ระบบสารสนเทศอาคารปฏิบัติการคณะวิศวกรรมศาสตร์

มหาวิทยาลัยนเรศวร

INFORMATION WORKSHOP BUILDING ENGINEERING

OF NARESUAN UNIVERSITY

นายปริญญา เขจรรักษ์ รหัส 42360552

นายเอกมล ชูเที่ยง รหัส 42360768

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ 28 เม.ย. 2553
เลขทะเบียน 1A9415824
เลขเรียกหนังสือ ว.ร.
มหาวิทยาลัยนเรศวร ว/4585
2545

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2545

ใบรับรองโครงงานวิจัย

หัวข้อโครงงาน ระบบสารสนเทศอาคารปฏิบัติการคณะวิศวกรรมศาสตร์มหาวิทยาลัยนเรศวร

ผู้ดำเนินงานโครงงาน นายปริญญา เขจรรักษ์ รหัส 42360552

นายเอกมล ชูเที่ยง รหัส 42360768

อาจารย์ที่ปรึกษา อาจารย์ภาณุพงศ์ สอนคม

สาขา วิศวกรรมคอมพิวเตอร์

ภาควิชา วิศวกรรมไฟฟ้าและคอมพิวเตอร์

ปีการศึกษา 2545

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้โครงงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์

คณะกรรมการตรวจสอบโครงงานวิจัย

.....ประธานกรรมการ

(อาจารย์สิทธิโชค เชาวกุล)

.....กรรมการ

(อาจารย์พรพิศุทธ์ วรจิรันตน์)

.....กรรมการ

(อาจารย์รัฐภูมิ วรานุสาสน์)

.....กรรมการ

(อาจารย์ภาณุพงศ์ สอนคม)

หัวข้อโครงงาน	**ระบบสารสนเทศอาคารปฏิบัติการคณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร**		
ผู้ดำเนินงานโครงงาน	**นายปริญญา**	**เขจรรักษ์**	**รหัส 42360552**
	นายเอกมล	**ชูเที่ยง**	**รหัส 42360768**
อาจารย์ที่ปรึกษา	**อาจารย์ภาณุพงศ์ สอนคม**		
สาขา	**วิศวกรรมคอมพิวเตอร์**		
ภาควิชา	**วิศวกรรมไฟฟ้าและคอมพิวเตอร์**		
ปีการศึกษา	2545		

บทคัดย่อ

ระบบเทคโนโลยีสารสนเทศ ได้เข้ามามีส่วนสำคัญในการดำเนินชีวิตในสังคมยุคใหม่ ช่วยในการจัดการ ระบบองค์กรได้อย่างมีประสิทธิภาพ การจัดการระบบสารสนเทศอาคารปฏิบัติการคณะวิศวกรรมศาสตร์ มหาลัยนเรศวร เริ่มจากการจัดซื้อวัสดุและอุปกรณ์ การจัดการทะเบียนครุภัณฑ์ การยืมครุภัณฑ์ การายงานประจำปี การจ้างซ่อม

โปรแกรมนี้ประกอบด้วย 2 ส่วน ในส่วนแรกใช้ภาษา Delphi ในการพัฒนาโปรแกรม ซึ่งการทำงาน ในลักษณะของ เครื่องคอมพิวเตอร์ลูกข่าย (Client) และเครื่องคอมพิวเตอร์แม่ข่าย (Server) โดยใช้ BDE (Borland Database Engine)และ ODBC เป็นเครื่องมือในการติดต่อกับฐานข้อมูลระหว่างเครื่องคอมพิวเตอร์ลูกข่าย (Client) และเครื่องคอมพิวเตอร์แม่ข่าย (Server) ในส่วนที่สองได้ใช้ ASP (Active Server Page) ในการพัฒนาทำ ให้ผู้ใช้สามารถค้นหาอุปกรณ์ รายงานการคืนอุปกรณ์ผ่านทางเครื่องคอมพิวเตอร์ที่ติดอยู่กับ ระบบเครือข่ายคอมพิวเตอร์ โดยไม่ต้องเดินทางมาที่ระบบปฏิบัติการ

Project Title	INFORMATION WORKSHOP BUILDING ENGINEERING OF NARESUAN UNIVERSITY
Name	Mr.Parinya Kejornlak ID.42360552
	Mr.Ekkamon Chooteang ID.42360768
Project Advisor	Mr.Panupong Sonkhom
Major	Computer Engineering
Department	Electrical and Computer Engineering
Acedemic Year	2002

ABSTRACT

Information Technology has become an essential component of everyday life in modern society. It help to manage organization systems Management Information Workshop Build Engineering of Naresuan University begin buying device, hire, repair device, loaning and returning device, accounting, Device and report end of year.

This computer program consists of two part Delphi 5 is employer for the first part. This part works on client-server mode by using BDE and ODBC, the tools which connect database between client and server. The other part employed ASP (Active Server Page) for programming. This allows user to search and check loaning while processing computer anywhere outside the workshop building.

กิตติกรรมประกาศ

ในการทำโครงงานวิศวกรรมครั้งนี้ คณะผู้จัดทำได้รับความอนุเคราะห์เป็นอย่างดีจากบุคคลหลายฝ่าย และหน่วยงานราชการอีกหลายหน่วยงาน สามารถทำงานได้สำเร็จลุล่วงมาด้วยดี คณะผู้จัดทำจึงขอกราบขอบพระคุณมา ณ ที่นี้

คณะผู้จัดทำขอกราบขอบพระคุณ อ.ภาณุพงศ์ สอนคม ซึ่งเป็นที่ปรึกษาโครงงาน ตามที่ท่านให้ความกรุณาแนะนำวิธีในการทำงาน ให้เข้าใจถึงการศึกษาอย่างเป็นระบบ ขั้นตอน ตามที่ท่านให้ความกรุณาแนะนำวิธีในการทำงานและชี้แนวทางแก้ไขในทุกขั้นตอนตลอดจนการศึกษา

ขอขอบพระคุณ เจ้าหน้าที่ฝ่ายพัสดุ ที่เอื้อเฟื้อเอกสารต่างๆ ในการทำโครงงานในครั้งนี้

นาย ปริญญา เขจรรักษ์

นาย เอกมล ชูเที่ยง

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....ก
บทคัดย่อภาษาอังกฤษ.....ข
กิตติกรรมประกาศ.....ค
สารบัญ.....ง
สารบัญตาราง.....ฉ
สารบัญรูป.....ช

บทที่ 1 บทนำ
1.1 หลักการและเหตุผล.....1
1.2 วัตถุประสงค์.....1
1.3 ขอบข่ายของงาน.....1
1.4 กิจกรรมการดำเนินงาน.....2
1.5 ผลที่คาดว่าจะได้รับ.....2
1.6 งบประมาณ.....3

บทที่ 2 หลักการและทฤษฎี
2.1 ระบบคืออะไร.....4
2.2 ระบบฐานข้อมูล.....5
2.3 สถาปัตยกรรมของระบบฐานข้อมูล.....6
2.4 แบบจำลองของฐานข้อมูลเชิงสัมพันธ์ (Relational Database Model).....6
2.5 โครงสร้างข้อมูลของฐานข้อมูลแบบ Relational.....7
2.6 การออกแบบฐานข้อมูล.....8
2.7 Data Flow Diagram (DFD).....9
2.8 Entity-Relationship Model.....12
2.9 การรีเลชั่นให้อยู่ในรูปแบบนอร์มัล(Normalization).....15
2.10 ขั้นตอนการออกแบบฐานข้อมูลโดยใช้โมเดล.....15
2.11 การติดต่อกับฐานข้อมูล.....17
2.12 ความสัมพันธ์ระหว่าง เดลไฟ(Delphi) กับฐานข้อมูล.....18

สารบัญ (ต่อ)

หน้า

2.13 ASP(Active Server Page) คืออะไร.....20

บทที่ 3 การออกแบบฐานข้อมูล

3.1 เข้าใจในปัญหา (Problem Recognition).....30

3.2 ศึกษาความเป็นไปได้ (Feasibility Study).....30

3.3 การวิเคราะห์ (Analysis).....30

3.4 การออกแบบ (Design).....31

3.5 ทำการออกแบบฐานข้อมูล.....39

บทที่ 4 การพัฒนาโปรแกรม

4.1 รูปแบบโปรแกรมการยืมคืนวัสดุและอุปกรณ์.....45

4.2 รูปแบบโปรแกรมการจัดซื้อหรือจัดจ้าง.....52

4.3 เวบไซต์ระบบสารสนเทศอาคารปฏิบัติการคณะวิศวกรรมศาสตร์มหาวิทยาลัยนเรศวร.....63

บทที่ 5 สรุปผล

5.1 สรุปผล.....67

5.2 ประเมินผลและข้อเสนอแนะ..... 67

5.3 ปัญหา ข้อเสนอแนะ และแนวทางแก้ไข.....68

ภาคผนวก ก.....69

ภาคผนวก ข.....73

เอกสารอ้างอิง.....80

ประวัติผู้เขียน.....81

สารบัญตาราง

ตารางที่ หน้า

1.1 แผนการดำเนินงานตลอดโครงงาน....2

2.1 สัญลักษณ์ER....16

2.2 รูปแบบการติดต่อแบบ DSNLess กับฐานข้อมูล Access และ SQL Server....26

2.3 ชื่อของตัวแปรประเภท Object ที่ใช้แทน Recordset ของ ADO....28

2.4 การใช้คำสั่ง Move ต่างๆในการเขียน ASP....29

3.1 Data Description/Data Dictionary....39

สารบัญรูป

รูปที่ หน้า

2.1 การแทนการไหลของข้อมูลเป็นลูกศร.....10

2.2 การแทนกระแสข้อมูลเป็นลูกศร.....10

2.3 การแทนคำนามที่อยู่นอกระบบ.....10

2.4 การแทนแหล่งเก็บข้อมูล.....11

2.5 การแทนสัญลักษณ์เพิ่มเติม.....11

3.1 Context Diagram.....31

3.2 แผนภาพ DFD ของกระบวนการ 0 ระดับที่ 1..... 32

3.3 แผนภาพ DFD ของกระบวนการที่ 1 ระดับที่ 1..... 33

3.4 แผนภาพ DFD ของกระบวนการที่ 1 ระดับที่ 2..... 34

3.5 แผนภาพ DFD ของกระบวนการที่ 2 ระดับที่ 1.....35

3.6 แผนภาพ DFD ของกระบวนการที่ 3 ระดับที่ 1..... 36

3.7 ER Diagram 37

3.8 โครงสร้างของฐานข้อมูลระบบสารสนเทศอาคารปฏิบัติการคณะวิศวกรรมศาสตร์.....38

4.1 แบบฟอร์มเมนูหลักระบบสารสนเทศอาคารปฏิบัติการคณะวิศวกรรมศาสตร์.....45

4.2 ลักษณะของโปรแกรมการยืมครุภัณฑ์.....46

4.3 ลักษณะโปรแกรมการยืมวัสดุ.....47

4.4 ลักษณะโปรแกรมการคืนวัสดุและครุภัณฑ์.....48

4.5 ลักษณะโปรแกรมรายงานการตรวจสอบวัสดุและครุภัณฑ์.....49

4.6 ลักษณะโปรแกรมรายงานการตรวจสอบครุภัณฑ์.....50

4.7 ลักษณะโปรแกรมการเพิ่มรายละเอียดอื่นๆ.....51

4.8 ลักษณะโปรแกรมการเพิ่มหน่วยงาน.....51

4.9 ลักษณะโปรแกรมเมนูหลักการจัดซื้อหรือจัดจ้าง.....52

4.10 ลักษณะโปรแกรมการจัดซื้อครุภัณฑ์.....53

4.11 การจัดซื้อหรือจัดจ้าง(การจัดซื้อวัสดุ).....54

4.12 ลักษณะโปรแกรมการจัดซื้อหรือจัดจ้าง(การจ้างซ่อม).....55

4.13 ลักษณะโปรแกรมหนังสืออนุมัติเสนอซื้อหรือจ้าง.....56

4.14 ลักษณะโปรแกรมการตรวจรับวัสดุและครุภัณฑ์.....57

สารบัญรูป (ต่อ)

รูปที่ หน้า

4.15 ลักษณะโปรแกรมการเบิกวัสดุและอุปกรณ์.....58

4.16 ลักษณะโปรแกรมรายงานการตรวจสอบ.....59

4.17 ลักษณะโปรแกรมการเพิ่มรายละเอียดอื่นๆ(การเพิ่มรายชื่อนิสิต).....60

4.18 ลักษณะโปรแกรมการเพิ่มรายละเอียดอื่นๆ(การเพิ่มหน่วยงาน).....61

4.19 ลักษณะโปรแกรมการเพิ่มรายละเอียดอื่นๆ(การเพิ่มรายการร้านค้า).....62

4.20 เวบไซต์หลักสำหรับการค้นหาข้อมูล.....63

4.21 หน้าจอเมื่อทำการสืบค้น Note Book แล้ว.....64

4.22 ตรวจสอบการยืมคืน.....65

4.23 การตรวจสอบการยืมคืน.....65

4.24 การตรวจสอบการยืมเกินเวลา.....66

บทที่ 1
บทนำ

1.1 หลักการและเหตุผล

เครื่องมือและอุปกรณ์ทางด้านวิศวกรรมศาสตร์ มีความสำคัญมากในการเรียนการสอนใน คณะวิศวกรรมศาสตร์ การจัดซื้อ การจัดสรรอุปกรณ์ การตรวจเช็คอุปกรณ์ บำรุงรักษา จึงมีความสำคัญมากเพื่อให้การใช้งานมีความสะดวก รวดเร็ว ปลอดภัยและเกิดประสิทธิภาพการใช้งานสูงสุด ในปัจจุบันคณะวิศวกรรมศาสตร์ ได้มีระบบการจัดการดังกล่าว ทุกๆอาคารปฏิบัติการมีลักษณะคล้ายคลึงกัน แต่ไม่มีการจัดเก็บข้อมูลที่แน่ชัด กล่าวคือ บันทึกข้อมูลลงกระดาษทุกขั้นตอนการทำงานและบางขั้นตอนที่เป็นการทำงานเดียวกัน แต่ปฏิบัติต่างกัน เช่นการยืมคืนอุปกรณ์ เป็นต้น ดังนั้นเพื่อให้อาคารปฏิบัติการดำเนินไปในทิศทางเดียวกันทุกอาคารปฏิบัติการภายในคณะวิศวกรรมศาสตร์ พร้อมทั้งสอดคล้องกับการบริหารงานของคณะวิศวกรรมศาสตร์ จึงได้มีการจัดระบบสารสนเทศอาคารปฏิบัติการคณะวิศวกรรมศาสตร์ขึ้นมา

1.2 วัตถุประสงค์

1.จัดระบบสารสนเทศของอาคารปฏิบัติการคณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร เพื่อตอบสนองการเรียนการสอนของนิสิตนักศึกษา อาจารย์ผู้สอน และพนักงานในอาคารปฏิบัติการคณะวิศวกรรมศาสตร์

2.จัดทำฐานข้อมูลในการจัดเก็บข้อมูลและโปรแกรมการจัดฐานข้อมูลของเครื่องมือและอุปกรณ์ของอาคารปฏิบัติการคณะวิศวกรรมศาสตร์

3.บันทึกข้อมูลลงในคอมพิวเตอร์ แทนการใช้กระดาษซึ่งง่ายต่อการตรวจสอบ

1.3 ขอบข่ายของงาน

โครงงานนี้จัดทำเกี่ยวกับระบบฐานข้อมูล ซึ่งมีการใช้เครื่องมือ(โปรแกรมคอมพิวเตอร์) เพื่อใช้ในการพัฒนาและนำเสนอโครงงานดังนี้

1ออกแบบฐานข้อมูลด้วย Microsoft Access 2000

2.ใช้ Delphi ในการเขียนโปรแกรมจัดการฐานข้อมูล

3ใช้ Visio ในการออกแบบแผนภาพ กระแสการไหลของข้อมูลเพื่อนำเสนอ

การใช้เครื่องมืออาจมีการเปลี่ยนแปลงตามความเหมาะสม

1.4 กิจกรรมการดำเนินงาน

ตาราง 1.1 ขั้นตอนการดำเนินงานโครงงาน

กิจกรรม	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค	เม.ย.	พ.ค.	มิ.ย.	ก.ค.	ส.ค.	ก.ย.
1.เขียนโครงสร้างการทำงาน	↔										
2.รวบรวมข้อมูลใน Shop		↔									
3.ศึกษาความเป็นไป เพื่อใช้ในการตัดสินใจ การใช้เครื่องมือในการพัฒนาโปรแกรม			↔								
4.ออกแบบระบบฐานข้อมูล			↔	↔							
5.เขียนโปรแกรมจัดการฐานข้อมูล					↔	→	→	→			
6.ทดลองใช้จริงกับระบบ								↔	→		
7.ทดลองระบบหากมีปัญหาเกิดขึ้นกับระบบ										↔	
8.ทำโครงงานฉบับร่าง										↔	↔
9.ปรับปรุงแก้ไขโครงงานให้สมบูรณ์											↔

1.5 ผลที่คาดว่าจะได้รับ

1.การใช้เครื่องมือและอุปกรณ์มีประสิทธิภาพสูงสุด มีความปลอดภัย ง่ายต่อการจัดเก็บและการค้นหา การรายงานสั่งซื้อเครื่องมือและอุปกรณ์ สามารถตรวจสอบได้จากฐานข้อมูล

2.มีฐานข้อมูลและโปรแกรมจัดเก็บฐานข้อมูลของเครื่องมือและอุปกรณ์ของอาคารปฏิบัติการคณะวิศวกรรมศาสตร์

3.ลดการใช้กระดาษในการบันทึกข้อมูลโดยการใช้ฐานข้อมูลในการจัดเก็บแทน

4.มีการขยายโครงการสู่คณะอื่น เพื่อการบริหารเครื่องมือและอุปกรณ์เป็นไปในทิศทางเดียวกันทั้งมหาวิทยาลัย

1.6 งบประมาณ

1.ค่าหนังสือ และเอกสารต่างๆ	700 บาท
2.ค่าจัดทำรูปเล่มรายงาน	800 บาท
3.อื่นๆ เช่น แผ่นโปรแกรมที่ใช้ในการพัฒนา ,ค่าถ่ายเอกสาร ฯลฯ	500 บาท
รวมค่าใช้จ่าย	2000 บาท

บทที่ 2
หลักการและทฤษฎี

การพัฒนาระบบงานฐานข้อมูลให้มีประสิทธิภาพนั้น เราจะต้องเข้าใจพื้นฐานการทำงานของระบบฐานข้อมูล องค์ประกอบ และคำศัพท์ต่างๆ ที่ใช้ในระบบฐานข้อมูล อีกทั้งยังต้องเข้าใจขั้นตอนต่างๆ ที่ใช้ในการพัฒนาระบบ เพื่อจะได้มองภาพรวมของระบบได้อย่างชัดเจนและส่งผลให้พัฒนาโปรแกรมมีประสิทธิภาพต่อไป โดยเนื้อหาที่เกี่ยวข้องมีดังต่อไปนี้

2.1 ระบบคืออะไร [2]

ระบบคือกลุ่มขององค์ประกอบต่างๆ ที่ทำงาน ร่วมกันเพื่อจุดประสงค์เดียวกัน ระบบอาจประกอบไปด้วยบุคลากร เครื่องมือ เครื่องใช้ พัสดุ วิธีการ ซึ่งทั้งหมดนี้จะต้องมีระบบจัดการอันหนึ่งเพื่อให้บรรลุจุดประสงค์อันเดียวกัน เช่น ระบบการเรียนการสอน มีจุดประสงค์เพื่อให้นักเรียนได้รับความรู้ในเนื้อหาวิชาที่สอน ระบบนี้จึงประกอบไปด้วยองค์ประกอบต่างๆ

เมื่อเราได้ศึกษาระบบใดระบบหนึ่ง เราควรจะต้องเข้าใจการทำงานของระบบนั้นให้ดี โดยการถามตัวเองตลอดเวลาด้วยคำถามนี้

1.ระบบทำอะไร (What)

2.ทำโดยใคร (Who)

3.ทำเมื่อไร (When)

4.ทำอย่างไร (How)

2.1.1 ระบบสารสนเทศ (Management Information System)

ระบบนี้ช่วย จัดการข้อมูลที่ต้องการใช้ในระบบธุรกิจ ช่วยเก็บตัวเลขและข่าวสารเพื่อช่วยในการดำเนินธุรกิจและการตัดสินใจ ตัวอย่างของระบบสารสนเทศ เช่น ระบบการเก็บเงินสินค้า เราต้องการที่จะทราบ ว่าลูกค้าแต่ละคนซื้ออะไร อยู่ที่ไหน สินค้าและจำนวนที่ขายให้แก่ลูกค้าแต่ละคนเป็นอย่างไร การจ่ายเงินของลูกค้าเป็นอย่างไร ติดค้างนานหรือไม่ หรือหนี้สูญ รวมทั้งจำนวนเงิน ที่ลูกค้าจะต้องชำระ ระบบสารสนเทศอาจจะใช้หรือไม่ใช้คอมพิวเตอร์ก็ได้ กรณีที่เราจะใช้คอมพิวเตอร์ ก็มีเหตุผลหลายอย่าง เช่น เราต้องการทราบข้อมูลอย่างรวดเร็ว หรือจำนวนลูกค้ามีมาก จึงสะดวกในการทำงานมากกว่าถ้าใช้คอมพิวเตอร์ แทนที่จะใช้คนหลายๆคน (Manual) เป็นต้น การที่ฝ่ายบริหาร ทราบข้อมูลต่างๆได้อย่างรวดเร็วจะช่วยตัดสินใจได้อย่างรวดเร็วและถูกต้อง ทำให้การวางแผนต่างๆได้ล่วงหน้าและทันท่วงที

2.2 ระบบฐานข้อมูล [1]

2.2.1 ระบบแฟ้มข้อมูล

ในอดีตองค์กรต่างๆมักจัดเก็บเอกสารไว้ในแฟ้มเอกสารต่างๆ ซึ่งมีความเกี่ยวข้องกับฐานข้อมูลน้อยหรือไม่มีเลย เช่น การจัดเก็บอุปกรณ์ใน shop มักทำการเก็บ เอกสารไม่เป็นระเบียบ ทำการบันทึกการยืมคืนอุปกรณ์ในกระดาษ และเมื่อมีอุปกรณ์มากขึ้น การค้นหาเอกสารต้อง ใช้เวลามากขึ้น และมีความยากลำบากมากขึ้น การจัดเก็บข้อมูลในคอมพิวเตอร์จึงถูก นำมาใช้ในองค์กรแทนการจัดเก็บในรูปแบบเดิม โดยรูปแบบการจัดเก็บในคอมพิวเตอร์นี้อยู่ใน รูปแฟ้มข้อมูล แฟ้มข้อมูลที่ถูก จัดเก็บรวมอยู่ด้วยกันนี้เรียกว่า "ระบบแฟ้มข้อมูล"

การใช้งานระบบแฟ้มข้อมูล จะต้องอาศัยโปรแกรมเมอร์พัฒนาโปรแกรม เพื่ออ่านข้อมูลจากแฟ้มข้อมูล จากแฟ้มต่างๆ ขึ้นมา ประมวลผล เช่น ภาษา COBOL FORTRAN BASIC เป็นต้น ทำให้มีข้อจำกัดในการเรียกใช้ข้อมูล ความซับซ้อนในการพัฒนาโปรแกรม และแต่ละโปรแกรมถูกผูกติดกับแฟ้มข้อมูลต่างๆ เช่น เมื่อต้องการเปลี่ยนแปลงโครงสร้างแฟ้มข้อมูล จึงจำเป็นต้องเขียนแฟ้มข้อมูลขึ้นมาใหม่

2.2.2 ปัญหาของระบบแฟ้มข้อมูล

ด้วยการจัดเก็บแฟ้มข้อมูลของระบบแฟ้มข้อมูลเป็นเอกเทศและกระจัดกระจาย เช่น ในแต่ละหน่วยงานในองค์กรมีการสร้างระบบแฟ้มข้อมูลขึ้นมาใช้งานภายในหน่วยงานของตัวเองทำให้เกิดปัญหาต่างๆ เช่น การจัดเก็บข้อมูลที่ซ้ำซ้อนกัน การขัดแย้งกันของข้อมูล

2.2.3 ระบบฐานข้อมูล

จากปัญหาต่างของระบบแฟ้มข้อมูล ทำให้เกิดการจัดเก็บข้อมูลในรูปแบบใหม่ที่เรียกว่า "ฐานข้อมูล (Database)" ฐานข้อมูลคือการนำเอาข้อมูลต่างๆที่มีความสัมพันธ์กันมาเก็บไว้ในที่เดียวกันรวมกันเป็นฐานข้อมูลทำให้เกิดการใช้ข้อมูลร่วมกัน

2.2.4 องค์ประกอบของระบบฐานข้อมูล

-ข้อมูล

-ฮาร์ดแวร์(Hardware)

-ซอฟแวร์(Software)

-ผู้ใช้ระบบฐานข้อมูล

2.2.5 ประโยชน์ของฐานข้อมูล

สามารถลดความซ้ำซ้อนของการเก็บข้อมูลได้

สามารถหลีกเลี่ยงความขัดแย้งกันของข้อมูลได้

ในแต่ละองค์สามารถใช้ข้อมูลร่วมกันได้

สามารถกำหนดระดับความปลอดภัยให้กับข้อมูลได้ โดยกำหนดระดับความสามารถในการเรียกใช้ข้อมูลของผู้ใช้แต่ละคน ให้แตกต่างกันตามความรับผิดชอบได้

สามารถรักษาความถูกต้องให้กับข้อมูลได้ โดยเฉพาะระบุกฎเกณฑ์ในการควบคุมความผิดพลาดในการป้อนข้อมูลผิด

ทำให้ข้อมูลเป็นอิสระจากโปรแกรมที่ใช้งานข้อมูลนั้นที่ทำให้ผู้พัฒนาโปรแกรมทำการแก้ไขโครงสร้างของข้อมูลได้โดยไม่มีผลกระทบต่อโปรแกรมที่เรียกใช้ข้อมูลนั้น

2.3 สถาปัตยกรรมของระบบฐานข้อมูล

สถาปัตยกรรมของฐานข้อมูลเป็นการอธิบาย ถึงรูปแบบและโครงสร้างของ ข้อมูลภายใน ระบบฐานข้อมูล โดยทั่วไประดับแนวความคิด ไม่ขึ้นอยู่กับโครงสร้าง ของระบบฐานข้อมูลนั้นๆ สำหรับสถาปัตยกรรมของระบบฐานข้อมูลที่นิยมใช้ได้แก่ สถาปัตยกรรม ANSI/SPRC ได้แบ่งออกเป็น 3 ระดับดังนี้

ระดับ Internal เป็นสถาปัตยกรรมในระดับที่เกี่ยวข้องกับโครงสร้างทางกายภาพในการเก็บข้อมูลในฐานข้อมูลมากที่สุดเนื่องจากเป็นระดับที่กล่าวถึงการจัดเก็บข้อมูล

ระดับ External เป็นระดับที่เกี่ยวข้องกับผู้ใช้มากที่สุด เนื่องจากเป็นระดับที่กล่าวถึงมุมมองที่มีต่อข้อมูลของผู้ใช้แต่ละคน

ระดับ Conceptual เป็นสถาปัตยกรรมที่กล่าวถึงโครงสร้างของข้อมูลในระดับแนวความคิดซึ่งเป็นภาพของข้อมูลที่ใช้แทนโครงสร้างทางกายภาพของข้อมูลที่แท้จริงที่เก็บอยู่ในฐานของข้อมูล

2.4 แบบจำลองของฐานข้อมูลเชิงสัมพันธ์ (Relational Database Model)

ฐานข้อมูลที่มีโครงสร้างในแบบ Relation ได้รับการพัฒนาขึ้นในแบบจำลองที่กล่าวถึงความสัมพันธ์ระหว่างข้อมูลที่มีชื่อว่า Relation Model ข้อมูลที่จัดเก็บอยู่ในฐานข้อมูลที่มีโครงสร้างข้อมูลในแบบ Relational จะถูกแยกจัดเก็บออกเป็นหน่วยย่อยๆ ที่เรียกว่า Relation หรือที่เรียกว่า Table ที่อยู่ในรูปตารางที่ประกอบด้วยชุดของแถวและชุดของสดมภ์ ข้อมูลที่ถูกจัดเก็บในตารางจะถูกแยกเป็นเอกเทศแต่สามารถนำมาสร้างความสัมพันธ์ร่วมกันได้ โดยความสัมพันธ์ที่สร้างมานี้อยู่ในรูปแนวความคิดมากกว่าทางกายภาพ

2.5 โครงสร้างข้อมูลของฐานข้อมูลแบบ Relational

ใน Relation Model ได้นิยามต่างๆคือ Relation,Tuple,Attribute,Domain เพื่อใช้อธิบายถึงข้อมูลใน Relation Model นี้

2.5.1 Relation

เป็นหน่วยที่ใช้จัดเก็บข้อมูลที่อยู่ในรูปแบบของตารางประกอบด้วย Row และ สดมภ์ Column สำหรับชื่อของแต่ละแถวของ Relation เรียกว่า Tuple ส่วนชื่อของแต่ละสดมภ์เรียกว่า Attribute

2.5.2 Domain

เป็นการนิยามของขอบเขตของค่าที่เป็นไปได้ให้กับข้อมูลในแต่ละ Attribute เช่นการกำหนดให้ค่าของเงินเดือนพนักงาน จะต้องมีค่ามากกว่า 0 หรือการกำหนดเพศของพนักงานจะต้องมีค่าเป็นชาย(M) หรือ หญิง(F)

2.5.3 Key

Key คือ Attribute หรือชุด Attribute ของที่ทำให้ข้อมูลแต่ละ มีค่าข้อมูลที่ไม่ซ้ำกัน เช่น Attribute EmpID ในตัวอย่างตารางในข้อ 4.1key นั้นแบ่งออกเป็นดังนี้

Primary key จะถูกใช้เป็น key สำหรับการตรวจสอบการซ้ำกันของข้อมูล ระหว่างที่ทำการป้อนข้อมูลหรือกำหนดข้อมูลใหม่ให้กับ Relation เช่น Attribute EmpID

Foreign Key จะเป็น Attribute ใด Attribute หนึ่งใน Relation ที่ใช้อ้างอิงไปยัง Attribute ที่ทำหน้าที่เป็น Primary Keyของอีกตารางหนึ่งใน"DepID" ของตารางแรกจะเป็น Foreign Key

Candidate Key เป็น ที่สามารถระบุตำแหน่งของแต่ละTupleใน Relstionได้

Null คือค่าของข้อมูลที่ไม่สามารถระบุค่าได้ ซึ่งอาจเกิดจากการกรอกข้อมูลไม่ครบถ้วน เช่น

2.5.4 กฎ Referentail Intrigrity

เป็นกฎที่กำหนดไว้ว่า "ฐานข้อมูลใดๆจะต้องไม่ปรากฏ Relation ที่มีข้อมูลที่มีค่าของ Foreign key ที่ไม่ขึ้นกับ Primary key ของ Relation อื่น"

2.5.5 กฎ Entity Intrigrity

เป็นกฎที่กำหนดว่า Primary key ของ Relation ใดๆห้ามมีค่าเป็น Null

2.6 การออกแบบฐานข้อมูล

ฐานนับเป็นส่วนที่สำคัญสำหรับงานสารสนเทศที่ใช้คอมพิวเตอร์ในการประมวลผล เนื่องจากเป็นส่วนที่ใช้จัดเก็บข้อมูลต่างๆ ซึ่งใช้เป็นอินพุตของระบบสารสนเทศจึงต้องให้ความสำคัญต่อการออกแบบฐานข้อมูลเช่นเดียวกันกับการออกแบบส่วนประมวลผล

2.6.1 วงจรชีวิตการพัฒนาระบบฐานข้อมูล (Data Life Cycle)

วงจรชีวิตของการพัฒนาระบบฐานข้อมูล เป็นขั้นตอนที่กำหนดขึ้น เพื่อใช้เป็นแนวทางในการพัฒนาระบบฐานข้อมูลขึ้นใช้งาน ประกอบด้วยขั้นตอนต่างๆ ดังนี้

