การจัดการงานด้านวัสดุครุภัณฑ์ ผ่านเครือข่ายอินเตอร์เนต สำหรับภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

ASSET MANAGEMENT VIA INTERNET FOR THE DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ 19 ส.ค. 2547
เลขทะเบียน 4700195
เลขเรียกหนังสือ
มหาวิทยาลัยนเรศวร

I4381907

ปร.
ป621ร
2546

นางสาวปิยะนาฏ เฟื่องจันทร์ รหัส 43370535

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร
ปีการศึกษา 2546

ใบรับรองโครงงานวิศวกรรม

หัวข้อโครงงาน	การจัดการงานด้านวัสดุครุภัณฑ์ที่เกี่ยวกับคอมพิวเตอร์ สำหรับภาควิชาวิศวกรรมคอมพิวเตอร์ ผ่านเครือข่ายอินเตอร์เน็ต
ผู้ดำเนินโครงงาน	นางสาว ปิยะนาฏ เฟื่องจันทร์ CPE 433705355
อาจารย์ที่ปรึกษา	อาจารย์ ดร. สุชาติ แย้มเม่น
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2546

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้โครงงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร วิศวกรรมศาสตร์บัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะกรรมการสอบโครงงานวิศวกรรม

.....ประธาน

(ดร. สุชาติ แย้มเม่น)

.....กรรมการ

(ดร. พนมขวัญ ริยะมงคล)

.....กรรมการ

(อาจารย์ พงศ์พันธ์ กิจสนาโยธิน)

หัวข้อโครงงาน	การจัดการงานด้านวัสดุและครุภัณฑ์ ผ่านเครือข่ายอินเตอร์เนต สำหรับภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์
ผู้ดำเนินโครงงาน	นางสาว ปิยะนาฏ เฟื่องจันทร์ รหัส 43370535
อาจารย์ที่ปรึกษา	ดร.สุชาติ แย้มเม่น
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2546

.....

บทคัดย่อ

โครงงานนี้เป็นการศึกษาเกี่ยวกับการจัดการงานด้านวัสดุครุภัณฑ์ ผ่านเครือข่ายอินเตอร์เน็ต เป็นโปรแกรมที่สามารถให้บริการกับฝ่ายเจ้าหน้าที่ที่รับผิดชอบงานด้านวัสดุ ครุภัณฑ์ ที่ให้ความสามารถในการยืม – คืน วัสดุ ครุภัณฑ์ ทั้งบอกสถานะของวัสดุครุภัณฑ์ด้วย พร้อมทั้งข้อมูลของบุคคลที่เกี่ยวข้องกับงานในส่วนนี้ได้ ซึ่งใช้ PHP ในการทำ และเก็บข้อมูลโดย My SQL ในการสร้างฐานข้อมูล

ผลที่ได้จากโครงงานนี้ คือ ได้โปรแกรมเกี่ยวกับการจัดการงานด้านวัสดุ ครุภัณฑ์ และสามารถใช้ได้กับภาควิชา ซึ่งสะดวกและง่ายกว่าระบบที่ใช้ในปัจจุบัน ที่ยังเก็บข้อมูลบนกระดาษ และสามารถให้ผู้ดูแลรับผิดชอบงานด้านวัสดุ ครุภัณฑ์ ทราบข้อมูลได้ตลอดเวลา

Project Title	Asset Management via Internet for the Department of Electrical and Computer Engineering
Name	Miss Piyanart Fuangjan ID. 43370535
Project Advisor	Dr.Suchat Yammen
Department	Electrical and Computer Engineering
Academic Year	2003

ABSTRACT

This project is to develop a program for managing the asset Department of the Electrical and Computer Engineering.There is borrowing or returning the asset process via internet when the data are recorded in the database.The status of the asset also is shown and checked via internet.

As a result of this project, we got the program that helps the officer manage the asset easily in stead of recording the data in the paper. Furthermore, an the officer can check the data any time that he wants via internet.

กิตติกรรมประกาศ

ผู้ทำโครงงาน ขอขอบพระคุณ ท่านอาจารย์ ดร. สุชาติ แย้มเม่น ที่ให้คำแนะนำเกี่ยวกับการทำโปรแกรม และการค้นหาข้อมูลเพื่อมาทำโครงงานนี้ ตลอดจนได้สละเวลาให้คำแนะนำทั้งภาคทฤษฎีและภาคปฏิบัติ เอาใจใส่ให้ความช่วยเหลืออย่างดียิ่งตลอดเวลาในการทำโครงงานนี้ ทั้งอาจารย์ในภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ทุกท่าน ที่คอยดูแลและให้ความสะดวกในการเล่าเรียน และทำโครงงานนี้

นางสาว ปิยะนาฏ เฟื่องจันทร์

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....ก

บทคัดย่อภาษาอังกฤษ.....ข

กิตติกรรมประกาศ.....ค

สารบัญ.....ง

สารบัญตาราง.....ช

สารบัญรูป.....ซ

บทที่ 1 บทนำ

1.1 หลักการและเหตุผล.....1

1.2 วัตถุประสงค์ของโครงงาน.....1

1.3 ขอบข่ายของโครงงาน.....2

1.4 ขั้นตอนการดำเนินงาน.....2

1.5 ผลที่คาดว่าจะได้รับ.....3

1.6 งบประมาณที่ใช้.....3

บทที่ 2 งานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีระบบเครือข่าย.....4

2.2 ทฤษฎีการสื่อสารบนอินเตอร์เนต.....17

2.3 ระบบฐานข้อมูล.....22

2.4 การทำNormalization.....36

2.5 สรุป.....43

บทที่ 3 การออกแบบโปรแกรมฐานข้อมูลการจัดการงานด้านวัสดุครุภัณฑ์

3.1 ขั้นตอนการดำเนินงาน.....44

บทที่ 4 การพัฒนาโปรแกรม

4.1 แผนภาพการทำงานของโปรแกรม.....58

4.2 รูปแบบโปรแกรมฐานข้อมูลการจัดการงานด้านวัสดุครุภัณฑ์.....64

สารบัญ(ต่อ)

บทที่ 5 บทสรุป

5.1 สรุปผล.....70

5.2 ประเมินผลและข้อเสนอแนะ.....70

5.3 ปัญหาข้อเสนอแนะและแนวทางการแก้ไข.....71

เอกสารอ้างอิง.....72

ประวัติผู้เขียนโครงงาน.....73

สารบัญตาราง

ตารางที่ หน้า

1.1 ขั้นตอนการดำเนินงาน....2

2.1 ตัวอย่าง Relation....26

2.2 ตัวอย่าง Relation....26

2.3 ตัวอย่าง Relation....27

2.4 ตัวอย่าง Relation....28

2.5 ตัวอย่าง Relation....28

2.6 ตัวอย่าง Entity....33

2.7 ตัวอย่าง Property....34

2.8 ตัวอย่าง Identity....34

2.9 ตัวอย่าง Relationship....35

2.10 ตัวอย่าง First Normal Form....37

ตัวอย่าง First Normal Form(ต่อ)....37

2.11 ตัวอย่าง Second Normal Form....39

2.12 ตัวอย่าง Third Normal Form....41

2.13 ตัวอย่าง Forth Normal Form....42

2.14 ตัวอย่าง Fourth Normal Form....43

ตัวอย่าง Fourth Normal Form (ต่อ)....43

3.1 ฐานข้อมูลโปรแกรมการจัดการงานด้านวัสดุครุภัณฑ์....54

สารบัญรูป

รูปที่		หน้าที่
2.1	รูปแบบโครงสร้าง OSI	5
2.2	ตัวอย่างการทำงานของชั้นสื่อสารในรูปแบบ OSI	11
2.3	รูปแบบโครงสร้าง TCP/IP	13
2.4	กฎการสื่อสารในรูปแบบTCP/IP	14
2.5	คลาสของเครือข่ายบนอินเทอร์เน็ต	19
2.6	ไอพีเป็นผู้ส่งข้อมูลผ่านเครือข่ายที่โยงอยู่ระหว่างทั้งสองเครื่อง	21
2.7	ชั้นของโพรโตคอล	22
2.8	การแทนกระแสข้อมูลเป็นลูกศร	30
2.9	การแทนกระแสข้อมูลเป็นลูกศรรูปวงกลม	30
2.10	การแทนนามที่อยู่นอกระบบ	30
2.11	การแทนแหล่งเก็บข้อมูล	31
2.12	การแทนสัญลักษณ์เพิ่มเติม	31
2.13	การปฏิบัติงานของบริษัท หนังสือไทย	32
3.1	แผนภาพ Context Diagram	45
3.2	แผนภาพ DFD Level 0	46
3.3	แผนภาพ DFD Level 1.0 Process1.0	47
3.4	แผนภาพ DFD Level 1 Process 2.0	48
3.5	แผนภาพ DFD Level 1 Process 3.0	49
3.6	แผนภาพ DFD Level 1 Process 4.0	50
3.7	แผนภาพ DFD Level 2 Process 4.1	51
3.8	แผนภาพ DFD Level 2 Process 4.2	52
3.9	แผนภาพ ER Diagram	53
4.1	โฟลชาร์ทระบบสต๊อกวัสดุครุภัณฑ์	58
4.2	โฟลชาร์ทโปรแกรมระบบฐานข้อมูลประวัติผู้ยืม	59
4.3	โฟลชาร์ทระบบการยืมวัสดุครุภัณฑ์	60
4.4	โฟลชาร์ทระบบการคืนวัสดุครุภัณฑ์	61
4.5	โฟลชาร์ทโปรแกรมระบบผู้ดูแลระบบ	62

สารบัญรูป (ต่อ)

รูปที่		หน้าที่
4.5	โฟลชาร์ทโปรแกรมระบบผู้ดูแลระบบ (ต่อ)	63
4.6	โฟลชาร์ทโปรแกรมแสดงระบบรายงาน	64
4.7	โปรแกรมการเข้าสู่ระบบ	64
4.8	โปรแกรมการเข้าสู่ระบบสำหรับผู้ดูแลระบบ	65
4.9	โปรแกรมการจัดการงานผู้ยืม/สมาชิก	65
4.10	โปรแกรมระบบบันทึกรายการวัสดุ ครุภัณฑ์	66
4.11	โปรแกรมการจัดการงานยืม คืน วัสดุ ครุภัณฑ์	67
4.12	โปรแกรมการจัดการระบบผู้ดูแลระบบ	68
4.13	โปรแกรมการจัดการระบบรายงาน	69

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของโครงงาน

ปัจจุบันอินเตอร์เน็ตเป็นเครือข่ายคอมพิวเตอร์ขนาดใหญ่ที่เชื่อมโยงเอาเครือข่าย คอมพิวเตอร์ย่อยๆ ทั่วโลกเข้าด้วยกันมีการแลกเปลี่ยนข่าวสารกันตลอด24 ชั่วโมง การติดต่อสื่อสารกับคอมพิวเตอร์ในอีกซีกโลกหนึ่งจะง่ายพอๆกับการคุยกับคอมพิวเตอร์ที่อยู่ในระยะทางใกล้ๆกันหรือเครื่องคอมพิวเตอร์ที่อยู่ในห้องเดียวกัน เมื่อไรก็ตามที่คอมพิวเตอร์เชื่อมต่อเข้ากับอินเตอร์เน็ตก็เปรียบเสมือนกับการติดต่อสื่อสารกันทั่วโลกซึ่งจะนำข้อมูลจากคอมพิวเตอร์เครื่องอื่นมาสู่คอมพิวเตอร์ของเราปัจจุบันนี้หลายๆองค์กรได้เชื่อมต่อระบบเครือข่ายของตนเพื่อนำองค์กรของตนเข้าสู่อินเตอร์เน็ต เพื่อรองรับการติดต่อสื่อสารการค้นคว้าหาข้อมูลต่างๆ รวมถึงความสะดวกในด้านการทำงานอื่นๆด้วย

จากการหาข้อมูลมาจากภายในสาขาคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร ทำให้ทราบว่าทางสาขาคอมพิวเตอร์ มีปัญหาเกี่ยวกับการจัดการวัสดุและครุภัณฑ์เกี่ยวกับคอมพิวเตอร์ เนื่องจากการ ยืม-คืน วัสดุและครุภัณฑ์ เพราะไม่ทราบว่าขณะนี้วัสดุและครุภัณฑ์เกี่ยวกับคอมพิวเตอร์ อยู่ที่ใครบ้าง ยืม-คืนไปในวันที่เท่าไร หมายเลขครุภัณฑ์หมายเลขอะไร จึงต้องใช้เวลานานมากในการค้นหาจากข้อมูลที่เป็นเอกสารทั้งหมด รวมทั้งเมื่อต้องการตรวจนับจำนวนวัสดุและครุภัณฑ์เกี่ยวกับคอมพิวเตอร์ แต่ละอย่างว่ามีจำนวนเท่าไรก็ต้องเสียเวลาในการตรวจนับอีกด้วย

ผู้เสนอโครงงานจึงมีแนวคิดที่จะพัฒนาระบบโดยการจัดทำเวปไซต์ (Web site) เพื่อติดตามวัสดุและครุภัณฑ์เกี่ยวกับคอมพิวเตอร์ ทางสาขาคอมพิวเตอร์ เพื่อเป็นประโยชน์ในการติดตามของเจ้าหน้าที่ผู้ที่รับผิดชอบให้สามารถแก้ไข เพิ่มข้อ ลบ Update ข้อมูล และพิมพ์ข้อมูลเป็นรายงานได้ รวมทั้งผู้ใช้บริการบุคคลอื่นๆก็ยังสามารถเข้าไปดูข้อมูลได้โดยไม่ต้องมาตรวจสอบจากเจ้าหน้าที่อีกด้วยโดยอาศัยเครือข่ายทางอินเตอร์เน็ต

1.2 วัตถุประสงค์ของโครงงาน

1. เพื่อลดความยุ่งยากของเจ้าหน้าที่ที่ทำการจัดการวัสดุและครุภัณฑ์เกี่ยวกับคอมพิวเตอร์ในการค้นหาเอกสารการยืม-คืนว่าบุคคลใดยืมไปหรือวัสดุและครุภัณฑ์เกี่ยวกับคอมพิวเตอร์ที่มีรหัสนี้อยู่ที่ไหน

2. **เพื่อความสะดวกของเจ้าหน้าที่ที่ทำการจัดการวัสดุและครุภัณฑ์เกี่ยวกับคอมพิวเตอร์ในการตรวจนับจำนวนวัสดุและครุภัณฑ์เกี่ยวกับคอมพิวเตอร์ ว่ามีของแต่ละอย่างใน จำนวนเท่าไร แก้ไข เพิ่ม ลบ Update และพิมพ์ข้อมูลเป็นรายงานได้**
3. **เพื่อให้บุคคลอื่นที่ไม่ได้ ทำการจัดการวัสดุและครุภัณฑ์เกี่ยวกับคอมพิวเตอร์เข้ามาดูข้อมูลโดยที่ไม่ต้องมาตรวจสอบจากเจ้าหน้าที่**

1.3 ขอบข่ายของโครงงาน

1. สามารถจัดเก็บข้อมูลของวัสดุและครุภัณฑ์เกี่ยวกับคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร ที่มีหมายเลข ชื่อ ยี่ห้อ และรุ่นของครุภัณฑ์ รวมทั้งหลักฐานการยืม-คืนวัสดุและครุภัณฑ์เกี่ยวกับคอมพิวเตอร์ ว่าใครยืมไป และอยู่ที่ไหน
2. สามารถตรวจนับจำนวนและครุภัณฑ์เกี่ยวกับคอมพิวเตอร์ว่ามีของแต่ละอย่างในจำนวนเท่าไร แก้ไขข้อมูล เพิ่มข้อมูล ลบข้อมูล Update และพิมพ์ข้อมูลเป็นรายงานได้
3. สามารถรายงานผลการจัดการวัสดุและครุภัณฑ์เกี่ยวกับคอมพิวเตอร์ได้โดยผ่านอินเตอร์เน็ต
4. สามารถใช้สิทธิของผู้เข้าดูและผู้ดูแลระบบ

1.4 ขั้นตอนการดำเนินงาน

กิจกรรม	ปี 2546		
	ส.ค	ก.ย	ต.ค.
1. เขียนโครงร่างการทำงานและการนำเสนอ รวมทั้งการรวบรวมข้อมูลที่เกี่ยวข้อง	←——→		
2. ศึกษาและค้นคว้าข้อมูลของการจัดการวัสดุและครุภัณฑ์เกี่ยวกับคอมพิวเตอร์และศึกษาหลักการในการเขียนโปรแกรม	←——→		
3. ทำการเขียนโปรแกรม		←——→	
4. ทำการทดสอบและแก้ไขโปรแกรม			←——→

1.5 ผลที่คาดว่าจะได้รับ

1. เจ้าหน้าที่สามารถค้นหาข้อมูลการจัดเก็บข้อมูลของวัสดุและครุภัณฑ์เกี่ยวกับคอมพิวเตอร์ รวมทั้ง การยืม-คืน การแก้ไข เพิ่ม ลบ Update และพิมพ์ข้อมูลเป็นรายงานได้ สะดวกขึ้น
2. เจ้าหน้าที่สามารถตรวจสอบจำนวนครุภัณฑ์แต่ละชนิดว่ามีเท่าไร
3. ผู้ใช้ระบบไม่จำเป็นต้องเดินทางมาสอบถามกับเจ้าหน้าที่โดยตรง และสามารถตรวจสอบข้อมูลและรายละเอียดได้สะดวกขึ้น

1.6 งบประมาณที่ใช้ นิสิต: คน: 1000 บาท

1.	ซื้อหนังสือ	700	บาท
2.	ค่าถ่ายเอกสาร	300	บาท
	รวม	1000	บาท

บทที่ 2
งานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงหลักการ และทฤษฎีที่เกี่ยวกับการสร้างโปรแกรมจัดการวัสดุและครุภัณฑ์ผลการศึกษาผ่านทางเครือข่ายอินเทอร์เน็ต โดยประกอบไปด้วยทฤษฎีที่เกี่ยวกับระบบเครือข่ายการสื่อสารบนอินเทอร์เน็ต การออกแบบฐานข้อมูล การเขียนโปรแกรมติดต่อกับฐานข้อมูลบนอินเทอร์เน็ต

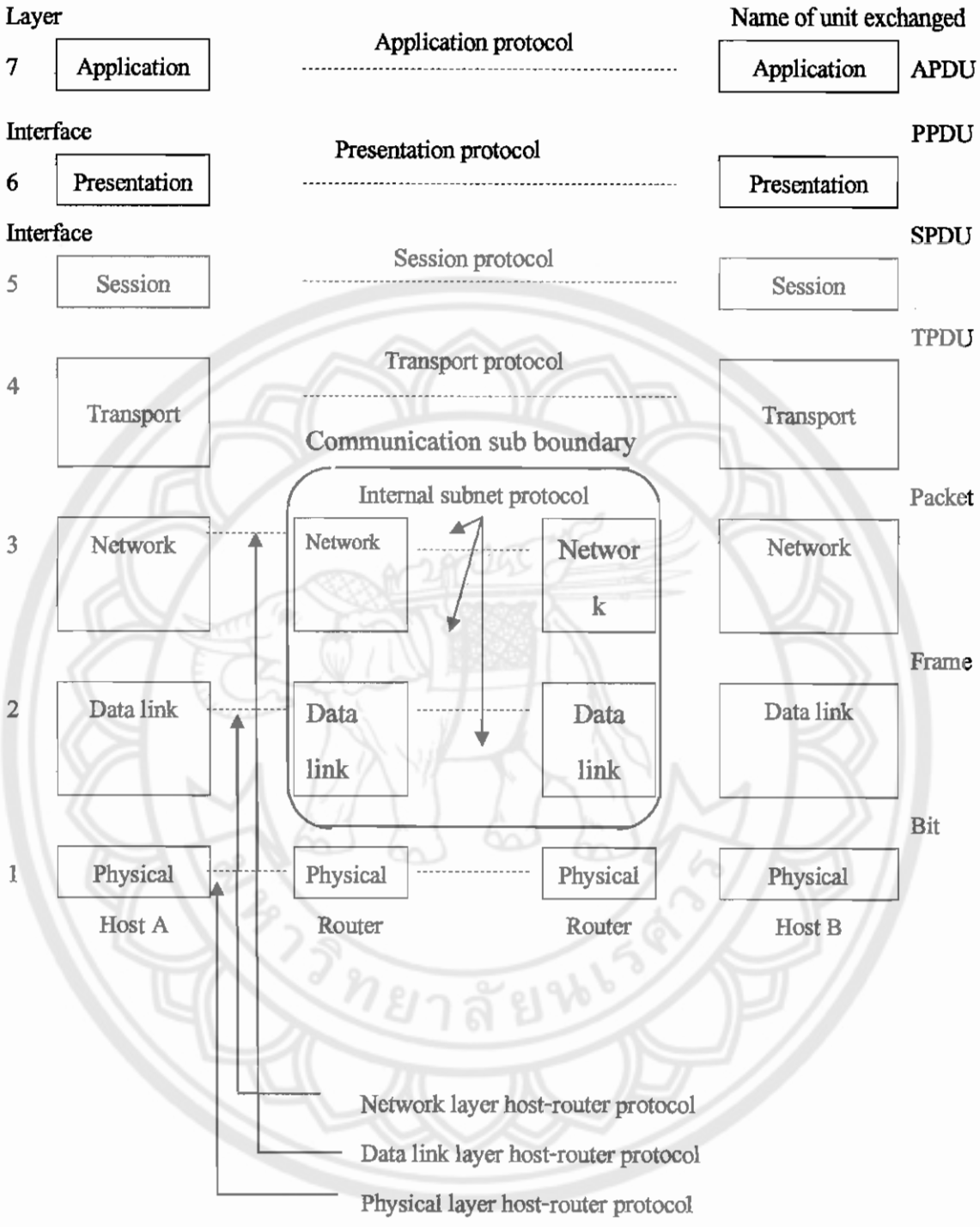
2.1 ทฤษฎีระบบเครือข่าย

2.1.1 รูปแบบเครือข่ายมาตรฐานสากล

องค์การ International Standards Organization (ISO) ได้กำหนดรูปแบบโครงสร้างมาตรฐานสากลสำหรับ การติดต่อสื่อสารระหว่างเครื่องคอมพิวเตอร์ไว้เรียกว่าOpen System Interconnection (OSI) ซึ่งมีอยู่ทั้งหมด 7 ชั้นสื่อสาร ตัวโครงสร้างเองได้เน้นความสำคัญของรูปแบบการติดต่อสื่อสาร ระหว่างระบบเปิด (Open systems) กับระบบปิด จึงสามารถนำไปใช้อ้างอิงได้ในระดับสากลอย่างแท้จริง

แนวความคิดของการกำหนดมาตรฐานเป็นแบบชั้นสื่อสาร (Layers)

1. ชั้นสื่อสารแต่ละชั้นถูกกำหนดขึ้นมาตามบทบาทที่แตกต่างกัน
2. แต่ละชั้นสื่อสารต้องทำหน้าที่ตามที่ได้รับมอบหมายอย่างดียิ่ง
3. แต่ละฟังก์ชันในชั้นสื่อสารใดๆ จะต้องกำหนดขึ้นมาโดยใช้แนวความคิดในระดับสากลเป็นวัตถุประสงค์หลัก
4. ขอบเขตความรับผิดชอบของแต่ละชั้นสื่อสาร จะต้องกำหนดขึ้นมาเพื่อจำกัดปริมาณการแลกเปลี่ยนข้อมูลและผลกระทบข้างเคียงระหว่างการติดต่อให้มีน้อยที่สุด
5. จำนวนของชั้นสื่อสารจะต้องมีมากพอที่จะแยกฟังก์ชันการทำงานที่แตกต่างกันให้อยู่คนละชั้น แต่จะต้องไม่มีมากเกินความจำเป็น



รูปที่ 2.1 รูปแบบโครงสร้าง OSI

(ที่มา : Computer Network, Andrew S. Tanenbaum)

2.1.1.1 ชั้นสื่อสารกายภาพ (The Physical Layer)

0 จะแทนด้วยกระแสไฟฟ้ากี่โวลต์ และบิต 1 จะต้องใช้กี่โวลต์, แต่ละบิตจะใช้ระยะเวลาในการส่งนานเท่าไร, การส่งเป็นแบบทางเดียวหรือสองทาง, จะเริ่มติดต่ออย่างไร, การติดต่อจะสิ้นสุดอย่างไร , และสายเคเบิ้ลมีกี่เส้นแต่ละเส้นใช้เพื่ออะไร เป็นต้น จะเห็นได้ว่ากฎระเบียบ สำหรับขั้นนี้จะเกี่ยวพันโดยตรงกับการรักษาของอุปกรณ์สัญญาณไฟฟ้า (หรือสัญญาณใดๆ) ขั้นตอนในการใช้อุปกรณ์เหล่านั้น และความสัมพันธ์กับสื่อที่ใช้รับ-ส่งสัญญาณ

2.1.1.2 ชั้นสื่อสารเชื่อมต่อข้อมูล (Data Link Layer)

หน้าที่หลักของชั้นเชื่อมต่อข้อมูลคือ ทำการรวบรวมข้อมูลต่อจากชั้นกายภาพ ตรวจสอบความถูกต้องของข้อมูล แล้วส่งข้อมูลที่ปราศจากข้อผิดพลาดนี้ให้กับชั้นควบคุมเครือข่ายต่อไป โดยปกติผู้ส่งข้อมูลจะแบ่งข้อมูลที่มีความยาวมากออกเป็นกลุ่มข้อมูลย่อยๆ แต่ละส่วนย่อยเรียกว่า ดาต้าเฟรม (Data frame) ซึ่งจะมีขนาดคงที่ประมาณสองหรือสามร้อยไบต์ หรืออย่างมากก็ไม่เกินสองถึงสามพันไบต์ชุดของดาต้าเฟรมสำหรับข้อมูลที่ต้องการส่งไปให้ผู้รับก็จะถูกส่งไปทีละเฟรมตั้งแต่เฟรมแรกไปจนครบทุกเฟรม ข้างฝ่ายผู้รับจะตอบสนองโดยการส่งดาต้าเฟรมพิเศษเรียกว่า เฟรมตอบรับ (Acknowledgement frame) ไปถึงผู้ส่งเพื่อเป็นการบอกให้ทราบว่าได้รับข้อมูลครบแล้ว กระบวนการรับ-ส่งข้อมูล ข้อมูลชุดนี้ก็จะเสร็จสิ้นสมบูรณ์

การรับ-ส่งข้อมูลในชั้นกายภาพนั้นจะไม่รับรู้ในเรื่องโครงสร้างข้อมูล คือจะมองเห็นข้อมูลว่าเป็นบิด 0 หรือบิต 1 กลุ่มหรือชุดหนึ่งที่เรียงตามลำดับ เรียกว่า กระแสบิต (Bit stream) จึงเป็นหน้าที่ของโปรแกรมในชั้นเชื่อมต่อข้อมูลจะต้องทำการตรวจสอบความถูกต้อง ซึ่งทำได้โดยการเพิ่มข้อมูลสำหรับการตรวจสอบติดไว้กับข้อมูลทุกเฟรม เช่น การเพิ่มข้อมูลส่วนหัวและส่วนหาง (Header and tailer) เข้าไปกับทุกเฟรมซึ่งจะใช้เป็นตัวกำหนดขอบเขตของดาต้าเฟรมด้วย

การส่งข้อมูลผ่านระบบใดๆ ก็ตาม ข้อมูลที่ส่งนั้นมีโอกาสจะเสียหายไปเลยก็ได้ โปรแกรมในชั้นเชื่อมต่อข้อมูลจะต้องสามารถตรวจสอบความผิดปกตินี้แล้วก็จะต้องมีวิธีการแก้ไข เช่น แจ้งให้ผู้ส่งข้อมูลได้ส่งข้อมูลชุดเดิมกลับมาใหม่ (เรียกเฟรมนี้ว่า Duplicate frame) อย่างไรก็ตามการส่งข้อมูลซ้ำทำให้เกิดปัญหาตามมาในกรณีที่ชุดข้อมูลไม่ได้สูญหายไปไหนเพียงแต่ใช้เวลาเดินทางมากกว่าปกติ ดังนั้นข้อมูลชุดเดียวกันก็จะมาถึงผู้ใช้ทั้งสองเฟรม โปรแกรมในชั้นนี้จะต้องหาวิธีตรวจสอบและต้องกำจัดเฟรมที่ซ้ำออกไป

2.1.1.3 ชั้นสื่อสารควบคุมเครือข่าย (Network Layer)

ชั้นควบคุมเครือข่ายมีหน้าที่รับผิดชอบในการควบคุมการติดต่อรับ-ส่งข้อมูลระหว่างเครื่องคอมพิวเตอร์ (เรียกว่าโหนด Node) ต่างๆ ในระบบเครือข่ายให้เป็นไปด้วยความเรียบร้อย สิ่งที่สำคัญที่สุดคือการกำหนดเส้นทางเดินของข้อมูลจากโหนดผู้ส่งข้อมูลไปตามโหนดต่างๆ จนถึงโหนดผู้รับข้อมูลในที่สุดโฮสต์บางกลุ่มจะกำหนดเส้นทางการเดินข้อมูลโดยศึกษาระบบเครือข่าย

แล้วสร้างตารางเส้นทางเดินข้อมูลแบบถาวร โฮสต์บางกลุ่มจะกำหนดเส้นทางการเดินข้อมูลในตอนเริ่มต้นของการสื่อสารดังนั้นการสื่อสารในครั้งต่อไป(ติดต่อกับโหนดเดิม) อาจจะเปลี่ยนไปใช้เส้นทางอื่นได้ โฮสต์ในกลุ่มที่มีวิธีการซับซ้อนมากจะกำหนดเส้นทางเดินข้อมูลในระดับแพ็กเก็ต กรณีที่มีผู้ส่งข้อมูลพร้อมๆ กันหลายจุดจะทำให้เกิดความคับคั่งของข้อมูลคล้ายกับสภาวะการจราจรในชั่วโมงเร่งด่วนซึ่งมีปริมาณรถยนต์มากทำให้การจราจรติดขัด โฮสต์ในกลุ่มนี้ก็จะปรับเส้นทางการเดินข้อมูลของแต่ละแพ็กเก็ตให้เหมาะสมกับสภาวะของระบบเครือข่ายตลอดเวลา

การส่งผ่านข้อมูลในระบบเครือข่ายอาจมีการบันทึก ผู้ส่ง ผู้รับ และปริมาณข้อมูลที่ไหลผ่านโฮสต์หรือเราเตอร์ต่างๆ เพื่อประโยชน์ทางด้านการคิดค่าบริการ ซึ่งจะมีความซับซ้อนมากขึ้นถ้าข้อมูลไหลผ่านระบบเครือข่ายที่มีการคิดอัตราค่าบริการที่แตกต่างกัน

