



ระบบสารสนเทศสำหรับการบริหารสถาบันการศึกษา

EDUCATIONAL MANAGEMENT INFORMATION SYSTEM

โดย

นายรัชชัย ขุนศรี รหัส 38013274  
นายพรชัย เลิศปรีดาผดุง รหัส 38013278  
นายยุทธพล พูลสวัสดิ์ รหัส 38013286

วัน เดือน ปี..... 15.ค.ค.2541  
เลขทะเบียน..... 038974  
เลขเรียกหนังสือ..... โ.มอ.๒๕ ๗๑๙๗

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
สาขาวิศวกรรมคอมพิวเตอร์  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2540

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ใดๆ  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

038974

ปีการศึกษา 2540

ระบบสารสนเทศสำหรับงานบริหารสถาบันการศึกษา

EDUCATIONAL MANAGEMENT INFORMATION SYSTEM



อาจารย์ที่ปรึกษา

รศ.ประทีป บัญญัตินพรัตน์

อ.บรรจง ปิยธำรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทบริหารศึกษาศาสตร์ 2540

ภาควิชา วิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบสารสนเทศสำหรับงานบริหารสถาบันการศึกษา

ผู้จัดทำ

1. นายรัชชัย ขุนศรี 38013274
2. นายพรชัย เลิศปรีดาผดุง 38013278
3. นายบุทรพล พูลสวัสดิ์ 38013286



.....อาจารย์ที่ปรึกษา

(รศ.ประทีป บัญญัตินพรัตน์)



.....อาจารย์ที่ปรึกษา

(อ.บรรจง ปิยะธำรง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ระบบสารสนเทศสำหรับงานบริหารสถาบันการศึกษา

ธวัชชัย ขุนศรี 38013274

พรชัย เลิศปริดาผดุง 38013278

ยุทธพล พูลสวัสดิ์ 38013286

รศ.ประทีป บัญญัติสินพรัตน์ อาจารย์ที่ปรึกษา

อ.บรรจง ปิยะธำรง อาจารย์ที่ปรึกษา

ปีการศึกษา 2540

## บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้ เรียบเรียงขึ้นจากผลงานการพัฒนาระบบฐานข้อมูลเชิงสัมพันธ์ แบบไคลเอ็นเซิร์ฟเวอร์ เพื่อใช้ในการทำงานของระบบสารสนเทศระบบสารสนเทศสำหรับงานบริหารสถาบันการศึกษา ซึ่งประกอบไปด้วยการกรอกข้อมูลเบื้องต้น การบันทึกข้อมูล การสอบถามข้อมูล และการพิมพ์รายงาน

การพัฒนา ดังกล่าวใช้ระบบจัดการฐานข้อมูลเชิงสัมพันธ์ไมโครซอฟต์ เอสคิวแอล เซิร์ฟเวอร์เวอร์ชัน 6.5 และเดลฟี (DELPHI) ในการเขียนโปรแกรมส่วนประมวลผลส่วนหน้า (Front-End) หรือส่วนติดต่อกับผู้ใช้ (User Interface) เป็นแบบกราฟฟิก โดยมีไดรฟ์เวอร์โอดีบีซี (ODBC : Open Database Connectivity) ทำหน้าที่ในการติดต่อระหว่างส่วนประมวลผลส่วนหน้าและส่วนจัดการฐานข้อมูล ในการพัฒนาระบบเริ่มจาก การหาความต้องการของผู้ใช้ การวิเคราะห์ ออกแบบ และติดตั้งในสโคปกว้าง ๆ

# EDUCATIONAL MANAGEMENT INFORMATION SYSTEM

Thawatchai Kunsree 38013274

Pornchai Lerdpredapadung 38013278

Yutthapol Phoonsawat 38013286

Assoc. Prof Prateep Bunyatnopparat (Advisor)

Bunjong Piyathumrong (Advisor)

1997

## Abstract

This thesis is a collection of the work on client/server relational database development on the information system and management, including basic entering data, noting a database, giving information and displaying reports.

This database development is implemented by MS SQL Server Version 6.5 relational database system and the front-end or user interface is programming in DELPHI which is design to be graphic. The system has an ODBC (ODBC : Open Database Connectivity ) driver connectivity the front-end and database server. This system is implementing new system work by take requirement, analysis, design, and installation to make the system wide scope but cover requirement.

# สารบัญ

	หน้า
บทที่ 1 บทนำ	1
บทที่ 2 ความรู้และทฤษฎีพื้นฐาน	3
2.1 โมเดลเชิงสัมพันธ์ ( Relational Model )	3
2.2 ระบบการจัดการฐานข้อมูลเชิงสัมพันธ์	7
2.3 การออกแบบฐานข้อมูลรวมแบบรีเลชันนอล	8
2.4 ระบบสารสนเทศเพื่อการจัดการ	9
2.5 การออกแบบฐานข้อมูลในรูปแบบบรรทัดฐาน	11
2.6 การออกแบบฐานข้อมูลโดยใช้วิธีในแอม	19
2.7 ระบบไคลเอ็นท์เซิร์ฟเวอร์	26
2.8 โอเพนดาต้าเบสคอนเน็คติวิตี	27
2.9 ความรู้เกี่ยวกับเครื่องมือและลักษณะของคลไฟล์ กับดาต้าเบสแบบ two-tiered	30
2.10 ระบบปฏิบัติการ Windows NT 4.0	33
2.11 แอปพลิเคชัน	35
บทที่ 3 การดำเนินงานและพัฒนาระบบ	42
3.1 ลักษณะและความสำคัญ	42
3.2 หลักการและเหตุผล	42
3.3 วัตถุประสงค์ของโครงการ	42
3.4 สิ่งที่ได้จากการนำระบบงานสารสนเทศ เพื่อการจัดการมาใช้	43
3.5 หน้าที่หลักของระบบสารสนเทศ	43
3.6 ความต้องการของผู้ใช้	43
3.7 ข้อมูลและสารสนเทศที่เป็นความต้องการขององค์กร	43
3.7.1 ฝ่ายส่งเสริมการศึกษา	43
3.7.2 ฝ่ายกิจกรรมนักศึกษา	46
3.7.3 ฝ่ายวางแผนและพัฒนา	48
3.7.4 ฝ่ายวิชาการ	49
3.8 ข้อเพิ่มเติมของระบบ	52
3.9 ขั้นตอนในการพัฒนาระบบ	52
3.10 การเขียนโค้ดโปรแกรม	54

# สารบัญ (ต่อ)

	หน้า
3.11 สรุปปัญหาและการดำเนินการ	66
บทที่ 4 การทดลองและผลการทดลอง	68
4.1 ฮาร์ดแวร์และซอฟต์แวร์ที่ใช้ในการทดลอง	68
4.2 ความต้องการทางด้านซอฟต์แวร์	68
4.3 การทดลองและผลการทดลอง	69
การทดลองที่ 1 การบันทึกและการอ่านข้อมูลจากฐานข้อมูล	69
การทดลองที่ 2 การแก้ไขข้อมูลเดิม	70
บทที่ 5 สรุป วิเคราะห์ และแนวทางในการพัฒนา	72
5.1 ความสามารถของโปรแกรม	72
5.2 ข้อจำกัดของโปรแกรม	72
5.3 แนวทางในการพัฒนาต่อไป	72
ภาคผนวก	
กิตติกรรมประกาศ	
บรรณานุกรม	

## บทคัดย่อ

ระบบสารสนเทศสำหรับงานบริหารสถาบันการศึกษา โรงเรียนเทคโนโลยีชลบุรีเป็นระบบที่จัดการเกี่ยวกับงานทางด้านการศึกษา และการบริหารงาน ของโรงเรียนเทคโนโลยีชลบุรี ซึ่งประกอบด้วย 4 ฝ่ายดังต่อไปนี้

1. ฝ่ายวิชาการ
2. ฝ่ายวางแผนและพัฒนา
3. ฝ่ายกิจกรรมนักศึกษา
4. ฝ่ายส่งเสริมการศึกษา

ซึ่งจะนำมาใช้แทนระบบเดิมของโรงเรียนเทคโนโลยีชลบุรี โดยข้อมูลจะถูกจัดเก็บไว้อย่างเป็นระบบ จะช่วยเพิ่มขีดความสามารถในการทำงาน และลดเวลาการทำงานของเจ้าหน้าที่ และผู้มาติดต่อ โดยเป็นระบบที่ออกแบบมาให้ทำงานบนระบบ Windows NT และมีความสามารถในการเก็บข้อมูลโดยใช้ระบบฐานข้อมูลเชิงสัมพันธ์ ( Relational Database System )

ในการพัฒนาระบบฐานข้อมูลดังกล่าว จะใช้ระบบการจัดการฐานข้อมูลเชิงสัมพันธ์ ( Relational Database Management System :RDBMS ) ของบริษัท Microsoft ที่ชื่อว่า Microsoft SQL Server Version 6.5 และใช้ Delphi ของบริษัท Borland ในการเขียนโปรแกรมส่วนที่เป็น Graphic User Interface (GUI) หรือ Font End ของระบบ โดยมี Driver ODBC (Open Database Connectivity) เป็น Driver ที่ทำหน้าที่ในการติดต่อระหว่าง Font End กับ Back End ซึ่งระบบงานนี้จะทำงานภายใต้ระบบ LAN (Local Area Network)

# บทที่ 1

## บทนำ

ปัจจุบันระบบงานในองค์กรต่างๆ มักมีข้อมูลที่มีความสำคัญต่อองค์กรเป็นจำนวนมาก จึงต้องมีการทำการจัดการฐานข้อมูล เนื่องจากสามารถทำให้ระบบการจัดเก็บข้อมูลเป็นได้อย่างสะดวก สามารถควบคุมความถูกต้องและความปลอดภัยของข้อมูลได้เป็นอย่างดี พร้อมทั้งยังสามารถอำนวยความสะดวกในการเรียกใช้และแก้ไขข้อมูลได้อย่างดียิ่ง

ระบบการจัดการฐานข้อมูลเชิงสัมพันธ์ ( Relational Database ) เป็นแนวทางหนึ่งที่มีผู้นิยมใช้กันมาก ดังนั้นจึงมีความจำเป็นที่จะต้องศึกษาการออกแบบฐานข้อมูลเชิงสัมพันธ์เกี่ยวกับข้อกำหนด ขั้นตอนการออกแบบ และการประยุกต์ไปใช้ในการแก้ปัญหาในระบบงาน

การเชื่อมต่อระบบเน็ตเวิร์กแบบไคลเอ็นท์เซิร์ฟเวอร์เหมาะกับการเชื่อมโยงเครือข่ายขนาดใหญ่ ซึ่งจะเป็นเครื่องไมโครคอมพิวเตอร์ที่ประกอบด้วยเซิร์ฟเวอร์ ( Server ) ทำหน้าที่เป็นผู้ให้บริการ และไคลเอ็นท์ ( Client ) เป็นผู้ขอใช้บริการจากเซิร์ฟเวอร์ การเชื่อมต่อแบบนี้มักจะนำไปใช้งานกับองค์กรขนาดใหญ่มีปริมาณข้อมูลจัดเก็บหรือใช้งานอยู่ในจำนวนมาก โดยจัดเก็บฐานข้อมูล ( Database ) อยู่ที่เซิร์ฟเวอร์และเครื่องไคลเอ็นท์ อื่นๆ จะดึงข้อมูลจากเซิร์ฟเวอร์ไปใช้

ระบบเน็ตเวิร์กแบบไคลเอ็นท์เซิร์ฟเวอร์มีค่าใช้จ่ายที่สูง โดยเฉพาะเครื่องเซิร์ฟเวอร์ที่ต้องรับงานหนักเป็นพิเศษ นอกจากค่าใช้จ่ายจะสูงแล้วระบบการทำงานจะมีความซ้ำซ้อน เนื่องจากระบบไคลเอ็นท์เซิร์ฟเวอร์ต้องมีการบริหารและวินิจฉัยปัญหาที่เกิดขึ้นกับกาทำงานอยู่อย่างเสมอ อย่างไรก็ตามระบบไคลเอ็นท์เซิร์ฟเวอร์ยังมีลักษณะที่ดีดังนี้

1. ความรวดเร็วสูงเพื่อให้สามารถตอบสนองความต้องการของผู้ใช้ได้หลายคนพร้อมๆกัน
2. มีมาตรการรักษาความปลอดภัยให้ข้อมูลมากเป็นพิเศษ เพื่อรองรับผู้รัยย่อยจำนวนมาก
3. ระบบบริหารที่มีประสิทธิภาพ เนื่องจากทรัพยากรหลักต่างๆถูกรวมไว้ที่เซิร์ฟเวอร์
4. สามารถขยายระบบงานได้สะดวก เป็นการยอมรับการเจริญเติบโตในอนาคตของหน่วยงาน

สำหรับการพัฒนาระบบงาน เริ่มจากการศึกษาระบบงานเดิม เพื่อหาความต้องการของผู้ใช้วิเคราะห์ระบบ ออกแบบระบบ และติดตั้งระบบให้สามารถใช้งานได้อย่างมีประสิทธิภาพมากที่สุด

รายละเอียดของปริญญาานิพนธ์ฉบับนี้ สามารถแบ่งได้ตามหัวข้อดังต่อไปนี้

บทที่ 1 บทนำกล่าวถึงรายละเอียดเกี่ยวกับปริญญาานิพนธ์ฉบับนี้

บทที่ 2 ทฤษฎีหลักการที่นำมาใช้ในการประกอบการพัฒนาระบบ เช่น แนวคิดเกี่ยวกับระบบการจัดการฐานข้อมูลแบบรีเลชันนอล การออกแบบฐานข้อมูลด้วยไนแอม ( NIAM ) การออกแบบฐานข้อมูลให้อยู่ในรูปบรรทัดฐานขั้นที่ห้า ( fifth normalization ) ความรู้เกี่ยวกับเอสคิวแอลเซิร์ฟเวอร์ ( SQL Server ) และความรู้เกี่ยวกับแคลไฟล์ซึ่งใช้ในการออกแบบการติดต่อกับผู้ใช้ ( User Interface )

- บทที่ 3 การดำเนินการและการพัฒนาระบบทั้งหมดในโครงการ ซึ่งประกอบด้วยการทำงานแอททริบิว ที่เกี่ยวข้องกับระบบการออกแบบโครงสร้างฐานข้อมูล และอัลกอริทึมในการเขียนโปรแกรม
- บทที่ 4 การทดสอบและผลการทดลอง ซึ่งอธิบายการเก็บข้อมูลลงฐานข้อมูล การค้นหา การเพิ่มเติมข้อมูล การแก้ไขข้อมูล
- บทที่ 5 บทวิจารณ์และสรุป



## บทที่ 2

# ความรู้และทฤษฎีพื้นฐาน

### 2.1 โมเดลเชิงสัมพันธ์ (Relational Model)

โมเดลเชิงสัมพันธ์เป็นโมเดลที่ใช้ในการอธิบายความสัมพันธ์ของข้อมูลที่ถูกเก็บด้วยระบบการจัดการฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management System : RDMS) ซึ่งเป็นผลงานของ ดร.คอดด์ (Codd) เสนอผลงานวิจัยในปี พ.ศ. 2513 โดยมีแนวความคิดในการพัฒนามาจากทฤษฎีทางคณิตศาสตร์ นั่นคือคิดว่าการแก้ปัญหาถ้าใช้ทฤษฎีทางคณิตศาสตร์ที่ถูกต้องแล้วไม่ว่าจะหาคำตอบมาจากวิธีการใดย่อมจะต้องได้ผลลัพธ์ที่เหมือนกัน ในปัจจุบันนี้มีการนำโมเดลนี้มาใช้กับเครื่องระดับเมนเฟรมลงไปถึงเครื่องระดับไมโครคอมพิวเตอร์ด้วย และก็เป็นที่ยอมรับกันแล้วว่า บรรดาผู้ใช้ระบบฐานข้อมูลจะมีความคุ้นเคยกับโมเดลเชิงโครงข่าย (Network Model) ที่มีมาก่อนหน้านี้

#### 2.1.1 การเปรียบเทียบระหว่างโมเดลเชิงสัมพันธ์กับโมเดลอื่น ๆ

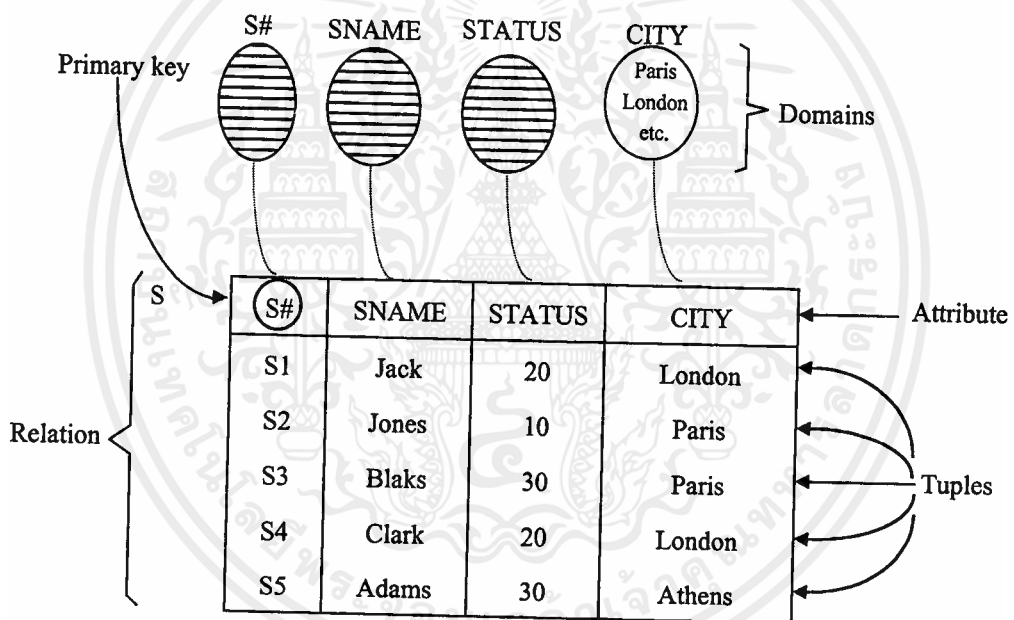
นอกเหนือจากความแพร่หลายของ โมเดลเชิงสัมพันธ์แล้ว ข้อดีของโมเดลเชิงสัมพันธ์ที่มีมากกว่าอีก 2 โมเดล มีดังนี้

1. โมเดลเชิงสัมพันธ์เป็นโมเดลที่สร้างความเข้าใจได้ง่ายกว่า เพราะภาพลักษณ์ของข้อมูลที่เก็บโดยโมเดลเชิงสัมพันธ์จะมาจากมุมมองของผู้ใช้ ซึ่งจะมีความซับซ้อนน้อยกว่าภาพลักษณ์ของข้อมูลที่เก็บโดยอีก 2 โมเดล
2. ระบบส่วนใหญ่ที่ใช้โมเดลเชิงสัมพันธ์นี้มักจะมีเครื่องมือที่ช่วยให้ผู้ใช้สามารถจัดการกับข้อมูลที่จัดเก็บด้วย โมเดลแบบอื่น
3. โมเดลเชิงสัมพันธ์นี้มีเครื่องมือที่ช่วยให้ผู้ใช้สามารถค้นพบปัญหาที่เกิดขึ้น ในการออกแบบระบบฐานข้อมูลได้โดยง่าย และยังง่ายในการแก้ไขการออกแบบที่ผิดพลาดนั้นด้วย
4. โมเดลเชิงสัมพันธ์เป็นโมเดลที่มีความสอดคล้องกับหลักการของฐานข้อมูล ผู้ใช้ไม่ต้องพะวงกับรายละเอียดของการจัดเก็บข้อมูลเหมือนกับการจัดข้อมูลของโมเดลอื่น
5. ภาษาที่ใช้ในการจัดการกับข้อมูลที่จัดเก็บระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (ภาษา SQL : Structure Query Language) เป็นภาษาแบบเซต โอเรียนต์ (set oriented) ซึ่งจะต่างกับภาษาที่ใช้ในการจัดการกับข้อมูลที่จัดเก็บด้วยระบบจัดการฐานข้อมูลของโมเดลอื่นที่เป็นภาษาแบบ record-at-a-time

แม้ว่าโมเดลเชิงสัมพันธ์จะมีข้อดีหลายประการดังกล่าว แต่ในปัจจุบันก็ยังมีจุดอ่อนที่มีการอ้างอิงถึงเสมอ คือ ระบบจัดการฐานข้อมูลแบบโมเดลเชิงสัมพันธ์นี้มักจะมีประสิทธิภาพในการใช้งานผู้ใช้ อีก 2 โมเดลไม่ได้ โดยเฉพาะในการประยุกต์ใช้งานขององค์กรขนาดใหญ่ ก็ได้มีการแย่งกลับในแง่ที่ว่า โมเดลเชิงสัมพันธ์เป็นโมเดลที่อายุการพัฒนาน้อยกว่าอีก 2 โมเดล จึงเป็นไปได้ว่าการพัฒนาที่ผ่านมาของ โมเดลเชิงสัมพันธ์ก็ยังมีจำนวนระดับขั้นที่ได้พัฒนาไปแล้วน้อยกว่าอีก 2 โมเดล ดังนั้นหากต้องการเปรียบเทียบการทำงานระหว่างโมเดลเชิงสัมพันธ์กับโมเดลอื่นควรทำการเปรียบเทียบที่ระดับการพัฒนาที่เท่ากัน

### 2.1.2 ศัพท์เฉพาะของโมเดลเชิงสัมพันธ์

จากการที่ข้อมูลที่เก็บด้วยโมเดลเชิงสัมพันธ์จะถูกเก็บไว้ในตารางที่ถูกเรียกว่า “รีเลชัน” โดยที่รีเลชันทุกรีเลชันอาจแสดงได้ในรูปของตาราง แต่ตารางบางตารางอาจไม่เป็นรีเลชันก็ได้



รูปที่ 2.1 แสดง โครงสร้างฐานข้อมูลแบบรีเลชัน

ตารางที่มีลักษณะเป็นรีเลชันจะต้องมีคุณสมบัติดังนี้

1. แต่ละช่องของตารางจะบรรจุข้อมูลได้เพียงค่าเดียว
2. ชื่อหัวข้อในแต่ละคอลัมน์มีความแตกต่างกัน อันเป็นชื่อของแอตทริบิวของเอนทิตี
3. ค่าข้อมูลในแต่ละคอลัมน์ คือ ค่าของแอตทริบิวของเอนทิตี
4. การเรียงลำดับคอลัมน์ไม่ถือว่ามีความสำคัญ
5. ข้อมูลแต่ละแถวจะต้องแตกต่างกัน

## 6. การเรียงลำดับแถวไม่ถือว่ามีความสำคัญ

ตารางที่มีคุณสมบัติดังกล่าวจะเรียกว่า รีเลชัน ดังนั้นเราจะได้นิยามของฐานข้อมูลเชิงสัมพันธ์ คือ ฐานข้อมูลที่เกิดจากการรวบรวมรีเลชันต่าง ๆ ที่มีความสัมพันธ์ระหว่างกัน

เราจะเรียกข้อมูลแต่ละแถวในแนวนอนของรีเลชันว่า ทัพเพิล (Tuple) และเรียกข้อมูลในแต่ละแถวในแนวตั้ง หรือแนวคอลัมน์ว่า แอททริบิว (Attribute) โดยที่คำว่า คีย์ (Key) จะหมายถึงข้อมูลที่เกิดจากแอตทริบิว 1 ตัวหรือหลายตัวก็ได้

แต่ละรีเลชันจะต้องมีสิ่งทีเรียกว่า คีย์หลัก (Primary Key) คือ คีย์คู่แข่ง (Candidate Key) ตัวที่เลือกขึ้นมา ซึ่งมักจะเลือกตัวที่สั้นกว่า กระทัดรัดที่สุด เพื่อประโยชน์ในการอ้างอิงและโอกาสป้อนข้อมูลผิดพลาดจะผิดพลาดน้อยลง

คีย์คู่แข่ง (Candidate Key) หรือ คีย์สำรอง (Alternate Key) คือกลุ่มของแอตทริบิวที่มีคุณสมบัติดังนี้

1. Uniqueness คือ มีค่า (value) ไม่ซ้ำ 2 แถว (tuple) ใด ๆ
2. Minimally คือกลุ่มของแอตทริบิวที่มีจำนวนน้อยที่สุด ที่จะทำให้คุณสมบัติ Uniqueness

หรือถ้าจะกล่าวให้ง่ายก็คือ แอททริบิว 1 ตัวหรือมากกว่า 1 ตัวก็ได้ ที่สามารถใช้เป็นตัวบอกได้ว่าเมื่อเราอ้างอิงแอตทริบิวชุดนี้แล้วเราจะได้ทัพเพิลที่ไม่ซ้ำกัน และแอตทริบิวอื่น ๆ ที่เหลือที่มีได้เป็นคีย์หลัก และไม่ได้เป็นส่วนใดส่วนหนึ่งของคีย์หลัก ก็จะถูกเรียกว่าเป็น Nonkey Attribute

คำว่า Null Value คือสิ่งที่มีลักษณะดังนี้

1. เป็นแอตทริบิวที่เกี่ยวข้องแต่ไม่ทราบค่า
2. เป็นแอตทริบิวที่ไม่เกี่ยวข้องกับแอตทริบิวตัวอื่นในทัพเพิลนั้นเลย

ส่วนคีย์ที่เป็นแอตทริบิวของรีเลชันอื่นที่ซ้ำกับแอตทริบิวที่เป็นคีย์หลักของรีเลชันหนึ่งจะเรียกว่าคีย์นอก (Foreign Key)

คำว่า โดเมน (Domain) จะหมายถึง กรอบของค่าต่าง ๆ ที่เป็นไปได้ เช่น โดเมนของแอตทริบิว วันที่ ก็จะหมายถึงวันที่ที่เป็นไปได้ คือมีค่าเท่ากับ 1 ถึง 31

แต่ในการเก็บค่าข้อมูลลงในรีเลชันนั้น บางกรณีที่เรามีการกำหนดโดเมนให้แอตทริบิวแล้ว แต่ข้อมูลที่จะถูกเก็บเข้าไปอาจถูกบรรจุเข้าไปในภายหลัง ลักษณะนี้จะทำให้เกิด ค่าว่าง (Null Value) ขึ้นชั่วขณะก่อนที่จะมีการบรรจุค่าข้อมูลที่อยู่ในโดเมนที่กำหนดไว้เข้าไป

เมื่อมีการจัดเก็บข้อมูลในฐานข้อมูลใด ๆ แล้วข้อมูลจะถูกแยกออกเป็นกลุ่มของข้อมูลเป็นชุดที่ประกอบด้วยแอตทริบิวต่าง ๆ ที่มีความสัมพันธ์กัน เช่น การเก็บข้อมูลของบุคลากรในโรงเรียน ก็อาจแยกเก็บเป็นกลุ่มของข้อมูล นักเรียน กลุ่มของข้อมูลอาจารย์ และกลุ่มของนักรักษาการโรง เป็นต้น กลุ่มของข้อมูลแต่ละกลุ่มนี้จะเรียกว่าเอนติตี้ (Entity) ซึ่งแต่ละเอนติตี้จะประกอบไปด้วยแอตทริบิวต่าง ๆ ที่มีความสัมพันธ์กัน เช่น เอนติตี้ของนักเรียนก็จะประกอบไปด้วย ชื่อ ที่อยู่ ชั้นเรียน เป็นต้น โดยแต่ละเอนติตี้จะมีความสัมพันธ์กัน

### 2.1.3 ความสัมพันธ์ระหว่างเอนทิตีแบ่งออกเป็น 3 ชนิด คือ

- ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (One to One)
- ความสัมพันธ์แบบหนึ่งต่อกลุ่ม (One to Many)
- ความสัมพันธ์แบบกลุ่มต่อกลุ่ม (Many to Many)