Database Innitial Study เป็นขั้นตอนแรกของการพัฒนาระบบฐานข้อมูล ในขั้นตอนนี้ ผู้พัฒนาระบบฐานข้อมูลจะวิเคราะห์ความต้องการต่างๆ ของผู้ใช้ เพื่อกำหนดจุดมุ่งหมาย ปัญหา ขอบเขตและกฎระเบียบต่างๆ ของระบบฐานข้อมูลที่จะพัฒนาขึ้นเพื่อใช้เป็นแนวทางในการออกแบบฐานข้อมูลต่อไป

Database Design เป็นขั้นตอนในการเอารายละเอียดต่างๆ ที่ได้จากการวิเคราะห์ในขั้นตอนแรกมากำหนดเป็นแนวทางในการออกแบบ ซึ่งแบ่งเป็น 3 ระดับคือ การออกแบบฐานข้อมูลในระดับ Conceptual,Logical และ Physical

Implementation and Loading เป็นขั้นตอนที่นำเอาโครงร่างต่างๆ ของระบบฐานข้อมูลที่ได้จากการออกแบบในขั้นตอนการออกแบบ มาสร้างเป็นข้อมูลที่จะใช้เก็บจริง รวมทั้งแปลงข้อมูลของระบบเดิมให้สามารถนำมาใช้งานในระบบฐานข้อมูลที่พัฒนาขึ้นมาใหม่

Testing and Evaluation เป็นขั้นตอนการทดสอบระบบฐานข้อมูลที่พัฒนาขึ้น เพื่อหาข้อผิดพลาดต่างๆ รวมทั้งการประเมินความสามารถของระบบฐานข้อมูล เพื่อนำไปเป็นแนวทางในการปรับปรุง ให้รองรับความต้องการของผู้ใช้ในด้านต่างๆ ได้อย่างครบถ้วนและถูกต้อง

Operation เป็นขั้นตอนที่เอาระบบฐานข้อมูลที่พัฒนาขึ้นเรียบร้อยแล้วมาใช้งานจริง

Maintainance and Evolution เป็นเรื่องที่เกิดขึ้นระหว่างการใช้ฐานข้อมูลจริง เพื่อบำรุงรักษาให้ระบบฐานข้อมูลทำงานได้อย่างมีประสิทธิภาพ รวมทั้งขั้นตอนการแก้ไข และการปรับปรุงฐานข้อมูลในกรณีที่มีการเพิ่ม หรือเปลี่ยนแปลงความต้องการของผู้ใช้

รายละเอียดที่ได้แต่ละขั้นตอนของการพัฒนาระบบฐานข้อมูลจะสามารถสะท้อนกลับไปยังขั้นตอนก่อนหน้านี้ ซึ่งจะช่วยปรับปรุง และแก้ไขข้อผิดพลาดในการออกแบบของขั้นตอนการออกแบบที่ผ่านมาได้เป็นอย่างดี

2.6.2 ขั้นตอนการออกแบบฐานข้อมูล

การออกแบบฐานข้อมูลแบ่งออกได้เป็น 3 รูปแบบดังนี้

การออกแบบฐานข้อมูลในระดับ conceptual การออกแบบฐานข้อมูลในระดับนี้ จะเป็นการกำหนดโครงร่างเริ่มต้น ที่จุดมุ่งหมายเพื่ออธิบายโครงสร้างหลักๆของข้อมูลภายในระบบฐานข้อมูลโดยไม่คำนึงถึงฐานข้อมูลที่นำมาใช้ การออกแบบในระดับนี้ มีความสำคัญมาก เนื่องจากโครงสร้างที่ได้จากการออกแบบในขั้นตอนนี้จะถูกนำไปใช้ในขั้นตอนต่อไป โครงร่างหรือที่เรียกว่า Schema ที่ได้จากการออกแบบในขั้นตอนนี้เรียกว่า Conceptual Schema

การออกแบบในระดับ Logical การออกแบบในระดับนี้จะเป็นระดับที่ต่อเนื่องมาจากระดับ Conceptual กล่าวคือ การออกแบบนี้จะอาศัยโครงร่างที่ได้จากการออกแบบในระดับ Conceptual มาปรับปรุงให้มีโครงสร้างเป็นไปตามโครงสร้างข้อมูลที่จะนำมาใช้งาน โดยยังไม่คำนึงถึงผลิตภัณฑ์ทางด้านฐานข้อมูลที่จะนำมาใช้งาน การออกแบบในขั้นตอนนี้ ต้องปรับปรุงโครงสร้างบางอย่างใน Conceptual Schema ให้สอดคล้องกับฐานข้อมูลที่จะนำมาใช้งาน การออกแบบในขั้นตอนนี้จึงต้องมีการตรวจสอบความถูกต้องของโครงร่างที่ออกแบบขึ้นมากับส่วนประมวลผลต่างๆ ที่ออกแบบไว้ รวมทั้งจะต้องแปลงโครงร่างต่างๆ ให้อยู่ในรูป Relation

การออกแบบฐานข้อมูลในระดับ Physical การออกแบบในระดับนี้ จะเป็นขั้นตอนสุดท้ายของการออกแบบฐานข้อมูลในขั้นตอนนี้ จะเป็นการปรับปรุงของโครงร่างของโครงสร้างที่ออกแบบเช่นเดียวกัน แต่การปรับปรุงโครงสร้างของการออกแบบฐานข้อมูลในขั้นตอนนี้ จะเป็นการนำเอาโครงร่างที่ได้จากการออกแบบในระดับ Logical มาปรับปรุงโครงสร้างให้เป็นไปตามโครงสร้างทางผลิตภัณฑ์ทางด้านฐานข้อมูลที่จะนำมาใช้งาน ผลลัพธ์ที่ได้จากการออกแบบในระดับนี้คือ โครงสร้างของระบบฐานข้อมูล ที่สามารถนำไปใช้ในการสร้างฐานข้อมูลจริง

2.7 Data Flow Diagram (DFD) [2]

กรรมวิธีวิเคราะห์ระบบอย่างมีโครงสร้างนั้น วิธีหนึ่งที่นิยมในทางปฏิบัติคือ การมองภาพรวมในรูปแบบการไหลของข้อมูล (Data Flow) โดยที่วิธีนี้จะช่วยให้นักวิเคราะห์สามารถแบ่งระบบให้เป็นระบบย่อยได้ง่ายขึ้นและสามารถตรวจสอบระบบได้สะดวก

การนำเสนอระบบแบบการไหลของข้อมูลนั้นจะใช้สัญลักษณ์แทนการบรรยายการทำงานของระบบ ซึ่งลักษณะที่จะใช้เป็นรูปวงกลม สี่เหลี่ยมจัตุรัส สี่เหลี่ยมผืนผ้าปลายเปิด เส้นโค้ง ลูกศร โดยนำสัญลักษณ์ต่อไปนี้มาเชื่อมต่อ การแสดงต่อเนื่องของข้อมูลและการประมวลผล

2.7.1 สัญลักษณ์ Data Flow Diagram(DFD)

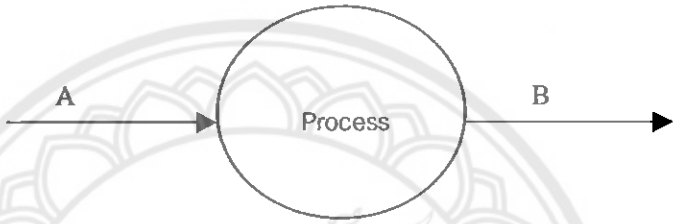
ในแผนภาพของ DFD จะประกอบด้วยสัญลักษณ์ต่างๆดังนี้

1.ลูกศร ใช้แทนการไหลของข้อมูลพร้อมชื่อของข้อมูลนั้นๆจะต้องกำกับไว้ด้วย

ชื่อข้อมูล

รูปที่ 2.1 การแทนการไหลของข้อมูลเป็นลูกศร

2.รูปวงกลม ใช้แทนกริยาการกระทำต่อข้อมูลที่ไหลเข้ามา โดยไม่คำนึงว่าเป็นการกระทำของคนหรือคอมพิวเตอร์ก็ตาม จะได้มาซึ่งผลลัพธ์การไหลออกของข้อมูล



A= ข้อมูลเข้า

B= ข้อมูลไหลออก

รูปที่ 2.2 การแทนกระแสข้อมูลเป็นลูกศร

3. รูปสี่เหลี่ยม ใช้แทนคำนามที่อยู่ภายนอกระบบซึ่งเป็นการกำเนิดของข้อมูลหรือการสิ้นสุดของข้อมูล โดยมีชื่ออยู่ในสี่เหลี่ยม

รูปที่ 2.3 การแทนคำนามที่อยู่นอกระบบ

4. รูปสี่เหลี่ยมผืนผ้าปลายเปิด เป็นตัวแทนของการเก็บข้อมูลหรือแฟ้มข้อมูล เสมือนเป็นตัวพักหรือช่วงขาดของการไหลของข้อมูลเพื่อนำไปเก็บเท่านั้น การกำหนดชื่อ ของแหล่งเก็บข้อมูลต้องอยู่ในสี่เหลี่ยม

ชื่อ

รูปที่ 2.4 การแทนแหล่งเก็บข้อมูล

5. สัญลักษณ์เพิ่มเติม จะใช้เพิ่มเติมลงในสัญลักษณ์ที่กล่าวมาข้างต้นเพื่อแสดงความเป็นสิ่งเดียวกัน แต่จะถูกกล่าวหลายๆครั้งในแผนภาพ

รูปที่ 2.5 การแทนสัญลักษณ์เพิ่มเติม

2.7.2 ลำดับชั้นใน Data Flow Diagram

ในการเขียน DFD นักวิเคราะห์ระบบจะต้องมองระบบจากภาพรวมก่อนจากนั้นมองลึกเข้าสู่รายละเอียดข้างในของระบบยิ่งมองลึกมากเท่าใด ก็จะยิ่งเห็นรายละเอียดของระบบย่อยได้มากขึ้นเท่านั้น

DFD ระดับที่ 0 ให้คิดว่าระบบทั้งระบบเป็น PROCESS หรือวงกลมหนึ่งวง มีลูกศรแทนที่ใช้ในที่ INPUT และ OUTPUT ตามที่จำเป็น

DFD ระดับที่ 1 ให้แตกวงกลมระดับที่ 0 ออกเป็นวงกลมย่อย 2-5 วง ตามความเหมาะสม

DFD ระดับที่ 2 ให้แตกวงกลมระดับที่ 1 ออกเป็นวงกลมย่อยอีกเท่าที่เป็นไปได้

DFD ระดับที่ 3 ถ้าจำเป็นก็ต้องตรวจดูว่า วงกลมในภาพระดับที่ 2 ยังมีความซับซ้อนจำเป็นที่นี้มีต้องแตกย่อย ก็ต้องแตกอีก ก็ต้องสร้างแผนภาพประกอบด้วยวงกลมย่อยแทนไว้วงกลมนั้นให้ได้รายละเอียดสุดท้าย

2.7.3 ทำไมต้องใช้ Data Flow Diagram

DFD ช่วยให้นักวิเคราะห์ระบบสามารถ สรุปข้อมูลที่เกี่ยวข้องกับระบบ, เข้าใจถึงปัญหาสำคัญของระบบและระบุส่วนการที่ซับซ้อน, เข้าใจถึงความสัมพันธ์ส่วนต่างๆของระบบและการประกอบกันเป็นระบบ, พัฒนาระบบได้อย่างมีประสิทธิภาพ, DFD เป็นเอกสารร่วมที่ทำให้นักวิเคราะห์ระบบ และผู้ใช้สามารถเข้าใจระบบและตรวจสอบความถูกต้องได้ทั้งสองฝ่าย

ในการตรวจสอบเรื่องของเวลาที่ใช้ในแต่ละขบวนการนั้น นักวิเคราะห์สามารถใช้ DFD เป็นเครื่องมือที่ทำให้ทราบถึงขอบเขตในการพัฒนารูปแบบของระบบ ว่ามีทางที่จะเป็นไปได้อย่างไรได้บ้าง

2.8 Entity-Relationship Model [1]

การออกแบบฐานข้อมูลใช้งานในระบบสารสนเทศใดๆต้องอาศัยแบบจำลองของข้อมูล เพื่อนำเสนอรายละเอียดต่างๆที่เกี่ยวข้องกับข้อมูลที่ออกแบบ แบบจำลองข้อมูลที่นิยมใช้ ได้แก่ Entity-Relationship Model

2.8.1 Semantic Model

แบบจำลองของข้อมูลในยุคแรกๆมีข้อจำกัด ในการนำเสนอรายละเอียดที่เกี่ยวข้องกับข้อมูลในฐานข้อมูล กล่าวคือ จะมีการนำเสนอเฉพาะรายละเอียดทางด้านโครงสร้างของฐานข้อมูล เช่น คุณสมบัติ Atomicity ของข้อมูล กฎระเบียบต่างที่ใช้ควบคุมความถูกต้องของข้อมูลฯ ลฯ เป็นต้น แต่ยังไม่มีการนำเสนอรายละเอียดทางด้านความหมายของข้อมูลภายในฐานข้อมูลนั้นๆ เช่น จำนวน และน้ำหนักของสินค้าว่ามีความสัมพันธ์กันอย่างไร หรือ Domain ของข้อมูลต่างๆ สามารถมีค่าอะไรได้บ้าง หรือข้อมูลใดที่ทำหน้าที่เป็น Candidate Key หรือForeign Key ฯลฯ เป็นต้น ดังนั้นจึงมีแนวคิดแบบจำลองของข้อมูลใหม่ขึ้นมาที่เรียกว่า "Semantic Model"

2.8.2 คำศัพท์ที่ใช้ใน Semantic Model

ใน Semantic Model ได้มีคำนิยามขึ้นแทนในความหมายต่างๆ ที่เรียกว่าดังนี้

1. Entity เปรียบเสมือนกับคำนามที่สามารถระบุได้ในความเป็นจริง อาจเป็นสิ่งที่จับต้องได้ เช่น นาย ไมเคิล หนังสือที่ชื่อ "ระบบฐานข้อมูล" หรือเป็นสิ่งที่เป็นรูปธรรมไม่สามารถจับต้องได้ เช่น จำนวนวันเวลาพักร้อนของพนักงาน ซึ่งเมื่อนำเอาแต่ละ entity มารวมกันภายใต้คุณลักษณะใดลักษณ์หนึ่งที่เหมือนกันแล้ว entity เหล่านั้นจะถูกเรียกว่า "entity set"

2. Property หรือ Attribute คือ ข้อมูลที่แสดงลักษณะคุณสมบัติของ Entity เช่น Propertyของ Entity set ที่ชื่อ "รถยนต์" ที่ประกอบด้วย หมายเลขทะเบียนรถยนต์ หมายเลขตัวถัง หมายเลขเครื่องยนต์ ยี่ห้อ รุ่น สี เป็นต้น

3. Identity แต่ละEntity ภายใต้ Entity Set เดียวกัน ถึงแม้ว่าจะต้องมี Property ที่เหมือนกัน แต่อย่างไรก็ตาม จะต้องมี Property ใน Property หนึ่ง ซึ่งเป็นเอกลักษณ์เฉพาะของ Entity นั้น เช่น Property หมายเลขประชาชน ของแต่ละ Entityใน Entity Set ประชาชน ซึ่งจะไม่มีหมายเลขใดที่ซ้ำกัน

สำหรับ Propertyที่สามารถนำมากำหนดเป็นเอกลักษณ์เฉพาะให้กับแต่ละ Entity นี้จะเรียกว่า Identity

4. Relationship ได้แก่ Entity Set ที่สร้างขึ้นจาก 2 Entity Set เดิมหรือมากกว่า เพื่อใช้แสดงความสัมพันธ์ระหว่างแต่ละ Entity ใน Entity เดิมเหล่านั้น ในการสร้าง Relationship อาจสร้างด้วยการเอาแต่ละ Entity ในแต่ละ Entity Set เดิมเหล่านั้นมาเชื่อมโยงข้อมูลกันภายใต้ค่าของ Property ที่เหมือนกัน ซึ่ง

ละ Entity ในแต่ละ Entity Set เดิมเหล่านั้นมาเชื่อมโยงข้อมูลกันภายใต้ค่าของ Property ที่เหมือนกัน ซึ่งการสร้างความสัมพันธ์ในลักษณ์นี้ Property ของ Relation จะเกิดจากการนำเอา property ของแต่ละ Entity Set มารวมกัน เช่น Relationship ชื่อ "สังกัดคณะ" ซึ่งเกิดจากการที่ Entity Set "นักศึกษา"และ"คณะ"มี Property "รหัสคณะ" ที่เหมือนกัน

5. Subtype และ Supertype เป็นEntity Set ที่สามารถแบ่งสมาชิกของ Entity Setนั้น ออกเป็น Entity Set ย่อยๆซึ่งสมาชิกของแต่ละ Entity ย่อยนั้น นอกเหนือจากจะต้องมี Propertyที่เหมือนกับหลักแล้ว ย่อยเหล่านั้น จะต้องมี ที่มีค่าซึ่งบ่งบอกถึงความแตกต่างของสมาชิกในแต่ละ Entity Set ย่อยเหล่านั้น สำหรับ Entity Set ย่อยเหล่านี้ จะถูกเรียกว่า "Subtype Entity" ส่วน จะถูกเรียกว่า "Supertype Entity" เช่น Entity Set "EMPLOYEE" ที่มี Property "MILITARY_STATUS" ทีกำหนดสถานภาพการเกณฑ์ทหารของพนักงาน ซึ่งจะปรากฏเฉพาะพนักงานชายเท่านั้น

2.8.3 Entity-Relationship Model

Semantic Model ที่นิยมใช้มากที่สุด ได้แก่ Entity-Relation Model หรือที่นิยมเรียกกันสั้นๆว่า "E-R Model" นับเป็นแบบจำลองที่ครอบคลุมปัญหาต่างๆ ที่กำหนดไว้ใน Semantic Model เนื่องจากมีรูปแบบที่ใช้แทนทุกๆความคิดที่กำหนดไว้ Semantic Model ซึ่งได้แก่ Entity,Property,Relationและ Subtype สำหรับแผนภาพที่สร้างขึ้นโดยใช้รูปภาพแบบต่างๆภายใน เพื่อแสดงความเป็นจริงต่างๆของข้อมูลในฐานข้อมูล จะเรียกว่า หรือที่นิยมเรียกกันสั้นๆว่าแผนภาพ

2.8.4 Cardinality Ratio

สมาชิกใน Entity ที่เกี่ยวข้องกับ Relation จะถูกเรียกว่า Participant ซึ่งจำนวนของParticipant นี้ จะเรียกว่า Degree ของ Relationship นั้น และจะถูกนำไปใช้กำหนดประเภทของ Relationship ที่เรียกว่า "Cardinality Ratio"

1.One-to-One เป็น Relationship ที่แต่ละ Participant ของ Entity หนึ่งจะมีความสัมพันธ์กับอีก Participant ของอีก Entity หนึ่งเพียง Participant เดียว

2.One-to-Many Relationshipเป็น Relationship ที่แต่ละ Participant ของ Entity หนึ่งมีความสัมพันธ์กับอีก Participant ของอีก Entity หนึ่งมากกว่า 1 Participant เช่น กรณีลูกค้าสามารถมีเงินฝากได้มากกว่า 1 บัญชี และแต่ละบัญชีเงินฝากจะต้องมีเจ้าของบัญชีเพียงคนเดียว

3.Many-to-Many Relationship เป็น Relationship ที่ Participant มากกว่า 1 Participant ของ Entity หนึ่ง มีความสัมพันธ์กับอีก Participant ของอีก Entity หนึ่งมากกว่า 1 Participant เช่น กรณีลูกค้าสามารถมีเงินฝากได้มากกว่า 1 บัญชี และแต่ละบัญชีเงินฝากสามารถมีเจ้าของบัญชีได้มากกว่า1 คน

2.8.5 คุณสมบัติของ E-R ที่ดี

เนื่องจากแผนภาพ ถูกใช้เป็นเครื่องมือในการนำเสนอความเป็นจริงที่เกี่ยวข้องกับข้อมูล ดังนั้นจึงควรที่จะมีคุณสมบัติดังนี้

1 Expressiveness แผน E-R ภาพที่ดี สามารถอธิบายโครงสร้างของข้อมูลไดเป็นอย่างดี และครบถ้วน

2. Simplicity จะต้องมีรูปแบบที่ง่ายต่อการเข้าใจ

3. Minimality จะต้องมีความชัดเจนและไม่สามารถตีความเป็นอย่างอื่นได้

2.8.6 การค้นหา Entity

สิ่งที่จำเป็นสำหรับการวาดแผนภาพ ได้แก่ การกำหนดและที่ใช้แทนข้อมูลต่างๆภายในฐานข้อมูลที่ต้องการออกแบบนั้น ที่มีขั้นตอนที่ค่อนข้างซับซ้อน แต่อย่างไรก็ตามผู้ออกแบบ สามารถที่จะกำหนดขึ้นภายในระบบอย่างคร่าวๆ โดยใช้วิธีการค้นหา ด้วยวิธีของ ซึ่งกำหนดไว้ว่า จะถูกกำหนดขึ้นจากสิ่งต่างๆ ที่ปรากฏอยู่ในความต้องการของผู้ใช้ดังต่อไปนี้

1.ขั้นตอนการทำงาน เอกสาร และรายงานต่างๆ ที่ปรากฏอยู่ในระบบ

2.อุปกรณ์ต่างๆที่ระบบใช้ในการติดต่อ

3.ระบบงานที่เกี่ยวข้องในการส่งผ่านข้อมูลระหว่างกัน

4.เหตุการณ์ที่เกิดขึ้นในระบบ และระบบจะต้องมีการบันทึกไว้เป็นหลักฐาน

5.บุคคลหรือสิ่งที่มีบทบาทต่อระบบงาน

6.สถานที่เกี่ยวข้องกับระบบงาน

7.หน่วยงานต่างๆในองค์กรที่เกี่ยวข้องกับระบบงาน

2.9 การรีเลชั่นให้อยู่ในรูปแบบนอร์มัล(Normalization)

รูปแบบนอร์มัลนี้ถูกนิยามขึ้นโดย E.F Codd โดยมี 5 ระดับ ได้มีการปรับปรุงระดับที่ 3 ให้รัดกุมขึ้นโดย Boyce และ Codd และให้ชื่อใหม่ว่า Boyce Codd Normal Form หรือ BCNF การทำรีเลชั่นในรูปแบบนอร์มัลนี้มีวัตถุประสงค์ในการจัดทำเพื่อ

1.ลดเนื้อที่ในการจัดเก็บข้อมูล

2.ลดปัญหาที่ข้อมูลไม่ถูกต้องเนื่องจากการจัดทำให้ข้อมูลใน Relation หนึ่งๆไม่ซ้ำกันทำให้ปรับปรุงข้อมูลง่ายที่จุดๆเดียว

3.ลดปัญหาในการ เพิ่ม แก้ไข และ ลบข้อมูลไม่ครบถ้วน

โดยในการใช้งานจริงเราพบว่าในการทำรีเลชันใช้เพียง 4 ระดับก็เพียงพอต่อการใช้งาน นิยามของการทำรีเลชั่นให้อยู่ในรูปแบบนอร์มัลทั้ง 4 ระดับมีดังนี้

1. First Normal Form:1NF นิยาม Relation ใดๆ อยู่ในรูปแบบของ 1NF ถ้า Relation นั้นๆ ไม่มีกลุ่มซ้ำกันคือไม่มีข้อมูลในบางช่องมากกว่า 1 ค่า

2. Second Normal Form:2NF นิยาม Relation ใดๆ จัดอยู่ในรูปแบบ 2NF ถ้าRelation นั้นเป็น 1NF และไม่มี Attribute ที่เป็น Nonkey ตัวใดตัวหนึ่งขึ้นกับส่วนใดส่วนหนึ่งของคีย์ นิยาม Nonkey Attribute คือ Attribute ที่ไม่ใช่ส่วนใดส่วนหนึ่งของ Primary key

3. Third Normal Form:3NF(BCNF) นิยาม Relation ใดๆ อยู่ในรูปของ 3NF ถ้า Relation นั้นๆเป็น 2NF และ Attribute ทุกตัวที่ไม่เป็น Primary key หรือส่วนประกอบของ Primary key ไม่มีคุณสมบัติในการกำหนดค่า Attribute อื่นๆ

4. Fouth Notmal Form:4NF นิยาม Relation ใดๆ อยู่ในรูปของ 4NF ถ้า Relation อยู่ในรูปแบบ 3NF และเป็น Relation ที่ไม่มีการขึ้นต่อกันเชิงกลุ่ม

นิยามใน Relation ที่ประกอบด้วย Attribute3 Attribute คือ A,B, และC การขึ้นต่อกันเชิงกลุ่มระหว่างB และ Aโดย B ขึ้นต่อ A หมายความว่าค่า1 ค่าของ A จะจับคู่กับ B ได้หลายค่าและเป็นอิสระกับ C

2.10 ขั้นตอนการออกแบบฐานข้อมูลโดยใช้โมเดล [1]

2.10.1 ศึกษาระบบงาน

ผู้ออกแบบฐานข้อมูลจะต้องศึกษาระบบงานที่จะจัดทำฐานข้อมูลก่อน โดยการศึกษาจะต้องทราบถึงลักษณะหน้าที่ของระบบงาน (Bussiness Function) ขั้นตอนการทำงานต่างๆ (Bussiness Rule) ซึ่งจะนำไปใช้ในการกำหนดความสัมพันธ์และควบคุมความคงสภาพของข้อมูล

2.10.2 กำหนดกลุ่มของข้อมูล (Entity)

เมื่อผู้ออกแบบได้ศึกษาระบบงานต่างๆ แล้วก็นำมากำหนดกลุ่มข้อมูลที่สนใจและต้องการจัดเก็บเป็นข้อมูลในระบบ

2.10.3 กำหนดความสัมพันธ์ระหว่างกลุ่มข้อมูล (Relationship)

เมื่อได้กลุ่มข้อมูลที่สนใจก็ทำการกำหนดความสัมพันธ์ตามกฎเกณฑ์ ต่างๆที่ได้ศึกษามา

2.10.4 กำหนดรายละเอียดของเอ็นติตี้ (Attribute)

ในการกำหนดเอ็นติตี้นั้นจะมีการกำหนดรายละเอียดของเอ็นติตี้ที่เก็บไว้เป็นฐานข้อมูล ซึ่งในการกำหนดรายละเอียดนี้เราก็สามารถได้เอ็นติตี้ที่สนใจเพิ่มขึ้นโดยกำหนดจากรายละเอียดที่มีข้อมูลมากและไม่สม่ำเสมอ จะเกิดเป็น Weak Entity ขึ้นมาอีกเอ็นติตี้

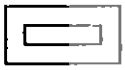
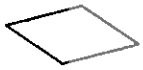
2.10.5 กำหนดคีย์หลัก(Primary key)

เมื่อกำหนดแอททริบิวในเอ็นติตี้ต่างๆ แล้วจะต้องทำการกำหนดแอททริบิวที่มีคุณสมบัติเป็นคีย์หลักคือ เป็นค่าไม่ซ้ำ มีค่าแน่นอนและไม่เป็นค่าว่าง บางครั้งก็ต้องกำหนดคีย์นอกให้ชัดเจนด้วย

2.10.6 สร้างโมเดลจำลองความสัมพันธ์ (Model)

เมื่อได้ข้อมูลต่างๆแล้ว ก็ทำการสร้างโมเดลจำลองความสัมพันธ์ขึ้นมานอกจากจะเป็นการจำลองระบบฐานข้อมูลแล้ว การนำมาเขียนเป็นโมเดลนี้ยังเป็นการทบทวนความถูกต้อง และความชัดเจนของฐานข้อมูลนั้นๆอีกด้วย ซึ่งในขั้นตอนนี้ก็เป็นการนำข้อมูลที่ได้จากขั้นตอนทั้งหมดมาเขียนให้อยู่ในรูปสัญลักษณ์ต่างๆที่ใช้ ก็ขึ้นกับว่าจะใช้โมเดลใด

ตารางที่ 2.1 สัญลักษณ์ ER

สัญลักษณ์	ความหมาย
	Entity
	Weak Entity
	ประเภทของความสัมพันธ์ระหว่าง Entity

2.11 การติดต่อกับฐานข้อมูล

ฐานข้อมูล คือ การเก็บข้อมูล(DATA) ไว้สำหรับการใช้งานร่วมกัน โดยปกติจะอยู่ในรูปแบบของตารางฐานข้อมูล (TABLE) ซึ่งมีไว้หลายตาราง ซึ่งในแต่ละตารางจะประกอบด้วยแถว (ROW) และคอลัมน์ (COLUMM) แต่ละแถวหมายถึงหนึ่งระเบียน(Record) ของข้อมูลในตาราง และแต่ละคอลัมน์ก็คือฟิลด์(Field) ต่าง ๆในตาราง

2.11.1 ประเภทของระบบฐานข้อมูล

1. ฐานข้อมูลแบบเชิงชั้น (Hierarchical Database) เป็นฐานข้อมูลที่คิดค้นโดยที่บริษัทไอบีเอ็ม (IBM) มีโครงสร้างเป็นชั้น ๆ เช่นเดียวกับผังการบังคับบัญชาขององค์การ (Organization Chart) การเชื่อมโยงข้อมูลจะใช้ตัวชี้ (Pointer) เพื่อโยระหว่างข้อมูลที่มีความสัมพันธ์กัน ตัวอย่างฐานข้อมูลประเภทนี้ได้แก่ IMS ซึ่งทำงานอยู่ในเครื่องเมนเฟรม

2. ฐานข้อมูลแบบโครงข่าย (Network Database) เป็นฐานข้อมูลที่มีโครงสร้างที่ซับซ้อนกว่าฐานข้อมูลแบบเชิงชั้น เป็นฐานข้อมูลที่คิดขึ้นมาเพื่อปรับปรุงข้อด้อยของฐานข้อมูลแบบเชิงชั้น ตัวอย่างฐานข้อมูลประเภทนี้ได้แก่ IDEMS

3. ฐานข้อมูลเชิงสัมพันธ์ (Relational Database) เป็นฐานข้อมูลที่คิดค้นโดย Dr.Codd ในปี 1970 ซึ่งมีจุดประสงค์ในการออกแบบ คือ เพื่อให้การจัดโครงสร้างของข้อมูลมีความง่าย โดยการเชื่อมโยงข้อมูลระหว่างตารางจะใช้ฟิลด์ข้อมูลแทนการใช้ตัวชี้ (Pointer) ที่ใช้ในระบบฐานข้อมูลทั้งสองแบบแรก การทำเช่นนี้ทำให้ง่ายในการจัดการฐานข้อมูล และง่ายในการเรียนรู้ที่จะใช้งาน ทำให้ระบบฐานข้อมูลประเภทนี้นิยมใช้กันแพร่หลายในปัจจุบัน โดยที่ระบบฐานข้อมูลทั้งสองแบบแรกแทบจะไม่มีการพัฒนา และไม่มีการผลิตโปรแกรมใหม่ ๆ ออกมาเลย

2.11.2 ส่วนประกอบของฐานข้อมูลเชิงสัมพันธ์

ดิกชันนารีข้อมูล (Data Dictionary / Directory หรือ DD/D) บางครั้งเรียกว่า Mata Data ซึ่ง เป็นส่วนที่ใช้เก็บนิยามเพื่ออธิบายองค์ประกอบของข้อมูล (Database) ที่จัดเก็บไว้ในระบบ