เมื่อแพ็กเก็ตเดินทางผ่านเครือข่ายย่อยระบบหนึ่งไปยังอีกระบบหนึ่งอาจเกิดปัญหาความแตกต่างระหว่างกันในด้านต่างๆ อาทิเช่น การใช้กฎการสื่อสารข้อมูลไม่เหมือนกัน หรือการใช้วิธีการกำหนดตำบลที่อยู่ไม่เหมือนกันปัญหาที่กล่าวถึงนี้เป็นความรับผิดชอบของโปรแกรมในชั้นควบคุมเครือข่ายที่จะต้องหาทางแก้ไขหรือปรับความแตกต่างระหว่างเครือข่ายต่างๆ ให้สามารถเข้าใจกันได้ ท้ายที่สุด การส่งข้อมูลแบบกระจ่ายข่าย(Broadcasting) ที่มีใช้ในบางระบบนั้นจะไม่มีปัญหาที่กล่าวถึงอยู่เลย ดังนั้นโปรแกรมในชั้นนี้จึงมีหน้าที่การทำงานน้อยมากหรืออาจไม่มีเลยก็ได้

2.1.1.4 ชั้นจัดการนำส่งข้อมูล (Transport Layer)

โปรแกรมในชั้นนำส่งข้อมูลมีหน้าที่หลักในการรับข้อมูลมาจากชั้นควบคุมหน้าต่างสื่อสาร ซึ่งอาจต้องแบ่งข้อมูลออกเป็นแพ็กเก็ตขนาดย่อม (ในกรณีที่ข้อมูลมีปริมาณมาก) หลายๆ แพ็กเก็ตแล้วจึงส่งข้อมูลทั้งชุดต่อไปให้โปรแกรมในชั้นควบคุมเครือข่าย ทางด้านโปรแกรมชั้นนำส่งข้อมูลของผู้รับก็จะทำหน้าที่ประกอบแพ็กเก็ตชุดนี้ให้กลับมารวมเป็นข้อมูลเดิม

ในภาวะปกติ การเชื่อมต่อการสื่อสารจะเป็นการจัดตั้งหน้าต่างสื่อสาร (Session) ระหว่างผู้ส่งและผู้รับตามที่เกิดขึ้น ถ้าต้องการเพิ่มประสิทธิภาพก็อาจสร้างโพรเซสของโปรแกรมนำส่งข้อมูลขึ้นมาหลายๆ โพรเซสเพื่อช่วยกันจัดส่งข้อมูลให้เร็วขึ้นแต่ถ้าเน้นในด้านความประหยัดก็อาจทำในทางตรงกันข้ามนั้นคือการยุบรวมโพรเซสให้เหลือจำนวนน้อยลงแล้วจึงจัดการให้โพรเซสที่เหลืออยู่ทำการส่งข้อมูลทั้งหมดโดยการใช้ช่องสื่อสารร่วมกัน

โปรแกรมในชั้นนี้เป็นผู้กำหนดประเภทของการให้บริการต่างๆ รวมไปถึงการอำนวยความสะดวกในการใช้ระบบเครือข่ายซึ่งแบ่งออกได้เป็น 3 ประเภท ประเภทแรกเป็นการให้บริการแบบจุด-ต่อ-จุด โดยเน้นการรับประกันความถูกต้องของข้อมูลเป็นสำคัญ ประเภทที่สองเน้นการให้บริการข้อมูล ข้อมูลในระดับแพ็กเก็ตซึ่งแม้ว่าจะไม่รับประกันการสูญหายของข้อมูลแต่ก็ให้ความคล่องตัวสูงกว่าแบบแรก(การรับประกันความถูกต้องของข้อมูลสามารถทำในชั้นอื่นได้)

ประเภทที่สามเป็นการส่งข้อมูลแบบกระจ่ายข่าวเพื่อประโยชน์ในการส่งข้อมูลชุดเดียวกันไปยังผู้ใช้หลายจุดพร้อมกัน

โปรแกรมในชั้นนำส่งข้อมูลติดต่อถึงกันผ่านช่องสัญญาเสมือน (Virtual channel) ระหว่างผู้ส่งและผู้รับโดยตรงเรียกว่าเป็นการติดต่อแบบ End- to -End Connection ในขณะที่โปรแกรมในสามชั้นแรกนั้นเป็นการติดต่อแบบ จุด- ต่อ- จุด ซึ่งผู้รับอาจไม่ใช่ผู้รับข้อมูลที่แท้จริง แต่เป็นเพียงโหนดตัวกลางในการรับแล้วส่งข้อมูลต่อไปตามเส้นทางเดินข้อมูลที่ถูกกำหนดไว้เท่านั้น

เครื่องโฮสต์ส่วนมากจะใช้ระบบปฏิบัติการที่ให้บริการแบบมัลติโปรแกรมมิ่ง (Multiprogramming) คือสามารถสร้างและใช้งานโพรเซสในชั้นการนำส่งข้อมูลได้หลายโพรเซสในขณะเดียวกัน จึงมีความจำเป็นที่จะต้องเพิ่มข้อมูลส่วนหัว (คือสาย H4 ในรูป 2.2) เข้าไปกับข้อมูลแต่ละแพ็กเก็ตเพื่อบอกให้ระบบปฏิบัติการโฮสต์ทราบว่าแพ็กเก็ตที่รับมานั้นเป็นของโพรเซสใด

นอกจากการใช้ช่องสื่อสารร่วมกันแล้วโปรแกรมในชั้นนำส่งข้อมูลจะต้องมีความสามารถในการจัดตั้งหน้าต่างสื่อสารกับโหนดอื่น ๆ ในระบบเครือข่ายและจัดการยกเลิกเมื่อการสื่อสารสิ้นสุดลง โปรมแกรมในชั้นยังต้องมีวิธีการกำหนดการตั้งชื่อให้แก่ตนเอง และแนะนำให้ผู้อื่นในระบบ ได้รู้จัก รวมทั้งเรียนรู้การกำหนดตำบลที่อยู่ของโหนดอื่นได้ อีกสิ่งหนึ่งที่จำเป็นมากคือการควบคุมการไหลของข้อมูล (Flow control) ซึ่งมีทั้งในระดับโฮสต์และระดับเราเตอร์โดยมีวัตถุประสงค์ในการควบคุมการรับ และส่งข้อมูลโดยเฉพาะในกรณีที่ผู้ส่งจัดการส่งข้อมูลเร็วเกินกว่าผู้ใช้จะทำงานได้ทัน

2.1.1.5 ชั้นสื่อสารควบคุมหน้าต่างสื่อสาร (Session Layer)

ชั้นควบคุมหน้าต่างสื่อสารเป็นผู้กำหนดวิธีการควบคุมการเชื่อมต่อระหว่างผู้รับข้อมูล และผู้ส่งข้อมูลตั้งแต่เริ่มต้นการสื่อสารไปจนยุติการสื่อสาร เช่น การติดต่อขอใช้โฮสต์จากเครื่องคอมพิวเตอร์ที่อยู่ไกลออกไป (Remote login) หรือการส่งแฟ้มข้อมูลระหว่างเครื่องคอมพิวเตอร์ผ่านระบบเครือข่ายโดยภาพรวมแล้วการให้บริการในชั้นนี้จะคล้ายกับบริการที่มีให้ในชั้นนำส่งข้อมูลแต่ในชั้นนี้จะให้บริการหลายอย่างที่เป็นประโยชน์มากกว่าสำหรับการประยุกต์ใช้งานบางประเภท

หน้าที่สำคัญอย่างหนึ่งคือบริหารการแลกเปลี่ยนข่าวสาร (Dialogue control) อันได้แก่การกำหนดให้การแลกเปลี่ยนข่าวสารเป็นไปแบบสองทางในเวลาเดียวกัน (Full duplex) หรือถ้าเป็นการสื่อสารแบบทางเดียวแต่สลับทิศได้ (Half duplex) ก็จะต้องเป็นผู้จัดลำดับให้ทั้งผู้ใช้และผู้ส่งทำการส่งข้อมูลได้คล้ายกับการควบคุมการสับหลีกรถไฟ

สำหรับการสื่อสารประเภทที่ต้องใช้ โทเก้น (Token) โปรแกรมในชั้นนี้จะเป็นผู้บริหารการใช้โทเก้นเพื่อให้โหนดต่างๆ ในระบบนั้นผลัดเปลี่ยนการครอบครองโทเก้นอย่างเป็นธรรม หรือถูกต้องตามลำดับความสำคัญ (Priority)

หน้าที่อีกประการหนึ่งได้แก่การแก้ปัญหาความล้มเหลวในการส่งข้อมูลขนาดใหญ่มากระหว่างโหนดต่างๆ ในกรณีที่ส่งข้อมูลเกิดล้มเหลวกลางคันโดยไม่มีการแก้ไขใดๆ โหนดทั้งสองก็จะต้องเริ่มต้นใหม่ถ้าเกิดการล้มเหลวขึ้นอีกก็จะต้องเริ่มต้นใหม่อีก วิธีการแก้ไขวิธีหนึ่งคือการแทรกจุดตรวจสอบความถูกต้อง (Checkpoints) เข้าไปจำนวนหนึ่ง (ขึ้นอยู่กับปริมาณข้อมูล) ในระหว่างการส่งข้อมูล จุดตรวจสอบทั้งหมดจะต้องถูกแทรกเข้าไปในข้อมูลที่ตำแหน่งเดียวกันของทั้งผู้ส่ง และผู้รับซึ่งเรียกว่าการ synchronization หากเกิดการล้มเหลวขึ้นโปรแกรมในชั้นนี้ของผู้รับก็จะค้นหาจุดตรวจสอบจุดสุดท้ายก่อนการล้มเหลวเพื่อลบข้อมูลส่วนที่อยู่หลังจุดตรวจสอบนั้นทิ้งไปแล้วแจ้งให้ผู้ส่งเริ่มต้นการส่งข้อมูลใหม่จากจุดตรวจสอบนั้นแทนที่จะต้องเริ่มต้นใหม่ทั้งหมด

2.1.1.6 ชั้นสื่อสารนำเสนอข้อมูล (Presentation layer)

โปรแกรมที่ทำงานในระดับชั้นควบคุมต้นๆ ที่กล่าวมานั้นจะให้ความสนใจในประสิทธิภาพของการรับ-ส่งข้อมูล และมองเห็นว่าข้อมูลคือกระแสบิต (Bit stream) หรือกระแสไบต์ (Byte stream) เท่านั้นโปรแกรมในชั้นนำเสนอข้อมูลจะมองข้อมูลว่าเป็นสิ่งที่มีรูปแบบ (Syntax) และความหมาย (Semantics) มากกว่ากระแสของบิตหรือไบต์ เช่น ข้อมูลที่เป็นตัวเลขบอกจำนวนเงินข้อมูลที่เป็นชื่อ และข้อมูลที่เป็นในเรียกเก็บเงิน เป็นต้น ความแตกต่างของการให้ความหมายข้อมูลของเครื่องคอมพิวเตอร์ในระบบต่าง ๆ เป็นปัญหาที่จะต้องได้รับการแก้ไขในระดับส่วนรวมไม่ใช่ให้แต่ละฝ่ายแก้ปัญหาโดยลำพัง การควบคุมรูปแบบ และความหมายของข้อมูล การใช้รหัสแทนข้อมูล เช่น ASCII หรือ Unicode หรือการแทนข้อมูลด้วยระบบ Little endian หรือ Big endian รวมถึงการเข้ารหัส และถอดรหัสข้อมูลสิ่งต่าง ๆ ที่กล่าวมานี้ล้วนแต่เป็นความรับผิดชอบของโปรแกรมในชั้นนี้

2.1.1.7 ชั้นสื่อสารการประยุกต์ (Application Layer)

ในปัจจุบัน มีจอภาพเทอร์มินัล(Terminals) อยู่หลายร้อยชนิดทั่วโลกซึ่งส่วนใหญ่จะไม่สามารถใช้ทดแทนหรือใช้งานร่วมกันได้ การติดต่อระหว่างเครื่องคอมพิวเตอร์ที่อยู่คนละระบบเครือข่ายจึงไม่อาจสื่อสารกันได้โดยสมบูรณ์ โปรแกรมในชั้นการประยุกต์จึงเข้ามีมีบทบาทสำคัญสองด้านคือ การเป็นตัวกลาง หรือส่วนติดต่อระหว่างโปรแกรมประยุกต์ (application Programs) กับโปรแกรมใน 6 ชั้นที่เหลือ และการกำหนดแบบมาตรฐานของจอ (Terminal type)

การกำหนดแบบมาตรฐานของจอนั้นไม่ได้เป็นการกำหนดวิธีสร้างจอเทอร์มินัลให้เหมือนกันแต่จะคล้ายกับการสร้างจอเทอร์มินัลเสมือน (Virtual terminal) ขึ้นบนจอเทอร์มินัลจริง ทั้งนี้เพื่อทำให้จอเทอร์นิมัลทุกชนิดในโลกมีความเข้าใจตรงกันเช่น ขนาดบริเวณที่ในการแสดงผล การเคลื่อนย้ายตำแหน่งเคอร์เซอร์ (Cursor) และการแสดงตัวอักษร ณ ตำแหน่งต่างๆ บนจอภาพ เป็นต้น จึงทำให้การใช้จอเทอร์มินัลเพื่อการสื่อสารบนระบบเครือข่ายเกิดขึ้นได้แม้ว่าจะใช้จอเทอร์มินัลต่างแบบกันก็ตาม

การสำเนาแฟ้มข้อมูลหรือการคัดลอก (Copy) แฟ้มข้อมูลผ่านระบบเครือข่ายก็อาจเกิดปัญหาได้ตัวอย่าง เช่น ในระบบปฏิบัติการหลายระบบมีวิธีการกำหนดชื่อแฟ้มข้อมูลที่แตกต่างไปจากวิธีที่ใช้ในระบบอื่น จึงจำเป็นจะต้องมีการแก้ไขหรือปรับแต่งส่วนที่ต่างกันให้สามารถเข้ากันได้ ถ้าหากปล่อยให้เป็นหน้าที่ของผู้ใช้โดยตรงแล้วก็จะทำให้เกิดความยุ่งยากมาก ทั้งนี้เพราะผู้ใช้บางส่วนอาจไม่มีความรู้มากพอที่จะแก้ไขได้ นอกจากนี้การส่งจดหมายอิเล็กทรอนิกส์ การใช้เทอร์มินัลสำหรับป้อนข้อมูลจากระยะไกล หรือการดูบัญชีรายชื่อแฟ้มข้อมูล (ในเครื่องคอมพิวเตอร์อื่นล้วนเป็นหน้าที่ของโปรมแกรมชั้นการประยุกต์ที่จะต้องอำนวยความสะดวกให้แก่ผู้ใช้ทั้งสิ้น)

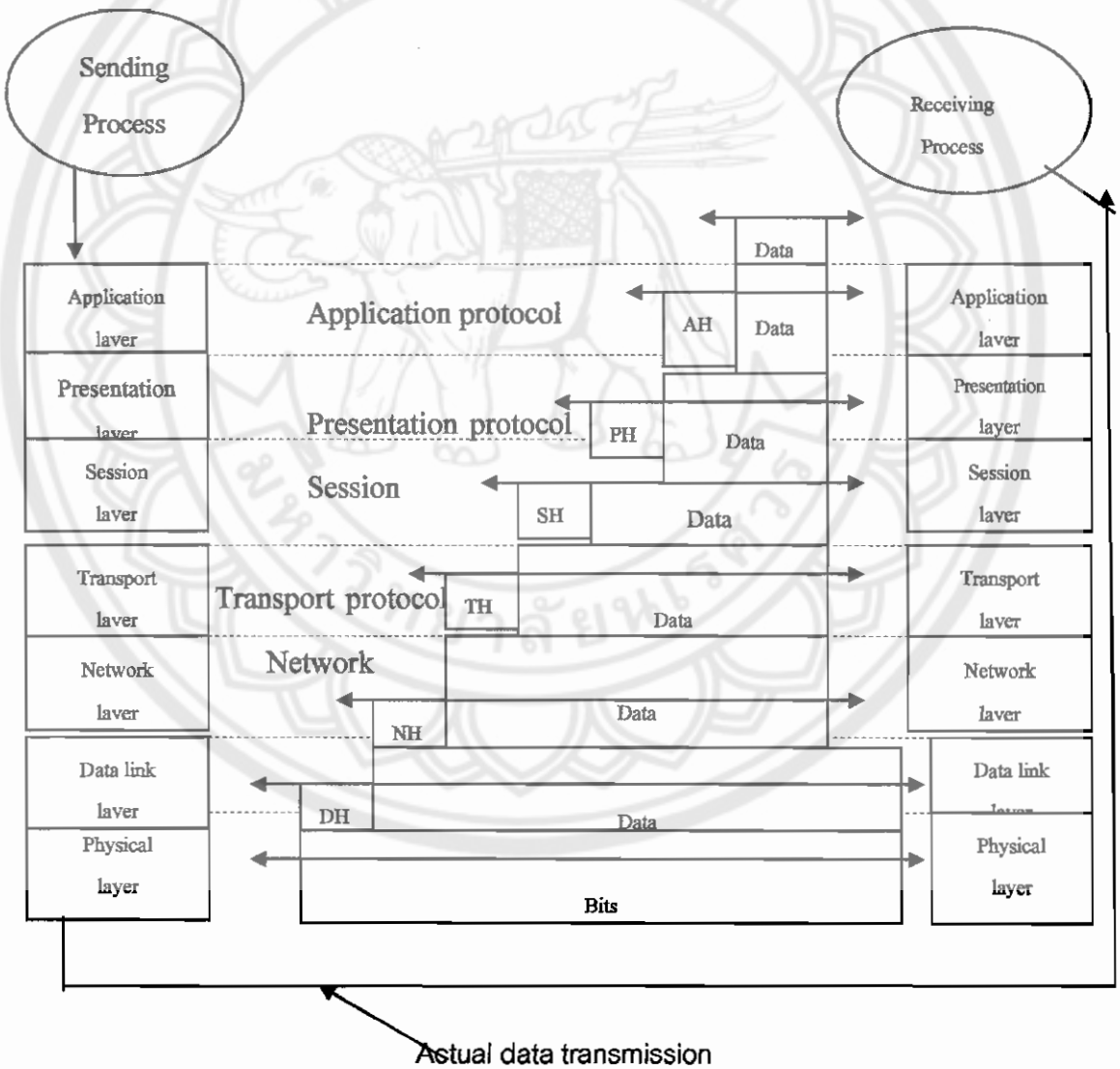
2.1.1.8 การส่งข้อมูลในรูปแบบมาตรฐานของ OSI

รูป 2.2 แสดงให้เห็นขั้นตอนที่ข้อมูลได้ถูกส่งจากผู่ส่งข้อมูลไปให้ผู้รับข้อมูล กระบวนการส่งข้อมูลเริ่มต้นจากโพรเซสที่ต้องการส่งข้อมูลส่งมอบให้กับโปรแกรมชั้นการประยุกต์ซึ่งจะเติมข้อมูลส่วนหัว (คือส่วน AH ในรูป 2.2) ที่ต้องการเข้าไปกับข้อมูลจริงส่งต่อไปให้โปรแกรมชั้นนำเสนอข้อมูล โปรแกรมในชั้นนั้นจะเข้าใจว่าข้อมูลที่ส่งมาทั้งหมดนั้นคือข้อมูลจริง (จะไม่ทราบว่า มีข้อมูลส่วน AH แทรกเข้ามาด้วยแล้ว) จึงเพิ่มเติมข้อมูลในส่วนที่ตนเองต้องการลงไปกับข้อมูลทั้งหมด (คือส่วน PH ในรูป 2.2) แล้วส่งต่อให้กับโปรแกรมในชั้นกำหนดหน้าต่างสื่อสาร กระบวนการนี้จะทำซ้ำไปเรื่อยๆ เมื่อผ่าน โปรแกรมในแต่ละชั้นสื่อสารจนถึงชั้นกายภาพซึ่งจะมองเห็นว่าข้อมูลที่ส่งมาจากชั้นเชื่อมต่อข้อมูลนั้นเป็นเพียงกระแสบิต (Bit stream) ที่จะต้องส่งออกไปเท่านั้น

ทางด้านผู้ใช้ข้อมูลจะเริ่มกระบวนการรับข้อมูลโดยชั้นกายภาพรับข้อมูลเข้ามาในลักษณะของกระแสบิตแล้วส่งต่อไปให้โปรแกรมขั้นเชื่อมต่อข้อมูล โปรแกรมนี้จะอาศัยข้อมูลส่วนหัว (DH) เพื่อทำความเข้าใจกับลักษณะของข้อมูลที่รับมารวมทั้งการตรวจสอบความถูกต้อง เสร็จแล้วจึงลบข้อมูลส่วนนี้ออกไป และส่งข้อมูลที่เหลือไปให้กับโปรแกรมในชั้นควบคุมเครือข่าย กระบวนการที่จะเป็นไปอย่างนี้จนในที่สุดข้อมูลก็ถูกส่งมาถึงชั้นการประยุกต์ในชั้นนี้ข้อมูลส่วนหัวที่โปรแกรมในแต่ละชั้น (ของผู้ส่ง) เพิ่มเติมเข้าไปนั้นได้ถูกโปรแกรมในชั้นนั้นๆ (ของผู้ใช้) ดึงออกไปใช้จนเหลือแต่ข้อมูลส่วนหัวของขั้นการประยุกต์(AH)เท่านั้น โปรแกรมในชั้นนี้ก็จะดึง

ออกไปใช้จนเหลือแต่ข้อมูลส่วนหัวของ(AH)เท่านั้น โปรแกรมในชั้นนี้ก็จะดึงข้อมูลส่วนสุดท้ายนี้ออกไปใช้งานแล้วจึงส่งข้อมูลที่เหลืออยู่ (คือข้อมูลจริง) ให้กับโพรเซสของผู้รับนำข้อมูลไปใช้งาน

แนวความคิดที่เป็นหลักสำคัญของกระบวนการนี้คือ การที่โปรแกรมในแต่ละชั้นคิดว่าการสื่อสารเป็นไปตามแนวนอน แม้ว่าในความเป็นจริงนั้นจะเป็นไปตราแนวตั้งดังที่แสดงในรูป 2.2 ทั้งนี้หมายความว่าโปรแกรมในแต่ละชั้นของทางด้านผู้ส่งจะแทรกข้อมูลที่ตนเองต้องการเข้าไปกับข้อมูลโดยมีวัตถุประสงค์ให้โปรแกรมในชั้นเดียวกันของทางผู้รับได้นำข้อมูลนั้นไปใช้ จึงเสมือนกับว่าโปรแกรม ในแต่ละชั้นของทางผู้ส่งจะทำงานกับโปรแกรมในชั้นเดียวกันของทางฝ่ายผู้รับโดยไม่สนใจว่าจะมีโปรแกรมในชั้นอื่นๆ อยู่ด้วย การทำงานของโปรแกรมในแต่ละชั้นจึงเป็นอิสระจากโปรแกรมในชั้นอื่น ๆ อย่างแท้จริง และจะปฏิบัติกับข้อมูลที่รับมาราวกับว่าเป็นข้อมูลจริงทั้งหมด(แม้ว่าจะมีข้อมูลของชั้นบน ๆ แทรกเข้ามาด้วยก็ตาม)



รูปที่ 2.2 ตัวอย่างการทำงานของชั้นสื่อสารในรูปแบบ OSI

(ที่มา : Computer Network, Andrew S. Tanenbaum)

2.1.2 รูปแบบระบบเครือข่าย TCP/IP

ระบบเครือข่ายระดับโลกที่มีใช้อยู่ในปัจจุบันคือระบบอินเทอร์เน็ตนั้นมีกำเนิดมาจากระบบเครือข่ายชื่อ APPANET ซึ่งได้รับการสนับสนุนให้ดำเนินการวิจัยโดยมีกระทรวงกลาโหมประเทศสหรัฐอเมริกาเป็นผู้ออกค่าใช้จ่าย ในยุคแรก ๆ นั้นเป็นระบบเครือข่ายที่เชื่อมการติดต่อระหว่างเครื่องคอมพิวเตอร์ในมหาวิทยาลัย และสถานที่ราชการหลายร้อยแห่งในสหรัฐอเมริกาเข้าด้วยกัน โดยใช้สายโทรศัพท์เช่า (Leased lines) เป็นสายสื่อสารหลัก ต่อมาเมื่อระบบการสื่อสาร แบบคลื่นวิทยุความถี่สูง และการสื่อสารดาวเทียมเริ่มเข้ามามีบทบาท และนำมาใช้ในระบบมากขึ้น ทำให้กฎการสื่อสาร (Protocol) ที่เคยใช้ได้ผลดีนั้นเกิดปัญหาไม่สามารถใช้งานได้อีกต่อไป กฎการสื่อสารรุ่นต่อมาจึงได้รับการออกแบบเพื่อนำมาใช้ทดแทนแบบเก่าโดยมีวัตถุประสงค์ในการเชื่อมการติดต่อระหว่างระบบที่มีความแตกต่างกันเรื่องหลัก ผลที่ได้คือกฎการสื่อสารที่เรียกว่ากฎสื่อสารมาตรฐานแบบ TCP/IP ซึ่งได้รับการปรับปรุงจนนำมาใช้งานจริงได้ในปี ค.ศ. 1974 การปรับปรุงรุ่นต่อมาสำเร็จในปี ค.ศ. 1988 รายละเอียดสามารถหาอ่านได้ใน (Clark,1988)

กฎสื่อสารมาตรฐานแบบ TCP/IP ยังมีวัตถุประสงค์หลักอีกสองข้อสำคัญคือ ความสามารถในการแก้ไขปัญหาที่เกิดขึ้นในระบบเครือข่าย เช่น ในกรณีที่ผู้ส่งและผู้รับยังคงมีการติดต่อกันอยู่แต่โหนดกลางที่ใช้เป็นผู้ช่วยรับ-ส่งข้อมูลเกิดเสียหายใช้การไม่ได้ หรือสายสื่อสารบางช่วงถูกตัดขาด กฎการสื่อสารนี้จะต้องสามารถจัดการหาทางเลือกอื่นเพื่อทำให้การสื่อสารดำเนินต่อไปได้โดยอัตโนมัติ ข้อที่สองคือจะต้องมีความอ่อนตัวต่อการสื่อสารข้อมูลได้หลายชนิดทั้งแบบที่ไม่มีความเร่งด่วน เช่น การจัดส่งแฟ้มข้อมูล และแบบที่ต้องการรับประกันความเร่งด่วนของข้อมูล เช่นการสื่อสารแบบ Real-time หรือการสื่อสารแบบโทรศัพท์ (Voice)

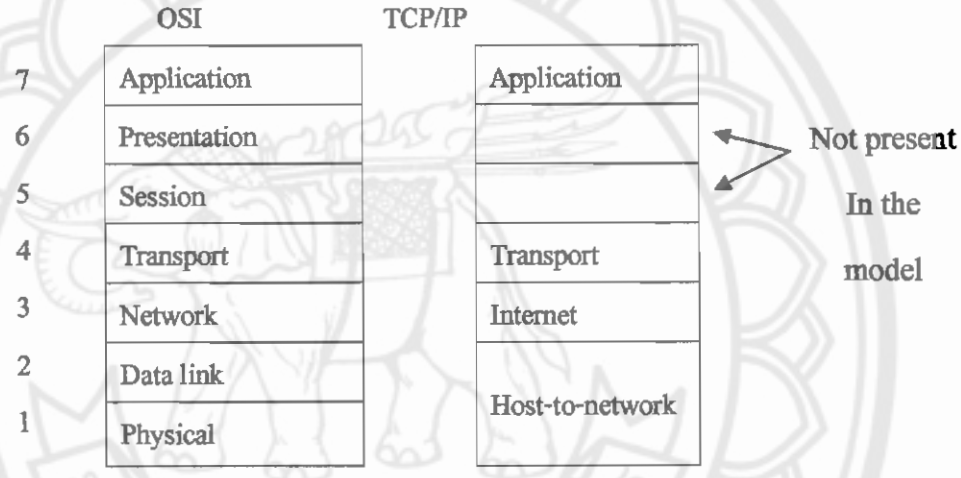
2.1.2.1 ชั้นสื่อสารอินเทอร์เน็ต (The Internet Layer)

จากความต้องการทั้งหมดที่กล่าวถึงทำให้เกิดเป็นวิธีการส่งข้อมูลแบบหนึ่งเรียกว่า ระบบเครือข่ายแบบสลับช่องสื่อสารระดับแพ็กเก็ต (Packet-switching network) ซึ่งเป็นการติดต่อแบบไม่ต่อเนื่อง (Connectionless) หลักการทำงานคือ การปล่อยให้ข้อมูลขนาดเล็กที่เรียกว่า แพ็กเก็ต สามารถไหลจากโหนดผู้ส่งแพ็กเก็ตออกมาเป็นชุดโดยมีจุดหมายปลายทางเดียวกันในระหว่างการเดินทางในเครือข่าย แพ็กเก็ตที่ถูกส่งออกมาเป็นตัวแรกก็ได้ จึงเป็นหน้าที่ของโปรแกรมในชั้น**สูงขึ้นไปที่จะต้องรวบรวมแพ็กเก็ตทั้งหมด แล้วจัดเรียงลำดับให้ถูกต้องก่อนที่จะส่งมอบให้กับโพรเซสผู้รับข้อมูล**

ระบบนี้เปรียบเทียบได้กับระบบไปรษณีย์ระหว่างประเทศ กรมไปรษณีย์ในแต่ละประเทศจะมีการจัดระเบียบต่างๆ เป็นของตนเองซึ่งอาจเหมือนหรือไม่เหมือนกันกับประเทศอื่นก็ได้ คนที่ต้องการส่งจดหมายต้องรับทราบระเบียบต่างๆ เฉพาะบริเวณบังคับใช้โดยกรมไปรษณีย์ในประเทศตนเองเท่านั้นเมื่อส่งจดหมายไปแล้วไปรษณีย์ของประเทศผู้ส่งก็จะจัดการส่งจดหมายฉบับนั้นไป

ตามที่อยู่ผู้รับซึ่งอยู่ในประเทศอื่น เป็นไปได้ว่าจดหมายจะได้รับการฝากส่งไปตามไปรษณีย์ของหลายๆ ประเทศก่อนที่จะไปถึงยังไปรษณีย์ของประเทศผู้รับจะเห็นได้ว่าในทันทีที่จดหมายเดินทางออกนอกประเทศไปแล้วนั้นจะไม่อยู่ในความควบคุมของไปรษณีย์ประเทศผู้ส่งอีกต่อไป ในขณะเดียวกันไปรษณีย์ของประเทศต่างๆ ที่จดหมายเดินทางผ่านอาจมีกฎระเบียบต่างกันออกไป เช่น มี แสตมป์เป็นของตนเอง แต่เนื่องจากได้มีการตกลงกันไว้แล้วจึงยอมรับและช่วยส่งจดหมายไปยังจุดต่อไปให้ กฎระเบียบที่แตกต่างกันนี้ผู้ส่งจดหมายไม่มีความจำเป็นที่จะต้องรับทราบเลย

ชั้นสื่อสารบนอินเทอร์เน็ตกำหนดรูปแบบแพ็กเก็ตและกฎการสื่อสารเรียกว่า IP(Internet Protocol) ดังแสดงในรูป 2.3 ซึ่งเปรียบเทียบ TCP/IP กับรูปแบบระบบเครือข่าย OSI งานสำคัญของ TCP/IP คือการนำแพ็กเก็ตไปส่งยังจุดหมายปลายทางให้ได้



