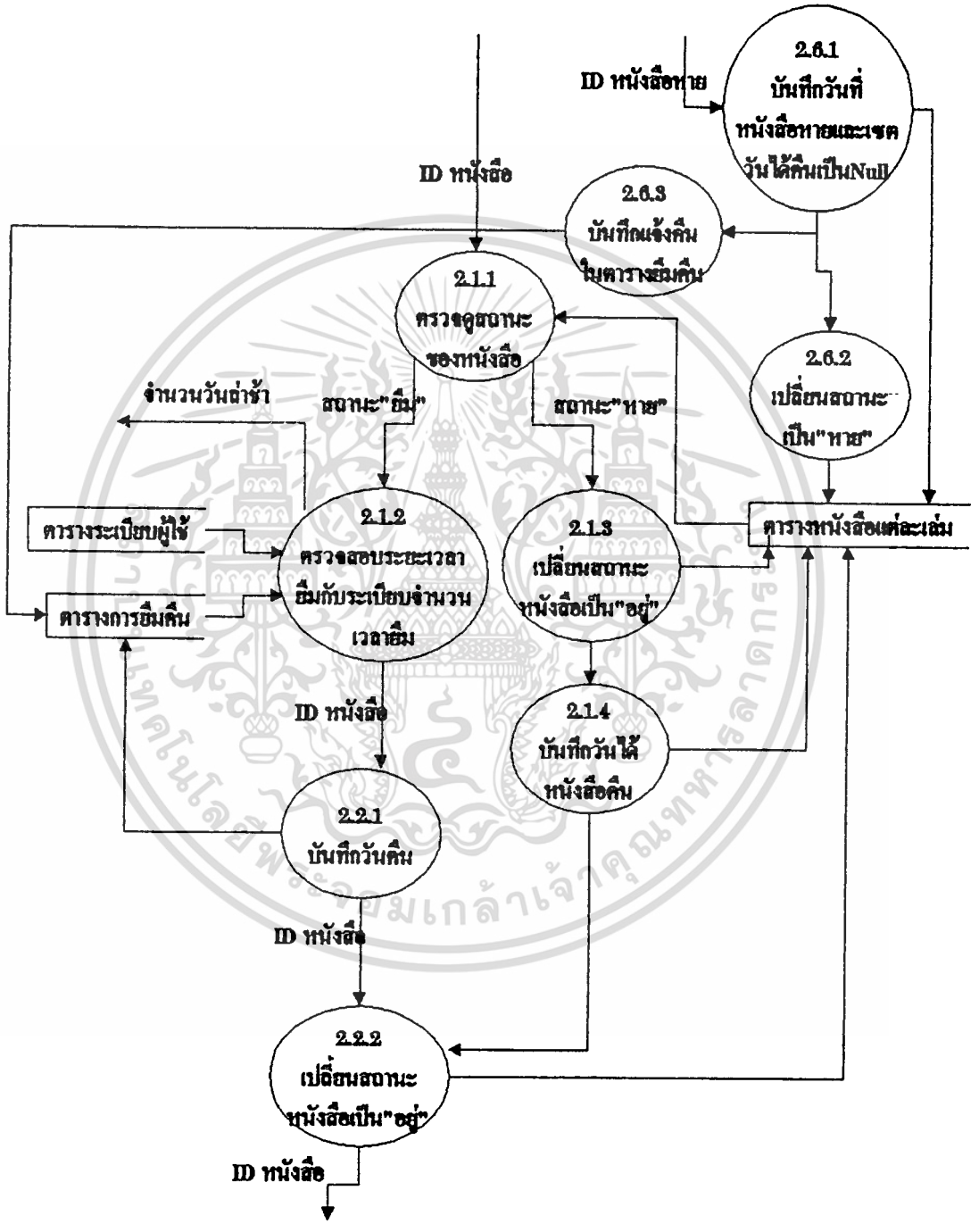


รูปที่ 4-22 คาต้าโฟลว์การยืมหนังสือในระดับตาราง

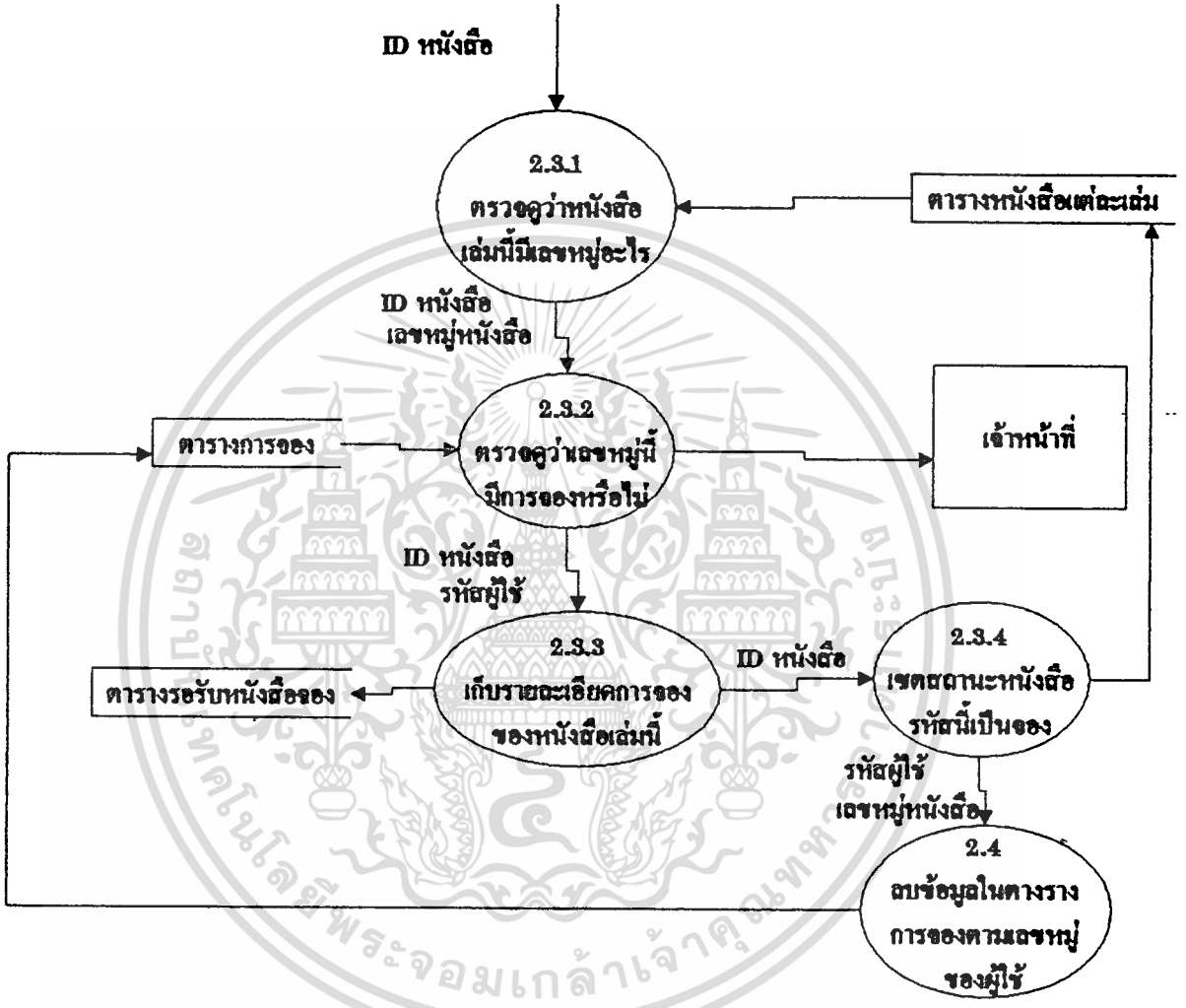


รูปที่ 4-23 คาต้าโฟลว์การยืมหนังสือในระดับตาราง(ต่อ)

4.4.1.2 การคืนหนังสือ

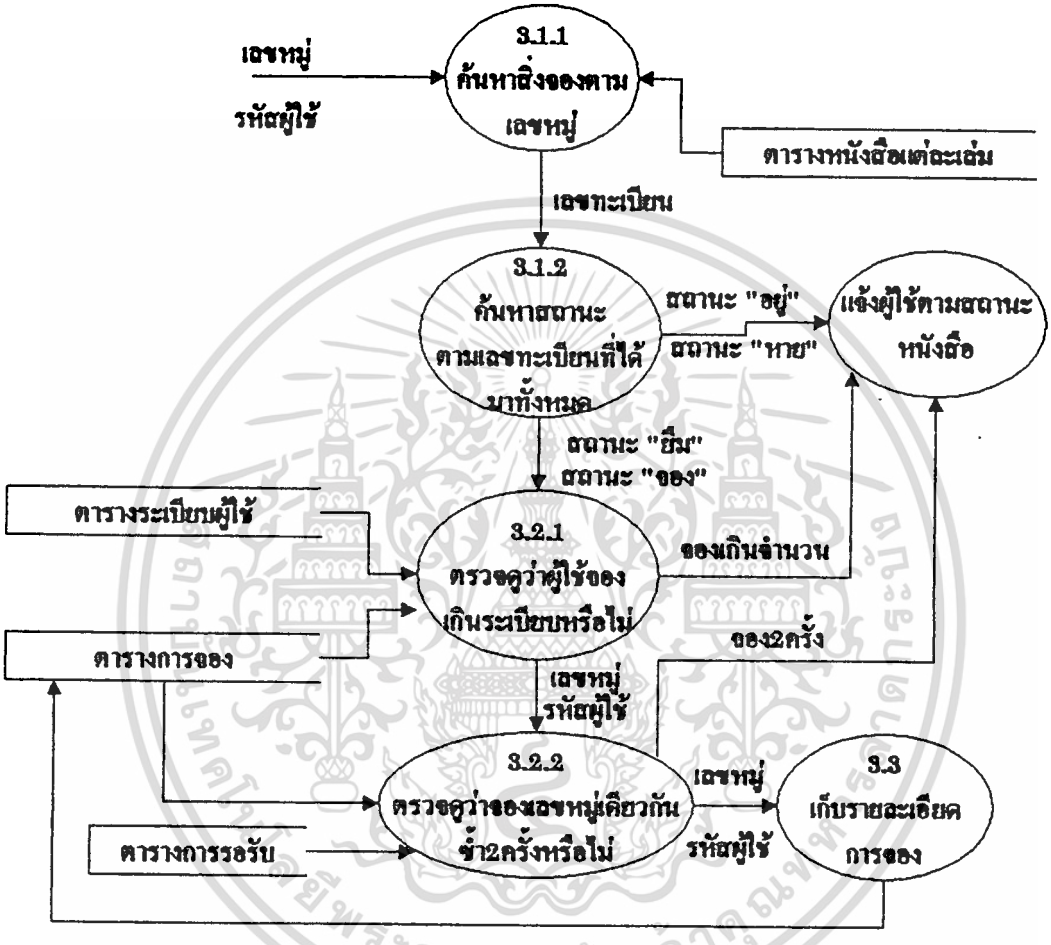


รูปที่ 4-24 คาต้าโฟลว์การคืนหนังสือในระดับตาราง

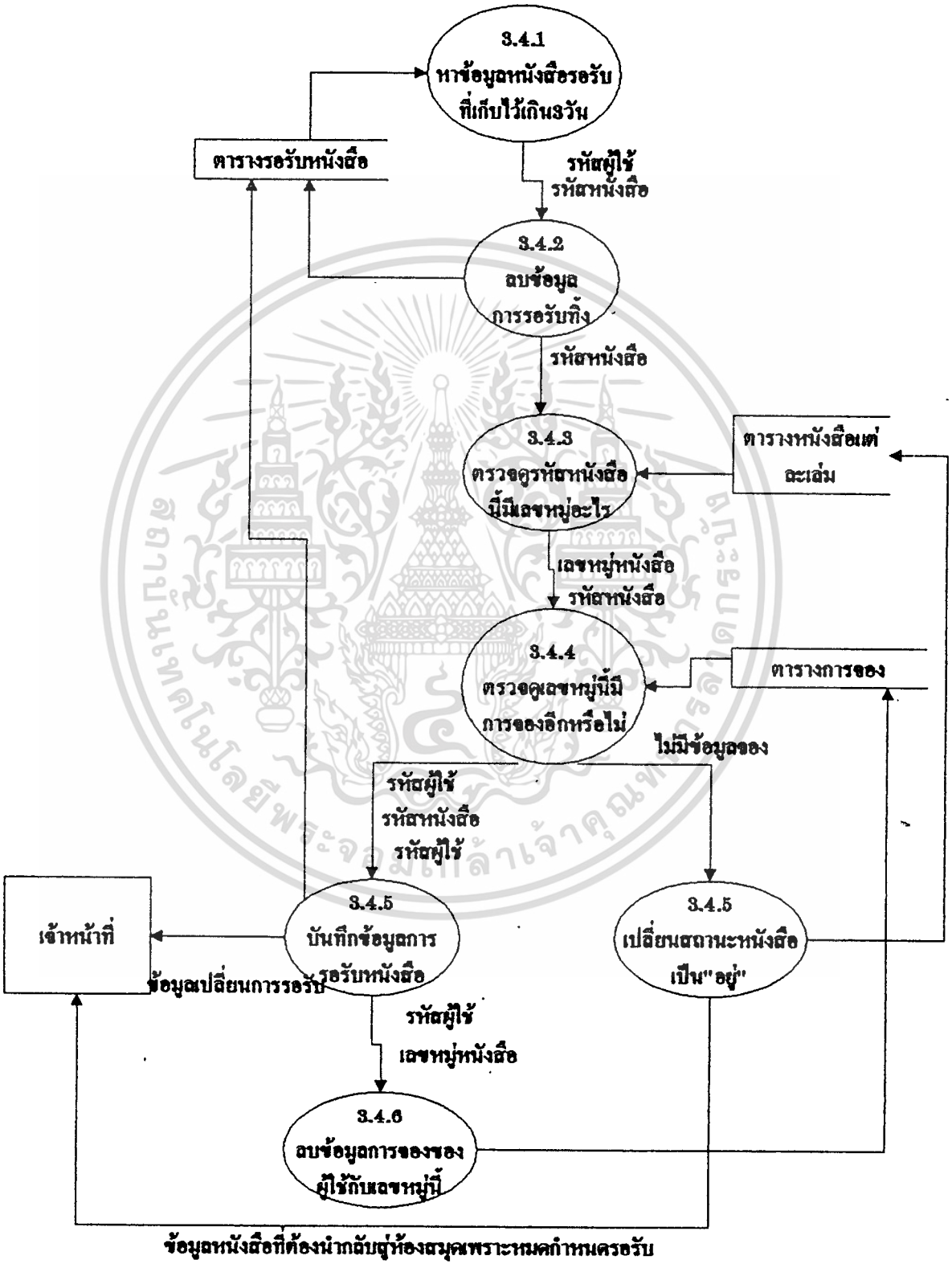


รูปที่ 4-25 คาต้าโฟลว์การคืนหนังสือในระดับตาราง (ต่อ)

4.4.1.3 การจองหนังสือ

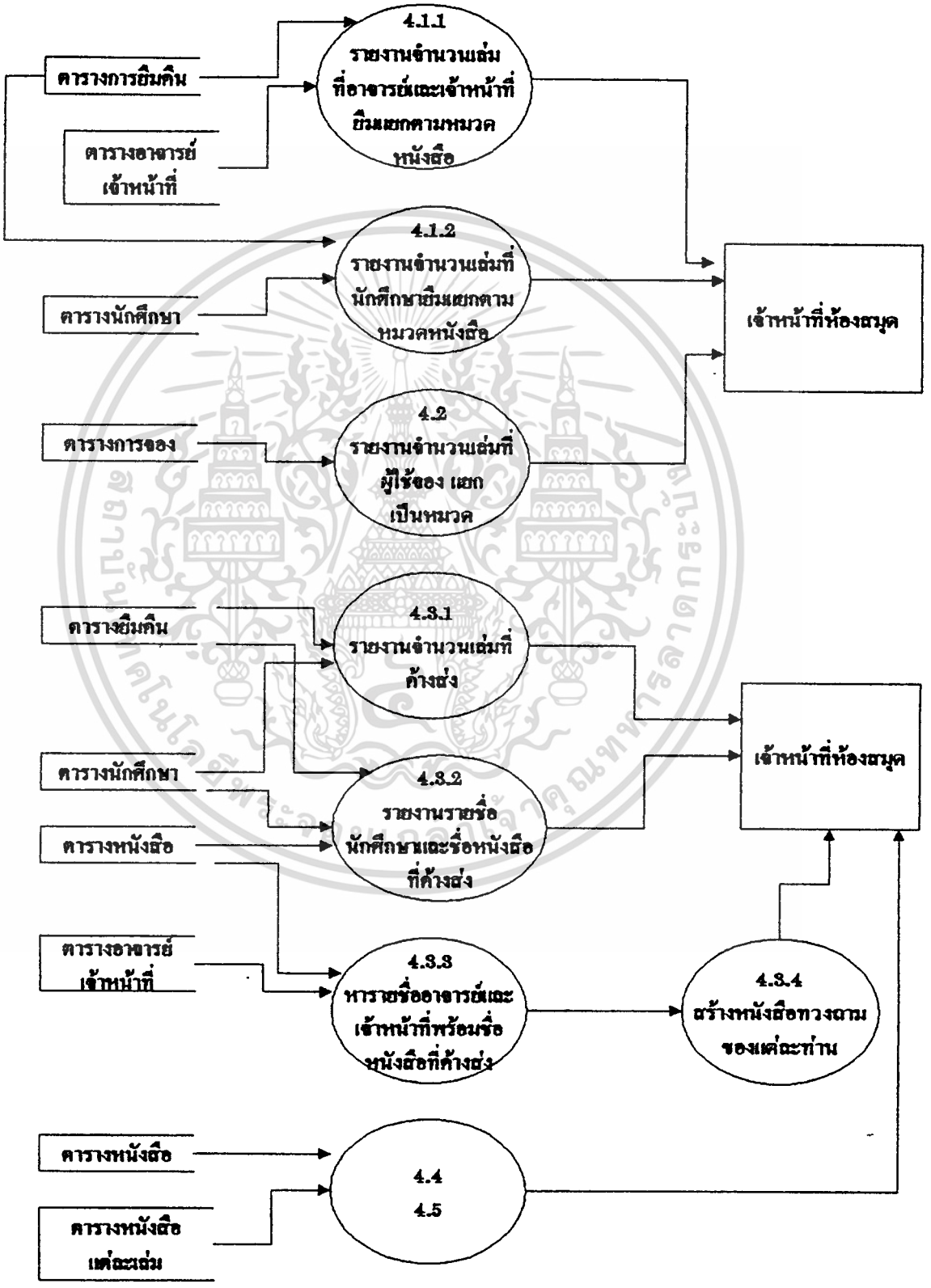


รูปที่ 4-26 คาต้าโฟลว์การจองหนังสือในระดับตาราง



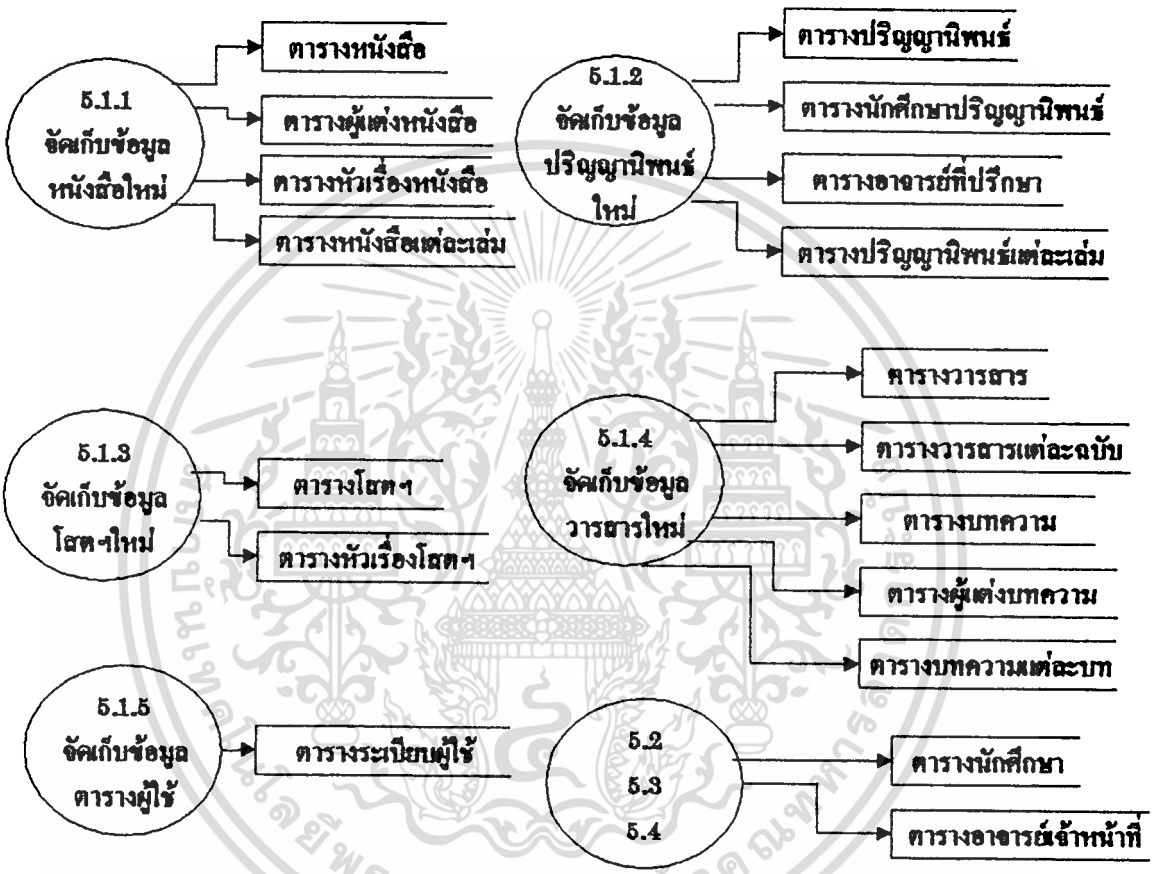
รูปที่ 4-27 คาต้าโฟลว์การจองหนังสือในระดับตาราง(ต่อ)

4.4.1.4 การสร้างรายงาน



รูปที่ 4-28 คาต้าโฟลว์การสร้างรายงานในระดับตาราง

4.4.1.5 การจัดเก็บข้อมูลใหม่



รูปที่ 4-29 ดาต้าโฟลว์การจัดเก็บข้อมูลใหม่ในระดับตาราง

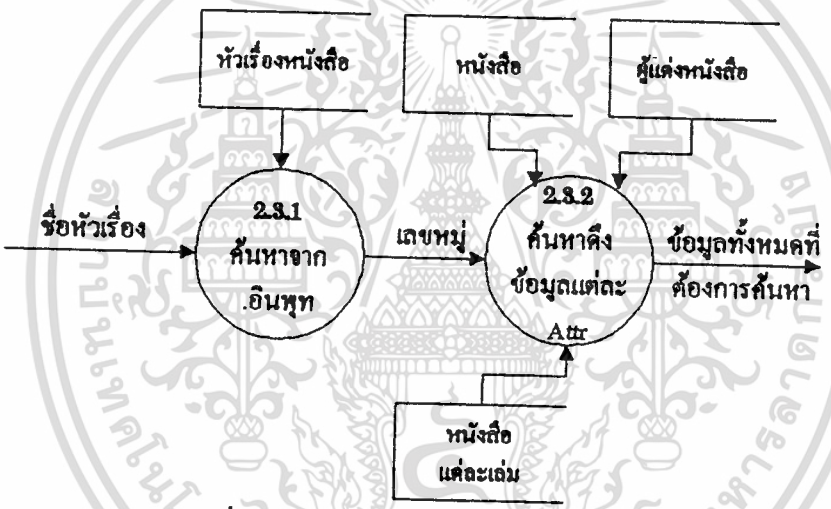
4.4.2 งานค้นหา

เราสามารถเขียนคาต้าโฟล์วไดอะแกรม ให้ลึกถึงระดับตารางโดยจะแตกเฉพาะส่วนที่ 2 (ส่วนค้นหา จากรูป 3.10) เพราะเป็นส่วนที่สำคัญที่สุดและเป็นส่วนที่ต้องทำงานกับตารางฐานข้อมูล โดยจะแบ่งออกเป็น

