

โปรแกรมการจัดการระบบเอกสารภายในองค์กรผ่านเครือข่าย

A network based system for Documents organization and management

นายธีรวิษณุ พิรพันธุ์

รหัสประจำตัว 43360429

นายเทพรัตน์ มุลกัณธา

รหัสประจำตัว 43360742

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ..... 1 / โศ.ย. 2553
เลขทะเบียน..... 1 4941994 ๘๒
เลขเรียกหนังสือ..... ๗๖.
มหาวิทยาลัยนเรศวร ๕๖๔๘ ๗

2547

ปริญญาานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2547



ใบรับรองโครงการวิศวกรรม

หัวข้อโครงการ	โปรแกรมการจัดการระบบเอกสารภายในองค์กรผ่านเครือข่าย		
ผู้ดำเนินโครงการ	นายธีรวิชญ์ พิศพันธ์ุ์	รหัสประจำตัว	43360429
	นายเพชรรัตน์ มุลกัณษา	รหัสประจำตัว	43360742
อาจารย์ที่ปรึกษา	อาจารย์ภาณุพงศ์ สอนคม		
อาจารย์ที่ปรึกษาร่วม	อาจารย์ไพศาล มุณีสว่าง		
สาขาวิชา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2547		

คณะวิศวกรรมศาสตร์มหาวิทยาลัยขอนแก่น อนุมัติให้โครงการฉบับนี้เป็นส่วนหนึ่งของการศึกษา
ตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์
คณะกรรมการสอบ โครงการวิศวกรรม

.....ประธานกรรมการ
(อาจารย์ภาณุพงศ์ สอนคม)

.....กรรมการ
(อาจารย์ไพศาล มุณีสว่าง)

.....กรรมการ
(อาจารย์แสงชัย มังกรทอง)

หัวข้อโครงการ	โปรแกรมการจัดการระบบเอกสารภายในองค์กรผ่านเครือข่าย	
ผู้ดำเนินโครงการ	นายธีรวิทย์ พิพันธ์ุ	รหัสประจำตัว 43360429
	นายเทพรัตน์ มูลกัณธา	รหัสประจำตัว 43360742
อาจารย์ที่ปรึกษา	อาจารย์ภาณุพงศ์ สอนคม	
อาจารย์ที่ปรึกษาร่วม	อาจารย์ไพศาล มุณีสว่าง	
สาขาวิชา	วิศวกรรมคอมพิวเตอร์	
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์	
ปีการศึกษา	2547	

บทคัดย่อ

โครงการนี้จัดทำขึ้นเพื่อพัฒนาโปรแกรมการจัดการระบบเอกสารภายในองค์กรผ่านเครือข่าย เพื่อให้การจัดการระบบเอกสารต่างๆภายในองค์กรมีประสิทธิภาพมากขึ้น สามารถนำเอกสารต่างๆออกมาใช้งานได้อย่างสะดวกรวดเร็ว และเป็นการประหยัดกระดาษที่ใช้ในการเก็บเอกสารซึ่งต้องใช้เป็นจำนวนมาก และเนื่องจากการที่เก็บเอกสารภายในระบบนี้จะช่วยเพิ่มความปลอดภัยของข้อมูลต่างๆได้ เพราะระบบมีการแบ่งระดับของผู้ที่จะเข้าถึงเอกสารในแต่ละระดับต่างกันตามหน้าที่ของแต่ละบุคคล นอกจากนี้ยังช่วยให้การส่งงาน การติดต่อสื่อสารกันภายในองค์กรมีความสะดวกและรวดเร็วมากยิ่งขึ้น

จากผลการทดสอบโปรแกรมการจัดการระบบเอกสารภายในองค์กรผ่านเครือข่าย พบว่าโปรแกรมสามารถทำงานได้ตามที่ความต้องการสามารถทำงานได้อย่างมีประสิทธิภาพ

Project Title	A network based system for documents organization and management		
Name	Mr. Teerawit Pheeraphan	ID.	43360429
	Mr. Theparat Moonkanta	ID.	43360742
Project Advisor	Mr. Panupong Sonkhom		
Co - Project Advisor	Mr. Phaisan Muncesawang		
Major	Computer Engineering		
Department	Electrical and Computer Engineering		
Academic Year	2004		

.....

ABSTRACT

This project studies a network based system for effective organization and management of documents. This system provides fast access to documents stored remotely on the network. The proposed system can separate users in different levels according to their job description for security purpose, as well as providing a better communication between workers. Apart from this, by using our system, we can reduce a large amount of papers since documents are stored in electronic form. From our experiment, we found that our system is very effective for organization and management of documents.

กิตติกรรมประกาศ

โครงการฉบับนี้สำเร็จลงได้ด้วยความอนุเคราะห์อันดีจาก อาจารย์ภาณุพงศ์ สอนคม ที่ได้กรุณาเป็นอาจารย์ที่ปรึกษาให้แนวคิดและให้ความช่วยเหลือตลอดจนสละเวลาอันแสนมีค่าเพื่อตรวจแก้ไขข้อบกพร่องต่าง ๆ จนกระทั่งสำเร็จลุล่วงไปได้ด้วยดี นอกจากนี้ยังได้รับความอนุเคราะห์จาก อาจารย์ไพศาล มุณีสว่าง ที่ได้สละเวลาอันแสนมีค่าเพื่อร่วมตรวจแก้ไขข้อบกพร่องต่าง ๆ และให้การช่วยเหลือในอีกหลาย ๆ ด้าน ทางผู้จัดทำขอกราบขอบพระคุณเป็นอย่างสูงไว้ ณ โอกาสนี้

ทั้งนี้ขอกราบขอบพระคุณอาจารย์ทุกท่านที่ได้อบรมสั่งสอนให้ความรู้ คำแนะนำและการช่วยเหลือเสมอมา รวมทั้งขอขอบพระคุณเพื่อน ๆ ที่ ๆ ทุกคนที่ได้ให้การช่วยเหลืออย่างเต็มที่ไม่ว่าจะในด้านการจัดทำโปรแกรมหรือการจัดทำเอกสาร

