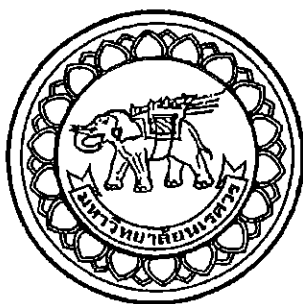




การพัฒนาระบบไอทีเฮลป์เดสก์สำหรับงานบริการผู้ใช้งานระบบเครือข่ายคอมพิวเตอร์

มหาวิทยาลัยนเรศวร

IT HELPDESK DEVELOPMENT FOR NARESUAN UNIVERSITY NETWORK

นาย วัชรพล เพชรต๋น รหัสนิติต 44370351

นาย วิกรม วิเศษพลกรัง รหัสนิติต 45371218

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ..... 25 พ.ค. 2553
เลขทะเบียน..... 1499 2839
เลขเรียกหนังสือ..... 8/5
มหาวิทยาลัยนเรศวร ๑๖๘ 17

2551

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

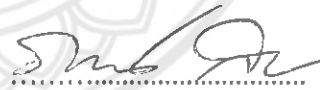
ปีการศึกษา 2551

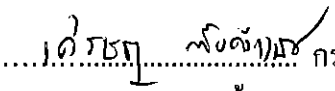


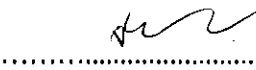
ใบรับรองโครงการวิศวกรรม

หัวข้อโครงการ	การพัฒนาระบบไอทีเซตพีเคตส์สำหรับงานบริการผู้ใช้งานระบบเครือข่ายคอมพิวเตอร์ มหาวิทยาลัยธนบุรี		
ผู้เสนอโครงการ	นายวัชรพล	เพชรตุ่น	รหัส 44370351
	นายวิกรม	วิเศษพลกรัง	รหัส 45371218
อาจารย์ที่ปรึกษา	อาจารย์ภาณุพงศ์ สอนคม		
สาขาวิชา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2551		

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยธนบุรี อนุมัติให้โครงการฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร วิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์
คณะกรรมการสอบโครงการวิศวกรรม

 ประธานกรรมการ
(อาจารย์ภาณุพงศ์ สอนคม)

 กรรมการ
(อาจารย์เศรษฐา ตั้งคำวานิช)

 กรรมการ
(อาจารย์จิราพร พุกสุข)

หัวข้อโครงการ	การพัฒนาระบบไอทีที่เสถียรและดีสำหรับงานบริการผู้ใช้งานระบบ เครือข่ายคอมพิวเตอร์ มหาวิทยาลัยนเรศวร		
ผู้ดำเนินโครงการ	นายวัชรพล	เพชรต่วน	รหัส 44370351
	นายวิกรม	วิเศษพลกรัง	รหัส 45371218
อาจารย์ที่ปรึกษา	อาจารย์ภาณุพงศ์	สอนคม	
สาขาวิชา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2551		

บทคัดย่อ

เนื่องด้วยระบบเครือข่ายในมหาวิทยาลัยนเรศวร มีผู้ใช้บริการเป็นจำนวนมาก ตั้งแต่อดีต บุคลากรและคณาจารย์ที่ทำงานอยู่ในมหาวิทยาลัย การให้บริการแก้ปัญหา และปรึกษาให้คำแนะนำ การใช้งานแก่ผู้ใช้บริการระบบเครือข่ายคอมพิวเตอร์ ในปัจจุบันยังไม่มีประสิทธิภาพเพียงพอ โดยการติดต่อขอรับบริการต่างๆ ยังเป็นเพียงระบบแมนนวล (Manual) เท่านั้น เมื่อผู้ใช้บริการต้องการ ขอคำปรึกษาหรือแจ้งปัญหาการใช้งานต่างๆ จะทำได้เพียงการโทรศัพท์หรือเข้ามาติดต่อโดยตรงที่ สำนักงานซึ่งมีความไม่สะดวก อีกทั้งการประสานงานของเจ้าหน้าที่ยังไม่มีประสิทธิภาพเท่าที่ควร ดังนั้นผู้ทำโครงการจึงได้จัดทำระบบการบริหารจัดการข้อมูลสำหรับงานบริการผู้ใช้งานระบบ เครือข่ายคอมพิวเตอร์ เพื่อรองรับปัญหาดังกล่าว ข้างต้น โดยมีการนำเทคโนโลยีสารสนเทศมาใช้ โดยเฉพาะ Web Application มาประยุกต์ใช้กับการแก้ปัญหาของผู้ใช้งานและกระบวนการของ เจ้าหน้าที่ และเพิ่มระบบแก้ไขปัญหาคด้วยตัวของผู้ใช้งานเอง (Self Help) โดยอ้างอิงทฤษฎีต้นไม้ (Decision Tree) มาใช้ในการวิเคราะห์คุณสมบัติความน่าจะเป็นมากที่สุด (Analytical Characterization) ที่ผู้ใช้งานต้องการคำตอบ

ผลที่ได้รับจากการพัฒนาระบบ IT Helpdesk นี้ มีประโยชน์ต่องานให้บริการระบบ เครือข่ายคอมพิวเตอร์ในมหาวิทยาลัยนเรศวร สามารถนำไปแก้ไขปัญหเกี่ยวกับระบบการทำงานของ เจ้าหน้าที่ และเพิ่มความสะดวกสบายแก่ผู้ใช้งาน อีกทั้งยังเพิ่มรายละเอียดต่างๆ ที่ระบบเดิมไม่มี คือ

1. ระบบการช่วยแก้ไขปัญหการใช้งานด้วยตนเอง (Self Help)
2. ระบบการแจ้งขอรับบริการ (Services Request)
3. ระบบการประสานงานของเจ้าหน้าที่และการจัดลำดับงาน (Services Queuing)
4. ระบบการบันทึกและรายงานสถิติการให้บริการและการรายงานผล (Services Report)

Project Title	IT Helpdesk Development For Naresuan University Network		
Name	Mr.Watcharapol	Peththoon	ID.44370351
	Mr.Wikrom	Wisapolkrang	ID.45371218
Project Advisor	Mr.Panupong	SornKhom	
Major	Computer Engineering		
Department	Electrical and Computer Engineering		
Academic Year	2008		

.....

ABSTRACT

The network systems in Naresuan University there are a lot of users, both students and personals and teachers of University. Server solves and consults, advice, using works to the server users of computer network systems which currently still not quality enough by the connection to all services request be Manual System only. When server users wants to consult request or inform the problem of using all working, it can use telephone only or come in contact directly at office which not convenience. Moreover, work integration of an officer not efficiency expected. Thus, the project maker then makes administration system the data for server with users the computer network systems for support a problem aforementioned by takes Communication Technology come to use, especially using Web site of Application for apply with server solves of users and the procedure of an officer and enhance the server solves systems by Self Help in analytical characterization most of a problem that the user wants the answer. (Decision Tree)

The result from development of this IT Helpdesk Systems be valuable to work of the computer network systems at Naresuan University, can takes to server solves, enhance an efficiency and convenience in the work of users. By have the detail as follows;

1. The system of Server Solve Help with Self Help
2. Inform System for Services Request
3. Work Integration System of an officer and Services Queuing
4. Record System and Statistics Report and Services Report

กิตติกรรมประกาศ

การจัดทำโครงการในครั้งนี้ สำเร็จลงไปได้ด้วยดีต้องขอกราบขอบพระคุณ คุณพ่อ คุณแม่ สำหรับการให้การสนับสนุน และความช่วยเหลือในด้านต่างๆ อาจารย์ ภาณุพงศ์ สอนคม ที่มาเป็นอาจารย์ที่ปรึกษาโครงการ กับอาจารย์ เศรษฐา ตั้งคำวานิช และ อาจารย์ จิราพร พุกสุข ที่มาเป็นกรรมการในการสอบโครงการนี้ ที่ให้ความดูแลเอาใจใส่ให้คำแนะนำในการทำงานเป็นอย่างดี ขอขอบพระคุณท่านคณะกรรมการสอบทุกท่านที่ได้สละเวลาอันมีค่า ครูอาจารย์ทุกท่านที่ประสิทธิ์ประสาทวิชาความรู้ให้ตลอดมา ตลอดจนเพื่อนๆทุกคนที่ได้ให้กำลังใจในการทำโครงการนี้

ผู้จัดทำ

นายวัชรพล เพชรดุ่น

นายวิกรม วิเศษพลกรัง



สารบัญ

หน้า

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

บทที่ 1 บทนำ

1.1 ความสำคัญและที่มาของการศึกษา	1
1.2 วัตถุประสงค์ของโครงการ.....	2
1.3 ขอบข่ายของโครงการ	2
1.4 ขั้นตอนของการดำเนินงาน	3
1.5 ประโยชน์ที่คาดว่าจะได้รับ	4

บทที่ 2 แนวคิดและทฤษฎีที่เกี่ยวข้อง

2.1 การให้บริการระบบเครือข่ายคอมพิวเตอร์ในมหาวิทยาลัยนเรศวร	5
2.2 การจัดทำและบริหารระบบ IT Helpdesk	7
2.3 เทคโนโลยีและเครื่องมือที่เกี่ยวข้อง	8
2.3.1 แนวคิดการพัฒนาระบบคอมพิวเตอร์ด้วยหลักการ Object Orientation	8
2.3.2 UML (Unified Modeling Language)	12
2.3.3 องค์ประกอบของ UML	15
2.3.4 Diagram	17
2.3.5 Visual Basic	25
2.3.6 .NET Framework	25
2.3.7 Visual Studio.NET และ ASP.NET	27
2.3.8 Microsoft SQL Server 2005 (MS SQL Server 2005)	28
2.3.9 ระบบฐานข้อมูล	28
2.3.10 เทคโนโลยีเหมืองข้อมูล (Data Mining)	38

สารบัญ (ต่อ)

หน้า

2.3.11 ทฤษฎีการตัดสินใจ (Decision Tree)	41
บทที่ 3 ขั้นตอนการดำเนินงาน	
3.1 วิธีดำเนินงานการศึกษาค้นคว้า.....	48
3.1.1 การศึกษาระบบการทำงานแบบเดิม.....	49
3.1.2 การวิเคราะห์และออกแบบระบบ	52
3.1.3 การพัฒนาระบบการให้บริการและแสดงลำดับการให้บริการ	57
3.1.4 การออกแบบโครงสร้างของเว็บเพจ	62
3.1.5 การประเมินผลระบบ	64
บทที่ 4 การใช้งานโปรแกรม	
4.1 ส่วนประกอบของตัวโปรแกรม	65
4.2 รูปแบบของโปรแกรมและวิธีการใช้งาน	65
บทที่ 5 บทสรุป	
5.1 สรุปผล	89
5.2 ข้อเสนอแนะ	90
เอกสารอ้างอิง	91
ประวัติผู้เขียนโครงงาน	92

สารบัญตาราง

ตารางที่	หน้า
1.1 ตารางการปฏิบัติงาน	3
2.1 เปรียบเทียบการวิเคราะห์และออกแบบระบบวิธีเดิมกับวิธีเชิงวัตถุ	8
2.2 ตัวอย่างตารางข้อมูลของนักศึกษาชั้นต้น	39
2.3 ตัวอย่างข้อมูลการลงทะเบียนเรียนของนักศึกษา	39
3.1 ผังขั้นตอนระบบการทำงานแบบเดิม (Work Flow Diagram)	50
3.2 รายละเอียดของ NU-NET Helpdesk Use Cases	61



สารบัญรูป

รูปที่	หน้า
2.1 ผังโครงสร้างการบริหารงานบริการระบบเครือข่ายคอมพิวเตอร์มหาวิทยาลัยนเรศวร.....	5
2.2 หน่วยบริการผู้ใช้งานและผู้ที่เกี่ยวข้องกับงานบริการเครือข่ายคอมพิวเตอร์.....	6
2.3 ตัวอย่าง Class นักศึกษา.....	9
2.4 การถ่ายทอดคุณสมบัติแบบ Multiple Inheritances.....	10
2.5 ความสัมพันธ์แบบ 2 ทิศทางของ Object หรือ Class แบบ Association.....	10
2.6 ความสัมพันธ์ระหว่าง Object หรือ Class แบบ Aggregation.....	11
2.7 ความสัมพันธ์ระหว่าง Object หรือ Class แบบ Composition.....	11
2.8 ความสัมพันธ์ระหว่าง Object หรือ Class แบบ Generalization.....	12
2.9 แสดงสัญลักษณ์ต่างๆ ของ UML Diagram.....	15
2.10 ตัวอย่างความสัมพันธ์แบบ Dependency.....	16
2.11 ตัวอย่างความสัมพันธ์แบบ Association.....	16
2.12 ตัวอย่างความสัมพันธ์แบบ Generalization.....	17
2.13 ตัวอย่างความสัมพันธ์แบบ Realization.....	17
2.14 ตัวอย่าง Use Case Diagram ที่สร้างจาก Visual UML.....	18
2.15 ตัวอย่าง Sequence ในขั้นตอนของการออกแบบ.....	19
2.16 ตัวอย่าง Collaboration Diagram ที่สร้างจาก Visual Diagram.....	20
2.17 ตัวอย่าง State Diagram ที่สร้างจาก Visual UML.....	21
2.18 ตัวอย่าง Activity Diagram ที่สร้างจาก Visual UML.....	22
2.19 ตัวอย่าง Class Diagram ที่สร้างจาก Visual UML.....	23
2.20 ตัวอย่าง Component Diagram.....	24
2.21 ตัวอย่าง Deployment Diagram.....	24
2.22 โมเดลของเทคโนโลยีคอมพิวเตอร์.....	26
2.23 ระบบการจัดการฐานข้อมูล DBMS.....	29
2.24 สถาปัตยกรรมของระบบการทำเหมืองข้อมูล.....	38
2.25 โครงสร้างคลังข้อมูล (Data Warehouse).....	39
2.26 ลักษณะแผนภูมิแบบต้นไม้.....	41
2.27 แผนภูมิแบบต้นไม้ที่มีโน้ตย่อยหลายระดับ.....	42

สารบัญรูป (ต่อ)

รูปที่	หน้า
2.28 แผนภูมิแบบต้นไม้ที่มีโหนดไม่เกิน 2 โหนดในแต่ละชั้น	42
2.29 แผนภูมิตัดสินใจแบบต้นไม้แสดงหลักเกณฑ์พิจารณาอนุมัติเงินกู้.....	43
2.30 การใช้สัญลักษณ์ในแผนผังตัดสินใจแบบต้นไม้.....	43
2.31 การเขียนอธิบายกระบวนการด้วยแผนภูมิดำเนินงานในแนวนอน.....	44
2.32 การเขียนอธิบายกระบวนการด้วยแผนภูมิดำเนินงานตามแนวนอน.....	45
3.1 ขั้นตอนการดำเนินงาน.....	47
3.2 หน่วยบริการผู้ใช้งานและผู้เกี่ยวข้องในระบบการทำงานแบบเดิม	48
3.3 ปัญหาของการให้บริการในระบบเดิม.....	50
3.4 ระบบ IT Helpdesk System Requirements	51
3.5 ผังระบบ NU-NET Helpdesk.....	52
3.6 ผัง Activities ของผู้ให้บริการ (User)	53
3.7 ผัง Activities ของเจ้าหน้าที่ซ่อมบำรุง (Net Staff)	54
3.8 ผัง Activities ของเจ้าหน้าที่ดูแลระบบ Helpdesk (Hdsa).....	54
3.9 ผังโครงสร้างของระบบแก้ปัญหาด้วยตนเอง (Self Help)	55
3.10 ผังขั้นตอนการขอรับบริการในส่วนของผู้ใช้งานทั่วไป	57
3.11 ผังขั้นตอนการขอรับบริการในส่วนของผู้ดูแลระบบ	58
3.12 ผังแสดง Use Case ของ NU-NET Helpdesk.....	59
3.13 โครงสร้างของเว็บเพจ	61
3.14 ผังแสดง Site Map ของเว็บไซต์ระบบ NU-NET Help Desk	62
4.1 หน้าสำหรับเจ้าหน้าที่ Login เข้าสู่ระบบ	65
4.2 แสดงข้อมูลภาระงานของพนักงาน	66
4.3 แสดงการส่งต่อภาระงานและการปิดงาน	67
4.4 แสดงการรายงานผลการดำเนินงานก่อนการส่งต่องานหรือปิดงาน	68
4.5 แสดงข้อมูลการประมวลผลการงานของพนักงานสำหรับ Hdsa	69
4.6 แสดงแบบฟอร์มบันทึกข้อมูล Service Request ของผู้ขอใช้บริการ	70
4.7 แบบฟอร์มใบงาน	71
4.8 ข้อมูลประวัติการซ่อมของอุปกรณ์	72

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.9 ข้อมูลรายละเอียดประวัติการซ่อมของอุปกรณ์	72
4.10 แสดงรายการข่าว	73
4.11 แสดงรายละเอียดของข่าว	73
4.12 หน้าแรกของเว็บไซต์	74
4.13 ข้อมูลการติดตามการดำเนินงานของเจ้าหน้าที่	75
4.14 รายละเอียดการดำเนินงาน	75
4.15 หน้าลงทะเบียนสมาชิก	76
4.16 หน้ากระดานสนทนา	77
4.17 หน้าตั้งหัวข้อกระทู้	78
4.18 หน้าแสดงความคิดเห็น	79
4.19 บริหารกระดานสนทนา	80
4.20 บริหารข้อมูลข่าว	81
4.21 บริหารข้อมูลประเภทปัญหา	82
4.22 บริหารข้อมูลอุปกรณ์	83
4.23 บริหารข้อมูลการซ่อมบำรุง	84
4.24 บริหารข้อมูลพนักงาน	85
4.25 บริหารข้อมูลทีมงาน	86
4.26 โปรแกรมสืบค้นข้อมูล	87
4.27 แสดงการสืบค้นปัญหา กรณีที่ปัญหาอยู่ Tree อื่น	88

บทที่ 1

บทนำ

1.1 ความสำคัญและที่มาของการศึกษา

ปัจจุบันระบบเครือข่ายคอมพิวเตอร์ในหน่วยงานการศึกษาต่างๆ โดยมีผู้ใช้บริการ ประกอบด้วยนิสิต นักศึกษาและบุคลากรเป็นจำนวนมาก และมีแนวโน้มที่จะมีจำนวนเพิ่มขึ้น จึงมีความจำเป็นอย่างยิ่งที่จะต้องมีการบริหารจัดการระบบเครือข่ายคอมพิวเตอร์ที่มี ประสิทธิภาพและรวดเร็ว ในการให้บริการย่อมมีเหตุต่างๆ เกิดขึ้นได้เสมอ ทั้งในด้านฮาร์ดแวร์ (Hardware), ซอฟต์แวร์ (Software) หรือการใช้งาน การให้บริการการแก้ปัญหาการใช้งาน ให้ ความรู้ ให้คำปรึกษาแก่ผู้ใช้งาน ตลอดจนการให้บริการซ่อมบำรุงจึงมีความสำคัญมาก จึงอยากศึกษา ระบบควบคุมเครือข่ายคอมพิวเตอร์ โดยมีงานบริการระบบเครือข่ายคอมพิวเตอร์ทำหน้าที่บริหารจัดการและให้บริการแก้ไขปัญหาการใช้งานต่างๆ ให้แก่ผู้ใช้งานระบบเครือข่ายคอมพิวเตอร์

ในการให้บริการแก้ปัญหา และปรึกษาให้คำแนะนำการใช้งานแก่ผู้ใช้บริการระบบ เครือข่ายคอมพิวเตอร์นี้ ปัจจุบันยังไม่มีประสิทธิภาพเพียงพอ โดยการติดต่อกันต่างๆ ยังเป็นเพียง ระบบแมนนวล (Manual) เท่านั้น เมื่อผู้ใช้บริการต้องการขอคำปรึกษาหรือแจ้งปัญหาการใช้งาน ต่างๆ จะทำได้เพียงการโทรศัพท์หรือเข้ามาติดต่อโดยตรงที่สำนักงานซึ่งมีความไม่สะดวก อีกทั้ง การประสานงานของเจ้าหน้าที่ยังไม่มีประสิทธิภาพเท่าที่ควร

เพื่อให้การประสานงานของเจ้าหน้าที่และการให้บริการแก่ผู้ใช้งานอย่างมีประสิทธิภาพ มากขึ้น สามารถให้บริการได้ตลอดเวลา และผู้ใช้งานสามารถแก้ไขปัญหาการใช้งานเบื้องต้นได้ด้วย ตนเอง จึงมีแนวคิดจะจัดทำระบบ IT Helpdesk เพื่อช่วยงานบริการผู้ใช้งาน โดยเกี่ยวข้องกับการ แนะนำในด้านต่างๆ ตลอดจนการประสานงานกับเจ้าหน้าที่ การเก็บรวบรวมข้อมูลการใช้งาน

ผู้ศึกษาเห็นว่า การพัฒนาระบบ IT Helpdesk นี้ มีประโยชน์ต่องานให้บริการระบบ เครือข่าย จึงเลือกที่จะทำการศึกษา โดยหวังว่าเมื่อสำเร็จแล้ว นอกจากจะเป็นการพัฒนา ความสามารถ ยังเป็นประโยชน์ต่อหน่วยงานและส่วนรวมอีกด้วย

1.2 วัตถุประสงค์ของโครงการ

1.2.1 เพื่อพัฒนาระบบ IT Helpdesk สำหรับใช้ในงานบริการผู้ใช้งานระบบเครือข่ายคอมพิวเตอร์ในมหาวิทยาลัยนเรศวร

1.2.2 เพื่อศึกษาวิธีการ และเครื่องมือต่างๆ ในการพัฒนา Web Application ด้วยเทคโนโลยี .NET Framework โดยมีเครื่องมือ ได้แก่ Visual Studio.NET, ASP.NET และ MS SQL Server 2005 รวมทั้งแนวคิดการเขียนโปรแกรมเชิงวัตถุ (Object-Oriented Programming)

1.3 ขอบเขตของโครงการ

ทำการพัฒนาระบบ IT Helpdesk สำหรับ หน่วยบริการผู้ใช้งาน งานบริการระบบเครือข่ายคอมพิวเตอร์ โดยมีผู้ให้บริการ ได้แก่ นิสิต นักศึกษาและบุคลากรของหน่วยงานการศึกษาต่างๆ องค์ประกอบระบบ มีดังนี้

- 1.3.1 การแนะนำการใช้งานและแก้ปัญหาการใช้งานระบบเครือข่ายเบื้องต้น
- 1.3.2 การแจ้งขอใช้บริการซ่อมบำรุงนอกสถานที่
- 1.3.3 การประสานงานและจัดลำดับงานของเจ้าหน้าที่ที่ให้บริการระบบเครือข่าย
- 1.3.4 การรับความคิดเห็นและข้อร้องเรียน
- 1.3.5 การรายงานสถิติการให้บริการ

1.4 ขั้นตอนของการดำเนินงาน

1.4.1 ศึกษาระบบการทำงานเดิม เพื่อใช้เป็นแนวทางการสร้างระบบใหม่

1.4.2 ศึกษาข้อมูลการเขียนโปรแกรม Visual Studio.NET, ASP.NET,

Microsoft SQL Server 2005

1.4.3 ศึกษาแนวคิดและทฤษฎีที่เกี่ยวข้องกับการพัฒนาระบบ IT Helpdesk

1.4.4 วิเคราะห์และออกแบบระบบ IT Helpdesk

1.4.5 เขียนโปรแกรมและทดสอบการใช้งาน

แผนการดำเนินงาน

ตารางที่ 1.1 ตารางการปฏิบัติงาน

กิจกรรม	พ.ย.	ธ.ค.	ม.ค.	ก.พ.
1. ศึกษาระบบการทำงานเดิมของหน่วยงาน บริการผู้ใช้งานระบบเครือข่าย เพื่อใช้เป็น แนวทางในการออกแบบระบบใหม่	↔			
2. ศึกษาแนวคิด ทฤษฎีที่เกี่ยวข้องของตัวอย่าง ระบบ IT Helpdesk รวมทั้งเครื่องมือที่ใช้ใน การพัฒนาระบบ เช่น Visual Studio.NET, ASP.NET, MS SQL Server และ Object- Oriented Programming	↔	↔		
3. วิเคราะห์และออกแบบระบบ		↔	↔	
4. พัฒนาระบบ IT Helpdesk			↔	↔
5. ทดสอบและประเมินผลระบบ				↔
6. สรุปและรายงานผล				↔

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1.5.1 ประโยชน์ต่อผู้ทำการศึกษา

- พัฒนาความรู้ความสามารถในการวิเคราะห์ออกแบบและพัฒนาระบบสารสนเทศและฐานข้อมูล
- เพิ่มความรู้ความเข้าใจวิธีการเขียนโปรแกรมเชิงวัตถุ
- เรียนรู้ระบบ IT Helpdesk
- เพิ่มความรู้ความสามารถในการบริหารจัดการงาน

1.5.2 ประโยชน์ต่อองค์กร

- ลดภาระงานและค่าใช้จ่ายในการให้บริการ
- ลดความซ้ำซ้อนและเพิ่มประสิทธิภาพการให้บริการ
- สามารถรองรับการใช้งานของผู้ใช้งานที่เพิ่มขึ้นได้อย่างทั่วถึง
- มีข้อมูลปัญหา และการให้บริการ เพื่อใช้ในการวางแผนพัฒนาการให้บริการระบบเครือข่ายคอมพิวเตอร์ต่อไป

1.5.3 ประโยชน์ต่อผู้ใช้งาน

- ผู้ใช้งานได้รับบริการที่สะดวก รวดเร็วขึ้น สามารถใช้งานระบบได้ตลอดเวลา
- ตอบสนองความต้องการของผู้ใช้ได้ดีขึ้น

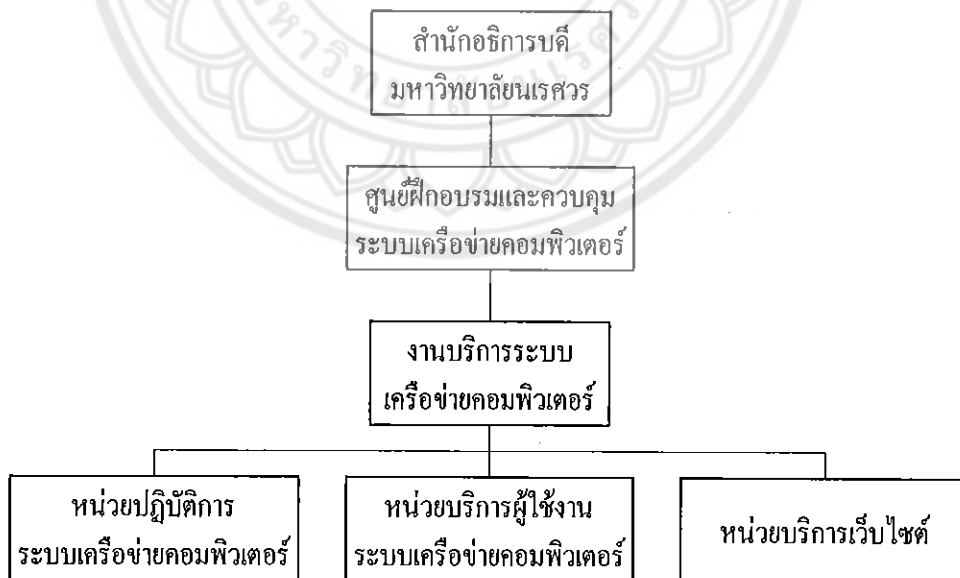
บทที่ 2

แนวคิดและทฤษฎีที่เกี่ยวข้อง

2.1 การให้บริการระบบเครือข่ายคอมพิวเตอร์ในมหาวิทยาลัยนเรศวร

ในการติดตั้งระบบเครือข่ายของหน่วยงานการศึกษาจะต้องทำการเชื่อมโยงทรัพยากรคอมพิวเตอร์ และอุปกรณ์ติดต่อสื่อสารจากเครือข่ายหลักรวมเข้ากับเครือข่ายที่พัฒนาขึ้นให้เป็นระบบใหญ่ และสามารถใช้ทรัพยากรร่วมกันได้นั้น โดยทำให้ผู้ใช้งานสามารถสืบค้นข้อมูลสารสนเทศเพื่อการเรียนการสอน และเพื่อการใช้งาน รวมไปถึงการบริหารงานภายในหน่วยงานการศึกษาได้อย่างมีประสิทธิภาพ และในปัจจุบันมีผู้ใช้งานบริการระบบเครือข่าย เช่น นิสิตนักศึกษา รวมไปถึงบุคลากรของหน่วยงานที่มีเป็นจำนวนมาก ซึ่งสามารถใช้บริการได้ทั้งในรูปแบบของระบบเครือข่ายท้องถิ่น (LAN) และการเชื่อมต่อจากภายนอก (Remote Access) ผ่านทางระบบโทรศัพท์ (Dialup Modem) โดยผู้ใช้งานแต่ละคนจะได้รับ User ID และ Password คนละ 1 ชุด เพื่อใช้ในการ Log In ใช้งานระบบเครือข่ายภายในหน่วยงานการศึกษานั้นๆ

การบริหารจัดการระบบเครือข่ายคอมพิวเตอร์ หรือ “การบริการระบบเครือข่ายคอมพิวเตอร์” โดยแบ่งออกเป็น 3 หน่วย คือ หน่วยปฏิบัติการระบบเครือข่าย หน่วยบริการผู้ใช้งาน และหน่วยบริการเว็บไซต์



รูปที่ 2.1 ผังโครงสร้างการบริหารงานบริการระบบเครือข่ายคอมพิวเตอร์
มหาวิทยาลัยนเรศวร

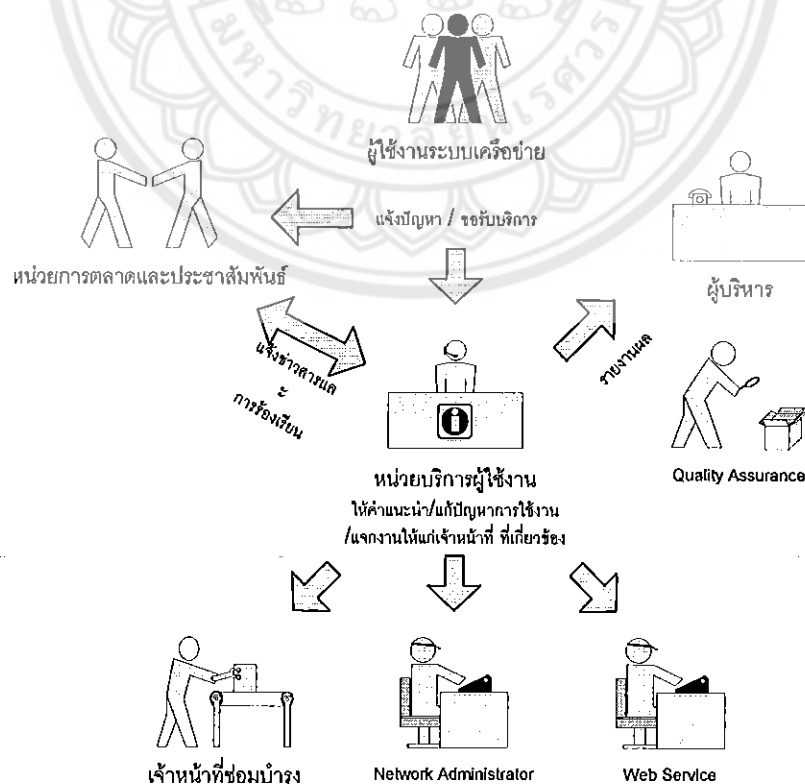
2.1.1 วัตถุประสงค์ของระบบเครือข่ายคอมพิวเตอร์

- ตอบสนองความต้องการเชื่อมโยงเครือข่ายคอมพิวเตอร์ระหว่างหน่วยงาน กับ คณะต่างๆ ในลักษณะเครือข่ายท้องถิ่น (Local Area Network: LAN)
- ออกแบบติดตั้งศูนย์ควบคุมระบบเครือข่ายข้อมูลภายใน (Campus Network)
- ฝึกอบรมคอมพิวเตอร์ให้กับบุคลากรของมหาวิทยาลัย (Computer Training - Center)
- พัฒนาการศึกษาระบบของมหาวิทยาลัยในการขยายโอกาสการเรียนรู้ไปสู่วิทยาเขตอื่นๆ อย่างทั่วถึง

2.1.2 ระบบให้บริการผู้ใช้งานระบบเครือข่ายคอมพิวเตอร์

หน่วยงานบริการผู้ใช้งาน ทำหน้าที่ให้คำปรึกษาและแก้ไขปัญหาการใช้งานระบบเครือข่าย รวมทั้งให้บริการซ่อมบำรุงระบบเครือข่ายและอุปกรณ์คอมพิวเตอร์ที่เกี่ยวข้อง (Helpdesk and Maintenance Service) โดยผู้ใช้งานระบบเครือข่ายสามารถติดต่อขอใช้บริการหรือแจ้งผ่านหน่วยงานประชาสัมพันธ์

นอกจากนี้หน่วยบริการผู้ใช้งานจะประสานงานกับหน่วยการตลาดและประชาสัมพันธ์ในเรื่องที่เกี่ยวกับการรื้อเรียนจากผู้ใช้งานระบบเครือข่าย และการแจ้งข่าวสารประชาสัมพันธ์ รวมทั้งต้องรายงานผลให้แก่หน่วยประกันคุณภาพและผู้บริหารทราบด้วย



