

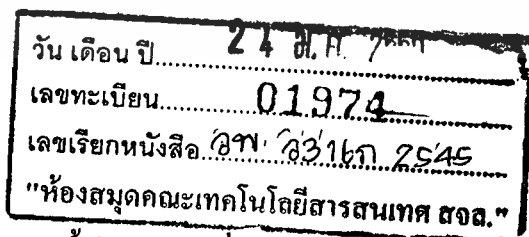
การพัฒนาระบบสารสนเทศโดยใช้ Oracle Designer6i CASE Tools ส่วนที่ 1

(กรณีศึกษา : ระบบทะเบียนนักศึกษา)

Development of an Information System using

Oracle Designer6i CASE Tools Part I

(Case Study : Student Registration System)



รายงานนี้เป็นส่วนหนึ่งของโครงการพัฒนาระบบงาน

หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

ภาคเรียนที่ 2 ปีการศึกษา 2545

คณะเทคโนโลยีสารสนเทศ

เอกสารนี้เป็นเอกสารสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ	การพัฒนาระบบสารสนเทศโดยใช้ Oracle Designer6i CASE Tools ส่วนที่ 1 (กรณีศึกษา : ระบบทะเบียนนักศึกษา)
นักศึกษา	นางสาว วราพร ถิ่นวัฒนาเลิศ
อาจารย์ที่ปรึกษา	ผศ. ดร. ประจวบ วานิชชัชวาล
ระดับการศึกษา	วิทยาศาสตร์มหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2545

บทคัดย่อ

โครงการพัฒนาระบบงานนี้มีวัตถุประสงค์ในการศึกษาและพัฒนาระบบสารสนเทศโดยใช้ CASE Tools Method มาทำการวิเคราะห์และออกแบบระบบ โดยเครื่องมือที่ใช้ คือ Oracle Designer6i ซึ่งจะช่วยให้ดำเนินการได้ง่ายและสะดวกต่อการปรับปรุงแก้ไข เนื่องจาก Oracle Designer6i นี้สนับสนุนหลักการงานทั้งแบบ Waterfall และ Spiral ทำให้ผลงานที่เข้าซ้่าซอลงได้สำหรับขั้นตอนในการพัฒนาเริ่มตั้งแต่การวิเคราะห์และออกแบบระบบโดยใช้ Entity Relationship Diagram, Data Flow Diagram, Function Hierarchy Diagram ลงบนเครื่องมือ Oracle Designer6i เพื่อทำการสร้างฐานข้อมูลและตัวโปรแกรม และจะทำการรัน Form โดยไปเรียก Oracle Form6i ให้โดยอัตโนมัติ

ระบบสารสนเทศของงานทะเบียนนักศึกษา ได้ถูกเลือกมาทำการศึกษาและพัฒนาโดยใช้ Oracle Designer6i เพื่อช่วยอำนวยความสะดวกในการจัดเก็บรวบรวมข้อมูล ประมวลผลข้อมูลตามขั้นตอนของการทำงาน อีกทั้งยังรวมถึงการแสดงผลการสืบค้นข้อมูลทางจอภาพ

Title	Development of an Information System using Oracle Designer6i CASE Tools Part I (Case Study : Student Registration System)
Student	Miss Waraporn Limwattnalert
Advisor	Asst. Prof. Prachuab Vanitchatchavan, Ph.D.
Level of Study	Master of Science in Information Technology
Major	Information Science
Academic Year	2002

ABSTRACT

In this project, the objective is to design and develop the information system using Oracle Designer6i which is one of the most popular CASE Tools. This tools supports the Waterfall model and Spiral model. It reduces the repeating processes in analyzing steps and helps system maintenance. The system design and development steps with Oracle Designer6i are composed of Entity Relationship Diagram, Data Flow Diagram, Function Hierarchy Diagram. All of the design results are used as a source for automatic database and applications generations. Generated form are executed by Oracle Form6i

The case study is the design and development of a student registration information system by using the Oracle Designer6i CASE Tool. This developed information system will be utilize for better preparing, processing, retrieving data and maintenance.

กิตติกรรมประกาศ

โครงการพัฒนาระบบงานนี้สำเร็จลุล่วงด้วยดีเนื่องมาจากความร่วมมือ และช่วยเหลือจากบุคคลหลายฝ่ายด้วยกัน โดยเฉพาะ ศศ.ดร.ประจวบ วานิชชัชวาล อาจารย์ที่ปรึกษาโครงการพัฒนาระบบงาน ที่คอยให้คำปรึกษาและแนะนำตลอดการทำโครงการพัฒนาระบบงานนี้ ซึ่งต้องขอขอบพระคุณอาจารย์เป็นอย่างสูง

ขอขอบคุณบิดา มารดาของข้าพเจ้า ที่ให้การเลี้ยงดู อบรมสั่งสอนข้าพเจ้าตลอดมา ตลอดจนเพื่อนๆ ที่คอยให้คำแนะนำและให้กำลังใจ และสุดท้ายนี้ขอขอบคุณคณะเทคโนโลยีสารสนเทศที่ให้ความอนุเคราะห์เครื่องคอมพิวเตอร์ในการทำโครงการพัฒนาระบบงานนี้

นางสาวรภาพร ลิ้มวัฒนาเลิศ
ผู้จัดทำ



สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญภาพ.....	VII
บทที่	
1: บทนำ.....	1
1.1 ความสำคัญและที่มา.....	1
1.2 วัตถุประสงค์ของโครงการ.....	1
1.3 ขอบเขตของโครงการ.....	2
1.4 ขั้นตอนและวิธีการดำเนินงาน.....	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	2
2. ทฤษฎีที่เกี่ยวข้อง.....	3
2.1 CASE Tools.....	3
2.1.1 Function Hierarchy Diagram.....	3
2.1.2 Dataflow Diagram	5
2.1.3 Entity Relationship Diagram	6
2.2 วงจรการพัฒนากระบวนการ (SDLC).....	8
2.3 Oracle Designer6i.....	9
3. การวิเคราะห์และออกแบบระบบงาน.....	15
3.1 ระบบทะเบียนนักศึกษา.....	15

3.2 Context Diagram ของระบบทะเบียนนักศึกษา.....	19
3.3 Dataflow Diagram ของระบบทะเบียนนักศึกษา.....	19
3.4 การวิเคราะห์และออกแบบฐานข้อมูล.....	22
4. การพัฒนาระบบงาน.....	30
4.1 การพัฒนาระบบส่วนฐานข้อมูล.....	30
4.2 การพัฒนาระบบส่วนโปรแกรมประยุกต์ใช้งาน.....	35
5. สรุปและข้อเสนอแนะ.....	49
บรรณานุกรม.....	51



สารบัญตาราง

ตารางที่	หน้า
3.1 ตาราง FACULTY	24
3.2 ตาราง DEPARTMENT	24
3.3 ตาราง STUDENT	25
3.4 ตาราง STUDENT_TYPES	25
3.5 ตาราง INSTRUCTOR	25
3.6 ตาราง CURRICULUM	26
3.7 ตาราง COURSE	26
3.8 ตาราง PREREQUISITE	27
3.9 ตาราง COURSE_OFFERED	27
3.10 ตาราง COURSE_SCHEDULE	28
3.11 ตาราง SEMESTER	28
3.12 ตาราง REGISTRATION_RECORD	29
3.13 ตาราง REGISTRATION_RECORD_DETAIL	29
4.1 กระบวนการทำงานของ Database Design Transformer	32

สารบัญรูป

รูปที่	หน้า
2.1	Function Hierarchy Diagram..... 4
2.2	องค์ประกอบของ Dataflow Diagram..... 6
2.3	สัญลักษณ์ของ Entity, Attribute และสัญลักษณ์ที่ใช้..... 7
2.4	สัญลักษณ์ของความสัมพันธ์ (Relationship)..... 8
2.5	Front Panel ของ Oracle designer 6i..... 10
3.1	Context Diagram ของระบบทะเบียนนักศึกษา..... 19
3.2	Dataflow Diagram Level-1 ของระบบทะเบียนนักศึกษาในส่วนข้อมูลคณะและ ภาควิชา และส่วนการลงทะเบียน..... 20
3.3	Dataflow Diagram Level-2 ของส่วนข้อมูลคณะและภาควิชา..... 21
3.4	Dataflow Diagram Level-2 ของส่วนการลงทะเบียน..... 22
3.5	Entity Relationship Diagram ของระบบทะเบียนนักศึกษาในส่วนข้อมูลคณะและ ภาควิชา และส่วนการลงทะเบียน..... 23
4.1	Entity Relationship Diagram ของระบบทะเบียนนักศึกษาในส่วนข้อมูลคณะและ ภาควิชา และส่วนการลงทะเบียน..... 31
4.2	Database Design Transformer..... 32
4.3	Server Model Diagram ของระบบทะเบียนนักศึกษา..... 33
4.4	ฟังก์ชัน Generate Database form Server Model..... 34
4.5	ตัวอย่าง Script DDL File ที่ได้จากการ Generate Database..... 34
4.6	Dataflow Diagram Level-1 ของระบบทะเบียนนักศึกษาในส่วนข้อมูลคณะและ ภาควิชา และส่วนการลงทะเบียน..... 36
4.7	Dataflow Diagram Level-2 ของส่วนข้อมูลคณะและภาควิชา..... 37
4.8	Dataflow Diagram Level-2 ของส่วนการลงทะเบียน..... 38
4.9	Function Hierarchy Diagram ของระบบทะเบียนนักศึกษาในส่วนข้อมูลคณะและ ภาควิชา และส่วนการลงทะเบียน..... 39
4.10	ฟังก์ชัน Entity Usages และฟังก์ชัน Attribute Usages 40

รูปที่	หน้า
4.11 การกำหนด Response Type ของฟังก์ชัน.....	41
4.12 การกำหนด Display Type ของแต่ละ Column.....	41
4.13 Application Design Transformer.....	42
4.14 Module Structure.....	43
4.15 Module Component Properties.....	44
4.16 LOV Structure.....	45
4.17 Generator Preference.....	46
4.18 Form การเพิ่มข้อมูลคณะและภาควิชา.....	47
4.19 Form การลงทะเบียนปกติ.....	47
4.20 Form การลงทะเบียนเพิ่มถอนรายวิชา.....	48



บทที่ 1

บทนำ

1.1 ความสำคัญและที่มา

เทคโนโลยีทางด้านซอฟต์แวร์คอมพิวเตอร์ได้มีการเปลี่ยนแปลงรวดเร็วเป็นอย่างมาก จากที่เคยเขียนโปรแกรมด้วย Third-Generation Language ต่อมาเปลี่ยนเป็น Fourth-Generation Language จนเมื่อหลายปีที่ผ่านมา CASE Tools ได้เริ่มเข้ามามีบทบาทสำคัญเพิ่มขึ้นเรื่อยๆ จนเป็นที่ยอมรับกันว่าเป็นเทคนิคในการพัฒนาระบบงานคอมพิวเตอร์ที่ฉลาดที่สุดในยุคปัจจุบัน และน่าจะเป็นเครื่องมือที่ใช้กันอย่างแพร่หลายในอนาคต เนื่องจาก CASE Tools เป็นเครื่องมือที่ช่วยให้ผู้พัฒนาระบบสามารถพัฒนาระบบงานที่ซับซ้อนได้ง่ายและรวดเร็วขึ้น และสะดวกต่อการปรับปรุงแก้ไข

สำหรับโครงการนี้ได้นำเอา CASE Tools ที่ชื่อว่า Oracle Designer6i มาช่วยในการออกแบบและพัฒนาระบบงานทะเบียนนักศึกษา ซึ่งเป็นระบบงานที่ค่อนข้างใหญ่และซับซ้อน มีหน่วยงานที่เกี่ยวข้องและข้อมูลจำนวนมาก การนำเอา CASE Tools มาช่วยในการพัฒนาระบบจัดการข้อมูลและจัดทำเอกสาร จึงเป็นสิ่งจำเป็นอย่างยิ่ง เนื่องจาก Oracle Designer6i นี้สนับสนุนการทำงานแบบ Spiral ทำให้ลดงานที่ซ้ำซ้อนลงได้ และมีส่วนการติดต่อกับผู้ใช้ (User Interface) ที่ง่ายต่อการใช้งาน โดย Oracle Designer6i นี้จะช่วยในการจัดการฐานข้อมูลของระบบ, ช่วยในการวิเคราะห์กิจกรรมต่างๆของระบบ และทำการออกแบบโครงสร้างของระบบ เพื่อนำไปสร้างเป็นโปรแกรมประยุกต์ใช้งานต่อไป อีกทั้งยังช่วยในการจัดทำเอกสารคู่มือและรายงานตลอดทุกขั้นตอนของการพัฒนาระบบ เรียกได้ว่าในขั้นตอนการพัฒนาระบบทุกขั้นตอนสามารถนำ Oracle Designer6i นี้ไปช่วยทั้งในการเขียน Diagram ต่างๆ การออกรายงาน จนกระทั่งสร้างตัวโปรแกรมที่เป็นโปรแกรมต้นแบบโดยง่ายให้

1.2 วัตถุประสงค์ของโครงการ

ในการออกแบบและพัฒนาระบบงาน มีวัตถุประสงค์ ดังนี้

1.2.1 ศึกษากระบวนการพัฒนาระบบสารสนเทศและการทำงานของ CASE Tools เพื่อใช้

เป็นเครื่องมือในการออกแบบและพัฒนาระบบงาน

1.2.2 ออกแบบและพัฒนาระบบทะเบียนนักศึกษาโดยใช้ Oracle Designer6i CASE Tool

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของโครงการ

สำหรับโครงการพัฒนาระบบงานนี้ จะทำการวิเคราะห์ ออกแบบ และพัฒนาระบบสารสนเทศทะเบียนนักศึกษาในส่วนงานการลงทะเบียนและการจัดการข้อมูลคณะและภาควิชา ซึ่งระบบงานนี้จะใช้เป็นกรณีศึกษา เพื่อที่จะแสดงให้เห็นว่าการใช้ Oracle Designer6i CASE Tool ช่วยให้การพัฒนาระบบง่าย รวดเร็วและมีประสิทธิภาพขึ้น อีกทั้งยังสะดวกต่อการปรับปรุงแก้ไข

1.4 ขั้นตอนและวิธีการดำเนินการ

- 1.4.1 ศึกษาหลักการและทฤษฎีต่างๆที่เกี่ยวข้อง เช่น ทฤษฎี CASE Method, ระบบฐานข้อมูลแบบ Relational, ภาษา PL/SQL เป็นต้น
- 1.4.2 ศึกษาหลักการงานและวิธีการใช้งาน Oracle Designer6i CASE Tool ออกแบบและพัฒนาระบบทะเบียนนักศึกษาโดยใช้ Oracle Designer6i CASE Tool ในการออกแบบ
- 1.4.3 สำหรับขั้นตอนในการพัฒนา เริ่มตั้งแต่การวิเคราะห์และออกแบบระบบโดยใช้ Entity Relationship Diagram, Data Flow Diagram, Function Hierarchy Diagram ลงบนเครื่องมือ Oracle Designer6i เพื่อทำการสร้างฐานข้อมูลและตัวโปรแกรม และจะทำการรัน Form โดยไปเรียก Oracle Form6i ให้โดยอัตโนมัติ ส่วนฐานข้อมูลจะถูกนำไปสร้างบนระบบจัดการฐานข้อมูล Oracle8i

1.5 ประโยชน์ที่คาดว่าจะได้รับ

ในการพัฒนาระบบ โดยใช้ CASE Tools คาดว่าจะได้รับประโยชน์ ดังนี้

- 1.5.1 จัดเก็บข้อมูลที่ใช้ในการพัฒนาระบบรวมไว้ที่เดียวกันอย่างครบถ้วน เพื่อลดเวลาในการสืบค้นและเวลาในการดูแล และปรับปรุงแก้ไขระบบ เช่น Requirement Specification, Program Specification, Source Code, และมาตรฐานการเขียนโปรแกรม เป็นต้น
- 1.5.2 เกิดความคล่องตัวในการประสานงานระหว่างผู้ที่เกี่ยวข้องกับการพัฒนาระบบ
- 1.5.3 ทำให้การพัฒนาระบบมีขั้นตอนการทำงานที่มีรูปแบบเป็นมาตรฐานเดียวกันทั้งองค์กร
- 1.5.4 แนวคิดการใช้ CASE Tools ไปพัฒนาระบบอื่นๆต่อไป

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

2.1 CASE Tools

CASE Tools (Computer-aided Software Engineering) เป็นเครื่องมือที่พัฒนาขึ้นมาเพื่อใช้ในการออกแบบและสร้างระบบสารสนเทศ ประกอบด้วยการวิเคราะห์ระบบ การออกแบบระบบ และการสร้างระบบ โดยจะเริ่มตั้งแต่การรวบรวมความต้องการทางธุรกิจของระบบที่จะพัฒนา แล้วนำความต้องการเหล่านั้นมาออกแบบเป็นโครงสร้างไว้ แล้วนำโครงสร้างดังกล่าวมาสร้างเป็นระบบสารสนเทศขึ้นมา ซึ่งวิธีการพัฒนาระบบสารสนเทศโดยใช้ CASE Tools นี้เรียกว่า CASE Method

ปัจจุบันนี้มีการนำเอา CASE Tools มาช่วยในการออกแบบ วิเคราะห์ ควบคุมและทำเอกสารในระบบงานใหญ่ๆมากขึ้น เนื่องจากบริษัท และองค์กรต่างๆเห็นว่า CASE Tools ช่วยในการสร้าง Diagram และช่วยในการแก้ไข ปรับปรุงงานในระบบงานที่มีขนาดใหญ่ได้เป็นอย่างดี อีกทั้ง CASE Tools ต่างๆที่พัฒนาขึ้นมาในปัจจุบัน มีองค์ประกอบที่ทำให้สามารถใช้งานได้ง่ายและมีประสิทธิภาพมากขึ้น ตัวอย่างเช่น มีการใช้ Graphics เข้ามาช่วยในการออกแบบและสร้าง Diagram ต่างๆ, มีตัว Generators สำหรับสร้างหน้าจอเพื่อใช้ในการแสดงผล Input/Output ของระบบ, มี Repository ที่ใช้เก็บข้อมูลเกี่ยวกับการออกแบบทั้งหมดไว้ในที่เดียวกัน เหมาะกับระบบงานใหญ่ๆที่มีการใช้งานแบบ Multi-user

ในการนำ CASE Tools มาใช้ในการวิเคราะห์และออกแบบระบบนั้น ช่วยให้ขั้นตอนในการพัฒนาระบบมีระเบียบวิธีที่เป็นมาตรฐาน ลดระยะเวลา ค่าใช้จ่าย ตลอดจนการดูแลรักษาระบบที่พัฒนาขึ้น ทำให้สามารถปรับปรุงเปลี่ยนแปลงได้ง่าย

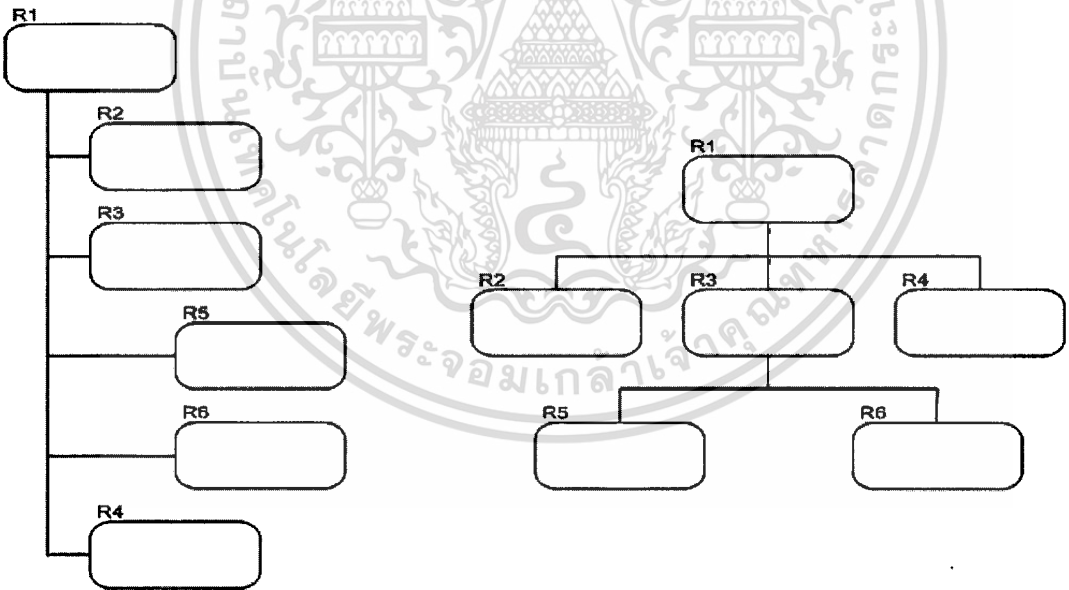
ในขั้นตอนการวิเคราะห์ระบบโดยใช้ CASE Tools มีการใช้เทคนิคในการสร้างไดอะแกรมต่างๆ ดังนี้

