

ระบบฐานข้อมูลการจัดการสารสนเทศอาคารสถานที่
BUILDING INFORMATION SYSTEM

นางสาว นัฐกร เอมะพลานนท์ รหัส 41360322
 นางสาว ปารีชาติ กุลดู รหัส 41360363

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
 สาขาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2544

ห้องสมุดคณะวิศวกรรมศาสตร์	509794
วันที่รับ 30 มิ.ย. 2544	นค.
เลข : 000 000 4400585	นศร.ต.
เลขเรียกหนังสือ 00	2544
มหาวิทยาลัยนเรศวร 76.9.03	
นศร.ต. 263926	

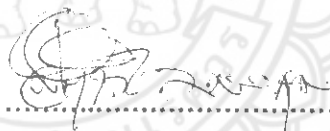
2544 C. 2

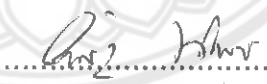


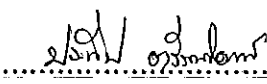
ใบรับรองโครงการวิจัย

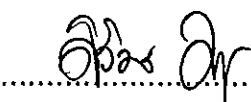
หัวข้อโครงการ	ระบบฐานข้อมูลการจัดการสารสนเทศอาคารสถานที่		
ผู้ดำเนินโครงการ	นางสาวนัฐกร	เอมะพลานนท์ รหัส 41360322	
	นางสาวปาริชาติ	กุดจุก รหัส 41360363	
อาจารย์ที่ปรึกษา	อาจารย์สิทธิโชค	เชาวกุล	
สาขา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2544		

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนครสวรรค์ อนุมัติให้ โครงการฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ คณะกรรมการสอบ โครงการวิจัย


..... ประธานกรรมการ
(อาจารย์ สิทธิโชค เชาวกุล)


..... กรรมการ
(อาจารย์ Wachirawit Pichpanit)


..... กรรมการ
(อาจารย์ ประทีป ศรีรัตนโอกาส)


..... กรรมการ
(อาจารย์ ศิริพร เคษะศิลาภักย์)

หัวข้อโครงการ	ระบบฐานข้อมูลการจัดการสารสนเทศอาคารสถานที่		
ผู้ดำเนินโครงการ	นางสาวนัฐกร	เอมะพลานนท์ รหัส 41360322	
	นางสาวปาริชาติ	กุลฐ	รหัส 41360363
อาจารย์ที่ปรึกษา	อาจารย์สิทธิโชค	เชาวกุล	
สาขา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2544		

บทคัดย่อ

โครงการนี้เป็นการศึกษาและพัฒนาโปรแกรมเกี่ยวกับการจัดการระบบฐานข้อมูลการใช้อาคาร โดยศึกษาจากความต้องการจริงที่เกิดขึ้นเกี่ยวกับการจัดเก็บข้อมูลแบบเดิมที่ยังไม่สามารถอำนวยความสะดวกให้กับผู้ใช้งานได้อย่างครบถ้วน ซึ่งโปรแกรมนี้อาจรองรับการจัดการข้อมูลอาคารในมหาวิทยาลัยได้ เช่น กลุ่มอาคารวิศวกรรมศาสตร์ และยังสามารถนำไปพัฒนาและประยุกต์ใช้กับอาคารแบบอื่นๆ ได้ โดยโปรแกรมนี้อาจแสดงข้อมูลต่างๆของแต่ละห้องภายในอาคาร เช่น อุปกรณ์เครื่องมือเครื่องใช้ต่างๆที่มีภายในห้อง ความจุของห้อง นอกจากนี้โปรแกรมยังสามารถกำหนดการจองห้องได้หากมีผู้ต้องการใช้ห้องในวันและเวลาที่ห้องนั้นว่างจากการใช้งาน ทั้งนี้การแสดงผลจะสามารถเรียกดูข้อมูลได้ผ่านระบบเครือข่าย และสามารถจัดการ แก้ไข เพิ่มเติม และเปลี่ยนแปลงข้อมูลได้ที่เครื่อง Server

Project Title	Building Information System	
Name	Miss Nattaporn Amapalanone	ID. 41360322
	Miss Parichat Kunju	ID. 41360363
Project Advisor	Mr. Sithichoke Chaowagul	
Major	Computer Engineering	
Department	Electrical and Computer Engineering	
Academic Year	2001	

.....

ABSTRACT

This program 's project is studied and developed for information system about building usage. Program is created by real requirement from old system ,that is not comfortable and completes for use. This Information Program can be use for University Building such as Engineering Building group and then it can be develop to another Organization , and program can show any data in many room or building ,and then can be set reservation for who what to reserve room on request time if it free for job. All main program working on network system , than can search , add , edit or change data from Server.

กิตติกรรมประกาศ

ผู้จัดทำโครงการ ขอขอบพระคุณอาจารย์ สิทธิโชค เชาวกุล เป็นอย่างสูงที่กรุณาสละเวลา ให้คำ
ปรึกษา และแนะนำทฤษฎี วิธีการที่เป็นประโยชน์ในการทำโครงการ

ขอขอบคุณอาจารย์ทุกท่านที่ได้ให้ความรู้ ความเข้าใจในวิชาต่างๆ จนสามารถนำมาประยุกต์ใช้
และให้คำแนะนำในการทำโครงการครั้งนี้ได้

และขอขอบคุณ นายเกรียงศักดิ์ เพิ่มพรม นิสิตวิศวกรรมศาสตร์ สาขาวิศวกรรมคอมพิวเตอร์ ที่
ได้ให้คำแนะนำและคำปรึกษาในการทำ โครงการครั้งนี้เป็นอย่างดี

นางสาว นัฐกร เหมะพลานนท์

นางสาว ปารีชาต กุลญ



สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	ก
บทคัดย่อภาษาอังกฤษ.....	ข
กิตติกรรมประกาศ.....	ค
สารบัญ.....	ง
สารบัญตาราง.....	ฉ
สารบัญรูป.....	ช
บทที่ 1 บทนำ	
1.1 ที่มาและความสำคัญของโครงการ.....	1
1.2 วัตถุประสงค์.....	1
1.3 ขอบข่ายการทำงาน.....	1
1.4 ขั้นตอนการดำเนินงาน.....	2
1.5 ผลที่คาดว่าจะได้รับ.....	2
1.6 งบประมาณที่ต้องใช้.....	3
บทที่ 2 ระบบฐานข้อมูลและการออกแบบ โปรแกรมด้วยภาษา JAVA	
2.1 ระบบสารสนเทศ.....	4
2.2 ระบบฐานข้อมูล.....	4
2.3 การออกแบบฐานข้อมูล.....	5
2.4 แผนภาพกระแสข้อมูล.....	6
2.5 Entity Relationship Model.....	7
2.6 ภาษาทางด้านฐานข้อมูล(SQL).....	12
2.7 ภาษา JAVA.....	14
2.7.1 JAVAกับการเขียนโปรแกรมเชิงวัตถุ.....	15
2.7.2 การเขียนโปรแกรมด้วย JAVA ...	16
2.7.3 การเขียนโปรแกรมแบบ GUI โดยใช้โปรแกรม JAVA.....	20
2.8 ภาษา JAVA กับ ฐานข้อมูล.....	31

สารบัญ(ต่อ)

หน้า

บทที่ 3 การออกแบบระบบฐานข้อมูลอาคารสถานที่	
3.1 ศึกษาและรวบรวมข้อมูลของระบบการเก็บข้อมูล เกี่ยวกับอาคารและอุปกรณ์ในอาคาร.....	34
3.2 รวบรวมความต้องการของผู้ใช้.....	38
3.3 วิเคราะห์ข้อมูลที่รวบรวมมา.....	39
3.4 แผนภาพ ER.....	39
3.5 ทำการออกแบบฐานข้อมูล.....	39
3.6 ออกแบบ ในส่วนของโปรแกรม(Application Design).....	39
บทที่ 4 การพัฒนาโปรแกรมระบบฐานข้อมูลสารสนเทศอาคารสถานที่	
4.1 แผนภาพการทำงานของโปรแกรม.....	43
4.2 รูปแบบโปรแกรมฐานข้อมูลสารสนเทศอาคารสถานที่.....	45
บทที่ 5 บทสรุป	
5.1 สรุปผล.....	55
5.2 ปัญหา ข้อเสนอแนะ และแนวทางแก้ไข.....	55
เอกสารอ้างอิง.....	56
ภาคผนวก	
ความรู้เบื้องต้นเกี่ยวกับ JDK (Java Development Kit).....	57
ประวัติผู้ทำโครงการ.....	60

สารบัญตาราง

ตารางที่	หน้า
2.1 Package บางส่วนของ Java	18
2.2 Component ของ Swing เมื่อเทียบกับ AWT.....	21
2.3 AWT Event Listener และ event types.....	28
2.4 Swing Event Listener และ event types	28



สารบัญรูป

รูปที่	หน้า
2.1 แผนภาพกระแสดำเนินข้อมูล.....	6
2.2 สัญลักษณ์แทนสิ่งที่อยู่ภายนอกระบบ.....	6
2.3 สัญลักษณ์แทนการประมวลผล.....	7
2.4 สัญลักษณ์แทนการไหลของข้อมูล.....	7
2.5 สัญลักษณ์แทนแหล่งเก็บข้อมูล.....	7
2.6 RegularEntity.....	8
2.7 WeakEntity.....	8
2.8 Simple Property.....	8
2.9 Composite Property.....	9
2.9 Key.....	9
2.11 Multilevel Property.....	10
2.12 Relationship.....	10
2.13 One-to-One Relationship.....	11
2.14 One-to-Many Relationship.....	11
2.15 Many-to-Many Relationship.....	11
2.16 ความสัมพันธ์ของตัวอย่าง Class.....	15
2.17 การทำงาน ของ Event Handling.....	25
2.18 ลำดับชั้นการถ่ายทอด ของ subclass มาจาก Superclass.....	26

สารบัญรูป(ต่อ)

รูปที่	หน้า
3.1 แผนภาพ DFD ลำดับที่ 0.....	40
3.2 แผนภาพ DFD ลำดับที่ 1.....	40
3.3 แผนภาพ DFD ลำดับที่ 2.....	41
3.4 ER Diagram ของระบบอาคารสถานที่.....	42
4.1 Flow Chart โปรแกรมระบบค้นหาข้อมูลอุปกรณ์.....	43
4.2 Flow Chart โปรแกรมระบบการจอง.....	44
4.3 ลักษณะของ โปรแกรมหน้าหลักของระบบการค้นหาข้อมูลอุปกรณ์ภายในอาคาร.....	45
4.4 ลักษณะของ โปรแกรมระบบการค้นหาข้อมูลอุปกรณ์ภายในอาคาร.....	46
4.5 ลักษณะของ โปรแกรมระบบการบันทึกข้อมูลลงฐานข้อมูลอาคาร.....	47
4.6 ลักษณะของ โปรแกรมระบบการแก้ไขข้อมูลในฐานข้อมูลอาคาร.....	48
4.7 ลักษณะของ โปรแกรมจัดการฐานข้อมูลการจองห้อง.....	49
4.8 ลักษณะของ โปรแกรมการค้นหาห้องว่าง.....	50
4.9 ลักษณะของฟอร์มการกรอกรายละเอียดการจองห้อง.....	51
4.10 ลักษณะของฟอร์มการแก้ไขข้อมูลการจอง.....	52
4.11 ฟอร์มแสดงข้อมูลที่หมดอายุการจอง.....	53
4.12 คั่นหารายละเอียดการจองห้อง.....	54

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของโครงการ

ปัจจุบันนี้มีอาคารสถานที่ขนาดใหญ่ เช่น อาคารเรียนในมหาวิทยาลัย อาคารหอประชุม เกิดขึ้นมากมาย ซึ่งในอาคารเหล่านั้นก็จะประกอบไปด้วยห้องแบบต่างๆ ที่มีขนาดใหญ่เล็กแตกต่างกันเป็นจำนวนมาก อีกทั้งมีการเปลี่ยนแปลงกิจกรรมในห้องหรือมีการเคลื่อนย้ายสิ่งของในห้องเหล่านั้นอยู่เสมอ ข้อมูลต่างๆ เหล่านี้ได้จัดเก็บไว้ในรูปแบบเอกสาร จึงส่งผลให้เกิดความยุ่งยากในการแก้ไขข้อมูล และเกิดการเสียเวลาในการค้นหาข้อมูลที่เป็นข้อมูลในปัจจุบัน

เนื่องจากวิทยาการด้านคอมพิวเตอร์ได้เจริญรุดหน้าเป็นอย่างมาก เราจึงนำคอมพิวเตอร์มาช่วยในการจัดเก็บข้อมูลที่เกี่ยวข้องกับห้องในอาคาร เช่น ลักษณะของห้อง ขนาดความจุ อุปกรณ์ในห้อง และกิจกรรมที่เกิดขึ้นภายในห้องนั้นๆ ซึ่งจะช่วยให้การค้นหาข้อมูลมีความสะดวกและรวดเร็วขึ้น และง่ายต่อการแก้ไขเปลี่ยนแปลงข้อมูล อีกทั้งยังช่วยจัดเก็บข้อมูลจากเอกสารปริมาณมากให้มีปริมาณน้อยลงด้วย

1.2 วัตถุประสงค์

1. เก็บและรวบรวมข้อมูลของอาคาร เช่น ข้อมูลของกลุ่มอาคารเรียนคณะวิศวกรรมศาสตร์ เพื่อนำไปออกแบบระบบฐานข้อมูลได้
2. ออกแบบฐานข้อมูลของห้องภายในอาคารได้
3. ศึกษาและเลือกเครื่องมือที่เหมาะสมเพื่อใช้ในการออกแบบ โปรแกรมจัดการระบบฐานข้อมูล และการออกแบบฐานข้อมูลได้
4. สามารถออกแบบ โปรแกรมจัดการระบบฐานข้อมูล และฐานข้อมูลของอาคารสถานที่ได้

1.3 ขอบข่ายการทำงาน

1. ศึกษาวิธีการออกแบบฐานข้อมูล
 - การเก็บรวบรวมข้อมูล
 - การเขียน Data Flow Diagram
 - การเขียนแผนภาพ ER และการทำ Normalization
2. รวบรวมข้อมูล ของอาคารเรียนคณะวิศวกรรมศาสตร์ จากเจ้าหน้าที่ ที่รับผิดชอบฝ่ายจัดการอาคารสถานที่

1.6 งบประมาณที่ต้องใช้ในการดำเนินงาน

นิสิต : คน : 1,000 บาท

หมึกดำ Epson Stylus 440 1 กล่อง 850 บาท

หมึกสี Epson Stylus 440 1 กล่อง 1,050 บาท

รวม 2,000 บาท



บทที่ 2

ระบบฐานข้อมูลและการออกแบบโปรแกรมด้วยภาษา JAVA

2.1 ระบบสารสนเทศ(Management Information System)

ระบบสารสนเทศ [1] คือการประมวลผลข้อมูล(Data)จำนวนมากให้เหลือสารสนเทศ (Information)จำนวนน้อยเพื่อนำมาใช้ในการตัดสินใจ ระบบนี้ช่วยจัดการข้อมูลที่ต้องการใช้ในระบบ ซึ่งอาจใช้หรือไม่ใช้คอมพิวเตอร์ก็ได้ กรณีที่นำคอมพิวเตอร์เข้ามาใช้ในระบบสารสนเทศก็เพื่อที่จะช่วยให้ เกิดความสะดวกในการทำงานมากขึ้นเช่น ทำให้ค้นหาข้อมูลได้รวดเร็วขึ้น หรือวางแผนงานต่างๆ ล่วงหน้าได้อย่างถูกต้องและทันทั่วทั้งที่เป็นที่ยอมรับว่าการจัดการข้อมูลอย่างมีประสิทธิภาพจะเป็น กุญแจสำคัญที่นำสู่ความสำเร็จได้

ฐานข้อมูลนับเป็นส่วนสำคัญสำหรับระบบงานสารสนเทศที่ใช้คอมพิวเตอร์ในการประมวลผล (Computer-based Information System) เนื่องจากเป็นส่วนที่ใช้เก็บข้อมูลต่างๆซึ่งใช้เป็น อินพุทของทุก ระบบงานสารสนเทศ ในการออกแบบงานสารสนเทศ จึงต้องให้ความสำคัญกับการออกแบบฐานข้อมูล เช่นเดียวการออกแบบในส่วนประมวลผล

2.2 ระบบฐานข้อมูล (Database Systems)

ฐานข้อมูล(Database) คือการนำเอาข้อมูลต่างๆที่มีความสัมพันธ์กันมาเก็บรวมกันไว้ภายในฐานข้อมูลเดียว และสามารถใช้อ้างอิงข้อมูลร่วมกันได้ ฐานข้อมูลที่จัดทำขึ้นเพื่อสนับสนุนการดำเนินงาน อย่างใดอย่างหนึ่งนั้นเรียกว่า ระบบฐานข้อมูล(Database Systems)

2.1 องค์ประกอบของระบบฐานข้อมูล จะประกอบไปด้วย

- ข้อมูล
- ฮาร์ดแวร์(Hardware)
- ซอฟต์แวร์(Software)
- ผู้ใช้ระบบฐานข้อมูล

2.2 ประโยชน์ของฐานข้อมูล

- สามารถลดความซ้ำซ้อนของการเก็บข้อมูลได้
- สามารถหลีกเลี่ยงความขัดแย้งกันของข้อมูลได้
- แต่ละหน่วยงานขององค์กร สามารถใช้อ้างอิงข้อมูลร่วมกันได้

- สามารถกำหนดระบบความปลอดภัยให้กับข้อมูลได้ โดยกำหนดระดับความสามารถในการเรียกใช้ข้อมูลของผู้ใช้แต่ละคน ให้แตกต่างกันตามความรับผิดชอบได้
- ทำให้ข้อมูลเป็นอิสระจาก โปรแกรมที่ใช้งานข้อมูลนั้น ทำให้ผู้พัฒนาโปรแกรมทำการแก้ไขโครงสร้างของข้อมูลได้โดยไม่มีผลกระทบกับโปรแกรมที่เรียกใช้ข้อมูล
- สามารถรักษาความถูกต้องให้กับข้อมูลได้ โดยการระบุกฎเกณฑ์ในการควบคุมความผิดพลาดในการป้อนข้อมูลผิด
- สามารถตอบสนองต่อความต้องการใช้ข้อมูลในหลายรูปแบบ
- สามารถกำหนดให้ข้อมูลมีรูปแบบที่เป็นมาตรฐานเดียวกันได้ เพื่อให้ผู้ใช้ข้อมูลในข้อมูลชุดเดียวกัน สามารถเข้าใจและสื่อสารถึงความหมายเดียวกัน

2.3 การออกแบบฐานข้อมูล

2.3.1 ขั้นตอนการออกแบบฐานข้อมูล แบ่งได้เป็น 3 ขั้นตอนดังนี้

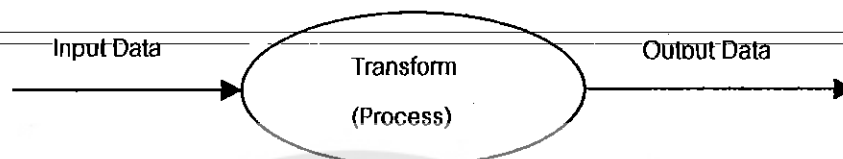
2.3.1.1 การออกแบบฐานข้อมูลในระดับ Conceptual การออกแบบฐานข้อมูลในระดับนี้จะเป็นการกำหนดโครงสร้างเริ่มต้น ที่มีจุดมุ่งหมายเพื่ออธิบายโครงสร้างหลักๆของข้อมูลภายในระบบฐานข้อมูลที่จะนำมาใช้ การออกแบบในระดับนี้มีความสำคัญมากเนื่องจาก โครงสร้างที่ได้จากการออกแบบในขั้นตอนนี้จะถูกนำไปใช้ในขั้นตอนต่อไป โครงร่างหรือที่เรียกว่า Schema ที่ได้จากการออกแบบในขั้นตอนนี้เรียกว่า Conceptual Schema

2.3.1.2 การออกแบบในระดับ Logical การออกแบบในระดับนี้เป็นระดับที่ต่อเนื่องมาจาก ระดับ Conceptual โดยอาศัยโครงร่างที่ได้จากการออกแบบในระดับ Conceptual มาปรับปรุงให้มี โครงสร้างที่เป็นไปตาม โครงสร้างข้อมูลที่จะนำมาใช้งาน โดยยังไม่คำนึงถึงผลิตภัณฑ์ทางด้านฐานข้อมูลที่จะใช้งาน การออกแบบในขั้นตอนนี้ต้องปรับปรุง โครงสร้างบางอย่างในConceptual Schema ให้สอดคล้องกับฐานข้อมูลที่จะนำมาใช้งาน การออกแบบในขั้นตอนนี้จึงต้องมีการตรวจสอบความถูกต้องของ โครงร่างที่ออกแบบขึ้นกับส่วนประมวลผลต่างๆที่ออกแบบไว้รวมทั้งจะต้องแปลงโครงร่างต่างๆให้อยู่รูป Relation