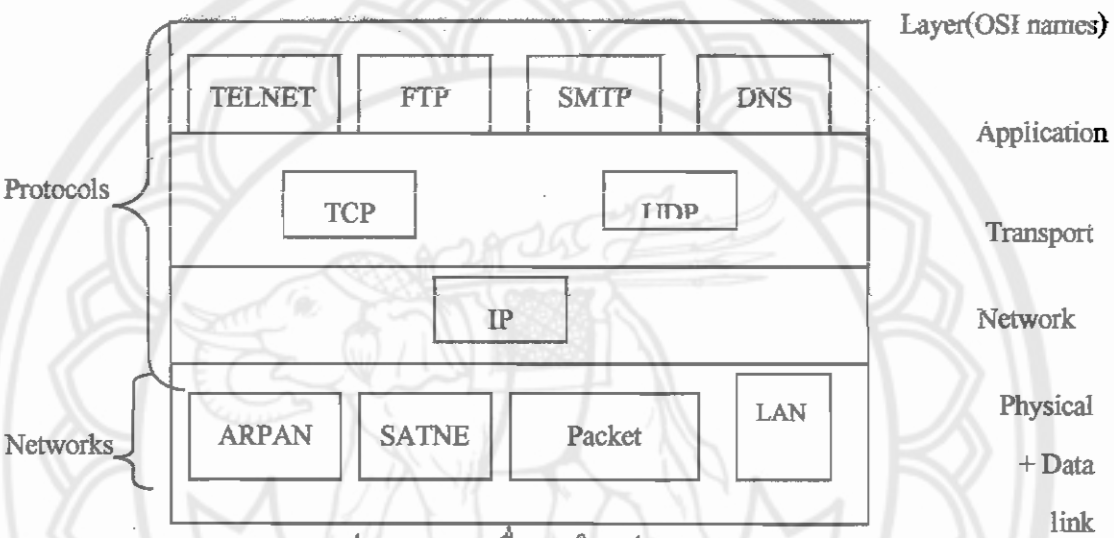
รูป 2.3 รูปแบบโครงสร้าง TCP/IP

(ที่มา : Computer Network , Andrew S. Tanenbaum

2.1.2.2 ชั้นสื่อสารนำส่งข้อมูล (Transport Layer)

ชั้นสื่อสารที่อยู่เหนือชั้น IP คือชั้นนำส่งข้อมูลซึ่งมีหน้าที่การทำงานเหมือนกันกับชั้นจัดการนำส่งข้อมูลของมาตรฐาน OSI แบ่งออกเป็นโพรโตคอลสองประเภท ประเภทแรกเรียกว่า TCP (Transmission Control Protocol) เป็นแบบที่มีการกำหนดช่วงการสื่อสารตลอดระยะเวลาการสื่อสาร (Connection-oriented) ซึ่งจะยอมให้มีการส่งข้อมูลในเป็นกระแสไบต์ (Byte stream) ที่ไว้ใจได้ (Reliable) โดยไม่มีข้อผิดพลาด ข้อมูลที่มีปริมาณมากจะถูกแบ่งออกเป็นส่วนเล็ก ๆ เรียกว่า Message ซึ่งจะถูกส่งไปยังผู้รับผ่านทางชั้นสื่อสารของอินเทอร์เน็ต ทางฝ่ายผู้รับจะนำ Message มาเรียงต่อกันตามลำดับเป็นข้อมูลตัวเดิม TCP ยังมีความสามารถในการควบคุมการไหลของข้อมูลเพื่อป้องกันไม่ให้ผู้ส่งส่งข้อมูลเร็วเกินกว่าที่ผู้รับจะทำงานได้ทันอีกด้วย

กฎการนำส่งข้อมูลแบบที่สองเรียกว่า UDP(User Datagram Protocol) เป็นการติดต่อแบบไม่ต่อเนื่อง (Connectionless) ไม่มีการตรวจสอบความถูกต้องของข้อมูลเหมือนกับแบบ TCP อย่างไรก็ตามวิธีการนี้มีข้อดีในด้านความรวดเร็วในการส่งข้อมูลจึงนิยมใช้ในระบบผู้ให้ และผู้ใช้บริการ (Client/Server system) ซึ่งมีการสื่อสารแบบถาม/ตอบ (Request/Reply) นอกจากนั้นยังใช้ในการส่งข้อมูลประเภทภาพเคลื่อนไหวหรือการส่งเสียง (Voice) ทางอินเทอร์เน็ต รูป 2.4 แสดงให้เห็นความสัมพันธ์ของ IP,TCP และ UDP กฎการสื่อสารแบบ IP นี้เป็นระบบที่ได้รับความนิยมมากและได้รับการนำไปใช้ในระบบสื่อสารหลายระบบ



รูป 2.4 กฎการสื่อสารในรูปแบบ TCP/IP

(ที่มา : Computer Network, Andrew S. Tanenbamu)

2.1.2.3 ชั้นสื่อสารการประยุกต์ (Application Layer)

ตามมาตรฐาน TCP/IP ไม่มีการกำหนดชั้นนำเสนอข้อมูลและชั้นควบคุมหน้าต่างสื่อสารตามที่ปรากฏในรูปแบบมาตรฐาน OSI ทั้งนี้จากประสบการณ์ที่ผ่านมาของการกำหนดมาตรฐานของ OSI พบว่าโปรแกรมสำหรับชั้นควบคุมการสื่อสารสองชั้นนี้จะมีประโยชน์ในการใช้งานจริงน้อยมากระบบเครือข่าย TCP/IP จึงตัดทั้งสองชั้นนี้ออกไป ดังนั้นการประยุกต์จึงกลายเป็นชั้นที่อยู่เหนือชั้นนำส่งข้อมูล

โปรแกรมในชั้นการประยุกต์จะรวมเอาหน้าที่การทำงานที่จะเป็นของสองชั้นที่หายไปไว้ที่นี่ อันได้แก่ โพรโตคอลสำหรับสร้างจอเทอร์มินัลเสมือน เรียกว่า TELNET โพรโตคอลสำหรับการจัดการแฟ้มข้อมูล เรียกว่า FTP และโพรโตคอลสำหรับการให้บริการจดหมายอิเล็กทรอนิกส์ เรียกว่า SMTP ดังรูป 2.4 โพรโตคอลสำหรับสร้างจอเทอร์มินัลเสมือนช่วยให้ผู้ใช้สามารถติดกับเครื่องโฮสต์ที่อยู่ไกลออกไปโดยผ่านทางอินเทอร์เน็ต และสามารถทำงานได้เสมือนกับว่ากำลังนั่งทำงานอยู่ที่เครื่องโฮสต์นั้น โพรโตคอลสำหรับการจัดการแฟ้มข้อมูลช่วยในการคัดลอกแฟ้มข้อมูล

มาจากเครื่องอื่นที่อยู่ในระบบเครือข่าย หรือส่งสำเนาแฟ้มข้อมูลไปยังเครื่องใดๆ ก็ได้ โพรโตคอลสำหรับให้บริการจดหมายอิเล็กทรอนิกส์ช่วยในการจัดส่งข้อความไปยังผู้ใช้ในระบบหรือรับข้อความที่มีผู้ส่งเข้ามา

นอกจากโพรโตคอลที่กล่าวถึงแล้วยังมีโพรโตคอลอีกจำนวนมากที่ได้รับการพัฒนาขึ้นมาใช้งาน เช่น โพรโตคอลสำหรับการบริหารที่อยู่ของโหนดต่างๆ ในระบบเครือข่าย เรียกว่า DNS (Domain Name Service) จะช่วยในการเก็บสำเนารายเชื่อของโหนดต่าง ๆ ในระบบที่เป็นทั้งชื่อในภาษาอังกฤษที่ผู้ใช้ทั่วไปรู้จัก และสามารถจดจำได้ง่าย และชื่อที่เป็นตัวเลขที่ใช้อ้างอิงในระหว่างการสื่อสาร โพรโตคอล NNTP ใช้ในการให้บริการข่าวสาร(News) แก่สมาชิกในระดับกลุ่มและโพรโตคอล HTTP ซึ่งช่วยในการสร้างรูปแบบหน้าจอขั้นก้าวหน้าซึ่งนิยมใช้ในระบบ WWW (World Wide Web) เป็นต้น

2.1.2.4 ชั้นโฮสต์-เครือข่าย (Host-to-Network Layer)

โพรโตคอลสำหรับการควบคุมการสื่อสารในชั้นนี้เป็นสิ่งที่ไม่มีการกำหนดรายละเอียดอย่างเป็นทางการ หน้าที่หลักคือการรับข้อมูลจากชั้นสื่อสาร IP มาแล้วส่งไปยังโหนดที่ระบุไว้ในเส้นทางเดินข้อมูลทางด้านผู้รับก็จะทำงานในทางกลับกันคือรับข้อมูลจากสายสื่อสารแล้วนำส่งให้กับโปรแกรมในชั้นสื่อสาร IP

การที่ไม่กำหนดรายละเอียดเอาไว้อาจเนื่องมาจากเหตุผลสามประการ ประการแรกคือ จากประสบการณ์ที่ได้รับในการกำหนดมาตรฐาน OSI พบว่าเทคโนโลยีทางด้านรับ-ส่งข้อมูลผ่านสื่อประเภทต่าง ๆ นั้นมีการเปลี่ยนแปลงอยู่เสมอ หากมีการกำหนดรายละเอียดก็อาจจะต้องแก้ไขกันอยู่บ่อย ๆ ประการที่สองคือ การสื่อสารที่เกิดขึ้นในระดับนี้เป็นการส่งข้อมูลในระดับกระแสบิต (Bit stream) ซึ่งไม่มีความเกี่ยวข้องกับโครงสร้างข้อมูลในระดับบนแต่อย่างใด ประการสุดท้ายการทำงานระดับนี้จะติดต่อหรือมีความสัมพันธ์กับอุปกรณ์สื่อสารข้อมูลโดยตรง ซึ่งอุปกรณ์เหล่านี้ล้วนแล้วแต่มีมาตรฐานสากลรองรับอยู่อย่างเหลือเฟือแล้ว

2.1.3 การเปรียบเทียบระหว่างมาตรฐาน OSI และ TCP

รูปแบบระบบเครือข่าย TCP/IP มีลักษณะทั่วไปคล้ายคลึงกับรูปแบบระบบเครือข่าย OSI อย่างมาก รูปแบบระบบทั้งสองมีรากฐานบนแนวความคิดเดียวกันคือการสร้างชั้นสื่อสารที่เป็นอิสระแก่กันและกันอย่างสิ้นเชิง รวมทั้งหน้าที่การทำงานในชั้นสื่อสารต่าง ๆ ก็มีความใกล้เคียงกันมากดังจะเห็นได้จากชั้นสื่อสารที่อยู่ในระดับบนสุดลงมาถึงชั้นจัดการนำส่งข้อมูลจะมีการติดต่อจากผู้ส่งถึงผู้รับแบบเสมือนว่าเป็นการติดต่อกันโดยตรง ซึ่งความแตกต่างของระบบเครือข่ายย่อยต่าง ๆ จะถูกปรับให้สามารถสื่อสารกันได้อย่างเรียบร้อย ในขณะที่ชั้นสื่อสารที่สูงกว่าชั้นจัดการนำส่งข้อมูลจะมีความเกี่ยวข้องกับโปรแกรมประยุกต์มากกว่าชั้นสื่อสารอื่น ๆ

อย่างไรก็ตามมาตรฐานทั้งสองแบบก็มีข้อแตกต่างกันอยู่พอสมควร ในที่นี้จะมุ่งความสนใจไปยังความแตกต่างของมาตรฐานทั้งสองแบบ ส่วนความแตกต่างทางด้านโพรโตคอลนั้นจะได้กล่าวถึงในบทต่อ ๆ ไป

องค์ประกอบความคิดพื้นฐานที่สำคัญของรูปแบบระบบเครือข่าย OSI คือ

1. ส่วนการให้บริการ (Service)
2. ส่วนการติดต่อ (Interface)
3. โพรโตคอล (Protocols)

วัตถุประสงค์หลักของรูปแบบระบบเครือข่าย OSI คือการแยกขอบเขตความรับผิดชอบและหน้าที่ขององค์ประกอบทั้งสามออกจากกันอย่างเด็ดขาด โดยที่แต่ละชั้นสื่อสารจะทำหน้าที่ให้บริการแก่ชั้นที่อยู่เหนือขึ้นไปหนึ่งขั้น การให้บริการจะบอกให้ทราบว่าชั้นสื่อสารนั้นๆ ให้บริการอะไรบ้าง แต่ไม่ได้บอกรายละเอียดในวิธีการให้บริการและวิธีการติดต่อขอใช้บริการ

ส่วนการติดต่อ เป็นการอธิบายถึงวิธีการที่ผู้ใช้คือโปรแกรมในชั้นสื่อสารที่อยู่เหนือขึ้นไปหนึ่งขั้นจะสามารถเรียกใช้บริการที่มีอยู่ได้อย่างไร โดยจะต้องมีการกำหนดชื่อบริการ ประเภทและจำนวนพารามิเตอร์ หรือข้อมูลที่ส่งเข้ามา และประเภทและจำนวนของผลลัพธ์

กฎการสื่อสาร หรือโพรโตคอล เป็นการกำหนดรายละเอียดการทำงานของบริการแต่ละชนิดซึ่งจะมีอิสระเต็มที่ในการเลือกวิธีการทำงานอย่างไรก็ได้เพื่อทำงานให้บรรลุตามวัตถุประสงค์ที่กำหนดไว้

องค์ประกอบที่กล่าวถึงนี้ แม้ว่าจะเกิดขึ้นมานานมากแล้ว เมื่อเปรียบเทียบกับเทคโนโลยีใหม่ในปัจจุบัน เช่น เทคโนโลยีการโปรแกรมเชิงวัตถุ (Object-oriented programming) กลับมีความสอดคล้องกันอย่างกลมกลืน นั่นคือ วัตถุ(Object) สามารถเปรียบเทียบได้กับชั้นสื่อสารแต่ละชั้นซึ่งมีฟังก์ชันในการบริการต่างๆ ที่วัตถุอื่น(ชั้นสื่อสารที่อยู่เหนือไปอีกชั้นหนึ่ง) สามารถเรียกใช้ได้ วัตถุมีการกำหนดความหมายของฟังก์ชันที่ให้บริการ ตัวแปรสำหรับฟังก์ชัน รวมทั้งผลลัพธ์ที่ได้ซึ่งก็เปรียบได้กับส่วนการติดต่อ ท้ายที่สุดกฎการสื่อสารหรือโพรโตคอลก็คือการกำหนดรายละเอียดคำสั่งต่างๆ สำหรับแต่และฟังก์ชันซึ่งจะไม่มีวัตถุอื่นเข้ามาก้าวก่ายนอกจากตัววัตถุนั้นเอง

ในส่วนของรูปแบบ TCP/IP นั้นไม่ได้มีการแยกความหมายและหน้าที่ของส่วนการให้บริการส่วนการติดต่อ และโพรโตคอลให้ชัดเจนตั้งแต่เริ่มต้น มีความพยายามที่จะกำหนดนิยามที่ชัดเจนให้กับรูปแบบ TCP/IP ให้เหมือนกับรูปแบบ OSI แต่ก็สามารถทำได้กับบริการเพียงสองชนิดคือ บริการส่ง IP แพ็กเก็ต และบริการรับ IP แพ็กเก็ต ความแตกต่างในข้อนี้แสดงให้เห็นว่า OSI นั้นมีการจัดโครงสร้างภายในที่ดีกว่า TCP/IP ซึ่งทำให้การหาโปรแกรมที่ใช้เทคโนโลยีใหม่มาทดแทนโปรแกรมที่ใช้อยู่เดิมในระบบ OSI นั้นสามารถทำได้ง่ายกว่ามาก

รูปแบบของ OSI ได้รับการออกแบบมาก่อนที่โพรโตคอลหลายอย่างจะได้รับการออกแบบที่สมบูรณ์และสร้างโปรแกรมขึ้นมาใช้งานจริง รูปแบบของ OSI จึงได้รับการออกแบบมา

อย่างมีอิสระทางด้านความคิดอย่างเต็มที่โดยไม่ผูกติดกับแนวความคิดของโพรโตคอลใด ๆ อย่างไรก็ตามผลเสียที่เกิดขึ้นจากการไม่คำนึงถึงโพรโตคอลอื่น ๆ ทำให้การกำหนดหน้าที่การทำงานหลายส่วนไม่มีความเหมาะสมในการใช้งานจริง ตัวอย่างที่เกิดขึ้นได้แก่ปัญหาในชั้นเชื่อมต่อข้อมูล แรกทีเดียวชั้นสื่อสารนี้กำหนดไว้สำหรับการสื่อสารแบบจุด-ต่อ-จุด ต่อมาวิธีการสื่อสารแบบกระจายข่าวเริ่มเป็นที่นิยมมากขึ้น แต่ก็ไม่อาจจะนำเทคโนโลยีการสื่อสารแบบนี้แทรกเข้าไปในรูปแบบ OSI คาดว่าจะมีระบบเครือข่ายเพียงหนึ่งเครือข่ายภายในแต่ละประเทศ แต่ความจริงที่ปรากฏก็คือประเทศที่มีประสบการณ์การใช้งานระบบเครือข่ายจะจัดตั้งระบบเครือข่ายย่อยขึ้นใช้งานมากมาย ทำให้วิธีตั้งชื่อโฮสต์และโหนดต่างต้องได้รับการปรับปรุงใหม่

ทางด้าน TCP/IP นั้นกลับอยู่ในสถานการณ์ตรงกันข้าม นั่นคือโพรโตคอลต่างๆ ได้รับการพัฒนามาเป็นอย่างดีก่อนที่รูปแบบเครือข่าย TCP/IP จะได้รับการพัฒนาขึ้นมาใช้งาน ดังนั้นการผสมผสานโพรโตคอลต่าง ๆ เข้ากับรูปแบบ TCP/IP จึงทำได้อย่างง่ายดาย อย่างไรก็ตามความอ่อนตัวที่มีอยู่นี้ทำให้รูปแบบ TCP/IP ไม่มีการกำหนดโครงสร้างที่ชัดเจน การเปรียบเทียบ TCP/IP กับรูปแบบเครือข่ายอื่น ๆ จึงเป็นเรื่องที่เป็นไปไม่ได้

ข้อแตกต่างที่เป็นรูปธรรมของทั้งสองรูปแบบนี้คือรูปแบบ OSI ประกอบด้วยชั้นสื่อสารเจ็ดชั้นในขณะที่ TCP/IP มีเพียงสี่ชั้น นอกจากนั้นรูปแบบ OSI ให้บริการส่งข้อมูลแบบไม่ต่อเนื่องและแบบต่อเนื่องในชั้นควบคุมเครือข่าย แต่ให้บริการเฉพาะแบบต่อเนื่องในชั้นนำส่งข้อมูล ส่วนรูปแบบ TCP/IP ให้บริการแบบไม่ต่อเนื่องในชั้นควบคุมเครือข่าย แต่ให้บริการทั้งสองชนิดในชั้นนำส่งข้อมูล เนื่องจากการให้บริการในชั้นนำส่ง ข้อมูลเป็นบริการที่ผู้ใช้สามารถนำไปใช้ได้ รูปแบบ TCP/IP จึงเปิดโอกาสให้ผู้ใช้สามารถเลือกวิธีการส่งข้อมูลที่ต้องการได้ ในขณะที่ผู้ใช้ OSI ไม่มีทางเลือก ซึ่งวิธีการติดต่อนี้มีส่วนสำคัญคือ การใช้โพรโตคอลแบบร้องขอและตอบสนอง (Request-response) เป็นอย่างมาก

2.2 ทฤษฎีการสื่อสารบนอินเทอร์เน็ต

ก่อนที่คอมพิวเตอร์จะติดต่อสื่อสารกันได้ทั้ง 2 เครื่องจะต้องใช้ภาษาเดียวกันก่อน โดยเครื่องคอมพิวเตอร์ทุกเครื่องที่ทำการติดต่อกับอินเทอร์เน็ตนั้นจะมีการใช้โพรโตคอล TCP/IP เพื่อให้คอมพิวเตอร์ทุกรุ่น ทุกแบบ สามารถติดต่อสื่อสารกันได้อย่างถูกต้องและเมื่อคอมพิวเตอร์ต่างๆ พูดคุยกันรู้เรื่องโดยใช้โพรโตคอลเดียวกัน การติดต่อสื่อสารก็จะเริ่มขึ้น โดยการติดต่อสื่อสารจะมีอยู่ 2 แบบคือแบบส่งข้อมูล และแบบรับข้อมูล โดยเครื่องคอมพิวเตอร์ที่ทำการส่งข้อมูลเราจะเรียกว่าเครื่องให้บริการ (Service) และเครื่องคอมพิวเตอร์ที่รับข้อมูลเราจะเรียกว่าเครื่องรับบริการ (Client) ซึ่งการติดต่อในลักษณะนี้เราเรียกว่าการติดต่อกันแบบ Client-Server

2.2.1 โดเมนเนม (Domain Name)

ทุก ๆ สถานที่ในอินเทอร์เน็ตจะมี "ที่อยู่" เฉพาะ ฉะนั้นการที่จะเข้าไปชมเว็บเพจนั้นก็จำเป็นที่เราจะต้องทราบที่อยู่ของเว็บนั้นเสียก่อน ซึ่งจะมีลักษณะเฉพาะที่เรียกว่า URL (Unitfrom Resource Locator) ดังนั้นแต่ละเว็บเพจจึงขึ้นต้นด้วย http:// ซึ่งย่อมมาจาก Hypertext Transfer Protocol

ส่วนโดเมนเนมก็คือชื่อสำหรับไว้เรียกเว็บไซต์ที่จะเข้าไปซึ่งมักจะพบในรูปแบบ เช่น www.yahoo.com หรืออาจเขียนโดเมนเนมเหมือนบ้านเลขที่เพราะแต่เดิมนั้นโดเมนเนมจะเป็นตัวเลข เช่น 123.456.7.8 แต่ตัวเลขยากต่อการจำจึงได้มีการใช้ตัวอักษรแทนชุดของตัวเลขซึ่งทำให้ง่ายต่อการใช้งานและจำง่ายด้วย โดยโดเมนเนมจะแยกรายละเอียดได้ดังนี้

www.***.com หมายถึง Top-Level-Domain Name

www.***.com หมายถึง ชื่อโดเมนที่ขอจดไว้ (Second-Level-Domain Name)

www.***.com หมายถึง สับย่อยของโดเมนเนม (Sub Domain Name)

ซึ่งปกติถ้าไม่มีการสร้างสับย่อยเพิ่มก็มักจะเป็นแบบ www.***.com แต่ถ้ามีการสร้างเพิ่มขึ้นก็จะเป็นแบบ yourmane.***.com

หรือ yourcompany.***.com

หรือ www.yourcompany.***.com

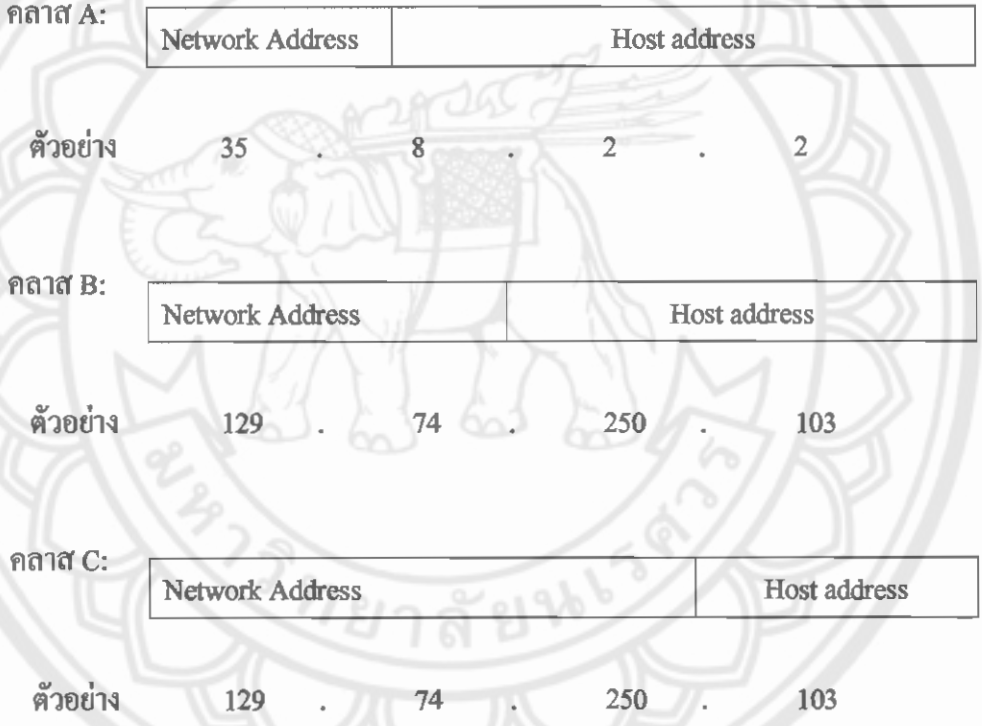
2.2.2 การกำหนดหมายเลขไอพี

ถึงแม้การใช้โดเมนเนมจะเป็นวิธีที่ดีที่สุดในอินเทอร์เน็ตยุคปัจจุบันแต่ความเข้าใจเรื่องวิธีกำหนดหมายเลขไอพีเวิร์กสเตชันเข้าสู่อินเทอร์เน็ตได้และในการค้นข้อมูลในอินเทอร์เน็ต

หมายเลขไอพีประกอบด้วยเลขสี่ส่วน หมายเลขที่หมายถึงคอมพิวเตอร์เครื่องใดเครื่องหนึ่งบนอินเทอร์เน็ต วิธีตั้งหมายเลขไอพีใช้แนวทางที่ทำในเครือข่ายขนาดใหญ่จัดการหมายเลขจำนวนมาก และเครือข่ายขนาดเล็กจัดการเฉพาะส่วนเล็กๆ ใน "หมายเลขไอพีทั้งหมด" คล้ายกับในเรื่องของบริการโทรศัพท์ที่บริษัทโทรศัพท์อนุญาตให้องค์กรที่ใหญ่มากออกหมายเลขโทรศัพท์ของทั้งจังหวัด และให้องค์กรเล็กออกหมายเลขโทรศัพท์ในบริเวณของชุมสายหนึ่งๆ

หมายเลขไอพีแบ่งเป็นระดับหรือคลาส ส่วนซ้ายสุดของหมายเลขไอพีเป็นตัวกำหนดคลาสถ้ามีค่าตั้งแต่ 0 ถึง 127 แสดงว่าเครื่องนั้นอยู่ในระดับเครือข่ายคลาส A ตามทฤษฎีเครือข่ายคลาส A มีโฮสต์คอมพิวเตอร์ได้ถึง 16,777,216 เครื่อง เครือข่ายระดับถัดมาคือคลาส B มีหมายเลขไอพีเริ่มจาก 128 ถึง 191 ตามทฤษฎีมีโฮสต์ได้ 65,536 เครื่อง หมายเลขในคลาส C เริ่มจาก 192 ถึง 233 แต่ละเครือข่ายมีโฮสต์ได้ถึง 256 เครื่อง รูปที่ 2.5 แสดงการแบ่งหมายเลขไอพีออกเป็นคลาสต่างๆ

อินเทอร์เน็ตเป็นเครือข่ายของคอมพิวเตอร์ที่ แต่ละเครือข่ายที่ประกอบกันขึ้นเป็นอินเทอร์เน็ตจะมีผู้ดูแลเครือข่าย (Network administrator) ของตนเอง งานอย่างหนึ่งของผู้ดูแลคือกำหนดหมายเลขไอพีให้กับโฮสต์จากกลุ่มหมายเลขไอพีที่ถูกกำหนดมาให้กับเครือข่ายนั้นๆ ในทางปฏิบัติ ขั้นตอนการกำหนดหมายเลขไอพีสำคัญต่อผู้ดูแลระบบมาก แต่แทบไม่มีความหมายต่อผู้ใช้เลย แต่แทบไม่มีความหมายต่อผู้ใช้เลย เมื่อผู้ดูแลเครือข่ายเลือกหมายเลยไอพีสำหรับคอมพิวเตอร์เครื่องใหม่ที่จะเชื่อมกันกับเครือข่ายได้แล้ว ผู้ดูแลยังต้องเลือกชื่อโฮสต์แบบโดเมนด้วย จากนั้นต้องบันทึกทั้งหมายเลขไอพี และชื่อโฮสต์ (รวมทั้งข้อมูลอื่นที่จำเป็น) ลงในเซิร์ฟเวอร์บริการชื่อ โดเมนสำหรับเครือข่ายนั้น เมื่อเสร็จแล้วผู้ใช้ก็สามารถใช้หมายเลขไอพีและเริ่มใช้ TCP/IP ได้



รูปที่ 2.5 คลาสของเครือข่ายบนอินเทอร์เน็ต

(ที่มี : The internet for everyone, Richard W.Wiggins)

2.2.3 TCP ส่งข้อมูลอย่างไร

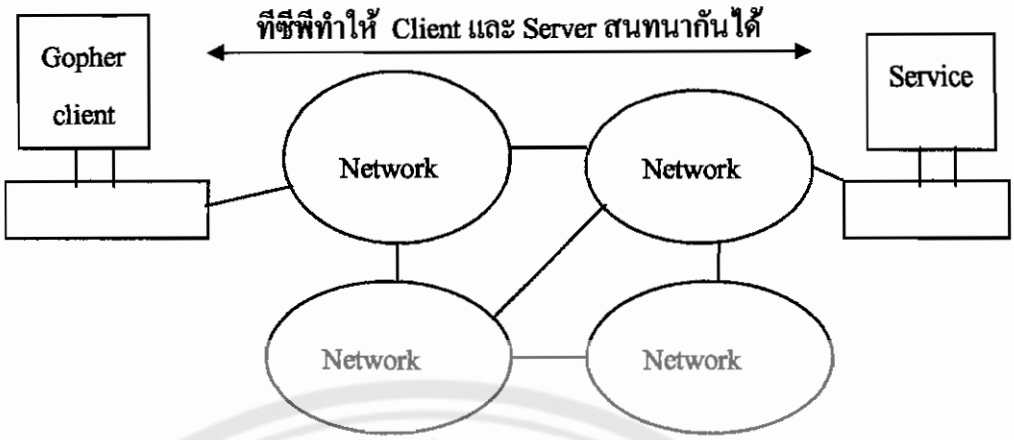
กระบวนการหรือขั้นตอนในการย้ายข้อมูลจากคอมพิวเตอร์ของคุณไปยังคอมพิวเตอร์อีกเครื่องบนเครือข่ายที่คุณใช้หรือบนอินเทอร์เน็ตเป็นเรื่องที่ค่อนข้างซับซ้อน และรายละเอียดว่าขั้นตอนที่ทำงานอย่างไรเป็นเรื่องยุ่งยากจนดูเหมือนว่าถ้าเอาข้อความที่ต้องการส่งพร้อมกับที่อยู่ปลายทางผูกติดกับลูกโป่งและปล่อยให้ลอยไปจะง่ายกว่า ถึงแม้ขั้นตอนนี้จะซับซ้อนแต่เชื่อถือได้