4.4.2.1 การค้นหาหนังสือ

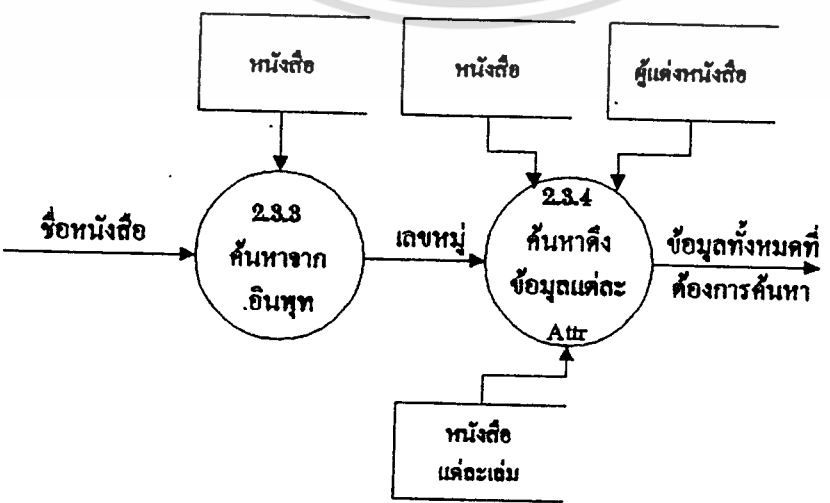
จะรับ อินพุท มาจากส่วนที่ 1 และ อินพุท ที่รับเข้ามา จะมีหลักๆ อยู่ 4 อย่าง คือ

1. หัวเรื่อง



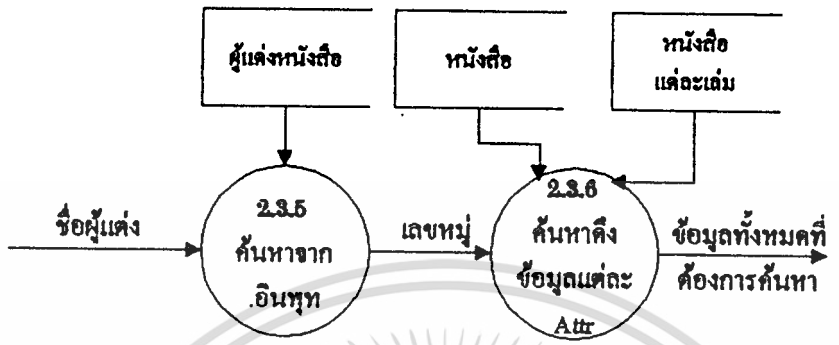
รูปที่ 4-30 คาต้าโฟลว์การค้นหาหนังสือตามหัวเรื่อง

2. ชื่อเรื่อง



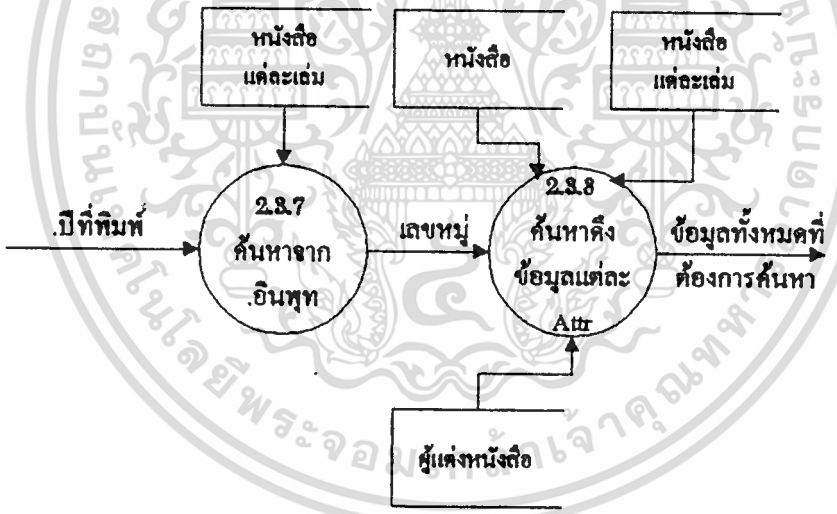
รูปที่ 4-31 คาต้าโฟลว์การค้นหาหนังสือตามชื่อเรื่อง

3.ชื่อผู้แต่ง



รูปที่ 4-32 คาต้าโฟลว์การค้นหาหนังสือตามชื่อผู้แต่ง

4.ปีที่พิมพ์

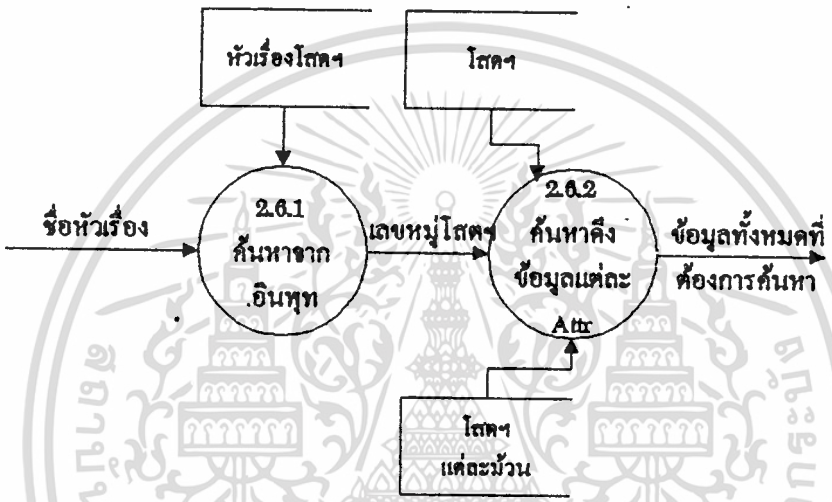


รูปที่ 4-33 คาต้าโฟลว์การค้นหาหนังสือตามปีที่พิมพ์

4.4.2.2 การค้นหาโสตทัศนูปกรณ์

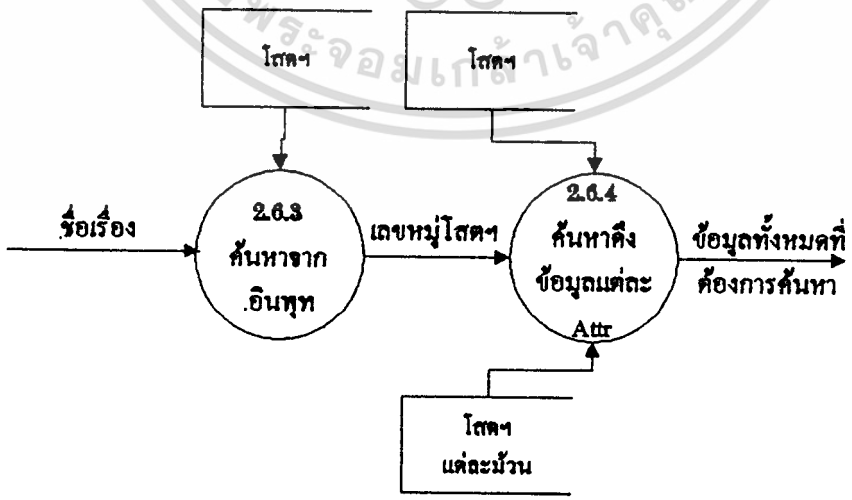
โสตทัศนูปกรณ์จะมีเฉพาะ วีดีโอ เท่านั้น และจาก ตารางที่ออกแบบไว้ ฉะนั้น ในการค้นหาจะมีหลักๆ อยู่ 2 อย่าง คือ

1.หัวเรื่อง



รูปที่ 4-34 คาต้าโฟลว์การค้นหาโสตฯตามหัวเรื่อง

2.ชื่อเรื่อง

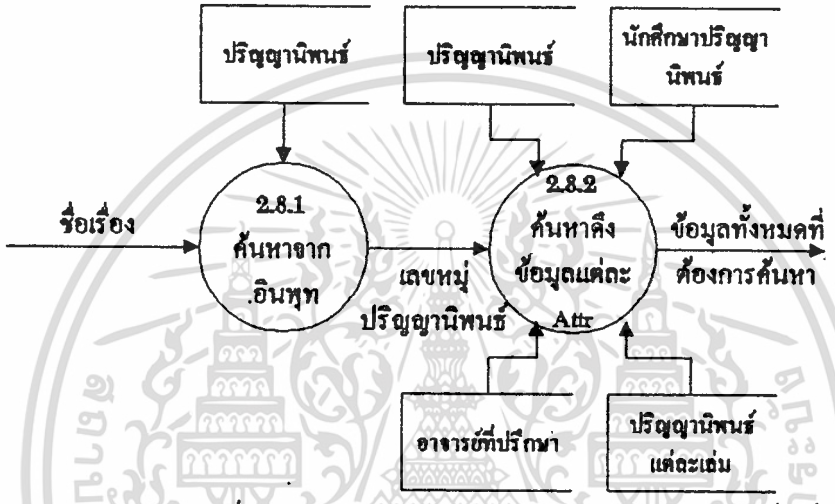


รูปที่ 4-35 คาต้าโฟลว์การค้นหาโสตฯตามชื่อเรื่อง

4.4.2.3 การค้นหาปริญญานิพนธ์

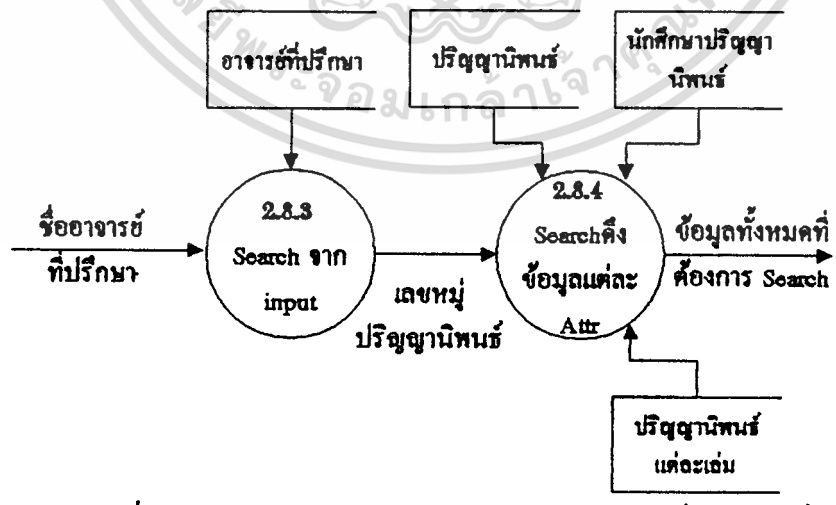
จากตารางที่ออกแบบไว้ เราสามารถค้นหาตาม key ต่างๆ ซึ่งจะมีหลักๆ อยู่ 5 อย่าง คือ

<u>1.ชื่อเรื่อง</u>



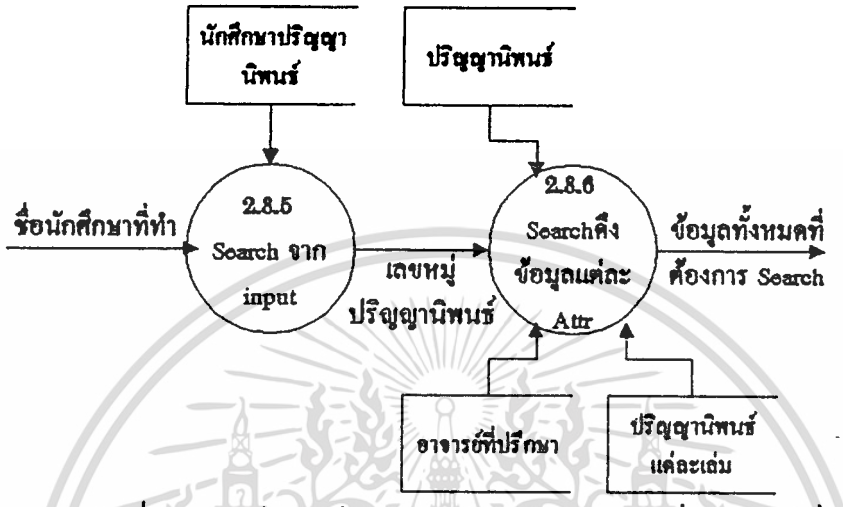
รูปที่ 4-36 คาต้าโฟลว์การค้นหาปริญญานิพนธ์ตามชื่อเรื่อง

<u>2.ชื่ออาจารย์ที่ปรึกษา</u>



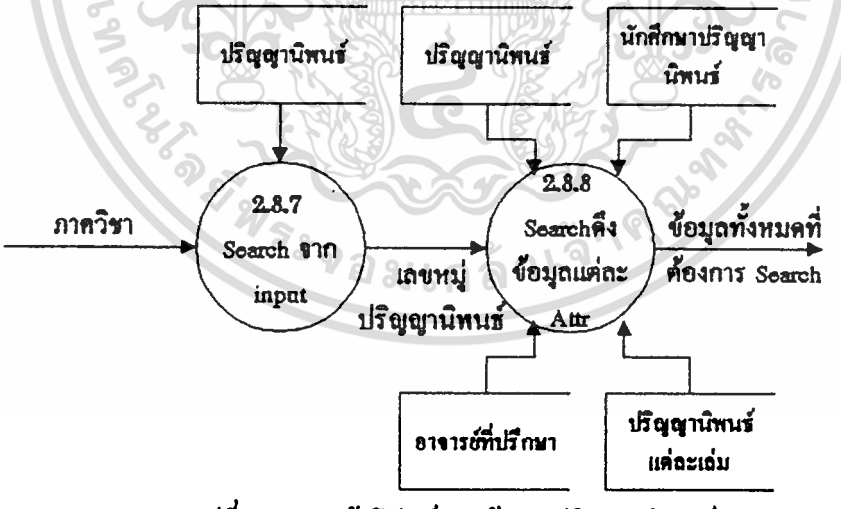
รูปที่ 4-37 คาต้าโฟลว์การค้นหาปริญญานิพนธ์ตามชื่ออาจารย์ที่ปรึกษา

3.ชื่อนักศึกษาที่ทำ



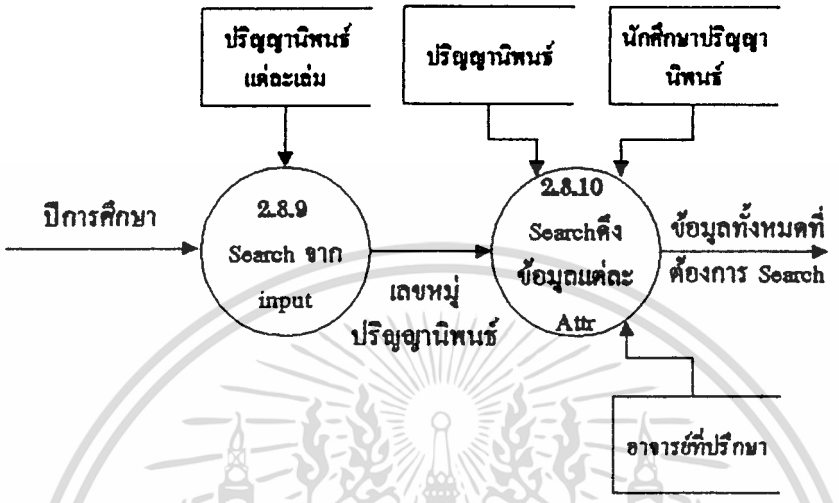
รูปที่ 4-38 คาต้าโฟลว์การค้นหาปริญญานิพนธ์ตามชื่อนักศึกษาที่ทำ

4.ภาควิชา



รูปที่ 4-39 คาต้าโฟลว์การค้นหาปริญญานิพนธ์ตามภาควิชา

5.ปีการศึกษา

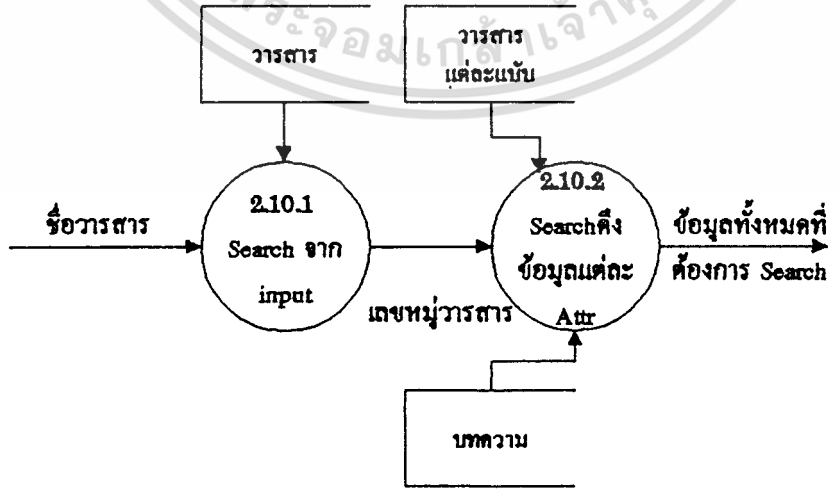


รูปที่ 4-40 คาต้าโฟลว์การค้นหาปริญญานิพนธ์ตามปีการศึกษา

4.4.2.4 การค้นหาวารสาร

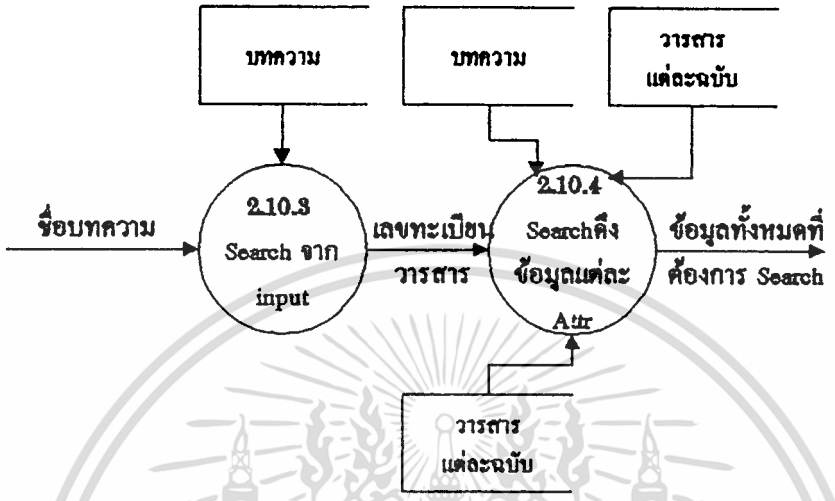
จากตารางที่ออกแบบไว้จะเห็นว่า จะแบ่งออกเป็น 2 ส่วน คือ ส่วนของบทความในวารสาร และ ส่วนของตัววารสาร ฉะนั้น เราสามารถค้นหาตาม key ต่างๆ ซึ่งจะมีหลักๆ อยู่ 3 อย่าง คือ

1.ชื่อวารสาร



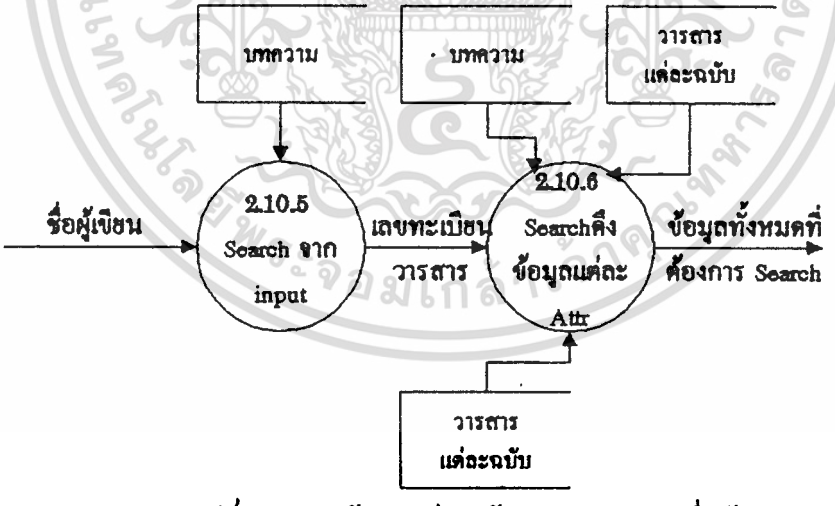
รูปที่ 4-41 คาต้าโฟลว์การค้นหาวารสารตามชื่อวารสาร

2.ชื่อบทความในวารสาร



รูปที่ 4-42 คาต้าโฟลว์การค้นหาวารสารตามชื่อบทความ

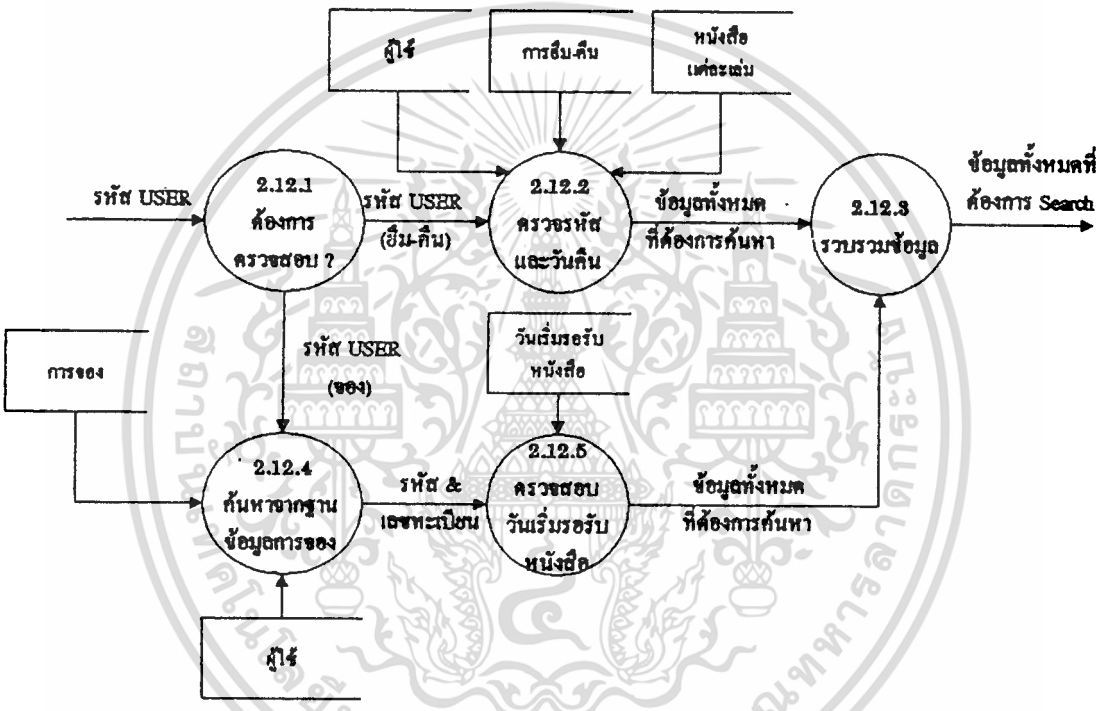
3.ชื่อผู้แต่ง/ผู้เขียน บทความ



รูปที่ 4-43 คาต้าโฟลว์การค้นหาวารสารตามชื่อผู้แต่งบทความ

4.4.2.5 การค้นหาข้อมูลผู้ใช้

จากความต้องการของห้องสมุดและ ตารางที่ออกแบบไว้จะเห็นว่า จะแบ่งออกเป็น 2 ส่วน คือ ข้อมูลการยืม-คืนหนังสือ และ ข้อมูลการจองหนังสือ ในการค้นหา เราจะค้นหาตามรหัสประจำตัวของผู้ใช้ ทั้งผู้ใช้ที่เป็นอาจารย์ เจ้าหน้าที่ และ นักศึกษา