2.3.1.3 การออกแบบฐานข้อมูลในระดับ Physical การออกแบบในระดับนี้จะเป็นขั้นตอนสุดท้ายของการออกแบบฐานข้อมูล ในขั้นตอนนี้จะเป็นการปรับปรุงโครงสร้างโดยการนำเอาโครงร่างที่ได้จากการออกแบบในระดับ Logical มาปรับปรุง โครงสร้างให้เป็นไปตามโครงสร้างของผลิตภัณฑ์ของฐานข้อมูลที่จะนำมาใช้งาน ซึ่งจะกำหนดโครงสร้างในการจัดเก็บ และวิธีในการเข้าถึงข้อมูล ผลที่ได้จากการออกแบบในระดับนี้คือ โครงสร้างของระบบฐานข้อมูล ที่สามารถนำไปใช้ในการสร้างตัวฐานข้อมูลจริง

2.4 แผนภาพกระแสข้อมูล(Data Flow Diagram , DFD)

DFD จะแสดงให้เห็นการไหลของข้อมูล (Information) และช่วยให้นักวิเคราะห์ระบบสามารถสรุปข้อมูลเกี่ยวกับระบบ มีเข้าใจถึงความสัมพันธ์ระหว่างส่วนต่างๆของระบบ และช่วยให้เข้าใจถึงปัญหาสำคัญหรือระบุส่วนของการทำงานที่ซ้ำซ้อนของระบบได้



รูปที่ 2.1 การไหลของข้อมูล

2.4.1 ลำดับขั้นของแผนภาพกระแสข้อมูล

Data Flow Diagram [2] แบ่งออกเป็น หลายลำดับขั้นตามความเหมาะสม โดยแบ่งจนกว่าจะได้รายละเอียดข้างในของระบบย่อยตามที่ต้องการ โดยเริ่มจาก

DFD ระดับที่ 0 : จะแสดงให้เห็นถึงระบบทั้งหมด โดยแทนทั้งระบบด้วยวงกลม 1 วงมีลูกศรเข้าแสดง input จากระบบภายนอกสู่ระบบของเรา และลูกศรชี้ออกที่แสดงถึงผลลัพธ์ออกจากระบบของเราสู่ระบบภายนอก

DFD ระดับที่ 1 : แยกวงกลมที่ได้จาก DFD ระดับที่ 0 ออกเป็นวงกลมย่อยอย่างน้อย 2 วงเชื่อมต่อกันด้วยลูกศร เพื่อแสดงรายละเอียดที่มากขึ้น เนื่องจากรายละเอียดในระดับที่แล้วยังไม่เพียงพอที่จะประมวลผลได้

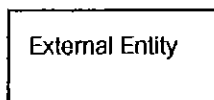
DFD ระดับที่ 2 : แยกวงกลมย่อยที่ได้จาก DFD ระดับที่ 1 ออกเป็นวงกลมย่อยลงไปอีกเพื่อให้ได้รายละเอียดที่มากขึ้น

หากวงกลมใดที่ได้จาก DFD ระดับที่ 2 ยังมีความซับซ้อนอยู่อีกจำเป็นจะต้องมีการแยกย่อยก็ให้แยกย่อยจนกว่าจะได้รายละเอียดตามที่ต้องการ

2.4.2 สัญลักษณ์ที่ใช้ใน แผนภาพกระแสข้อมูล(DFD Notation)

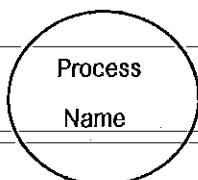
ประกอบด้วยสัญลักษณ์ต่างๆดังต่อไปนี้

2.4.2.1 สัญลักษณ์พื้นผิว : ใช้เป็นสัญลักษณ์แทนสิ่งที่อยู่ภายนอกระบบ



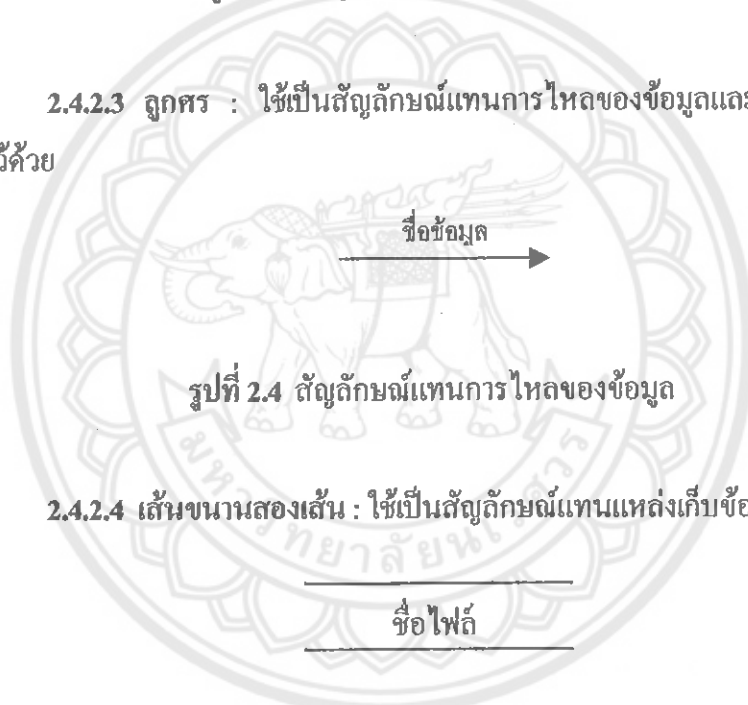
รูปที่ 2.2 สัญลักษณ์แทนสิ่งที่อยู่ภายนอกระบบ

2.4.2.2 วงกลม : ใช้เป็นสัญลักษณ์แทนการประมวลผล แทนกริยาการกระทำต่อ
ข้อมูลไหลเข้ามาโดยไม่คำนึงถึงว่าเป็นการกระทำโดยคอมพิวเตอร์หรือคน



รูปที่ 2.3 สัญลักษณ์แทนการประมวลผล

2.4.2.3 ลูกศร : ใช้เป็นสัญลักษณ์แทนการไหลของข้อมูลและจะต้องมีชื่อของข้อมูล
นั้นๆกำกับไว้ด้วย



รูปที่ 2.4 สัญลักษณ์แทนการไหลของข้อมูล

2.4.2.4 เส้นขนานสองเส้น : ใช้เป็นสัญลักษณ์แทนแหล่งเก็บข้อมูล โดยมีชื่อกำกับ

ชื่อไฟล์

รูปที่ 2.5 สัญลักษณ์แทนแหล่งเก็บข้อมูล

2.5 Entity Relationship Model

Entity Relationship Model หรือที่นิยมเรียกกันสั้นๆว่า E-R Model นี้ เป็น Data Model รูปแบบหนึ่ง ที่นิยมใช้ในการนำเสนอรายละเอียดต่างๆ ที่เกี่ยวข้องกับข้อมูล ในฐานข้อมูล ที่ออกแบบ ในการออกแบบฐานข้อมูลขึ้น ใช้ในงานในระบบสารสนเทศ เนื่องจาก E-R Model เป็นแบบจำลองที่มี รูปภาพที่ใช้แทนโครงสร้างทางด้าน Abstraction ต่างๆ ได้เป็นอย่างดี ซึ่งประกอบด้วยสัญลักษณ์ต่างๆ ต่อไปนี้

2.5.1 Entity เป็นรูปภาพที่ใช้แทน Class ของสิ่งของต่างๆ ซึ่งอาจเป็นทั้ง สิ่งที่ต้องจับต้องได้ เช่น ห้องเรียน หรือไม่สามารถจับต้องได้ เช่น เวลาเรียน ซึ่งใน E-R Model จะแบ่ง Entity ออกเป็น 2 ประเภทดังนี้

2.5.1.1 Regular Entity (Strong Entity) ได้แก่ Entity ที่ประกอบด้วยสมาชิกที่มีคุณสมบัติ ซึ่งบ่งบอกถึงเอกลักษณ์ของแต่ละสมาชิคนั้น รูปภาพที่ใช้แทน Entity ได้แก่ รูปสี่เหลี่ยมผืนผ้า โดยมีชื่อของ Entity นั้นอยู่ภายใน

Classroom

รูปที่ 2.6 Regular Entity

2.5.1.2 Weak Entity สมาชิกของ Entity ชนิดนี้ เป็นสมาชิกที่ไม่สามารถมีคุณสมบัติ ซึ่งบ่งบอกถึงเอกลักษณ์ของแต่ละสมาชิคนั้นได้ จะต้องอาศัยคุณสมบัติใดคุณสมบัติหนึ่งของ Regular Entity มาประกอบกับคุณสมบัติของตัวเอง รูปภาพที่ใช้แทน Entity ได้แก่ รูปสี่เหลี่ยมผืนผ้า 2 รูป ซ้อนกัน โดยมีชื่อของ Entity นั้นอยู่ภายใน

Order - Detail

รูปที่ 2.7 Weak Entity

2.5.1.3 Property หรือ Attribute เป็นข้อมูลที่แสดงลักษณะและคุณสมบัติของ Entity จะสามารถแบ่งย่อยได้ดังนี้

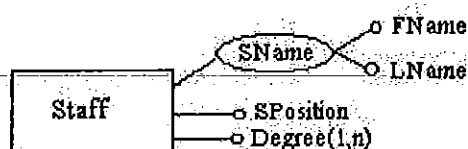
- Simple Property ได้แก่ Property ที่ค่าภายใน Property นั้นไม่สามารถแบ่งย่อยได้อีก เช่น รหัสหมายเลขครุภัณฑ์ ชื่ออุปกรณ์ รูปภาพที่ใช้แทน Property ชนิดนี้ ได้แก่ วงรีที่มีเส้นเชื่อมต่อไปยัง Entity ที่เป็นเจ้าของ Property นั้น โดยมีชื่อของ Property นั้น กำกับอยู่หลังวงกลม

ReserveRoom

o ResRoom Typ
o ResRoom Status
o ResRoom Capa

รูปที่ 2.8 Simple Property

- Composite Property ได้แก่ Property ที่ค่าภายใน Property นั้นสามารถแยกเป็น Property ย่อยได้อีก รูปภาพที่ใช้แทน Property ประเภทนี้ จะใช้วงกลมที่ต่อเชื่อมกับวงรี ของ Simple Property ที่เป็นเจ้าของ Composite Property นั้นแทน



รูปที่ 2.9 Composite Property

- Key เป็น Property หรือกลุ่มของ Property ที่มีค่าในแต่ละสมาชิกไม่ซ้ำกันเลย ซึ่งถูกนำมาใช้กำหนดความเป็นเอกลักษณ์ ให้กับแต่ละสมาชิกใน Entity รูปภาพที่ใช้แทน Key จะใช้วงกลมสี่ค่า ที่มีเส้นเชื่อมต่อไปยัง Entity ที่เป็นเจ้าของ Property นั้น โดยมีชื่อของ Property นั้นกำกับอยู่หลังวงกลม



รูปที่ 2.10 Composite Property

- single Value เป็น Property ที่มีค่าข้อมูลภายใต้ค่าของ Property ใด Property หนึ่ง เพียงค่าเดียว เช่น Property "Equip_ID" ซึ่งอุปกรณ์แต่ละชิ้นจะมีหมายเลขครุภัณฑ์ได้เพียงค่าเดียวจะมีรหัสประจำตัวเพียงค่าเดียวดังตัวอย่างต่อไปนี้

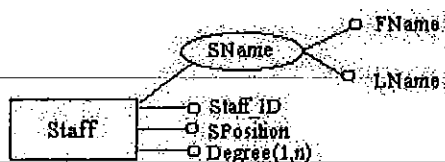
Equip_ID = {(จอมอนิเตอร์เครื่อง5,ww11111),(จอมอนิเตอร์เครื่อง18,ww22222)}

สำหรับรูปภาพที่ใช้แทน Property ประเภทนี้ จะใช้รูปเดียวกับ Simple Property

- Multi-valued Property เป็น Property ที่มีค่าของข้อมูลได้หลายค่าภายใต้ค่าของ Property ใด Property หนึ่ง เช่น Property "Room" ที่ใช้ระบุนิวิชาเรียนในแต่ละห้อง ซึ่งแต่ละห้องจะมีวิชาที่ใช้ห้องเรียนนั้นได้หลายวิชา ดังตัวอย่างข้อมูลต่อไปนี้

Room = {(En509,Database System , Control System),(EN310,Economics,Japanese I)}

รูปภาพที่ใช้แทน Property ประเภทนี้จะใช้วงกลมที่มีเส้นเชื่อมต่อไปยัง Entity ที่เป็นเจ้าของ Property โดยมีชื่อของ Property นั้นพร้อมกับ Mapping Cardinality กำกับอยู่ด้านหลัง



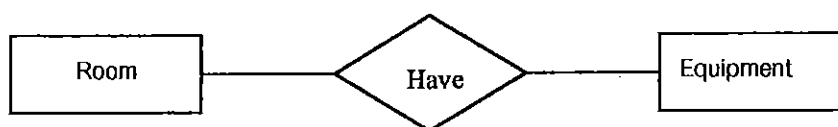
รูปที่ 2.11 Multi-valued Property

- Derived Property เป็น Property ที่ค่าของข้อมูล ได้มาจากการนำค่าของ Property อื่นๆ มาทำการคำนวณ ซึ่งค่าของ Property ประเภทนี้ จะต้องเปลี่ยนแปลงทุกครั้งเมื่อมีการเปลี่ยนแปลงค่าของ Property ที่ถูกนำค่ามาคำนวณเช่น Property Tot_Sal ซึ่งได้มาจากผลรวมของค่าใน Property "Income" ของ Entity "Employee" ซึ่งเป็นเงินเดือนที่พนักงานแต่ละคนได้รับในแต่ละเดือนสำหรับรูปภาพที่ใช้แทน Property ประเภทนี้ จะใช้วงกลมที่มีเส้นเชื่อมต่อไปยัง Entity ที่เป็นเจ้าของ Property โดยมีชื่อของ Property นั้นพร้อมกับคำว่า "Derived Data" กำกับอยู่ด้านหลัง

2.5.3 Relationship โดยทั่วไป จะกำหนดขึ้นจาก Entity ที่มี Property ร่วมกันสมาชิกของ Relationship จึงเกิดจากการจับคู่กันระหว่างสมาชิกของ Entity ที่มารวมกันภายใต้ Relationship นั้น

Relationship ที่สร้างขึ้นนี้ จะใช้แทนความสัมพันธ์ใดความสัมพันธ์หนึ่งระหว่างสมาชิกของ Entity ที่มารวมกันภายใต้ Relationship นั้น ดังนั้นในการตั้งชื่อให้ Relationship จึงควรที่จะตั้งชื่อที่แสดงถึงความสัมพันธ์นั้นๆ

สำหรับรูปภาพที่ใช้แทน Relationship ได้แก่ รูปสี่เหลี่ยมข้าวหลามตัดที่มีชื่อของ Relationship นั้น อยู่ภายในรูปภาพ Relationship จะไม่สามารถปรากฏอยู่เดี่ยวๆ ได้ จะต้องปรากฏคู่กับ Entity เสมอ เช่น



รูปที่ 2.12 Relationship

สำหรับ Relationship ที่ใช้กับ Weak Entity ก็จะใช้รูปภาพเดียวกันกับรูปภาพของ Relationship โดยทั่วไป

Relationship ระหว่าง Entity ใดๆ ไม่จำเป็นต้องมีเพียง Relationship เดียว ถ้าความสัมพันธ์ระหว่างสมาชิกใน Entity เหล่านั้นมีมากกว่า 1 ความสัมพันธ์ สมาชิกใน Entity ที่เกี่ยวข้องกับ Relationship จะถูกเรียกว่า Participant ซึ่งจำนวนของ Participant นี้ จะถูกนำไปใช้ในการกำหนดประเภทของ Relationship ดังนี้

2.5.3.1 One-to-One Relationship เป็น Relationship ที่แต่ละ Participant ของ Entity หนึ่ง จะมีความสัมพันธ์กับอีก Participant ของอีก Entity หนึ่งได้เพียง Participant เดียวเช่นกรณีที่บุคคล 1 คน สามารถจองห้องได้เพียง 1 ห้อง



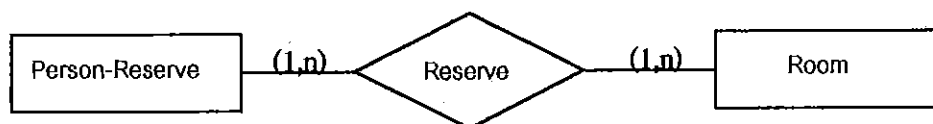
รูปที่ 2.13 One-to-One Relationship

2.5.3.2 One-to-Many Relationship เป็น Relationship ที่แต่ละ Participant ของ Entity หนึ่ง จะมีความสัมพันธ์กับอีก Participant ของอีก Entity หนึ่งมากกว่า 1 Participant เช่นกรณีที่บุคคล 1 คน สามารถจองห้องได้หลายห้อง



รูปที่ 2.14 One-to-Many Relationship

2.5.3.3 Many-to-Many Relationship เป็น Relationship ที่ Participant มากกว่า 1 Participant ของ Entity หนึ่ง จะมีความสัมพันธ์กับอีก Participant ของอีก Entity หนึ่งมากกว่า 1 Participant เช่นกรณีที่บุคคล 1 คน สามารถจองห้องได้มากกว่าห้อง และแต่ละห้องก็สามารถมีบุคคลมาจองได้มากกว่า 1 คน



รูปที่ 2.15 Many-to-Many Relationship

แผนภาพ E-R ที่ดีนั้นจะต้องสามารถอธิบายโครงสร้างของข้อมูลได้อย่างดีและ ครบถ้วนรวมทั้งต้องมีรูปแบบที่ง่ายต่อความเข้าใจ มีความชัดเจน และต้องไม่ซ้ำซ้อนด้วย

2.6 ภาษาทางด้านฐานข้อมูล(Query Language)

ผลิตภัณฑ์ทางด้านฐานข้อมูลที่มีโครงสร้างแบบ Relational จำเป็นจะต้องมีภาษาทางด้านฐานข้อมูล หรือที่เรียกว่า Query Language เช่น ภาษา Structured Query Language (SQL)

ภาษา Query-By-Example(QBE) และภาษา Quel เป็นต้น ซึ่งภาษาทางด้านฐานข้อมูลที่นิยมใช้มากที่สุดภาษาหนึ่งได้แก่ Structured Query Language หรือที่เรียกโดยย่อว่า SQL นั่นเอง

อย่างไรก็ตามภาษา SQL ของแต่ละผลิตภัณฑ์ยังคงมีข้อแตกต่างกันในรายละเอียดทางด้านการใช้งานจึงได้มีการกำหนดมาตรฐานของ SQL ขึ้นโดยมีชื่อว่า Application Architecture Database Interface (SAA-SQL)

2.6.1 คำสั่งต่างๆของภาษา SQL สามารถแบ่งตามลักษณะการใช้งานออกเป็น 3 กลุ่ม ได้ดังนี้

2.6.1.1 กลุ่มคำสั่ง Data Definition Language(DDL) เป็นกลุ่มคำสั่งที่ใช้สำหรับสร้างฐานข้อมูลหรือใช้กำหนดโครงสร้างให้กับ Relation ภายใตฐานข้อมูล เช่น การเพิ่ม เปลี่ยนแปลง ลบ Attribute ของ Relation เป็นต้น

2.6.1.2 กลุ่มคำสั่ง Data Manipulation Language (DML)เป็นกลุ่มคำสั่งที่พัฒนาขึ้นตามแนวคิดของ Relation Algebra และ Record Relational Calculus โดยประกอบด้วยคำสั่งที่ใช้สำหรับเพิ่ม เปลี่ยนแปลง หรือลบข้อมูลในฐานข้อมูล

2.6.1.3 กลุ่มคำสั่ง Data Query Language เป็นกลุ่มคำสั่ง DML ประเภทหนึ่ง ที่ใช้ในการเลือกข้อมูลจาก Relation ขึ้นมาแสดงผลตามรูปแบบที่ต้องการ

2.6.2 รูปแบบของกลุ่มคำสั่ง Data Query Language

รูปแบบของกลุ่มคำสั่ง จะประกอบด้วย 3 ส่วนหลักดังนี้

2.6.2.1 Select เป็นส่วนของคำสั่งที่ใช้กำหนดรายชื่อ Field ของ Table ที่ต้องการให้แสดงผล

2.6.2.2 From เป็นส่วนของคำสั่งที่ใช้ในการกำหนดรายชื่อ Table ที่เป็นแหล่งข้อมูลที่ต้องการเลือกมาแสดงผล

2.6.2.3 Where เป็นส่วนของคำสั่งที่ใช้ในการกำหนดเงื่อนไขในการเลือกข้อมูลจาก Table ที่กำหนดไว้ในส่วนของ From มาแสดงผลสำหรับคำสั่งในส่วนนี้ถ้าไม่กำหนดจะถือว่าต้องการทุก Record ใน Table นั้น

นำทั้ง 3 ส่วนมาประกอบกันเป็นประโยคคำสั่ง จะมีรูปแบบดังนี้

Select.....From.....[Where.....]