โพรโตคอลหลักที่ใช้กันอยู่บนอินเทอร์เน็ตมีชื่อว่า TCP/IP ความจริง ทีซีพี และไอพีเป็นโพรโตคอลสองโพรโตคอลที่มีหน้าที่ต่างกัน หน้าที่ของทีซีพีคือสร้างกลไกในการส่งข้อมูลไปมาระหว่างคอมพิวเตอร์ที่เชื่อถือได้ ซึ่งหมายถึงว่า ตัวข้อมูลที่ส่งผ่านมาต้องถูกต้องครบถ้วนสมบูรณ์ เรียงลำดับข้อมูลอย่างถูกต้อง และไม่มีการซ้ำซ้อนของข้อมูล หน้าที่ของไอพีคือเคลื่อนย้ายกลุ่มของข้อมูลที่เรียกว่า "ดาต้าแกรม (Datagram) ให้ผ่านเครือข่ายที่โยงใยกันอย่างซับซ้อน(เรียกว่า "อินเทอร์เน็ตเวิร์ก",Internetwork) ซึ่งคั่นระหว่างคอมพิวเตอร์สองเครื่องที่ต้องการติดต่อกันอยู่

เมื่อคอมพิวเตอร์เครื่องหนึ่งต้องการสนทนากับคอมพิวเตอร์อีกเครื่องบนอินเทอร์เน็ต คอมพิวเตอร์จะเริ่มเปิดการเชื่อมต่อที่ทีซีพีกับคอมพิวเตอร์อื่น พูดในทางปฏิบัติคือ เมื่อโปรแกรมประยุกต์เช่นเทลเน็ต หรือโกเฟอร์เซิร์ฟเวอร์ มันจะเริ่มเปิดการเชื่อมต่อทีซีพี ขั้นตอนนี้อาจเทียบได้กับการทำงานของเครือข่ายโทรศัพท์ เมื่อคนกดปุ่มโทรศัพท์ โทรศัพท์ไปยังที่อยู่หนึ่ง (หมายเลขโทรศัพท์) จะมีมือลึกลับ (ระบบสวิทชิ่ง) เรียกไปยังที่อยู่นั้นบนเครือข่ายให้ หลังจากที่มีผู้รับโทรศัพท์ คนก็สามารถแลกเปลี่ยนข้อมูลกันได้จนกว่าทั้งคู่ตัดสินใจวางโทรศัพท์

การเชื่อมต่อระหว่างคอมพิวเตอร์สองเครื่องโดยใช้ทีซีพีทำให้สามารถส่งข้อมูลได้พร้อมกันสองทาง อาจนึกภาพว่าขั้นตอนนี้เป็นการส่งข้อมูลสองสายจากปลายด้านหนึ่งไปยังอีกด้าน ข้อมูลในสายหนึ่งวิ่งในทิศตรงข้ามกับอีกสาย ทีซีพียังมีวิธีการให้ด้านหนึ่งบอกกับอีกด้านว่า "นี่! ข้อมูลนี้เป็นข้อมูลสำคัญนะ" วิธีการนี้มีประโยชน์มาก เช่น ทำให้โปรแกรมไคลเอ็นต์ของเทลเน็ตบอกโปรแกรมเซิร์ฟเวอร์ว่ากดแป้น Enter แล้ว และต้องการให้เซิร์ฟเวอร์ประมวลผลคำสั่งได้ ทีซีพียังมีวิธีการ "ควบคุมการไหลของข้อมูล" เพื่อไม่ให้คอมพิวเตอร์ที่รับข้อมูลได้รับข้อมูลจากฝ่ายส่งมากเกินไป

ทีซีพีช่วยให้โปรแกรมคอมพิวเตอร์สองโปรแกรมแลกเปลี่ยนข้อมูลกันบนเครือข่าย หรือบนกลุ่มเครือข่าย (อินเทอร์เน็ต) ได้ทีซีพีต้องใช้บริการของไอพีในการย้ายข้อมูลระหว่างส่วนย่อยๆ ของเครือข่าย ข้อมูลที่ส่งจะถูกแบ่งเป็นกลุ่มย่อยเรียก ด้าต้าแกรม ซึ่งข้อมูลหนึ่งชุดอาจแบ่งเป็นหนึ่งดาต้าแกรมหรือมากกว่านั้น ดาต้าแกรมแต่ละตัวต้องวิ่งผ่านเครือข่ายหลายเครือข่ายก่อนจะถึงจุดหมายเลยไอพี บนแต่ละเครือข่ายมีอุปกรณ์ที่เรียกว่า "เราเตอร์" (Router) เราเตอร์เป็นตัวตรวจสอบว่าหมายเลขไอพีปลายทางอยู่บนเครือข่ายอื่นหรือไม่ ถ้าอยู่เราเตอร์จะส่งดาต้าแกรมต่อไปยังเครือข่ายอื่น การแบ่งงานระหว่างทีซีพี และไอพีแสดงอยู่ในรูปที่ 2.6



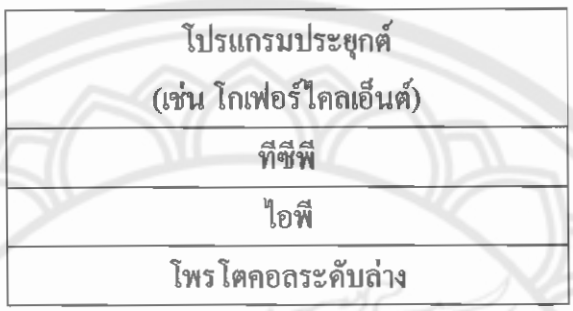
รูปที่ 2.6 ไอพีเป็นผู้ส่งข้อมูลผ่านเครือข่ายที่โยงอยู่ระหว่างทั้งสองเครื่อง
(ที่มา : The internet for everyone. Richard W. Wiggins)

ข้อมูลของคุณต้องถูกแปลงหลายครั้งหลายคราวกว่าที่จะถึงมือผู้รับ เช่นเมื่อคุณเลือกเอกสารในโกเฟอร์ โปรแกรมไคลเอ็นต์ของคุณจะเปิดการเชื่อมต่อแบบทีซีพีไปยังเซิร์ฟเวอร์พร้อมกับขอให้เซิร์ฟเวอร์ส่งเอกสารกลับมาโดยใช้การเชื่อมต่อนั้น คำขอของคุณอาจส่งไปตามเส้นทางหนึ่งบนอินเทอร์เน็ต แต่เอกสารที่ส่งกลับมาอาจใช้เส้นทางใหม่ที่ไม่เหมือนเดิมก็ได้ ความจริงไอพีอาจแบ่งดาต้าแกรมออกเป็นส่วนย่อยลงไปอีก เรียก แฟรกเม้นต์(Fragment) เพื่อให้ส่งได้อย่างมีประสิทธิภาพ แฟรกเม้นต์อาจถูกส่งแยกไปตามเส้นทางต่างๆ กัน ไอพีอาจส่งดาต้าแกรมโดยไม่เรียงลำดับ นอกจากนี้ไอพีไม่สามารถรับรองว่าดาต้าแกรมส่งถึงจุดหมายได้เรียบร้อยเป็นหน้าที่ของทีซีพีที่ต้องตรวจสอบว่าข้อมูลถึงจุดหมายทั้งหมดหรือไม่ ถ้าไม่ถึงจะต้องขอให้ส่งข้อมูลมาใหม่

เหตุใดจึงมีการอนุญาตให้ส่งข้อมูลแยกออกจากกันไปหลายเส้นทางก่อนถึงจุดหมายเลขไอพี เหตุผลหนึ่งคือ ลักษณะของเครือข่าย เช่นมีประสิทธิภาพมากกว่าและเรียกกันว่า "แพ็คเก็ตสวิชชิ่ง(Packet-Switching) ถ้าจะเปรียบเทียบกับถนน ถ้าเส้นทางใดติดขัด และมีเส้นทางอื่นที่สามารถไปถึงจุดหมายเดียวกันได้ ก็ควรไปทางนั้น นอกจากนี้ถ้าเส้นทางไหนเกิดใช้ไม่ได้ การทำงานแบบนี้ทำให้ข้อมูลถูกส่งไปทางอื่นได้ เครือข่ายคอมพิวเตอร์พิเศษที่มีหน้าที่ตัดสินใจว่าจะส่งดาต้าแกรมไปเส้นทางคือ เราเตอร์ งานของเราเตอร์คือส่งดาต้าแกรมไปยังเครือข่ายอื่นเมื่อจุดหมายปลายทางของดาต้าแกรมไม่ได้อยู่บนเครือข่ายของตน เมื่อมีเส้นทางให้เลือกหลายเส้น เราเตอร์เลือกใช้เส้นทางบางเส้นมากกว่าเส้นทางนั้นเร็วกว่าหรือเพราะผู้ดูแลระบบกำหนด

การส่งข้อมูลของคุณยังต้องใช้การทำงานในอีกระดับคือ ข้อมูลต้องส่งผ่านสื่อกลาง เช่น สายอีเธอร์เนต(Ethernet) สายเคเบิ้ลไฟเบอร์ออพติก หรือสายโทรศัพท์ ไอพีต้องใช้บริการของซอฟต์แวร์สำหรับ "ควบคุมการใช้สื่อสาร"(Media access control) ในการส่งและรับข้อมูลผ่านสื่อเหล่านี้ รายละเอียดว่าสื่อกลางเหล่านี้ใช้งานได้อย่างไรเป็นเรื่องที่ไม่สำคัญต่อผู้ใช้ พูดคร่าว ๆ คือ ไอพีต้องให้ความสามารถของ "โพรโตคอลระดับล่าง" อย่างเช่นอีเธอร์เน็ตเพื่อส่งข้อมูลไปตามสาย

การสื่อสาร เหมือนกับที่ทีซีพีต้องใช้ความสามารถของไอพีในการสร้างและส่งคาต้าแกรม แนวความคิดนี้อธิบายได้จากขึ้นของโพรโตคอลในรูปที่ 2.7 ซึ่งใช้อธิบายการสื่อสารแบบ TCP/IP ได้อย่างคร่าว ๆ โชคดีที่เราไม่จำเป็นต้องเข้าใจการทำงานนี้ละเอียดทุกขั้นตอนเพื่อใช้อินเทอร์เน็ต แต่ความเข้าใจเรื่องนี้มีประโยชน์เมื่อคุณใช้เครือข่าย เช่นเมื่อคุณส่งข้อมูลแล้วมีการหยุดเป็นบางครั้ง ก็แสดงว่าดาต้าแกรมอาจวนหาทางออกอยู่ในเส้นทางที่ติดขัด นอกจากนี้ยังเป็นประโยชน์ในการเลือกวิธีเชื่อมต่อกับอินเทอร์เน็ตแบบต่างๆ



รูปที่ 2.7 ชั้นของโพรโตคอล

(ที่มี : The internet for everyone, Richard W.Wiggins)

การพัฒนาระบบการจัดการฐานข้อมูลในปัจจุบัน เราจำเป็นที่ต้องรู้จักการทำงานของระบบฐานข้อมูล องค์ประกอบ และคำศัพท์ต่าง ๆ ที่มีใช้ในการออกแบบระบบฐานข้อมูลเพื่อจะได้ใช้เป็นความรู้ในการออกแบบ และพัฒนาระบบฐานข้อมูลได้ ทั้งยังสามารถมองถึงปัญหาของระบบและแก้ไข เพื่อให้ระบบมีประสิทธิภาพสูงสุดได้ ซึ่งจำแนกเนื้อหาได้ดังนี้

2.3ระบบฐานข้อมูล

ระบบแฟ้มข้อมูล

ในอดีตระบบจัดเก็บข้อมูลยังมีการจัดเก็บเป็นแฟ้มเอกสาร เพื่อเป็นหมวดหมู่ ซึ่งอาจจะเป็นข้อมูลที่มีความสำคัญ หรือไม่ความสำคัญ ปะปนกันไป เช่น การจัดหมวดหมู่ของยา มักจัดตามกลุ่มอาการที่ใช้รักษา และเมื่อหน่วยงาน หรือองค์กรต่าง ๆ มีขนาดใหญ่ขึ้น เอกสารและข้อมูลทางเภสัชกรรมจึงมีมากขึ้น เมื่อมีความจำเป็นต้องใช้หรือตรวจเช็คข้อมูล จึงยากและลำบากที่จะค้นหา จึงได้มีการนำเครื่องคอมพิวเตอร์เข้ามาช่วยในการทำงานของหน่วยงาน หรือ องค์กรนั้น ๆ ซึ่งรูปแบบของข้อมูลจะมีอยู่ใน รูปแฟ้มข้อมูล เมื่อแฟ้มข้อมูลรวมกันอยู่มาก ๆ หรืออาจจะมีความสัมพันธ์กัน เราจึงนำมาไว้ด้วยกันเรียกว่า “ระบบแฟ้มข้อมูล”

ซึ่งการที่เราจะนำข้อมูลที่อยู่ในระบบแฟ้มข้อมูลมาใช้นั้น เราจะต้องอาศัยผู้ที่มีความรู้เข้าช่วยซึ่งเรียกว่า “โปรแกรมเมอร์” เพื่ออ่านข้อมูลต่าง ๆ เข้ามาช่วยในการโปรเซสของข้อมูล ซึ่ง

ใช้ภาษาต่าง ๆ เช่น ฟอร์แทน, เบสิค เป็นต้น ซึ่งมีผลตามมาคือ ข้อจำกัดของการเรียกใช้ข้อมูล และความซับซ้อนในการพัฒนาโปรแกรม เช่นเมื่อต้องการเปลี่ยนแปลงแฟ้มข้อมูล โปรแกรมเมอร์ต้องเขียนโปรแกรมขึ้นมาใหม่

ปัญหาของระบบแฟ้มข้อมูล

เมื่อมีการจัดเก็บข้อมูลอย่างกระจัดกระจาย ของแต่ละหน่วยงาน หรือแต่ละ องค์กร ซึ่งอาจจะทำให้เกิดปัญหาของ ข้อมูลซ้ำซ้อน เกิดการขัดแย้งของข้อมูลทำให้ผู้ใช้เข้าใจผิดเมื่อต้องการใช้ข้อมูลได้

ระบบฐานข้อมูล

จากการที่เกิดปัญหาจากหัวข้อที่ผ่านมา เราจึงต้องมีวิธีในการจัดเก็บข้อมูล ซึ่งเรานำมาเก็บในรูปแบบ "ฐานข้อมูล" นั้นเอง ซึ่งมีหลักสำคัญคือ การนำข้อมูลที่มีความสัมพันธ์กันมาไว้ในฐานข้อมูลเดียวกัน เพราะต้องอาศัยข้อมูลของกันและกันในฐานข้อมูลนั้น ๆ

องค์ประกอบของระบบฐานข้อมูล

1. ข้อมูล
2. ฮาร์ดแวร์
3. ซอฟแวร์
4. ผู้ใช้หรือผู้ออกแบบฐานข้อมูล

หน้าที่ของ DBMS

สำหรับหน้าที่ของโปรแกรม DBMS มีดังนี้

1. ทำหน้าที่แปลงคำสั่งที่ใช้จัดการกับข้อมูลภายในฐานข้อมูล ให้อยู่ในรูปแบบที่ฐานข้อมูลเข้าใจ
2. ทำหน้าที่ในการนำคำสั่งต่างๆ ซึ่งได้รับการแปลแล้ว ไปสั่งให้ฐานข้อมูลทำงาน เช่น การเรียกใช้ข้อมูล (Retrieve) การจัดเก็บข้อมูล (Update) การลบข้อมูล (Delete) การเพิ่มข้อมูล (Add) เป็นต้น
3. ทำหน้าที่ป้องกันความเสียหายที่จะเกิดกับข้อมูลภายในฐานข้อมูล โดยจะคอยตรวจสอบว่าคำสั่งใดที่สามารถทำงานได้ และคำสั่งใดที่ไม่สามารถทำงานได้
4. ทำหน้าที่รักษาความสัมพันธ์ของข้อมูลภายในฐานข้อมูลให้มีความถูกต้องอยู่เสมอ
5. ทำหน้าที่เก็บรายละเอียดต่าง ๆ ที่เกี่ยวข้องกับข้อมูลภายในฐานข้อมูลไว้ใน Data Dictionary ซึ่งรายละเอียดเหล่านี้จึงมีจะถูกเรียกว่า "ข้อมูลของข้อมูล" (Metadata)

ประโยชน์ของฐานข้อมูล

1. สามารถลดความซ้ำซ้อนของการเก็บข้อมูลได้

2. สามารถหลีกเลี่ยงความขัดแย้งข้อมูลได้
3. ในแต่ละหน่วยงานสามารถใช้ข้อมูลเดียวกันได้
4. สามารถรักษาความถูกต้องให้กับข้อมูลได้
5. สามารถกำหนดระดับความปลอดภัยให้กับข้อมูลได้
6. ทำให้ข้อมูลเป็นอิสระจากโปรแกรมคือเมื่อต้องการแก้ไขข้อมูลจะไม่ส่งผลกระทบต่อตัวโปรแกรมเลย

สถาปัตยกรรมของระบบฐานข้อมูล

เป็นการอธิบายถึง รูปแบบและโครงสร้างของข้อมูล โดยทั่วไปตามแนวความคิด ไม่ขึ้นกับโครงสร้างของฐานข้อมูลนั้น สถาปัตยกรรมที่นิยมใช้กันคือ

สถาปัตยกรรม ANSI/SPARC แบ่งเป็น 3 ระดับคือ

1. ระดับ Internal ซึ่งจะเกี่ยวข้องกับระดับกายภาพ ในการจัดเก็บข้อมูลในฐานข้อมูลมากที่สุด ซึ่งจะเกี่ยวกับการจัดเก็บข้อมูล
2. ระดับ External ซึ่งจะเกี่ยวกับผู้ใช้มากที่สุด จะพูดถึงมุมมองและความคิดของผู้ใช้ในแต่ละคน
3. ระดับ Conceptual จะกล่าวถึง โครงสร้างข้อมูลในระดับแนวคิดแสดงออกมาในรูปแบบ ของภาพของโครงสร้างทางกายภาพ

Mapping

ในสถาปัตยกรรม ANSI/SPARC นี้มุมมองที่มีต่อข้อมูลในสถาปัตยกรรมในระดับที่สูงกว่า สามารถที่จะถ่ายทอดมุมมองนั้นไปยังสถาปัตยกรรมในระดับที่ต่ำกว่าได้ เช่น การถ่าย-ทอดมุมมองที่มีต่อข้อมูลจากสถาปัตยกรรมในระดับ Conceptual ไปยังสถาปัตยกรรมในระดับ Internal เพื่อนำโครงสร้างของข้อมูลในระดับ Conceptual Schema ไปแปลงเป็นโครงสร้างของข้อมูลในระดับกายภาพเพื่อใช้ในการจัดเก็บข้อมูลเป็นต้นสำหรับการถ่ายทอดมุมมองจากสถาปัตยกรรมในระดับที่สูงกว่าไปยังระดับที่ต่ำกว่านี้ จะเรียกว่า "การทำ Mapping" ในการทำ Mapping ตามสถาปัตยกรรม ANSI/SPARC สามารถแบ่งออกได้เป็น 2 ลักษณะดังนี้

สถาปัตยกรรมของระบบฐานข้อมูลกับ Database Administrator

Database Administrator (DBA) เป็นบุคคลที่มีหน้าที่ในการกำหนดกลยุทธ์และกุศโลบายในการใช้ข้อมูลภายในฐานข้อมูล รวมทั้งจัดหาเทคนิคที่จำเป็นต่อกลยุทธ์และกุศโลบายที่กำหนดขึ้น ดังนั้นผู้ใช้ทำหน้าที่เป็น DBA จึงมีความเกี่ยวข้องกับสถาปัตยกรรมของระบบฐานข้อมูลดังนี้

19 S.A. 2547 ห้องสมุดคณะวิศวกรรมศาสตร์ 4700195

1. เป็นผู้กำหนดโครงสร้างของข้อมูลสถาปัตยกรรมในระดับ Conceptual เช่นการกำหนดว่า ข้อมูลใดบ้างที่ควรที่จะจัดเก็บไว้ในฐานข้อมูล และมีโครงสร้างเป็นอย่างไร

2. เป็นผู้กำหนดโครงสร้างของข้อมูลในสถาปัตยกรรมในระดับ Internal ซึ่งได้แก่โครงสร้างทางกายภาพของข้อมูลที่ใช้ในการจัดเก็บเช่น ขนาดของแต่ละ Field ประเภทของข้อมูล เป็นต้น

3. เป็นผู้ที่ทำหน้าที่ตรวจสอบโครงสร้างข้อมูลที่กำหนดขึ้นว่าสามารถรับตัวมุมมองหรือความต้องการในการใช้ข้อมูลของผู้ใช้ในสถาปัตยกรรมในระดับ Extemal หรือไม่

4. เป็นผู้กำหนดการรักษาความปลอดภัย และกฎที่ใช้ในการควบคุมความถูกต้องให้กับข้อมูลซึ่งเป็นส่วนหนึ่งของการกำหนดโครงสร้างของข้อมูลในสถาปัตยกรรมในระดับ Conceptual

5. เป็นผู้กำหนดวิธีในการสำรองข้อมูล (Data Backup) และการกู้ข้อมูลที่เสียหายกลับมาใช้งาน (Data Recovery)

6. เป็นผู้ควบคุมให้ระบบฐานข้อมูลมีประสิทธิภาพ และทันสมัยตามความต้องการที่เปลี่ยนไป

สถาปัตยกรรมของระบบฐานข้อมูลกับ Database Management System

Database Management System (DBMS) เป็นส่วนที่มีความสัมพันธ์กับสถาปัตยกรรมของระบบฐานข้อมูลเนื่องจาก DBMS เป็นโปรแกรมที่ทำหน้าที่รับคำสั่งต่างทั้งในกลุ่มคำสั่ง DML และกลุ่มคำสั่ง DDL ที่ผู้ใช้กำหนดในสถาปัตยกรรมในระดับ Conceptual มาแปลงให้อยู่ในรูปแบบของโครงสร้างข้อมูลในสถาปัตยกรรมในระดับ Conceptual และ External ตามลำดับ เพื่อนำมาแสดงผลต่อผู้ใช้

แบบจำลองของฐานข้อมูลเชิงสัมพันธ์ (Relation Database Model)

เป็นโครงสร้างในแบบ Relational พัฒนาจากแบบจำลองที่กล่าวถึงความสัมพันธ์ระหว่างข้อมูลที่เรียกว่า Relation Model ข้อมูลที่จัดเก็บในแบบนี้ จะเก็บเป็นส่วนย่อย ๆ ที่เรียกว่า Table อยู่ในตารางที่มีความสัมพันธ์ของสดมภ์และแถว โดยข้อมูลแต่ละตารางจะเก็บเป็นเอกเทศ แต่สามารถเอามาสัมพันธ์กันได้ตามแนวความคิด

โครงสร้างของฐานข้อมูลแบบ Relational

ใน Relation Model ได้นิยามต่างๆ คือ Relation, Attribute, Domain เพื่ออธิบายถึงข้อมูลใน Relation Model นี้

i 438907
ปร.
ป 621 ก.
2546

Relation

เป็นการจัดเก็บข้อมูลแบบตารางมีแถวและสดมภ์ ซึ่งของแต่ละแถวเรียกว่า "Tuple" ส่วนชื่อของแต่ละสดมภ์เรียกว่า "Attribute" ดังตารางที่ 2.2

ตารางที่ 2.1 ตัวอย่างที่ Relation

EmpID	Name	Last name	Sex	Salary	DepID
00I	เกียรติศักดิ์	สิงห์ศิริเจริญกุล	M	10,000	02
001	พรเทพ	เกิดจันทึก	M	10,000	03

คุณสมบัติของ Relation

คุณสมบัติของ Tuple และ Attribute ของแต่ละ Relation จะประกอบด้วย

1. เนื่องจาก Relational ใน Relational Model อยู่ในรูปแบบของเซตทางคณิตศาสตร์ ที่ภายในเซตจะต้องประกอบด้วยสมาชิกที่มีค่าไม่ซ้ำกัน ดังนั้น ภายใน Relation ใด ๆ จึงต้องมี Attribute ใด Attribute หนึ่ง ที่ทำให้แต่ละ Tuple ใน Relation มีข้อมูลที่ไม่ซ้ำกัน เช่น Relation "POPULAR" ที่ใช้เก็บข้อมูลของประชากรในประเทศไทยซึ่งถึงแม้ว่าจะมีบุคคลที่มีชื่อและนามสกุลที่ซ้ำกัน เช่น ประชากรที่ชื่อ "นายสมบูรณ์ สุขมาก" แต่ข้อมูลในแต่ละ Tuple ของ Relation "POPULAR" ก็จะไม่ปรากฏข้อมูลที่ไม่ซ้ำกันดังตัวอย่างข้อมูลต่อไปนี้

ตารางที่ 2.2 ตัวอย่าง Relation

POPULAR

ID	Name	Surname	Sex
1230000100	สมบูรณ์	สุขมาก	M
1230000101	สมเกียรติ	เจริญพร	M
1230000102	สมบูรณ์	สุขมาก	M
1230000103	น้ำฝน	ม่วงทอง	F

2. ด้วยเหตุผลเช่นเดียวกับข้อที่ 1 ลำดับที่ของสมาชิกภายในเซตใด ๆ จะไม่มีผลต่อเซตนั้น ดังนั้น ภายใน Relation จึงไม่มีการกำหนดลำดับที่ให้กับแต่ละ Tuple ใน Relation กล่าวคือ จะไม่มีการกล่าวถึงคำว่า Tuple แรก หรือ Tuple สุดท้าย หรือ Tuple ลำดับที่ 5 หรือ Tuple ที่ผ่านมาใน Relation

3. ภายใน Relation จะไม่มีการกำหนดลำดับที่ให้กับแต่ละ Attribute เนื่องจากในการอ้างอิง Attribute ใน Relation จะใช้ชื่อของ Attribute นั้นในการอ้างอิง ดังนั้น จึงไม่มีการกล่าวถึงคำว่า Attribute แรก หรือ Attribute สุดท้ายหรือ Attribute ลำดับที่ 5 หรือ Attribute ที่ผ่านมา หรือ Attribute ถัดไปใน Relation เช่นเดียวกับ tuple

4. ค่าในทุก Attribute ของ Relation จะต้องมีคุณสมบัติ Atomicity ซึ่งเป็นคุณสมบัติที่กำหนดให้ค่าของข้อมูลในแต่ละ Attribute ของ Relation จะต้องมีความหมายใด ความหมายหนึ่งเพียงความหมายเดียวไม่ใช่กลุ่มของสิ่งใดสิ่งหนึ่ง หรือกล่าวอีกนัยหนึ่ง ข้อมูลในแต่ละ Attribute ของ Relation จะต้องไม่ใช่ข้อมูลในลักษณะ Repeating Group เช่น กรณีที่พนักงานสามารถสังกัดฝ่ายได้มากกว่า 1 ฝ่าย ข้อมูลในการ Attribute "DeptID" ของแต่ละ Tuple ของ Relation "EMPLOYEE" ซึ่งใช้เก็บรหัสของฝ่ายที่พนักงานแต่ละคนสังกัด จะไม่สามารถจัดเก็บทุกรหัสของฝ่ายที่พนักงานคนนั้นสังกัดภายใน Tuple เดียวดังตัวอย่างข้อมูลต่อไปนี้ได้

ตารางที่ 2.3 ตัวอย่าง Relation

EMPLOYEE

EmpID	Name	Surname	Sex	Salary	DeptID
00001	สมบูรณ์	สุขมาก	M	10,000	01,03
00002	สมเกียรติ	เจริญพร	M	8,000	02
00003	จันจิรา	แจ้งเกิด	F	12,000	03
00004	น้ำฝน	ม่วงทอง	F	9,500	01

จากรูปจะสังเกตเห็นว่า ข้อมูลที่เก็บอยู่ใน Attribute "DeptID" พนักงานที่ชื่อ "สมบูรณ์" มีค่ามากกว่า 1 ค่า ซึ่งจากรูปคือ รหัสฝ่าย "01" และ "03" ดังนั้น Relation นี้จึงต้องมีการแยก ข้อมูลในลักษณะ Repeating Group นั้นออกมาเป็น Tuple ใหม่ดัง ตัวอย่างข้อมูลต่อไปนี้

ตารางที่ 2.4 ตัวอย่าง Relation

EMPLOYEE

EmpID	Name	Surname	Sex	Salary	DeptID
00001	สมบูรณ์	สุขมาก	M	10,000	01
00001	สมบูรณ์	สุขมาก	M	10,000	03
00002	สมเกียรติ	เจริญพร	M	8,000	02
00003	จันจิรา	แจ้งเกิด	F	12,000	03
00004	น้ำฝน	ม่วงทอง	F	9,500	01

สำหรับคุณสมบัติของ Relation ในข้อนี้ ได้บังคับให้ทุก Relation มีรูปแบบการจัดเก็บข้อมูลที่เป็นไปตามรูปแบบแรก (First Normal Form) ตามที่กำหนดไว้ในการทำ Nomalization ให้กับ Relation ต่างๆ สำหรับรายละเอียดของการทำ Nomalization ให้กับ Relation ต่าง ๆ สำหรับรายละเอียดของการทำ Nomalization ให้กับ Relation จะกล่าวถึงในลำดับต่อไป

5. ชื่อของแต่ละ Attribute ใน Relation เดียวกัน จะต้องมีชื่อที่ไม่ซ้ำกัน

6. ค่าที่ปรากฏในแต่ละ Attribute เดียวกันจะต้องให้แทนข้อมูลที่มีความหมายเดียวกัน เช่น Attribute "EmpID" ของ Relation "EMPLOYEE" จะต้องใช้เก็บข้อมูลรหัสพนักงานของแต่ละ Tuple เท่านั้น จะไม่สามารถใช้เก็บข้อมูลอื่น ๆ เช่น เพศ ได้ดังตัวอย่างข้อมูลต่อไปนี้

ตารางที่ 2.5 ตัวอย่าง Relation

EMPLOYEE

EmpID	Name	Surname	Sex	Salary	DeptID
00001	สมบูรณ์	สุขมาก	M	10,000	01
00002	สมเกียรติ	เจริญพร	M	8,000	02
00003	จันจิรา	แจ้งเกิด	F	12,000	03
00004	**น้ำฝน**	**ม่วงทอง**	**F**	**9,500**	**01**

ขั้นตอนการออกแบบฐานข้อมูล

การออกแบบฐานข้อมูลแบ่งออกได้เป็น 2 ขั้นตอนดังนี้

การออกแบบฐานข้อมูลในระดับ Conceptual

การออกแบบฐานข้อมูลในระดับนี้จะเป็นการกำหนดโครงร่างเริ่มต้น ที่มีจุดมุ่งหมายเพื่ออธิบายโครงสร้างหลัก ๆ ของข้อมูลภายในระบบฐานข้อมูลโดยไม่คำนึงถึง ฐานข้อมูลที่นำมาใช้ การออกแบบในระดับนี้มีความสำคัญมาก เนื่องจากโครงสร้างที่ได้จากการออกแบบในขั้นตอนนี้จะถูกนำไปใช้ในขั้นตอนต่อไปนี้ โครงร่างหรือที่เรียกว่า Schema ที่ได้จากการออกแบบในขั้นตอนนี้เรียกว่า Conceptual Schema