ข้อมูล (Database) เป็นส่วนหนึ่งที่เก็บข้อมูลต่าง ๆที่ผู้ใช้ต้องการ

Database Engineเป็นซอฟแวร์ที่บริหารฐานข้อมูล ทำหน้าที่บริหารระบบโดยรับความต้องการจากผู้ใช้งานแล้วแปลความหมายของคำสั่ง จากนั้นทำการปรับปรุงข้อมูลในฐานข้อมูล (Database) โดยใช้ข้อมูล Meta Data จากดิกชันนารีข้อมูล ส่วนประกอบของฐานข้อมูลเชิงสัมพันธ์ ข้อมูลที่จัดเก็บไว้ในฐานข้อมูลจะอยู่ในรูปแบบโครงสร้างคล้ายกับโครงสร้างของแฟ้มข้อมูล โดยที่ในระบบฐานข้อมูลจะจัดเก็บข้อมูลในลักษณะที่แยกฐานข้อมูลออกจากกัน จำนวนฐานข้อมูลจะมากหรือน้อยขึ้นกับการออกแบบระบบของนักวิเคราะห์ระบบ ภายในฐานข้อมูลจะประกอบด้วยตาราง (Table) หลาย ๆ ตาราง แต่ละตารางประกอบด้วยระเบียน (Record) และฟิลด์ (Field) นอกจากการจัดเก็บตารางแล้วฐานข้อมูลยังสามารถจัดเก็บข้อมูลอื่น ๆ ที่ช่วยในการทำงานได้ด้วย

2.12 ความสัมพันธ์ระหว่าง เดลไฟ(Delphi) กับฐานข้อมูล [4]

การติดต่อกับฐานข้อมูลนั้น เดลไฟ(Delphi) มีคอมโพเนนต์ที่สามารถเชื่อมต่อ เพื่อจัดการกับฐานข้อมูลที่เก็บอยู่ในฐานข้อมูลทุกประเภท ไม่ว่าจะเป็นการเพิ่ม ลบ แก้ไข หรือการเรียกข้อมูล โดยผู้ใช้ไม่ต้องเขียนชุดคำสั่งใด ๆ ในโปรแกรมเลย ก็สามารถสร้างแอพพลิเคชันอย่างง่าย ๆ ที่ทำงานกับฐาน

ข้อมูลขึ้นมาได้สามารถใช้ Delphi จัดการกับฐานข้อมูลที่เป็นแบบง่าย ๆ ซึ่งได้แก่ ดีเบส (dBASE) หรือ พาราด็อกซ์ (Paradox) และ ไมโครซอฟแอคเซส (MS Access) ซึ่งพวกนี้จะเรียกว่าเป็น Local Database คือ เป็นฐานข้อมูลที่ทำงานในเครื่องนั่นเอง หรือเรียกอีกอย่างหนึ่งว่า เป็นแบบ File_oriented Database ก็ได้ เพราะเป็นการเก็บข้อมูลไว้ในโครงสร้างของไฟล์ นอกจากนี้ยังสามารถใช้งานกับระบบฐานข้อมูลที่เป็น Database Server ต่าง ๆเช่น SQL Server หรือ InterBase ซึ่งอาจจะทำงานอยู่ในเครื่องเดียวกันกับแอพพลิเคชันในกรณีของ Local Interbase หรืออาจจะทำงานอยู่บนเครื่อง Server เครื่องอื่น ซึ่งเรียกว่าเป็นการใช้ ฐานข้อมูลในแบบ Client/Server ก็ได้ ใช้งานฐานข้อมูลของ Dephi นั้น จะใช้ Borland Database (BDE) เป็นตัวจัดการในการเข้าถึงข้อมูล ไม่ว่าจะเป็นข้อมูลแบบ Local ที่เป็นไฟล์ในเครื่องที่ทำงานนั้นอยู่เอง หรือ Client/Server ที่อยู่ต่างเครื่องก็ตาม ล้วนแล้วแต่จะต้องใช้ BDE ในการเข้าถึงข้อมูลทั้งสิ้น ความสัมพันธ์ระหว่างแอพลิเคชันที่สร้างด้วย Delphi กับฐานข้อมูลที่เชื่อมต่อกันด้วย BDE

2.12.1 เครื่องมือในการทำงานกับฐานข้อมูล

เดลไฟ มีเครื่องมือที่ช่วยให้เราสามารถทำงานกับฐานข้อมูลได้ง่ายและสะดวก ทั้งในส่วนของการเชื่อมต่อกับฐานข้อมูล การสร้างฐานข้อมูล และการจัดการฐานข้อมูล เครื่องมือเหล่านี้ได้แก่ BDM (Borland Database Engine) ,IDAPI (Borland Independent Database Application Programming Interfage),Database Decktop(DBD),Data pump และ SQL Monitor

2.12.2 การติดต่อกับฐานข้อมูลด้วย BDE เดลไฟ (Dephi)

สามารถติดต่อกับฐานข้อมูลได้หลายวิธี สำหรับวิธีที่สะดวกที่สุดคือการติดต่อผ่านอินเตอร์เฟส (Interface) BDE (Borland Database Engine) ซึ่งจะทำหน้าที่เป็นตัวเชื่อม (interface) และล่าม(Translate) เพื่อทำการติดต่อกับฐานข้อมูลชนิดต่าง ๆ กัน เช่น ดีเบส (dBase) พาราด็อกซ์ (Paradox) และ SQL server เป็นต้น เดลไฟ มี BDE Administator เป็นแอพลิเคชันที่ใช้จัดการกับ BDE ไดรเวอร์ (Driver) ในการเชื่อมต่อกับฐานข้อมูลต่าง ๆเป็นต้น ซึ่งสามารถเปิด BDE Administator ซึ่ง BDE Administator แบ่งออกเป็น 2 ส่วน คือ All Database Aliasea (ด้านซ้าย) และDefinition (ด้านขวา) ทางด้านซ้ายประกอบไปด้วยเพจ 2 เพจ คือ เพจ Dataabase แสดงเอเลียส และเพจ Configuration แสดงค่าที่กำหนดของ BDE และไดรเวร์ของฐานข้อมูลต่าง ๆ

2.12.3 BDE (Borland Database Engine)

เป็นเครื่องมือที่ถูกออกแบบมาเพื่อช่วยให้การพัฒนาแอพลิเคชันบน Window สำหรับภาษา C C++ และDephi สามารถใช้กับฐานข้อมูลได้หลายประเภท เช่น ดีเบส (dBase) พาราด็อกซ์ (Paradox) ไมโครซอฟต์เอกเซส (Microsofe Access) โอราเคิล (Oracle) ซิสเบส (Sysbase) อินเตอร์เบส (Interbase) และอินฟอร์มิกซ์ (Informix) เป็นต้น โดยที่ พาราด็อกซ์ (Parsadox) และดีเบส (dBase) เป็นฐานข้อมูลมาตราฐานในการใช้งาน สามารถอธิบายความสัมพันธ์ระหว่างแอพลิเคชัน BDE และฐานข้อมูล

2.12.4 IDPI (Borland Independent Database Appication Programming Interface)

ย่อมาจาก Borland Independent Database Appication Programming Interface หมายถึง โปรแกรมซึ่งทำหน้าที่ประสานการทำงานระหว่าง BDE กับ แอพลิเคชันที่เราสร้างขึ้นด้วย Dephi ในการติดต่อและดำเนินการกับฐานข้อมูล (Database) แต่ละชนิดนั้น แอพลิเคชันในเดลไฟ ไม่ได้ติดต่อกับฐานข้อมูล (Database)นั้นโดยตรง แต่จะติดต่อผ่าน BDE โดยมี IDAPI ทำหน้าที่ประสานงาน ดังนั้นชนิดของฐานข้อมูล (Database) ที่เดลไฟ ดำเนินการได้จึงขึ้นอยู่กับเวอร์ชันของ BDE

2.12.5 Database Desktop

เป็นแอพลิเคชันในกลุ่มของเดลไฟ ใช้แสดง จัดเรียง แก้ไข ค้นหาข้อมูลจากตาราง รวมทั้งสร้างตารางข้อมูล โดยรูปแบบของ ตารางที่สามารถสร้าง ได้มีด้วยกันหลายรูปแบบ เช่นพาราด็อกซ์ (Paradox) ดีเบส (dBase) หรือ SQL เป็นต้น การเปิด Database Decktopทำได้โดยเลือกเมนูของ Tool > Database Decktop เมื่อเลือกแล้วจะปรากฏวินโดว์ของ Database Decktop

2.12.6 เอเลียส (Alias)

คือชื่อเราตั้งขึ้นเพื่อใช้แทนไดเร็กทอรี เช่น ถ้าเราต้องการใช้งานไดเร็กทอรี " C:\Program Files\Common Files\Borland Shared\Data\" จะเห็นได้ว่า ชื่อไดเร็กทอรีนี้ยาว เราจึงสร้างเอเลียส ขึ้นมาเพื่อใช้แทนไดเร็กทอรีดังกล่าวใน Database Decktop จะมี Alias Manager ที่ใช้ในการจัดการกับเอเลียส (Alias) หรือ ตาราง เป็นต้น รวมทั้งจัดการข้อมูลในตารางได้ด้วย นอกจากนั้นแล้วยังใช้คำสั่ง SQL ในการดึงข้อมูลออกมาแสดงได้ตามต้องการ

2.12.7 คิวรี (Query)

เป็นวิธีการแสดงกลุ่มของข้อมูลที่ต้องการจากตาราง โดยจะอยู่ในรูปของคำถาม เช่น หนังสือเล่มใดขายดีที่สุด และขายได้กี่เล่ม หรือ จำนวนหนังสือของสำนักพิมพ์หนึ่งมีทั้งหมดกี่เล่ม เป็นต้น นอกจากจะได้ข้อมูลที่ต้องการแล้ว ยังสามารถจัดการกับข้อมูลที่ได้ เช่น เรียงลำดับ หาค่าเฉลี่ย หาค่าสูงสุด หรือค่าต่ำสุด เป็นต้น

2.12.8 วิธีการสร้างแอพลิเคชันฐานข้อมูลจากคอมโพเนนต์

ก่อนการสร้างแอพลิเคชันฐานข้อมูล เราควรทำความเข้าใจหลักการทำงานของคอมโพเนนต์ต่าง ๆก่อน เนื่องจากคอมโพเนนต์ที่ใช้กับฐานข้อมูลนั้นมีอยู่ด้วยกัน 2 ส่วน คือคอมโพเนนต์ในเพจ Data Access กับคอมโพเนนต์เพจ Data Controlสามารถอธิบายความสัมพันธ์ระหว่าง คอมโพเนนต์จากทั้ง 2 เพจ และการติดต่อกับฐานข้อมูล ได้ ส่วนที่ติดต่อกับฐานข้อมูลโดยตรงคือส่วนของ Dataset ซึ่งเป็นคอมโพเนนต์ที่ทำหน้าที่ดึงกลุ่มของข้อมูลจากตารางที่เราสนใจในฐานข้อมูล โดยคอมโพเนนต์ที่เป็น Dataset ได้แก่คอมโพเนนต์ตาราง (Table)คอมโพเนนต์คิวรี(Query) คอมโพเนนต์ StroedProc

คอมโพเนนต์ NestedTable ซึ่งคอมโพเนนต์ทั้ง 4 ตัวนี้อยู่ในเพจ Data Access แต่ Dataset จะแสดงข้อมูลจากฐานข้อมูลไม่ได้ เนื่องจากเป็นคอมโพเนนต์แบบ Non-Visual แต่การสร้างแอพลิเคชันฐานข้อมูลส่วนมากจำเป็นต้องมีส่วนที่ใช้แสดงข้อมูล ดังนั้น Delphi 5.0 จึงมีคอมโพเนนต์ที่เราเรียกว่า Data-aware ซึ่งก็คือคอมโพเนนต์ในเพจ Data Control เป็นคอมโพเนนต์ที่สามารถนำข้อมูลจาก Dataset มาแสดงผลได้ แต่การนำข้อมูลมาแสดงนั้น จะต้องมีตัวกลางในการเชื่อมต่อระหว่างคอมโพเนนต์ของทั้ง 2 กลุ่ม ซึ่งเราเรียกว่า คอมโพเนนต์แหล่งข้อมูล (Data Source)

2.12.9 คอมโพเนนต์แหล่งข้อมูล (Data Soure)

เป็นคอมโพเนนต์ที่ใช้เชื่อมความสัมพันธ์ระหว่างกลุ่มข้อมูลกับคอมโพเนนต์ที่ใช้แสดงข้อมูล โดยที่คอมโพเนนต์ทุกตัวในเพจ Data Control จะต้องทำการอ้างถึงคอมโพเนนต์นี้เพื่อกำหนดแหล่งข้อมูลที่จะนำแสดงในคอมโพเนนต์นั้น ๆ

2.12.10 คอมโพเนนต์ฐานข้อมูล (Database)

ใช้จัดการกับฐานข้อมูล มีประโยชน์ในการควบคุมการเข้ามาใช้ข้อมูลในฐานข้อมูล โดยสามารถกำหนดให้มีการป้อนชื่อผู้ใช้และรหัสผ่านได้ การเชื่อมต่อคอมโพเนนต์นี้กับฐานข้อมูลเราจะเชื่อมต่อด้วยชื่อฐานข้อมูล หรืออาจเป็นเอเลียส(Aliase) ที่สร้างก็ได้

2.13 ASP(Active Server Page) คืออะไร [3]

ASP คือเทคโนโลยีในการพัฒนาแอพพลิเคชั่นสำหรับเวบเพจ ที่พัฒนาขึ้นโดยบริษัทไมโครซอฟท์ เป็นเทคโนโลยีที่ออกแบบเพื่อให้การทำงานด้านการสร้างแอพพลิเคชั่นบนเวบไซต์สามารถทำได้เร็วขึ้น ซึ่ง ASP จะมีจุดเด่นในการพัฒนาและจัดการแอพพลิเคชันบนเว็บเซิร์ฟเวอร์

ASP จะทำงานในลักษณะเป็นโปรแกรมภาษา (Interpreter) ที่ใช้ในการตีความเว็บเพจที่เขียนขึ้นมาโดยใช้ภาษา VBScript ,Jscript หรือ Perl ที่ประกอบด้วยส่วนที่เป็นการเขียนแบบ ASP ซึ่งบราวเซอร์ เช่น Internet Exproler หรือ Netscape Nevigator ไม่สามารถนำไปแสดงผลได้ เมื่อ ASP ตีความภาษาส่วนนี้แล้วก็จะส่งผลลัพธ์ไปเป็นเอกสารที่อยู่ในรูปแบบ HTML ไปยังบราวเซอร์ ซึ่งบราวน์เซอร์จะนำส่วนนี้ไปแสดงผลต่อไป

Active Server Pages เป็นเทคโนโลยีในฝั่งของเซิร์ฟเวอร์ ดังนั้นออบเจกต์ต่างๆ ของ Active Server Pages จะทำงานอยู่ที่ฝั่งเซิร์ฟเวอร์ นั่นคือการประมวลผล การเรียกใช้งานต่างๆ จะเกิดขึ้นบนเซิร์ฟเวอร์เท่านั้น มีแต่เพียงผลลัพธ์การทำงานเท่านั้นที่ส่งออกไปยังไคล์เอ็นต์ที่เรียกใช้งาน

2.13.1 หลักการทำงานของ ASP

เนื่องจาก ASP จะทำงานโดยมีตัวแปรและเอ็กซ์คิวต์ที่ฝั่งเซิร์ฟเวอร์ อาจเรียกการทำงานว่าเป็นเซิร์ฟเวอร์ไซด์ (Server Side) โดยการทำงานจะเริ่มต้นที่ผู้ใช้ส่งความต้องการผ่านเว็บบราวเซอร์ทาง HTTP (HTTP Request) ซึ่งอาจจะเป็นการกรอกแบบฟอร์ม หรือใส่ข้อมูลที่ต้องการ ข้อมูลเหล่านั้นจะเป็นเอกสาร ASP (เอกสารนี้จะมีส่วนขยายเป็น asp เช่น search.asp เป็นต้น) เมื่อเอกสาร ASP เข้ามาถึงเว็บเซิร์ฟเวอร์ก็จะถูกส่งไปให้ ASP จะทำหน้าที่แปลคำสั่ง (โดย ASP Interpreter) แล้วเอ็กซ์คิวต์คำสั่งนั้น ซึ่ง ASP อาจเรียกใช้ออบเจ็กต์ คอมโพเนนต์ หรือ ADO (เพื่อใช้ฐานข้อมูล) หลังจากนั้น ASP จะสร้างผลลัพธ์ในรูปเอกสาร HTML ส่งกลับไปเว็บเซิร์ฟเวอร์เพื่อส่งต่อไปให้บราวเซอร์แสดงผลทางฝั่งผู้ใช้ต่อไป (HTTP Response) ซึ่งลักษณะการทำงานแบบนี้คล้ายกับการทำงานของ CGI (Common Gateway Interface) หรืออาจกล่างได้ว่า ASP ก็คือโปรแกรม CGI ประเภทหนึ่งก็ได้

2.13.2 การเขียนสคริปต์ในรูปแบบของ Active Server Pages

สำหรับการเขียนสคริปต์ในรูปแบบของ Active Server Pages เราสามารถใช้ได้ตั้งแต่ NotePad , เวิร์ดโพรเซสเซอร์ หรือแม้กระทั่งเครื่องมือพัฒนาแอพพลิเคชันสำหรับอินเตอร์เน็ตโดยเฉพาะ เช่น Visusl Interdev

สำหรับสคริปต์ที่ใช้เขียนนั้น ASP ได้นั้น ณ ขณะนี้สามารถใช้ได้ 2 ภาษา ได้แก่ VBScript กับ Jscript (เป็นต้น Java Script ในเวอร์ชันของไมโครซอฟท์)

2.13.3 การเขียนสคริปต์ในรูปแบบของ Active Server Page

สำหรับการเขียนสคริปต์ในรูปแบบของ Active Server Page เราสามารถใช้ได้ตั้งแต่ NotePad, เวิร์ดโพรเซสเซอร์ หรือแม้กระทั่งเครื่องมือพัฒนาแอพพริเคชันสำหรับอินเทอร์เนตโดยเฉพาะ เช่น Visual Interdev

สำหรับสคริปต์ที่ใช้เขียนขึ้นนั้น ASP ได้นั้น ณ ขณะนี้สามารถใช้ได้ 2 ภาษา ได้แก่ VBScript กับ JScript (เป็น Java Script ในเวอร์ชันของไมโครซอฟท์)

แอพพลิเคชัน ASP นั้นจะเป็นแท็กซ์ไฟล์ที่มีนาใสกุลเป็น .asp ซึ่งจะประกอบด้วยข้อความ Tag ของ HTML และคำสั่งสคริปต์ (ตามแต่จะเลือกใช้ VBScript หรือ JScript)

แอพพลิเคชัน ASP จะมีใช้ <% กับ %> ปิดหัวท้ายสคริปต์ของ ASP (คล้าย ๆ กับการใช้ < และ > ปิดหัวท้าย Tag HTML) ซึ่งจะทำให้ ASP Interpreter รู้ว่านี่เป็นสคริปต์ของ ASP (ซึ่งจะถูกประมวลผลที่ Web Server เท่านั้น) แต่เราต้องการให้ผลลัพธ์ของการประมวลผลสคริปต์ที่เว็บเซิร์ฟเวอร์ไปแสดงผลที่บราวเซอร์ของผู้ใช้ เราจะปิดหัวท้ายของสคริปต์นั้นด้วย <%= กับ %>

จากตัวอย่างจะเห็นได้ว่าสคริปต์ของ ASP จะกระจายไปตลอดทั้งเอกสาร ทั้งส่วนที่เก็บ Sub Routine ของ VBScript (ในตอนต้น) ส่วนที่เป็น HTML (ซึ่งอยู่ตอนท้าย) จะสังเกตว่าแอพพลิเคชัน

ASP บางตัวอาจจะเริ่มด้วยการปิดหัวท้ายสคริปต์ด้วย <%@ กับ %> (แบบนี้เรียกว่า Directive) ซึ่งจะเป็นการบอกเว็บเซิร์ฟเวอร์ทราบว่านี่เป็นแอพพลิเคชัน ASP

2.13.4 การเพิ่มประสิทธิภาพแอพพลิเคชัน ASP ด้วย Sever Side Include (SSI)

นอกจากเราจะใช้ความสามารถของตัวภาษา,HTML หรือ ActiveX Control แล้วเรายังสามารถใช้ข้อมูลที่มีอยู่ที่เว็บเซิร์ฟเวอร์ มาเพิ่มประสิทธิภาพของแอพพลิเคชัน ASP โดยการสั่งให้เซิร์ฟเวอร์ทำงานบางอย่าง หรือส่งข้อมูลบางอย่างที่เราต้องการ แล้วเก็บผลการทำงานทั้งหมดลงใน HTML (ซึ่งก็จะถูกส่งให้บราวเซอร์)

-#include : แปะเพิ่มเข้าไปในเอกสาร HTML หรือในแอพพลิเคชัน ASP เราอาจจะต้องการแปะแท็กซ์ๆฟล์เข้าไปในเอกสารที่เป็นผลลัพธ์จากการประมวลผลที่ Web Server มีรูปแบบดังนี้

<!--#include file= "filename"-->สำหรับไฟล์ที่ถูกนำมาแปะเราสามารถเลือกได้จากไฟล์ไดเร็กทอรีปัจจุบัน

<!--#include file= "data.txt"-->ไฟล์จาก Virtual Directory จะต้องมีการระบุพาธโดยอ้างอิงจาก Virtual Directory

<!--#include file= "/script/tools/data.txt"-->-#config : ระบุรูปแบบการแสดงผล ที่ต้องการเราสามารถกำหนดรูปแบบของการแสดงผลไม่ว่าจะเป็น Error Message, ข้อมูลวันเวลาหรือขนาดของไฟล์ไปแสดงให้บราวเซอร์แสดงผลได้ตามที่ต้องการ ดังตัวอย่าง

<!--#config timefmt= "d/%m/%y%H:%M:%S"- - >-#echo : แสดงค่าตัวแปรของระบบบางตัวลงไปในเอกสาร HTML เป็นการแสดงค่าตัวแปรของระบบ เช่น วัน-เวลาที่เซิร์ฟเวอร์,ชื่อของเอกสาร,หรือข้อมูลที่ส่งจากบราวเซอร์ไปยังเว็บเซิร์ฟเวอร์ เป็นต้น

-#exec : รันแอพพลิเคชัน แล้วนำผลลัพธ์มาใส่ในเอกสาร HTML มี 2 รูปแบบ ได้แก่

1.สั่งให้รันคำสั่งกับไฟล์ Executable <!--#exec cmd= "ShowLocation.exe"--> สั่งรันไฟล์ ShowLocation.exe

2.สั่งให้รันแอพพลิเคชันของ ISAPI เช่นสั่งให้เอ็กซ์คิวต์แอพพลิเคชัน ASP ตัวอื่น ๆ (เพราะแอพพลิเคชัน ASP ก็ถือเป็นแอพพลิเคชัน ISAPI) <!--#exec cgi= "/Project/ASP/Greeting.asp"-->สั่งให้รันเอ็กซ์คิวต์ไฟล์ Greeting.asp

-#fastmod : แสดงวันเวลาที่แก้ไขไฟล์ล่าสุดลงไปในเอกสาร HTML ดังตัวอย่าง <!--#fastmod file= "Data.txt"--> แสดงวันเวลาแไขล่าสุดของไฟล์ Data.txt

-#fsize : แสดงขนาดของไฟล์ลงในเอกสาร HTML ดังตัวอย่าง <!--#fsize file= "Data.txt"--> แสดงขนาดไฟล์ Data.txt

2.13.5 การสร้างแอพพลิเคชัน ASP ด้วยออบเจ็กต์

เนื่องจาก ASP สร้างมาจากพื้นฐานของ COM (Component Object Model) ทุกสิ่งที่เป็นองค์ประกอบของ ASP จึงถูกมองว่าเป็นออบเจ็กต์ไปหมด โดยจะทำงานทั้งเซิร์ฟเวอร์ สำหรับออบเจ็กต์ของ ASP แบ่งออกเป็น 2 พวกคือ ออบเจ็กต์ที่ผู้ใช้สร้างขึ้นเอง และบิวต์อินออบเจ็กต์ (Buit-in Objects) สำหรับบิวต์อินออบเจ็กต์ใน ASP 3.0 มี 7 ตัว (เพิ่ม ASPError จากเดิมที่ ASP 2.0 ไม่มี) ดังนี้

-Application Object	เสมือนตัวแทนในการจัดการแอพพลิเคชัน ASP
-Session Object	จัดการผู้ใช้งานที่เข้าใช้งานแอพพลิเคชัน ASP
-Server Object	จัดการและบริหารทรัพยากรของเว็บเซิร์ฟเวอร์
-ObjectContext Object	จัดการเกี่ยวกับทรานแซคชั่น
-Respones Object	จัดการข้อมูลที่ถูกส่งจากเว็บเซิร์ฟเวอร์ไปยังบราวเซอร์
-Request Object	จัดการการข้อมูลจากบราวเซอร์
-ASPError Object	จัดการข้อมูลที่เป็นรายละเอียดของการเกิดข้อผิดพลาด

ภายในออบเจ็กต์จะประกอบไปด้วย คอลเล็กชั่น (Collection), พร็อพเพอร์ตี้ (Property), เมธอด (Method) และอีเวนต์ (Event) มากมาย ซึ่งสามารถหาศึกษาเพิ่มเติมได้ในหนังสือที่เกี่ยวกับ ASP ทั่วไป

2.13.6 ระดับการใช้งานตัวแปรในแอพพลิเคชัน ASP

การใช้งานตัวแปรในแอพพลิเคชัน ASP เราสามารถกำหนดขอบเขตให้แต่ละตัวได้ตามความเหมาะสม ซึ่งแบ่งได้เป็น 3 ขอบเขตหรือ 3 ระดับ คือ

1.Application Level จะมีขอบเขตครอบคลุมการใช้งานตลอดทั้งแอพพลิเคชัน มีสิทธิ์ใช้งานได้จากผู้ใช้งาน ซึ่งจะเป็นเหมือนการแชร์ข้อมูลระหว่างกัน

2.Session Level จะจำกัดขอบเขตการใช้งานอยู่เฉพาะแต่ละผู้ใช้งาน เป็นเหมือนสภาพแวดล้อมการทำงานของแต่ละผู้ใช้งาน

3.Pages Level จะจำกัดขอบเขตการใช้งานอยู่ในแต่ละเว็บเพจ บางทีถูกเรียกว่า Procedure Level สำหรับการใช้งานตัวแปรในแต่ละระดับ

2.13.7 การใช้ ASP ในการติดต่อฐานข้อมูลด้วย ADO (Active Data Object)

Active Data Object คือชุดของออบเจ็กต์ที่ถูกนำมาใช้ในการติดต่อกับ Database Server สำหรับ Active Server Pages (ASP) โดยจะใช้ ADODB สร้างออบเจ็กต์ที่เกี่ยวข้องกับการติดต่อและดึงข้อมูลจากฐานข้อมูลเพื่อมาใช้งาน โดยออบเจ็กต์ต่าง ๆที่เกี่ยวข้องกับ ADO โครงสร้างของ ADO Object โดยจะประกอบไปด้วย

-Connection Object

เป็นออบเจ็กต์แรกที่ถูกสร้างขึ้นสำหรับเชื่อมต่อกับฐานข้อมูล นอกจากนี้ยังสามารถส่งคำสั่งไปเอ็กซ์คิวต์เพื่อให้ได้ผลลัพธ์ โดยปกติแล้วจะได้เป็นเร็คคอร์ดกลับมา ซึ่งทำให้ Recordset Object ถูกสร้างโดยอัตโนมัติ Connection Object ประกอบด้วย

Error Object แทนข้อผิดพลาดแต่ละอย่าง หากการเชื่อมต่อทำไม่เสร็จ ข้อผิดพลาดเหล่านี้จะเกิดขึ้นและถูกรวมกันเป็น Error Collection, Property Object แทนคุณลักษณะแต่ละอย่างของ Connection Object ซึ่งรวมกันเป็น Properties Object, Command Objectเป็นออบเจ็กต์คำสั่งที่ถูกส่งไปเพื่อเอ็กซ์คิวต์ให้ได้เร็คคอร์ดเซ็ตกลับมาเป็นผลลัพธ์ Command Object มีความสารถในการกำหนดพารามิเตอร์ต่าง ๆ ไปกับการส่งคำสั่งปกติได้ Command Object ประกอบด้วย, Parameter Object แทนพารามิเตอร์แต่ละค่าที่ส่งไปกับคำสั่งเพื่อเอ็กซ์คิวต์ ซึ่งรวมกันเป็น Parameter Collection, Property Object แทนคุณลักษณะแต่ละอย่างของ Command Object ซึ่งรวมกันเป็น Properties Collection, Recordset Object

เป็นออบเจ็กต์ที่ได้มาจากการส่งคำสั่งเพื่อให้ได้เร็คคอร์ดเซ็ตเป็นผลลัพธ์กลับนอกจากนี้การสร้าง Command Object สามารถทำได้โดยตรง ซึ่งทำให้สามารถเปลี่ยนแปลงคุณลักษณะต่าง ๆให้เหมาะสมได้ Command Object ประกอบด้วย, Field Object แทนแต่ละคอลัมน์หรือฟิลด์ของผลลัพธ์ที่ได้ ซึ่งถูกรวมกันเป็น Field Collection นอกจากนี้ยังมี Properties Collection ของแต่ละฟิลด์, Property Object แทนคุณลักษณะแต่ละอย่างของ Recordset Object ซึ่งรวมกันเป็น Properties Collection

2.13.8 รูปแบบการเชื่อมต่อฐานข้อมูล

ในการติดต่อกับแฟ้มข้อมูลของ ASP นั้นจะอาศัย ADODB (Active Data Object Database) เพื่อติดต่อและจัดการกับข้อมูลในฐานข้อมูลนั้น สำหรับขั้นตอนการติดต่อแฟ้มข้อมูลของ ASP ที่ต้องการติดต่อกับแฟ้มข้อมูลจะต้องอาศัยไดว์เวอร์ของแฟ้มข้อมูลนั้นช่วย ซึ่งวิธีการติดต่อมี 3 วิธีคือ, ODBC (Open Database Connectivity), DSNLess, OLEDB

ในที่นี้จะขอกล่าวรูปแบบการติดต่อฐานข้อมูลเพียง 2 แบบเท่านั้นที่คิดว่าอยู่ในเกณฑ์ดีคือแบบ ODBC และ DSNLess โดยทั้ง 2 แบบใช้ ADODB

1. การติดต่อฐานข้อมูลโดยใช้ ODBC การติดต่อแบบนี้จะต้องมีการกำหนดชื่อ DSN (Data Source Name) ให้กับ ODBC โดยจะต้องกำหนดชื่อไฟล์ฐานข้อมูลและชนิดของไดร์เวอร์ว่าเป็นอะไร ข้อดีสำหรับการติดต่อแบบนี้จะสะดวกถ้าต้องการเปลี่ยนฐานข้อมูลจาก Access ไปเป็นอย่างอื่นเช่น SQL Server หรือ Oracle เป็นต้น เพราะว่าสามารถใช้ชื่อเดิมแต่เปลี่ยนคุณสมบัติภายในเท่านั้น หลังจากนั้นก็สามารถใช้งานได้ต่อไป โดยไม่จำเป็นต้องแก้โปรแกรมใหม่เพราะ ASP ไม่ได้สนใจว่าภายในใช้ไดร์เวอร์อะไรอยู่ ดังนั้นการเขียนคำสั่งเพื่อติดต่อฐานข้อมูลAccess หรือ SQL Server จะเขียนเหมือนกัน คือ

```
Set ObjDB=Server.CreateObject("ADODB.Connection")
ObjDB.Open DSN-Name, "" , ""
```

ห้องสมุดคณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

14941524

โดยที่ DSN-Name คือชื่อของ DSN ที่สร้างและกำหนดไดร์เวอร์แล้ว ส่วนข้อเสีย : คือ จะต้องไปเซ็ตอัพ ODBC และสร้าง DSN ที่เซิร์ฟเวอร์นั้น