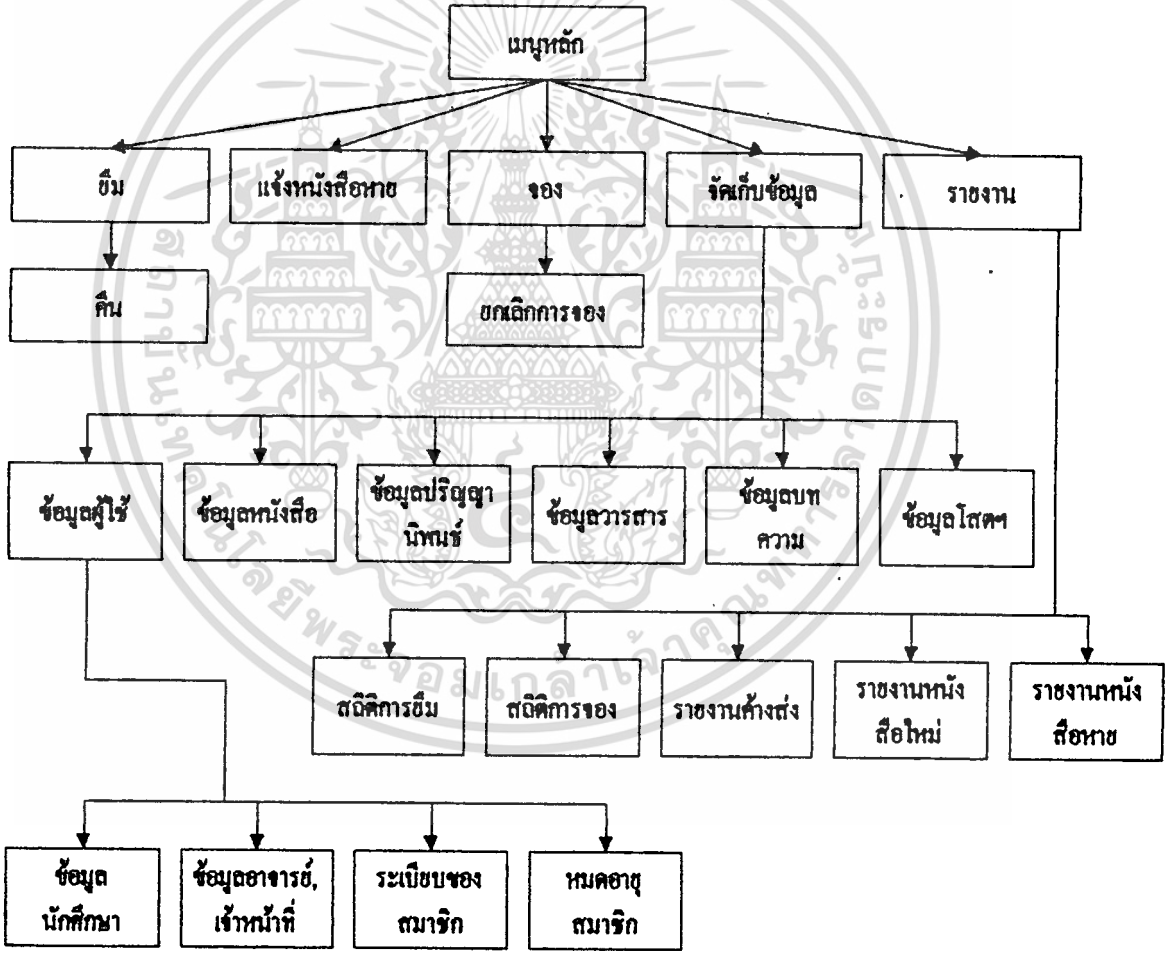
รูปที่ 4-44 คาต้าโฟลว์การค้นหาข้อมูลผู้ใช้

4.5 ออกแบบหน้าจอและรายงานของระบบงานห้องสมุด

4.5.1 งานบริการ

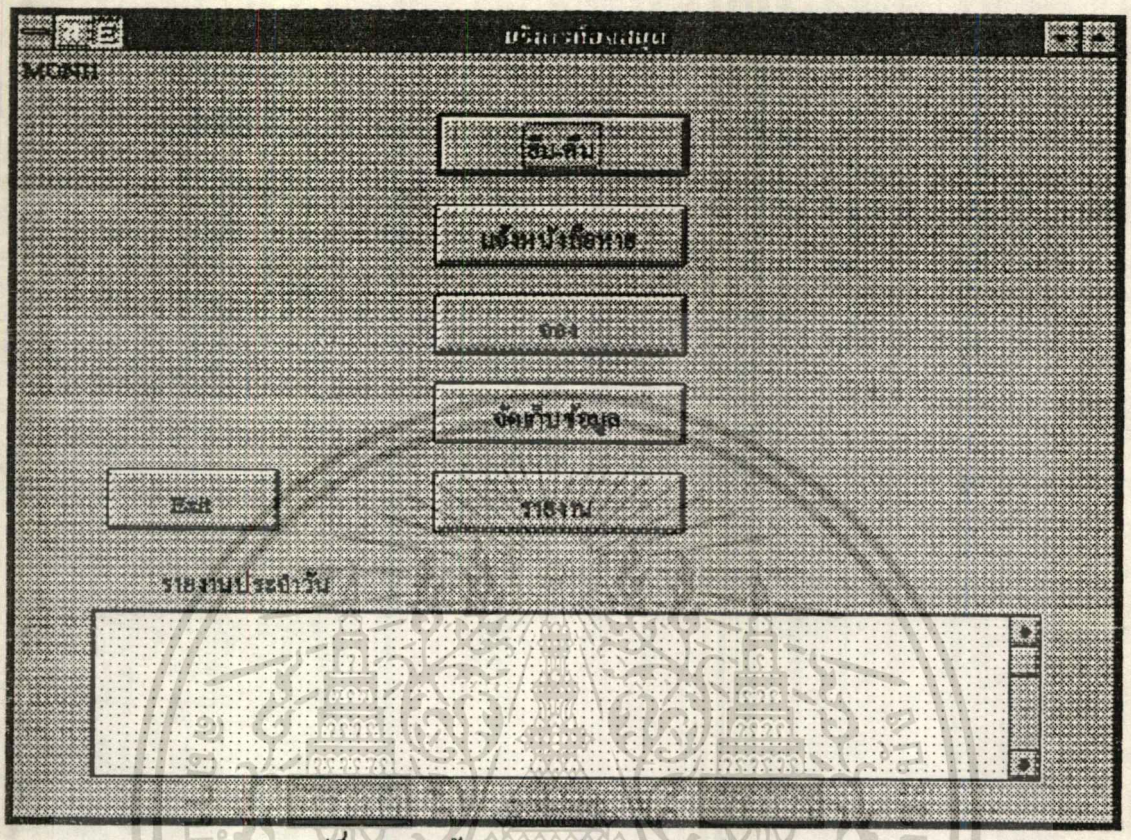
4.5.1.1 การออกแบบหน้าจอ

จากที่ทราบแล้วว่าในส่วนงานบริการมีหน้าที่สำคัญ 5 อย่างด้วยกัน ดังนั้นในการออกแบบหน้าจอจึงได้แบ่งตามลักษณะของงานซึ่งโครงสร้างของหน้าจอ มีลักษณะดังต่อไปนี้

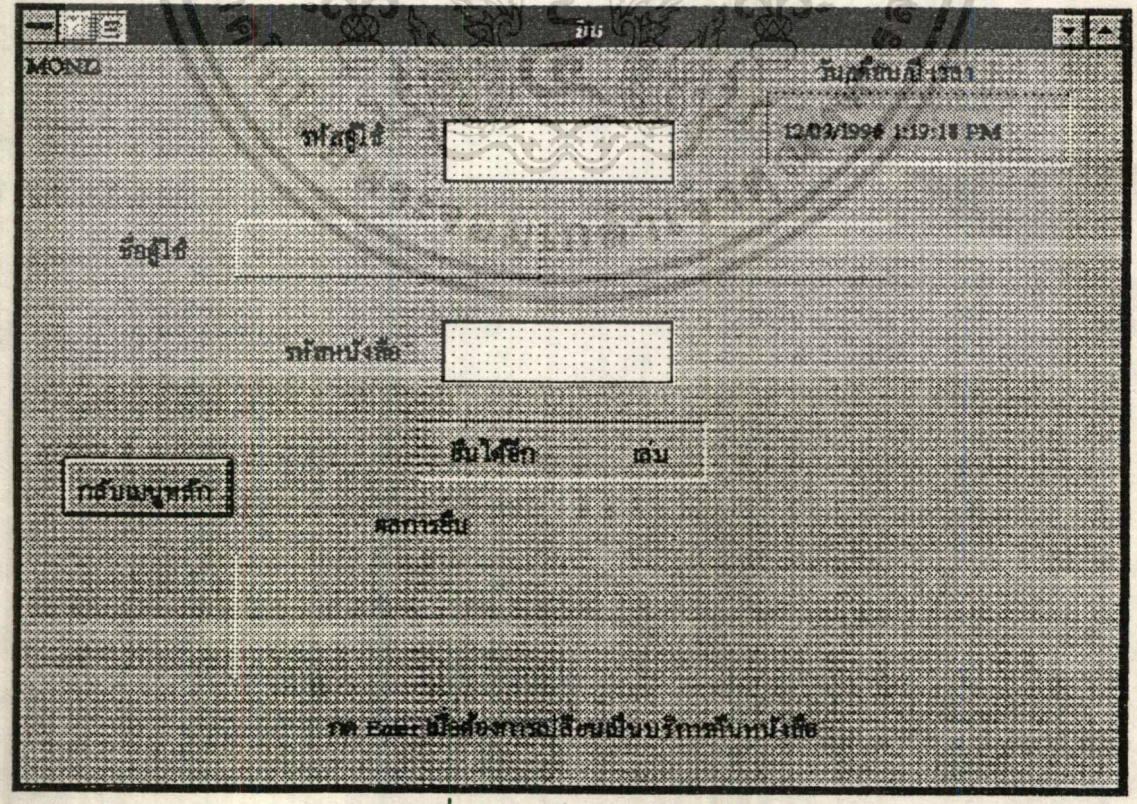


รูปที่ 4-45 โครงสร้างของหน้าจองานบริการ

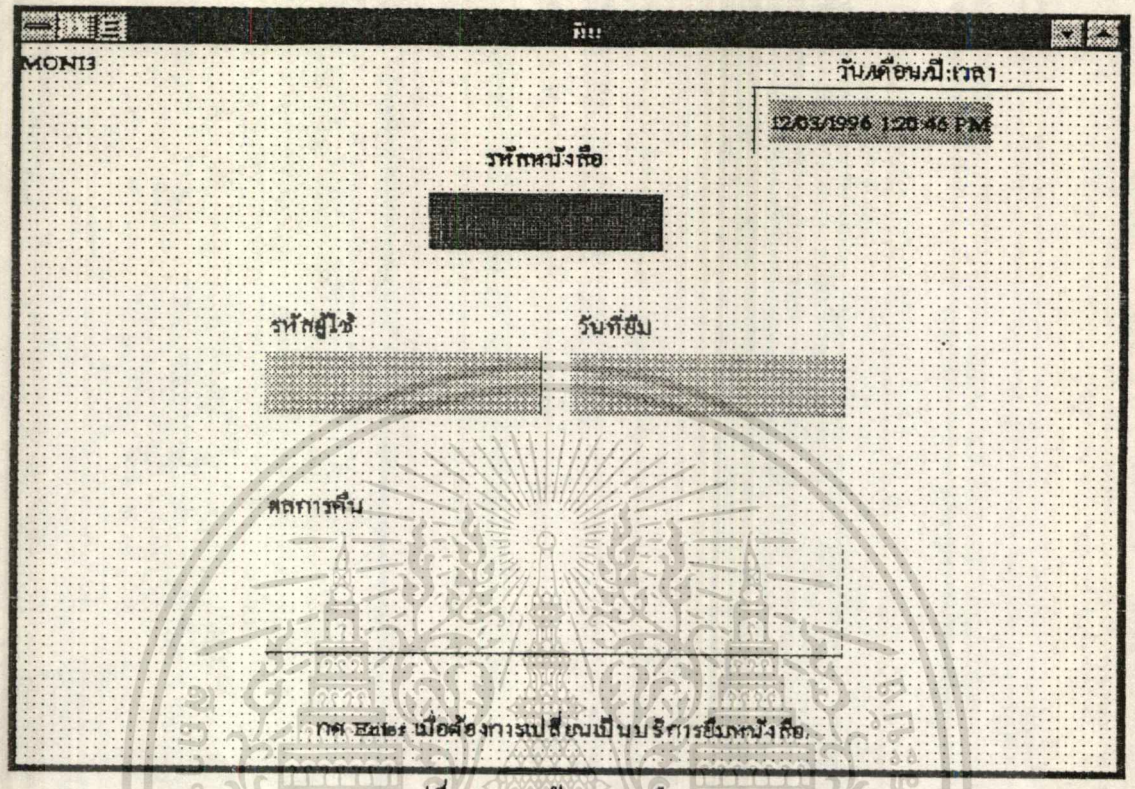
ต่อจากนี้จะเป็นหน้าจอต่างๆของส่วนบริการ ที่ออกแบบไว้ก่อนที่จะเขียนโปรแกรมต่อไป