การออกแบบในระดับ Logical

การออกแบบในระดับนี้จะเป็นระดับที่ต่อเนื่องมาจากระดับ Conceptual กล่าวคือ การออกแบบในระดับนี้จะอาศัยโครงร่างที่จากการออกแบบในระดับ Conceptual มาปรับปรุงให้มีโครงสร้างที่เป็นไปตามโครงสร้างข้อมูลที่จะนำมาใช้งาน โดยยังไม่คำนึงถึงผลิตภัณฑ์ทางด้านฐานข้อมูลที่จะนำมาใช้งาน การออกแบบในขั้นตอนนี้ต้องปรับปรุงโครงสร้างบางอย่างใน Conceptual Shcema ให้สอดคล้องกับฐานข้อมูลที่จะนำมาใช้งาน การออกแบบในขั้นตอนนี้จึงต้องมีการตรวจสอบความถูกต้องของโครงร่างที่ออกแบบขึ้นกับส่วนประมวลผลต่าง ๆ ที่ออกแบบไว้รวมทั้งจะต้องแปลงโครงร่างต่าง ๆ ให้อยู่ในรูป Relation

การออกแบบฐานข้อมูลในระดับ Physical

การออกแบบในระดับนี้จะเป็นขั้นตอนสุดท้ายของการออกแบบฐานข้อมูลในขั้นตอนนี้ จะเป็นการปรับปรุงโครงสร้างของโครงร่างที่ออกแบบเช่นเดียวกัน แต่การปรับปรุงโครงสร้างของการออกแบบฐานข้อมูลในขั้นตอนนี้ จะเป็นการนำเอาโครงร่างที่ได้จากการออกแบบในระดับ Logical มาปรับปรุงโครงสร้างให้เป็นไปตามโครงสร้างของผลิตภัณฑ์ทางด้านฐานข้อมูลที่จะนำมาใช้งาน ผลลัพธ์ที่ได้จากการออกแบบในระดับนี้คือ **โครงสร้างของระบบฐานข้อมูล ที่สามารถนำไปใช้ในการสร้างตัวฐานข้อมูลจริง**

Data Flow Diagram(DFD)

กรรมวิธีการวิเคราะห์ระบบอย่างมีโครงสร้างนั้น วิธีหนึ่งนิยมในทางปฏิบัติคือ การมองภาพรวมในรูปแบบการไหลของข้อมูล (Data Flow) โดยที่วิธีนี้จะช่วยให้นักวิเคราะห์สามารถแบ่งระบบเป็นระบบเป็นระบบย่อยได้ง่ายขึ้นและสามารถตรวจสอบได้สะดวก

การนำเสนอระบบแบบการไหลของข้อมูลนั้นจะใช้สัญลักษณ์แทนการบรรยายการทำงานของระบบ ซึ่งลักษณะที่จะใช้จะเป็นรูปวงกลม สี่เหลี่ยมจัตุรัสสี่เหลี่ยมผืนผ้าปลายเปิด เส้นโค้ง ลูกศร โดยนำสัญลักษณ์เหล้านี้มาเชื่อมต่อ การแสดงการต่อเนื่องของข้อมูลและการประมวลผล

สัญลักษณ์ Data Diagram (DFD)

ในแผนภาพของ DFD จะประกอบด้วยสัญลักษณ์ต่างๆ ดังนี้

1. ลูกศร ใช้แทนการไหลของข้อมูลพร้อมกับชื่อของข้อมูลนั้น ๆ จะต้องกำกับไว้ด้วย

รูปที่ 2.8 การแทนกระแสข้อมูลเป็นลูกศร

2. รูปวงกลม ใช้แทนกริยาการกระทำต่อข้อมูลที่ไหลเข้ามา โดยไม่คำนึงถึงว่าจะเป็นการกระทำโดยคนหรือคอมพิวเตอร์ก็ตาม จะได้มาซึ่งผลลัพธ์ที่จะไหลออกจากวงกลมภายในวงกลมจะระบุคำสั้นๆ ที่จะใช้แทนการกระทำต่อข้อมูล

A = ข้อมูลเข้า

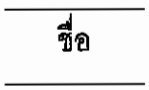
B = ข้อมูลออก

รูปที่ 2.9 การแทนกระแสข้อมูลเป็นลูกศร

3. รูปสี่เหลี่ยม ใช้แทนนามที่อยู่ภายนอกระบบซึ่งเป็นการกำเนิดของข้อมูลหรือสิ้นสุดของข้อมูล โดยมีชื่ออยู่ในสี่เหลี่ยม

รูปที่ 2.10 การแทนนามที่อยู่นอกระบบ

4. รูปสี่เหลี่ยมผื่นผ้าปลายเปิด เป็นตัวแทนของแหล่งเก็บข้อมูลหรือ แฟ้มข้อมูลเสมือนเป็นตัวพักหรือช่วงขาดของการไหลของข้อมูลเพื่อนำไปเก็บเท่านั้น การกำหนดชื่อของแหล่งเก็บข้อมูลต้องอยู่ในสี่เหลี่ยม



รูปที่ 2.11 การแทนแหล่งเก็บข้อมูล

4. **สัญลักษณ์เพิ่มเติม** จะใช้เติมลงในสัญลักษณ์ที่กล่าวมาข้างต้นเพื่อแสดงความเป็นสิ่งเดียวกัน แต่จะถูกกล่าวหลายๆ ครั้งในแผนภาพ

รูปที่ 2.12 การแทนสัญลักษณ์เพิ่มเติม

ลำดับชั้นใน Data Flow Diagram

ในการเขียน DFD นักวิเคราะห์ระบบจะต้องมองระบบจากภาพรวมก่อนจากนั้นมองลึกเข้าสู่รายละเอียดข้างในของระบบยิ่งมองลึกมากเท่าใด ก็ยิ่งเห็นรายละเอียดของระบบย่อยได้มากขึ้นเท่านั้น

DFD ระดับที่ 0

ให้คือว่าระบบทั้งระบบเป็น PROCESS หรือวงกลมหนึ่งวง มีลูกศรแทน INPUT และ OUTPUT ตามที่จำเป็น

DFD ระดับที่ 1

ให้แตกวงกลมที่ลำดับ 0 ออกเป็นวงกลมย่อย 2-5 วงตามความเหมาะสม

DFD ระดับที่ 2

ให้แตกวงกลมที่ลำดับ 1 ออกเป็นวงกลมย่อยลงไปอีกเท่าที่จะทำได้

DFD ระดับที่ 3

ถ้าจำเป็นก็ต้องตรวจดูว่า วงกลมใดในภาพลำดับที่ 2 ยังมีความซับซ้อนที่จำเป็นต้องแตกย่อยก็ต้องแตกย่อย ก็ต้องสร้างแผนภาพประกอบด้วยวงกลมย่อยแทนวงกลมนั้นให้ได้รายละเอียดสุดท้าย

ทำไมจึงต้องใช้ Data Flow Diagram

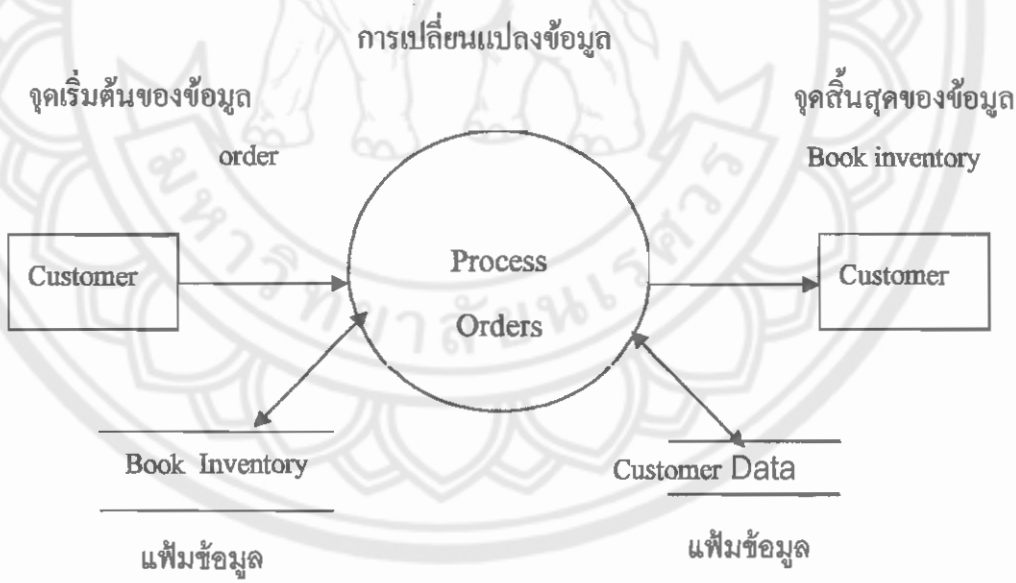
เหตุผลที่ต้อง DFD เป็นแผลภาพของระบบก็เพราะ DFD ทำให้นักวิเคราะห์ระบบสามารถ

- สรุปข้อมูลที่เกี่ยวกับระบบ
- เข้าใจถึงปัญหาสำคัญของระบบและระบุส่วนของการทำงานที่ซ้ำซ้อน
- เข้าใจถึงความสัมพันธ์ระหว่างส่วนต่างๆ ของระบบการประกอบกันเป็นระบบ
- พัฒนาระบบได้อย่างมีประสิทธิภาพ

DFD เป็นเอกสารร่วมที่ช่วยให้นักวิเคราะห์ระบบและผู้ใช้สามารถเข้าใจระบบและตรวจสอบความถูกต้องได้สองฝ่าย

ในการตรวจสอบเรื่องเวลาที่ใช้ในแต่ละขบวนการนั้น นักวิเคราะห์ สามารถใช้ DFD เป็นเครื่องมือที่ช่วยให้ทราบถึงขอบเขตในการพัฒนารูปแบบของระบบว่ามีทางที่จะเป็นไปได้อย่างไรบ้าง

ตัวอย่างการใช้ Data Flow Diagram



รูปที่ 2.13 การปฏิบัติงานของบริษัท หนังสือไทย

เป็นแผนภาพแสดงการสั่งซื้อหนังสือจากร้านหนังสือไทยลูกค้าในช่องสี่เหลี่ยมซ้ายมือสั่งสินค้ามายังวงกลม สิ่งที่จะต้องทำก็คือ ตัดสต๊อกของหนังสือที่ถูกสั่งออกไปพร้อมกับใบเก็บเงินไปยังลูกค้าด้านขวามือ โดยรูปแบบของสี่เหลี่ยมจัตุรัส ของลูกค้าทั้งด้านซ้ายและขวามีลักษณะเหมือนกัน จะหมายถึงสิ่งเดียวกัน

Entity-Relationship Model

การออกแบบฐานข้อมูลใช้งานในระบบงานสารสนเทศใดๆ ต้องอาศัยแบบจำลองของข้อมูลเพื่อนำเสนอรายละเอียดต่าง ๆ ที่เกี่ยวข้องกับข้อมูลในฐานข้อมูลที่ออกแบบ แบบจำลองของข้อมูลที่นิยมใช้ ได้แก่ Entity-Relationship Model

Semantic Model

แบบจำลองของข้อมูลในยุคแรก ๆ มีข้อจำกัดในการนำเสนอรายละเอียดที่เกี่ยวข้องกับข้อมูล กล่าวคือ จะมีการนำเสนอเฉพาะรายละเอียดทางด้านโครงสร้างของฐานข้อมูลเช่น คุณสมบัติ Atomicity ของข้อมูล กฎระเบียบต่าง ๆ ที่ใช้สำหรับควบคุมความถูกต้องของข้อมูล เป็นต้น แต่ยังไม่มีการนำเสนอรายละเอียดทางด้านความหมาย (Semantic)ของข้อมูลภายในฐานข้อมูลนั้น ๆ เช่น จำนวน และน้ำหนักของสินค้าว่ามีความสัมพันธ์กันอย่างไร หรือ Domain ของข้อมูลต่าง ๆ สามารถมีค่าเป็นอะไรได้บ้าง หรือข้อมูลใดที่ทำหน้าที่เป็น Candidate Key หรือ Foreign Key เป็นต้น ดังนั้น จึงมีการคิดค้นแบบจำลองของข้อมูลใหม่ขึ้นที่เรียกว่า "Semantic Model"

คำศัพท์ที่ใช้ใน Semantic Model

ใน Semantic Model ได้มีการนิยามคำขึ้นแทนข้อมูลในความหมายต่าง ๆ ที่เรียกว่า Concept ดังนี้

Entity

Entity เปรียบเหมือนกับคำนามที่สามารถระบุได้ในความเป็นจริง อาจเป็นสิ่งที่จับต้องได้ เช่น นายไมเคิล หนังชื่อที่ชื่อ "ระบบฐานข้อมูล" หรือเป็นสิ่งที่อยู่ในรูปนามธรรมที่ไม่สามารถจับต้องได้ เช่น จำนวนวันลาพักร้อนของพนักงาน ซึ่งเมื่อนำแต่ละ Entity มารวมกันภายใต้คุณลักษณะใดลักษณะหนึ่งที่เหมือนกันแล้ว Entity เหล่านั้น จะถูกเรียกว่า "Entity Set" ดังตัวอย่างดังนี้

ตารางที่ 2.6 ตัวอย่าง Entity

Entity ของนาย ก.

นาย ก.	ไทย

Entity ของนาย ข.

นาย ข.	**ไทย**

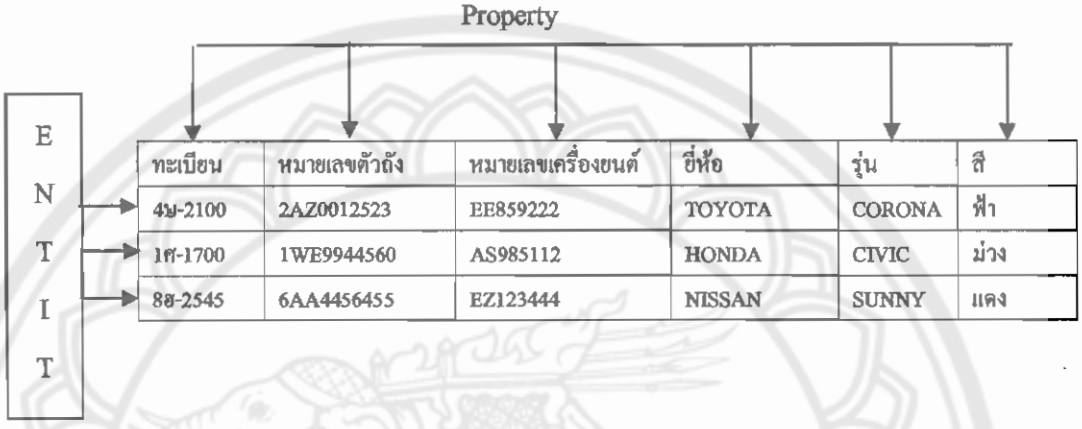
Entity Set "ประชากรสัญชาติไทย"

นาย ก.	**ไทย**
นาย ข.	**ไทย**

Property หรือ Attribute

Property หรือ Attribute คือข้อมูลที่แสดงลักษณะและคุณสมบัติของ Entity เช่น Property ของ Entity Set ที่ชื่อ "รถยนต์" ที่ประกอบด้วย หมายเลขทะเบียนรถยนต์ หมายเลขตัวถัง หมายเลขเครื่องยนต์ ยี่ห้อ รุ่น สี ดังรูป

ตารางที่ 2.7 ตัวอย่าง Property



ทะเบียน	หมายเลขตัวถัง	หมายเลขเครื่องยนต์	ยี่ห้อ	รุ่น	สี
4ษ-2100	2AZ0012523	EE859222	TOYOTA	CORONA	ฟ้า
1ศ-1700	1WE9944560	AS985112	HONDA	CIVIC	ม่วง
8ฮ-2545	6AA4456455	EZ123444	NISSAN	SUNNY	แดง

Identity

แต่ละ Entity ภายใต้ Entity Set เดียวกัน ถึงแม้ว่าจะต้องมี Property ที่เหมือนกันแต่อย่างไรก็ตาม จะต้องมี Property ใน Property หนึ่ง ซึ่งเป็นเอกลักษณ์เฉพาะของ Entity นั้นเช่น Property "หมายเลขประชาชน" ของแต่ละ Entity ใน Entity Set "ประชาชน" ซึ่งจะไม่มีหมายเลขใดที่ซ้ำกัน ดังตารางข้างล่าง

ตารางที่ 2.8 ตัวอย่าง Identity

Indentity

หมายเลขบัตรประชาชน	ชื่อ	นามสกุล	เพศ	สัญชาติ	วันเดือนปีเกิด
123456789	แดง	สด	ชาย	ไทย	12/12/22
987654321	ดำ	ดี	หญิง	ไทย	12/12/22

สำหรับ Property ที่สามารถนำมากำหนดเป็นเอกลักษณ์เฉพาะให้กับแต่ละ Entity นี้จะเรียกว่า "Identity"

Relationship

ได้แก่ Entity Set ที่สร้างขึ้นจาก 2 Entity Set เดิมหรือมากกว่า เพื่อใช้แสดงความสัมพันธ์ระหว่างแต่ละ Entity ใน Entity Set เดิมเหล่านั้นในการสร้าง Relationship อาจสร้างด้วยการนำเอา

แต่ละ Entity ใน Entity Set เดิมเหล่านั้นมาเชื่อมโยงข้อมูลกันภายใต้ค่าของ Property ที่เหมือนกัน ซึ่งการสร้างความสัมพันธ์ในลักษณะนี้ Property ของ Relationship จะเกิดจากการนำเอา Property ของแต่ละ Entity Set มารวมกัน เช่น Relationship ที่ชื่อ "สังกัดคณะ" ซึ่งเกิดจากการที่ Entity Set "นักศึกษา" และ "คณะ" มี Property "รหัสคณะ" ที่เหมือนกัน ดังรูป

ตารางที่ 2.9 ตัวอย่าง Relationship

Entity Set "นักศึกษา"

รหัสนักศึกษา	ชื่อ – สกุล	เพศ	รหัสคณะ
380012	เอก โท	ชาย	02
381202	ตรี จัตวา	หญิง	01
380052	เอก ตรี	ชาย	03

Entity Set "คณะ"

รหัสคณะ	คณะ
01	วิศวกรรมศาสตร์
02	บริหารธุรกิจ

Relationship "สังกัดคณะ"

รหัสนักศึกษา	ชื่อ – สกุล	เพศ	รหัสคณะ	คณะ
380012	เอก โท	ชาย	02	บริหารธุรกิจ
381202	ตรี จัตวา	หญิง	01	วิศวกรรมศาสตร์
380052	เอก ตรี	ชาย	03	วิทยาศาสตร์

Entity-Relationship model

Semantic Model ที่นิยมใช้มากที่สุดได้แก่ Entity-Relationship Model หรือที่นิยมเรียกกันสั้นๆว่า "E-R Model" E-R Model นับเป็นแบบจำลองที่ครอบคลุมนิยามต่างๆ ที่กำหนดไว้ใน Semantic Model เนื่องจากมีรูปแบบที่ใช้แทนทุกๆ แนวความคิดที่กำหนดไว้ Semantic Model ซึ่งได้แก่ Entity, Property, Relation และ Subtype สำหรับแผนภาพที่สร้างขึ้นโดยใช้รูปภาพต่าง ๆ ภายใน E-R Model เพื่อแสดงความเป็นจริงต่างๆ ของข้อมูลในฐานข้อมูล จะเรียกว่า Entity-Relationship Diagram หรือที่นิยมเรียกกันสั้น ๆ ว่าแผนภาพ E-R (E-RDiagram)

2.4 การทำงาน Normalization

การออกแบบฐานข้อมูลด้วย E-R Model มีจุดมุ่งหมายเพื่อนำเสนอข้อเท็จจริงต่างๆ ที่เกี่ยวข้องกับข้อมูล โดยไม่ได้คำนึงถึงว่า ฐานข้อมูลที่ออกมานั้น จะมีปัญหาทางด้านความซับซ้อนของข้อมูลความถูกต้องของข้อมูล ความผิดพลาดในการเพิ่ม ลบ และแก้ไข หรือไม่ ดังนั้น จึงต้องมีวิธีการตรวจสอบ และแก้ไขปัญหาต่างๆ เหล่านี้ วิธีการดังกล่าว "การทำ Normalization"

2.4.1 Normalization

เป็นวิธีการที่ใช้ในการตรวจสอบ และแก้ไขปัญหาทางด้านความซับซ้อนของข้อมูล โดยดำเนินการให้ข้อมูลในแต่ละ relation อยู่ในรูปที่เป็นหน่วยเล็กที่สุดที่ไม่สามารถแตกออกเป็นหน่วยย่อยๆ ได้อีก โดยยังคงความสัมพันธ์ระหว่างข้อมูลใน Relation ต่างๆ ไว้ตามหลักการที่กำหนด ไว้ใน Relation Model

การทำ Normalization นี้เป็นการดำเนินงานอย่างเป็นระดับ ที่กำหนดไว้ด้วยกัน เป็นขั้นตอน ตามปัญหาที่เกิดขึ้นในตอนขั้นๆ ซึ่งแต่ละขั้นตอนจะมีชื่อตามโครงสร้างข้อมูลกำหนดไว้ดังนี้

1. ขั้นตอนการทำ First Normal Form (1NF)
2. ขั้นตอนการทำ Second Normal Form (2NF)
3. ขั้นตอนการทำ Third Normal Form (3NF)
4. ขั้นตอนการทำ Boyce-Codd Normal Form (BCNF)
5. ขั้นตอนการทำ Fourth Normal Form (4NF)

ในแต่ละขั้นตอนของการทำ Normalization จะมีการระบุรูปแบบของโครงสร้างของข้อมูลที่ควรจะเรียกว่า Normal Form ไว้ ซึ่งโครงสร้างเหล่านี้จะสามารถแก้ไขปัญหาที่เกิดขึ้นในโครงสร้างที่ระบุนี้จะสามารถแก้ไขปัญหาที่เกิดขึ้นในโครงสร้างข้อมูลของขั้นตอนก่อนหน้าได้หรือกล่าวอีกนัยหนึ่ง การทำ Normalization ในขั้นตอน จะต้องอาศัยผลที่ได้จากการทำ Normalization ในขั้นตอนก่อนหน้า มาปรับปรุงเพื่อให้มีโครงสร้างเป็นไปตามโครงสร้างเป็นไปตามโครงสร้างที่กำหนดไว้ในขั้นตอนนั้นๆ แต่อย่างไรก็ตาม ในการทำ Normalization ไม่จำเป็นต้องเริ่มจากขั้นตอนการทำ First Normal form เสมอไป กล่าวคือ การทำ First Normal form และสิ้นสุดในขั้นตอนการทำ Fourth Normal form เสมอไป

2.4.2 First Normal Form(1NF)

เป็นขั้นตอนสำหรับปรับโครงสร้างของข้อมูลของ Relation เพื่อให้ทุก Attribute ของ Relation มีคุณสมบัติ Atomicity กล่าวคือ โครงสร้างข้อมูลของ Relation ในแบบ INF นี้จะต้องประกอบด้วย Attribute ที่ไม่อยู่ในรูป Repeating Group เช่น ตัวอย่างข้อมูลต่อไปนี้

ตารางที่ 2.10 ตัวอย่าง First Normal Form

ORDER

CUST-NO	CUST-NAME	CITY	ZONE-SALE	ORDER-CONTENT	
					ORDER-QTY
C001	นารี เกิดสว่าง	อยุธยา	001	P001	24
				P003	30
				P004	50
C002	สลักจิต สว่างภพ	ศรีสะเกษ	002	P001	29
				P002	40
				P004	30

จากตารางจะสังเกตเห็นว่า (Attribute "CUST_NO") 1 คนสามารถมีรายการสินค้าที่สั่งซื้อ (Attribute "CUST_NAME") ได้มากกว่า 1 รายการ ดังนั้นจึงกล่าวได้ว่า Attribute "CUST_NO" นี้มีความสัมพันธ์กับ Attribute "ORDER_CONTENT" ในแบบ Repeating Group ส่งผลให้ Relation นี้มีโครงสร้างที่ไม่สอดคล้องกับ 1NF ดังนั้นจึงต้องทำการ Normalization โดยการแปลง Attribute ที่อยู่ในรูป Repeating Group ให้มีคุณสมบัติ Atomicity พร้อมกับกำหนดให้ Attribute ดังกล่าวเป็น Relation Key ของ Relation ดังนั้น จากตัวอย่างข้างต้นจึงถูกแปลงให้อยู่ในรูปดังนี้

ตารางที่ 2.10 (ต่อ) ตัวอย่าง First Normal Form

ORDER

CUST_NO	CUST_NAME	CITY	ZONE_SALE	PRODUCT_ID	ORDER_QTY
C0001	นารี เกิดสว่าง	อยุธยา	001	P001	24
C0001	นารี เกิดสว่าง	อยุธยา	001	P003	30
C0001	นารี เกิดสว่าง	อยุธยา	004	P004	50
C0002	สลักจิต สว่างภพ	ศรีสะเกษ	001	P001	29

จะสังเกตเห็นว่า แต่ละค่าเป็น Repeating Group ของ Attribute "PRODUCT_ID" และ "ORDER_QTY" จะถูกแยกออกมา Tuple ใหม่ พร้อมกำหนดให้ Attribute "PRODUCT_ID" ทำหน้าที่ Relation Key ร่วมกับ Attribute "CUST_NO"

อย่างไรก็ตาม Relation ที่อยู่ในรูป 1NF ถึงแม้จะทำให้ทุก Attribute มีคุณสมบัติ Atomicity แต่กลับเกิดปัญหาความซ้ำซ้อนกันของข้อมูล (Redundacy) ขึ้นใน Attribute "CUST_NO"

"CUST_NAME","CITY" และ "ZONE_SALE" และก่อให้เกิดปัญหาทางด้าน Anomaly ตามมา ดังนี้

1. **Insert Anomaly**

เมื่อพิจารณาจากตัวอย่าง จะสังเกตเห็นว่าการเพิ่มข้อมูลลูกค้า จะทำได้ก็ต่อเมื่อลูกค้านั้นมีรายการสั่งซื้อสินค้าแล้วเท่านั้น หมายถึงว่า Relation นี้จะไม่สามารถจัดเก็บข้อมูลของลูกค้าที่ยังไม่มีการสั่งซื้อสินค้าได้ เช่น เมื่อต้องการเพิ่มขึ้นลูกค้ารหัส "C006" โดยที่ยังไม่มีการกำหนดการสั่งซื้อสินค้าใน Attribute "PROSUCT_ID" จะไม่สามารถกระทำได้ เนื่องจากการ Attribute "PRODUCT_ID" นี้ถูกกำหนดให้เป็น Relation Key จึงไม่สอดคล้องตามกฎของ Entity Integrity Rule ในส่วนที่ว่า Attribute หรือกลุ่ม Attribute ที่เป็น Relation Key จะมีค่าเป็น Null ไม่ได้

2. **Delete Anomaly**

เมื่อพิจารณาจากตัวอย่าง จะสังเกตเห็นว่า การลบข้อมูลรายการสั่งซื้อ (Attribute "PRODUCT_ID" และ "ORDER_QTY" บาง Tuple ใน Relation นี้จะทำให้ข้อมูลลูกค้าบางคนสูญหายไป เช่น เมื่อลบข้อมูลรายการสั่งซื้อของสินค้ารหัส "P005" ของลูกค้ารหัส "C003" จึงทำให้ข้อมูลลูกค้ารหัส "C003" ถูกลบตามไปด้วย

3. **Update Anomaly**

เมื่อพิจารณาตัวอย่าง จะสังเกตเห็นว่า การปรับปรุงข้อมูลใน Tuple ที่มีค่าของข้อมูลซ้ำซ้อนกันไม่ครบถ้วน อาจทำให้เกิดความขัดแย้งของข้อมูลที่ซ้ำซ้อนกันนั้นได้ เช่น การเปลี่ยนชื่อจาก "สลักจิต สว่างภาพ" เป็น "รินลณี สว่างภพ" ของลูกค้ารหัส "C002" ซึ่งถ้าแก้ไขไม่ครบแล้ว จะทำให้ ลูกค้ารหัส "C002" มีชื่อทั้ง "สลักจิต สว่างภาพ" และ รินลณี สว่างภพ"

2.4.3 Second Normal Form(2NF)

ในการทำ Normalization ในขั้นตอน Second Normal Form จำเป็นต้องรู้จักถึง Prime Attribute และ Nonprime Attribute เนื่องจาก Attribute ทั้ง 2 ประเภทนี้ จะมีความสำคัญต่อการทำ Normalization แบบ Second Normal Form

Prime Attribute ได้แก่ ทุก Attribute ที่ทำหน้าที่เป็น Relation Key ของ Relation ส่วน Nonprime Attribute ได้แก่ Attribute ที่ไม่ได้เป็นส่วนหนึ่งของ Relation Key

1. ต้องมีโครงสร้างเป็นไปตามโครงสร้างของ 1NF
2. ทุก Nonprime Attribute จะต้องไม่ขึ้นกับ Relation Key ที่อยู่ในรูป Subset

ตัวอย่างเช่น Relation "ORDER1" ในตัวอย่าง ที่ผ่านมา ซึ่งเป็น Relation ที่มีคุณสมบัติของ 1NF จะสังเกตเห็นว่า Attribute (Cust No, Product ID) เป็น Attribute ที่ทำให้ข้อมูลในแต่ละ Tuple มีค่าไม่

ซ้ำกัน ดังนั้น Attribute ทั้ง 2 จึงทำหน้าที่เป็น Relation Key ซึ่งสามารถเขียนได้ด้วย Functional Dependency ได้ดังนี้

FD : CUST_NO,PRODUCT_ID ⟶
CUST NAME.CITY.ZONE SALE,ORDER O

เมื่อพิจารณาค่าของAttribute"CUST_NO", "CUST_NAME", "CITY","ZONE_SALE" จะสังเกตว่า Tuple ที่ประกอบขึ้นจาก Attribute เหล่านี้ จะมีข้อมูลกันซ้ำกันเป็นชุดๆ และมีเพียง Attribute "ORDER_QTY" เท่านั้น ที่มีค่าเปลี่ยนตามค่าของ Relation Key ดังนั้นจึงสามารถเขียนด้วย Functional Dependency ได้ดังนี้

d1 : CUST_NO,PRODUCT_ID ⟶ ORDER_QTY
d2 : CUST_NO ⟶ CUST_NAME,CITY,ZONE_SALE

ใน d2 จะสังเกตเห็นว่า Attribute "CUST_NAME","CITY", และ "ZONE_SALE" เป็น Nonprime Attribute ของ Relation ที่ไม่ได้ขึ้นอยู่กับเฉพาะ Relation "ORDER1" จึงไม่มีคุณสมบัติเป็นไปตามคุณสมบัติของ 2NF จึงต้องแตก Relation "ORDER1" ออกเป็น 2 Relation ตาม d1 และ d2 ดังนี้