#### 2.1.3.1 ความสัมพันธ์แบบหนึ่งต่อหนึ่ง

ความสัมพันธ์แบบหนึ่งต่อหนึ่งระหว่างเอนทิตี หมายความว่า เมื่อเอนทิตีหนึ่งมีข้อมูลของคีย์หลักค่าหนึ่ง ค่าของข้อมูลดังกล่าวก็จะมีความสัมพันธ์กับค่าข้อมูลคีย์หลักของอีกเอนทิตีหนึ่ง เพียงค่าเดียวเท่านั้น เช่น หากเรากำหนดให้ความสัมพันธ์ระหว่างเอนทิตีนักเรียนกับเอนทิตีผู้ปกครองสามารถอ้างอิงถึงผู้ปกครองได้เพียงคนเดียวเท่านั้น และในทางตรงกันข้ามก็ต้องเป็นจริงด้วย คือ เมื่อเราอ้างอิงถึงผู้ปกครองคนใดคนหนึ่งแล้วก็สามารถอ้างอิงถึงนักเรียนคนเดียวเท่านั้น

ชื่อนักเรียน	ชื่อผู้ปกครอง
A	a
B	b
C	c

#### 2.1.3.2 ความสัมพันธ์แบบหนึ่งต่อกลุ่ม

ความสัมพันธ์แบบหนึ่งต่อกลุ่มระหว่างเอนทิตี หมายความว่า เมื่อเอนทิตีหนึ่งมีข้อมูลของคีย์หลักค่าหนึ่ง ค่าข้อมูลดังกล่าวก็จะมีความสัมพันธ์กับข้อมูลของคีย์หลักของอีกเอนทิตีหนึ่ง ได้หลายค่า เช่น หากเรากำหนดให้ความสัมพันธ์ระหว่างเอนทิตีนักเรียนกับเอนทิตีผู้ปกครองเป็นแบบหนึ่งต่อกลุ่มแล้ว จะหมายความว่า การที่เราอ้างอิงถึงนักเรียนคนใดคนหนึ่ง ก็จะสามารถอ้างอิงถึงผู้ปกครองได้เพียงคนเดียวเท่านั้น และในทางตรงกันข้ามก็จะมีผลหมายความว่า เมื่อเราอ้างอิงถึงผู้ปกครองคนใดคนหนึ่งแล้วก็สามารถอ้างอิงถึงนักเรียนคนเดียวเท่านั้น แต่ผู้ปกครองที่เราอ้างอิงถึงอาจจะอ้างอิงถึงนักเรียนคนเดียวก็เป็นได้

ชื่อนักเรียน	ชื่อผู้ปกครอง
A	a
B	a
C	c

#### 2.1.3.3 ความสัมพันธ์แบบกลุ่มต่อกลุ่ม

ความสัมพันธ์แบบกลุ่มต่อกลุ่มระหว่างเอนทิตี หมายความว่า ค่าข้อมูลคีย์หลักของเอนทิตีหนึ่งที่แตกต่างกันอาจอ้างอิงถึงค่าข้อมูลของคีย์หลักของอีกเอนทิตีหนึ่งได้ค่าเดียวกันหรือหลายค่าก็ได้ เช่น หากเรากำหนดให้ความสัมพันธ์ระหว่างเอนทิตีนักเรียนกับเอนทิตีผู้ปกครองเป็นแบบกลุ่มต่อกลุ่มแล้วจะหมายถึง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความว่า การที่อ้างอิงถึงนักเรียนคนใดคนหนึ่งหรือหลายคน ก็จะสามารถอ้างอิงถึงผู้ปกครองคนเดียว  
กันได้ และในทางกลับกัน เมื่อเราอ้างอิงถึงผู้ปกครองคนหนึ่งหรือหลายคนก็สามารถอ้างอิงถึงนักเรียนคน  
เดียวกันก็ได้

ชื่อนักเรียน	ชื่อผู้ปกครอง
A	a
B	a
C	c
C	d

นอกจากนี้ยังมีเอนทิตีบางประเภทที่การอ้างอิงตัวมันได้อย่างสมบูรณ์นั้นจะต้องอ้างอิงเอนทิ  
ตีอื่นเสมอ เช่น หากระบบฐานข้อมูลของบุคลากรในโรงเรียนมีเอนทิตีของลูกนัการการ ไรงด้วยแล้ว การ  
อ้างอิงเอนทิตีนี้ไ้จะต้องอ้างอิงเอนทิตีนัการการ ไรงด้วยเสมอ เอนทิตีประเภทนี้เราเรียกว่า เอนทิตีชนิด  
อ่อน (Weak entity)

## 2.2 ระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management System)

การวิเคราะห์ฐานข้อมูลใด ๆ ว่าเป็นระบบฐานข้อมูลเชิงสัมพันธ์ (Relational Database  
System) หรือไม่ มีสิ่งต้องพิจารณา 3 ประการดังนี้

### 2.2.1 โครงสร้าง ฐานข้อมูล (Data Structure)

โครงสร้างข้อมูลในระดับ ลอจิคอล (Logical Structure) ที่ใช้ต้องเป็นริเลขันเท่านั้น ไม่มี  
พอยน์เตอร์ (Pointer) หรืออินเดกซ์ (Index) มาใช้อยู่ด้วย โดยจะใช้ตารางมาแสดงความสัมพันธ์ระหว่าง  
ข้อมูลได้ และต้องมีคุณสมบัติดังที่ได้กล่าวไปแล้วข้างต้น

### 2.2.2 กฎที่ใช ควบคุม (Integrity constrain)

เป็นกฎที่ใช้ควบคุมโครงสร้างข้อมูล ประกอบด้วย

2.2.2.1 เอนทิตีอินทิกริตี (Entity Integrity) กล่าวว่า แอททริบิวที่เป็นส่วนของคีย์หลักจะต้องไม่  
เป็นค่าว่าง ทั้งนี้เพราะคีย์หลักเปรียบเสมือนสิ่งที่ใช้ระบุตำแหน่งในระดับลอจิคอล (Logical Address) ของ  
แถวหรือทัพเพิลนั้น ถ้าคีย์หลักเป็นค่าว่างจะทำให้เราไม่สามารถเข้าถึงข้อมูลซึ่งอยู่ในแถวนั้นบนระดับลอจ  
ิคอล ได้ อีกเหตุผลหนึ่งคือ ถ้าคีย์หลักเป็นค่าว่างซ้ำกันก็จะทำให้สูญเสียคุณสมบัติของความไม่ซ้ำ  
(Uniqueness) ไป

2.2.2.2 เรฟเฟอเรนเชียลอินทิกริตี (Referential Integrity) กล่าวว่าแอททริบิวที่เป็นคีย์นอก  
(Foreign key) ต้องมีค่าตรงกับคีย์หลัก (Primary key) หรือไม่ก็มีค่าเป็นค่าว่าง (Null Value)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.2.3 ภาษาจัดการฐานข้อมูล (Data Manipulation)

ภาษาที่ใช้ในการจัดการข้อมูลต้องเป็นภาษารีเลชันแนลคอมพลีท (Relational Complete Language) คืออย่างน้อยต้องมีความสามารถเทียบเท่ารีเลชันแนลอัลจีบรา (Relational Algebra) หรือรีเลชันแนลแคลคูลัส (Relational Calculus)

### 2.3 การออกแบบฐานข้อมูลรวมแบบรีเลชันแนล

การจัดเก็บข้อมูลจำนวนมาก โดยใช้เทคโนโลยีฐานข้อมูลช่วยในการจัดเก็บต้องคำนึงถึงวิธีการออกแบบระบบฐานข้อมูล และระบบจัดการฐานข้อมูล (DBMS) ที่เหมาะสม

ปัจจุบันวิธีการออกแบบฐานข้อมูลที่รู้จักกันอย่างแพร่หลายได้แก่ การออกแบบฐานข้อมูลในรูปแบบบันทึกรฐาน (Normalization) ซึ่งเป็นวิธีที่มีแนวคิดในการปรับปรุงคุณสมบัติของรีเลชันเป็นขั้นตอน อย่างมีระบบเพื่อลดโอกาสที่จะเกิดความผิดพลาดเนื่องจากการเปลี่ยนแปลงข้อมูล การออกแบบฐานข้อมูลด้วยวิธีนอร์มัลไลเซชันนี้ เริ่มต้นด้วยการกำหนดยูนิเวอร์แซลรีเลชัน (Universal Relation) พร้อมทั้งกำหนดความสัมพันธ์ระหว่างแอททริบิวต์ (Attribute) ต่าง ๆ ของยูนิเวอร์แซลรีเลชันในรูปของฟังก์ชันแนลดีเพนเดนซ์ (Functional Dependency) มัลติแวลลูดีเพนเดนซ์ (Multivalued Dependency) และจอยน์ดีเพนเดนซ์ (Join Dependency) แล้วทำให้รีเลชันในแต่ละขั้นตอนมีคุณสมบัติตรงตามที่กำหนดไว้ ตั้งแต่ขั้นตอนที่ 1 ถึงขั้นตอนที่ 5 ผลที่ได้ในขั้นตอนสุดท้ายจะได้ fifth Normal Form (5NF)

ถึงแม้วิธีนี้เป็นวิธีที่มีขั้นตอนเป็นระบบ แต่ก็เป็วิธีที่ค่อนข้างยุ่งยากและซับซ้อน โดยเฉพาะอย่างยิ่งเมื่อมีระบบงานขนาดใหญ่มาก แอททริบิวต์มีจำนวนมาก การออกแบบด้วยวิธีนอร์มัลไลเซชันจึงเป็นเรื่องลำบาก

การออกแบบระบบฐานข้อมูลรวมแบบรีเลชันแนล (Integrated Relational Database System) โดยใช้ในแอม (NIAM : Nijssen's Information System Analysis Method) เป็นเครื่องมือในการออกแบบเนื่องจากแนวคิดที่ให้ Conceptual Schema มีพื้นฐานมาจากโครงสร้างภาษารวมชาติ ใช้รูปประโยคที่มีประธาน กริยา กรรม วิธีแสดงรูปแบบความสัมพันธ์ระหว่างข้อมูล เป็นแบบจำลองที่มีความหมาย และมีเครื่องหมายแสดงความสัมพันธ์ของข้อมูลและข้อจำกัดของข้อมูลได้อย่างชัดเจน นอกจากนั้นยังสามารถแปลง Conceptual Schema เป็น Relational Database Schema ซึ่งจะอยู่ในรูปของ Fifth Normal Form และเนื่องจากวิธีการนี้ใช้รูปสัญลักษณ์ที่แสดงความสัมพันธ์ของข้อมูลและง่ายต่อการเข้าใจ ดังนั้นจึงสะดวกในการออกแบบฐานข้อมูลของระบบงานใหญ่ ๆ

เนื่องจากเป็นที่ยอมรับกัน โดยทั่วไปว่า ระบบฐานข้อมูลรีเลชันแนลเป็นเครื่องมือที่เหมาะสมในการช่วยพัฒนาระบบสารสนเทศ (Information System) โดยเฉพาะอย่างยิ่งระบบสารสนเทศที่มีจุดประสงค์ ในการสนับสนุนการจัดการ (Management Information System : MIS) ดังนั้นการออกแบบฐานข้อมูลรวม สำหรับองค์กรจึงเป็นสิ่งสำคัญอย่างยิ่ง เพราะฐานข้อมูลที่ได้จะเป็น โครงสร้างข้อมูลหลัก สำหรับรองรับระบบสารสนเทศดังกล่าว

## 2.4 ระบบสารสนเทศเพื่อการจัดการ

ในปัจจุบันฐานข้อมูลส่วนใหญ่ถูกออกแบบมาเพียงเพื่อช่วยงานในระดับปฏิบัติการให้สำเร็จเป็นระบบงานอิสระเท่านั้น การที่ฐานข้อมูลจะสามารถรองรับงานในระดับบริหารได้นั้นต้องมีองค์ประกอบหลายด้าน เนื่องจากผู้บริหารใช้ข้อมูลสารสนเทศในการบริหาร ทั้งแบบที่เป็นรายงานประจำที่ที่ต้องรับทราบ และข้อมูลที่ต้องการในขณะใดขณะหนึ่ง โดยไม่ได้เตรียมรายงานไว้ก่อนเพื่อใช้ในการตัดสินใจได้ทันเหตุการณ์ การจะตอบสนองความต้องการด้านข้อมูลแก่ผู้บริหาร ได้อย่างรวดเร็วและถูกต้อง ระบบฐานข้อมูลดังกล่าวจะต้องมีคุณสมบัติต่อไปนี้

### 2.4.1 ข้อมูลมีความซ้ำซ้อนน้อยที่สุด

กล่าวคือข้อมูลต้องไม่มีการเก็บซ้ำซ้อน ซึ่งอาจทำให้เกิดความขัดแย้งของข้อมูล อันเนื่องมาจากการเปลี่ยนแปลงข้อมูลเฉพาะที่ คุณสมบัติข้อนี้ต้องอาศัยการออกแบบระบบฐานข้อมูลที่ดี ดังนั้นการเลือกใช้วิธีในแอมในการออกแบบฐานข้อมูล ซึ่งเป็นประโยชน์อย่างมากเพราะเมื่อแปลงรูปแบบจำลองเป็นรีเลชันแล้วจะอยู่ในรูป 5NF (fifth Normal Form) ในทันที

### 2.2.4 มีการจัดระบบคลังข้อมูล (Inventory of data)

ผู้รับผิดชอบฐานข้อมูลต้องทราบว่าข้อมูลใดถูกเก็บไว้ในตารางชื่ออะไร เพื่อที่ว่าเมื่อมีคำถามจากผู้บริหารจะได้หาข้อมูลเพื่อตอบคำถามได้ในทันทีด้วยภาษาฐานข้อมูล เมื่อคำนึงถึงวัตถุประสงค์นี้ร่วมกับเรื่องของการลดความซ้ำซ้อน ตารางข้อมูลที่ถูกสร้างขึ้นจะต้องได้รับการออกแบบอย่างดีและการสร้างตารางจะทำได้โดยกลุ่มของผู้รับผิดชอบฐานข้อมูลเท่านั้น ผู้ใช้ในระดับปฏิบัติการไม่ควรมีสิทธิ์ในการสร้างตารางข้อมูลเอง และเนื่องจากข้อมูลที่เข้ามาในระบบฐานข้อมูลจะถูกนำเข้าโดยระบบงานระดับปฏิบัติการ ตามหน่วยงานย่อยขององค์กร ซึ่งในแต่ละหน่วยงานจะมีสิทธิ์ในการจัดการข้อมูลในระดับใดก็ได้ ผ่านใครเป็นผู้นำข้อมูลเข้า ใครมีสิทธิ์แก้ไขข้อมูล และใครมีสิทธิ์เพียงเรียกใช้ข้อมูล เพื่อที่จะได้ให้สิทธิ์ที่ถูกต้องบนตารางหรือส่วนของตารางที่สมควรให้ใช้ หลักการนี้เรียกว่า หลักการระบบคลังข้อมูล

ในระบบจัดการฐานข้อมูลปัจจุบัน ส่วนมากจะมีพจนานุกรมข้อมูล (Data Dictionary) หรือที่บางระบบเรียกว่า System Catalog ที่ประกอบไปด้วยตารางซึ่งเก็บรายละเอียดของสิ่งที่มีความสำคัญต่อฐานข้อมูล เช่น รายละเอียดเกี่ยวกับผู้ใช้ตาราง วิว อินเด็กซ์ เป็นต้น การแก้ไขพจนานุกรมข้อมูลนี้ระบบจัดการฐานข้อมูล จะจัดการเองอัตโนมัติเมื่อมีการเปลี่ยนแปลงใด ๆ ของตารางและผู้ใช้ขึ้น นับว่าเป็นเครื่องมือในการบริหารคลังข้อมูล ที่มีประโยชน์มาก

### 2.4.3 สามารถตอบคำถามเฉพาะกิจ

ในปัจจุบันระบบจัดการฐานข้อมูลจะใช้ภาษาฐานข้อมูล ภาษาฐานข้อมูลดังกล่าวเช่น SQL, QUEL ซึ่งเป็นภาษาที่สามารถเข้าใจได้ง่าย ใช้เขียนโปรแกรมในการเรียกดูข้อมูลในตารางหนึ่ง หรือหลายตารางได้โดยสะดวกรวดเร็ว ผู้ใช้ที่เชี่ยวชาญสามารถใช้ภาษาฐานข้อมูลตอบคำถามที่เกิดขึ้นในขณะใดขณะหนึ่ง อันเกี่ยวข้องกับข้อมูลในฐานข้อมูลได้อย่างทันที โดยไม่จำเป็นต้องเขียนโปรแกรมในภาษาอื่น เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เช่น โคบอล ซี หรือ ปาสคาล ซึ่งเสียเวลานานกว่ามากจนอาจไม่ทันต่อความต้องการใช้ข้อมูลเพื่อการตัดสินใจ นอกจากนี้ยังช่วยประหยัดงบประมาณ และเวลาในการจัดทำโปรแกรมในภาษาอื่นดังกล่าว เพื่อรองรับคำถามที่เป็นไปได้สำหรับทุกคำถามไว้ล่วงหน้า

#### 2.4.4 ความปลอดภัยของข้อมูลสูง

ข้อนี้มีความสำคัญมาก เพราะหากทุกคนสามารถเรียกดูและและเปลี่ยนแปลงข้อมูลในฐานข้อมูลทั้งหมดได้ อาจก่อให้เกิดความเสียหายต่อข้อมูลอันเป็นผลให้ข้อมูลที่มี เป็นข้อมูลที่ผิดพลาดไม่มีประโยชน์ต่อการดำเนินงานใด ๆ ข้อมูลบางส่วนอาจเป็นข้อมูลที่ไม่อาจเปิดเผยได้หรือเป็นข้อมูลเฉพาะของผู้บริหาร หากไม่มีการจัดการด้านความปลอดภัยของข้อมูล ฐานข้อมูลก็จะไม่สามารถใช้เก็บข้อมูลบางส่วน ดังนั้นในระบบจัดการฐานข้อมูลส่วนใหญ่นอกจากจะมีการรักษาความปลอดภัยของข้อมูล โดยใช้รหัสผู้ใช้และรหัสผ่านแล้ว ยังมีความสามารถในการแบ่งระดับผู้ใช้ฐานข้อมูลออกเป็นหลายระดับ เช่น

- DBA ผู้ที่สามารถสร้างและจัดการตารางและข้อมูลทั้งหมดในฐานข้อมูลรวมทั้งการเพิ่มผู้ใช้หรือระงับการใช้งานของผู้ใช้
- RESOURCE ผู้ที่สามารถสร้างและจัดการตาราง และข้อมูลทั้งหมดในตารางของตน และสามารถอนุญาตให้ผู้อื่นใช้ตารางของตนในทางใด ๆ
- CONNECT ผู้ที่สามารถเรียกดู เพิ่มเติม ลบ และแก้ไขข้อมูล หรือบางส่วนของความสามารถข้างต้น ในตารางที่ได้รับการอนุญาตให้ใช้

นอกจากนั้นผู้ออกแบบฐานข้อมูลสามารถใช้วิว (VIEW) เพื่อประโยชน์ในการรักษาความปลอดภัยของข้อมูลได้เป็นอย่างดี โดยการสร้างวิวที่ทำตัวเสมือนเป็นตารางของผู้ใช้จริง ๆ และข้อมูลที่ปรากฏในวิวจะเป็นข้อมูลที่เกี่ยวข้องกับงานของผู้ใช้เท่านั้น

#### 2.4.5 สามารถจัดการการไข ข ข้อมูลในเวลาเดียวกัน (ConcerrencyControl)

ระบบจัดการฐานข้อมูลของระบบงานขนาดใหญ่มักจะเป็นระบบแบบผู้ใช้หลายคน (Multiuser) ซึ่งบางหน่วยงานที่มีการใช้ข้อมูลร่วมกัน บางครั้งอาจมีความต้องการเข้าถึงข้อมูลในตารางเดียวกันหรือข้อมูลเดียวกันพร้อม ๆ กัน

ดังนั้นระบบจัดการฐานข้อมูล จำเป็นต้องมีความสามารถที่จะให้ผู้ใช้หลายคนเข้าถึงข้อมูลเดียวกันได้อย่างถูกต้อง ระบบที่ใช้สำหรับงานด้านนี้คือ Concerrency Control Mechanism โดยจะทำการ Lock ข้อมูลในระดับของคอลัมน์ แถว ตาราง หรือ อาจจะทั้งฐานข้อมูล แล้วแต่ความสามารถของระบบจัดการฐานข้อมูล เพื่อจุดประสงค์ต่าง ๆ กัน โดยจะจัดลำดับของงานตามเวลาหรือความสำคัญของงาน มีหลักการ ดังนี้

- Share Lock (read lock) คือขณะที่ผู้ใช้คนหนึ่งกำลังอ่านข้อมูลอยู่ ผู้อื่นก็จะสามารถอ่านข้อมูลเดียวกันได้ แต่ไม่สามารถเปลี่ยนแปลงข้อมูล หรือจัดการรูปแบบตารางที่กำลังอ่านอยู่ได้

- Exclusive Lock (write lock) ขณะที่ผู้ใช้หนึ่งกำลังทำการเปลี่ยนแปลงข้อมูลหรือจัดการรูปแบบของตาราง ผู้อื่นไม่สามารถใช้ตารางหรือคอลัมน์หรือบางแถวที่กำลังถูกเปลี่ยนแปลงนั้นได้

กิจกรรมทั้งหมดนี้ระบบจัดการฐานข้อมูลเป็นผู้จัดการให้โดยผู้ใช้ไม่ต้องรับรู้ เป็นการแบ่งเบาภาระในการพัฒนาระบบงานเป็นอย่างมาก ถ้าไม่ใช่ระบบจัดการฐานข้อมูล (ใช้ระบบไฟล์ธรรมดา) ผู้พัฒนาโปรแกรมจะต้องจัดการสิ่งเหล่านี้เองทั้งหมด

## 2.5 การออกแบบฐานข้อมูลในรูปบรรทัดฐาน

การออกแบบฐานข้อมูลในรูปบรรทัดฐาน เป็นการออกแบบตารางฐานข้อมูลเพื่อใช้งานกับระบบจัดการฐานข้อมูลแบบรีเลชัน (ฐานข้อมูลที่ไม่อยู่ในรูปบรรทัดฐานนั้นไม่สามารถที่จะนำไปใช้ได้กับระบบจัดการฐานข้อมูลแบบรีเลชัน) โดยมีจุดประสงค์ในการออกแบบก็เพื่อที่จะให้ได้ตารางฐานข้อมูลที่ไม่มีความซ้ำซ้อนเกิดขึ้น ซึ่งฐานข้อมูลใดก็ตามที่ยังมีความซ้ำซ้อนอยู่อาจเป็นสาเหตุทำให้เกิดการประมวลผลข้อมูลที่ไม่ตรงกับความเป็นจริง (Database Inconsistency And Modification Anomalies) ในหัวข้อนี้จะกล่าวถึงข้อเสียของตารางฐานข้อมูลที่ซ้ำซ้อนและแนวทางแก้ไข โดยใช้วิธีการทำให้ตารางฐานข้อมูลอยู่ในรูปบรรทัดฐานขั้นสูงตั้งแต่ขั้นแรก (First Normalization) จนถึงขั้นที่ห้า (Fifth Normalization) ซึ่งถือว่าเป็นขั้นสูงสุดที่สามารถกำจัดความซ้ำซ้อนของข้อมูลได้อย่างสิ้นเชิง

### 2.5.1 ฐานข้อมูลที่อยู่ในรูปบรรทัดฐานขั้นแรก

ตารางข้อมูลที่สามารถใช้ได้กับระบบจัดการฐานข้อมูลแบบรีเลชันนั้นจะต้องมีคุณสมบัติอย่างน้อยที่สุดเป็นตารางฐานข้อมูลที่อยู่ในรูปบรรทัดฐานขั้นแรก ซึ่งต้องมีคุณสมบัติดังต่อไปนี้

- ไม่มีทัพเพิลที่ซ้ำกัน คุณสมบัตินี้นิยามจากความจริงที่ว่าข้อมูลที่อยู่ในรีเลชันก็คือเซตของทัพเพิล ซึ่งในทางคณิตศาสตร์แล้วสมาชิกในเซตต้องไม่ซ้ำกัน เหตุผลที่ตามมาก็คือว่า ทุกรีเลชันจะต้องมีกุญแจหลักเพื่อแยกความแตกต่างระหว่างทัพเพิล
- ลำดับของทัพเพิล ไม่มีความสำคัญ คุณสมบัตินี้นิยามจากความจริงที่ว่าข้อมูลที่อยู่ในรีเลชันก็คือเซตของทัพเพิล ซึ่งในทางคณิตศาสตร์แล้วลำดับของสมาชิกในเซตไม่มีความสำคัญ
- ลำดับของแอททริบิวต์ ไม่มีความสำคัญ คุณสมบัตินี้นิยามจากความจริงที่ว่าข้อมูลประกอบขึ้นจากเซตของแอททริบิวต์ ซึ่งในทางคณิตศาสตร์แล้วลำดับของสมาชิกในเซตไม่มีความสำคัญ
- เมื่อกำหนดแถวและหลักของตารางจะต้องได้ค่าของข้อมูลเพียงค่าเดียวเท่านั้น

S#	P#	QTY
S1	P1	300
S1	P4	200
S1	P3	400
S1	P4	200
S1	P5	100
S1	P6	100
S2	P1	300
S2	P2	400
S3	P2	200
S4	P2	200
S4	P4	300
S4	P5	400

S#	P		Q
	P#	QTY	
S1	P1	300	
	P2	200	
	P3	400	
	P4	200	
	P5	100	
	P6	100	
S2	P1	300	
	P2	400	
S3	P2	200	
	P2	200	
S4	P4	300	
	P5	400	

(a) Duplicate tuples

(b) Repeating groups

รูปที่ 2.2 แสดงฐานข้อมูลที่ไม่อยู่ในรูปบรรทัดฐานหรือเรียกอีกอย่างหนึ่งว่าไม่เป็นรีเลชัน

S#	P#	QTY
S1	P1	300
S1	P2	200
S1	P3	400
S1	P4	200
S1	P5	100
S1	P6	100
S2	P1	300
S2	P2	400
S3	P2	200
S4	P2	200
S4	P4	300
S4	P5	400

รูปที่ 2.3 แสดงตารางฐานข้อมูลที่อยู่ในรูปบรรทัดฐานขั้นแรก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.5.2 ฐานข้อมูลที่อยู่ในรูปบรรทัดฐานขั้นที่สอง

### PROJECT-DATA

PERSON_ID	PROJ#	PROJ_BUDGET	TOTAL_TIME_SPENT_BY_PERSON ON PROJECT
P1	PROJ1	20	20
P3	PROJ1	20	16
P2	PROJ2	17	35
P2	PROJ3	84	42
P3	PROJ2	17	17
P2	PROJ1	20	83
P4	PROJ3	84	41
-	PROJ4	90	-

รูปที่ 2.4 แสดงตารางฐานข้อมูลที่ไม่อยู่ในรูปบรรทัดฐานขั้นที่สอง

ฐานข้อมูลที่อยู่ในรูปบรรทัดฐานขั้นที่สองต้องมีคุณสมบัติดังต่อไปนี้

— ต้องเป็นฐานข้อมูลที่อยู่ในรูปบรรทัดฐานขั้นแรก

ทุกแอททริบิวต์ที่ไม่ใช่คีย์หลักของตารางต้องขึ้นอยู่กับคีย์หลักเท่านั้นจะขึ้นอยู่กับสับเซตของคีย์หลักไม่ได้ หรือกล่าวอีกนัยหนึ่งว่าทุกแอททริบิวต์ที่ไม่ใช่คีย์หลักของตารางต้อง Fully Dependence คีย์หลัก