2. การติดต่อฐานข้อมูลโดยใช้ DSNLess การติดต่อแบบนี้ไม่ใช้ DSN นั่นหมายความว่า ไม่จำเป็นต้องสร้าง DSN หรืออาจกล่าวได้ว่าไม่มีการติดต่อกับ ODBC นั่นเอง ข้อดี ของการติดตั้งแบบนี้ก็คือจะตัดขั้นตอนการเซ็ตอัพ ODBC ทั้งหมด ทำให้สะดวกในการใช้งาน ไม่จำเป็นต้องเดินทางไปเซ็ตอัพที่เซิร์ฟเวอร์ ส่วนข้อเสีย คือ ถ้าต้องการเปลี่ยนชนิดของฐานข้อมูล เช่น จาก Access เป็น SQL Server จำเป็นจะต้องไปแก้ไขรูปแบบการติดต่อทุกไฟล์ที่กล่าวถึงการติดต่อนั้น สำหรับรูปแบบการติดต่อแบบ DSNLess กับฐานข้อมูลที่สำคัญ ๆมีดังนี้

ม5.
ป/458
254

ตารางที่ 2.2 รูปแบบการติดต่อแบบ DSNLess กับฐานข้อมูล Access และ SQL Server

Access	Set ObjDB=Server.CreateObject("ADODB.Connection") ObjDB.Open "DRIVER={Microsoft Access Driver (*.mdb)};DBQ=path and Database" โดยที่ path and Database คือ พาธและตำแหน่งของไฟล์ฐานข้อมูล
SQL Server	Set ObjDB=Server.CreateObject("ADODB.Connection") ObjDB.Open "DRIVER={SQL Server}; SERVER=server-name;DATABASE=database-name UID=user-name;PWD=passward" โดยที่ server-name คือ ชื่อเซิร์ฟเวอร์ หรือ IP ของเซิร์ฟเวอร์ก็ได้ database-name คือ ชื่อของฐานข้อมูล user-name คือ ชื่อของผู้ใช้ passward คือ รหัสผ่านของผู้ใช้

2.13.9 วิธีการเขียนโปรแกรมในการติดต่อกับฐานข้อมูล

การติดต่อกับฐานข้อมูลด้วย ADO ในแต่ละครั้งที่ทำการติดต่อกับฐานข้อมูลสิ่งที่ควรทำอันดับแรก คือการสร้างส่วนที่เชื่อมต่อ หรือ Connection โดยการสร้างออบเจ็กต์สำหรับการติดต่อ แล้วใช้เมธอด Open เปิดการเชื่อมต่อนั้น ดังตัวอย่าง

```
Set Conn=Server.CreateObject("ADODB.Connection")
Conn.open "mydsn", "admin", ""
```

ในบรรทัดแรกเป็นการสร้างออบเจ็กต์สำหรับการติดต่อชื่อ Conn ไว้ก่อน แล้วจึงเปิดการเชื่อมต่อนั้นด้วยเมธอด Open โดยมีการกำหนดการเชื่อมต่อกับ Data Source Name ชื่อ "mydsn" เข้าติดต่อด้วย Login Name คือ "admin" ซึ่งไม่มีรหัสผ่าน (Password) ในการติดต่อแสดงด้วย "" การเปิด Recordset ของ ADO หลังจากที่ได้สร้างออบเจ็กต์การติดต่อเรียบร้อยแล้ว เราสามารถสร้าง Recordset Object เพื่อเอ็กซ์คิวต์คำสั่งในการดึงข้อมูลจากฐานข้อมูลมาได้ ดังรูปแบบต่อไปนี้

```
Set Recordset=Server.CreateObject("ADODB.Recordset")
Recordset.Open
```

โดยที่ Recordset คือ ชื่อของตัวแปรที่ใช้แทน Recordset ของ ADO, Sourceคือ คำสั่ง SQL,ชื่อ Table,ชื่อ Stored Procedure ที่ใช้สร้าง Recordset, Connection คือ ชื่อของตัวแปรประเภท Object ที่ใช้แทนการติดต่อกับไฟล์ฐานข้อมูล, CursorType คือ ประเภทของ Cursor ของ Recordset ที่ต้องการสร้างขึ้น ซึ่งระบุหรือไม่ก็ได้, Locktype คือ ประเภทของการ Lock ข้อมูลของRecordset ที่ต้องการสร้างขึ้น ซึ่งระบุหรือไม่ก็ได้, Option คือค่าที่ใช้ระบุควบคู่ในส่วน Source เพื่อระบุการทำงานให้กับ OLEDB Provider ซึ่งระบุหรือไม่ก็ได้, การปิด Recordset ของ ADO, เมื่อต้องการปิด Recordset ให้ยกเลิกตัวแปรประเภท Object ที่แทนแต่ละ Recordset นั้นด้วยคำสั่งดังนี้, Recordset.Close โดยที่ Recordset คือ ชื่อตัวแปรประเภท Object ที่ใช้แทน Recordset ของ ADO การยกเลิกการติดต่อกับฐานข้อมูลด้วย ADO เมื่อต้องการยกเลิกการติดต่อกับฐานข้อมูล ให้ยกเลิกตัวแปรประเภท Object ที่แทนการติดต่อนั้นด้วย Method "Close" ด้วยคำสั่งดังนี้ "Conn.Close" โดยที่ Conn คือ ชื่อของตัวแปรประเภท Object ที่ใช้แทนการติดต่อกับฐานข้อมูล, รูปแบบในการอ้างอิงถึง Field ของ ADO สำหรับรูปแบบการอ้างอิงถึง Field ของ Recordset แบบ ADO สามารถทำได้ดังนี้ "Recordset(Field)" โดยที่ Recordset คือ ชื่อของตัวแปรประเภท Object ที่ใช้แทน Recordset ของ ADO,Field คือ ชื่อ Field ในฐานข้อมูลที่ถูกเปิดโดย Recordset, Method ที่ใช้ในการจัดการกับข้อมูลของ ADOเมื่อต้องการเพิ่ม แก้ไข และลบข้อมูลภายใน Recordset ของ ADO จะอาศัย Method ดังนี้ตาราง คำสั่งที่ใช้ในการเพิ่ม,บันทึก,ลบ Recordset ของ ADO

ตารางที่ 2.3 ชื่อของตัวแปรประเภท Object ที่ใช้แทน Recordset ของ ADO

ชื่อ	หน้าที่	รูปแบบของคำสั่ง
AddNew	เพิ่มข้อมูลให้กับ Recordset	Recordset,AddNew
Update	บันทึกข้อมูลให้กับ Recordset	Recordset,Update
Delete	ลบข้อมูลออกจาก Recordset	Recordset,Delete

โดยที่ Recordset คือ ชื่อของตัวแปรประเภท Object ที่ใช้แทน Recordset ของ ADO,การเลื่อนตำแหน่งของ Record ใน ADO การเลื่อนตำแหน่ง (Pointer) ของ Record ใน Recordset ของ ADO จะอาศัย Method ดังนี้

ตารางที่ 2.4 การใช้คำสั่ง Move ต่างๆในการเขียน ASP

Method	หน้าที่
MoveFirst	ใช้เลื่อน Pointer ของ Record ใน Recordset ไปยัง Record แรกในRecordset
MoveLast	ใช้เลื่อน Pointer ของ Record ใน Recordset ไปยัง Record สุดท้ายในRecordset
MoveNext	ใช้เลื่อน Pointer ของ Record ใน Recordset ไปยัง Record ถัดไปในRecordset
MovePreviou	ใช้เลื่อน Pointer ของ Record ใน Recordset ไปยัง Record ก่อนหน้าในRecordset

ซึ่งทั้ง 4 Method นี้จะมีรูปแบบของคำสั่งเดียวกันดังนี้

Recordset.MoveFirst

Recordset.MoveLast

Recordset.MoveNext

Recordset.MovePrevious

โดยที่ Recordset คือ ชื่อของตัวแปรประเภท Object ที่ใช้แทน Recordset ของ ADO ทั้งหมดนี้ก็เป็นการใช้ ASP ในการติดต่อฐานข้อมูลผ่านทางอินเตอร์เน็ต โดยที่การเข้าถึงข้อมูลแบบ ADO นั้นจะต้องกระทำผ่านตัวแปรประเภท Object 2 ตัว ที่กำหนดขึ้นแทนการติดต่อกับข้อมูล และ Recordset ที่แทนข้อมูลใน Table ต่าง ๆ ซึ่งด้วยวิธีนี้ จะทำให้ตัวแปรประเภท Object ที่กำหนดขึ้นแทน Recordset สามารถเรียกใช้ Method และ Property ต่าง ๆของ Recordset ได้ทั้งหมด โดยที่ Recordset คือ ชื่อของตัวแปรประเภท Object ที่ใช้แทน Recordset ของ ADO,การเลื่อนตำแหน่งของ Record ใน ADO การเลื่อนตำแหน่ง (Pointer) ของ Record ใน Recordset ของ ADO จะอาศัย Method , รูปแบบในการอ้างอิงถึง

แทนข้อมูลใน Table ต่าง ๆ ซึ่งด้วยวิธีนี้ จะทำให้ตัวแปรประเภท Object ที่กำหนดขึ้นแทน Recordset สามารถเรียกใช้ Method และ Property ต่าง ๆของ Recordset ได้ทั้งหมด

บทที่ 3
การออกแบบฐานข้อมูล

3.1 เข้าใจในปัญหา (Problem Recognition)

1. การเก็บข้อมูลเกี่ยวกับวัสดุ และครุภัณฑ์จะเก็บ บันทึกและรายงานในกระดาษต่อการค้นหา การเก็บรักษาและการสำรองข้อมูล

2. ขั้นตอนการยืมครุภัณฑ์ ยากต่อการเข้าใจและปฏิบัติในทิศทางเดียวกัน

3. รายงานการตรวจสอบประจำปีมีรูปแบบที่ไม่เหมือนกัน ทำการตรวจสอบได้ยากในการทำงานจริง

4. กรณีครุภัณฑ์ชำรุด หากมีการใช้งาน อาจมีอันตรายได้

5. การจัดเก็บกรณีเคลื่อนย้ายครุภัณฑ์ จะไม่มีการบันทึก

3.2 ศึกษาความเป็นไปได้ (Feasibility Study)

จากปัญหาที่เกิดขึ้นคือ การทำงานด้านการจัดการฐานข้อมูลขั้นพื้นฐาน เกี่ยวกับวัสดุและครุภัณฑ์ของแต่ละภาควิชาไม่เป็นในทิศทางเดียวกัน ทำให้ยากต่อการค้นหา การตรวจสอบ หากมีการจัดการข้อมูลขั้นพื้นฐานในการจัดเก็บข้อมูล ประวัติครุภัณฑ์ การรายงานการตรวจสอบในทิศทางเดียวกัน จะทำให้ง่ายต่อการบริการและจัดการเกี่ยวกับครุภัณฑ์

กรณีที่ไม่ดำเนินการจัดการระบบการจัดการเกี่ยวกับวัสดุครุภัณฑ์ อาจเกิดอันตรายต่อผู้ใช้งาน การสูญหายของอุปกรณ์ การขาดแคลนอุปกรณ์ที่สำคัญในการเรียนการสอนเป็นต้น

3.3 การวิเคราะห์ (Analysis)

1. และครุภัณฑ์ การให้บริการการยืมคืนครุภัณฑ์ การรายงานการตรวจสอบประจำปี

2. ทำการรวบรวมจากพัสดุคณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร ในการเสนอซื้อครุภัณฑ์ เสนอซื้อวัสดุ เสนอจ้างซ่อมครุภัณฑ์ ซึ่งกระบวนการทางภาควิชาจะเสนอต่อทำการรวบรวมข้อมูลอาคารปฏิบัติการ เช่น จัดทำทะเบียนครุภัณฑ์ การจัดเก็บวัสดุพัสดุคณะ

3. ทำการรวบข้อมูลจากครูช่าง ในการจัดการเกี่ยวกับวัสดุและครุภัณฑ์

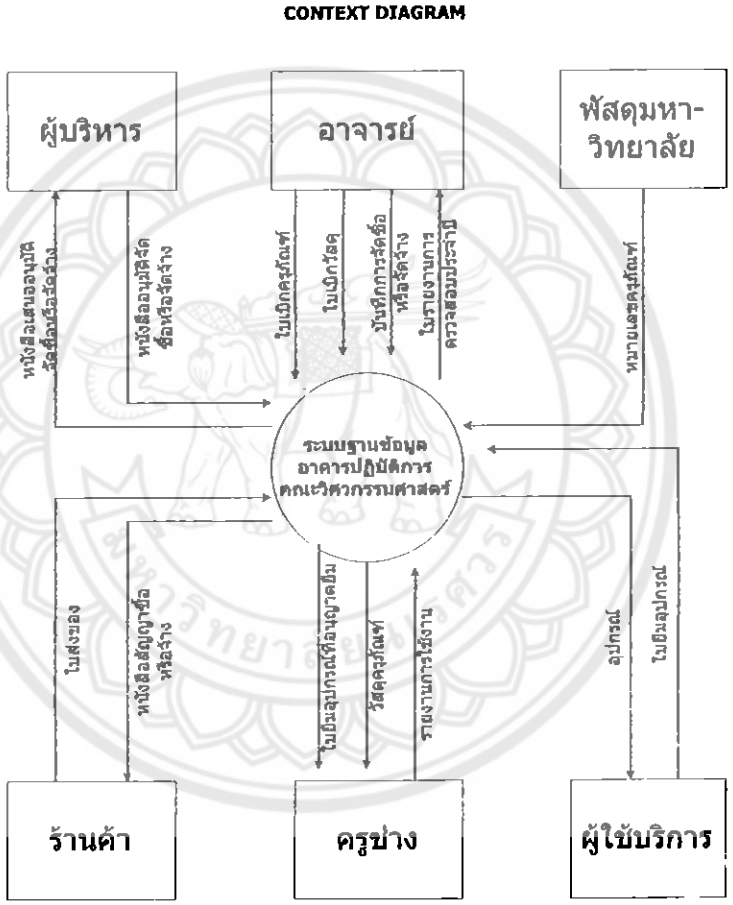
4. ทำการรวบรวมข้อมูลจากอาจารย์และนิสิต ซึ่งใช้งานวัสดุและครุภัณฑ์

3.4 การออกแบบ (Design)

3.4.1 รวบรวมข้อมูลและแบบฟอร์มตัวอย่าง

ของการรายงานทะเบียนประจำปี การยืมวัสดุและอุปกรณ์ของแต่ละภาควิชา ฟอร์มขั้นตอนการจัดซื้อวัสดุและครุภัณฑ์ การจ้างซ่อมจากพัสดุคณะวิศวกรรมศาสตร์ ออกแบบความต้องการของระบบ (Require-ment) โดยใช้ Data Flow Diagram ในการออกแบบและอธิบาย ซึ่งแบ่งเป็นขั้นและระบบได้ดังนี้

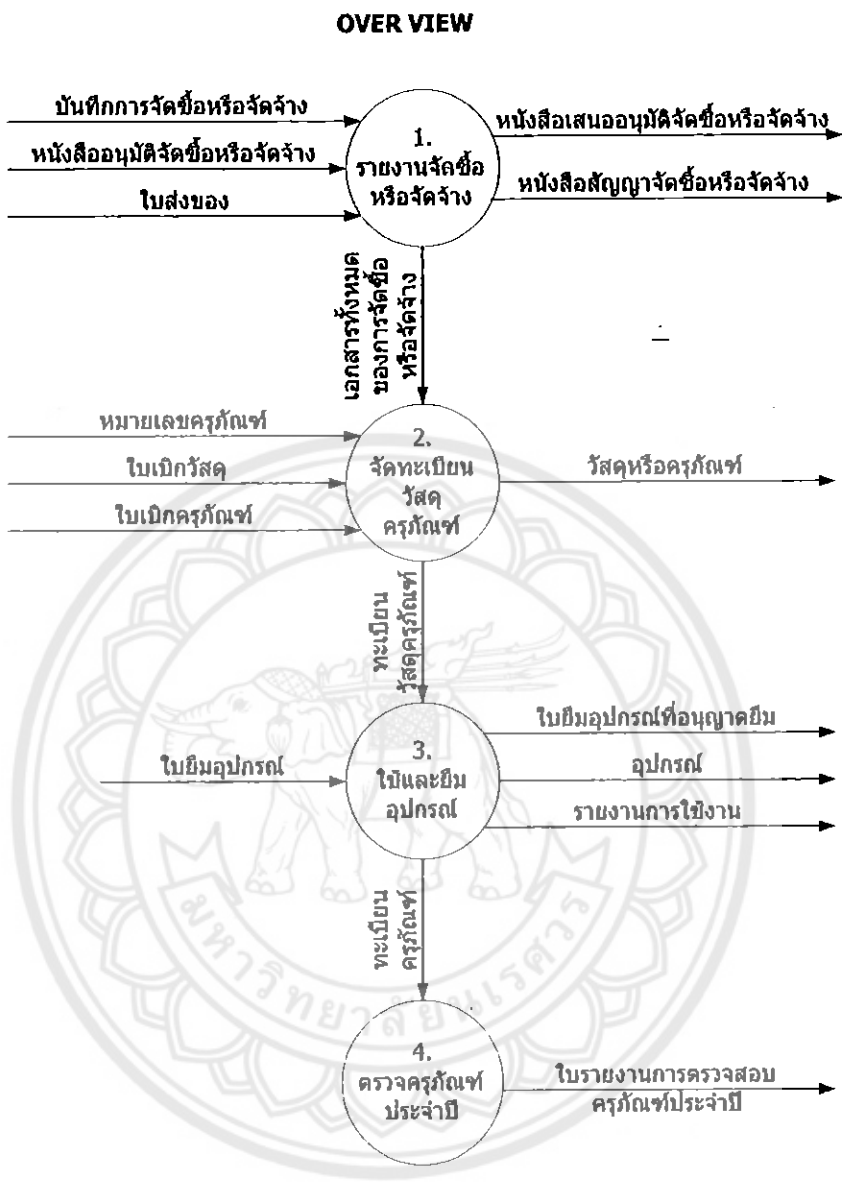
3.4.1.1 DFD ระดับที่ 0



รูปที่ 3.1 Context Diagram

แผนภาพการไหลของข้อมูล ของระบบสารสนเทศอาคารปฏิบัติการคณะวิศวกรรมศาสตร์ มหาวิทยาลัยขอนแก่น จะมองภาพรวมการทำงานของระบบ โดยมี output ที่อยู่นอกระบบ ประกอบด้วย ร้านค้า ครูช่าง ผู้ใช้บริการ ผู้บริหาร อาจารย์และพัสดุมหาวิทยาลัย

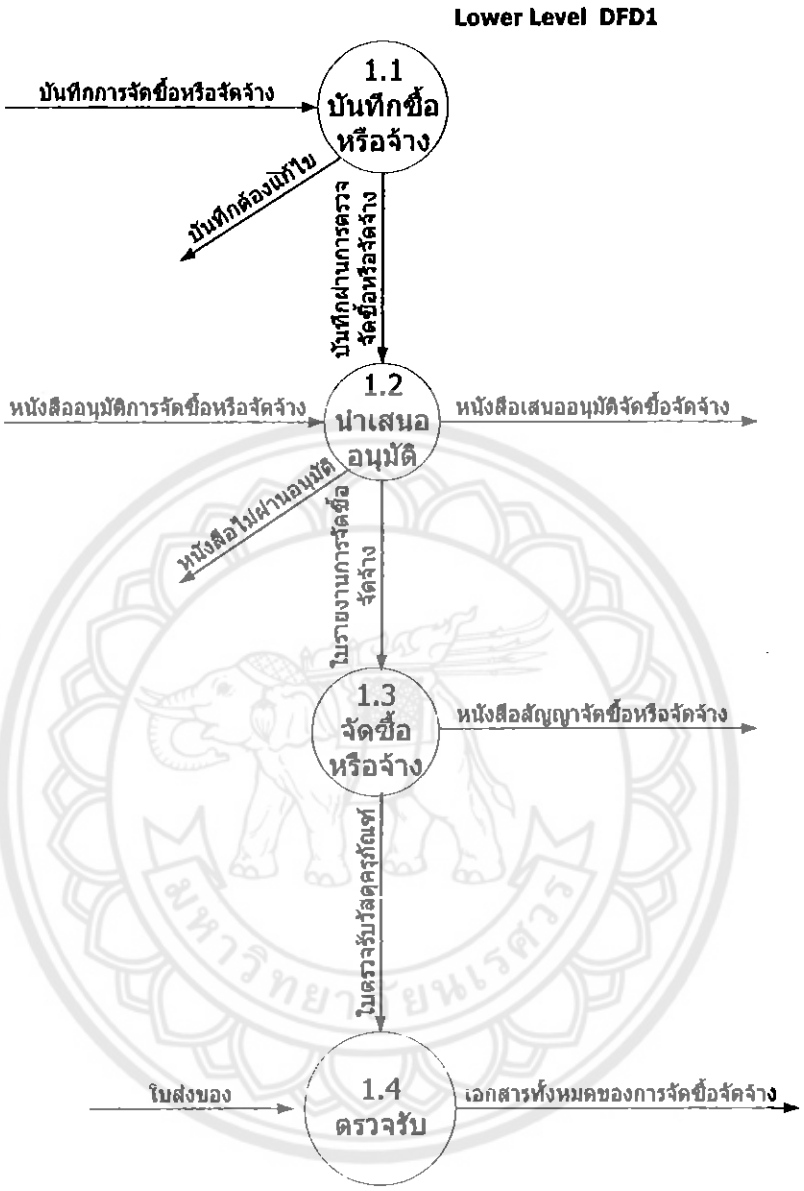
3.4.1.2 Process 0 Level 1



รูปที่ 3.2 แผนภาพ DFD ของกระบวนการ 0 ระดับที่ 1

แผนภาพการไหลของข้อมูล เป็นภาพการทำงานรวมของระบบในระดับที่ 1 นี้แสดงรายละเอียดของการะบวนการที่ 0 ในแผลภาพการไหลของข้อมูลในระดับที่ 0 ออกเป็น 4 กระบวนการย่อย ประกอบด้วย รายงานจัดซื้อหรือจัดจ้าง จัดทะเบียนวัสดุและครุภัณฑ์ ใช้และยืมอุปกรณ์ และตรวจครุภัณฑ์ประจำปี

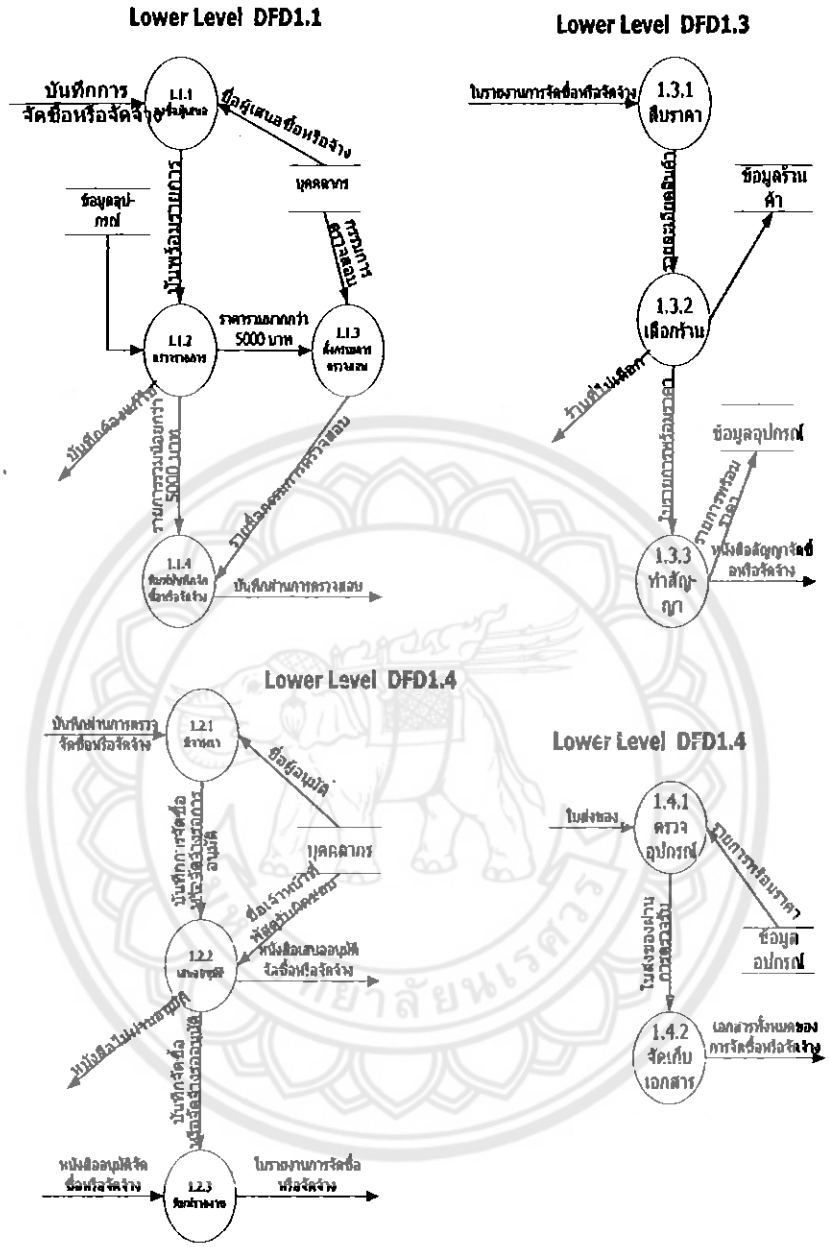
3.4.1.3 Process 1 Level 1



รูปที่ 3.3 แผนภาพ DFD ของกระบวนการที่ 1 ระดับที่ 1

จากแผนภาพการไหลของข้อมูลของกระบวนการที่ 1 ในระดับที่ 1 สามารถแบ่งแยกกระบวนการทำงานย่อยประกอบด้วย ทำการบันทึกซื้อหรือจ้าง นำเสนออนุมัติ ทำการจัดซื้อหรือจัดจ้าง และทำการตรวจรับ

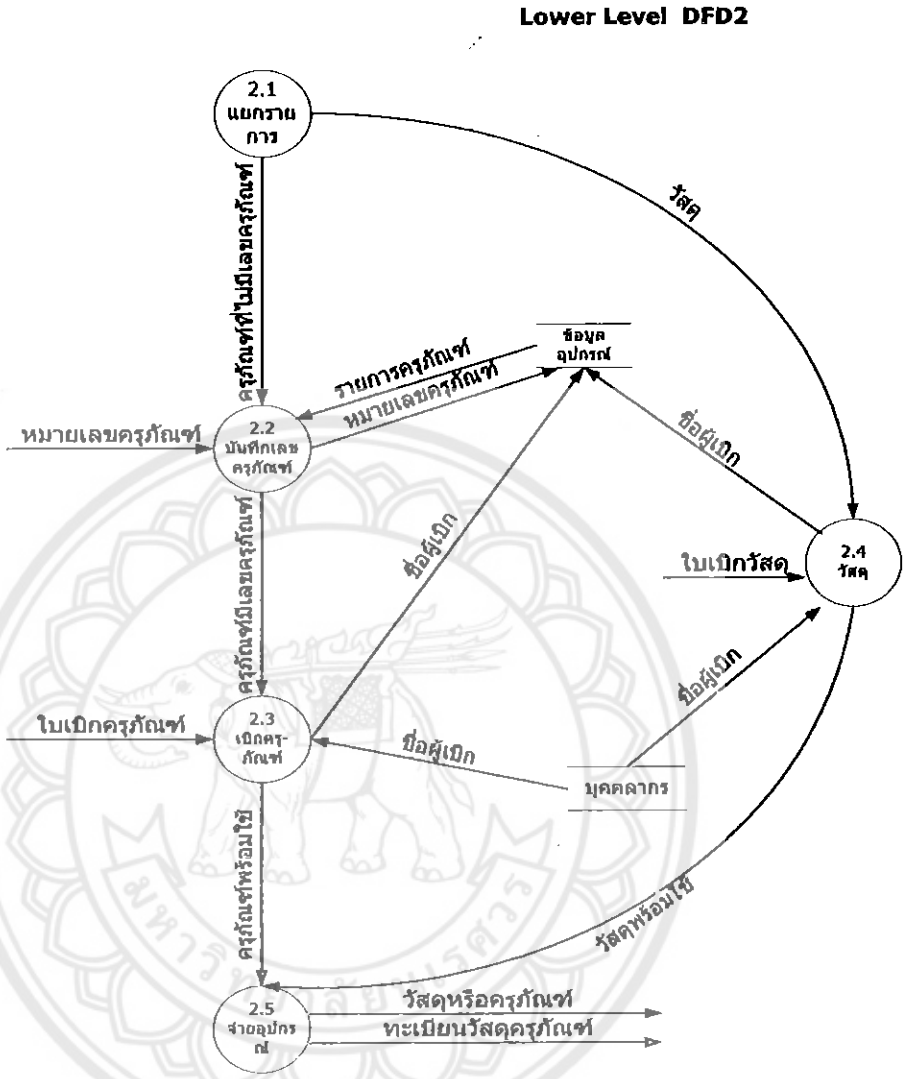
3.4.1.4 Process 1 Level 2



รูปที่ 3.4 แผนภาพ DFD ของกระบวนการที่ 1 ระดับที่ 2

เป็นการแตกรายละเอียดของ Process ที่ 1 ระดับที่ 1 เพื่อให้เห็นรายละเอียดมากขึ้น

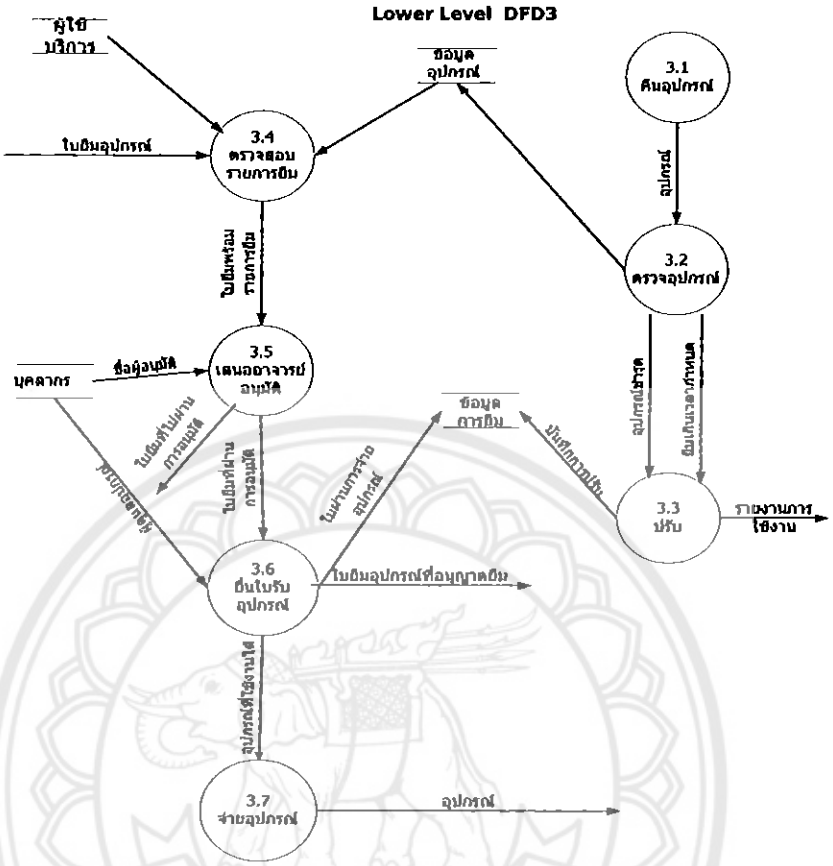
3.4.1.5 Process 2 Level 1



รูปที่ 3.5 แผนภาพ DFD ของกระบวนการที่ 2 ระดับที่ 1

จากแผนภาพการไหลของข้อมูลของกระบวนการที่ 2 ในระดับที่ 1 สามารถแบ่งแยกกระบวนการทำงานย่อยประกอบด้วย ทำการแยกรายการ บันทึกเลขครุภัณฑ์ ทำการเบิกครุภัณฑ์ ทำการเบิกวัสดุ และจ่ายอุปกรณ์