ท้ายนี้ขอกราบขอบพระคุณ บิดา-มารดา ผู้ที่ทุ่มเททั้งชีวิตและจิตใจเพื่อให้เกิดการศึกษาที่ดีที่สุดและสิ่งต่าง ๆ ที่ดีที่สุดแก่ผู้จัดทำ และคอยดูแลความสำเร็จในแต่ละก้าวของผู้จัดทำนอกจากนี้ยังคอยเป็นห่วงเป็นใยและให้กำลังใจเสมอมา จนสามารถทำงานครั้งนี้จนสำเร็จลุล่วงด้วยดี

นายธีรวิทย์ ฬิรพันธุ์

นายเทพรัตน์ มุลกัณษา

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ก
บทคัดย่อภาษาอังกฤษ.....	ข
กิตติกรรมประกาศ.....	ค
สารบัญ.....	ง
สารบัญรูป.....	ช
บทที่ 1 บทนำ	
1.1 ที่มาและความสำคัญของโครงการ.....	1
1.2 วัตถุประสงค์ของโครงการ.....	1
1.3 ขอบข่ายของโครงการ.....	1
1.4 กิจกรรมดำเนินการ.....	2
1.5 ผลที่คาดว่าจะได้รับ.....	3
1.6 งบประมาณที่ใช้.....	3
บทที่ 2 หลักการและทฤษฎีที่เกี่ยวข้อง	
2.1 ไคลเอนต์ (Client) และ เซิร์ฟเวอร์ (Server).....	4
2.2 โพรโตคอล TCP/IP.....	5
2.2.1 โพรโตคอล TCP (Transmission Control Protocol).....	5
2.2.2 โพรโตคอล IP (Internet Protocol).....	5
2.3 โครงสร้างของโปรโตคอล TCP/IP.....	6
2.3.1 Process layer หรือ Application Layer.....	7
2.3.2 Host-to-Host layer หรือ Transport Layer.....	8
2.4 ความรู้พื้นฐานเกี่ยวกับ Visual Basic.....	10
2.4.1 จุดเด่นของ Visual Basic.....	11
2.4.2 การเขียนโปรแกรมเพื่อสร้างแอปพลิเคชันจาก Visual Basic.....	11
2.4.3 MS Winsock Control 6.....	11

สารบัญ(ต่อ)

หน้า

บทที่ 3 การศึกษาและพัฒนาโปรแกรม

3.1 เครื่องมือที่ใช้ในการพัฒนาโปรแกรม.....	17
3.2 การออกแบบระบบ.....	17
3.2.1 เซิร์ฟเวอร์ (Server).....	17
3.2.2 ไคลเอนต์ (Client).....	17
3.3 ส่วนอธิบายโปรโตคอลของการติดต่อสื่อสาร.....	17
ส่วนที่ 1 ไคลเอนต์ --> เซิร์ฟเวอร์.....	17
ส่วนที่ 2 เซิร์ฟเวอร์ --> ไคลเอนต์.....	18
3.4 โครงสร้างและหลักการทำงานของโปรแกรม.....	19
3.4.1 ล็อกอิน (Login) / ล็อกออฟ (Logoff).....	19
3.4.2 ข้อความ (message).....	21
3.4.3 อัปโหลด (Upload) / ดาวน์โหลด (Download).....	24
3.4.4 ส่งอีเมล (Send e-mail) / สมุดรายชื่อ (Address Book).....	25
3.4.5 ค้นหาไฟล์หรือเอกสาร (Search file).....	27
3.4.6 การสนทนา (Chat).....	28
3.4.7 การจัดการฐานข้อมูลยูสเซอร์ (User Database).....	29
3.5 การออกแบบในส่วนของอินเทอร์เน็ตเฟสที่ติดต่อกับผู้ใช้.....	30
3.5.1 ส่วนของเซิร์ฟเวอร์.....	31
3.5.2 ส่วนของไคลเอนต์.....	31

บทที่ 4 การทดสอบและวิเคราะห์การทำงาน

4.1 ส่วนของเซิร์ฟเวอร์ (Server).....	37
4.2 ส่วนของไคลเอนต์ (Client).....	37
4.2.1 ล็อกอิน (Login).....	38
4.2.2 ข้อความ (message).....	38
4.2.3 อัปโหลด (Upload) / ดาวน์โหลด (Download).....	40
4.2.4 ส่งอีเมล (Send e-mail).....	41

สารบัญ(ต่อ)

	หน้า
4.2.5 สมุดรายชื่อ (Address Book).....	42
4.2.6 ค้นหาไฟล์หรือเอกสาร (Search file).....	43
4.2.7 การสนทนา (Chat).....	44
4.3 ปัญหาที่พบบ่อยขณะทำการทดสอบและวิธีแก้ไข.....	44

บทที่ 5 สรุปผลการดำเนินงานและข้อเสนอแนะ

5.1 สรุปผลการดำเนิน โครงการ.....	47
5.2 ประโยชน์ที่ได้รับจากการทำโครงการ.....	47
5.3 ข้อเสนอแนะ.....	48
5.4 แนวทางในการพัฒนาโปรแกรมในอนาคต.....	48
ภาคผนวก.....	50
เอกสารอ้างอิง.....	56
ประวัติผู้เขียนโครงการ.....	57

สารบัญรูป

รูปที่	หน้า
2.1 TCP/IP Model และ OSI Model.....	6
2.2 Protocol Stack.....	7
2.3 ลำดับชั้นการทำงานของโปรโตคอล.....	7
2.4 Port ที่ใช้ในโปรโตคอลต่างๆ.....	9
2.5 การเรียกใช้โปรโตคอล TCP และ UDP.....	10
2.6 Components MS Winsock Control 6.....	12
2.7 Microsoft Winsock Control 6.0.....	12
2.8 Properties Winsock Dialog.....	13
3.1 แผนภาพแสดงการทำงานของการล็อกอิน.....	20
3.2 แผนภาพแสดงการทำงานของการล็อกออฟ.....	21
3.3 แผนภาพแสดงการทำงานของข้อความ.....	22
3.4 แผนภาพแสดงการทำงานของการส่งข้อความ.....	23
3.5 แผนภาพแสดงการทำงานของอ็อปโหลด / ดาวน์โหลด.....	25
3.6 แผนภาพแสดงการทำงานของการส่งอีเมล.....	26
3.7 แผนภาพแสดงการทำงานของสมุดรายชื่อ.....	27
3.8 แผนภาพแสดงการทำงานของการค้นหาไฟล์.....	28
3.9 แผนภาพแสดงการทำงานของการสนทนา.....	29
3.10 แผนภาพแสดงการทำงานของฐานข้อมูลยูสเซอร์.....	30
3.11 อินเทอร์เน็ตของเซิร์ฟเวอร์.....	31
3.12 อินเทอร์เน็ตของ.....	31
3.13 อินเทอร์เน็ตของข้อความ.....	32
3.14 อินเทอร์เน็ตของข้อความ.....	32
3.15 อินเทอร์เน็ตของอ็อปโหลด / ดาวน์โหลด.....	33
3.16 อินเทอร์เน็ตของการส่ง.....	33
3.17 อินเทอร์เน็ตของสมุดรายชื่อ.....	34
3.18 อินเทอร์เน็ตของการเพิ่มรายชื่อ.....	34
3.19 อินเทอร์เน็ตของการแก้ไขรายชื่อ.....	35
3.20 อินเทอร์เน็ตของการสนทนา.....	35