ตารางที่ 2.11 ตัวอย่าง Second Normal Form

CustOrder

CUST_NO	PRODUCT_ID	ORDER_QTY
C001	P001	24
C001	P003	30
C001	P004	50
C002	P001	29

Cust

CUST_NO	CUST_NAME	CITY	ZONE_SALE
C001	นารี เกิดสว่าง	อยุธยา	001
C002	สลักจิต สว่างภพ	ศรีสะเกษ	002
C003	สุทิศา เจ็กสกุล	เชียงใหม่	004
C004	ฟ้า เพิ่มพร	ศรีสะเกษ	002
C009	ต้นสาย ตันเจริญ	เชียงใหม่	004

สำหรับโครงสร้างของ Relation "CustOrder" และ "Cust" นี้จะสังเกตเห็นว่าสามารถแก้ปัญหา Anomaly ที่เกิดขึ้นใน Relation "ORDER1" ได้ดังนี้

- Relation "Cust" สามารถเพิ่มข้อมูลค้าได้โดยไม่ต้องมีการสั่งซื้อสินค้าที่เกิดขึ้น เนื่องจากข้อมูลการสั่งซื้อสินค้าจะถูกแยกจัดเก็บใน Relation "CustOder" จึงสามารถแก้ปัญหา Insert Anomaly ที่เกิดขึ้นได้
- สามารถลบข้อมูลรายการสั่งซื้อสินค้ารหัส "P005" ของลูกค้ารหัส "C003" ใน Relation "CustOrder" ได้โดยไม่ส่งผลต่อข้อมูลของลูกค้ารหัส "C003" เนื่องจากข้อมูลลูกค้าถูกแยกจัดเก็บใน Relation "Cust" จึงสามารถแก้ปัญหา Delecte Anomaly ที่เกิดขึ้นได้
- เนื่องจากข้อมูลของลูกค้าที่ซ้ำซ้อนจะถูกแยกมาจัดเก็บใน Relation "Cust" ดังนั้น การเปลี่ยนแปลงรายละเอียดข้อมูลลูกค้าจึงกระทำกับ Relation "Cust" เพียง Relation เดียว จึงก่อให้เกิดปัญหา Update Anomaly

2.4.4 Third Normal Form(3NF)

สำหรับ Relation ที่จะมีโครงสร้างในแบบ 3NF จะต้องมีคุณสมบัติดังนี้

1. ต้องมีคุณสมบัติของ 2NF

2. ต้องไม่มี Functional Dependency เกิดขึ้นระหว่าง Nonprime Attribute ด้วยกันเองที่เรียกว่า "Transitive Dependency" จาก Relation "Cust" ในหัวข้อที่ผ่านมา ถึงแม้ว่าจะมีโครงสร้างเป็นไปตามคุณสมบัติของ 2NF แต่จะสังเกตเห็นว่า ค่าของ Attribute "CITY" และ "ZONE_SALE" ถ้าปรากฏข้อมูลที่มีค่าซ้ำซ้อนกันอยู่เป็นคู่ ๆ หรือกล่าวอีกนัยหนึ่งทั้ง 2 Attribute สามารถที่จะระบุค่าระหว่างกันได้ กล่าว คือ เมื่อระบุค่าให้กับ Attribute "ZONE_SALE" จะสามารถทราบถึงชื่อเมืองใน Attribute "CITY" ได้ ซึ่งความสัมพันธ์ในลักษณะนี้ จะเรียกว่า Transitive Dependency ดังนั้น Relation นี้จึงขาดคุณสมบัติของ 3NF และยังก่อให้เกิดปัญหาความผิดพลาดทางด้าน Anomaly ดังนี้

1. **Insert Anomaly**

ใน Relation "Cust" เมื่อต้องการเพิ่มข้อมูลให้กับ Attribute "CITY" และ "ZONE_SALE" ซึ่งมีความสัมพันธ์ในแบบ Transitive Dependency จะไม่สามารถกระทำได้เนื่องจากข้อมูลใน 2 Attribute นี้ ไม่ใช่ Relation Key ดังนั้น จึงต้องเพิ่มข้อมูลนี้ของลูกค้ารายใหม่ให้กับ Attribute "CUST_NO" และ "CUST_NAME" ตามไปด้วย

2. Update anomaly

ใน Relation "Cust" จะสังเกตเห็นว่า เมื่อมีการแก้ไขข้อมูลใน Attribute "ZONE_SALE" เท่ากับ "004" ให้ครบถ้วน เนื่องจากเมื่อทำการแก้ไขข้อมูลไม่ครบถ้วนจะก่อให้เกิดข้อมูลใน Relation ที่มีความขัดแย้งกันได้

3. Delete Anomaly

ใน Relation "Cust" จะสังเกตเห็นว่า ถ้ามีการลบข้อมูลของ Tuple ที่จัดเก็บข้อมูลในกลุ่ม Transitive Dependency ที่ปรากฏอยู่เพียงชุดเดียวใน Relation จะส่งผลให้ข้อมูลในกลุ่ม Transitive Dependency นั้นสูญหายไปจาก Relation ได้เช่น เมื่อทำการลบข้อมูลใน Tuple ของลูกค้ารหัส "C001" นอกจากจะทำให้ข้อมูลของลูกค้ารหัส "C001" หายไปแล้ว ยังส่งผลให้ข้อมูลของเขตการขายที่อยุธยาสูญหายไปด้วย

จากปัญหา Anomaly ที่เกิดขึ้นจาก Transitive Dependency เหล่านี้ จึงต้องทำการแยก Nonprime Attribute ที่ก่อให้เกิด Transitive Dependency ของ Relation "Cust" ออกมาเป็น Relation ใหม่ ซึ่งจากตัวอย่าง ได้แก่ Attribute "CITY" และ "ZONE_SALE" ดังนี้

ตารางที่ 2.12 ตัวอย่าง Third Normal Form

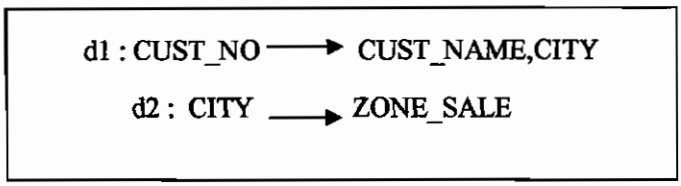
Cust2

CUST_NO	CUST_NAME	CITY
C001	นารี เกิดสว่าง	อยุธยา
C002	สลักจิต สว่างภพ	ศรีสะเกษ
CUST_NO	CUST_NAME	CITY
C003	สุทิศา เจ็กสกุล	เชียงใหม่
C004	ฟ้า เพิ่มพร	ศรีสะเกษ

CityZone

CITY	ZONE_SALE
อยุธยา	001
ศรีสะเกษ	002
เชียงใหม่	004

ซึ่งสามารถเขียนด้วย Functional Dependency ได้ดังนี้



ข้อสังเกต ในการแยก Nonprime Attribute ที่ก่อให้เกิด Transitive Dependency ออกมาเป็น Relation ใหม่ มีหลักการอยู่ 2 ข้อดังนี้

1. แยก Nonprime Attribute ที่ก่อให้เกิด Transitive Dependency ออกเป็น Relation ใหม่
2. กำหนดให้ Nonprime Attribute ที่เป็นตัวระบบค่า ให้เป็น Relation Key ของ Relation ใหม่

2.4.5 Fourth Normal Form(4NF)

สำหรับ Relation ที่จะมีโครงสร้างแบบ 4NF จะต้องมีคุณสมบัติดังนี้

1. ต้องมีคุณสมบัติของ BCNF
2. ต้องไม่ปรากฏความสัมพันธ์ระหว่าง Attribute ในแบบ Multi-value Dependency เช่น ตัวอย่างข้อมูลของ Relation "EMPOLYEE_SKILL" ซึ่งใช้จัดเก็บข้อมูลเกี่ยวกับความสามารถของพนักงาน ทางด้านการใช้คอมพิวเตอร์ (Attribute "COMPUTER_SKILL") และทางด้านภาษาต่างประเทศ (Attribute "LANGUAGE_SKILL") ดังนี้

ตารางที่ 2.13 ตัวอย่าง Forth Normal Form

EMPLOYEE_SKILL

EMPLOYEE_SKILL	COMPUTER_SKILL	LANGUAGE_SKILL
1267	Word Processing	ฝรั่งเศส
1267	Word Processing	เยอรมัน
345	Spreadsheets	สเปน
1267	Spreadsheets	ฝรั่งเศส
1267	Spreadsheets	เยอรมัน
1345	COBOL	สเปน
1193	Word Processing	ฝรั่งเศส

จากตัวอย่างข้อมูล จะสังเกตเห็นว่า Relation นี้ มีคุณสมบัติเป็น BCNF แต่ยังไม่เป็น 4NF เนื่องจาก ปรากฏโครงสร้างข้อมูลในแบบ Multi-value Dependency กล่าวคือ เมื่อระบุค่าของ

Attribute "EMPLOYEE#" ซึ่งทำหน้าที่เป็น Determinant จะสามารถแสดงค่าของ Attribule "COMPUTER_LSKILL" และ "LANGUGE_SKILL" ที่ทำหน้าที่เป็น Dependency ได้มากกว่า 1 ค่า

ดังนั้น จึงต้องแบ่ง Relation นี้ออกเป็น Relation ใหม่ตามโครงสร้างข้อมูลแบบ Multi-value Dependency ที่ปรากฏอยู่ ซึ่งได้แก่ Attribule "COMPUTER_LSKILL" และ "LANGUGE_SKILL" ดังนี้

ตารางที่ 2.14 ตัวอย่าง Fourth Normal Form

COMPUTER_SKILL

EMPLOYEE#	COMPUTER_SKILL
1267	Word Processing
1345	Spradsheets
1267	Speadsheets
1345	COBOL
1193	Word Processing

ตารางที่ 2.14 (ต่อ) ตัวอย่าง Fourth Normal Form

LANGUAGE_SKILL

EMPLOYEE#	LANGUAGE_SKILL
1267	ฝรั่ง
1267	เยอรมัน

2.5 สรุป

โครงสร้างของฐานข้อมูลที่ออกแบบขึ้น ควรที่จะนำมาปรับปรุงโดยใช้วิธีของการทำ Normalization เพื่อปรับเปลี่ยนโครงสร้างของ Relation ต่างๆ ที่ได้ออกแบบไว้ ให้มีโครงสร้างที่เป็นไปตามคุณสมบัติของ Relation ที่กำหนดไว้ใน Relation Model รวมทั้งมีโครงสร้างที่เหมาะสมต่อการนำไปใช้งานในการทำ Normalization จะมีอยู่ด้วยกัน 5 ขั้นตอน แต่ส่วนใหญ่ในทางปฏิบัติแล้วการทำ Normalization จะกระทำถึงขั้นตอน Boyce-Codd Normal Form (BCNF) เท่านั้นก็เพียงพอแล้ว

บทที่ 3

การออกแบบโปรแกรมฐานข้อมูลฝ่ายเจ้าหน้าที่วัสดุครุภัณฑ์

3.1 ขั้นตอนการดำเนินงาน

1. ศึกษาทฤษฎีและหลักการพื้นฐานในการทำ Database การเขียน ER- Diagram การใช้ DFD และการเขียนคำสั่งเพื่อการทำงาน My SQLรวบรวมข้อมูลเกี่ยวกับวัสดุครุภัณฑ์ต่าง ๆ และข้อมูลที่เกี่ยวข้อง เช่น การแบ่งวัสดุครุภัณฑ์ การคืน การยืม สถานะวัสดุครุภัณฑ์

2. สำรวจและรวบรวมความต้องการของผู้ใช้ (Requirement) ซึ่งมีดังนี้

   2.1 ความต้องการข้อมูลของสต๊อกวัสดุครุภัณฑ์ ได้แก่

   2.1.1 ความต้องการในการยืมและคืนวัสดุครุภัณฑ์ วันที่ลงทะเบียน วันที่กำหนดคืนวัสดุครุภัณฑ์ ราคาซื้อ ชนิดของวัสดุครุภัณฑ์ รหัสวัสดุครุภัณฑ์ สถานะของวัสดุครุภัณฑ์

   2.1.2 ความต้องการด้านการจัดสต๊อกวัสดุครุภัณฑ์ คือ ถ้าวัสดุครุภัณฑ์ในสต๊อกเกิดการสูญหายหรือเสียหายชำรุด จะฟ้องการสั่งเพิ่ม Order ได้ไหม และจะลบวัสดุครุภัณฑ์ได้ยังไง ถ้าวัสดุครุภัณฑ์นั้นไม่ต้องการหรือว่าไม่ใช้แล้ว

   2.1.3 การกำหนดหมายการรับ – คืน วัสดุครุภัณฑ์

   2.2 ความต้องการด้านบุคลากรที่เกี่ยวข้องกับเจ้าหน้าที่รับผิดชอบฝ่ายวัสดุครุภัณฑ์

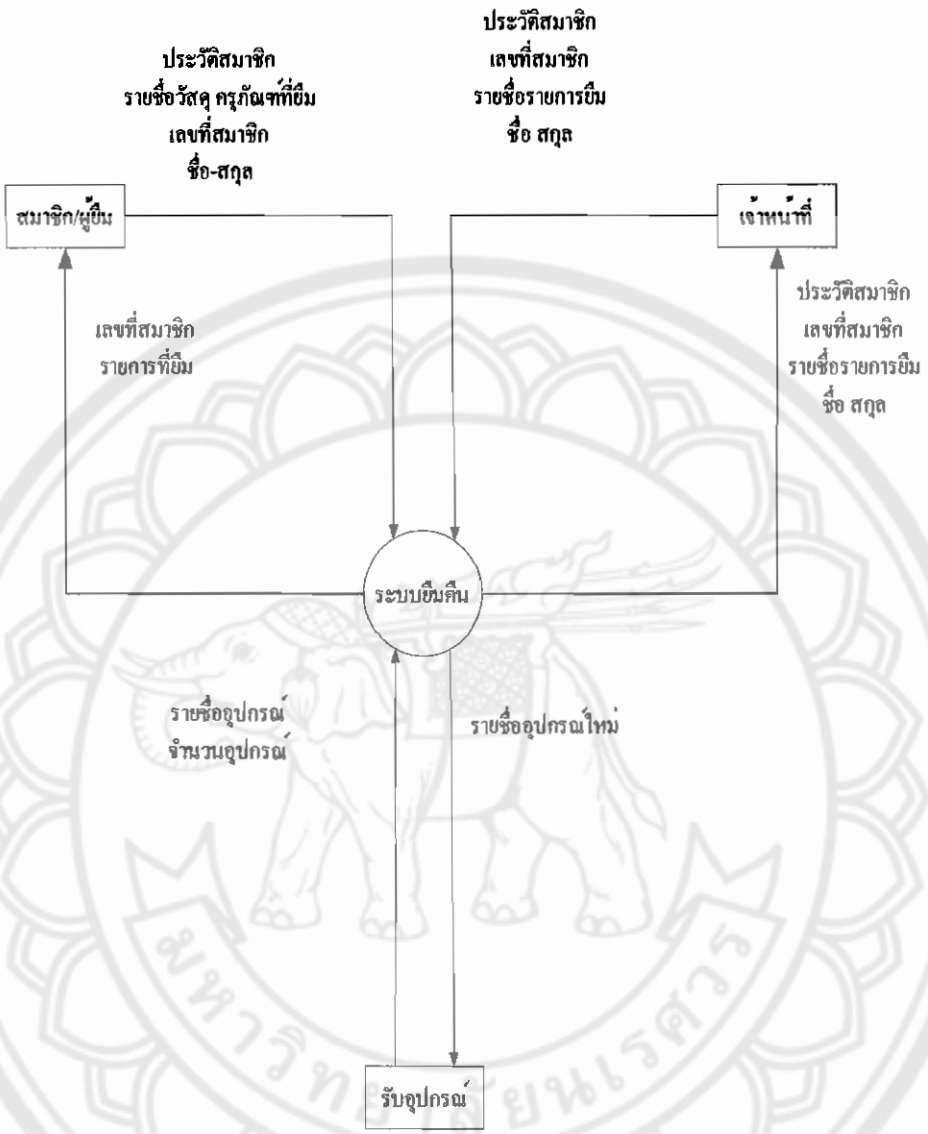
   2.2.1 ต้องการประวัติของบุคลากร เช่น เจ้าหน้าที่ที่ดูแลวัสดุครุภัณฑ์ เป็นต้น

   2.2.2 ความต้องการด้านข้อมูลของผู้ใช้วัสดุครุภัณฑ์

   2.3 สามารถเรียกดูข้อมูลวัสดุครุภัณฑ์ทั้งหมดผู้ยืม ในกรณีต้องการใช้ทั้งใหม่และเก่าได้ หรือค้นหาชื่อวัสดุครุภัณฑ์และประเภทวัสดุครุภัณฑ์

3. นำข้อมูลที่ได้ทั้งหมดมารวบรวมและวิเคราะห์เพื่อดู ส่วนข้อมูลเข้าและ ส่วนแสดงผลของการทำโปรแกรม และการเขียน DFD รวมทั้ง Context Diagram ได้

แผนภาพ Context Diagram

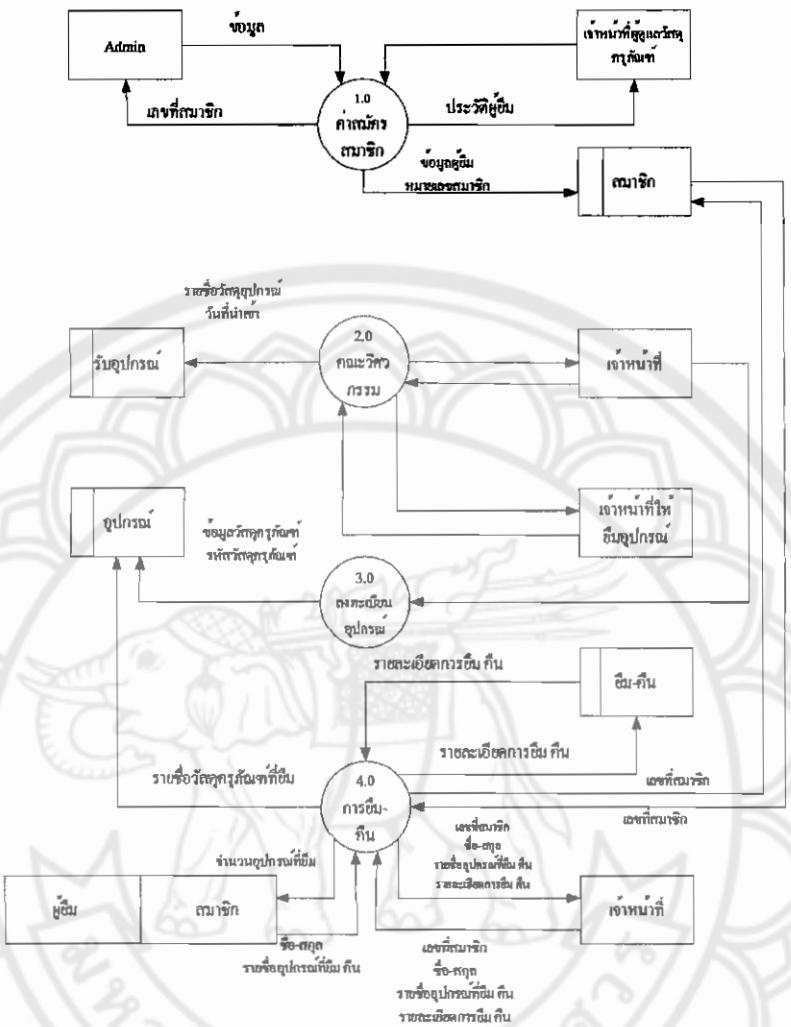


รูปที่ 3.1 แผนภาพ Context Diagram

จากแผนภาพเราจะเห็นได้ว่ามีส่วนประกอบของระบบยืมคืนที่มีข้อมูลของ

1. สมาชิก/ผู้ยืม ที่จะส่ง ประวัติสมาชิก , รายชื่อวัสดุครุภัณฑ์ที่ยืม , เลขที่สมาชิก , ชื่อ - สกุล เข้าสู่ระบบ
2. **เจ้าหน้าที่ ที่จะส่ง ประวัติสมาชิก , รายชื่อวัสดุครุภัณฑ์ที่ยืม , เลขที่สมาชิก , ชื่อ - สกุล เข้าสู่ระบบ**
3. **รับอุปกรณ์ ที่จะส่ง รายชื่ออุปกรณ์ , จำนวนอุปกรณ์ เข้าสู่ระบบ**

แผนภาพ Data Flow Diagram Level 0



รูปที่ 3.2 แผนภาพ DFD Level 0

จะเห็นได้ว่าขั้นตอนที่ 4 จะมองภาพรวมของทั้งระบบได้อย่างคร่าว ๆ คือได้ Input และ Output นั่นเอง

จากแผนภาพการไหลของ Level นี้ เราจะแสดงรายละเอียดของกระบวนที่ 0 ใน Level 0 ออกเป็น 4 ส่วนประกอบด้วย

1. การสมัครสมาชิก
2. **คณะวิศวกรรมศาสตร์**
3. **ลงทะเบียนอุปกรณ์**
4. **การยืม / การคืน**

จากแผนภาพประกอบไปด้วย

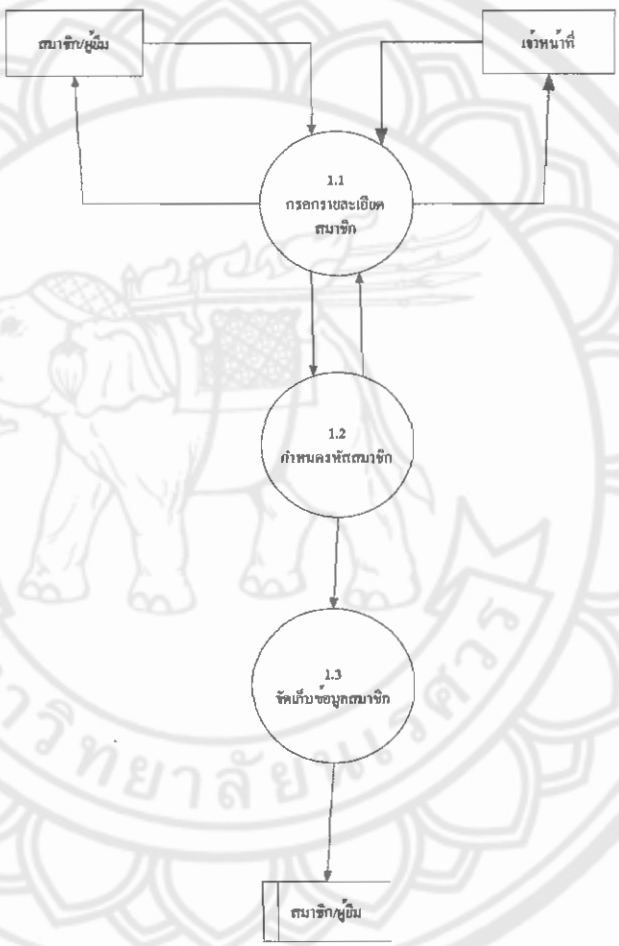
Process 1.0 ระบบการสมัครสมาชิก จะเก็บตัวข้อมูล Admin เจ้าหน้าที่ผู้ดูแลวัสดุครุภัณฑ์ และแฟ้มข้อมูลสมาชิก

Process 2.0 ระบบคณะวิศวกรรมศาสตร์ จะเก็บตัวข้อมูลของเจ้าหน้าที่, เจ้าหน้าที่ที่ให้ยืม, วัสดุครุภัณฑ์ และ แฟ้มข้อมูลการรับอุปกรณ์

Process 3.0 ระบบลงทะเบียนอุปกรณ์ จะเก็บตัวข้อมูลของแฟ้มอุปกรณ์ และ เจ้าหน้าที่

Process 4.0 ระบบการยืมคืน จะเก็บข้อมูลของสมาชิกผู้ยืม, เจ้าหน้าที่ , แฟ้มข้อมูลสมาชิก, แฟ้มข้อมูลการยืมคืน และ แฟ้มข้อมูลอุปกรณ์

แผนภาพ Data Flow Diagram Level 1 Process 1.0



รูปที่3.3 DFD Level 1 Process 1.0

จากแผนภาพประกอบไปด้วย

Process 1.1 กรอกลายละเอียดสมาชิก จะเก็บข้อมูลของสมาชิกผู้ยืม, เจ้าหน้าที่

Process 1.2 กำหนดรหัสสมาชิก รับข้อมูลจาก Process 1.1

Process 1.3 จัดเก็บข้อมูลสมาชิก รับข้อมูลจาก Process 1.2 และส่งไปยังแฟ้มข้อมูลสมาชิกผู้ยืม

แผนภาพ Data Flow Diagram Level 1 Process 2.0

รูปที่3.4 DFD Level 1 Process 2.0

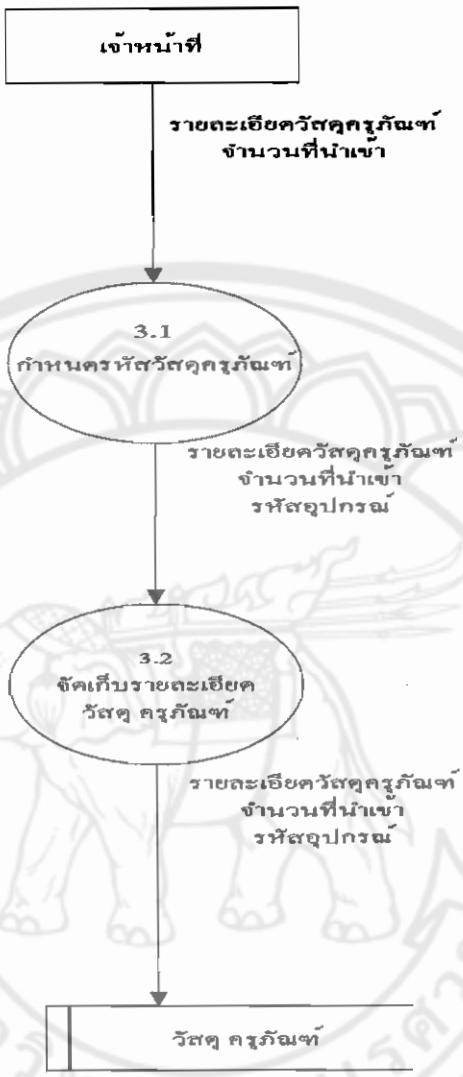
จากแผนภาพการไหลประกอบด้วย

Process 2.1 อุปกรณ์ใหม่ จะเก็บข้อมูลของเจ้าหน้าที่, รายชื่ออุปกรณ์ใหม่

Process 2.2 ตรวจรับอุปกรณ์ จะเก็บข้อมูลของเจ้าหน้าที่, รายชื่ออุปกรณ์ที่ซื้อ

Process 2.3 ตรวจรับข้อมูล จะเก็บข้อมูลจาก Process 2.2 ที่ส่งรายชื่ออุปกรณ์ที่ซื้อ, จำนวนที่ซื้อ วันที่นำเข้า และ ส่งข้อมูลไปยังแฟ้มข้อมูลการจัดเก็บข้อมูล

แผนภาพ Data Flow Diagram Level 1 Process 3.0



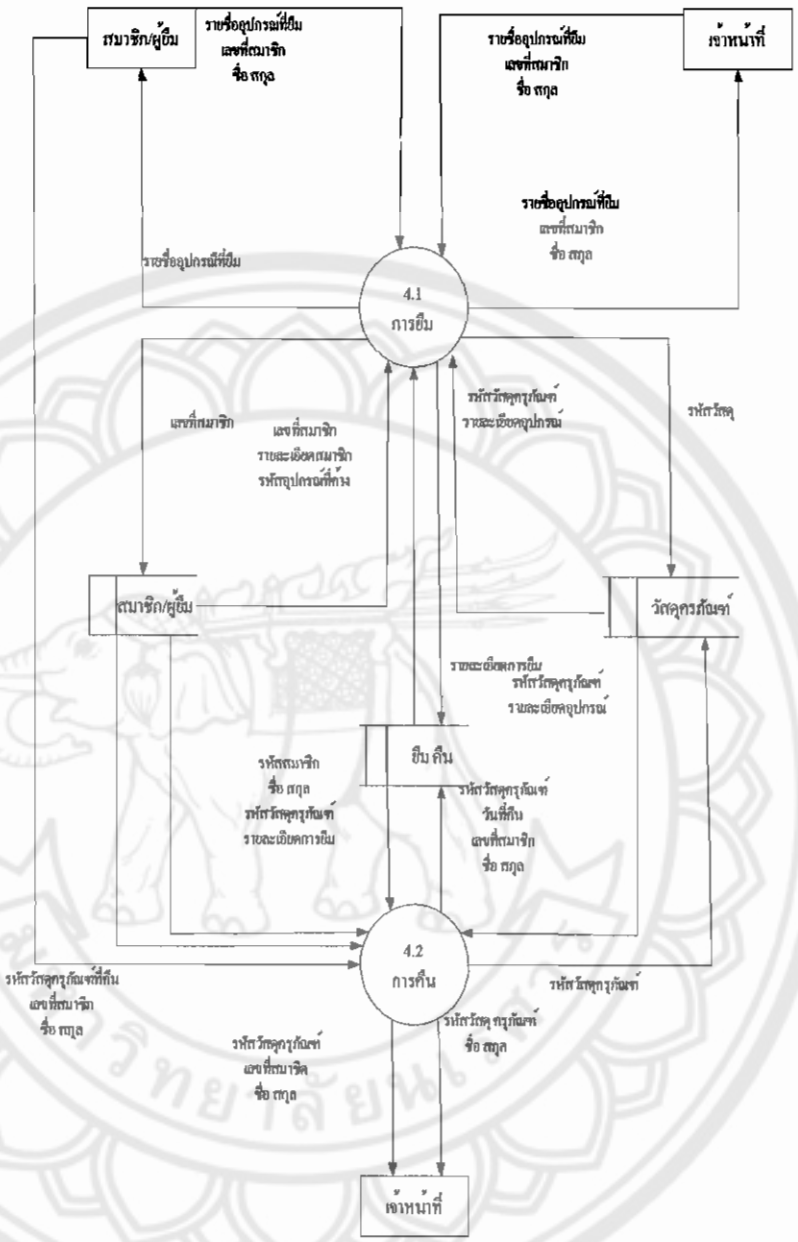
รูปที่3.5 DFD Level 1 Process 3.0

จากแผนภาพการไหลประกอบด้วย

Process 3.1 กำหนดรหัสวัสดุครุภัณฑ์ จะรับข้อมูลและเก็บข้อมูลจากเจ้าหน้าที่ที่จะส่งรายละเอียด**วัสดุครุภัณฑ์ จำนวนที่นำเข้า**

Process 3.2 จัดเก็บรายละเอียดวัสดุครุภัณฑ์ รับข้อมูลจาก Process 3.1 ที่จะส่งข้อมูลรายละเอียดวัสดุครุภัณฑ์, จำนวนที่เข้า และ รหัสวัสดุครุภัณฑ์ และ ส่งไปยังแฟ้มข้อมูลของวัสดุครุภัณฑ์