2.1.1 Function Hierarchy Diagram

เป็นการเขียนกิจกรรมทั้งหมดที่ต้องทำภายในองค์กรหรือในหน่วยงานหนึ่งๆ โดยการนำงานย่อยแต่ละอย่างหรือฟังก์ชันทางธุรกิจ (Business Function) มาเขียนไล่เป็นลำดับชั้น ฟังก์ชันที่อยู่บนจะถูกแตกเป็นฟังก์ชันย่อยๆในลำดับชั้นถัดลงมา จนกระทั่งได้โครงสร้างขององค์กรเป็นเอกสารนี้เป็นเอกสารที่ส่งมอบไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Hierarchy ที่สมบูรณ์ และครอบคลุมการทำงานทั้งหมด รวมทั้งสามารถเข้าใจการทำงานนั้นได้ การแตกย่อยของฟังก์ชันก็จะสิ้นสุด ในการแตกฟังก์ชันย่อยนี้ลำดับก่อนหลังที่ปรากฏในฟังก์ชันไม่ใช่สิ่งสำคัญ (ไม่แน่ใจว่าจะต้องเรียงลำดับก่อนหลัง) แต่จะเน้นถึงลำดับชั้นของฟังก์ชันในแต่ละระดับ (Level) มากกว่า ฟังก์ชันที่อยู่ในชั้นที่สูงกว่าจะถูกอธิบายโดยฟังก์ชันที่อยู่ด้านล่างที่แตกออกมาเรื่อยๆ นอกจากนี้การเขียน

สัญลักษณ์ที่ใช้แสดงฟังก์ชันทางธุรกิจ (Business Function) จะเป็นรูปกรอบสี่เหลี่ยมที่มีมุมขอบมน (แต่ใน Oracle Designer6i CASE Tools จะใช้เป็นรูปสี่เหลี่ยมผืนผ้า) ภายในกรอบคือรายละเอียดของฟังก์ชัน ที่มุมซ้ายบนแสดงชื่อสั้นๆไว้เป็นเลเบล (Label) เพื่อให้สะดวกในการเรียกและอาจมีเครื่องหมาย \oplus ที่มุมขวาบน เพื่อแสดงว่าฟังก์ชันดังกล่าวยังสามารถแตกรายละเอียดฟังก์ชันนั้น ออกเป็นฟังก์ชันย่อยๆลงไปได้อีก แต่รายละเอียดไม่ได้ปรากฏให้เห็นบนแผนภาพในขณะนั้น การเลือกใช้ชื่อฟังก์ชันจะต้องบ่งบอกถึงคำจำกัดความว่าฟังก์ชันนั้นมีหน้าที่ทำอะไรอย่างชัดเจน ชื่อของฟังก์ชันจะต้องขึ้นต้นด้วยคำกริยาและจะต้องไม่ระบุถึงวิธีปฏิบัติ นอกจากนี้ควรหลีกเลี่ยงการใช้คำศัพท์เฉพาะทาง, อักษรย่อ, คำย่อที่อาจสื่อความหมายกำกวม



รูปที่ 2.1 Function Hierarchy Diagram

อีกองค์ประกอบหนึ่งที่แสดงอยู่ในโมเดล (Model) คือ เหตุการณ์ (Event) จะแสดงด้วยสัญลักษณ์ลูกศรที่มีข้อความภายในลูกศร เหตุการณ์ดังกล่าวอาจเป็นเหตุการณ์ผลลัพธ์ (Outcome) ใดๆ หรือเป็นเหตุการณ์ใดๆที่กระตุ้น (Trigger) ให้แต่ละฟังก์ชันเกิดขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับการแสดงการย่อยของฟังก์ชันอาจจะแสดงได้ทั้งแนวนอน (Horizontal Layout), แนวตั้ง (Vertical Layout) หรือแบบผสม (Hybrid Layout) ก็ได้ ซึ่งตามทฤษฎีแล้วถือว่าเหมือนกันไม่ว่าจะแสดงในรูปแบบใด ขึ้นอยู่กับเนื้อหาในการใช้งานและความสะดวกในการใช้งาน

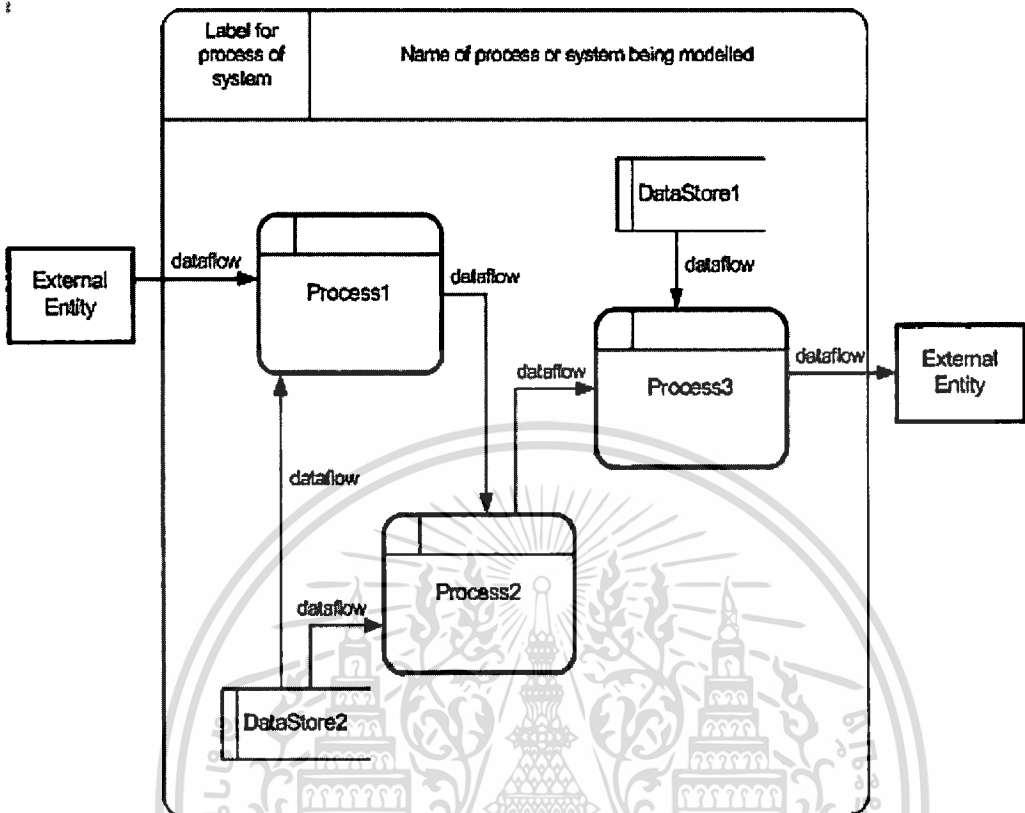
2.1.2 Dataflow Diagram

ไดอะแกรมชนิดนี้ใช้แสดงรายละเอียดว่ากิจกรรมในองค์กรนั้นต้องใช้ข้อมูลอย่างไร โดยจะแสดงให้เห็นว่าข้อมูลใดจะถูกจัดเก็บ และแสดงให้เห็นว่าข้อมูลต่างๆจะถูกดำเนินการผ่านโพรเซสใดบ้าง

ประโยชน์ของไดอะแกรมชนิดนี้ คือ ใช้ช่วยในการตัดสินใจว่าควรจะทำการแตกโพรเซสลงไปเป็นโพรเซสย่อยอีกหรือไม่ จนกระทั่งได้ไดอะแกรมในระดับที่เป็นรายละเอียดที่สุดแล้ว ไดอะแกรมนั้นก็เหมาะที่จะใช้ในการกำหนด Specification สำหรับงานออกแบบ, เขียนโปรแกรม, เขียนเอกสารคู่มือ หรือใช้ออกแบบอุปกรณ์ต่างๆเพื่อใช้ในระบบต่อไป มีข้อสังเกตว่าไดอะแกรมนี้ไม่สามารถอธิบายการทำงานภายในโพรเซสหรือฟังก์ชันได้ และไม่สามารถบอกได้ว่าเมื่อใดจึงจะเกิดการ ทำงานของโพรเซสใด เพราะไม่มีการแสดง Event ไว้ในไดอะแกรมนี้เลย

นอกจากโพรเซสแล้ว ไดอะแกรมนี้จะมีส่วนประกอบสำคัญอีก 3 ส่วน คือ

- **Dataflow** ใช้แสดงข้อมูลใดที่ถูกแลกเปลี่ยนระหว่าง 2 โพรเซส หรือระหว่างโพรเซสกับ Datastore สัญลักษณ์ของ Dataflow คือเส้นตรงที่มีหัวลูกศรชี้ไปทิศทางที่เกิดการไหลของข้อมูล ซึ่งอาจเกิดได้ทั้ง 2 ทางก็จะเป็นเส้นตรงที่มีหัวลูกศรทั้ง 2 ปลาย
- **Datastore** ใช้แทนหน่วยข้อมูลที่ถูกจัดเก็บ โดยจะมีการกำหนดชื่อไว้ใช้อย่างถึงจากโพรเซสต่างๆ สัญลักษณ์ของ Datastore จะเป็นรูปกรอบสี่เหลี่ยมตัดมุมที่เปิดด้านหนึ่ง ภายในกรอบแสดงชื่อย่อและคำอธิบาย
- **External Entity** เป็นสัญลักษณ์ที่ใช้แสดงสิ่งใดๆที่อยู่นอกขอบเขตระบบงาน แต่สัมพันธ์กับระบบด้วยการส่งผ่านของข้อมูลถึงกันอยู่



รูปที่ 2.2 องค์ประกอบของ Dataflow Diagram

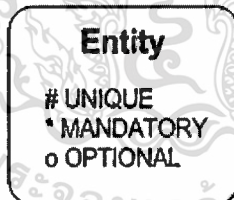
2.1.3 Entity Relationship Diagram

เป็น โมเดลที่ใช้แสดงความสัมพันธ์ของข้อมูลระหว่าง Entity เพื่อใช้กำหนดข้อมูลที่ระบบต้องการ เป็นวิธีการ (Algorithm) ที่สามารถออกแบบฐานข้อมูลให้มีลักษณะเป็น โครงสร้าง (Data Structure) ที่มีข้อมูลอยู่บนนั้นเลย ซึ่ง ER Model นี้มีรูปแบบการนำเสนอที่สามารถมองจากโมเดลแล้วเข้าใจได้โดยง่าย

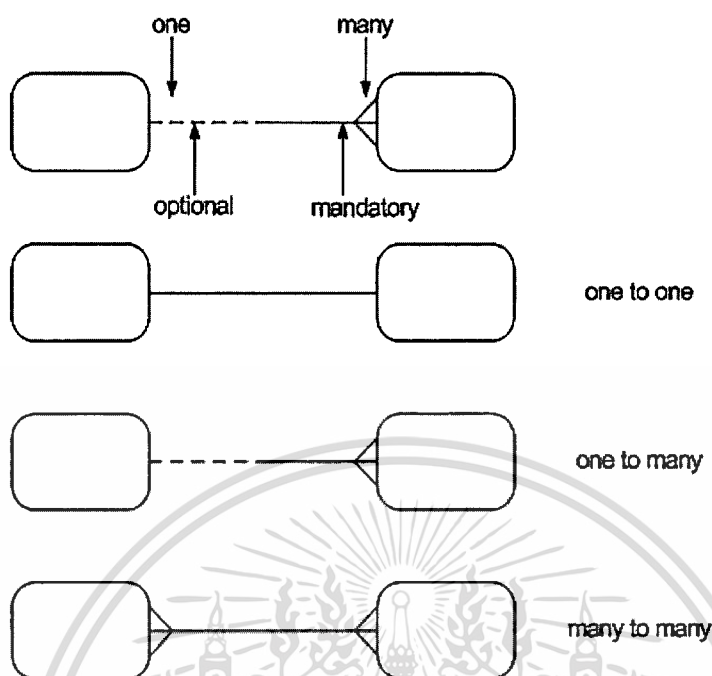
ส่วนประกอบของ Entity Relationship Diagram

- เอนติตี้ (Entity) คือ สิ่งของหรือสิ่งที่เราสนใจหรือหัวข้อที่เป็นหลักสำคัญ อาจจะเป็นสิ่งที่มีอยู่จริง หรือถูกสมมติขึ้นมาก็ได้ จะเป็นสิ่งที่บรรจุหรือเกี่ยวข้องกับข้อมูลด้านต่างๆที่จำเป็นต้องรู้ เช่น คน, สถานที่, สิ่งของ, สินค้า, การสั่งซื้อ เป็นต้น โดยจะแสดงอยู่ในกรอบสี่เหลี่ยมขอบมน (Soft Box)

- แอททริบิวต์ (Attribute) คือ รายละเอียดต่างๆที่บ่งบอกได้ถึงคุณภาพ, ลักษณะ, ประเภท, จำนวน หรือข้อกำหนดคุณลักษณะที่เกี่ยวข้องกับสถานะของเอนทิตี โดย แอททริบิวต์สามารถแบ่งออกเป็น 3 ลักษณะ คือ
 - Unique Identifier (UID) เป็นแอททริบิวต์ที่ค่าในแอททริบิวต์นั้นไม่ซ้ำกัน และไม่สามารถที่จะเป็นค่า Null ได้ ใช้เครื่องหมาย # หน้าแอททริบิวต์นั้นๆ
 - Mandatory Attribute เป็นแอททริบิวต์ที่จะต้องมียค่าอยู่เสมอ ไม่สามารถที่จะเป็นค่า Null ได้ ใช้เครื่องหมาย * หน้าแอททริบิวต์นั้นๆ
 - Optional Attribute เป็นแอททริบิวต์ที่อาจจะมีค่าหรือไม่มีก็ได้ สามารถเป็นค่า Null ได้ ใช้เครื่องหมาย o หน้าแอททริบิวต์นั้นๆ
- ความสัมพันธ์ (Relationship) คือ สิ่งที่สำคัญในการเชื่อมโยงระหว่าง 2 เอนทิตีซึ่งสามารถแสดงความสัมพันธ์ได้ทั้งที่เป็น Optional (เส้นประ) หรือความสัมพันธ์แบบ Mandatory (เป็นเส้นทึบ) คือ จะต้องมียค่าข้อมูลเสมอ ไม่สามารถเป็นค่า Null ได้ และยังสามารถแสดงถึงความสัมพันธ์ในลักษณะที่เป็นหนึ่งต่อหนึ่ง (One-to-One) หรือหนึ่งต่อกลุ่ม (One-to-Many) หรือกลุ่มต่อกลุ่ม (Many-to-Many)



รูปที่ 2.3 สัญลักษณ์ของ Entity, Attribute และสัญลักษณ์ที่ใช้



รูปที่ 2.4 สัญลักษณ์ของความสัมพันธ์ (Relationship)

2.2 วงจรการพัฒนากระบวนการ (System Development Life Cycle)

ระบบงานต่างๆ ไม่ว่าจะเป็นงานด้านธุรกิจ การศึกษา หรืออื่นๆ ในปัจจุบันนี้ มีความซับซ้อนเพิ่มมากขึ้น การวิเคราะห์ระบบงานต่างๆ เหล่านี้จึงเกิดความยุ่งยากอันเนื่องมาจากความซับซ้อนของระบบงานเอง ด้วยเหตุนี้การพัฒนาระบบงานควรจะมีมาตรฐานขึ้น เพื่อช่วยลดความซับซ้อนที่เกิดขึ้นภายในระบบงานได้ รวมไปถึงทำให้เกิดข้อผิดพลาดได้น้อยที่สุด

วงจรพัฒนาระบบงานก็เป็นหลักการหนึ่งที่น่าสนใจในการพัฒนาระบบเพื่อตอบสนองความต้องการเหล่านั้น โดยในวงจรพัฒนาระบบงานนี้จะแบ่งออกเป็น 7 ขั้นตอน ดังนี้

1. Problem Definition

ในขั้นตอนนี้ เป็นขั้นตอนการศึกษาระบบ และการกำหนดขอบเขตในการศึกษา รวมไปถึงลักษณะของปัญหาที่ผู้ใช้ประสบอยู่ โดยจะต้องทำการค้นหาให้ได้อย่างชัดเจนว่า สิ่ง que ผู้ใช้ต้องการในระบบนี้คืออะไร หรือว่าปัญหาที่เกิดขึ้นในการทำงานคืออะไรบ้าง

2. Requirement Analysis

ในขั้นตอนนี้ จะต้องทำการวิเคราะห์การทำงานของระบบปัจจุบัน เพื่อทำการค้นหาวิธีการในการแก้ไขปัญหาที่เกิดขึ้นในระบบปัจจุบัน หรือวิธีการที่จะทำให้สามารถตอบสนอง

ความต้องการที่เพิ่มขึ้นของผู้ใช้ได้ใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรรมใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. Specification

เป็นขั้นตอนที่จะต้องระบุให้ได้ว่า ระบบจะพัฒนาไปในแนวทางใด ด้วยวิธีการใด และมีความจำเป็นที่จะต้องมียุทธศาสตร์ประกอบอะไรบ้าง โดยจะต้องมีความชัดเจน เพื่อให้สามารถนำไปทำการออกแบบระบบได้

4. Design

เป็นขั้นตอนที่จะนำสิ่งที่ได้ระบุเอาไว้มาทำการออกแบบระบบงาน โดยการออกแบบจะต้องให้มีความสอดคล้องกับการทำงานในระบบปัจจุบัน และจะต้องสามารถรองรับความต้องการ และสามารถแก้ปัญหาที่เกิดขึ้นของผู้ใช้ได้

5. Coding

เป็นขั้นตอนที่นักพัฒนาระบบเริ่มที่จะทำการเขียนโปรแกรม เพื่อให้มีความสอดคล้องกับความต้องการของผู้ใช้ และเป็นไปตามสิ่งที่ได้ออกแบบไว้

6. Testing

เป็นขั้นตอนในการทดสอบระบบที่พัฒนาขึ้นมา เพื่อเป็นการยืนยันว่าระบบที่พัฒนาขึ้นมา มีความถูกต้องสมบูรณ์ สอดคล้องกับที่ได้ตกลงกันเอาไว้หรือไม่

7. Operation and Maintenance

เป็นขั้นตอนที่ผู้ใช้เริ่มนำระบบไปใช้งานจริงภายใต้สภาวะแวดล้อมจริงๆ เมื่อใช้ระบบไปสักระยะหนึ่งอาจจะเกิดความผิดพลาดขึ้น ซึ่งจะเข้าสู่ส่วนของการบำรุงรักษาระบบ โดยในขั้นตอนนี้ก็จะมี การปรับปรุงเปลี่ยนแปลง หรือแก้ไขให้รองรับกับความต้องการของผู้ใช้ที่เพิ่มขึ้น