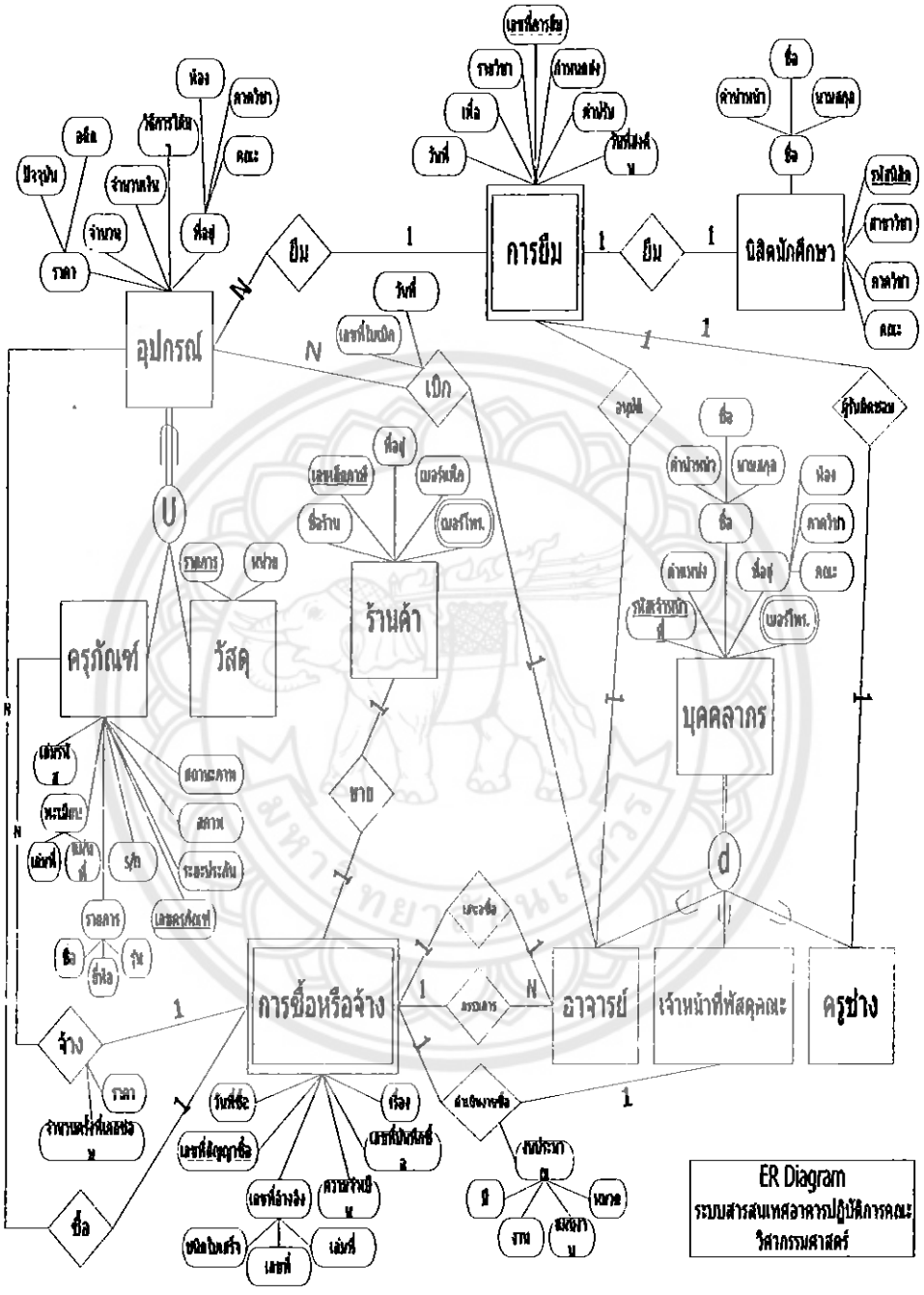
3.4.1.6 Process 3 Level 1



รูปที่ 3.6 แผนภาพ DFD ของกระบวนการที่ 3 ระดับที่ 1

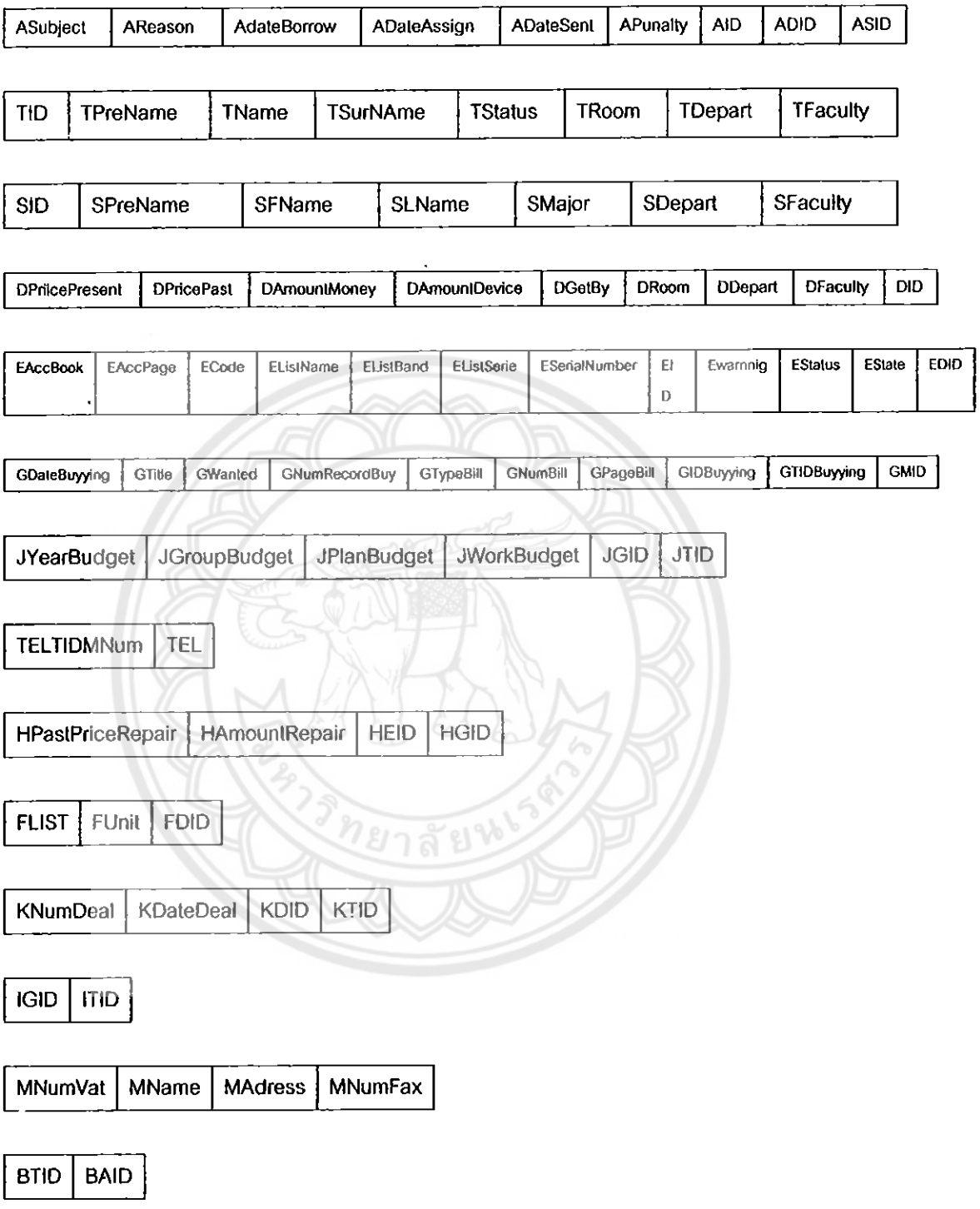
จากแผนภาพการไหลของข้อมูลของกระบวนการที่ 3 ในระดับที่ 1 สามารถแบ่งแยกกระบวนการทำงานย่อยประกอบด้วย คืนอุปกรณ์ ตรวจอุปกรณ์ การปรับ ตรวจสอบการยืม เสนออาจารย์อนุมัติและจ่ายอุปกรณ์

3.4.2 การออกแบบฐานข้อมูลจาก Requirement (Data Flow Diagram; DFD) โดยใช้ Entity-Relationship Model(ER Diagram) ในการออกแบบและอธิบาย



รูปที่ 3.7 ER Diagram ของฐานข้อมูลระบบสารสนเทศอาคารปฏิบัติการคณะวิศวกรรมศาสตร์มหาวิทยาลัยนเรศวร

จากรูปที่ 3.7 สามารถเขียนเป็น Relation Model ได้ดังนี้



รูปที่ 3.8 โครงสร้างของฐานข้อมูลอาคารปฏิบัติการคณะวิศวกรรมศาสตร์มหาวิทยาลัยนเรศวร

3.5 ทำการออกแบบฐานข้อมูล

ทำการออกแบบฐานข้อมูลระบบสารสนเทศอาคารปฏิบัติการคณะวิศวกรรมศาสตร์มหาวิทยาลัยนเรศวร โดยการสร้าง Data Description/Data Dictionary ดังแสดงในตารางที่ 3.1

ตารางที่ 3.1 Data Description/Data Dictionary

BORROW_COMMITTEE_TABLE.mdb

FIELDNAME	TYPE	DESCRIPTION
BCBID	Text(50)	เลขที่การยืม
BCPID	Text(50)	รหัสอาจารย์
BCName	Text(50)	ชื่อ-สกุล
BCSubject	Text(50)	ความรับผิดชอบ

BORROW_COMMITTEE_TABLE.mdb

FIELDNAME	TYPE	DESCRIPTION
BLDID	Text(50)	รหัสครุภัณฑ์
BLBID	Text(50)	รหัสใบยืม
BLName	Text(50)	ชื่อ,ยี่,ห้อรุ่น

BORROW_TABLE.mdb

FIELDNAME	TYPE	DESCRIPTION
Bsubject	Text(50)	รายวิชาที่เรียน
BdateBorrow	Date/Time	วันที่ยืม
BdateAssign	Date/Time	กำหนดส่งคืน
BdateSent	Date/Time	วันที่คืน
Bpunalty	Number	ค่าปรับ
BPIDBorrow	Text(50)	ชื่อผู้ยืม
BPIDPay	Text(50)	ชื่อผู้จ่าย
BMID	Text(50)	ภาควิชาที่ออกใบยืม
BdateLate	Number	จำนวนวันที่ส่งช้า
BcheckOld	Yes/No	ตรวจสอบสถานะ

ตารางที่ 3.1(ต่อ) Data Description/Data Dictionary

BUYYING_COMMITTEE_TABLE.mdb

FIELDNAME	TYPE	DESCRIPTION
GCGID	Text(50)	เลขที่บันทึกซื้อ
GCPID	Text(50)	รหัสอาจารย์
GCName	Text(50)	ชื่อ
GCSubject	Text(50)	ความรับผิดชอบ

BUYYING_LIST_TABLE.mdb

FIELDNAME	TYPE	DESCRIPTION
GLDID	Text(50)	รหัสรายการอุปกรณ์
GLName	Text(50)	รายการอุปกรณ์
GLGID	Text(50)	รหัสใบเลขที่บันทึกซื้อ
GLPricePerUnit	Number	ราคาต่อหน่วย
GLUnit	Text(50)	หน่วย
GLAmountUnit	Number	จำนวน
GLAmountMoney	Number	จำนวนเงิน

BUYYING_TABLE.mdb

FIELDNAME	TYPE	DESCRIPTION
GID	Text(50)	เลขที่บันทึกซื้อ
GdateBuy	Date/Time	วันที่ซื้อ
Glist	Text(50)	รายการพร้อมจำนวน
Gwanted	Text(50)	ความจำเป็นในการซื้อ
GtypeBill	Text(50)	ชนิดใบเสร็จ
GnumBill	Text(50)	เลขที่
GpageBill	Text(50)	เล่มที่
GproNum	Text(50)	เลขที่สัญญาซื้อ
GPIDBuy	Text(50)	รหัสผู้เสนอซื้อ

ตารางที่ 3.1(ต่อ) Data Description/Data Dictionary

BUYYING_TABLE.mdb(ต่อ)

FIELDNAME	TYPE	DESCRIPTION
GSID	Text(50)	รหัสร้านค้า
GsumMoney	Text(50)	ราคารวมการซื้อ
GMID	Number	รหัสภาควิชา
GCheck	Yes/No	ใบที่ผ่านการตรวจสอบ
Gagree	Yes/No	ใบที่ผ่านการเห็นชอบแล้ว

DEAL_TABLE.mdb

FIELDNAME	TYPE	DESCRIPTION
KGID	Text(50)	เลขที่ใบเบิก
KdateDeal	Date/Time	วันที่เบิก
KPID	Text(50)	รหัสผู้เบิก

DEVICE_E_TABLE.mdb

FIELDNAME	TYPE	DESCRIPTION
EDID	Text(50)	เลขครุภัณฑ์ทางภาควิชา
Ename	Text(50)	รายการขยาย
EserialNumber	Text(50)	S/N
EID	Text(50)	เลขครุภัณฑ์
Estate	Yes/No	สถานะภาพ(อยู่,ไม่อยู่)
Estatus	Yes/No	สถาพใช้งาน(ใช้ได้,ใช้ไม่ได้)

ตารางที่ 3.1(ต่อ) Data Description/Data Dictionary

DEVICE_F_TABLE.mdb

FIELDNAME	TYPE	DESCRIPTION
FID	Text(50)	รหัสวัสดุ
FList	Text(50)	รายการ
FMID	Text(50)	รหัสภาควิชา

DEVICE_TABLE.mdb

FIELDNAME	TYPE	DESCRIPTION
DID	Text(50)	เลขครุภัณฑ์ทางภาควิชา
DAccBook	Text(50)	ทะเบียนเล่มที่
DAccPage	Text(50)	ทะเบียนแผ่นที่
DListName	Text(50)	ชื่อรายการ
DListBand	Text(50)	ยี่ห้อรายการ
DListSerie	Text(50)	รุ่นรายการ
DGetBy	Text(50)	วิธีการได้มา
DRoom	Text(50)	ห้องที่อยู่
DMID	Text(50)	รหัสภาควิชา
DGID	Text(50)	เลขที่ใบบันทึกซื้อ/เลขที่ใบสั่งซื้อ
DYearBudget	Text(50)	งบประมาณ
Dwarnning	Text(50)	ระยะเวลาประกัน
DCheck	Yes/No	ตรวจประจำปี

ตารางที่ 3.1(ต่อ) Data Description/Data Dictionary

HIRE_TABLE.mdb

FIELDNAME	TYPE	DESCRIPTION
HDID	Text(50)	รหัสครุภัณฑ์
HGID	Text(50)	เลขที่บันทึกจ้าง
HList	Text(50)	รายการ
HMoney	Number	ราคา
HLastDate	Date/Time	วันซ่อมหลังสุด

MAJOR_TABLE.mdb

FIELDNAME	TYPE	DESCRIPTION
MID	Text(50)	สาขาวิชา
MName	Text(50)	ชื่อสาขาวิชา
MDepart	Text(50)	ภาควิชา
MFaculty	Text(50)	คณะ

PERSON_TABLE.mdb

FIELDNAME	TYPE	DESCRIPTION
PID	Text(50)	รหัสอาจารย์,รหัสนิสิต,รหัสบุคลากร
PPreName	Text(50)	คำนำหน้าชื่อ
PName	Text(50)	ชื่อ
PSurName	Text(50)	นามสกุล
PStatus	Text(50)	ตำแหน่ง
PRoom	Text(50)	ห้องที่ทำงาน
PMID	Text(50)	รหัสภาควิชา
PTel	Text(50)	เบอร์โทรศัพท์
PPassWord	Text(50)	รหัสผ่าน

ตารางที่ 3.1(ต่อ) Data Description/Data Dictionary

STORE_TABLE.mdb

FIELDNAME	TYPE	DESCRIPTION
SNumVat	Text(50)	เลขเสียภาษี
SName	Text(50)	ชื่อร้านค้า
SAddress	Text(50)	ที่อยู่ร้านค้า
STel	Text(50)	เบอร์โทรศัพท์

STUDENT_TABLE.mdb

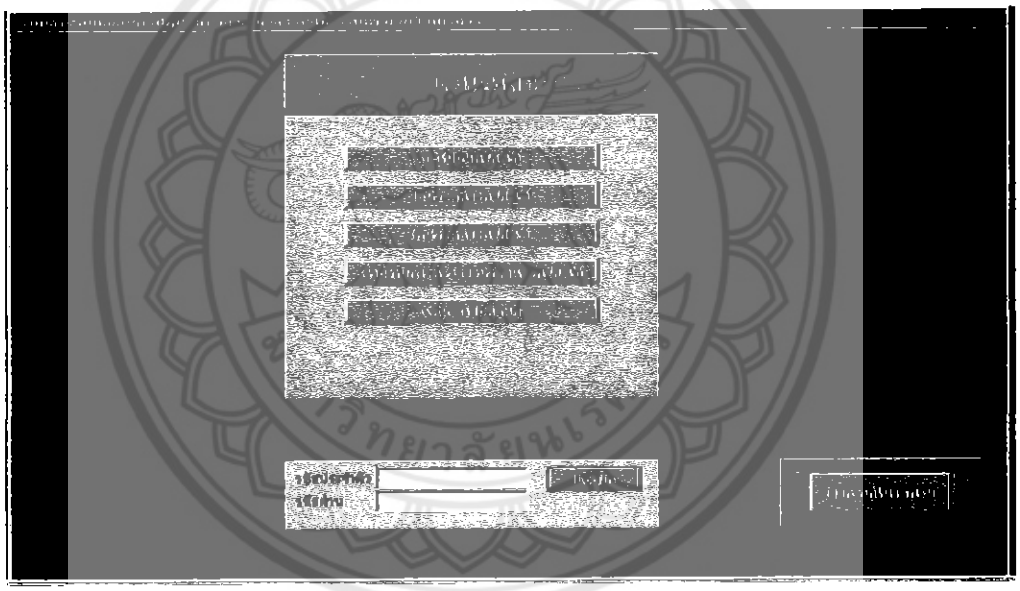
FIELDNAME	TYPE	DESCRIPTION
PID	Text(50)	รหัสประจำตัว
PPreName	Text(50)	คำนำหน้า
PName	Text(50)	ชื่อ
PSurName	Text(50)	นามสกุล
PMID	Text(50)	รหัสสาขา
PTel	Text(50)	เบอร์โทรศัพท์

บทที่ 4
การพัฒนาโปรแกรม

โปรแกรมระบบสารสนเทศอาคารปฏิบัติการคณะวิศวกรรมศาสตร์มหาวิทยาลัยนเรศวร เขียนขึ้นโดยใช้โปรแกรม Delphi 5 เป็นส่วนหน้าจอติดต่อกับผู้ใช้และจัดการกับฐานข้อมูลที่สร้างขึ้นด้วย Microsoft Access 2000 ในการพัฒนาโปรแกรมมีขั้นตอนแนวคิดและลักษณะโปรแกรมต่อไปนี้

4.1 รูปแบบโปรแกรมการยืมคืนวัสดุและอุปกรณ์

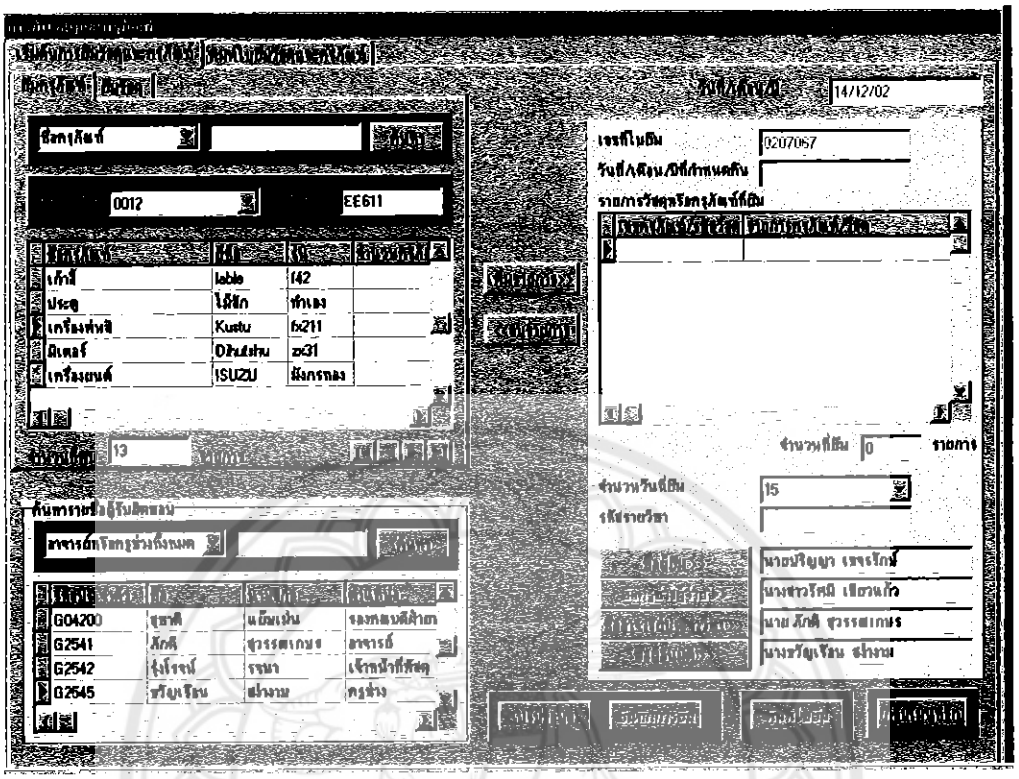
4.1.1 เมนูหลักระบบสารสนเทศอาคารปฏิบัติการคณะวิศวกรรมศาสตร์มหาวิทยาลัยนเรศวร



รูปที่ 4.1 แบบฟอร์มเมนูหลักระบบสารสนเทศอาคารปฏิบัติการคณะวิศวกรรมศาสตร์มหาวิทยาลัยนเรศวร

ฟอร์มเมนูหลักประกอบด้วย ทะเบียนครุภัณฑ์ ยืมวัสดุและครุภัณฑ์ คืนวัสดุและครุภัณฑ์ รายงานการตรวจสอบวัสดุและครุภัณฑ์และเพิ่มรายละเอียดอื่นๆ เมื่อเราจะเข้าไปใช้โปรแกรม ก็ทำการ Log In เข้าไปโดยใส่รหัสประจำตัวกับรหัสผ่าน

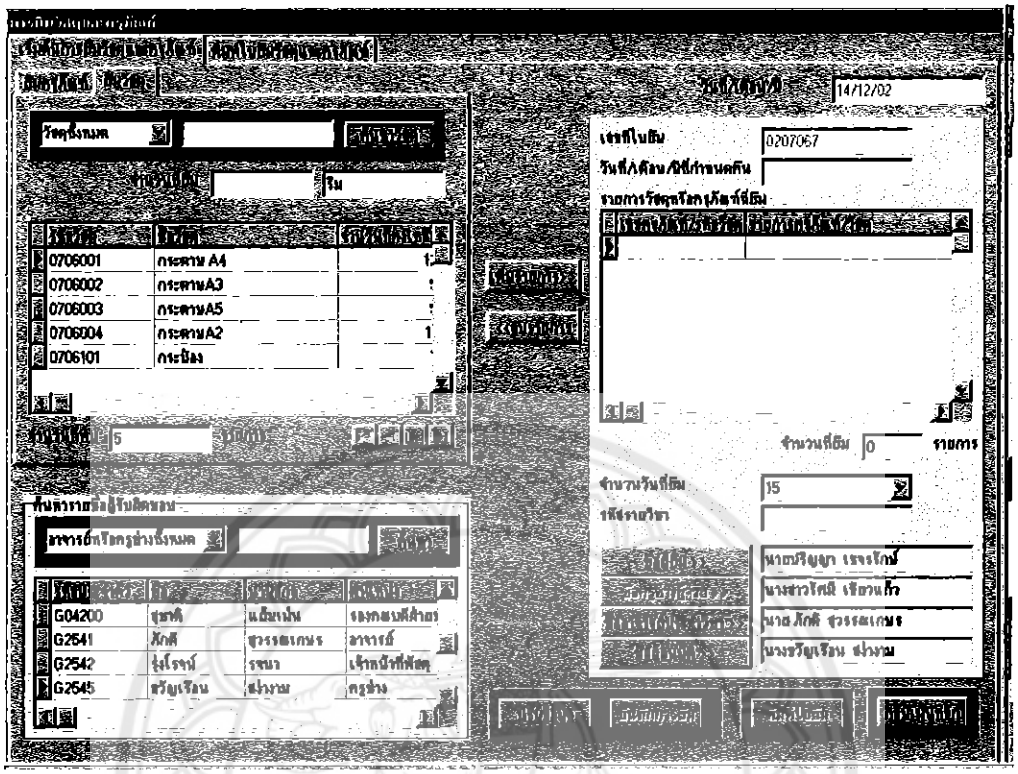
4.1.2 ระบบการยืมคืนครุภัณฑ์และวัสดุ



รูปที่ 4.2 ลักษณะของโปรแกรมการยืมครุภัณฑ์

สามารถค้นหาครุภัณฑ์ได้โดยจะค้นหาครุภัณฑ์ทั้งหมด ค้นหาตามชื่อครุภัณฑ์ หรือเลขครุภัณฑ์ เมื่อเราได้รายการครุภัณฑ์ที่จะยืมแล้ว ในส่วนของการค้นหารายชื่อผู้ที่รับผิดชอบโดยสามารถค้นหาได้จาก นิสิตทั้งหมด ชื่อนิสิต นามสกุลนิสิต รหัสนิสิต อาจารย์หรือครูช่างทั้งหมด ชื่ออาจารย์หรือครูช่าง นามสกุลอาจารย์หรือครูช่าง และรหัสอาจารย์หรือรหัสครูช่าง ในส่วนของรายการวัสดุและครุภัณฑ์ก็เพิ่มรายการเข้าไป โดยการเพิ่มจำนวนที่ยืม จำนวนวันที่ยืม รหัสนิสิต รหัสรายวิชา ในส่วนของชื่อผู้ยืม ชื่ออาจารย์ประจำภาควิชา และชื่อผู้อนุมัติ ทำการปรับข้อมูล เป็นการเสร็จการยืมครุภัณฑ์

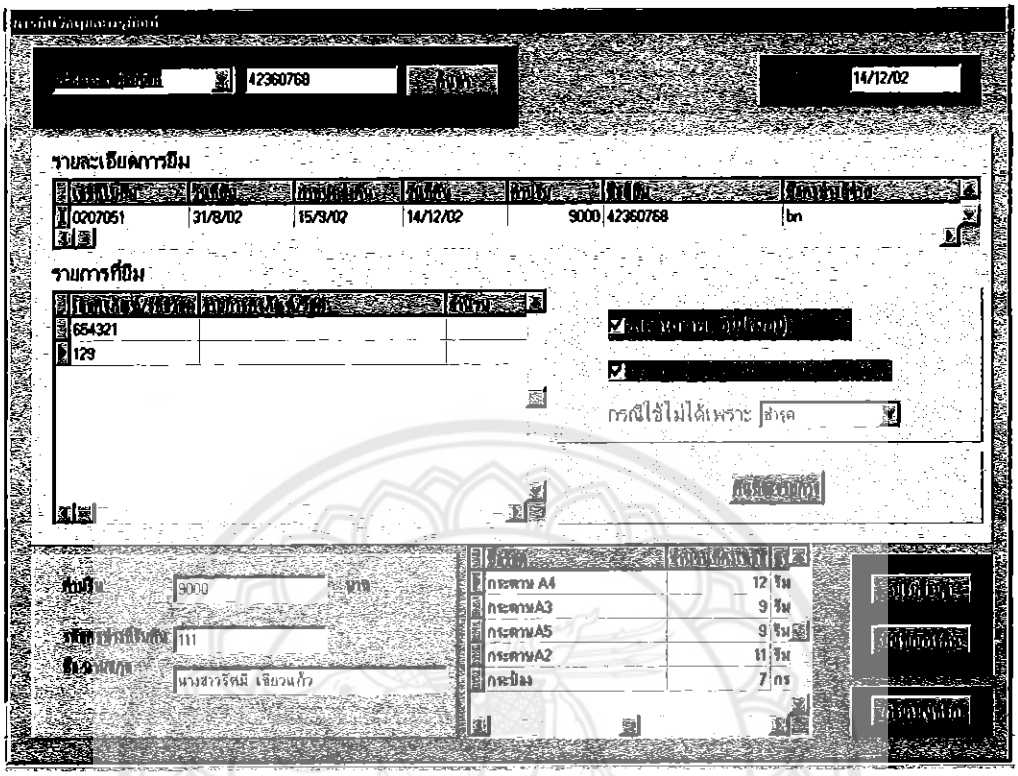
4.1.3 ระบบการยืมวัสดุ



รูปที่ 4.3 ลักษณะโปรแกรมการยืมวัสดุ

ในส่วนของการยืมวัสดุสามารถค้นหาวัสดุได้จาก วัสดุทั้งหมด หรือรหัสวัสดุ แล้วจะปรากฏรายการมาดังรูป แล้วใส่จำนวนที่ยืม ในส่วนของการค้นหารายชื่อผู้ที่รับผิดชอบและรายการวัสดุและครุภัณฑ์ที่ยืมหน้าจอสีเหลือง ทำเหมือนกับการยืมครุภัณฑ์

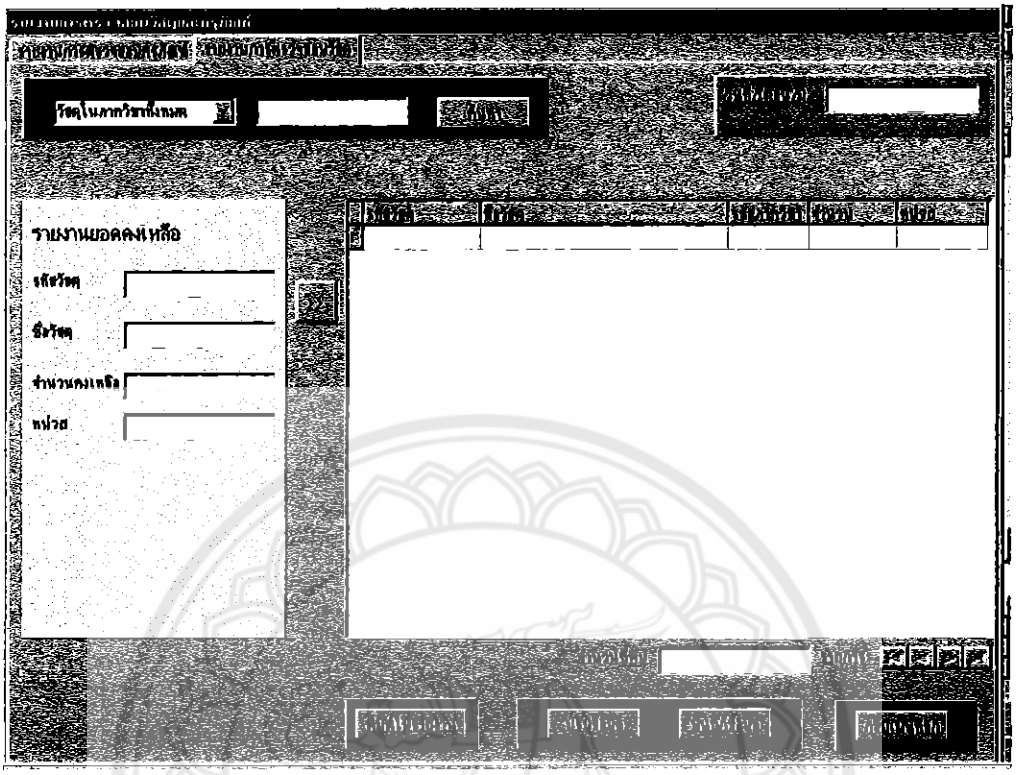
4.1.4 ระบบการคืนวัสดุและครุภัณฑ์



รูปที่ 4.4 ลักษณะโปรแกรมการคืนวัสดุและครุภัณฑ์

ทำการค้นหารายละเอียดการยืม โดยค้นหาได้จาก รหัสประจำตัวผู้ยืม ใบยืมทั้งหมด เลขที่ใบยืม ชื่อผู้ยืม หรือรหัสประจำตัวผู้ยืม แล้วก็มาใส่เครื่องหมายถูกที่สถานภาพเพื่อยืนยันว่าได้ทำการคืนแล้ว ทำการปรับข้อมูล แล้วบันทึกข้อมูล ในส่วนของหน้าจอสีม่วงจะบอกค่าปรับ รหัสครูช่างที่รับคืนและชื่อนามสกุลของครูช่าง