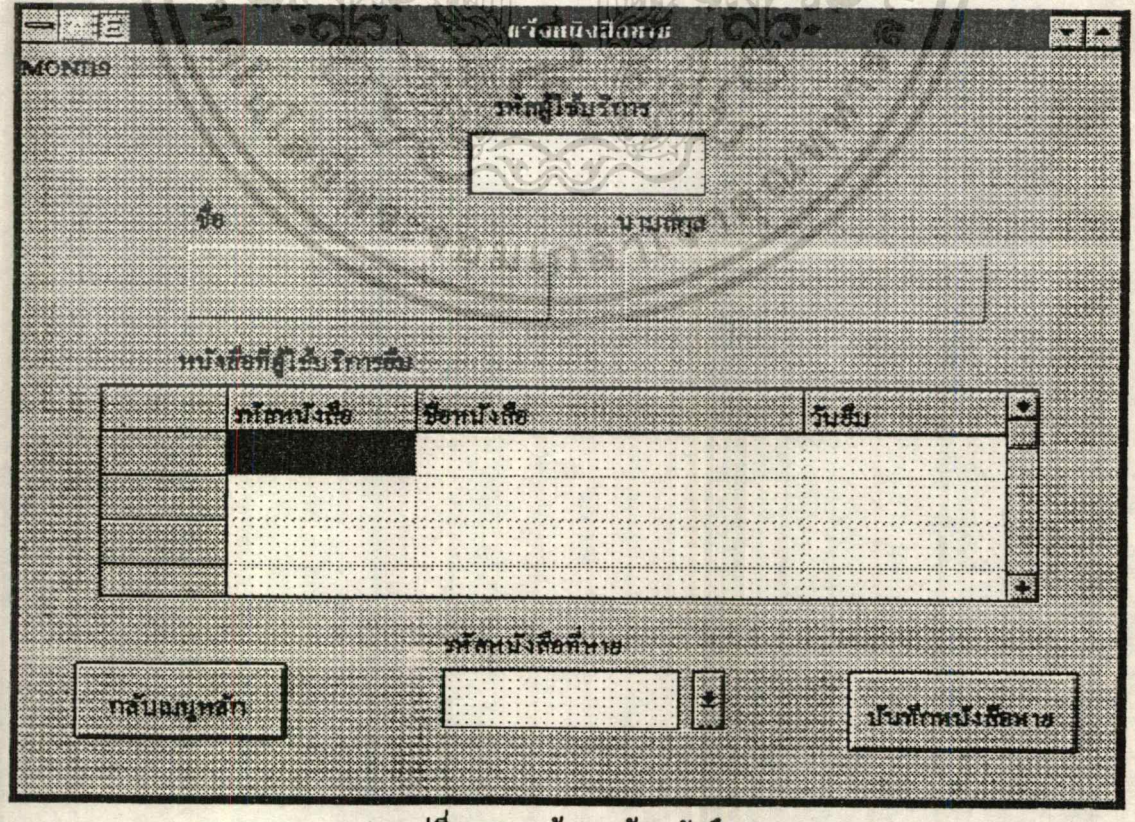
รูปที่ 4-46 หน้าจอเมนูหลักของงานบริการ



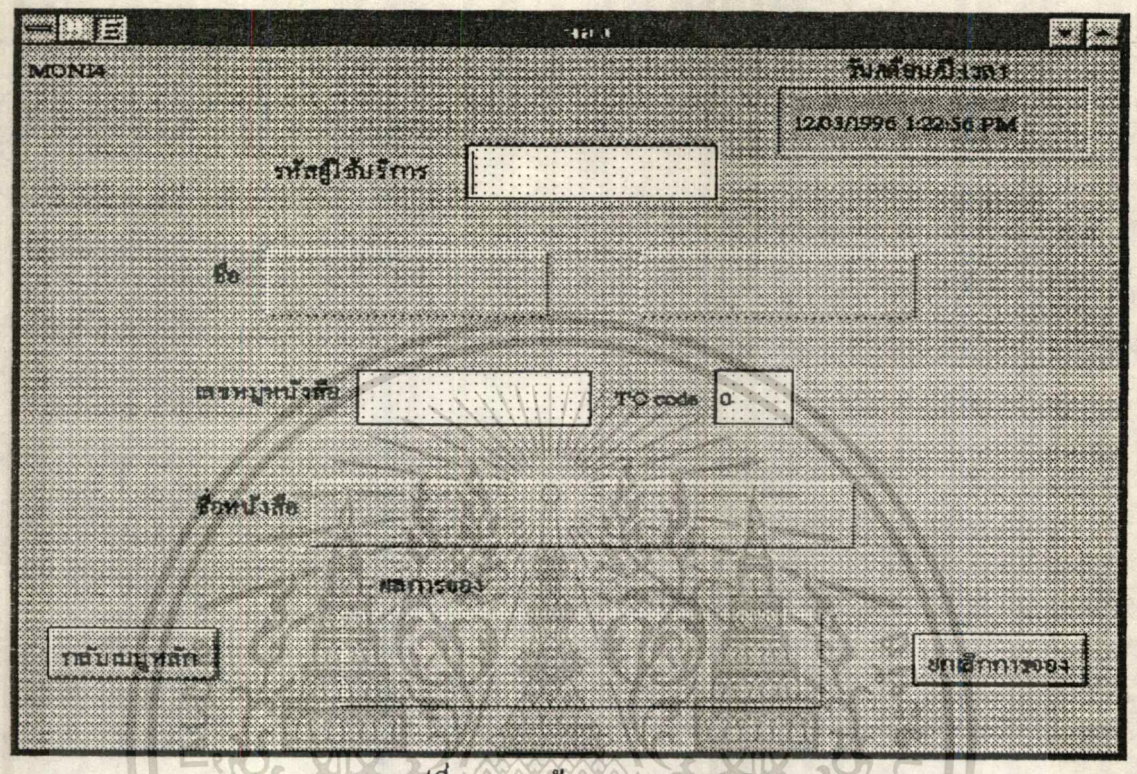
รูปที่ 4-47 หน้าจอการยืม



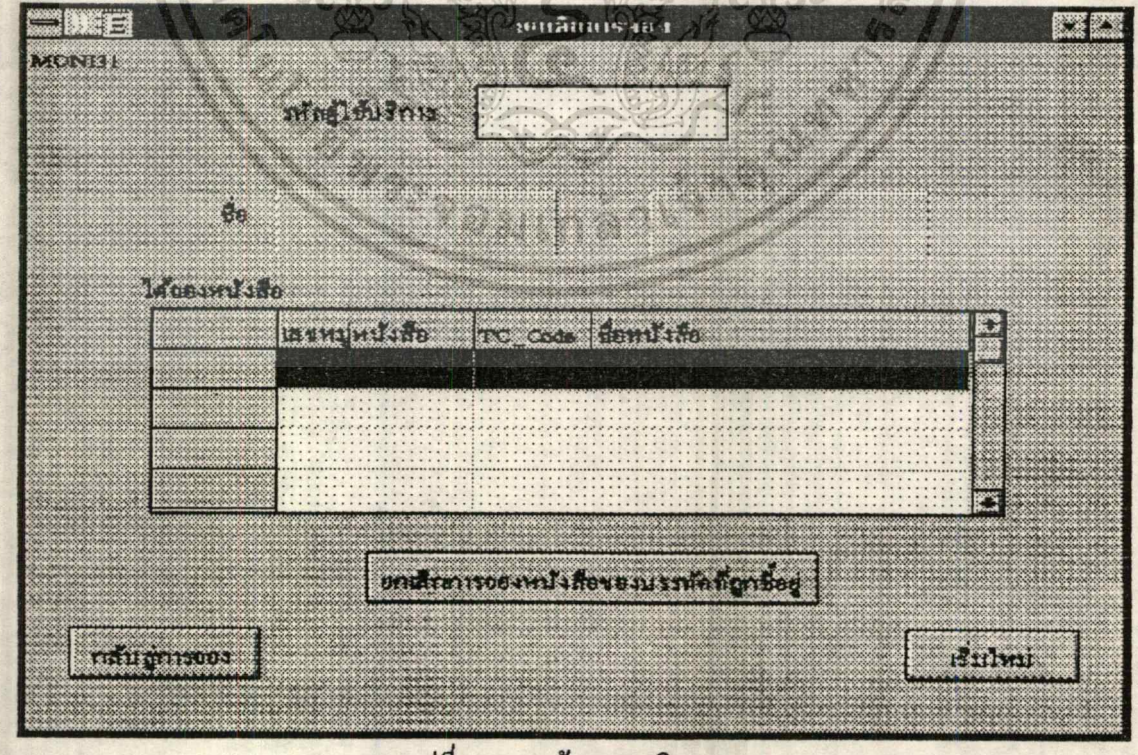
รูปที่ 4-48 หน้าจอการคืน



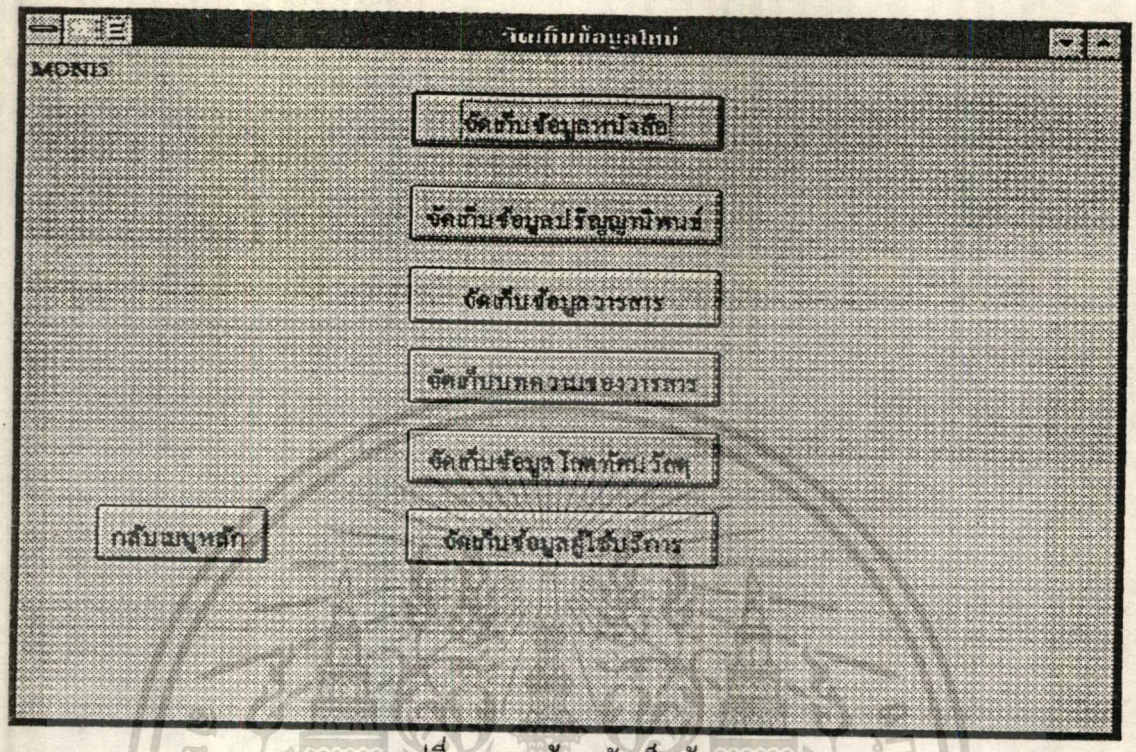
รูปที่ 4-49 หน้าจอแจ้งหนังสือหาย



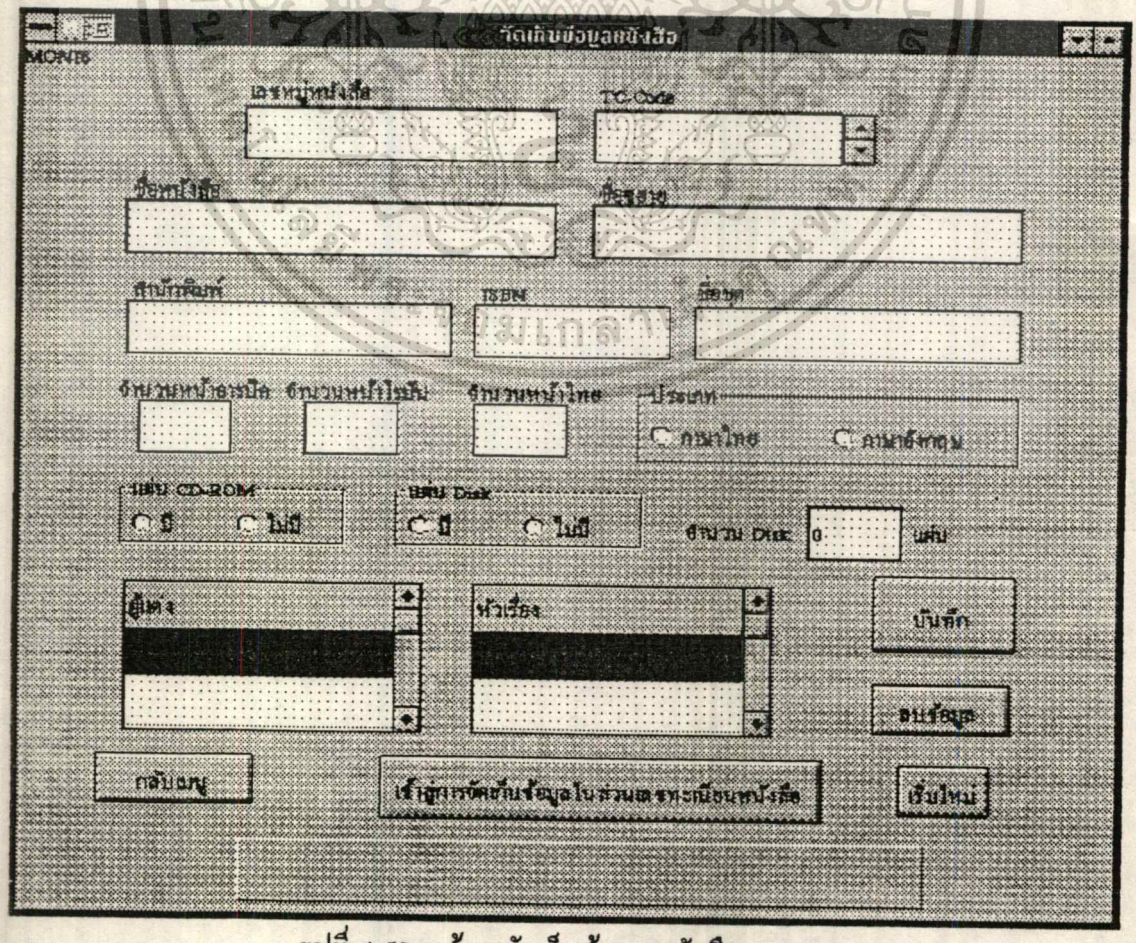
รูปที่ 4-50 หน้าจอการจอง



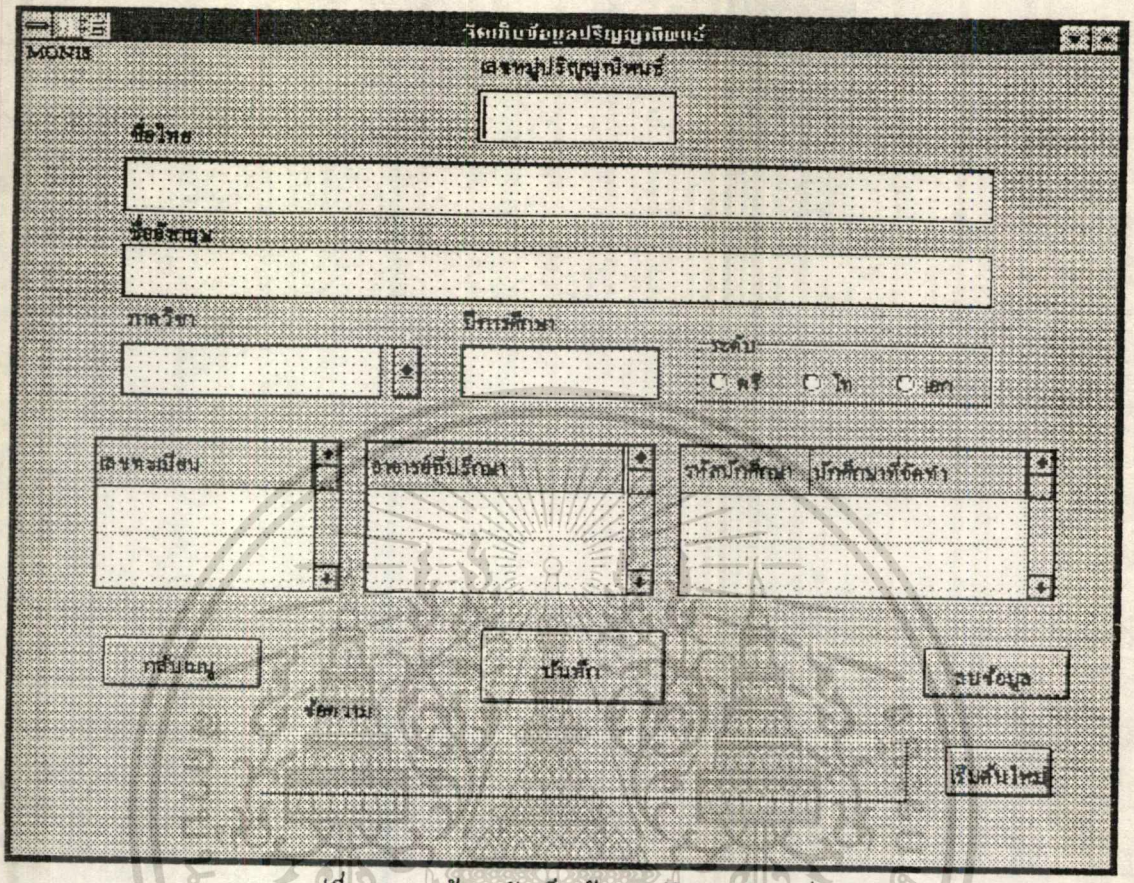
รูปที่ 4-51 หน้าจอยกเลิกการจอง



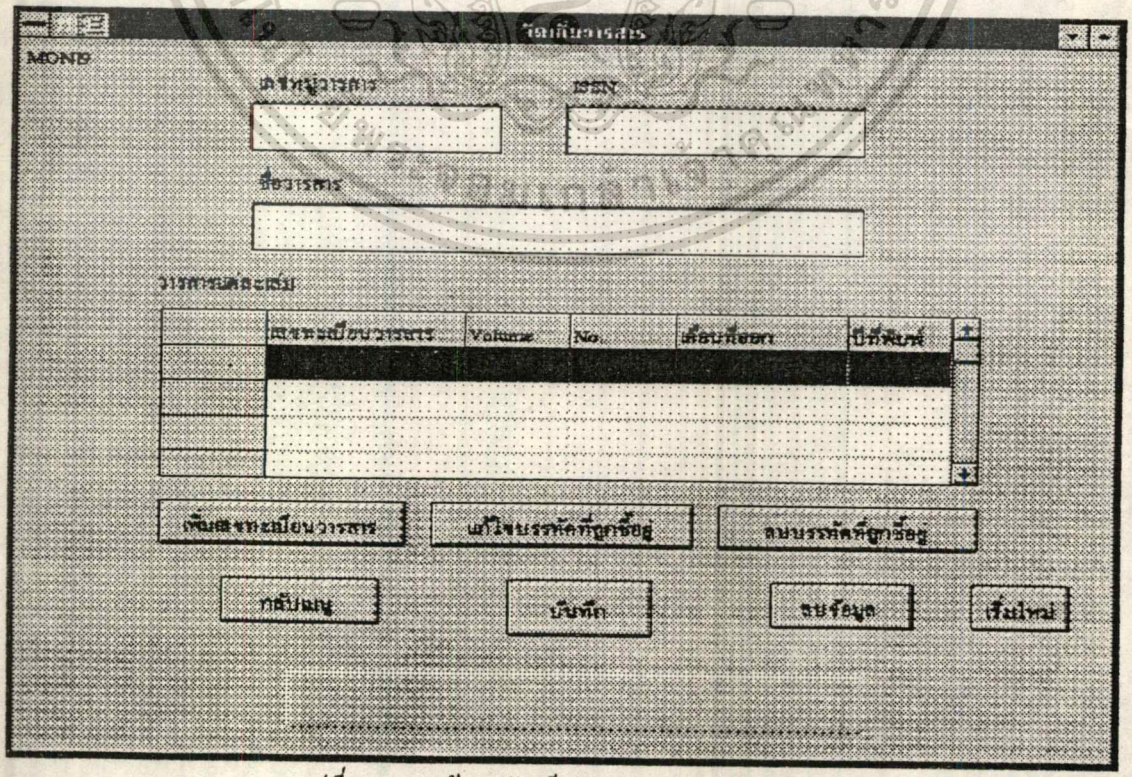
รูปที่ 4-52 หน้าจอจัดเก็บข้อมูล



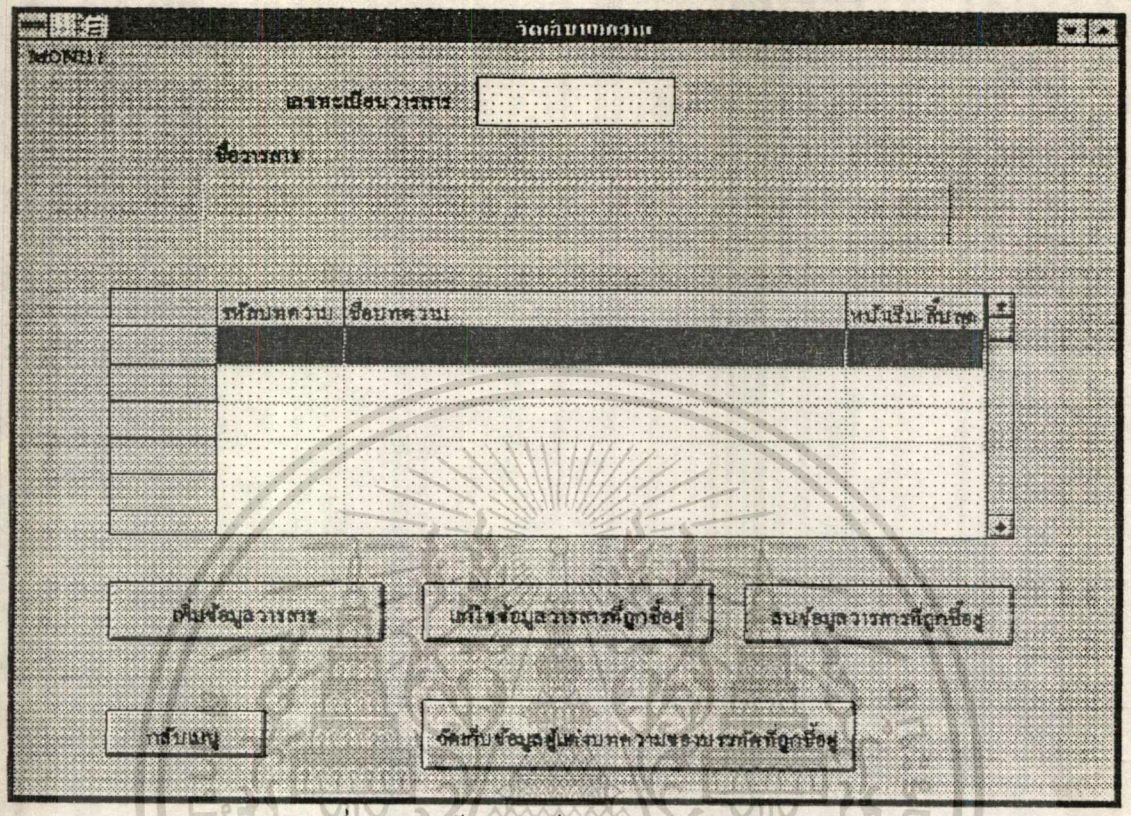
รูปที่ 4-53 หน้าจอจัดเก็บข้อมูลหนังสือ



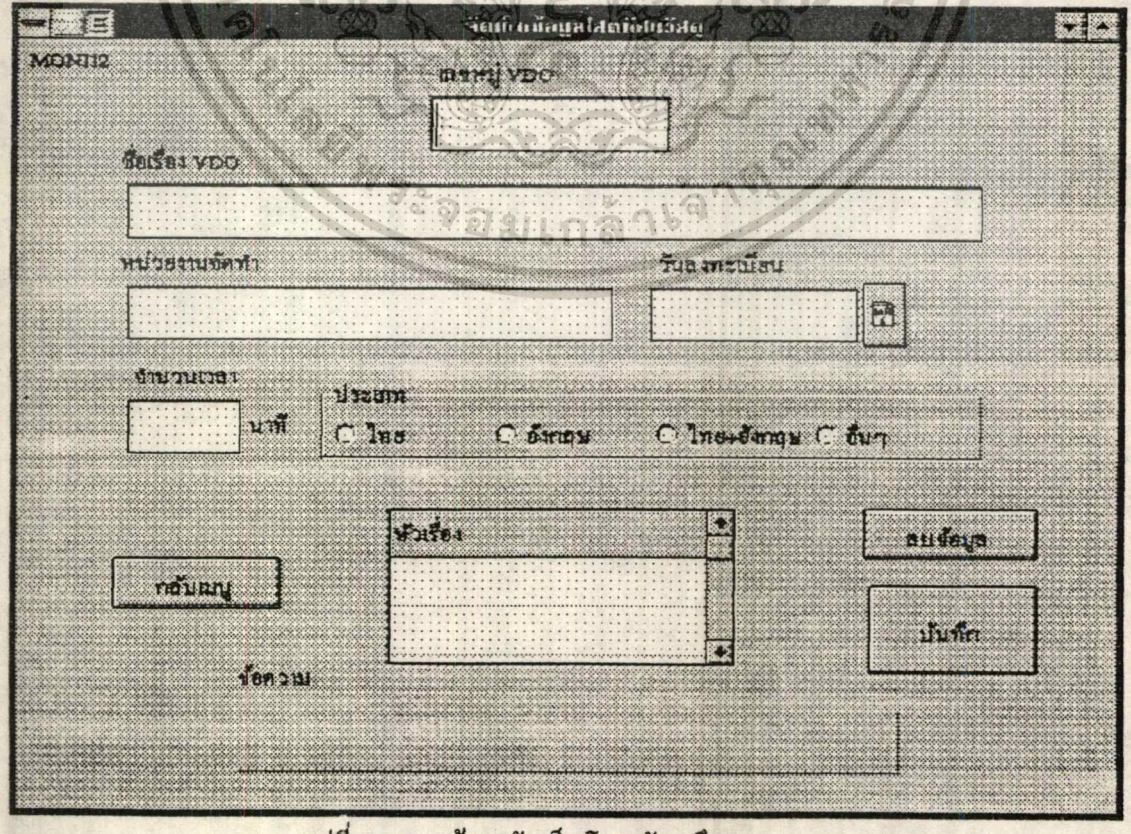
รูปที่ 4-54 หน้าจอจัดเก็บข้อมูลปริญญานิพนธ์