2.3 Oracle Designer6i

บริษัท Oracle เป็นบริษัทหนึ่งที่ได้มีการพัฒนา CASE Tools ขึ้นมา ซึ่ง Oracle Designer6i ก็เป็น CASE Tool ตัวหนึ่งของบริษัท Oracle ที่ได้ทำการพัฒนาให้สามารถใช้พัฒนาระบบงานได้ง่าย รวดเร็ว และสร้าง Application ได้อย่างรวดเร็ว มีประสิทธิภาพ แล้วยังรองรับการออกแบบระบบฐานข้อมูลแบบ Object และฐานข้อมูลแบบ Relational Database ได้โดยไม่มีปัญหาใด ๆ

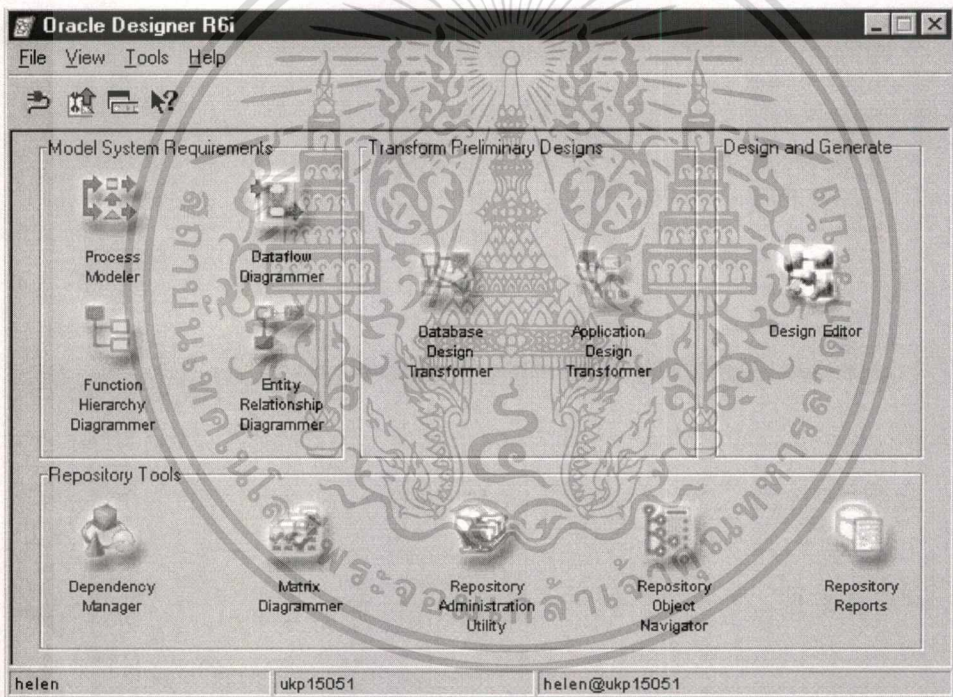
การเก็บรวบรวมข้อมูลต่าง ๆ ของ Oracle Designer6i นั้นจะเก็บไว้ที่ส่วนกลางที่เรียกว่า Repository ซึ่งจะเก็บ Object, Meta Data ที่เกี่ยวกับการวิเคราะห์ ออกแบบและการสร้างระบบ ของระบบที่ทำการพัฒนา

ความสามารถในการสร้างระบบสารสนเทศของ Oracle Designer6i นั้นมีความสามารถที่ครอบคลุมการออกแบบระบบ กล่าวคือ สามารถที่จะสร้าง Forms ต่าง ๆ Web Applications ซึ่งมีเอกสารเป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสนับสนุนภาษา Java Reports ของระบบสามารถสร้าง Menu รวมไปถึงสามารถที่จะสร้าง Document ของระบบได้อีกด้วย

2.3.1 ส่วนประกอบของ Oracle Designer 6i

การทำงานของ Oracle Designer 6i มี 2 ส่วน คือ ฟังก์ชันไคลเอนต์ และฟังก์ชันเซิร์ฟเวอร์ ในส่วนของฟังก์ชันไคลเอนต์จะเกี่ยวข้องกับเครื่องมือ และ Utility ที่ใช้ในการออกแบบ การพัฒนา และการจัดการรายละเอียดต่าง ๆ ภายในระบบ และในส่วนของฟังก์ชันเซิร์ฟเวอร์จะมีความเกี่ยวข้องกับการจัดการ Repository ที่เป็นศูนย์รวมของข้อมูลและออบเจกต์ทั้งหมดที่มีภายในระบบ ซึ่งหน้าจอ Front Panel ของ Oracle Designer 6i มีลักษณะดังรูปที่ 2.5



รูปที่ 2.5 Front Panel ของ Oracle designer 6i

2.3.1.1 การทำงานฟังก์ชันไคลเอนต์

บนฟังก์ชันไคลเอนต์มีเครื่องมือต่างๆที่สนับสนุนทุกขั้นตอนของการพัฒนาระบบ ดังนี้

- **Analysis and Design Tools**

เครื่องมือในการวิเคราะห์และออกแบบระบบ ประกอบด้วย 3 ส่วนหลักๆ ได้แก่

1. Model System Requirements

เป็นเครื่องมือที่ใช้ในการวิเคราะห์ความต้องการของผู้ใช้ สร้างออกมาเป็นโมเดล ซึ่งจะเป็นอินพุตของ Transformer ต่อไป ซึ่งประกอบด้วย

- Process Modeler

เป็นเครื่องมือที่ช่วยให้ผู้ออกแบบระบบสามารถสร้างโมเดลของธุรกิจตามวัตถุประสงค์ขององค์กร เพื่ออธิบาย Business Process ออกมาในลักษณะรูปภาพ และ Flow การทำงานของธุรกิจ และแสดงความสัมพันธ์ของแต่ละ Process เพื่อให้ผู้พัฒนาระบบและผู้ใช้ระบบมองระบบออกมาในภาพเดียวกัน ช่วยให้เกิดความร่วมมือในการสร้างและแก้ไข Model ร่วมกัน เพื่อให้ได้ Model ที่ตรงกับความต้องการทางธุรกิจที่จะพัฒนาระบบ และสามารถนำไปใช้พัฒนาต่อในขั้นตอนอื่นได้อย่างถูกต้อง

- Function Hierarchy Diagrammer

เป็นเครื่องมือที่ใช้แสดงการแบ่งย่อยของกิจกรรมต่างๆ ในองค์กร ว่าในองค์กรนั้นมีฟังก์ชันการทำงานอะไรบ้าง และแบ่งย่อยฟังก์ชันเหล่านั้นลงไปจนกระทั่งรู้รายละเอียดของงานครบถ้วน

- Dataflow Diagrammer

เป็นเครื่องมือที่ใช้ในการทำแผนภาพแสดงการไหลของข้อมูลภายในธุรกิจ ซึ่งจะแสดงรายละเอียดงานของระบบว่ามีการใช้ข้อมูลอะไรบ้าง และมีการแลกเปลี่ยนข้อมูลกันอย่างไร

- Entity Relationship Diagrammer

เป็นเครื่องมือที่ใช้แสดงความต้องการภายในระบบขององค์กร และกำหนดข้อมูลที่ระบบงานนั้นๆต้องการ โดยมีการสร้างโมเดลที่มีลักษณะเป็นกราฟพิคเพื่อใช้แสดงความสัมพันธ์ระหว่าง Entity ว่าเกี่ยวข้องกันอย่างไรในระบบงาน และทำการกำหนด Entities, Attributes, Relationship, และ Domains ทั้งหมดของระบบงานลงใน Repository

2. Transform Preliminary Designs

เป็นเครื่องมือที่ใช้ในการเปลี่ยนแปลงการรูปแบบโครงสร้างอย่างง่ายที่ได้จากขั้นตอนการวิเคราะห์ระบบ ให้เป็นการออกแบบเบื้องต้น ซึ่งประกอบด้วย

- Database Design Transformer (DDT)

เป็นเครื่องมือที่ใช้ในการจัดการแผนผังการออกแบบฐานข้อมูลที่ได้จากการ

ออกแบบ ER Diagram ตามความต้องการของระบบในขั้นตอนของ Entity

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Relationship Diagrammer ให้อยู่ในรูปของข้อมูลที่สามารถที่จะเปลี่ยนให้เป็นฐานข้อมูลได้

- Application Design Transformer (ADT)

เป็นเครื่องมือที่ใช้ในการจัดการข้อมูลที่ได้จากขั้นตอนของ Function Hierarchy Diagrammer และ Dataflow Diagrammer เพื่อให้อยู่ในรูปแบบที่สามารถทำการสร้างเป็นแอปพลิเคชันที่ตั้งอยู่บนพื้นฐานของฟังก์ชัน และ Business Process ที่ได้จากความต้องการของระบบได้

3. Design and Generate

กลุ่มเครื่องมือที่ใช้ในการออกแบบแนวทางของระบบให้เป็นฟอร์ม (Form) ที่สอดคล้องและตรงตามความต้องการขององค์กรสำหรับผู้ที่ทำหน้าที่จัดการระบบและผู้ที่ออกแบบระบบ โดยมีการสร้างองค์ประกอบต่างๆ ในฝั่งของเซิร์ฟเวอร์ และแอปพลิเคชัน ในฝั่งของไคลเอนต์จากรายละเอียดเรคอร์ดที่อยู่ภายใน Repository เครื่องมือที่ทำหน้าที่ในส่วนนี้ คือ

- Design Editor

เป็นเครื่องมือที่ให้สภาพแวดล้อมในการออกแบบและการสร้างแอปพลิเคชันของทั้งฝั่งไคลเอนต์และฝั่งเซิร์ฟเวอร์ รวมไปถึงในสภาพแวดล้อมเดียวกัน ใช้สำหรับทั้งการออกฐานข้อมูล และการออกแบบแอปพลิเคชัน

● **Generators** ใช้ในการสร้างส่วนต่างๆ ของระบบ โดยแบ่งเป็น 2 ส่วน คือ

- **Server Generators** สามารถสร้างฐานข้อมูลของออร์เคิล, ODBC และฐานข้อมูลประเภทอื่นได้

- **Client Generators** สามารถสร้างแอปพลิเคชันได้ในหลายแพลตฟอร์ม ดังนี้

* Oracle Forms และ Oracle Reports

* Oracle Web PL/SQL Application

* Visual Basic Project

* MS Help Application

● **Utilities and Services**

นอกเหนือจาก Client tools และ Repository แล้ว Oracle Designer 6i ยังสนับสนุน Utilities และ Services ต่างๆ ได้แก่

- Repository Services ประกอบด้วย

* **Repository Object Navigator** เป็นเครื่องมือที่ทำให้สามารถเข้าถึงออบเจกต์ทั้งหมดใน Repository ใช้ในการจัดเก็บและจัดการออบเจกต์ทั้งหมดที่มีในการพัฒนาระบบ และมีการจัดเก็บลงใน Repository โดยจะมีการใช้งานร่วมกับ Repository Administration Utility เพื่อใช้ในการดูแลความสัมพันธ์ทั้งหมดที่อยู่ภายในและการเข้าถึงออบเจกต์ของคลังข้อมูลส่วนกลาง นอกจากนี้ผู้ดูแลระบบยังใช้ Repository Object Navigator สำหรับควบคุมและกำหนดสิทธิ์ในการเข้าถึงระบบด้วย

* **Matrix Diagrammer** เป็นเครื่องมือที่ใช้ทำ Cross-check ระหว่างอิลิเมนต์ที่มีรูปแบบต่างกันได้ เช่น สามารถทำ Cross-check ระหว่าง Business Functions และ Entities

* **Repository Reports** เป็นเครื่องมือที่ใช้สำหรับสร้างรายงาน (Report) ของข้อมูลใน Repository

* **Repository Administration Utility** ใช้สำหรับการสร้างและดูแล Repository รวมถึงการควบคุมการเข้าถึง Repository

- **Online Documentation** เป็นเอกสารออนไลน์เกี่ยวกับวิธีการใช้และหลักการทำงานของ Oracle Designer 6i

2.3.1.2 การทำงานฝั่งเซิร์ฟเวอร์

มีส่วนประกอบหลักในการจัดการอยู่ 2 ส่วน คือ

1) Repository

เปรียบเสมือนเป็นหัวใจของ Oracle Designer เนื่องจาก Repository อำนวยความสะดวกในการออกแบบและสร้างแอปพลิเคชันที่รวดเร็วและมีประสิทธิภาพ

Oracle Repository เป็นชุดเครื่องมือสำหรับเก็บและจัดการ Meta Data ของแอปพลิเคชัน ซึ่งสนับสนุนการทำงานในสภาพแวดล้อมแบบ Multi-user ทำให้สนับสนุนการพัฒนาระบบแบบเป็นทีม เพราะสามารถเข้าถึง Repository ร่วมกันได้ นอกจากนี้เมื่อมีการจัดการเกิดขึ้นกับอิลิเมนต์ใดก็ตามใน Repository จะไม่เกิดการเปลี่ยนแปลงกับ Definition ของอิลิเมนต์นั้นจริงๆ การเปลี่ยนแปลงจะเป็นแบบ Logical ไม่ใช่ Physical ผู้ที่ทำหน้าที่สร้าง, ดูแล และกำหนดสิทธิ์ในการเข้าถึง Repository เรียกว่า Repository Administration

2) API (Application Programmatic interface)

เป็นส่วนที่ทำหน้าที่ในการ Update Repository ซึ่งจะ Update ทันที เมื่อมีการเปลี่ยนแปลงและแก้ไขข้อมูลใดๆผ่าน Tools ต่างๆของ Oracle Designer 6i



บทที่ 3

การวิเคราะห์และออกแบบระบบงาน

สำหรับในส่วนของโครงการนั้น ได้ทำการพัฒนาระบบสารสนเทศโดยใช้ Oracle Designer6i CASE Tools โดยใช้ระบบงานทะเบียนนักศึกษาเป็นตัวอย่างในการศึกษาและพัฒนา โดยจะเริ่มตั้งแต่การวิเคราะห์ การออกแบบ และการพัฒนาระบบ ซึ่งในบทนี้จะกล่าวถึงการวิเคราะห์และออกแบบระบบงาน ในส่วนการพัฒนาระบบงานนั้นจะกล่าวในบทต่อไป

3.1 ระบบทะเบียนนักศึกษา

เป็นระบบที่สำนักทะเบียนและประมวลผลใช้ในการทำงานเกี่ยวกับข้อมูลทางด้านการศึกษาภายในสถานศึกษา ซึ่งในระบบทะเบียนนักศึกษแบ่งการทำงานเป็น 7 ส่วนดังนี้

3.1.1 ส่วนข้อมูลคณะและภาควิชา

เป็นส่วนที่เกี่ยวข้องกับการจัดการรายละเอียดข้อมูลของคณะ ภาควิชา และอาจารย์ โดยฝ่ายทะเบียนของแต่ละคณะ มีหน้าที่ดังนี้

- กำหนดตารางเรียนและตารางสอบ จากนั้นจะแจ้งให้ส่วนการลงทะเบียนทราบ
- เป็นส่วนที่ติดต่อกับอาจารย์ผู้สอน โดยตรง
- แจ้งผลการเรียนที่ได้รับจากอาจารย์ผู้สอนให้ส่วนการประมวลผล
- แต่งตั้งอาจารย์ที่ปรึกษาให้แก่นักศึกษา

3.1.2 ส่วนข้อมูลหลักสูตร หมวดวิชา รายวิชา

ในส่วนนี้จะเกี่ยวข้องกับการจัดการรายละเอียดข้อมูลของหลักสูตร, หมวดวิชา, รายวิชา, อาจารย์ผู้สอน, และจำนวนหน่วยกิตที่บังคับของหลักสูตร

โดยข้อมูลของรายวิชาประกอบด้วย รายวิชาที่อยู่ในหลักสูตร, รายวิชาที่เปิดสอน และรายวิชาต่อเนื่อง (Prerequisite)

3.1.3 ส่วนข้อมูลนักศึกษา

ในส่วนนี้จะเป็นการจัดการเกี่ยวกับรายละเอียดข้อมูลของนักศึกษา ทั้งข้อมูลส่วนตัว เช่น ชื่อ นามสกุล ที่อยู่ หมายเลขโทรศัพท์ วันเดือนปีเกิด เป็นต้น และข้อมูลที่เกี่ยวข้องกับการศึกษา เช่น

ไม่ว่าการฉ้อโกงทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วันที่เริ่มเข้าเรียน วันที่สำเร็จการศึกษา หน่วยกิตสะสม ผลการเรียนเฉลี่ยสะสม สถานภาพทางการศึกษา เป็นต้น

นอกจากนั้นยังมีส่วนของการจัดการเอกสารที่เกี่ยวข้องกับนักศึกษา เช่น เอกสารแต่งตั้งอาจารย์ที่ปรึกษา, เอกสารอนุมัติลาพักการศึกษา เป็นต้น

3.1.4 ส่วนการลงทะเบียน

ในส่วนนี้จะประกอบด้วย การลงทะเบียนปกติ, การลงทะเบียนเพิ่มวิชา, การลงทะเบียนถอนวิชา, การรักษาสถานภาพการศึกษา และการลาพักการศึกษา

ในการลงทะเบียนปกติ จะมีขั้นตอนการตรวจสอบ ดังนี้

- ตรวจสอบสิทธิ์ในการลงทะเบียนของนักศึกษา ได้แก่ สถานภาพทางการศึกษา คือ จะต้องเป็นนักศึกษาที่มีสถานภาพปกติ หรือรอพินิจ ไม่เป็นนักศึกษาที่พ้นสภาพไปแล้ว , สถานภาพการชำระเงิน คือ จะต้องไม่ค้างชำระเงินกับสถานศึกษา และสถานภาพการยืมหนังสือห้องสมุด คือ จะต้องไม่ค้างส่งหนังสือห้องสมุด
- ตรวจสอบรายวิชาที่จะลงทะเบียนว่ามีการเปิดสอนหรือไม่ และมีการลงทะเบียนซ้ำกับภาคการศึกษาที่ผ่านมาหรือไม่ ถ้าซ้ำก็จะต้องตรวจสอบว่าในการลงทะเบียนครั้งที่ผ่านมา นักศึกษาได้ผลการเรียนในรายวิชานั้นสูงกว่ากำหนดหรือไม่ จากนั้นจะตรวจสอบวิชาต่อเนื่อง (Prerequisite) ว่าวิชาที่จะลงทะเบียนนั้นต้องการวิชาต่อเนื่องหรือไม่ ถ้าต้องการจะต้องตรวจสอบต่อว่าในการลงทะเบียนครั้งที่ผ่านมาได้ลงทะเบียนวิชาต่อเนื่องครบทุกวิชา

- ตรวจสอบจำนวนหน่วยกิตรวมที่ทำการลงทะเบียนว่าเป็นไปตามที่กำหนดหรือไม่

ในการลงทะเบียนเพิ่มวิชา จะมีขั้นตอนการตรวจสอบ ดังนี้

- ตรวจสอบว่ามีการลงทะเบียนปกติมาก่อนหรือไม่
- ตรวจสอบสถานภาพการชำระเงิน คือ จะต้องไม่ค้างชำระเงินในการลงทะเบียนครั้งที่ผ่านมา
- ตรวจสอบข้อมูลรายวิชาที่ลงทะเบียนเพิ่ม ซึ่งจะมีขั้นตอนการตรวจสอบเช่นเดียวกับการลงทะเบียนปกติ

ในการลงทะเบียนถอนวิชา จะมีขั้นตอนการตรวจสอบ ดังนี้

- ตรวจสอบว่าวิชาที่จะถอนนั้นมีการลงทะเบียนมาก่อนหรือไม่
- ตรวจสอบสถานภาพการชำระเงิน คือ จะต้องไม่ค้างชำระเงินในการลงทะเบียนครั้งที่ผ่านมา

ผ่านมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งการลงทะเบียนถอนวิชาจะแบ่งออกเป็นช่วงการถอนวิชาภายในกำหนด และช่วงการถอนวิชาภายหลังกำหนด ซึ่งจะมีผลต่อการบันทึกข้อมูลการลงทะเบียนต่างกัน

- การถอนวิชาภายในกำหนด จะมีผลให้ลบรายวิชานั้นๆออกจากข้อมูลการลงทะเบียน
- การถอนวิชาภายหลังกำหนด จะไม่ลบรายวิชานั้นๆออกจากข้อมูลการลงทะเบียน แต่ผลการเรียนจะเป็น W (Withdrawn) แต่จะไม่ถูกนำมาคิดเกรดรวมกับวิชาอื่นๆ