จากตารางฐานข้อมูลในรูปที่ 4 เป็นตารางฐานข้อมูลที่อยู่ในรูปบรรทัดฐานขั้นแรกที่มีคีย์หลักของตารางคือแอททริบิวต์ PERSON\_ID รวมกับแอททริบิวต์ PROJ# ตารางฐานข้อมูลนี้ยังไม่เป็นตารางฐานข้อมูลในรูปบรรทัดฐานขั้นที่สอง เนื่องจากแอททริบิวต์ PROJECT\_BUDGET สามารถขึ้นอยู่กับแอททริบิวต์ที่เป็นสับเซตของคีย์หลักคือ PROJ# เท่านั้น ทำให้เกิดความซ้ำซ้อนของตารางฐานข้อมูลขึ้น ได้แก่ เมื่อใดก็ตามที่เรากำหนด PROJ# จะได้ผลลัพธ์ของ PROJECT\_BUDGET ซึ่งเป็นค่าเดียวกันจำนวนหลายแถว ข้อเสียคือ ถ้าต้องการแก้ไขข้อมูลของ PROJ# จะต้องแก้ไขข้อมูลของ PROJ\_BUDGET ด้วย ค่าที่สอดคล้องกันจะแก้ไขเพียงแอททริบิวต์ใดแอททริบิวต์หนึ่งไม่ได้ จากเหตุผลดังกล่าวทำให้ข้อมูลที่ถูกเก็บลงในตารางฐานข้อมูลนี้อาจไม่ตรงกับความจริงได้ถ้าเมื่อใดก็ตามที่มีการแก้ไข PROJ# หรือ PROJECT\_BUDGET เพียงแอททริบิวต์เดียว วิธีการกำจัดความซ้ำซ้อนของตารางฐานข้อมูลนี้เราต้องแยกตารางฐานข้อมูลนี้ออกเป็นสองตารางดังในรูปที่ 5 ซึ่งตารางฐานข้อมูลทั้งสองที่เกิดขึ้นใหม่นี้จัดอยู่ในตารางฐานข้อมูลในรูปบรรทัดฐานขั้นที่สอง

PROJECT		WORK		
PROJ#	PROJ_BUDGET	PERSON_ID	PROJ#	TOTAL_TIME_SPENT BY PERSON ON PROJECT
PROJ1	20	P1	PROJ1	20
PROJ2	17	P3	PROJ1	16
PROJ3	84	P2	PROJ2	35
PROJ4	90	P2	PROJ3	42
		P3	PROJ2	17
		P3	PROJ1	83
		P4	PROJ3	41

รูปที่ 2.5 แสดงตารางฐานข้อมูลที่อยู่ในรูปบรรทัดฐานขั้นที่สอง

### 2.5.3 ฐานข้อมูลที่อยู่ในรูปบรรทัดฐานขั้นที่สาม

ฐานข้อมูลที่อยู่ในรูปบรรทัดฐานขั้นที่สามต้องมีคุณสมบัติดังต่อไปนี้

- ต้องเป็นฐานข้อมูลที่อยู่ในรูปบรรทัดฐานขั้นที่สอง
- ทุกแอททริบิวต์ที่ไม่ใช่คีย์หลักของตารางต้องไม่ขึ้นอยู่กับแอททริบิวต์ที่ไม่ใช่คีย์หลักของตารางด้วยตัวเอง

PROJECTS		
PROJ#	MANAGER	DATE_OF_BIRTH
P1	Joe	Jan 63
P3	Vicki	March57
P2	Joe	Jan 63
P4	Marilyn	July 53

รูปที่ 2.6 แสดงตารางฐานข้อมูลที่อยู่ในรูปบรรทัดฐานขั้นที่สาม

จากตารางข้อมูลในรูปที่ 6 เป็นตารางฐานข้อมูลที่อยู่ในรูปบรรทัดฐานขั้นที่สอง เนื่องจากคีย์หลักของตารางประกอบด้วยแอททริบิวต์เพียงแอททริบิวต์เดียว คือ PROJ# ตารางฐานข้อมูลนี้ยังไม่มีเป็นตารางฐานข้อมูลที่อยู่ในรูปบรรทัดฐานขั้นที่สาม เนื่องจากแอททริบิวต์ที่ไม่ได้เป็นคีย์หลักของตารางคือ DATE\_OF\_BIRTH สามารถขึ้นอยู่กับแอททริบิวต์ MANAGER ซึ่งเป็นแอททริบิวต์ที่ไม่เป็นคีย์หลักของตารางเช่นเดียวกัน ทำให้เกิดความซ้ำซ้อนของตารางฐานข้อมูลขึ้น ได้แก่ เมื่อไรก็ตามที่เรากำหนด MANAGER จะได้ผลลัพธ์ของ DATE\_OF\_BIRTH ด้วยค่าที่สอดคล้องกันจะแก้ไขเพียงแอททริบิวต์ใดแอททริบิวต์หนึ่งไม่ได้ จากเหตุผลดังกล่าวทำให้ข้อมูลที่เก็บลงในตารางฐานข้อมูลอาจไม่ตรงกับความจริงได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าเมื่อไรก็ตามที่มีการแก้ไข MANAGER หรือ DATE\_OF\_BIRTH เพียงแอททริบิวเดียว วิธีการกำจัดความซ้ำซ้อนของตารางฐานข้อมูลนี้เราต้องแยกตารางฐานข้อมูลนี้ออกเป็นสองตารางดังรูปที่ 7 ซึ่งตารางฐานข้อมูลทั้งสองที่เกิดขึ้นใหม่นี้จัดอยู่ในตารางฐานข้อมูลในรูปแบบรทฐานขั้นที่สาม

PROJECTS		MANAGER	
PROJ#	MANAGER	MANAGER	DATE_OF_BIRTH
P1	Joe	Joe	Jan 63
P2	Joe	Vicki	March 57
P3	Vicki	Marilyn	Jan 63
P4	Marilyn		

รูปที่ 2.7 แสดงตารางฐานข้อมูลที่อยู่ในรูปแบบรทฐานขั้นที่สาม

2.5.4 ฐานข อมูลที่อยู่ในรูปแบบรทฐานขั้น Boyce/Codd

ฐานข้อมูลที่อยู่ในรูปแบบรทฐานขั้น Boyce/Codd ต้องมีคุณสมบัติดังต่อไปนี้

- ต้องเป็นฐานข้อมูลที่อยู่ในรูปแบบรทฐานขั้นที่สาม
- ทุกแอททริบิวที่เป็นคีย์หลักของตารางจะต้องไม่ขึ้นอยู่กับสับเซตของคีย์หลักอันอื่น หรือกล่าวอีกนัยหนึ่งว่า ทุก Determinant ต้องเป็น Candidate Key

ATTENDANCE				
TEACHER	SEMESTER	SUBJECT	SECTION	ATTENDANCES
Joe	1/88	COBOL	SECTION1	35
Jenny	1/88	MATHS	SECTION1	40
Gregory	2/88	UNIX	SECTION2	33
Jenny	1/88	MATHS	SECTION2	42
Gregopy	2/88	UNIX	SECTION1	47
Joe	1/88	COBOL	SECTION2	50
Joe	1/88	COBOL	SECTION2	12

รูปที่ 2.8 แสดงตารางฐานข้อมูลที่ไม่อยู่ในรูปแบบรทฐานขั้น Boyce/Codd

จากตารางฐานข้อมูลในรูปที่ 8 เป็นตารางฐานข้อมูลที่ไม่อยู่ในรูปแบบรทฐานขั้นที่สาม เนื่องจากจากแอททริบิวที่ไม่เป็นคีย์หลักของตารางมีเพียงแอททริบิวเดียวคือ ATTENDANCES (ไม่มีแอททริบิวที่ไม่ใช่คีย์หลักของตารางตัวอื่นมาขึ้นกับแอททริบิว ATTENDANCES ) โดยมีคีย์หลักของตารางจำนวนสองตัวได้แก่ กลุ่มแอททริบิว {TEACHER, SEMESTER, SECTION} กับ {SUBJECT, SEMESTER, SECTION} ตารางฐานข้อมูลนี้ยังไม่เป็นตารางฐานข้อมูลในรูปแบบรทฐานขั้น BOYCE/CODD เนื่องจากเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น เมื่ออนุญาตให้นำไปเผยแพร่โดยไม่ผ่านการคัดค้านไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แอททริบิวต์ที่เป็นส่วนหนึ่งในคีย์หลักของตารางตัวหนึ่งคือ TEACHER ขึ้นอยู่กับสับเซตของคีย์หลักตัวที่สองคือ แอททริบิวต์ SEMESTER ร่วมกับ SUBJECT ทำให้เกิดความซ้ำซ้อนของตารางฐานข้อมูลขึ้นได้แก่ เมื่อไรก็ตามที่เรากำหนด SEMESTER กับ SUBJECT จะได้ผลลัพธ์ของ TEACHER ซึ่งเป็นค่าเดียวกันจำนวนหลายแถว ข้อเสียคือ ถ้าต้องการแก้ไขข้อมูลของ SEMESTER กับ SUBJECT จะต้องแก้ไขข้อมูลของ TEACHER ด้วย ค่าที่สอดคล้องกันจะแก้ไขเพียงแอททริบิวต์ใดแอททริบิวต์หนึ่งไม่ได้ จากเหตุผลดังกล่าวทำให้ข้อมูลที่ถูกเก็บลงในตารางฐานข้อมูลนี้อาจไม่ตรงกับความจริงได้ ถ้าเมื่อไรก็ตามที่มีการแก้ไข SEMESTER กับ SUBJECT หรือ TEACHER เพียงแอททริบิวต์เดียว วิธีการกำจัดความซ้ำซ้อนของตารางฐานข้อมูลนี้ เราต้องแยกตารางฐานข้อมูลออกเป็นสองตาราง ดังในรูปที่ 9 ซึ่งตารางฐานข้อมูลทั้งสองที่เกิดขึ้นใหม่นี้จัดอยู่ในตารางฐานข้อมูลในรูปแบบรทฐานขั้น Boyce/Codd

TEACHER	SEMESTER	SUBJECT
Joe	1/88	COBOL
Jenny	1/88	MATHS
Gregory	2/88	UNIX

SEMESTER	SUBJECT	SECTION	ATTENDANCES
1/88	COBOL	SECTION1	35
1/88	MATHS	SECTION1	40
2/88	UNIX	SECTION2	35
1/88	MATHS	SECTION2	33
2/88	UNIX	SECTION1	47
1/88	COBOL	SECTION2	50
1/88	COBOL	SECTION3	12

รูปที่ 2.9 แสดงตารางฐานข้อมูลในรูปแบบรทฐานขั้น Boyce/Codd

### 2.5.5 ฐานข้อมูลในรูปแบบรทฐานขั้นที่สี่

ฐานข้อมูลในรูปแบบรทฐานขั้นที่สี่ มีคุณสมบัติดังต่อไปนี้

- ต้องเป็นฐานข้อมูลในรูปแบบรทฐานขั้น Boyce/Codd
- ต้องไม่มี multivalue dependency (MVD) มากเกินหนึ่งตัว

← CTX →

COURSE	TEACHER	TEXT
Physics	Prof. Green	Basic Mechanics
Physics	Prof. Green	Principle of Optics
Physics	Prof. Brown	Basic Mechanics
Physics	Prof. Brown	Principle of Optics
Maths	Prof. Green	Basic Mechanics
Maths	Prof. Green	Vector Analysis
Maths	Prof. Green	Trigonometry

รูปที่ 2.10 แสดงตารางฐานข้อมูลที่ไม่อยู่ในรูปบรรทัดฐานขั้นที่สี่

จากตารางฐานข้อมูลในรูปที่ 10 เป็นตารางฐานข้อมูลที่อยู่ในรูปบรรทัดฐานขั้น Boyce/Codd เนื่องจากมีคีย์หลักเพียงตัวเดียว ตารางฐานข้อมูลนี้ยังไม่เป็นตารางฐานข้อมูลในรูปบรรทัดฐานขั้นที่สี่ เนื่องจากมี มากกว่าหนึ่งตัว ได้แก่ เมื่อเรากำหนด จะได้เซตของ และเซตของ ข้อเสียของความซ้ำซ้อนที่เกิดขึ้นได้แก่ เมื่อต้องการเพิ่มข้อมูลเพียงว่า วิชา สามารถสอนด้วย เราจำเป็นอย่างยั้งที่จะต้อง ข้อมูลนี้ถึง 3 ทัพเพิล เนื่องจาก ที่ใช้ในการสอนวิชา มีด้วยกันสามเล่ม ดังนั้นจะเห็นได้ว่าเราไม่สามารถจะ ข้อมูลที่ละ 1 ทัพเพิลได้ถ้ามี เกิดขึ้นมากกว่าหนึ่งตัวในตารางเดียวกัน วิธีการกำจัดความซ้ำซ้อนของตารางข้อมูลนี้เราต้องแยกตารางฐานข้อมูลนี้ออกเป็นสองตารางดังในรูปที่ 11 ซึ่งตารางฐานข้อมูลทั้งสองที่เกิดขึ้นใหม่นี้จัดอยู่ในตารางฐานข้อมูลในรูปบรรทัดฐานขั้นที่สี่

← CT →

COURSE	TEACHER
Physics	Prof. Green
Physics	Prof. Brown
Math	Prof. Green

← CX →

COURSE	TEXT
Physics	Basic Mechanics
Physics	Principles of Optics
Math	Basic Mechanics
Math	Vector Analysis
Math	Trigonometry

รูปที่ 2.11 แสดงตารางฐานข้อมูลที่อยู่ในรูปบรรทัดฐานขั้นที่สี่

2.5.6 ฐานข้อมูลที่อยู่ในรูปบรรทัดฐานขั้นที่ห้า

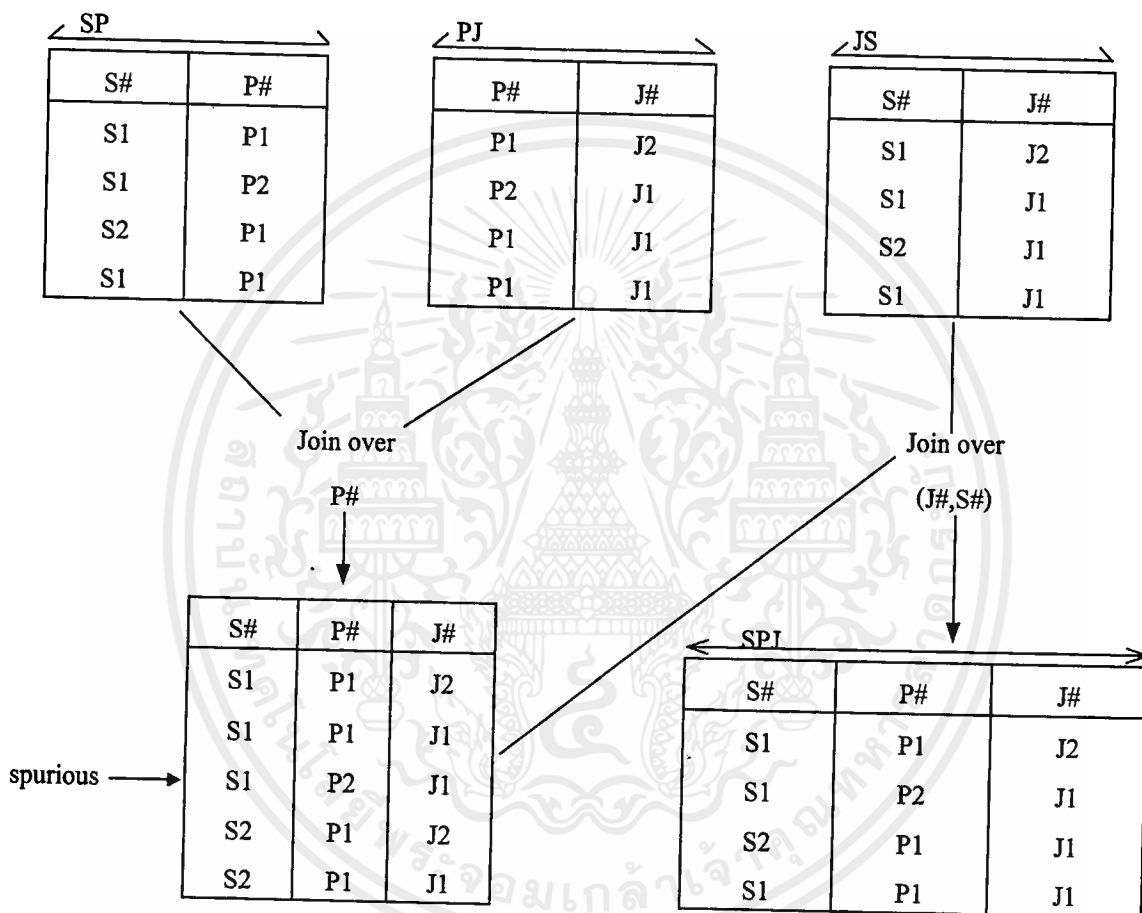
ฐานข้อมูลที่อยู่ในรูปบรรทัดฐานขั้นที่ห้า มีคุณสมบัติดังต่อไปนี้

- ต้องเป็นฐานข้อมูลที่อยู่ในรูปบรรทัดฐานขั้นที่สี่
- ผลลัพธ์ของการ Join ของ JD (Join Dependency) ใน R ต้องเป็น Lossless Join

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

S#	P#	J#
S1	P1	J2
S1	P2	J1
S2	P1	J1
S1	P1	J1

รูปที่ 2.12 แสดงตารางฐานข้อมูลที่ไม่อยู่ในรูปบรรทัดฐานขั้นที่ห้า



รูปที่ 2.13 แสดงตารางฐานข้อมูลที่อยู่ในรูปบรรทัดฐานขั้นที่ห้า

จากตารางฐานข้อมูลในรูปที่ 12 แสดงตารางฐานข้อมูลที่อยู่ในรูปบรรทัดฐานขั้นที่สี่ เนื่องจากทุกแอททริบิวต์รวมเป็นคีย์หลักของตาราง และ involves no nontrivial Fds or MVDs และจากตารางฐานข้อมูลในรูปที่ 13 เราแยกตาราง SPJ ออกเป็น 3 ตารางคือ (SP, PJ, JS) จะเห็นว่าเมื่อทำการ join ตารางทั้งสามจะทำให้ผลลัพธ์ของข้อมูลในตารางมีค่าเท่ากับตารางเดิมทุกประการ และถ้าทำการ join ตารางแค่เพียงสองตารางจะทำให้ข้อมูลของตารางที่เกิดขึ้นใหม่มีจำนวนมากขึ้นซึ่งไม่ถูกต้อง แสดงให้เห็นว่าเราไม่สามารถที่จะแยกตารางออกเป็นสองตารางได้แต่แยกเป็นสามตารางได้ ซึ่งการ join ของตารางทั้งสามสามารถอธิบายได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



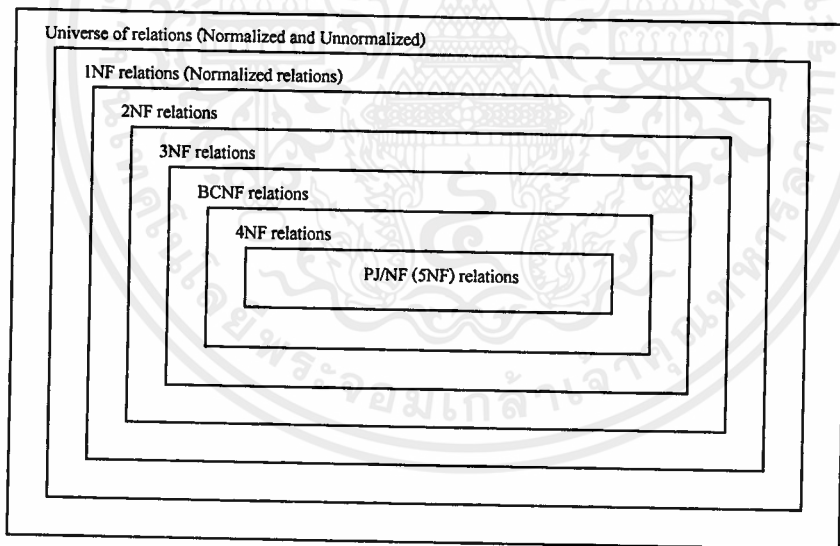
ถ้ามีข้อมูล (S1, P1) ปรากฏในตาราง SP  
 และมีข้อมูล (P1, J1) ปรากฏในตาราง PJ  
 และมีข้อมูล (J1, S1) ปรากฏในตาราง JS  
 แล้วจะต้องมีข้อมูล (S1, P1, J1) ปรากฏในตาราง SPJ

หรือกล่าวอีกนัยหนึ่งว่า

ถ้ามีข้อมูล (S1, P1, J2), (S2, P2, J2) ปรากฏในตาราง SPJ  
 แล้วจะต้องมีข้อมูล (S1, P1, J1) ปรากฏในตาราง SPJ ด้วยเช่นกัน

ดังนั้นลักษณะความซ้ำซ้อนของตาราง SPJ คือ ถ้าเราต้องการแทรกข้อมูล (S2, P1, J1) เราจะต้องแทรกข้อมูล (S1, P1, J1) ด้วยเช่นเดียวกัน แต่ในทางกลับกันไม่จำเป็น และถ้าเราต้องการลบข้อมูล (S1, P1, J1) ข้อมูลอื่นจะต้องถูกลบทิ้งด้วยอีกแถวหนึ่ง จะเห็นว่าการแทรกและลบข้อมูลของตาราง SPJ มีความยุ่งยากมากและเกิดความผิดพลาดได้ง่าย ดังนั้นถ้าเราแยกตาราง SPJ ออกเป็นสามตารางแล้ว ปัญหาต่าง ๆ ที่เกิดขึ้นจะหมดไป ซึ่งตารางฐานข้อมูลทั้งสามที่เกิดขึ้นใหม่จัดอยู่ในตารางฐานข้อมูลในรูปบรรทัดฐานขั้นที่ห้า เพราะการ join กันของตารางทั้งสามเป็น Lossless join

ที่ได้กล่าวมาทั้งหมดนี้ทำให้สามารถแสดงความสัมพันธ์ระหว่างฐานข้อมูลในรูปบรรทัดฐานขั้นต่าง ๆ ในรูปของเซตได้ดังรูปที่ 14 คือ



รูปที่ 2.14 แสดงความสัมพันธ์ของตารางฐานข้อมูลในรูปบรรทัดฐานขั้นต่าง ๆ

### 2.6 การออกแบบฐานข้อมูลโดยวิธีไนแอม (NIAM : Nijssen's Information Analysis Methodology)

ไนแอมมีขั้นตอนในการออกแบบอยู่ 9 ขั้นตอน (DSDP 9 STEP : Conceptual Schema Design Procedure)

ขั้นตอนที่ 1 : กำหนดขอบเขตของงาน (Universal of Discourse : UoD) และความจริงที่เกิดขึ้นภายในขอบเขตของงานที่กำหนดไว้

ขั้นตอนที่ 2 : วาด Conceptual Schema Diagram โดยคร่าว ๆ จากความจริงในขอบเขตงาน

ขั้นตอนที่ 3 : จัดรูปของ Schema ให้เป็นระเบียบและหาชนิดความจริงที่ได้รับ ข้อมูลมาจากชนิดความจริงอื่น

ขั้นตอนที่ 4 : เติมสัญลักษณ์แสดง Uniqueness Constraints

ขั้นตอนที่ 5 : ตรวจสอบความถูกต้องของชนิดความจริง

ขั้นตอนที่ 6 : เติมสัญลักษณ์แสดง Lexical, Mandatory Role, Subtype Constrains

ขั้นตอนที่ 7 : ตรวจสอบ Unique Identifier ของแต่ละชนิดเอนติตี้

ขั้นตอนที่ 8 : เติมสัญลักษณ์แสดง Equality, Exclusion, Subset Constraints

ขั้นตอนที่ 9 : ตรวจสอบความสมบูรณ์ของ Conceptual Schema ว่าต้องสอดคล้องกับตัวอย่างข้อมูล และ ไม่มีความซ้ำซ้อนของข้อมูล

### 2.6.1 ส่วนประกอบพื้นฐานของโนแอม

1. ชนิดเอนติตี้ (Entity Type)
2. ชนิดเลเบล (Label Type)
3. ชนิดความจริง (Fact Type)
4. ชนิดอ้างอิง (Reference Type)
5. ข้อจำกัดเพื่อความถูกต้องของข้อมูล (Integrity Constraints)

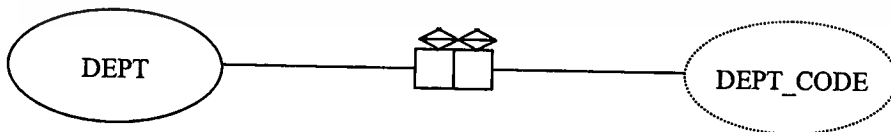
สัญลักษณ์และตัวอย่างการใช้ส่วนประกอบพื้นฐานของแบบจำลองโนแอม แสดงดังรูป



รูปที่ 2.15 แสดงสัญลักษณ์ของชนิดเอนติตี้ภาควิชา

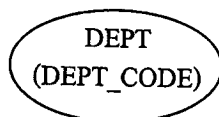


รูปที่ 2.16 แสดงสัญลักษณ์ของชนิดเลเบลรหัสภาควิชา



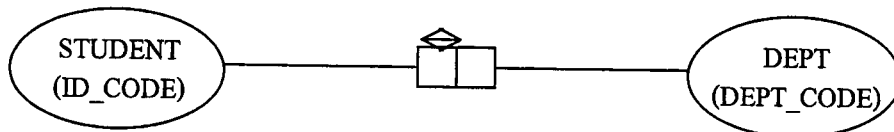
รูปที่ 2.17 แสดงความสัมพันธ์อ้างอิงแบบ one to one

(ภาควิชาใด ๆ จะมีรหัสภาควิชาเพียงรหัสเดียวเท่านั้นและไม่ซ้ำกับภาควิชาอื่น)



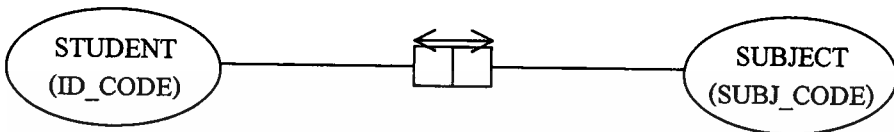
รูปที่ 2.18 แสดงการเขียนความสัมพันธ์อ้างอิงแบบ one to one อย่างย่อ

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับเจ้าของหนังสือเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.19 แสดงความจริงแบบ many to one

(นักศึกษาหนึ่งคนจะสังกัดภาควิชาได้เพียงภาคเดียวแต่ภาควิชาใด ๆ สามารถมีนักศึกษาในสังกัดได้มากกว่าหนึ่งคน)



รูปที่ 2.20 แสดงความจริงแบบ many to many

(นักศึกษาหนึ่งคนสามารถลงทะเบียนเรียนได้หลายวิชา แต่ละวิชาที่เปิดสอนสามารถรับนักศึกษาได้มากกว่าหนึ่งคน แต่นักศึกษาที่ลงทะเบียนเรียนวิชาใด ๆ แล้วจะลงทะเบียนซ้ำวิชาเดิมไม่ได้ : ไม่เป็นจริงในทางปฏิบัติ)

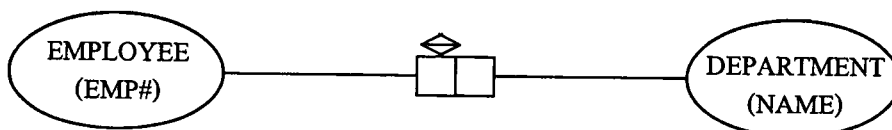
ชนิดเอนติตี้เป็นเซต (set) ซึ่งมีสมาชิกเป็นตัวอย่างเอนติตี้ (Entity instance) เช่น ภาควิชา A (ภาควิชาโทรคมนาคม) ภาควิชา B (ภาควิชาวิศวกรรมคอมพิวเตอร์) เป็นตัวอย่างเอนติตี้ของชนิดเอนติตี้ภาควิชา

เครื่องหมายความสัมพันธ์ที่เป็นส่วนเชื่อมโยงระหว่างชนิดเอนติตี้ และชนิดเอนติตี้ หรือชนิดเลเบลนั้น เรียกว่า บทบาท (role) จะเขียนความหมายของบทบาทนั้นไว้ภายในหรือข้าง ๆ สัญลักษณ์ของมัน