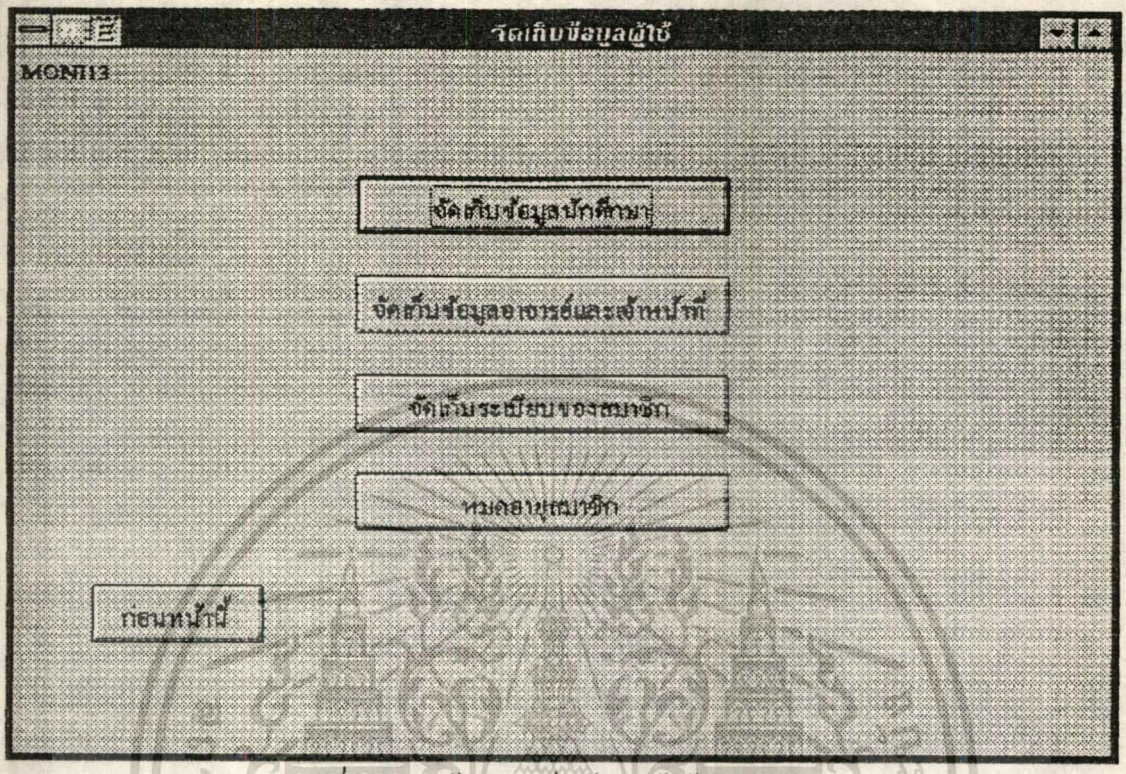
รูปที่ 4-55 หน้าจอจัดเก็บวารสาร



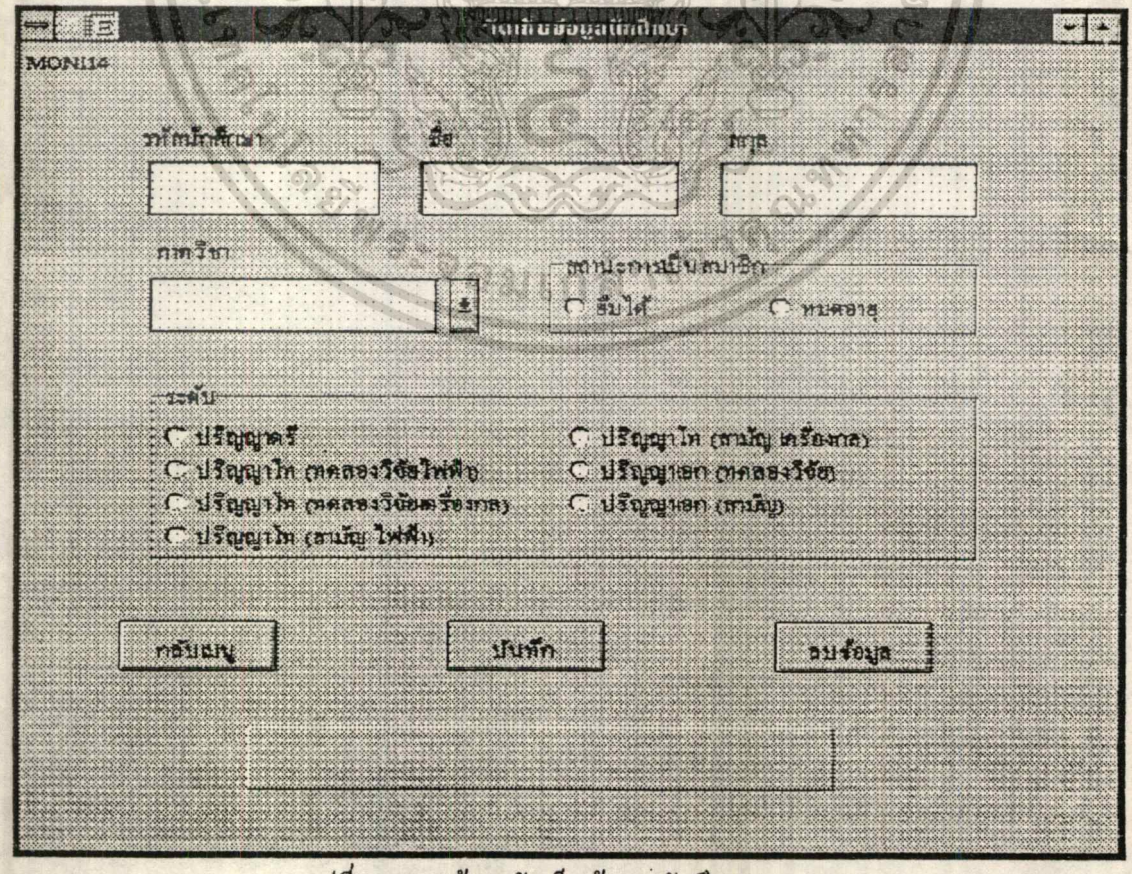
รูปที่ 4-56 หน้าจอจัดเก็บบทความ



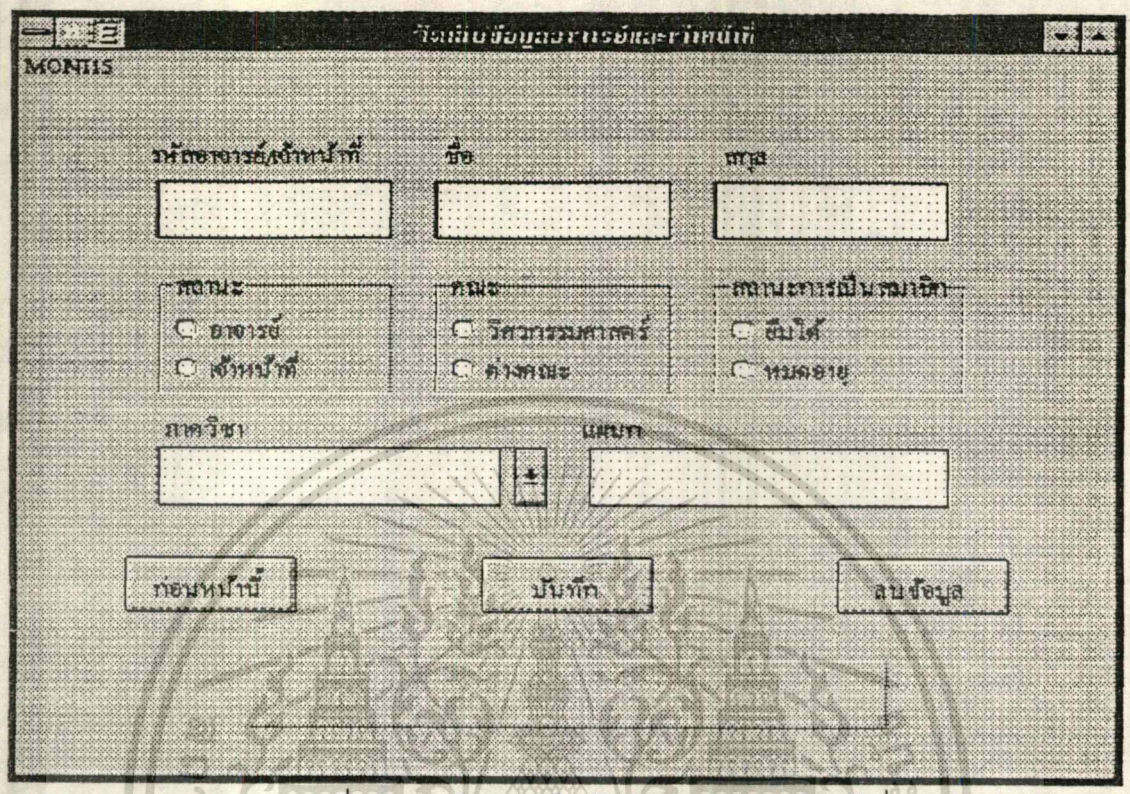
รูปที่ 4-57 หน้าจอจัดเก็บโสตทัศนศึกษา



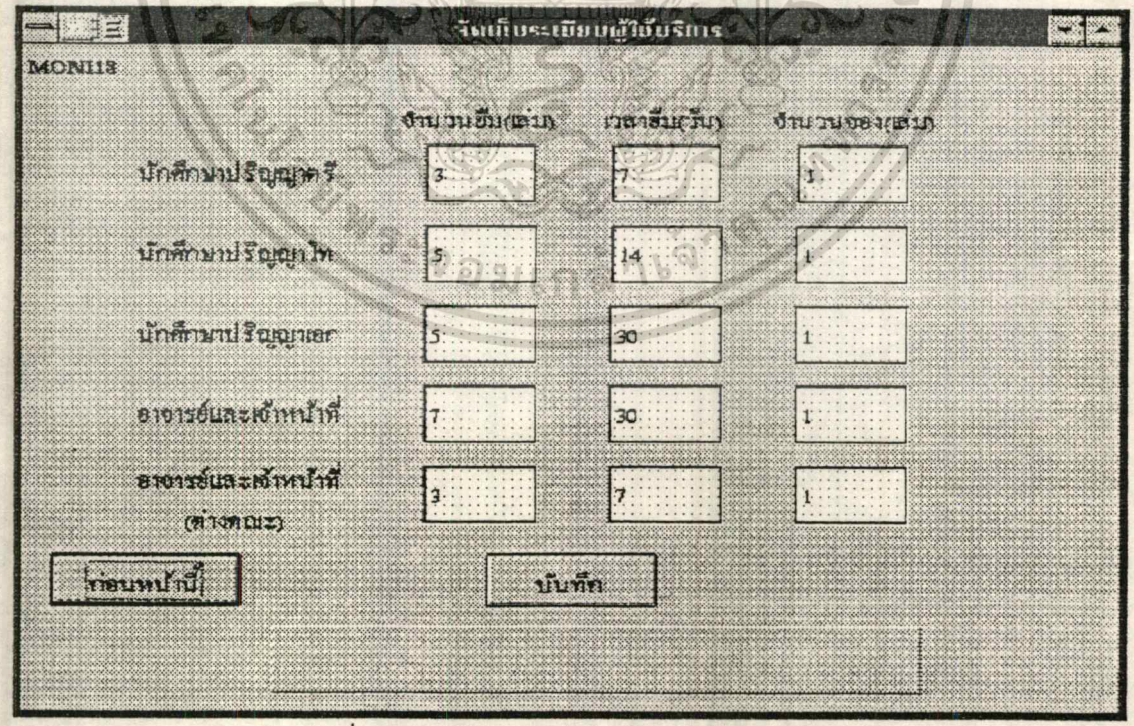
รูปที่ 4-58 หน้าจอจัดเก็บข้อมูลผู้ใช้



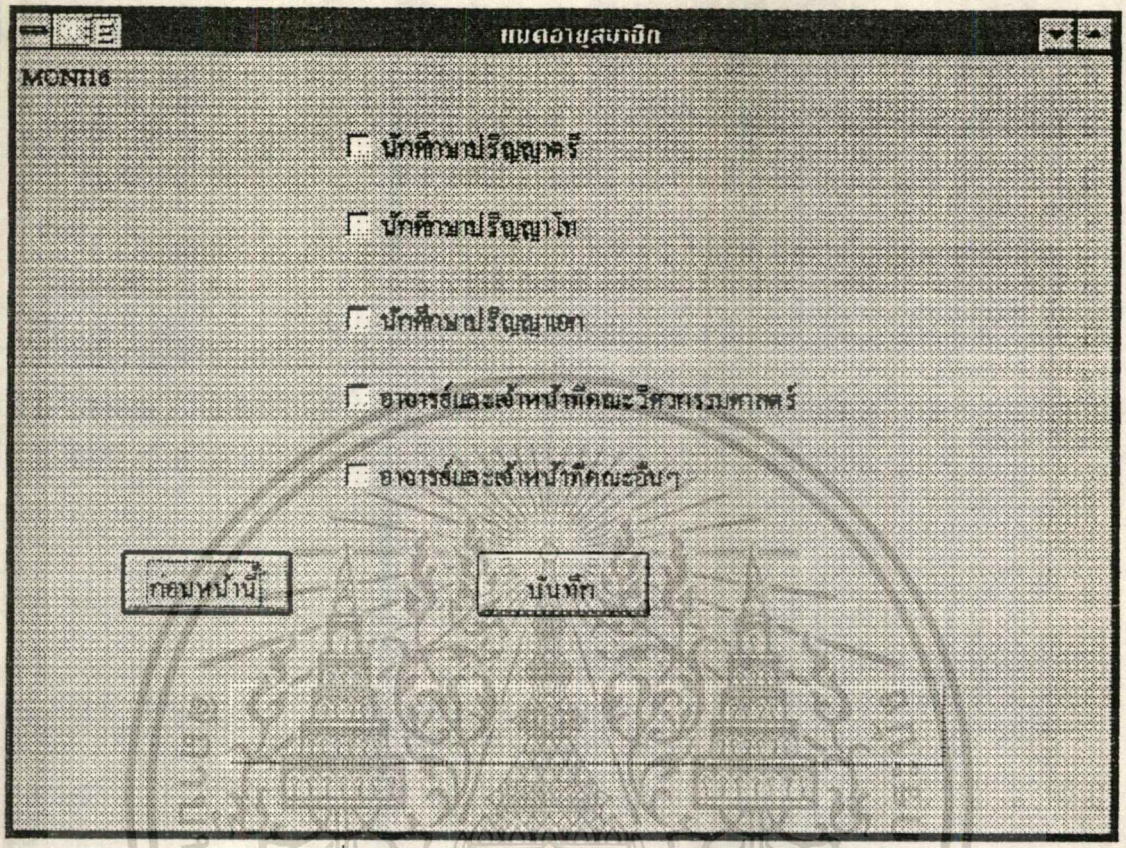
รูปที่ 4-59 หน้าจอจัดเก็บข้อมูลนักศึกษา



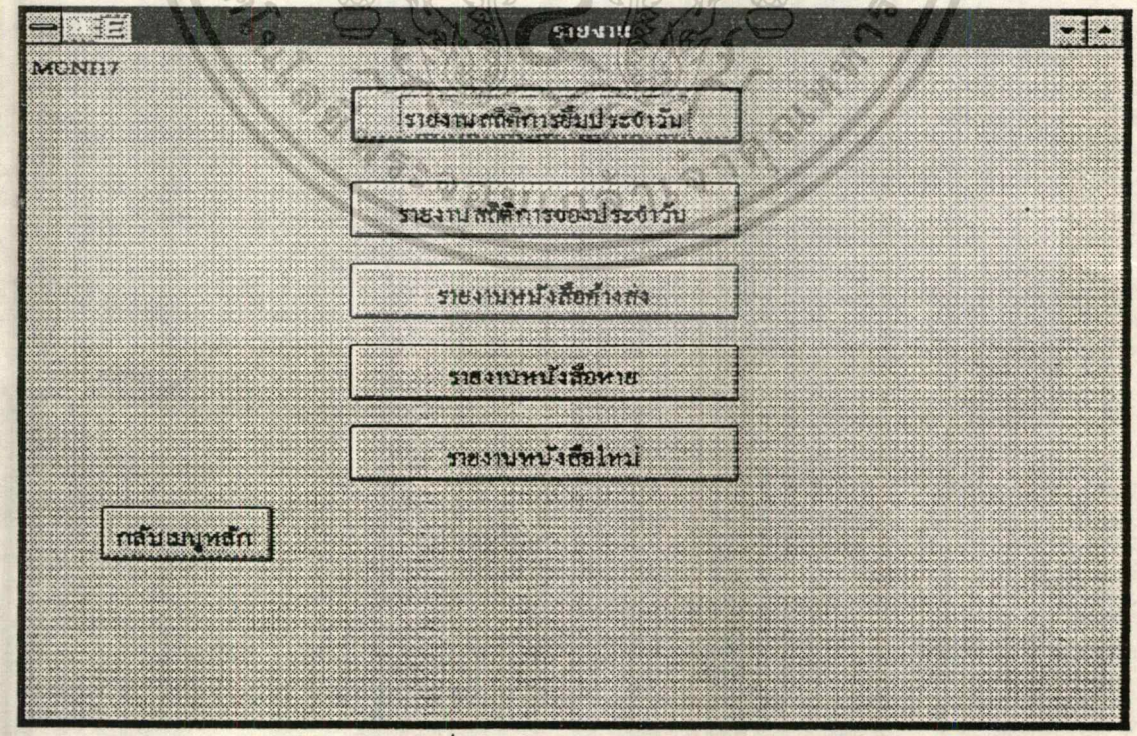
รูปที่ 4-60 หน้าจอจัดเก็บข้อมูลอาจารย์และเจ้าหน้าที่



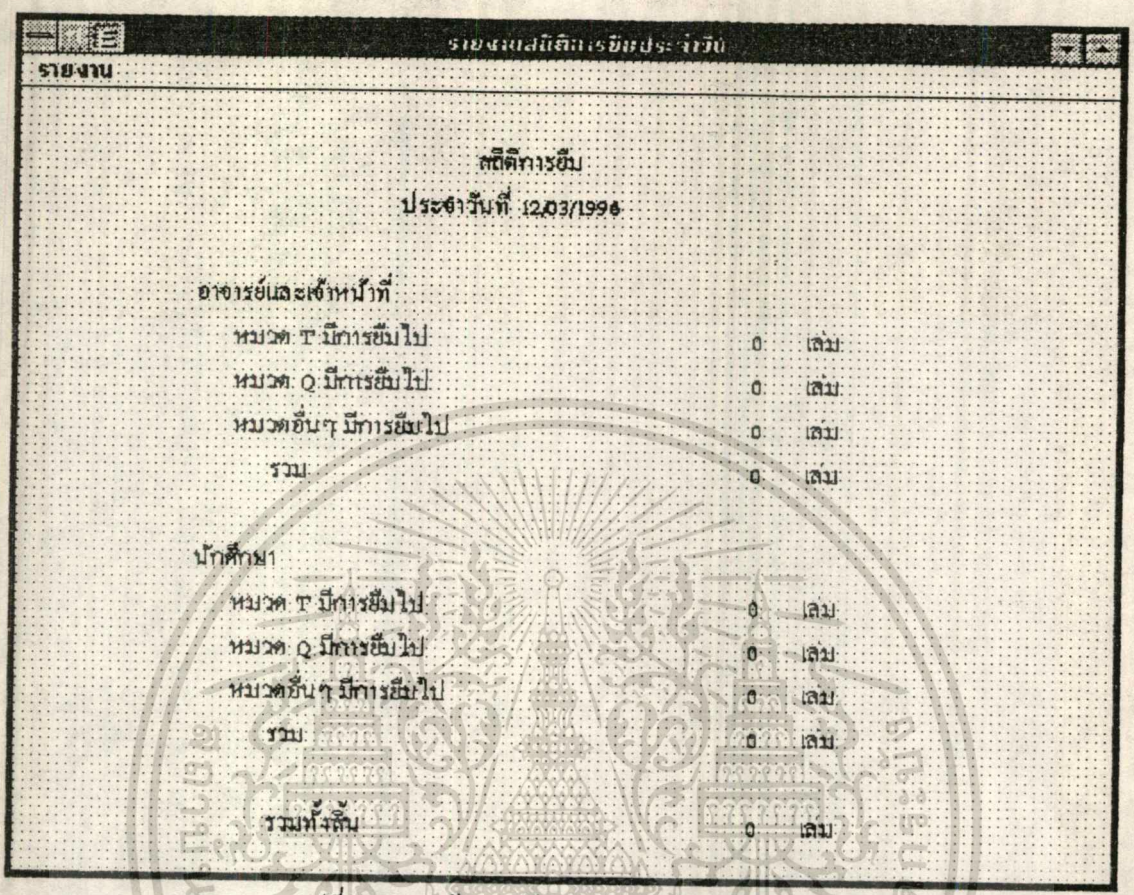
รูปที่ 4-61 หน้าจอจัดเก็บระเบียบสมาชิก



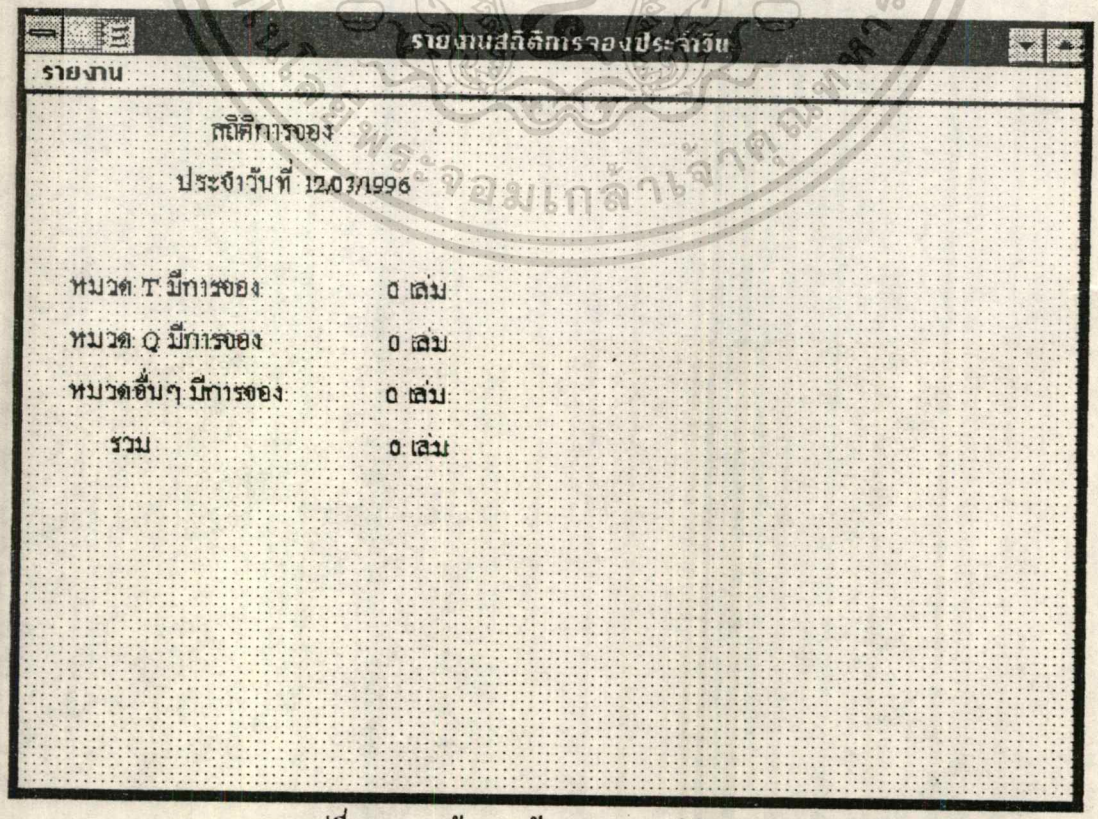
รูปที่ 4-62 หน้าจอจัดเก็บหมดอายุสมาชิก



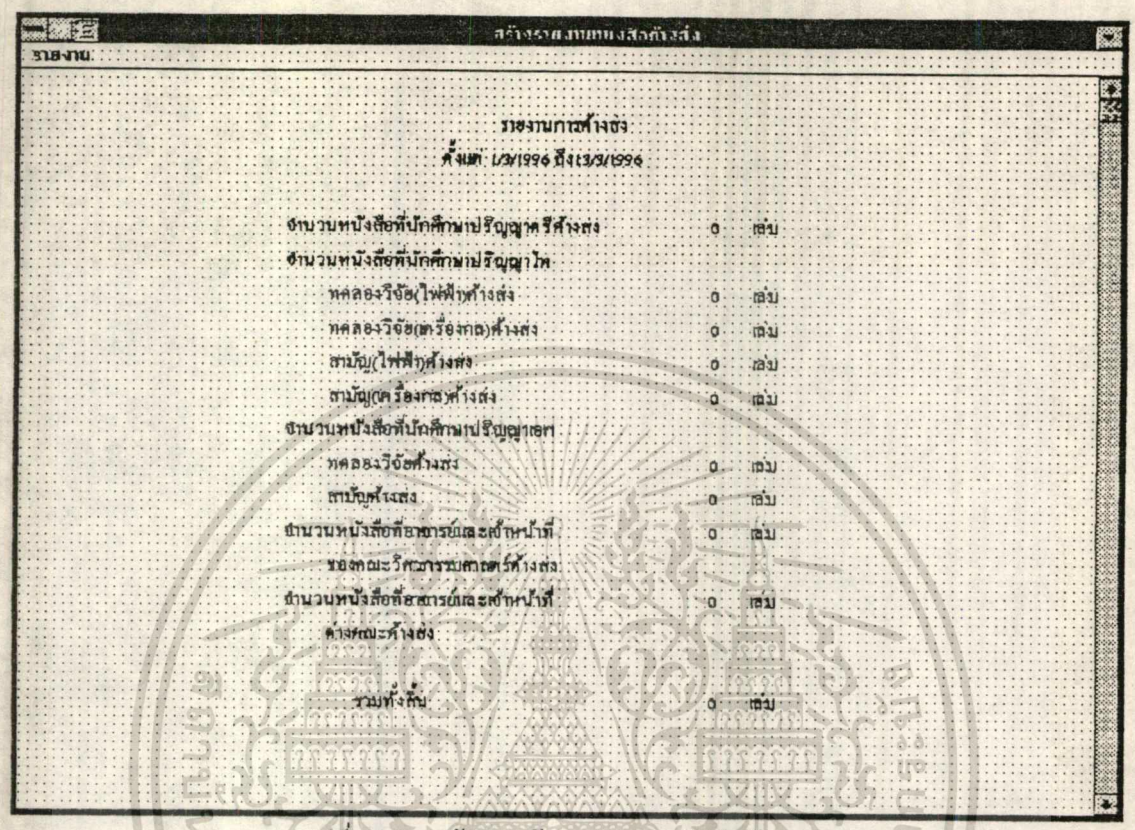
รูปที่ 4-63 หน้าจอรายงาน



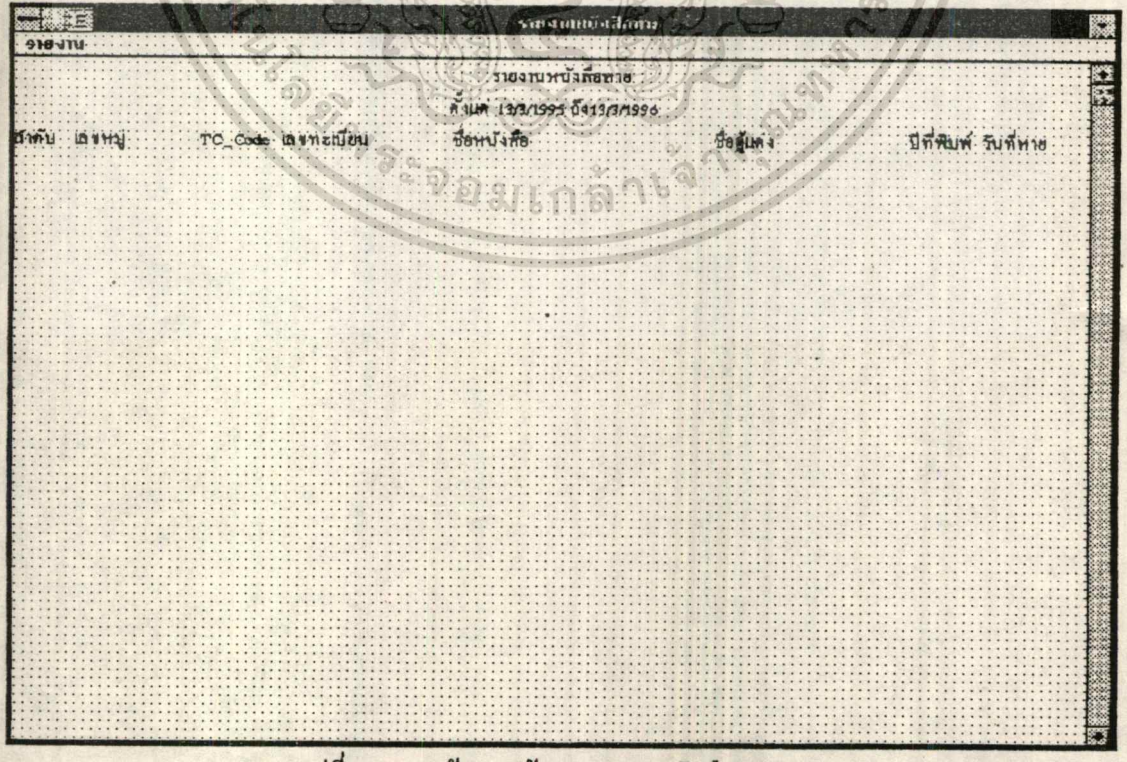
รูปที่ 4-64 หน้าจอสร้างรายงานสถิติการยืมประจำวัน



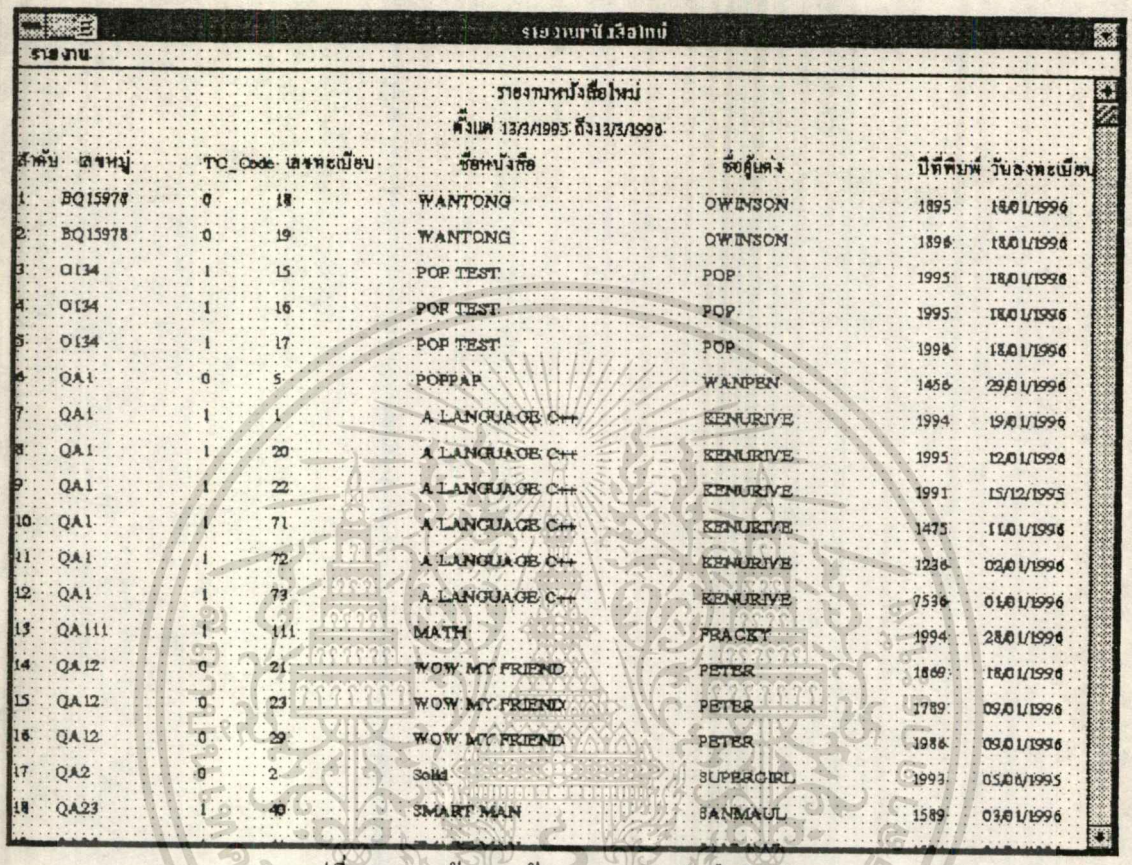
รูปที่ 4-65 หน้าจอสร้างรายงานสถิติการจองประจำวัน



รูปที่ 4-66 หน้าจอสร้างรายงานหนังสือค้างส่ง



รูปที่ 4-67 หน้าจอสร้างรายงานหนังสือหาย



รายงานหนังสือใหม่

รายงาน

รายงานหนังสือใหม่

ตั้งแต่ 13/3/1995 ถึง13/3/1996

ลำดับ	เลขหมู่	TC_Code	เลขทะเบียน	ชื่อหนังสือ	ชื่อผู้แต่ง	ปีที่พิมพ์	วันลงทะเบียน
1	BQ15978	0	18	WANTONG	OWINSON	1895	18/01/1996
2	BQ15978	0	19	WANTONG	OWINSON	1896	18/01/1996
3	Q134	1	15	POP TEST	POP	1995	18/01/1996
4	Q134	1	16	POP TEST	POP	1995	18/01/1996
5	Q134	1	17	POP TEST	POP	1996	18/01/1996
6	QA1	0	5	POPPAP	WANPEN	1456	29/01/1996
7	QA1	1	1	A LANGUAGE C++	KENURIVE	1994	19/01/1996
8	QA1	1	20	A LANGUAGE C++	KENURIVE	1995	12/01/1996
9	QA1	1	22	A LANGUAGE C++	KENURIVE	1991	15/12/1995
10	QA1	1	71	A LANGUAGE C++	KENURIVE	1475	11/01/1996
11	QA1	1	72	A LANGUAGE C++	KENURIVE	1236	02/01/1996
12	QA1	1	73	A LANGUAGE C++	KENURIVE	7536	01/01/1996
13	QA111	1	111	MATH	FRACKT	1994	28/01/1996
14	QA12	0	21	WOW MY FRIEND	PETER	1869	18/01/1996
15	QA12	0	23	WOW MY FRIEND	PETER	1789	09/01/1996
16	QA12	0	29	WOW MY FRIEND	PETER	1986	09/01/1996
17	QA2	0	2	Solid	SUPERGIRL	1993	05/06/1995
18	QA23	1	40	SMART MAN	SANMAUL	1589	03/01/1996

รูปที่ 4-68 หน้าจอสร้างรายงานหนังสือใหม่

4.5.1.2 การออกแบบรายงาน

ต่อไปนี้เป็นรายงานทั้งหมดที่แอพพลิเคชั่นจะสร้างให้กับเจ้าหน้าที่บรรณารักษ์

1. สถิติการยืม

สถิติการยืม

ประจำวันที่ _____

นักศึกษา

หมวด T มีการยืมไป _ เล่ม

หมวด Q มีการยืมไป _ เล่ม

หมวดอื่นๆ มีการยืมไป _ เล่ม

รวม _เล่ม

อาจารย์และเจ้าหน้าที่

หมวด T มีการยืมไป _ เล่ม

หมวด Q มีการยืมไป _ เล่ม

หมวดอื่นๆ มีการยืมไป _ เล่ม

รวม _เล่ม

รวมทั้งสิ้น _เล่ม

2. สถิติการจอง

สถิติการจอง

ประจำวันที่ _____

หมวด T มีการจอง _ เล่ม

หมวด Q มีการจอง _ เล่ม

หมวดอื่นๆ มีการจอง _ เล่ม

รวม _เล่ม

3. รายงานสถิติการค้างส่งการค้างส่ง

รายงานการค้างส่ง

ตั้งแต่วัน___ถึง___

จำนวนหนังสือที่นักศึกษาปริญญาตรีค้างส่ง _เล่ม

จำนวนหนังสือที่นักศึกษาปริญญาโท

ทดลองวิจัย(ไฟฟ้า)ค้างส่ง _เล่ม

ทดลองวิจัย(เครื่องกล)ค้างส่ง _เล่ม

สามัญ(ไฟฟ้า)ค้างส่ง _เล่ม

สามัญ(เครื่องกล)ค้างส่ง _เล่ม

จำนวนหนังสือที่นักศึกษาปริญญาเอก

ทดลองวิจัยค้างส่ง _เล่ม

สามัญค้างส่ง _เล่ม

จำนวนหนังสือที่อาจารย์และเจ้าหน้าที่ _เล่ม
ของคณะวิศวกรรมศาสตร์ค้างส่ง

จำนวนหนังสือที่อาจารย์และเจ้าหน้าที่ _เล่ม
ต่างคณะค้างส่ง

รวมทั้งสิ้น _เล่ม

4. รายงานแสดงรายชื่อผู้ค้างส่ง

รายชื่อ____ที่ค้างส่ง
ตั้งแต่วัน___ถึง___

รหัสนักศึกษา ชื่อนักศึกษา เลขหมู่หนังสือ เลขทะเบียนหนังสือ ชื่อหนังสือ วันยืม

5. ใบทวงถามหนังสือค้างส่งสำหรับอาจารย์และเจ้าหน้าที่

ตัวอย่างใบทวงถามหนังสือ

เรียน อาจารย์_____

ท่านได้ยืมหนังสือจากห้องสมุดคณะวิศวกรรมศาสตร์ และขณะนี้หนังสือที่ท่านยืมได้เลยกำหนดเวลาส่งแล้ว

รายชื่อหนังสือที่ท่านยืมได้แก่

เลขหมู่หนังสือ ชื่อหนังสือ วันที่ยืม

1.