3.1.5 ส่วนการเงิน

ในส่วนนี้จะเป็นการจัดการเกี่ยวกับรายละเอียดค่าใช้จ่ายต่างๆ ทั้งส่วนที่เกี่ยวข้องกับการลงทะเบียน และส่วนอื่นๆ ซึ่งจะเป็นค่าใช้จ่ายในการออกเอกสารตามใบคำร้อง เช่น ใบรับรองผลการเรียน (Transcript), ใบแจ้งผลการเรียน, ใบรับรองต่างๆ เป็นต้น

สำหรับค่าใช้จ่ายในการลงทะเบียน ส่วนการเงินจะนำข้อมูลจากส่วนการลงทะเบียนมาทำการคำนวณค่าใช้จ่ายที่นักศึกษาจะต้องชำระ โดยคิดจากหน่วยกิตที่ลงทะเบียนรวมกับค่าธรรมเนียมต่างๆ แล้วพิมพ์ใบชำระเงินให้นักศึกษาไปชำระเงินที่ฝ่ายการเงิน

3.1.6 ส่วนการประมวลผล

ในส่วนการประมวลผล จะแบ่งการทำงานออกเป็น 2 ส่วน คือ การบันทึกผลการเรียน และการปรับสถานภาพของนักศึกษา โดยสถานภาพของนักศึกษา จะแบ่งออกเป็น สถานภาพการศึกษา, สถานภาพการชำระเงิน และสถานภาพการยืมหนังสือห้องสมุด

ในการบันทึกผลการเรียน จะรับข้อมูลผลการเรียนจากฝ่ายทะเบียนคณะ ซึ่งทางฝ่ายทะเบียนคณะจะได้รับผลการเรียนในแต่ละรายวิชาจากอาจารย์ผู้สอน และในการบันทึกผลการเรียนอาจจะต้องมีการปรับเปลี่ยนแก้ไขผลการเรียน กรณีที่นักศึกษาได้ผลการเรียนเป็น I ซึ่งเมื่อทางอาจารย์ผู้สอนอนุมัติเปลี่ยนแปลงผลการเรียนแล้ว จะแจ้งให้ฝ่ายทะเบียนคณะทราบ จากนั้นจะส่งผลการเรียนให้ส่วนการประมวลผลเพื่อทำการแก้ไขและบันทึกผลการเรียนใหม่

ในการบันทึกผลการเรียนสำหรับนักศึกษาระดับปริญญาโท-เอก ถ้ารายวิชาใดได้ผลการเรียนต่ำกว่า C+ ถือว่าต้องลงทะเบียนรายวิชานั้นใหม่

สำหรับการปรับสถานภาพของนักศึกษาจะแบ่งเป็น 3 ส่วน ดังนี้

1) การปรับสถานภาพการศึกษา จะพิจารณาจากผลการเรียนเฉลี่ยสะสม โดยแบ่งการพิจารณาเป็น 2 ระดับ คือ ระดับปริญญาตรี และระดับปริญญาโท-เอก

● ระดับปริญญาตรี

- ถ้าผลการเรียนเฉลี่ยสะสมสูงกว่า 2.00 ให้ปรับสถานภาพเป็น “ปกติ (Normal)”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ถ้าผลการเรียนเฉลี่ยสะสมต่ำกว่า 1.00 ให้ปรับสถานภาพเป็น “พ้นสภาพ (Retired)”
- ถ้าผลการเรียนเฉลี่ยสะสมต่ำกว่า 2.00 แต่สูงกว่า 1.00 ให้ปรับสถานภาพเป็น “รอพินิจ (Probation)”
- ถ้าสถานภาพปัจจุบันเป็น “รอพินิจ (Probation)” และผลการเรียนเฉลี่ยประจำภาคต่ำกว่า 2.00 ให้ปรับสถานภาพเป็น “พ้นสภาพ (Retired)”
- ถ้าสถานภาพปัจจุบันเป็น “รอพินิจ (Probation)” และผลการเรียนเฉลี่ยประจำภาคสูงกว่า 2.00 แต่ผลการเรียนเฉลี่ยสะสมยังคงต่ำกว่า 2.00 ให้ปรับสถานภาพเป็น “รอพินิจ (Probation)”
- **ระดับปริญญาโท-เอก**
 - ถ้าผลการเรียนเฉลี่ยสะสมสูงกว่า 3.00 ให้ปรับสถานภาพเป็น “ปกติ (Normal)”
 - ถ้าผลการเรียนเฉลี่ยสะสมต่ำกว่า 2.50 ให้ปรับสถานภาพเป็น “พ้นสภาพ (Retired)”
 - ถ้าผลการเรียนเฉลี่ยสะสมต่ำกว่า 3.00 แต่สูงกว่า 2.50 ให้ปรับสถานภาพเป็น “รอพินิจ (Probation)”
 - ถ้าสถานภาพปัจจุบันเป็น “รอพินิจ (Probation)” และผลการเรียนเฉลี่ยประจำภาคต่ำกว่า 3.00 ให้ปรับสถานภาพเป็น “พ้นสภาพ (Retired)”
 - ถ้าสถานภาพปัจจุบันเป็น “รอพินิจ (Probation)” และผลการเรียนเฉลี่ยประจำภาคสูงกว่า 3.00 แต่ผลการเรียนเฉลี่ยสะสมยังคงต่ำกว่า 3.00 ให้ปรับสถานภาพเป็น “รอพินิจ (Probation)”

2) การปรับสถานภาพการชำระเงิน

ในส่วนนี้จะได้รับข้อมูลจากฝ่ายการเงิน โดยทางฝ่ายการเงินจะส่งรายชื่อนักศึกษาที่ค้างชำระเงินมาให้ จากนั้นก็จะทำการปรับสถานภาพการชำระเงินของนักศึกษาเป็นค้างชำระเงิน

3) การปรับสถานภาพการยืมหนังสือห้องสมุด

ในส่วนนี้จะได้รับข้อมูลจากห้องสมุด โดยห้องสมุดจะส่งรายชื่อนักศึกษาที่ค้างส่งหนังสือห้องสมุดมาให้ จากนั้นก็จะทำการปรับสถานภาพการยืมหนังสือห้องสมุดเป็นค้างส่งหนังสือ

3.1.7 ส่วนการออกรายงาน

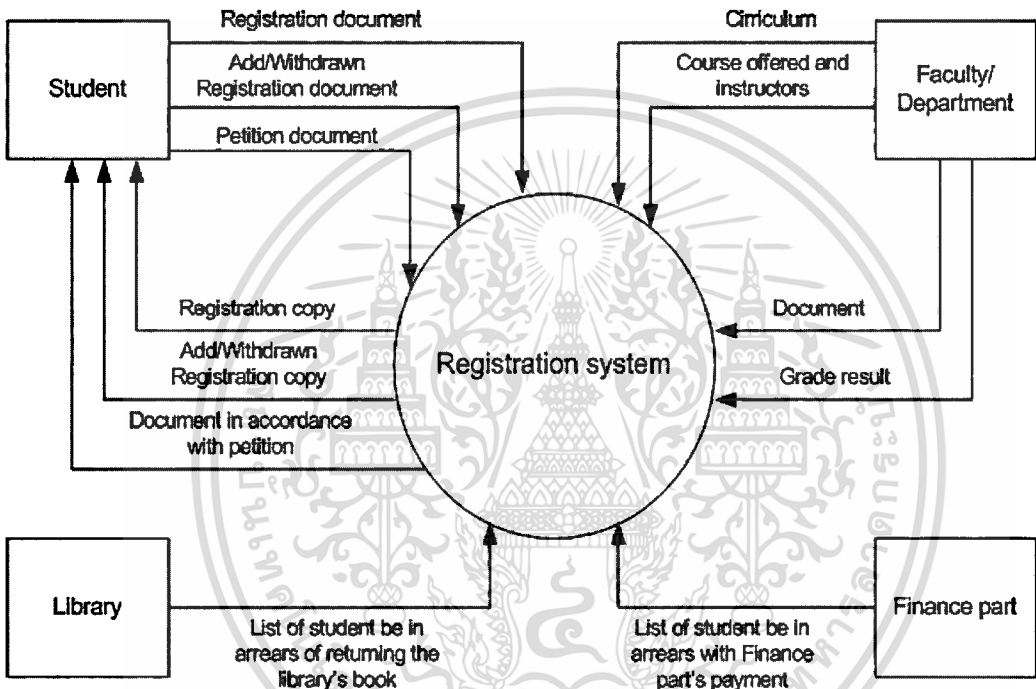
ในส่วนนี้จะออกรายงานที่เกี่ยวข้องกับทะเบียนนักศึกษาในส่วนต่างๆ เช่น

- รายงานแสดงรายชื่อนักศึกษาที่มีสิทธิ์เข้าเรียนในแต่ละรายวิชา
- รายงานแสดงรายชื่อนักศึกษาที่มีสิทธิ์เข้าสอบ
- เอกสารตามใบคำร้อง เช่น ใบรับรองผลการเรียน (Transcript) ใบแจ้งผลการเรียน ใบรับรองต่างๆ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 Context Diagram ของระบบทะเบียนนักศึกษา

จากการศึกษาขั้นตอนการทำงานของระบบทะเบียนนักศึกษา ได้ทำการสรุปการทำงานออกมาโดยใช้ Dataflow Diagram เพื่อช่วยให้สามารถเข้าใจการทำงานของระบบได้ดียิ่งขึ้น เนื่องจากเป็นแผนภาพที่ใช้แสดงการไหลของข้อมูลทั้งระบบ ซึ่ง Context Diagram ของระบบทะเบียนนักศึกษาแสดงดังรูปที่ 3.1



รูปที่ 3.1 Context Diagram ของระบบทะเบียนนักศึกษา

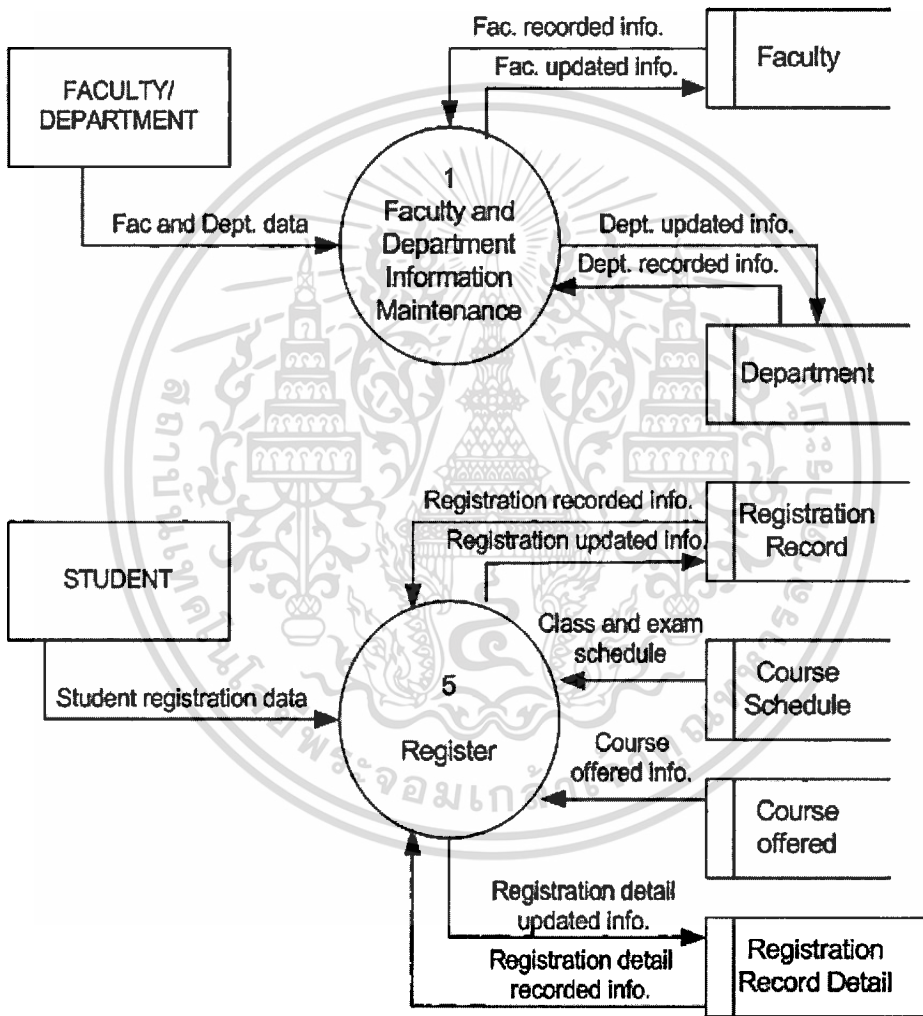
3.3 Dataflow Diagram ของระบบทะเบียนนักศึกษา

จากแผนภาพ Context Diagram แสดงให้เห็นการทำงานโดยรวมของระบบทะเบียนนักศึกษา ในส่วนของ Dataflow Diagram จะแสดงให้เห็นการไหลของข้อมูลและขั้นตอนการทำงานของระบบในรายละเอียด แสดงให้เห็นว่ากิจกรรมแต่ละอย่างในงานลงทะเบียนนั้น มีขั้นตอนการทำงานอย่างไร

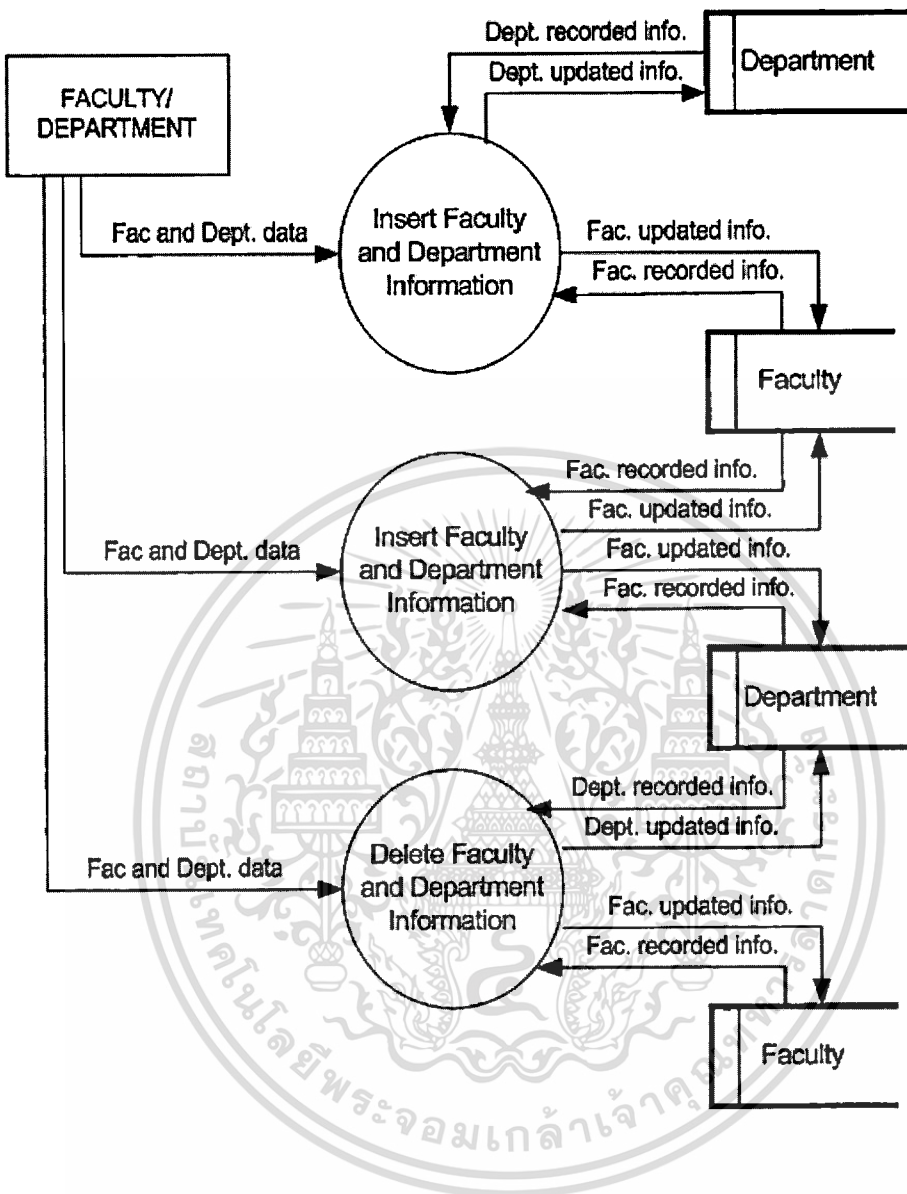
สำหรับ Dataflow Diagram นั้นสามารถแสดงระดับของข้อมูลในระดับย่อยๆ ลงไปได้หลายระดับ ซึ่งในการสร้าง Dataflow Diagram ของระบบทะเบียนนักศึกษาในส่วนข้อมูลคณะและภาควิชา และส่วนการลงทะเบียนนี้ ได้แสดงระดับย่อยไว้ 2 ระดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

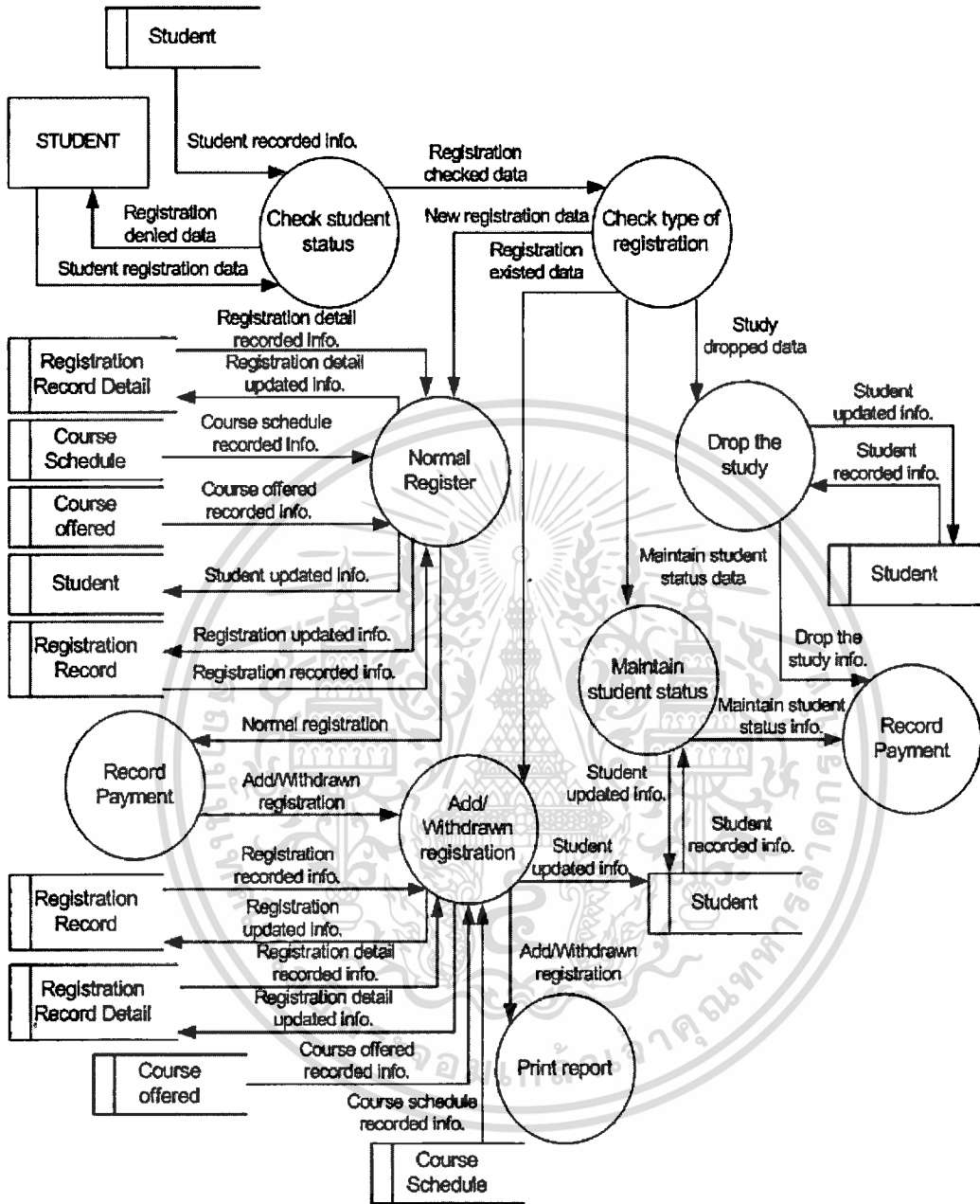
โดย Dataflow Diagram Level-1 จะแสดง Dataflow Diagram ของระบบทะเบียนนักศึกษา ในส่วนข้อมูลคณะและภาควิชา และส่วนการลงทะเบียน ดังรูปที่ 3.2 ส่วนในรูปที่ 3.3 ถึงรูปที่ 3.4 จะเป็น Dataflow Diagram Level-2 ซึ่งแสดงให้เห็นถึงรายละเอียดการทำงานของทั้ง 2 ส่วนข้างต้น โดยในรูปที่ 3.3 จะเป็น Dataflow Diagram Level-2 ของส่วนข้อมูลคณะและภาควิชา และในรูปที่ 3.4 จะเป็น Dataflow Diagram Level-2 ของส่วนการลงทะเบียน