การแปลงข้อมูลที่วิเคราะห์มาให้อยู่ในรูปแบบจำลอง ก่อนอื่นต้องนำข้อมูลมากำหนดเป็นชนิดเอนติตี้และเลเบลให้เรียบร้อยเสียก่อน จึงนำชนิดเอนติตี้ที่ได้มาเขียนประโยค ความจริงมูลฐาน (Elementary Fact)

แล้วเอาความจริงทั้งหมดที่ได้มาเขียนเป็นแบบจำลอง และเติมข้อจำกัดต่าง ๆ ลงไปตามความเป็นจริงในขอบเขตของงาน

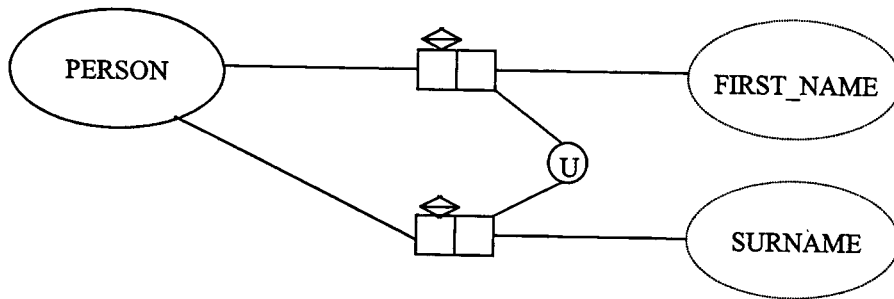
ตัวอย่างการใช้ข้อจำกัดต่าง ๆ แสดงไว้ดังนี้



รูปที่ 2.21 แสดง intra fact type uniqueness constraint

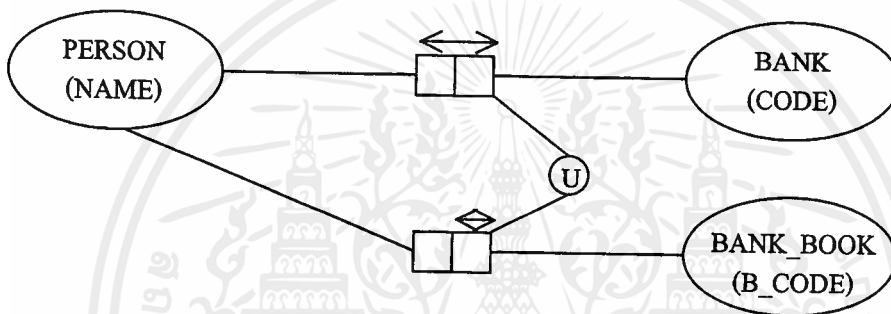
(Employee หนึ่งคนจะมีที่ทำงานได้ที่เดียวเท่านั้น)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



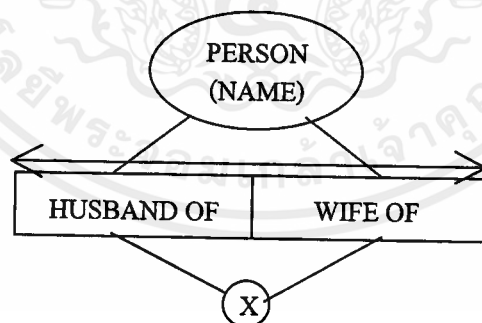
รูปที่ 2.22 แสดงการใช้ inter fact type uniqueness constraint

(บุคคลหนึ่งจะมีชื่อ 1 ชื่อ นามสกุล 1 นามสกุล ชื่อบางคนอาจจะซ้ำกันและนามสกุลบางคนอาจจะซ้ำกัน แต่ชื่อรวมกับนามสกุลจะต้องไม่ซ้ำกัน)



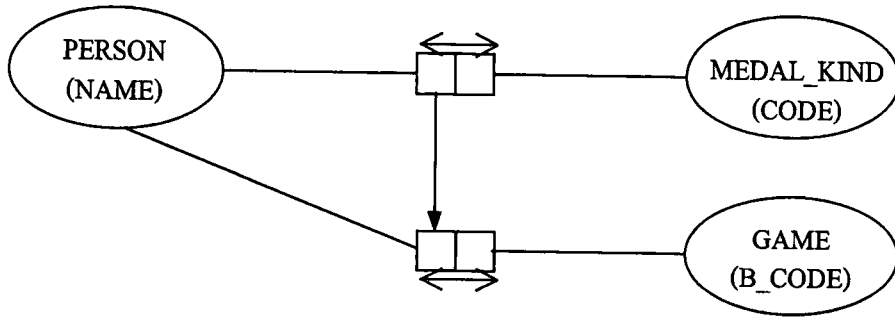
รูปที่ 2.23 แสดงการใช้ equality constraint

(ถ้าบุคคลหนึ่งเป็นลูกค้ำของธนาคารใดแล้ว บุคคลนั้นต้องมีสมุดบัญชีของธนาคารนั้นด้วย หรือในทางกลับกันถ้าบุคคลใดมีสมุดบัญชีของธนาคารใดแล้ว ก็เป็นลูกค้ำของธนาคารนั้นด้วย)



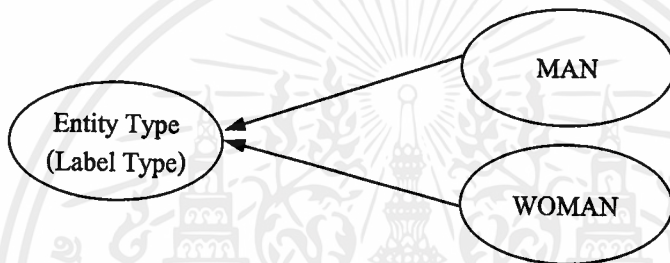
รูปที่ 2.24 แสดงการใช้ exclusion constraint

(บุคคลใดเป็นภรรยาของอีกบุคคลหนึ่งแล้ว บุคคลนั้นต้องไม่เป็นสามีของบุคคลใด ๆ ในทางกลับกันบุคคลที่เป็นสามีของอีกบุคคลหนึ่งแล้ว จะต้องไม่เป็นสามีของบุคคลใดเลย)



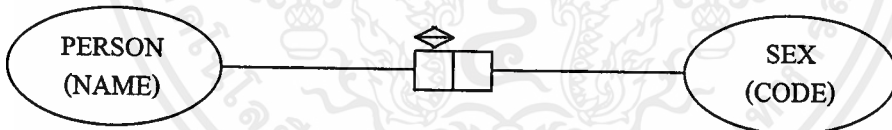
รูปที่ 2.25 แสดงการใช้ subset constraint

(บุคคลที่ชนะเลิศการแข่งขันกีฬาทุกคนจะต้องเป็นบุคคลที่เล่นกีฬาแต่บุคคลที่เล่นกีฬาไม่จำเป็นต้องชนะเลิศการแข่งขันกีฬาทุกคน)



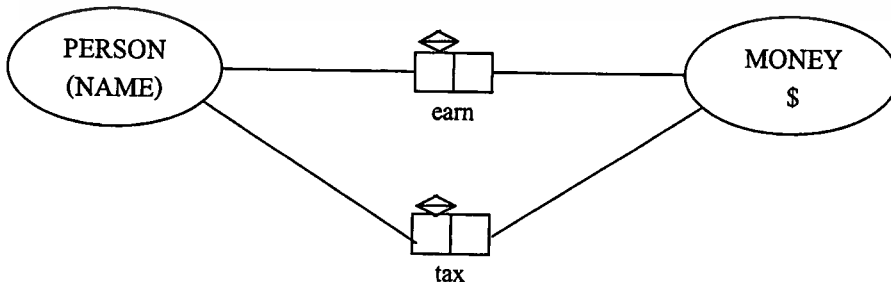
รูปที่ 2.26 แสดงการใช้ subtype constraint

(เอนิตีต์ทุกตัวของชนิดเอนิตีต์ผู้ชาย และชนิดเอนิตีต์ผู้หญิง ต่างก็เป็นสมาชิกเอนิตีต์บุคคล)



รูปที่ 2.27 แสดงการใช้ mandatory constraint, lexical constraint

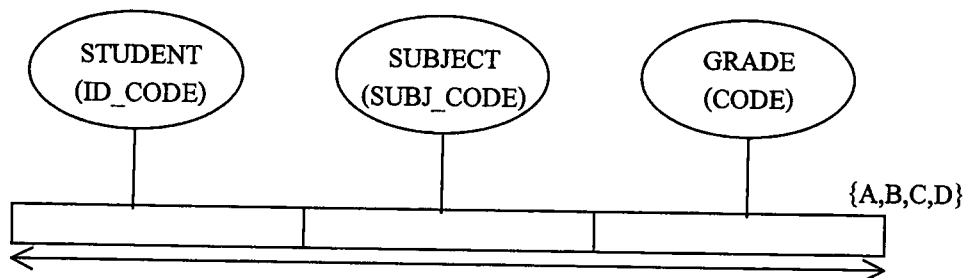
(บุคคลทุกคนต้องมีเพศและสมาชิกของเอนิตีต์เพศมีเพียง M:Male และ F:Female)



รูปที่ 2.28 หมายความว่า Employee ทุกคนต้องมีรายได้ และทุกคนต้องเสียภาษี

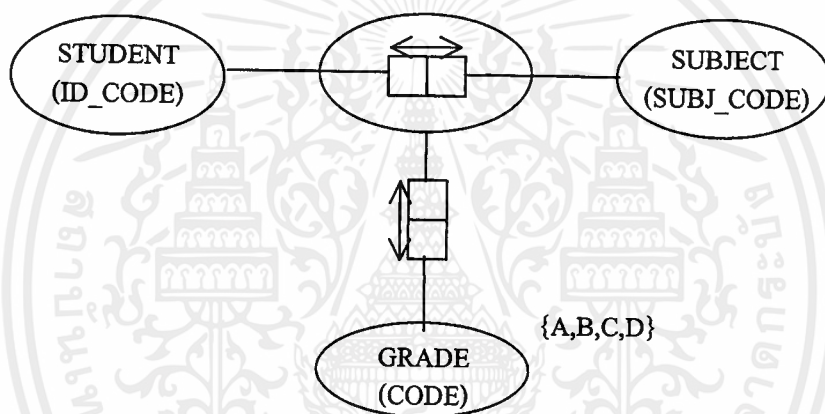
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากตัวอย่าง ความจริงแบบ Binary Fact Type ที่แสดงไว้ข้างต้นแล้วยังมีตัวอย่างชนิดอื่นอีกเช่น



รูปที่ 2.29 Ternary Fact Type

(ข้อมูลการเรียนของนักศึกษาทุกคนจะต้องมีทั้งรหัสวิชาและเกรด)



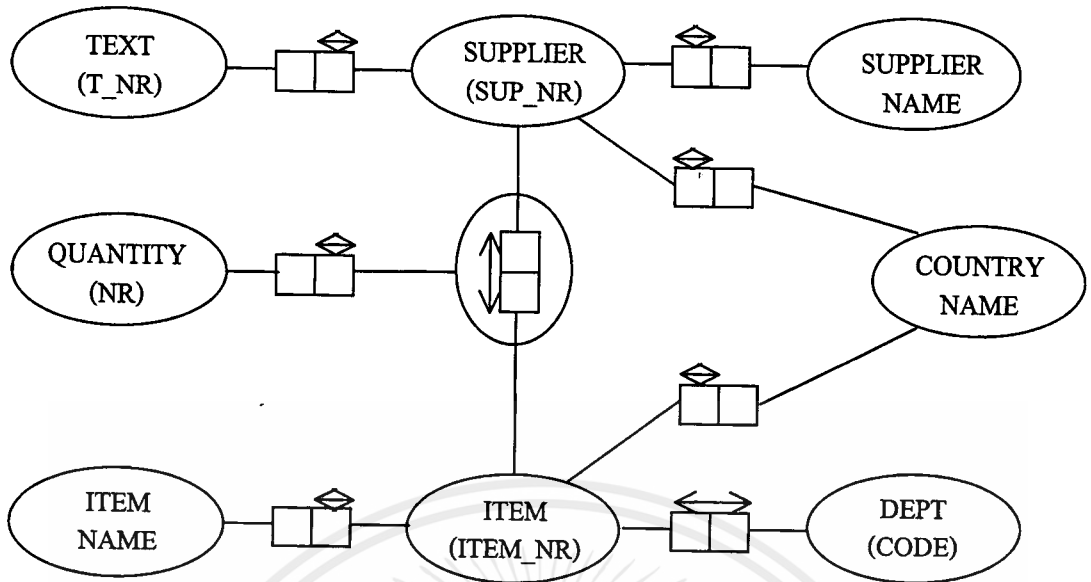
รูปที่ 2.30 Nested Fact Type มีความหมายเหมือนรูปที่ 29

### 2.6.2 ONF อัลกอริทึม (The Optimal Normal Form Algorithm)

เป็นวิธีการจัดกลุ่มความจริงใน Conceptual Schema ให้เป็น Relational Database Schema โดยมีหลักการ โดยสรุปดังต่อไปนี้

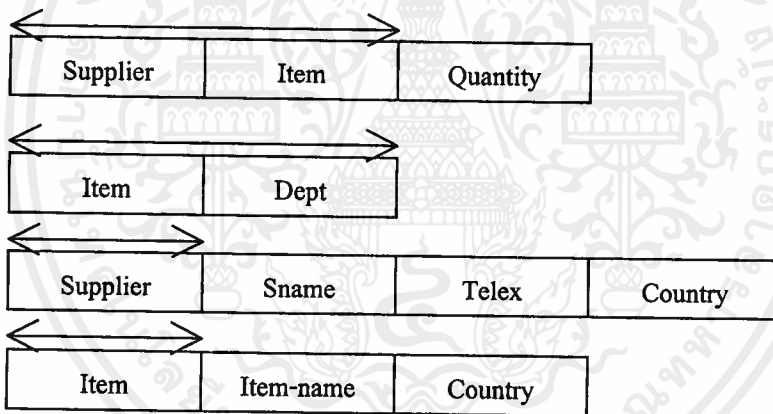
1. สร้าง 1 รีเลชัน สำหรับชนิดความจริงแบบไบนารี ซึ่งมีความสัมพันธ์แบบ many to many โดยที่ Unique Identifier ของชนิดเอนติตี้ที่เกี่ยวข้องทั้งหมดเป็น Primary Key
2. สร้าง 1 รีเลชัน สำหรับแต่ละชนิดความจริงแบบ n-ary โดยที่ Unique Identifier ของชนิดเอนติตี้ซึ่งมี Role ถูกบังคับด้วย Uniqueness Constraint
3. พิจารณาชนิดเอนติตี้ที่เกี่ยวข้องกับความจริงแบบไบนารีที่มีความสัมพันธ์แบบ one to one หรือ many to one โดยที่ role ของชนิดเอนติตี้เหล่านี้ถูกบังคับด้วย Uniqueness Constraint ให้สร้างรีเลชัน โดยมี Unique Identifier ของชนิดเอนติตี้เหล่านั้นเป็น Candidate Key

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.31 ตัวอย่างแบบจำลองข้อมูล (Conceptual Schema)

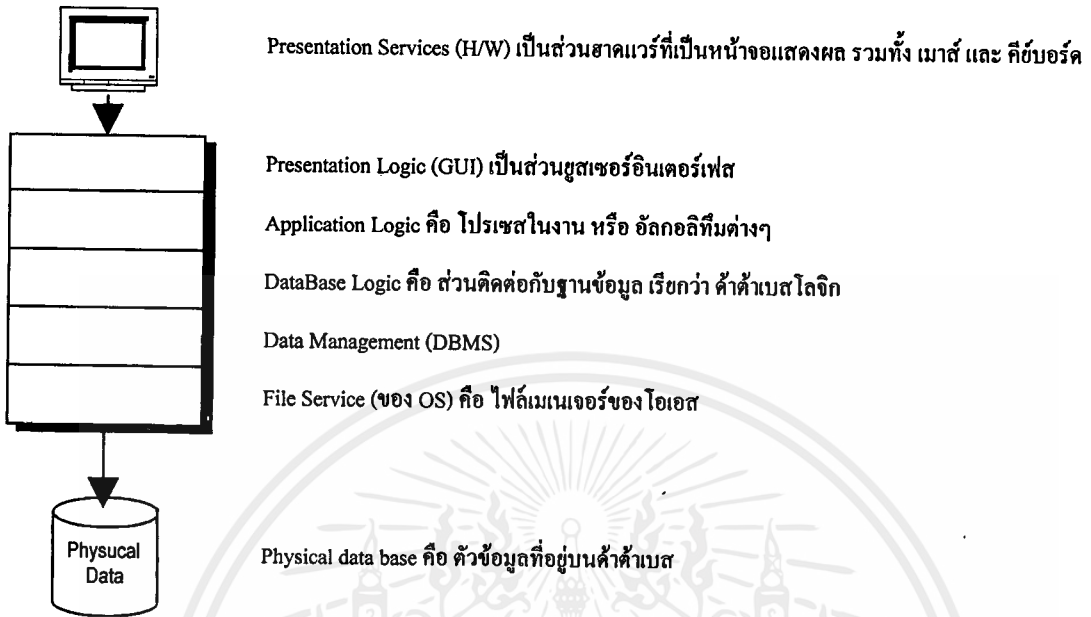
จาก Conceptual Schema ในรูปที่ 31 เมื่อใช้ ONF อัลกอริทึมจะได้รีเลชันดังนี้



รูปที่ 2.32 รีเลชันของแบบจำลองรูปที่ 31

## 2.7 ระบบไคลเอ็นท์เซิร์ฟเวอร์ (Client/Server)

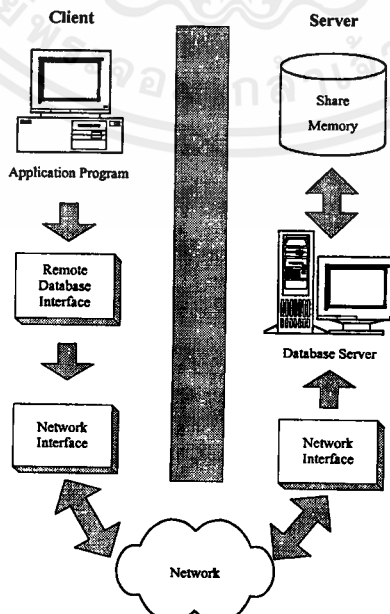
ไคลเอ็นท์เซิร์ฟเวอร์ หมายถึง การแบ่งโปรเซส (Process) ออกเป็น 7 ส่วนไว้ตามฮาร์ดแวร์ (Hardware) ต่าง ๆ ดังรูปที่ 33



รูป 2.33 การแบ่งโปรเซสออกเป็นส่วน ๆ ไว้ตามฮาร์ดแวร์

ไคลเอ็นท์ (Client) หมายถึง ระบบที่ทำการร้องขอการบริการจากส่วนเซิร์ฟเวอร์ โดยเปรียบไคลเอ็นท์เป็นผู้ให้บริการ (Requester) ทำการขอบริการ (Service) จากเซิร์ฟเวอร์

เซิร์ฟเวอร์ (Server) หมายถึง ระบบที่ให้บริการตามที่ไคลเอ็นท์ (Client) ร้องขอ โดยเซิร์ฟเวอร์หนึ่งตัวสามารถให้บริการกับไคลเอ็นท์ ได้มากกว่าหนึ่งบริการ (Service)



รูปที่ 2.34 แสดงลักษณะการเชื่อมต่อไคลเอ็นท์กับดาต้าเบสเซิร์ฟเวอร์ (Database Server)

จากรูปที่ 34 แสดงความสัมพันธ์ของการทำงานระหว่างไคลเอ็นท์และเซิร์ฟเวอร์ในลักษณะของดาต้าเบสเซิร์ฟเวอร์ (Database Server) การไหลของข้อมูลและการควบคุมของระบบฐานข้อมูลบนไคลเอ็นท์เซิร์ฟเวอร์ (Database Client/Server)

โปรแกรมแอปพลิเคชัน (Application Program) ประกอบด้วย Database Access Request เช่น SQL Statement ในระหว่างการคอมไพล์ (Compile) และลิงค์ (Link) ส่วนประกอบต่าง ๆ ของ Program นั้น Remote Database Access Stub จาก Database Application Library จะทำการเชื่อมต่อกันเป็นโปรแกรมที่สมบูรณ์ จากนั้น Remote Database Interface จะทำการจัดรูปแบบของ Database Access Request เพื่อส่งผ่านไปทางเครือข่าย (Network) โดยผ่านส่วนของการเชื่อมต่อเครือข่าย (Network Interface) ไปยังส่วนเซิร์ฟเวอร์

ในส่วนเซิร์ฟเวอร์นั้น Network Interface จะทำหน้าที่รับ Message ที่ส่งมาจากส่วนไคลเอ็นท์ และนำส่งต่อไปยังดาต้าเบสเซิร์ฟเวอร์ เพื่อทำการประมวลผลตามคำร้องขอ (Request) ที่ได้รับและส่งผลลัพธ์กลับคืนสู่โปรแกรมแอปพลิเคชัน

## 2.8 โอเพนดาตา เบสคอนเนกติวิตี (ODBC : Open Database Connectivity)

เป็นมาตรฐานการอินเตอร์เฟสซึ่งทำให้โปรแกรมประยุกต์สามารถเข้าถึงข้อมูลที่เก็บอยู่ในระบบฐานข้อมูล ซึ่งถูกดูแลโดย DBMS (Database Management System) โดยใช้คำสั่งของ SQL เป็นมาตรฐานในการติดต่อ โดยก่อนที่จะมีการพัฒนา ODBC ขึ้นมานั้น การพัฒนาแอปพลิเคชันจะขึ้นกับ DBMS ซึ่งจะมีลักษณะเป็น Embedded SQL ซึ่งประสิทธิภาพของ Embedded SQL นี้จะขึ้นอยู่กับฮาร์ดแวร์และสภาพแวดล้อมของระบบปฏิบัติการ ซึ่งจะทำให้ซอฟต์แวร์จะต้องถูก Compile ใหม่สำหรับแต่ละสภาพแวดล้อม

การที่แอปพลิเคชัน ใช้ ODBC เป็นตัวติดต่อเพื่อการเข้าถึงข้อมูลนั้นจะทำให้แอปพลิเคชันสามารถเข้าถึงข้อมูล DBMS ที่แตกต่างกันได้ ซึ่งทำให้ผู้พัฒนาแอปพลิเคชันสามารถนำแอปพลิเคชันที่พัฒนาแล้วไปใช้กับ DBMS ตัวใดก็ได้ โดยที่ผู้ใช้สามารถเพิ่มเติมโมดูล (Database Driver) ที่จะ link แอปพลิเคชันให้ใช้กับ DBMS ที่ต้องการได้

### 2.8.1 ส่วนประกอบในการอินเตอร์เฟสกับ ODBC

1. Library ของฟังก์ชัน ซึ่งจะช่วยให้แอปพลิเคชันติดต่อกับ DBMS เพื่อที่จะเอ็กรหัสคิวรี่คำสั่งแล้วดึงข้อมูลออกมา
2. ไวยากรณ์ของ SQL ซึ่งยึดถือตามมาตรฐานของ X/Open และ SQL Access Group (SAG) SQL CAE Specification (1992)
3. มาตรฐานรหัสข้อผิดพลาด (Error Code)
4. วิธีมาตรฐานในการติดต่อและการล็อกเข้า DBMS
5. มาตรฐานประเภทของข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.8.2 ข้อดีในการอินเตอร์เฟซกับ ODBC

1. สามารถกำหนดคำสั่ง SQL ได้ทั้งในซอสโค้ด (Source code) ของแอปพลิเคชันนั้น และขณะทำงาน (Run time)
2. ซอสโค้ดเดียวกันสามารถเข้าถึงข้อมูลที่เกี่ยวข้องบน DBMS คนละตัวกันได้
3. โพรโทคอลในการสื่อสารข้อมูล (Data Communication Protocol) ระหว่าง แอปพลิเคชันแต่ละตัวไม่มีความสำคัญ
4. สามารถปรับแต่งรูปแบบของข้อมูลที่รับและส่งระหว่างแอปพลิเคชันกับตัว ODBC ได้ตามต้องการ

### 2.8.3 ส่วนประกอบของโอดีบีซี (ODBC Component)

1. แอปพลิเคชัน (Application) มีหน้าที่หลักในการเข้าถึงข้อมูลภายในฐานข้อมูล คือ การเรียกใช้ฟังก์ชันของ ODBC และการส่งคำสั่ง SQL และการรับข้อมูลผลลัพธ์
2. ไดรฟ์เวอร์แมนเนเจอร์ (Driver Manager) จะทำการ โหลดไดรฟ์เวอร์ในนามของ แอปพลิเคชัน
3. ไดรฟ์เวอร์ (Driver) ปฏิบัติตามฟังก์ชันที่เรียกมาจากแอปพลิเคชันส่งคำสั่ง SQL ไปยังดาต้าซอส (Datasource) และรอรับผลตอบสนองเพื่อส่งต่อไปยังแอปพลิเคชัน นอกจากนี้ยังสามารถแก้ไขปรับปรุงรูปแบบของการร้องขอจาก แอปพลิเคชัน เพื่อให้สอดคล้องกับ DBMS
4. ดาต้าซอส (Data source) ประกอบด้วยข้อมูลที่ผู้ใช้ต้องการจะเข้าถึง รวมไปถึงระบบปฏิบัติการที่เกี่ยวข้องกับ และเน็ตเวิร์กแพลตฟอร์ม (Network Platform)

### 2.8.4 แอปพลิเคชัน

แอปพลิเคชัน จะทำงานต่าง ๆ ผ่าน ODBC Interface ดังต่อไปนี้

1. ทำการร้องขอการติดต่อไปยังดาต้าซอส
2. ส่ง SQL ไปยังดาต้าซอส
3. กำหนดเนื้อที่และรูปแบบของข้อมูลสำหรับเก็บผลลัพธ์จากการประมวลผลคำสั่ง SQL
4. ร้องขอผลลัพธ์
5. ประมวลผลข้อผิดพลาดที่อาจจะเกิดขึ้น
6. รายงานผลลัพธ์กลับไปยังผู้ใช้
7. ร้องขอการคอมมิต (Commit) และการโรลแบค (Rollback) สำหรับการควบคุมทรานแซกชัน
8. ยกเลิกการติดต่อจากดาต้าซอส

## 2.8.5 ไครฟเวอร์แมนเนเจอร์

ไครฟเวอร์แมนเนเจอร์เป็นไดนามิกลิงค์ไลบรารี (Dynamic-link library, DLL) จุดประสงค์หลักของไครฟเวอร์แมนเนเจอร์คือการ โหลดไครฟเวอร์ โดยจะทำตามลำดับดังต่อไปนี้

1. ใช้ ODBC.INI ไฟล์เพื่อแมปชื่อของคาคำขอสเพื่อค้นหาไครฟเวอร์ดีแอลแอล
2. ทำการประมวลผลโดย ODBC Initialization Call
3. จัดการ ODBC ฟังก์ชันสำหรับแต่ละไครฟเวอร์
4. จัดการพารามิเตอร์ให้ถูกต้องสำหรับการเรียกใช้ ODBC

## 2.8.6 ไครฟเวอร์

ไครฟเวอร์เป็นดีแอลแอลที่อิมพลิเมนต์ ODBC ฟังก์ชันคอล และทำการติดต่อกับข้อมูลที่คาคำขอส โดยไครฟเวอร์แมนเนเจอร์จะทำการ โหลดไครฟเวอร์เมื่อแอปพลิเคชันได้เรียกใช้ฟังก์ชัน SQL Brouwer Connect SQL Connect หรือ SQL Driver Connect โดยไครฟเวอร์จะทำตามขั้นตอนดังต่อไปนี้เพื่อตอบรับการเรียก ODBC ฟังก์ชันของแอปพลิเคชัน

1. สร้างการติดต่อไปยังคาคำขอส
2. ส่งการร้องขอไปยังคาคำขอส
3. แปลงข้อมูลให้เป็นรูปแบบอื่น ถ้าเป็นการเรียกโดยแอปพลิเคชัน
4. ส่งผลลัพธ์ไปยังแอปพลิเคชัน
5. ถ้าเกิดข้อผิดพลาดขึ้นจะมีการส่ง ข้อผิดพลาดกลับไปยังแอปพลิเคชันที่เรียกมา
6. ทำการประกาศและจัดการเกี่ยวกับเคอร์เซอร์ถ้าจำเป็น
7. อินนิเชียลทรานแซกชันถ้าคาคำขอสนั้น ๆ ต้องการ