ต่อไปนี้เป็นตัวอย่าง ที่แสดงถึงการใช้ SQLในการจัดการกับฐานข้อมูล ที่มักพบเห็นได้โดยทั่วไป

- ค้นหาข้อมูล

```
SELECT * FROM employees
WHERE dept = 1
ORDER BY salary DESC,name
```

- สร้าง Table

```
CREATE TABLE employees
(
  id          int PRIMARY KEY,
  name       char(25) NOT NULL,
  address    char(25) NOT NULL,
  city       char(25) NOT NULL,
  zip        char(25) NOT NULL,
  dept       int,
  phone      char(12),
  salary     int
)
```

- เก็บข้อมูลใหม่

```
INSERT INTO employees VALUES
(
  "John Doe",
  "10725 Java Drive",
  "Mountain View",
  "CA 94040",
  1,
  "415-960-1300",
  60000
)
```

- ลบ Table

```
DROP TABLE employees
```

2.7 ภาษา JAVA

ทางบริษัท Sun Microsystems ซึ่งเป็นผู้พัฒนาภาษา Java ขึ้น ได้แจกเครื่องมือฟรีให้กับผู้พัฒนา
งานด้วย Java เรียกเครื่องมือชุดนี้ว่า “Java 2 SDK Standard Edition” หรือ J2SDK โดยบริษัท Sun ได้
เพิ่ม Feature ต่างๆ ให้มีมากขึ้น และ มีประสิทธิภาพมากขึ้น รวมทั้งได้เพิ่ม “Class Libraries” หรือที่มัก
เรียกว่า “Application Programmer Interface” หรือ APIs โดย APIs นี้จะมีโปรแกรมสำเร็จรูปพร้อมให้
เลือกนำมาใช้งานได้เลย ซึ่งในปัจจุบัน Java ได้รับการพัฒนามาจนถึง J2SDK V1.3 แล้ว

Java นั้นนอกจากจะเป็นภาษาสำหรับเขียนโปรแกรม(application programming) ซึ่งเป็น
ลักษณะของโปรแกรมเชิงวัตถุ(object oriented programming) [3] แล้วยังสามารถนำมาใช้งานบน Web
ได้อีกด้วย จึงทำให้ Java เป็น โปรแกรมที่ได้รับความนิยมและถูกพัฒนาขึ้นอย่างต่อเนื่อง

2.7.1 JAVA กับการเขียนโปรแกรมเชิงวัตถุ (Object Oriented Programming)

การเขียนโปรแกรมเชิงวัตถุ หรือ OOP เป็นวิธีการในการพัฒนาและจัดระเบียบในการเขียน
โปรแกรมรูปแบบหนึ่ง ที่มององค์ประกอบต่างๆให้เป็นวัตถุ (Object) โดยแต่ละวัตถุก็จะมีการจัด
ระเบียบวิธี (Method) ในการปฏิบัติที่แตกต่างกันไป แต่สามารถนำมาใช้รวมกันได้

Java เป็นโปรแกรมภาษาโปรแกรมหนึ่งทำงานในลักษณะของ OOP

2.7.1.1. Object

หัวข้อสำคัญของ OOP คือ Object หรือวัตถุ หมายถึง Entity หรือสิ่งที่มีตัวตน และนำไป
ใช้ในการประมวลผลข้อมูล โดยใน Object จะต้องมีข้อมูล(Data) ที่ใช้ในการอธิบายตนเอง และการ
กระทำต่างๆ ที่ Object สามารถกระทำได้

2.7.1.2 Class

เปรียบเทียบต้นแบบที่ใช้ในการสร้าง Object โดย Class จะประกอบไปด้วยลักษณะเฉพาะ
ของข้อมูลใน Object พร้อมทั้งรายละเอียดของ Action ที่ Object มีลักษณะเฉพาะของข้อมูล คือ
Attributes หรือ Member Function

ประโยชน์ของ Class ของ OOP คือ

1. การแบ่งโปรแกรมออกเป็น Class ทำให้โปรแกรมที่สร้างขึ้นมา มีความง่ายในการแก้ไข เนื่องจาก
แต่ละ Class ไม่เกี่ยวข้องกัน (ถึงแม้จะทำงานร่วมกัน)
2. เมื่อแต่ละ Class ไม่มีผลอย่างใดกับ Class อื่น เมื่อมีการแก้ไข ครั้งนั้น จึงสามารถนำ Class ต่างๆมา
ใช้กับโปรแกรมอื่นๆ ได้

2.7.1.3 Method

Method คือ ระเบียบวิธีในการทำงานของงานใดงานหนึ่ง แสดงเป็นขั้นตอน หรือก็คือการนำคำ
สั่งต่างๆของ Java มาประกอบกันเป็น โปรแกรมนั่นเอง

2.7.1.4 Instantiation

การสร้าง Object ขึ้นมา เพื่อให้สามารถใช้งาน Method ใดๆภายใน Class ได้ นั่นคือ การเรียกใช้ Method ใดๆจะเรียกใช้ด้วยวิธีปกติไม่ได้ ใน Java ต้องเรียกใช้ผ่าน Object ที่สร้างขึ้นมาก่อนภายใต้ Class ที่มี Method นั้นอยู่

การสร้าง Object ใหม่ โดยใช้คำสั่ง

```
ClassName ObjectName = new ClassName();
```

การเรียกใช้ Method จะ ใช้รูปแบบดังนี้

```
ObjectName.MethodName()
```

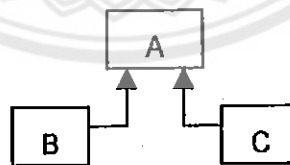
2.7.1.5 Data Encapsulation

สามารถซ่อนข้อมูลภายในของแต่ละ Object เพื่อไม่ให้ Object อื่นมองเห็น ดังนั้นเมื่อมีความต้องการใช้ข้อมูลเกิดขึ้น จะต้องทำการร้องขอไปยัง Object ที่มีข้อมูล ประโยชน์ก็คือเมื่อต้องการแก้ไขข้อมูลหรือ Action ใดๆ ภายใน Object หนึ่งจะไม่ส่งผลถึง Object อื่น

2.7.1.6 Inheritance

OOP นั้นมีความสัมพันธ์กับ โครงสร้างแบบชั้น (Hierarchy) หรือเรียกอีกอย่างหนึ่งว่า “การสืบทอดคุณสมบัติ”

โดยสิ่งที่อยู่ด้านบนถ่ายทอดคุณสมบัติต่างๆ ลงมาให้กับสิ่งที่อยู่ด้านล่าง เช่น



รูปที่ 2.16 ความสัมพันธ์ของตัวอย่าง Class

ถ้า A มีคุณสมบัติอยู่ 3 อย่าง ก็จะถ่ายทอดคุณสมบัติทั้ง 3 อย่าง มาให้ B กับ C แต่ทั้ง B และ C อาจมีคุณสมบัติอื่นเพิ่มเติมขึ้นมาได้อีก

2.7.2 การเขียนโปรแกรมด้วย JAVA

2.7.2.1 ชนิดของโปรแกรม JAVA

- Java Application

เป็นการนำ Java มาเขียนเป็น โปรแกรมที่มาใช้งาน ใ้อิสระ (Stand Alone Program) เหมือนกับการเขียน โปรแกรมภาษาระดับสูงอื่นๆ เช่น C++ Pascal ทั้งนี้สามารถนำ แอปพลิเคชันไปใช้งานที่คอมพิวเตอร์ต่างแพลตฟอร์ม (Platform) ไม่ว่าจะเป็น PC Macintosh หรืออื่นๆ

- Java Applets

เป็นการนำ Java มาเขียนเป็น โปรแกรมเช่นกัน แต่ไม่สามารถเรียกใช้ตามลำพังเหมือน Application แต่จะต้องนำไปใส่ไว้ในเอกสาร HTML แล้วใช้โปรแกรม Web Browser เช่น Netscape, Internet Explorer หรือใช้ โปรแกรมของ Java ชื่อ AppletViewer เพื่อเรียกผลลัพธ์ก็ได้

แต่ทั้ง 2 ชนิด จะมีการแปลงรหัสโปรแกรมต้นฉบับที่เรียกว่า “Source Code” หรือ “Source Program” ไปเป็นรหัสที่คอมพิวเตอร์เข้าใจก่อน เรียกว่า “Machine Code” หรือ “Object Code” ขึ้นตอนในการแปลงเป็นดังนี้

1. สร้าง Source Program ขึ้นมาก่อน โดยกำหนดนามสกุลเป็น *.java
2. นำไฟล์ที่ได้มาแปลงเป็น Bytecode ด้วยวิธีการที่เรียกว่า compile จะได้ไฟล์ใหม่ 1 ไฟล์ โดยมีชื่อเดิม แต่นามสกุลเป็น *.class

ถ้าเป็น Application

3. นำไฟล์ที่ได้จากข้อ 2 มาเรียกใช้ เรียกว่า Interpret โดยจะเรียกใช้ที่คอมพิวเตอร์ต่าง Platform กันได้ (ในการใช้ผลลัพธ์จะใช้คำสั่ง “java” Interpreter ของภาษา Java แล้วตามด้วยชื่อไฟล์ที่ได้จากข้อ 2 แต่ไม่ต้องใส่นามสกุล *.class ลงไปด้วย)

ถ้าเป็น Applets

4. สร้างเอกสาร HTML ขึ้นมา
5. นำชื่อไฟล์ที่ได้จากข้อ 2 ใส่ลงในเอกสาร HTML โดยใช้คำสั่งของ Applet
6. เรียกผลลัพธ์ผ่านทาง Web Browser หรือ AppletViewer

2.7.2.2 โครงสร้างของ Class (Class Structure)

1. ภายใน Class จะมี Method หลักชื่อ “main” (จะเป็น Method ที่โปรแกรมจะเรียกใช้ เป็นอันดับแรกเสมอ) และท้ายชื่อของ Method จะต้องตามด้วยเครื่องหมายวงเล็บ (กรณีที่ไม่มีการส่งค่า Parameter ใดๆ มาด้วย)

2. ชื่อของ Class หรือ Method ใดๆ ที่ตั้งไว้ ต้องระวังเรื่องอักษรพิมพ์ใหญ่และพิมพ์เล็ก เนื่องจาก compiler ของ Java จะแปลงเป็นคนละตัวกัน

3. ภายใน Class และ Method หนึ่งๆ จะต้องมีเครื่องหมายปีกกาเปิด ({) เพื่อแสดงจุดเริ่มต้น และมีเครื่องหมายปีกกาปิด (}) เพื่อแสดงจุดสิ้นสุดของ Class นั้น

4. แต่ละคำสั่งที่ใช้ภายใน Class (ยกเว้นบรรทัด Header ของ Class และ Method) จะต้องปิดท้ายด้วยเครื่องหมาย Semicolon (;) เสมอ เช่น theApp.Run();

5. บรรทัดที่แสดงชื่อ Class จะต้องมี Keyword ว่า “class” เสมอ และไม่ต้องปิดท้ายด้วย Semicolon (;)

6. หน้าชื่อของตัวแปร , class หรือ Method จะมี Keyword เพื่อระบุขอบเขต (Accessibility) ของการเรียกใช้โดยแบ่งเป็น static, public, private และ void

static ใช้นิยามตัวแปร และ Method ที่ต้องการให้ค่าคงที่ สามารถนำไปใช้ได้ทุกๆ ส่วนของ Class โดยค่านั้นจะไม่มีเปลี่ยนแปลง

public ใช้นิยามตัวแปร , Method และ Class ใดๆ เพื่อให้สามารถนำไปใช้ได้กับ Class หรือ โปรแกรมอื่นได้

private ใช้นิยามตัวแปร หรือ Method เพื่อให้สามารถใช้ได้เฉพาะภายใน Class ที่สร้างขึ้นมานั้น

protected ใช้นิยามตัวแปร หรือ Method ที่ใช้เฉพาะ Class ที่สร้างขึ้นมาด้วยวิธีการ Inheritance

void ใช้นิยาม Method โดยกำหนดให้ไม่มีการส่งค่าใดๆกลับมาให้ Method นี้

7. Java ไม่ได้กำหนดรูปแบบการจัดเรียง Method ใน Class ไว้ ดังนั้นจึงสามารถวาง Method ได้ตามอิสระ แต่ทั้งนี้เพื่อความสะดวก นิยมวาง main Method ไว้ด้านบนสุดของโปรแกรม โดยทุกโปรแกรมจะต้องมี Method

```
Public static void main(String args[]) หรือ
```

```
Public static void main(String[] args)
```

8. สามารถมี Class ได้มากกว่า 1 Class ใน 1 โปรแกรม และโปรแกรมใดๆ ก็สามารถเรียกใช้ Class และ Method ในโปรแกรมอื่นๆ ได้ (ถ้าโปรแกรมนั้นกำหนดให้สามารถเรียกใช้ได้) สำหรับ Class ที่ใช้เป็นชื่อโปรแกรมจะต้องมี Keyword ว่า “public” ด้วยเสมอ (ดังนั้นมีเพียง 1 public class ใน 1 โปรแกรมเท่านั้น)

2.7.2.3 Package (JAVA Class Library)

Package เปรียบเสมือนเป็นห้องสมุดที่เก็บรวบรวม Classes ต่างๆเอาไว้มากมาย โดยจะแยกเก็บไว้เป็น Directory และภายในแต่ละ Class จะมี Method มากมายให้เลือกนำมาใช้ให้เหมาะสมกับงานที่เราต้องการ โดยเราไม่ต้องเขียนขึ้นมาเอง

Package ถูกเรียกใช้ด้วยคำสั่ง import ซึ่งจะต้องวางไว้บนสุดของโปรแกรมเสมอ เพื่อให้ compiler นำ Package ที่อ้างถึงมาเชื่อมโยงเข้ากับ โปรแกรม ก่อนที่จะเรียกใช้ Method ต่างๆเหล่านั้นภายในโปรแกรม โดยจะใช้เครื่องหมายจุด (.) คั่นระหว่างชื่อ Package และชื่อ Class เช่น

```
import java.util.*;
```

เป็นการเรียกใช้ Package ชื่อ "java.util" ทุก Class ที่อยู่ใน Package นี้ แต่จะเป็นการเปลี่ยนหน่วยความจำ ดังนั้นจึงนิยมระบุไว้ด้วยว่าเป็น Class ไค เช่น

```
import java.util.Calendar.*;
```

Java มี Package อยู่มากมาย เรียกว่า Standard Packages เพื่อให้เรียกใช้งานได้ต่างๆ กัน เช่น

ตารางที่ 2.1 Package บางส่วนของ Java

Package	คำอธิบาย
Java.applet	ประกอบไปด้วย Class ที่จำเป็นต่อการสร้าง Applet ซึ่ง Class ใน Package ทำงานได้ภายใต้ Netscape เวอร์ชัน 2.0 ขึ้นไป HotJava หรือ Browser ที่เทียบเคียงกันได้
Java.lang	สนับสนุนงานจัดการด้านการคำนวณตัวเลข, ข้อมูลตัวเลข และข้อมูลแบบ Arrays จัดเป็น Class พื้นฐานของ Java ที่ทุกโปรแกรมต้องมี Java จึงจัดเตรียมไว้ให้อัตโนมัติ ดังนั้นจึงไม่จำเป็นต้อง import เข้ามาไว้ในโปรแกรม
Java.awt	AWT = Alternative Window Toolkit ประกอบด้วย Class ที่ใช้ในการสร้าง GUI (Graphic User Interface) โดยอิสระต่อแพลตฟอร์ม ภายในยังแบ่งเป็น Package ย่อยอีก เช่น java.awt.peer, java.awt.image เป็นต้น นอกจากนั้นยังมี Routine สำหรับการวาดรูป (Drawing)

ตารางที่ 2.1 (ต่อ) Package บางส่วนของ Java

Package	คำอธิบาย
Java.io	ประกอบด้วย Class ที่ปฏิบัติงานด้าน I/O
Java.net	ประกอบด้วย Class ที่ใช้สร้างการติดต่อกับเครือข่าย มักใช้ควบคู่กับ java.io สำหรับอ่านและเขียนข้อมูลจากเครือข่าย
Java.util	ประกอบด้วย Class ที่เป็น Utilities เช่น การเข้ารหัส, การถอดรหัส, การเรียกใช้วันที่และเวลา
Java.swing	ประกอบด้วย Class ที่ช่วยในการสร้าง "Swing" GUI

2.7.2.4 การสร้าง Object (Object Instantiation)

เมื่อต้องการเรียกใช้งาน Class ใดก็ตาม จะต้องทำการสร้าง Object ขึ้นมาใหม่ เพื่อเตรียมไว้ให้สามารถใช้งานได้กับ Class นั้นๆ โดยเฉพาะ เช่น Java มี Class ชื่อ "String" ซึ่งเราสามารถนำมาใช้ในการใส่ข้อมูลลงไป แต่เราใส่ข้อมูลลงใน Class ไม่ได้ ต้องใส่ลงใน Object เท่านั้น ดังนั้นจึงต้องมีการสร้าง Object ขึ้นมา โดยมีขั้นตอนดังนี้

1. กำหนดว่า Object อยู่ใน Class ใด เช่น

```
string S1;
```

ให้ Object S1 อยู่ใน class String (สายอักษร)

2. สร้าง Object ขึ้นมาพร้อมใส่ข้อมูลลงไป

```
S1 = new String("Hello World");
```

หรือรวมเป็นคำสั่งเดียว คือ

```
String S1 = new String("Hello World");
```

เรียกการสร้าง Object ใหม่ว่า "Object Instantiation"

เรียกการสร้าง Object พร้อมกับการใส่ข้อมูลนี้เรียกว่า "Object Initialization"

2.7.2.5 การเรียกใช้ Method

แบ่ง Method ออกเป็น 2 ประเภท ได้แก่

1. Method ที่สร้างขึ้นเอง โดยผู้เขียนโปรแกรม จะเขียน Method ต่างๆ ขึ้นใช้เองตามต้องการ Method ที่สร้างขึ้นเองนี้ อาจเป็น Method ที่อยู่ใน Class เดียวกัน หรือ ต่าง Class กันก็ได้
2. Method ที่มีอยู่แล้ว จะได้มาพร้อม Package ของ JDK โดย Method เหล่านี้จะอยู่ใน Class Libraries ต่างๆ ที่เราต้อง Import เข้ามาใช้ในตอนต้นของโปรแกรม จึงไม่จำเป็นต้องสร้าง Object ขึ้นมาใหม่ ก็สามารถเรียกใช้ Method ใน Class ได้เลย เช่น Method “println”

```
System.out.println("Hello World");
```

ซึ่งเป็น Method หนึ่งใน Class ชื่อ “System.out” ของ Package ชื่อ “java.lang” ใช้ส่งข้อความโดยจะส่ง “Hello World” ไปให้กับ Method นี้ด้วย เพื่อใช้ในการแสดงผลทางจอภาพ

2.7.3 การเขียนโปรแกรมแบบ GUI (GUI Programming) โดยใช้โปรแกรม JAVA

การเขียนโปรแกรมแบบ GUI [4] นิยมสร้างให้เป็นจอภาพแบบ Graphic เป็นการเขียนโปรแกรมที่พยายามให้มนุษย์สื่อสารและทำงานกับโปรแกรมได้อย่างมีประสิทธิภาพยิ่งขึ้นด้วยการเข้าใจ interface หน้าที่ ขั้นตอนการทำงานและข้อความต่างๆของ interface ได้ง่ายขึ้น

2.7.3.1 AWT

เครื่องมือพื้นฐานในการเขียนโปรแกรมด้วย GUI ใน JAVA จะใช้ AWT (Abstract Windowing Toolkit – java.awt) โดยในการเขียนโปรแกรมด้วย GUI นั้น จะต้องมีส่วนประกอบ 3 ส่วน ประกอบไปด้วย Components , Containers และ Layout Managers

- AWT Components

AWT Components เป็น object ที่ java ได้สร้างมาให้ใช้งานได้เลย (building block) ประกอบไปด้วย class ต่อไปนี้ Component Container Button Canvas CheckBox Choice Dialog Frame Label List Menu MenuItem Panel Scrollbar TextArea TextComponent และ TextField

- AWT Containers

Containers จัดเป็น Component อย่างหนึ่งที่สามารถนำ Component อื่นๆ มาใส่เอาไว้ได้ ประกอบไปด้วย class ต่อไปนี้ Applet Container Dialog File Dialog Frame Panel ScrollPane และ Window

- Layout Managers

ใช้สำหรับจัดแสดง Container ออกทางจอภาพ ประกอบไปด้วย class ต่อไปนี้ BorderLayout ,