รูปที่ 3.2 Dataflow Diagram Level-1 ของระบบทะเบียนนักศึกษา
ในส่วนข้อมูลคณะและภาควิชา และส่วนการลงทะเบียน



รูปที่ 3.3 Dataflow Diagram Level-2 ของส่วนข้อมูลคณะและภาควิชา



รูปที่ 3.4 Dataflow Diagram Level-2 ของส่วนการลงทะเบียน

3.4 การวิเคราะห์และออกแบบฐานข้อมูล

การออกแบบฐานข้อมูลเป็นการรวบรวมข้อมูลที่เกี่ยวข้องกับระบบทั้งหมดเอาไว้แล้ว

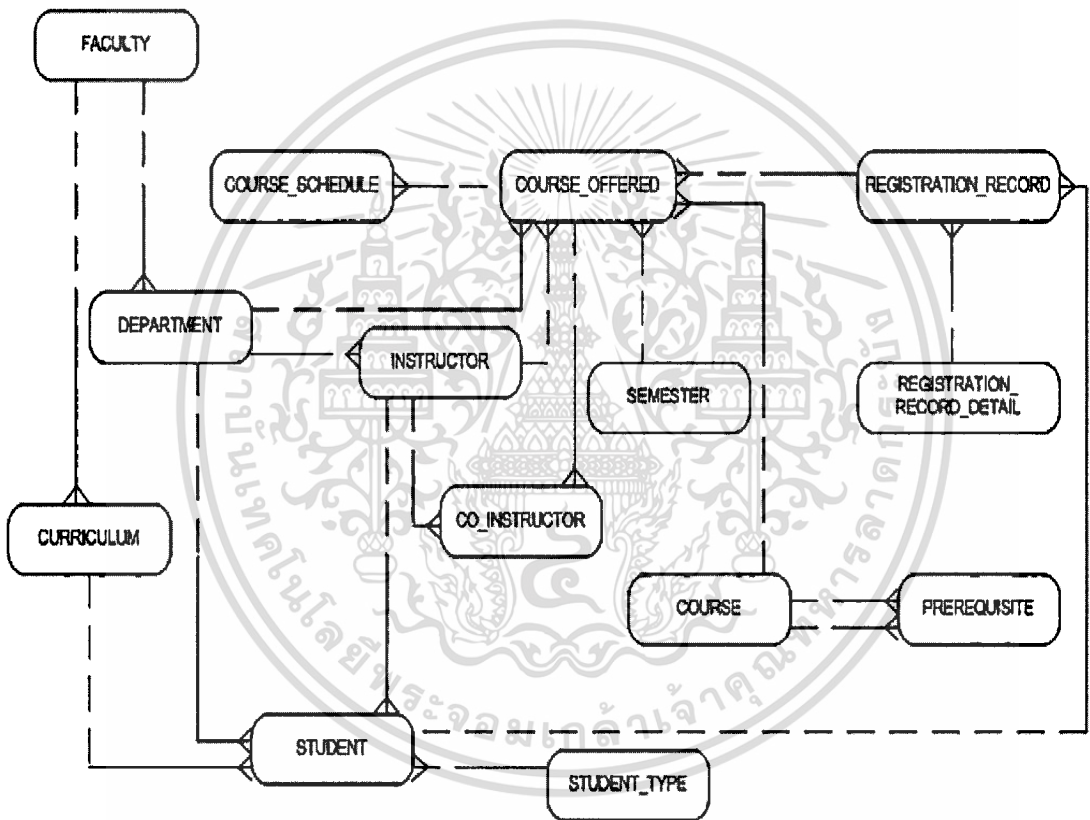
นำมาพิจารณาว่าควรจะจัดเก็บอย่างไร จะมี Entity อะไรบ้าง โดยแสดงออกมาในรูปของ Table

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และ Column โดยเขียนเป็นแบบจำลองแสดงความสัมพันธ์ของข้อมูล (Entity Relationship Diagram) เพื่อแสดงให้เห็นความสัมพันธ์ของข้อมูลแต่ละ Table รวมถึงการระบุขอบเขตของข้อมูล (Domains) และเงื่อนไขการตรวจสอบความถูกต้องของข้อมูล (Constraints) ของแต่ละ Column ด้วย

ข้อมูลที่เกี่ยวข้องกับระบบทะเบียนนักศึกษาในส่วนข้อมูลคณะและภาควิชา และส่วนการลงทะเบียน สามารถแสดงความสัมพันธ์ออกมาเป็น Entity Relationship Diagram ดังรูปที่ 3.5



รูปที่ 3.5 Entity Relationship Diagram ของระบบทะเบียนนักศึกษา
ในส่วนข้อมูลคณะและภาควิชา และส่วนการลงทะเบียน

- รายละเอียดของฐานข้อมูลของระบบ

จะเป็นรายละเอียดของตารางที่เกี่ยวข้องกับส่วนข้อมูลคณะและภาควิชา และส่วนการลงทะเบียน ในฐานข้อมูล ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FACULTY ข้อมูลคณะ					
Seq.	Key	Name	Data Type	Len.	Comment
1	PK	FAC_ID	Varchar2	6	Faculty Id
2		FAC_NAME_E	Varchar2	40	Faculty Name in English
3		FAC_NAME_T	Varchar2	60	Faculty Name in Thai

ตารางที่ 3.1 ตาราง FACULTY

DEPARTMENT ข้อมูลภาควิชา					
Seq.	Key	Name	Data Type	Len.	Comment
1	PK	DEPT_FAC_ID	Varchar2	6	Faculty Id
2	PK	DEPT_ID	Varchar2	6	Department Id
3		DEPT_NAME_E	Varchar2	40	Department Name in English
4		DEPT_NAME_T	Varchar2	60	Department Name in Thai

ตารางที่ 3.2 ตาราง DEPARTMENT

STUDENT ข้อมูลนักศึกษา					
Seq.	Key	Name	Data Type	Len.	Comment
1	PK	ST_ID	Varchar2	15	Faculty Id
2	FK	ST_FAC_ID	Varchar2	6	Faculty Id
3	FK	ST_DEPT_ID	Varchar2	6	Department Id
4	FK	ST_CURR_ID	Varchar2	6	Curriculum Id
5	FK	ST_TYPE_ID	Varchar2	2	Student Type Id
6	FK	ST_INST_ID	Varchar2	10	Instructor Id
7		ST_TTITLE	Varchar2	10	Student Title
8		ST_F_NAME_T	Varchar2	30	Student First name in Thai

เอกสารนี้เป็นเอกสารราชการใช้สำหรับการศึกษาระดับปริญญาตรี เอกสารนี้ได้รับการจัดเก็บไว้ในระบบสารสนเทศของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9	ST_L_NAME_T	Varchar2	30	Student Last name in Thai
10	ST_GENDER	Varchar2	2	Gender
11	ST_DOB	DATE		Date of Birth
12	ST_E_MAIL	Varchar2	30	E-mail Address
13	ST_ADM_DATE	DATE		Date of Admission
14	ST_GRAD_DATE	DATE		Date of Graduated
15	ST_ACC_CR_A	NUMBER	(3,0)	Academic Credit Acquired
16	ST_ACC_CR_E	NUMBER	(3,0)	Academic Credit Entry
17	ST_GPA	NUMBER	(4,3)	GPA
18	ST_ACC_STATUS	CHAR	2	Academic Status
19	ST_FIN_STATUS	CHAR	1	Financial Status
20	ST_LIB_STATUS	CHAR	1	Library Status

ตารางที่ 3.3 ตาราง STUDENT

STUDENT_TYPES ข้อมูลประเภทนักศึกษา					
Seq.	Key	Name	Data Type	Len.	Comment
1	PK	ST_TYPE_ID	Varchar2	2	Student Types Id
2	PK	ST_TYPE_NAME	Varchar2	20	Student Types Name

ตารางที่ 3.4 ตาราง STUDENT_TYPES

INSTRUCTOR ข้อมูลอาจารย์					
Seq.	Key	Name	Data Type	Len.	Comment
1	PK	INST_ID	Varchar2	10	Instructor Id
2		INST_NAME_T	Varchar2	40	Instructor Name in Thai
3		INST_NAME_E	Varchar2	40	Instructor Name in English

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4	FK	INST_FAC_ID	Varchar2	6	Faculty Id
5	FK	INST_DEPT_ID	Varchar2	6	Department Id

ตารางที่ 3.5 ตาราง INSTRUCTOR

CURRICULUM ข้อมูลหลักสูตร					
Seq.	Key	Name	Data Type	Len.	Comment
1	PK	CURR_ID	Varchar2	6	Curriculum Id
2		CURR_FAC_ID	Varchar2	6	Faculty Id
3		CURR_NAME_T	Varchar2	60	Curriculum Name in Thai
4		CURR_NAME_E	Varchar2	40	Curriculum Name in English
5		CURR_DEG_NAME_T	Varchar2	60	Degree Name in Thai
6		CURR_DEG_NAME_E	Varchar2	40	Degree Name in English
7		CURR_LEVEL	CHAR	1	Curriculum Level

ตารางที่ 3.6 ตาราง CURRICULUM

COURSE ข้อมูลวิชา					
Seq.	Key	Name	Data Type	Len.	Comment
1	PK	C_ID	Varchar2	12	Course Id
2		C_TITLE_T	Varchar2	60	Course Title in English
3		C_TITLE_E	Varchar2	40	Course Title in Thai
4		C_CR	NUMBER	(2,0)	Course Credit
5		C_HR_LLECT	NUMBER	(2,0)	Course Hour Lecture
6		C_HR_LAB	NUMBER	(2,0)	Course Hour Lab
7	FK	C_GC_ID	Varchar2	2	Grade Category Id

ตารางที่ 3.7 ตาราง COURSE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PREREQUISITE ข้อมูลวิชาต่อเนื่อง					
Seq.	Key	Name	Data Type	Len.	Comment
1	PK	PRE_R_ID	NUMBER	(10,0)	Prerequisite NO.
2	FK	PRE_R_C_ID	Varchar2	12	Course Id
3	FK	PRE_R_REQ_C_ID	Varchar2	12	Prerequisite Id

ตารางที่ 3.8 ตาราง PREREQUISITE

COURSE_OFFERED ข้อมูลวิชาที่เปิดสอน					
Seq.	Key	Name	Data Type	Len.	Comment
1	PK	CO_YEAR	NUMBER	(4,0)	Year
2	PK	CO_SEMESTER	NUMBER	(2,0)	Semester
3	PK	CO_C_ID	Varchar2	12	Course Id
4	PK	CO_SECTION	NUMBER	(2,0)	Section
5		CO_LIMIT	NUMBER	(2,0)	Student Amount Limit
6	FK	CO_FAC_ID	Varchar2	6	Faculty Id
7	FK	CO_DEPT_ID	Varchar2	6	Department Id
8	FK	CO_INST_ID	Varchar2	10	Instructor Id

ตารางที่ 3.9 ตาราง COURSE_OFFERED

COURSE_SCHEDULE ข้อมูลวิชาที่เปิดสอน					
Seq.	Key	Name	Data Type	Len.	Comment
1	PK	CO_YEAR	NUMBER	(4,0)	Year
2	PK	CO_SEMESTER	NUMBER	(2,0)	Semester
3	PK	CO_C_ID	Varchar2	12	Course Id
4	PK	CO_SECTION	NUMBER	(2,0)	Section

เอกสารนี้เป็นเอกสารประกอบการใช้งานเท่านั้น ไม่สามารถนำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5		CS_DAY	DATE		Day of Course Offered
6		CS_TIME_START	TIME		Start Time of Course Offered
7		CS_TIME_END	TIME		Finished Time of Course Offered
8		CS_MID_DATE	DATE		Midterm Exam Date
9		CS_MID_TIME_START	TIME		Midterm Exam Start Time
10		CS_MID_TIME_END	TIME		Midterm Exam Finished Time
11		CS_FINAL_DATE	DATE		Final Exam Date
12		CS_FINAL_TIME_START	TIME		Final Exam Start Time
13		CS_FINAL_TIME_END	TIME		Finished Time of Final Exam

ตารางที่ 3.10 ตาราง COURSE SCHEDULE

SEMESTER ข้อมูลวิชาที่เปิดสอน					
Seq.	Key	Name	Data Type	Len.	Comment
1	PK	CO_YEAR	NUMBER	(4,0)	Year
2	PK	CO_SEMESTER	NUMBER	(2,0)	Semester
3	PK	SE_START	DATE		Start Date of Semester
4	PK	SE_END	DATE		End Date of Semester
5		SE_ADD_LIMIT	DATE		Due Date of Add Course
6	FK	SE_W_LIMIT	DATE		Due Date of Withdrawn
7	FK	SE_DROP_LIMIT	DATE		Due Date of Drop Course

ตารางที่ 3.11 ตาราง SEMESTER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

REGISTRATION_RECORD ข้อมูลการลงทะเบียนเรียน					
Seq.	Key	Name	Data Type	Len.	Comment
1	PK	CO_YEAR	NUMBER	(4,0)	Year
2	PK	CO_SEMESTER	NUMBER	(2,0)	Semester
3	PK	CO_C_ID	Varchar2	12	Course Id
4	PK	CO_SECTION	NUMBER	(2,0)	Section
5	FK	RR_ST_ID	Varchar2	15	Student Id
6		RR_CR	NUMBER	(2,0)	Credit of Course
7		RR_GRADE	Varchar2	2	Grade
8		RR_DATE	DATE		Registration Date
9		RR_TX	Varchar2	1	Transfer Course Mark
10		RR_USER	Varchar2	15	Operator Name
11		RR_RD2_ID	NUMBER	(10,0)	Receipt No.

ตารางที่ 3.12 ตาราง REGISTRATION_RECORD

REGISTRATION_RECORD_DETAIL ข้อมูลรายละเอียดเพิ่มเติมจากการลงทะเบียนเรียน					
Seq.	Key	Name	Data Type	Len.	Comment
1	PK	RR_D_DETAIL_ID	NUMBER	(10,0)	Registration Record ID
2	FK	CO_YEAR	NUMBER	(4,0)	Year
3	FK	CO_SEMESTER	NUMBER	(2,0)	Semester
4	FK	CO_C_ID	Varchar2	12	Course Id
5	FK	CO_SECTION	NUMBER	(2,0)	Section
6	FK	RR_ST_ID	Varchar2	15	Student Id
7		RR_D_DETAIL	Varchar2	40	Registration Record Detail

ตารางที่ 3.13 ตาราง REGISTRATION_RECORD_DETAIL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การพัฒนาระบบงาน

ในการออกแบบระบบงานทะเบียนนักศึกษาโดยใช้เครื่องมือของ Oracle Designer6i นั้น จะต้องวิเคราะห์ ออกแบบ และพัฒนาระบบงานใน 2 ส่วนหลัก คือ

- การวิเคราะห์และออกแบบระบบฐานข้อมูล (Database System)
- การออกแบบโปรแกรมประยุกต์ใช้งาน (Application Program)

โดยจะต้องทำการออกแบบและพัฒนาในส่วนของระบบฐานข้อมูลก่อน เพื่อที่จะสามารถระบุถึงข้อมูลที่โปรแกรมประยุกต์ใช้งานจะนำไปใช้ได้

สำหรับขั้นตอนการพัฒนาระบบที่จะกล่าวต่อไปนั้น จะไม่แสดงรายละเอียดของวิธีการใช้งานเครื่องมือต่างๆของ Oracle Designer6i ที่ใช้ในการออกแบบโคอะแกรมต่างๆ ได้แก่ Entity Relationship Diagram, Dataflow Diagram, Function Hierarchy Diagram ซึ่งเทคนิคในการออกแบบโคอะแกรมเหล่านี้สามารถศึกษาได้ในบทที่ 2

4.1 การพัฒนาระบบส่วนฐานข้อมูล

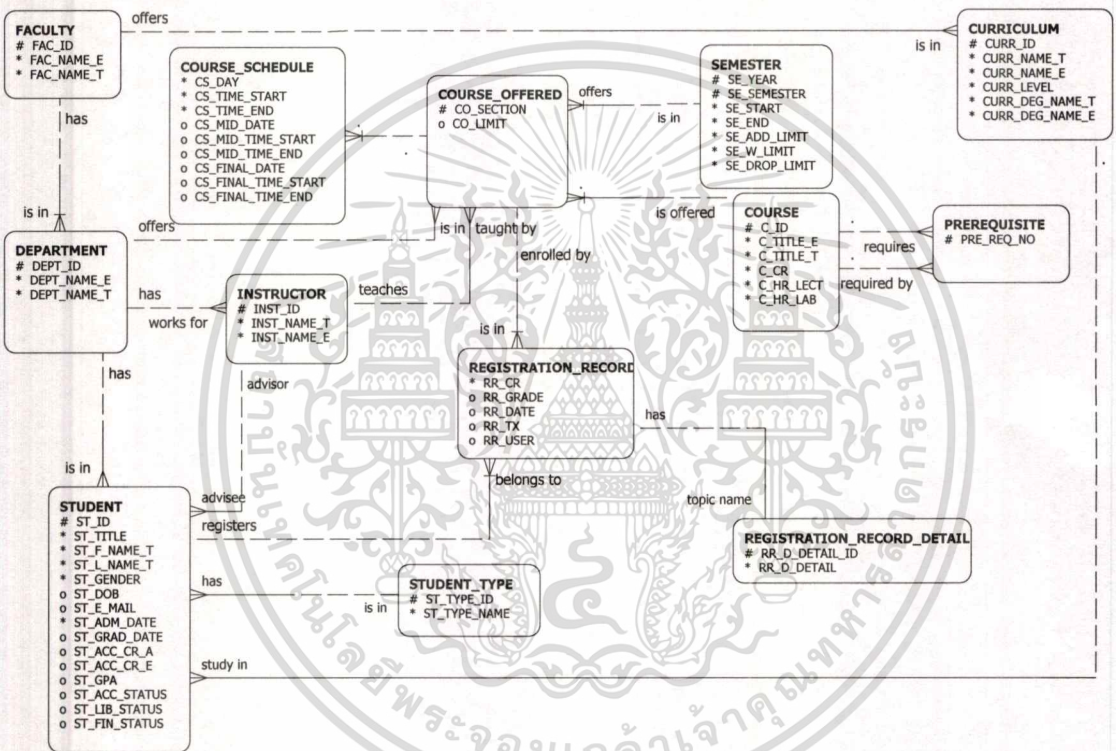
ในการพัฒนาระบบส่วนฐานข้อมูลนี้ จะนำข้อมูลที่เกี่ยวข้องทั้งหมดที่ได้ออกแบบไว้ในบทที่แล้วมาเขียนลงในเครื่องมือ Entity Relationship Diagrammer โดยจะบันทึก Entity, Attribute, Relationship, และข้อมูลอื่นๆที่เกี่ยวข้องในรูปของ Entity Relationship Diagram จากนั้นนำ Entity Relationship Diagram ที่ได้ไปแปลงให้อยู่ในรูปแบบที่เรียกว่า First-cut Data Design โดยใช้ Utility ที่เรียกว่า Database Design Transformer ซึ่งจะทำการแปลง Data Model ที่ออกแบบไว้ให้อยู่ในรูปแบบของ Database Design โดย Entity, Attribute, Unique Identifier (UID), Relationship จะถูกแปลงเป็น Table, Column, Primary Key, Foreign Key และ Constraint ต่างๆ ตามลำดับ จากนั้นนำผลลัพธ์ที่ได้มาสร้างเป็น Server Model Diagram โดยใช้ Utility ที่เรียกว่า Design Editor เพื่อให้ง่ายแก่การทำ ความเข้าใจและปรับแต่งข้อมูลก่อนการนำไปสร้างเป็นฐานข้อมูลจริง เช่น เพิ่มหรือแก้ไขคอลัมน์ต่างๆได้ และเมื่อปรับแต่งจนเป็นที่เรียบร้อยแล้ว ขั้นตอนถัดมาคือ การสร้างฐานข้อมูล โดยใช้ฟังก์ชัน Generate Server Model Definition ซึ่งอาจจะสร้างเป็น Table จริงๆเก็บอยู่ในฐานข้อมูล หรืออาจจะสร้างเป็น Script DDL File สำหรับนำไปสร้างเป็นฐานข้อมูลต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.1 การสร้าง Entity Relationship Diagram