2.

3.

จึงขอความกรุณานำหนังสือตามรายชื่อดังกล่างส่งคืนกลับห้องสมุดด้วย

จึงเรียนมาเพื่อทราบ

เจ้าหน้าที่ห้องสมุด

6. รายงานหนังสือหาย

รายงานหนังสือหาย

ประจำเดือน_____

เลขหมู่หนังสือ เลขทะเบียนหนังสือ ชื่อหนังสือ ปีพิมพ์ ชื่อผู้แต่งคนแรก วันที่หาย

7. รายงานหนังสือใหม่

รายงานหนังสือใหม่

ประจำเดือน_____

เลขหมู่หนังสือ เลขทะเบียนหนังสือ ชื่อหนังสือ ปีพิมพ์ ชื่อผู้แต่งคนแรก วันลงทะเบียน

4.5.2 งานค้นหา

การออกแบบหน้าจอ

ในการออกแบบหน้าจอของส่วนค้นหานั้น จากที่เคยกล่าวมาแล้ว จะสามารถแบ่งได้ 5 หัวข้อดังนี้

1. การค้นหาหนังสือ
2. การค้นหาวิทยานิพนธ์
3. การค้นหาวารสาร
4. การค้นหาโสตทัศนูปกรณ์ (V.D.O. เท่านั้น)
5. การค้นหาข้อมูลของผู้ใช้

เนื่องจากความคล้ายคลึงกันของระบบงาน จะกล่าวรวมหัวข้อที่ 1 และ 2 เข้าด้วยกัน

1) การค้นหาหนังสือ & การค้นหาวิทยานิพนธ์

การค้นหาหนังสือและการค้นหาวิทยานิพนธ์ จะพัฒนาจากระบบงานเดิมโดยได้มีการปรับปรุงระบบให้ดีขึ้นตามความต้องการของผู้ใช้ และความเหมาะสม โดยปรับเปลี่ยนส่วนต่างๆดังนี้

1. ปรับเปลี่ยนระบบงานเดิมที่เป็น Text base ให้เป็น GUI

2. ปรับปรุงคีย์ที่ใช้ในการค้นหา จากข้อมูลของระบบงานเดิมที่ใช้ในปัจจุบันสามารถทราบถึงการใช้คีย์ในการค้นหาข้อมูลของผู้ใช้ ว่ามักใช้คีย์ใดในการค้นหา หรือไม่ใช้คีย์ใด จึงมีการปรับเปลี่ยนให้เหมาะสมและมีการเพิ่มคีย์ ที่คิดว่าจะเป็นประโยชน์ในการค้นหาเพิ่มเข้าไปด้วย

เนื่องจากระบบงานเดิมสามารถค้นหาโดยใช้คีย์เดียว ก็ปรับปรุงให้สามารถใช้ หลายคีย์ ในการค้นหาได้ และสามารถกรองข้อมูลให้เป็นข้อมูลที่ผู้ใช้ต้องการมากที่สุด ทำให้ข้อมูลมีคุณภาพมากขึ้น เช่น ถ้า ผู้ใช้ ทราบข้อมูลของชื่อหนังสือกับชื่อผู้แต่ง ก็สามารถป้อน ทั้ง 2 อย่างเข้าไป เพื่อช่วยในการค้นหาให้เร็วและข้อมูลที่ได้ตรงกับที่ผู้ใช้ต้องการ จากเดิมที่จะต้องเลือกว่าจะค้นหาตามคีย์อะไรอย่างใดอย่างหนึ่งแล้วป้อนข้อมูลนั้นลงไป เป็นต้น

สามารถสรุป คีย์ ที่ใช้ในการค้นหาได้ดังนี้

หนังสือ : -ชื่อหนังสือ -ชื่อหัวเรื่อง -ชื่อผู้แต่ง
-ปีที่พิมพ์ -เลขหมู่

ปรัชญานิพนธ์ : -เลขหมู่ -ชื่อเรื่อง -ชื่ออาจารย์
-นักศึกษา , รหัสนักศึกษา -ปีการศึกษา -ภาควิชา

3.ปรับปรุง เอาท์พุท ให้เหมาะสม โดยจากระบบงานเดิม เมื่อค้นหาได้แล้วจะแสดงข้อมูลทั้งหมดที่เก็บไว้ซึ่งข้อมูลบางอย่างผู้ใช้ไม่ได้สนใจ เพราะไม่มีประโยชน์กับตัวผู้ใช้เลย นอกจากจะเป็นประโยชน์กับเจ้าหน้าที่ห้องสมุดเท่านั้น จึงมีการคัดเลือก แอททริบิวท์ ที่คิดว่ามีประโยชน์กับผู้ใช้ทั่วไป ซึ่งผู้ใช้สามารถนำไปค้นหาตัวหนังสือต่อได้เท่านั้นมาแสดง จะช่วยให้การค้นหาของผู้ใช้เร็วขึ้นและสามารถแสดงข้อมูลต่อ 1 หน้าจอได้มากขึ้นด้วย โดยแบ่งได้ดังนี้

<u>หนังสือ</u> จะแสดง แอททริบิวท์ ดังนี้

เลขหมู่	TC_CODE	ชื่อหนังสือ	สำนักพิมพ์	หนังสือทั้งหมดที่มีอยู่	หนังสือที่อยู่ในห้องสมุด

จะเห็นว่า มีการบอกสถานะหนังสือด้วย เป็นการช่วยผู้ใช้ไม่ต้องไปหาหนังสือไม่ได้อยู่ในห้องสมุด

<u>ปริญญานิพนธ์</u> จะแสดง แอททริบิวท์ ดังนี้

เลขหมู่ปริญญานิพนธ์	ชื่อเรื่อง	ภาควิชา	หลักสูตร	ปีการศึกษา

แต่ถ้าผู้ใช้ต้องการทราบรายละเอียดหนังสือทั้งหมด ก็สามารถทำได้ด้วยการใช้เลขหมู่ไปหารายละเอียดที่ต้องการ

2.) การค้นหาวารสาร

เป็นการค้นหาที่เพิ่มเติมขึ้นมาใหม่ โดยสามารถค้นหาตาม คีย์ ได้ดังนี้

-เลขทะเบียนวารสาร

-ชื่อวารสาร , ฉบับที่ , ปีที่พิมพ์ , เลขหมู่วารสาร

-ช่วงการออกวารสารโดยสามารถเลือกได้เป็น รายเดือน , รายสัปดาห์ , รายปักษ์

-ชื่อผู้เขียน

-ชื่อบทความ

-รหัสบทคาม

ในหัวข้อนี้ ถ้าผู้ใช้รู้ข้อมูลอะไรสามารถเลือกใส่ได้และจะนำข้อมูลไปค้นหา โดยแบ่งแอททริบิวท์ ที่ใช้แสดงได้ดังนี้

2.1 ข้อมูลวารสารแต่ละเล่ม แบ่งเป็น 2 ส่วน

-ข้อมูลเกี่ยวกับวารสารนั้น

เลขหมู่	ชื่อวารสาร	ISSN	เลขทะเบียนวารสาร	ฉบับที่	.ปีที่	ช่วงการออก

-ข้อมูลเกี่ยวกับบทความในวารสาร

รหัสบทความ	ชื่อบทความ	ชื่อผู้แต่ง	หน้าเริ่มต้น - สิ้นสุด

2.2 ข้อมูลบทความ จะแสดงข้อมูลให้ผู้ใช้หาบทความที่ต้องการ

รหัสบทความ	ชื่อบทความ	ชื่อวารสาร	ฉบับที่	.ปีที่	ช่วงการออก	ผู้จัดทำ

3) การค้นหาโสตทัศนูปกรณ์ (เฉพาะ V.D.O.)

จากการตรวจสอบความต้องการและดูจากตารางที่ได้สร้างไว้สามารถค้นหาตาม คีย์ ได้ดังนี้

-ประเภท (ไทย / อังกฤษ) เนื่องจาก V.D.O. จะมี 2 ภาษาให้เลือก

-ชื่อเรื่อง

-หัวเรื่อง

-เลขหมู่

เมื่อผู้ใช้ป้อนข้อมูลลงไปจะแสดงผลดังนี้

เลขหมู่โสต	ชื่อเรื่อง	หน่วยงานที่จัดทำ	ความยาว	ภาษา	วันรับโสต

4) การค้นหาข้อมูลของผู้ใช้

ผู้ใช้แบ่งเป็น 3 ประเภท

1. นักศึกษา
2. อาจารย์ - เจ้าหน้าที่ในคณะ
3. อาจารย์ - เจ้าหน้าที่ต่างคณะ

ในการใช้บริการผู้ใช้ต้องป้อนรหัสของตนเองลงไปเพื่อค้นหาข้อมูลต่างๆ ซึ่งแบ่งเป็น 2 ส่วน

4.1 ข้อมูลเกี่ยวกับการยืม - คืนหนังสือ

ในส่วนนี้จะให้ผู้ใช้บริการสอบถามข้อมูลของผู้ใช้เองหรือบุคคลอื่น ถ้ารู้รหัสในขั้นตอนนี้จะเป็นการสอบถามว่ายืมหนังสืออะไรไปบ้าง ถึงกำหนดส่งหรือยัง ถ้าเกินกำหนดแล้วก็จะคำนวณค่าปรับให้

โดยจะแสดงข้อมูลดังนี้

เลขหมู่	TC_CODE	เลขทะเบียน	ชื่อเรื่อง	วันที่ยืม	วันที่คืน

และสามารถนำข้อมูลแสดงไปค้นหารายละเอียดหนังสือเล่มนั้นได้อีกด้วย

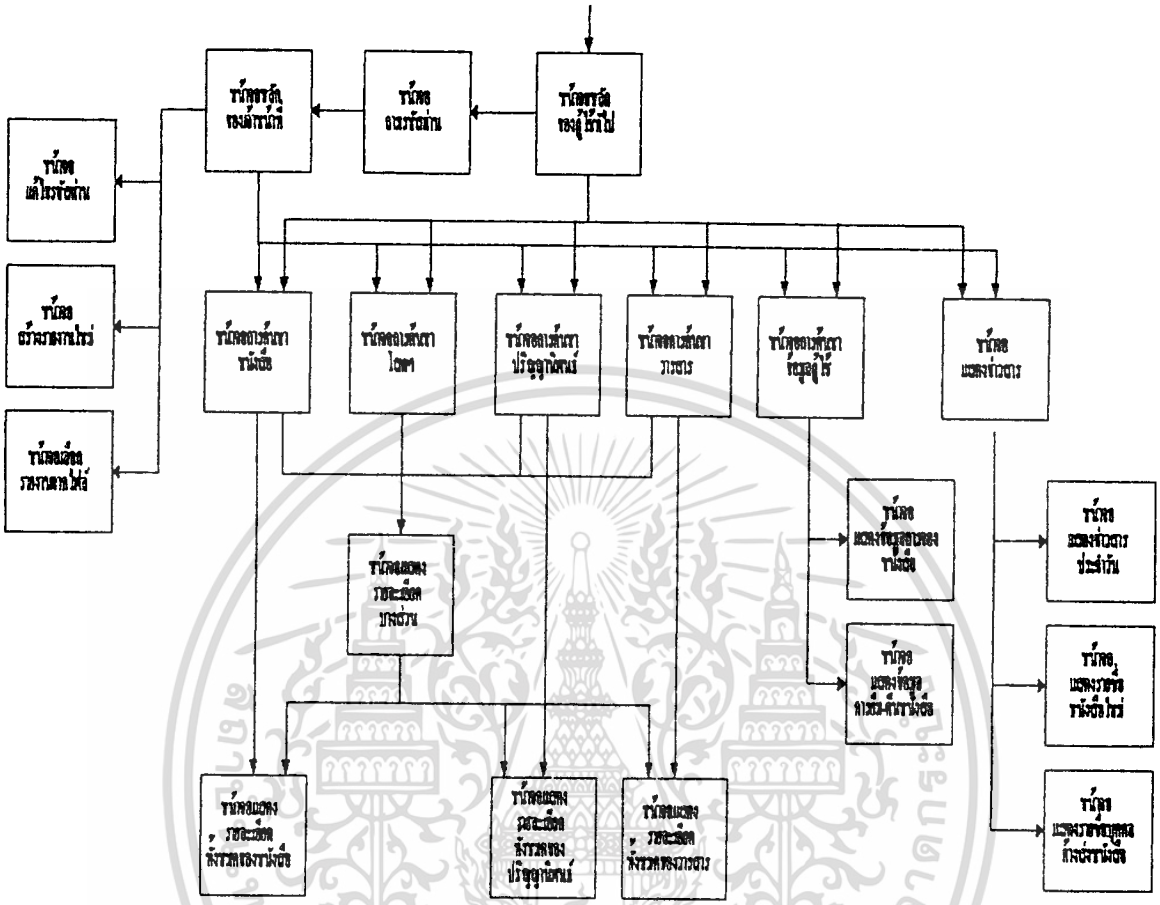
4.2 การค้นหาข้อมูลเกี่ยวกับการจอง

ในส่วนนี้ผู้ใช้บริการจะใช้ค้นหาข้อมูลของตนเองว่าจองอะไรไปบ้างและจะได้รับหนังสือที่จองเมื่อไร เพราะปกติผู้ใช้ต้องไปสอบถามเจ้าหน้าที่แต่ด้วยบริการนี้สามารถค้นหาได้เอง

โดยจะแสดงข้อมูลดังนี้

เลขหมู่	TC_CODE	เลขทะเบียน	ชื่อเรื่อง	วันที่รอรับ	วันหมดกำหนด

รูปดังที่จะแสดงต่อไปนี้เป็นโครงสร้างของหน้าจอที่ออกแบบไว้ทั้งหมด

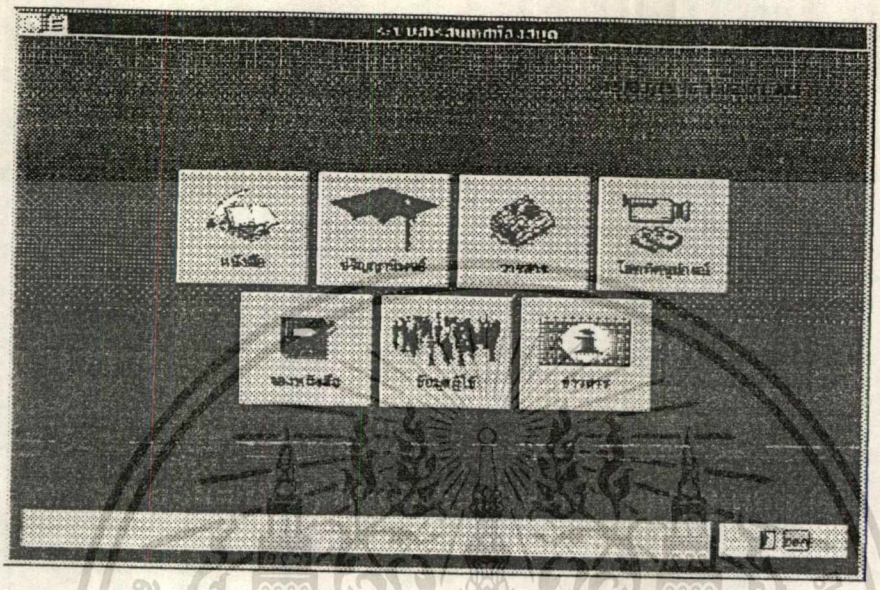


รูปที่ 4-69 โครงสร้างของหน้าจอการค้นหา

ต่อจากนี้จะเป็นหน้าจอต่างๆของส่วนการค้นหา ที่ออกแบบไว้ก่อนที่จะเขียนโปรแกรมต่อไป

1. เมนูหลัก - จะมี 2 หน้าจอ คือ

1.1 เมนูหลักสำหรับผู้ใช้ทั่วไป - ใช้ในการค้นหาข้อมูลที่ต้องการ



รูปที่ 4-70 หน้าจอเมนูหลักสำหรับผู้ใช้ทั่วไป

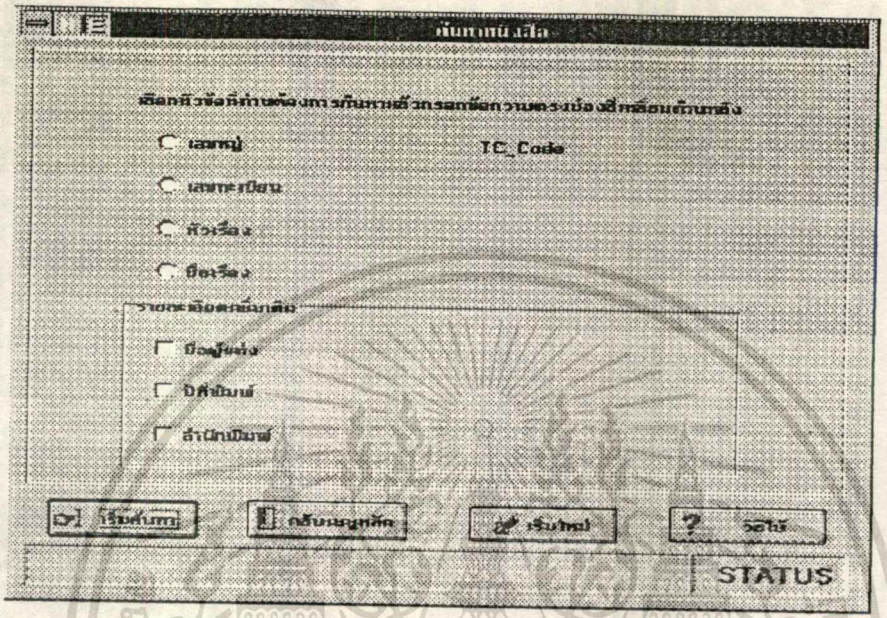
1.2 เมนูหลักสำหรับเจ้าหน้าที่ - เพิ่มเติมส่วนสร้างรายงานใหม่ และ ส่วนแก้ไขรหัสผ่าน