FlowLayout , CardLayout , GridLayout และ GridBagLayout

2.7.3.1.2 Swing

เนื่องจาก AWT เมื่อมาใช้ในการเขียน GUI program นั้นจะมีปัญหาบางอย่างเกิดขึ้น เช่น ลื่น เปลี่ยนหน่วยความจำของคอมพิวเตอร์ และ GUI ที่สร้างจาก AWT จะปรับเปลี่ยนไปตาม platform ที่ใช้ ถ้าใช้ต่างกันก็อาจได้ผลลัพธ์หรือรูปร่างหน้าตาของ GUI ที่แตกต่างไปจากที่ต้องการได้ ทำให้มีการพัฒนา package ชื่อ “ Swing ” ขึ้นมาช่วยสนับสนุนการทำงานให้ดีขึ้น Swing Components จะเรียกว่า เป็น “pure java components ” เนื่องจากถูกสร้าง จัดการ และแสดงผลได้อย่างสมบูรณ์ภายใน JAVA ดังนั้น ไม่ว่าจะนำไปใช้ ใน platform ใด ผลลัพธ์ก็จะไม่เปลี่ยนแปลง แต่ไม่ได้เป็นการนำมาใช้งานแทน AWT เพราะยังคงใช้ AWT อยู่ และ Swing เองก็ยังคงต้อง extends class บาง class จาก AWT มาใช้งานด้วย

Swing เป็น API (Application Programming Interface) ทั่วหนึ่งใน Java Foundation class (JFC) ซึ่งเป็นส่วนหนึ่งของ JDK โดย API ใน JFC ประกอบไปด้วย Accessibility Drag-and-Drop Java2D Swing และ Input Method Framework

ตารางที่ 2.2 Component ของ Swing เมื่อเทียบเท่ากับ AWT

AWT Component	Swing JFC
Applet	JApplet
Component	JComponent
Container	JContainer
Button	JButton
Canvas	JCanvas
CheckBox	JCheckBox
Dialog	JDialog
Frame	JFrame
Label	JLabel
List	JList
Menu	JMenu
MenuBar	JMenuBar
MenuItem	JMenuItem

ตารางที่ 2.2 (ต่อ) Component ของ Swing เมื่อเทียบกับ AWT

AWT Component	Swing JFC
TextComponent	JtextComponent
TextFiield	JtextField
Panel	Jpanel
Scrollbar	Jscrollbar
TextArea	JTextArea

Swing JFC ที่ไม่มีใน AWT ได้แก่ JComboBox JRadioButton JTable Box JColorChooser JComboBox JDeskTopPane JInternalFrame JLayeredPane JSeparator JSlider JSplitPane JTextPane JToggleButton JToolBar JToolTip JTree และ JViewPort

2.7.3.1.3 Swing Components

เครื่องมือต่างๆที่นำมาใช้ในการสร้าง GUI เรียกว่า class โดย class จะอยู่ใน Package ชื่อ javax.swing เช่น class JButton , class JCheckbox , class JLabel ทั้งนี้ในแต่ละ class ก็จะมี Constructor Method ของตนเอง ที่จะถูกเรียกใช้เมื่อมีการสร้าง Object ใหม่ภายใต้ class ต่างๆ

ดังนั้นคำสั่งที่ใช้ในงานต่างๆ ใน GUI จึงเป็นคำสั่งในการสร้าง object ใหม่ เป็นส่วนใหญ เช่น

```
Jbutton myButton = new JButton();
```

หมายความว่า สร้างปุ่มว่างขึ้นมา 1 ปุ่ม (new JButton()) ซึ่งคำสั่งนี้จะเรียกใช้ Constructor ของ Class JButton เพื่อสร้างปุ่มว่างขึ้นมา โดยปุ่มว่างนี้ จะเป็น Object ที่ชื่อ myButton

2.7.3.1.4 Container

เป็นส่วนประกอบสำคัญของ Interface จะใช้สำหรับสร้าง Component ที่มีหลายๆ Components อยู่ภายใน เช่น Button, TextField ,RadioButton ฯลฯ อยู่นั้นคือ Container ก็คือ Component ที่ประกอบขึ้นจาก Components หลายตัว ในทุก Container จะมี Layout Manager ที่ทำหน้าที่ในการจัดรูปแบบของแต่ละ Component ใน Container

Method ที่ใช้กับ Container มี 2 methods ที่สำคัญคือ

add ใช้สำหรับใส่ Component ลงใน content pane(พื้นที่ของ Container)

setLayout ใช้สำหรับกำหนดชนิดของ Layout Manager ที่ต้องการใช้งาน ใน 1 Container จะมี Layout Manager ได้เพียงชนิดเดียวเท่านั้น

2.7.3.1.5 Layout Manager

ประเภทที่มักนำมาใช้งานได้แก่

- FlowLayout เป็น Layout ง่ายที่สุดคือ จะวาง Component ในแนวนอนจากซ้ายสุดไปขวาสุด เมื่อหมดบรรทัดก็จะขึ้นบรรทัดใหม่ให้ ถ้าไม่ได้กำหนดการวางตำแหน่งของ Component ไว้ทุก Component จะถูกจัดให้วางอยู่กลางบรรทัดเสมอ

- GridLayout ลักษณะคล้ายกับตารางคือ กำหนดแถวและคอลัมน์ก่อน จากนั้นจึงนำ Component ไปใส่ไว้ในแต่ละ Cell ที่สร้างขึ้น

- BorderLayout จะแบ่งพื้นที่ออกเป็น 5 ส่วนคือ North , South, East , West และ Center โดยต้องเพิ่ม Component ลงในส่วนใดๆตามที่ต้องการ

- Panel เป็นการจัดวาง Component ต่างๆไว้ในพื้นที่เล็กๆอย่างเป็นสัดส่วน โดยใช้ Layout Manager ใดๆ ก็ได้ เรียกว่า Panel จากนั้นจึงนำเนื้อที่นี้ไปวางในจอภาพใหญ่อีกทีหนึ่ง ซึ่งจอภาพใหญ่ก็จะจัดวางด้วย Layout Manager อีกแบบหนึ่ง

ตัวอย่าง การเขียน โปรแกรมโดยใช้ package Swing และ AWT

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class RedButton extends JFrame {
    private JButton redButton;

    RedButton () {
        Container c = getContentPane();
        c.setLayout(new FlowLayout());
        redButton = new JButton("Red");
        c.add(redButton);

        public static void main(String[] args) {
            RedButton f = new RedButton();
            f.setSize(300,200);
            f.setVisible(true);
```

```
f.addWindowListener(new WindowAdaper() {
    public void windowClosing (WindowEvent e)
        {System.exit(0); }
});
}
```

คำอธิบาย

1. คำสั่ง import ที่ใช้งาน ได้แก่

```
import javax.swing.*;
```

สำหรับการเรียกใช้งาน Component ในที่นี้คือ JButton

```
import java.awt.*;
```

สำหรับเรียกใช้ method หรือ class อื่นๆ ของ AWT ที่ยังคงต้องใช้งานอยู่ เช่น Container ,
FlowLayout, ฯลฯ

```
import java.awt.event.*;
```

สำหรับใช้ในการตรวจสอบเหตุการณ์ต่างๆที่เกิดขึ้น ในที่นี้คือ ใช้ในที่นี้คือ ใช้ในการตรวจสอบเหตุการณ์การออกจากโปรแกรม

2. ในการกำหนดพื้นที่สำหรับแสดงผล Components ต่างๆและรูปแบบการวางตำแหน่งของ Components จะใช้ 2 คำสั่งต่อไปนี้

```
Container c = get ContenePane();
```

ใช้กำหนดพื้นที่ของ Container ซึ่งใช้สำหรับเป็นที่แสดงผล Components โดยใช้กำหนดให้ Object "c" เป็นตัวแทนของพื้นที่ของ Container

```
c.setLayout (new FlowLayout() );
```

ใช้กำหนดชนิดของ Layout Manager ในโปรแกรมนี้ใช้แบบ FlowLayout คือ วาง Component เรียงลำดับกันจากซ้ายไปขวา

3. ในการสร้างปุ่ม จะใช้คำสั่งต่อไปนี้

```
redButton = new JButton("Red");
```

หมายถึง ให้สร้างปุ่ม 1 ปุ่ม ชื่อ redButton โดยให้ใส่ข้อความลงบนปุ่มด้วย คือข้อความว่า Red โดยต้องกำหนดชนิดของ redButton เป็น JButton ด้วยหรืออาจกำหนดโดยใช้คำสั่งเป็น JButton
redButton = new JButton("Red"); ก็ได้เช่นกัน

```
c.add(redButton);
```

กำหนดให้ redButton แสดงไว้ใน object ชื่อ "c" ซึ่งก็คือ Container ที่มี Layout Manager แบบ FlowLayout ตามที่เรากำหนดไว้

ศุภมาส

ว.ศ. ๒๕๕๖

2.7.4 Events และ Event Handling

2.7.4.1 แนวคิดเกี่ยวกับ Event Handling

๒๕๕๖

Event Handling เป็นขบวนการที่ Application ตอบสนองต่อการกระทำใดๆ (action) ที่เกิดขึ้นต่อ Component ใดๆใน Application โดยการกระทำนั้นจะเกิดจากภายนอก Application โดยผู้ใช้งาน (user) ซึ่งระบบปฏิบัติการ (Operating System) ของเครื่องคอมพิวเตอร์ที่ใช้งานอยู่ จะทำหน้าที่สังเกตการณ์ว่ามีการกระทำใดๆเกิดขึ้นบ้าง เช่น การกดแป้นพิมพ์ใดๆ (keystrokes) หรือการกดปุ่มที่เมาส์ (mouse click) เมื่อมีการกระทำใดๆเกิดขึ้นกับ Component ใด Component นั้นก็จะสร้าง Event หรือเหตุการณ์ ไปยัง Component อื่นที่สนใจต่อการเกิดเหตุการณ์นั้นๆ การกระทำใดๆ 1 อย่าง อาจสร้าง Events ได้มากกว่า 1 Event ได้ เช่นการคลิกเมาส์ 1 ครั้ง จะเกิดเหตุการณ์ 3 เหตุการณ์ คือ mousePressed , mouseReleased และ mouseClicked เป็นต้น ทั้งนี้ แต่ละโปรแกรมจะตัดสินใจว่าจะได้ตอบการกระทำนั้นอย่างไรบ้าง ซึ่งก็ขึ้นอยู่กับการเขียนโปรแกรม

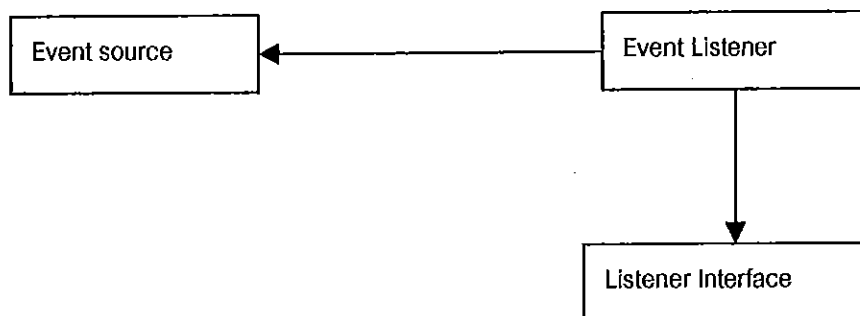
ในการเขียนโปรแกรมกับ GUI นี้ ทุก Components ที่อยู่บนจอภาพสามารถมีเหตุการณ์ใดๆเกิดขึ้นได้ตลอดเวลา Components เหล่านี้จะเรียกว่า Event Sources หรือแหล่งกำเนิดเหตุการณ์ หรือเรียกสั้นๆว่า Source

Source แต่ละชนิดจะมีเหตุการณ์ที่แตกต่างกันไป เช่น ถ้า Source เป็น Button จะมีเหตุการณ์การคลิกเมาส์เลือกปุ่ม (mouse clicks) เหตุการณ์ต่างๆของแต่ละ Component นั้นจะมีอยู่มากมายดังนั้นจึงจำเป็นต้องมีการลงทะเบียน(register)ดักฟัง(listen) เหตุการณ์ที่ต้องการด้วย

Source ใดที่มีการลงทะเบียนดักฟังเหตุการณ์ไว้จะเรียกว่า listener ส่วนเหตุการณ์ที่เกิดขึ้นและมีการตรวจจับได้จะเรียกสั้นๆว่า event listener หรือเรียกสั้นๆว่า event

Source ใดที่ไม่ได้ทำการลงทะเบียนดักฟังเหตุการณ์ไว้ ก็จะไม่มีผลใดๆ เมื่อเกิดเหตุการณ์ขึ้นและ Listener สามารถมีเหตุการณ์ให้เลือกตรวจสอบได้มากกว่า 1 เหตุการณ์ สำหรับ Listener นั้น ถ้าเหตุการณ์ที่เกิดขึ้น ไม่ได้กำหนดการตรวจสอบไว้ใน Listener นั้น ก็จะไม่มีผลใดๆเกิดขึ้น

สรุปการทำงานง่ายๆด้วยภาพต่อไปนี้



4400585

QA
76.9.D3
2639x 8
25440.2

รูปที่ 2.17 การทำงาน ของ Event Handling

Event Source จะเกิดเหตุการณ์ขึ้นได้มากกว่า 1 เหตุการณ์ แต่จะมีการ register ไว้ที่ Event Listener เมื่อมีเหตุการณ์ที่ register ไว้เกิดขึ้น Event Listener ก็จะไปเรียกใช้ Listener Interface ที่เกี่ยวข้องมาทำงาน

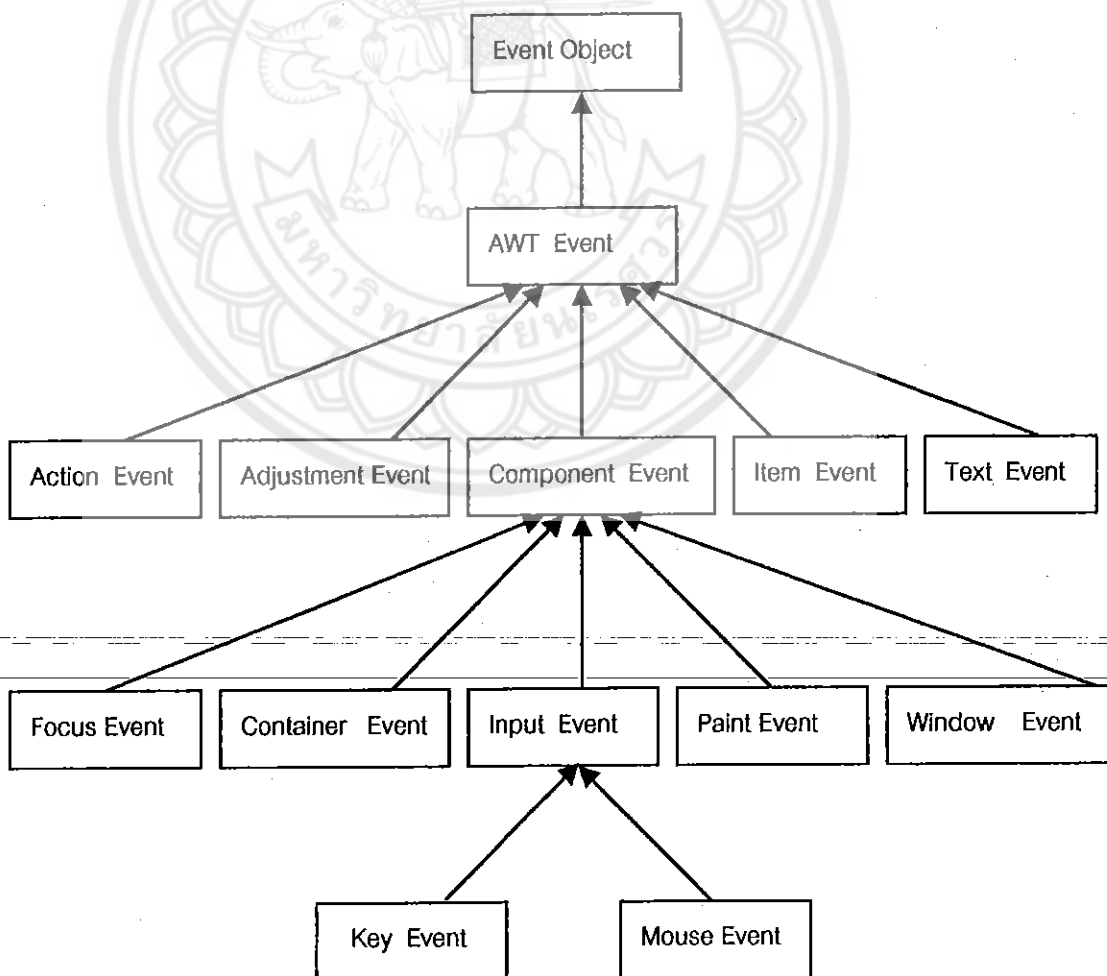
2.7.4.2 ชนิดของ Event (Event Types)

แบ่งออกเป็น 2 ชนิดคือ AWT Events และ Swing Events

2.1 AWT Events เป็นเหตุการณ์ที่เกิดจาก AWT Components โดยแบ่งเป็น 2 ชนิดคือ Component Events และ Semantic Events

2.2 Swing Events เป็นเหตุการณ์ที่เกิดจาก Swing Components และจะไม่ซ้ำกับเหตุการณ์ใน AWT

Event Handling ใน JAVA เป็น Object-Oriented อย่างหนึ่ง เหตุการณ์ต่างๆจะเรียกใช้กันเป็นลำดับชั้น (hierarchy) โดยการถ่ายทอด (inherit) ของ subclass มาจาก Superclassแสดงได้ดังรูป



รูปที่ 2.18 ลำดับชั้นการถ่ายทอด ของ subclass มาจาก Superclass

Event Object ถือเป็น Superclass ของทุกๆ event ใน AWT โดย Subclass หลักก็คือ Swing Component ก็จะมี event object ของตนเองเช่นกัน แต่จะ Event Object ไม่ใช่ AWT Event

2.7.4.3 Event Listeners

Event Listener หรือ Listener เป็น Components หรือ Objects ที่สนใจต่อเหตุการณ์ที่เกิดขึ้นจาก Objects หรือ Components อื่นๆ โดย Listeners จะคอยดักฟัง (listen) เหตุการณ์ต่างๆตลอดเวลาเมื่อเหตุการณ์ที่ต้องการเกิดขึ้น ก็จะไปทำงานตามที่โปรแกรมกำหนดไว้ทันที แต่ถ้ามีเหตุการณ์ที่ตัวเองไม่ได้สนใจเกิดขึ้น ก็จะไม่ทำงานอะไรเช่นกัน

การที่ listeners จะดักฟังเหตุการณ์ใดๆได้นั้น จะต้องทำการ register หรือลงทะเบียนเพื่อขอดักฟังด้วยการลงทะเบียนนี้จะทำได้โดยใช้ Method ที่แตกต่างกัน ดังนี้

```
Public void addXXXListener ( XXXListener )
```

โดย XXX จะเป็นชื่อของ Event Types

สำหรับ Methods ต่างๆที่ใช้ใน Component Class ได้แก่

```
Public void addComponentListener(ComponentListener)
Public void addFocusListener(FocusListener)
Public void addInputMethodListener(InputMethodListener)
Public void addKeyListener(KeyListener)
Public void addMouseListener(MouseListener)
Public void addMouseMotionListener(MouseMotionListener)
```

การยกเลิกการ Register ได้ โดยใช้ Methods ตามคู่ที่ใช้ ดังนี้

```
Public void addXXXListener ( XXXListener )
Public void removeXXXListener ( XXXListener )
```

AWT Event Listener และ event types ต่างๆที่มักนำมาใช้เสมอ แสดงได้ดังตารางที่ 2.4

ตารางที่ 2.3 AWT Event Listener และ event types

Listener interface	Interface methods
ActionListener	actionPerformed(ActionEvent)
MouseListener	mouseClicked(MouseEvent)
	mouseEntered(MouseEvent)
	mouseExited(MouseEvent)
	mousePressed(MouseEvent)
	mouseReleased(MouseEvent)
KeyListener	keyPressed(KeyEvent)
	keyReleased(KeyEvent)
	keyTyped(KeyEvent)

Swing Event Listener และ event types ต่างๆที่มักนำมาใช้เสมอ แสดงได้ดังตารางต่อไปนี้

ตารางที่ 2.4 Swing Event Listener และ event types

Listener interface	Interface methods
MenuItemListener	menuItemClicked(MenuItemEvent)
	menuItemDeselected(MenuItemEvent)
	menuItemSelected(MenuItemEvent)
TableModelListener	columnAdded(TableModelEvent)
	columnMarginChanged(TableModelEvent)
	columnMove(TableModelEvent)
	columnRemoved(TableModelEvent)
	columnSelectionChanged(TableModelEvent)
TableModelListener	tableChanged(TableModelEvent)

2.7.4.4 รูปแบบการเขียน Event Handling

สามารถเขียนโปรแกรมเพื่อตรวจสอบเหตุการณ์ที่เกิดขึ้นได้ 3 รูปแบบคือ

1. การแยกเป็น Method
2. การใช้ inner class

3. การใช้ anonymous inner class

ตัวอย่าง การเขียนโปรแกรมสำหรับตรวจสอบด้วยการแยกเป็น Method โดยต้องการเขียนโปรแกรม โดยมีการทำงานคือ ถ้ามีการคลิกที่ปุ่ม Change ก็จะเปลี่ยนสีของ background ไปเป็นสีแดง

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

class ButtonPanel extends JPanel implements ActionListener {
    private JButton redButton;
    public ButtonPanel()
    {
        redButton = new JButton("Red");
        add(redButton);
        redButton.addActionListener(this);
    }
    public void actionPerformed(ActionEvent evt) {
        Object source = evt.getSource();
        Color color = getBackground();
        If (source == redButton) color = Color.red;
        setBackground(color);
        Repaint();
    }
}

class ButtonFrame extends JFrame {
    public ButtonFrame() {
        setSize(300,200);
        addWindowListener(new WindowAdapter()
        { public void windowClosing(WindowEvent e)
        { System.exit(0); }
        });
        Container c = getContentPane();
        c.add(new ButtonPanel() );
    }
}
```

```

    }
}

public class ButtonTest {
    public static void main(String[] args) {
        JFrame f = new JFrame();
        f.show();
    }
}