เราจะนำฐานข้อมูลที่ออกแบบไว้มาเขียนลงในเครื่องมือ Entity Relationship Diagrammer ของ Oracle Designer 6i โดยการสร้าง Entity Relationship Diagram จะเกี่ยวข้องกับกระบวนการ Entity, Attribute, Relationship รวมถึงการกำหนด Domain และ Unique Identifier (UID)

สำหรับ Entity Relationship Diagram ของระบบทะเบียนนักศึกษาในส่วนข้อมูลคณะและภาควิชา และส่วนการลงทะเบียน แสดงดังรูปที่ 4.1



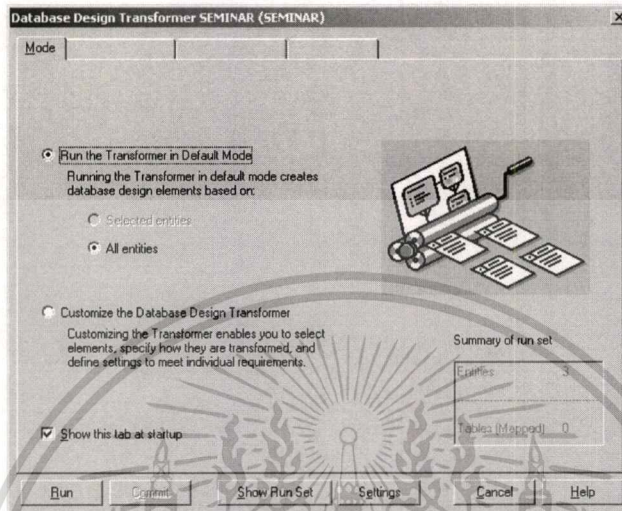
รูปที่ 4.1 Entity Relationship Diagram ของระบบทะเบียนนักศึกษา ในส่วนข้อมูลคณะและภาควิชา และส่วนการลงทะเบียน

4.1.2 การสร้าง First-cut Data Design

เมื่อเราได้ทำการออกแบบและสร้าง Entity Relationship Diagram เรียบร้อยแล้ว ก่อนที่จะสร้างฐานข้อมูลได้นั้น จะต้องแปลง Entity Relationship Diagram ให้อยู่ในรูปแบบที่เรียกว่า First-cut Data Design ก่อน โดยใช้ Utility ของ Oracle Designer6i ที่เรียกว่า Database Design Transformer ดังรูปที่ 4.2 ซึ่งจะเลือกแปลงทุก Entity หรือเฉพาะบาง Entity ก็ได้ นอกจากนี้สามารถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่จะตั้งค่าต่างๆได้ เช่น ขนาดของ Table Space, Constraint หรือ Index ต่างๆ โดยจะมีผลต่อ Object ต่างๆที่อยู่ใน Oracle Designer6i ดังตารางที่ 4.1



รูปที่ 4.2 Database Design Transformer

Data Model	Data Design
Entity	Table
UID	Primary key
Attribute	Column
Relationship	Foreign key column และ constraint

ตารางที่ 4.1 กระบวนการทำงานของ Database Design Transformer

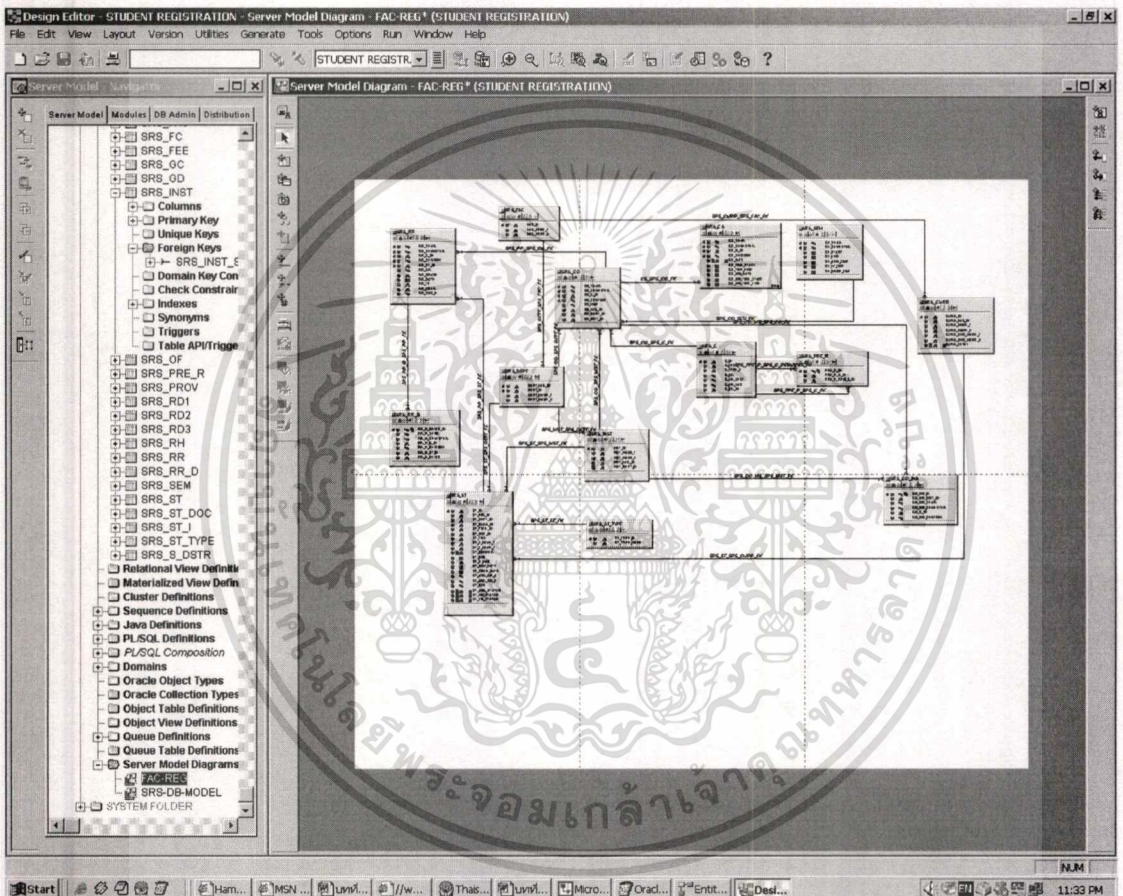
จากตารางที่ 4.1 เป็นการแสดงความสัมพันธ์ระหว่างข้อมูลก่อนแปลง (Data Model) กับ ข้อมูลหลังแปลง (Data Design) ซึ่งเป็นกระบวนการที่ใช้ในการแปลงโมเดลที่ออกแบบไว้ให้อยู่ในรูปที่พร้อมจะนำไปทำงานในขั้นตอนต่อไป

4.1.3 การสร้าง Server Model Diagram

เมื่อทำการแปลงข้อมูลจาก Entity Relationship Diagram ให้อยู่ในรูป First-cut Data Design แล้ว จากนั้นเราจะใช้ Utility ของ Oracle Designer6i ที่เรียกว่า Design Editor มาทำการเอกสารนี้เป็นเอกสารที่ส่งวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สร้าง Server Model Diagram เพื่อให้ง่ายต่อการทำความเข้าใจและปรับแต่งข้อมูลก่อนการนำไปสร้างเป็นฐานข้อมูลจริง เช่น เพิ่มหรือแก้ไขคอลัมน์ต่างๆได้

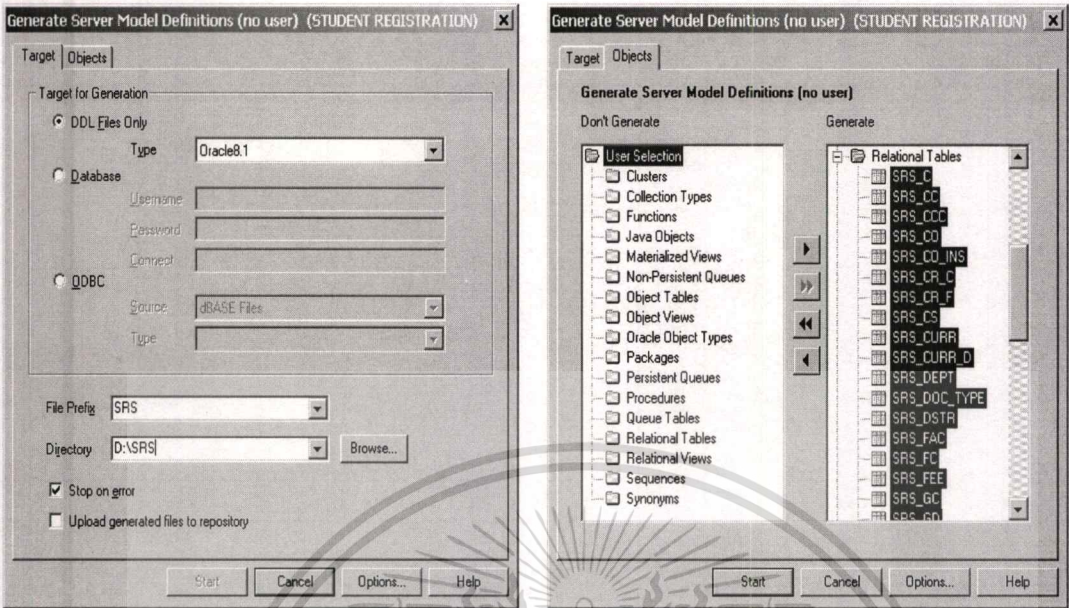
สำหรับ Server Model Diagram ของระบบทะเบียนนักศึกษาแสดงดังรูปที่ 4.3 โดยในหน้าต่างหลักของ Design Editor จะแบ่งเป็น 2 ส่วน หน้าต่างทางด้านซ้ายแสดง Server Model Navigator ส่วนหน้าต่างทางด้านขวาแสดง Server Model Diagram



รูปที่ 4.3 Server Model Diagram ของระบบทะเบียนนักศึกษา

4.1.4 การสร้างฐานข้อมูล

เมื่อเราได้ทำการปรับแต่งข้อมูลต่างๆใน Server Model Diagram เรียบร้อยแล้ว ขั้นตอนถัดมาคือ การสร้างฐานข้อมูล โดยเราจะใช้ฟังก์ชัน Generate Database form Server Model ซึ่งสามารถเลือกที่จะสร้างเป็น Table จริงๆเก็บอยู่ในฐานข้อมูล หรือจะสร้างเป็น Script DDL File สำหรับนำไปสร้างเป็นฐานข้อมูลต่อไป ดังรูปที่ 4.4 เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 ฟังก์ชัน Generate Database form Server Model

จากรูปที่ 4.4 แสดงการแปลง Object ต่างๆ ไปเป็น Database ซึ่งประกอบด้วย Relational Table ที่อยู่ใน Server Model Diagram ที่สร้างไว้ และ Object อื่นๆที่เกี่ยวข้อง หลังจาก Generate แล้วจะได้ Script File ที่เก็บคำสั่งในการสร้าง Database Object ต่างๆ เพื่อนำไปสร้างใน Database ต่อไป ตัวอย่าง Script File ที่อยู่ในรูปของ Database Definition Language แสดงดังรูปที่ 4.5

```

SRS_FAC.tab - Notepad
E:\BACKUP\SRS_FAC.tab
Generated for Oracle 8.1 on Wed Dec 25 23:45:23 2002 by Server Generator 6.5.52.1.0

PROMPT Creating Table 'SRS_ST_DOC'
CREATE TABLE SRS_ST_DOC
(ST_DOC_ID NUMBER(10,0) NOT NULL
,ST_DOC_TYPE_ID VARCHAR2(6) NOT NULL
,ST_DOC_ST_ID VARCHAR2(15) NOT NULL
,ST_DOC_DATE DATE
,ST_DOC_DATA BLOB NOT NULL
)
/

COMMENT ON COLUMN SRS_ST_DOC.ST_DOC_ID IS 'Document Number ID'
/

COMMENT ON COLUMN SRS_ST_DOC.ST_DOC_TYPE_ID IS 'Document Type Identifier'
/

COMMENT ON COLUMN SRS_ST_DOC.ST_DOC_ST_ID IS 'Student Identifier'
/

COMMENT ON COLUMN SRS_ST_DOC.ST_DOC_DATE IS 'Document Date'
/

COMMENT ON COLUMN SRS_ST_DOC.ST_DOC_DATA IS 'Document Data'
/

```

เอกสารนี้เป็นเอกสารรูปที่ 4.5 ตัวอย่าง Script DDL File ที่ได้จากการ Generate Database ใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.5 เป็นการแสดงตัวอย่าง Script File ที่ใช้สร้าง Table เมื่อนำ Script File นี้ไปรัน จะได้ Table ใน Database ที่เลือกไว้

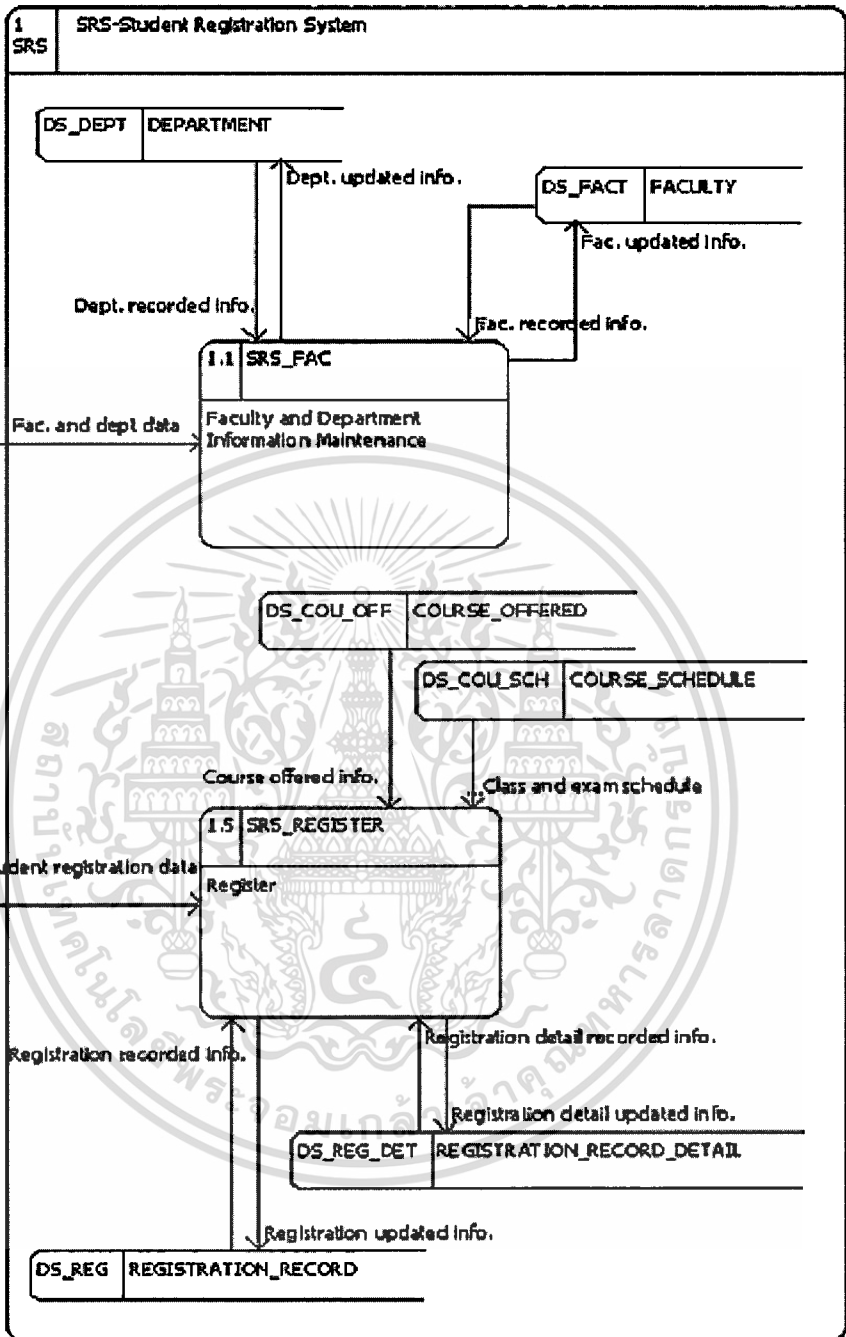
4.2 การพัฒนาระบบส่วนโปรแกรมประยุกต์ใช้งาน

ในการพัฒนาระบบงานส่วนโปรแกรมประยุกต์ใช้งาน จะสร้างกระบวนการทั้งหมดไว้ใน Dataflow Diagram เพื่อแสดงการไหลของข้อมูลทั้งหมดของระบบ โดยใช้เครื่องมือ Dataflow Diagrammer ของ Oracle Designer6i ซึ่งช่วยให้เราสามารถสร้างและปรับปรุงฟังก์ชันในระบบงาน, Datastore, Dataflow และสิ่งภายนอกอื่นๆ (External) ที่ถูกเก็บไว้ใน Repository

เมื่อสร้าง Dataflow Diagram เรียบร้อยแล้ว เราสามารถใช้เครื่องมือ Function Hierarchy Diagrammer ทำการสร้าง Function Hierarchy Diagram ให้โดยอัตโนมัติ จากนั้นเราจะแปลง Function Hierarchy Diagram ให้อยู่ในรูปของ Module โดยใช้ Utility ที่ชื่อว่า Application Design Transformer จะทำให้ได้ Module ที่จะนำไปสร้างเป็นโปรแกรมประยุกต์ใช้งานต่อไป ในส่วนของ Module ที่ได้มา จะต้องนำมาปรับแต่งเพิ่มเติมในรายละเอียดการทำงาน โดยใช้ Utility ที่ชื่อว่า Design Editor เพื่อให้มีความสมบูรณ์พร้อมนำไปใช้งานมากที่สุด ก่อนที่จะนำมาสร้างเป็นโปรแกรมประยุกต์ใช้งานโดยใช้ฟังก์ชัน Generate Module