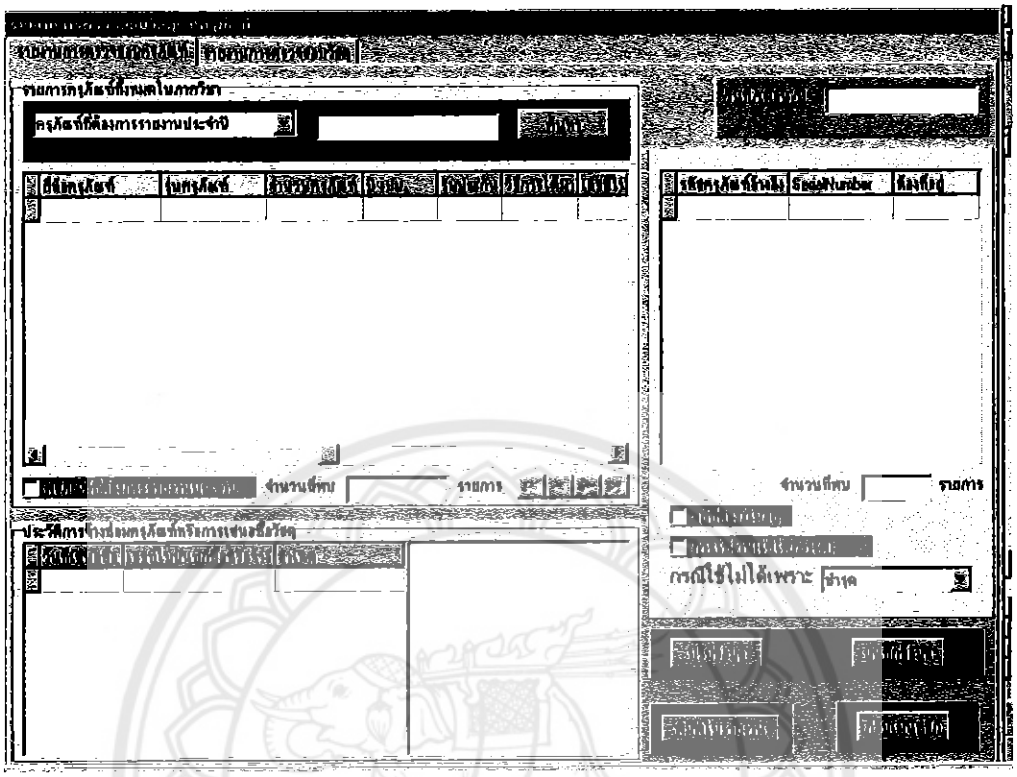
4.1.5 ระบบรายงานการตรวจสอบวัสดุ



รูปที่ 4.5 ลักษณะโปรแกรมรายงานการตรวจสอบวัสดุและครุภัณฑ์

ทำการค้นหาตามรายการ วัสดุทั้งหมด วัสดุในภาควิชาทั้งหมด ชื่อวัสดุในภาควิชา รหัสวัสดุในภาควิชา หรือใบบันทึกซื้อ เสร็จแล้วก็ทำการปรับข้อมูลและบันทึกรายการ ซึ่งในหน้าจอนี้จะมีในส่วนของการพิมพ์ใบรายงาน

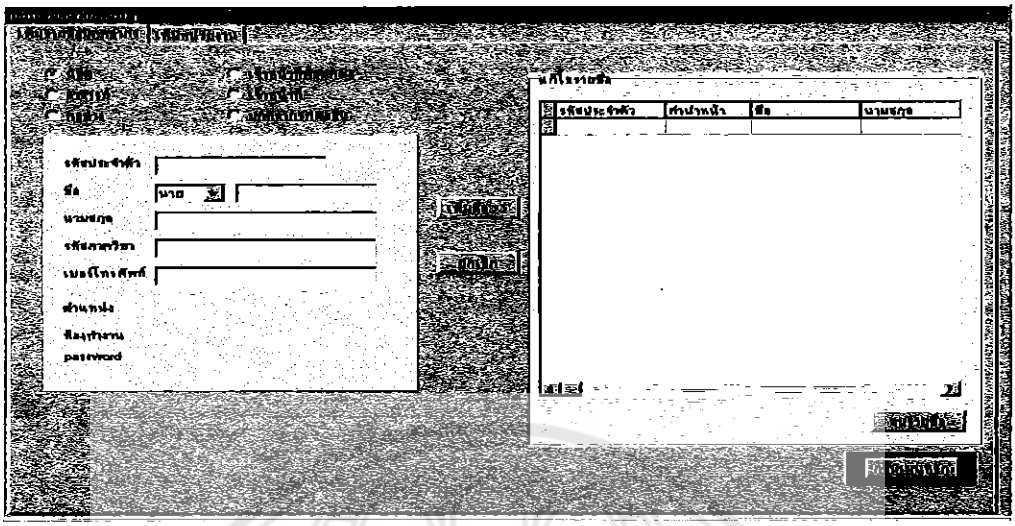
4.1.6 ระบบรายงานการตรวจสอบครุภัณฑ์



รูปที่ 4.6 ลักษณะโปรแกรมรายงานการตรวจสอบครุภัณฑ์

ทำการค้นหารายการครุภัณฑ์ได้จาก ครุภัณฑ์ที่ต้องการรายงานประจำปี ครุภัณฑ์ในภาควิชาทั้งหมด ครุภัณฑ์ทั้งหมดทุกภาควิชา ชื่อครุภัณฑ์ เลขครุภัณฑ์ ปีงบประมาณที่ซื้อหรือใบบันทึกซื้อ แล้วทำการตรวจว่าครุภัณฑ์ อยู่ที่ห้องหรือไม่อยู่ การใช้งานใช้ได้หรือไม่ได้ แล้วทำการปรับข้อมูลและบันทึกข้อมูล ในส่วนแบบฟอร์มนี้จะมีการพิมพ์ใบรายงาน

4.1.7 ระบบการเพิ่มรายชื่อบุคลากร



รูปที่ 4.7 ลักษณะโปรแกรมการเพิ่มรายละเอียดอื่นๆ

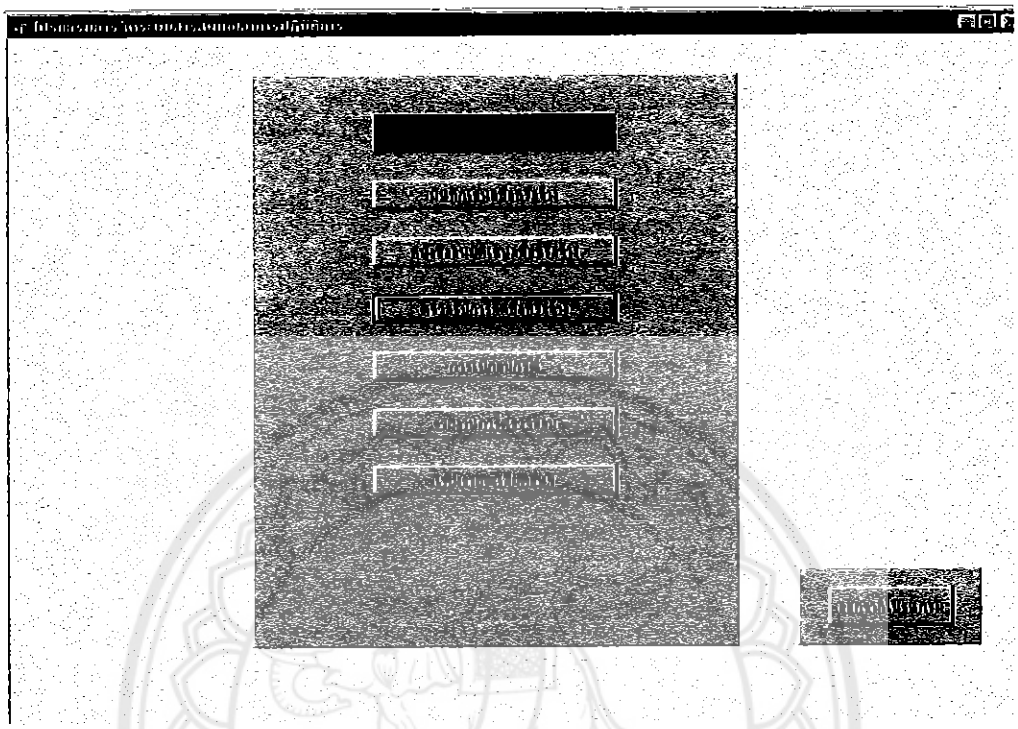
จะประกอบด้วยการเพิ่มในส่วนของ นิสิต อาจารย์ ครูช่าง เจ้าหน้าที่พัสดุคณะ เจ้าหน้าที่ และบุคลากรคณะอื่น

4.1.8 ระบบการเพิ่มหน่วยงาน

รูปที่ 4.8 ลักษณะโปรแกรมการเพิ่มหน่วยงาน

4.2 รูปแบบโปรแกรมการจัดซื้อหรือจัดจ้าง

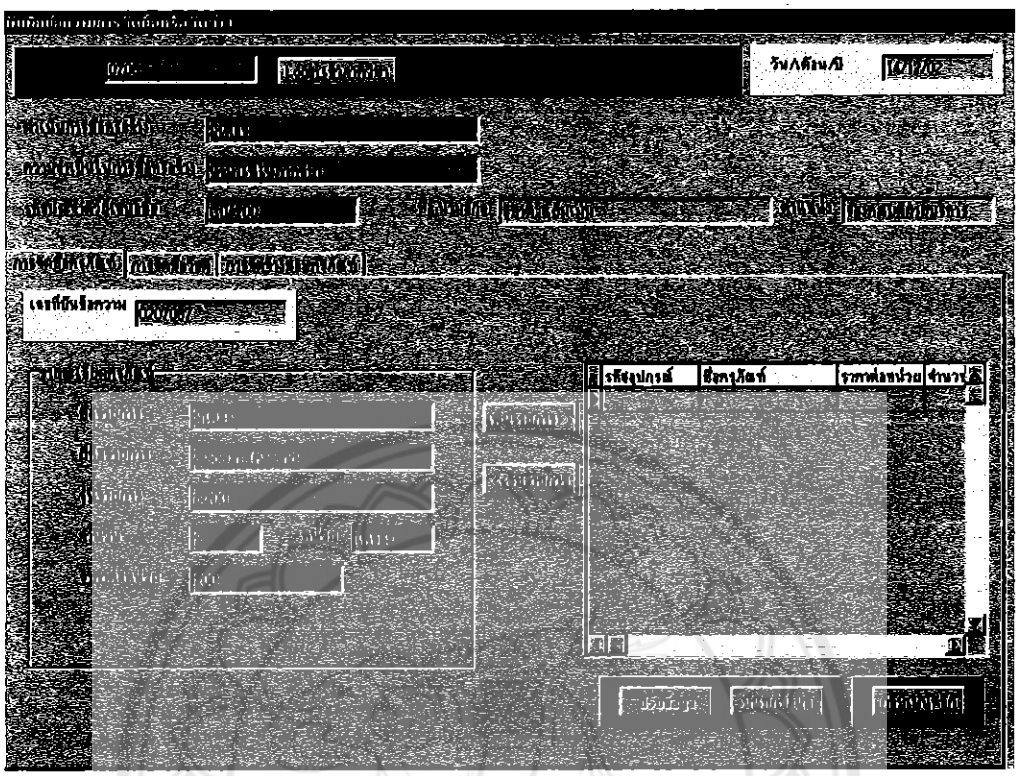
4.2.1 ลักษณะโปรแกรมของเมนูหลัก



รูปที่ 4.9 ลักษณะโปรแกรมเมนูหลักการจัดซื้อหรือจัดจ้าง

จะประกอบด้วย บันทึกเสนอซื้อหรือเสนอจ้าง หนังสืออนุมัติเสนอซื้อหรือเสนอจ้าง ตรวจรับการจัดซื้อหรือจัดจ้าง การเบิกอุปกรณ์ รายงานการตรวจรับ และเพิ่มรายละเอียดอื่นๆ

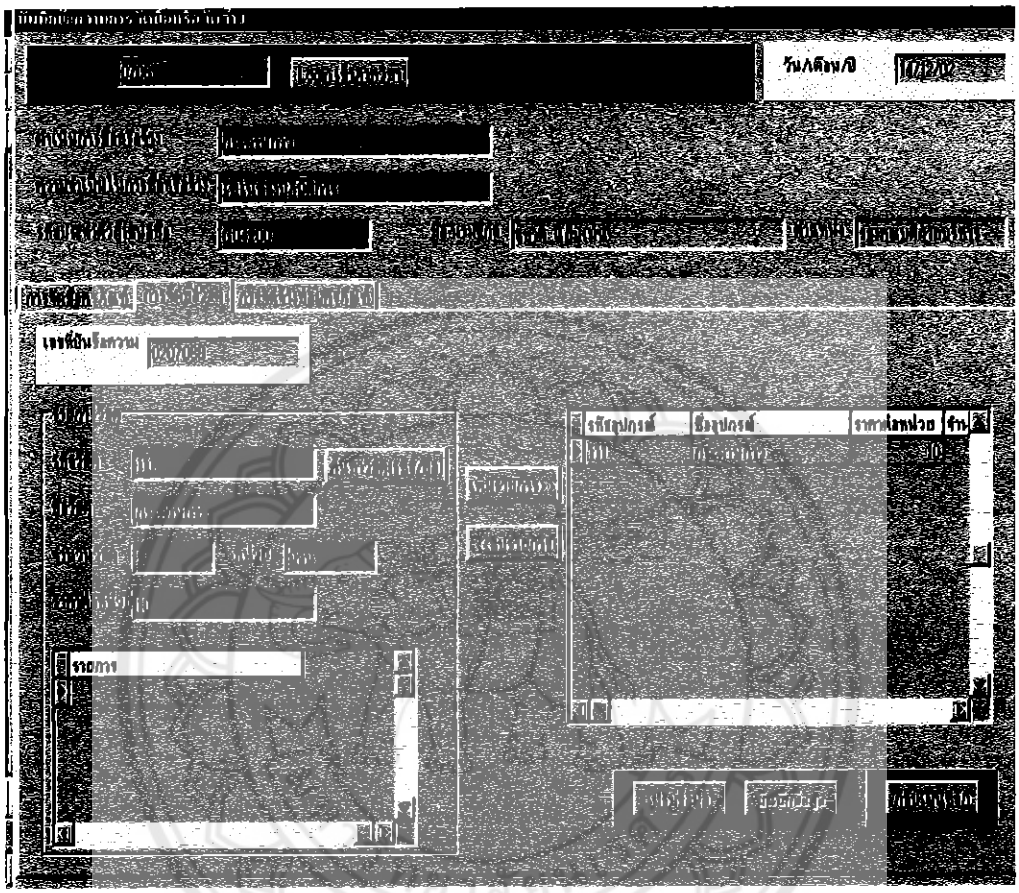
4.2.2 ระบบบันทึกเสนอซื้อหรือจ้าง(การจัดซื้อครุภัณฑ์)



รูปที่ 4.10 ลักษณะโปรแกรมการจัดซื้อครุภัณฑ์

ต้องกรอกในส่วนของพื้นที่สีฟ้าให้ครบก่อน อันดับแรกทำการ Log in เข้าภาควิชา เสร็จแล้วทำการกรอกรายละเอียดดังตัวอย่างเมื่อกรอกเสร็จแล้วทำการปรับข้อมูลและบันทึกข้อมูล

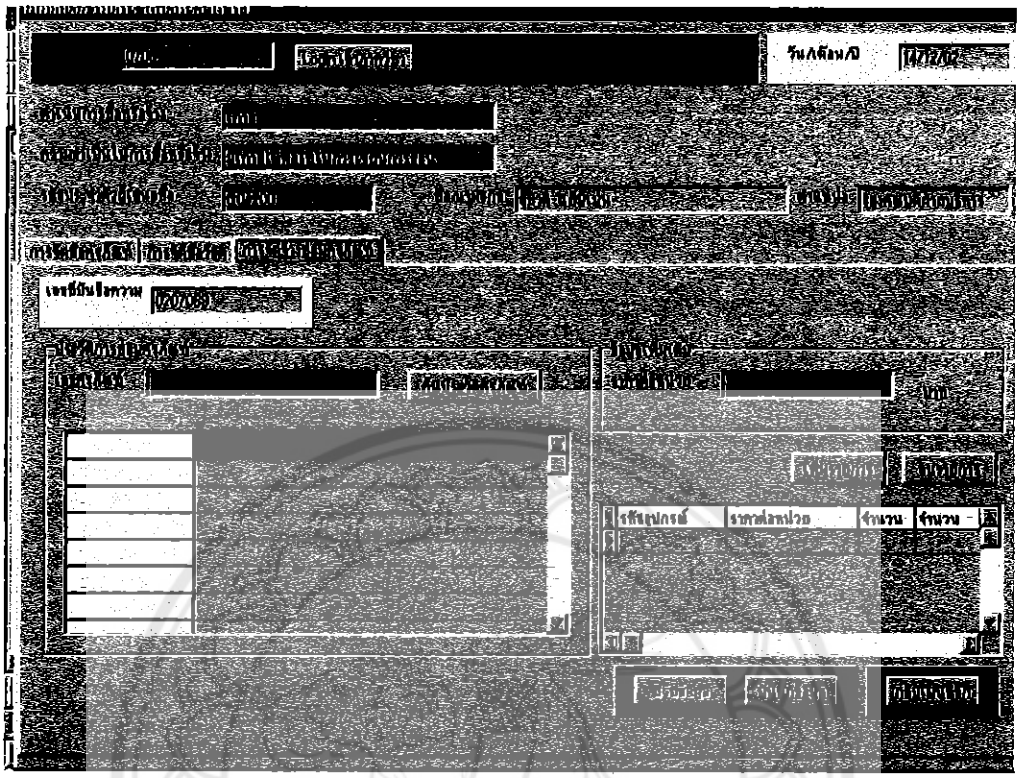
4.2.3 ระบบบันทึกเสนอซื้อหรือจ้าง(การจัดซื้อวัสดุ)



รูปที่ 4.11 การจัดซื้อหรือจัดจ้าง(การจัดซื้อวัสดุ)

เมื่อทำการ Log in เข้าภาควิชาแล้ว ทำการกรอกพื้นที่สีฟ้าก่อนเสร็จแล้ว ก็กรอกรายละเอียดอื่นๆ ตามตัวอย่างเสร็จแล้วทำการปรับข้อมูลและบันทึกข้อมูล

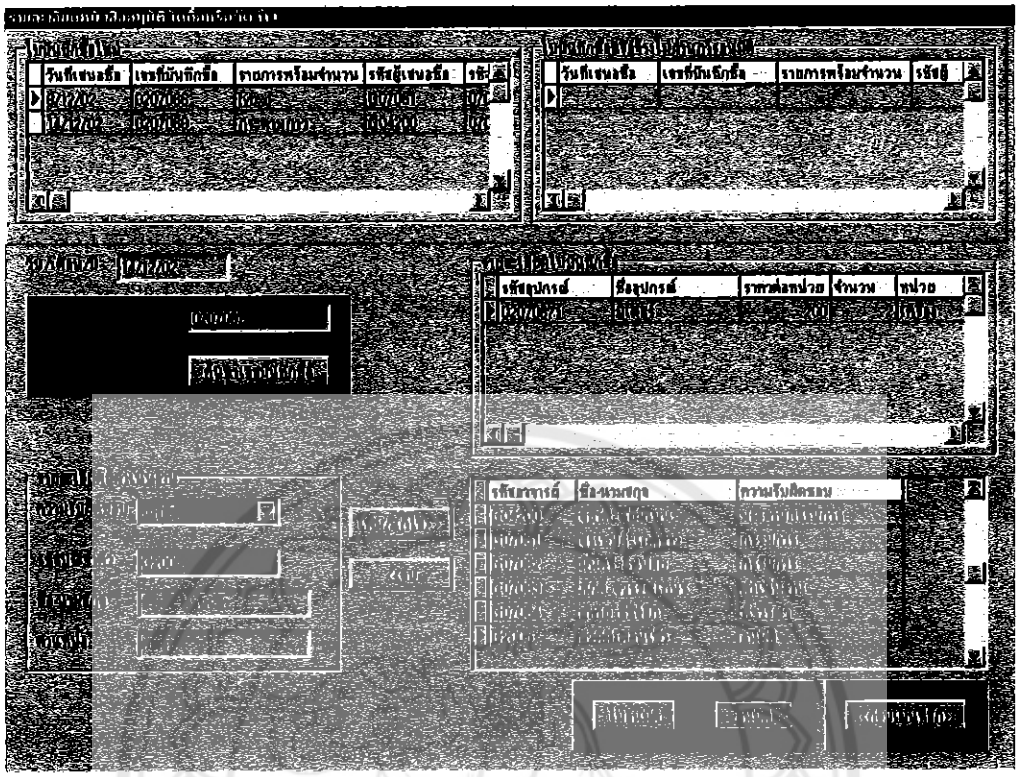
4.2.4 ระบบบันทึกเสนอซื้อหรือจ้าง(การจัดจ้างซ่อม)



รูปที่ 4.12 ลักษณะโปรแกรมการจัดซื้อหรือจัดจ้าง(การจ้างซ่อม)

ทำการกรอกในพื้นที่สีฟ้า เสร็จแล้วค้นหาประวัติการซ่อมครุภัณฑ์ จะขึ้นรายการมาให้ ในส่วนของราคาต่อหน่วยทางพัสดุจะต้องทำการสืบราคาแล้ว เมื่อเสร็จแล้วทำการปรับข้อมูลและบันทึกข้อมูล

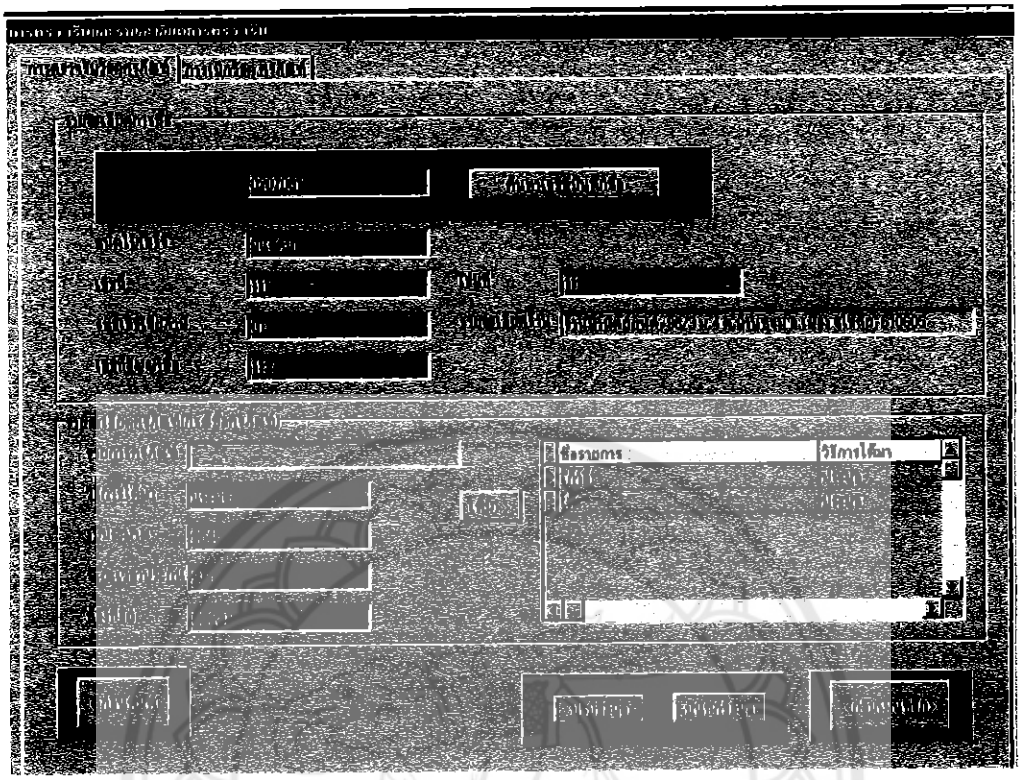
4.2.5 ระบบหนังสืออนุมัติเสนอซื้อหรือจ้าง



รูปที่ 4.13 ลักษณะโปรแกรมหนังสืออนุมัติเสนอซื้อหรือจ้าง

ทำการค้นหาเลขใบบันทึกซื้อหรือจ้าง จะขึ้นรายละเอียดของอุปกรณ์มาให้ หลังจากนั้นในส่วนของรายละเอียดผู้ดำเนินงานจะประกอบด้วย ประธานกรรมการจะเป็นผู้ที่เสนอซื้อ กรรมการมีกี่คนก็ได้ ผู้ดำเนินงาน ผู้พิจารณา และผู้อนุมัติจะเป็นคณะบดี เสร็จแล้วจึงทำการอนุมัติหรือไม่อนุมัติ

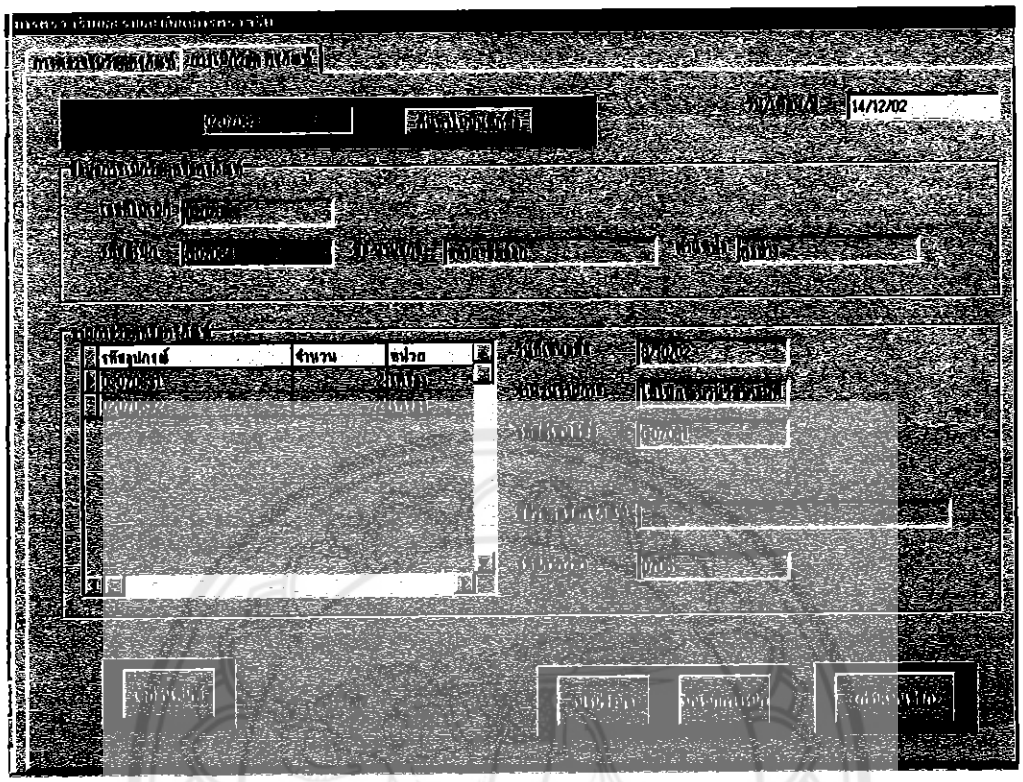
4.2.6 ระบบการตรวจรับวัสดุและครุภัณฑ์



รูปที่ 4.14 ลักษณะโปรแกรมการตรวจรับวัสดุและครุภัณฑ์

ใส่เลขที่ใบบันทึกซื้อจะขึ้นชื่อรายการมาให้ในส่วนของสีเหลือง เสร็จแล้วก็กรอกสีฟ้าให้เสร็จดัง ตัวอย่างเสร็จแล้วทำการปรับข้อมูลและบันทึกข้อมูล

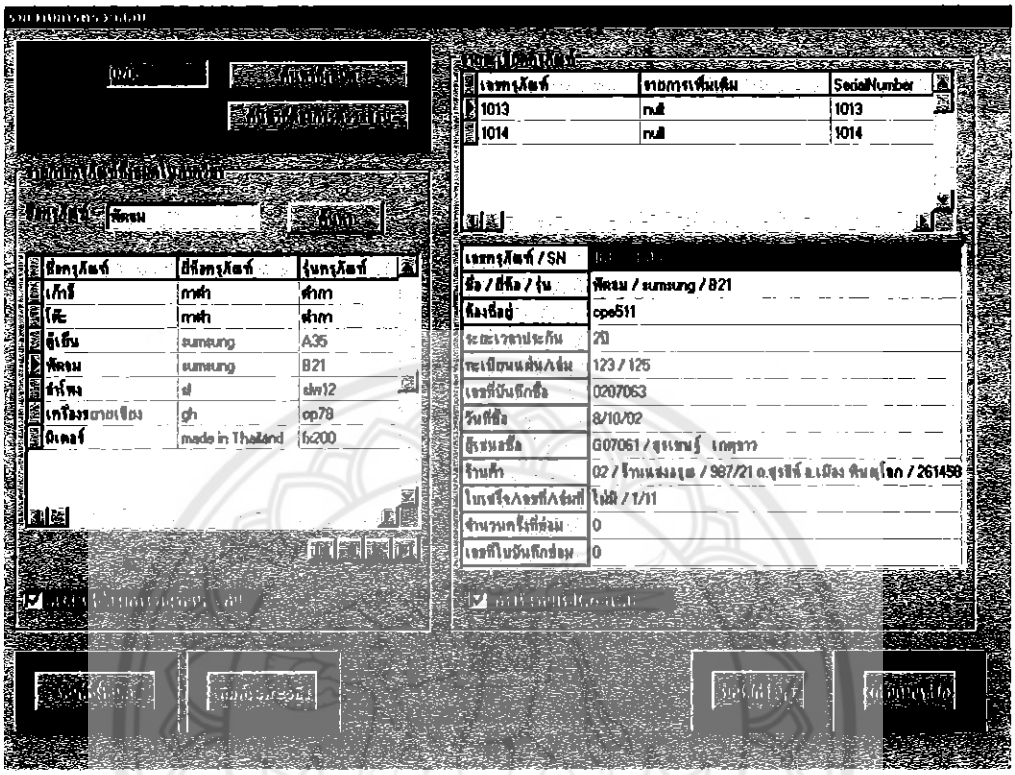
4.2.7 ระบบการเบิกวัสดุและอุปกรณ์



รูปที่ 4.15 ลักษณะโปรแกรมการเบิกวัสดุและอุปกรณ์

ในส่วนของฟอร์มนี้เป็นส่วนที่ครูช่างเบิกวัสดุและอุปกรณ์มาจากพัสดุคณะ เพื่อนำมาไว้ในอาคารปฏิบัติการคณะวิศวกรรมศาสตร์ ทำการกรอกเลขที่ใบบันทึกซื้อ ก็จะขึ้นรายละเอียดของวัสดุและครุภัณฑ์ ครูช่างจะทำการใส่รหัสผู้เบิกแล้วทำการปรับข้อมูลและบันทึกข้อมูล