```

จะทำการตรวจสอบ โดยแยกส่วนสำหรับตรวจสอบออกมาเป็น Method หนึ่งคือ

```
public void actionPerformed(ActionEvent evt)
```

และใช้คำสั่ง implements ActionListener ที่ class ที่มีการ Register ดังนี้

```
class ButtonPanel extends JPanel implements ActionListener
```

เหตุการณ์ที่เกิดขึ้น จะถูกส่งมาเก็บไว้ใน object ชื่อ evt ซึ่งเราสามารถนำ object นี้มาตรวจสอบได้ โดย Method ที่ใช้ในการตรวจสอบหาชื่อปุ่มก็คือ getSource โดยคำสั่งที่ใช้ก็คือ

```
Object source = evt.getSource();
```

ค่าที่ได้ก็คือ ชื่อของปุ่มที่ถูกคลิก ซึ่งจะมีชนิดเป็น object เสมอ

การทำงานของโปรแกรมนี้ก็คือ ถ้ามีการคลิกปุ่ม redButton ก็จะเปลี่ยนสีของ background ไปเป็นสีแดง โดยใช้ method ชื่อ setBackground ในการเปลี่ยนสีจอภาพ

2.8 ภาษา Java กับ ฐานข้อมูล

2.8.1 JDBC(Java Database Connectivity)

JDBC [5] คือ relational database API classes(วิธีการติดต่อกับระบบฐานข้อมูลเชิงสัมพันธ์) สำหรับ Java Applets และ Java Application สามารถใช้ได้กับคอมพิวเตอร์ทุกชนิด (Platform) ใช้กับ database ได้หลายชนิดพัฒนาจากบริษัท Sun-Microsystem-JDBC ได้รับการพัฒนาโดยมีรูปแบบคล้ายกับ ODBC(Open Database Connectivity) ของบริษัท Microsoft เพราะว่าทั้งคู่อุอยู่บนพื้นฐานของ X/Open SQL CLI (Call Level Interface)เช่นเดียวกัน และทางบริษัท Sun ยังได้พัฒนาให้ JDBC สามารถเชื่อมต่อกับ ODBC ได้อีกด้วย

โดยในปัจจุบันการเชื่อมต่อ database ผ่าน JDBC ที่เราพบเห็นบ่อยๆ จะเป็นแบบการต่อผ่าน ODBC และ การต่อกับ database ที่มี JDBC Driver อยู่ในตัวเอง และอีกแบบคือ แบบ Three-tier ที่มี ตัวกลาง(broker) เป็นตัวผ่าน โดยไม่ต้องการ Native Libraries ของ database นั้น ซึ่งโดยปกติ เราต้องนำ Native Libraries ของ database ที่เราใช้อยู่ไปลงในฝั่ง client ด้วย (concept ของ two- tier)อย่างเช่น การใช้ CORBA RMI เป็นต้น

2.8.2 JDBC-ODBC

ODBC คือ API (Application Programming Interface – เป็นข้อกำหนดรูปแบบของฟังก์ชันที่ใช้กับ ODBC เป็นวิธีการติดต่อกับฐานข้อมูลรูปแบบหนึ่ง ของ บริษัท Microsoft วิธีการติดต่อกับฐานข้อมูลรูปแบบอื่นๆ ของบริษัท Microsoft ที่เกี่ยวข้องกับฐานข้อมูลคล้ายกับ ODBC คือ

- OLE DB (OLE Database) : ใช้งานในระดับต่ำ ต้องเขียนโปรแกรมในระบบ Low-level
- ADO (ActiveX Data Objects) : ไม่สามารถใช้ได้กับ Java
- RDS (Remote Data Service) : ไม่สามารถใช้ได้กับ Java และไม่สามารถใช้งานในแบบ

Portability ได้

ODBC ถูกใช้งานอย่างกว้างขวางในการติดต่อกับระบบฐานข้อมูลเชิงสัมพันธ์(Relational Database)โดยทั่วไป และตัว ODBC ถูกสร้างมาจากโปรแกรมภาษา C เป็นส่วนใหญ่ ดังนั้นในการเขียนโปรแกรมเพื่อเรียกใช้งานฟังก์ชันของ ODBC-จึงไม่ถายนักถ้าเขียนโดยภาษา Java แต่ถึงอย่างไรก็ตาม Java ได้สร้าง API ในชื่อ JDBC-ODBC Bridge เพื่อเป็นสะพานเชื่อมต่อกับ ODBC ด้วยภาษาจาวาให้มากยิ่งขึ้น

JDBC ติดต่อกับระบบฐานข้อมูลด้วยคำสั่ง SQL ดังนั้นจึงไม่เป็นการยากที่จะเขียน โปรแกรมด้วย Java และให้ใช้งานฐานข้อมูลได้

เครื่องมือที่ช่วยในการเขียน โปรแกรม Java ที่เชื่อมต่อกับ Database นั้น ได้แก่ java.sql package (เราเรียก class ของ Java ว่า package)ซึ่งเราจะเรียกใช้ package นี้ได้โดยคำสั่ง import java.sql . *

เราสามารถสร้าง โปรแกรมที่เชื่อมต่อกับ database ได้โดย จะมีเครื่องมือที่เกี่ยวข้องดังนี้คือ

- Java 2 (JDK)

- DBMS

- Driver ของ DBMS : JDBC-ODBC Bridge

ในขณะที่ผู้พัฒนาโครงการ ใช้ Microsoft Access เนื่องจากมี driver ของ Microsoft Access อยู่ใน ODBC แล้ว และเลือกใช้ JDBC-ODBC Bridge เพื่อติดต่อกับฐานข้อมูล เพื่อความสะดวกในการเขียนโปรแกรม

2.8.3 ตัวอย่างในการ connect database แบบการต่อ ผ่าน ODBC ของ Java

ขั้นตอนที่ 1. มีการ import JDBC API มาใช้

```
import java.sql.*;
public class Jdbc {
public static void main (String []args){
```

ขั้นตอนที่ 2. ประกาศตัวแปร Statement กับ ResultSet

```
Statement stmt;
ResultSet rs
try {
```

ขั้นตอนที่ 3 เรียกใช้ odbc driver

```
//Load Driver ขึ้นมาใช้และ DBMS ต้อง support JDBC ด้วย
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
```

ขั้นตอนที่ 4. ทำการ connect database ผ่าน jdbc - odbc

```
Connection con = DriverManager.getConnection("jdbc:odbc:db1","","");
```

ขั้นตอนที่ 5. ทำการสร้าง Statement มาใช้

```
stmt = con.createStatement();
```

ขั้นตอนที่ 6 สร้าง sql statement

```
String sql = "Select * From Student"
```

ขั้นตอนที่ 7. นำ result ที่เราดึงจาก DBMS มา จากการ Query มาใส่ใน ResultSet

```
rs = stmt.executeQuery(sql);
```

ขั้นตอนที่ 8. ทำไปเรื่อยๆจนหมด resultset

```
while (rs.next()){  
    // ค่าแต่ละ field ใน table Student มา print ทางหน้าจอ  
    System.out.println(rs.getInt("id"));  
    System.out.println(rs.getString("name"));  
    System.out.println(rs.getString("position"));  
}  
} catch(Exception e){ }
```

ขั้นตอนที่ 9. ต้องมีการ close connection และ statement ทุกครั้งเพื่อลดการ lag ของ memory

```
stmt.close();  
con.close();  
}  
}
```



บทที่ 3

การออกแบบระบบฐานข้อมูลอาคารสถานที่

3.1 ศึกษาและรวบรวมข้อมูลของระบบการเก็บข้อมูลเกี่ยวกับอาคารและอุปกรณ์ในอาคารที่เป็นระบบงานเดิม(ใช้ข้อมูลของอาคารเรียนคณะวิศวกรรมศาสตร์)

3.1.1 ระบบงานการจัดตารางการใช้ห้องเรียน(Room Timetable)ของคณะวิศวกรรมศาสตร์

3.1.1.1 ทางฝ่ายวิชาการงานจัดการเรียนการสอนของมหาวิทยาลัยจะทำหน้าที่จัดตารางการใช้ห้องเรียนของวิชาพื้นฐาน โดยใช้โปรแกรมสำเร็จรูปจัดตารางเรียน

ส่วนวิชาเฉพาะของแต่ละคณะนั้น ทางฝ่ายวิชาการของแต่ละคณะจะส่งตารางการใช้ห้องเรียนมาให้เจ้าหน้าที่ฝ่ายวิชาการงานจัดการเรียนการสอนของมหาวิทยาลัยจากนั้น ทางเจ้าหน้าที่ของมหาวิทยาลัยจะรวบรวมข้อมูลและส่งตารางการใช้ห้องเรียนเป็นเอกสารสมบูรณ์กลับมาที่แต่ละคณะ

3.1.1.2 เลขานุการคณะวิศวกรรมศาสตร์ รับผิดชอบตารางการใช้ห้องเรียนของอาคารเรียน เมื่อมีการขอเปลี่ยนแปลงการใช้ห้องเรียน จะแบ่งเป็น 2 กรณี

1. ขอเปลี่ยนแปลงห้องเรียนตอนต้นเทอมตลอดทั้งภาคเรียน เมื่อมีผู้ต้องการจองห้อง ซึ่งก็จะเป็นทั้ง อาจารย์หรือนิสิต มาแจ้งความจำเป็นขอเปลี่ยนห้องเรียนหรือเวลาเรียน เจ้าหน้าที่ฝ่ายวิชาการงานจัดการเรียนการสอนของมหาวิทยาลัย จะตรวจสอบจากเอกสารตารางการใช้ห้องเรียน(Room Timetable)ว่าห้องที่ผู้จองต้องการจองนั้นว่างตรงกับเวลาที่ต้องการหรือไม่ ถ้าหากว่าห้องเรียนนั้นไม่ว่างตามเวลาที่ต้องการก็จะเปลี่ยนเป็นห้องอื่นที่เหมาะสม หรือถ้ามีความจำเป็นต้องใช้ห้องเรียนนั้นจริงๆ ก็จะเปลี่ยนเป็นช่วงเวลาอื่นให้

ถ้าหากว่าผู้ต้องการจองห้องไม่ทราบว่าห้องใดบ้างที่มีอุปกรณ์และขนาดของห้องที่เหมาะสมที่จะใช้งานตามต้องการ เช่น ถ้าผู้ต้องการจองห้องต้องการห้องที่มี เครื่องปรับอากาศ ขนาดความจุห้อง 80 คน ทางเจ้าหน้าที่ฝ่ายวิชาการงานจัดการเรียนการสอนของมหาวิทยาลัย จะต้องทำการตรวจสอบจากเอกสารแผนผังอุปกรณ์ไฟฟ้าในอาคารเรียน และขนาดของห้อง ของแต่ละอาคารซึ่งได้รับข้อมูลที่เป็นเอกสารมาจากนักวิชาการฝ่ายงานอาคารของแต่ละคณะ เพื่อค้นหาห้องที่มีอุปกรณ์ไฟฟ้าคือ เครื่องปรับอากาศ และมีขนาดความจุห้องมากกว่า 80 คน ตามที่ต้องการ หรือผู้มาจองห้อง ต้องการจองห้องที่มีอุปกรณ์พิเศษ เช่น โปรเจคเตอร์และ คอมพิวเตอร์ ทางเจ้าหน้าที่จะต้องทำการตรวจสอบจากเอกสารอุปกรณ์การเรียนการสอน ที่ได้รับมาจากนักวิชาการฝ่ายโสตทัศนศึกษาของแต่ละคณะ เพื่อค้นหาห้องที่มีลักษณะตามนั้น จากนั้นเจ้าหน้าที่ฝ่ายวิชาการงานจัดการเรียนการสอนของมหาวิทยาลัย จะตรวจสอบจากเอกสารตารางการใช้ห้องเรียน(Room Timetable) อีกครั้งเพื่อให้ได้ห้องเรียนที่

ตรงกับความต้องการ หลังจากทำการแก้ไขเปลี่ยนแปลงเรียบร้อยแล้ว ก็จะจัดทำตารางการใช้ห้องเรียน(Room Timetable)ฉบับสมบูรณ์ แล้วส่งให้เลขานุการคณะวิศวกรรมศาสตร์ จากนั้นเลขานุการคณะวิศวกรรมศาสตร์ ก็จะพิมพ์ตารางการใช้ห้องเรียนของแต่ละห้อง คิดไว้ที่หน้าห้องเรียนทุกห้อง

2 ขอเปลี่ยนแปลงห้องเรียนในโอกาสพิเศษ เช่น การจัดสัมมนา ขอใช้ ในงานชมรมต่างๆ ชั่วโมงเรียนที่ต้องใช้อุปกรณ์พิเศษ ฯลฯ การค้นหาห้องเรียนและตรวจสอบเวลาว่าง มีลักษณะเดียวกับขอเปลี่ยนแปลงห้องเรียนตลอดทั้งภาคเรียนแต่เป็นหน้าที่รับผิดชอบของเลขานุการคณะวิศวกรรมศาสตร์โดยจะ ตรวจสอบห้องเรียนจากเอกสารตารางการใช้ห้องเรียน(Room Timetable)ที่ได้รับมาจากเจ้าหน้าที่ฝ่ายวิชาการงานจัดการเรียนการสอนของมหาวิทยาลัย และจะมีเอกสารการขออนุมัติใช้ห้อง บันทึกว่า มีบุคคลกลุ่มใด มาใช้ห้องเวลาใด ใช้ในกิจกรรมใด และต้องการใช้ อุปกรณ์ใดบ้าง เพื่อที่จะติดต่อกับทางแม่บ้านให้เปิดห้องให้ตามเวลา

3.1.2 ตัวอย่างตารางการใช้ห้องเรียน (Room Timetable)

2/2543

Room timetable – EN 205. (Wks 1-16)

..... Monday

Day/Time Monday 08:00-09:00 Wks 1-16

Course 252112 gr.101-104

Staff F01036 Sujiya, F01005 PraMote P., F01027 Anchalee

Rooms EN 207, EN 210, EN 212, EN 205

..... Tuesday

Day/Time Tuesday 08:00-09:30 Wks 1-16

Course 252112 gr.101-104

Staff F01036 Sujiya, F01005 PraMote P., F01027 Anchalee

Rooms EN 207, EN 210, EN 212, EN 205

Room timetable – EN 207. (Wks 1-16)

.....Monday

Day/Time Monday 08:00-09:00 Wks 1-16

Course 252112 gr.101-104

Staff F01036 Sujiya, F01005 PraMote P., F01027 Anchalee

Rooms EN 207, EN 210, EN 212, EN 205

Day/Time Monday 14:00-15:30 Wks 1-16

Course 255111 gr.101-104

Staff F01010 Prirode, F01009 Watcharapom, F01042 Sriwan, F01040 Anamai

Rooms EN 207, EN 210, EN 212, EN 205

.....Tuesday

Day/Time Tuesday 08:00-09:30 Wks 1-16

Course 252112 gr.101-104

Staff F01036 Sujiya, F01005 PraMote P., F01027 Anchalee

Rooms EN 207, EN 210, EN 212, EN 205

Room timetable – EN 210. (Wks 1-16)

.....Monday

Day/Time Monday 08:00-09:00 Wks 1-16

Course 252112 gr.101-104

Staff F01036 Sujiya, F01005 PraMote P., F01027 Anchalee

Rooms EN 207, EN 210, EN 212, EN 205

3.1.3 รายละเอียดเกี่ยวกับกลุ่มอาคารเรียนคณะวิศวกรรมศาสตร์ ทั้งหมด 8 อาคารเรียน ประกอบไปด้วย

1. อาคารเรียนรวม(Engineering Building , EN) เปิดให้นักศึกษาใช้ได้ทั้งมหาวิทยาลัย
2. อาคารเรียนภาควิชาวิศวกรรมอุตสาหกรรมและภาควิชาวิศวกรรมเครื่องกล(Industrial Mechanical Engineering , IE) เปิดให้นักศึกษาใช้ได้ทั้งมหาวิทยาลัย เฉพาะชั้น 6 และ 7

3. อาคารเรียนภาควิชาวิศวกรรมโยธา(Civil Engineering , CE) ,มีครุภัณฑ์ประกอบการเรียนการสอน เฉพาะชั้น 5 ใช้และใช้ในการเรียนการสอนภายในคณะวิศวกรรมศาสตร์เท่านั้น

4. อาคารเรียนภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์(Electrical – Computer Engineering , EE) ยังไม่มีครุภัณฑ์ประกอบการเรียนการสอน และใช้ในการเรียนการสอนภายในคณะวิศวกรรมศาสตร์เท่านั้น

5. อาคารปฏิบัติการ 1 ภาควิชาโยธา (Lab-CE)ใช้ในเฉพาะคณะวิศวกรรมศาสตร์

6. อาคารปฏิบัติการ 2 ภาควิชาวิศวกรรมไฟฟ้า (Lab EE)ใช้ในการเรียนการสอนภายในคณะวิศวกรรมศาสตร์เท่านั้น

7. อาคารปฏิบัติการ 3 ภาควิชาวิศวกรรมอุตสาหกรรม (Lab IE)ใช้ในการเรียนการสอนภายในคณะวิศวกรรมศาสตร์เท่านั้น

8. อาคารปฏิบัติการ 4 ภาควิชาวิศวกรรมเครื่องกล (Lab ME)ใช้ในการเรียนการสอนภายในคณะวิศวกรรมศาสตร์เท่านั้น

โดยรหัสของห้องเรียนภายในกลุ่มอาคารเรียนคณะวิศวกรรมศาสตร์ มีทั้งหมด 5 ตัวด้วยกันเป็นตัวอักษร 2 ตัว ตัวเลข 3 ตัว มีความหมายดังนี้

- อักษร 2 ตัวแรก เป็นรหัสของอาคารเรียน เช่น EN ,IE ,CE,EE เป็นต้น
 - เลขตัวแรก เป็นรหัสชั้น
 - เลข 2 ตัวสุดท้ายเป็น เลขที่ห้อง
- ยกตัวอย่างเช่น EN 205
- EN หมายถึง อาคารเรียนรวม
 - 2 หมายถึง ชั้นที่ 2
 - 05 หมายถึง ห้องที่ 5

3.1.4 ข้อมูลอุปกรณ์ในห้อง

ฝ่ายโสตทัศนศึกษาทำหน้าที่จัดการเอกสารเหล่านี้โดยจะมีการสำรวจทุกภาคเรียน และแต่ละห้องจะมีเอกสารรายละเอียดของอุปกรณ์ทั้งหมดในห้อง ชื่ออุปกรณ์ จำนวน และ หมายเลขครุภัณฑ์

3.1.5 ตัวอย่างเอกสาร

ชื่ออุปกรณ์	จำนวน	หมายเลขครุภัณฑ์
ชุดเครื่องขยายเสียง	1ชุด	XXXXXXXXXX
โปรเจคเตอร์	-	-
เก้าอี้	80	XXXXXXXXXX – XXXXXXXXXXXX

3.2 รวบรวมความต้องการของผู้ใช้ เพื่อเป็นแนวทางในการออกแบบฐานข้อมูล ซึ่งความต้องการที่รวบรวมมาได้มีดังนี้

1. ต้องการเก็บข้อมูลห้องกับอุปกรณ์ภายในห้อง ได้แก่ รหัสอาคารเรียน เลขที่ห้อง ขนาดความจุห้อง อุปกรณ์ที่ใช้ในการเรียนการสอน อุปกรณ์ไฟฟ้า-โต๊ะ-เก้าอี้

2. ต้องการเก็บข้อมูลกิจกรรมที่เกิดขึ้นภายในห้อง ได้แก่ ชื่อกิจกรรม หรือวิชาเรียน เวลาที่ใช้ห้อง วันที่ใช้ห้อง บุคคลที่ใช้ห้อง