## 2.8.7 ชนิดของไครฟเวอร์

ODBC กำหนดชนิดของไครฟเวอร์ไว้ 2 รูปแบบ คือ

1. ซิงเกิลเทียร์ (Single-tier) ไครฟเวอร์ชนิดนี้จะมีหน้าที่เอ็กซีคิวท์ฟังก์ชันที่ส่งมาจาก ODBC รวมทั้งคำสั่ง SQL ด้วย ดังนั้น ไครฟเวอร์ชนิดนี้จึงมีหน้าที่บางส่วนของคาคำขอสอยู่ด้วย
2. มัลติเฟลิตเทียร์ (Multiple-tier) ไครฟเวอร์ชนิดนี้จะมีหน้าที่เอ็กซีคิวท์เฉพาะฟังก์ชันที่ส่งมาจากODBCเท่านั้น ส่วนคำสั่งเอซคิวแอลจะส่งไปให้กับตัวคาคำขอสซึ่งระบบฯ หนึ่งอาจจะประกอบด้วยไครฟเวอร์ทั้งสองชนิดเลขก็ได้

### 2.8.8 ระดับข อตกลงของโอดีบีซี (ODBC Conferment Level)

ในการใช้มาตรฐาน ODBC ในการเชื่อมต่อนั้นจะต้องมีข้อตกลงมาตรฐาน เพื่อที่จะใช้คน พัฒนาส่วนของไครฟเวอร์สามารถครอบคลุมฟังก์ชันต่าง ๆ ที่จำเป็นได้ครบ และทางด้านแอฟพลิเคชั่น สามารถเลือกใช้ไครฟเวอร์ตามฟังก์ชันมาตรฐาน ได้โดยง่าย

โดยระดับข้อตกลงของไครฟเวอร์แบ่งออกเป็น

1. ODBC API ซึ่งแบ่งย่อยออกได้เป็น
  - Core API อ้างอิงมาตรฐานของ X/OPEN&SQL Access Group Call Level Interface
  - Level1 API
  - Level2 API
 ซึ่งสองระดับสุดท้ายถือว่าเป็นระดับ Extended Functionality
2. ODBC SQL Grammar และ ODBC SQL Datatypes แบ่งออกเป็น
  - Minimum SQL Grammar ซึ่งเป็นระดับพื้นฐาน
  - Core SQL Grammar ยึดตามมาตรฐานที่ระบุใน X/OPEN&SQL Access Group SQL CAE Specification (1992)
  - Extended SQL Grammar Support Extension SQL ของ DBMS ทั่วไป

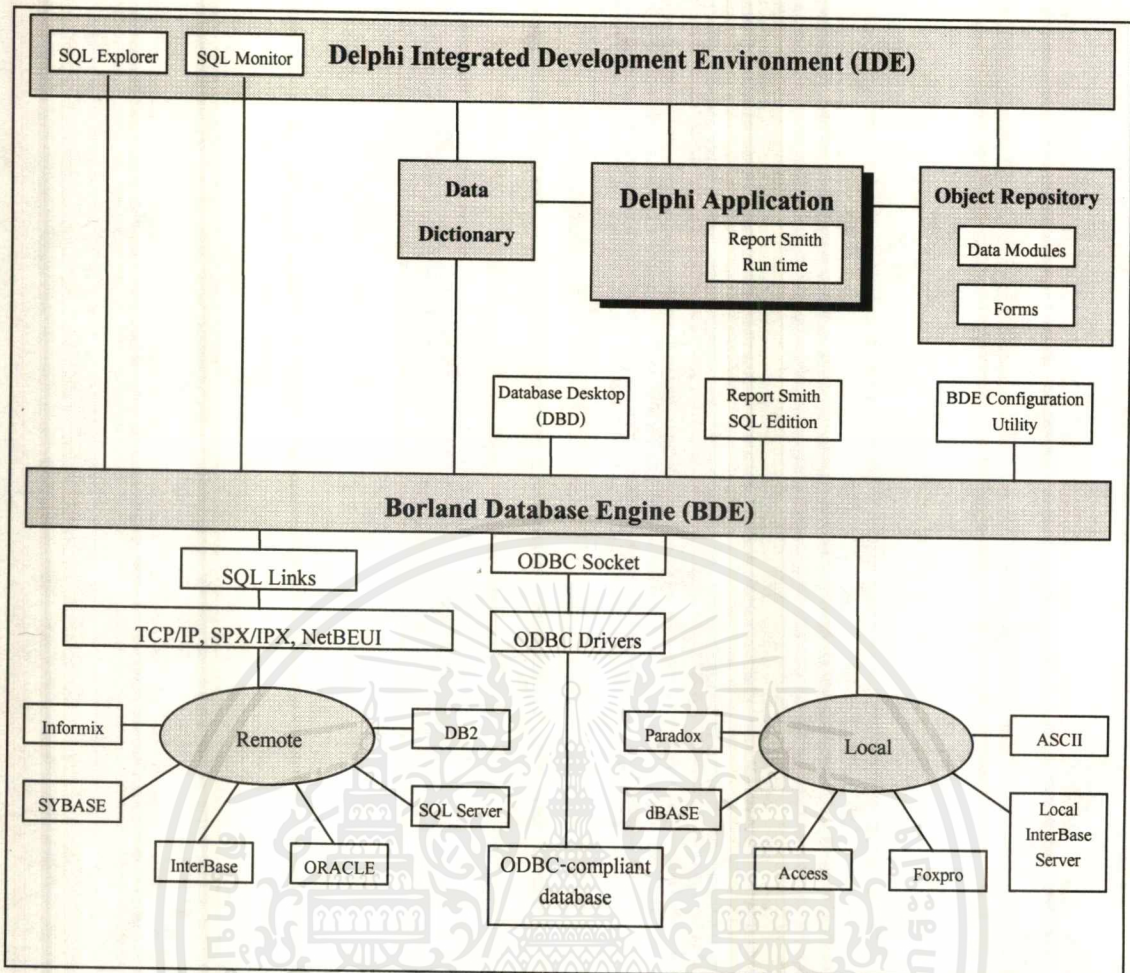
### 2.8.9 คอนเน็คชันและทรานแซกชัน (Connection & Transaction)

ก่อนที่โปรแกรมประยุกต์จะสามารถติดต่อกับ ODBC จะต้องทำการร้องขอ ODBC เพื่อของ Environment Handle, henv และก่อนที่จะสามารถติดต่อกับดาต้าซอลได้นั้น แอฟพลิเคชั่นจะต้องร้องขอ เพื่อของ Connection Handle, hdbc และยังคงต้องใช้ henv, hdbc ไว้สำหรับการอ้างอิงภายหลังด้วย

### 2.9 ความรู้ เกี่ยวเครื่องมือและลักษณะของเคลไฟกับดาต ามแบบ two-tiered (Delphi database tools and two-tiered database application)

วิธีการเชื่อมต่อกับฐานข้อมูล โดยใช้เคลไฟ (Delphi) สามารถทำได้โดยผ่านบีดีอี (BDE : Borland Database Engine) ใน BDE มี Built-in Native Driver สำหรับพาราดอก (Paradox), ดีเบส (dBase), แอซเซสซ์ (Access), ฟอกซ์โปร (Foxpro) และ เท็กซ์ไฟล์ (Text File) นอกจากนี้ยังสามารถเพิ่มไดร์เวอร์ สำหรับออราเคิล (Oracle), ซิเบส (Sybase), อินเทอร์เบส (Interbase), อินฟอร์มิกซ์ (Informix) และอื่นๆ ได้อีกด้วย

ในเคลไฟ สามารถทำการเข้าถึงข้อมูลผ่าน ODBC ได้ด้วย เพราะใน BDE มี ODBC Socket สำหรับผู้ใช้ที่ต้องการใช้ ODBC Driver เพื่อเข้าถึงข้อมูล

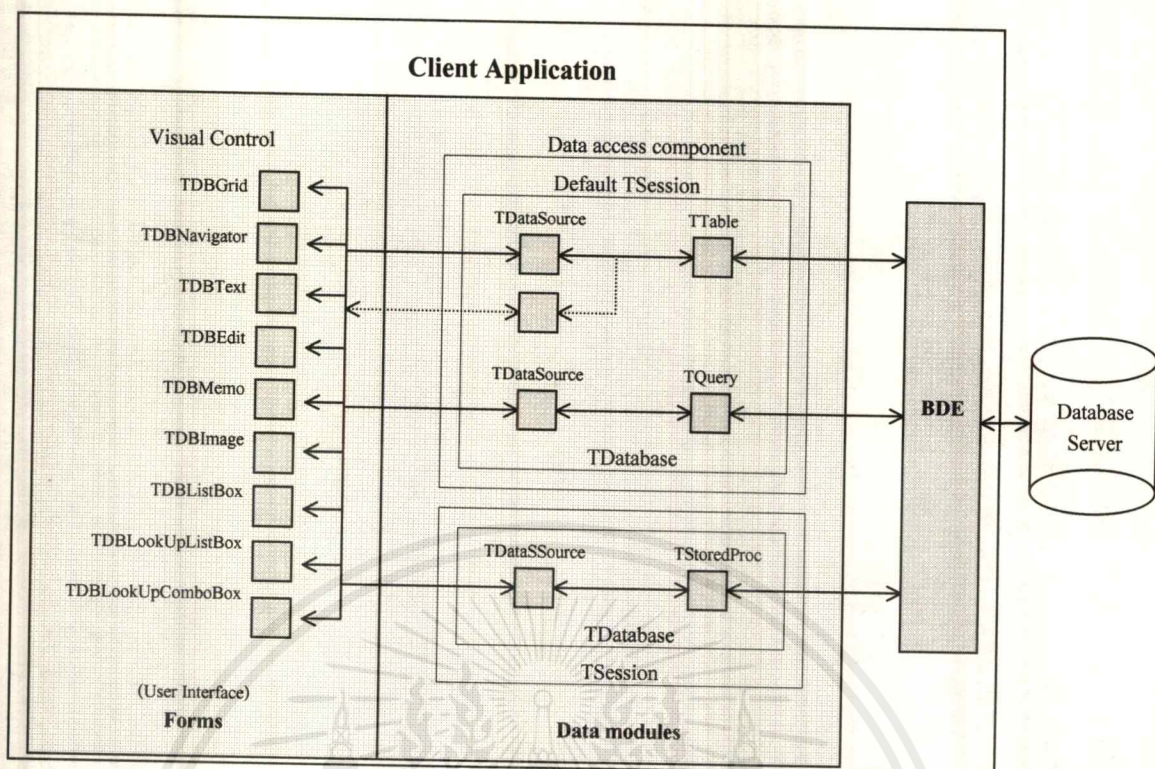


รูป 2.35 Delphi Client/Server Suite two-tiered development environment

#### ข้อดีของ BDE ที่มีในเดลไฟ








1. สามารถ Share Data กับ BDE Application อื่น ได้โดยไม่เกิด Conflict เนื่องจากการ Access Data ทุกตัวที่ทำถูกควบคุมโดย BDE
2. สามารถ Access Data โดยใช้ Bidirectional Scrollable Cursor ได้ถึงแม้ว่า Database Server จะไม่สนับสนุน Bidirectional Cursor
3. สามารถใช้งานกับ Data ที่มาจาก Plat Form ที่แตกต่างกันได้ เช่น สามารถ Query โดยผ่านการ Join ระหว่าง Data บน SQL Server Table กับ Paradox Table และ ORACLE Table ได้
4. BDE ทำการควบคุมการแปลง Data Type ให้โดยอัตโนมัติ ซึ่ง User ไม่จำเป็นต้องทำการแปลง Data Type เอง
5. สามารถทำการเคลื่อนย้าย Data ได้สะดวกและรวดเร็วระหว่าง Plat Form ที่แตกต่างกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้















รูป 2.36 Two-tiered database application architecture

### 2.9.1 เคลไฟฟ้าต แอคเซสคอมโปเนนท์ (Delphi Data Access component summary)

- 
**TDatabase** ทำหน้าที่จัดการกับการติดต่อกับฐานข้อมูล เหมาะกับ Remote Database Server
- 
**TTable** ทำหน้าที่จัดการเกี่ยวกับการอ่านและเขียนข้อมูลลงตาราง สร้างตาราง และสร้างอ็เดกซ์ (Index)
- 
**TQuery** ทำหน้าที่จัดการเกี่ยวกับการใช้ SQL ในการอ่านและเขียนข้อมูล
- 
**TStoredProc** ทำหน้าที่ควบคุมและกำหนดให้ผู้ใช้สามารถทำงาน โดยผ่านค่าพารามิเตอร์ (Parameter) เข้าไปยัง Stored Procedure นั้น
- 
**TDataSource** ทำหน้าที่จัดการเกี่ยวกับการติดต่อระหว่าง TTable, Tquery หรือ TStoredProc กับส่วนดาต้าคอนโทรล (Data Control Component)
- 
**TBatchMove** ทำหน้าที่เกี่ยวกับการคัดลอกตารางโดยพิจารณาทั้งโครงสร้างและชนิดข้อมูล
- 
**TReport** ทำหน้าที่จัดการเกี่ยวกับการพิมพ์รายงาน โดยใช้ Report Smith (เครื่องมือที่ใช้ในการทำรายงานที่มีในเคลไฟ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.9.2 เบลโฟดาต คอนโทรลคอมโปเนนท์ (Delphi Data Controls component summary)

	TDBGrid	ใช้ในการแสดงผล Data Set ในรูปตาราง
	TDBNavigator	ใช้ในการควบคุม Record ของ Data Set
	TDBText	ใช้ในการแสดงผลตัวอักษรของ Field ใน Data Set ที่ระบุโดยไม่สามารถแก้ไขข้อมูล
	TDBEdit	ใช้ในการแสดงผลตัวอักษรของ Field ใน Data Set ที่ระบุโดยสามารถแก้ไขข้อมูลได้ถ้ากำหนดให้มีการเปลี่ยนแปลงข้อมูล
	TDBMemo	ใช้ในการแสดงผลในรูปแบบชุดอักษรจำนวนมาก
	TDBImage	ใช้ในการแสดงข้อมูลที่เป็นรูปภาพ
	TDBListBox	ใช้ในการแสดง List ของค่าที่เป็นไปได้สำหรับ Field ใน Windows List Box
	TDBCComboBox	ใช้ในการแสดง List ของค่าที่เป็นไปได้สำหรับ Field ใน Windows List Box โดยสามารถซ่อน Windows List Box ได้
	TDBCheckBox	ใช้ในการแสดงข้อมูลในลักษณะที่เป็นไปได้เพียง 2 ค่าคือ Check กับ UnCheck
	TDBRadioGroup	ใช้ในการแสดงผลค่าที่เป็นไปได้สำหรับ Field ใน Data Set โดยจะมี Check Box ในแต่ละ Value ที่เป็นไปได้
	TDBLookupListBox	ใช้ในการแสดง Content ของ Field ใน Data Set อื่นในรูปแบบ List Box เพื่อนำค่าไปใช้ใน Data Set ที่ต้องการ
	TDBLookupComboBox	ใช้ในการแสดง Content ของ Field ใน Data Set อื่นในรูปแบบ Combo Box เพื่อนำค่าไปใช้ใน Data Set ที่ต้องการ

## 2.10 ระบบปฏิบัติการ Windows NT 4.0

Windows NT ได้กลายมาเป็นมาตรฐานของระบบปฏิบัติการ สำหรับไมโครคอมพิวเตอร์ ไม่เพียงแต่ติดต่อกับผู้ใช้ด้วยระบบกราฟฟิก แต่ยังมีความสามารถในการทำงานหลายงาน (Multitask) และปรับปรุงระบบจัดการหน่วยความจำด้วย โดยเฉพาะความสามารถที่เพิ่มเข้ามา เพื่อขยายประโยชน์การใช้งานของเครื่องคอมพิวเตอร์ต่างชนิดกันตลอดจนการเชื่อมต่อคอมพิวเตอร์ในระบบเน็ตเวิร์กง่าย เพื่อให้ผู้ใช้สามารถติดต่อกันได้ วินโดวส์เอ็นที (Windows NT) สามารถทำงานได้โดยปราศจากดอส (DOS) บนเครื่องไมโครคอมพิวเตอร์ (Microcomputer)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะของวินโดวส์เอ็นที (Windows NT) ในลักษณะที่ว่า พอททาบิลิตี (Portability) หมายความว่า วินโดวส์เอ็นที (Windows NT) พร้อมทำงานบนไมโครโปรเซสเซอร์ 32 Bits ของเครื่องคอมพิวเตอร์ใด ๆ ผลที่เกิดขึ้นจากความสามารถในหลายระบบคือ รูปแบบ (scalability) ความสามารถในการปรับระบบปฏิบัติการใช้กับระบบฮาร์ดแวร์ (hardware) หลายรูปแบบรวมถึงระบบรักษาความปลอดภัย ทำให้ควบคุมสิทธิการใช้งานทรัพยากรคอมพิวเตอร์ของผู้ใช้แต่ละคน รองรับการเชื่อมต่อเครื่องคอมพิวเตอร์ระบบเน็ตเวิร์ก (network) สามารถติดต่อและใช้ข้อมูลร่วมกันได้ปกติ รวมไปถึงโปรแกรมประยุกต์ที่ใช้ POSIX Standard (POSIX :The Portable Operating System Interface สร้างโดย IEEE เพื่อกำหนด API ที่ใช้เป็นมาตรฐานสำหรับบริการของระบบปฏิบัติการ) ส่วนด้านการใช้งานหน่วยความจำ วินโดวส์เอ็นที (Windows NT) สามารถใช้งานหน่วยความจำได้มากถึง 4 จิกะไบต์ และการเข้าถึงข้อมูลในฮาร์ดดิส (Harddisk) แบบ 64 บิต (Bits) ทำให้แทบไม่มีการจำกัดขนาดฮาร์ดดิสเลย

วินโดวส์เอ็นที (Windows NT) สามารถใช้งานโปรแกรมประยุกต์ได้มากกว่า 1 โปรแกรม ในเวลาเดียวกันที่รู้จักกันว่าเป็น time sharing system ทำให้รองรับการทำงานหลายผู้ใช้ที่ติดต่อผ่าน terminal โดยโปรแกรมประยุกต์ที่แตกต่างกันและเป็น preemptive multitasking ซึ่งระบบปฏิบัติการจะทำการควบคุมแทนที่จะให้โปรแกรมคืนการควบคุมให้ โปรแกรมประยุกต์บน วินโดวส์เอ็นที (Windows NT) สามารถถูกแบ่งออกเป็นหลาย thread ซึ่งสามารถทำงานไปพร้อม ๆ กันได้ด้วย และจะเพิ่มประสิทธิภาพโดยเฉพาะเครื่องที่มีไมโครโปรเซสเซอร์มากกว่า 1 ตัว ขึ้นไปด้วยกระบวนการ symmetric multiprocessing วินโดวส์เอ็นที (Windows NT) จะควบคุมการเข้าถึงหน่วยความจำและอุปกรณ์อย่างเข้มงวด นอกจากนี้ระบบ file ที่ก้าวหน้าของระบบวินโดวส์เอ็นที (Windows NT) จะเป็นแบบ fault-tolerance ซึ่งเป็นความสามารถในการทำงานต่อไปอย่างน่าเชื่อถือ แม้ว่าจะเกิดความผิดพลาดจากอุปกรณ์ฮาร์ดแวร์ และมีอยู่ในระบบ file แบบ NTFS (NT file system) ทำหน้าที่รักษาขั้นตอนการทำงาน (transaction log) เพื่อแน่ใจในความมั่นคงของโครงสร้างฮาร์ดดิสก์แม้จะล้มเหลวด้วยเหตุการณ์ไม่คาดถึง

สำหรับความต้องการที่จำเป็นสำหรับเซิร์ฟเวอร์ (Server) บนระบบเน็ตเวิร์ก (network) ก็ สามารถหาได้ใน version ที่เรียกว่า วินโดวส์เอ็นที Advanced Server โดยเพิ่มความสามารถทางด้าน fault tolerance ให้มากขึ้น รวมถึงการทำ disk mirroring และ disk striping ร่วมกับการทำ parity ด้วยวิธี disk mirroring วินโดวส์เอ็นที (Windows NT) จะเก็บสำเนาของฮาร์ดดิสก์ตัวหนึ่งไว้บนฮาร์ดดิสก์อีกตัวหนึ่ง การทำ disk mirroring จะน่าเชื่อถือและมีประสิทธิภาพ อย่างไรก็ตาม เมื่อใช้ disk striping ร่วมกับ parity ก็จะมีเพิ่ม การบันทึกข้อมูลในส่วนที่สามารถทำให้ข้อมูลกู้กลับมาได้ วินโดวส์เอ็นที (Windows NT) ยังเพิ่มความสามารถในการควบคุมระบบ network ทำให้สามารถ

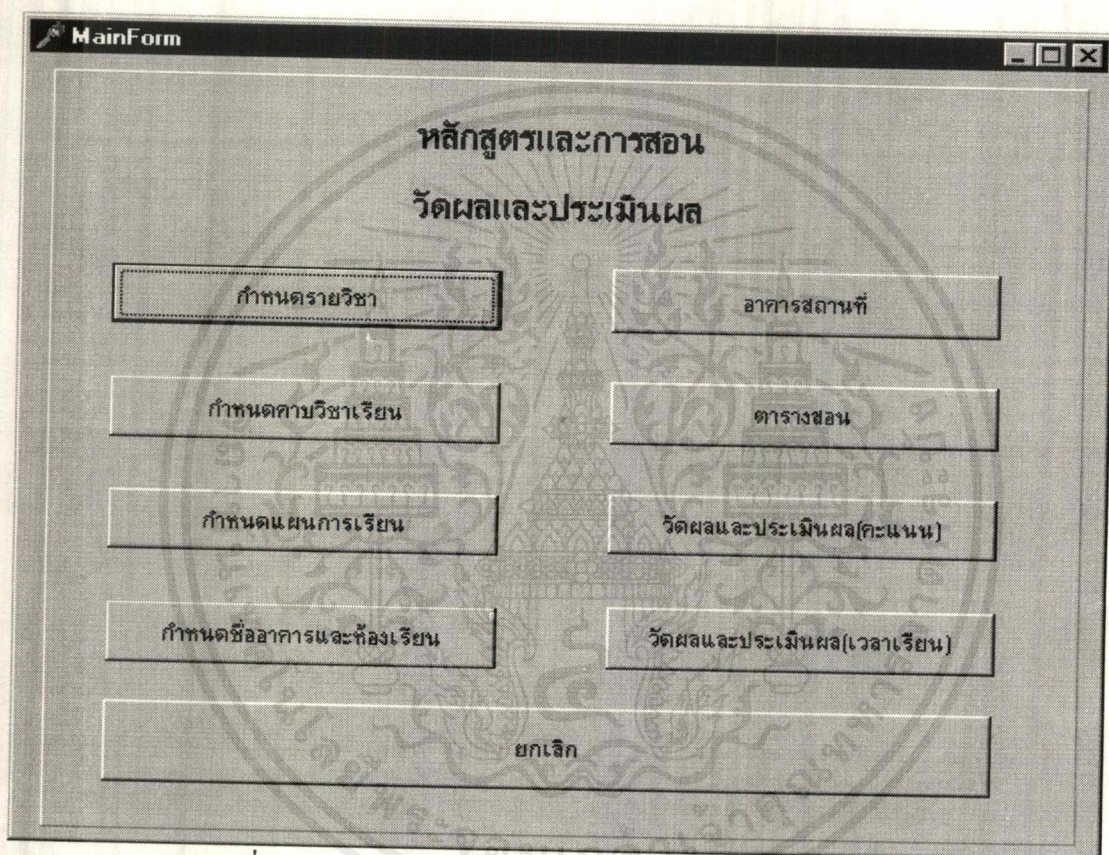
- Sharing files คือ การใช้ file ร่วมกัน
- Sharing Printers คือ การใช้เครื่องพิมพ์ร่วมกัน
- Exchanging electronic mail คือ ระบบ mail ใน network
- Maintenaning group schedules การจัดการตารางกลุ่มผู้ใช้ซึ่งอาจนำไปใช้ในการนัดพบบุคคล
- Running distributed applications ความสามารถในการรองรับการใช้งานโปรแกรมประยุกต์ที่กระจายกัน

ไปโดยรองรับการทำงานแบบ RPC (remote procedure call) ซึ่งทำให้สามารถพัฒนาโปรแกรมประยุกต์ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้เชิงพาณิชย์ การค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อติดต่อกับโปรแกรมประยุกต์ อื่นที่อยู่บน network เช่นการใช้ RPC เพื่อรวบรวมข้อมูลเข้าด้วยกันจากระบบฐานข้อมูลจากเครื่องคอมพิวเตอร์ที่อยู่บนระบบ network วินโดว์เอ็นที RPC จะยึดมาตรฐาน DCE (The Open System Foundation's Distributed Computing Environment)

## 2.11 แอปพลิเคชัน

“ระบบสารสนเทศเพื่องานปฏิบัติการและงานบริหาร” เป็นงานจัดการฐานข้อมูลให้มีประสิทธิภาพ และแยกหน้าที่การทำงานเป็นส่วน ๆ ตามลักษณะงาน ต่อไปนี้เป็นตัวอย่างงานและการใช้งาน



รูปที่ 2.37 เมื่องานหลักสูตรและการสอน-งานวัดผลและประเมินผล

### 2.11.1 งานหลักสูตรและการสอน

ประกอบไปด้วยงานที่เป็นการกำหนดหลักสูตร โดยรับมาจากกระทรวงซึ่งจะมีหลักสูตรของอาชีวะและราชวมงคล ตามความเหมาะสม แล้วนำมาบรรจุเป็นหลักสูตรการสอนของชั้นปีนั้น ๆ รวมทั้งการจัดเวลาเรียน กำหนดอาจารย์ผู้สอน กำหนดวิชาเรียนให้กับแผนกต่าง ๆ ดังต่อไปนี้

1. กำหนดรายวิชา
2. กำหนดคาบวิชาเรียน
3. กำหนดแผนการเรียน
4. กำหนดชื่ออาคารและห้องเรียน
5. อาคารสถานที่

เอกสารนี้เป็นเอกสารที่สงวนเวลาสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 6. ตารางสอน

ซึ่งงานดังกล่าวข้างต้นเป็นงานที่อยู่ในหน้าที่ความรับผิดชอบของ “งานหลักสูตรและการสอน” ทั้งสิ้น ซึ่งมีหน้าที่หลักในการกำหนดหลักสูตรและการเรียนการสอนให้กับแผนการเรียนต่าง ๆ และบุคลากรที่เกี่ยวข้อง ซึ่งจะได้แจกแจงรายละเอียดดังต่อไปนี้

## 1. กำหนดรายวิชา

เป็นการกำหนดแผนกและชื่อวิชา โดยแผนกนี้มีวิชาเรียนอะไรบ้าง เรียนกี่หน่วย และแต่ละวิชาต้องผ่านวิชาอะไรมาก่อน

รหัสวิชา	รหัสแผนก	ชื่อวิชา
111-1112	013811	Japan
111-4577	012561	English
117-1155	013812	fhfh
119-4146	013812	Wing Back

รูปที่ 2.38 การกำหนดรายวิชา

แบ่งเป็น 2 ส่วนคือ ส่วนแผนกเป็นกำหนดแผนก และชื่อวิชาเป็นการกำหนดรายละเอียดของวิชาที่มีการบรรจุลงตามหลักสูตร

เมื่อทำการคีย์ข้อมูลเรียบร้อยแล้ว กดปุ่ม “บันทึก” เพื่อบันทึกข้อมูลลงฐานข้อมูลเมื่อบันทึกเสร็จจะปรากฏดังรูป และสามารถตรวจเช็คข้อมูลถูกบันทึกลงฐานข้อมูลโดยเลื่อนดูข้อมูลในตาราง