รูปที่ 2.2 หน่วยบริการผู้ใช้งานและผู้ที่เกี่ยวข้องงานบริการเครือข่ายคอมพิวเตอร์

2.2 การจัดทำและบริหารระบบ IT Helpdesk

ปัจจุบันเมื่อมีการนำเทคโนโลยีสารสนเทศมาใช้มากขึ้น ปัญหาที่เกิดจากเทคโนโลยีสารสนเทศก็มีมากขึ้นตามไปด้วย จึงเกิดหน่วยงานที่เรียกว่า Helpdesk โดยความหมายของ Helpdesk คือเจ้าหน้าที่ที่คอยให้คำปรึกษา, แก้ไขปัญหาในกรณีที่ผู้ใช้มีปัญหาเกี่ยวกับเครื่องคอมพิวเตอร์ หรือระบบงาน คอยวางแผน, ประสานงาน และสนับสนุนกระบวนการทางธุรกิจ (สนับสนุนระบบ และผู้ใช้)

2.2.1 หน้าที่และความรับผิดชอบของ Helpdesk

- เป็นศูนย์กลางในการติดต่อกับผู้ใช้บริการทั้งหมด
- บันทึกข้อมูลปัญหา และวิธีการในการแก้ปัญหา
- จัดให้มีการจัดการกับข้อผิดพลาดที่เกิดขึ้นและติดตามความก้าวหน้าในการแก้ไขตลอดจนจัดหาให้มีผู้รับผิดชอบแก้ไขปัญหาให้เหมาะสม
- เพิ่มประสิทธิภาพในการแก้ไขข้อผิดพลาดหรือปัญหาต่างๆ ที่เกิดขึ้น

ระบบ Helpdesk หรือระบบ Call Service มีการให้บริการมานานแล้วในธุรกิจต่างๆ ส่วนใหญ่ทำหน้าที่ให้คำปรึกษา แก้ไขปัญหาการใช้งาน และรับแจ้งการขอบริการต่างๆ โดยเจ้าหน้าที่ให้บริการ จนกระทั่งเมื่อระบบอินเทอร์เน็ตได้รับการพัฒนามากขึ้นและมีการใช้งานอย่างแพร่หลาย จึงมีการประยุกต์ใช้เทคโนโลยีสารสนเทศกับระบบ Helpdesk กลายเป็นระบบ IT Helpdesk

การจัดทำ IT Helpdesk อาจพัฒนาขึ้นได้เอง หรือใช้ซอฟต์แวร์สำเร็จรูป ทั้งนี้ขึ้นอยู่กับรูปแบบการใช้งานและงบประมาณ โดยทั่วไประบบ IT Helpdesk มักมีส่วนประกอบ เช่น

- การแนะนำหน่วยงาน
- การแจ้งข่าวสาร และการประชาสัมพันธ์
- การแนะนำวิธีการใช้งานและการแก้ไขปัญหาการใช้งานผลิตภัณฑ์ต่างๆ
- การรับข้อเสนอแนะและการร้องเรียนจากลูกค้า
- การตรวจสอบสถานการณ์ให้บริการ การสั่งซื้อ หรือการจัดส่งสินค้า
- การสนับสนุนลูกค้าในรูปแบบต่างๆ

2.2.2 หัวข้อที่ควรพิจารณาในการบริหารระบบ IT Helpdesk ได้แก่

- ประเภท รูปแบบและเนื้อหาของบริการที่จะให้มีระบบ Helpdesk (Type Format and Context)
- ความต้องการของลูกค้าและการบริหารความสัมพันธ์กับลูกค้า (Customer Requirement and Customer Relation Management)
- สถิติของปัญหาที่ผ่านมา (Trouble Statistic)
- ทรัพยากร เทคโนโลยี และเครื่องมือที่ต้องการ (Resource, Technology and Tools)

- การจัดเตรียมและฝึกอบรมเจ้าหน้าที่ (Requirement and Training)
- ระบบการทำงานและจัดลำดับการให้บริการ (Work Flow and Queuing)
- การวัดและประเมินผลการให้บริการและผลตอบกลับจากลูกค้า (Measurement and Evaluation)
- ระบบฐานข้อมูล (Database System)
- องค์ความรู้ (Knowledge)

2.3 เทคโนโลยีและเครื่องมือที่เกี่ยวข้อง

2.3.1 แนวคิดการพัฒนาระบบคอมพิวเตอร์ด้วยหลักการ Object Orientation

แนวคิดเชิงวัตถุ (Object Oriented) หมายถึง การใช้ Object เป็นตัวหลักในการพิจารณาความเป็นจริงต่างๆที่เกิดขึ้นในโลก โดยมองทุกสิ่งในโลกเป็นวัตถุทั้งหมด และมองว่ากิจกรรมที่เกิดขึ้นในโลกนี้เกิดจากความสัมพันธ์และปฏิสัมพันธ์ระหว่างวัตถุ

ตารางที่ 2.1 เปรียบเทียบการวิเคราะห์และออกแบบระบบวิธีเดิมกับวิธีเชิงวัตถุ

วิธีเดิม	วิธีเชิงวัตถุ
เริ่มต้นจากการวิเคราะห์เอกสารผลลัพธ์และการทำงานของระบบงานเดิม	เริ่มต้นการวิเคราะห์จาก Object ที่สามารถเห็นได้ชัดเจน
แตกการทำงานออกเป็นหน่วยย่อยๆ	แบ่งกลุ่มของ Object ตามคุณลักษณะ
องค์ประกอบต่างๆของระบบ เช่น การประมวลผล การออกรายงาน การคำนวณ จะเกี่ยวพันกัน การเปลี่ยนแปลงจะกระทบซึ่งกันและกัน	แต่ละ Object เป็นอิสระต่อกัน การเปลี่ยนแปลงจะไม่กระทบกัน
การปรับเปลี่ยนระบบต้องแก้ไข Source Code	การปรับเปลี่ยนระบบ ทำได้โดยการเปลี่ยน Attributes, Functions ของ Object
Tools ที่สนับสนุนมีน้อยลง	Tools ที่สนับสนุนมีมากขึ้น

(อ้างอิงจาก หนังสือ Object-Oriented Software Engineering)

2.3.1.1 วัตถุ (Objects)

- วัตถุ (Object) คือหน่วยสนใจของระบบที่ทำให้เกิดเหตุการณ์ วัตถุเป็นได้ทั้งสิ่งที่สามารถจับต้องได้ (เช่น โต๊ะ รถยนต์ คอมพิวเตอร์ คน) และวัตถุที่ไม่สามารถจับต้องได้ (เช่น บริษัท ฝ่ายต่างๆ หลักสูตร)
- การสื่อสารระหว่างวัตถุเรียกว่า Message

2.3.1.2 Class (Class)

- Class คือกลุ่มของ Object ที่มีโครงสร้างพื้นฐานพฤติกรรมเดียวกัน Object ที่มีคุณสมบัติเดียวกัน ก็จะรวมกลุ่มอยู่ใน Class เดียวกัน
- Class และ Object มีความคล้ายกันมากจนทำให้หลายคนสงสัยว่าเป็นสิ่งเดียวกันหรือไม่ ในความเป็นจริง Class ถือว่าเป็น นามธรรม (Abstract) ในขณะที่ Object นั้นเป็นสิ่งที่มีความตัวตน (Concrete) กล่าวคือ Class เป็นเหมือนพิมพ์เขียวของ Object โดยที่ Class จะไม่สามารถทำงานได้ แต่ Object สามารถทำงานได้
- การทำงานของ Object จะเป็นไปตามคุณสมบัติที่กำหนดไว้ใน Class และ Object ทุกตัวก็ต้องอยู่ใน Class ดังนั้น Class และ Object จึงเป็นสิ่งที่คู่กันเสมอ
- Class นอกจากจะมีชื่อ Class กำกับแล้ว ยังมี คุณสมบัติ (Attributes) และ หน้าที่การทำงาน (Operations หรือ Methods)

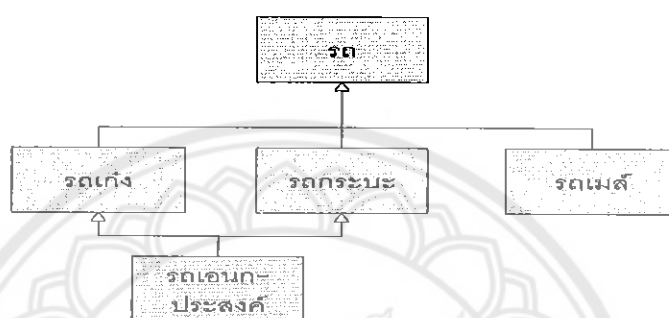
Student
StudentID
FirstName
LastName
Address
Telephone
Birthdate
Faculty
GPA
Register()
Drop()
Withdraw()

(อ้างอิงจาก หนังสือ Object-Oriented Software Engineering)

รูปที่ 2.3 ตัวอย่าง Class นักศึกษา

2.3.1.3 การสืบทอดคุณสมบัติ (Inheritance)

- การสืบทอดคุณสมบัติ คือ การที่ Subclass ได้รับการถ่ายทอดคุณสมบัติ มาจาก Superclass แล้วผนวกคุณสมบัติพิเศษเพิ่มเข้าไป
- สัญลักษณ์ที่ใช้ คือ ลูกศรหัวรูปสามเหลี่ยมชี้จาก Subclass ไปยัง SuperClass
- หลักของการสืบทอดคุณสมบัติจะทำให้ความสัมพันธ์ระหว่าง Object มีความชัดเจนยิ่งขึ้น กล่าวคือถ้ามีความสัมพันธ์ที่ชัดเจนมากขึ้นเท่าใด จะส่งผลให้การออกแบบระบบงานง่ายขึ้น



(อ้างอิงจาก หนังสือ Object-Oriented Software Engineering)

รูปที่ 2.4 การถ่ายทอดคุณสมบัติแบบ Multiple Inheritances

2.3.1.4 ความสัมพันธ์ระหว่าง Object ประกอบด้วย

2.3.1.4.1 Association

เป็นความสัมพันธ์ระหว่าง Object หรือ Class แบบ 2 ทิศทาง

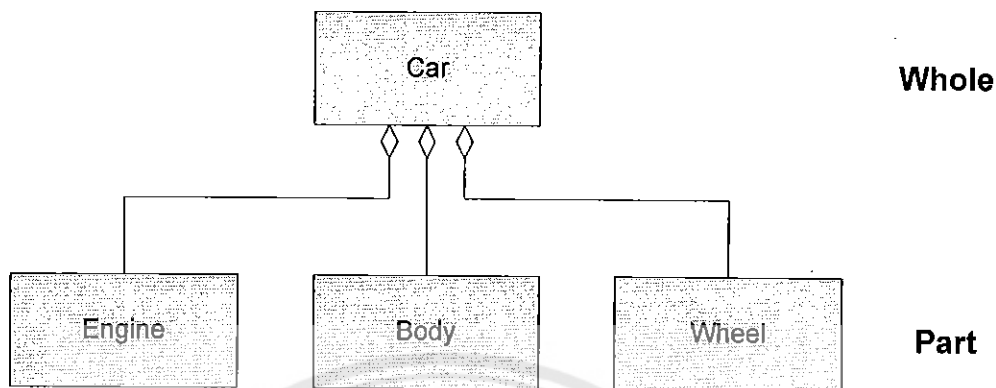


(อ้างอิงจาก หนังสือ Object-Oriented Software Engineering)

รูปที่ 2.5 ความสัมพันธ์แบบ 2 ทิศทางของ Object หรือ Class แบบ Association

2.3.1.4.2 Aggregation

เป็นความสัมพันธ์ระหว่าง Object หรือ Class แบบ “Whole-Part” หรือ “is part of” โดยจะมี Class ที่ใหญ่ที่สุดที่เป็น Object หลัก และมี Class อื่นเป็นส่วนประกอบ

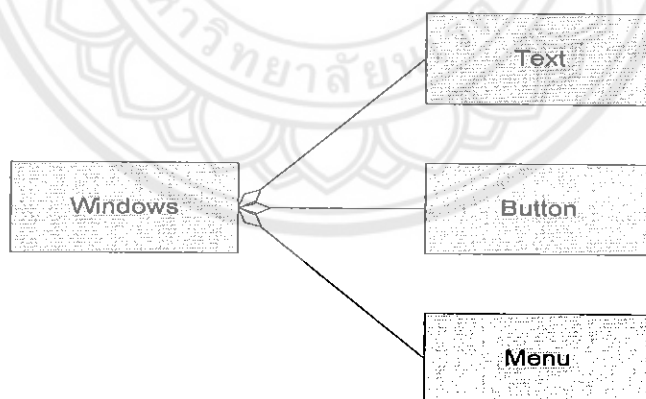


(อ้างอิงจาก หนังสือ Object-Oriented Software Engineering)

รูปที่ 2.6 ความสัมพันธ์ระหว่าง Object หรือ Class แบบ Aggregation

2.3.1.4.3 Composition

เป็นความสัมพันธ์ระหว่าง Object หรือ Class แบบขึ้นต่อกันและมีความเกี่ยวข้องกันเสมอ โดยจะมี Class ซึ่งเป็นองค์ประกอบของ Class อื่นที่ใหญ่กว่า เมื่อ Class ที่ใหญ่กว่าถูกทำลาย Class ที่เป็นองค์ประกอบก็จะถูกทำลายไปด้วย

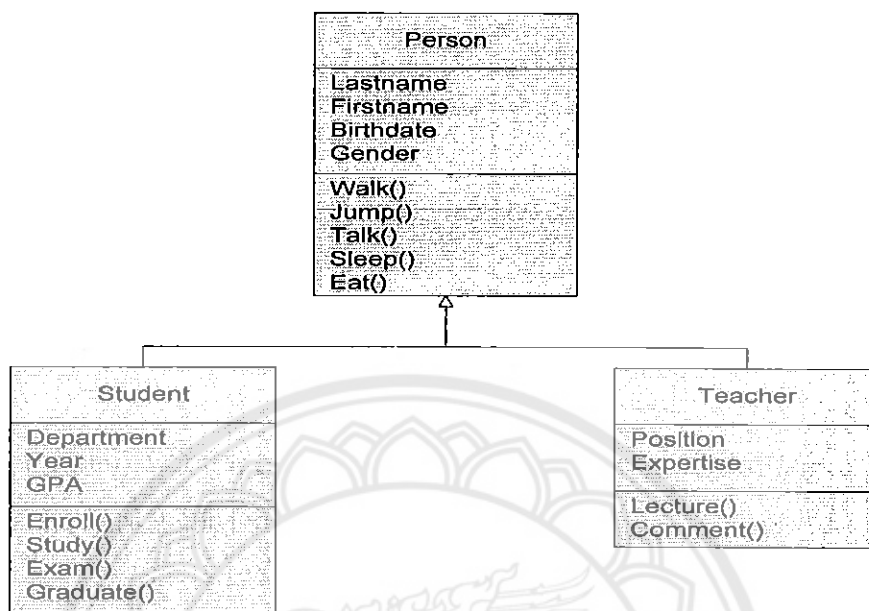


(อ้างอิงจาก หนังสือ Object-Oriented Software Engineering)

รูปที่ 2.7 ความสัมพันธ์ระหว่าง Object หรือ Class แบบ Composition

2.3.1.4.4 Generalization

เป็นความสัมพันธ์ระหว่าง Object หรือ Class ในลักษณะของการสืบทอดคุณสมบัติจาก Class หนึ่ง (Superclass) ไปยังอีก Class หนึ่ง (Subclass)



(อ้างอิงจาก หนังสือ Object-Oriented Software Engineering)

รูปที่ 2.8 ความสัมพันธ์ระหว่าง Object หรือ Class แบบ Generalization

2.3.2 UML (Unified Modeling Language)

UML คือโมเดลมาตรฐานที่ใช้หลักการออกแบบ OOAD (Object-Oriented Analysis and Design) รูปแบบของภาษา UML จะมี Notation ซึ่งเป็นสัญลักษณ์ที่นำไปใช้ในโมเดลต่างๆ UML จะมีข้อกำหนดกฎระเบียบต่างๆในการโปรแกรม โดยกฎระเบียบต่างๆ จะมีความหมายต่อการเขียนโปรแกรม ดังนั้นการใช้ UML จะต้องทราบความหมายของ Notation ต่างๆ เช่น Generalize, Association, Dependency, Class และ Package สิ่งเหล่านี้มีความจำเป็นอย่างยิ่งต่อความดีความของการออกแบบระบบก่อนนำไป Implement ระบบงานจริง ในปัจจุบันมีเครื่องมือมากมายที่สามารถแปลงโมเดล UML เป็นโค้ดภาษาต่างๆ ยกตัวอย่างเช่น JAVA, C# และ Visual Basic เป็นต้น

2.3.2.1 เครื่องมือในการเขียน UML

UML อาจใช้โปรแกรมสำหรับการวาดรูปต่างๆ เช่น Paint, Photoshop, PowerPoint, Visio หรือโปรแกรมอะไรก็ตามที่สามารถวาดรูปได้ นอกจากการใช้โปรแกรมแล้ว การวาดรูปลงบนกระดาษก็สามารถใช้ได้เช่นเดียวกัน อย่างไรก็ตามวิธีการต่างๆเหล่านี้ จะมีเครื่องอำนวยความสะดวกในการทำงานให้ ดังนั้น จึงมีผู้ผลิตซอฟต์แวร์หลายค่ายได้ทำการสร้างเครื่องมือช่วยออกแบบ (Case Tool) สำหรับทำงานกับ UML โดยเฉพาะขึ้นมากมาย เช่น Rational Rose, Borland Together, Visual UML หรือ Magic Draw เป็นต้น โดยเครื่องมือเหล่านี้จะสามารถทำการ

ออกแบบ UML โปรแกรมต่างๆและทำการนำเข้าโค้ดกลับเข้ามาเพื่อให้อยู่ในรูปโมเดลได้ เรียกว่า Generate Code และการ Import กลับเข้ามาอยู่ในโมเดล UML ว่า “Round Trip Engineer” กระบวนการนี้มีประโยชน์มากในการอัปเดตโมเดลและโค้ดให้ตรงกันเสมอ ซึ่งจะต้องมีการควบคุมให้ดี เนื่องจากจะเป็นการทำงานร่วมกันระหว่างนักออกแบบระบบ (UML) และ Developer (Coding)

2.3.2.2 Visual Modeling

UML เป็น Visual Modeling ซึ่งจะทำหน้าที่ในการแสดงโครงสร้างการทำงานของซอฟต์แวร์ให้ออกมาในโมเดลที่สามารถมองเห็นได้โดยการสื่อออกมาในรูปของ Diagram ด้วยวิธีการแบบนี้จะทำให้ Model, Implement และ Coding มีความสอดคล้องกันไปไปในแนวทางเดียวกัน

UML ถูกนำมาใช้ในการพัฒนาซอฟต์แวร์อย่างกว้างขวาง และทำให้การทำงานมีคุณภาพ กล่าวคือ

1. ช่วยลดระยะเวลาในการพัฒนาระบบงาน
2. ช่วยเพิ่มความสามารถในการทำงาน
3. ช่วยเพิ่มคุณภาพของระบบงาน
4. รองรับระบบงานเดิม
5. ช่วยในการสื่อสารระหว่างทีมผู้พัฒนาระบบงาน

ข้อดีของ UML คือ

1. สามารถแสดงส่วนประกอบในการสร้างโปรเจกต์ในรูปของ OOP
2. เชื่อมแนวคิดกับการออกแบบระบบโดยใช้ Object-Oriented Code
3. ง่ายต่อการทำความเข้าใจและสามารถแปลงเป็นโค้ดโปรแกรมได้

2.3.2.3 Use Case Driven

กระบวนการการพัฒนาโครงการใดๆนั้นมีวัตถุประสงค์เพื่อทำการสนับสนุนการทำงานของ User เป็นหลัก หรือสนับสนุนความต้องการของผู้ใช้ ทั้งนี้ User ไม่ได้หมายถึงคนเพียงอย่างเดียว อาจหมายถึงระบบอื่นที่อยู่ภายนอกระบบก็ได้ User จะมีการทำกิจกรรมใดๆต่อระบบ เช่น การสอดบัตรพลาสติกลงในเครื่องอ่านบัตร การกดปุ่มตอบสนองต่อหน้าจอที่ปรากฏต่อผู้ใช้ การใช้งานเครื่องเอทีเอ็ม (Automatic Teller Machine) และการรับเงินที่เครื่องส่งออกมา โดยเอทีเอ็มหรือระบบจะมีการสร้างลำดับของงานขึ้นมาก่อน จากนั้นจึงส่งผลลัพธ์ให้ User หรือผู้ใช้กระบวนการตอบสนองของระบบอย่างมีลำดับ (Sequence) เพื่อให้ได้งานตามต้องการในลักษณะนี้เรียกว่า “Use Case”

Use Case จึงเป็นฟังก์ชันการทำงานของระบบที่กำหนดให้ผลลัพธ์ หรืองานตามที่ User ต้องการ (Capture Functional Requirement) นักพัฒนาระบบจะนำเอา Use Case มาสร้างเป็น โมเดล ที่สามารถอธิบายฟังก์ชันการทำงานที่สมบูรณ์ของระบบต่อไป

อย่างไรก็ตาม Use Case ไม่ได้เป็นแค่เพียงสิ่งที่บอกถึงความต้องการของระบบเท่านั้น แต่ ยังมีความสำคัญอย่างมากต่อการออกแบบ, การสนับสนุน, การทดสอบและการติดตั้งใช้งานระบบ จาก Use Case Model ที่เกิดขึ้น Developer จะทำการสร้างลำดับของการออกแบบและพัฒนาระบบ, การแตกรายละเอียด หรือ Use Case Driven ดังนั้น Use Case Driven จึงหมายถึงกระบวนการ การพัฒนาระบบโดยมีการไหลของงานเกิดจากการใช้ Use Case เป็นตัวกำหนด Use Case และดำเนิน ไปเป็นวงจร

2.3.2.4 Interactive and Incremental Development

ในโลกแห่งความเป็นจริง การทำงานไม่สามารถทำให้เสร็จได้ในขั้นตอนเดียว และวิธีการทำงานโดย “Water Fall Model” ซึ่งเป็นวิธีการทำงานแบบดั้งเดิมจะต้องทำงานให้เสร็จในรอบเดียว นับว่าเป็นความเสี่ยงอย่างยิ่งต่อความล้มเหลวของโครงการ เนื่องจากการทำงานในยุคปัจจุบันมักมีการเปลี่ยนแปลงสิ่งแวดล้อมต่างๆเกิดขึ้นเสมอ เช่น เปลี่ยนแปลงความต้องการ เปลี่ยนแปลง เทคโนโลยี เป็นต้น ดังนั้นจะเห็นว่าในสภาพการทำงานจริงๆ มักมีการทำงานแบบแก้ไขใหม่วนซ้ำเรื่อยๆ นั่นก็คือ กระบวนการ Iteration นั่นเอง (การวนทำซ้ำใหม่เมื่อพบข้อผิดพลาดหรือบกพร่อง) วิธีการนี้จึงมีความสำคัญมาก นอกจากนี้ยังพบว่าการทำงานโดยการค่อยๆ เพิ่มงานเข้าไปในงานเดิม ที่ทำเสร็จแล้วเรื่อยๆ โดยการแบ่งขอยงานใหญ่ๆ ออกเป็นงานย่อยๆ เมื่อเกิดข้อผิดพลาดขึ้นจะมีผลกระทบต่องานเพียงส่วนย่อยที่กำลังดำเนินการอยู่เท่านั้น ไม่ใช่กระทบหมดทั้งโครงการ วิธีการทำงานเช่นนี้เรียกว่าการทำงานแบบ Incremental จะเห็นว่าการทำงานแบบการเพิ่มระดับการทำงาน และกระบวนการทำซ้ำจะให้ผลดีกว่าการทำงานแบบดั้งเดิม หรือ “Water Fall Model” (มีความเสี่ยงมากกว่าเนื่องจากหากพบข้อผิดพลาดขึ้น จะมีผลกระทบต่อโครงการมากกว่า) ดังนั้นในยุคปัจจุบัน จึงนิยมที่จะทำงานโดยอาศัยหลักการเพิ่มระดับการทำงานและกระบวนการทำซ้ำ

บ่อยครั้งที่ กระบวนการเพิ่มระดับการทำงานและกระบวนการทำซ้ำ ถูกนำมาใช้กันอย่าง สลับสน แต่มันมีข้อแตกต่างกันกล่าวคือ กระบวนการเพิ่มระดับการทำงาน จะเป็นการทำงานโดย อาศัยการเพิ่มเติมส่วนงานเข้าไปในงานเดิมเรื่อยๆ เพื่อให้มีความสมบูรณ์มากขึ้น เช่น การสร้างบ้าน ที่ค่อยๆ มีการต่อเติมส่วนประกอบย่อยเข้าไปหลังจากทำงานส่วนหนึ่งเสร็จไปแล้ว เช่น สร้าง หอ้งน้ำ ต่อด้วยห้องนอน ประตู เป็นต้น

การทำงานในปัจจุบันมักพบว่าโปรเจกต์ต่างๆ จะมีขนาดใหญ่ ดังนั้นจึงนิยมทำการแบ่งงาน ออกเป็นโปรเจกต์ย่อยๆ หลายๆ ส่วน แต่ละส่วนก็จะมีการทำ Iteration และการเพิ่มเติมเข้าไปใน

รายละเอียดหรืองานเข้าไปในงานเดิม หรือ Incremental การแบ่งส่วนงานออกเป็นงานย่อยจะส่งผลดี คือ

- การแบ่งส่วนงานจะสามารถทำงานร่วมกับ Use Case จำนวนมากได้
- ลดความเสี่ยงในการทำงาน

2.3.2.5 Use Case Analysis Model

วัตถุประสงค์ของ Analysis Model

- ต้องการแยกการวิเคราะห์ระบบ
- ต้องการภาพรวมของระบบ
- ศึกษาเพื่อสร้างระบบที่เกี่ยวข้องกับระบบ หรือสิ่งที่มีอยู่แล้ว

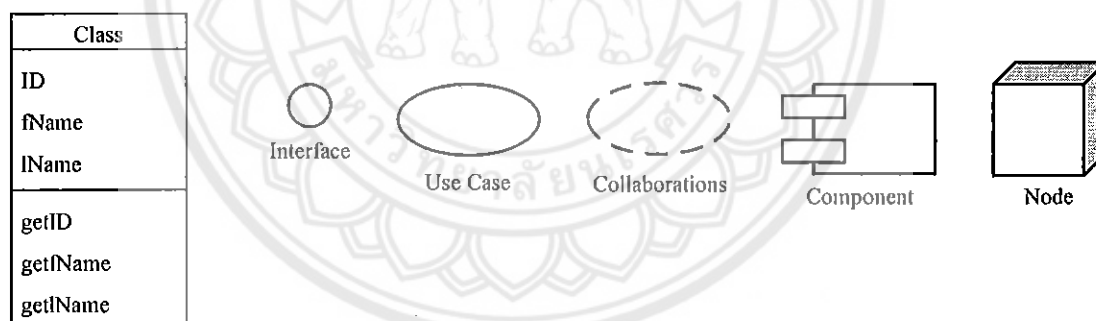
2.3.3 องค์ประกอบของ UML

UML สามารถแบ่งองค์ประกอบออกเป็น 3 กลุ่ม ดังนี้

2.3.3.1 Things คือ สัญลักษณ์หรือสิ่งต่างๆที่นำมาใช้สร้าง UML Diagram แบ่งออกเป็น 4

หมวดคือ

1. Structural Things หรือหมวดโครงสร้าง เป็นคำนำมาใช้สำหรับ UML ส่วนใหญ่ จะเป็นส่วน Static ได้แก่ Use Case, Interface, Class, Collaboration, Component, Node



(อ้างอิงจาก หนังสือ Schaum's Outline of UML)

รูปที่ 2.9 แสดงสัญลักษณ์ต่างๆ ของ UML Diagram

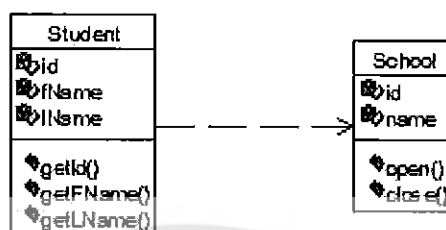
2. Behavioral Things หรือหมวดพฤติกรรม ได้แก่ส่วนที่เป็น Dynamic แสดงถึงพฤติกรรมของระบบ ประกอบด้วยสองส่วนใหญ่ๆ คือ Interaction และ State Machine

3. Grouping Things หรือหมวดการจัดกลุ่มหมู่ ได้แก่ Package

4. Annotation Things หรือหมวดคำอธิบาย ได้แก่ Note

2.3.3.2 Relationships ความสัมพันธ์ใน UML ประกอบไปด้วย

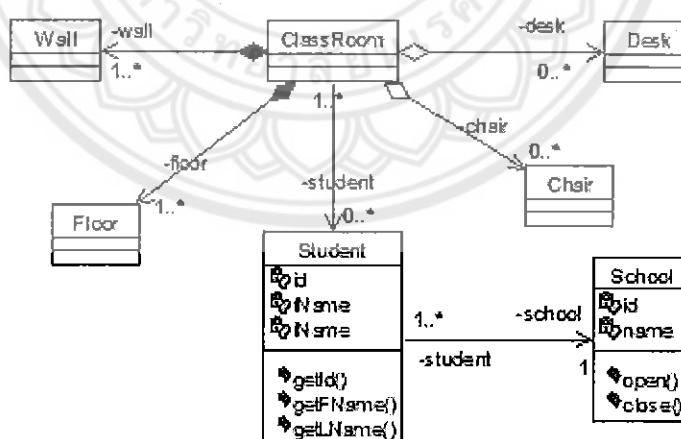
1. Dependency หรือ ความขึ้นอยู่กับกันจะให้ความหมายว่าเมื่อเกิดการเปลี่ยนแปลงที่ส่วนหนึ่งแล้วจะส่งผลกระทบต่ออีกส่วนหนึ่งที่ลากเส้นมาสัมพันธ์กัน เช่น การเปลี่ยนแปลงของโรงเรียนจะมีผลกระทบของนักเรียน เป็นต้น



(อ้างอิงจาก หนังสือ Schaum's Outline of UML)

รูปที่ 2.10 ตัวอย่างความสัมพันธ์แบบ Dependency

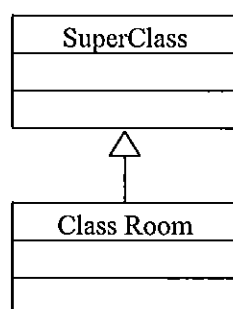
2. Association หรือ ความสัมพันธ์จะแสดงถึงความสัมพันธ์ระหว่าง Object ความสัมพันธ์ยังสามารถแยกความสัมพันธ์ออกเป็นแบบธรรมดา ความสัมพันธ์แบบ Aggregation, Composite, ทิศทางความสัมพันธ์ นอกจากนี้ยังสามารถกำหนด Multiplicity ให้กับความสัมพันธ์ได้อีกด้วย



(อ้างอิงจาก หนังสือ Schaum's Outline of UML)

รูปที่ 2.11 ตัวอย่างความสัมพันธ์แบบ Association

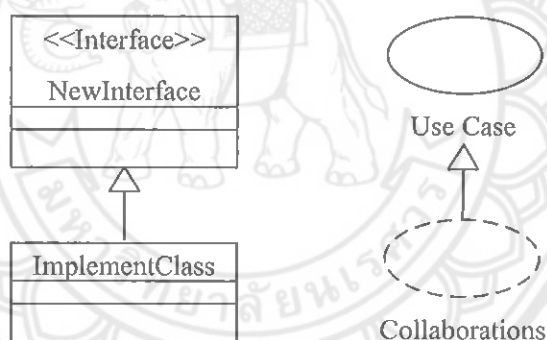
3. Generalization หรือ การสืบทอดคุณสมบัติ หรือ Inheritance นั่นเอง



(อ้างอิงจาก หนังสือ Schaum's Outline of UML)

รูปที่ 2.12 ตัวอย่างความสัมพันธ์แบบ Generalization

4. Realization หรือ การทำให้ทำงานได้จริง เช่น Interface ถูก Realize โดย Class, Use Case ถูก Realize โดย Collaborations



(อ้างอิงจาก หนังสือ Schaum's Outline of UML)

รูปที่ 2.13 ตัวอย่างความสัมพันธ์แบบ Realization

2.3.4 Diagram

Diagram ของ UML แบ่งออกได้เป็น 9 Diagram หลัก ดังนี้

2.3.4.1 Use Case Diagram ในการพัฒนาระบบงานใดๆนั้น การเก็บรวบรวมความต้องการของผู้ใช้มีความสำคัญมาก และจะทำในระยะแรกๆ ของการพัฒนาระบบงานเสมอ Use Case Diagram เป็น Diagram ที่ทำหน้าที่

- เป็นเทคนิคในการสร้างแบบจำลองเพื่อใช้อธิบายหน้าที่ของระบบใหม่ หรือระบบปัจจุบัน
- กระบวนการสร้าง Use Case เป็นแบบ Iteration

- ความต้องการของระบบจะได้จากลูกค้า/ผู้ใช้ และผู้พัฒนาระบบ
- องค์ประกอบจะมี Use Case, Actor, Use Case Relation และ System

Chart ID : Use Case Diagram1
 Chart Name : Use Case Diagram1
 Chart Type : UML Use Case Diagram