สารบัญรูป(ต่อ)

รูปที่	หน้า
3.21 อินเตอร์เฟซการจัดการฐานข้อมูลยูสเซอร์ (เพิ่มยูสเซอร์ใหม่).....	36
3.22 อินเตอร์เฟซการจัดการฐานข้อมูลยูสเซอร์ (แก้ไขข้อมูลยูสเซอร์).....	36
4.1 ภาพแสดงเซิร์ฟเวอร์.....	37
4.2 ภาพแสดงการล็อกอิน.....	38
4.3 ภาพแสดงข้อความ.....	38
4.4 ภาพแสดงข้อความใหม่.....	39
4.5 ภาพแสดงการส่งข้อความ.....	39
4.6 ภาพแสดงการดาวน์โหลด.....	40
4.7 ภาพแสดงการอัปโหลด.....	40
4.8 ภาพแสดงการอัปโหลด(2).....	41
4.9 ภาพแสดงการส่งอีเมล.....	41
4.10 ภาพแสดงสมุดรายชื่อ.....	42
4.11 ภาพแสดงการเพิ่มรายชื่อลงในสมุดรายชื่อ.....	42
4.12 ภาพแสดงการแก้ไขรายชื่อในสมุดรายชื่อ.....	43
4.13 ภาพแสดงการค้นหาไฟล์.....	43
4.14 ภาพแสดงการสนทนา.....	44
4.15 Error connection.....	45
4.16 Not found ServerIP.ini.....	45
4.17 Path not found.....	46
6.1 การติดตั้งเซิร์ฟเวอร์.....	50
6.2 การติดตั้งเซิร์ฟเวอร์(2).....	51
6.3 การติดตั้งเซิร์ฟเวอร์(3).....	51
6.4 การติดตั้งเซิร์ฟเวอร์(4).....	52
6.5 การติดตั้งไคลเอนต์(1).....	53
6.6 การติดตั้งไคลเอนต์(2).....	53
6.7 การติดตั้งไคลเอนต์(3).....	54
6.8 การติดตั้งไคลเอนต์(4).....	54

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของโครงการ

เนื่องจากองค์กรต้องมีข้อมูลข่าวสารจำนวนมาก เพื่อให้แข่งขันในตลาดการค้าที่รุนแรงในปัจจุบันได้อย่างมีประสิทธิภาพ องค์กรที่จะประสบความสำเร็จ ต้องมีระบบข่าวสารข้อมูลที่ทันสมัย รวดเร็ว และสามารถเชื่อมโยงข้อมูลได้อย่างมีประสิทธิภาพ นอกจากนี้เอกสารจำนวนมากที่มีอยู่ภายในองค์กรนั้น ต้องใช้เวลามากในการค้นหาข้อมูลบนเอกสารในส่วนที่ต้องการ ในกรณีที่เอกสารถูกเก็บไว้นานอาจเกิดการชำรุดหรือสูญหาย ซึ่งในปัจจุบันปัญหาที่เกิดขึ้นเหล่านี้ น่าจะลดน้อยลงเนื่องจากความทันสมัยของเทคโนโลยีจึงควรนำเทคโนโลยีเหล่านี้มาใช้ให้เกิดประโยชน์ โดยแทนเอกสารที่อยู่ในรูปของกระดาษด้วยเอกสารที่อยู่ในรูปของ แฟ้มข้อมูลที่ถูกรักษาไว้ในเครื่องคอมพิวเตอร์เพื่อลดปริมาณของการใช้กระดาษในการเก็บเอกสาร ซึ่งแฟ้มข้อมูลจะถูกจัดเก็บด้วยโปรแกรมที่มีระบบรักษาความปลอดภัยที่สามารถป้องกันข้อมูลที่เป็นความลับ หรือการแก้ไข คัดแปลงข้อมูลของบุคคลที่มีได้รับอนุญาต และสามารถแบ่งระดับของการเข้าถึงข้อมูลของแต่ละบุคคลได้ ซึ่งจะช่วยให้การทำงานเป็นไปอย่างรวดเร็วและมีประสิทธิภาพมากขึ้น

1.2 วัตถุประสงค์ของโครงการ

1. เพื่อให้การสื่อสารภายในองค์กรมีความสะดวกรวดเร็วมากยิ่งขึ้น
2. เพื่อรักษาความลับทางเอกสารภายในองค์กร
3. เพื่อลดปริมาณการใช้กระดาษในการจัดเก็บเอกสาร ซึ่งจะช่วยให้ประหยัดค่าใช้จ่ายภายในองค์กรและเป็นการประหยัดทรัพยากรธรรมชาติ

1.3 ขอบข่ายของโครงการ

1. สามารถส่งเอกสารไปยังเครื่องลูกข่าย โดยที่สามารถตรวจสอบได้ว่าสถานะของเอกสารที่ถูกส่งไปยังเครื่องลูกข่ายนั้น ได้ถูกอ่านหรือเปิดดูแล้วหรือยัง โดยที่เครื่องลูกข่ายนั้นจะมีการส่งข้อมูลที่บอกสถานะของเอกสารกลับมายังเครื่องแม่ข่าย ทำให้สามารถทราบสถานะของเอกสารได้
2. เป็นโปรแกรมที่มีรูปแบบในการสื่อสารระหว่างกันเป็นของตนเอง โดยผ่านทางระบบเครือข่ายภายในองค์กรเพื่อเพิ่มความปลอดภัยในระบบเอกสาร
3. สามารถเลือกการส่งเอกสารผ่านทางอินเทอร์เน็ต ไปยังอีเมลล์ของบุคคลที่ต้องการได้ ในกรณีที่ผู้รับจำเป็นที่จะต้องรับเอกสารนั้นขณะที่อยู่ภายนอกองค์กร