3.2.1 ระบบงานที่จัดทำขึ้นใหม่

1. จัดทำฐานข้อมูล ตารางการใช้งานใช้ห้อง
2. จัดทำฐานข้อมูล รายละเอียดการจองห้อง
3. จัดทำฐานข้อมูล อุปกรณ์ภายในห้อง
4. โปรแกรมจัดการฐานข้อมูล โปรแกรมจะมีความสามารถดังนี้คือ ค้นหาข้อมูลรายละเอียดการใช้ห้องเรียน แต่ละห้องได้ ค้นหาข้อมูลอุปกรณ์ภายในห้องเรียนได้ ส่วนการเพิ่มข้อมูล ลบข้อมูล ปรับปรุงข้อมูล นั้น โปรแกรมจะทำการตรวจสอบรหัสผ่านของเจ้าหน้าที่ด้วย

3.2.2 ระบบงานการจัดตารางการใช้ห้องเรียน(Room Timetable)ของคณะวิศวกรรมศาสตร์ เมื่อใช้โปรแกรมที่พัฒนาขึ้น

1. ทางฝ่ายวิชาการงานจัดการเรียนการสอนของมหาวิทยาลัยจะทำหน้าที่จัดตารางการใช้ห้องเรียนของวิชาพื้นฐาน โดยใช้โปรแกรมสำเร็จรูปจัดตารางเรียน ส่วนวิชาเฉพาะของแต่ละคณะนั้น ทางฝ่ายวิชาการของแต่ละคณะจะส่งตารางการใช้ห้องเรียนมาให้เจ้าหน้าที่ฝ่ายวิชาการงานจัดการเรียนการสอนของมหาวิทยาลัย หลังจากนั้นทางเจ้าหน้าที่ของมหาวิทยาลัยจะรวบรวมข้อมูลและส่งตารางการใช้ห้องเรียนเป็นเอกสารมาที่แต่ละคณะ

2. เมื่อมีการขอเปลี่ยนแปลงการใช้ห้องเรียน จะแบ่งเป็น 2 กรณี

2.1 ขอเปลี่ยนแปลงห้องเรียนตลอดทั้งภาคเรียน เมื่อมีผู้ต้องการจองห้องซึ่งก็จะเป็นทั้งอาจารย์หรือนิสิต มาแจ้งความจำนงขอเปลี่ยนห้องเรียนหรือเวลาเรียน เจ้าหน้าที่ฝ่ายวิชาการจะใช้โปรแกรมเพื่อตรวจสอบห้องเรียนที่ต้องการจองได้ หากห้องเรียนมีเวลาว่างตามที่ต้องการ ก็ทำการจองห้องได้ในทันทีและข้อมูลก็จะถูกเก็บอยู่ที่ ฐานข้อมูล ตารางการใช้ห้องเรียน (Room Timetable) หากว่าห้องเรียนนั้นไม่ว่างตามเวลาที่ต้องการก็จะใช้โปรแกรมค้นหาห้องอื่นที่เหมาะสมให้ หรือถ้ามีความจำเป็นต้องใช้ห้องเรียนนั้นจริงๆ ก็จะเปลี่ยนเป็นช่วงเวลาอื่นให้ จากนั้นก็ทำการจองห้องเรียนนั้นๆ และข้อมูลก็จะถูกเก็บอยู่ที่ ฐานข้อมูล ตารางการใช้ห้องเรียน (Room Timetable)

ถ้าหากว่าผู้ต้องการของห้องไม่ทราบว่าห้องใดบ้างที่มีอุปกรณ์และขนาดของห้องที่เหมาะสมที่จะใช้งานตามต้องการ เช่น ถ้าผู้ต้องการของห้องต้องการห้องที่มี เครื่องปรับอากาศ ทางเจ้าหน้าที่ฝ่ายวิชาการจะใช้โปรแกรมเพื่อค้นหาห้องที่มีเครื่องปรับอากาศ หรือ ผู้ต้องการของห้อง ต้องการของห้องที่มี โปรเจคเตอร์และ คอมพิวเตอร์ ทางเจ้าหน้าที่จะใช้โปรแกรมเพื่อค้นหาห้องที่มี โปรเจคเตอร์และ คอมพิวเตอร์ และอุปกรณ์อื่นๆ ตามที่ต้องการได้ จากนั้นเจ้าหน้าที่ฝ่ายวิชาการจะใช้โปรแกรมตรวจสอบกับฐานข้อมูล ตารางการใช้ห้องเรียน(Room Timetable) อีกครั้งเพื่อให้ได้ห้องเรียนที่ตรงกับอุปกรณ์และเวลาที่ผู้มาของต้องการ หลังจากทำการแก้ไขเปลี่ยนแปลงเรียบร้อยแล้ว ก็จัดทำตารางการใช้ห้องเรียน(Room Timetable)ฉบับสมบูรณ์ แล้วเลขานุการคณะวิศวกรรมศาสตร์ก็จะพิมพ์ตารางการใช้ห้องเรียนของแต่ละห้อง คัดไว้ที่หน้าห้องเรียนทุกห้อง

2.2 ขอเปลี่ยนแปลงห้องเรียนในโอกาสพิเศษ เช่น การจัดสัมมนา ขอใช้ในงานชมรมต่างๆ ชั่วโมงเรียนที่ต้องใช้อุปกรณ์พิเศษ ฯลฯ การค้นหาห้องเรียนและตรวจสอบเวลาว่าง มีลักษณะเดียวกับขอเปลี่ยนแปลงห้องเรียนตลอดทั้งภาคเรียน แต่จะต้องมีฐานข้อมูลอีกฐานข้อมูลหนึ่งเพื่อบันทึกว่ามีบุคคลกลุ่มใด มาใช้ห้องเวลาใด ใช้ในกิจกรรมใดด้วย โดยการทำเรื่องของของห้องจะมีช่วงเวลาที่กำหนดว่าต้องมาก่อนเป็นเวลาเท่าใด เพื่อที่เลขานุการคณะวิศวกรรมศาสตร์ จะได้ติดต่อกับทางแม่บ้านให้เปิดห้องให้ได้ตามเวลา

3.3 วิเคราะห์ข้อมูลที่รวบรวมมา

นำมาเขียนแผนภาพการไหลของข้อมูล(Data Flow Diagram)

3.4 แผนภาพ ER

เขียนขึ้นเพื่อดูความสัมพันธ์ของข้อมูลในแต่ละระบบ

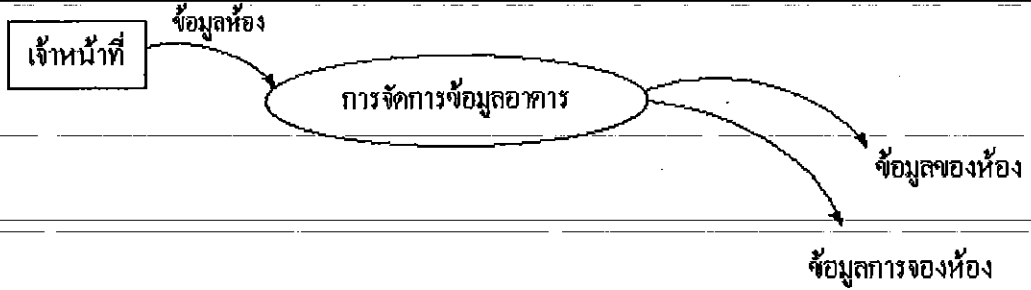
3.5 ทำการออกแบบฐานข้อมูล

โดยใช้วิธีการ Normalization เพื่อให้ข้อมูลที่จัดเก็บในแต่ละระบบไม่ซ้ำกัน

3.6 ออกแบบในส่วนของโปรแกรม(Application Design)

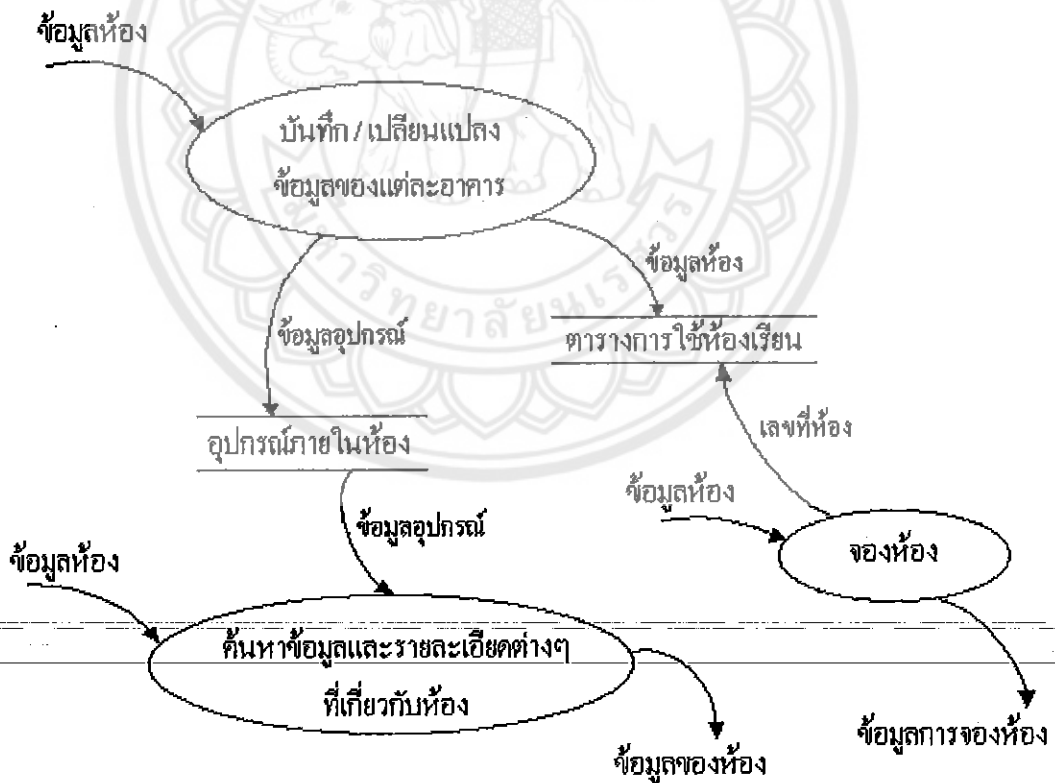
ออกแบบโปรแกรมจากนั้นพัฒนาต้นแบบ(Prototype)ของระบบงาน ปรับปรุงระบบ และแก้ไขข้อผิดพลาด แล้วนำโปรแกรมที่ปรับปรุงแล้วมาทดสอบและแก้ไขข้อผิดพลาดของโปรแกรม

DFD0



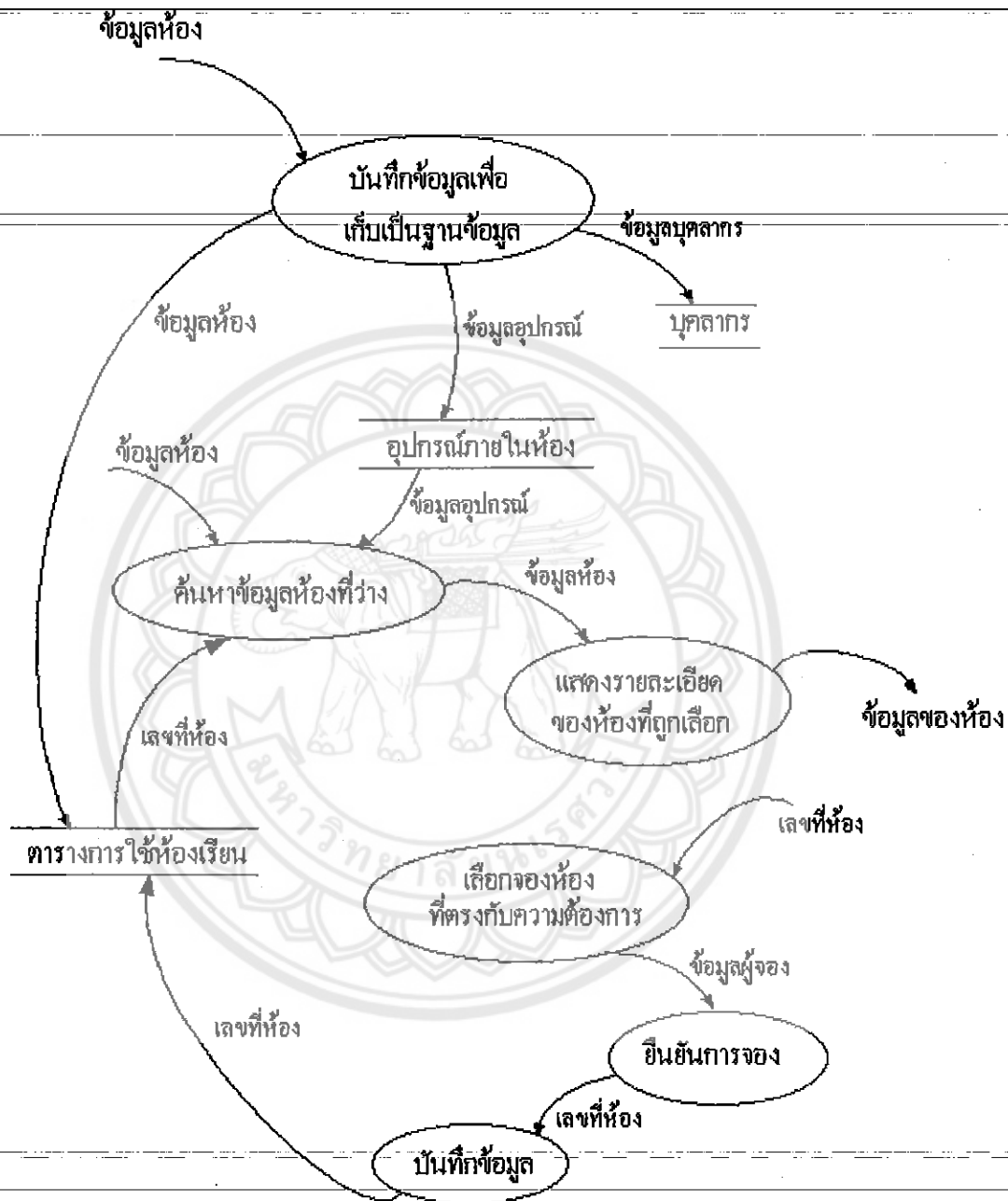
รูปที่ 3.1 DFD level 0

DFD1

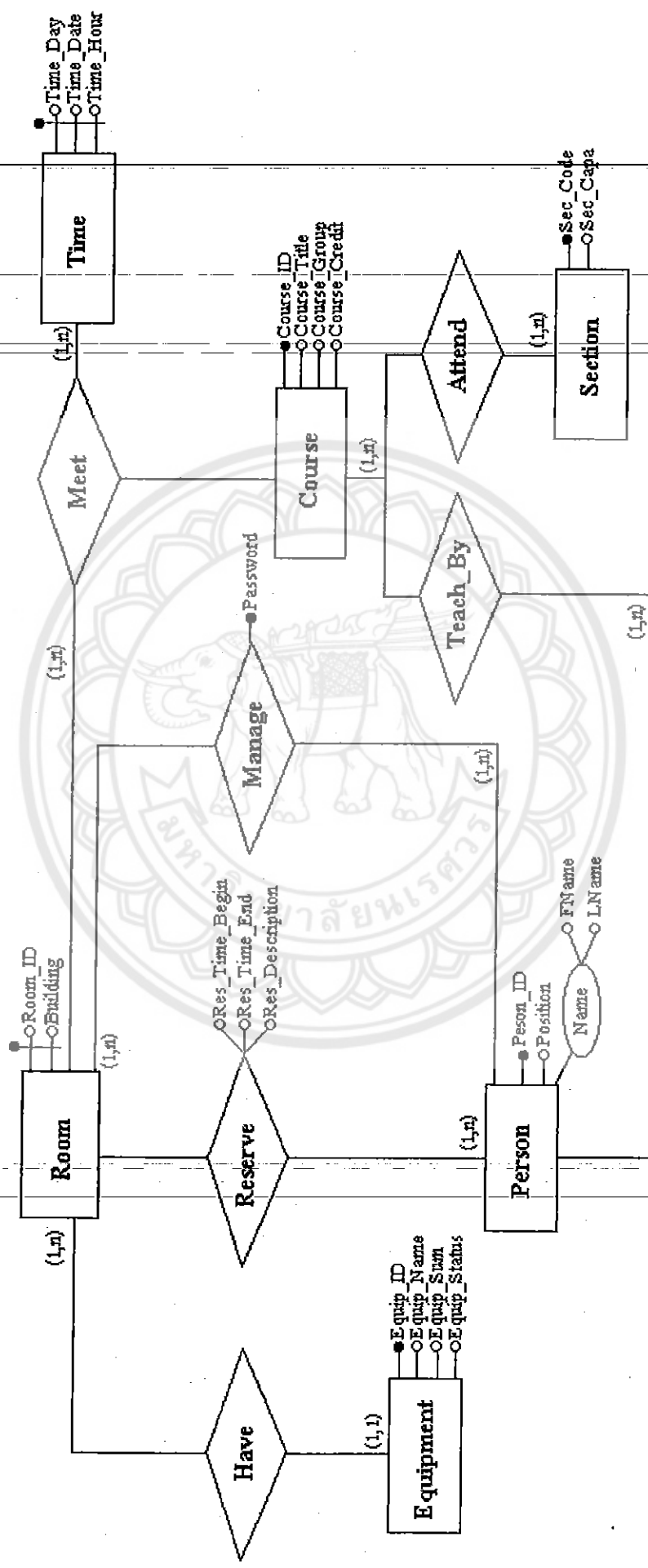


รูปที่ 3.2 DFD level 1

DFD2



รูปที่ 3.3 DFD level 2



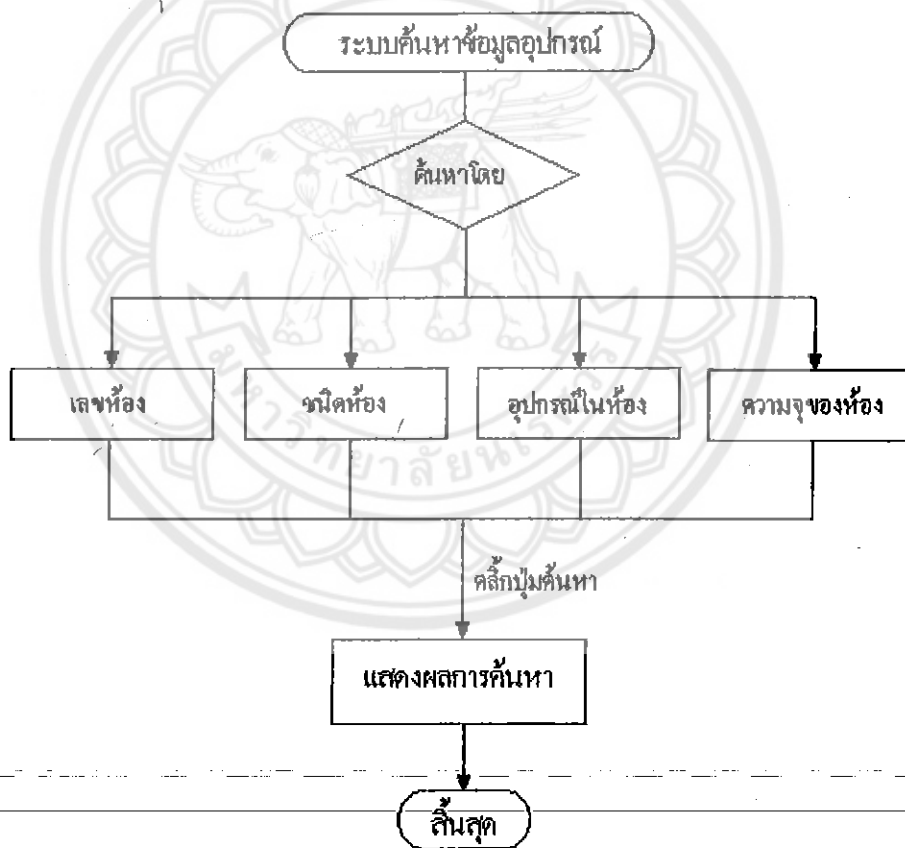
รูปที่ 3.4 ER Diagram

การพัฒนาโปรแกรมระบบฐานข้อมูลสารสนเทศอาคารสถานที่

โปรแกรมจัดการฐานข้อมูลนี้ถูกเขียนขึ้นโดยใช้โปรแกรม JDK 1.2 (Java Development Kit Version 1.2) ติดต่อกับฐานข้อมูล Microsoft Access 97 ผ่าน ODBC โดยการพัฒนาโปรแกรมมีขั้นตอน แนวคิด และ ลักษณะ โปรแกรมดังนี้