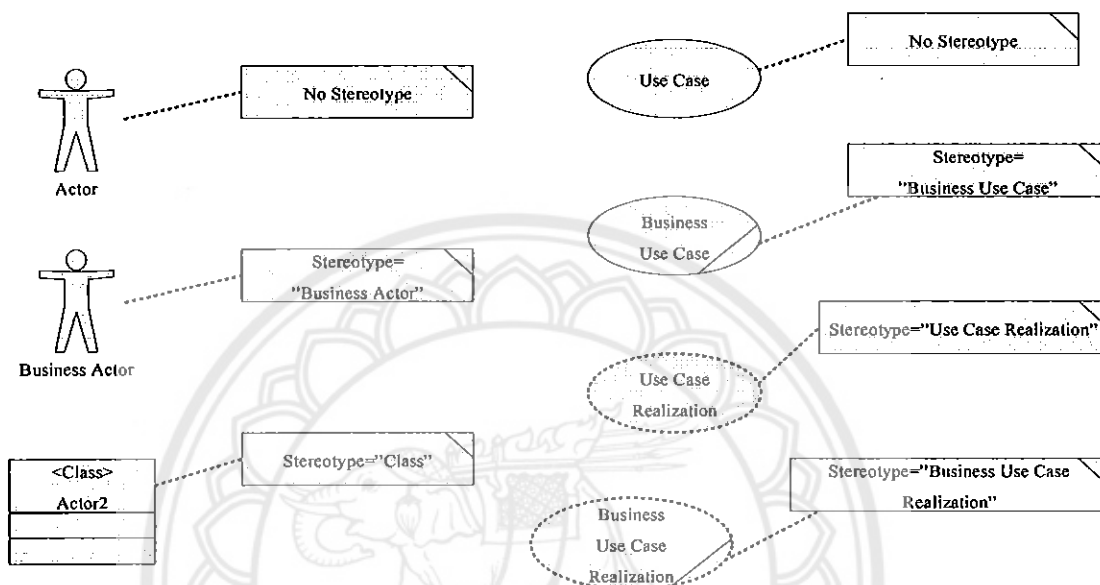
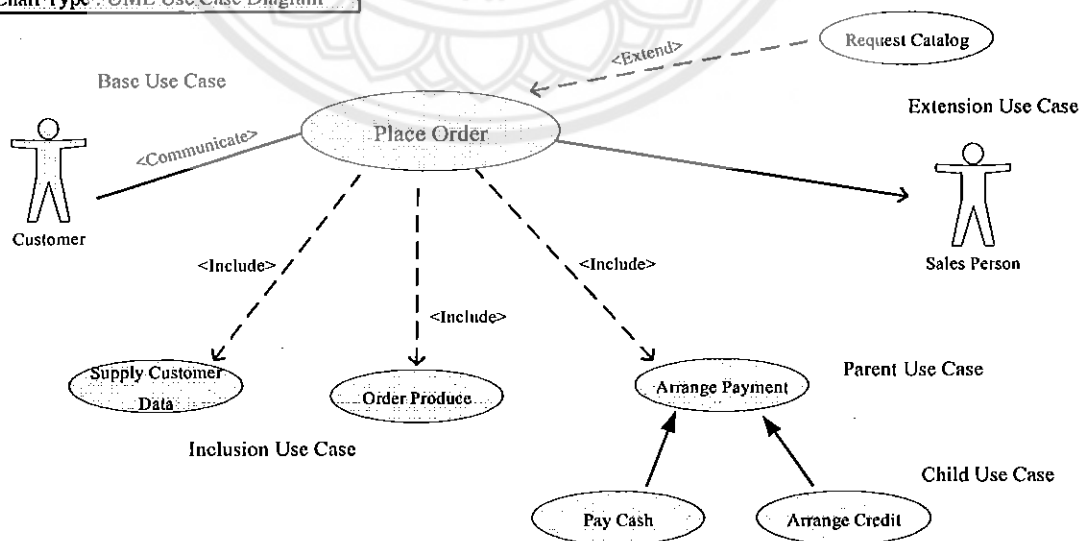


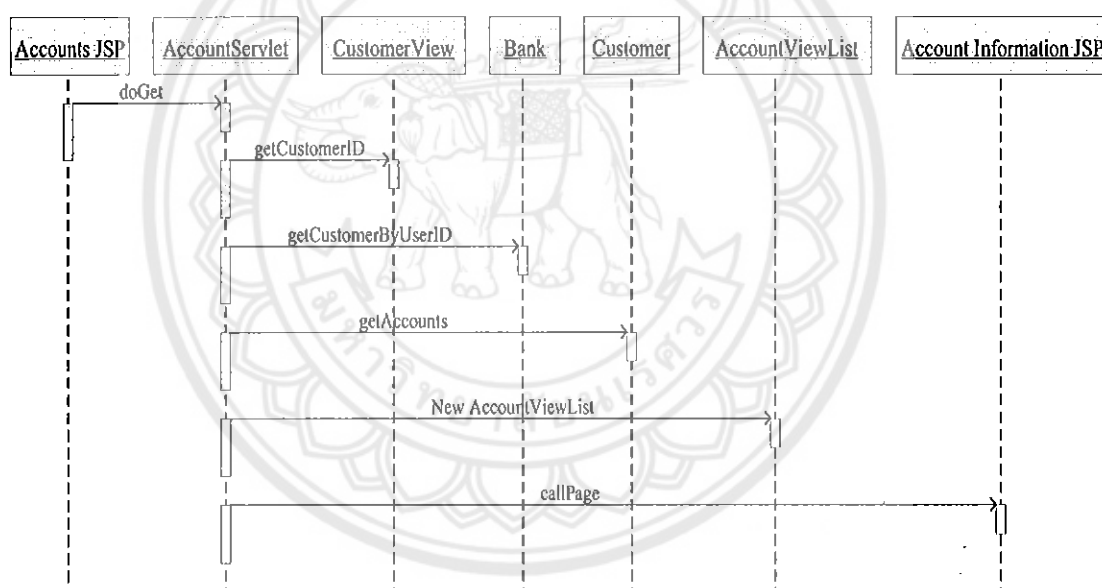
Chart ID : Use Case Relationships
 Chart Name : Use Case Relationships
 Chart Type : UML Use Case Diagram



(อ้างอิงจาก หนังสือ Schaum's Outline of UML)

รูปที่ 2.14 ตัวอย่าง Use Case Diagram ที่สร้างจาก Visual UML

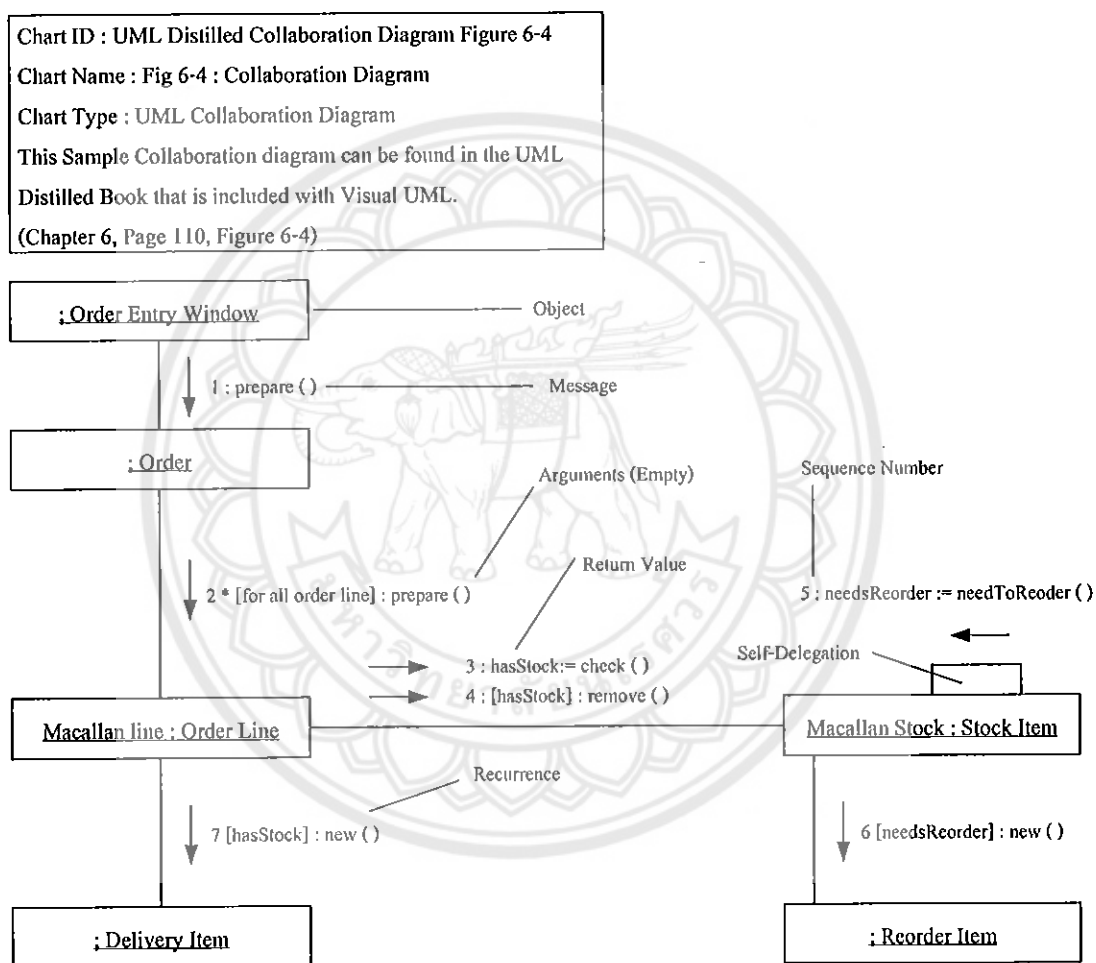
2.3.4.2 Sequence Diagram เป็น Diagram ที่บอกลำดับการทำงานของระบบ โดยมี Object และเวลาเป็นตัวกำหนดลำดับของงาน Sequence Diagram เป็น Diagram ซึ่งแสดงปฏิสัมพันธ์ (Interaction) ระหว่าง Object ตามลำดับของเหตุการณ์ที่เกิดขึ้น ณ เวลาที่กำหนด Message ที่เกิดขึ้นระหว่าง Class จะสามารถนำไปสู่การสร้าง Method ใน Class ที่เกี่ยวข้องได้ จากตัวอย่าง Sequence ข้างต้นจะทำการออกแบบสำหรับการทำรายการบัญชีการเงินของธนาคาร โดยจะเริ่มจากหน้าจอ JSP ซึ่งถือเป็น User Interface (UI) ประเภทหนึ่งจะทำหน้าที่ติดต่อกับผู้ใช้ เพื่อรับข้อมูลจากการใช้งานโดย User จากนั้นเมื่อกดปุ่มคำสั่งใดๆ ผ่านหน้าจอ JSP ก็จะมีการเรียกใช้ Servlet โดยเรียกผ่าน Method doPost ของ Account Servlet และทำการส่งค่าและเรียกใช้ Class ต่างๆที่เกี่ยวข้องเพื่อทำการประมวลผลตาม Business Process ต่อไป Servlet และ Class ที่ทำการประมวลผลต่างๆจึงเป็นตัวควบคุมการทำงาน หรือ Controller นั่นเอง ส่วน Class ที่ทำหน้าที่เก็บข้อมูลต่างๆ จะเรียกว่า Entity Class



(อ้างอิงจาก หนังสือ Schaum's Outline of UML)

รูปที่ 2.15 ตัวอย่าง Sequence ในขั้นตอนของการออกแบบ

2.3.4.3 Collaboration diagram ทำหน้าที่เช่นเดียวกับ Sequence Diagram แต่รูปแบบและลักษณะการเขียนจะต่างกัน หรืออาจกล่าวได้ว่า Collaboration Diagram ก็คือรูปอีกรูปแบบหนึ่งของ Sequence Diagram เมื่อได้ Sequence Diagram แล้ว Tool บางชนิดสามารถ Generate Collaboration Diagram ให้ได้เลยหรือในทางกลับกัน เมื่อสร้าง Collaboration Diagram เสร็จแล้วก็จะสามารถ Generate Sequence Diagram ได้โดยอัตโนมัติ ซึ่งถือว่าทั้งสอง Diagram สะท้อนภาพกันและกันอยู่นั่นเอง



(อ้างอิงจาก หนังสือ Schaum's Outline of UML)

รูปที่ 2.16 ตัวอย่าง Collaboration Diagram ที่สร้างจาก Visual Diagram

2.3.4.4 State Diagram ประกอบด้วยสถานะต่างๆของ Object และเหตุการณ์ต่างๆที่ทำให้สถานะของ Object เปลี่ยนและการกระทำที่เกิดขึ้นเมื่อสถานะของระบบเปลี่ยนไป สามารถบอกสถานะของ Object ได้ โดยจะให้ความสนใจ ณ เวลาใดๆ Object นั้นมีสถานะเป็นแบบใด

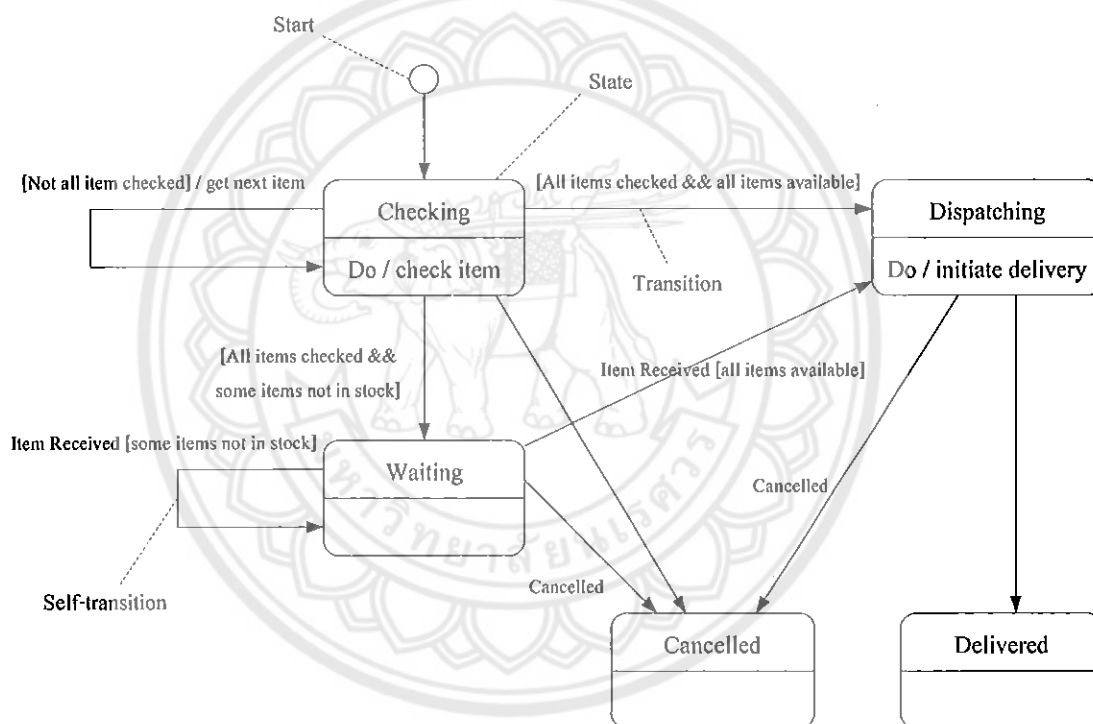
Chart ID : UML Distilled Figure 8-2

Chart Name : Fig 8-2 : State Diagram

Chart Type : UML State Diagram

This sample State diagram can be found in the UML Distilled Book that is included with Visual UML.

(Chapter 8, Page 124, Figure 8-2)



(อ้างอิงจาก หนังสือ Schaum's Outline of UML)

รูปที่ 2.17 ตัวอย่าง State Diagram ที่สร้างจาก Visual UML

2.3.4.5 Activity Diagram แสดงลำดับกิจกรรมของการทำงาน (Flow) สามารถแสดงทางเลือกที่เกิดขึ้นได้ Activity Diagram จะแสดงขั้นตอนการทำงานในการปฏิบัติการ โดยประกอบไปด้วยสถานะต่างๆที่เกิดขึ้นระหว่างการทำงาน และผลจากการทำงานในขั้นตอนต่างๆ

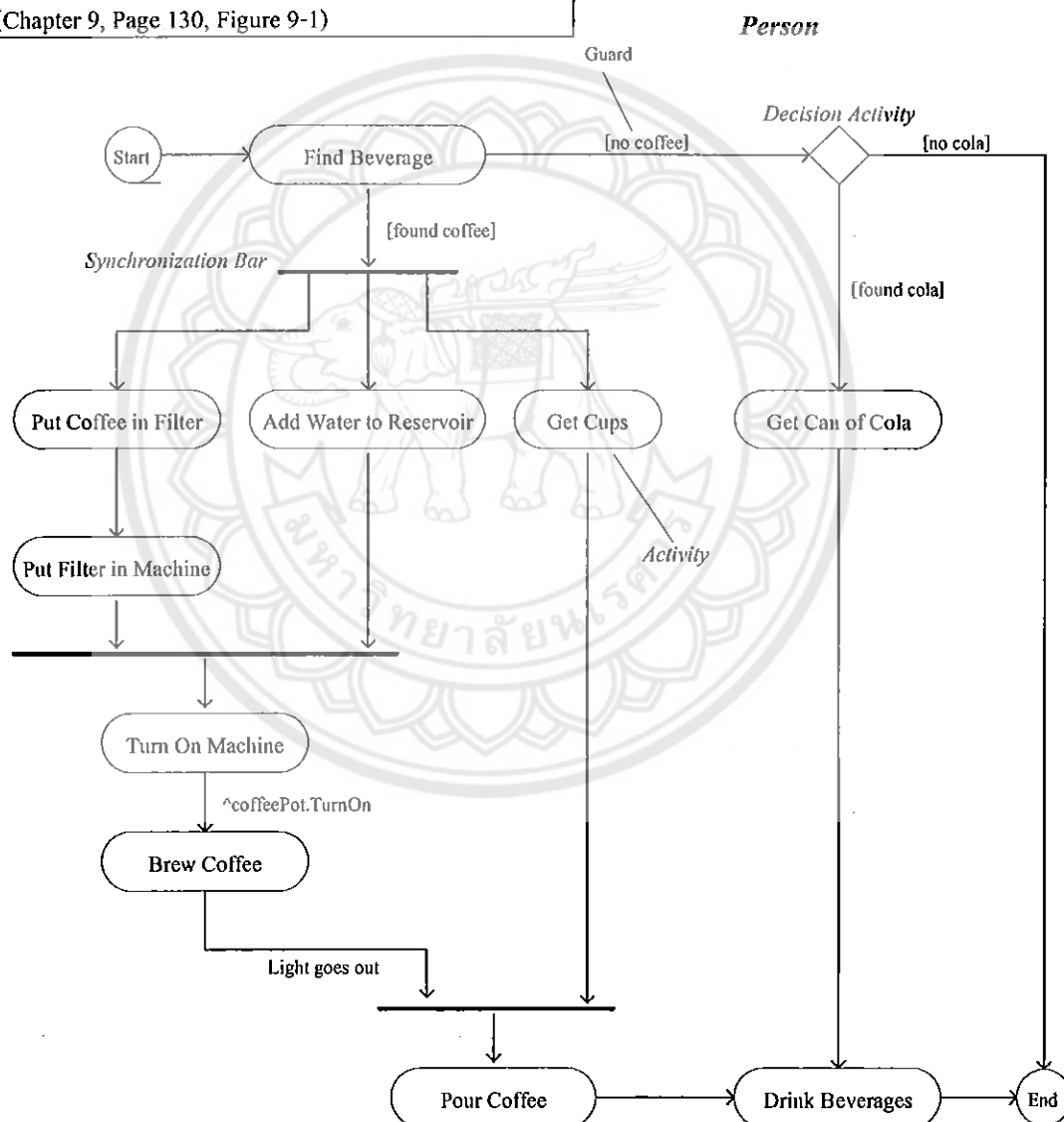
Chart ID : UML Distilled Figure 9-1

Chart Name : Fig 9-1 : Activity Diagram

Chart Type : UML Activity Diagram

This sample Activity diagram can be found in the UML Distilled Book that is included with Visual UML.

(Chapter 9, Page 130, Figure 9-1)



(อ้างอิงจาก หนังสือ Schaum's Outline of UML)

รูปที่ 2.18 ตัวอย่าง Activity Diagram ที่สร้างจาก Visual UML

2.3.4.6 Class Diagram ประกอบด้วย Class และความสัมพันธ์ต่างๆ ระหว่าง Class เช่น Dependency, Generalization, Association เป็นต้น Class Diagram ยังสามารถทำการแสดงรายละเอียดภายใน Class แต่ละ Class ได้ว่ามี Method อะไรบ้าง Field และ Attribute เป็นอย่างไร

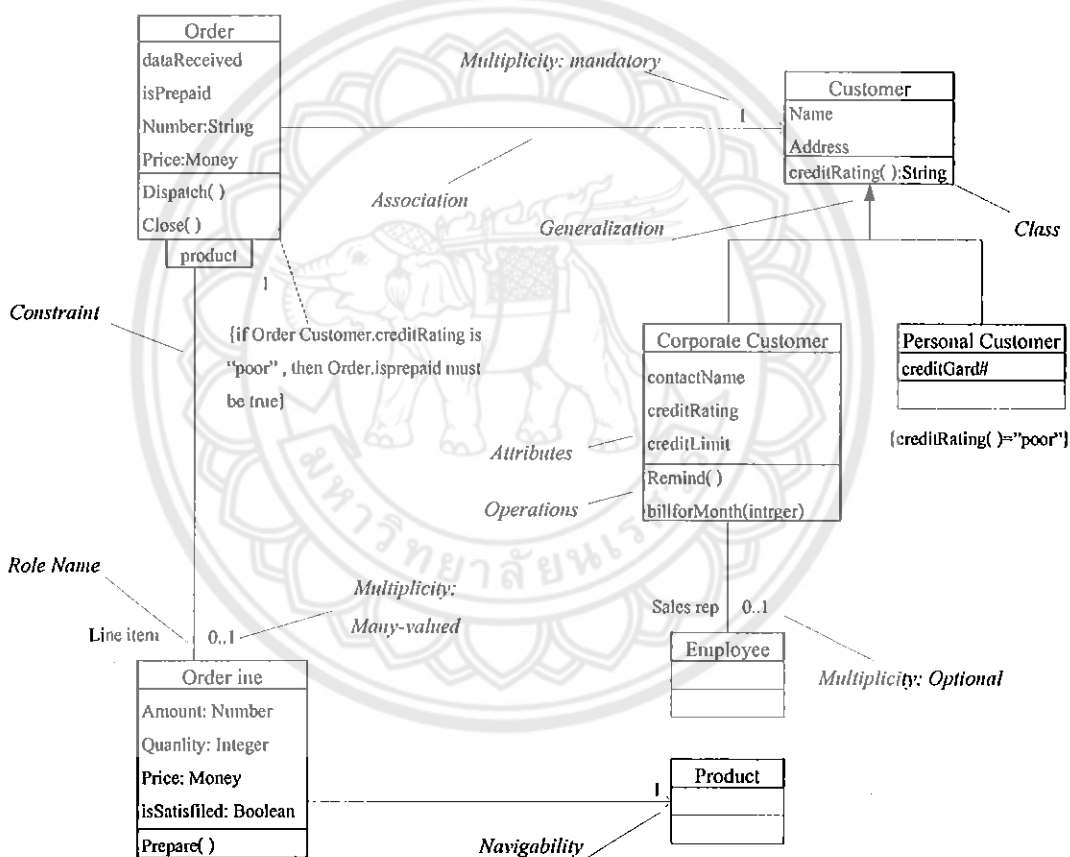
Chart ID : UML Distilled Figure 4-1

Chart Name : Figure 4-1 : Class Diagram

Chart Type : UML Class Diagram

This sample Class diagram can be found in the UML Distilled Book that is included with Visual UML.

(Chapter 4, Page 54, Figure 4-1 & 4-3)

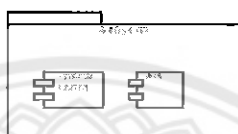


(อ้างอิงจาก หนังสือ Schaum's Outline of UML)

รูปที่ 2.19 ตัวอย่าง Class Diagram ที่สร้างจาก Visual UML

2.3.4.7 Object Diagram ประกอบด้วย Object และ Relation ระหว่าง Object โดยแต่ละ Object จะแสดง Instance ของแต่ละ Class ที่มีในระบบ และความสัมพันธ์ต่างๆ ระหว่าง Class เช่น Dependency, Generalization, Association จะมีลักษณะเช่นเดียวกับใน Class Diagram

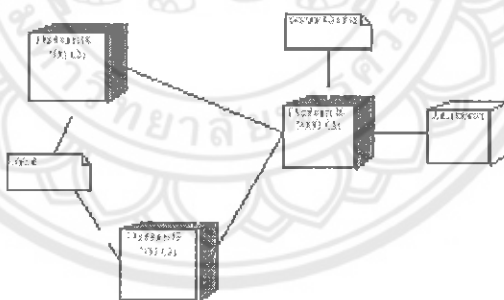
2.3.4.8 Component Diagram เป็น Diagram ซึ่งแสดงโครงสร้างทางกายภาพของซอฟต์แวร์ โดยจะประกอบด้วยองค์ประกอบซึ่งอยู่ในรูปต่างๆ เช่น Binary, Texts และ Executables ภายใน Component Diagram ก็จะมีความสัมพันธ์แสดงอยู่เช่นเดียวกับ Class Diagram, Object Diagram



(อ้างอิงจาก หนังสือ Schaum's Outline of UML)

รูปที่ 2.20 ตัวอย่าง Component Diagram

2.3.4.9 Deployment Diagram เป็นสิ่งที่สามารถทำการแสดงระบบสถาปัตยกรรมของ ฮาร์ดแวร์และซอฟต์แวร์ ตลอดจนความสัมพันธ์ระหว่างฮาร์ดแวร์และซอฟต์แวร์



(อ้างอิงจาก หนังสือ Schaum's Outline of UML)

รูปที่ 2.21 ตัวอย่าง Deployment Diagram

1499 7839

พ.ร.

ว 8877 ก 2551

2.3.5 Visual Basic

Visual Basic เป็นภาษาโปรแกรมภาษาหนึ่งพัฒนาโดยบริษัทไมโครซอฟต์ เป็นภาษาที่ใช้พัฒนาโปรแกรมหรือ Application ที่ทำงานภายใต้ระบบปฏิบัติการวินโดวส์และวินโดวส์เอ็นที (Window NT) การใช้งาน Visual Basic ค่อนข้างง่าย โปรแกรมเมอร์สามารถวาดและวางองค์ประกอบต่างๆ บนหน้าจอเพื่อติดต่อกับผู้ใช้ (User Interface) ได้ตามต้องการ เมื่อวาดหน้าจอได้เสร็จก็เขียนโปรแกรมซึ่งเป็นลักษณะแบบมีโครงสร้างทางภาษากลายคลึงกับภาษาอังกฤษ เพื่อเป็นการเชื่อมโยงความสัมพันธ์ขององค์ประกอบแต่ละตัวบนหน้าจอเข้าด้วยกัน ให้ทำงานอย่างสัมพันธ์กันตามที่โปรแกรมเมอร์ต้องการตามหลักการของ Object Oriented นั่นเอง

2.3.6 .NET Framework

เป็นแพลตฟอร์มใหม่ และเปลี่ยนแปลงไปอย่างสิ้นเชิง ที่ถูกสร้างขึ้นโดย บริษัทไมโครซอฟต์เพื่อใช้ในการพัฒนา Application .Net Framework ถูกออกแบบมาเพื่อให้สามารถถูกใช้จากภาษาใดๆ ก็ได้ รวมถึง C# ด้วย รวมถึง C++, Visual Basic, Jscript, Delphi และอื่นๆ เพื่อให้สิ่งเหล่านี้เป็นไปได้ จึงเกิดภาษาเหล่านี้ ขึ้นมาในรูปแบบของเวอร์ชันเฉพาะ สำหรับคอมพิวเตอร์อีกด้วย ได้แก่ภาษา Managed C++, Visual Basic.Net, Jscript .Net, Borland C#, Delphi8 เป็นต้น และมีอีก ใหม่นานี้ ที่กำลังพัฒนา และปล่อยออกสู่ท้องตลาดอยู่ ตลอดเวลา ไม่ใช่เพียงแค่ภาษาทั้งหมดเหล่านี้ จะมีการเข้าถึง .NET Framework เท่านั้น แต่มันยังสามารถสื่อสาร กับภาษาอื่นๆ ได้อีกด้วย

จุดประสงค์ของเทคโนโลยีคอตเน็ต มีไว้สำหรับผู้พัฒนาแบบมืออาชีพที่พัฒนาระบบงานสำหรับองค์กรตั้งแต่งานขนาดเล็กจนถึงงานระดับใหญ่มากในองค์กรและระหว่างองค์กร โดยครอบคลุมการประยุกต์งานทั้งแบบดั้งเดิม ที่เรียกว่าแบบ Host Base, แบบ Console Application, แบบ Client Server, แบบ Multi tier และแบบประยุกต์บนเว็บ หากผู้พัฒนาระบบงานได้ทดลองใช้หรือศึกษาจากเครื่องมือ เช่น การใช้ภาษา C# จะพบว่าเมื่อใช้พัฒนาระบบงานจะมีเครื่องมือช่วยทำให้การพัฒนาได้หลากหลายรูปแบบตั้งแต่ Console Application, Window Application หรือ Web Application

ไมโครซอฟต์ ได้ทางสถาปัตยกรรมของเทคโนโลยีคอตเน็ตไว้เป็น 6 ระดับดังรูป



รูปที่ 2.22 โมเดลของเทคโนโลยีคอตเน็ต

ระดับเว็บเซอร์วิส เป็นระดับบนสุดของคอตเน็ตที่ใช้สำหรับให้บริการแก่ผู้ใช้ ระดับนี้เป็นระดับการทำงานโดยผู้ใช้ติดต่อผ่านเว็บ เพื่องานต่างๆ โดยเฉพาะ งานทางด้าน E-Commerce, E-Business ตลอดจนการบริการประชาชนด้าน E-Service การบริการบนเว็บเน้นให้มีการโต้ตอบผ่านเว็บแบบสองทิศทาง ทั้งที่ติดต่อโดยผู้ใช้ หรือติดต่อระหว่างคอมพิวเตอร์กับคอมพิวเตอร์เองโดยอัตโนมัติ การเขียนโปรแกรมปัจจุบันจึงเน้นรูปแบบ Use Interface ที่อยู่บนเว็บ

ระดับเฟรมเวิร์กและไลบรารี เป็นกลุ่มของเฟรมเวิร์กและโปรแกรมที่เก็บเป็นไลบรารี ที่ให้ผู้พัฒนาเรียกไปใช้ กลุ่มนี้ทำให้การพัฒนาระบบงานกระทำได้ง่าย ดี และรวดเร็ว ตัวเฟรมเวิร์กประกอบด้วย ASP.NET (Active Server Pages) ส่วนนี้สำหรับผู้พัฒนาเว็บเพื่อการบริการให้ได้ดี และทำการโต้ตอบผ่านทางเว็บได้ง่าย ผู้พัฒนาระบบบริการบนเว็บสามารถทำได้โดยใช้ Active Server Page นี้ได้รวดเร็วและมีคุณภาพดี ADO.NET (ActiveX Data Objects) ส่วนนี้มีไว้สำหรับ

การพัฒนากระบวนการที่มีความจำเป็นที่จะเชื่อมโยงกับฐานข้อมูล และการประมวลผล Object ต่างๆ ส่วนที่ทำให้การประยุกต์การแลกเปลี่ยนข้อมูลระหว่างกัน และยังรวมถึงไลบรารีที่ทำให้ผู้พัฒนาระบบงานได้ดี โดยเฉพาะในเรื่องวินโดว์ฟอร์ม เป็นการเขียนโปรแกรมผ่านฟอร์มของวินโดว์ที่มีกราฟิกต่างๆ ให้เราหยิบมาใช้ได้

มาตรฐานการแลกเปลี่ยนข้อมูล เป็นการสร้างมาตรฐานการแลกเปลี่ยนข้อมูลบนฐาน XML ที่ทำให้ไม่ขึ้นกับแพลตฟอร์มที่ใช้ ส่วนสำคัญในส่วนนี้ประกอบด้วย SOAP (Simple Object Access Protocol) เป็นเสมือนสิ่งที่อยู่ตรงกลางสำหรับการเชื่อมโยงกัน โดยมีการนิยามสำหรับการเขียนเชื่อมโยงกับเว็บเพื่อให้เบราว์เซอร์ทำงานได้ทันที ส่วนนี้เราเรียกว่า WSDL (Web Service Description Language)

เครื่องมือสำหรับการพัฒนาระบบงาน มีการสร้างสิ่งแวดล้อมให้ผู้พัฒนาระบบงานทำได้ง่าย ส่วนนี้ไมโครซอฟต์ได้สร้าง Visual Studio ขึ้นมาใหม่และให้ชื่อว่า Visual Studio.Net เป็นเครื่องมือที่ใช้สร้างซอฟต์แวร์รวม โดยมีคอมไพเลอร์ที่จะสร้าง Object ต่างๆ ภายใต้เครื่องมือนี้ทำให้เราสร้างระบบงานบนวินโดว์ การสร้างระบบงานบนเว็บ การสร้างระบบงานบนคอนโซล การสร้างเครื่องมือเหล่านี้ไมโครซอฟต์ได้สร้างฐานร่วมกัน

Model Component ก่อนการพัฒนาคอเนกต์ มีการวางโครงสร้างโมเดลกันแล้ว โดยจากกลุ่มใหญ่สามกลุ่มที่พยายามวางรากฐานของ Component เพื่อให้ฐานการพัฒนาของซอฟต์แวร์แบบ Component มีมาตรฐานและใช้งานร่วมกัน

โมเดล Object เป็นระดับที่สร้างอยู่บนโมโนรูปที่ทำให้ทุกสิ่งทุกอย่างเป็น Object ภายใต้แนวคิด Object Oriented เพื่อที่จะกำหนด Spec ของภาษาร่วมกันได้ เพื่อให้ Object ส่วนนี้เป็นกลางไม่ว่าจะมาจากภาษาใด ก็จะได้ตามกรอบของ Object ที่กำหนดไว้