รูปที่ 2.39 หลังจากกดปุ่ม “บันทึก”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อต้องการแก้ไขเลือกข้อมูลที่ต้องการแก้ไข แล้วกดปุ่ม“แก้ไข” ดังรูปจะเห็นได้ว่า ข้อมูลที่แก้ไขจะเป็นเฉพาะส่วนชื่อวิชาเท่านั้น แล้วทำการแก้ไขในส่วนที่ต้องการ บันทึกเมื่อแก้ไขเสร็จ ยกเลิกเมื่อไม่ต้องการแก้ไข

รูปที่ 2.40 การแก้ไขข้อมูล

เมื่อต้องการลบวิชาที่ไม่ต้องการทิ้งให้เลือกข้อมูลตัวนั้นแล้วกดปุ่มลบ จะปรากฏข้อความเพื่อยืนยันการลบข้อมูล กด “Yes” ข้อมูลจะถูกลบ กด “No” ยกเลิกการลบข้อมูล

รูปที่ 2.41 แสดงข้อความยืนยันการลบข้อมูล

ส่วนการกำหนดแผนกเมื่อต้องการเพิ่มข้อมูลแผนกให้คลิกเมาส์ ตรงช่องรหัสแผนกแล้วกดปุ่ม Ctrl และ จะแสดงดังรูป แล้วคีย์ข้อมูลที่ต้องการลงไป

รูปที่ 2.42 การบันทึกข้อมูลแผนก

## 2. กำหนดคาบวิชาเรียน

รายเรียน

เป็นการกำหนดว่าคาบเรียนแต่ชั่วโมงเริ่มและสิ้นสุดเมื่อใด แล้วนำข้อมูลนี้ไปกำหนดเป็นตารางเรียน

คาบที่	เวลา
1	08.31
2	09.30
3	10.30
4	11.30
5	12.30
6	13.30
7	14.30
8	15.30

รูปที่ 2.43 การกำหนดคาบวิชาเรียน

สำหรับปุ่มการทำงานต่าง ๆ จะคล้ายกับการกำหนดรายวิชา

## 3. กำหนดแผนการเรียน

เป็นการกำหนดวิชาเรียนรายบุคคล ซึ่งกรณีนี้อาจเป็นนักศึกษา ที่ย้ายโรงเรียนมาจากที่อื่น ลงเรียนเพิ่ม หรือไม่ผ่านวิชาต้องเรียนซ้ำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักสูตรและการสอน

กำหนดแผนการเรียนรายบุคคล

ปีการศึกษา  ภาคเรียนที่

รหัสนักศึกษา  ชื่อ  นามสกุล

รหัสวิชา  วิชา  หลักสูตร

รหัสแผนก  แผนก

จำนวนข้อมูล

บันทึก  แก้ไข  ลบ  ยกเลิก

รูปที่ 2.44 กำหนดแผนการเรียนรายบุคคล

#### 4. กำหนดชื่ออาคารและห้องเรียน

เป็นการกำหนดชื่ออาคารเพื่อนำไปใช้ในการกำหนดตารางสอนซึ่งจะทำให้ทราบว่า ห้องเรียน ณ ที่นั้น ๆ ถูกใช้ไปที่ห้อง

หลักสูตรและการสอน

ชื่ออาคาร

อาคาร  ชื่ออาคาร

อาคาร	ชื่ออาคาร
A	สำนักงาน
AA	TingTong1
AC	Thai
B	HongKok
BB	ไฟฟ้ากำลัง

จำนวนข้อมูล 13 : Record

บันทึก  แก้ไข  ลบ  ยกเลิก

รูปที่ 2.45 กำหนดชื่ออาคาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 5. อาคารสถานที่

เป็นการกำหนดโค้ดห้องเรียนเพื่อง่ายต่อการระบุห้องใช้งาน และแต่ละห้องใช้ทำอะไรบ้าง เช่น เป็น LAB เป็นห้องเรียนทฤษฎี เป็นต้น

หลักสูตรและการสอน

อาคารสถานที่

รหัสห้องเรียน  อาคาร  ชื่ออาคาร

ชั้นที่  หมายเลขห้อง

รหัสห้องเรียน	อาคาร	ชั้นที่	หมายเลขห้อง
A101	A	1	1
A301	BB	3	1
A401	A	4	1
A909	A	9	9
AC703	AC	7	3
B305	B	3	5

จำนวนข้อมูล 12 : Record

บันทึก  แก้ไข  ลบ  ยกเลิก

รูปที่ 2.46 กำหนดอาคารสถานที่

### 6. ตารางสอน

เป็นการกำหนดวันเวลา อาคารสถานที่ ชื่ออาจารย์ผู้สอน รหัสแผนก เพื่อกำหนดให้ทราบว่าวันเวลานั้นเรียนหรือสอนวิชาอะไร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักสูตรและการสอน

ตารางสอน

วัน  คาบที่  เวลา

รหัสวิชา  วิชา  รหัสห้องเรียน

นักเรียนห้อง  รหัสแผนก  แผนก

รหัสอาจารย์  ชื่อ  นามสกุล

วัน	เวลา	รหัสห้อง	รหัสวิชา	รหัสแผนก	นักเรียนห้อง	รหัสอาจารย์

บันทึก  แก้ไข  ลบ  ยกเลิก

รูปที่ 2.47 กำหนดตารางสอน

จากแอปพลิเคชันที่ได้อกล่าวมาข้างต้นการใช้งานส่วนมากจะเป็นในแนวทางเดียวกันหมด จะเห็นได้ว่าการใช้งานง่ายและไม่ซับซ้อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

### การดำเนินงานและพัฒนา

#### 3.1 ลักษณะและความสำคัญ

ในปัจจุบันการใช้งานข้อมูลของโรงเรียน ประสบปัญหาในการดำเนินการ เนื่องจากไม่มีความสะดวกในการใช้งานฐานข้อมูล เพื่อทำการแก้ไขให้ทันเหตุการณ์ หรือการเรียกใช้ฐานข้อมูลเพื่อตรวจสอบความถูกต้อง ก่อนการออกหลักฐานทางการศึกษาต่าง ๆ ดังนั้นจึงต้องการฐานข้อมูลที่ครอบคลุม

#### 3.2 หลักการและเหตุผล

ปัจจุบันงานทะเบียนเป็นงานขั้นพื้นฐานในด้านการบริหารงานทางการศึกษา แต่ในปัจจุบันมีการสำรวจพบว่างานทะเบียน อันประกอบไปด้วย การรับสมัครนักศึกษาใหม่ การจัดการเรียนตารางสอน การลงทะเบียน การออกเกรด การออกใบทรานสคริปต์ และการจัดการเกี่ยวกับข้อมูลนักศึกษา อาจารย์ ห้องเรียน ชั้นเรียน ผู้ปกครองต่างๆ มีปัญหาเนื่องจากจำนวนระบบข้อมูลมีขนาดใหญ่ และมีความยุ่งยากในการจัดทำ ก่อให้เกิดความล่าช้าในการทำงาน และความผิดพลาดต่างๆที่เกิดจากระบบไม่ดีพอ ซึ่งสามารถนำระบบเทคโนโลยีสารสนเทศเข้ามาช่วยจัดการหรือทดแทนแรงงานที่ต้องใช้ในการจัดทำเอกสารต่างๆ ได้ โดยผลที่ได้รับอย่างชัดเจนในการนำเทคโนโลยีสารสนเทศเข้ามาใช้ คือ ลดแรงงาน บุคลากร ลดความผิดพลาด ได้ผลงานมากกว่าในเวลาเท่ากัน สามารถให้บริการโดยไม่ขึ้นกับเวลา และลดผลต่างๆที่เกิดจากข้อจำกัดของมนุษย์

#### 3.3 วัตถุประสงค์ของโครงการ

เนื่องจากระบบงานทะเบียนมีความต้องการใช้ระบบคอมพิวเตอร์ เพื่อนำมาช่วยงานในระบบต่างๆที่เกี่ยวข้องกับทะเบียนประวัติ และการลงทะเบียนของนักศึกษา โดยที่ทุกระบบต้องมีการติดต่อเป็นแบบ GUI (Graphical User Interface) นอกจากนี้แก้ปัญหาที่พบได้ในการดำเนินการในปัจจุบันและเพิ่มประสิทธิภาพของระบบย่อยๆแล้ว ระบบฐานข้อมูลที่เหมาะสมของงานทะเบียนมีความจำเป็นต้องได้รับการพัฒนาขึ้นมาเป็นระบบสารสนเทศเพื่อการจัดการหรือระบบทะเบียนนักศึกษาเพื่อการจัดการการศึกษาให้สอดคล้องกับงานของฝ่ายต่างๆภายในสำนักงานที่ต้องเกี่ยวข้องกับงานทะเบียนประวัตินักศึกษา และการลงทะเบียนนักศึกษาโดยมีวัตถุประสงค์ที่สำคัญคือ

1. เพื่อช่วยให้ผู้บริหารและผู้ปฏิบัติงานเข้าใจกระบวนการปฏิบัติงานได้ดีขึ้น และสามารถเพิ่มประสิทธิภาพให้กระบวนการปฏิบัติงานที่รับผิดชอบให้มีความกระชับและคล่องตัวขึ้น
2. เพื่อลดขั้นตอนกระบวนการปฏิบัติงานประจำวัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เพื่อจัดเตรียมองค์กรให้มีขีดความสามารถในการสร้างงานและรองรับงานทะเบียน ประวัตินักศึกษาและการลงทะเบียนเรียนของนักศึกษาที่จะขยายตัวเพิ่มขึ้นในอนาคต

### 3.4 สิ่งที่ได้จากการนำระบบงานสารสนเทศเพื่อการจัดการมาใช้

การออกแบบระบบงานสารสนเทศเพื่อการจัดการ เป็นการศึกษารวบรวมข้อมูลจากเอกสารที่เกี่ยวข้อง และการสอบถามความต้องการของผู้ใช้ระบบ ระบบงานสารสนเทศมีวัตถุประสงค์ในการออกแบบเพื่อนำมาใช้ในการแบ่งเบาภาระ และอำนวยความสะดวกในการทำงานของผู้ใช้เป็นหลัก เพื่อการจัดการข้อมูล หรือเอกสารที่เกี่ยวข้องกับนักศึกษา และเพื่อเพิ่มประสิทธิภาพในการทำงาน การจัดเก็บข้อมูล การแก้ไขข้อมูล การค้นหาข้อมูล การจัดพิมพ์เอกสารให้เป็นไปอย่างรวดเร็วและถูกต้อง อีกทั้งยังช่วยในการลดขั้นตอนต่างๆ อาทิ ในการคำนวณคะแนนเฉลี่ยสะสม ในการค้นหาวิชาเรียนที่นักศึกษาลงทะเบียน เพื่อนำมาใส่เอกสารทางการศึกษา เป็นต้น ทำให้งานทะเบียนสามารถตอบสนองความต้องการในด้านการบริการแก่นักศึกษาได้เป็นอย่างดี

### 3.5 หน้าที่หลักของระบบสารสนเทศ

หน้าที่หลักของระบบสารสนเทศ เพื่อเป็นเครื่องมือ ในการจัดการข้อมูล บนระบบฐานข้อมูล และประมวลผลความต้องการที่จะทำการจัดเก็บรวบรวมไว้ โดยข้อมูลเหล่านี้ เป็นข้อมูลที่เกี่ยวข้องกับระบบบริหารงานวิทยาลัย เพื่อเป็นฐานข้อมูลที่ทันต่อความต้องการ และพัฒนาต่อไปในอนาคต

### 3.6 ความต้องการของผู้ใช้

ณ ปัจจุบันระบบงานเดิมที่เป็นอยู่ ไม่สามารถรองรับการทำงานที่จะขยายต่อไปในอนาคตได้ และโปรแกรมที่ใช้งานบางส่วนมีปัญหา ไม่สามารถทำงานได้ตามต้องการหรือเกิด Error ขึ้น ซึ่งระบบงานนี้ไม่มีการพัฒนางานเพิ่มเติม จึงจำเป็นต้องหา Solution ที่เข้ามารองรับ ฐานข้อมูลที่เป็นอยู่ในปัจจุบัน และในการทำงาน ดังนั้นจึงเล็งเห็นว่า Relational Database สามารถเข้ามารองรับการทำงานนี้ได้

### 3.7 ข้อมูลและสารสนเทศที่เป็นความต้องการขององค์กร

ความต้องการของระบบข้อมูลสารสนเทศเพื่อการจัดการ (Management Information System) หรือระบบบริหารงานวิทยาลัย เพื่อการจัดการ การศึกษาและของผู้ใช้ฝ่ายต่าง ๆ ที่เกี่ยวข้องประกอบด้วย

#### 3.7.1 ส่งเสริมการศึกษา

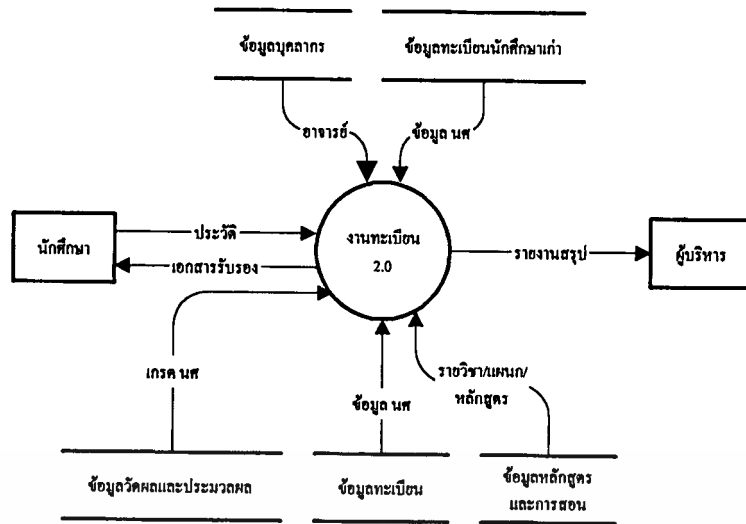
##### 3.7.1.1 งานทะเบียน

- กำหนดฝ่ายต่าง ๆ
  1. กำหนดวุฒิการศึกษา
  2. ประเภท/แผนกวิชา

- ประวัตินักศึกษา

เอกสารนี้เป็นเอกสารที่แจ้งให้บริการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. บันทึก
  2. ค้นหา
  3. เปลี่ยนแปลงชั้นเรียน
  4. พิมพ์ใบรับรอง
  5. กำหนดอาจารย์ที่ปรึกษา
  6. สรุปจำนวนนักศึกษา
  7. สรุปจำนวนจำแนกตามภูมิสำเนา
  8. ตรวจสอบสถานศึกษาเดิม
- จัดการทั่วไป
    1. แสดงผลสอบรายวิชา
    2. แสดงผลการสอบรายบุคคล
    3. พิมพ์ผลการสอบรายบุคคล
    4. การเลื่อนชั้นเรียน
    5. แสดงรายชื่อนักศึกษาที่คะแนนสะสมต่ำกว่าเกณฑ์
    6. การเปลี่ยนแปลงสาขาวิชาเรียน
  - สถานภาพนักศึกษา
    1. การกำหนดการฟื้นสถานภาพนักศึกษา
    2. ดูข้อมูลนักศึกษาที่ฟื้นสภาพ
    3. การคืนสถานภาพนักศึกษา
    4. สรุปจำนวนนักศึกษาที่ฟื้นสภาพ
  - นักศึกษาที่จบการศึกษา
    1. กำหนดรายชื่อนักศึกษาที่จบการศึกษา
    2. พิมพ์ใบรบ.1
    3. สร้างแฟ้มสำรองข้อมูลนักศึกษาที่จบ
    4. ลบข้อมูลนักศึกษาที่จบออก
  - ข้อมูลเดิมของนักศึกษาที่จบแล้ว
    1. ทะเบียนประวัติ
    2. ผลการเรียนรายบุคคล
    3. สรุปจำนวนนักศึกษาที่จบในรุ่นนั้น ๆ

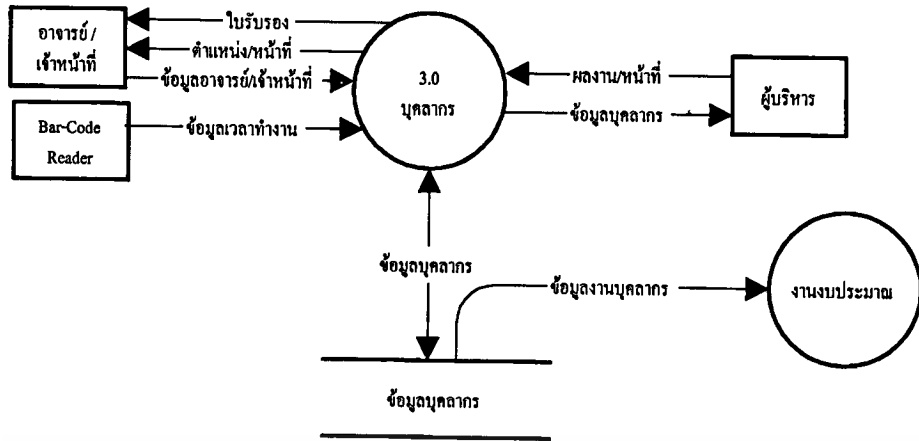


รูปที่ 3.1 ดาต้าโฟลว์สำหรับงานทะเบียน

### 3.7.1.2 งานบุคลากร

- กำหนดฝ่าย/แผนก
- บันทึกข้อมูลบุคลากร
- ค้นหาข้อมูลบุคลากร
- การมาทำงาน
  1. บันทึกประจำวัน
  2. สรุปการมาทำงานตามรายการ
  3. สรุปการมาทำงานรายบุคคล
- สรุปจำนวนบุคลากร
- สรุปจำนวนจำแนกตามระดับชั้น
- สรุปจำนวนจำแนกตามวุฒิการศึกษา
- พิมพ์ประวัติบุคคล
  1. ประวัติบุคคล
  2. ประวัติการปรับเงินเดือน
  3. ประวัติการทำงาน
  4. ประวัติการอบรม
  5. ประวัติการรับเครื่องราชอิสริยาภรณ์
  6. ประวัติการศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 คาด้าไฟล์สำหรับงานบุคลากร

### 3.7.1.3 งานพัสดุ

- บันทึกข้อมูลการสั่งซื้อ
- ค้นหาข้อมูลการสั่งซื้อ
  1. ลำดับที่
  2. เลขที่คำสั่งซื้อ
  3. บุคคลหรือหน่วยงานที่เป็นเจ้าของ
  4. สถานที่เก็บ
- พิมพ์รายงานข้อมูลของพัสดุที่ต้องการ

### 3.7.1.4 งานชำระค่าเทอม

- บันทึกข้อมูลการลงทะเบียนของนักศึกษา เช่น จำนวนเงิน เล่มที่ใบเสร็จ เลขที่ใบเสร็จ
- สรุปรยอดการลงทะเบียนรายวัน รายสัปดาห์ รายเดือน รายปี
- รายงานนักศึกษาที่ยังไม่ได้ชำระเงินค่าลงทะเบียน
- รายงานการชำระค่าลงทะเบียนจำแนกตามห้องของนักศึกษา
- รายงานการชำระค่าลงทะเบียนจำแนกตามเล่มที่และเลขที่ใบเสร็จ
- รายงานประวัติการชำระค่าลงทะเบียนรายบุคคล

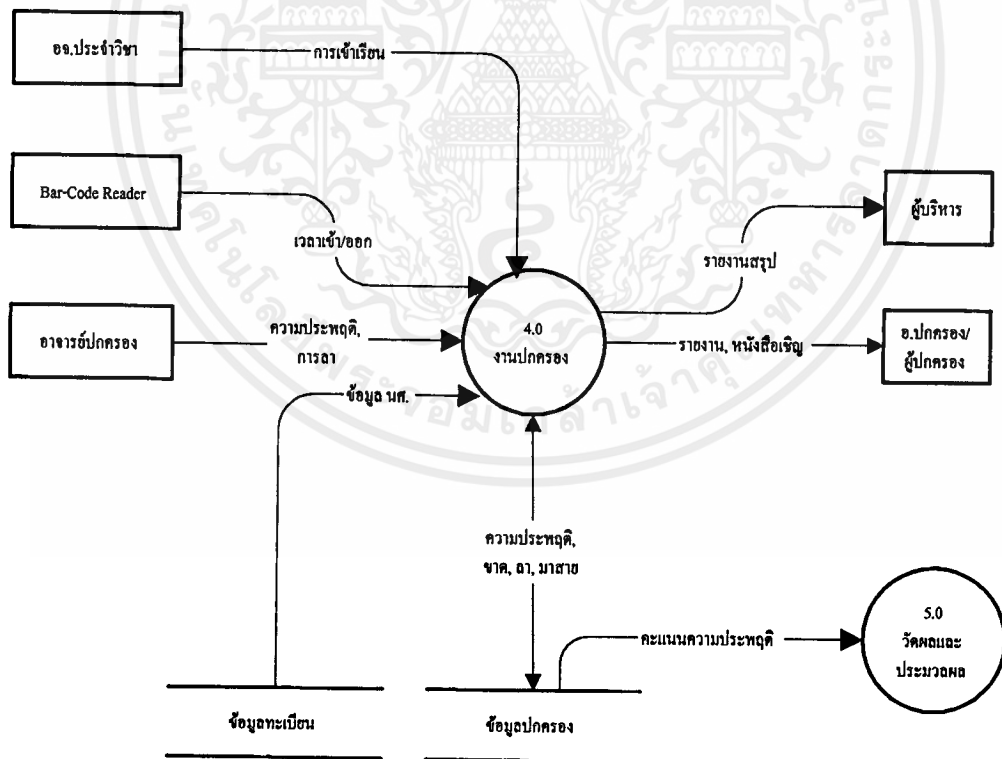
## 3.7.2. ฝ่ายกิจกรรมนักศึกษา

### 3.7.2.1 งานปกครอง

- บันทึกข้อมูลความคิดเห็นความชอบ
- พิมพ์รายงานการบันทึกข้อมูลความคิดเห็นความชอบ
- แสดงประวัติการทำความคิดเห็นความชอบของนักศึกษารายบุคคล
- ค้นหาข้อมูลการทำความคิดเห็นความชอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. พิมพ์รายงานประวัติการทำควมดีความชอบของนักศึกษารายบุคคล
2. ลบข้อมูลการทำควมดีความชอบ โดยระบุเทอมและปีการศึกษาที่ต้องการ
3. บันทึกข้อมูลการทำควมดี
4. พิมพ์รายงานการบันทึกข้อมูลควมดี
5. แสดงประวัติการทำควมดีของนักศึกษารายบุคคล
6. ค้นหาข้อมูลการทำควมดี
7. พิมพ์รายงานประวัติการทำควมดีของนักศึกษารายบุคคล
8. ลบข้อมูลการทำควมดี โดยระบุเทอมและปีการศึกษาที่ต้องการ
9. บันทึกข้อมูลการมาเรียนของนักศึกษา
10. พิมพ์รายชื่อนักศึกษาที่มาเรียนตามเวลาที่ระบุ
11. แสดงประวัติการมาเรียนของนักศึกษารายบุคคล
12. ค้นหาและพิมพ์ข้อมูลการมาเรียน
13. พิมพ์ประวัติการมาเรียนของนักศึกษารายบุคคล
14. ลบข้อมูลการมาเรียนของนักศึกษา โดยระบุภาคและปีการศึกษา
15. สรุปจำนวนวันลา-ขาด จำแนกตามชั้นเรียน



รูปที่ 3.3 คำศัพท์สำหรับงานปกครอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.7.3. ฝ่ายวางแผนและพัฒนา

#### 3.7.3.1 งานงบประมาณ

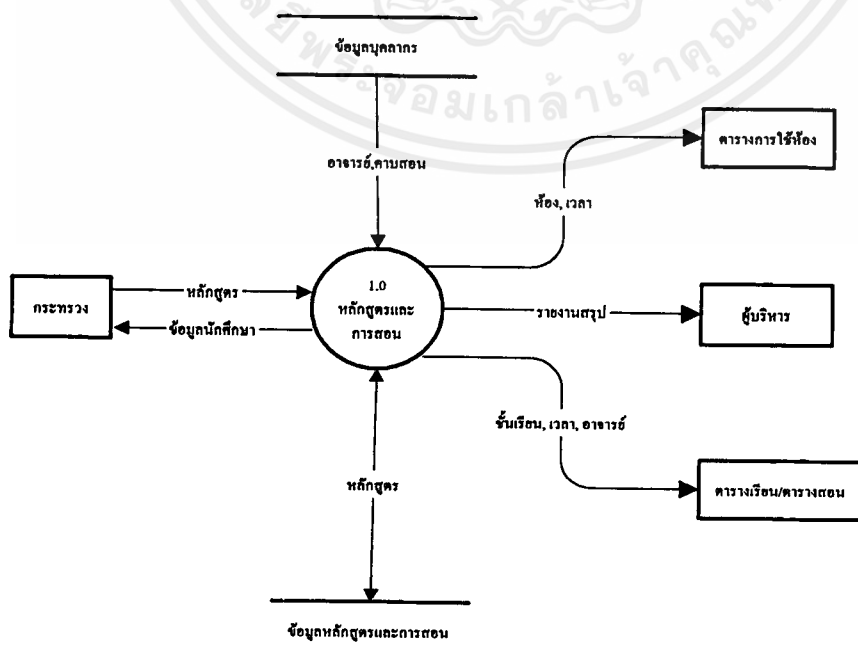
- บันทึกข้อมูลต่าง ๆ ของงบประมาณ เช่น
  1. เลขที่งบประมาณ
  2. ชนิดของงบประมาณ
  3. จำนวนงบประมาณ
  4. จุดประสงค์ของงบประมาณ
- ค้นหาข้อมูลของงบประมาณต่าง ๆ ที่ได้บันทึกไว้แล้วได้โดยสะดวก โดยการค้นหาสามารถใช้รายการต่าง ๆ ที่ได้บันทึกไว้แล้วเป็นข้อกำหนดในการค้นหาข้อมูล เช่น
  1. ค้นหาตามเลขที่งบประมาณ
  2. ค้นหาข้อมูลตามชื่อของงบประมาณ
  3. ค้นหาข้อมูลตามชื่อหรือหน่วยงานที่ได้รับงบประมาณไป
- สามารถสรุปข้อมูลของงบประมาณที่ได้บันทึกไว้แล้วเพื่อที่จะทำให้สามารถทราบว่า
  1. จำนวนงบประมาณที่มีอยู่
  2. จำนวนเงินงบประมาณทั้งหมด
 ซึ่งการสรุปข้อมูลสามารถแสดงได้ทั้งในรูปแบบตารางและแผนภูมิ
- พิมพ์รายงานต่าง ๆ
  1. รายงานข้อมูลของแต่ละงบประมาณ
  2. รายงานงบประมาณจำแนกตาม
    - 2.1 ชนิดของงบประมาณ
    - 2.2 ผู้อนุมัติจ่าย
    - 2.3 ผู้รับเงินงบประมาณนั้น ๆ
  3. รายงานสรุปงบประมาณในรูปแบบตาราง
  4. รายงานสรุปงบประมาณในรูปแบบแผนภูมิ
- บันทึกข้อมูลต่าง ๆ ของโครงการในแต่ละงบประมาณได้ เช่น
  1. เลขที่โครงการ, ชื่อโครงการ
  2. ผู้รับผิดชอบโครงการ
  3. จำนวนเงินงบประมาณที่ใช้
  4. จุดประสงค์ของโครงการ
  5. ระยะเวลาเริ่มต้น, เวลาสิ้นสุดโครงการ
- สรุปจำนวนโครงการในแต่ละงบประมาณได้ เพื่อที่จะทำให้สามารถทราบ
  1. จำนวนโครงการในแต่ละงบประมาณ
  2. จำนวนเงินที่ใช้ในแต่ละโครงการ
  3. จำนวนเงินที่เหลือหลังจากที่โครงการทั้งหมดได้ใช้ไปแล้ว