รูปที่ 4-71 หน้าจอเมนูหลักสำหรับเจ้าหน้าที่

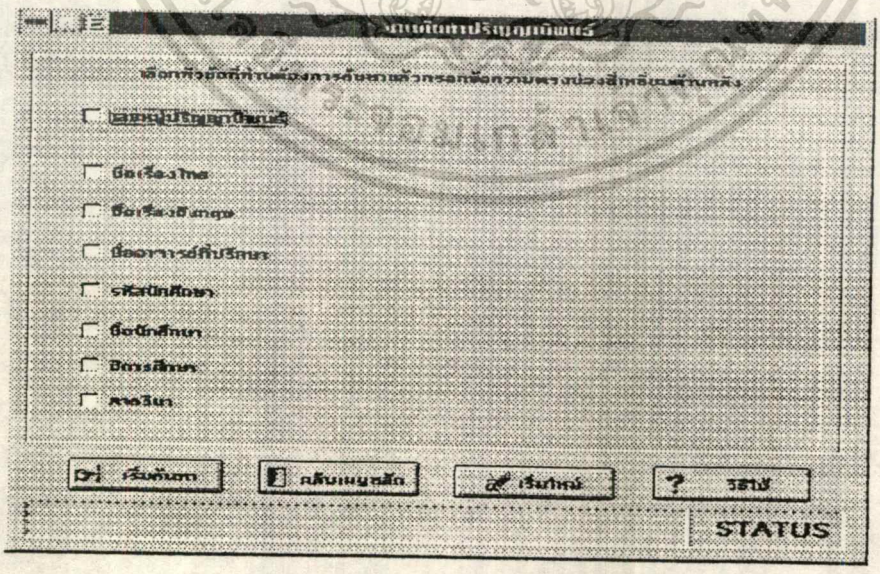
2.หน้าจอสำหรับค้นหา - มี 4 หน้าจอ

2.1 หน้าจอค้นหาหนังสือ



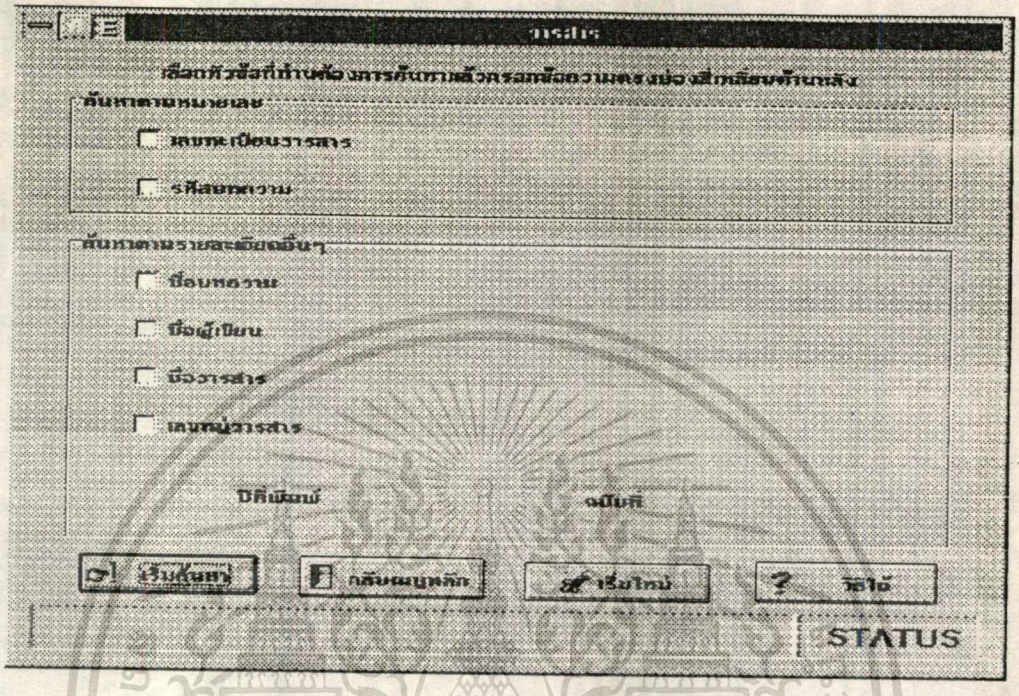
รูปที่ 4-72 หน้าจอค้นหาหนังสือ

2.2 หน้าจอค้นหาปริญญานิพนธ์



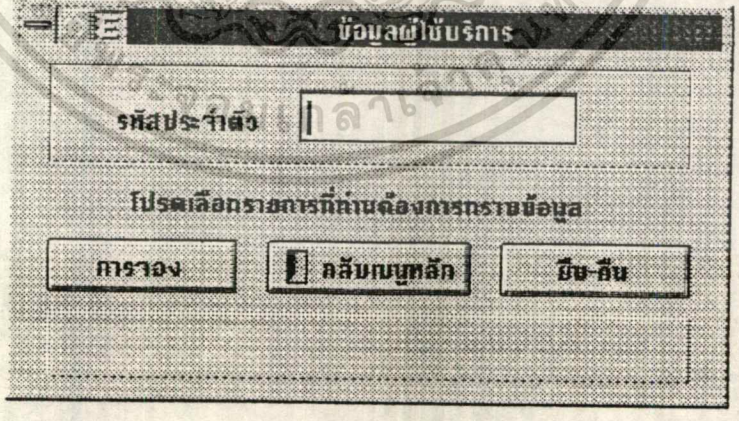
รูปที่ 4-73 หน้าจอค้นหาปริญญานิพนธ์

2.3 หน้าจอค้นหาโสตทัศนูปกรณ์



รูปที่ 4-74 หน้าจอค้นหาโสตทัศนูปกรณ์

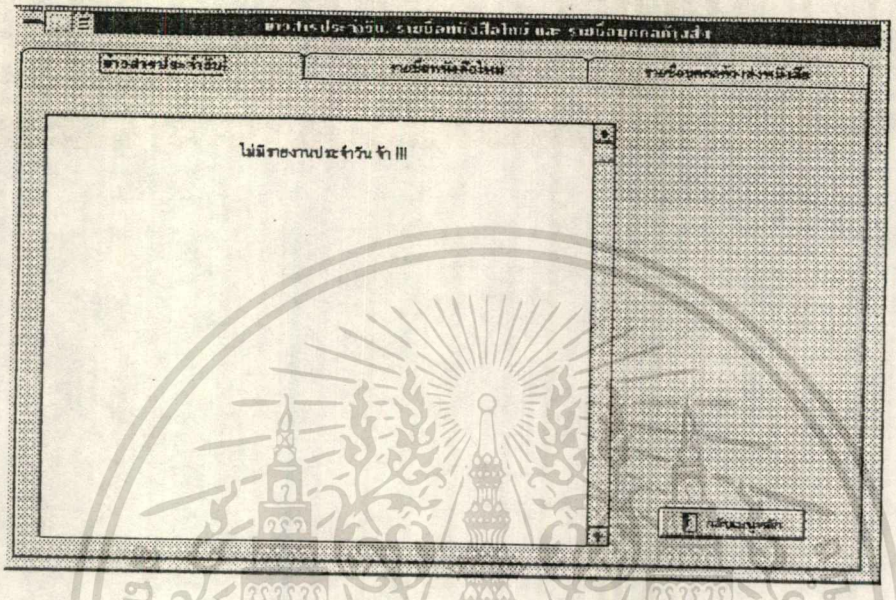
2.4 หน้าจอค้นหาข้อมูลผู้ใช้



รูปที่ 4-75 หน้าจอค้นหาข้อมูลผู้ใช้

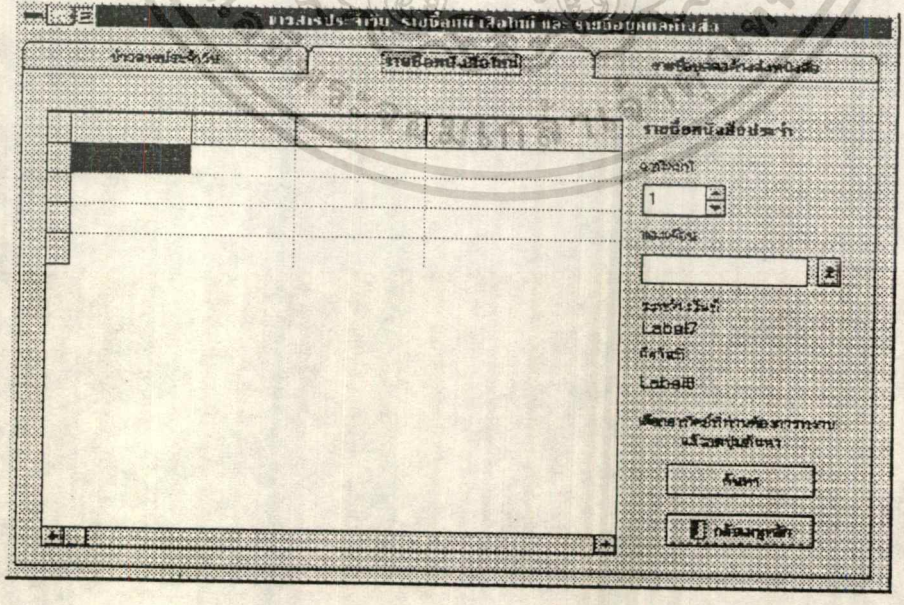
3. ส่วนแสดงข่าวสาร - มีการแสดงข่าวสารจากทางห้องสมุดให้ผู้ใช้ทราบ 3 เรื่อง

3.1 แสดงข่าวสารประจำวัน



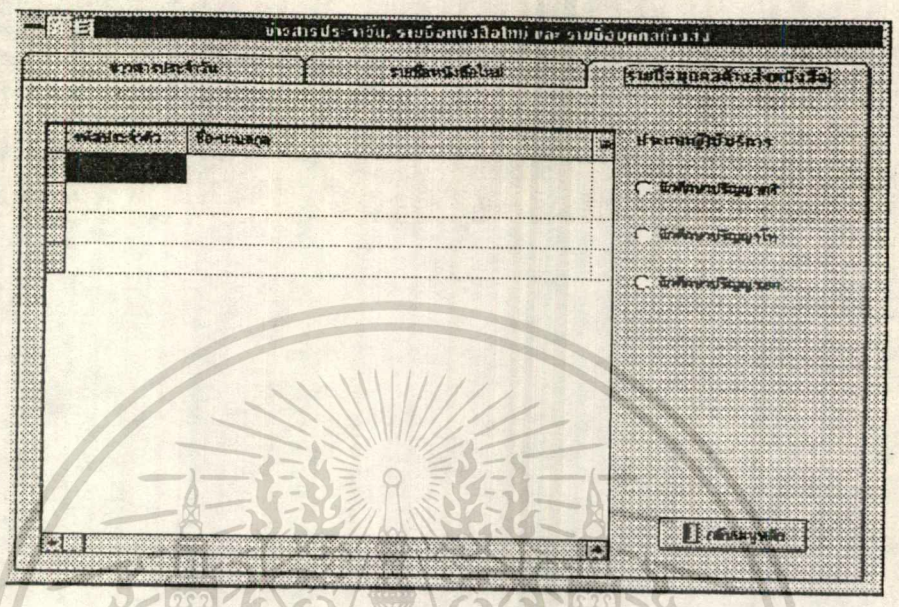
รูปที่ 4-76 หน้าจอแสดงข่าวสารประจำวัน

3.2 แสดงรายชื่อหนังสือใหม่ - สามารถค้นหาหนังสือใหม่เป็นช่วงสัปดาห์ได้



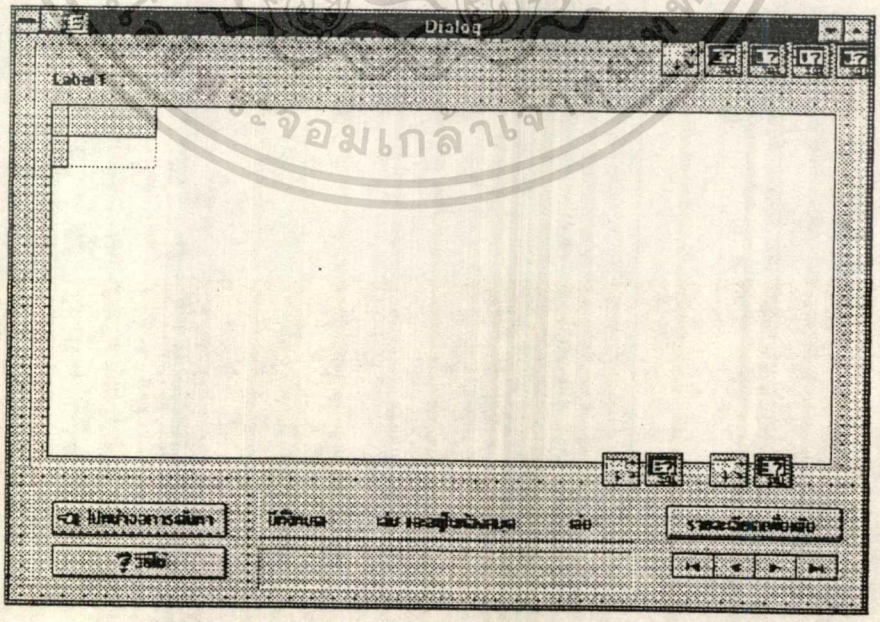
รูปที่ 4-77 หน้าจอแสดงรายชื่อหนังสือใหม่

3.3 แสดงรายชื่อบุคคลค้างส่งหนังสือ



รูปที่ 4-78 หน้าจอแสดงรายชื่อบุคคลค้างส่งหนังสือ

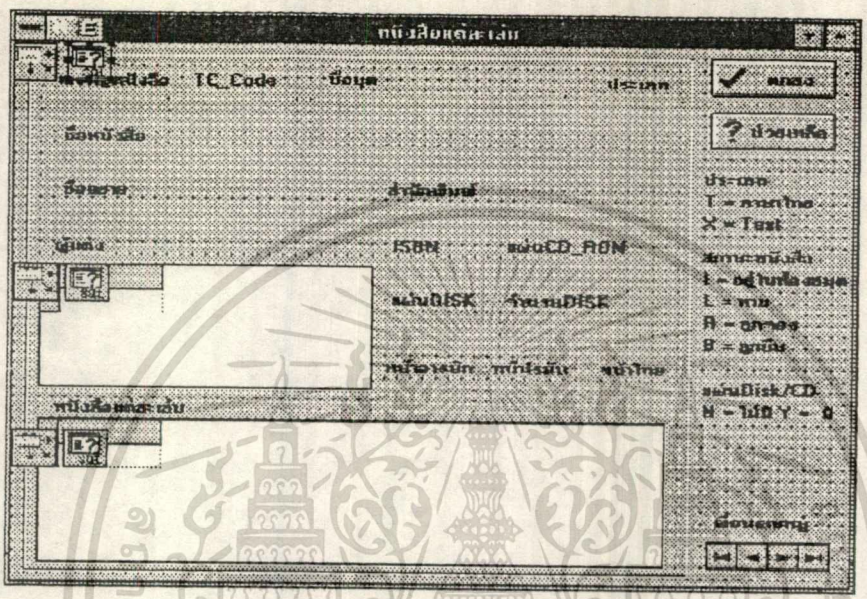
4. หน้าจอแสดงผลสำหรับข้อมูลบางส่วน - ทั้ง หนังสือ, โสตทัศนูปกรณ์, วารสาร และ ปริญญานิพนธ์ ใช้ร่วมกัน



รูปที่ 4-79 หน้าจอสำหรับข้อมูลบางส่วน

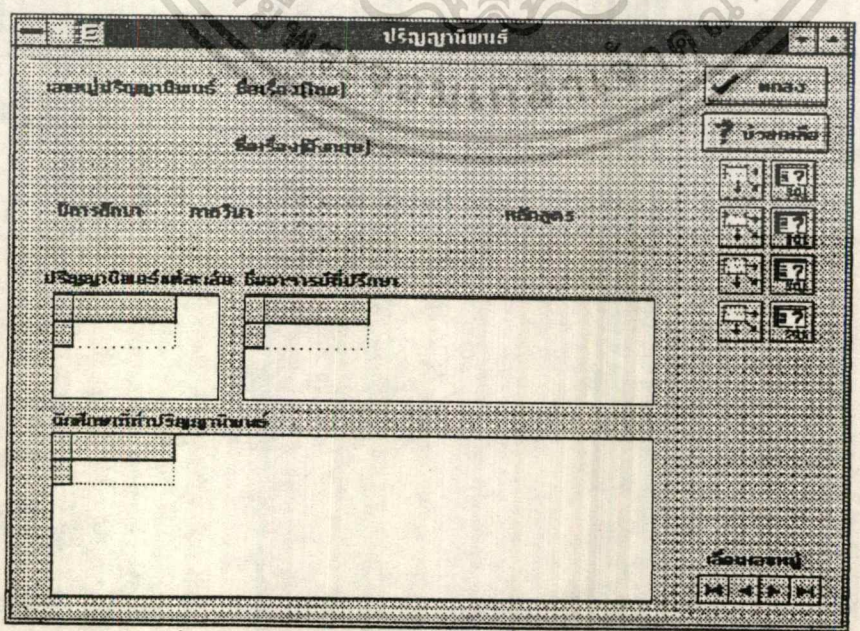
5 หน้าจอแสดงผลสำหรับข้อมูลทั้งหมด

5.1 หน้าจอแสดงผลสำหรับข้อมูลทั้งหมดของหนังสือ



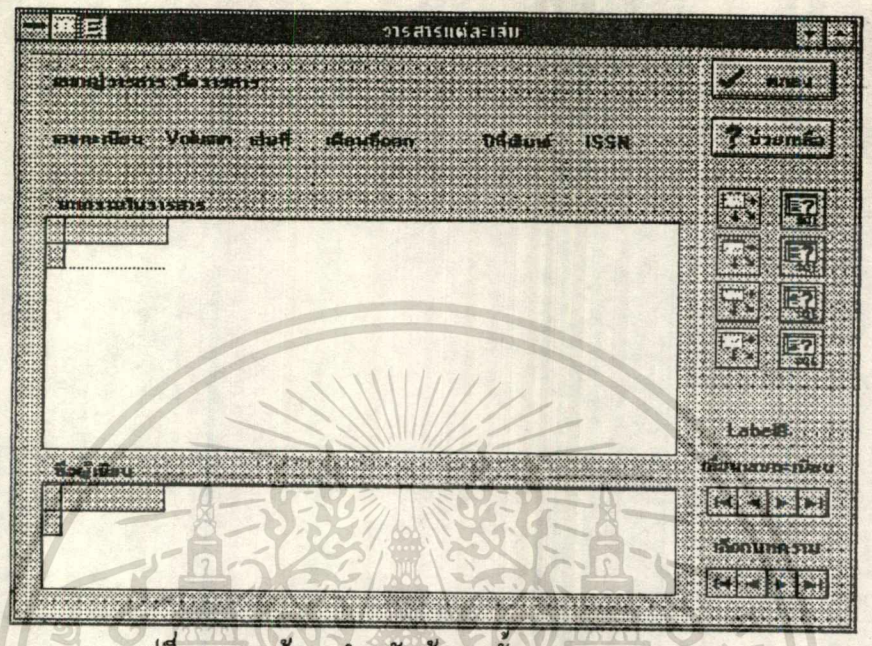
รูปที่ 4-80 หน้าจอสำหรับข้อมูลทั้งหมดของหนังสือ

5.2 หน้าจอแสดงผลสำหรับข้อมูลทั้งหมดของปริญญานิพนธ์



รูปที่ 4-81 หน้าจอสำหรับข้อมูลทั้งหมดของปริญญานิพนธ์

5.3 หน้าจอแสดงผลสำหรับข้อมูลทั้งหมดของวารสาร

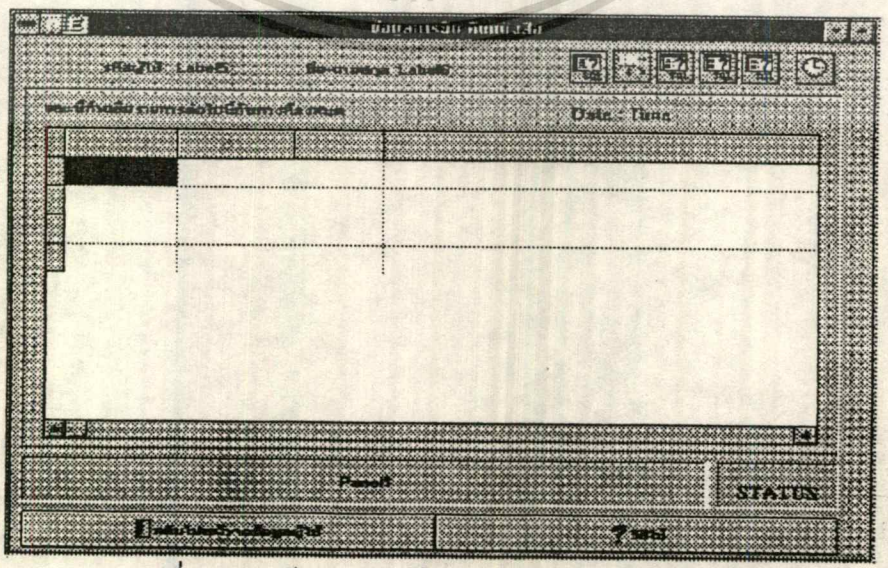


รูปที่ 4-82 หน้าจอสำหรับข้อมูลทั้งหมดของวารสาร

● หมายเหตุ หน้าจอแสดงผลสำหรับข้อมูลทั้งหมดของโสตทัศนูปกรณ์ จะแสดงในหน้าจอแสดงข้อมูลบางส่วน เพราะ ฟิลด์ที่จัดเก็บมีน้อยสามารถแสดงในหน้าจอนี้ได้ทั้งหมด

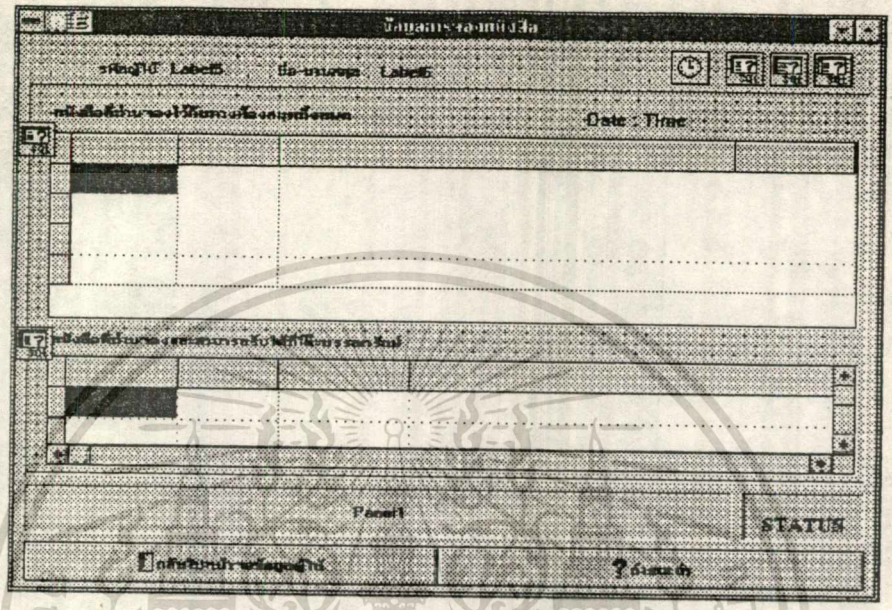
6. หน้าจอแสดงข้อมูลผู้ใช้

6.1 หน้าจอแสดงข้อมูลการยืม- คืนหนังสือ จะมีการบอกว่ายืมไปกี่เล่มจากที่ยืมได้ทั้งหมดกี่เล่ม และ แจ้งค่าปรับด้วย



รูปที่ 4-83 หน้าจอแสดงข้อมูลการยืม- คืนหนังสือ

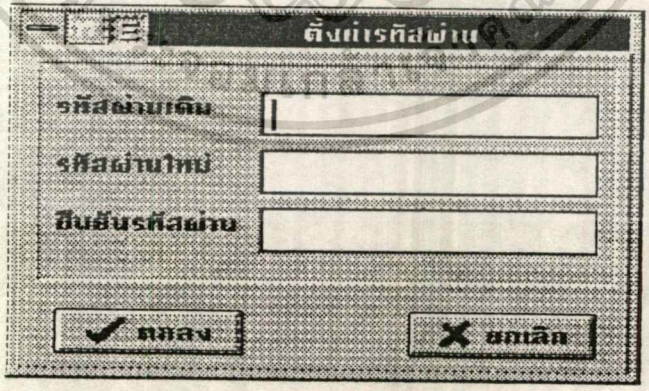
6.2 หน้าจอแสดงข้อมูลการจองหนังสือ จะมีการบอกว่าจองไปกี่เล่มจากที่จองได้ทั้งหมดกี่เล่ม และ แจ้งด้วยว่าหนังสือที่จองสามารถรับได้หรือยัง



รูปที่ 4-84 หน้าจอแสดงข้อมูลการจองหนังสือ

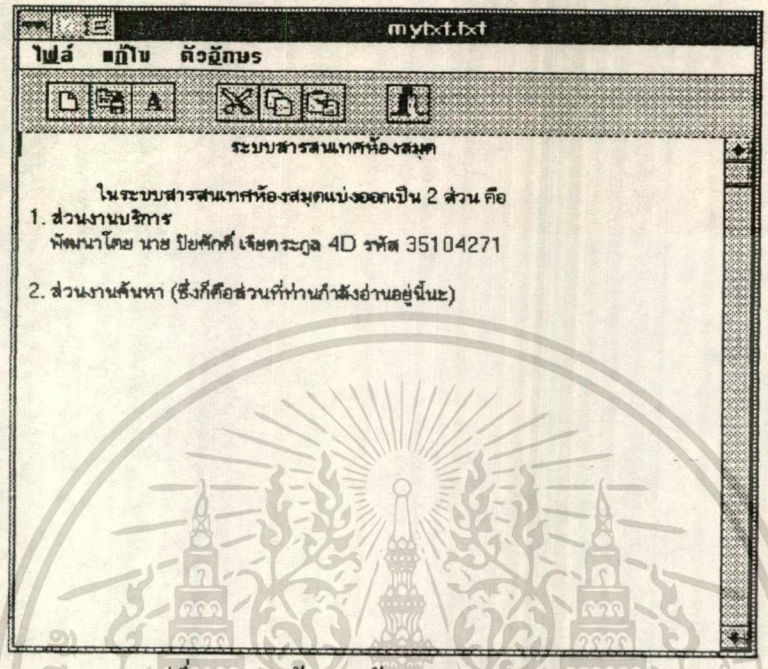
7. หน้าจอสำหรับเจ้าหน้าที่ จะแบ่งเป็น 3 ส่วน

7.1 หน้าจอเปลี่ยนรหัสผ่านใหม่



รูปที่ 4-85 หน้าจอเปลี่ยนรหัสผ่านใหม่

7.2 หน้าจอสร้างรายงานประจำวัน



รูปที่ 4-86 หน้าจอสร้างรายงานประจำวัน

7.3 หน้าจอเลือกรายงานประจำวันจากไฟล์



รูปที่ 4-87 หน้าจอเลือกรายงานประจำวันจากไฟล์

หลังจากได้ออกแบบหน้าจอครบแล้ว ก็เป็นขั้นตอนการออกแบบภาษาเอสคิวแอลที่ใช้ในการดึงข้อมูลจากตารางจัดเก็บข้อมูลต่างๆโดยให้สัมพันธ์กับหน้าจอที่ออกแบบและคาต้าโฟลว์ของระบบงาน จากนั้นก็ทำการออกแบบโปรแกรมก่อนที่จะเขียนจริง โดยสร้างสเตทไดอะแกรม และซูโดโค้ด และต่อมาก็ทำการเขียนโปรแกรม

ในหัวข้อการออกแบบภาษาเอสคิวแอล,สเตทไดอะแกรม,ซูโดโค้ด ของโปรแกรมจะไม่ขอกล่าวในที่นี้ทั้งหมด เพราะจะนำไปรวมใน คู่มือการใช้งานของโปรแกรมเมอร์