ระดับ Common Language Runtime เปรียบเสมือนเครื่องจักรที่ทำงานตามคำสั่งที่ต้องการ โดยมีการสร้างรหัสคำสั่งเหมือนเป็นภาษาเครื่องให้ทำงานได้ จึงทำให้คอเนกต์เป็นกลไกที่จะทำงานได้ตั้งแต่เครื่อง PDA, Palm, Pocket PC, Mobile Phone, PC หรือเครื่องอื่นๆ โดยไม่ขึ้นกับฮาร์ดแวร์และระบบปฏิบัติการ

2.3.7 Visual Studio.NET และ ASP.NET

Visual Studio.NET เป็นซอฟต์แวร์ที่ใช้เป็นเครื่องมือในการพัฒนา Web Service และ Application ต่างๆ โดยได้รับการออกแบบและพัฒนาโดย บริษัทไมโครซอฟต์ เพื่อให้สามารถสร้าง Web Application ตามหลักการของ .NET Framework และให้ง่ายต่อการพัฒนา Web Service มีลักษณะเป็น IDE (Integrated Development Environment) คือสามารถรวมเอาเครื่องมือที่ใช้ในการพัฒนาซอฟต์แวร์ทั้งหมดไว้ในที่เดียวกัน และทำงานในสภาพแวดล้อมเดียวกัน โดยมีภาษาหลักที่ใช้ คือ Visual Studio.NET, Visual C++ .NET และ Visual C# .NET

ASP.NET (Active Server page .NET) คือ ภาษาโปรแกรมที่ทำงานบนเครื่องบริการเว็บ IIS หรือ Internet Information Service เป็นภาษาสคริปต์ทำงานฝั่งเครื่องบริการ (Server-Side Script) พัฒนาโดยบริษัทไมโครซอฟต์ โดยมีรากฐานภาษามาจากภาษาเบสิก สำหรับ ASP.NET จะต้องทำงานบน .NET Framework

2.3.8 Microsoft SQL Server 2005 (MS SQL Server 2005)

เป็นซอฟต์แวร์จัดการระบบฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management System หรือ RDBMS) พัฒนาขึ้นโดย บริษัทไมโครซอฟต์ เป็นซอฟต์แวร์บริหารระบบฐานข้อมูลรุ่นล่าสุดในตระกูล MS SQL Server มีความสามารถและเสถียรภาพสูงใกล้เคียงกับฐานข้อมูล Oracle แต่ใช้ทรัพยากรน้อยกว่า เหมาะสมสำหรับฐานข้อมูลขนาดปานกลางถึงขนาดใหญ่ และได้รับความนิยมแพร่หลายมากในปัจจุบัน

Microsoft SQL Server 2005 และ Visual Studio.NET ได้รับการพัฒนาให้ใช้งานร่วมกันได้เป็นอย่างดีจึงเหมาะสมที่จะเลือกใช้เป็นซอฟต์แวร์จัดการระบบฐานข้อมูลสำหรับการพัฒนาระบบคอมพิวเตอร์และ เป็นโปรแกรมจัดการฐานข้อมูลระดับเซิร์ฟเวอร์ที่มีขีดความสามารถในการรองรับข้อมูลขนาดใหญ่ที่ช่วยให้การบริหารจัดการฐานข้อมูลมีประสิทธิภาพ รวดเร็ว สนองตอบต่อความต้องการขององค์กรขนาดใหญ่ และยังเป็นระบบฐานข้อมูล และโซลูชันการวิเคราะห์ที่สมบูรณ์แบบ ซึ่งนำเสนอความน่าเชื่อถือและประสิทธิภาพ ด้านการขยายระบบที่เว็บและองค์กรธุรกิจต้องการ ด้วยการรองรับ XML และ HTTP ทำให้การเข้าถึงและการแลกเปลี่ยนข้อมูลนั้นทำได้ง่ายขึ้น ในขณะที่ความสามารถในการวิเคราะห์อันทรงพลังยังช่วยเพิ่มคุณค่าของข้อมูล และด้วยความพร้อมของระบบที่ดีขึ้น ทำให้ความสามารถในการทำงานอย่างต่อเนื่อง ของระบบนั้นสูงขึ้น, จัดการกับงานในแต่ละวันโดยอัตโนมัติได้ดีขึ้น รวมทั้งปรับปรุงเครื่องมือด้านการเขียนโปรแกรม และการพัฒนาความเร็วด้านการบริการได้ดีขึ้นเช่นกัน

2.3.9 ระบบฐานข้อมูล

ฐานข้อมูล (Database) ความยุ่งยากจากการนำข้อมูลจากไฟล์ข้อมูลมาใช้งานที่ต้องขึ้นอยู่กับโปรแกรมที่ใช้จัดการกับข้อมูลทำให้ประสิทธิภาพในการใช้งานลดลง ยากต่อการขยายตัวในอนาคต และในบางกรณีไฟล์ข้อมูลที่มีอยู่ไม่สามารถนำมาใช้งานร่วมกับอีกไฟล์ข้อมูลที่ถูกติดกับอีกโปรแกรมหนึ่ง เป็นผลให้ต้องเสียค่าดูแล บำรุงรักษาค่อนข้างมากก่อให้เกิดระบบการจัดเก็บข้อมูลแบบใหม่ที่เรียกว่า “ฐานข้อมูล” (Database) ขึ้นมา

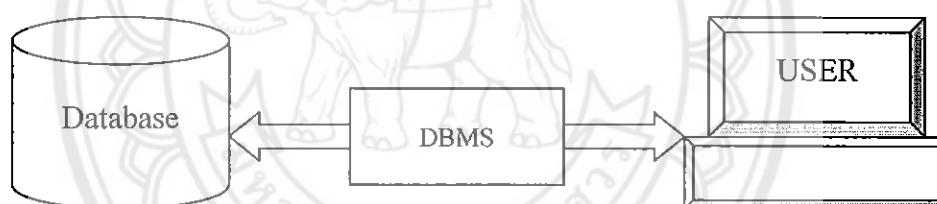
ฐานข้อมูลเป็นระบบการจัดเก็บข้อมูลที่แตกต่างไปจากไฟล์ข้อมูล ทั้งนี้เพราะฐานข้อมูลใช้วิธีการรวบรวมข้อมูลที่มีความสัมพันธ์เกี่ยวเนื่องกันไว้ในตารางที่เรียกว่าตารางสัมพันธ์ (Relation Table) มีหลักการพื้นฐานมาจากทฤษฎีเซต (Theory of Set) ซึ่งเป็นคณิตศาสตร์แบบหนึ่ง ข้อมูลจะถูกนำเสนอในรูปของตาราง เพราะสามารถทำความเข้าใจได้ง่าย สะดวกต่อการค้นหาและนำข้อมูล

ขึ้นมาแสดงผลเพื่อความสะดวกในการจัดเก็บและเรียกใช้งาน ไม่ได้เก็บข้อมูลทุกอย่างไว้ทีเดียว เหมือนไฟล์ข้อมูล ทำให้ฐานข้อมูลมีข้อจำกัดในเรื่องของการเลือกข้อมูลที่จะนำมาจัดเก็บในตาราง โดยข้อมูลบางส่วนอาจถูกแยกออกไปเก็บไว้ในอีกตารางหนึ่ง ทำให้การทำงานกับฐานข้อมูลต้องเกี่ยวข้องกับการจัดการข้อมูลที่อยู่ ในตารางตั้งแต่ตารางเดียวไปจนถึงหลาย ๆ ตาราง

2.3.9.1 ระบบการจัดการฐานข้อมูล

เนื่องจากข้อมูลทั้งหมดถูกเก็บไว้ในตาราง แต่ระบบฐานข้อมูลนั้นไม่อนุญาตให้มีการติดต่อกับข้อมูลที่อยู่ภายในได้โดยตรง ต้องกระทำผ่านระบบจัดการฐานข้อมูล (Database Management System) หรือ DBMS จึงจะใช้งานข้อมูลได้

หน้าที่หลักของ DBMS คือ ทำให้การเรียกใช้ข้อมูลจากฐานข้อมูลเป็นอิสระจากส่วนฮาร์ดแวร์ (Data Independence) โดยทำหน้าที่จัดการและควบคุมความถูกต้อง ความซ้ำซ้อน และความสัมพันธ์ระหว่างข้อมูลภายในฐานข้อมูลแทนโปรแกรมเมอร์ ส่งผลให้ผู้ใช้สามารถเรียกใช้ข้อมูลจากฐานข้อมูลได้โดยไม่ต้องทราบถึงโครงสร้างทางกายภาพของข้อมูลในระดับลึกแบบเดียวกับโปรแกรมเมอร์ ทำให้การใช้งานฐานข้อมูลกระจายไปยังกลุ่มผู้ใช้ทั่วไปไม่จำกัดอยู่เฉพาะในกลุ่มโปรแกรมเมอร์เหมือนในอดีตอีกต่อไป



รูปที่ 2.23 ระบบการจัดการฐานข้อมูล DBMS

2.3.9.2 ประโยชน์ของฐานข้อมูล

ประโยชน์ของการนำข้อมูลที่มีความสัมพันธ์กันมาใช้งานร่วมกันเป็นฐานข้อมูลมีดังต่อไปนี้

- ลดความซ้ำซ้อนของข้อมูล (Data Redundancy)
- หลีกเลี่ยงความขัดแย้งของข้อมูล (Data Inconsistency)
- แต่ละหน่วยในองค์กรสามารถใช้ข้อมูลร่วมกันได้
- กำหนดรูปแบบข้อมูลให้เป็นมาตรฐานเดียวกัน ทำให้ผู้ใช้ข้อมูลสามารถเข้าใจและสื่อสารถึงความหมายเดียวกัน
- กำหนดระดับความสามารถในการเรียกใช้ข้อมูลของผู้ใช้แต่ละคนให้แตกต่างกัน ตามความรับผิดชอบ
- รักษาความถูกต้องของข้อมูลได้

- ตอบสนองความต้องการใช้ข้อมูลในหลายรูปแบบ
- สามารถแก้ไขโครงสร้างข้อมูลได้อย่างอิสระ

ตัวอย่างคำสั่ง Microsoft SQL Server 2005

1. คำสั่งสร้างฐานข้อมูล

รูปแบบคำสั่ง

```
CREATE DATABASE database_name
[ ON
  [ < filespec > [ ,...n ] ]
  [ , < filegroup > [ ,...n ] ]
]
[ LOG ON { < filespec > [ ,...n ] } ]
[ COLLATE collation_name ]
[ FOR LOAD | FOR ATTACH ]
```

ตัวอย่าง

```
CREATE DATABASE Employees
ON
(NAME = Empl_dat,
FILENAME = 'd:\sample data dir\',
SIZE = 10,
MAXSIZE = 50,
FILEGROWTH = 5)
```

2. คำสั่งการลบฐานข้อมูล

รูปแบบคำสั่ง

```
DROP DATABASE database_name [ ,...n ]
```

ตัวอย่าง

```
DROP DATABASE pubs, newpubs
```

3. การแก้ไขฐานข้อมูล

รูปแบบคำสั่ง

```

ALTER DATABASE database
{ ADD FILE < filespec > [ ,...n ] [ TO FILEGROUP filegroup_name ]
| ADD LOG FILE < filespec > [ ,...n ]
| REMOVE FILE logical_file_name
| ADD FILEGROUP filegroup_name
| REMOVE FILEGROUP filegroup_name
| MODIFY FILE < filespec >
| MODIFY NAME = new_dbname
| MODIFY FILEGROUP filegroup_name {filegroup_property | NAME =
new_filegroup_name }
| SET < optionspec > [ ,...n ] [ WITH < termination > ]
| COLLATE < collation_name >
}

```

ตัวอย่าง

```

ALTER DATABASE Test1
ADD FILE
(
NAME = Test1dat2,
FILENAME = 'c:\Program Files\Microsoft SQL Server\MSSQL\Data\t1dat2.ndf',
SIZE = 5MB,
MAXSIZE = 100MB,
FILEGROWTH = 5MB
)

```

4. การสร้างตาราง

รูปแบบคำสั่ง

```

CREATE TABLE [database.[owner.]table_name
(
Column_name datatype [identity | constraint | NULL | NOT NULL]
| column_name AS computed_column_expression
[... ]

```

)

[ON { filegroup | DEFAULT }]

โดยที่	Table_name	คือชื่อตารางที่ต้องการสร้าง
	Column_name	คือชื่อของคอลัมน์ต่าง ๆ ที่กำหนดให้
	Datatype	คือประเภทของข้อมูลที่ได้กล่าวมาแล้ว

ตัวอย่าง

```
CREATE TABLE tbl_employees
(
  EmploysID int IDENTITY NOT NULL,
  FirstName varchar (30) NOT NULL,
  LastName varchar (30) NOT NULL,
  StartWorkDate datetime
)
```

5. การแก้ไขตาราง

รูปแบบคำสั่ง

```
ALTER TABLE table
{ [ ALTER COLUMN column_name
  { new_data_type [ ( precision [ , scale ] ) ]
  [ COLLATE < collation_name > ]
  [ NULL | NOT NULL ]
  | {ADD | DROP } ROWGUIDCOL }
]
| ADD
{ [ < column_definition > ]
| column_name AS computed_column_expression
} [ ,...n ]
| [ WITH CHECK | WITH NOCHECK ] ADD
{ < table_constraint > } [ ,...n ]
| DROP
{ [ CONSTRAINT ] constraint_name
```

```
| COLUMN column } [ ,...n ]
| { CHECK | NOCHECK } CONSTRAINT
{ ALL | constraint_name [ ,...n ] }
| { ENABLE | DISABLE } TRIGGER
{ ALL | trigger_name [ ,...n ] }
}
```

ตัวอย่าง

```
ALTER TABLE MyTable
ADD AddDate smalldatetime NULL
CONSTRAINT AddDateDflt
DEFAULT getdate() WITH VALUES
```

6. การลบตาราง

รูปแบบคำสั่ง

```
DROP TABLE table_name
```

ตัวอย่าง

```
DROP TABLE titles1
```

7. แทรกตาราง

รูปแบบคำสั่ง

```
INSERT [ INTO]
{table_name WITH(<table_hint_limited>[...n])
| view_name
| rowset_function_limited
}
{[(column_list)]
{VALUES
({DEFAULT | NULL | expression}[...n])
| derived_table
| execute_statement
}
```

```
}
| DEFAULT VALUES
```

ตัวอย่าง

```
INSERT INTO new_authors
SELECT TOP 10 *
FROM authors
```

8. การแสดงตาราง

รูปแบบคำสั่ง

```
SELECT select_list
[ INTO new_table ]
FROM table_source
[ WHERE search_condition ]
[ GROUP BY group_by_expression ]
[ HAVING search_condition ]
[ ORDER BY order_expression [ ASC | DESC ] ]
```

ตัวอย่าง

```
USE pubs
SELECT au_fname, au_lname, phone AS Telephone, city, state
FROM authors
ORDER BY au_lname ASC, au_fname ASC
```

9. การอัปเดตตาราง

รูปแบบคำสั่ง

```
UPDATE
{
table_name WITH ( < table_hint_limited > [ ...n ] )
| view_name
| rowset_function_limited
}
SET
```

```

{ column_name = { expression | DEFAULT | NULL }
| @variable = expression
| @variable = column = expression } [ ,...n ]
{ { [ FROM { < table_source > } [ ,...n ] ]
[ WHERE
< search_condition > ] }
|
[ WHERE CURRENT OF
{ { [ GLOBAL ] cursor_name } | cursor_variable_name }
] }
[ OPTION ( < query_hint > [ ,...n ] ) ]

```

ตัวอย่าง

```

UPDATE authors
SET authors.au_fname = 'Annie'
WHERE au_fname = 'Anne'

```

10. การเลือกทำงานตามเงื่อนไข โดยใช้คำสั่ง CASE

รูปแบบคำสั่ง

```

CASE input_expression
WHEN when_expression THEN result_expression
[ ...n ]
[
ELSE else_result_expression
]
END

```

ตัวอย่าง

```

SELECT Category =
CASE type
WHEN 'popular_comp' THEN 'Popular Computing'
WHEN 'mod_cook' THEN 'Modern Cooking'
WHEN 'business' THEN 'Business'

```

```

WHEN 'psychology' THEN 'Psychology'
WHEN 'trad_cook' THEN 'Traditional Cooking'
ELSE 'Not yet categorized'
END,
CAST (title AS varchar(25)) AS 'Shortened Title',
Price AS Price
FROM titles
WHERE price IS NOT NULL
ORDER BY type, price

```

11. การเลือกทำงานตามเงื่อนไขโดยใช้คำสั่ง IF

รูปแบบคำสั่ง

```

IF Boolean_expression { sql_statement | statement_block }
[
ELSE
{ sql_statement | statement_block }
]

```

ตัวอย่าง

```

IF (SELECT COUNT (price)
FROM titles
WHERE title_id LIKE 'TC%' AND price BETWEEN 10 AND 20) > 0
BEGIN
SET NOCOUNT ON
SET @msg = 'There are several books that are a good value between $10 and
$20. These books are : '
PRINT @msg
SELECT title
FROM titles
WHERE title_id LIKE 'TC%' AND price BETWEEN 10 AND 20
END
ELSE

```

```

BEGIN
SET NOCOUNT ON
SET @msg = 'There are no books between $10 and $20. You might consider the
following books that are under $10.'
PRINT @msg
SELECT title
FROM titles
WHERE title_id LIKE 'TC%' AND price < 10
END

```

12. การเลือกทำงานตามเงื่อนไข โดยใช้คำสั่ง WHILE

รูปแบบคำสั่ง

```

WHILE Boolean_expression
{ sql_statement | statement_block }
[ BREAK ]
{ sql_statement | statement_block }
[ CONTINUE ]

```

ตัวอย่าง

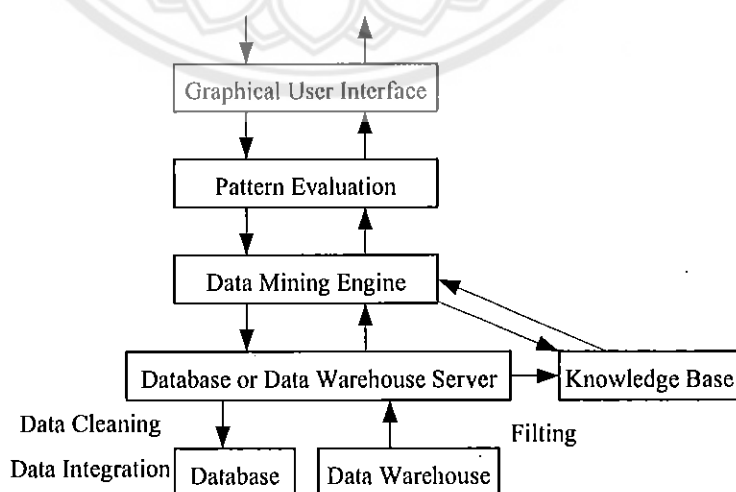
```

WHILE (SELECT AVG(price) FROM titles) < $30
BEGIN
UPDATE titles
SET price = price * 2
SELECT MAX(price) FROM titles
IF (SELECT MAX(price) FROM titles) > $50
BREAK
ELSE
CONTINUE
END
PRINT 'Too much for the market to bear'

```

2.3.10 เทคโนโลยีเหมืองข้อมูล (Data Mining)

การทำเหมืองข้อมูล คือกระบวนการค้นหาสารสนเทศหรือข้อความรู้ที่อยู่ในฐานข้อมูลขนาดใหญ่ที่ซับซ้อน เพื่อนำข้อความรู้ที่ได้ไปใช้ประโยชน์ในการตัดสินใจ สารสนเทศที่ได้อาจนำมาสร้างการพยากรณ์หรือสร้างตัวแบบสำหรับการจำแนกหน่วยหรือกลุ่ม หรือแสดงความสัมพันธ์ระหว่างหน่วยต่างๆ หรือให้ข้อสรุปของสาระในฐานข้อมูล การทำเหมืองข้อมูลประกอบขึ้นด้วยการนำกระบวนการทางสถิติและการเรียนรู้ผ่านระบบคอมพิวเตอร์ เพื่อสร้างตัวแบบ กฎเกณฑ์ รูปแบบ การพยากรณ์และข้อความรู้ จากฐานข้อมูลขนาดใหญ่ โดยการทำเหมืองข้อมูลมีขั้นตอนการดำเนินงานหลายขั้นตอนซึ่งต้องอาศัยเทคนิคหรือวิธีการต่างๆ เช่น วิธีการจัดกลุ่ม การค้นหาความสัมพันธ์ การพยากรณ์ เป็นต้น การดำเนินงานมักอยู่ในลักษณะของการสร้างตัวแบบ (Modeling) ที่อธิบายความเป็นไปหรือสถานการณ์หนึ่งที่เกิดขึ้นแล้ว หรือที่เราทราบคำตอบ แล้วนำตัวแบบนี้มาใช้อธิบายสถานการณ์ที่ยังไม่เกิดขึ้น หรือที่ไม่ทราบคำตอบ ตัวแบบเหล่านี้อาจเป็นแบบที่เรียบง่ายไปจนถึงตัวแบบที่ยุ่งยากซับซ้อน และอาจใช้การผสมผสานแนวคิดหรือเครื่องมือต่างๆ เข้าด้วยกัน เพื่อที่จะสามารถสกัดข้อความรู้ที่อยู่ในข้อมูลขนาดใหญ่ได้ โดยใช้เทคโนโลยีคลังข้อมูล (Data Warehouse) เข้ามาช่วยในการจัดการข้อมูลเพื่อเพิ่มประสิทธิภาพของการทำเหมืองข้อมูล ดังนั้น ถ้ามีฐานข้อมูลขนาดใหญ่ที่มีข้อมูลคุณภาพดี เทคโนโลยีการทำเหมืองข้อมูลจะช่วยให้การค้นหาหรือแสวงหาโอกาสทางธุรกิจใหม่ โดยการทำเหมืองข้อมูลจะก่อให้เกิดกระบวนการอัตโนมัติในการค้นพบสารสนเทศหรือข้อความรู้ในฐานข้อมูลขนาดใหญ่ ด้วยการใช่วิธีการเช่นการพยากรณ์แนวโน้มและพฤติกรรมผู้บริโภคแบบอัตโนมัติ หรือเกิดกระบวนการอัตโนมัติในการค้นพบรูปแบบที่ไม่เคยรู้จักมาก่อน ด้วยการใช่วิธีการค้นหาเข้าไปในรายละเอียดของฐานข้อมูลเพื่อหารูปแบบที่ซ่อนอยู่ในฐานข้อมูลนั้น



(อ้างอิงจาก หนังสือ Principles of Data Mining)

รูปที่ 2.24 สถาปัตยกรรมของระบบการทำเหมืองข้อมูล

2.3.10.1 ประเภทของข้อมูลที่สามารถทำเหมืองข้อมูล

1. Relational Database เป็นฐานข้อมูลที่จัดเก็บอยู่ในรูปแบบของตาราง โดยในแต่ละตารางจะประกอบไปด้วยแถวและคอลัมน์ ความสัมพันธ์ของข้อมูลทั้งหมดสามารถแสดงได้โดย Entity-relationship (ER) model

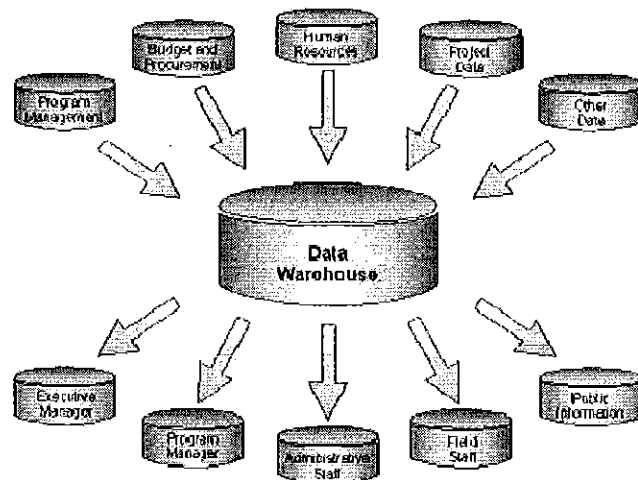
ตารางที่ 2.2 ตัวอย่างตารางข้อมูลของนักศึกษาชั้นต้น

ID	Sex	Address	001	002	...	Major	GPA.
1	Male	Bangkok	Medium	Low	...	Elec.	2.3
2	Female	Non-Bangkok	High	High	...	Civil	3.2

ตารางที่ 2.3 ตัวอย่างข้อมูลการลงทะเบียนเรียนของนักศึกษา

ID	Subject	Section	Term	Year	Grade
1	001	1	1	2550	C+
1	002	1	1	2550	D
1	005	1	1	2550	B+

2. คลังข้อมูล (Data Warehouse) ฐานข้อมูลขนาดใหญ่ที่รวบรวมข้อมูลทั้งจากแหล่งข้อมูลภายในและภายนอกองค์กร หลายแหล่งหลายช่วงเวลา โดยข้อมูลในคลังข้อมูลจะถูกนำมาใช้เพื่อสนับสนุนการตัดสินใจบริหารงานของผู้บริหาร โดยเฉพาะการเป็นข้อมูลพื้นฐานให้กับระบบงาน



(อ้างอิงจาก หนังสือ Principles of Data Mining)

รูปที่ 2.25 โครงสร้างคลังข้อมูล (Data Warehouse)

ลักษณะเฉพาะของคลังข้อมูล

- Subject-Oriented คลังข้อมูลถูกออกแบบมาเพื่อมุ่งเน้นไปในแต่ละเนื้อหาที่สนใจ ไม่ได้เน้นไปที่การทำงานหรือกระบวนการ
- Integration การรวบรวมข้อมูลจากหลายฐานข้อมูลปฏิบัติการเข้าด้วยกัน และทำให้ข้อมูลมีมาตรฐานเดียวกัน
- Time Valiancy ข้อมูลในคลังข้อมูล จะต้องจัดเก็บโดยกำหนดช่วงเวลาเอาไว้ โดยจะสัมพันธ์กับการดำเนินธุรกิจของหน่วยธุรกิจนั้น
- Nonvolatile ข้อมูลในคลังข้อมูลจะไม่เปลี่ยนแปลงบ่อย ไม่ว่าจะเป็นการเพิ่มเติมข้อมูลใหม่ หรือการปรับปรุงแก้ไขข้อมูลเดิมที่บรรจุอยู่แล้ว ผู้ใช้ทำได้เพียงการเข้าถึงข้อมูลเท่านั้น
- Inflow การนำข้อมูลจากฐานข้อมูลอื่นเข้าสู่คลังข้อมูลทั้งงานข้อมูลภายในและภายนอกองค์กร โดยในขั้นนี้อาจมีการเปลี่ยนแปลงโครงสร้างข้อมูล
- Upflow การเพิ่มคุณค่าให้กับข้อมูลด้วยเพื่อให้ข้อมูลอยู่ในรูปแบบที่เป็นประโยชน์มากที่สุด เช่นการจัดกลุ่มข้อมูลหาค่าทางสถิติที่ซับซ้อน จัดข้อมูลให้อยู่ในรูปแบบมาตรฐาน
- Downflow การปรับปรุงเปลี่ยนแปลงข้อมูลเก่า และไม่อยู่ในเนื้อหาที่องค์กรสนใจ ออกไปจากคลังข้อมูลขององค์กร
- Outflow เป็นขั้นตอนที่ผู้ใช้เรียกใช้ข้อมูลในคลังข้อมูลผ่านเครื่องมือต่างๆ

- Metaflow การจัดเก็บแหล่งที่มาของข้อมูลนั้น รวมถึงแหล่งเก็บในคลังข้อมูล และข้อมูลอื่นที่เกี่ยวข้องที่เชื่อมโยงไปภายนอก

3. Transactional Database ประกอบด้วยข้อมูลที่แต่ละ Transaction แทนเหตุการณ์ในขณะใดขณะหนึ่ง เช่น ใบเสร็จรับเงิน

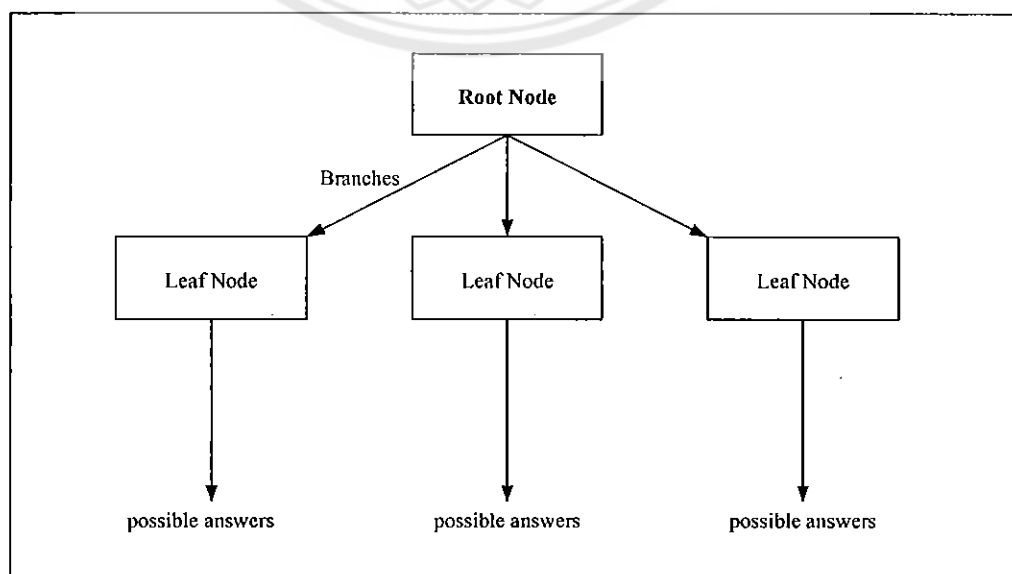
4. Advance Database เป็นฐานข้อมูลที่จัดเก็บในรูปแบบอื่น ๆ เช่น ข้อมูลแบบ Object-Oriented, ข้อมูลที่เป็น Text File, ข้อมูล Multimedia ความเชื่อมโยงระหว่าง Data Mining และ Data Warehouse

- ในการเลือกส่วนย่อยๆ ของ Record และ Field ที่ตรงประเด็น Data Mining จะต้องการความสามารถในการคิวรีข้อมูลของ Data Warehouse
- การศึกษาผลที่ได้จากการทำ Data Mining จะเป็นประโยชน์อย่างมาก ถ้าหากมีการสืบค้นข้อมูลอย่างมีแบบแผนต่อไปในอนาคต ซึ่ง Data Warehouse จะเป็นแหล่งจัดเก็บข้อมูลภายหลังไว้ให้

2.3.11 ทฤษฎีการตัดสินใจ (Decision Tree)

Decision Tree เป็นวิธีหนึ่งที่สำคัญใน Classification โดย Decision Tree จะมีลักษณะเป็น Flowchart เหมือนโครงสร้างต้นไม้ ที่แต่ละ Node แสดงคุณลักษณะ (Attribute) ที่ใช้ทดสอบข้อมูล แต่ละกิ่งแสดงผลในการทดสอบและ Leaf node แสดงกลุ่มหรือ Class ที่กำหนดไว้ ซึ่ง Decision Tree นี้ง่ายต่อการปรับเปลี่ยนเป็น Classification Rule

เมื่อมีข้อมูลที่ต้องการที่จะจัดกลุ่มก็จะนำ Attribute ต่างๆของข้อมูลนั้นไปเทียบกับ Decision Tree นี้ตามเส้นทางใน Tree จนกระทั่งได้ Class ปลายทาง ซึ่งก็คือกลุ่มของข้อมูลที่เหมือนกัน

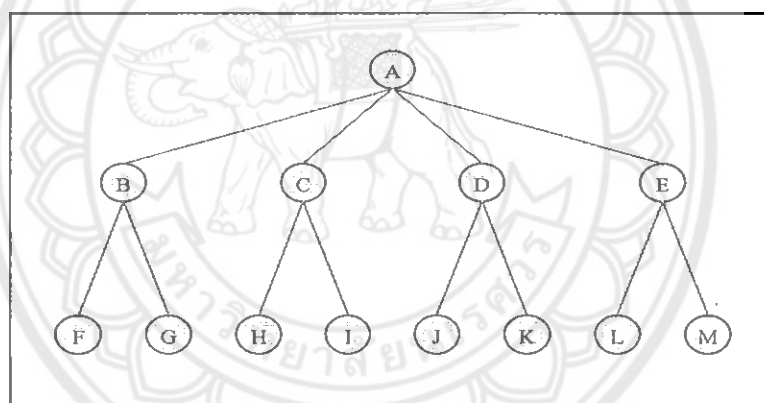


รูปที่ 2.26 ลักษณะแผนภูมิแบบต้นไม้

2.3.11.1 องค์ประกอบแผนภูมิแบบต้นไม้

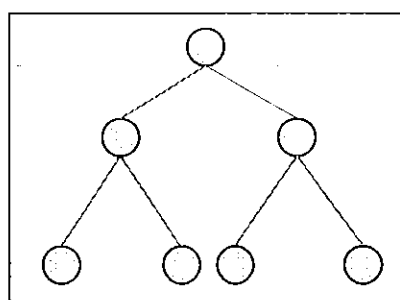
องค์ประกอบของแผนภูมิแบบต้นไม้จะเรียกตามชื่อองค์ประกอบต้นไม้ ได้แก่ ราก (Root) กิ่ง (Branch) และใบ (Leaf) โดยจุดที่แตกเป็นกิ่งก้านแต่ละจุดเรียกว่า โหนด (Node) จากตัวอย่างในรูปที่ 2.26 นั้น ตำแหน่งโหนดบนสุดเรียกว่าราก เส้นลูกศรที่ลากจากโหนดเรียกว่ากิ่ง และสิ่งที่อยู่ปลายกิ่งเรียกว่า ใบ