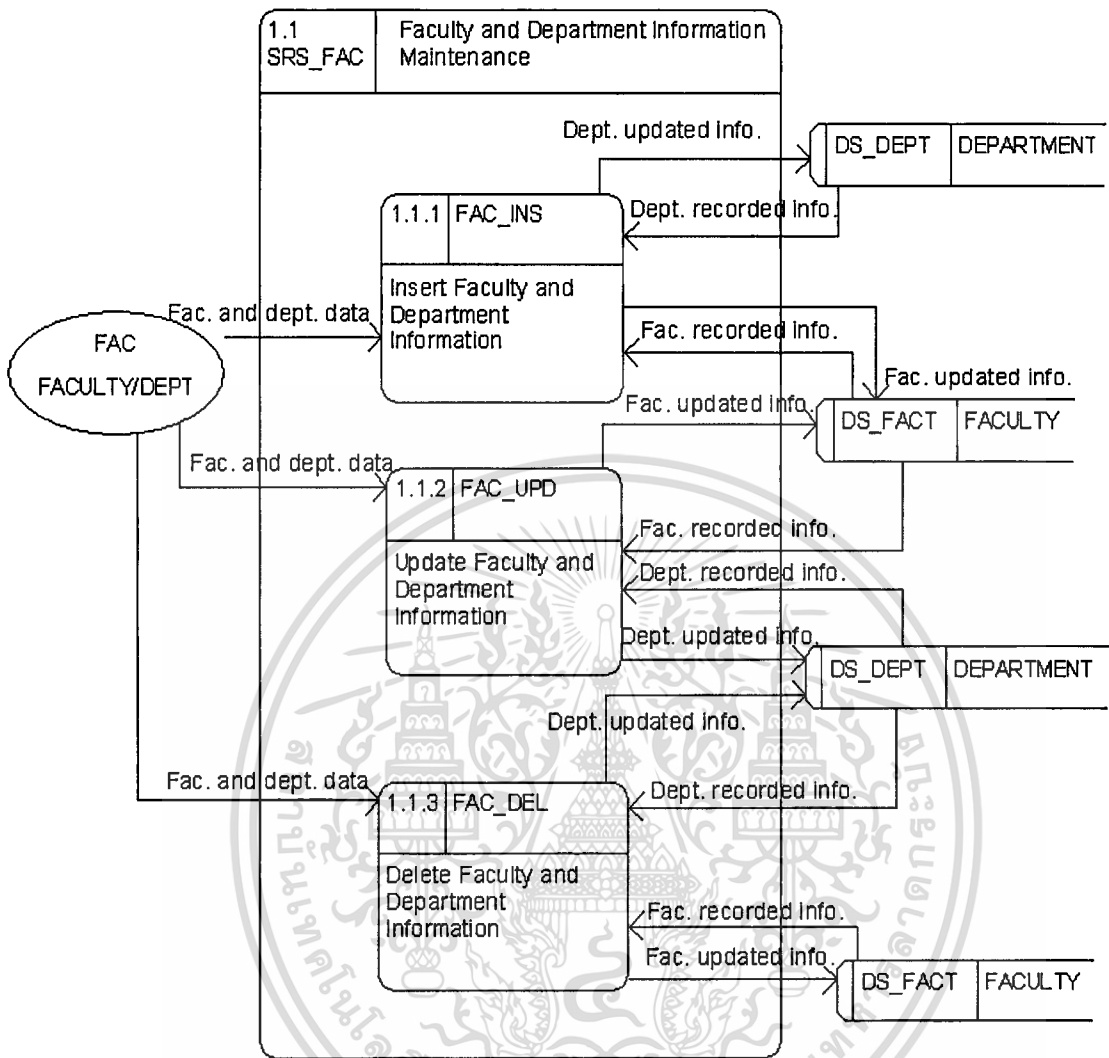
4.2.1 การสร้าง Dataflow Diagram

จาก Dataflow Diagram ของระบบทะเบียนนักศึกษาในส่วนข้อมูลคณะและภาควิชา และ ส่วนการลงทะเบียนที่เราได้ออกแบบไว้ นำมาเขียนโดยใช้เครื่องมือ Dataflow Diagrammer จะได้ Dataflow Diagram Level-1 ของระบบทะเบียนนักศึกษาในส่วนข้อมูลคณะและภาควิชา และ ส่วนการลงทะเบียน ดังรูปที่ 4.6 ในรูปที่ 4.7 จะเป็น Dataflow Diagram Level-2 ของส่วนข้อมูลคณะและภาควิชา และในรูปที่ 4.8 จะเป็น Dataflow Diagram Level-2 ของส่วนการลงทะเบียน

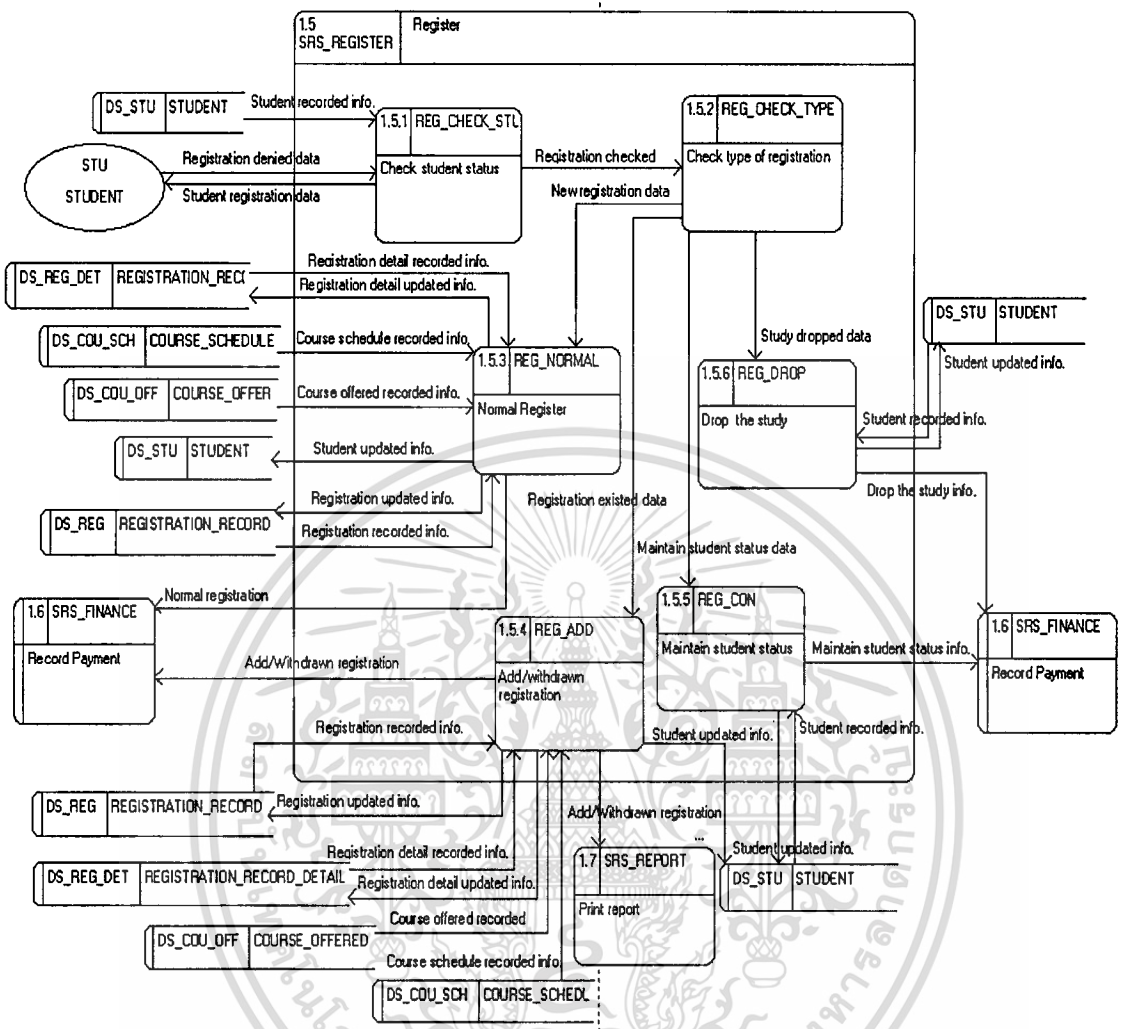


รูปที่ 4.6 Dataflow Diagram Level-1 ของระบบทะเบียนนักศึกษา
ในส่วนข้อมูลคณะและภาควิชา และส่วนการลงทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



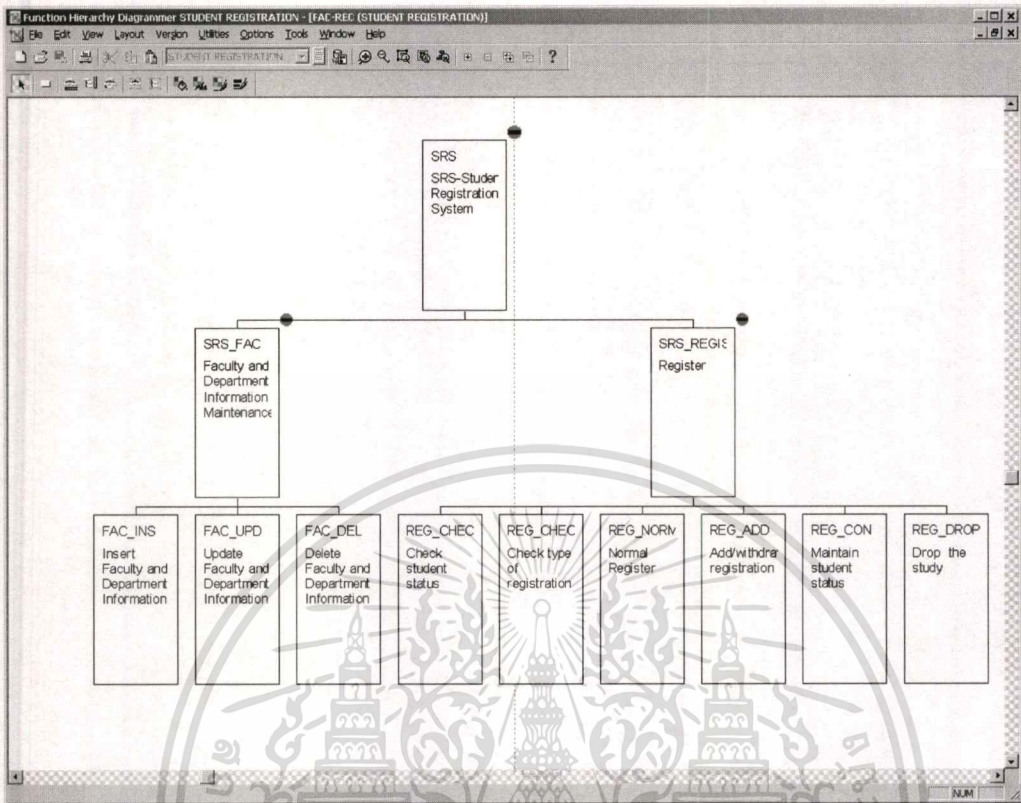
รูปที่ 4.7 Dataflow Diagram Level-2 ของส่วนข้อมูลคณะและภาควิชา



รูปที่ 4.8 Dataflow Diagram Level-2 ของส่วนการลงทะเบียน

4.2.2 การสร้าง Function Hierarchy Diagram

เมื่อสร้าง Dataflow Diagram เรียบร้อยแล้ว เราสามารถใช้เครื่องมือ Function Hierarchy Diagrammer ทำการสร้าง Function Hierarchy Diagram ให้โดยอัตโนมัติ ซึ่งทำให้เรามองเห็นภาพรวมของฟังก์ชันการทำงานทั้งหมดของระบบ สำหรับ Function Hierarchy Diagram ของระบบทะเบียนนักศึกษาในส่วนข้อมูลคณะและภาควิชา และส่วนการลงทะเบียน แสดงดังรูปที่ 4.9



รูปที่ 4.9 Function Hierarchy Diagram ของระบบทะเบียนนักศึกษา
ในส่วนข้อมูลคณะและภาควิชา และส่วนการลงทะเบียน

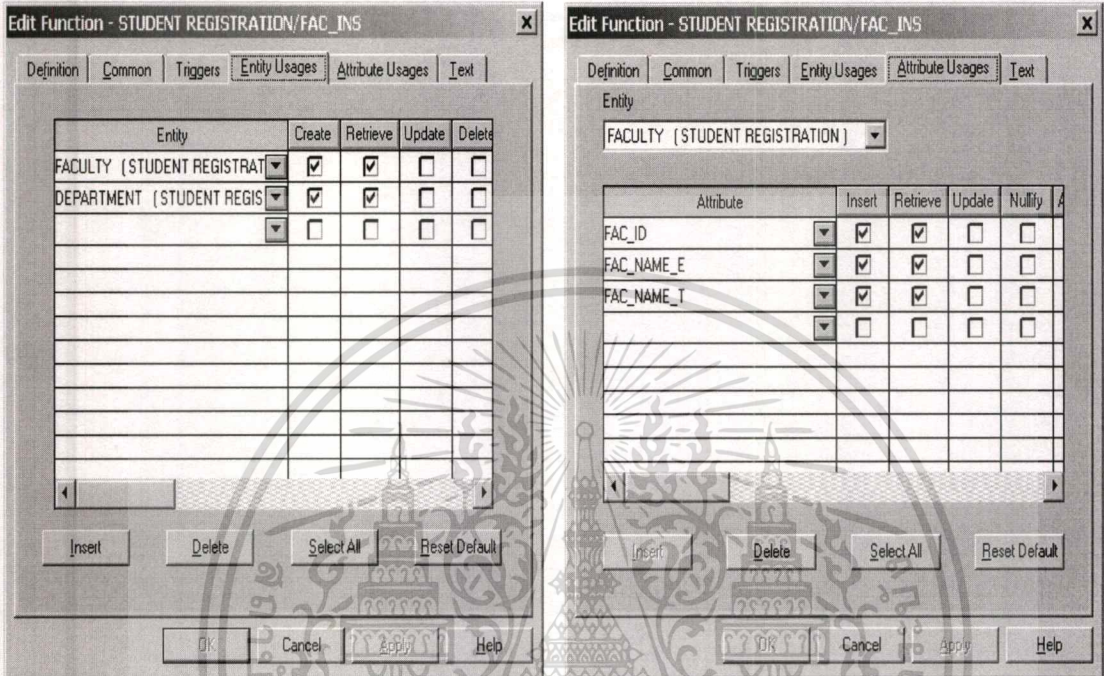
4.2.3 การสร้าง Application Module

เมื่อเราได้ Function Hierarchy Diagram เรียบร้อยแล้ว จากนั้นจะทำการสร้าง Application Module โดยมีขั้นตอนการทำงาน ดังนี้

- พิจารณา Atomic Function ว่าฟังก์ชันใดเป็น Manual และฟังก์ชันใดเป็น Automated เพื่อที่จะนำฟังก์ชันที่เป็น Automated ไปสร้างเป็น Application Module
- ก่อนที่จะกำหนดข้อมูลที่ต้องการใช้ให้กับ Automated Function ข้อมูลที่ต้องการใช้ทั้งหมด จะต้องมีการกำหนดไว้เรียบร้อยแล้วจาก Entity Relationship Diagram ที่สร้างขึ้นโดยใช้ Entity Relationship Diagrammer
- ทำการระบุข้อมูลจากฐานข้อมูลที่สร้างขึ้นที่แต่ละ Application Module ต้องการใช้งาน โดยใช้ฟังก์ชัน Entity Usages และฟังก์ชัน Attribute Usages ดังรูปที่ 4.10 ซึ่งเป็นการ

กำหนดข้อมูลที่จะใช้ให้กับฟังก์ชัน และกำหนดว่าฟังก์ชันนั้นจะจัดการกับข้อมูลอย่างไร เอกสารนี้เป็นเอกสารที่ส่งมอบไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Create, Retrieve, Update, Delete) เพื่อให้ Application Module ที่จะได้มีความถูกต้องสมบูรณ์มากที่สุด



รูปที่ 4.10 ฟังก์ชัน Entity Usages และฟังก์ชัน Attribute Usages

- กำหนดคุณสมบัติของฟังก์ชันในส่วนของ Response Type ว่าต้องการให้ผลลัพธ์แสดงออกมาในรูปแบบใด มีอยู่ 2 รูปแบบ คือ Immediate และ Overnight ถ้าเรากำหนด Response Type เป็น Immediate จะให้ผลลัพธ์เป็น Screen แต่ถ้าเรากำหนดเป็น Overnight จะให้ผลลัพธ์เป็น Utility หรือ Report แสดงดังรูปที่ 4.11

รูปที่ 4.11 การกำหนด Response Type ของฟังก์ชัน

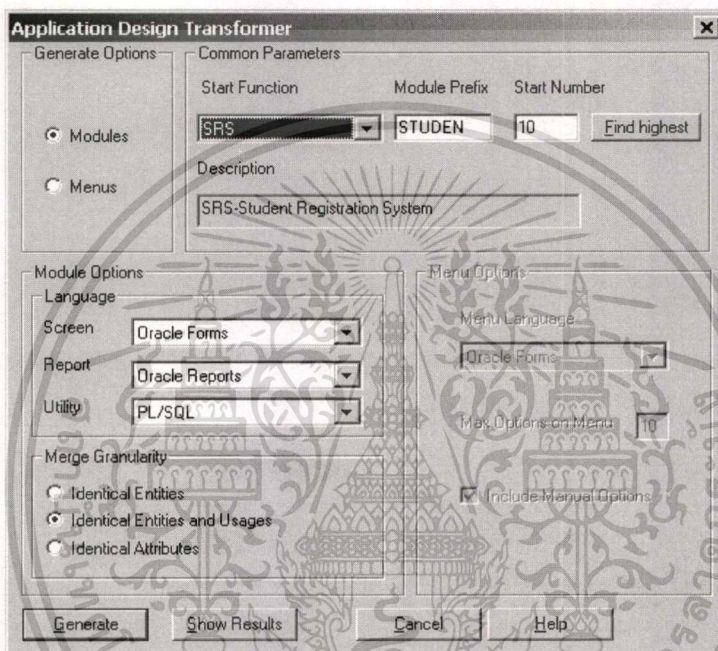
- กำหนดค่าของ Display Type ของแต่ละ Column ในแต่ละ Table ว่าต้องการให้ Column นั้นๆแสดงผลลัพธ์ออกทางหน้าจอในรูปแบบใด เช่น Text, Radio Group, Pop List, Checkbox เป็นต้น แสดงดังรูปที่ 4.12

Column	Prompt	Display type	Width	Height
FAC_ID	Faculty Id	Text	6	
FAC_NAME_T	Faculty Name in Thai	Text	60	
FAC_NAME_E	Faculty Name in English	Text	40	

รูปที่ 4.12 การกำหนด Display Type ของแต่ละ Column

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ใช้ Application Design Transformer ในการแปลง Function และข้อมูลที่ใช้ให้กลายเป็น Application Module ในรูปแบบ Application First-cut Data Design โดยจะต้องกำหนดว่าต้องการจะสร้างเป็น Module หรือ Menu และกำหนด Module Option ว่าต้องการให้สร้าง Application Module ในรูปของภาษาใด และที่สำคัญจะต้องกำหนดฟังก์ชันเริ่มต้นที่จะทำการแปลง แสดงดังรูปที่ 4.13



รูปที่ 4.13 Application Design Transformer

- Application Module ที่สร้างขึ้นจะเป็น Candidate Module ซึ่งถ้าเราต้องการนำ Module ไปใช้งาน จะต้องเปลี่ยนคุณสมบัติ Candidate ของ Module จาก Yes เป็น No เพื่อยอมรับ Candidate Module นั้น

4.2.3 การปรับปรุงแก้ไข Application Module

หลังจากเราใช้ Application Design Transformer สร้าง Application Module เรียบร้อยแล้ว เราจำเป็นต้องปรับปรุงแก้ไข Application Module ก่อนจะทำการสร้าง Form ซึ่งมีขั้นตอนดังนี้

- สร้าง Module Diagram

เริ่มแรกจะทำการสร้าง Module Diagram โดยใช้ Design Editor ซึ่งสามารถแสดง

Module Diagram ได้ใน 2 มุมมอง คือ Data View และ Display View

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น ขอสงวนสิทธิ์ในนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