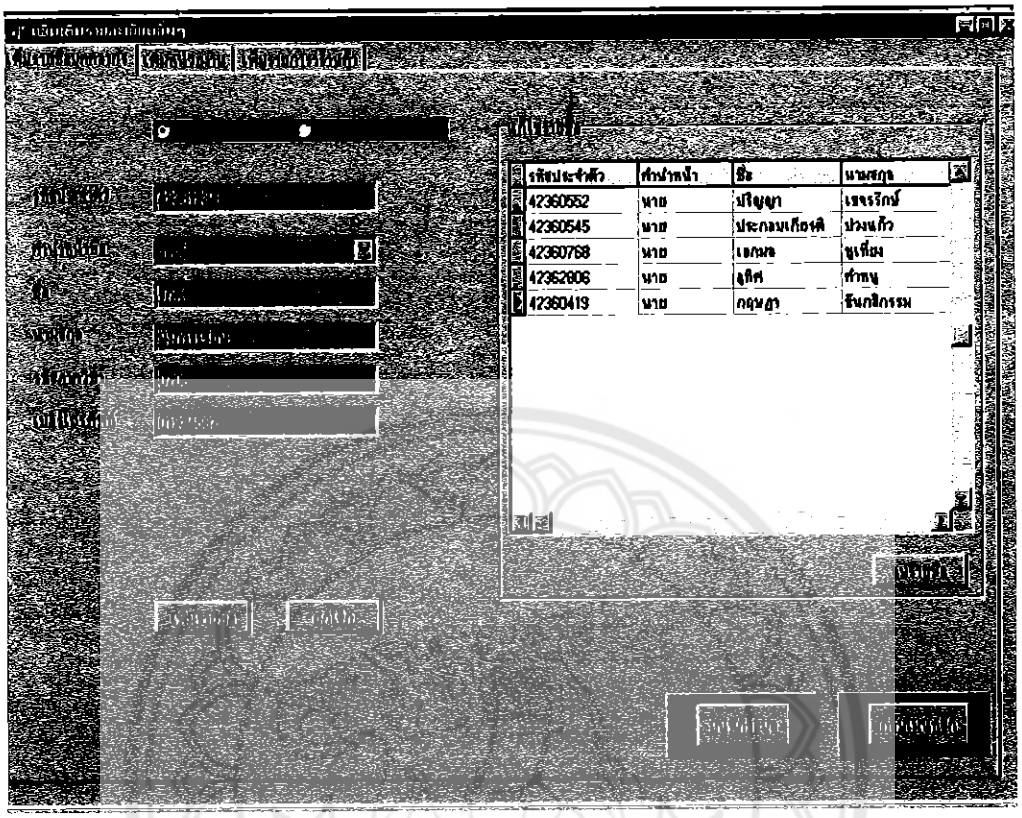
4.2.8 ระบบรายงานการตรวจสอบ



รูปที่ 4.16 ลักษณะโปรแกรมรายงานการตรวจสอบ

ทำการ Log in เข้าภาควิชา เสร็จแล้วทำการค้นหาครุภัณฑ์โดยค้นหาแบบทั้งหมด หรือค้นหาเฉพาะที่ต้องการตรวจสอบ เสร็จแล้วจะขึ้นรายละเอียดครุภัณฑ์ครุภัณฑ์มาให้พร้อมทั้งประวัติ ในส่วนของการค้นหาครุภัณฑ์จะค้นหาตามชื่อก็ได้ โดยทำการกรอกชื่อครุภัณฑ์ที่ต้องการ ในแบบฟอร์มนี้จะมีส่วนของการพิมพ์ เสร็จแล้วทำการปรับข้อมูลและทำการบันทึกข้อมูล

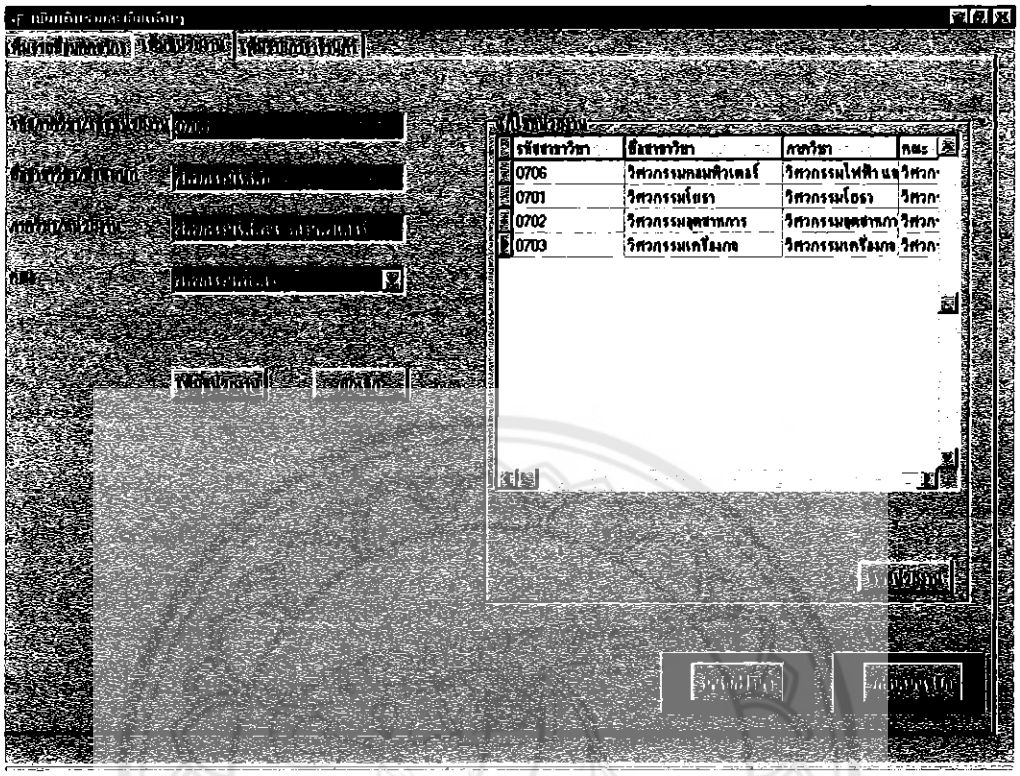
4.2.9 ระบบการเพิ่มรายละเอียดอื่นๆ(การเพิ่มรายชื่อนิสิต)



รูปที่ 4.17 ลักษณะโปรแกรมการเพิ่มรายละเอียดอื่นๆ(การเพิ่มรายชื่อนิสิต)

ทำการเพิ่มรายชื่อนิสิตโดยการกรอกข้อมูลในพื้นที่สีฟ้าตามตัวอย่าง เสร็จแล้ว ทำการเพิ่มรายชื่อโดยคลิกปุ่มการเพิ่มรายชื่อ เสร็จแล้วทำการบันทึกข้อมูล แต่ถ้าต้องการลบรายชื่อก็ทำการคลิกปุ่มลบรายชื่อ

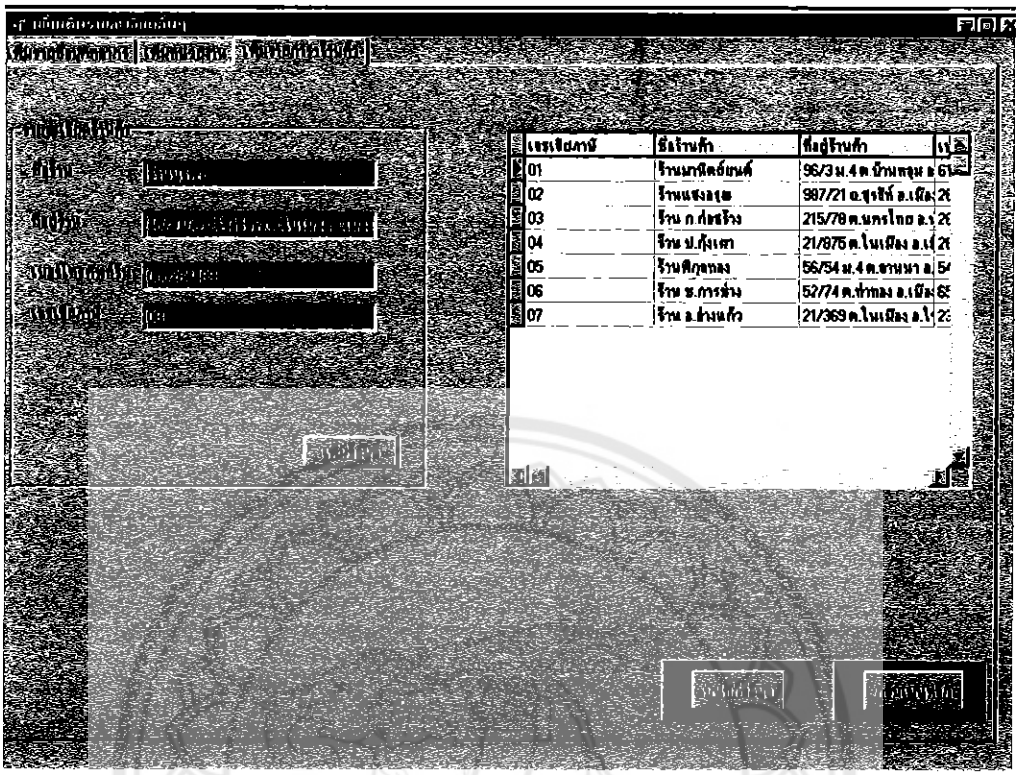
4.2.10 ระบบการเพิ่มรายละเอียดอื่นๆ(การเพิ่มหน่วยงาน)



รูปที่ 4.18 ลักษณะโปรแกรมการเพิ่มรายละเอียดอื่นๆ(การเพิ่มหน่วยงาน)

ทำการเพิ่มหน่วยงานที่มีใหม่ โดยการกรอกข้อมูลในพื้นที่สีฟ้าให้เรียบร้อย เสร็จแล้วทำการคลิกปุ่มเพิ่มหน่วยงาน แล้วทำการบันทึกโดยคลิกปุ่มบันทึก ถ้าหน่วยงานไหนที่ไม่ต้องการให้ทำการลบทิ้งโดยคลิกปุ่มลบหน่วยงาน

4.2.11 ระบบการเพิ่มรายละเอียดอื่นๆ(เพิ่มรายการร้านค้า)

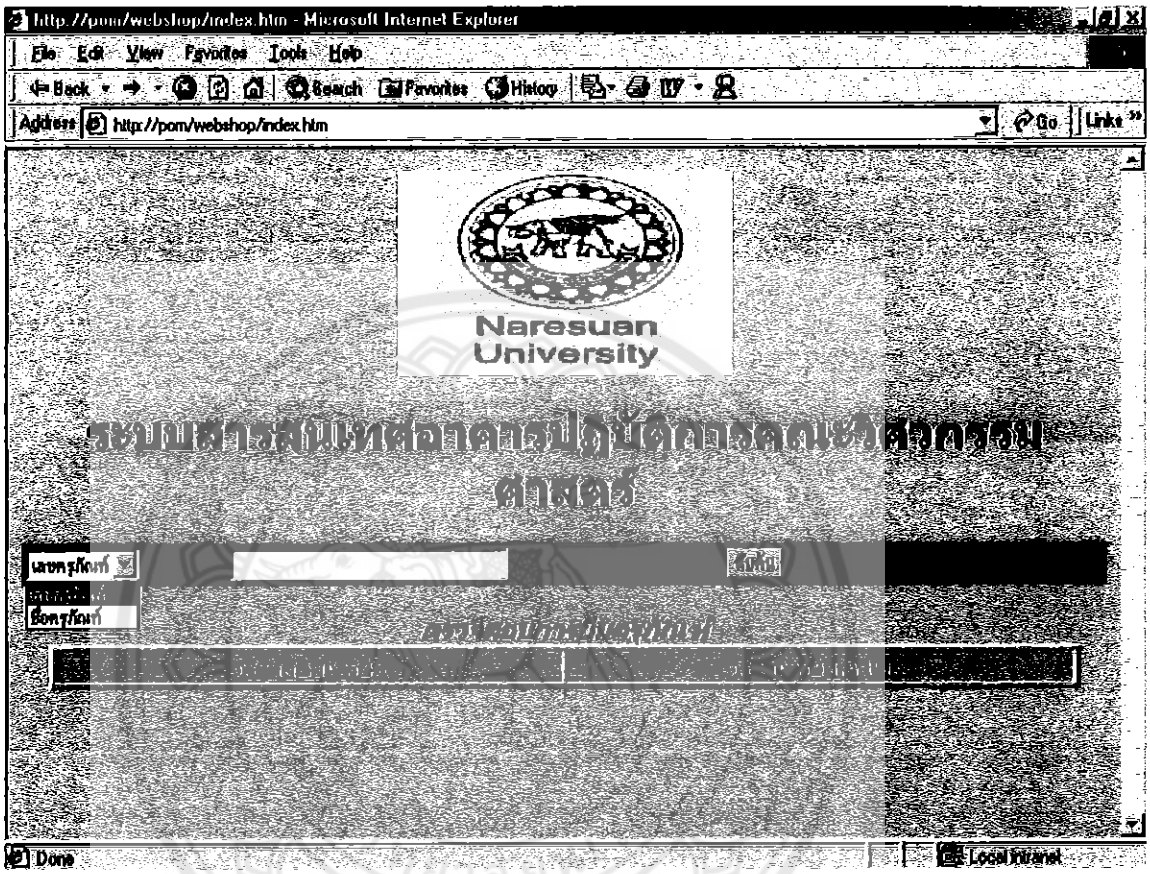


รูปที่ 4.19 ลักษณะโปรแกรมการเพิ่มรายละเอียดอื่นๆ(การเพิ่มรายการร้านค้า)

ทำการกรอกข้อมูลในส่วนของพื้นที่สีฟ้าตามตัวอย่างให้ครบ หลังจากนั้นทำการเพิ่มรายการร้านค้า โดยการคลิกปุ่มเพิ่มข้อมูล เสร็จแล้วก็ทำการบันทึกข้อมูล เป็นการเสร็จสิ้นการเพิ่มรายการร้านค้า

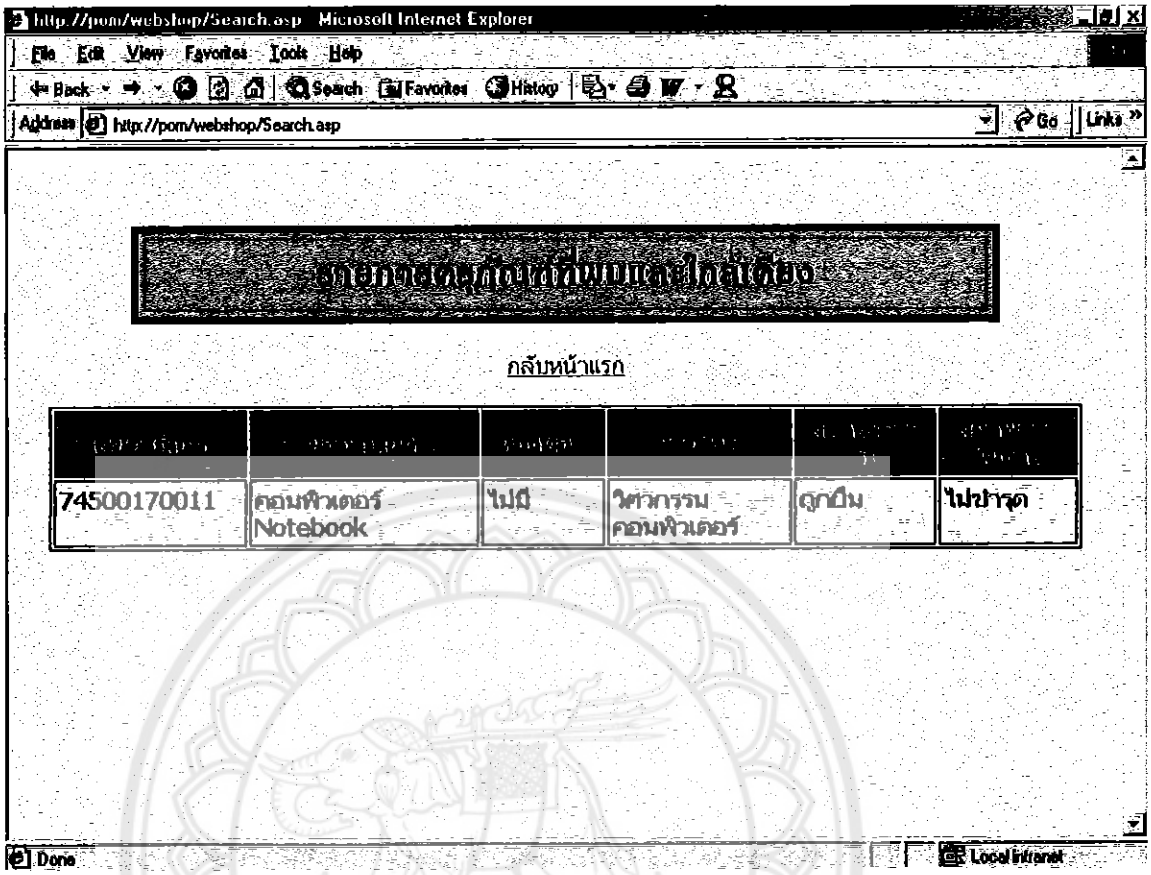
4.3 เวบไซต์ระบบสารสนเทศอาคารปฏิบัติการคณะวิศวกรรมศาสตร์มหาวิทยาลัยนเรศวร

4.3.1 หน้าจอหลักสำหรับค้นหารายละเอียดต่างๆ



รูปที่ 4.20 เวบไซต์หลักสำหรับการค้นหาข้อมูล

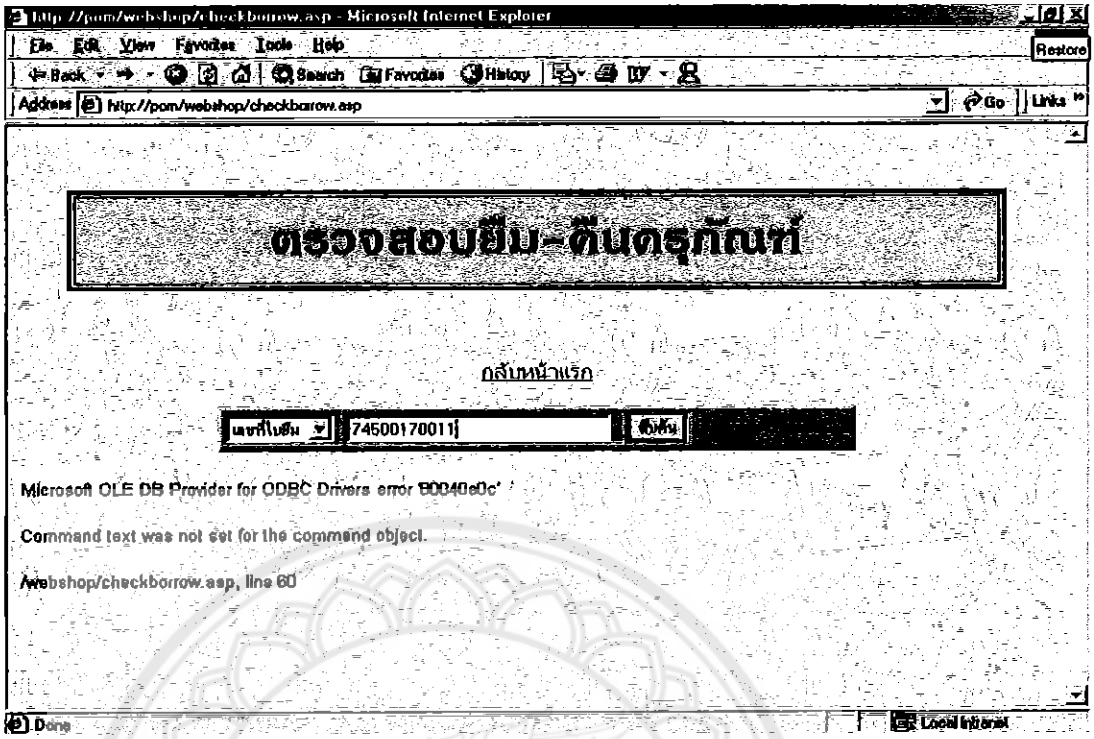
ในส่วนของหน้าจอนี้ จะค้นหาครุภัณฑ์แบบ ค้นหาเลขครุภัณฑ์ หรือ ค้นหาแบบชื่อครุภัณฑ์ จะเป็นการสืบค้นว่าครุภัณฑ์ตัวที่เราจะยืมอยู่หรือไม่ ถ้าอยู่ อยู่ที่ห้องไหน สถานภาพการใช้งาน ใช้ได้ไหม เช่น ทำการสืบค้น Note Book แล้วจะปรากฏดังรูปที่ 4.21



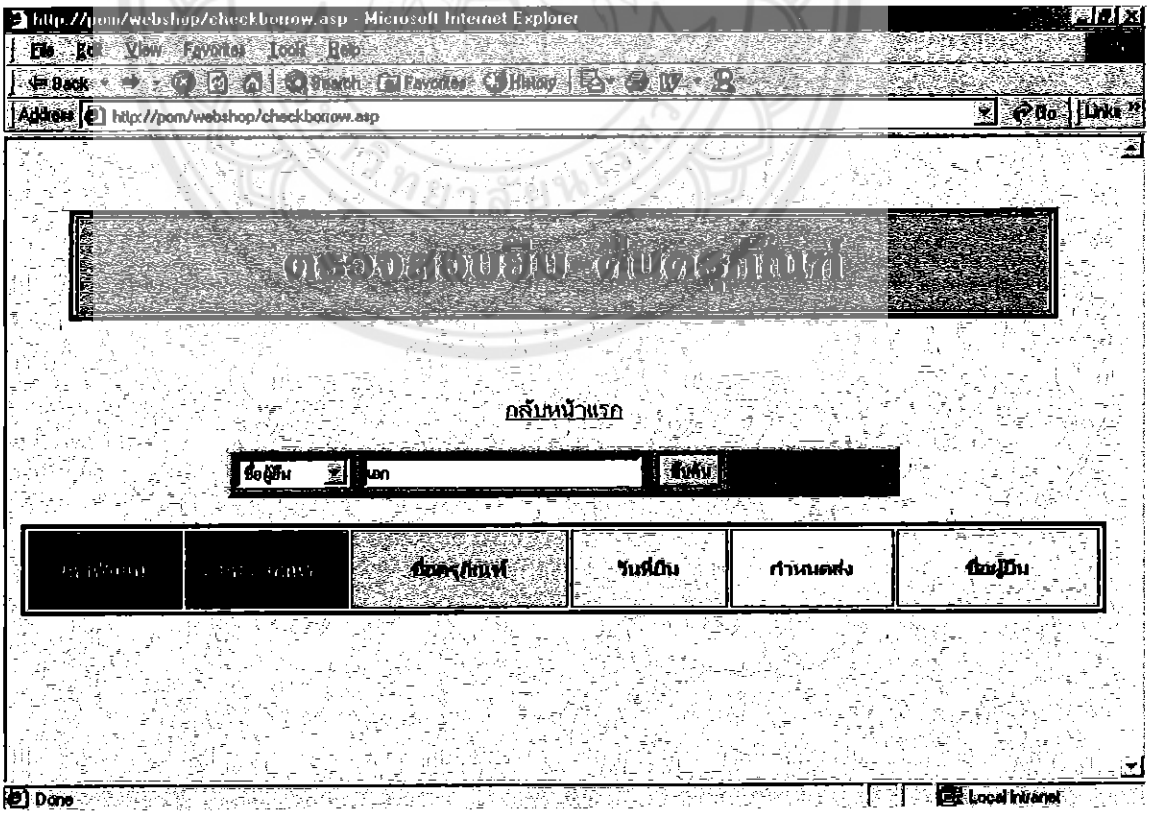
รูปที่ 4.21 หน้าจอเมื่อทำการสืบค้น Note Book แล้ว

4.3.2 ในส่วนของการตรวจสอบการยืม

เป็นการตรวจดูว่าครุภัณฑ์ตัวที่เราสนใจที่จะขอยืม ตอนนี้ถูกใครยืมไปหรือไม่ ถ้าถูกยืม ใครยืมไป หรือถ้าหากต้องการทราบว่า นาย ก. ยืมอะไรไป สามารถตรวจสอบดูได้ โดยในการค้นหา ให้เลือก เลขที่ใบยืม เลขครุภัณฑ์ หรือชื่อผู้ยืม ดังแสดงใน รูปที่ 4.22 และ รูปที่ 4.23



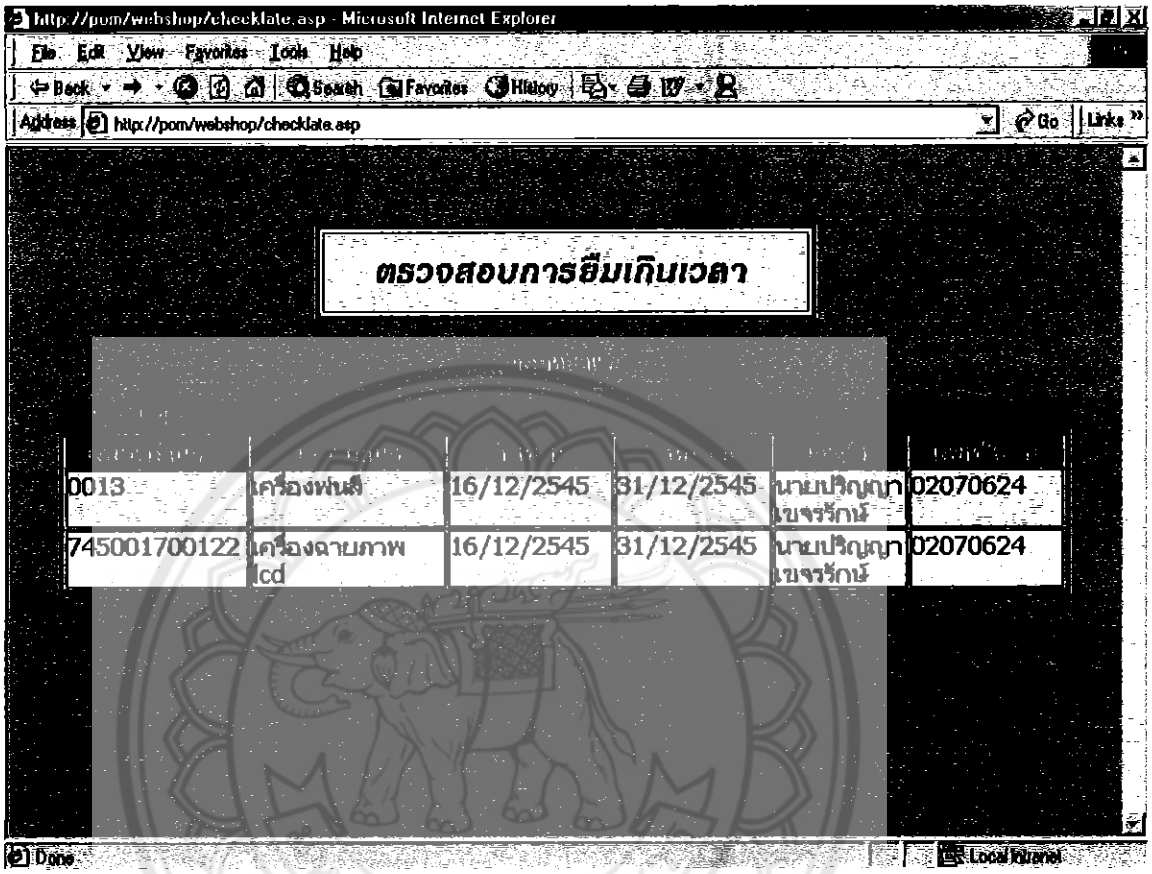
รูปที่ 4.22 ตรวจสอบการยืมคืน



รูปที่ 4.23 การตรวจสอบการยืมคืน

4.3.3 การตรวจสอบการยืมเกินเวลา

สามารถที่จะตรวจสอบได้ว่า มีใครยืมวัสดุและครุภัณฑ์เกินเวลาบ้าง จะขึ้นรายชื่อ



รูปที่ 4.24 การตรวจสอบการยืมเกินเวลา

บทที่ 5
บทสรุป

5.1 สรุปผล

1. ระบบสารสนเทศการยืมคืนวัสดุและครุภัณฑ์ สามารถทำการยืมคืนแทนระบบเก่าที่ใช้ในการยืมคืนมาใช้ระบบสารสนเทศแทน ทำให้การทำงานเร็วขึ้น การยืมคืนมีประสิทธิภาพ สามารถทำการค้นหาวัสดุและครุภัณฑ์ที่ต้องการจะยืมได้อย่างรวดเร็ว ทำให้ทราบข้อมูลว่าวัสดุและครุภัณฑ์ตัวไหนอยู่หรือไม่อยู่ ชำรุดหรือไม่ชำรุด พร้อมที่จะยืมได้ไหม ทำให้ครูช่างทำการตรวจสอบวัสดุและครุภัณฑ์ผ่านทางระบบสารสนเทศได้ สามารถพิมพ์ใบยืมคืนวัสดุและครุภัณฑ์ได้

2. ในส่วนของเวบไซต์การยืมคืน ทำให้นิสิตสะดวกต่อการยืมคืนข้อมูล ทำให้นิสิตดูข้อมูลวัสดุและครุภัณฑ์ผ่านทางเวบไซต์ ว่าวัสดุและครุภัณฑ์ตัวไหนอยู่ที่อาคารปฏิบัติการหรือถูกใครยืมไป หรือชำรุดไม่สามารถที่จะยืมได้ โดยไม่ต้องเดินทางมาดูที่อาคารปฏิบัติการ

3. ระบบสารสนเทศ การจัดซื้อหรือจัดจ้าง สามารถที่ ทำการบันทึกเสนอซื้อหรือจ้างได้ ทำการกรอกรายละเอียดวัสดุและครุภัณฑ์ สามารถทำการอนุมัติจัดซื้อหรือจัดจ้างได้ โดยการอนุมัติผ่านทางระบบสารสนเทศได้ สามารถที่จะตรวจรับวัสดุและครุภัณฑ์ที่ได้ทำการตรวจรับจารร้านค้า โดยเจ้าหน้าที่พัสดุคณะทำการตรวจรับ และตรวจสอบวัสดุและครุภัณฑ์ สามารถทำการเบิกวัสดุและครุภัณฑ์จากพัสดุคณะมาไว้ที่อาคารปฏิบัติการผ่านทางระบบสารสนเทศ โดยครูช่างเป็นผู้ทำการเบิกวัสดุและครุภัณฑ์ได้ สามารถทำการตรวจสอบวัสดุและครุภัณฑ์ และรายงานการตรวจสอบได้ สามารถทำการเพิ่มรายละเอียดในส่วนของการเพิ่มรายละเอียดชื่อบุคลากรได้ เพิ่มหน่วยงานได้ และเพิ่มรายการร้านค้าได้

5.2 ประเมินผลและข้อเสนอแนะ

จากผลของโครงงานเมื่อเทียบกับวัตถุประสงค์ที่ตั้งไว้ได้ผลว่า

1. จากการดำเนินงานที่ผ่านมาสามารถออกแบบฐานข้อมูลได้ตามโครงสร้างฐานข้อมูลในบทที่ 3

2.เครื่องมือที่ใช้ในการพัฒนาโปรแกรมระบบสารสนเทศอาคารปฏิบัติการคณะวิศวกรรมศาสตร์มหาวิทยาลัยนเรศวร ได้แก่ โปรแกรม Delphi 5 ในการเขียนโปรแกรม และใช้ Microsoft Access 2000 เป็นฐานข้อมูล เนื่องจากโปรแกรม Microsoft Access 97 เป็นฐานข้อมูลที่ใช้กันอย่างแพร่หลาย

3.ระบบสารสนเทศอาคารปฏิบัติการคณะวิศวกรรมศาสตร์มหาวิทยาลัยนเรศวร ที่ได้พัฒนาขึ้นประกอบด้วยส่วน ระบบการยืมคืนวัสดุและครุภัณฑ์ ประกอบด้วย ทะเบียนครุภัณฑ์, ยืมวัสดุหรือครุภัณฑ์, คืนวัสดุหรือครุภัณฑ์, รายงานการตรวจสอบวัสดุหรือครุภัณฑ์, เพิ่มรายละเอียดอื่นๆ ระบบการ

จัดซื้อหรือจัดจ้าง บันทึกเสนอซื้อหรือจ้าง หนังสืออนุมัติเสนอซื้อหรือจ้าง ตรวจรับการจัดซื้อหรือจัดจ้าง การเบิกอุปกรณ์ รายงานการตรวจสอบ เพิ่มรายละเอียดอื่นๆ