เนื่องจากโครงสร้างต้นไม้อาจจะมีหลายชั้น ดังนั้น กว่าจะถึงใบก็อาจจะมีกิ่งที่แตกออกมาเป็นโหนดย่อยๆ โดยโหนดย่อยที่ยังไม่ใช่ใบจะถูกเรียกว่าโหนดลูก (Child node) และโหนดที่แตกออกมาเป็นโหนดลูกเรียกว่าโหนดพ่อแม่ (Parents node) และเนื่องจากทิศทางของแผนภูมิประเภทนี้จะไล่ลำดับจากบนลงล่างและจากซ้ายไปขวา ดังนั้น การเขียนผังอาจจะมีหัวลูกศรไว้ในฐานที่เข้าใจ ดังรูปที่ 2.27



รูปที่ 2.27 แผนภูมิแบบต้นไม้ที่มีโหนดย่อยหลายระดับ

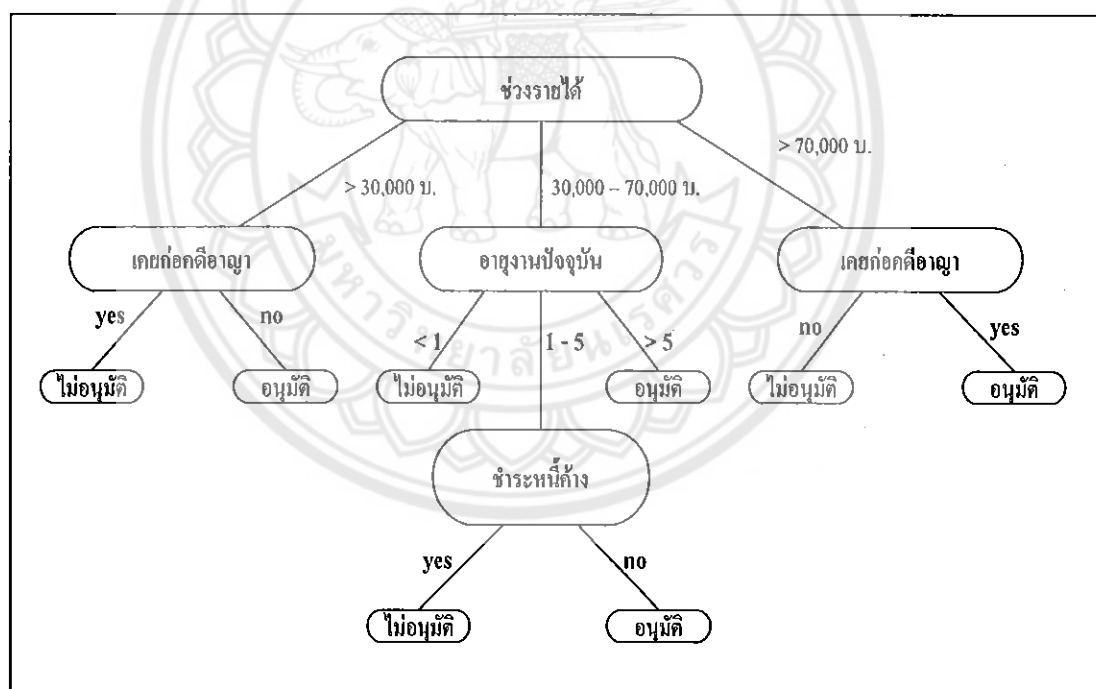
แผนภูมิต้นไม้แบบไบนารี (Binary Tree Diagram หรือ B- Tree Diagram) เป็นแผนภูมิแบบต้นไม้ที่มีกิ่งแตกออกมาได้ไม่เกิน 2 โหนดในแต่ละชั้น ดังรูปที่ 2.28



รูปที่ 2.28 แผนภูมิแบบต้นไม้ที่มีโหนดไม่เกิน 2 โหนดในแต่ละชั้น

การเขียนแผนผังโดยทั่วไปจะแสดงเงื่อนไขไว้ที่โหนด แสดงสิ่งที่เลือกไว้ที่กิ่ง และแสดงผลการตัดสินใจไว้ที่ใบ จากตัวอย่างจะแสดงหลักเกณฑ์อนุมัติเงินกู้โดยนำคุณสมบัติส่วนบุคคลหลายด้านของผู้ขอู้มาประกอบกันเป็นเงื่อนไขพิจารณา โดยเริ่มต้นจะพิจารณาจากรายได้ต่อเดือน ถ้ารายได้อยู่ในระดับต่ำหรือระดับสูงจะต้องพิจารณาประวัติทางคดีอาญา แต่ถ้ารายได้ในระดับกลางก็จะต้องพิจารณาความมั่นคงทางการเงิน เป็นต้น

แผนภูมิตัดสินใจแบบต้นไม้ (Decision Tree Diagram หรือเรียกกันทั่วไปว่า Decision Tree) เป็นแผนภูมิแบบต้นไม้ที่แสดงผลลัพธ์ที่มาจากการเลือกเงื่อนไขต่างๆ ที่กำหนดไว้ ตั้งแต่เงื่อนไขเบื้องต้น และเงื่อนไขลำดับต่อๆ มาจนถึงเงื่อนไขสุดท้ายที่เกี่ยวข้องในการนำมาพิจารณาประกอบร่วมกันดังตัวอย่างในรูปที่ 2.29



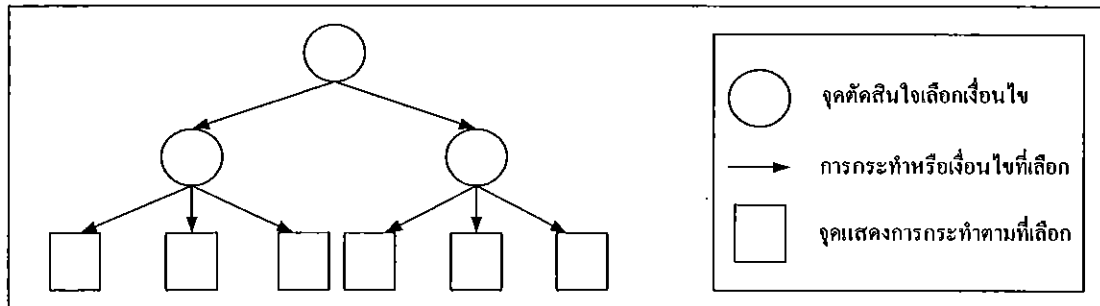
รูปที่ 2.29 แผนภูมิตัดสินใจแบบต้นไม้แสดงหลักเกณฑ์พิจารณาอนุมัติเงินกู้

2.3.11.2 องค์ประกอบของแผนผังสำหรับอธิบายกระบวนการ

แผนผังที่อธิบายกระบวนการทำงานจะประกอบด้วยโหนดเงื่อนไขตัดสินใจ และมีก้านแสดงการเลือกเป็นตัวแทนเชื่อมกับโหนดเงื่อนไขถัดไปตามลำดับที่สอดคล้องกับกระบวนการทำงาน และเชื่อมต่อกันไปเรื่อยๆ จนได้ผลลัพธ์การตัดสินใจสุดท้ายว่าจะดำเนินการอย่างไรต่อไป

การเขียนแผนผังประเภทนี้มีกนนิยมใช้สัญลักษณ์เพื่อแสดงความหมายเจาะจง ดังรูปที่ 2.30 อย่างไรก็ตามอาจจะมีการกำหนดสัญลักษณ์แบบอื่นด้วยแต่ก็คงตามลักษณะแนวคิด

แบบเดิม



รูปที่ 2.30 การใช้สัญลักษณ์ในแผนผังตัดสินใจแบบต้นไม้

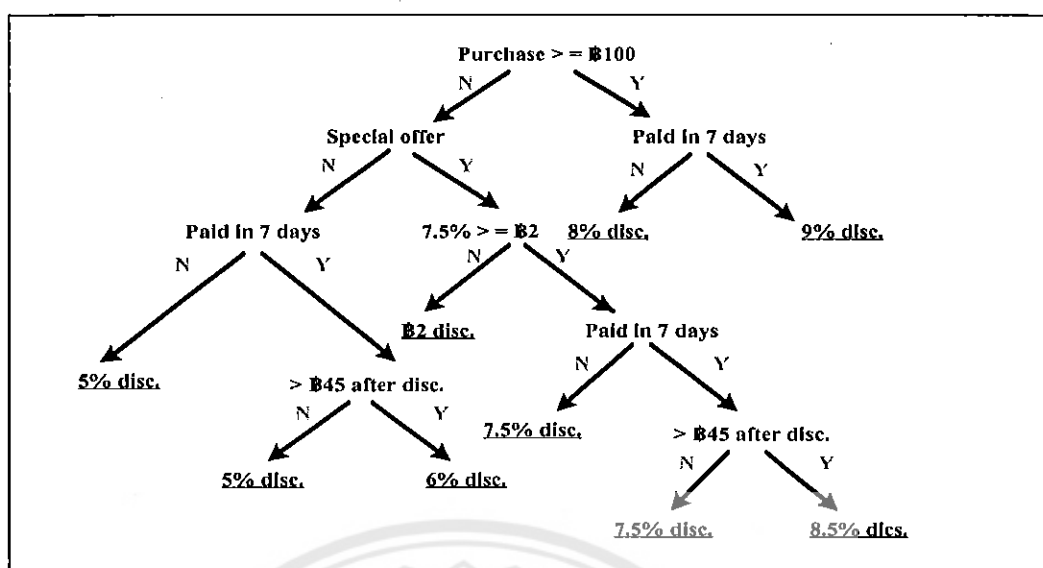
2.3.11.3 การเขียนแผนผังในภาคปฏิบัติ

การเขียนคำอธิบายการประมวลผลแบบต้นไม้ไม่นิยมใช้กับกระบวนการที่มีลักษณะการทำงานที่ซับซ้อนและต้องมีการเก็บลำดับขั้นตอนการตัดสินใจ ส่วนใหญ่นิยมใช้แผนผังแบบ Binary ซึ่งจะมีทางเลือกการตัดสินใจพื้นฐานของเงื่อนไข 2 ทาง (ใช่ กับ ไม่ใช่) เพราะมีความสะดวกในการเขียนและตรวจสอบกระบวนการและยังสอดคล้องกับวิธีการคำนวณทางตรรกะของคอมพิวเตอร์ (0 กับ 1 หรือ True กับ False)

นอกจากนี้การใช้สัญลักษณ์อาจจะแตกต่างกันไปหรือไม่ให้ความสำคัญมาก ทั้งนี้เพราะว่าโดยหลักการไล่ผังจะเริ่มจากโหนดของรากไปจนถึงโหนดท้ายซึ่งไม่มีความซับซ้อน ดังนั้น การเขียนผังอาจจะตัดรายละเอียดปลีกย่อยออก เช่น หัวลูกศร สัญลักษณ์เฉพาะ อย่างไรก็ตาม การใช้สัญลักษณ์หรือการเลือกรูปแบบการเขียนผังควรมีรูปแบบที่ไปในทางเดียวกัน

2.3.11.4 เขียนโครงสร้างต้นไม้ การเขียนโครงสร้างต้นไม้จากเงื่อนไขที่ถูกจัดลำดับเรียบร้อยแล้ว โดยมีขั้นตอนดังนี้

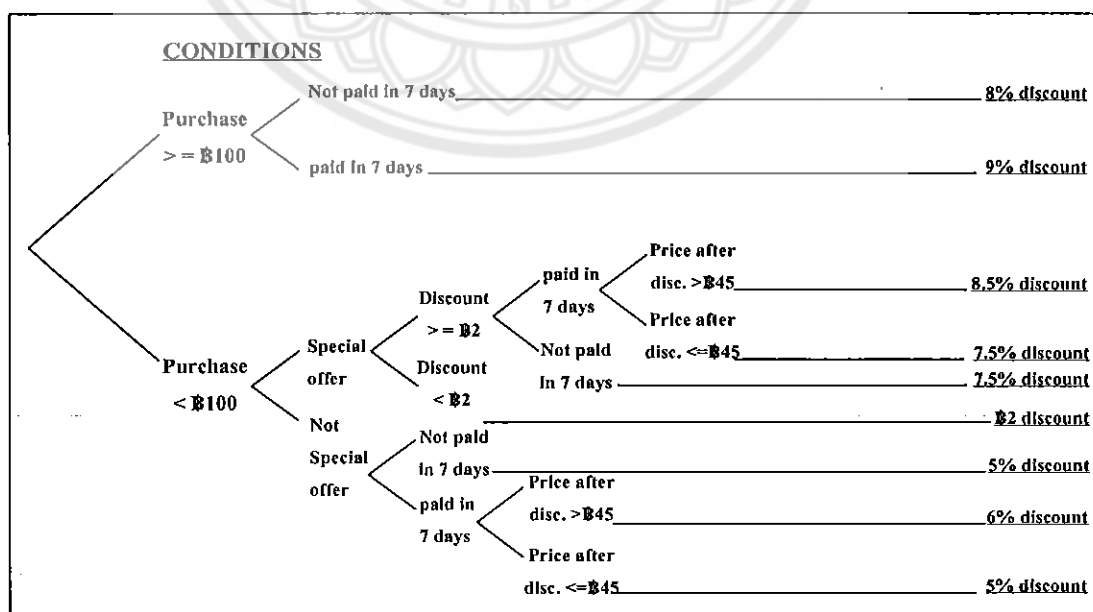
- เริ่มเขียนที่รากต้นไม้ก่อน โดยใช้เงื่อนไขแรกเป็นจุดเริ่มต้น
 - แดกกิ่งจากเงื่อนไขแรกไปสู่เงื่อนไขถัดไป ทำตามขั้นตอนที่ 2 จนครบทุกเงื่อนไขที่เกี่ยวข้องกับการตัดสินใจ
 - ที่กิ่งให้เขียนสิ่งที่ตัดสินใจเลือก หรือการกระทำที่เป็นผลลัพธ์ของการตัดสินใจ
- เลือกจากโครงสร้างลำดับการตัดสินใจที่ได้ในตัวอย่างที่ผ่านมา สามารถเขียนโครงสร้างต้นไม้ ได้ดังรูปที่ 2.31



รูปที่ 2.31 การเขียนอธิบายกระบวนการด้วยแผนภูมิต้นไม้ในแนวนิ่ง

จากตัวอย่างในรูปที่ 2.31 เราจะเห็นได้ว่า จุดตัดสินใจจะเริ่มที่ตำแหน่งบนสุดตำแหน่งแรก และจะมีทางเลือก 2 ทาง คือ ใช่ (Y) หรือไม่ใช่ (N) เพื่อที่จะไปสู่จุดตัดสินใจจุดต่อไป โดยแต่ละจุด จะมี 2 ทางเลือกเช่นเดียวกัน ได้ลงมาเรื่อยๆ จนถึงตำแหน่งสุดท้าย คือคำตอบส่วนลดที่ถูกค่าได้

สำหรับแผนภูมิต้นไม้บางที่อาจจะเขียนในแนวนอนโดยให้รากอยู่ทางซ้ายมือ ทั้งนี้รูปการใช้งานก็ยังเหมือนแผนผังแบบแนวนิ่งทุกประการ หรืออาจจะปรับรูปแบบการเขียนโครงต้นไม้ใหม่ ดังรูปที่ 2.32



รูปที่ 2.32 การเขียนอธิบายกระบวนการด้วยแผนภูมิต้นไม้ตามแนวนอน

2.3.11.5 ข้อพิจารณาในการเลือกใช้แผนผังตัดสินใจแบบต้นไม้

เมื่อต้องการนำแผนผังการตัดสินใจแบบต้นไม้มาใช้งาน ให้พิจารณาความเหมาะสมดังนี้
ข้อได้เปรียบ-เสียเปรียบของแผนผังต้นไม้

- โครงสร้างแบบต้นไม้มีลำดับชัดเจน ทำให้เห็นเงื่อนไขหรือการกระทำได้ชัดเจน
และการตรวจสอบความถูกต้องทำได้ง่ายขึ้น แต่ต้องมั่นใจว่า ลำดับการตัดสินใจในกระบวนการ
ทำงานมีความถูกต้องแล้วก่อนที่จะเขียนแผนผังต้นไม้

- มีความสะดวกและง่ายต่อการทำความเข้าใจแต่ต้องใช้พื้นที่มากในการเขียน
ลักษณะการอธิบายที่เหมาะสมสำหรับแผนผังแบบต้นไม้ แผนผังตัดสินใจแบบ
ต้นไม้เหมาะสมสำหรับอธิบายกระบวนการทำงานที่มีลักษณะดังต่อไปนี้

- เงื่อนไขในกระบวนการทำงานมีลำดับก่อนหลังชัดเจน
- มีเงื่อนไขที่แตกย่อยต่อเนื่องออกไปหลากหลาย (หรือต้องแตกกิ่งสู่เงื่อนไข
ต่อไปหลายขั้น)

2.3.11.6 การหาความน่าจะเป็นมากที่สุดในการวิเคราะห์ข้อมูล ที่ใช้ในการหาค่าปัญหา
จาก Decision Tree

1. การคำนวณหาค่า I (Information) ของตาราง และคำนวณหาค่า I ของ Field

$$I(s_1, s_2, \dots, s_m) = - \sum_{i=1}^m \frac{s_i}{S} \log_2 \frac{s_i}{S}$$

สมการที่ 2.1 การคำนวณหาค่า I (Information) ของตาราง

2. การคำนวณหาค่า Entropy เป็นการวัดปริมาณค่าของฟิลด์แต่ละฟิลด์

$$E(A) = \sum_{j=1}^v \frac{s_{1j} + \dots + s_{mj}}{S} I(s_{1j}, \dots, s_{mj})$$

สมการที่ 2.2 การหาค่า Entropy ของแต่ละฟิลด์

3. การคำนวณหาค่า Information Gained ของแต่ละ Field (Attribute)

$$\text{Gain}(A) = I(s_1, s_2, \dots, s_m) - E(A)$$

สมการที่ 2.3 การหาค่า Gain ของฟิลด์

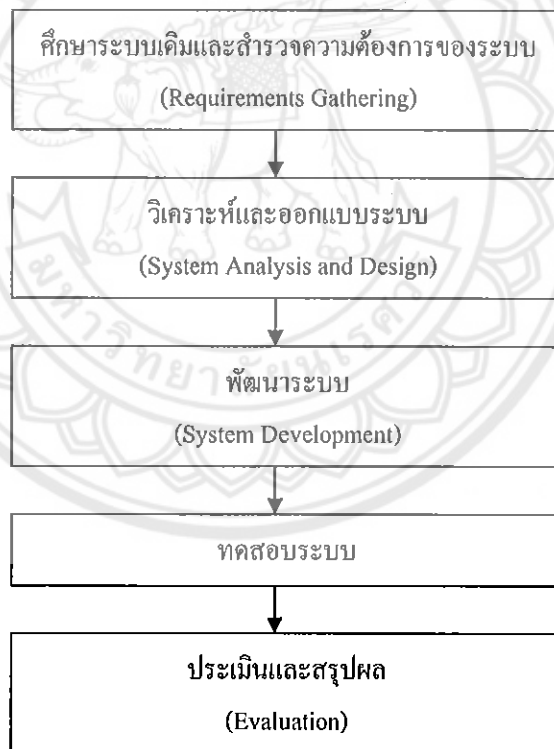
บทที่ 3

ขั้นตอนการดำเนินงาน

3.1 วิธีดำเนินงานการศึกษาค้นคว้า

ในการศึกษาค้นคว้าครั้งนี้ มีขั้นตอนการดำเนินการ ดังนี้

1. ศึกษาระบบการทำงานเดิมของหน่วยงานบริการผู้ใช้งานระบบเครือข่าย เพื่อใช้เป็นข้อมูลในการพัฒนาระบบใหม่ (Requirement Survey)
2. วิเคราะห์และออกแบบระบบ (System Analysis and Design)
3. พัฒนาระบบ (System Development)
4. ทดสอบและติดตั้ง Testing Server (Testing)
5. ประเมินผลระบบและสรุปผล (Evaluation)



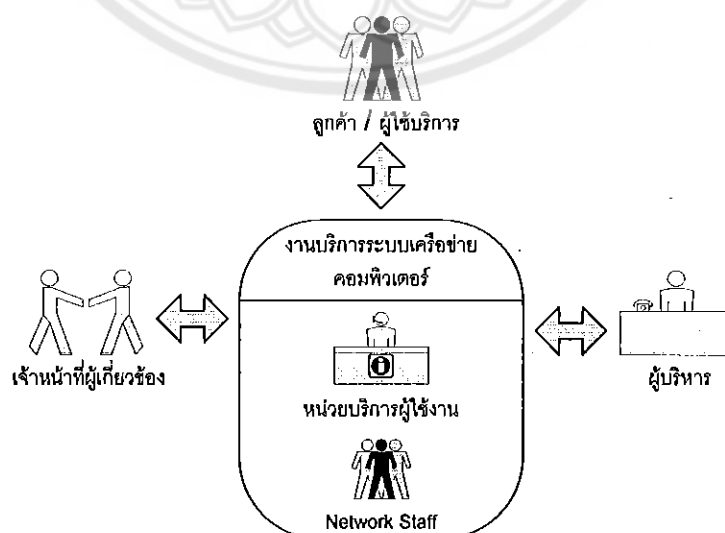
รูปที่ 3.1 ขั้นตอนการดำเนินงาน

3.1.1 การศึกษาระบบการทำงานแบบเดิม (Requirements Gathering)

ในการดำเนินการพัฒนาระบบ IT Helpdesk นี้ เริ่มต้นจากการศึกษาระบบงานเดิมและสำรวจความต้องการในการใช้งานระบบ IT Helpdesk หลังจากนั้นก็นำข้อมูลที่ได้ไปวิเคราะห์และออกแบบระบบที่จะทำการพัฒนา เมื่อทำการพัฒนาระบบเสร็จแล้ว ก็จะนำระบบมาทดสอบ และทำการประเมินระบบต่อไป

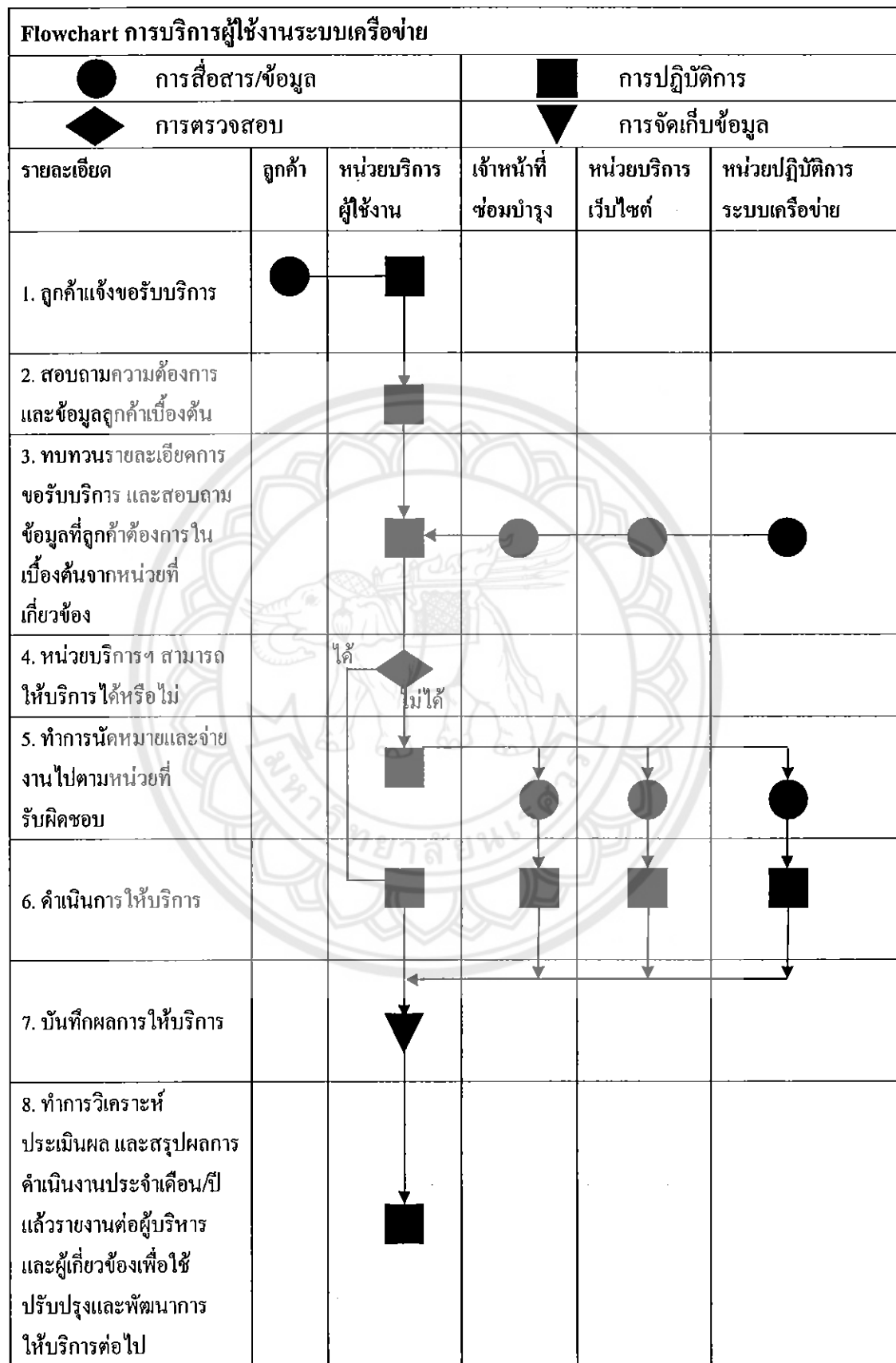
3.1.1.1 ระบบการทำงานแบบเดิม งานให้บริการระบบเครือข่ายฯ มีความเกี่ยวข้องกับบุคคลและหน่วยงานต่างๆ ได้แก่

- ลูกค้าหรือผู้ใช้บริการ (Users) ซึ่งอาจเป็นนิสิต นักศึกษาหรือบุคลากรของหน่วยงานนั้น จะเป็นผู้แจ้งปัญหาหรือแจ้งขอรับบริการซ่อมบำรุง
- เจ้าหน้าที่หน่วยซ่อมบำรุง (Net Staff) มีหน้าที่ให้บริการซ่อมบำรุงและแก้ไขปัญหาที่เกี่ยวข้องกับระบบเครือข่ายคอมพิวเตอร์ ทั้งในส่วนของฮาร์ดแวร์และซอฟต์แวร์ โดยหน่วยซ่อมบำรุงเป็นหน่วยงานในสังกัดของ งานบริการเครือข่ายคอมพิวเตอร์
- เจ้าหน้าที่หน่วยบริการผู้ใช้งาน (Helpdesk) เป็นหน่วยงานในสังกัดของ งานบริการระบบเครือข่ายคอมพิวเตอร์เช่นกัน ทำหน้าที่ให้บริการแนะนำการใช้งานระบบเครือข่าย และแก้ปัญหาการใช้งานเบื้องต้น เป็นผู้รับแจ้งการขอใช้งานบริการต่างๆ จากผู้ใช้งาน และติดต่อประสานงานในการจัดส่งเจ้าหน้าที่ซ่อมบำรุงไปให้บริการแก่ผู้ใช้งาน
- เจ้าหน้าที่งานอื่นๆ ที่เกี่ยวข้อง ได้แก่ หน่วยงานตลาดและประชาสัมพันธ์ ซึ่งจะแจ้งข่าวสารต่างๆ และผู้บริหารซึ่งจะคอยติดตามการทำงานของงานบริการระบบเครือข่ายคอมพิวเตอร์



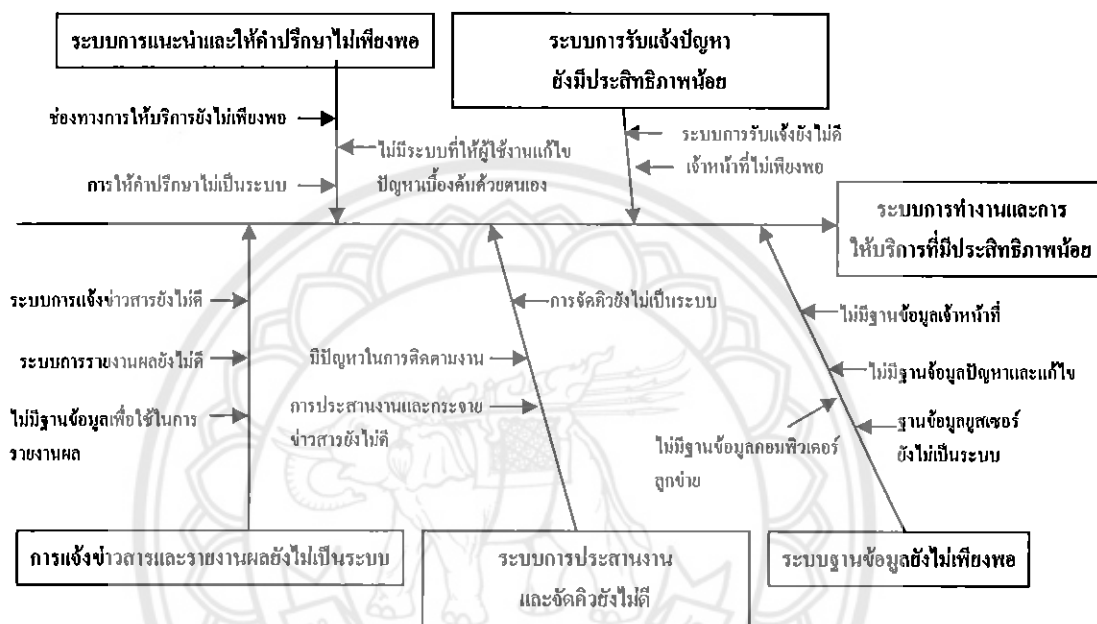
รูปที่ 3.2 หน่วยบริการผู้ใช้งานและผู้เกี่ยวข้องในระบบการทำงานแบบเดิม

ตารางที่ 3.1 ผังขั้นตอนระบบการทำงานแบบเดิม (Work Flow Diagram)



3.1.1.2 ปัญหาของระบบเดิมและความต้องการของระบบที่จะทำการพัฒนาใหม่
 เพื่อให้การพัฒนาระบบ IT Helpdesk เป็นไปในทิศทางที่ถูกต้อง สามารถปรับปรุงระบบ
 การทำงาน และตอบสนองความต้องการของผู้ใช้ได้อย่างเหมาะสม โดยมีหัวข้อที่สำคัญ ดังนี้

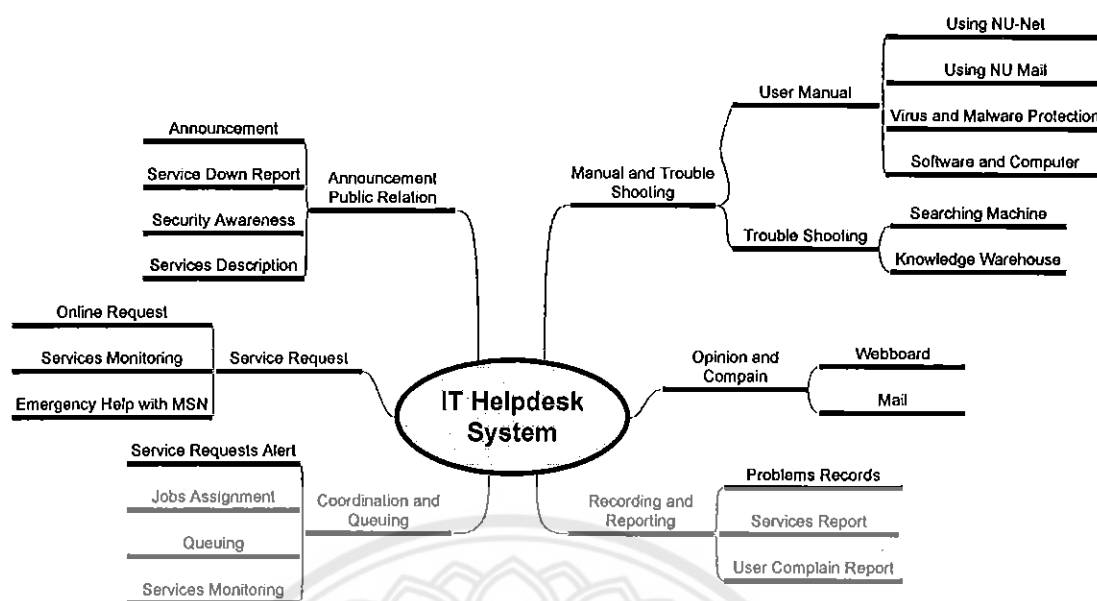
- ปัญหาการใช้บริการระบบเครือข่ายที่เคยเกิดขึ้นและวิธีการแก้ปัญหา
- กระบวนการทำงานที่ควรปรับปรุง
- เอกสาร ข้อมูล และผู้ที่เกี่ยวข้อง



รูปที่ 3.3 ปัญหาของการให้บริการในระบบเดิม

จากการสำรวจความต้องการของระบบพบว่ามีปัญหาและความต้องการ ดังนี้

1. การให้คำแนะนำและแก้ปัญหาการใช้งานควรนำระบบเว็บไซต์เข้ามาช่วยเพื่อเพิ่มประสิทธิภาพการให้บริการ
2. การแจ้งขอรับบริการซ่อมบำรุง ควรให้สามารถแจ้งออนไลน์ได้ผ่านทางเว็บไซต์
3. การประสานงานของเจ้าหน้าที่และการจัดลำดับการให้บริการซ่อมบำรุงควรนำระบบเทคโนโลยีสารสนเทศต่างๆเข้ามาช่วยในการดำเนินงาน
4. ควรมีระบบฐานข้อมูลในการจัดเก็บสถิติของปัญหาและการให้บริการ