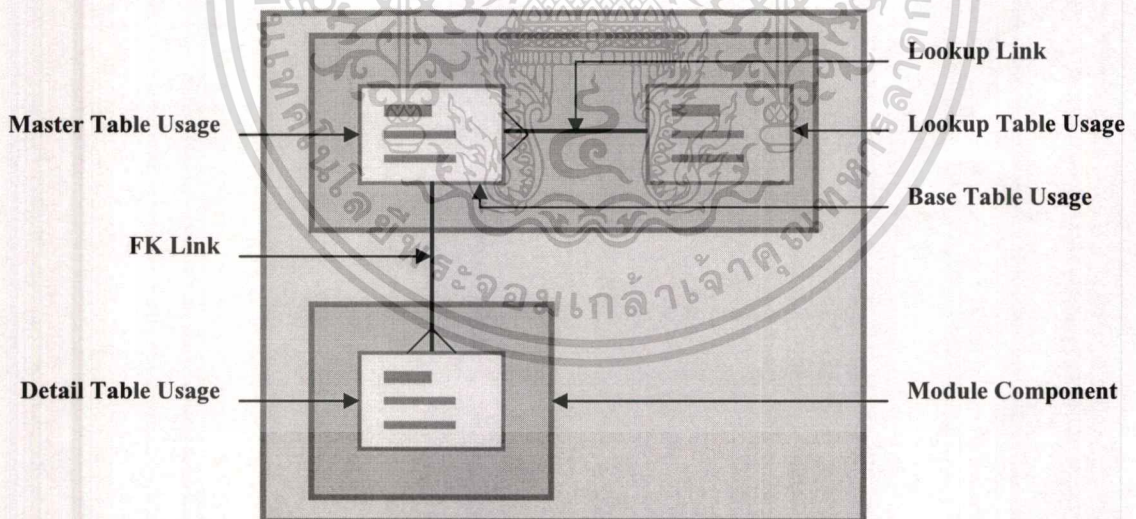
- Data View จะแสดงโครงสร้างของโมดูล ว่าโมดูลประกอบด้วย Table Usage อะไรบ้าง และแสดงความสัมพันธ์ระหว่าง Table Usage ต่างๆ
- Display View จะแสดงหน้าต่างของ Layout View ว่าโมดูลนี้มี Window และ Canvas อะไรบ้าง, มี Item ใดบ้างและเรียงลำดับอย่างไร

- จัดโครงสร้างของโมดูล

เริ่มจากการจัดลำดับตำแหน่งของ Module Component จากนั้นพิจารณาความสัมพันธ์ของ Table Usage ต่างๆ ได้แก่

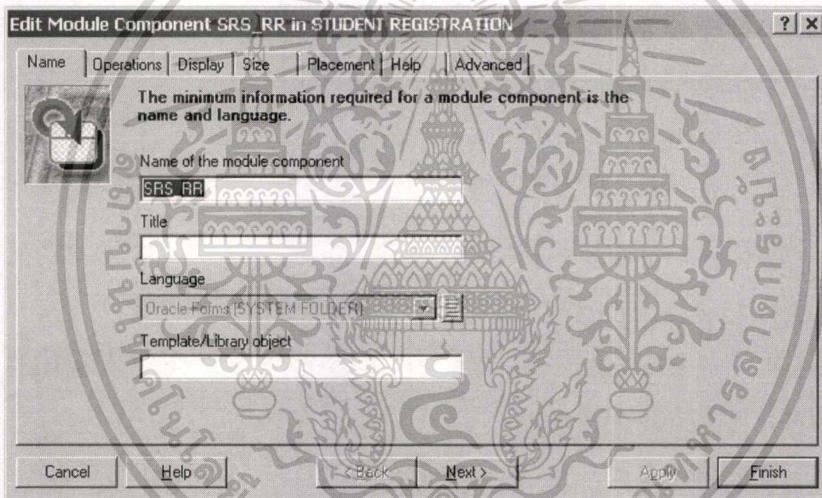
- ความสัมพันธ์แบบ Master-Detail ว่าจะให้อันใดเป็น Master Table และอันใดเป็น Detail Table โดยจะต้องวาง Master Table ไว้ด้านบน Detail Table
- ความสัมพันธ์แบบ Lookup ว่าจะให้อันใดเป็น Base Table และอันใดเป็น Lookup Table โดยจะต้องวาง Lookup Table ไว้ด้านขวาของ Base Table

จากนั้นสร้าง Link เชื่อม Table Usage และที่สำคัญความสัมพันธ์ของ Table Usage ทั้งแบบ Master-Detail Link และแบบ Lookup Link จะต้องมี Foreign Key



รูปที่ 4.14 Module Structure

- สร้าง Bound Item ซึ่งเป็นการกำหนด Module Data Usage โดยกำหนดว่าแต่ละโมดูลใช้คอลัมน์อะไรบ้าง
- กำหนดคุณสมบัติของ Module Component ได้แก่
 - ชื่อของ Module Component
 - กำหนด Allowable Operation ต่างๆ ได้แก่ Insert(Ins), Update(Upd), Delete(Del) และ Allow Query? โดย Application Design Transformer จะกำหนดคุณสมบัติเหล่านี้ตาม Function Entity Usage ซึ่งเราสามารถปรับเปลี่ยนได้
 - จำนวนแถวที่จะแสดงใน 1 บล็อกข้อมูล ซึ่งเราควรที่จะกำหนดจำนวนแถวที่ต้องการให้แสดงออกมาเอง เพราะ โดย Default จะกำหนดเป็น Maximum

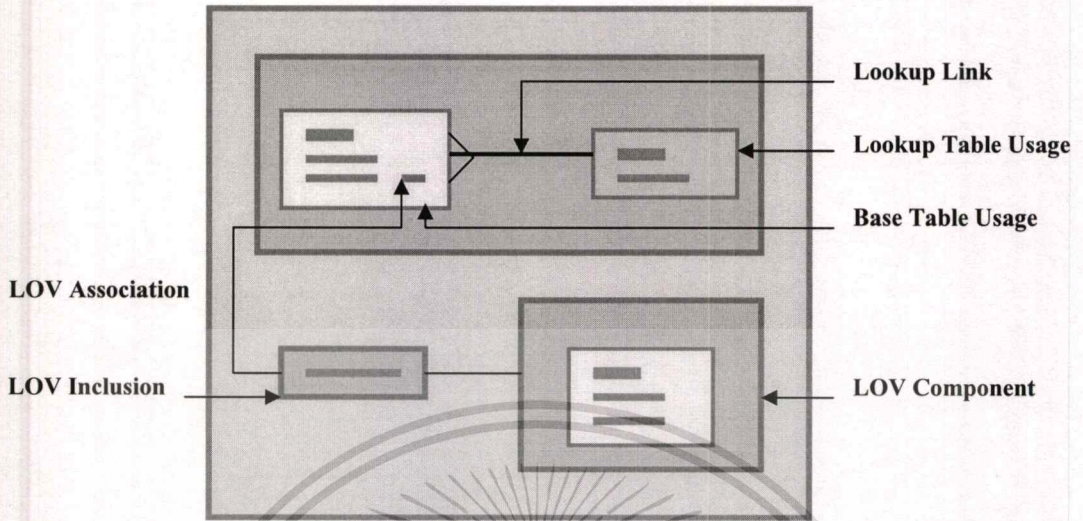


รูปที่ 4.15 Module Component Properties

- สร้าง LOV Component

หลังจากที่เรากำหนด Lookup Table เรียบร้อยแล้ว เราสามารถใช้ Utilities ที่ชื่อว่า Default List of Value เพื่อทำการสร้าง LOV ทั้งหมดที่โมดูลต้องการให้อัตโนมัติ โดย LOV Component ช่วยให้ผู้ใช้สามารถป้อนค่าและคิวรีข้อมูลได้ถูกต้องและรวดเร็ว โดยจะให้ผู้เลือกค่าจาก List of Value ที่แสดง

นอกจากนี้เราจะต้องสร้าง LOV Association เพื่อเรียกใช้ LOV Inclusion ใน Module Component ได้

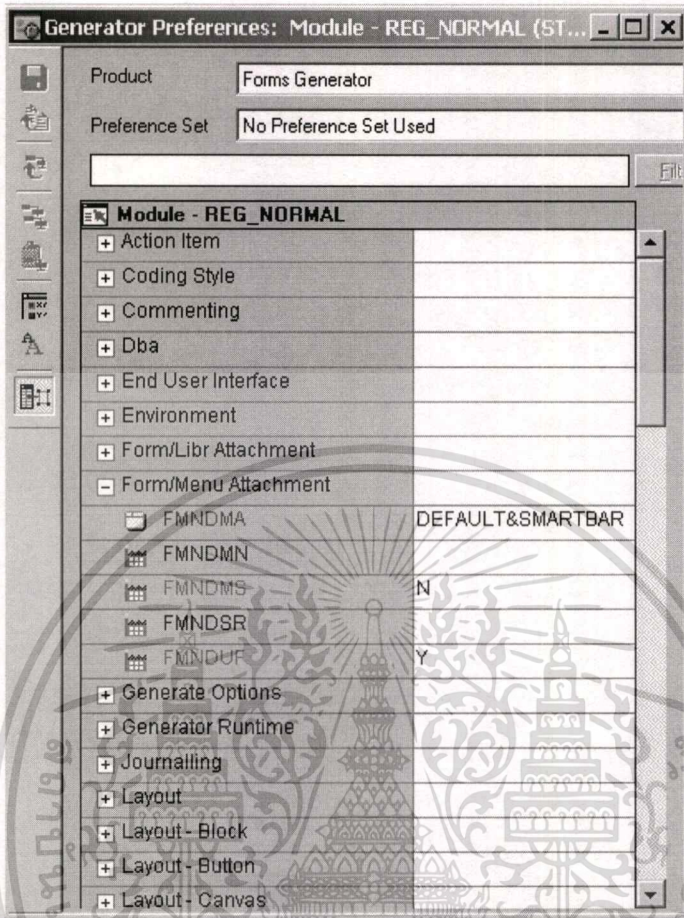


รูปที่ 4.16 LOV Structure

- จัดเรียงลำดับของ Item ต่างๆ โดยเราจะต้องเปลี่ยนมุมมองของ Module Diagram ให้เป็น Display View ก่อน

4.2.4 การสร้าง Form

เพื่อให้ Form ที่เราจะสร้างขึ้นมีรูปร่างหน้าตาตรงตามความต้องการ เราจะต้องทำการกำหนดค่า Preference ต่างๆ ก่อนที่เราจะทำการสร้าง Form เช่น กำหนดให้มี Smartbar, กำหนดให้แสดงปุ่ม LOV เป็นต้น



รูปที่ 4.17 Generator Preference

เมื่อเลือกโมดูลที่ต้องการ Generate แล้ว จากนั้นเรียก Form Generator เพื่อทำการสร้าง Form และจะทำการรัน Form โดยไปเรียก Oracle Form6i ให้โดยอัตโนมัติ ซึ่งในรูปที่ 4.18, รูปที่ 4.19 และรูปที่ 4.20 แสดงตัวอย่าง Form ที่ได้จากการ Generate ของ Oracle Designer6i ซึ่งจะเห็นได้ว่าหน้าตาของ Form ที่ได้นั้นมีการจัดเรียงไม่สวยงามเท่าที่ควร หากต้องการให้ได้ Form ที่มีหน้าตาสวยงาม ควรจะกำหนดค่า Preference เพิ่มเติม หรือปรับแต่งโดยใช้ Layout Editor ของ Oracle Form6i

Oracle Forms Runtime

Action แก้ไข เรียกค้น บล็อก เรคอร์ด ฟิลด์ หน้าต่าง วิธีย่

Faculty

Faculty

Faculty Id 0001 Faculty Name in Thai วิทยาศาสตร์

Faculty Name in English Science

Department

Department Id 0001 Department Name in Thai วิทยาการคอมพิวเตอร์

Department Name in English Computer Science

FRM-40400: Transaction เซลล์สีนแล้ว: 2 เรคคอร์ดที่ถูกนำไม่ใช้และบันทึก
เรคคอร์ด: 1/1

รูปที่ 4.18 Form การเพิ่มข้อมูลคณะและภาควิชา

Oracle Forms Runtime

Action แก้ไข เรียกค้น บล็อก เรคคอร์ด ฟิลด์ หน้าต่าง วิธีย่

Normal Register: Window 0

Student Id 44067000 Student Firstname in Thai นนเรศ

Student Lastname in Thai คุ้มดีแท้

C Id	Course Title in English	Se Year	Se Semester	Co Section	Rr Cr
0001	Introduction to Computer	2003	1	1	3
0002	Data Structure and Algorithm	2003	1	1	3

วิธีย่

เรคคอร์ด: 2/2

รายการแสดงค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 4.19 Form การลงทะเบียนนักศึกษาอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Oracle Forms Runtime

Action แก้ไข เรียกค้น บล็อก เรคอร์ด ฟิลต์ หน้าต่าง วิธืใช้

Add Withdraw Register: Window 0

Student Id Student Firstname in Thai

Student Lastname in Thai

C Id	Course Title in English	Se Year	Se Semester	Co Section	Rr Cr
0001	Introduction to Computer	2003	1	1	3
0002	Data Structure and Algorithm	2003	1	1	3
0003	Database System	2003	1	1	3

เรคคอร์ด: 3/3

รายการแสดงทำ

รูปที่ 4.20 Form การลงทะเบียนเพิ่มถอนรายวิชา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปและข้อเสนอแนะ

จากการศึกษา Oracle Designer6i CASE Tools เพื่อนำมาจัดทำโครงการพัฒนาระบบฉบับนี้ พบว่าเครื่องมือนี้มีประโยชน์อย่างมากต่อการพัฒนาระบบ โดยเฉพาะระบบงานที่มีขนาดใหญ่ อย่างระบบงานทะเบียนนักศึกษา ซึ่งสามารถสรุปผลการทำโครงการพัฒนาระบบฉบับนี้ได้ดังนี้

5.1 ข้อดีของการนำโปรแกรม Oracle Designer6i มาพัฒนาระบบ

- การนำ Oracle Designer6i CASE Tools มาใช้ในการวิเคราะห์และออกแบบระบบนั้น ช่วยให้ขั้นตอนในการพัฒนาระบบมีระเบียบวิธีที่เป็นมาตรฐาน ลดระยะเวลา ค่าใช้จ่าย ตลอดจนการดูแลรักษาระบบที่พัฒนาขึ้น ทำให้สามารถปรับปรุงเปลี่ยนแปลงได้ง่าย
- มี Repository ที่ใช้เก็บข้อมูลทั้งหมดเกี่ยวกับการพัฒนาระบบ ไว้อย่างครบถ้วนในที่เดียวกัน ทำให้ง่ายต่อการสืบค้นและปรับปรุงแก้ไข ช่วยอำนวยความสะดวกในการทำงานร่วมกันระหว่างบุคคลต่างๆที่เกี่ยวข้องกับการพัฒนาระบบ เนื่องจากข้อมูลถูกจัดเก็บ และสามารถเรียกดูได้ที่ศูนย์กลางที่เดียว ทำให้ได้ข้อมูลเดียวกันและเข้าใจตรงกัน เหมาะกับระบบงานใหญ่ๆที่มีการใช้งานแบบ Multi-user นอกจากนี้ยังมีระบบรักษาความปลอดภัยที่ดี ในการควบคุมการเข้าถึงข้อมูลที่อยู่ใน Repository
- สนับสนุนแนวคิดในการพัฒนาระบบงานแบบ Spiral ซึ่งสามารถหมุนวนกลับไปตรวจสอบและทำซ้ำได้ เหมาะกับระบบงานที่ต้องพัฒนาหรือเปลี่ยนแปลงเรื่อยๆ ทำให้ช่วยลดงานของผู้ออกแบบและพัฒนาระบบลงได้ นอกจากนี้ทำให้การออกแบบและการพัฒนามีประสิทธิภาพดียิ่งขึ้น
- เมื่อต้องการเปลี่ยนแปลงแก้ไขในขั้นตอนใดขั้นตอนหนึ่งของการพัฒนาระบบ ก็สามารถไปแก้ไขที่จุดนั้นได้ทันที สะดวกและรวดเร็วกว่าที่จะต้องทำเองทั้งหมด เช่น เมื่อต้องการเปลี่ยนแปลงฐานข้อมูล ก็สามารถกลับมาแก้ไขที่ Entity Relationship Diagram ได้ทันที และทำการ Generate ตารางใหม่
- การศึกษาและพัฒนาระบบต่อในภายหลัง สามารถทำได้สะดวกและรวดเร็วขึ้น ผู้ที่มาพัฒนาระบบต่อสามารถเข้าไปดูข้อมูลต่างๆที่ออกแบบไว้ใน Repository แล้วปรับแต่งให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตรงกับ Requirement ใหม่ตามต้องการ ก็สามารถสร้างโปรแกรมการใช้งานออกมาได้ ไม่ต้องเสียเวลาในการเริ่มต้นออกแบบระบบใหม่ทั้งหมด

- ช่วยให้สามารถสร้างฐานข้อมูลได้ง่ายขึ้น โดยไม่จำเป็นต้องรู้ภาษา Database Definition Language
- ช่วยในการจัดทำเอกสารคู่มือและรายงานตลอดทุกขั้นตอนของการพัฒนาระบบ

5.2 ข้อจำกัดของการนำโปรแกรม Oracle Designer6i มาพัฒนาระบบ

- ในการใช้โปรแกรม Oracle Designer6i ผู้ใช้จะต้องมีความรู้ในการพัฒนาระบบให้ดีพอสมควร รวมถึงต้องใช้เวลาในการศึกษาการใช้งาน CASE Tool นี้ เนื่องจากมีการใช้งานที่ค่อนข้างซับซ้อน เพื่อให้สามารถนำมาใช้พัฒนาระบบได้อย่างมีประสิทธิภาพ
- ในการพัฒนาร่วมกันเป็นทีม จะต้องมีการตกลงกันถึงมาตรฐานต่างๆที่จะใช้ในการพัฒนา เช่น มาตรฐานในการตั้งชื่อ Entity เป็นต้น เนื่องจากโปรแกรมไม่มีกฎเกณฑ์บังคับ
- การสร้างโปรแกรมประยุกต์ใช้งานโดยใช้ฟังก์ชันใน Oracle Designer6i ยังมีจุดอ่อนในการจัด Layout ของโปรแกรมประยุกต์ใช้งานที่จะสร้างขึ้น หากต้องการให้โปรแกรมมีหน้าตาสวยงามตามที่ต้องการ อาจจะต้องนำมาปรับแต่งด้วยโปรแกรมอื่นๆ เช่น Oracle Form6i, Visual Basic เป็นต้น

บรรณานุกรม

พัชรี ประกอบกิจ. 2542. “การพัฒนาระบบซ่อมโทรศัพท์เคลื่อนที่โดยใช้โปรแกรม CASE”.

กรุงเทพฯ : ปริญญาานิพนธ์ หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

สุกิจ กุชชัยสิทธิ์. 2542. “การพัฒนาระบบสารสนเทศโดยใช้ CASE TOOL ORACLE

DESIGNER/2000 : กรณีศึกษา แผนกพัสดุและจัดซื้อ”. กรุงเทพฯ : ปริญญาานิพนธ์ หลักสูตรวิทยาศาสตรมหาบัณฑิต คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

Oracle Corporation. 1999. **Oracle Designer Tutorial**. [Online].

Available: http://www.uncg.edu/irc/oracle/DesignerDoc_Tut/cCTU71/toc.htm.

Oracle Corporation. 2000. **Oracle Designer 6i Product Overview**. [Online].

Available: http://technet.oracle.com/products/designer/pdf/otn_des6i_pover_wp.pdf

Oracle Corporation. 2001. **Oracle iDS Designer: First Class**.



ประวัติผู้เขียน

ชื่อ นางสาวราพร ถิมวัฒนาเลิศ
วันเดือนปีเกิด 4 กุมภาพันธ์ พ.ศ. 2522
สัญชาติ ไทย
สถานที่เกิด อ.หาดใหญ่ จ.สงขลา
การศึกษา จบปริญญาตรี คณะวิทยาศาสตร์ สาขาวิทยาการคอมพิวเตอร์
มหาวิทยาลัยสงขลานครินทร์ วิทยาเขตหาดใหญ่



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้