- มีรายงานเพื่อแสดงรายละเอียดของโครงการต่าง ๆ เช่น
  1. รายงานแสดงโครงการทั้งหมดของแต่ละงบประมาณ
  2. รายงานแสดงโครงการทั้งหมดโดยจำแนกตามงบประมาณ
  3. รายงานแสดงโครงการตามที่มาของข้อมูล เช่น
    - 3.1 ข้อมูลโครงการของผู้รับผิดชอบ
    - 3.2 ข้อมูลโครงการในระยะเวลาที่เริ่มต้นหรือสิ้นสุดที่ต้องการทราบ

### 3.7.4. ฝ่ายวิชาการ

#### 3.7.4.1 งานหลักสูตรและการสอน

- กำหนดรายชื้อวิชา
- กำหนดเวลาคาบเรียน
- กำหนดแผนการเรียน
- กำหนดชื่ออาคารและห้องเรียน
- กำหนดวิชาเรียนแยกตามชั้นปี/กลุ่ม
- จัดทำตารางสอน
- กำหนดวิชาเรียนรายบุคคล
- ตารางสอนส่วนอาจารย์
- พิมพ์แบบฟอร์มบันทึกคะแนนดิบ
- ตารางสอนส่วนนักศึกษา
- ตารางสอนตามห้องเรียน

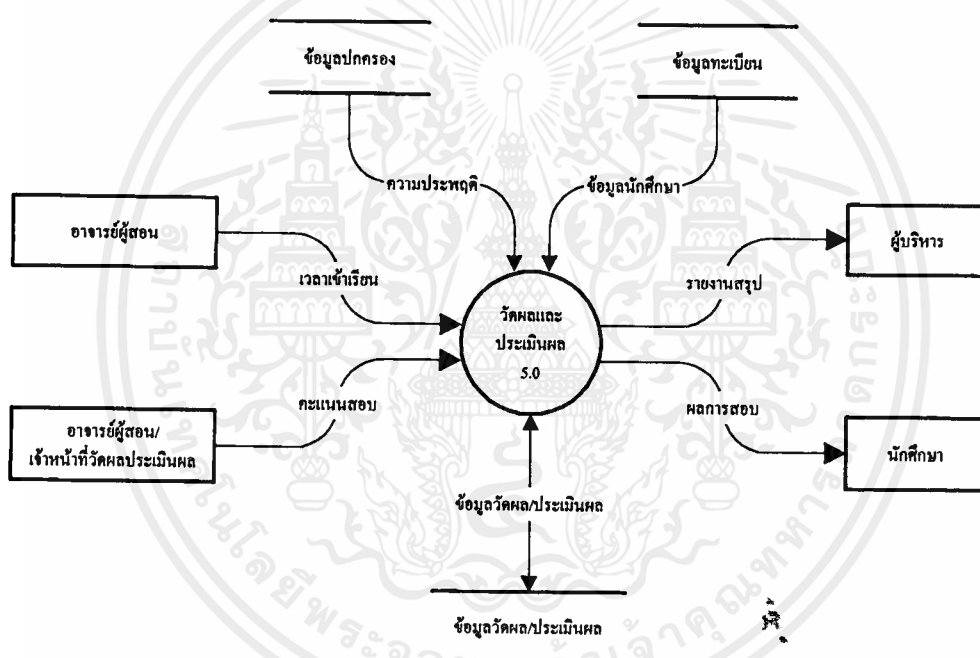


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### รูปที่ 3.4 คาด้าไฟล์สำหรับงานหลักสูตรและการสอน

#### 3.7.4.2 งานวัดผลและประเมินผล

- บันทึกคะแนนสอบรายวิชา
- แสดงผลสอบรายวิชา
- คำนวณเกรดรายวิชา
- แสดงผลสอบรายบุคคล
- บันทึกเกียรยวิชา
- พิมพ์รายงานผลสอบรายบุคคล
- แสดงผลสอบวิชากิจกรรม
- พิมพ์รายงานผลสอบรายชั้นเรียน



รูปที่ 3.5 คาด้าไฟล์สำหรับงานวัดผลและประเมินผล

#### 3.7.4.3 งานห้องสมุด

- บันทึกข้อมูลหนังสือ
- บันทึกการยืม/คืนหนังสือ
  1. บันทึกการยืม
  2. บันทึกการคืน
  3. ประวัติการยืม/คืนหนังสือ
- บันทึกข้อมูลทั่วไป
  1. หมวดหมู่หนังสือ
  2. ประเภทหนังสือ

3. สำนักพิมพ์
  4. แต่ง/ผู้แปล
- ค้นหาหนังสือตามส่วนที่สนใจ เช่น ตามหมวดหมู่หนังสือ ชื่อหนังสือ ผู้แต่ง สำนักพิมพ์ เป็นต้น
  - สรุปข้อมูล/จำนวนหนังสือ จำแนกตามหมวดหมู่หนังสือและประเภทหนังสือ
  - สรุปการยืม/คืนหนังสือประจำวัน
  - สรุปรายชื่อหนังสือที่เกินกำหนดคืน

#### 3.7.4.4 งานฝ่ายวิชาการ

- บุคลากรที่ปฏิบัติงานฝ่ายวิชาการ ประกอบด้วย
  1. ผู้ช่วยอาจารย์ใหญ่ฝ่ายวิชาการ
  2. หัวหน้าแผนกสามัญ
  3. หัวหน้าแผนกบริหารธุรกิจ
  4. หัวหน้าแผนกช่างยนต์
  5. หัวหน้าแผนกช่างไฟฟ้า
  6. หัวหน้าแผนกช่างฝึกฝีมือพื้นฐาน
  7. หัวหน้าแผนกช่างอิเล็กทรอนิกส์
  8. ครู-อาจารย์ประจำวิชา
  9. บรรณารักษ์ห้องสมุด
- ขอบข่ายงานฝ่ายวิชาการ
  1. การจัดการเรียนการสอนตามหลักสูตร
  2. จัดแผนการเรียนการสอนให้นักเรียน
  3. พิจารณาให้ครู-อาจารย์ ได้สอนวิชาต่าง ๆ ตรงความต้องการ ความถนัดและความสามารถ
  4. จัดตารางเรียน ตารางสอน ที่อำนวยความสะดวกทั้งครู-อาจารย์ และนักเรียน
  5. จัดทำปฏิทินการศึกษา
  6. จัดนิเทศน์การศึกษาภายใน โรงเรียนเพื่อปรับปรุงการสอนให้มี ประสิทธิภาพ จัดระบบการสอนแทน เมื่อขาดครู-อาจารย์ ในวันทำการสอนปกติ
  7. ส่งเสริมการใช้อุปกรณ์การศึกษาที่เป็นประโยชน์ต่อการเรียนการสอนอย่างกว้างขวาง
  8. จัดส่งเสริมการแข่งขันทักษะวิชาชีพ เพื่อพัฒนาคุณภาพการเรียนการสอน
  9. จัดอบรมเสริมสร้างความรู้ทักษะประสบการณ์ให้แก่ครู-อาจารย์เพื่อพัฒนาบุคลากร
  10. จัดส่งเสริมสนับสนุนให้การจัดกิจกรรมเสริมหลักสูตรเพื่อพัฒนาการเรียนการสอน

ทุกด้าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. จัดส่งนักเรียนไปฝึกงานตามสถานประกอบการ และหน่วยงานต่าง ๆ
12. จัดเรียนซ่อมเสริม เพื่อปรับคะแนนให้สูงกว่าเกณฑ์ขั้นต่ำ
13. จัดการวัดผลและการประเมินผลการเรียนการสอน
14. จัดห้องสมุดให้เป็นแหล่งรวมวิทยาการแขนงต่าง ๆ ทุกสาขา เพื่อให้ความรู้ ความบันเทิงใจ ความจริงใจ และเพื่อส่งเสริมการเรียนรู้
15. ตรวจสอบผลการเรียน และจัดเก็บผลการเรียนของนักเรียนทุกระดับชั้นไว้เป็นหลักฐาน
16. จัดเก็บผลการเรียนของนักเรียนทุกระดับชั้นไว้เป็นหลักฐาน และจัดออกหลักฐานการศึกษาให้นักเรียน เช่น ใบรับรองผลการเรียน รบ.1๓ ฯลฯ
17. ศึกษาปัญหาต่าง ๆ ที่เกิดจากการเรียนการสอนเพื่อการแก้ไขปรับปรุง

### 3.8 ข้อเพิ่มเติมของระบบ

1. การออกรายงานต้องสามารถแสดงได้ทั้งบนจอและพิมพ์ออกทางเครื่องพิมพ์
2. ระบบจะต้องรวมขั้นตอนการ Load ข้อมูลจากระบบเดิม เข้าสู่ระบบที่จัดตั้งใหม่
3. ระบบต้องสามารถเพิ่มระบบเสริมตามต้องการได้โดยง่าย
4. ระบบต้องสามารถเก็บข้อมูลได้ทั้งภาษาไทยและภาษาอังกฤษ

### 3.9 ขั้นตอนในการพัฒนาระบบ

ในการปฏิบัติงานจำเป็นต้องมีการวางแผนงานปฏิบัติงาน ซึ่งมีขั้นตอนการปฏิบัติงานดังนี้

1. สรุปขอบเขตของงานที่จะทำ โดยพิจารณาจากความต้องการของ User ( พนักงานที่เกี่ยวข้องกับการทำงานโดยตรง )
2. ศึกษาขั้นตอนการดำเนินงานจริงของฝ่ายต่างๆ ที่เกี่ยวข้อง กับการทำงานของระบบ และเขียนออกมาเป็น Context Diagram และ Data Flow Diagram จนถึง Level สุดท้าย
3. นำ Data Flow Diagram ที่เขียนมา ไปให้ User ทำการ Verify ความถูกต้อง และทำการแก้ไขถ้ามีจุดบกพร่องจาก Data Flow Diagram ที่เขียนขึ้นมาให้ถูกต้องและแน่นอนมากที่สุด
4. นำ Data Flow Diagram ที่ถูกต้องมาเขียนความสัมพันธ์ในรูปแบบของ NIAM จากนั้นทำการ Map จาก NIAM ออกมาให้อยู่ในรูปของ Table และทำการ Tuning Table ที่ Map ออกมาให้เหมาะสมยิ่งขึ้นจากนั้นทำการระบุนิคมของข้อมูล ลักษณะของข้อมูล ขนาดของข้อมูลและตัวอย่างของข้อมูลที่จะเก็บลง Table ให้ชัดเจน
5. นำ Table ที่ออกแบบมาได้มาตรวจสอบความถูกต้อง ด้วยการจำลอง Table ที่ออกแบบมาด้วยการเรียกข้อมูลโดยใช้ภาษา SQL ( SQL Query ) จาก Microsoft Access จากนั้นจึงนำไปให้ User ทำการ Verify ความถูกต้องและครบถ้วนของข้อมูลที่ต้องการจะเก็บและนำมาใช้ภายหลัง และทำการแก้ไขถ้ามีข้อผิดพลาดที่ตรวจพบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

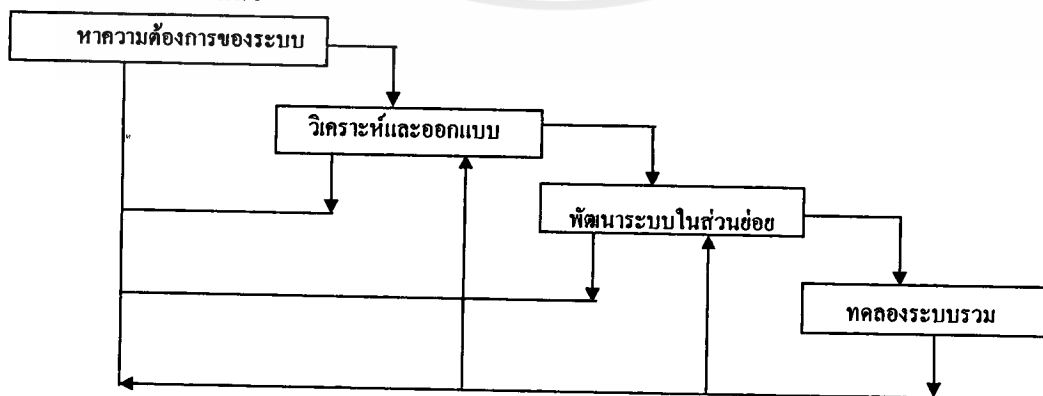
6. ทำการออกแบบหน้าจอ (User Interface) โดยยึดลำดับการทำงานของ User เป็นหลัก ซึ่งในการออกแบบหน้านั้นจะใช้ โปรแกรมที่จะใช้ โปรแกรมที่จะใช้เป็น Tools ในการเขียนโปรแกรม คือ “Borland Delphi”
7. นำหน้าจอไปเสนอต่อ User ให้ User เสนอความต้องการที่อาจแตกต่างไปจากที่ได้ออกแบบมาแล้ว จากนั้นการแก้ไขให้ตรงกับความต้องการของ User มากที่สุดเพื่อความสะดวกในการใช้งานของ User
8. เขียนโปรแกรม เพื่อควบคุมการทำงานของระบบทั้งหมดโดยใช้ภาษา “Borland delphi”
9. ทำการ Implement โปรแกรม ที่เขียนขึ้นมาและนำไปให้ User ทดลองใช้และทำการแก้ไขเมื่อเกิดข้อผิดพลาดขึ้นจนกว่าจะทำงานได้ ตรงตามขอบเขตที่ได้วางเอาไว้ตั้งแต่ต้นและตามความต้องการของ User
10. เขียน Document อธิบายส่วนต่าง ๆ ของระบบและการทำงานทั้งหมดของโปรแกรม
11. เขียนคู่มือการใช้โปรแกรม ให้กับ User เพื่อความสะดวกในการใช้งานต่อไป หลักการและวิธีการในการพัฒนาระบบ

ในการพัฒนาระบบใช้วิธีการออกแบบ 2 วิธี คือ

1. วอเตอร์ฟอลโมเดล (Waterfall Model)
2. โปรโตไทป์ (Prototyping Model)

โดยที่ในส่วนของการออกแบบหน้าจอซึ่งเป็นส่วนที่โปรแกรมจะต้องติดต่อกับผู้ใช้นั้นใช้วิธีโปรโตไทป์โมเดล กล่าวคือจะมีการออกแบบหน้าจอที่มีลักษณะคล้ายกับการใช้งานจริงแล้วนำไปให้ผู้ใช้งาน ทดลองใช้งาน จากนั้นจะนำข้อผิดพลาดและข้อเสนอแนะต่าง ๆ กลับมาแก้ไขที่หน้าจอใหม่ ทำเช่นนี้ไปจนกว่าผู้ใช้งานพอใจ และสามารถใช้งานได้สะดวกกว่าหรืออย่างน้อยต้องเทียบเท่ากับระบบเดิม

ในส่วนของการพัฒนาระบบอื่น ๆ เช่น การหาความต้องการของระบบ (Requirement) การออกแบบซอฟต์แวร์ (Software Design) การทดสอบระบบ (System testing) ใช้การออกแบบแบบวอเตอร์ฟอลโมเดล เพราะในขั้นตอนการทำงานอาจมีการย้อนกลับไปแก้ไขในขั้นตอนก่อน ๆ ที่ได้กระทำเสร็จเรียบร้อยแล้ว เพราะว่าการออกแบบอาจเกิดทำให้มองเห็นปัญหาที่ทำให้ต้องมีการเปลี่ยนแปลงแก้ไขในขั้นตอนที่ได้กระทำมาแล้ว



รูปที่ 3.6 แสดงวอเตอร์ฟอลโมเดล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.10 การเขียนโค้ดโปรแกรม (Program coding)

แบ่งรูปแบบหลักของโปรแกรมในส่วนที่ทำการติดต่อกับฐานข้อมูลได้ 3 ส่วนดังนี้คือ

1. ส่วนที่ใช้ในการเพิ่มข้อมูลลงฐานข้อมูล
2. ส่วนที่ใช้ในการค้นหาข้อมูลจากฐานข้อมูลเพื่อแสดงผลและแก้ไข
3. ส่วนที่ใช้สำหรับการลบข้อมูลออกจากฐานข้อมูล

โดยแต่ละส่วนจะมีทั้งโปรแกรมที่เป็นภาษา SQL และที่เป็นภาษาเคลไฟล์

#### 1. ส่วนที่ใช้ในการเพิ่มข้อมูลลงฐานข้อมูล

SQL Statement ใช้ Stored Procedure ดังนี้

```
Create Procedure ProcName (@Parameter1 Type(Size), @Parameter2 Type(Size), ...)
```

```
Begin
```

```
Insert Into TableName (Column1, Column2, ...)
```

```
Values (@Parameter1, @Parameter2, ...)
```

```
End
```

Delphi Statement

```
Begin
```

```
With DataModule.StoredProc Do
```

```
Begin
```

```
Close;
```

```
Prepare;
```

```
ParamsByName('@Parameter1').AsType := Component1.Text;
```

```
ParamsByName('@Parameter2').AsType := Component2.Text;
```

```
.
```

```
.
```

```
Try
```

```
ExecProc;
```

```
Except
```

```
CheckErrAndShowErrMsg;
```

```
End;
```

```
Close;
```

```
End;
```

```
DataModule.TableName.Refresh;
```

```
CountRec := DataModule.TableName.RecordCount;
```

```
End;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยทำการ Set Property ของ TStoredProc ให้ Stored Procedure Name ก็คือชื่อ Stored Procedure ใน Database Server

## 2. ส่วนที่ใช้ในการค้นหาข้อมูลจากฐานข้อมูลเพื่อแสดงผลและแก้ไข

SQL Statement ใช้ Query Component โดยเขียน SQL โค้ดใน Property ดังนี้

Select \*

From TableName

Where (Column1 = :Parameter1) And (Column2 = :Parameter2) ...

Group By ColumnX

Order By ColomnA

Delphi Statement

Begin

With DataModule.Query Do

Begin

Close;

RequestLive := True; // ถ้ากำหนดให้มีการแก้ไขข้อมูลได้จะต้องกำหนด

CachedUpdates := True; // RequestLive และ CachedUpdates เป็น True

Prepare;

ParamsByName('Parameter1').AsType := Component1.Text;

ParamsByName('Parameter2').AsType := Component2.Text;

.

.

.

Try

Open;

Except

CheckErrAndShowErrMsg;

End;

End;

NextFormName.Show;

NowFormName.Hide;

End;

ให้

หลังจากการแก้ไขข้อมูลต่างๆเรียบร้อยแล้วจะทำการบันทึกสิ่งที่เปลี่ยนแปลงลงฐานข้อมูลจะทำได้โดยเขียนโปรแกรมดังนี้

```
Begin
    DataModule.Query.ApplyUpdates;
    DataModule.Query.Close;
    DataModule.Table.Refresh;
    PrevFormName.Show;
    NowFormName.Close;
End;
```

### 3. ส่วนที่ใช้สำหรับการลบข้อมูลออกจากฐานข้อมูล

SQL Statement ใช้ Stored Procedure ดังนี้

```
Create Procedure ProcName (@Parameter1 Type(Size), @Parameter2 Type(Size), ...)
```

```
Begin
```

```
Delete From TableName
```

```
Where (Column1 = @Parameter1) And (Coloumn2 = @Parameter2) ...
```

```
End
```

Delphi Statement

```
Begin
```

```
With DataModule.StoredProc Do
```

```
Begin
```

```
Close;
```

```
Prepare;
```

```
ParamsByName('@Parameter1').AsType := Component1.Text;
```

```
ParamsByName('@Parameter2').AsType := Component2.Text;
```

```
.
```

```
Try
```

```
ExecProc;
```

```
Except
```

```
CheckErrAndShowErrMsg;
```

```
End;
```

```
DeleteCount := Count;
```

```
End;
```

```
End;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 3.1 ข้อมูลรายวิชา

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Subj_Code	Char	12	-	P.K.	รหัสวิชา
Dept_Code	Char	6	-	P.K.	รหัสแผนก
Credit	Tinyint	1	-	-	หน่วยกิต
Theory	Tinyint	1	-	-	ทฤษฎี
Operate	Tinyint	1	-	-	ปฏิบัติ
Subj_desc	Char	255	-	-	คำอธิบายรายวิชา
Yearth	Char	1	-	-	ปีที่
Semester	Tinyint	1	-	-	ภาคเรียน
Subj_Name	Char	30	-	-	ชื่อวิชา

ตาราง 3.2 ข้อมูลแผนก

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Dept_Code	Char	6	-	P.K.	รหัสแผนก
Department	Char	35	-	-	ชื่อแผนก
Branch	Char	35	-	-	ประเภท(ช่างอุตสาหกรรม)
Course	Char	35	-	-	หลักสูตร(อาชีพ)
Level1	Char	6	-	-	ระดับชั้น(ปวช)

ตาราง 3.3 ข้อมูลตารางสอน

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Day_Of_Week	Char	9	-	P.K.	วันในสัปดาห์
Period_No	Char	2	-	P.K.	เวลา
Room_Code	Char	10	-	P.K.	รหัสห้อง
Subj_Code	Char	12	-	-	รหัสวิชา
Class	Char	6	-	-	ห้องเรียน
Employee_No	Char	10	-	-	รหัสพนักงาน
Dept_Code	Char	6	-	-	รหัสแผนก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 3.4 ข้อมูลอาคารสถานที่

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Room_Code	Char	10	-	P.K.	รหัสห้องเรียน
Bldg	Char	3	-	-	ตึก(อาคาร)
Floor_No	Char	3	-	-	ชั้น
Room_No	Char	6	-	-	หมายเลขห้อง

ตาราง 3.5 ข้อมูลชื่ออาคาร

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Bldg	Char	3	-	P.K.	ตึก(อาคาร)
Bldg_Name	Char	35	-	-	ชื่ออาคาร

ตาราง 3.6 ข้อมูลอาจารย์ประจำชั้น

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Class	Char	6	-	P.K.	นักเรียนห้อง
Employee_No	Char	10	-	-	รหัสอาจารย์

ตาราง 3.7 ข้อมูลคาบวิชา

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Period_No	Char	2	-	P.K.	คาบวิชา
Time1	Char	7	-	-	เวลา

ตาราง 3.8 ข้อมูลนักศึกษา

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
ID	Char	10	-	P.K.	รหัสนักศึกษา
F_Name	Char	12	-	-	คำนำหน้าชื่อ
Name1	Char	35	-	-	ชื่อนักศึกษา
L_Name	Char	35	-	-	นามสกุลนักศึกษา
F_NameE	Char	12	-	-	คำนำหน้าชื่อ(ENG)
NameE	Char	35	-	-	ชื่อ(English)
L_NameE	char	35	-	-	นามสกุล(English)
Sex	Bit	1	-	-	เพศ
Certification	Char	35	-	-	ระดับวุฒิการศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 3.8 ข้อมูลนักศึกษา (ต่อ)

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Am_Pm	Char	5	-	-	นักศึกษารอบ
Address1	Char	50	-	-	ที่อยู่
Disctric	Char	35	-	-	ตำบล
Zone	Char	35	-	-	อำเภอ
Province	Char	35	-	-	จังหวัด
ZipCode	Char	5	-	-	รหัสไปรษณีย์
Tel	Char	20	-	-	เบอร์โทรศัพท์
Religion	Char	16	-	-	ศาสนา
Birth_Yr	Datetime	8	-	-	วันเกิด
Nation	Char	16	-	-	สัญชาติ
Race	Char	16	-	-	เชื้อชาติ
Weight	Real	4	2	-	น้ำหนัก
Hight	Real	4	2	-	ส่วนสูง
Status1	Char	10	-	-	สถานะนักศึกษา
Spacial	Char	35	-	-	ความสามารถพิเศษ
Reward	Char	35	-	-	รางวัลที่เคยได้รับ
Guard	Char	75	-	-	ผู้ปกครอง
Relate	Char	15	-	-	ความสัมพันธ์
Father	Char	75	-	-	บิดา
Mather	Char	75	-	-	มารดา
FMAddress	Char	100	-	-	ที่อยู่พ่อแม่
FMPProvince	Char	35	-	-	จังหวัด
FMZIP	Char	5	-	-	รหัสไปรษณีย์
FMTel	Char	20	-	-	โทรศัพท์
Dept_Code	Char	6	-	-	รหัสแผนก
Class	Char	6	-	-	นักเรียนห้อง
Enter_Date	Datetime	8	-	-	วันเข้าเรียน
Old_Certify	Char	35	-	-	วุฒิการศึกษาเดิม
Old_Credit	Char	20	-	-	หน่วยกิตที่ได้
School	Char	35	-	-	ชื่อโรงเรียนที่จบมา
Ex_Credit	Tinyint	1	-	-	หน่วยกิตยกเว้น
Now_School	Char	35	-	-	สังกัดโรงเรียน

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้บริการไว้ใช้ในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 3.8 ข้อมูลนักศึกษา (ต่อ)

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Blood	Char	2	-	-	กลุ่มเลือด
Disease	Char	35	-	-	โรคประจำตัว
Doc_Note	Char	35	-	-	เอกสารประกอบ
Pay_Note	Char	35	-	-	การจ่ายค่าเล่าเรียน
Old_ID	Char	15	-	-	เลขประจำตัวเดิม
Old_Roll	Char	16	-	-	รบ. ชุดที่เล่นที่
Old_Class	Char	15	-	-	ชั้นเรียนสุดท้าย
Old_GPA	Real	4	-	-	ผลการเรียนที่จบมา
Finish_Date	Datetime	4	-	-	วันสำเร็จการศึกษา
Trans_No	Char	16	-	-	เลขที่ Trans
Roll	Char	16	-	-	เลขที่ รบ
StatusLern	Char	45	-	-	สถานะนักศึกษา
NowYear	Char	4	-	-	ปีปัจจุบัน

ตาราง 3.9 ข้อมูลกำหนดฝ่าย/แผนก

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Employee_No	Char	10	-	P.K.	รหัสนักศึกษา
Department1	Char	100	-	-	แผนก
Position1	Char	100	-	-	ตำแหน่ง
Salary1	Tinyint	1	-	-	เงินเดือน
Date1	Datetime	8	-	-	วัน/เดือน/ปี

ตาราง 3.10 ข้อมูลโรงเรียน

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
School	Char	35	-	P.K.	ชื่อสถานศึกษา
Address1	Char	100	-	-	ที่อยู่
Province	Char	25	-	-	จังหวัด
Tel	Char	25	-	-	โทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 3.11 ข้อมูลประวัติบุคลากร

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Employee_No	Char	10	-	P.K.	รหัสพนักงาน
F_Name	Char	12	-	-	คำนำหน้าชื่อ
Name1	Char	35	-	-	ชื่อ
L_Name	Char	35	-	-	นามสกุล
Age	Tinyint	1	-	-	อายุ
Weight	Real	4	2	-	น้ำหนัก
Hight	Real	4	2	-	ส่วนสูง
Sex	Bit	1	-	-	เพศ
Blood	Char	2	-	-	กรุ๊ปเลือด
Race	Char	16	-	-	เชื้อชาติ
Nation	Char	16	-	-	สัญชาติ
Religion	Char	16	-	-	ศาสนา
Disease	Char	25	-	-	โรคประจำตัว
Birth_Yr	Datetime	8	-	-	วัน/เดือน/ปี เกิด
Birth_Prov	Char	25	-	-	จังหวัดที่เกิด
Home_Id	Char	20	-	-	บ้านเลขที่(ปัจจุบัน)
Land	Char	30	-	-	ตรอก/ซอย
Street	Char	25	-	-	ถนน
District	Char	30	-	-	ตำบล
Zone1	Char	35	-	-	อำเภอ
Province	Char	35	-	-	จังหวัด
ZipCode	Tinyint	1	-	-	รหัสไปรษณีย์
Tel	Char	20	-	-	โทรศัพท์
C_Home_Id	Char	20	-	-	บ้านเลขที่(ทะเบียน)
C_Land	Char	30	-	-	ตรอก/ซอย
C_Street	Char	25	-	-	ถนน
C_District	Char	30	-	-	ตำบล
C_Zone1	Char	35	-	-	อำเภอ
C_Province	Char	35	-	-	จังหวัด
C_ZipCode	Tinyint	1	-	-	รหัสไปรษณีย์
C_Tel	Char	20	-	-	โทรศัพท์
ID_No	Char	10	-	-	บัตรประชาชน