แผนภาพ Data Flow Diagram Level 1 Process 4.0



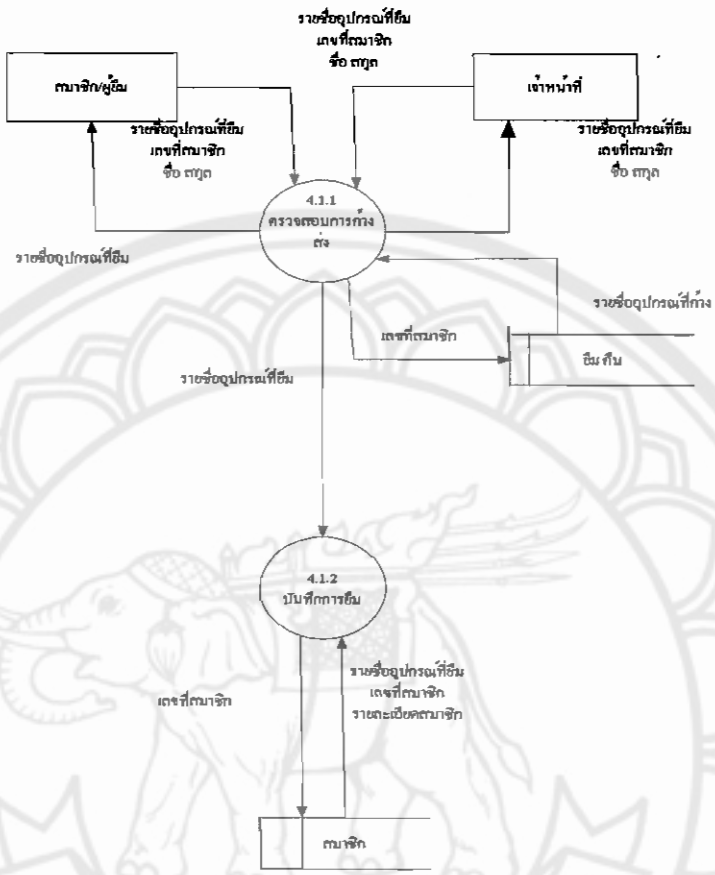
รูปที่3.6 DFD Level 1 Process 4.0

จากแผนภาพการไหลประกอบด้วย

Process 4.1 การยืม จะเก็บข้อมูลและรับข้อมูลจากสมาชิก/ผู้ยืม, เจ้าหน้าที่, แฟ้มข้อมูลของสมาชิกผู้ยืม/คืน และ แฟ้มข้อมูลของวัสดุครุภัณฑ์

Process 4.2 การคืน จะเก็บข้อมูลและรับส่งข้อมูลจากสมาชิก/ผู้ยืม, แฟ้มข้อมูลของสมาชิก/ผู้ยืม และ แฟ้มข้อมูลของวัสดุครุภัณฑ์ จากนั้นจะส่งข้อมูลต่อไปยังเจ้าหน้าที่

แผนภาพ Data Flow Diagram Level 2 Process 4.1



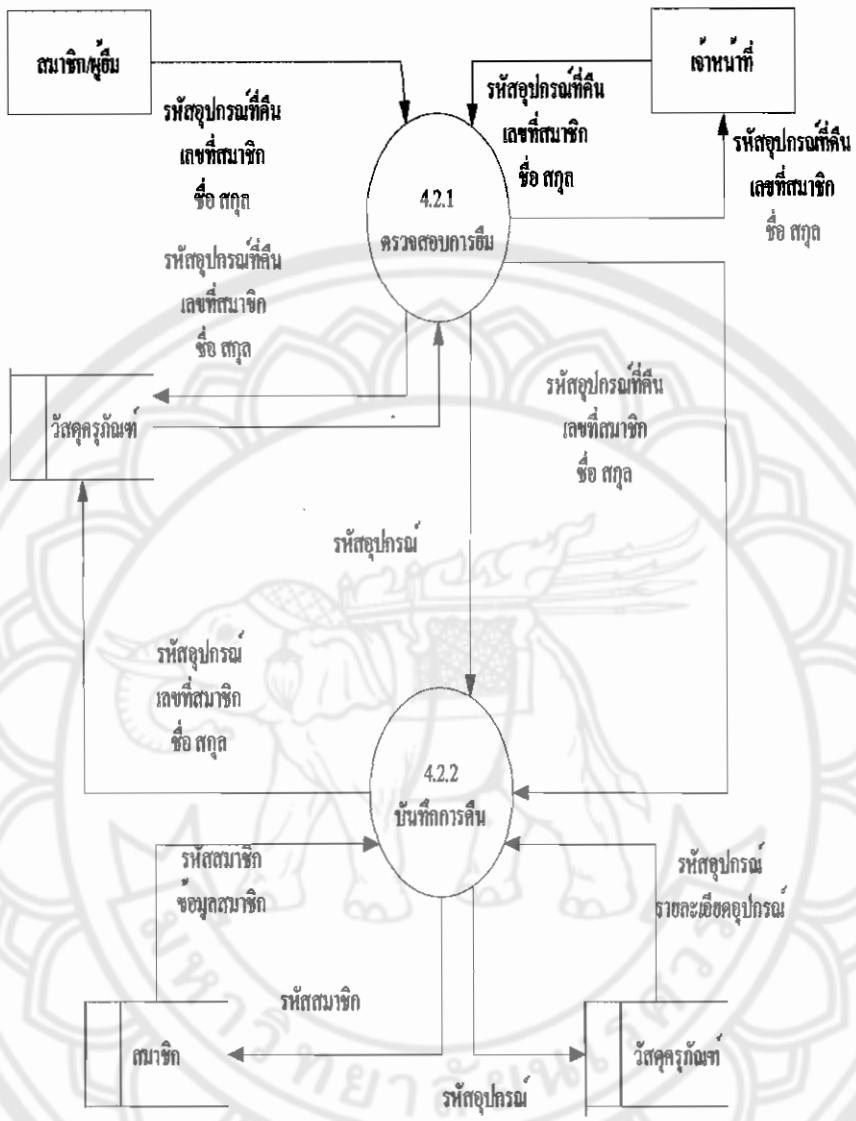
รูปที่3.7 DFD Level 2 Process 4.1

จากแผนภาพการไหลประกอบด้วย

Process 4.1.1 ตรวจสอบการค้างส่ง จะเก็บข้อมูลและรับส่งข้อมูลจากสมาชิก/ผู้ยืม, เจ้าหน้าที่ และแฟ้มข้อมูลของการยืมคืน

Process 4.1.2 บันทึกการยืม จะรับข้อมูลจาก Process 4.1.1 ที่จะส่งรายชื่ออุปกรณ์ที่ยืมมาให้ แล้วส่งข้อมูลเลขที่สมาชิกส่งไปยัง แฟ้มข้อมูลของสมาชิก

แผนภาพ Data Flow Diagram Level 1 Process 4.2



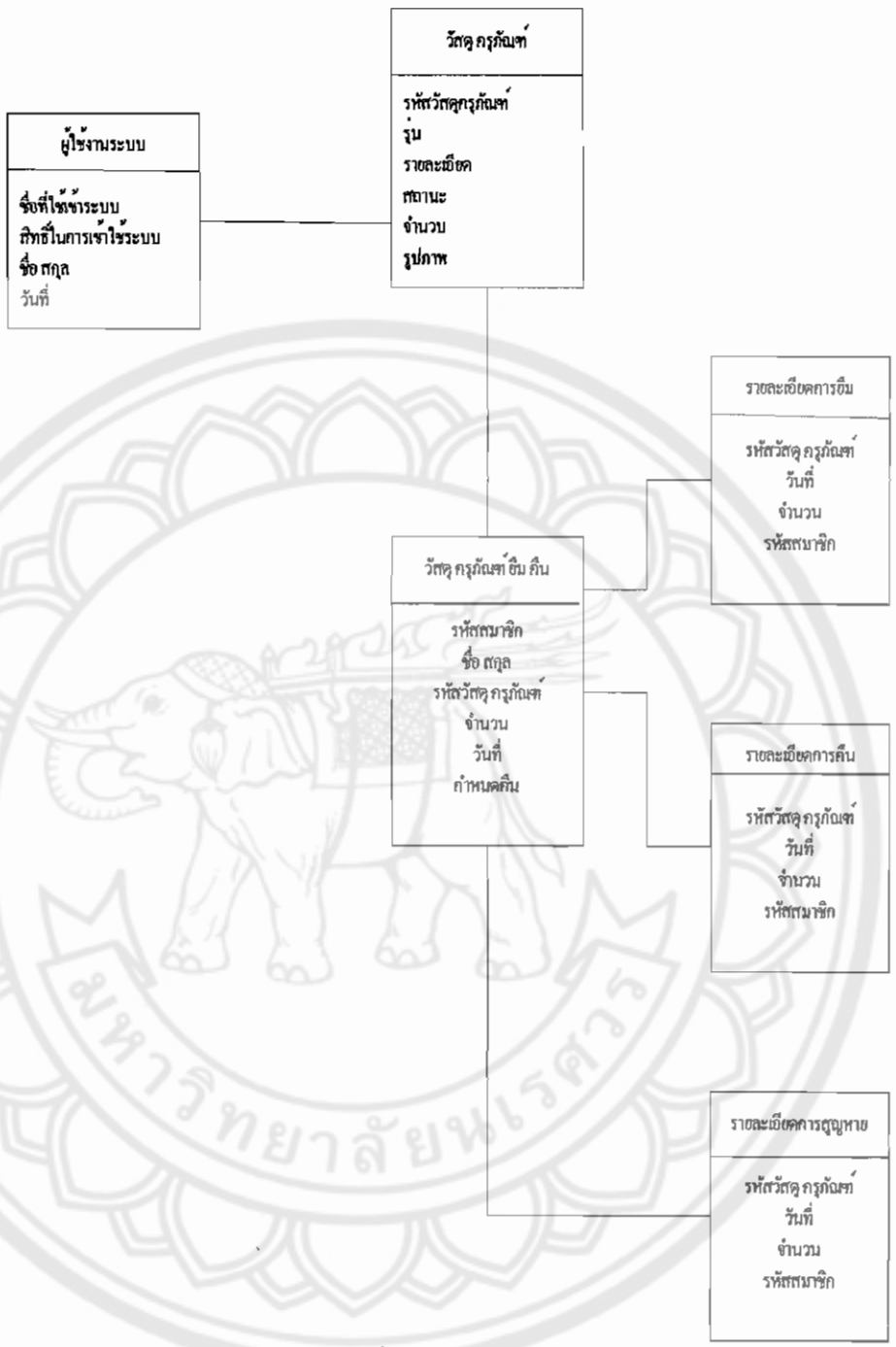
รูปที่3.8 DFD Level 2 Process 4.2

จากแผนภาพการไหลประกอบด้วย

Process 4.2.1 ตรวจสอบการยืม จะเก็บข้อมูลและรับส่งข้อมูลจากสมาชิก/ผู้ยืม, เจ้าหน้าที่ และแฟ้มข้อมูลวัสดุครุภัณฑ์

Process 4.2.2 บันทึกการคืนจะเก็บข้อมูลและรับส่งข้อมูลจาก Process 4.2.1 ที่ส่งรหัสอุปกรณ์ที่คืน, เลขที่สมาชิก, ชื่อสกุล มาให้ และ ยังรับข้อมูลมาจาก แฟ้มข้อมูลของสมาชิก และแฟ้มข้อมูลวัสดุครุภัณฑ์

ER Diagram



รูปที่3.9 ER Diagram

5. ออกแบบ ของโปรแกรมการจัดการฐานข้อมูลฝ่ายเจ้าหน้าที่วัสดุครุภัณฑ์
6. ฐานข้อมูลโปรแกรมการจัดการฐานข้อมูลฝ่ายเจ้าหน้าที่วัสดุครุภัณฑ์

ตารางที่3.1ฐานข้อมูลผู้ใช้งานระบบ

6.1 ฐานข้อมูลผู้ใช้งานระบบ

ชื่อที่ใช้เข้าระบบ	รหัสผ่าน	สิทธิ์ในการเข้าใช้ระบบงาน

ประกอบไปด้วย

- ชื่อที่ใช้เข้าระบบ
- รหัสผ่าน
- สิทธิ์ในการเข้าใช้ระบบ

6.2 ฐานข้อมูลรายการครุภัณฑ์

เลขที่สมาชิก	ประเภท	รุ่น	รหัสวัสดุครุภัณฑ์	รายละเอียด	สถานที่จัดเก็บ
จำนวนที่มี	รูปภาพ	หมายเหตุ	ชื่อ	ราคา	สถานะ

ประกอบไปด้วย

- เลขที่สมาชิก
- ประเภท
- รุ่น
- รหัสวัสดุครุภัณฑ์
- รายละเอียดวัสดุครุภัณฑ์
- สถานที่จัดเก็บ
- จำนวนที่มี
- รูปภาพ
- หมายเหตุ
- จัดเก็บ
- ชื่อ
- ราคา
- สถานะ

6.3 ฐานข้อมูลการยืมครุภัณฑ์

รหัสวัสดุครุภัณฑ์	สถานที่	วันที่	เลขที่	ชื่อนามสกุลผู้ยืม
สาขาวิชา	ภาควิชา	ประเภท	เหตุผลในการยืม	วันกำหนดคืน

ประกอบไปด้วย

- รหัสวัสดุครุภัณฑ์
- สถานที่ ที่อยู่
- วันที่
- เลขที่สมาชิก
- ชื่อ สกุล ผู้ยืม
- สาขาวิชา
- ภาควิชา
- ประเภท
- เหตุผลในการยืม
- วันกำหนดคืน

6.4 ฐานข้อมูลรายละเอียดของพัสดุ – ครุภัณฑ์

รหัส	ประเภท	รุ่น	รหัสวัสดุ – ครุภัณฑ์	รายละเอียด
สถานที่	สถานะ	จำนวน	รูปภาพ	หมายเหตุ

ประกอบไปด้วย

- รหัสวัสดุครุภัณฑ์
- ประเภท
- รุ่น
- รายละเอียด
- สถานที่ ที่อยู่
- สถานะ
- จำนวน

- **รูปภาพ**
- **หมายเหตุ**

6.5 ฐานข้อมูลเก็บรายละเอียดของพัสดุ – ครุภัณฑ์ ยืม – คืน

รหัสวัสดุครุภัณฑ์	**สถานที่**	**วันที่**	**เล่มที่**	**ชื่อ นามสกุลผู้ยืม**
ประเภทผู้ยืม	รหัสประจำตัวผู้ยืม	รวมรายการ	วันที่กำหนดคืน	หมายเหตุ

ประกอบไปด้วย

- รหัสวัสดุครุภัณฑ์
- สถานที่ ที่อยู่
- วันที่
- เลขที่สมาชิก
- ชื่อ สกุลผู้ยืม
- ประเภทผู้ยืม
- รหัสประจำตัวผู้ยืม
- รวมรายการ
- วันกำหนดคืน
- หมายเหตุ

6.6 ฐานข้อมูลในการคืนวัสดุครุภัณฑ์

รหัส	วันที่	เลขที่	รายละเอียด	จำนวน
รหัสวัสดุครุภัณฑ์	**หมายเหตุ**			

ประกอบไปด้วย

- **เลขที่สมาชิก**
- **วันที่**
- **รายละเอียด**

- **จำนวน**
- **รหัสวัสดุครุภัณฑ์**
- **หมายเหตุ**

6.7 ฐานข้อมูลพัสดุ – ครุภัณฑ์ สูญหาย

รหัส	วันที่	เลขที่	รายละเอียด	จำนวน
รหัสวัสดุครุภัณฑ์	หมายเหตุ			

ประกอบไปด้วย

- เลขที่สมาชิก
- วันที่
- รายละเอียดวัสดุครุภัณฑ์
- จำนวน
- รหัสวัสดุครุภัณฑ์
- หมายเหตุ

6.8 ฐานข้อมูลบันทึกการปฏิบัติงาน

รหัส	วันที่	รายละเอียด	รหัสวัสดุครุภัณฑ์	หมายเหตุ

ประกอบไปด้วย

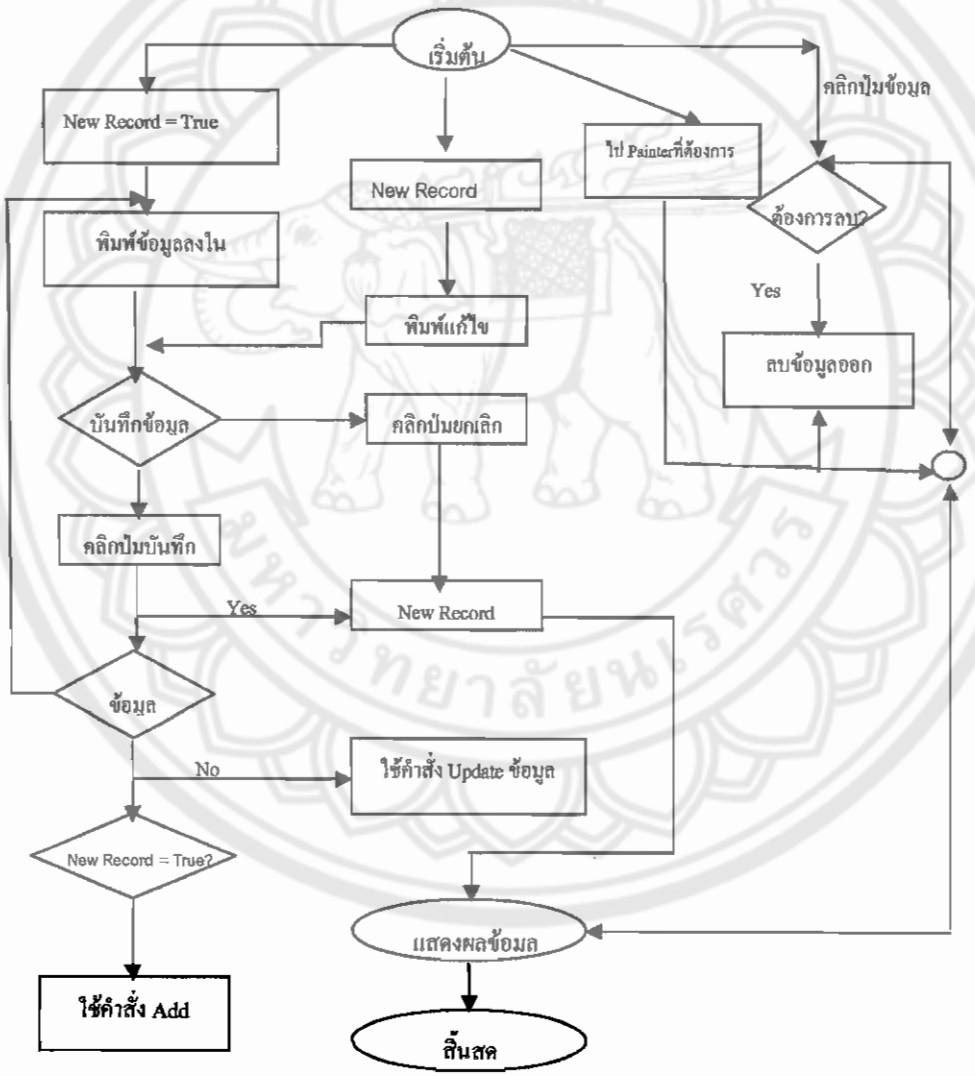
- เลขที่สมาชิก
- วันที่
- **รายละเอียด**
- **รหัสวัสดุครุภัณฑ์**
- **หมายเหตุ**

บทที่ 4
การพัฒนาโปรแกรม

โปรแกรมการจัดการงานด้านวัสดุ ครุภัณฑ์ สำหรับภาควิชา วิศวกรรมไฟฟ้าและคอมพิวเตอร์ ถูกเขียนโดยใช้โปรแกรมภาษา PHP เป็นส่วนติดต่อกับผู้ใช้ และ My SQL เป็นส่วนฐานข้อมูล มีแนววิชา ดังนี้

4.1 แผนภาพการทำงานของโปรแกรม

ระบบสต๊อกวัสดุครุภัณฑ์



รูปที่ 4.1 โฟลชาร์ทโปรแกรมระบบสต๊อกวัสดุครุภัณฑ์

จากจุดเริ่มต้นมีทางเลือกอยู่ 3 ทางที่จะจัดการกับระบบวัสดุครุภัณฑ์ คือ

1. การเพิ่มข้อมูล เมื่อเข้าสู่ระบบวัสดุครุภัณฑ์จะมี Textbox ให้พิมพ์ข้อมูลลงใน Textbox จากนั้นให้คลิกที่ปุ่มเพิ่มข้อมูล ระบบจะทำการ Add ข้อมูลลงในฐานข้อมูล
2. การแก้ไข เมื่อเข้าสู่ระบบวัสดุครุภัณฑ์จะมี Textbox ที่มีข้อมูลอยู่แล้ว ให้ลบข้อมูลเก่าออกแล้วพิมพ์ข้อมูลที่ต้องการลงไป จากนั้นให้คลิกที่ปุ่มแก้ไขข้อมูล ระบบจะทำการ Update ข้อมูลลงในฐานข้อมูล
3. การลบข้อมูล เมื่อเข้าสู่ระบบวัสดุครุภัณฑ์ให้เลือกเรคคอร์ดของข้อมูลที่ต้องการลบจากนั้นให้คลิกที่ปุ่มลบ ระบบจะถามว่าต้องการลบข้อมูลแน่หรือถ้าต้องการลบให้คลิกที่ใช่ ระบบจะทำการ Update ข้อมูลลงในฐานข้อมูล

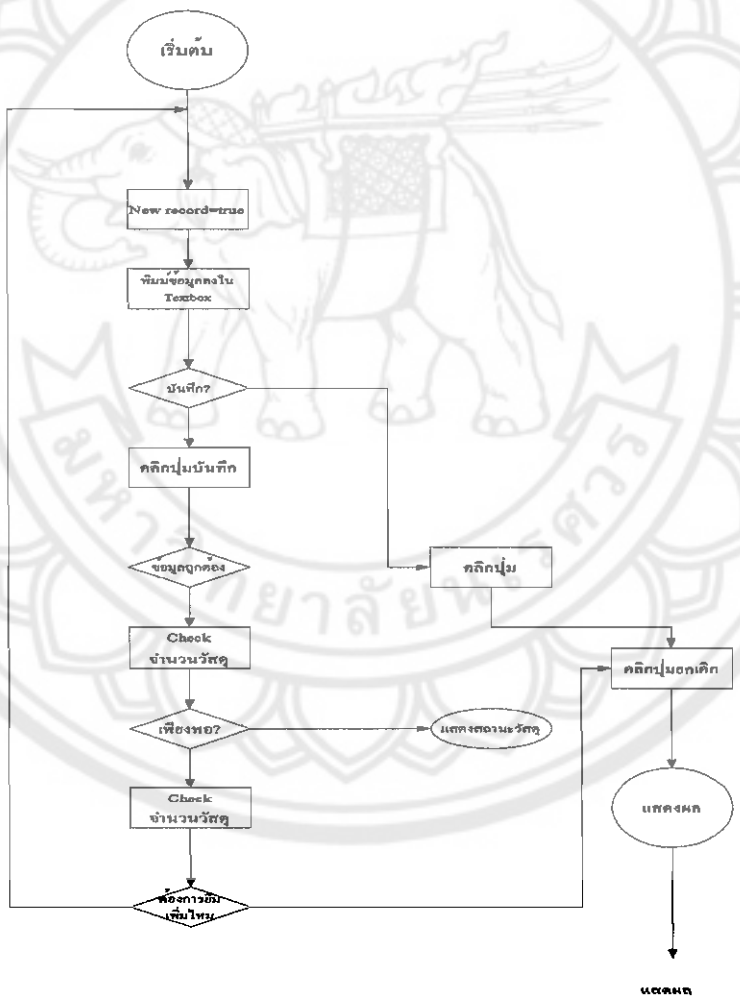
ระบบฐานข้อมูลประวัติผู้ยืม

รูปที่ 4.2 โฟลชาร์ทโปรแกรมระบบฐานข้อมูลประวัติผู้ยืม

จากจุดเริ่มต้นมีทางเลือกอยู่ 3 ทางที่จะจัดการกับระบบสมาชิก/ผู้ยืม คือ

1. การเพิ่มข้อมูล เมื่อเข้าสู่ระบบสมาชิก/ผู้ยืม จะมี Textbox ให้พิมพ์ข้อมูลลงใน Textbox จากนั้น ให้คลิกที่ปุ่มเพิ่มข้อมูล ระบบจะทำการ Add ข้อมูลลงในฐานข้อมูล
2. การแก้ไข เมื่อเข้าสู่ระบบสมาชิก/ผู้ยืม จะมี Textbox ที่มีข้อมูลอยู่แล้ว ให้ลบข้อมูลเก่าออก แล้วพิมพ์ข้อมูลที่ต้องการลงไป จากนั้นให้คลิกที่ปุ่มแก้ไขข้อมูล ระบบจะทำการ Update ข้อมูลลงในฐานข้อมูล
3. การลบข้อมูล เมื่อเข้าสู่ระบบสมาชิก/ผู้ยืม ให้เลือกเรคคอร์ดของข้อมูลที่ต้องการลบจากนั้น ให้คลิกที่ปุ่มลบ ระบบจะถามว่าต้องการลบข้อมูลแน่หรือ? ถ้าต้องการลบให้คลิกที่ใช่ ระบบจะทำการ Update ข้อมูลลงในฐานข้อมูล

ระบบการยืมวัสดุครุภัณฑ์

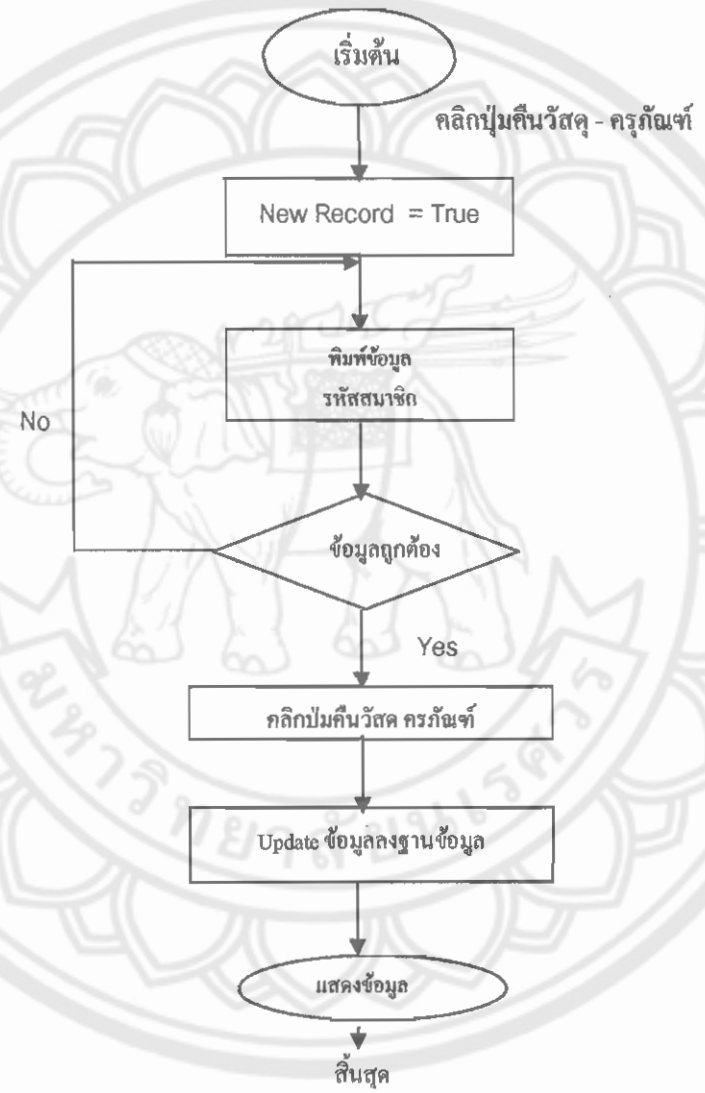


รูปที่ 4.3 โฟลชาร์ทโปรแกรมระบบการยืมวัสดุครุภัณฑ์

จากแผนภาพการเริ่มจะเข้าสู่ Textbox ของรหัสสมาชิก เมื่อกรอกข้อมูลสมาชิกเข้าไป ระบบจะดึงฐานข้อมูลของประวัติสมาชิก/ผู้ยืม ขึ้นมา แล้วสอบถามว่าจะยืมวัสดุครุภัณฑ์ หมายเลข

รหัสอะไร จะมี Textbox ให้ใส่ข้อมูลหมายเลขรหัสวัสดุครุภัณฑ์ เมื่อใส่ข้อมูลลงไปแล้วคลิ๊กที่ปุ่มยืม ระบบจะกลับไปตรวจสอบสถานะของวัสดุครุภัณฑ์ หมายเลขรหัสนั้นว่า ยืมได้หรือไม่ ถ้ายืมได้จะมีกล่องข้อความขึ้นมาบอกว่า ยืมเรียบร้อยแล้ว และถ้าต้องการยืมต่อก็สามารถยืมวัสดุครุภัณฑ์ต่อได้เลยโดยทำตามวิธีการเดิม แต่ถ้ารหัสวัสดุครุภัณฑ์นี้ยืมไม่ได้จะมีกล่องข้อความขึ้นมาบอกว่ายืมไม่ได้