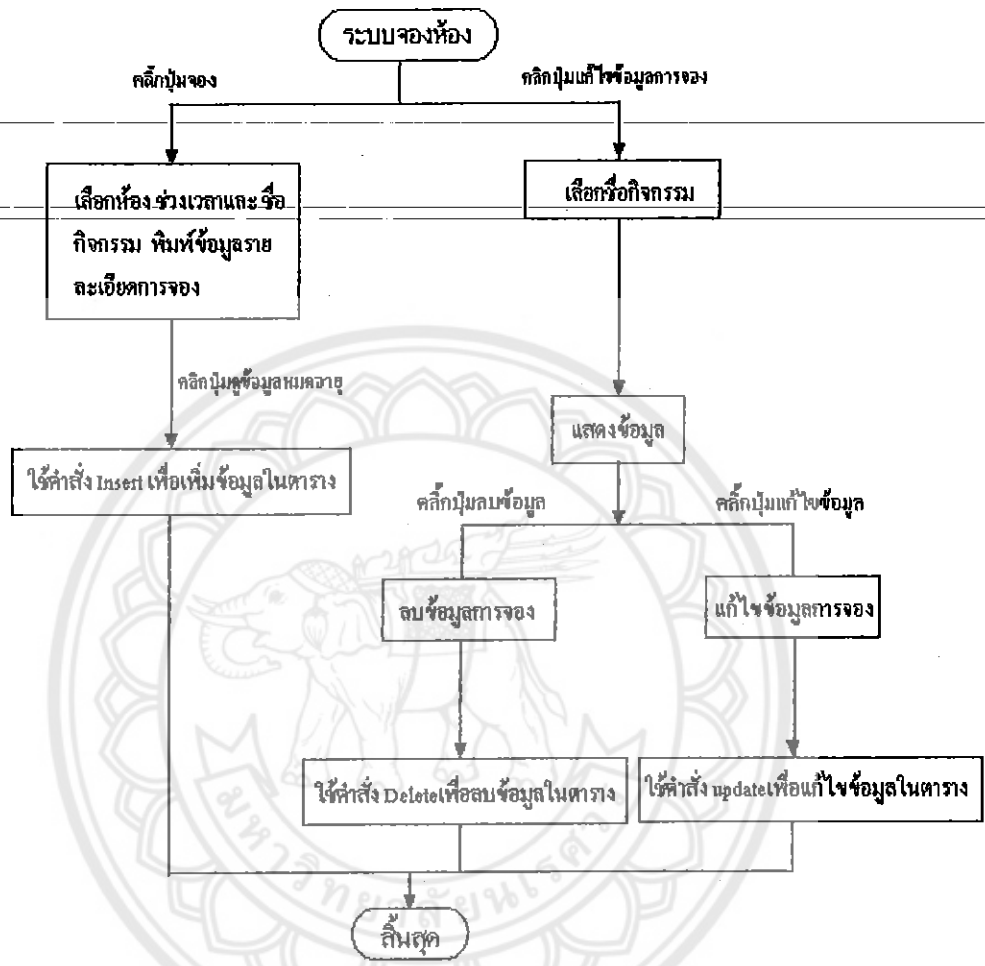
4.1 แผนภาพการทำงานของโปรแกรม

ระบบค้นหาข้อมูลอุปกรณ์



รูปที่ 4.1 Flow Chart โปรแกรมระบบค้นหาข้อมูลอุปกรณ์

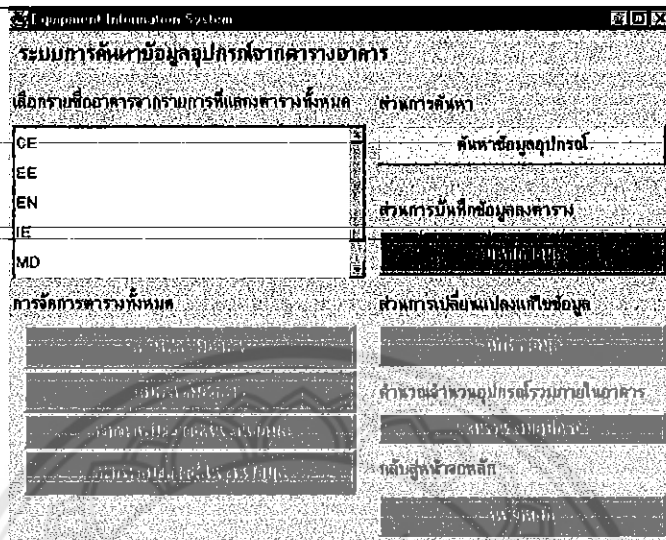
ระบบการจอง



รูปที่ 4.2 Flow Chart โปรแกรมระบบการจอง

4.2 รูปแบบโปรแกรมฐานข้อมูลสารสนเทศอาคารสถานที่

ระบบการค้นหาข้อมูลอุปกรณ์ภายในอาคาร

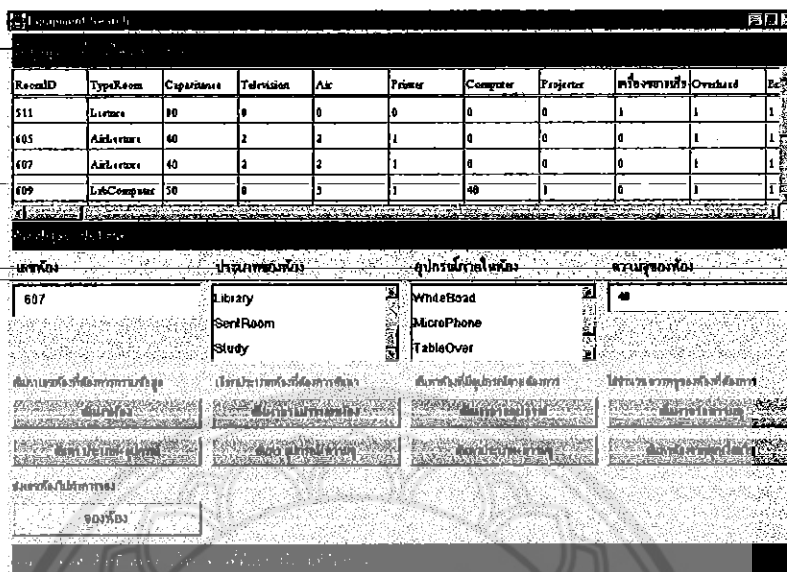


รูปที่ 4.3 ลักษณะของ โปรแกรมหน้าหลักของระบบการค้นหาข้อมูลอุปกรณ์ภายในอาคาร

เป็นหน้าหลักที่เชื่อมต่อกับส่วนต่างๆที่เกี่ยวข้องกับข้อมูลอาคาร โดยจะมีการแสดงรายชื่ออาคารทั้งหมดที่มีในฐานข้อมูล สามารถเลือกได้ว่าจะค้นหา บันทึก แก้ไขข้อมูล ที่ตารางอาคารไหน สามารถหาจำนวนรวมอุปกรณ์แต่ละอุปกรณ์ได้จากอาคารที่มี โดยเป็นการคำนวณจากข้อมูลตัวเลขของอุปกรณ์แต่ละชนิด

ส่วนการจัดการตารางอาคาร สามารถสร้างตารางอาคารที่จะมีห้อง ประเภทของห้อง และ ความจุเป็นข้อมูลหลัก หรือลบตารางที่ทำการสร้างขึ้นมาได้ จากนั้นสามารถเพิ่มหรือลบชนิดของอุปกรณ์ที่จะเพิ่มเข้าไปในฐานข้อมูล โดยใช้การเพิ่ม หรือ ลบ Field

ส่วนการค้นหา



รูปที่ 4.4 ลักษณะของโปรแกรมระบบการค้นหาข้อมูลอุปกรณ์ภายในอาคาร

ส่วนการค้นหาข้อมูลอุปกรณ์จากตารางอาคารแต่ละอาคารที่ได้เลือกไว้ โปรแกรมสามารถค้นหาห้องต่างๆ ได้จากการ

- ใส่เลขห้อง จะ ได้ข้อมูลของห้องนั้นจำนวนหนึ่งระเบียน
- เลือกชนิดของห้องหรือประเภทของห้อง จากรายการที่กำหนดให้ ซึ่งเป็นรายชื่อประเภทของห้องทั้งหมดที่มีภายในฐานข้อมูลอาคารนั้น เช่น ห้องเรียน ห้องประชุม ระเบียนที่ได้จะแสดงจำนวนห้องทั้งหมดที่ตรงกับความต้องการ
- ค้นหาอุปกรณ์ที่มีในแต่ละห้อง โดยแสดงเป็นรายการซึ่งสามารถเลือกได้มากกว่าหนึ่งรายการ(สังเกตการเลือกได้จากแถบอักษรด้านล่าง) การค้นหาจะแสดงเฉพาะห้องที่มีอุปกรณ์เครื่องใช้ที่ได้เลือกไว้
- ใส่ค่าความจุของห้อง หรือ ขนาดของห้อง เช่น 20 (ห้องจุได้ประมาณ 20 คน)

โปรแกรมจะค้นหาห้องที่มีความจุมากกว่าหรือเท่ากับที่ใส่ค่าเข้าไป

ค้นหาข้อมูลได้จากการกดที่ปุ่มค้นหาต่างๆ ภายใต้ข้อกำหนดนั้น ในส่วนปุ่มการค้นหาแล้วที่ สองเป็นการค้นหามากกว่าหนึ่งเงื่อนไข โดยใช้ข้อกำหนดที่มีให้คือ ประเภทของห้อง อุปกรณ์ และ ค่าความจุของห้อง มารวมกัน

เมื่อค้นหาห้องที่ตรงกับความต้องการ ได้แล้วก็สามารถส่งค่าเลขห้องที่ได้เลือกไว้ไปในส่วนของการจองห้องต่อไป

ส่วนการบันทึกข้อมูลลงฐานข้อมูลอาคาร

RoomID	TypeRoom	Capacity	Television	Air	Printer	Computer	Projector	เครื่องขยายเสียง	Overhead	อื่นๆ
310	ClassRoom	100	5	4	0	1	1	1	1	2
305	Lecture	75	0	0	0	0	0	1	1	0
511	Lecture	66	0	0	0	0	0	1	1	1
602	Lecture	60	0	0	0	0	0	1	0	3

เลขที่อาคาร	216	ประเภทของอาคาร	
เลขที่อาคารย่อยและเลขอาคาร	Lecture 40 6 6 1	TypeRoom	
บันทึกอุปกรณ์ภายในอาคาร		Capacity	
		Television	
		Air	
บันทึกอุปกรณ์ภายในอาคาร		Printer	
		Computer	
		Projector	
		เครื่องขยายเสียง	
		Overhead	
		อื่นๆ	

รูปที่ 4.5 ลักษณะของ โปรแกรมระบบการบันทึกข้อมูลลงฐานข้อมูลอาคาร

ในแต่ละครั้งที่สร้างฐานข้อมูลจะต้องมีการเก็บข้อมูลลงตาราง ซึ่งการบันทึกจะต้องใส่เลขห้องและข้อมูลอื่นที่ต้องการ โดยสามารถเลือกได้ว่าจะใส่ที่คอลัมน์อุปกรณ์ไหนบ้างจากรายการแสดงอุปกรณ์ทั้งหมด การใส่ค่าข้อมูลนั้นต้องใส่ให้ตรงกับอุปกรณ์ที่เลือกทั้งจำนวนและลำดับหรือถ้าหากต้องการใส่ข้อมูลทั้งหมดในหนึ่งระเบียน(ข้อมูลสำหรับหนึ่งห้อง) ก็สามารถใส่ข้อมูลทั้งหมดได้เลยโดยไม่ต้องเลือกจากรายการ แต่ให้เลือกการบันทึกที่ปุ่มบันทึกข้อมูลทั้งระเบียนได้เลย

ข้อมูลในส่วน TypeRoom หรือ ชนิดของห้อง เป็นได้ทั้งตัวอักษรหรือตัวเลข แต่ในส่วนอื่น(ความจุ กับ ข้อมูลอุปกรณ์)ควรใส่เป็นตัวเลข

การบันทึกข้อมูลลงฐานข้อมูลแต่ละครั้งได้ทำพร้อมกับการสร้างตารางเวลากิจกรรมห้องแต่ละห้องไปด้วย เป็นการสร้างความสะดวกให้กับผู้ใช้ และยังทำให้แน่ใจได้ว่ามีห้องที่ตรงกันในแต่ละฐานข้อมูล

ส่วนการแก้ไขข้อมูลในฐานข้อมูลอาคาร

RoomID	TypeRoom	Capacity	Television	Air	Printer	Computer	Projector	เตียงหยกเตียง	Overhead	โต๊ะ
665	AirLesson	40	2	2	1	0	0	0	1	1
667	AirLesson	40	2	2	1	0	0	0	1	1
669	LabComputer	50	0	3	1	40	1	0	1	1
Lobby	Lobby	0	0	5	0	1	0	0	0	2
Study	Study	0	1	1	0	0	0	0	0	2

แก้ไขข้อมูลที่ต้องการแก้ไข

RoomID:

RoomID:

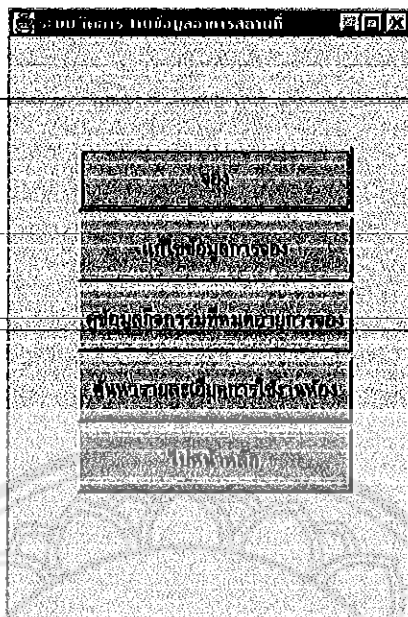
แก้ไขข้อมูลเป็น

รูปที่ 4.6 ลักษณะของโปรแกรมระบบการแก้ไขข้อมูลในฐานข้อมูลอาคาร

ข้อมูลในตารางอาคารสามารถแก้ไขได้ โดยการเลือกว่าจะกระทำกับข้อมูลที่ระบุเป็น(ห้อง) ไหน และที่คอตัมน์ใด (อุปกรณ์ชนิดใด) แล้วใส่ข้อมูลที่ต้องการแก้ไขลงในช่อง แก้ไขข้อมูลเป็น จากนั้นเลือกที่ปุ่มแก้ไข โปรแกรมก็จะทำการเปลี่ยนแปลงข้อมูลให้ตามที่กำหนด

ถ้าต้องการลบข้อมูล ก็สามารถลบได้ที่ละหนึ่งระบุ หรือก็คือการลบข้อมูลทั้งหมดในหนึ่งห้องที่ทำการเลือกไว้นั่นเอง

ระบบการจองห้อง



รูปที่ 4.7 ลักษณะของโปรแกรมจัดการฐานข้อมูลการจองห้อง

รายละเอียดการทำงานในฟอร์มการจองมีดังนี้

- สามารถเลือกจองห้องได้โดยการกดปุ่มจอง
- สามารถลบ-แก้ไขข้อมูลการจองได้โดยการกดปุ่มแก้ไขข้อมูลการจอง
- สามารถเรียกดูข้อมูลที่หมดอายุการจองได้โดยการกดปุ่มดูข้อมูลกิจกรรมที่หมดอายุการจอง
- สามารถค้นหารายละเอียดการใช้งานห้องได้โดยการกดปุ่มค้นหารายละเอียดการใช้งานห้อง

ส่วนการค้นหาห้องว่าง

ช่วงเวลา	ห้องว่าง
12:00-12:30	
12:30-13:00	
13:00-13:30	305301
13:30-14:00	305301
14:00-14:30	305301
14:30-15:00	305301
15:00-15:30	305301
15:30-16:00	305301

รูปที่ 4.8 ลักษณะของ โปรแกรมการค้นหาห้องว่าง

รายละเอียดการทำงานในฟอร์มการจองมีดังนี้

- ระบุช่วงเวลาและวันที่ต้องการทราบเลขที่ห้องที่ว่างสามารถที่จะทำการจองได้โดยอาจจะระบุหรือไม่ระบุรหัสอาคารก็ได้ จากนั้นกดปุ่มดูห้องว่าง รหัสอาคารและเลขที่ห้องก็จะแสดงขึ้นที่ตารางแสดงห้องว่าง เพื่อให้เราเลือกห้องได้ตามต้องการ
- กรอกรหัสอาคารและเลขที่ห้องที่ได้เลือกแล้วลงที่ช่องที่แสดงผลข้อมูลด้านล่าง สามารถเรียกดูตารางเวลาของกิจกรรมต่างๆที่เกิดขึ้นในห้องนั้นได้โดยเลือกกดปุ่มดูตารางเวลา สามารถเรียกดูตารางอุปกรณ์ในห้องนั้นได้โดยเลือกกดปุ่มดูตารางอุปกรณ์
- กรอกรหัสอาคาร เลขที่ห้อง วัน และ ช่วงเวลาที่ต้องการจอง โดยสามารถเลือกจองได้หลายวันในช่วงเวลาเดียวกันโดยกดปุ่ม Cntl ที่แป้นพิมพ์ขณะที่กดเลือกวันจากช่องแสดงผลวัน เมื่อเลือกข้อมูลครบแล้ว เมื่อกดปุ่มจอง ฟอร์มชื่อกิจกรรมจะแสดงขึ้นมาให้เลือกชื่อกิจกรรมที่ต้องการจองหรืออาจบันทึกชื่อกิจกรรมใหม่ลงในฐานข้อมูลก็ได้ โดย ถ้าต้องการเพิ่มชื่อกิจกรรมใหม่ ก็ทำการกดปุ่มเพิ่มชื่อกิจกรรมใหม่ หลังจากกดปุ่ม ช่องที่แสดงผลข้อมูลจะกลายเป็นช่องว่าง เพื่อที่จะคอยรับค่าข้อมูลที่ป้อนเข้าไป และปุ่มบันทึกชื่อกิจกรรมจะใช้ได้ หลังจากกรอกข้อมูลเรียบร้อยแล้ว ทำการบันทึกโดยกดปุ่มบันทึกชื่อกิจกรรม ชื่อกิจกรรมนั้นก็จะไปแสดงผลที่ช่องแสดงผลกิจกรรม

- ถ้าต้องการจอง เมื่อเลือกกิจกรรมที่ต้องการจองได้โดยการเลือกชื่อกิจกรรมจากช่องแสดงผลกิจกรรม ข้อมูลจะถูกนำมาแสดงที่ช่องแสดงผลรหัสวิชาและช่องแสดงผลชื่อกิจกรรม ถ้าต้องการจองห้องนี้โดยกิจกรรมนี้แน่นอน ก็ทำการกดปุ่มจอง ข้อมูลทั้งหมดจะถูกส่งไปหน้ารายละเอียดของการจองห้อง เพื่อบันทึกรายละเอียดการจองต่อไป

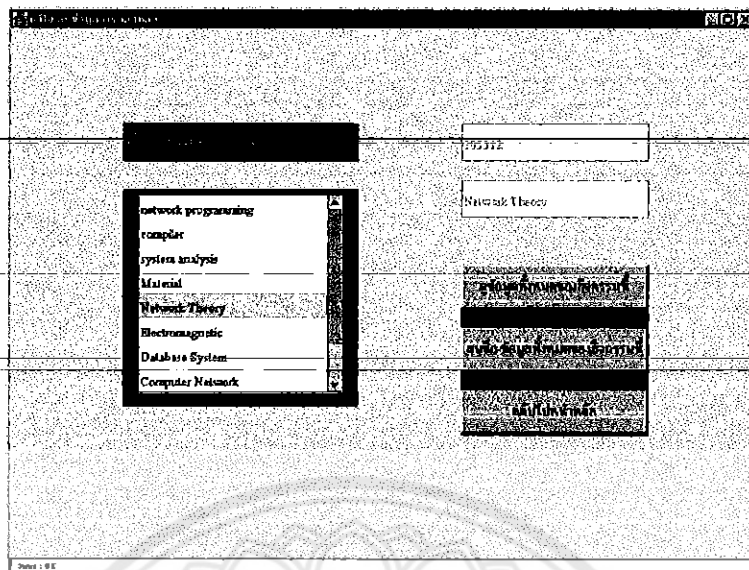
รายละเอียดการจองห้อง

รูปที่ 4.9 ลักษณะของฟอร์มการกรอกรายละเอียดการจองห้อง

การจัดเก็บข้อมูลในฟอร์มจองห้องนี้ สิ่งที่จะจัดเก็บประกอบไปด้วย รหัสอาคาร เลขที่ห้อง วัน-เวลาที่ต้องการจองห้อง วันที่เริ่มจอง วันสิ้นสุดการจอง และชื่อผู้จองห้อง รายละเอียดการทำงาน ของฟอร์มจองห้องมีดังนี้