4. โปรแกรมสามารถส่งบทสนทนาโต้ตอบกันได้ เพื่อความสะดวกในการสั่งงานภายในองค์กร และสามารถบันทึกบทสนทนาได้
5. โปรแกรมสามารถแบ่งระดับของการเข้าถึงเอกสารในระดับต่างๆภายในองค์กรโดยการใช้พาสเวิร์ดโดยที่พาสเวิร์ดของแต่ละบุคคลจะมีความสามารถในการเข้าถึงข้อมูลในระดับที่ต่างกันขึ้นอยู่กับหน้าที่และความจำเป็น
6. โปรแกรมสามารถค้นหาข้อมูลของเอกสาร ที่ถูกจัดเก็บโดยแบ่งตามประเภทของเอกสารไว้อย่างเป็นระบบ

1.4 กิจกรรมดำเนินการ

กิจกรรม	ปี 2546						ปี 2547					
	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.
เริ่มศึกษาและวางแผนการทำงาน	←		→									
ศึกษาระบบการจัดเก็บเอกสาร	←		→									
ศึกษาทฤษฎีที่เกี่ยวข้อง	←		→									
ศึกษาระบบเน็ตเวิร์ค			←		→							
ศึกษาแอปพลิเคชันที่เกี่ยวข้อง			←		→							
สร้างโปรแกรมจากระบบ			←		→							
ทดลอง ตรวจสอบแก้ไขโปรแกรมจากปัญหา							←		→			
จัดทำเอกสาร			←		→							
ส่งโครงการฉบับสมบูรณ์										←		→

1.5 ผลที่คาดว่าจะได้รับ

1. ได้โปรแกรมที่มีความสามารถในการส่งเอกสารไปยังเครื่องลูกข่ายโดยสามารถตรวจสอบสถานะของเอกสารที่ถูกส่งไปได้ หรือสามารถเลือกการส่งเอกสารผ่านทางอินเทอร์เน็ตไปยังอีเมลล์ของบุคคลที่ต้องการ, ส่งบทสนทนาโต้ตอบกันและสามารถบันทึกบทสนทนาได้, สามารถค้นหาข้อมูลของเอกสาร ที่ถูกจัดเก็บโดยแบ่งตามประเภทของเอกสาร, และโปรแกรมยังสามารถแบ่งระดับของการเข้าถึงเอกสารในระดับต่างๆภายในองค์กรได้
2. การสื่อสารภายในองค์กรมีความสะดวกรวดเร็วมากยิ่งขึ้น
3. ปริมาณการใช้กระดาษลดน้อยลง ประหยัดค่าใช้จ่ายภายในองค์กรและเป็นการประหยัดทรัพยากรธรรมชาติ

1.6 งบประมาณที่ใช้

1. ค่าหนังสือที่ใช้ประกอบการค้นคว้า	1500	บาท
2. ค่าจัดทำเอกสาร	500	บาท
3. ค่าอุปกรณ์ในระบบ Network	1000	บาท
รวมเป็นเงินทั้งสิ้น	<u>3000</u>	บาท

* ทุกรายการสามารถถัวเฉลี่ย



บทที่ 2

หลักการและทฤษฎีที่เกี่ยวข้อง

ในปัจจุบันการสื่อสารผ่านระบบเครือข่าย (Network) รวมทั้งระบบอินเทอร์เน็ต เป็นที่แพร่หลายมากในแง่ของการใช้งาน, ใช้บริการ, อำนวยความสะดวกในชีวิตประจำวันต่างๆ เนื่องจากระบบสื่อสารที่รวดเร็วและสามารถเชื่อมโยงเข้ากับหลาย ๆ ระบบได้ ทำให้สามารถนำมาประยุกต์ใช้งานผ่านระบบเครือข่ายได้หลากหลายรูปแบบ ซึ่งจะนำไปประยุกต์ใช้ในการควบคุมอุปกรณ์หรือโปรแกรมต่างๆ

2.1 ไคลเอนต์ (Client) และ เซิร์ฟเวอร์ (Server)