ระบบการคืนวัสดุ ครุภัณฑ์

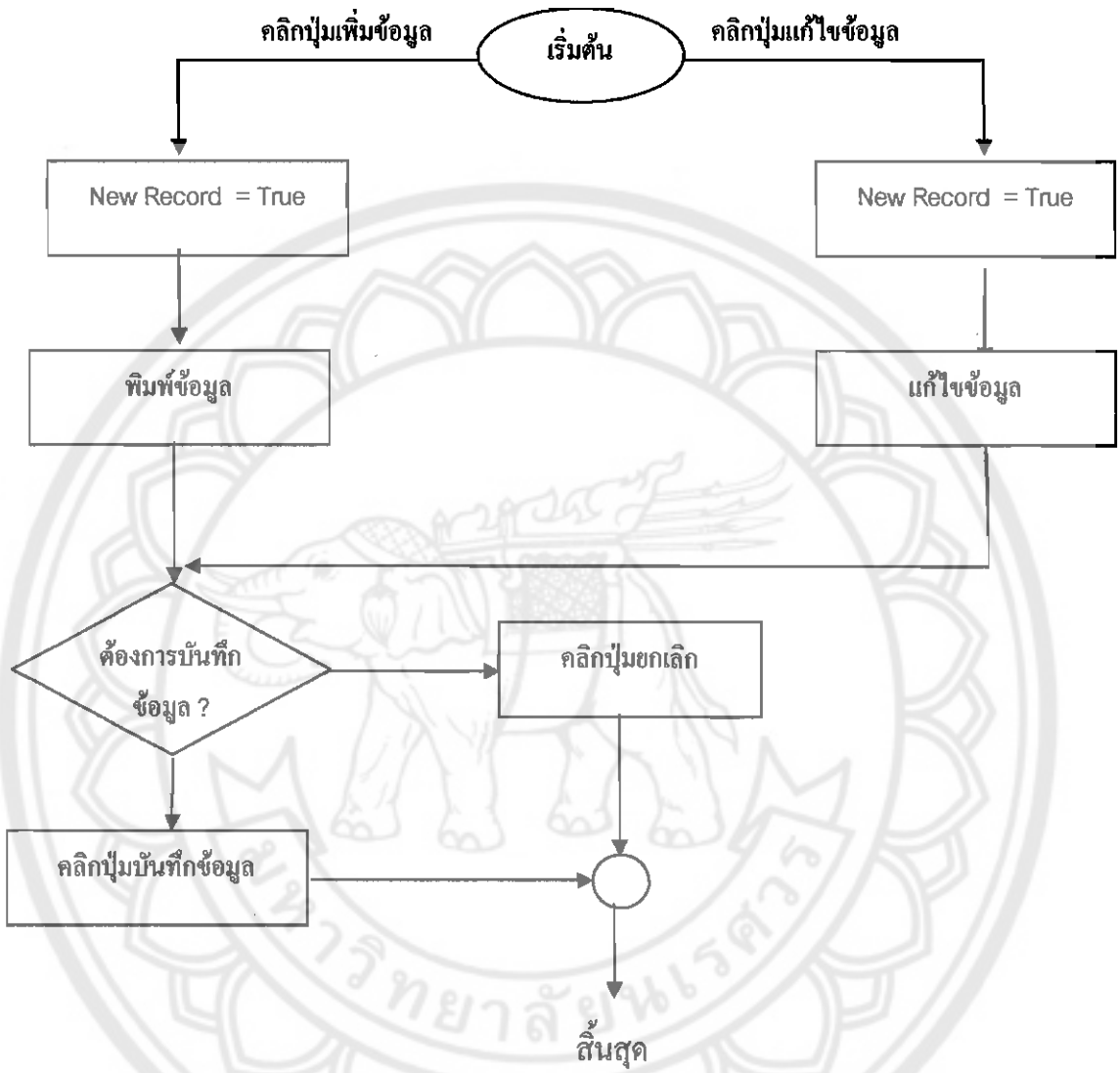


รูปที่ 4.4 โฟลชาร์ทโปรแกรมระบบการคืนวัสดุครุภัณฑ์

จากแผนภาพการเริ่มจะเข้าสู่ระบบการคืนวัสดุครุภัณฑ์ จะมี Textbox ของรหัสสมาชิก เมื่อกรอกข้อมูลสมาชิกเข้าไป ระบบจะดึงฐานข้อมูลของประวัติสมาชิก/ผู้ยืม ขึ้นมา แล้วระบบจะบอกขึ้นมาว่าสมาชิกรหัสนี้ยืมวัสดุครุภัณฑ์ที่มีข้อมูลอะไรบ้าง ถ้าต้องการจะคืน ให้คลิ๊กที่ปุ่มคืน ระบบ

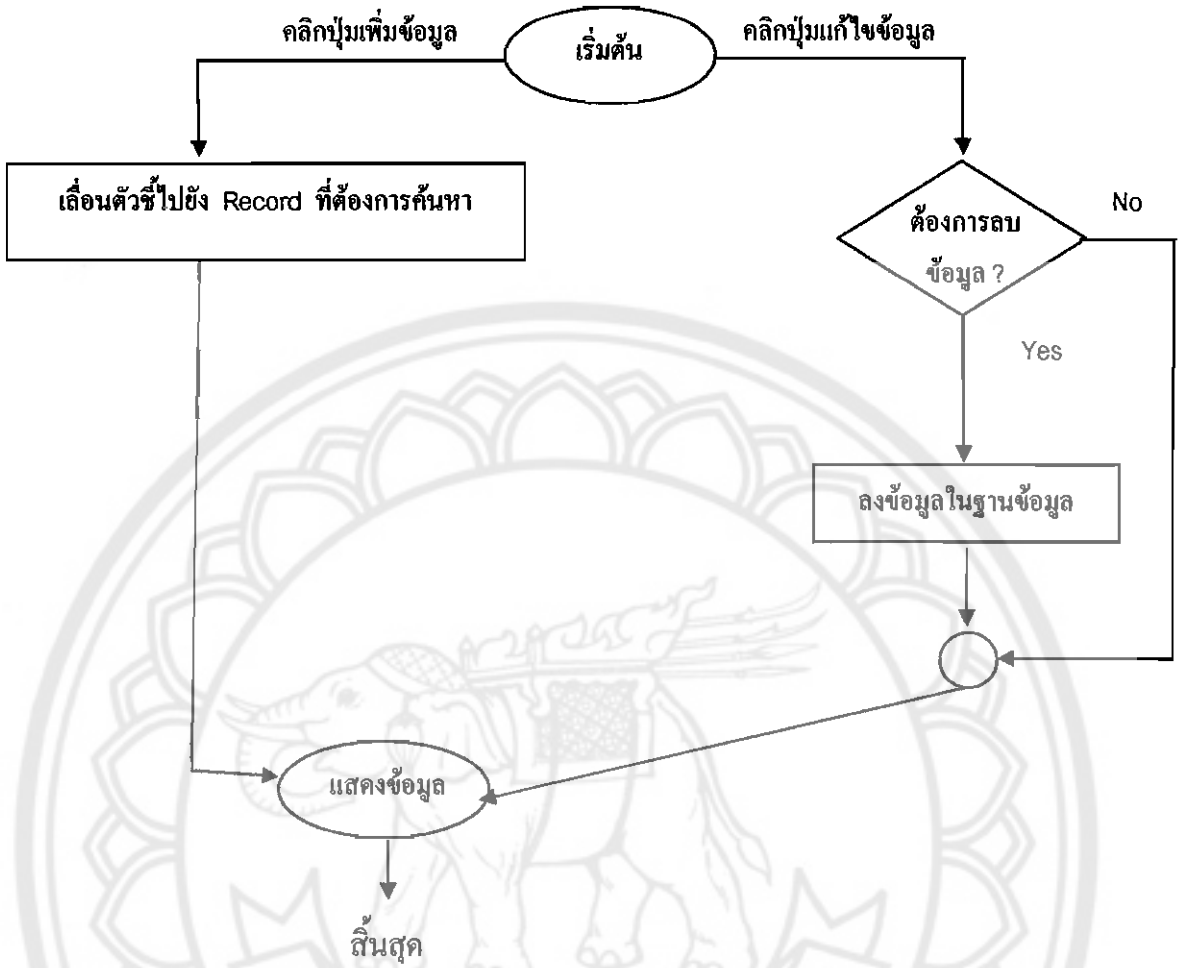
จะทำการ Update ข้อมูลลงฐานข้อมูลว่าคืนแล้ว และถ้าเกิดวัสดุครุภัณฑ์เกิดการสูญหาย ให้คลิ๊กที่ปุ่มหายระบบจะทำการ Update ข้อมูลลงฐานข้อมูลว่าหาย

ระบบประวัติผู้ดูแลระบบ



รูปที่4.5 โฟลชาร์ทโปรแกรมระบบผู้ดูแลระบบ

ระบบประวัติผู้ดูแลระบบ (ต่อ)



รูปที่4.5 (ต่อ) โฟลชาร์ทโปรแกรมระบบผู้ดูแลระบบ

จากจุดเริ่มต้นมีทางเลือกอยู่ 3 ทาง ที่จะจัดการกับระบบผู้ดูแลระบบ คือ

1. การเพิ่มข้อมูล เมื่อเข้าสู่ระบบผู้ดูแลระบบ จะมี Textbox ให้พิมพ์ข้อมูลลงใน Textbox จากนั้นให้คลิ๊กที่ปุ่มเพิ่มข้อมูล ระบบจะทำการ Add ข้อมูลลงในฐานข้อมูล
2. การแก้ไข เมื่อเข้าสู่ระบบผู้ดูแลระบบ จะมี Textbox ที่มีข้อมูลอยู่แล้ว ให้ลบข้อมูลเก่าออก แล้วพิมพ์ข้อมูลที่ต้องการลงไป จากนั้นให้คลิ๊กที่ปุ่มแก้ไขข้อมูล ระบบจะทำการ Update ข้อมูลลงในฐานข้อมูล
3. การลบข้อมูล เมื่อเข้าสู่ระบบผู้ดูแลระบบ ให้เลือกเรคคอร์ดของข้อมูลที่ต้องการลบจากนั้นให้คลิ๊กที่ปุ่มลบ ระบบจะถามว่าต้องการลบข้อมูลแน่หรือ? ถ้าต้องการลบให้คลิ๊กที่ใช่ ระบบจะทำการ Update ข้อมูลลงในฐานข้อมูล

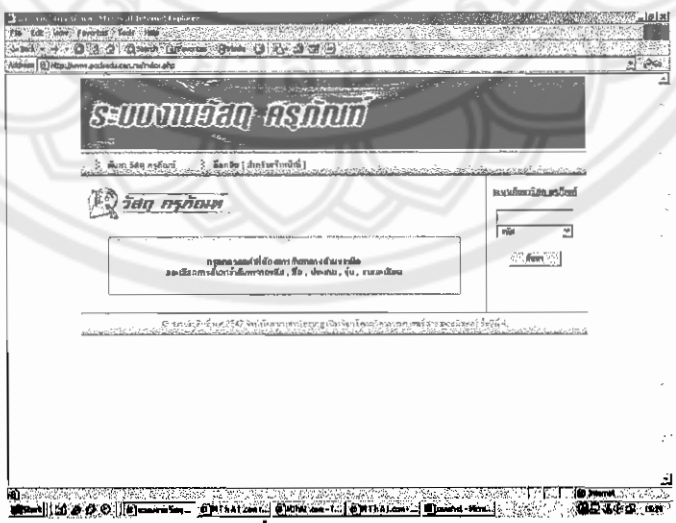
ระบบรายงาน

รูปที่ 4.6 โฟลชาร์ทโปรแกรมแสดงระบบรายงาน

จากแผนภาพการเริ่มต้น เมื่อเข้าสู่ระบบรายงานจะมีข้อมูลให้เลือกว่าจะดูข้อมูลของ การยืม การคืน การสูญหาย ซึ่งสามารถดูได้ทั้งรายวัน รายเดือน และรายปี เมื่อเลือกได้แล้วให้คลิ๊กที่ปุ่มรายการที่ต้องการจะดู ระบบจะแสดงผลรายงานนั้นๆ ขึ้นมาให้ดู

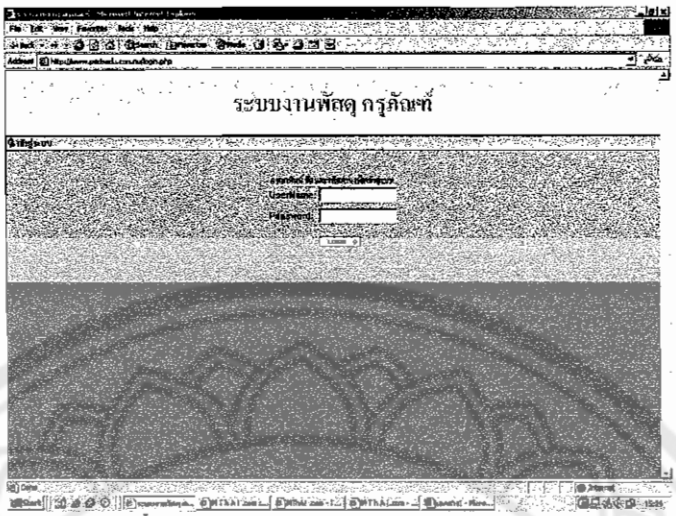
รูปแบบของโปรแกรมระบบการจัดการงานด้านวัสดุครุภัณฑ์ สำหรับภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ มหาวิทยาลัยนเรศวร

การเข้าสู่ระบบ



รูปที่ 4.7 การเข้าสู่ระบบ

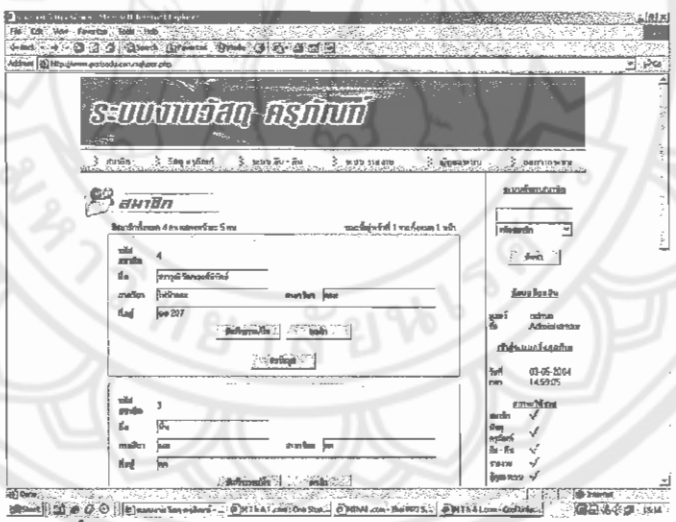
การเข้าสู่ระบบสำหรับผู้ดูแลระบบ



รูปที่4.8 การเข้าสู่ระบบสำหรับผู้ดูแลระบบ

เป็นหน้าจอที่ผู้ดูแลระบบต้องใส่รหัสผ่านเพื่อเข้าไปใช้งานระบบ

ระบบประวัติผู้ยืม/สมาชิก



รูปที่4.9 ลักษณะของโปรแกรมการจัดการงานสมาชิก

การจัดเก็บข้อมูลลงไป ฟอร์มผู้ยืม/สมาชิก สิ่งที่จัดเก็บประกอบไปด้วยข้อมูลพื้นฐานของสมาชิก/ผู้ยืม เช่น ที่อยู่ ชื่อ ชั้นปี รหัสนิสิต คณะ ภาควิชา และเบอร์ที่ติดต่อได้ เพื่อที่จะทำให้เราทราบว่า สมาชิก/ผู้ยืมมีภูมิลำเนาอยู่ที่ไหน และสามารถติดต่อได้อย่างไร ในกรณีที่ต้องการแจ้งปัญหาหรือในการติดตาม รายละเอียดของการทำงานของฟอร์มผู้ยืม/สมาชิก มีดังนี้

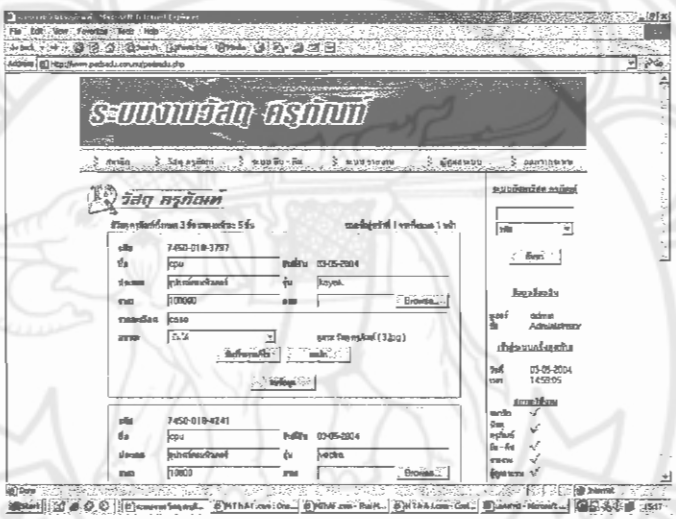
- **การเพิ่มข้อมูล หลังจากกดปุ่มเพิ่มข้อมูล ช่องที่แสดงผลข้อมูล (Text Box) จะกลาย**

เป็นช่องว่าง เพื่อที่จะคอยรับค่าข้อมูลที่ป้อนเข้าไป เมื่อพิมพ์ข้อมูลเสร็จก็สามารถกดปุ่มเพิ่มข้อมูล หรือลบข้อมูลก็ได้ ในกรณีที่เกิดการกรอกข้อมูลผิดพลาด ก็สามารถคลิกที่ปุ่มคอนโทรลการแก้ไขข้อมูลได้หลังจากนั้น ซึ่งทำการลบข้อมูลที่ผิดพลาด แล้วคลิกปุ่มบันทึกข้อมูลใหม่ได้

- การลบข้อมูลเมื่อต้องการลบข้อมูลเราต้องทำการเลื่อนคอมพิวเตอร์ไปยังเรคคอร์ดที่ต้องการลบ (โดยการค้นหาจากช่องค้นหาข้อมูล) หลังจากนั้น ก็ทำการกดปุ่มลบข้อมูล โปรแกรมก็จะทำการลบข้อมูล ออกจากฐานข้อมูล

- สำหรับการค้นหาข้อมูล เราสามารถค้นหาหาข้อมูลได้จากรหัสสมาชิก/ผู้ยืม

ระบบบันทึก/รายการวัสดุ ครุภัณฑ์

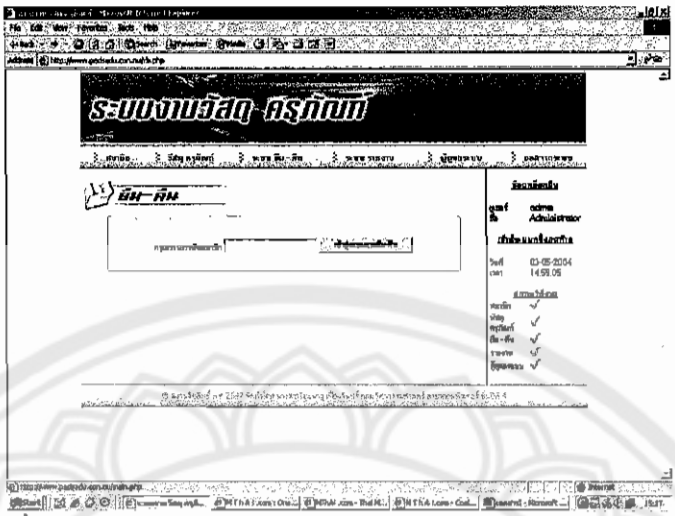


รูปที่ 4.10 ลักษณะของโปรแกรมระบบบันทึกรายการ วัสดุ – ครุภัณฑ์

ลักษณะการทำงานก็จะคล้ายกับฟอร์มสมาชิก/ผู้ยืม การจัดเก็บข้อมูลในฟอร์มรายการวัสดุครุภัณฑ์ก็จะประกอบไปด้วยข้อมูลพื้นฐานของวัสดุ ครุภัณฑ์ ชิ้นนั้น ๆ เช่น ชื่อ รหัสวัสดุครุภัณฑ์ รุ่น ราคา วันที่ซื้อเข้ามา

- การเพิ่มข้อมูล หลังจากกดปุ่มเพิ่มข้อมูล ช่องที่แสดงผลข้อมูล (Text Box) จะกลายเป็นช่องว่าง เพื่อที่จะคอยรับค่าข้อมูลที่ป้อนเข้าไป เมื่อพิมพ์ข้อมูลเสร็จก็สามารถกดปุ่มเพิ่มข้อมูล หรือลบข้อมูลก็ได้ ในกรณีที่เกิดการกรอกข้อมูลผิดพลาด ก็สามารถคลิกที่ปุ่มคอนโทรลการแก้ไขข้อมูลได้หลังจากนั้น ซึ่งทำการลบข้อมูลที่ผิดพลาด แล้วคลิกปุ่มบันทึกข้อมูลใหม่ได้
- การลบข้อมูล เมื่อต้องการลบข้อมูล เราต้องทำการเลื่อนคอมพิวเตอร์ ไปยังเรคคอร์ดที่ต้องการลบ (โดยการค้นหาจากช่องค้นหาข้อมูล) หลังจากนั้น ก็ทำการกดปุ่มลบข้อมูล โปรแกรมก็จะทำการลบข้อมูล ออกจากฐานข้อมูล
- การค้นหาข้อมูล เราสามารถค้นหาข้อมูลจากรหัสวัสดุ – ครุภัณฑ์

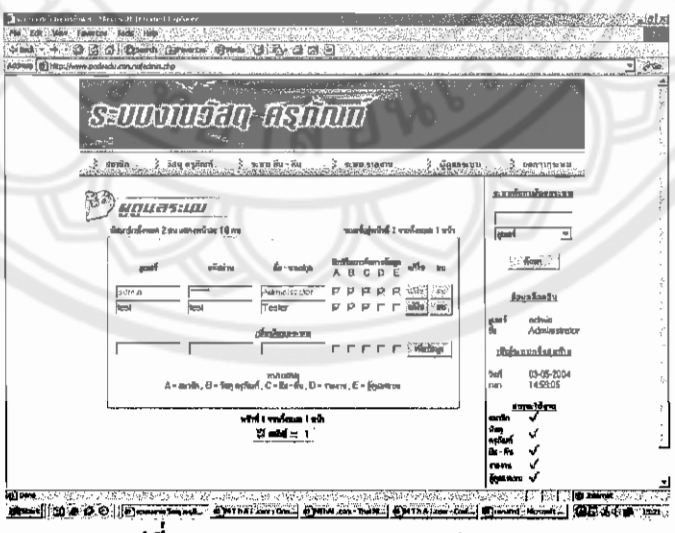
ระบบการยืมวัสดุ – ครุภัณฑ์



รูปที่4.11 ลักษณะของโปรแกรมระบบการยืม คืนวัสดุ – ครุภัณฑ์

การยืมวัสดุ ครุภัณฑ์ ก่อนอื่นจะต้องกรอกรหัสสมาชิกเข้ามาก่อนระบบจะทำการขึ้นประวัติของสมาชิก/ผู้ยืมขึ้นมา จากนั้นจะมี Text Box ถามขึ้นมาว่าจะต้องการยืมวัสดุครุภัณฑ์ รหัสอะไร เมื่อกรอกข้อมูลรหัสลงไปแล้วจะขึ้นข้อมูลวัสดุครุภัณฑ์ขึ้นมา และบอกสถานะของวัสดุครุภัณฑ์ชิ้นนั้นด้วยว่า สามารถยืมได้หรือไม่

ระบบผู้ดูแลระบบ



รูปที่4.12 โปรแกรมระบบระบบผู้ดูแลระบบ

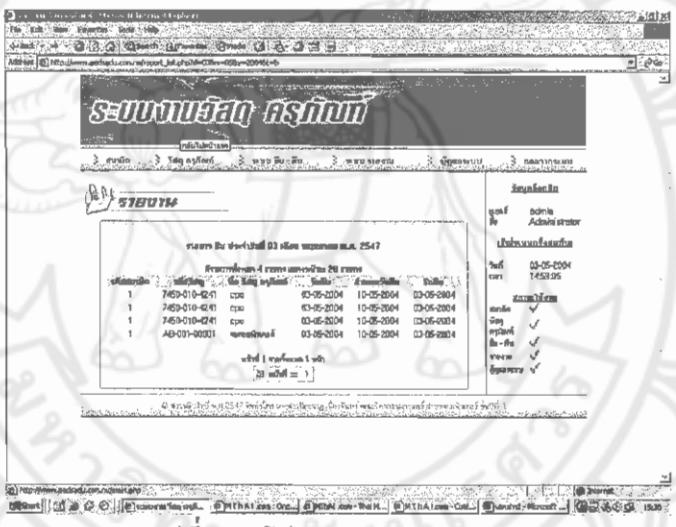
- ระบบเจ้าหน้าที่ จะคล้ายกับระบบสมาชิก/ผู้ยืม คือ สามารถเพิ่มข้อมูล แก้ไขข้อมูล และลบข้อมูลได้เหมือนกัน

- การเพิ่มข้อมูล หลังจากกดปุ่มเพิ่มข้อมูล ช่องที่แสดงผลข้อมูล (Text Box) จะกลายเป็นช่องว่าง เพื่อที่จะคอยรับคำข้อมูลที่ป้อนเข้าไป เมื่อพิมพ์ข้อมูลเสร็จก็สามารถกดปุ่มเพิ่มข้อมูล หรือลบข้อมูลก็ได้ ในกรณีที่เกิดการกรอกข้อมูลผิดพลาด ก็สามารถคลิกที่ปุ่มคอนโทรลการแก้ไขข้อมูลได้หลังจากนั้น ซึ่งทำการลบข้อมูลที่ผิดพลาด แล้วคลิกปุ่มบันทึกข้อมูลใหม่ได้

- การลบข้อมูล เมื่อต้องการลบข้อมูล เราต้องทำการเลื่อนคอมพิวเตอร์ไปยังเรคคอร์ดที่ต้องการลบ (โดยการค้นหาจากช่องค้นหาข้อมูล) หลังจากนั้น ก็ทำการกดปุ่มลบข้อมูล โปรแกรมก็จะทำการลบข้อมูล ออกจากฐานข้อมูล

- การแก้ไขข้อมูล สามารถคลิกที่ปุ่มแก้ไข และทำการลบข้อมูลที่ผิดพลาดแล้วบันทึกข้อมูลลงใหม่

ระบบรายงาน



รูปที่4.13 โปรแกรมระบบรายงาน

โปรแกรมระบบรายงานนี้ จะทำงานโดยการคลิกที่ปุ่มรายงาน จะมีหน้าจอต่อไปขึ้นมาถามว่า ต้องการที่จะดูรายงานแบบไหน เช่น รายงานการยืมของวันนี้ เดือนนี้ หรือรายงานการคืนของว่ามีเท่าไร มีอะไรบ้าง

MISSING

บทที่ 5
บทสรุป

5.1 สรุปผล

1. **ฐานข้อมูลประวัติสมาชิก/ผู้ยืม จะเก็บข้อมูลเกี่ยวกับประวัติพื้นฐานของสมาชิก/ ผู้ยืม เช่น ที่อยู่ ชื่อ ชั้นปี รหัสนิสิต คณะ ภาควิชา และเบอร์ที่ติดต่อได้ ในส่วนของโปรแกรมเจ้าหน้าที่** สามารถเพิ่มข้อมูล ลบข้อมูล แก้ไขข้อมูล ประวัติสมาชิก/ผู้ยืมได้ และสามารถค้นหาข้อมูลตามรหัสสมาชิก/ผู้ยืมได้

2. ฐานข้อมูลการบันทึกรายการวัสดุครุภัณฑ์ จะเก็บข้อมูลของ วัสดุครุภัณฑ์ชิ้นนั้น ๆ เช่น รหัสวัสดุครุภัณฑ์ ชื่อ รุ่น ราคา เป็นต้น ชิ้นส่วนของโปรแกรม เจ้าหน้าที่สามารถเพิ่มข้อมูล ลบข้อมูล แก้ไขข้อมูล รายการวัสดุครุภัณฑ์ได้ และสามารถค้นหาข้อมูลตามรหัสวัสดุครุภัณฑ์ได้

3. ฐานข้อมูลการยืม – คืน จะเก็บข้อมูลของวัสดุครุภัณฑ์ที่ยืมไป เช่น สมาชิก/ผู้ยืม รหัสอะไร รหัสวัสดุครุภัณฑ์ ชื่อ รุ่น วันที่ยืม เป็นต้น ในส่วนของโปรแกรมเจ้าหน้าที่สามารถค้นหาข้อมูลจากรหัสสมาชิก/ผู้ยืม จะรู้ว่ารหัสสมาชิก/ผู้ยืม ยืมวัสดุ ครุภัณฑ์อะไรไป และนำมาคืนแล้วหรือไม่

4. ฐานข้อมูลประวัติเจ้าหน้าที่ จะเก็บข้อมูลของเจ้าหน้าที่ เช่น ชื่อ ที่อยู่ เบอร์ติดต่อ เป็นต้น ในส่วนของโปรแกรมสามารถแก้ไขข้อมูล เพิ่มข้อมูลหรือลบข้อมูลได้

5. ฐานข้อมูลรายงานจะเก็บข้อมูลรายงาน การยืมของ การคืนของ ของสมาชิก/ผู้ยืมและนำมาสรุปผลออกมาเป็นรายงาน

6. ผู้ดูแลระบบ สามารถแก้ไขข้อมูลในระบบได้ทุกอย่าง เช่น สามารถจำกัดสิทธิ์การใช้งานของระบบ เป็นต้น

5.2 ประเมินผลและข้อเสนอแนะ

จากผลของโครงงานเมื่อเปรียบเทียบกับวัตถุประสงค์ที่ตั้งไว้ ได้ผลว่า

1. จากการดำเนินงานที่ผ่านมาสามารถออกแบบฐานข้อมูลได้ ตามโครงสร้างฐานข้อมูลในบทที่ 3
2. **เครื่องมือที่ใช้ในการพัฒนาโปรแกรมการจัดการงานด้านวัสดุ ครุภัณฑ์ ได้แก่ โปรแกรมภาษา PHP ในการเขียนโปรแกรม และใช้MY SQL เป็นฐานข้อมูลของระบบ เนื่องจากโปรแกรม SQL เป็นฐานข้อมูลที่ใช้กันอย่างแพร่หลาย**
3. **โปรแกรมระบบการจัดการงานด้านวัสดุ ครุภัณฑ์ ที่ได้พัฒนาขึ้นประกอบด้วยส่วนของโปรแกรมที่จัดการฐานข้อมูลต่าง ๆ ดังนี้**
   1) **ฐานข้อมูลประวัติสมาชิก/ผู้ยืม**

2) ฐานข้อมูลการบันทึกรายการวัสดุครุภัณฑ์

3) ฐานข้อมูลการยืม – คืน วัสดุครุภัณฑ์

4) ผู้ดูแลระบบ

5.3 ปัญหา ข้อเสนอแนะ และแนวทางการแก้ไข

1. โครงงานวิศวกรรม "การจัดการงานด้านวัสดุ ครุภัณฑ์ สำหรับภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ ผ่านเครือข่ายอินเตอร์เน็ต" อยู่บนพื้นฐานของการออกแบบฐานข้อมูลวัสดุครุภัณฑ์จริง โครงสร้างของฐานข้อมูลที่ออกแบบมานั้น อาจไม่สมบูรณ์ตามทฤษฎี เพราะมีข้อจำกัดหลายอย่าง เช่น การค้นหาชื่อ ยังต้องค้นหาจากรหัสสมาชิก/ผู้ยืม ซึ่งบางครั้งสมาชิก/ผู้ยืม อาจลืมได้

2. โปรแกรมที่ได้สามารถทำงานได้ตามจุดประสงค์แต่ว่าในส่วนการติดต่อกับผู้ใช้ การใช้งานอาจจะใช้งานได้ไม่สะดวกมากนัก

เอกสารอ้างอิง

[1] กิติศักดิ์ เจริญโภคานนท์. Application PHP 4 . พิมพ์ครั้งที่ 2 กรุงเทพฯ: ซัคเซส มีเดีย จำกัด, บริษัท . 2544.

[2] ดวงแก้ว สวามิภักดิ์. ระบบฐานข้อมูล. กรุงเทพฯ : ซีเอ็ด ยูเคชั่น จำกัด, บริษัท 2534.

[3] อำไพ พรประเสริฐสกุล. การวิเคราะห์และออกแบบฐานข้อมูล. พิมพ์ครั้งที่3. กรุงเทพฯ:ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ

[4] บุญฤทธิ์ กู้เกียรติกูล. ระบบฐานข้อมูล. ขอนแก่น. หน่วยสารบรรณงานบริหารและธุรการ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยขอนแก่น. 2 542

[5] มณีโชติ สมานไทย. สร้างและปรับแต่งเว็บเพจอย่างง่ายด้วย HTML. กรุงเทพฯ: เอช เอ็น กรุ๊ป จำกัด , บริษัท. 2544