4.6 การออกแบบโปรแกรม

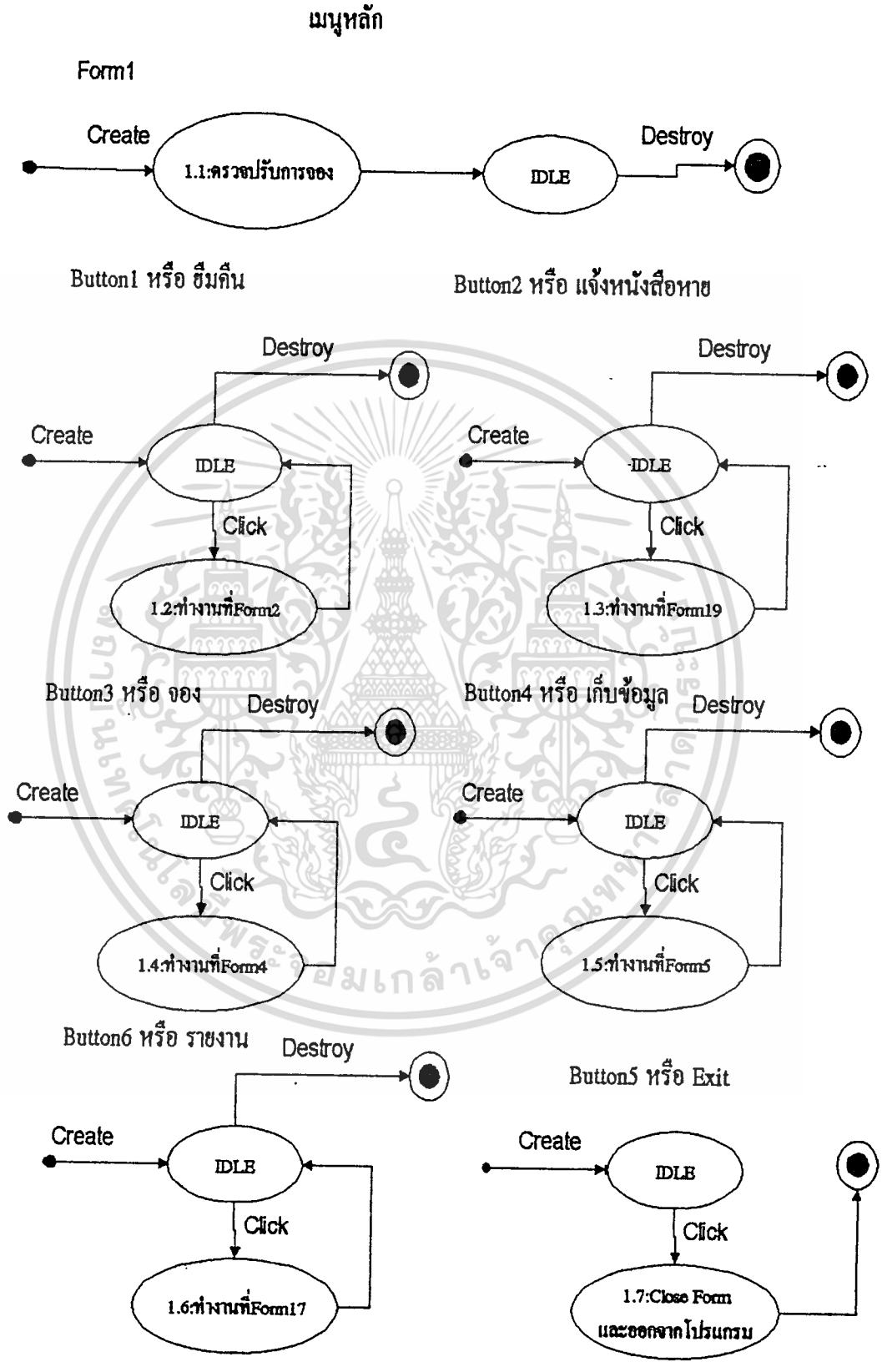
ในตัวอย่างจะขอกล่าวเฉพาะหน้าจอเมนูหลัก,หน้าจอการยืม,หน้าจอการคืน โดยขั้นตอนในการออกแบบได้เขียนในรูปเสตทไดอะแกรม และซูโดโค้ด

1. หน้าจอเมนูกลัก

เป็นหน้าจอแรกที่ผู้ใช้เห็นเมื่อเริ่มแรกทำงานโดยหน้าจอนี้จะเชื่อมกับทุกส่วนของงานบริการได้แก่ ส่วนยืม-คืน ,จอง , แจ้งหนังสือหาย,สร้างรายงาน,จัดเก็บข้อมูล จึงออกแบบให้เป็น ปุ่มต่างๆ เพื่อให้คลิกซ์ด้วยเมาส์หรือ กด enter ที่ปุ่มนั้น ซึ่งแต่ละส่วนของหน้าจอหรือแต่ละคอมโพเนนต์ มีการทำงานดังเสตทไดอะแกรมที่จะได้แสดงต่อไป

จะเห็นว่าหน้าจอหลักนี้ประกอบด้วยองค์ประกอบหลักด้วย ฟอร์มหมายเลข 1 และ ปุ่มอีก 6 ปุ่ม และแต่ละปุ่มก็มีหน้าที่การทำงานแตกต่างกันไป ตามเหตุการณ์ที่เกิด เช่น ปุ่มหมายเลข1 เมื่อมีการ Create ปุ่มขึ้นมา หมายความว่าเมื่อหน้าจอนี้ถูกสร้างและแสดงปุ่มขึ้นมาให้เห็น หากมีการคลิกซ์ปุ่มนั้นก็จะไปทำงานที่ ฟอร์มหมายเลข 2 (หน้าจอยืมคืน) ถ้าไม่มีการคลิกซ์ใดๆ ก็จะอยู่ในสถานะ Idle เพื่อรอจนกว่าจะมีการคลิกซ์เข้ามา หรือ หากหน้าจอนี้ถูกเลิกใช้ (เกิดเหตุการณ์Destroy) ก็เป็นผลให้ปุ่มถูกทำลายไปด้วย ส่วนปุ่มอื่นๆก็มีหลักการในการทำงานเช่นเดียวกัน

ในส่วนของฟอร์มหมายเลข 1 นั้น หลังจากถูกสร้างแล้วก็จะทำ ฟังก์ชั่นการตรวจปรับการจองก่อนหมายถึงเมื่อผู้ใช้ได้จองหนังสือไว้ และหนังสือที่ถูกจองนั้นกลับมายังห้องสมุดแล้ว ก็จะเก็บหนังสือเล่มนี้ไว้ให้ผู้จองไว้ 3 วัน ซึ่งหากไม่มีใครมารับก็จะเปลี่ยนมือไปให้ผู้จองถัดไปหรือนำกลับไปยังชั้นหนังสือในห้องสมุดต่อไป เมื่อตรวจปรับการจองแล้ว ฟอร์มก็จะมีสถานะ Idle รอจนกว่าจะออกจากโปรแกรม (Destroy) จึงเป็นการสิ้นสุดของฟอร์มนี้



รูปที่ 4-88 เสตทไดอะแกรมของหน้าจอเมนูหลัก

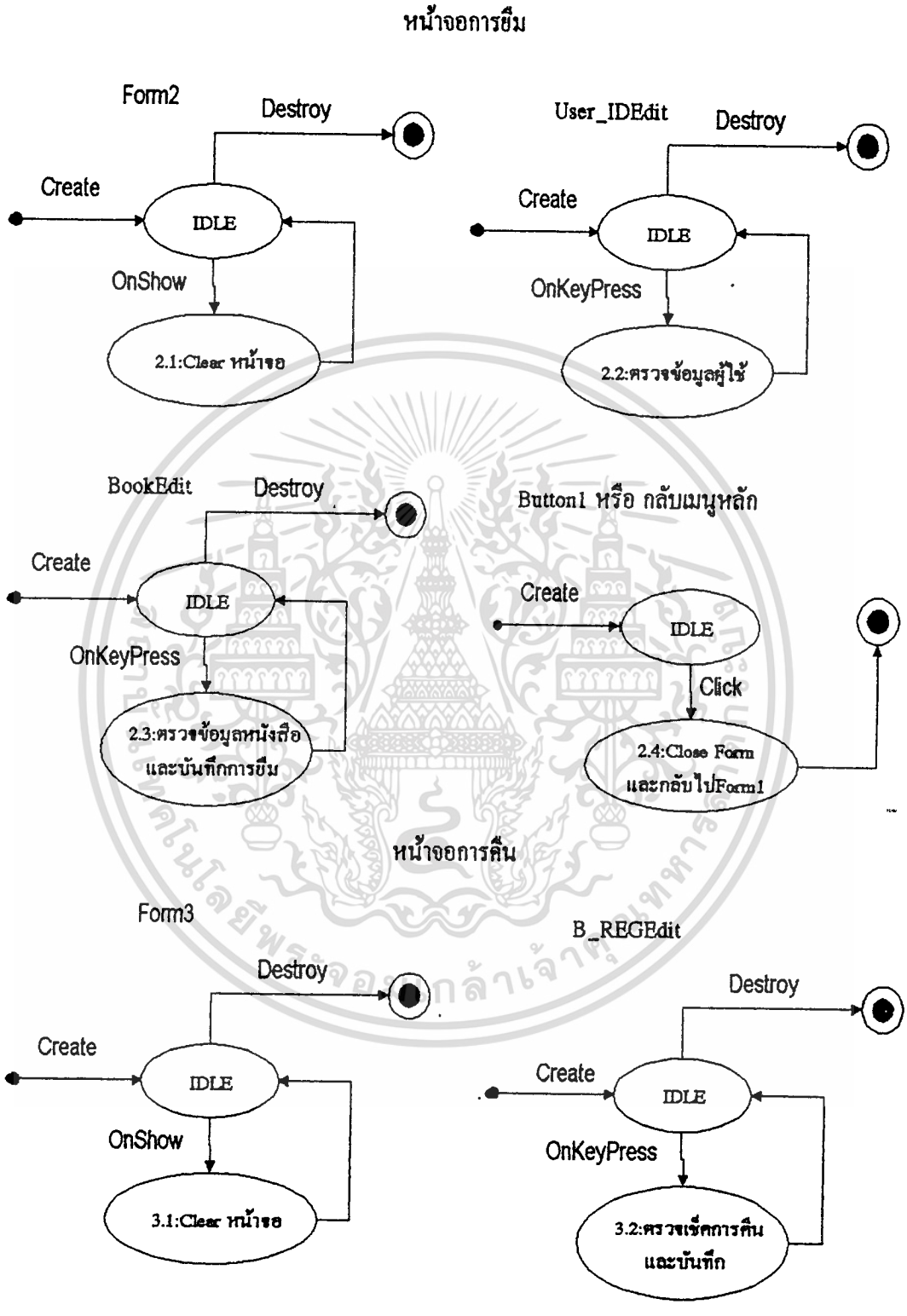
2. หน้าจอการยืม-คืน

ในส่วนหน้าจอการยืม จะเห็นว่าในหน้าจอมีส่วนในให้เติมรหัสนักศึกษา(User_IDEdit) และรหัสหนังสือ(B_REGEdit) ดังนั้น ในการทำงานของคอมโพเนนต์ User_IDEdit หลังจากที่หน้าจอถูกสร้างขึ้นมาแล้ว ก็จะอยู่ในสถานะ Idle เพื่อรอการกดคีย์บอร์ดเมื่อมีการกดเมื่อไหร่ ก็ให้ไปทำงานในฟังก์ชั่นตรวจข้อมูลผู้ใช้ ซึ่งในส่วนนี้จะต้องตรวจก่อนว่าเป็นคีย์ enter หรือไม่ หากเป็นคีย์ enter แล้วก็จะทำการตรวจดูว่าผู้ใช้หมดอายุสมาชิกหรือยัง และ ยืมหนังสือครบแล้วหรือยัง (การทำงานในรายละเอียดทั้งหมดของฟังก์ชั่นได้อธิบายไว้เป็นซูโดโค้ดแล้ว แต่ ในจุดนี้จะอธิบายในส่วนของสเตทเมนต์เท่านั้น ทำให้ยังไม่ลงลึกถึงการทำงานทุกขั้นตอน) หากตรวจข้อมูลผู้ใช้ผ่านแล้วก็จะไปรอรับรหัสหนังสือที่จะยืมต่อไป

การทำงานของคอมโพเนนต์ BookEdit ก็จะกระโดดไปทำงานในฟังก์ชั่นตรวจข้อมูลหนังสือ และบันทึกการยืมเมื่อมีการกดคีย์ใดๆ ซึ่งในฟังก์ชั่นของการตรวจข้อมูลและบันทึกการยืมก็จะเช็คว่าเป็นคีย์ enter หรือไม่ หากเป็นค่อยทำการตรวจข้อมูลหนังสือและบันทึกการยืมจริงๆ นั่นหมายความว่าผู้ใช้ต้องใส่รหัสหนังสือแล้วกด enter ตามมา จึงจะตรวจข้อมูลและบันทึก

การทำงานของคอมโพเนนต์ Button1 เมื่อมีการกดหรือคลิกซ์ก็จะกลับไปยังหน้าจอเมนูหลักหรือ Form1

ในส่วนของหน้าจอการคืนก็มีลักษณะคล้ายๆของหน้าจอการยืมคือมีส่วนรอรับรหัสหนังสือที่จะคืนคือ B_REGEdit เมื่อมีการกดคีย์ใดๆก็จะกระโดดเข้าไปทำงานในฟังก์ชั่น ตรวจเช็คการคืนและบันทึก โดยในฟังก์ชั่นจะตรวจก่อนว่าเป็นการกดคีย์ enter หรือไม่จึงจะให้เข้าไปตรวจและบันทึกจริง (รายละเอียดของฟังก์ชั่นดูได้จากซูโดโค้ดในโปรแกรมเมอร์แมนนวล)



รูปที่ 4-89 สเตทไดอะแกรมของหน้าจอยืม-คืน

4.7 การพัฒนาแอพพลิเคชั่นโดยใช้เดลไฟ

ในการโปรแกรมสามารถแบ่งออกเป็น 2 ขั้นตอนใหญ่ คือการเตรียมโปรแกรม กับการรันทดสอบ กล่าวคือ

การเตรียมโปรแกรมแบ่งออกเป็น 3 ขั้นตอนรอง คือ

1. กำหนดคอมโปเนนต์

-โดยเลือกแล้วกำหนดตำแหน่งและขนาดในฟอร์ม

-จะได้เป็นออปเจ็กต์ มีชื่อตามชื่อคอมโปเนนต์ต่อท้ายด้วยหมายเลข

2. ให้ค่าพรอพเพอร์ตี้ส์ในหน้า Properties โดยคลิ๊กเม้าส์ ณ หัวข้อคือค่าพรอพเพอร์ตี้ส์ที่ต้องการแล้วให้ค่า

3. ป้อนโปรแกรม โดยดับเบิลคลิ๊กเม้าส์ ณ อีเวนต์ที่ต้องการในหน้า Events จะแสดงโครงโปรแกรมให้ป้อนโปรแกรม

ส่วนในการรันทดสอบสามารถที่จะรันโปรแกรมเพื่อการทดสอบ หรืออาจถือว่าเป็นการทดลองรันได้ทุกขณะที่ต้องการ โดยเลือกรายการ Run/Run หรือเลือกปุ่ม Run ที่สปีดบาร์ และสามารถยุติการรันได้โดยปิดหน้าต่างตามปกติซึ่งเมื่อจะเตรียมโปรแกรมต่อให้ยุติการรันก่อน

ต่อจากนี้ไปจะเป็นตัวอย่างในการโปรแกรมของงานห้องสมุดของเมนเมนูในส่วนงานบริการ

1. ในการสร้างเมนเมนูจะเริ่มจาก สร้างหน้าจอให้เหมือนกับที่ออกแบบโดยเลือกคอมโปเน้น Button สำหรับปุ่มต่างๆมา 6 คอมโพเนนต์ซึ่งทำโดยดับเบิ้ลคลิ๊กที่ Button คอมโพเนนต์ในกล่องอุปกรณ์ จากนั้นก็เลือกคอมโพเนนต์ Label เพื่อแสดงป้ายอักษร

2. จากนั้นก็ให้ค่า พรอพเพอร์ตีส์ ในหน้า Properties ของแต่ละออฟเจ็กต์ โดย

คอมโพเน้น Button1 เปลี่ยนพรอพเพอร์ตีส์แคปชั่น(caption) เป็น “ยืม-คืน”

คอมโพเน้น Button2 เปลี่ยนพรอพเพอร์ตีส์แคปชั่น เป็น “แจ้งหนังสือหาย”

คอมโพเน้น Button3 เปลี่ยนพรอพเพอร์ตีส์แคปชั่น เป็น “จอง”

คอมโพเน้น Button4 เปลี่ยนพรอพเพอร์ตีส์แคปชั่น เป็น “จัดเก็บข้อมูล”

คอมโพเน้น Button5 เปลี่ยนพรอพเพอร์ตีส์แคปชั่น เป็น “รายงาน”

คอมโพเน้น Button6 เปลี่ยนพรอพเพอร์ตีส์แคปชั่น เป็น “Exit”

คอมโพเน้น Label1 เปลี่ยนพรอพเพอร์ตีส์แคปชั่น เป็น “รายงานประจำวัน”

3. เริ่มป้อนโปรแกรม โดยดับเบิ้ลคลิ๊กอีเว้นต์ที่ต้องการในหน้า Event โดย คอมโพเน้น Button1 ให้เลือกอีเว้นต์ OnClick หรืออาจดับเบิ้ลคลิ๊กที่คอมโพเน้น Button1 เลยก็ได้ซึ่งก็มีความเหมือนเลือกอีเว้นต์ onClick เดลไฟจะสร้างโค้ดที่จำเป็นให้หมด เราเพียงเขียนโค้ดที่ต้องการลงในอีเว้นต์นั้นเท่านั้นเอง เช่นในคอมโพเน้น Button1 นี้ได้เขียนโค้ดว่า

```
procedure TForm1.Button1Click(Sender: TObject);
begin
        Form1.Hide;
        Form2.ShowModal;
        Form1.Show;
end;
```

จากโค้ดจะเห็นว่าเราได้เขียนโค้ดเพียง 3 บรรทัดเท่านั้นเอง โดยมีความหมายว่า เมื่อมีการคลิ๊กที่ Button1 จะซ่อนหน้าจอปัจจุบันแล้วย้ายไปทำงานที่หน้าจอ 2 (Form2) จากนั้นเมื่อ หน้าจอ 2 ถูกปิดก็จะกลับมาทำงานที่หน้าจอเมนเมนู ต่อ

ในส่วนคอมโพเน้นอื่นๆ ก็มีลักษณะเช่นกันคือ ต้องเลือกอีเว้นต์ก่อน แล้วก็เขียนโค้ด ในอีเว้นต์นั้นซึ่งก็เขียนเป็นภาษาปาสคาลธรรมดานั่นเอง และในหนึ่งคอมโพเน้นก็สามารถเลือกอีเว้นต์ได้มากกว่า 1 อีเว้นต์ ตามแต่ต้องการ

เมื่อทำการรันโปรแกรมหากเกิดเหตุการณ์ใดๆ ที่ตรงกับเหตุการณ์ที่มีการเขียนโค้ดไว้ โปรแกรมก็จะกระโดดเข้าไปทำงานในโค้ดส่วนนั้น

4.8 สรุป

ในบทนี้ได้กล่าวถึงการพัฒนาระบบ หลังจากทราบระบบงานของห้องสมุดแล้วว่ามีการดำเนินงานเป็นอย่างไรในบทที่แล้ว โดยเริ่มการออกแบบฐานข้อมูลแบบไนแอม เมื่อได้ตารางของฐานข้อมูลมาแล้ว ก็นำตารางที่ได้มาออกแบบงานของห้องสมุดเพื่อให้ใช้ตารางเหล่านี้ โดยการแตกย่อยดาต้าโฟลว์เดิมให้ลึกจนถึงระดับตาราง จากนั้นก็เริ่มออกแบบหน้าจอของแอพพลิเคชั่นที่จะนำไปใช้งานจริงให้สัมพันธ์กับดาต้าโฟลว์ แล้วก็เริ่มออกแบบโปรแกรมและทำการพัฒนาแอพพลิเคชั่นโดยใช้เดลไฟในขั้นตอนถัดมา ในบทนี้ได้ยกตัวอย่างวิธีพัฒนาแอพพลิเคชั่นโดยใช้เดลไฟไว้พอสังเขป

บทที่ 5
บทวิจารณ์และสรุป

ในการพัฒนาระบบงานของห้องสมุดโดยใช้เดลไฟนั้น ผู้พัฒนาระบบจะต้องมีพื้นฐานความรู้เกี่ยวกับ ภาษาปาสคาลมาก่อน และโดยเฉพาะแอพพลิเคชั่นที่ต้องเกี่ยวข้องกับฐานข้อมูลแล้ว ผู้พัฒนาจะต้องมีพื้นฐานในด้านฐานข้อมูลพอสมควร เช่น ภาษาเอสคิวแอล ,ทรานแอกชั่น (Tranaction) ,การออกแบบตารางฐานข้อมูล เป็นต้น

ในปริญญานิพนธ์เล่มนี้ ส่วนใหญ่เน้นในเรื่องการออกแบบมากกว่า ที่จะมาอธิบายว่า เขียนโปรแกรมอย่างไร ซึ่งเป็นเพราะการออกแบบนั้นเป็นส่วนสำคัญที่สุดในการพัฒนาระบบงานใดๆก็ตาม หากออกแบบระบบดีแล้ว การเขียนโปรแกรมก็สามารถทำได้อย่างรวดเร็วและมีปัญหาน้อย ปัญหาส่วนใหญ่จึงเกิดเฉพาะ ด้านความไม่คุ้นเคยกับการโปรแกรมแบบวิชวลของเดลไฟเสียมากกว่า

เราสามารถใช้เดลไฟโดยแม็ป(map) จากสเตทไดอะแกรมเข้ากับระบบติดต่อกับผู้ใช้ได้อย่างเหมาะเจาะ อีกทั้งเดลไฟมีการเขียนโปรแกรมออกเป็นส่วนๆตามเหตุการณ์(Event) ก็ยิ่งทำให้การเขียนโปรแกรมนั้นง่ายยิ่งขึ้น

ในด้านไคลน์เซิร์ฟเวอร์ เดลไฟนั้นสนับสนุนโอดีบีซีอยู่แล้ว ดังนั้นจึงสามารถติดต่อกับฐานข้อมูลใดๆ ที่สนับสนุนโอดีบีซี ซึ่งทำให้ไม่ต้องยึดติดกับดีบีเอ็มเอสยี่ห้อใดยี่ห้อหนึ่ง

เป็นที่คาดว่า เมื่อโครงงานนี้นำไปใช้ก็สามารถทำประโยชน์ให้กับการบริการของห้องสมุดของคณะวิศวกรรมศาสตร์ได้หลายๆด้าน คือ

1. ทำให้การค้นหาและการจองหนังสือ โดยผู้ใช้สะดวกรวดเร็วยิ่งขึ้นเพราะได้มีการออกแบบให้เป็นระบบที่ง่ายและน่าใช้(User Friendly) มากที่สุด
2. ทำให้การบริการ ยืม-คืน ในห้องสมุดมีประสิทธภาพมากขึ้น
3. ทำให้การรวบรวมและเก็บข้อมูลของห้องสมุด มีประสิทธิภาพ และเป็นระเบียบเรียบร้อยขึ้น
4. สามารถคิดเงินค่าปรับเนื่องจากการคืนล่าช้า ได้อย่างถูกต้อง และรวดเร็ว
5. ผู้ใช้บริการและเจ้าหน้าที่ มีความสะดวกสบายในการทำงานยิ่งขึ้น เพราะออกแบบให้มีฟังก์ชั่นที่อำนวยความสะดวกต่อผู้ใช้และเจ้าหน้าที่ให้ครอบคลุมมากที่สุด

กิตติกรรมประกาศ

ปริญญานิพนธ์เล่มนี้สามารถลุล่วงไปได้เนื่องมาจากการช่วยเหลือจากบุคคลต่อไปนี้ ซึ่งผู้จัดทำวิทยานิพนธ์ต้องขอขอบพระคุณเป็นอย่างสูง

ผศ.ดร.ศุภมิตร จิตตะยโศธร ผู้ให้คำปรึกษา และ แนะนำแนวทาง

พี่ๆ ห้องรองคณบดีฝ่ายระบบสารสนเทศ

เพื่อน ๆ ทุกคน ที่คอยให้กำลังใจ คอยช่วยเหลือในการทำปริญญานิพนธ์นี้ ให้สำเร็จลุล่วงลงด้วยดี



หนังสืออ้างอิง

1. James Rumbaugh , Michael Blaha , William Premerlani , Frederick Eddy , William Lorensen , "Object Oriented Modeling and Design" , PRENTICE HALL ,1991.

2. "Database Application Developer's Guide" , Borland International.

3. "ODBC 2.0 Programmer 's Reference and SDK Guide" , Microsoft Press, 808 p, 1993.