ไคลเอนต์จะเป็นเครื่องคอมพิวเตอร์ที่ร้องขอบริการหรือข้อมูลจากเครื่องเซิร์ฟเวอร์ โดยผ่านทางโปรแกรมต่างๆ เช่น เว็บเบราว์เซอร์, เอาท์ลุคเอ็กเพรส, เอ็มไออาร์ซี ซึ่งเครื่องเซิร์ฟเวอร์ จะเป็นเครื่องที่ให้บริการต่างๆแก่เครื่องที่เป็นไคลเอนต์ เช่น เว็บเซิร์ฟเวอร์, เมล์เซิร์ฟเวอร์, ไออาร์ซีเซิร์ฟเวอร์ โดยมากโปรแกรมบนอินเทอร์เน็ตจะถูกติดตั้งลงในเครื่องที่เป็นไคลเอนต์ ทำให้เครื่องที่เป็นไคลเอนต์ มีความสามารถในการรับและส่ง อีเมล, เปิดชมเว็บไซต์, ทำการติดต่อสื่อสารกับกลุ่มข่าว (Newsgroups) ต่าง ๆ และสามารถโหลดไฟล์ที่ต้องการ ส่วนโปรแกรมอีกประเภทหนึ่งจะต้องติดตั้งกับเครื่องที่เป็นเซิร์ฟเวอร์ โปรแกรมประเภทนี้จะให้บริการ แก่ไคลเอนต์ที่ร้องขอใช้บริการ เครื่องเซิร์ฟเวอร์สามารถที่จัดการกับเครื่องไคลเอนต์ได้หลายเครื่องและพร้อมกันนั้นก็ยังสามารถจัดการกับงานต่างๆที่อยู่บนเครื่องไปพร้อมกัน เพราะว่าเทคโนโลยี ซ็อกเก็ต (Socket) ที่มีใช้งานในปัจจุบันบนอินเทอร์เน็ตมีความเสถียรภาพซ็อกเก็ตที่มีการใช้งานอยู่บน ไมโครซอฟท์ วินโดวส์ จะถูกเรียกว่า วินโดวส์ซ็อกเก็ต หรือจะเรียกสั้นๆว่า วินซ็อก (Winsock) ความหมายของซ็อกเก็ต คือ เครื่องมือของโปรแกรมที่จะถูกใช้ในการในการส่งและรับข้อมูลผ่านทางหมายเลขพอร์ตของ TCP/IP ที่กำหนด โปรแกรมจะสร้างซ็อกเก็ตได้ตามที่ต้องการเพื่อใช้ในการทำงาน แต่ 1 ซ็อกเก็ต จะต้องทำงานกับ 1 พอร์ตของ TCP/IP เท่านั้น โปรแกรมฝั่งไคลเอนต์จะสร้างซ็อกเก็ตและทำการกำหนดหมายเลขพอร์ตโดยวิธีการสุ่มหมายเลขขึ้นมา แต่ทางฝั่งเซิร์ฟเวอร์จะไม่เป็นอย่างนั้น โปรแกรมฝั่งเซิร์ฟเวอร์จะต้องทำตามข้อกำหนด ที่ได้มีการกำหนดไว้ใน TCP/IP พอร์ตซึ่งจะเป็นมาตรฐาน ตัวอย่าง : หมายเลข TCP/IP พอร์ต สำหรับ FTP เซิร์ฟเวอร์ คือ 21, และ สำหรับเว็บเซิร์ฟเวอร์ คือ 80 มีข้อกำหนดที่ครอบคลุม (Global Arrangement) สำหรับการเรียกใช้บริการ (Services) โดยจะมีการกำหนดหมายเลขของพอร์ตที่ไคลเอนต์ควรที่จะส่งคำร้องขอบริการ ดังตารางด้านล่าง

ตารางที่ 2.1 หมายเลขของพอร์ต

Protocol	Port	Description
SMTP	25	Simple Mail Transfer Protocol
POP3	110	Post Office Protocol
NNTP	119	Network News Transfer Protocol
FTP	21	File Transfer Protocol
HTTP	80	Hyper Text Transfer Protocol

โกลเอนต์จะเริ่มต้นสร้างการติดต่อผ่านเน็ตเวิร์ก (network sessions) กับเซิร์ฟเวอร์ โดยผ่านทาง network protocols ตัวใดตัวหนึ่งแล้วจะสร้างซ็อกเก็ต และกำหนดให้มันติดต่อยังเซิร์ฟเวอร์ที่ต้องการ เมื่อซ็อกเก็ตได้รับที่อยู่ (address) และ หมายเลขพอร์ต (Port) ของเซิร์ฟเวอร์แล้วมันก็จะติดต่อยังเซิร์ฟเวอร์นั้นทันที

ระบบเครือข่ายใช้โปรโตคอลมาตรฐานชื่อ TCP/IP ในการสื่อสารผ่านระบบเพื่อติดต่อกับเครื่องคอมพิวเตอร์อื่น ๆ โปรโตคอล TCP/IP นั้นประกอบด้วยส่วนที่สำคัญ 2 ส่วนก็คือ

- TCP (Transmission Control Protocol)
- IP (Internet Protocol)

ในการติดต่อสื่อสารกันจริงๆแล้วเราคงจะไม่สามารถเห็นขั้นตอนการทำงานของระบบได้ เพราะเป็นการทำงานของ Software&Hardware

2.2 โปรโตคอล TCP/IP

2.2.1 โปรโตคอล TCP (Transmission Control Protocol) เป็นโปรโตคอลที่มีการรับส่งข้อมูล โดยจะไม่คำนึงถึงปริมาณข้อมูลที่จะส่งไป แต่จะแบ่งข้อมูลเป็นส่วนย่อยๆก่อน แล้วจึงจะส่งไปยังปลายทางอย่างต่อเนื่องเป็นลำดับข้อมูล หากข้อมูลส่วนใดส่วนหนึ่งสูญหายไป ก็จะส่งข้อมูลส่วนนั้นใหม่อีกครั้ง สำหรับปลายทางก็จะทำหน้าที่เรียงส่วนของข้อมูลย่อยให้ต่อเนื่องกัน และประกอบกันกลับเป็นข้อมูลทั้งหมด ซึ่งจะแยกข้อมูลที่ไม่ถูกต้องออก

2.2.2 โปรโตคอล IP (Internet Protocol) ทำหน้าที่ให้บริการส่งผ่านข้อมูลเพื่อส่งข้ามไปยังเครือข่ายใดๆได้อย่างถูกต้อง ตัวโปรโตคอล IP จะทำงานแบบ Packet switching คือมีการส่งข้อมูลผ่านสวิทช์ (switch) ไปยังปลายทาง โดยข้อมูลจะเดินทางไปยังเครือข่ายต่างๆผ่านสวิทช์นี้ไปเรื่อยๆจนกว่าจะถึงปลายทาง ซึ่งในข้อมูลของโปรโตคอล IP จะมีข้อมูลของหมายเลข IP ปลายทางที่จะส่งข้อมูลไปและเมื่อถึงเครือข่ายปลายทางแล้ว จะมีกลไกแปลงหมายเลขฮาร์ดแวร์ประจำเครื่องที่ถูกต้องอีกทีหนึ่งด้วย

2.3 โครงสร้างของโปรโตคอล TCP/IP

โปรโตคอล TCP/IP มีการจัดแบ่งกลไกการทำงานออกเป็นชั้นๆ หรือ layer เหมือนกับมาตรฐาน OSI Model ในแต่ละ layer ของโปรโตคอล TCP/IP จะประกอบด้วย