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้กับโรงเรียนเพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้ทำซ้ำโดยไม่ขออนุญาตจากโรงเรียน

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 3.11 ข้อมูลประวัติบุคลากร (ต่อ)

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Issue_Date	Datetime	8	-	-	วัน/เดือน/ปี (ที่ออก)
Expire_Date	Datetime	8	-	-	วัน/เดือน/ปี(สิ้นสุด)
Issue_Zone	Char	35	-	-	ณ อำเภอ
Issue_Prov	Char	35	-	-	จังหวัด
Tax_No	Char	10	-	-	เลขผู้เสียภาษี
Status1	Char	10	-	-	สถานะภาพ
Mar_Name	Char	75	-	-	ชื่อคู่สมรส
Mar_Licence	Char	10	-	-	ทะเบียนสมรส
Mar_Date	Datetime	8	-	-	วัน/เดือน/ปี (ที่จด)
Mar_Zone	Char	35	-	-	ณ อำเภอ
Mar_Prov	Char	35	-	-	จังหวัด
Mar_Job	Char	35	-	-	อาชีพคู่สมรส
Mar_WkCompany	Char	50	-	-	บริษัท
Mar_Tel	Char	20	-	-	โทรศัพท์
Son	Tinyint	1	-	-	บุตรชาย
Daughter	Tinyint	1	-	-	บุตรสาว
Father	Char	75	-	-	บิดา
Father_Status	Char	10	-	-	สถานะภาพ
Fa_Age	Tinyint	1	-	-	อายุ
Fa_Add	Char	150	-	-	ที่อยู่(บิดา)
Fa_Job	Char	50	-	-	อาชีพ(บิดา)
Mather	Char	75	-	-	มารดา
Ma_Status	Char	10	-	-	สถานะภาพ
Ma_Age	Tinyint	1	-	-	อายุ
Ma_Add	Char	150	-	-	ที่อยู่(มารดา)
Ma_Job	Char	50	-	-	อาชีพ(มารดา)
Brother	Tinyint	1	-	-	จำนวนพี่
Sister	Tinyint	1	-	-	จำนวนน้อง
Work_Pos	Char	40	-	-	ตำแหน่งงาน
S_Work	Datetime	8	-	-	วันที่เริ่มทำงาน
TypeOfGrad	Char	30	-	-	การเกณฑ์ทหาร
Grad_Date	Datetime	8	-	-	วันที่ทำการเกณฑ์

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ การใช้งานที่นอกเหนือจากนี้โดยไม่ขออนุญาตหรือได้รับอนุญาตเป็นการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 3.11 ข้อมูลประวัติบุคลากร (ต่อ)

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Graded_Of	Char	50	-	-	สังกัด
Graded_Date	Datetime	8	-	-	วันที่ประจำการ
Memo_Skip	Char	60	-	-	ยกเว้นเพราะ
Document1	Char	20	-	-	เอกสาร
Doc_uncomp	Char	100	-	-	เอกสารที่ไม่ครบ

ตาราง 3.12 ข้อมูลพี่น้อง

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Employee_No	Char	10	-	P.K.	รหัสพนักงาน
Kin_Name	Char	75	-	P.K.	ชื่อ-สกุล(พี่น้อง)
Kin_Company	Char	100	-	-	บริษัทที่ทำงาน
Kin_Pos	Char	50	-	-	ตำแหน่ง

ตาราง 3.13 ข้อมูลประวัติการศึกษา

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Employee_No	Char	10	-	P.K.	รหัสพนักงาน
Institute	Char	200	-	-	ชื่อสถานศึกษา
Certification	Char	60	-	-	วุฒิการศึกษา
Year1	Datetime	8	-	-	ปีที่สำเร็จ
GPA	Real	4	2	-	เกรดเฉลี่ย

ตาราง 3.14 ข้อมูลวุฒิการศึกษา

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Institute	Char	200	-	P.K.	ชื่อสถานศึกษา
Certification	Char	60	-	P.K.	วุฒิการศึกษา
CLevel	Char	50	-	-	ระดับการศึกษา

ตาราง 3.15 ข้อมูลประวัติการทำงาน

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Employee_No	Char	10	-	P.K.	รหัสพนักงาน
Company	Char	35	-	P.K.	บริษัท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 3.15 ข้อมูลประวัติการทำงาน (ต่อ)

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Position1	Char	30	-	-	ตำแหน่ง
Salary	Money	8	-	-	เงินเดือน
Total_Date1	Char	30	-	-	เวลาที่ทำงาน

ตาราง 3.16 ข้อมูลประวัติการอบรม

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Employee_No	Char	10	-	P.K.	รหัสพนักงาน
Semina_Subj	Char	35	-	-	หัวข้อการอบรม
Semina_Org	Char	35	-	-	หน่วยงานที่จัด
Semina_Dept	Char	35	-	-	สถานที่อบรม
Start_Date1	Datetime	8	-	P.K.	วัน/เดือน/ปี(เริ่ม)
Finish_Date1	Datetime	8	-	-	วัน/เดือน/ปี(สิ้นสุด)

ตาราง 3.17 ข้อมูลสถานที่ทำงาน

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Company	Char	100	-	P.K.	บริษัท
Comp_Add	Char	255	-	-	ที่อยู่
Comp_Type	Char	100	-	-	ลักษณะงานที่ทำ
Comp_Tel	Char	25	-	-	เบอร์โทรศัพท์
Comp_Fax	Char	25	-	-	เบอร์แฟก

ตาราง 3.18 ข้อมูลการมาทำงาน

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Card_ID	Char	10	-	P.K.	รหัสบัตรประจำตัว
Date1	Datetime	8	-	P.K.	วัน/เดือน/ปี
Time1	Datetime	8	-	P.K.	เวลา
TypeOfDonkey	Char	20	-	-	ประเภทการลา
Remark	Char	35	-	-	หมายเหตุ
User_ID	Char	10	-	-	ผู้บันทึก
Time_Stamp	Datetime	8	-	-	เวลาบันทึก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 3.19 ข้อมูลความประพฤติ

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
ID	Char	10	-	P.K.	รหัสนักศึกษา
Date1	Datetime	8	-	P.K.	วัน/เดือน/ปี
Time1	Datetime	8	-	P.K.	เวลา
Err_Code	Char	10	-	-	รหัสความผิด
Memo	Char	255	-	-	หมายเหตุ

ตาราง 3.20 ข้อมูลความผิด

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Err_Code	Char	10	-	P.K.	รหัสความผิด
Err_Name	Char	35	-	-	ประเภทความผิด
Err_Point	Tinyint	1	-	-	จำนวนคะแนน

ตาราง 3.21 ข้อมูลเวลาเรียน

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
ID	Char	10	-	P.K.	รหัสนักศึกษา
Subj_Code	Char	12	-	P.K.	รหัสวิชา
Period_Std	Tinyint	1	-	-	คาบเวลา

ตาราง 3.22 ข้อมูลคะแนน

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
ID	Char	10	-	P.K.	รหัสนักศึกษา
Semeter	Tinyint	1	-	P.K.	ภาคการศึกษา
Year1	Char	4	-	P.K.	ปีการศึกษา
Subj_Code	Char	12	-	P.K.	รหัสวิชาเรียน
Dept_Code	Char	6	-	-	รหัสแผนก
Point	Tinyint	1	-	-	คะแนน
Grade	Char	2	-	-	เกรด
Err_Point	Tinyint	1	-	-	คะแนนความประพฤติ
Remark	Char	30	-	-	หมายเหตุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 3.23 ข้อมูลประวัติการฝึกงาน

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
ID	Char	10	-	P.K.	รหัสนักศึกษา
Company	Char	100	-	-	บริษัท
Start_Date	Datetime	8	-	-	เริ่มฝึกงาน
Finish_Date	Datetime	8	-	-	สิ้นสุดการฝึกงาน
Prac_Result	Char	30	-	-	ผลการฝึกงาน

ตาราง 3.24 ข้อมูลเงินเดือน

COLUMNNAME	TYPE	SIZE	SCALE	KEY	DETAIL
Employee_No	Char	10	-	P.K.	รหัสพนักงาน
Year1	Datetime	8	-	P.K.	วัน/เดือน/ปี(เริ่ม)
Salary	Money	8	-	-	เงินเดือน

### 3.11 สรุปปัญหาและการดำเนินงาน

#### 3.11.1 User

- ปัญหา 1. การติดต่อระหว่าง User กับทีมงาน ลำบากเพราะระยะทางไกลและติดต่อกับผู้รับผิดชอบไม่ค่อยได้
2. User เปลี่ยนแปลงความต้องการบ่อยครั้งมากเป็นผลให้การออกแบบโครงสร้างข้อมูล (Table) และการ Coding Program ลำบาก

#### 3.11.2 Network

- ปัญหา 1. อุปกรณ์ไม่พร้อมในการติดตั้ง เช่น Hub, PC, LAN Card
2. ไอพี แอดเดรส (IP Address) มีการซ้ำกันกับไอพี แอดเดรส (IP Address) ในภาค ซึ่งสาเหตุมาจาก Network ในภาควิศวกรรมคอมพิวเตอร์ในระยะนั้นมีการแจกจ่ายไอพี แอดเดรส (IP Address) โดยดีเอชซีพีเซิร์ฟเวอร์ (DHCP Server) ซึ่งทางเราไม่ทราบ และเมื่อกำหนดไอพี แอดเดรส (IP Address) ลงไปในเครื่องที่ทำ Project จะมีการซ้ำกันของไอพี แอดเดรส (IP Address) ในบางครั้ง
3. การติดต่อระหว่างไคลเอ็นท์/เซิร์ฟเวอร์ (Client/Server) เกิดปัญหาคือ ขณะใช้งานเฉพาะในกลุ่มของโปรเจก (Project) ไม่มีการเชื่อมต่อกับเน็ตเวิร์ค (Network) ของ ภาค จะใช้งานได้ปกติ แต่เมื่อมีการเชื่อมต่อ Network ของ กลุ่มโปรเจก (Project) เข้ากับเน็ตเวิร์ค (Network) ของภาควิศวกรรมคอมพิวเตอร์ ในขณะที่มีผู้ใช้ (User) ใช้งานมาก ทำให้การติดต่อระหว่างไคลเอ็นท์/เซิร์ฟเวอร์ (Client/Server) ไม่สามารถติดต่อได้

- การแก้ไข 1. จัดหา Hub, PC, LAN Card บางส่วนมาเอง
2. Set IP Address ไปนอกช่วง IP Address ของ DHCP Server
3. ไม่เชื่อมต่อกับ Network ของภาควิชาวิศวกรรมคอมพิวเตอร์

### 3.11.3 Development tools Delphi 3 Client/Server

- ปัญหา 1. Software ที่นำมาใช้งานในครั้งแรกเป็น Version 3 Client/Server Confidential ซึ่งมีปัญหา คือ เมื่อ Application ที่พัฒนาขึ้นมา มีการ run และ connect กับ Data Base Server แล้ว cursor mouse จะขึ้นเป็นสัญลักษณ์ นาฬิกาทราย SQL และค้าง ไม่สามารถใช้ Mouse ได้
2. ใช้ DB Grid ร่วมกับ Ttable Component เมื่อมีการนำข้อมูลจากฐานข้อมูลมาแสดงผลใน DB Grid และมีการ Scroll ข้อมูลที่หน้าจอ จะมีการซ้ำกันของข้อมูลที่แสดงผล แต่ในฐานข้อมูลยังคงถูกต้องเหมือนเดิม
3. การนำรูปภาพมาแสดงผล ในชุดของ Delphi 3 Client/Server Suit จะมี Component สำหรับ แสดงรูปภาพประเภท JPEG มาให้ ซึ่ง Component ตัวนี้ไม่สามารถใช้งานแบบ Run Time ได้

- การแก้ไข 1. หา Delphi 3 Client/Server Suit ชุดสมบูรณ์มาใช้งานแทน
2. เปลี่ยนแนวทาง โดยใช้ DB Grid ร่วมกับ Stored Procedure ที่มีการ Select ข้อมูลขึ้นมาทั้งหมดแทน
3. หา Component ตัวอื่นมาใช้งานแทน

### 3.11.4 Windows NT 4.0 Server Thai

- ปัญหา 1. การที่ Windows NT ไม่เห็น Network เนื่องจาก Network Interface Card ไม่มี Software ที่ใช้ในการเซ็ทค่าต่างบน Card
- การแก้ไข 1. Download Software สำหรับเซ็ท Card และ Driver มาใช้งาน และทำการเซ็ท Config ของ Card ให้เหมาะสม

# บทที่ 4

## การทดลองและผลการทดลอง

### 4.1 ฮาร์ดแวร์และซอฟต์แวร์ที่ใช้ในการทดลอง

#### 4.1.1 คอมพิวเตอร์ที่ทำหน้าที่ควบคุมระบบเครือข่าย (Server)

1. มีหน่วยประมวลผลกลาง เพนเทียม 166 เมกะเฮิร์ตขึ้นไป
2. มีหน่วยความจำหลัก ขนาดไม่ต่ำกว่า 32 MB
3. มีหน่วยความจำชนิดความเร็วสูงภายใน CPU ขนาดไม่ต่ำกว่า 16 KB และภายนอก CPU ขนาดไม่น้อยกว่า 512 KB
4. เครื่องอ่าน CD-ROM
5. Floppy Disk Drive ขนาด 3.5 นิ้ว ความจุไม่ต่ำกว่า 1.44 MB จำนวน 1 หน่วย
6. จอภาพ ขนาด 14 นิ้ว
7. แป้นพิมพ์ (Key Board) ที่มีอักษรภาษาไทย/อังกฤษ ตัวเลขและเครื่องหมาย สัญลักษณ์พิเศษอย่างน้อย 101 คีย์
8. มีเมาส์ที่มี 2 ปุ่ม จำนวน 1 ชุด

#### 4.1.2 เครื่องไมโครคอมพิวเตอร์ที่ทำหน้าที่เป็นเทอร์มินอลในระบบเครือข่าย (Client)

1. มีหน่วยประมวลผลกลาง เพนเทียม 100 เมกะเฮิร์ตขึ้นไป
2. มีหน่วยความจำหลัก 16 MB เป็นอย่างต่ำ
3. Floppy Disk Drive ขนาด 3.5 นิ้ว ความจุไม่ต่ำกว่า 1.44 MB จำนวน 1 หน่วย
4. จอภาพ ขนาด 14 นิ้ว
5. แป้นพิมพ์ (Key Board) ที่มีอักษรภาษาไทย/อังกฤษ ตัวเลขและเครื่องหมาย สัญลักษณ์พิเศษอย่างน้อย 101 คีย์
6. มีเมาส์ที่มี 2 ปุ่ม จำนวน 1 ชุด
7. เครื่องพิมพ์ HP Laser Jet จำนวน 1 เครื่อง

### 4.2 ความต้องการทางด้านซอฟต์แวร์

#### 4.2.1 เครื่องที่ทำหน้าที่เป็นเซิร์ฟเวอร์

1. ระบบปฏิบัติการ ไมโครซอฟท์วินโดวส์เอ็นที เวอร์ชัน 4.0 (Microsoft Windows NT Version 4.0)
2. โปรแกรมระบบจัดการฐานข้อมูล ไมโครซอฟท์วินโดวส์ เอสคิวแอลเซิร์ฟเวอร์ เวอร์ชัน 6.5 (Microsoft SQL Server Version 6.5)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.3 การทดลองและผลการทดลอง

#### การทดลองที่ 1 การบันทึกและการอ่านข้อมูลกับฐานข้อมูล

1. รันโปรแกรม แล้วเข้าสู่หน้าจอการกำหนดข้อมูล
2. คลิกเลือกการกำหนดวิชาเรียน เมื่อกรอกข้อมูลเข้าแล้วกดปุ่มตกลงเพื่อบันทึกข้อมูลที่กรอก ลงฐานข้อมูล

หลักสูตรและการสอน

แผนก

รหัสแผนก 012222    แผนก ช่างเชื่อม    ระดับ ปวส

ประเภท ช่างอุตสาหกรรม    หลักสูตร ราชมงคล 2541

ชื่อวิชา

รหัสวิชา 777-1999    วิชา Electronic II

หน่วยกิต 2    ทฤษฎี 2    ปฏิบัติ 0    ปีที่ 3    ภาคเรียนที่ 2

คำอธิบายรายวิชา ต้องผ่านวิชา Electronic I มาก่อน

รหัสวิชา	รหัสแผนก	ชื่อวิชา
111-1112	013811	Japan
111-4577	012561	English
117-1155	013812	fhfh
119-4146	013812	Wing Back

Bank Problem    หน่วยกิต 2

จำนวนข้อมูล 12 : Record

บันทึก    แก้ไข    ลบ    ยกเลิก

รูปที่ 4.1 แสดงการรับข้อมูล

3. สคอตล์อัฟ-ดาวน์ เพื่อดูข้อมูล ถ้ามีข้อมูลแสดงว่ามีกรบันทึกลงจริง และที่แสดงขึ้นมาให้เห็นนั้น คืออ่านข้อมูลมาจากฐานข้อมูล

**หลักสูตรและการสอน**

แผนก

รหัสแผนก 012222    แผนก    ระดับ    ประเภท    หลักสูตร

ชื่อวิชา

รหัสวิชา    วิชา    หน่วยกิต    ทฤษฎี    ปฏิบัติ    ปีที่    ภาคเรียนที่

คำอธิบายรายวิชา

รหัสวิชา	รหัสแผนก	ชื่อวิชา
333-3333	013811	Project
777-1999	012222	Electronic II
789-1246	023850	บัณฑิต
961-1123	012561	High Risk

ต้องผ่านวิชา Electronic I มาก่อน    หน่วยกิต 2

จำนวนข้อมูล 13 : Record

บันทึก     แก้ไข     ลบ     ยกเลิก

รูปที่ 4.2 แสดงข้อมูลที่อ่านจากฐานข้อมูล

ผลการทดลอง สามารถรับข้อมูลเข้าไปและอ่านข้อมูลจากฐานข้อมูลได้

การทดลองที่ 2 การแก้ไขข้อมูลเดิม

1. เลือกข้อมูลที่ต้องการ กดปุ่มแก้ไขดังรูป

**แก้ไข\_กำหนดรายชื่อวิชา**

รหัสแผนก 012222    แผนก ช่างเชื่อม

รหัสวิชา 777-1999    วิชา Electronic II

หน่วยกิต 2    ทฤษฎี 2    ปฏิบัติ 0    ปีที่ 3    ภาคเรียนที่ 2

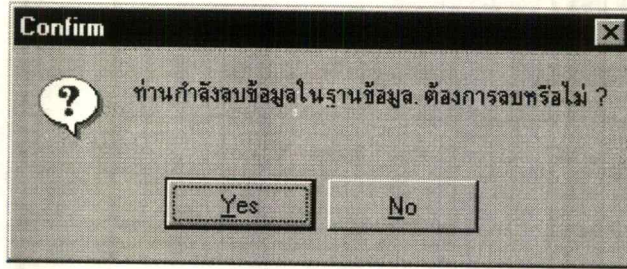
คำอธิบายรายวิชา ต้องผ่านวิชา Electronic I มาก่อน

บันทึก     ยกเลิก

รูปที่ 4.3 การแก้ไขการกำหนดรายชื่อวิชา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. เมื่อเลื่อนดูข้อมูลจะเห็นว่าข้อมูลที่เพิ่งบันทึกนั้นอยู่ในฐานข้อมูล
2. ทำการลบข้อมูลที่อยู่หน้าจอ จะแสดงข้อความยืนยันการลบ



รูปที่ 4.4 ยืนยันการลบข้อมูล

3. กดปุ่ม Yes เพื่อทำการลบข้อมูล
4. ใช้ตัวเลื่อน ๆ ดูข้อมูล ไม่พบข้อมูลในฐานข้อมูล แสดงว่าถูกลบไปแล้ว
5. เมื่อกด No ข้อมูลจะไม่ถูกลบ

ผลการทดลอง สามารถทำการแก้ไขข้อมูลในฐานข้อมูลได้ โดยสามารถ เปลี่ยนแปลง  
ลบข้อมูลได้ซึ่งหน้าจอที่รับข้อมูลทุกหน้าจอจะมีวิธีคล้าย ๆ กัน

## บทที่ 5

# สรุป วิจารณ์ และแนวทางในการพัฒนา

### 5.1 ความสามารถของโปรแกรม

1. สามารถจัดเก็บข้อมูลต่าง ๆ พร้อมทั้งแก้ไขแบบออนไลน์ (Online) ได้
2. ข้อมูลต่าง ๆ จะถูกจัดการโดยผ่านการเชื่อมต่อแบบกราฟฟิก (Graphic) ซึ่งใช้งานได้สะดวกกว่าบนเท็กซ์โหมด (Text Mode)
3. ระบบทำงานบน ระบบเครือข่าย (Network) ซึ่งผู้ที่มีสิทธิ์ในการเข้าถึงข้อมูลได้ ก็สามารถใช้โปรแกรมนี้เรียกใช้ข้อมูลได้ ลดความซ้ำซ้อนของข้อมูล สามารถใช้งานของข้อมูลร่วมกันอย่างมีประสิทธิภาพ รวมถึงความรวดเร็วในการจัดการข้อมูล
4. ความถูกต้องและความน่าเชื่อถือของข้อมูลเพิ่มมากขึ้น
5. ตรวจสอบและค้นหาข้อมูลได้อย่างสะดวกรวดเร็ว
6. สามารถบันทึกข้อมูลและแก้ไขข้อมูลได้

### 5.2 ข้อจำกัดของโปรแกรม

1. เนื่องจากระบบมีความซ้ำซ้อนประกอบกับเวลาที่จำกัดซึ่งอาจทำให้ขั้นตอนการทำงาน รวมถึงหลักการต่าง ๆ คลาดเคลื่อนไปบ้าง
2. ในการใช้งานจริงอาจพบข้อผิดพลาด ในการทำงานของโปรแกรมเนื่องจากยังไม่ได้ทดสอบกับข้อมูลจริง
3. เออร์เรอร์เมสเสจส์ (Error Message) นั้นยังไม่มีการจัดการที่ดีพอ ในบางครั้งเออร์เรอร์เมสเสจส์ ที่แจ้งกับผู้ใช้อาจไม่ตรงกับความต้องการของผู้ใช้ เพราะเป็นการบอกค่าความผิดพลาดโดยรวมเท่านั้น
4. ในส่วนการจัดการรายงานยังไม่สามารถทำงานได้
5. เวลาในการเขียนโปรแกรมมีน้อยมากในการที่จะทำให้ระบบสมบูรณ์ได้ดี
6. ความต้องการของผู้ใช้เปลี่ยนแปลงขณะที่ทำการเขียนโปรแกรมส่วนนั้นๆ ไปแล้ว

### 5.3 แนวทางในการพัฒนาต่อไป

สิ่งต่าง ๆ ที่ต้องปรับปรุงตัว โปรแกรม ให้มีประสิทธิภาพและสมบูรณ์ที่สุดในการใช้งาน รวมไปถึงความพึงพอใจของผู้ใช้งานมีดังนี้คือ

1. ตัวโปรแกรมที่ยังต้องมีการพัฒนาเพิ่มเติมหลังจากที่ทดลองใช้
2. บางหัวข้อการทำงานนั้นยังไม่สมบูรณ์เพียงพอ ในการนำไปใช้งาน แต่งานที่ออกมาทั้งหมดถือว่าครอบคลุมการทำงานที่ผู้ใช้งานต้องการ(สำหรับข้อมูลที่คีย์เข้า)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เนื่องจากระบบงานที่พัฒนาเป็นเพียงหำระบบในทั้งหมดเกำระบบ คังนั้งานที่ออกมาค่อนข้างหยาบเนื่องจากรายละเอียดมีมาก และความต้องการเปลี่ยนแปลงบ่อยแต่ก็ไ้ทำการแก้ไข และบันทึกไว้เป็นแนวทางในการแก้ไขแล้ว

4. การพัฒนาที่รวดเร็วผู้ไ้ใช้ต้องมีความสามารถในการเขียนโปรแกรม โดยเขียน โดย “Borland DelPhi” และถ้ำทราบว่าการทำงานของแต่ละ คอมโพเน้นท์ของเคลไฟล์ “Delphi” ทำงานอย่างไรบ้าง จะสามารถพัฒนาในส่วนการติดค้อกับผู้ใช้ไ้ได้เร็วและสมบูรณ์แบบที่สุค ในส่วนของจัดการเรื่องความเร็วในการติดค้อฐานข้อมูลนั้น จะต้องบริหารการจัดการ การเปิด-ปิดฟอร์ม (Form) ของ แอปพลิเคชัน (Application) ให้ดี

5. ในส่วนรีเลชันนอลดาต้าเบส (Relational Database) ไมโครซอฟต์ เอสคิวแอล เซิร์ฟเวอร์เวอร์ชัน 6.5 (Microsoft SQL Server 6.5) การจัดการกับทริกเกอร์ (Trigger) สโตร์โพรซีเจอร์ (Store Proceder) จะทำให้ประสิทธิในเรื่ององความเร็วและทราฟฟิก (Traffic) ในเน็ตเวอร์ (Network) น้อย รวมไปถึงการจัดการฐานข้อมูลอย่างมีประสิทธิภาพ (Performance Tunning) ในขณะข้อมูลมีจำนวนมาก ๆ การรีคัพเวอร์รี่ (Recovery) เมื่อฐานข้อมูลเกิดปัญหาขึ้น (Event Problem) การสำรองข้อมูลและการนำข้อมูลลงฐานข้อมูล (Backup and Recovery)

6. การจัดการในส่วนวินโดว์เอ็นที (Windows NT) การจำกัดสิทธิต่าง ๆ การบริหารระบบเน็ตเวอร์ (Network Management) การเลือกใช้โปรโตคอล (Protocol) การจัดการในเรื่องรีโมทแอกเซส (Remote Access) การเรพพลิเคตาต้า (Data Replicate) รวมถึงการเลือกชนิดการ การจัดการของวินโดว์เอ็นที (Windows NT) เช่น การลง (Installation) แบบไพร์มารีโดเมน (Primary Domain) หรือแบบเซิร์ฟเวอร์ตัวเดียว (Standalone Server) และสิ่งทีสำคัญเมื่อวินโดว์เอ็นที (Windows NT) เกิดเฟลล (Fail) ต้องสามารถสามารถอัพขึ้นมาให้ไ้ได้ (Startup)

7. ในส่วนของรายงานเป็นการนำเอาข้อมูลทีไ้ได้ กรอกลงฐานข้อมูลขึ้นมาแสดงยังไม่สมบูรณ์

## กิตติกรรมประกาศ

ปริญญานิพนธ์นี้สำเร็จลงด้วย เนื่องมาจากความร่วมมือ และความช่วยเหลือ  
จากทุก ๆ ท่านที่เกี่ยวข้อง

รศ.ประทีป บัญญัติสินพรัตน์ ผู้ให้ความรู้ทางด้าน MIS เพื่อใช้ในการดำเนินงาน  
วางแผน และวิเคราะห์ รวบรวมข้อมูลอย่างมีประสิทธิภาพ

อาจารย์บรรจง ปิยธำรง ผู้ให้แนวทางในการค้นคว้าระบบ Network และนำมา  
ประยุกต์ใช้งาน ในเรื่องของ Client/Server

คณาจารย์โรงเรียนเทคโนโลยีชลบุรี ผู้ให้รายละเอียดเกี่ยวกับระบบงานบริหาร  
สถาบันการศึกษา

ขอบคุณบริษัทเว็ลด์โฟนซ็อบ จำกัด ที่ให้ยืมเครื่องและ แลนการ์ดในการทำ  
โปรเจก รวมทั้งโทรศัพท์ที่ใช้ในการติดต่อทางไกลระหว่างจังหวัด

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



บรรณานุกรม

1. นฤเลิศ เอี่ยมทัศนาศนา, “การเขียนโปรแกรมเคลไฟล์”,  
บริษัท ซีอีคยูเคชั่น จำกัด (มหาชน), 2539
2. “Database Application Developer’s Guide”, Borland International,  
Inc., 199p
3. C.J DATE, “An Introduction To Database Systems:”, 8339p., 1995



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้