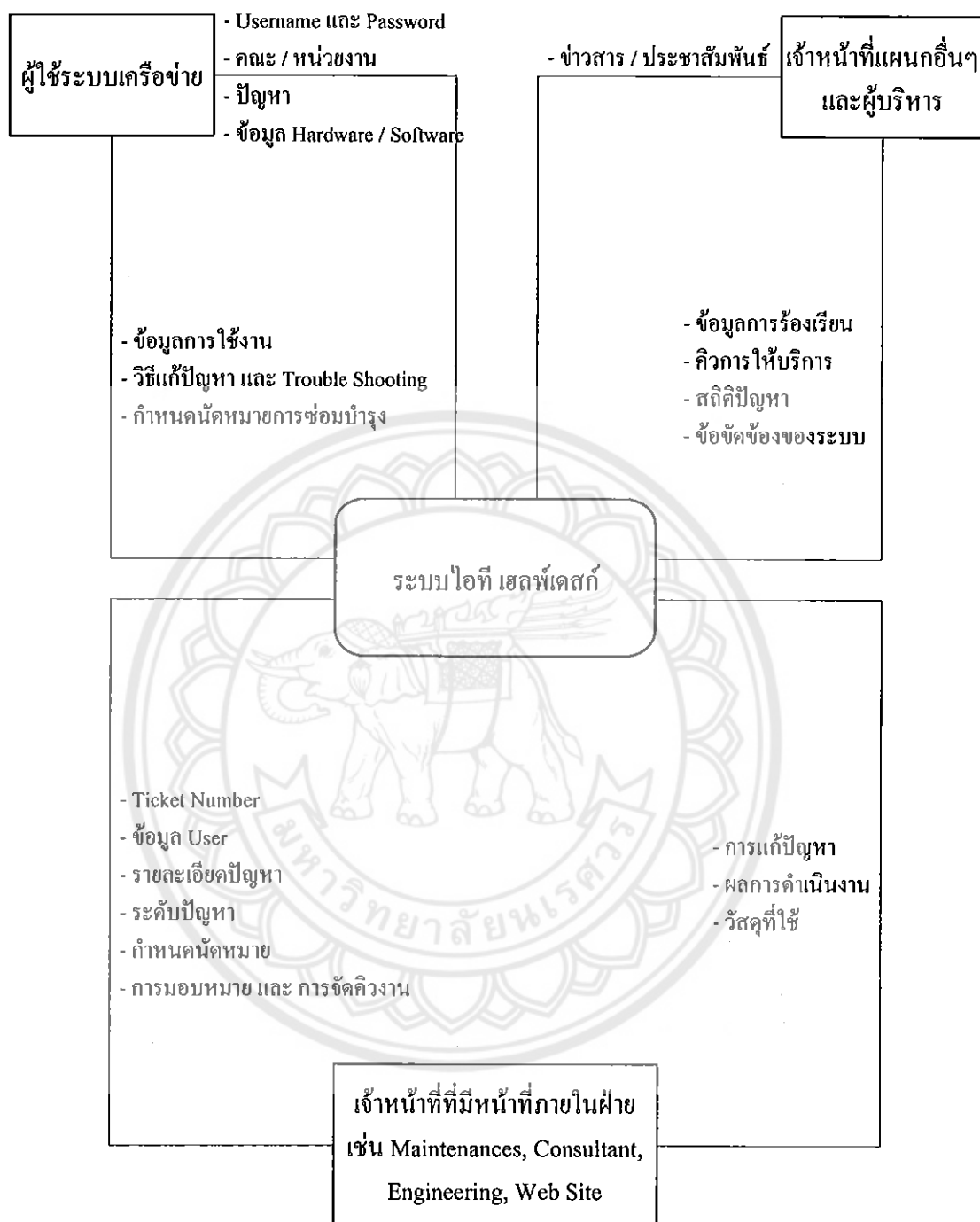


รูปที่ 3.4 ระบบ IT Helpdesk System Requirements

3.1.2 การวิเคราะห์และออกแบบระบบ

แนวทางการพัฒนาระบบ IT Helpdesk จะทำการนำเทคโนโลยีสารสนเทศ โดยเฉพาะระบบ Web Application มาประยุกต์ใช้กับการให้บริการแก้ปัญหของผู้ใช้งานระบบเครือข่ายและกระบวนการทำงานของเจ้าหน้าที่ผู้ให้บริการ โดยมีส่วนประกอบดังนี้

1. การแจ้งข่าวสาร ประชาสัมพันธ์ (Announcement)
2. การแนะนำการใช้งานและแก้ปัญหการใช้งานระบบเครือข่ายเบื้องต้นด้วยตนเอง (Self Help)
3. การรับแจ้งปัญหาการใช้งานและขอใช้บริการซ่อมบำรุงนอกสถานที่ (Service Request)
4. การประสานและจัดลำดับงานของเจ้าหน้าที่ให้บริการระบบเครือข่าย (Coordinating and Queuing)
5. การรับความคิดเห็นและข้อร้องเรียน (Web Board)
6. การเก็บสถิติปัญหา ผลให้บริการการแก้ไข รวมทั้งการรายงานผล (Services Report)

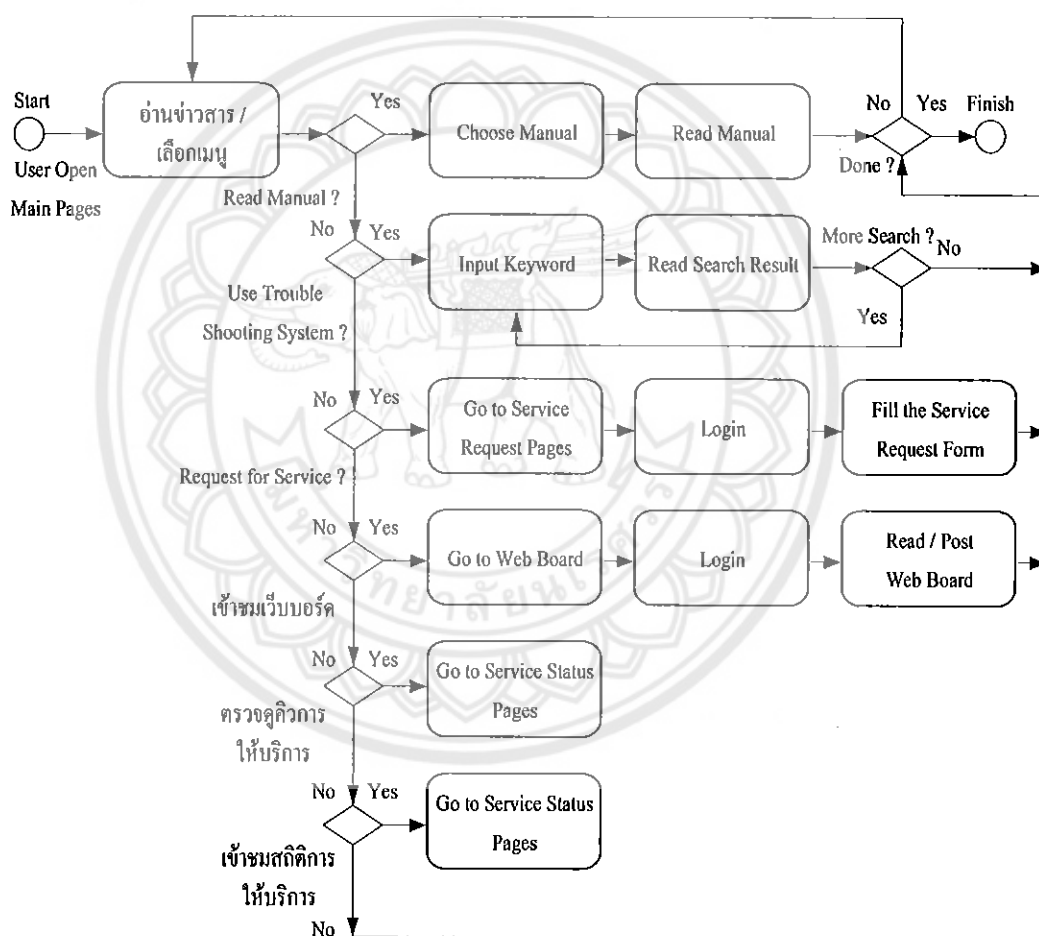


รูปที่ 3.5 ผังระบบ NU-NET Helpdesk

3.1.2.1 ระบบ NU-NET Helpdesk แบ่งได้เป็น 3 ส่วนหลักๆ คือ

1. ส่วนของผู้ใช้บริการ (User)

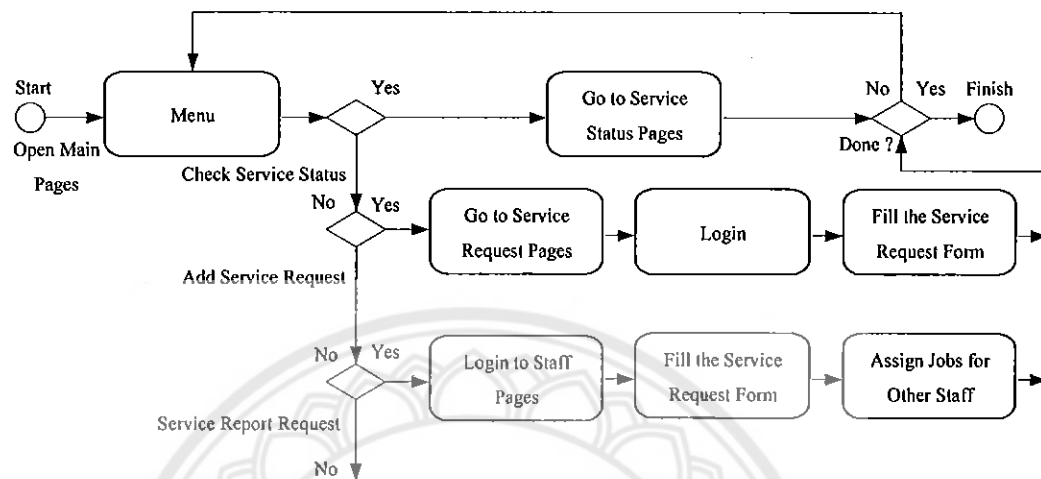
- การรับส่งข่าวสาร ประชาสัมพันธ์
- วิธีการใช้งานและการแก้ไขปัญหาเบื้องต้นด้วยตนเอง
- การแจ้งขอการซ่อมบำรุง
- การตรวจสอบลำดับการให้บริการ
- สามารถเข้าสู่สถิติการให้บริการซ่อมบำรุงได้
- การเสนอความคิดเห็นและการร้องเรียนผ่านทางเว็บบอร์ด



รูปที่ 3.6 พัง Activities ของผู้ให้บริการ (User)

2. ส่วนของเจ้าหน้าที่ซ่อมบำรุง (Net Staff)

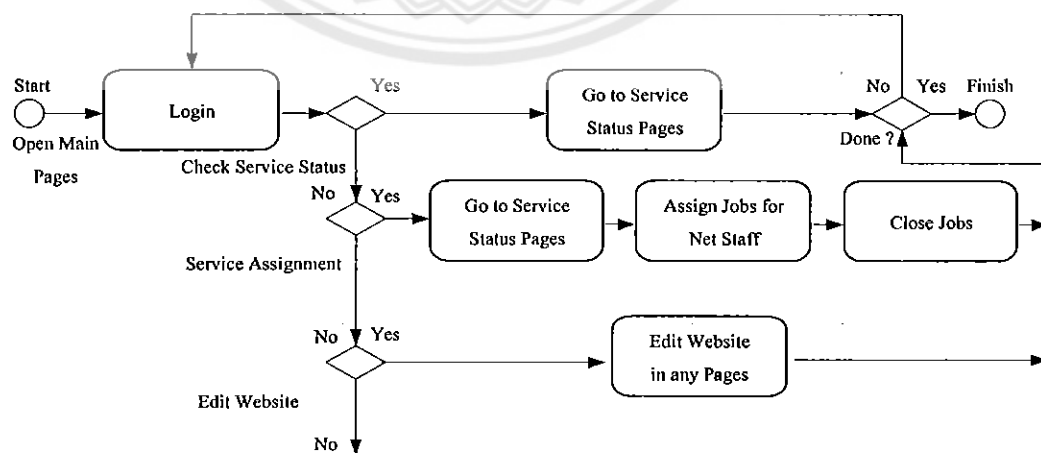
- สามารถใช้งานทุกอย่างได้เหมือน User
- การรายงานผลการให้บริการซ่อมบำรุงหลังจากงานเสร็จ



รูปที่ 3.7 ผัง Activities ของเจ้าหน้าที่ซ่อมบำรุง (Net Staff)

3. ส่วนของเจ้าหน้าที่ดูแลระบบ Helpdesk (Hdsa)

- สามารถดำเนินการได้ทุกส่วนของระบบ
- การติดตามสถานการณ์ให้บริการ
- มอบหมายงานให้กับเจ้าหน้าที่ซ่อมบำรุง
- แจ้งข่าวสาร ประชาสัมพันธ์

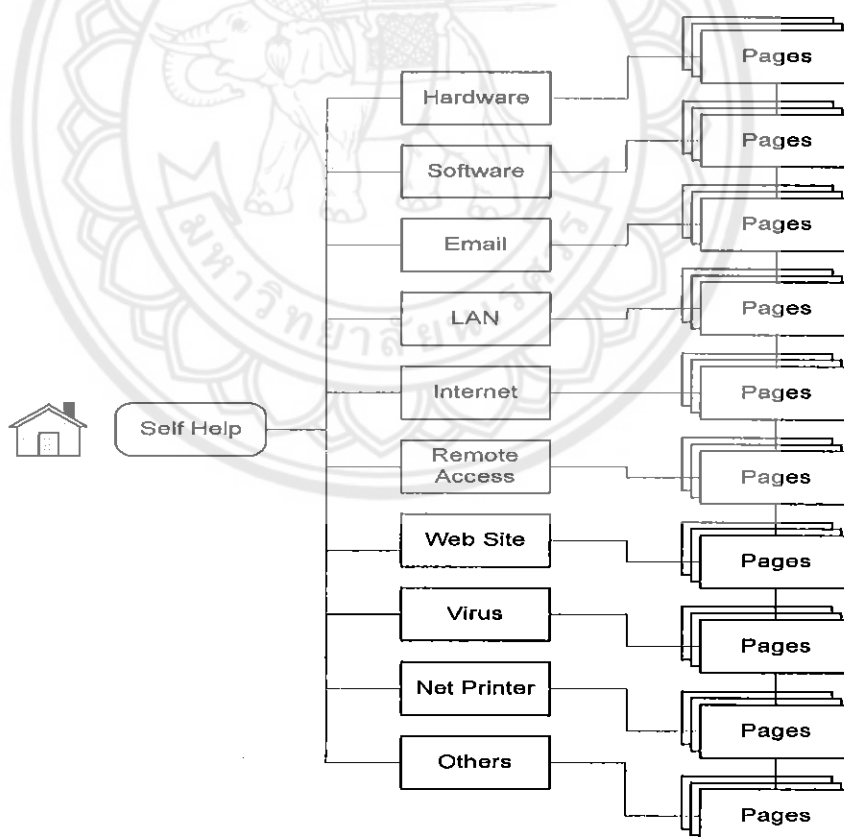


รูปที่ 3.8 ผัง Activities ของเจ้าหน้าที่ดูแลระบบ Helpdesk (Hdsa)

3.1.2.2 ระบบแก้ปัญหาการใช้งานระบบเครือข่ายเบื้องต้นด้วยตนเอง (Self Help)

เป็นเว็บเพจสำหรับการแนะนำการใช้งาน หรือ การแก้ปัญหาการใช้งานระบบเครือข่าย ฮาร์ดแวร์และซอฟต์แวร์คอมพิวเตอร์ โดยแบ่งปัญหาเป็น 10 ประเภท ดังนี้

- ปัญหาเกี่ยวกับ Hardware
- ปัญหาเกี่ยวกับ Software
- ปัญหาเกี่ยวกับ E-mail
- ปัญหาเกี่ยวกับระบบเครือข่าย (LAN)
- ปัญหาเกี่ยวกับการใช้งาน Internet
- ปัญหาการเชื่อมต่อ NU-NET จากโมเด็ม (Remote Access)
- ปัญหาเกี่ยวกับ Website
- ปัญหาเกี่ยวกับ Virus Computer และ Malware
- ปัญหาการใช้งาน Network Printer
- ปัญหาการใช้งานอื่นๆ

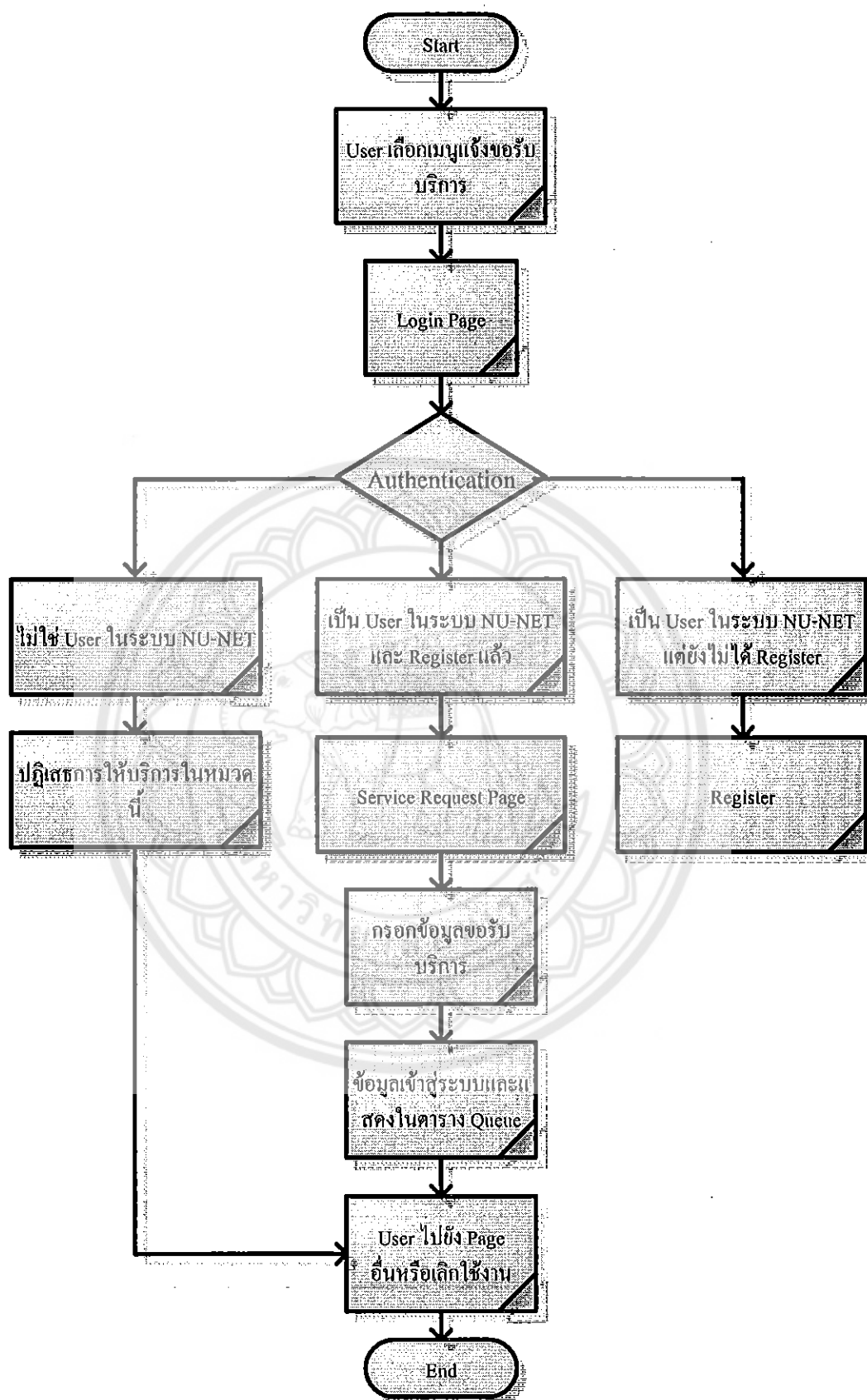


รูปที่ 3.9 ผังโครงสร้างของระบบแก้ปัญหาด้วยตนเอง (Self Help)

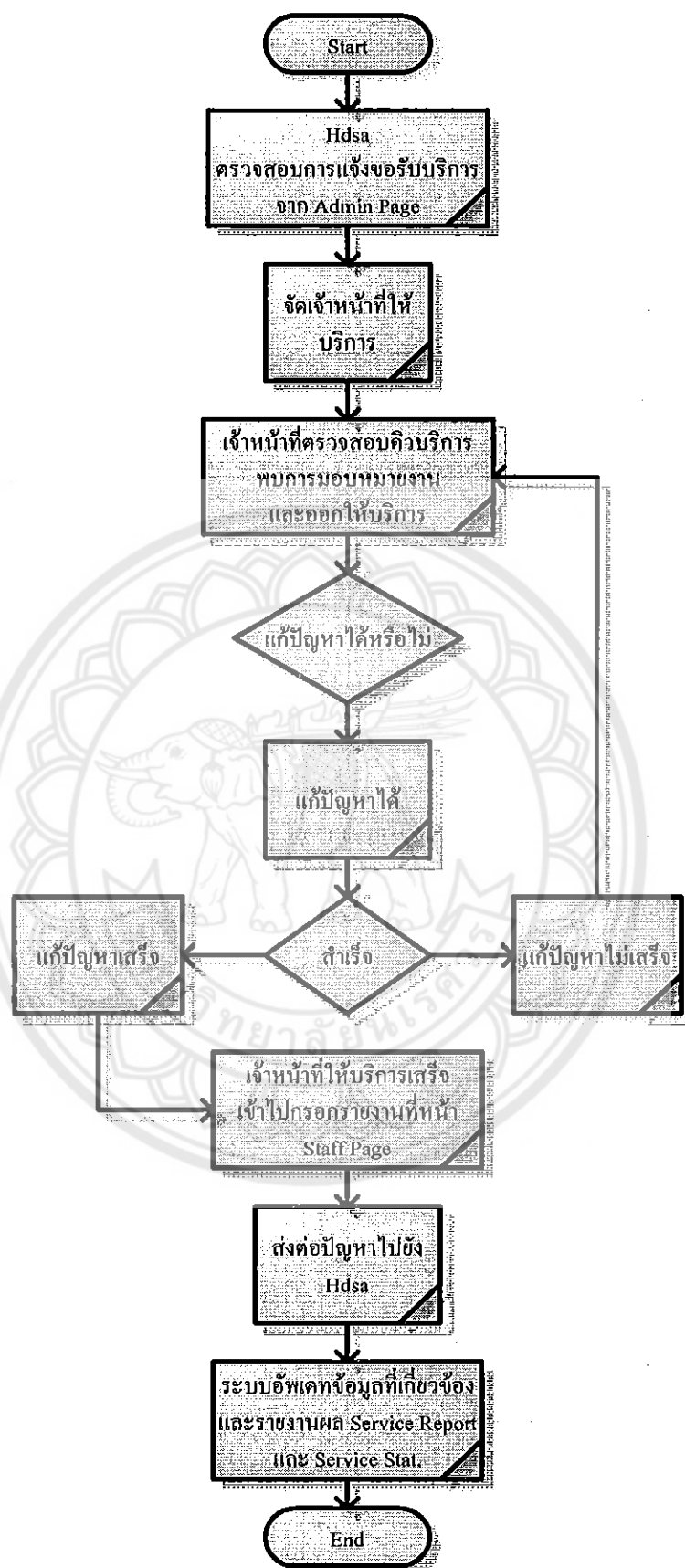
3.1.3 การพัฒนาระบบการขอรับบริการและแสดงลำดับการให้บริการ ประกอบด้วย

- การขอรับบริการ (Services Request)
- การจัดลำดับการให้บริการ (Queuing)
- การประสานงานและมอบหมายงานให้กับเจ้าหน้าที่ (Jobs Assignment)
- การรายงานผลการให้บริการ (Services Report)





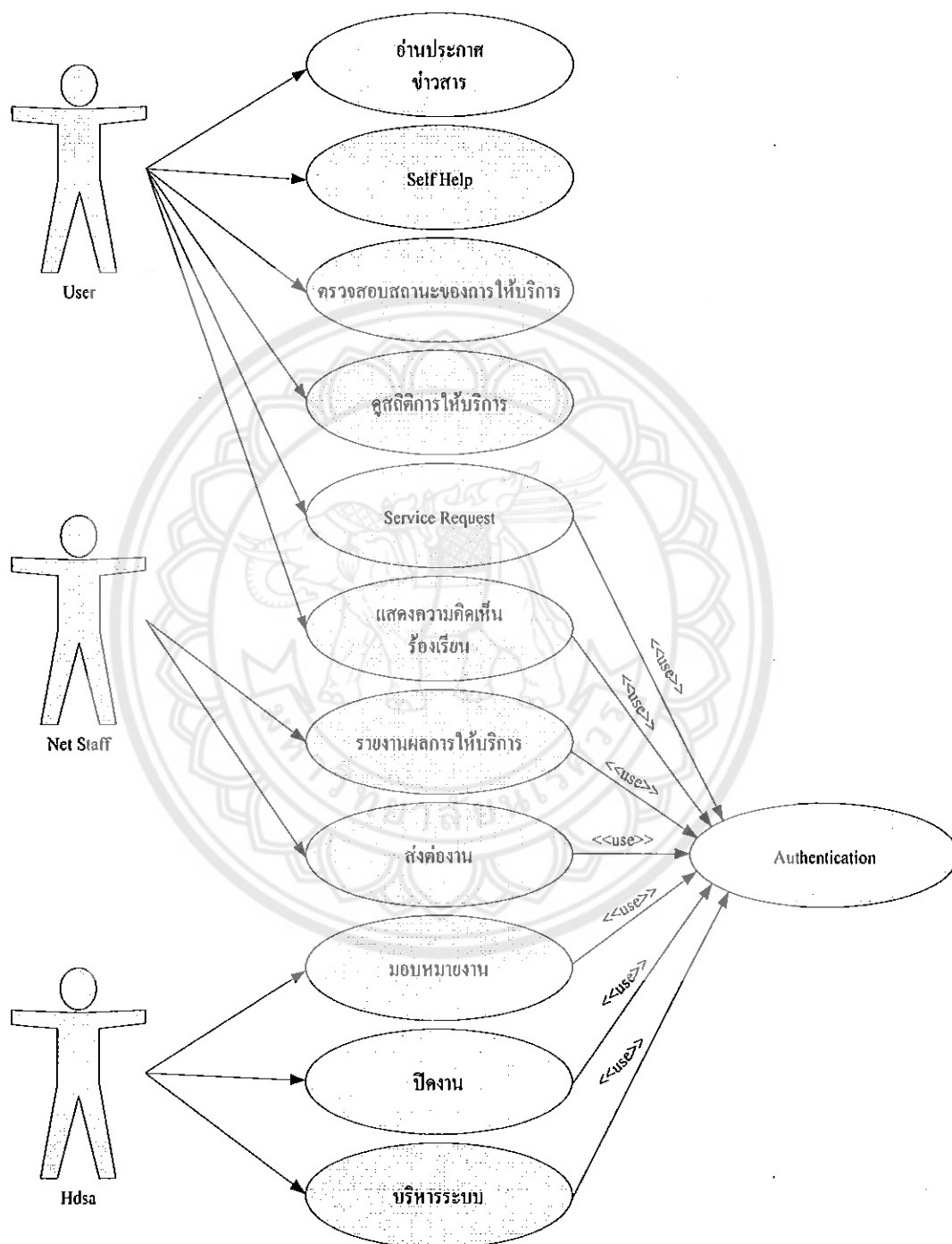
รูปที่ 3.10 ฟังก์ชันตอนการขอรับบริการในส่วนของผู้ใช้งานทั่วไป



รูปที่ 3.11 ผังขั้นตอนการขอรับบริการในส่วนของผู้ดูแลระบบ

3.1.3.1 NU-NET Help Desk Use Cases

ยูสเคสไดอะแกรมแสดงให้เห็นถึงความสัมพันธ์ของผู้ใช้ระบบ (Actor) และการทำงานของระบบ (Use Cases)



รูปที่ 3.12 ฟังแสดง Use Case ของ NU-NET Helpdesk

ตารางที่ 3.2 รายละเอียดของ NU-NET Helpdesk Use Cases

Actors	Use Case	Description
Users	อ่านประกาศข่าวสาร	อ่านข้อมูลข่าวสารที่อยู่ในหน้า Announce Page
	ระบบแก้ไขปัญหาเบื้องต้นด้วยตนเอง (Self Help)	อ่านวิธีการแก้ปัญหการใช้งาน
	ตรวจสอบสถานะ การให้บริการ	ตรวจสอบผลการแจ้งขอรับบริการซ่อมบำรุง และลำดับการให้บริการ
	คู่มือการให้บริการ	คู่มือการให้บริการซ่อมบำรุงที่ผ่านมา
	Services Request	แจ้งขอรับบริการซ่อมบำรุง
	แสดงความคิดเห็น / ร้องเรียน	แสดงความคิดเห็น หรือร้องเรียนผ่านทางเว็บบอร์ด
	Log in	แสดงตัวตนเพื่อเข้าสู่ระบบในกรณีที่ต้องการเข้าไปยังหน้า Service Request, Web Board รวมทั้งการตรวจสอบกรณีที่ต้องการเข้าสู่ระบบของเจ้าหน้าที่ทั้ง Net Staff และ Hdsa
Net Staff	รายงานผลการให้บริการ	กรอกรายงานการให้บริการเมื่อให้บริการเสร็จ
Hdsa	มอบหมายงาน	มอบหมายงานและจัดคิวการให้บริการแก่เจ้าหน้าที่ Net Staff ตามความเหมาะสม
	บริหารระบบ	เพิ่ม ลด แก้ไข ในข้อมูลในส่วนต่างๆของระบบ

3.1.3.2 การแสดงสถิติการให้บริการ

ระบบ NU-NET Help Desk จะแสดงสถิติการให้บริการที่ผ่านมา โดยแบ่งเป็น 3 ส่วน คือ

1. สถิติผู้ขอรับบริการ แบ่งเป็นนิสิต บุคลากร และรวมทั้งหมด
2. สถิติของการให้บริการ โดยแบ่งตามประเภทของปัญหา
3. สถิติการให้บริการของเจ้าหน้าที่แต่ละคน (Work Load)

3.1.4 การออกแบบโครงสร้างของเว็บเพจ

3.1.4.1 การออกแบบหน้าจอ

การออกแบบหน้าจอการใช้งาน คำนึงถึงความชัดเจนในการนำเสนอและการนำทางของเว็บไซต์ (Web Navigation) ที่ชัดเจน ไม่สับสน โดยการแสดงผล ออกแบบให้มองเห็นหน้าจอได้ชัดเจนทั้งขนาดความละเอียดหน้าจอแบบ 800x600 Pixels และ 1024x768 Pixels

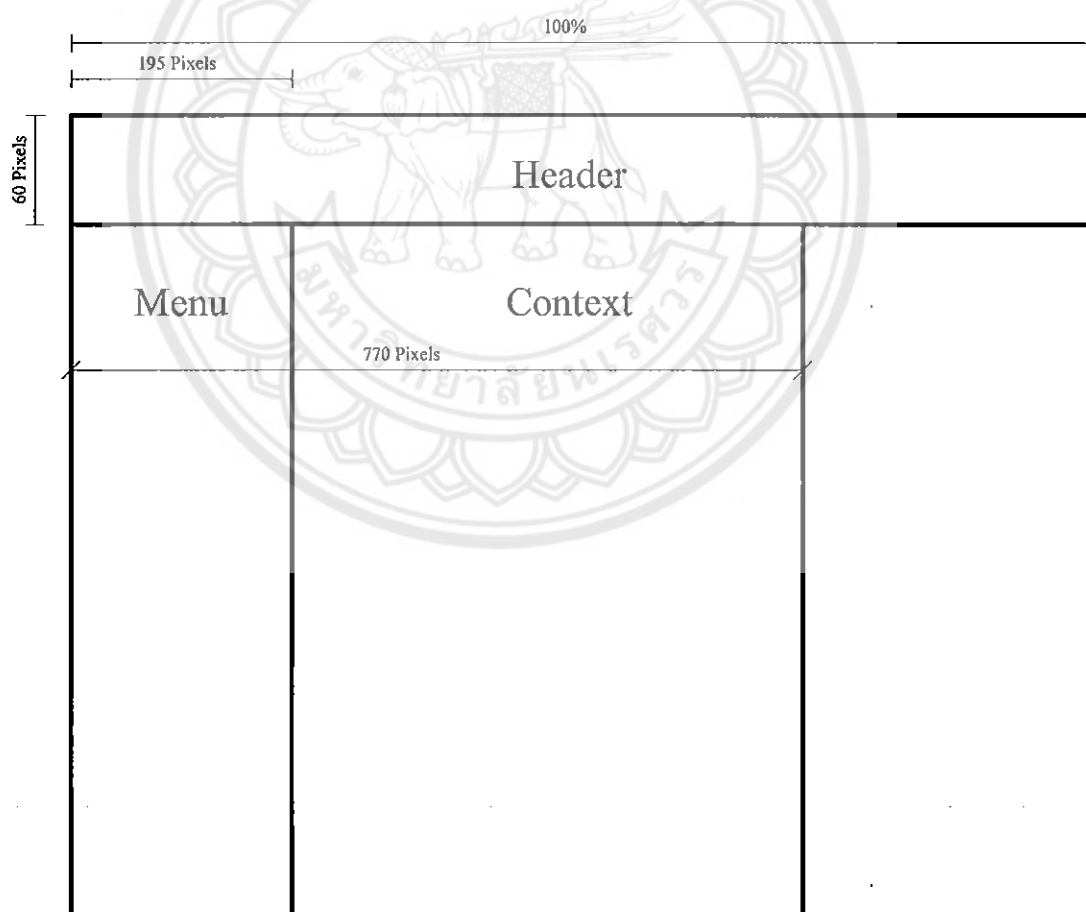
ส่วนประกอบหลักของแต่ละหน้าจอ ได้แก่

Header เป็นการแสดงชื่อของระบบ NU-NET Helpdesk โดยจะอยู่ด้านบนสุดของเว็บเพจ

Menu เป็นส่วนที่แสดงเมนูการใช้งาน โดยจะอยู่ด้านซ้ายของเว็บเพจ

Context เป็นส่วนของเนื้อหา โดยจะอยู่ด้านบน จะแสดงว่าเนื้อหาหน้านั้นๆ เป็นเรื่องอะไร ซึ่งจะสัมพันธ์กับระบบนำทางของเว็บไซต์