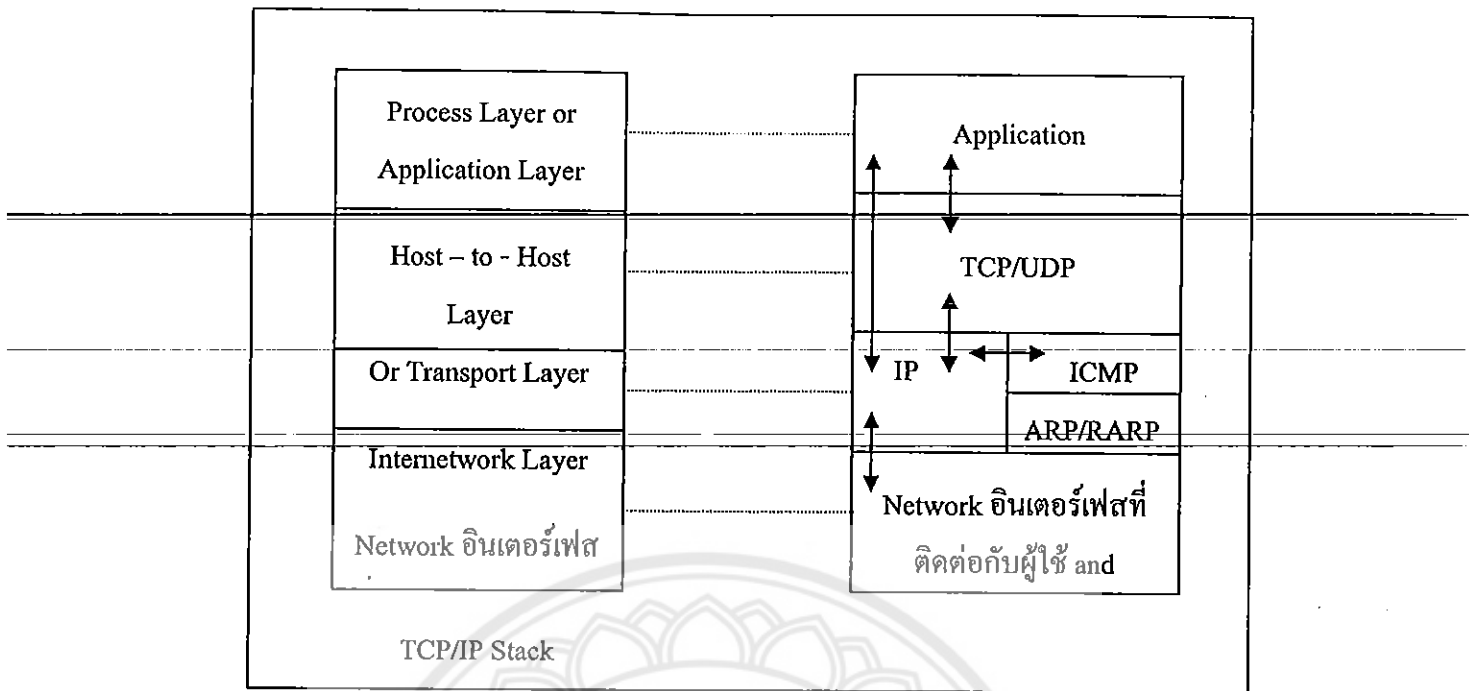
- Process layer หรือ Application Layer
- Host-to-Host layer หรือ Transport Layer
- Internetwork-layer
- Network อินเทอร์เน็ตที่ติดต่อกับผู้ใช้ layer

โดยเมื่อเปรียบเทียบกับมาตรฐาน OSI model แล้วจะเป็นดังรูปที่ 2.1 ซึ่งเราจะเห็นว่าบาง Layer ของโปรโตคอล TCP/IP เทียบได้กับมาตรฐาน OSI model ถึงสอง Layer และบาง Layer ก็จะทำหน้าที่เกี่ยวกับหลายๆ Layer ของ OSI model ตัวอย่างเช่น ในส่วน Network อินเทอร์เน็ตที่ติดต่อกับผู้ใช้ layer โปรโตคอล TCP/IP จะเทียบได้กับการนำเอา Data Link layer และ Physical layer ของมาตรฐาน OSI model มารวมกันเป็นต้น

ftp, telnet, mail application	Process	Application	7
	Layer	Presentation	6
โปรโตคอล TCP, UDP	Host-to-Host	Session	5
	Layer	Transport	4
โปรโตคอล IP	Internetwork	Network	3
	Layer	Data Link	2
ไครเวอร์ Ethernet, Token-Ring และอื่นๆ	Network	Physical	1
	อินเทอร์เน็ต		
	TCP/IP	OSI Model	

รูปที่ 2.1 TCP/IP Model และ OSI Model

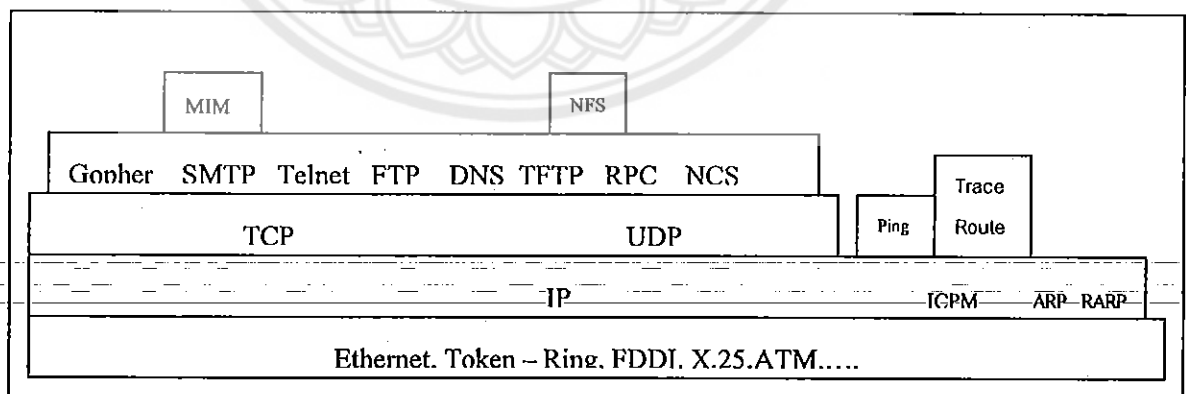
ในแต่ละกลไกของโปรโตคอล TCP/IP จะมีโปรโตคอลอื่นๆ ในชุดของ TCP/IP ร่วมทำงานอยู่ด้วย จึงทำให้เป็นที่มาของชื่อเรียก Protocol Stack เนื่องจากมีโปรโตคอลซ้อนทับกันอยู่เพื่อช่วยกันทำงานดังรูปต่อไปนี้