5.3 ปัญหา ข้อเสนอแนะ และแนวทางแก้ไข

1. ระบบสารสนเทศอาคารปฏิบัติการคณะวิศวกรรมศาสตร์มหาวิทยาลัยนเรศวรมีเนื้อหา ข้อมูลมาก โครงสร้างฐานข้อมูลที่ออกแบบมานั้นอาจจะไม่สมบูรณ์ตามทฤษฎี เพราะมีข้อจำกัดหลายๆอย่างๆ จากการใช้งานที่ผู้ใช้

2. ระบบสารสนเทศที่ออกแบบมานี้การใช้งานค่อบข้างใช้งานได้ไม่สะดวกมากนัก ซึ่งในอนาคตผู้ดำเนินงานโครงงานจะแก้ไขให้ใช้งานได้สะดวกและเหมาะสมกับผู้ใช้งานมากยิ่งขึ้น

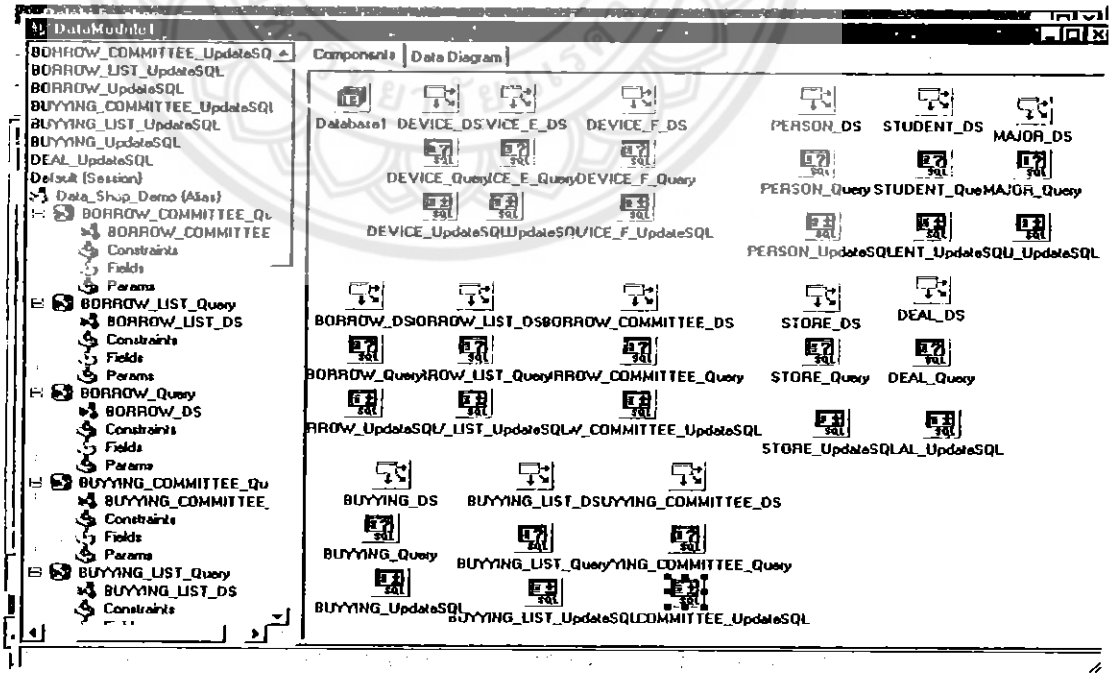
ภาคผนวก ก

การเขียนโปรแกรมโดยใช้ Data component ในการเข้าถึงฐานข้อมูล

1. การใช้ Data Module

Data Module เป็นออบเจ็คประเภทหนึ่งซึ่งปกติแล้วจะใช้เป็นคอมโพเนนต์ประเภท Data Acess โดยรวบรวมฟังก์ชันการทำงานและควบคุม Data Acess ไว้ที่เดียวกัน เพื่อสะดวกแก่การใช้งานร่วมกันตลอดทั้งแอพพลิเคชัน Data Module เป็นเหมือนฟอร์มอันหนึ่งซึ่งเราสามารถวางคอมโพเนนต์อื่นๆลงได้ เพียงแต่คอมโพเนนต์ที่จะวางลงใน Data Module ได้จะต้องเป็นคอมโพเนนต์ประเภท Non-Visual เท่านั้น เช่น Dialogs หรือ Data Acess เป็นต้น เพราะ Data Module เป็นเสมือนฟอร์มที่มองไม่เห็นนั่นเอง

ในการทำงานเราจะมอง Data Module เป็นยูนิตหนึ่งยูนิตซึ่งสามารถเรียกใช้ได้จากยูนิตอื่นๆ การใช้ Data Module จะมีประโยชน์ในกรณีที่เรามีการใช้ Table หรือ Qurey ใดๆซ้ำกันในหลายๆฟอร์มหรือหลายๆ แอพพลิเคชัน เราสามารถกำหนดคอมโพเนนต์ Table หรือ Qurey เพื่อใช้งานกับเทเบิลหรือคำสั่ง SQL เหล่านั้นไว้ใน Data Module เพียงยูนิตเดียว ซึ่งจะช่วยให้การแก้ไข ทำได้ง่ายและสะดวกแก่การใช้งานร่วมกันในหลายในหลายๆฟอร์ม



2. การใช้คอมโพเนนต์ Database

Database เป็นคอมโพเนนต์ที่ใช้ควบคุมสำหรับการทำงานกับฐานข้อมูล เช่น การติดต่อกับฐานข้อมูล การควบคุม Transaction เป็นต้น โดยปกติการเข้าถึงฐานข้อมูลต่างๆนั้น เราจะต้องกำหนดเอเลียสไว้ที่ BDE ก่อน ตัวแอพพลิเคชันถึงจะเชื่อมต่อกับฐานข้อมูลต่างๆได้ ซึ่งนอกจากนี้แล้วเราสามารถกำหนดเอเลียสไว้ที่คอมโพเนนต์ Database เพื่อใช้ในการเชื่อมต่อกับฐานข้อมูลได้โดยไม่ต้องสร้างเอเลียสไว้ที่ BDE ทำให้สามารถเข้าถึงฐานข้อมูลได้โดยไม่ต้องสร้างเอเลียสในเครื่องนั้นๆ

2.1 การติดต่อกับฐานข้อมูลด้วยคอมโพเนนต์ Database

ในการติดต่อกับฐานข้อมูลโดยใช้ Database ทำได้ 2 วิธีคือ การติดต่อกับฐานข้อมูลโดยตรง ซึ่งจะต้องกำหนดคุณสมบัติของ Driver และฐานข้อมูลที่จะติดต่อโดยจะกำหนดค่าในพร็อพเพอร์ตี้ DriverName และ Params การติดต่อผ่านเอเลียสที่กำหนดไว้ใน BDE โดยกำหนดค่าในพร็อพเพอร์ตี้ AliasName

การเข้าถึงฐานข้อมูลทำได้โดยกำหนดพร็อพเพอร์ตี้ Connected เป็น True เพื่อที่จะเข้าไปทำงานต่างๆกับฐานข้อมูล ซึ่งในการกำหนดพร็อพเพอร์ตี้ต่างๆของ Database นั้นจะสะดวกทำได้ง่ายและสะดวกโดยใช้ Database Editor

3. การควบคุม Transaction

Transaction คือ กลุ่มของขั้นตอนการทำงานใดๆ ซึ่งอาจประกอบไปด้วยขั้นตอนเดียว หรือ หลายๆ ขั้นตอนก็ได้ โดยที่งานเหล่านั้นจะต้องทำให้เสร็จทุกขั้นตอนจึงจะถือว่า Transaction สมบูรณ์ แต่ถ้าขั้นตอนใดขั้นตอนหนึ่งทำไม่สำเร็จ Transaction นั้นก็จะไม่สมบูรณ์ ตัวอย่างของการทำงานแบบ Transaction ที่คุ้นเคยในชีวิตประจำวันของเรา เช่น เมื่อเราถอนเงินจากบัญชีหนึ่ง แล้วนำเงินไปฝากอีกบัญชีหนึ่ง จะเป็นการโอนเงินระหว่าง 2 บัญชี ซึ่งถือว่าเป็นงาน 1 Transaction แต่ประกอบไปด้วยขั้นตอนการทำงาน 2 ขั้นตอน แต่ถ้าเราต้องการฝากเงินเพียงอย่างเดียว การฝากเงินก็เป็นหนึ่ง Transaction เหมือนกันแต่มีขั้นตอนการทำงานเพียงขั้นตอนเดียวเท่านั้น

แต่ละ Transactionถ้าขั้นตอนต่างๆเสร็จสมบูรณ์ จะกล่าวได้ว่าการทำงาน Transaction นั้นสมบูรณ์ แต่ถ้าขั้นตอนใดขั้นตอนหนึ่งเกิดข้อผิดพลาดขึ้น ทำงานไม่สำเร็จ ขั้นตอนที่ทำเสร็จไปก่อนหน้านี้ก็จะถูกยกเลิกทั้งหมด เช่น การโอนเงินระหว่างบัญชี ถ้าขั้นตอนในการถอนเงินได้ทำจนเสร็จเรียบร้อยแล้ว และเมื่อจะฝากเงินเข้าอีกบัญชีหนึ่ง แต่เกิดข้อผิดพลาดที่ทำให้ไม่สามารถฝากเงินเข้าบัญชีได้ การถอนเงินในบัญชีแรกก็จะถูกยกเลิกด้วย ทำให้ Transaction การโอนเงินนั้นไม่สมบูรณ์

3.1 เมธอดที่สำคัญ

StartTransaction	เป็นเมธอดสำหรับการกำหนดจุดเริ่มต้นของ Transaction
Commit	เป็นเมธอดสำหรับใช้สำหรับการกำหนดจุดสิ้นสุดของ Transaction และบอกว่าการทำงานเสร็จสมบูรณ์
Rollback	เป็นเมธอดที่ใช้ยกเลิกการทำงานภายใน Transaction ทั้งหมดซึ่งจะมีผลให้ข้อมูลต่างๆที่เก็บอยู่ในฐานข้อมูลยังมีค่าเหมือนเมื่อตอนก่อนเริ่มTransaction

4. การจัดการ Exception สำหรับฐานข้อมูล

เรื่องของการจัดการ Exception ได้เคยกล่าวไว้แล้วในบทที่ 8 ในหัวข้อการจัดการกับข้อผิดพลาดในโปรแกรม ส่วนการเขียนโปรแกรมด้านฐานข้อมูลจะมี Exception อยู่ 2 ชนิด คือ EdatabaseError และ EDBEngineError

4.1 EdatabaseError

เป็น Exception ที่เกิดขึ้นเมื่อมีข้อผิดพลาดกับฐานข้อมูล โดยตัวคอมโพเนนต์จะเป็นผู้พบปัญหานี้ ตัวอย่างเช่นการเปิดเทเบิลไม่สำเร็จ การกำหนดพารามิเตอร์ของคำสั่ง SQL ไม่ครบถ้วน เป็นต้น ตัวอย่างต่อไปนี้แสดงการจัดการ Exception ในกรณีที่เปิดเทเบิลไม่สำเร็จ

```
try
Table.Active :=True;
Except
on EdatabaseError do
ShowMessage('Open table not sucess');
end;
```

สำหรับ Exception ประเภท EdatabaseError นี้เราจะไม่ทราบสาเหตุของข้อผิดพลาดที่เกิดขึ้นว่าเป็นเพราะอะไร เนื่องจาก Delphi จะใช้ BDE ในการเข้าถึงฐานข้อมูล ดังนั้นรายละเอียดของข้อผิดพลาดที่เกิดขึ้นกับฐานข้อมูลจะอยู่ใน EDBEnineError

4.2 EDBEngineError

เป็นข้อผิดพลาดที่เกิดขึ้นใน BDE โดย Exception จะสืบทอดคุณสมบัติมาจาก EdatabaseError โดยมีพร็อพเพอร์ตี้เพิ่ม 2 ตัวคือ ErrorCount และ Errors ซึ่งเก็บรายละเอียดของข้อผิดพลาดที่เกิดขึ้น

4.3 Cached Update

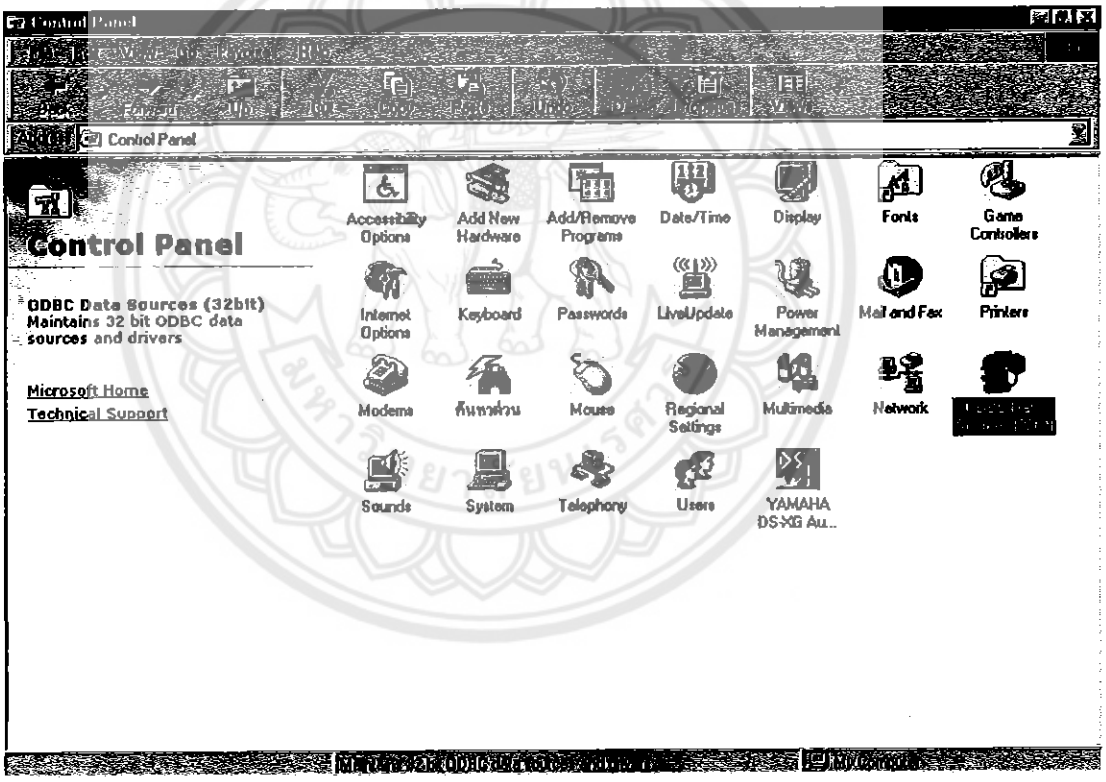
Cached Update เป็นวิธีการหนึ่งที่ทำให้เราสามารถแก้ไขข้อมูลพร้อมกันได้ทีละหลายๆเรคคอร์ด โดยพักข้อมูลไว้ในส่วนที่เรียกว่า Cache แล้วค่อยบันทึกข้อมูลทั้งหมดลงฐานข้อมูลจริงๆ การบันทึกข้อมูลเป็นกลุ่มอย่างนี้ จะเป็นการทำงานครั้งเดียวทำให้การควบคุม Transaction ได้อย่างมีประสิทธิภาพ ไม่ต้องล็อกฐานข้อมูลไว้นานๆระหว่างที่ผู้ใช้แก้ไขข้อมูล การทำงานของ Cached Update จะกำหนดได้ใน DataSet (Table,Query,StoredProc) โดยมีพร็อพเพอร์ตี้และเมธอดที่สำคัญสำหรับจัดการ Cached Update ดังนี้ CachedUpdate เป็นพร็อพเพอร์ตี้ที่ใช้สำหรับกำหนดให้ Cachd Update ทำงานหรือไม่ โดยกำหนดค่าเป็น True หรือ False ซึ่งเราจะต้องกำหนดพร็อพเพอร์ตี้นี้เป็นTrue ก่อนที่จะทำการแก้ไขข้อมูล และเมื่อกำหนดให้เป็น False จะยกเลิกและทิ้งการแก้ไขทั้งหมดไป, ApplyUpdates เป็นเมธอดที่ใช้สำหรับสั่งนำข้อมูลทั้งหมดที่แก้ไขบันทึกลงฐานข้อมูลจริงๆ, CancelUpdates เป็นการยกเลิกแก้ไขทั้งหมด

ภาคผนวก ข

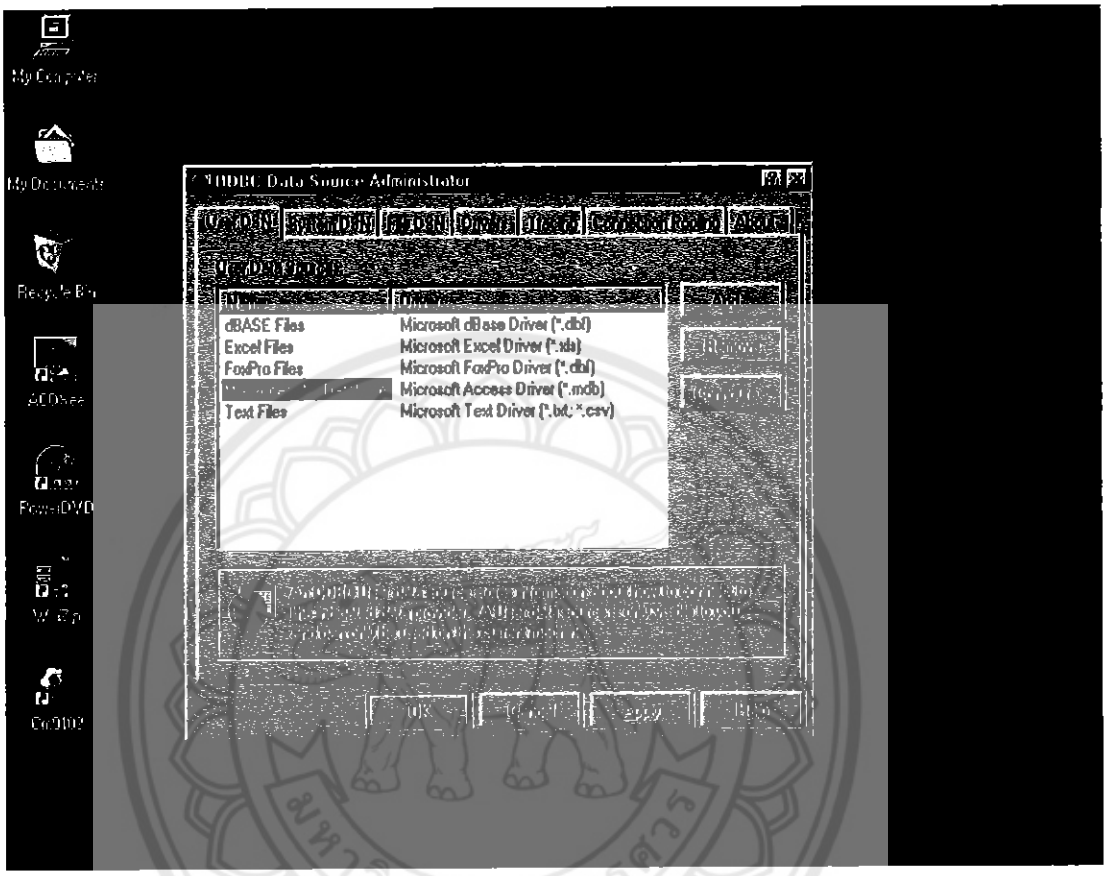
การเขียนโปรแกรมติดต่อกับฐานข้อมูล Microsoft Access

ตัวอย่างต่อไปนี้เป็นตัวอย่างในการติดต่อกับฐานข้อมูล Microsoft Access

1 เปิด Control Panal ของวินโดว์ขึ้นมา จากนั้นดับเบิ้ลคลิกที่ไอคอน ODBC Data Sources(32 bit) จะปรากฏไดอะล็อบล็อกซ์ ODBC Data Source Administrator ขึ้นมา ดังรูป

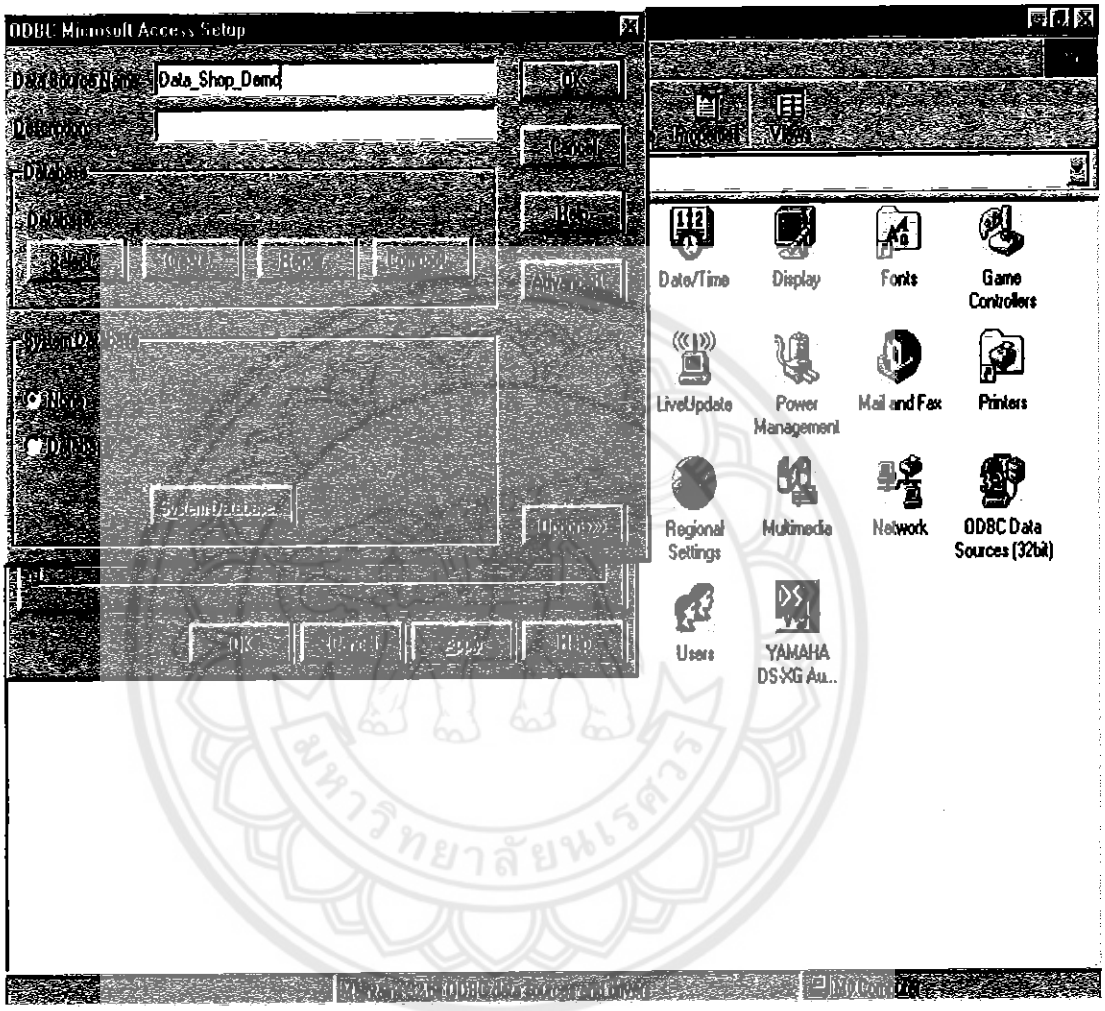


2.คลิกปุ่ม Add จะปรากฏไดอะล็อกบ็อกซ์ Create New Data Source ขึ้นมาเพื่อให้เราสร้างแหล่งข้อมูลขึ้นใหม่

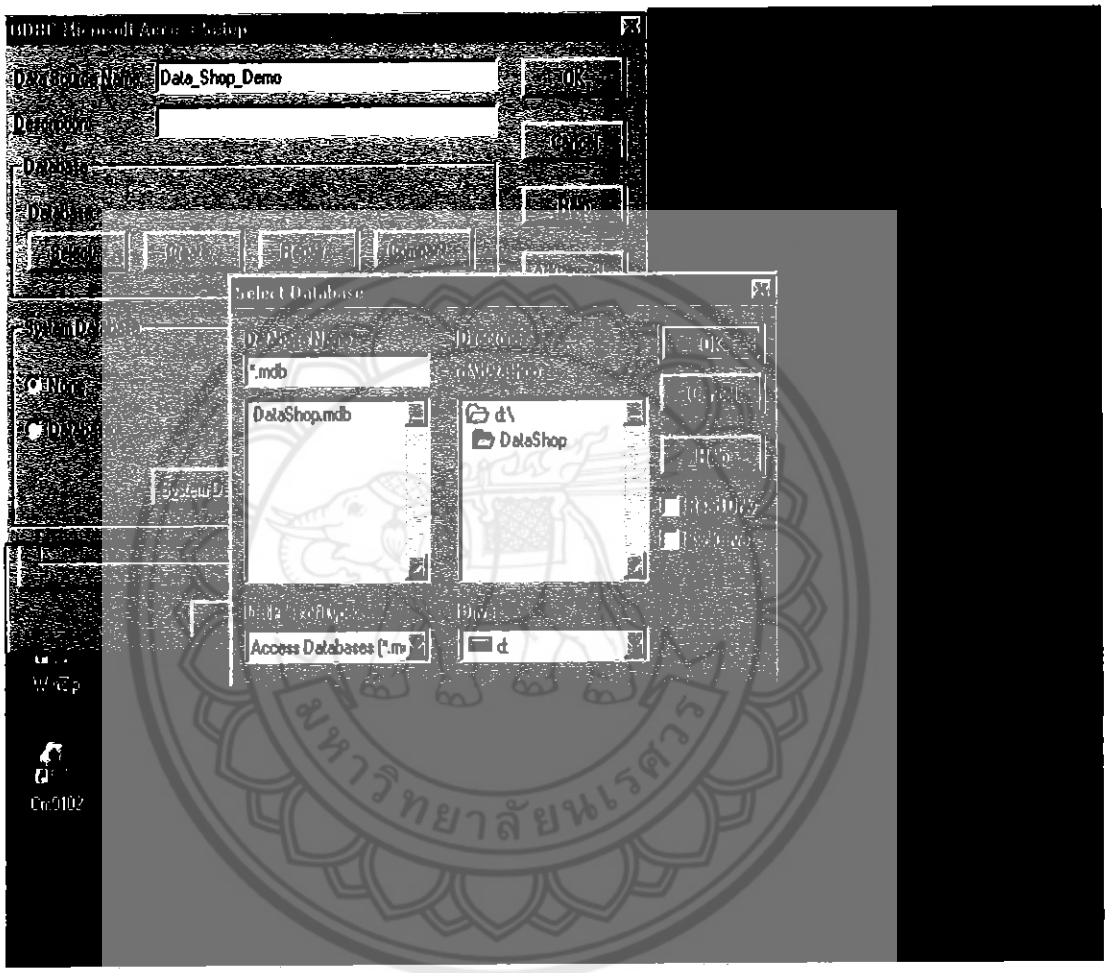


3. เลือก Microsoft Access Driver(*.mdb)แล้วคลิกที่ Finish จะปรากฏไดอะล็อกบล็อกซ์ ODBC Microsoft Access Setup ขึ้นมาเพื่อให้เรากรอกรายละเอียดต่างๆเกี่ยวกับแหล่งข้อมูลที่ต้องการจะสร้างขึ้น

4.เรากำหนดชื่อของแหล่งข้อมูลที่เราจะสร้างขึ้นในช่อง Data Source Name และคำอธิบายในช่อง Data Source Name และคำอธิบายในช่อง Description จากนั้นคลิกที่ปุ่ม Select เพื่อเลือกไฟล์ฐานข้อมูลที่ต้องการอ้างอิงถึง



5.ไฟล์ฐานข้อมูลในตัวอย่างนี้เป็นไฟล์ฐานข้อมูล Access ดังนั้นจึงมีส่วนขยายไฟล์เป็น *.mdb ในกรณีที่ฐานข้อมูลถูกเก็บไว้บนเซิร์ฟเวอร์ ก็ให้เลือกไฟล์ฐานข้อมูลจากเซิร์ฟเวอร์ เมื่อเลือกไฟล์เสร็จแล้ว คลิก OK จะกลับมายังไดอะล็อกบ็อกซ์ ODBC Microsoft Access Setup ที่มีฐานข้อมูลเป็นชื่อไฟล์ที่เราเลือก ดังรูป



6.เมื่อข้อมูลทุกอย่างถูกกำหนดให้กับแหล่งข้อมูลตัวใหม่แล้ว คลิก เพื่อปิดไดอะล็อกบ็อกซ์ ODBC Microsoft Access Setup จากนั้นไดอะล็อกบล็อกซ์ ODBC Data Source Administrator ก็จะแสดงขึ้นมาพร้อมกับชื่อ Data Source ตัวใหม่ที่เราเพิ่งสร้างขึ้นด้วย ดังรูป

ขั้นตอนทั้งหมดจะเป็นการสร้าง Data Source New MS Access ขึ้นมาเพื่อนำไปใช้งานในการสร้างแอพพลิเคชั่นฐานข้อมูลต่อไป เมื่อต้องการจบการทำงานให้คลิกปุ่ม OK

ในการสร้างแอพลิเคชันเพื่อเรียกใช้แหล่งข้อมูลที่เราสร้างขึ้นนั้นจะเหมือนกับที่เราได้สร้างมา เพียงแต่ในการเลือกฐานข้อมูลที่จะนำมาใช้ในแอพพลิเคชัน ก็ให้เราเลือกฐานข้อมูลที่เราได้สร้างมาเท่านั้น จะเป็นการเรียกใช้ข้อมูลจากแหล่งข้อมลที่เรากำหนด

เอกสารอ้างอิง

[1] กิตติ ภักดีวัฒนะกุล และ จำลอง ครูอุตสาหะ **คัมภีร์ระบบฐานข้อมูล** พิมพ์ครั้งที่ 2 กรุงเทพฯ:ไทยเจริญการพิมพ์, หจก.,2542

[2] อำไพ พรประเสริฐกุล **การวิเคราะห์และออกแบบระบบ** พิมพ์ครั้งที่ 3 กรุงเทพฯ: ศูนย์เทคโนโลยีอิเล็คทรอนิกส์และคอมพิวเตอร์แห่งชาติ

[3] ธวัชชัย สุริยะทองธรรม และธาริน สิทธิธรรมชารี **สร้างเว็บเพจอย่างไร้ขีดจำกัดด้วย ASP** กรุงเทพฯ : ซัดเซส มีเดีย จำกัด,บริษัท

[4] สังจะ จรัสรุ่งรวีวร และจักรพงษ์ สุขประเสริฐ **คู่มือเรียน Delphi 5** พิมพ์ครั้งที่ 1 กรุงเทพฯ: ดวงกมลสมัยจำกัด,บริษัท;2543.

ประวัติผู้ทำโครงงาน

ชื่อ	นาย ปริญญา เขจรรักษ์
วันเดือนปีเกิด	5 มกราคม 2522
สถานที่เกิด	จังหวัด มุกดาหาร
ประวัติการศึกษา	สำเร็จการศึกษาตอนต้น โรงเรียนบัวขาว
	จังหวัด กาฬสินธุ์ พ.ศ.2537
	สำเร็จการศึกษาตอนปลาย จากโรงเรียนบัวขาว
	จังหวัด กาฬสินธุ์ พ.ศ.2540

ชื่อ	นาย เอกมล ชูเที่ยง
วันเดือนปีเกิด	22 กันยายน 2522
สถานที่เกิด	จังหวัด สุโขทัย
ประวัติการศึกษา	สำเร็จการศึกษาตอนต้น โรงเรียนสุโขทัยวิทยาคม
	จังหวัด สุโขทัย พ.ศ.2537
	สำเร็จการศึกษาตอนปลาย โรงเรียนสุโขทัยวิทยาคม
	จังหวัด สุโขทัย พ.ศ.2540