รูปแบบของหน้าจอที่จะถูกใช้ ทั้งในส่วนของผู้ใช้งานทั่วไป, เจ้าหน้าที่บริการระบบเครือข่าย และเจ้าหน้าที่ดูแลระบบเซิร์ฟเวอร์

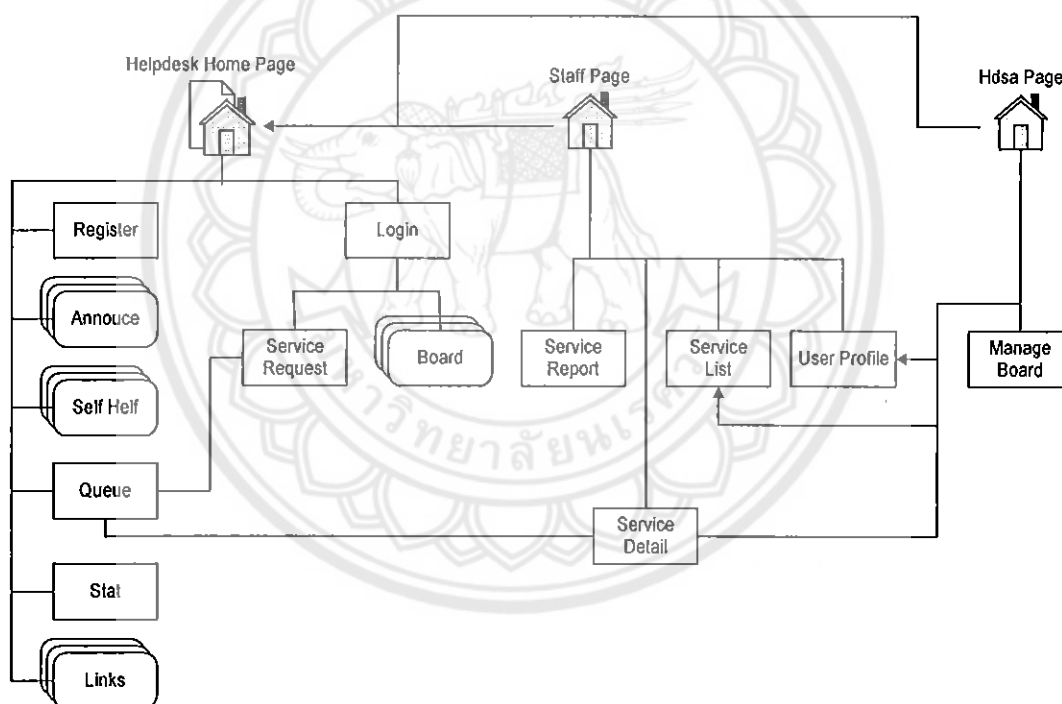


รูปที่ 3.13 โครงสร้างของเว็บเพจ

3.1.4.2 โครงสร้างของเว็บไซต์

โครงสร้างของเว็บไซต์ จะแบ่งเป็น 3 ส่วนหลักตามกลุ่มผู้ใช้งานระบบ ได้แก่

1. ส่วนของผู้ขอใช้บริการ (Users) โดยจะเริ่มต้นที่ Helpdesk Home Page สามารถเข้าไปยังหน้าต่างๆได้ โดยไม่ต้องล็อกอิน ยกเว้นหน้าในส่วนของการแจ้งขอรับบริการ (Service Request) และ หน้าเว็บบอร์ด ผู้ที่จะเข้าใช้งานได้ต้องมีรหัสยูสเซอร์เนมและพาสเวิร์ดของผู้ใช้ระบบ NU-NET ที่ถูกต้อง จึงจะสามารถใช้งานได้
2. ส่วนของเจ้าหน้าที่ผู้ให้บริการ (Net Staff) จะต้องผ่านการล็อกอิน เพื่อยืนยันตัวตนของเจ้าหน้าที่ก่อนเข้าใช้งาน
3. ส่วนของผู้ดูแลระบบ Helpdesk (Hdsa) จะต้องผ่านการล็อกอิน เพื่อยืนยันตัวตนของเจ้าหน้าที่ก่อนการใช้งานเช่นกัน โดยผู้ดูแลระบบจะเป็นผู้จัดการระบบทั้งหมด รวมทั้งมอบหมายงานให้กับเจ้าหน้าที่ผู้ให้บริการด้วย



รูปที่ 3.14 แสดง Site Map ของเว็บไซต์ระบบ NU-NET Help Desk

3.1.5 การประเมินผลระบบ

ผู้ศึกษาได้ดำเนินการสร้างแบบประเมินคุณภาพ ดังนี้

1. ศึกษาจากเอกสารและงานวิจัยที่เกี่ยวข้อง
2. ศึกษาจากคำราที่เกี่ยวกับการพัฒนาเว็บไซต์และการสร้างแบบสอบถาม
3. สร้างแบบสอบถามแบบมาตรฐานตามมาตราส่วน 5 ระดับ
4. นำแบบสอบถามเสนอต่ออาจารย์ที่ปรึกษา เพื่อพิจารณาตรวจสอบนำแบบสอบถามมาทบทวนปรับปรุงและแก้ไข



บทที่ 4

การใช้งานโปรแกรม

4.1 ส่วนประกอบของตัวโปรแกรม

โปรแกรมจะแบ่งการทำงานออกเป็น 3 ส่วน ได้แก่ ส่วนของผู้ใช้งานทั่วไป, เจ้าหน้าที่ผู้ให้บริการระบบเครือข่าย และเจ้าหน้าที่ดูแลระบบ Helpdesk

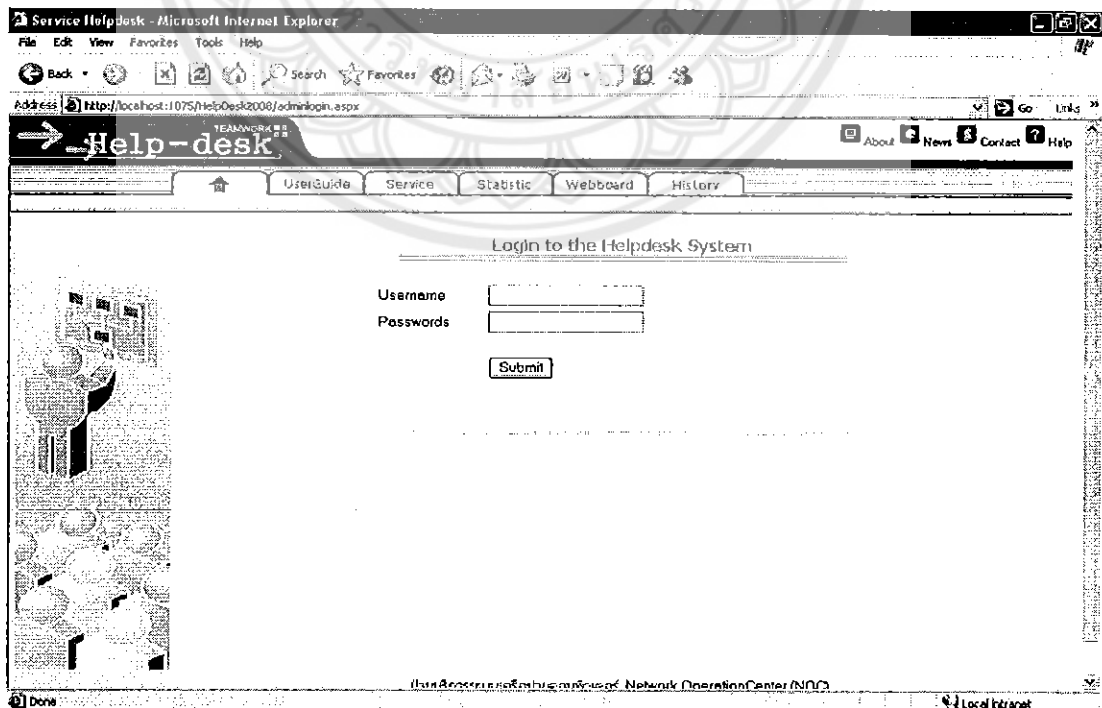
4.2 รูปแบบและวิธีการใช้งานของโปรแกรม

หน้าต่างหลักของโปรแกรม ซึ่งเป็นการแสดงผลแบบเว็บเพจ ในหน้านี้จะหน้าที่ให้ ผู้ใช้งานทำการล็อกอิน (Login to the System) เพื่อเข้าสู่ระบบการทำงาน และเพื่อเป็นการแสดง ตัวตนของผู้ใช้งานแต่ละระดับ เช่น ระดับ User, ระดับ Net Staff หรือระดับ Hdsa (เจ้าหน้าที่ดูแล Helpdesk)

ส่วนของเจ้าหน้าที่

(Adminlogin.aspx)

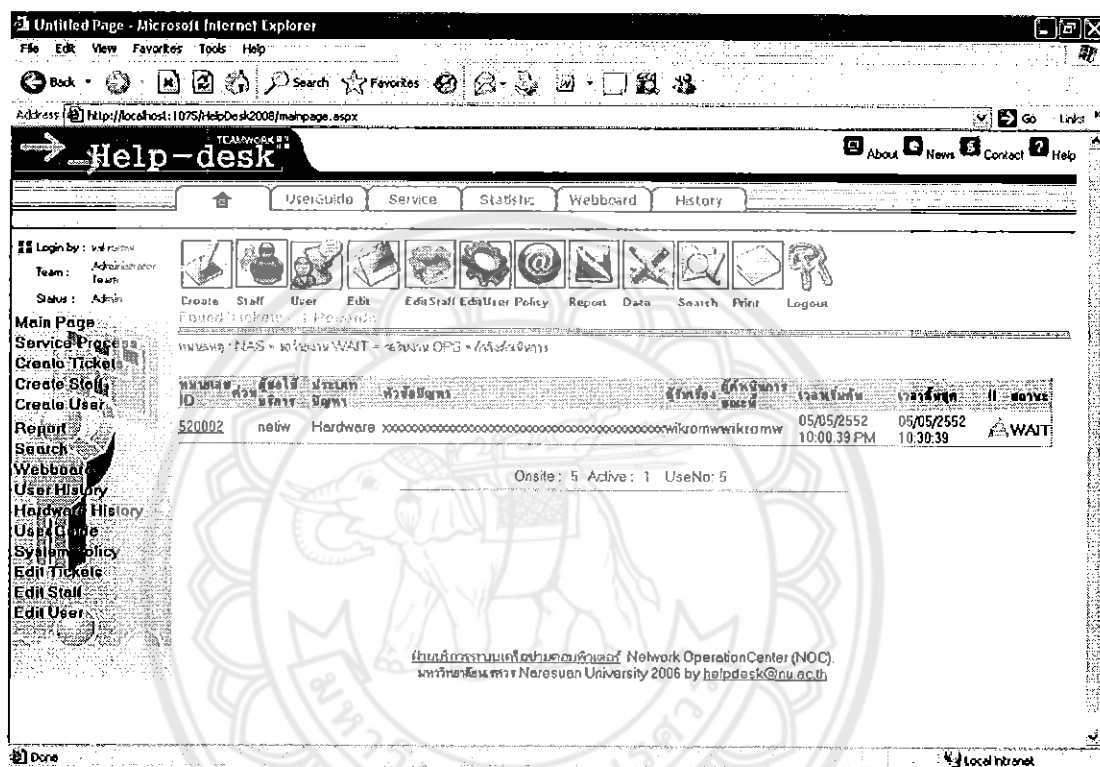
เจ้าหน้าที่ผู้ที่ลงทะเบียนและมีหน้าที่เกี่ยวข้อง หลังจากได้รับการจัดการสิทธิ์การเข้าถึง ข้อมูลแล้ว ทำการ Login ด้วย Username และ Passwords ของมหาวิทยาลัย เพื่อใช้งานระบบ



รูปที่ 4.1 หน้าสำหรับเจ้าหน้าที่ Login เข้าสู่ระบบ

(Mainpage.aspx)

หน้าแรกเมื่อเจ้าหน้าที่ Login เข้าสู่ระบบ ใช้บริหารจัดการงานของตัวเอง โดยสามารถเข้าถึงข้อมูลเฉพาะบุคคลได้เท่านั้น ประกอบด้วยเมนูหลัก และเมนูย่อยต่างๆ ที่ใช้ลิงค์ไปยังส่วนต่างๆ ของโปรแกรม และแสดงภาระงาน กรณีที่ได้รับมอบหมายงานจาก Hdsa ให้ไปปฏิบัติหน้าที่ตามนัดหมาย ซึ่งจะมีการ Refresh Page ทุกๆ 20 วินาที เพื่อ Update ข้อมูลภาระงานที่เข้ามาใหม่



รูปที่ 4.2 แสดงข้อมูลภาระงานของพนักงาน

(Forwardincident.aspx)

เมื่อเจ้าหน้าที่ได้รับมอบหมายงานแล้ว ต้องทำการรับงานที่ได้รับมอบหมาย โดยการคลิกที่ Accept เพื่อรับ Ticket เป็นลำดับแรก จากนั้น สถานะของ Ticket จะเปลี่ยนจาก WAIT (รอจ่ายงาน) เป็น OPS (อยู่ระหว่างดำเนินการ) เมื่อการดำเนินงานเสร็จสิ้น และถูกส่งกลับมายัง Hdsa สามารถ Close Ticket เพื่อปิดงานได้

ID	สถานะ	ผู้รับผิดชอบ	ทีม	เวลาที่ส่งต่องาน	ปัญหา	รายละเอียด
1	Income Process	wikromw	Administrator Team	05/05/2552 10:21:32	xxxxxx	xxxxxx

รูปที่ 4.3 แสดงการส่งต่อภาระงานและการปิดงาน

(Forwardticket.aspx)

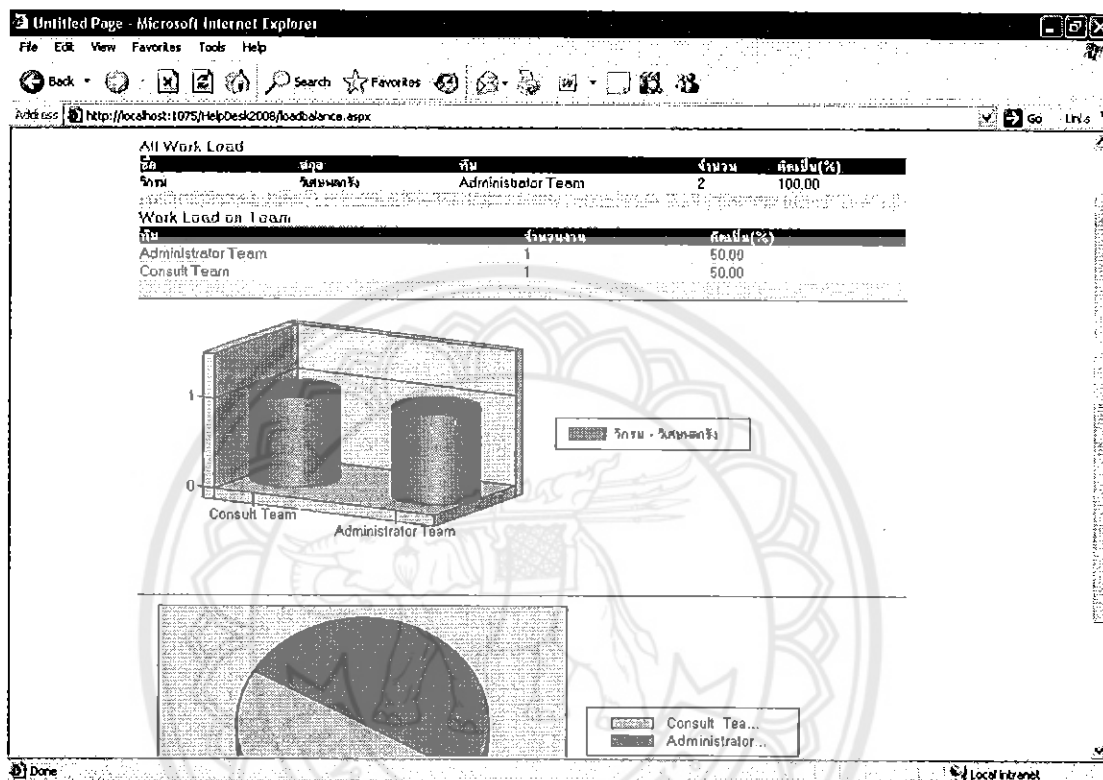
หน้าส่งต่องาน และรายงานผลการดำเนินงาน เมื่อดำเนินงานที่ได้รับมอบหมายเสร็จสิ้นในส่วนที่รับผิดชอบ ต้องกลับมากรอกผลการดำเนินการ และรายละเอียดต่างๆ หรือหากงานทั้งหมดยังไม่เสร็จสิ้น ต้องมีการประสานงานไปยังหน่วยอื่น ให้ทำการ Forward Ticket ไปยังลำดับความรับผิดชอบต่อไป

Ticket ID	520002
Status	WAIT
Priority	HW1
Request	ขอแจ้งรายการ (Request): Hardware
Category	Medium
Requested On	05/05/2552 10:00:39 P
Username	ไปรษณีย์
รายละเอียดการดำเนินการ	
E-Mail to LANNA	☐
E-Mail to NOC	☐
เลขที่โทรศัพท์	19388
บันทึก	

รูปที่ 4.4 แสดงการรายงานผลการดำเนินงานก่อนการส่งต่องานหรือปิดงาน

(Loadbalance.aspx)

Hdsa ซึ่งมีหน้าที่รับและกระจายงานไปยังเจ้าหน้าที่ สามารถดูโหลดการทำงานของเจ้าหน้าที่แต่ละคน เพื่อตัดสินใจว่าควรจะมอบหมายงานให้เจ้าหน้าที่คนใด โดยจะแสดงสถิติเป็นเปอร์เซ็นต์และกราฟ ของงานต่อวัน ทั้งวัน และของทีมทั้งหมด



รูปที่ 4.5 แสดงข้อมูลการประมวลผลภาระงานของพนักงานสำหรับ Hdsa

(Createticket.aspx)

Hdsa หรือผู้มีสิทธิในการสร้าง Ticket สามารถใช้แบบฟอร์มในการรับ Service Request จากผู้ขอใช้บริการ โดยเรียกฐานข้อมูลบุคคลผู้ขอใช้บริการขึ้นมาแสดงและลงรายละเอียดของการขอใช้บริการเช่น ประเภทการขอใช้บริการ เอกสารแนบ ช่องทางการแจ้งปัญหา ประเภทของปัญหา เลขครุภัณฑ์ เป็นต้น เมื่อบันทึกเสร็จสิ้นแล้ว โปรแกรมจะ Print เอกสารเป็น แบบฟอร์มออกมา

The screenshot shows a web browser window titled 'Help-desk' with the URL 'http://localhost:1075/HelpDesk2008/CreateTicket.aspx'. The page features a navigation menu on the left with options like 'Main Page', 'Service Process', 'Create Tickets', 'Create Staff', 'Create User', 'Report', 'Search', 'Webboard', 'User History', 'Hardware History', 'User Guide', 'System Policy', 'Edit Tickets', 'Edit Staff', and 'Edit User'. The main content area is titled 'Create Ticket' and contains a form with the following fields and options:

- Team:** Admin
- Status:** Admin
- นามสกุล:** ID 520003
- ชื่อ:** [Text Field]
- ประเภทขอใช้บริการ:** Radio buttons for 'ขอใช้บริการ (Request)' and 'ขอแจ้งปัญหา (Problem)'. 'Request' is selected.
- Upload File:** A button labeled 'Upload File' and a 'Browse...' button.
- ช่องทางแจ้งปัญหา:** A dropdown menu with 'Walk in' selected.
- สถานที่ขอใช้บริการ:** A text field.
- Serial/รหัสอุปกรณ์:** A text field.
- สถานะ:** A dropdown menu with 'Income Process' selected.
- ระดับขอแจ้งปัญหา:** A dropdown menu with 'Medium' selected.
- เวลาประมาณการ:** A dropdown menu with 'เลือกตามใจ' selected.
- เวลาเริ่ม:** A text field.
- ทีมที่รับผิดชอบ:** A dropdown menu with 'Administrator Team' selected.
- จำนวน:** A text field.

At the bottom of the form, there are buttons for 'บันทึก' (Save) and 'ยกเลิก' (Cancel). The status bar at the bottom indicates 'Local intranet'.

รูปที่ 4.6 แสดงแบบฟอร์มบันทึกข้อมูล Service Request ของผู้ขอใช้บริการ

(Printdoc.aspx)

หน้าออกแบบฟอร์มเอกสาร Service Request

Untitled Page - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Search Favorites

Address http://localhost:1075/helpDesk2008/printdoc.aspx?T_ID=520001 Go

1 of 1 100% Find Next Select a format Export

แบบฟอร์มการให้บริการ
งานบริการระบบเครือข่ายคอมพิวเตอร์

รหัส : FM-NET-002
ฉบับปัจจุบันที่ : 3

Case No. 520001 วันที่ 5/19/2009 12:00:00 AM

ข้าพเจ้า: รศ.ดร.เนติ ศกุล: วารชยุช

ชื่อหน่วยงาน/คณะ: หน่วยงานอื่นๆ ติดต่อ/สาขา: ศูนย์บริการวิชาการ

โทรศัพท์ภายใน: 4269 โทรศัพท์เคลื่อนที่: E-Mail: netiw@nu.ac.th

ส่วนที่ 1 ส่วนรับ Helpdesk

1.ประเภทการขอใช้บริการ อาจารย์มี

2.ช่องทางสายเร่ง Phone ปณเขต: SW2

3.ประเภทบริการ ขอใช้บริการ (Request)

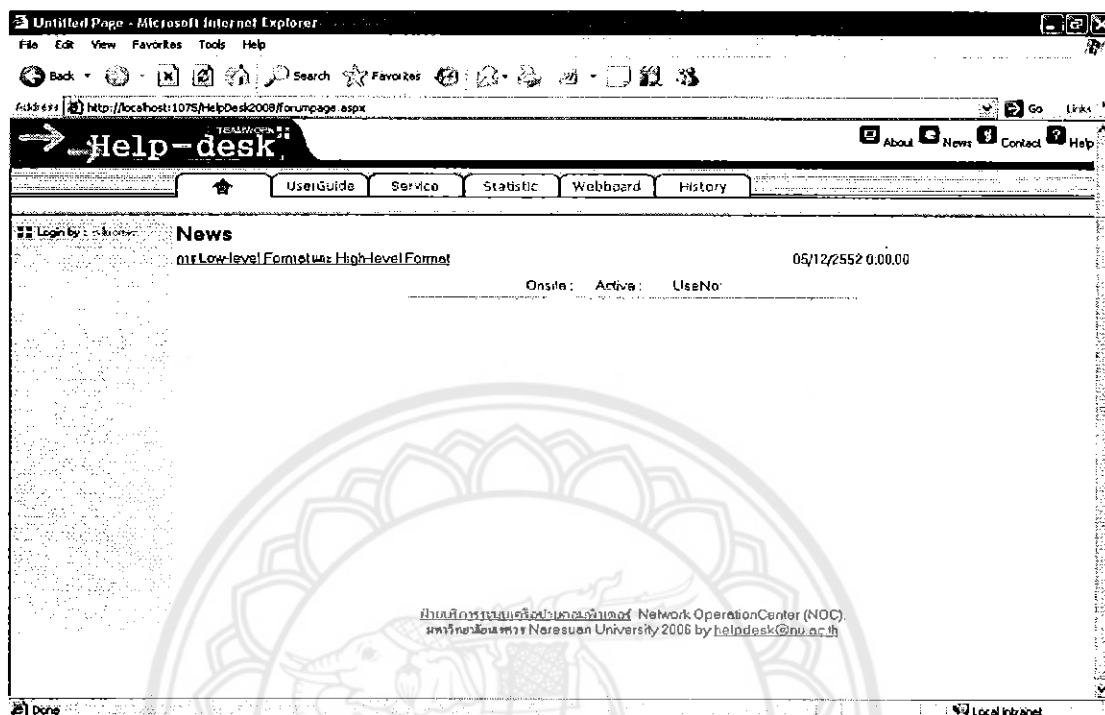
4.ระดับความเร่งด่วน Medium เวลาเริ่มรับ: 02/05/2552 10:49:47 เวลาสิ้นสุด: 02/05/2552 11:03:47

Done Local intranet

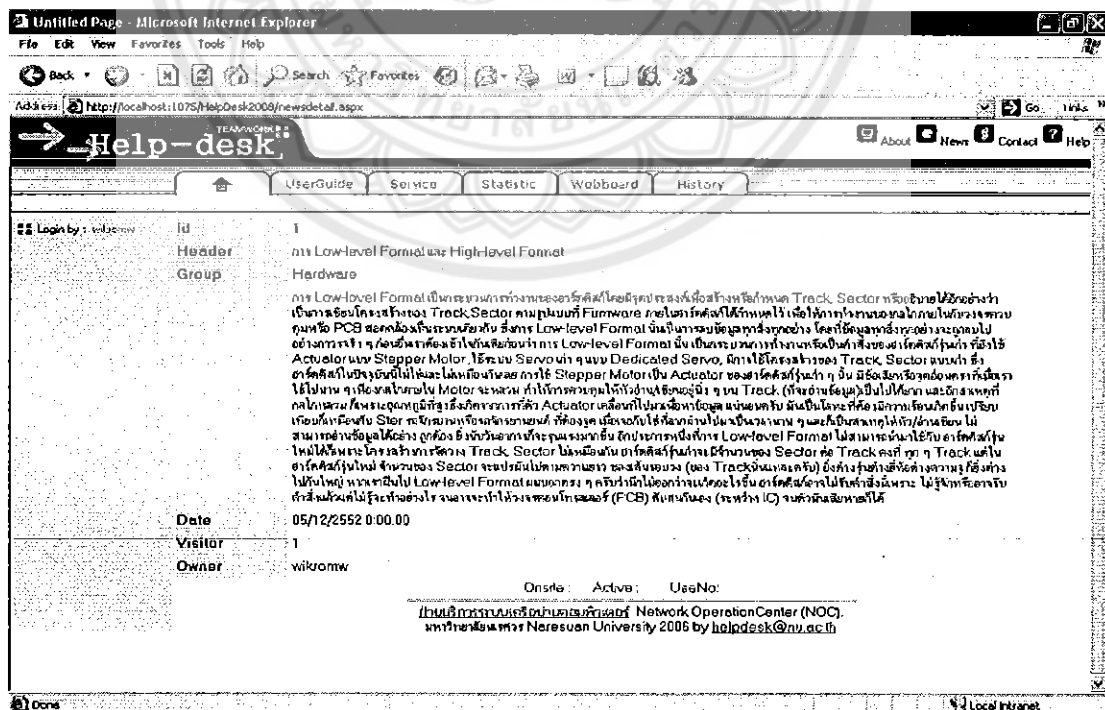
รูปที่ 4.7 แบบฟอร์มใบงาน

(Forumpage.aspx, Newsdetail.aspx)

หน้าแสดงข่าวสารและบทความต่างๆ



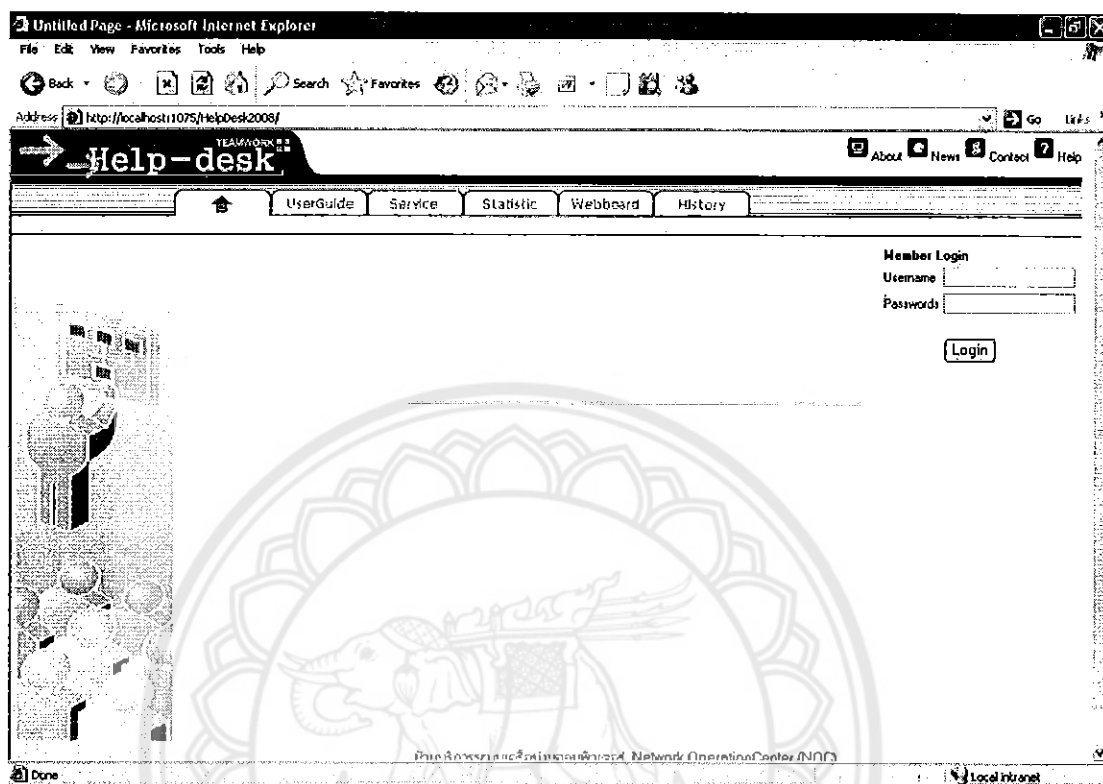
รูปที่ 4.10 แสดงรายการข่าว



รูปที่ 4.11 แสดงรายละเอียดของข่าว

(Default.aspx)

แสดงเป็นหน้าแรกของเว็บไซต์ และใช้เป็นหน้า User Login



รูปที่ 4.12 หน้าแรกของเว็บไซต์

(Register.aspx)

หน้าลงทะเบียนของผู้ใช้บริการ กรณีที่สมัครด้วยตัวเอง

Help-desk

Address: http://localhost:1075/helpdesk2008/register.aspx

Navigation: Home, UserGuide, Service, Statistic, Webboard, History

Form Fields:

- ชื่อ:
- นามสกุล:
- สถานะ:
- หน่วยงาน:
- ภาควิชา:
- E-mail:
- โทรศัพท์:
- ที่อยู่:
- Username:
- Passwords:
- Confirm Password:

ตกลง

Onsite : Active : UseNo,

ให้บริการศูนย์คอมพิวเตอร์ Network Operation Center (NOC)
มหาวิทยาลัยนเรศวร Naresuan University 2008 by helpdesk@nu.ac.th

Status: Done Local intranet

รูปที่ 4.15 หน้าลงทะเบียนสมาชิก

(Webboard.aspx)

หน้ารวมกระทู้ต่าง เพื่อแลกเปลี่ยนความรู้และตอบปัญหาของผู้ใช้งาน

Help-desk

Home UserGuide Service Statistic Webboard History

Login by: Helpdesk

กระทู้	หัวข้อกระทู้	ผู้โพสต์	โพสเมื่อ	จำนวนผู้อ่าน
0002	แจ้ง CD ฝากไว้ทางโลก (ฟรี) ในเน็ต	Admin	17/05/2552 22:32:15	3
0001	ขอความรู้	watchapol	15/02/2552 14:27:15	6

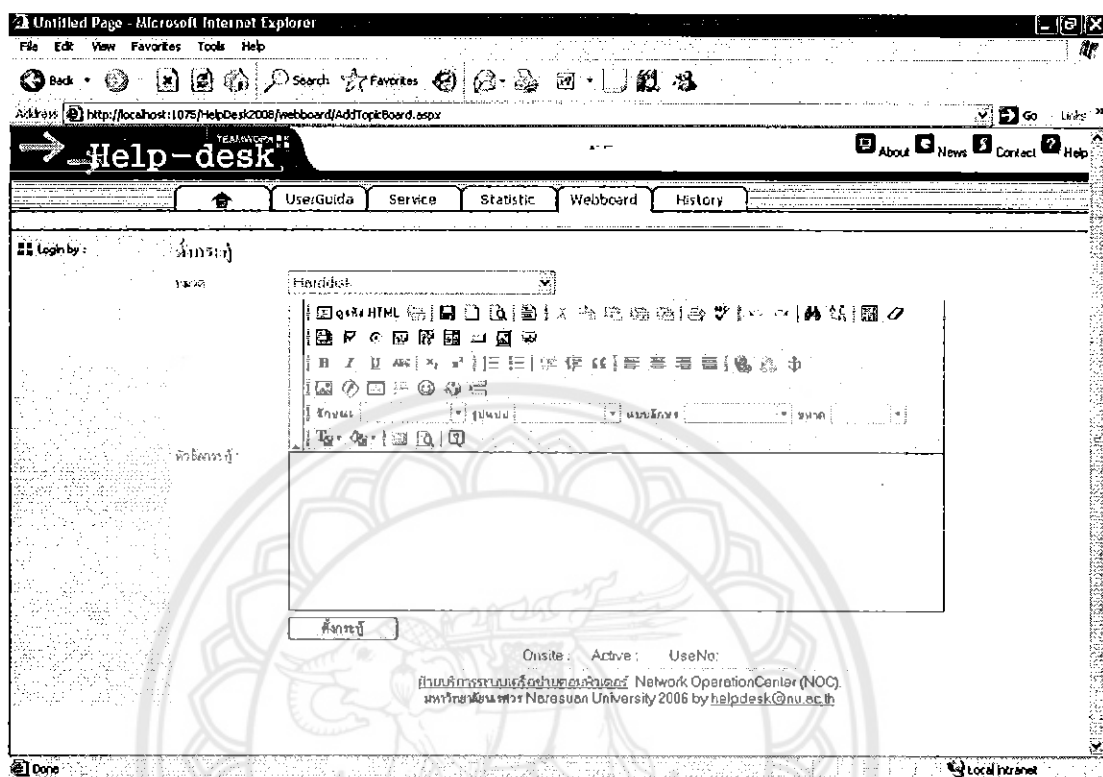
Onsite Active UseNo:

ศูนย์บริการและพัฒนาคอมพิวเตอร์ Network Operation Center (NOC)
มหาวิทยาลัยนเรศวร Naresuan University 2006 by helpdesk@nu.ac.th

รูปที่ 4.16 หน้ากระดานสนทนา

(Addtopicboard.aspx)

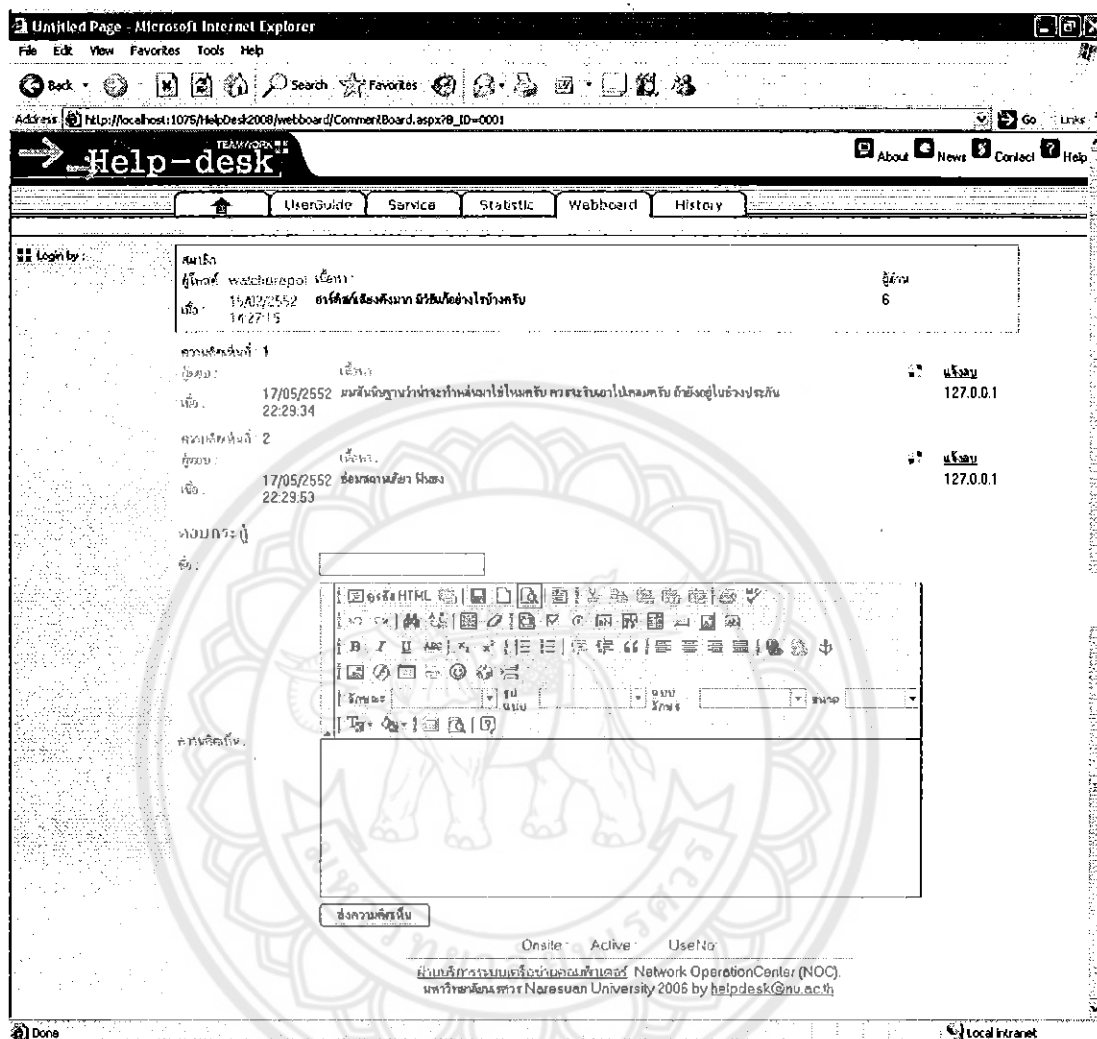
หน้าตั้งกระทู้บนกระดานสนทนา



รูปที่ 4.17 หน้าตั้งหัวข้อกระทู้

(Commentboard.aspx)

หน้าแสดงความคิดเห็นและตอบปัญหาบนกระดานสนทนา



รูปที่ 4.18 หน้าแสดงความคิดเห็น

(Adminboard.aspx)

หน้าบริหารจัดการข้อมูลของกระดานสนทนา

Unfiled Page - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites

Address http://localhost:1075/helpdesk2008/adminboard.aspx

Go

Help-desk

About News Contact Help

UserGuide Service Statistic Webboard History

Login by: Anonymous

Team: Administration Team

Status: Admin

บริหารจัดการข้อมูลเว็บบอร์ด

ค้นหา ดูทั้งหมด

เพิ่มรายการใหม่

ID	ชื่อของกระดาน	หมวด	จำนวน	วันที่	สถานะ
0002	Antrui	CD-Rom	ส่วน CD กลางโศภน เสาไฟแดง 55/1 โหล	17/05/2552 22:32:15	3
0001	watcharapol	Harddisk	ฮาร์ดดิสก์	15/02/2552 14:27:15	6

รายการทั้งหมดในหน้า 2 รายการ

ID	ชื่อสมาชิก	วันที่	IP Address	สถานะ
2	ชื่อสมาชิก	17/05/2552 22:29:53	127.0.0.1	
1	ชื่อสมาชิก	17/05/2552 22:29:34	127.0.0.1	

Online: 5 Active: 1 UseNo: 5

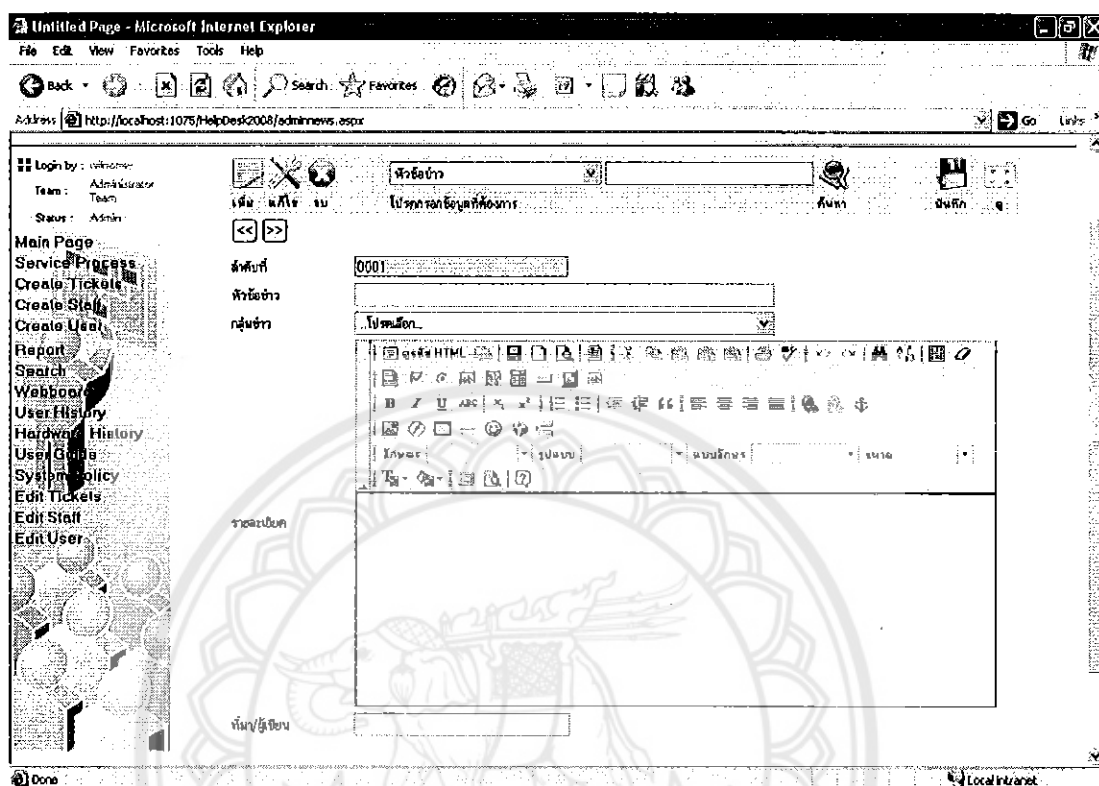
พัฒนาระบบเว็บไซต์และเว็บไซต์ Network Operation Center (NOC)
มหาวิทยาลัยนเรศวร Naresuan University 2006 by helpdesk@nu.ac.th

Done Local Internet

รูปที่ 4.19 บริหารกระดานสนทนา

(Adminnews.aspx)

หน้าบริหารจัดการข้อมูลข่าว



รูปที่ 4.20 บริหารข้อมูลข่าว

(Manageclass.aspx)

หน้าบริหารข้อมูลประเภทปัญหา

Unlited Page - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Search Favorites

Address http://localhost:1075/helpDesk2008/manageclass.aspx

Help-desk

About News Contact Help

UserGuide Service Statistic Webboard History

Login by: welcome
Team: Administrator
Status: Admin

Main Page
Service Process
Create Ticket
Create Staff
Create User
Report
Search
Webboard
User History
Hardware History
User Guide
System Policy
Edit Tickets
Edit Staff
Edit User

รหัส	ประเภท
13	การฟื้นฟูและสำรองข้อมูล (Implement)
12	การวางแผน (Planning)
11	การป้องกัน (Securing)
10	การตั้งค่า (Configuring)
9	การติดตั้ง (Setting)
8	การกำหนดค่า (Scheduling)
7	การป้องกัน (Safeguarding)
6	การตรวจสอบสถานะ (Monitoring)
5	การจัดการ (Managing)
4	การแสดงผล (Deploy)
3	การสร้าง (Creating)
2	การตรวจสอบ (Checking)
1	การฟื้นฟูและสำรองข้อมูล (Backing up and restoring)

Onsite: 5 Active: 1 UseNo: 5

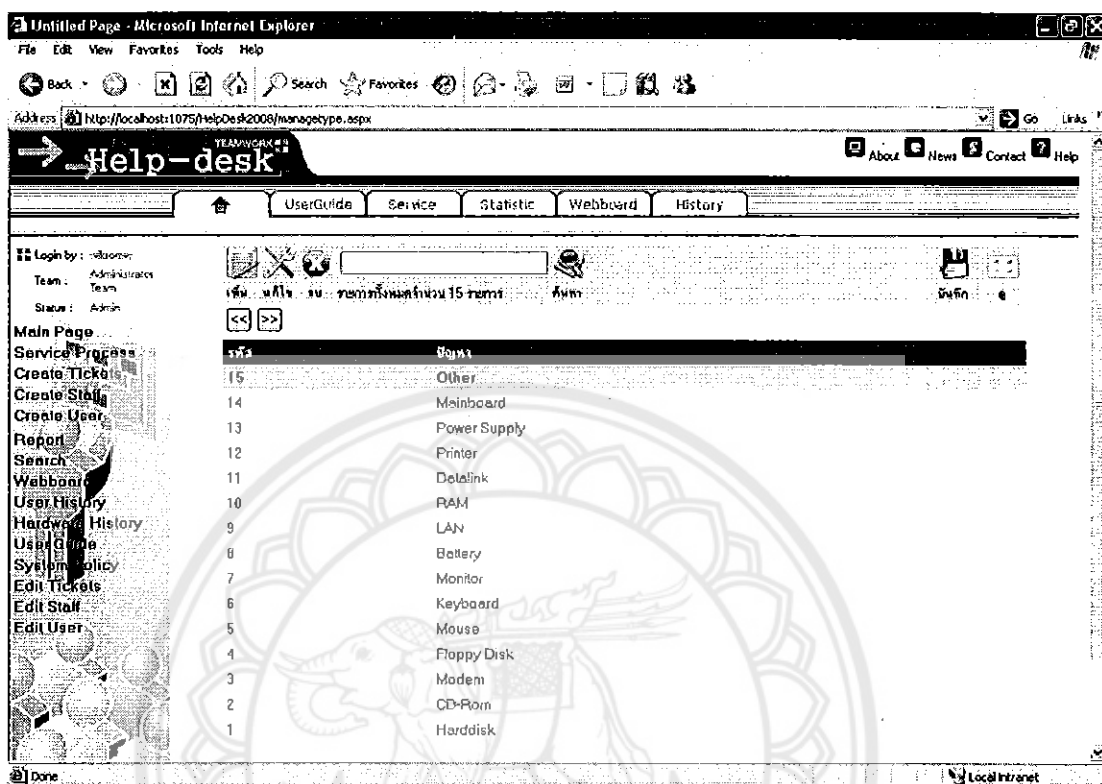
ผ่านบริการระบบเครือข่ายคอมพิวเตอร์ Network OperationCenter (NOC)

Done Local Internet

รูปที่ 4.21 บริหารข้อมูลประเภทปัญหา

(Managetype.aspx)

หน้าบริหารข้อมูลประเภทอุปกรณ์



รูปที่ 4.22 บริหารข้อมูลอุปกรณ์

(Managehelp)

หน้าบริหารจัดการข้อมูลการซ่อมบำรุง

Untitled Page - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Search Favorites

Address http://localhost:1075/helpDesk2008/managehelp.aspx

Go Link

Help-desk

About News Contact Help

UserGuide Service Statistic Webboard History

Login by: welcome
Team: Administrator
Status: Admin

Main Page
Service Process
Create Tickets
Create Staff
Create User
Report
Search
Webboard
User History
Hardware History
User Guide
System Policy
Edit Tickets
Edit Staff
Edit User

Node รายการ
Node รายการ
Node รายการ

Node รายการ

Node	รายการ	สถานะ	ชนิด	ประเภท	ผู้รับใช้	เมื่อ
0021	สถานะของเครื่องคอมพิวเตอร์	พบข้อผิดพลาด	การตรวจสอบ (Checking)	Monitor	admin	18/05/2552 9:48:46
0020	สถานะของเครื่องคอมพิวเตอร์	พบข้อผิดพลาด	การตรวจสอบ (Checking)	Monitor	admin	18/05/2552 9:46:38
0019	สถานะของเครื่องคอมพิวเตอร์	พบข้อผิดพลาด	การตรวจสอบ (Checking)	Monitor	admin	18/05/2552 9:45:05
0018	สถานะของเครื่องคอมพิวเตอร์	พบข้อผิดพลาด	การตรวจสอบ (Checking)	Monitor	admin	18/05/2552 9:42:15
0017	สถานะของเครื่องคอมพิวเตอร์	พบข้อผิดพลาด	การตรวจสอบ (Checking)	Monitor	admin	18/05/2552 9:16:07
0016	สถานะของเครื่องคอมพิวเตอร์	พบข้อผิดพลาด	การตรวจสอบ (Checking)	Monitor	admin	18/05/2552 9:13:50
0015	สถานะของเครื่องคอมพิวเตอร์	พบข้อผิดพลาด	การตรวจสอบ (Checking)	Monitor	admin	18/05/2552 9:11:34
0014	สถานะของเครื่องคอมพิวเตอร์	พบข้อผิดพลาด	การตรวจสอบ (Checking)	Monitor	admin	18/05/2552 9:11:34

Done

รูปที่ 4.23 บริหารข้อมูลการซ่อมบำรุง

(Staffmanage.aspx)

หน้าบริหารจัดการข้อมูล และจัดการสิทธิ์การเข้าถึงข้อมูล

Microsoft Internet Explorer - Untitled Page

Address: http://localhost:1075/helpdesk2008/staffmanage.aspx

Help-desk

Navigation: UserGuide Service Statistic Webboard History

Left Menu:

- Main Page
- Service Process
- Create Tickets
- Create Staff
- Create User
- Report
- Search
- Webboard
- User History
- Hardware History
- User Guide
- System Policy
- Edit Tickets
- Edit Staff
- Edit User

Username	ชื่อ	นามสกุล	ตำแหน่ง	ทีมงาน	งานประจำ	อีเมล	โทรศัพท์
Sorunoo	ศรุต	อดุลย์ชัย	Consult Team	งานบริการระบบเครือข่ายคอมพิวเตอร์		sorunoo@nu.ac.th	815338700
sorunuda	ศรุต	อติพร	Engineer Team	งานบริการระบบเครือข่ายคอมพิวเตอร์		sorunuda@nu.ac.th	869348797
wikromw	วิกรม	วิเศษชัย	Administrator Team	NOC		wikromw@nu.ac.th	1531

Onsite: 5 Active: 1 UseNo: 5

© 2006 by helpdesk@nu.ac.th

รูปที่ 4.24 บริหารข้อมูลพนักงาน

(Teammanage.aspx)

หน้าบริหารจัดการข้อมูลทีมงาน

Help-desk

Teammanage.aspx

Login by: natiw
Team: Administration
Status: Admin

Main Page
Service Process
Create Tickets
Create Staff
Create User
Report
Search
Webboard
User History
Hardware History
User Group
System Policy
Edit Tickets
Edit Staff
Edit User

ลำดับที่	ทีม	รายละเอียด
7	Website Team	ผู้ดูแลเว็บไซต์
6	Administrator Team	ผู้ดูแลระบบ
5	Officer Team	ผู้บริหาร ผู้ตรวจประเมิน
4	Consult Team	ประสานงาน ศึกษาค้นคว้า
3	Engineer Team	ควบคุมดูแล ระบบ หรืออุปกรณ์ระบบเครือข่าย ตลอดจนการดูแลงานเทคนิคต่าง ๆ
2	Maintenance Team	การบำรุง บำรุง บำรุง อุปกรณ์คอมพิวเตอร์ และอุปกรณ์ที่เกี่ยวข้อง
1	Help Desk Team	การบริการ การติดต่อ ข่าสาร ตลอดจนศึกษาปัญหาการร้องเรียนเพื่อปรับปรุงการทำงาน หรือการประสานงานกับผู้ใช้บริการและลูกค้าทั่วไป

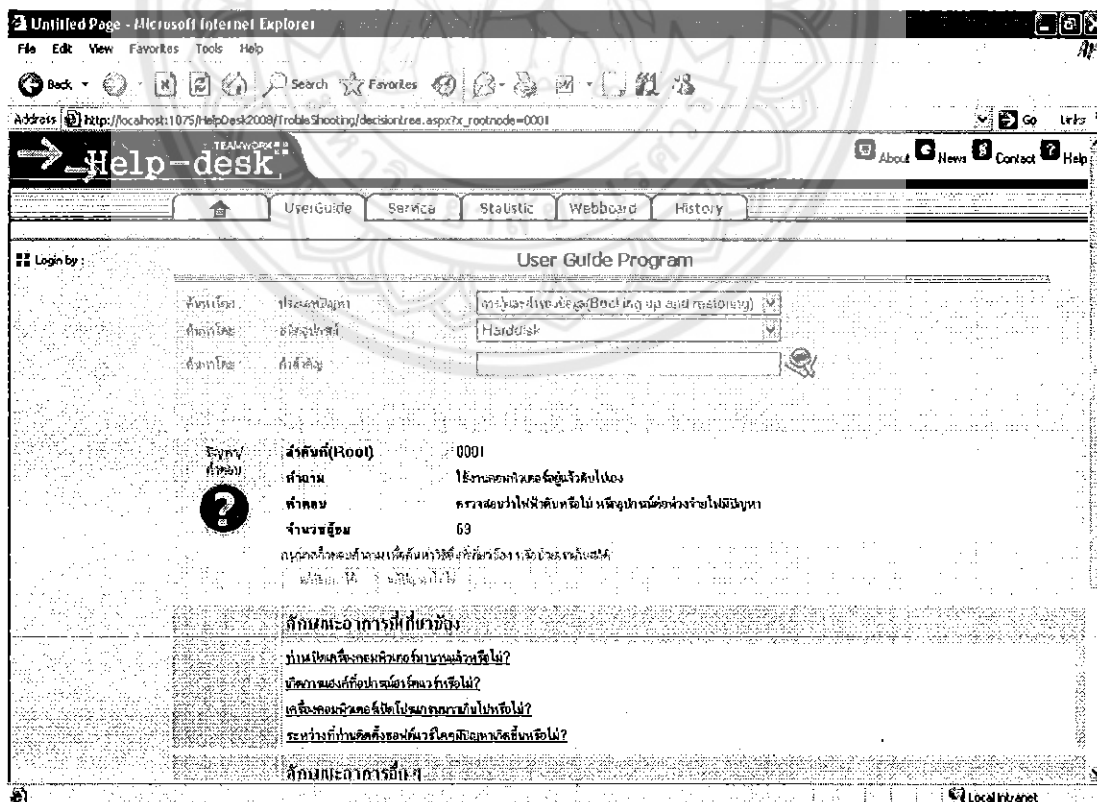
OnSite: 5 Active UseNo: 5

ฝ่ายบริการระบบเครือข่ายคอมพิวเตอร์ Network Operation Center (NOC)
มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี 2006 by helpdesk@kmutt.ac.th

รูปที่ 4.25 บริหารข้อมูลทีมงาน

(Decisiontree.aspx)

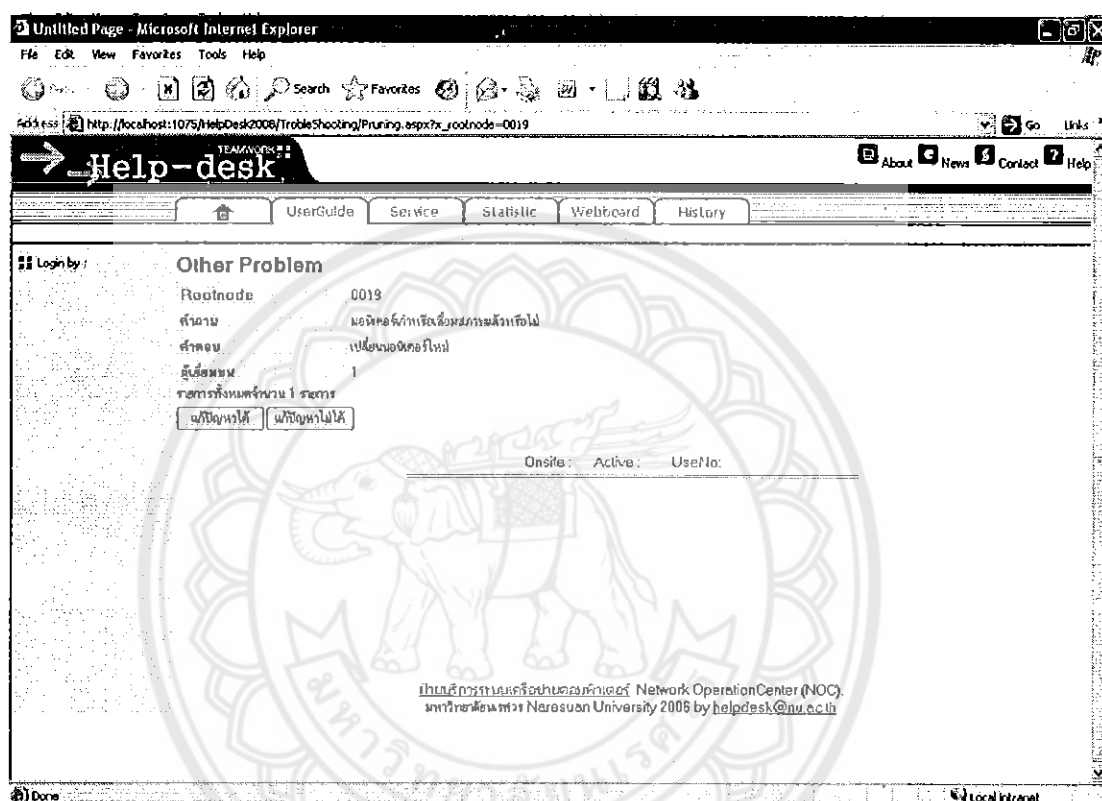
หน้าช่วยเหลือสืบค้นข้อมูล สำหรับผู้ใช้บริการที่ต้องการคำปรึกษาแนะนำในการแก้ไขปัญหา สามารถค้นหาโดยเลือกคนหาจากประเภทของปัญหา ประเภทของฮาร์ดแวร์ และค้นหาโดยคำค้น เมื่อค้นหาพบ โปรแกรมจะแสดง คำตอบที่ผู้ใช้เลือกจากรายการคำถาม โดยจะมีตัววัดผลการทำงานในการแก้ไขปัญหาว่าได้ผลหรือไม่ จากการคลิกที่ปุ่ม แก้ปัญหาได้ และแก้ปัญหาไม่ได้ หากคำตอบดังกล่าวไม่สามารถแก้ปัญหาได้ โปรแกรมจะช่วยเหลือผู้ใช้โดยแสดงรายการคำถามที่เกี่ยวข้องทั้งหมด ซึ่งแบ่งออกเป็นสองส่วนคือ คำถามที่เกี่ยวข้องกับปัญหา และคำถามทั้งหมด โดยคำถามที่เกี่ยวข้องกับปัญหาจะจัดอยู่ในกลุ่มของต้นไม้ปัญหาที่อยู่บนคันเดียวกัน เรียงลำดับจากค่า Information Gain จากมากไปน้อย ส่วนคำถามอื่นๆที่ไม่ได้ออยู่บนต้นไม้ใดเลย จะแสดงตามประเภทปัญหา หรือ ประเภทของฮาร์ดแวร์โดยเรียงลำดับจากสถิติการเข้าชมมากที่สุด การช่วยเหลือในการค้นหาปัญหาจะกระทำกระบวนกรซ้ำเดิมไปเรื่อยๆจนกว่า จะแก้ปัญหาของผู้ใช้โปรแกรมได้ นั่นคือสรุปปลายทางของ Tree นั้นเอง หากยังไม่พบคำตอบที่ต้องการ โปรแกรมจะทำการเรียกคำถามจากฐานข้อมูลของ Tree อื่นมาแสดง อนึ่ง กระบวนการวัดผลคำตอบของปัญหา จะเป็นตัวแปรสำคัญที่จะนำไปประมวลผลหาค่า Information Gain ซึ่งมีผลต่อความถูกต้องของการแก้ไขปัญหา และลดระยะเวลาในการเข้าถึงข้อมูลของผู้ใช้โปรแกรมอื่นๆต่อไป



รูปที่ 4.26 โปรแกรมสืบค้นข้อมูล

(Pruning.aspx)

หน้า Pruning เป็นการแสดงการค้นหาปัญหาที่อยู่นอกต้นไม้ปัจจุบัน หรือ ปัญหาที่ไม่ได้จัดอยู่ในต้นไม้ใดๆ เลย โดยเรียงลำดับตามสถิติการเข้าชม ซึ่งเมื่อคำตอบของปัญหาสามารถแก้ไข ปัญหาได้ โปรแกรมจะจัดปัญหาดังกล่าว เข้ามารวมอยู่ในกลุ่มของต้นไม้เดียวกัน (Post Pruning) แต่ถ้าไม่สามารถแก้ปัญหาก็ โปรแกรมจะค้นหาต่อไป



รูปที่ 4.27 แสดงการสืบค้นปัญหา กรณีที่ปัญหาอยู่ Tree อื่น

บทที่ 5

บทสรุป

5.1 สรุปผล

การศึกษาค้นคว้าเรื่องการพัฒนาระบบ IT Helpdesk สำหรับงานบริการผู้ใช้งานระบบเครือข่ายคอมพิวเตอร์ สรุปผลดังนี้

ผู้ศึกษาได้นำข้อมูลจากการเก็บผลสำรวจของระบบ มาทำการวิเคราะห์และออกแบบระบบตามแนวคิดแบบ Object-Oriented Analysis and Design (OOAD) โดยใช้ร่วมกับการพัฒนาฐานข้อมูล โดยได้ออกแบบระบบให้มีส่วนประกอบหลัก ดังนี้

1. การแนะนำและแก้ไขปัญหาการใช้งาน (Self Help)
2. การแจ้งขอรับบริการ (Services Request)
3. การประสานงานของเจ้าหน้าที่และจัดลำดับการให้บริการ (Services Queuing)
4. การรับข้อเสนอแนะ และการร้องเรียน (Web Board)
5. การแสดงสถิติการให้บริการ

กลุ่มผู้ใช้งานระบบ แบ่งเป็น 3 กลุ่ม คือ ผู้ใช้งานทั่วไป (Users), เจ้าหน้าที่ผู้ให้บริการระบบเครือข่าย (Net Staff) และ ผู้จัดการระบบ Helpdesk (HDSA) โดยแต่ละกลุ่มจะมีสิทธิในการเข้าถึงระบบทั้ง 6 ส่วนได้แตกต่างกัน จำแนกโดยระบบล็อกอินซึ่งใช้ในการยืนยันตัวตนของผู้ใช้

การพัฒนาระบบใช้โปรแกรม Visual Studio .NET (ภาษา C) ในส่วนระบบฐานข้อมูลใช้โปรแกรม Microsoft SQL Server 2005 ในการจัดทำ โดยประยุกต์ใช้แนวคิดเกี่ยวกับ Object-Relational Mapping ซึ่งได้ทำการพัฒนาระบบให้มีส่วนประกอบระบบ 4 ส่วน ดังนี้

1. ระบบการช่วยแก้ไขปัญหาการใช้งานด้วยตนเอง (Self Help)
2. ระบบการแจ้งขอรับบริการ (Services Request)
3. ระบบการประสานงานของเจ้าหน้าที่และการจัดลำดับงาน (Services Queuing)
4. ระบบการบันทึกและรายงานสถิติการให้บริการและการรายงานผล (Services Report)

5.2 ข้อเสนอแนะ

การศึกษาโครงการนี้ เป็นการพัฒนาระบบเพียงบางส่วนของระบบ Helpdesk ทั้งหมด โดยยังไม่รวมถึงระบบฐานข้อมูลฮาร์ดแวร์ต่างๆ และคอมพิวเตอร์ เนื่องจากมีฮาร์ดแวร์จำนวนมาก และการสืบค้นจำนวนและประวัติต่างๆ ยังไม่สามารถหาได้ครบถ้วน แต่ละหน่วยงานยังไม่ได้เก็บข้อมูลในส่วนนี้ได้เพียงพอ

นอกจากนี้ระบบ Helpdesk ควรมีระบบการเตรียมพร้อมสำหรับการแก้ปัญหาที่เกิดขึ้นในระดับต่างๆ เช่น เหตุฉุกเฉินหรือกรณี Service Down เป็นต้น รวมทั้ง ควรมีความสามารถในการประมาณการเวลาในการให้บริการซ่อมบำรุง และสามารถจัดสรรมอบหมายงานให้แก่เจ้าหน้าที่ได้โดยอัตโนมัติ



เอกสารอ้างอิง

- [1] Timothy C., Lethbridge and Lobert Laganière.. **Object-Oriented Software Engineering**. Singapore : McGraw-Hill Company,Inc. 2002.
- [2] Simon Bennett, John Skelton and Ken lunn. **Schaum's Outline of UML**. Singapore : McGraw-Hill Company,Inc. 2005.
- [3] Manfred Knobloch and Matthias Kopp. **Web Design with XML**. Great Britain : Biddies Ltd. 2003.
- [4] Dev.co.th., LTD"ASP.NET". [Online]. Available: <http://www.dev.co.th/newdev/more.asp>. 1995.
- [5] SiamDev.co.th., LTD".NET Framework" [Online]. Available: <http://www.siamdev.com/>. 2003.
- [6] Scott Klein. **Professional SQL Server™ 2005 XML**. Indianapolis : Wiley Publishing,Inc. 2006.
- [7] David Hand, Heikki Mannila and Padhric Smyth. **Principles of Data Mining**. London : Bradford Book. 2001.
- [8] สุภชัย สมพานิช. สร้างระบบงานฐานข้อมูลด้วย Visual Basic.NET ฉบับโปรแกรมเมอร์. นนทบุรี : ไอดีซี. 2546.

ประวัติผู้เขียนโครงการ

ชื่อ นาย วัชรพล เพชรต๋น
ภูมิลำเนา 25/5 หมู่ 4 ต.ธานี อ.เมือง จ.สุโขทัย 64000
ประวัติการศึกษา

- จบระดับมัธยมศึกษาที่โรงเรียนสรรพวิทยาคม
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 4
สาขาวิชาคอมพิวเตอร์ ภาควิชาไฟฟ้าแล้วคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail : hall_13@hotmail.com

ชื่อ นาย วิกรม วิเศษพลกรัง
ภูมิลำเนา 91 หมู่ 6 ต.คลองกระจง อ.สวรรคโลก จ.สุโขทัย 64110
ประวัติการศึกษา

- จบระดับมัธยมศึกษาที่โรงเรียนสวรรคค่อนันต์วิทยาคม
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 4
สาขาวิชาคอมพิวเตอร์ ภาควิชาไฟฟ้าแล้วคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail : wlkrom_6@hotmail.com