รูปที่ 2.2 Protocol Stack

จากรูปจะเห็นได้ว่า มีโปรโตคอลในแต่ละระดับซ้อนทับกันอยู่หลายตัวด้วยกัน ซึ่งเราจะพูดกันในรายละเอียดต่อไป การซ้อนกันเป็นชั้นๆ หรือแต่ละ Layer นี้หากเป็น OSI Model จะมีข้อบังคับให้แต่ละชั้นติดต่อได้กับเฉพาะชั้นที่ติดกับตนเองเท่านั้น แต่สำหรับ TCP/IP Stack แล้วจะเห็นว่าบางชั้นสามารถละเลยหรือข้ามไปติดต่อกับชั้นอื่นที่ไม่ติดกับตนได้

2.3.1 Process layer หรือ Application Layer



รูปที่ 2.3 ลำดับชั้นการทำงานของโปรโตคอล

จากรูป 2.3 แสดงลำดับชั้นการทำงานของโปรโตคอล TCP/IP เทียบกับมาตรฐาน OSI model นั้น ในชั้นบนสุดเรียกว่า Process layer ทำงาน 2 หน้าที่เทียบได้กับ Application layer และ Presentation layer ในชั้นนี้จะรองรับการทำงานของแอปพลิเคชันต่างๆ ที่ทำงานเป็นโปรเซส อยู่ใน

เครื่องเซิร์ฟเวอร์ ที่ให้บริการและเครื่องที่ขอใช้บริการ หรือ โคลเอนต์ (โคลเอนต์) ซึ่งจะติดต่อกันผ่านโปรโตคอลเฉพาะแอฟพลิเคชันที่หนึ่ง ตัวอย่างเช่น เมื่อผู้ใช้งานอินเทอร์เน็ตต้องการถ่ายโอนไฟล์หรือดาวน์โหลด ข้อมูลจากเครื่องเซิร์ฟเวอร์ที่ให้บริการ โดยอาจจะเรียกใช้โปรแกรม ftp โคลเอนต์ ที่เข้าไปเช่น โปรแกรม ws_ftp ติดต่อกับโปรเซส ftp ที่กำลังให้บริการอยู่ที่เครื่องเซิร์ฟเวอร์ จากนั้นตัวโปรเซส ftp ก็จะเรียกใช้โปรโตคอล FTP (File Transfer Protocol) เพื่อทำการถ่ายโอนไฟล์นี้ หรือถ้าผู้ใช้ต้องการเรียกใช้งานคอมพิวเตอร์เครื่องที่อยู่ห่างไกลออกไปด้วยการใช้โปรแกรม telnet ที่เครื่องเซิร์ฟเวอร์ให้บริการตัวโปรเซส Telnet ที่ทำงานอยู่ก็จะเรียกใช้โปรโตคอล Telnet เพื่อติดต่อกัน หรือในกรณีที่มีการเรียกใช้โปรแกรม web browser เช่น Netscape Navigator เพื่อเรียกดูเว็บเพจในเว็บไซท์ CNN ที่เครื่องซึ่งให้บริการเว็บของ CNN ก็จะมีโปรเซส HTTP (HyperText Transfer Protocol) ทำงานอยู่และจะติดต่อกับผู้ใช้ผ่าน โปรโตคอล HTTP เป็นต้น

การทำงานของแอฟพลิเคชันต่างๆ จะอยู่ที่ Process layer นี้ และมีการติดต่อกันตามแต่ละโปรโตคอลเฉพาะแล้วแต่แอฟพลิเคชันที่ใช้งาน จากการที่ Process layer ของ TCP/IP รองรับให้โปรโตคอลอื่นทำงานได้หลายโปรเซสและหลายโปรโตคอลได้พร้อมกันนั้น ทำให้ผู้ใช้สามารถใช้งานได้หลายๆอย่างพร้อมกัน เช่น เปิดโปรแกรม Internet Explorer เพื่อเรียกดูเว็บเพจพร้อมกับใช้งานโปรแกรม Outlook Express เพื่อรับส่งอีเมลล์ไปพร้อมกันได้โดยไม่ต้องรอให้ทำงานอย่างหนึ่งอย่างใดเสร็จก่อน หรือในปัจจุบันมีการพัฒนาโปรแกรม web browser ให้สามารถเรียกใช้งานโปรโตคอลได้มากขึ้น ทำให้เราสามารถใช้งานโปรแกรม web browser โอนถ่ายไฟล์ข้อมูลที่ใช้โปรโตคอล FTP ได้โดยไม่ต้องไปหาโปรแกรมอื่นมาใช้

โปรโตคอลหลายๆที่ทำงานใน Process layer ซึ่งผู้ใช้งานมักจะคุ้นเคยกันดีได้แก่ FTP (File Transfer Protocol) , Telnet, HTTP (HyperText Transfer Protocol) และ SMTP (Simple Mail Transfer protocol) นอกจากนี้ยังมีโปรโตคอลอื่นที่อยู่เบื้องหลัง ซึ่งทำงานโดยที่ผู้ใช้ไม่สามารถมองเห็นได้จากโปรแกรมหรือไม่ได้ที่การใช้งานโดยตรง เช่น