- การเพิ่มข้อมูล จะทำการเพิ่มข้อมูลในฐานข้อมูล โดยเก็บค่าข้อมูลดังนี้ รหัสอาคาร เลขที่ห้อง วัน-เวลาที่ต้องการจองห้อง รหัสกิจกรรม ชื่อกิจกรรม วันที่เริ่มจอง วันสิ้นสุดการจอง และชื่อผู้จองห้อง ซึ่งข้อมูลรหัสอาคาร เลขที่ห้อง วัน-เวลาที่ต้องการจองห้อง รหัสกิจกรรม ชื่อกิจกรรม นี้ จะได้รับค่ามาจากฟอร์มเลือกกิจกรรมแล้ว เมื่อกรอกข้อมูลวันที่เริ่มจอง วันสิ้นสุดการจอง และชื่อผู้จองห้อง เรียบร้อยแล้ว ทำการบันทึกข้อมูลได้ โดยกดปุ่มเพิ่มข้อมูล
- ถ้าหากเลือกจองวันเป็นหลายวัน เช่น จันทร์ กับ พุธ ให้ทำการเพิ่มข้อมูลที่ละวันดังนี้ ทำการเลือกข้อมูลวันในช่องแสดงผลวันเป็นวันจันทร์ก่อน จากนั้นทำการบันทึกข้อมูลได้โดยกดปุ่มเพิ่มข้อมูล แล้วจึงเลือกข้อมูลวันเป็นวันพุธ แล้วบันทึกข้อมูลด้วยการกดปุ่มเพิ่มข้อมูลอีกครั้ง

ลบ-แก้ไขข้อมูลการจอง



รูปที่ 4.10 ลักษณะของฟอร์มการแก้ไขข้อมูลการจอง

การแก้ไข-ลบข้อมูลการจองห้อง มีวิธีการทำงานโดย เมื่อกดปุ่มแก้ไขข้อมูลการจองที่หน้าจอหลักแล้ว จะแสดงฟอร์มแก้ไข-ลบข้อมูลการจองห้อง ขึ้นมาให้เลือกชื่อกิจกรรมที่เราต้องการแก้ไข ขั้นตอนการทำงานของฟอร์มแก้ไข-ลบข้อมูลมีดังนี้

- การลบ และ แก้ไขรายละเอียดของข้อมูล เมื่อเราเลือกชื่อกิจกรรมที่เราต้องการแก้ไขแล้ว ให้กดปุ่มดูข้อมูลทั้งหมดของกิจกรรมนี้ จากนั้นฟอร์มแสดงข้อมูลก็จะแสดงผลข้อมูลของงานกิจกรรมที่เราเลือกขึ้นมาทุกกระเบียด และกำหนดให้ใช้ปุ่มแก้ไขและปุ่มลบได้ ถ้าหากต้องการแก้ไขระเบียนข้อมูล เมื่อเลือกกระเบียดที่ต้องการแก้ไขได้แล้วทำการกรอกข้อมูล และกดปุ่มแก้ไขข้อมูลถ้าหากต้องการลบระเบียนข้อมูล เมื่อเลือกกระเบียดที่ต้องการลบได้แล้วทำการลบข้อมูล โดยการกดปุ่มลบข้อมูล สามารถเลือกดูข้อมูลของทุกกระเบียดที่มีชื่อกิจกรรมเดียวกันได้โดยกด << หรือ >>

- การลบข้อมูลทั้งหมดของกิจกรรมหนึ่งรวมทั้งลบชื่อกิจกรรมออกจากฐานข้อมูลกิจกรรมด้วยนั้นสามารถทำได้โดย เมื่อเราเลือกชื่อกิจกรรมที่เราต้องการลบแล้ว ให้กดปุ่มลบข้อมูลทั้งหมดของกิจกรรมนี้ จากนั้นฟอร์มแสดงข้อมูลก็จะแสดงผลข้อมูลของงานกิจกรรมที่เราเลือกขึ้นมาทุกกระเบียด และกำหนดให้ใช้ปุ่มลบได้เพียงปุ่มเดียว แล้วทำการลบข้อมูลโดยการกดปุ่มลบข้อมูล สามารถเลือกดูข้อมูลของทุกกระเบียดที่มีชื่อกิจกรรมเดียวกันได้โดยกด << หรือ >>

ข้อมูลหมดอายุ

รูปที่ 4.11 ฟอรัมแสดงข้อมูลที่หมดอายุการจอง

ในฟอรัมแสดงผลข้อมูลหมดอายุ มีการทำงานดังนี้

- เมื่อกดปุ่มดูข้อมูลหมดอายุจากหน้าจอหลักข้อมูลที่หมดอายุคือมีค่าเวลาสิ้นสุดการจองน้อยกว่าเวลาปัจจุบัน ก็จะถูกนำมาแสดงผล
- ถ้าหากต้องการลบทะเบียนข้อมูลนี้ ให้กดปุ่มลบข้อมูล
- หรือหากต้องการแก้ไข เช่นต้องการให้จองได้นานขึ้น ก็ทำการเปลี่ยนแปลงข้อมูลวันที่สิ้นสุดการจอง แล้วกดปุ่มแก้ไขข้อมูล

ค้นหารายละเอียดการใช้งานห้อง

ID	ชื่อ	รหัสห้อง	วันที่	เวลา	เวลา	วันที่	วันที่
305352	database system	IE	201	ศุกร์	19:00:00	20:00:00	วันที่ 2001-01-01 : 2002-01-01
305352	database system	IE	201	จันทร์	12:00:00	13:30:00	วันที่ 2001-01-01 : 2002-01-01

รูปที่ 4.12 ค้นหารายละเอียดการใช้งานห้อง

ฟอร์มค้นหารายละเอียดการใช้งานห้องมีการทำงานดังนี้

- เลือกเงื่อนไขที่ต้องการค้นหาโดยกดที่ปุ่มเลือกหน้าเงื่อนไขนั้น
- เลือกค้นหา โดยรหัส/เลขที่ห้อง จะต้องกรอกข้อมูลลงในช่องข้อมูลรหัสอาคารและเลขที่ห้องอย่างน้อย 1 ตัว
- เลือกค้นหา โดยเวลาจะทำการค้นหาข้อมูลทั้งหมดที่อยู่ในช่วงเวลาที่เลือก
- เลือกค้นหา โดยวันจะทำการค้นหาข้อมูลทั้งหมดที่มีการใช้งานห้องในวันนั้นออกมาแสดงผล
- เลือกค้นหา โดยรายชื่อกิจกรรม จะทำการค้นหาข้อมูลทั้งหมดที่มีชื่อกิจกรรมตามที่ได้เลือกไว้จากช่องแสดงผลรายชื่อกิจกรรม

จากนั้นกดปุ่มค้นหา ข้อมูลที่ค้นหามาได้ตามเงื่อนไขต่างๆข้างต้นจะถูกนำมาแสดงผลที่ตารางค้นหาข้อมูลด้านล่าง

บทที่ 5

บทสรุป

5.1 สรุปผล

โปรแกรมระบบฐานข้อมูลการจัดการสารสนเทศอาคารสถานที่ (Building Information System) นี้ได้จัดทำขึ้นเพื่ออำนวยความสะดวกให้กับผู้ใช้ในการค้นหาข้อมูลและกำหนดจุดประสงค์การใช้ห้องต่างๆภายในอาคารได้โดยตัวโปรแกรมได้พัฒนาขึ้นมาจากโปรแกรมJavaซึ่งเป็นโปรแกรมที่นำมาใช้งานได้อิสระสามารถนำไปใช้งานได้กับคอมพิวเตอร์ต่างแพลตฟอร์ม (Platform) กัน

ความสามารถของโปรแกรมจัดการระบบฐานข้อมูล และฐานข้อมูลของอาคารสถานที่ที่มีดังนี้

5.1.1 เก็บรวบรวม แก้ไข ข้อมูลอุปกรณ์ของอาคาร และแสดงผลข้อมูลได้

5.1.2 เก็บรวบรวม แก้ไข ข้อมูลกิจกรรมต่างๆที่เกิดขึ้นในห้องแต่ละห้องได้

5.1.3 ค้นหาห้องที่มีอุปกรณ์ตรงกับความต้องการเพื่อนำเลขที่ห้อง ไปทำการจองห้องได้ โดย ใน 1 ห้อง ในชื่อกิจกรรมและ ช่วงเวลาเดียวกันสามารถจองได้หลายวัน เช่น

EN 202 Computer Network 10:00-12:00 จันทร์

EN 202 Computer Network 10:00-12:00 อังคาร

5.1.4 ตรวจสอบข้อมูลที่หมดอายุการจองได้

5.1.5 ค้นหาข้อมูลรายละเอียดการใช้งานห้องตามเงื่อนไขที่กำหนดได้ ประกอบด้วย

5.1.5.1 ค้นหาโดยรหัสอาคารและเลขที่ห้อง

5.1.5.2 ค้นหาโดยระบุวัน

5.1.5.3 ค้นหาโดยระบุช่วงเวลา

5.1.5.4 ค้นหาโดยใช้ชื่อกิจกรรม

5.2 ปัญหา ข้อเสนอแนะ และแนวทางแก้ไข

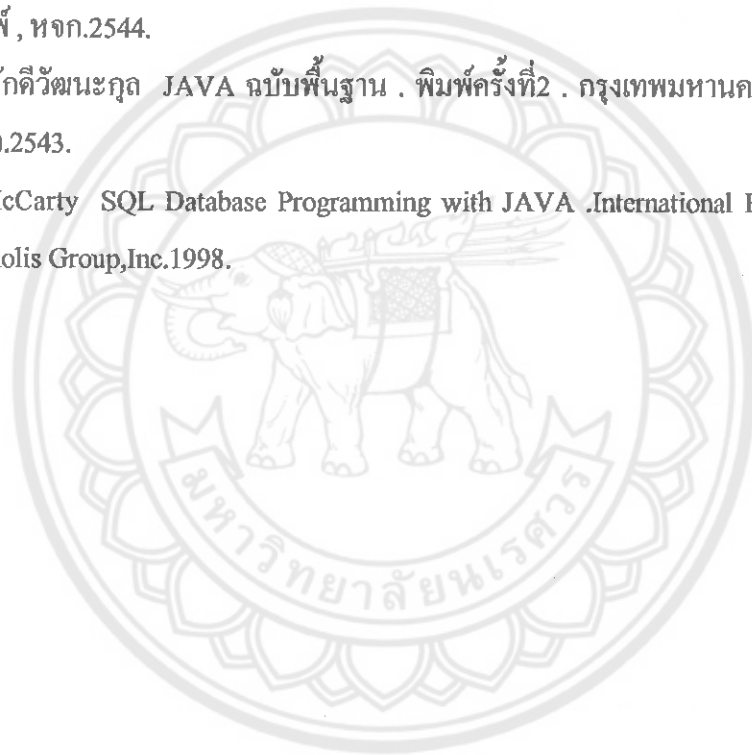
5.2.1 เนื่องจากการจัดเก็บข้อมูลการใช้งานห้องต้องการสร้างตารางของห้องทุกห้องขึ้นแล้วเก็บรหัสกิจกรรมลงภายในตารางนั้น ดังนั้นฐานข้อมูลตารางการใช้งานห้องจะประกอบไปด้วยตารางห้องต่างๆ เป็นจำนวนมาก ซึ่งต้องใช้หน่วยสำรองข้อมูลที่มีขนาดใหญ่เพียงพอสำหรับการจัดเก็บ

5.2.2 จากการศึกษาพบว่า การใช้ JAVA ติดต่อกับฐานข้อมูลโดยผ่าน JDBC-ODBC Bridge นั้น พบว่ามีความไม่เสถียรเกิดขึ้น

5.2.3 Microsoft Access มีข้อจำกัดทางด้านการใช้งานแบบ Multiuser หากต้องการใช้งาน ในแบบที่เป็น Multiuser ควรเปลี่ยนตัวจัดการฐานข้อมูล(Database Management System) เป็นชนิดอื่น

เอกสารอ้างอิง

- [1] กิตติ ภัคดีวัฒนะกุล และจำลอง ครูอุตสาหะ คัมภีร์ระบบฐานข้อมูล.พิมพ์ครั้งที่2 . กรุงเทพมหานคร :ไทยเจริญการพิมพ์ , หจก.; 2542.
- [2] อ่ำไพ พรประเสริฐกุล การวิเคราะห์และออกแบบระบบ .พิมพ์ครั้งที่ 3 . กรุงเทพมหานคร :ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ.2543.
- [3]กิตติ ภัคดีวัฒนะกุล JAVA ฉบับโปรแกรมเมอร์ . พิมพ์ครั้งที่1 . กรุงเทพมหานคร: ไทยเจริญการพิมพ์ , หจก.2544.
- [4] กิตติ ภัคดีวัฒนะกุล JAVA ฉบับพื้นฐาน . พิมพ์ครั้งที่2 . กรุงเทพมหานคร: ไทยเจริญการพิมพ์ , หจก.2543.
- [5] Bill McCarty SQL Database Programming with JAVA .International Edition . Arizona :The Coriolis Group,Inc.1998.



ภาคผนวก

ความรู้เบื้องต้นเกี่ยวกับ JDK (Java Development Kit)

ชุดพัฒนา Java

ก่อนการใช้งานภาษา Java ต้องติดตั้งชุดพัฒนา Java หรือที่เรียกว่า "Java Development Kit, JDK " ซึ่งประกอบด้วยเครื่องมือที่จำเป็นในการพัฒนาโปรแกรม ลงบนเครื่องคอมพิวเตอร์ที่ใช้อยู่ โดย

อาจหา Download ได้จาก <http://www.java.sun.com> และเพิ่ม Path ไว้ที่ไฟล์ AUTOEXEC.BAT

ปกติจะมีคำสั่ง path ระบุไว้ดังนี้

```
path = c:\windows;
```

ให้เพิ่มโดยระบุ Directory ที่โปรแกรม Java อยู่ ดังนี้

```
path = c:\windows; c:\jdk1.2\bin;
```

และเพื่อใช้ในการค้นหาโปรแกรม Java ที่เขียนขึ้นถูกเก็บอยู่ที่ใด เช่น c:\prog ให้กำหนดคำสั่งดังนี้

```
set class =;c:\j_prog
```

JDK ของภาษา Java นั้น จะติดต่อกับผู้ใช้ผ่านทาง command Line (ภาษาอื่น เช่น Visual Basic จะติดต่อแบบ Graphic User Interface (GUI) ดังนั้นการเรียกใช้ JDK จะต้องเรียกใช้จาก command Line และการสร้างไฟล์ซึ่งเป็น Source File (ไฟล์นามสกุล *.java) จะสร้างจากเอดิเตอร์ที่เป็น ASCII Text ทั่วไป เช่น MS-DOS Editor , NotePad

การใช้งาน JAVAC (Java Compiler)

javac เป็นตัวคอมไพเลอร์ภาษา Java ทำหน้าที่แปล Source File ที่สร้างขึ้น(*.java) ให้เป็นไฟล์ที่เป็น Byte Code (*.class) โดยมีรูปแบบคำสั่งคือ

```
javac [option] filename.java
```

Option ของเครื่องมือ JAVAC มีดังนี้

Option	คำอธิบายหน้าที่
Nowrite	บอก compiler ว่าให้คอมไพล์โดยไม่ต้องสร้างไฟล์ส่วนขยาย .class
Nowarn	บอก compiler ว่าไม่ต้องแสดงคำเตือน ความผิดพลาดที่ไม่ร้ายแรง(Warning)
Verbose	บอก compiler ว่าให้แสดงข้อมูลต่างๆ ขณะคอมไพล์ ข้อมูลที่แสดงประกอบด้วย class ที่เกี่ยวข้อง, เวลาที่ใช้กับการโหลด class, เวลาที่ใช้ในการคอมไพล์
d dir	บอก compiler ให้รู้ว่าใช้ Directory ตามที่กำหนดมาตามหลัง (dir แทน Directory ที่ต้องการ)
Classpath dirs	บอก compiler ให้มองเข้าไปใน Directory ที่กำหนดตามหลัง option นี้ และแสดง Directory ของ class ที่ ByteCode อ้างอิงถึงออกมา
0	บอก compiler ให้ทำ Method ของ โปรแกรมเป็น Inline
G	เพื่อจัดเตรียม Byte Code ไว้ใช้สำหรับการ Debug
Debug	บอก compiler ให้รัน โหมคการ Debug พร้อมแสดงขั้นตอนการคอมไพล์

การใช้งาน JAVA (Java Interpreter)

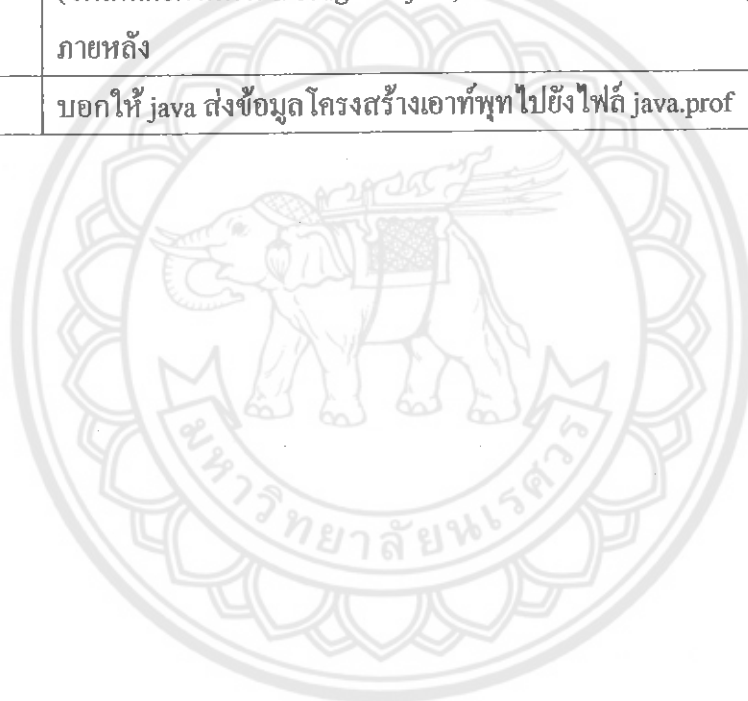
java เป็นเครื่องมือที่ใช้สำหรับแสดงผล Application ซึ่ง โปรแกรมที่จะรันต้องผ่านการคอมไพล์ด้วย javac ได้ไฟล์ *.class มาแล้ว รูปแบบคำสั่งคือ

```
java [Option] classname
```

Option ของเครื่องมือ JAVA มีดังนี้

Option	คำอธิบายหน้าที่
Help	บอกให้ java แสดงรายละเอียด Option ทั้งหมด บนจอแสดงผล
VerSion	บอกให้ java แสดงเวอร์ชันของ JDK
V	บอกให้ java แสดง class ที่ถูกโหลดขึ้นมาทั้งหมด
(-Chcksource)	กำหนดให้ java ตรวจสอบ Source Code ที่ยังไม่ได้คอมไพล์ว่า ใหม่กว่าไฟล์ class ที่มีอยู่หรือไม่ ซึ่งถ้าใหม่กว่าก็จะคอมไพล์ให้
Noasyncgc	Turn Off การทำ Garbage Collection แบบ Asynchronous
Verbosegc	บอกให้ java แสดงการเกิด Garbage

Option	คำอธิบายหน้าที่
verify	บอกให้ java ตรวจสอบ class ที่ถูกโหลดขึ้นมาทั้งหมด
noveriy	Turn Off การตรวจสอบ class ของออปชัน verify
verifyremote	บอกให้ java ตรวจสอบ class ที่ import หรือได้รับการถ่ายทอดมา
mx val	ใช้กำหนด Heap สูงสุด (ที่ val)
Ms val	ใช้กำหนด Heap เริ่มต้น
ss val	ใช้กำหนด Stack ให้ภาษา C
oss val	ใช้กำหนด Stack ให้ภาษา Java
debug	(ไฟล์ที่เคยผ่านการ Debug ด้วย java) สร้างรหัสผ่านในการใช้ java กับ ไฟล์นี้ ภายหลัง
prof	บอกให้ java ส่งข้อมูล โครงสร้างเอาท์พุท ไปยัง ไฟล์ java.prof



ประวัติผู้ทำโครงการ

ชื่อ	นางสาว นัฐกร เหมะพลานนท์
วัน เดือน ปีเกิด	29 ตุลาคม 2522
สถานที่เกิด	จังหวัดพิษณุโลก
ประวัติการศึกษา	มัธยมศึกษาตอนต้น โรงเรียนเฉลิมขวัญสตรี จังหวัด พิษณุโลก พ.ศ.2537 มัธยมศึกษาตอนปลาย โรงเรียนเฉลิมขวัญสตรี จังหวัด พิษณุโลก พ.ศ.2540

ชื่อ	นางสาว ปารีชาต กุลดู
วัน เดือน ปีเกิด	10 มกราคม 2525
สถานที่เกิด	จังหวัดพิษณุโลก
ประวัติการศึกษา	มัธยมศึกษาตอนต้น โรงเรียนเฉลิมขวัญสตรี จังหวัด พิษณุโลก พ.ศ.2538 มัธยมศึกษาตอนปลาย ศูนย์การศึกษานอกโรงเรียน จังหวัด พิษณุโลก พ.ศ.2540