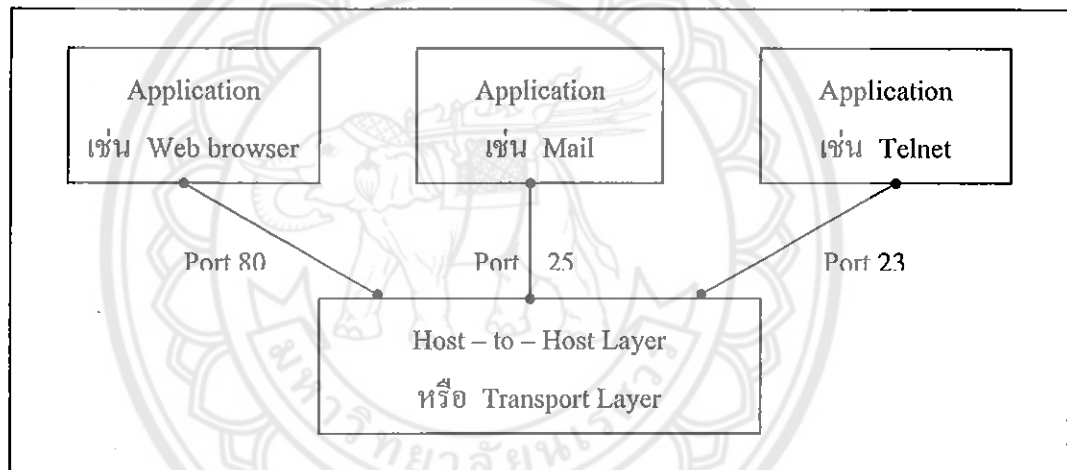
- โปรโตคอล DNS (Domain Name System) ที่ทำหน้าที่แปลงข้อมูลชื่อ domain name หรือชื่อเว็บไซท์ทั้งหลายให้เป็นหมายเลข IP Address
- โปรโตคอล SNMP (Simple network management Protocol) ใช้ในการควบคุมและตรวจสอบอุปกรณ์ที่อยู่ในเครือข่าย
- โปรโตคอล DHCP (Dynamic Host Configuration Protocol) ทำหน้าที่แจกจ่ายข้อมูลพารามิเตอร์ของเครือข่ายให้กับเครื่องลูกข่ายที่เชื่อมต่ออยู่

2.3.2 Host-to-Host layer หรือ Transport Layer

ผู้ใช้งานอินเทอร์เน็ตเคยสงสัยหรือไม่ว่าเครื่องเซิร์ฟเวอร์ที่ให้บริการต่างๆ เช่น เว็บเซิร์ฟเวอร์นั้น เมื่อมีผู้เข้ามาเรียกใช้บริการพร้อมกันหลายคน จะมีวิธีการส่งข้อมูลกลับไปยังต้นทาง

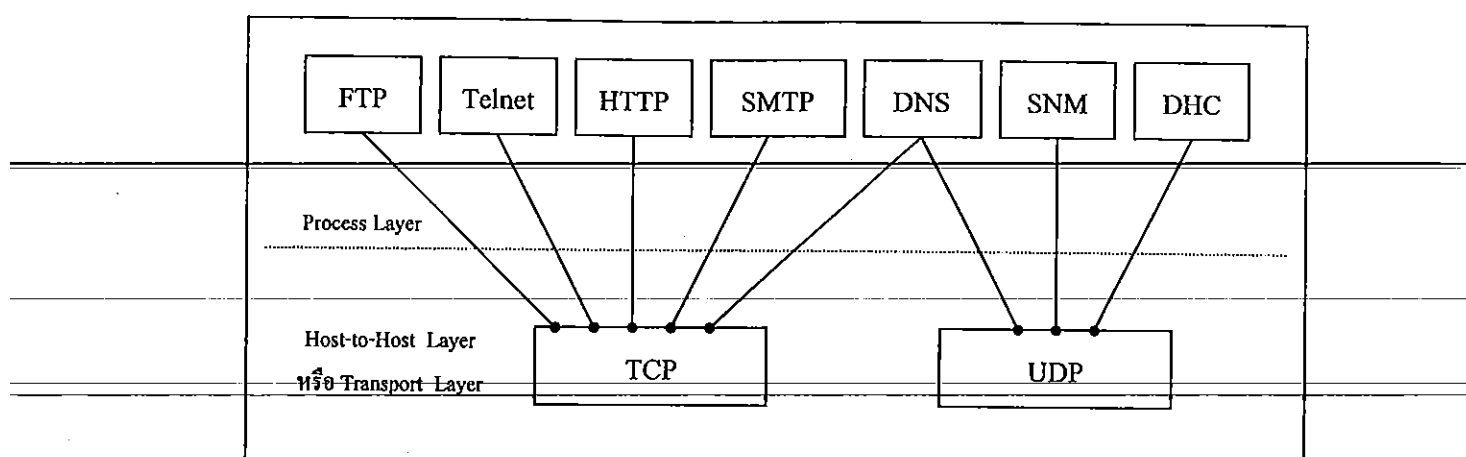
ได้อย่างไร โดยไม่ผิดพลาด ซึ่งบางครั้งผู้ใช้รายหนึ่งอาจจะเปิดโปรแกรม web browser ซ้อนกันเพื่ออ่านข้อมูลจากเว็บเพจอื่นๆ พร้อมกันไปได้ ดังนั้นระบบจะทราบได้อย่างไรถึงการจัดส่งข้อมูลได้อย่างถูกต้อง

การทำงานที่ชั้นของ Host-to-Host layer นี้จะมีบทบาทในการจัดการต่อจาก Process layer บางครั้งเรามักเรียกชั้น Host-to-Host ว่าเป็น Transport layer ซึ่งไม่ใช่ชั้นของ Transport layer ในมาตรฐาน OSI Model การทำงานของ Host-to-Host layer นี้จะมีการสร้าง Connection หรือการเชื่อมต่อกันระหว่างแอปพลิเคชันกับ Host-to-Host layer โดยจุดที่เชื่อมกันเพื่อรับส่งข้อมูลนี้เรียกว่า Port หรือ Socket (คำว่า Port ในที่นี้ไม่ได้หมายถึง Port ทางฮาร์ดแวร์) และในแต่ละแอปพลิเคชันก็จะสร้างการเชื่อมต่อผ่าน Port ได้พร้อมกันหลายแอปพลิเคชัน ซึ่งการใช้งาน Port ของแต่ละแอปพลิเคชันที่อยู่ในชั้น Process layer จะแตกต่างกันตามหมายเลขที่กำหนดไว้ และแต่ละโปรโตคอลจะมีการใช้งาน Port หมายเลขต่างๆ ไม่ซ้ำกัน ตามรูป



รูปที่ 2.4 Port ที่ใช้ในโปรโตคอลต่างๆ

เมื่อแอปพลิเคชันทำงานผ่านโปรโตคอลในชั้น Process Layer จะมีการส่งผ่านข้อมูลไปยัง Host-to-Host layer ที่ชั้นนี้จะมีการเชื่อมต่อผ่าน Port ที่กำหนด ทำให้การรับส่งข้อมูลในแต่ละโปรโตคอลทำได้ถูกต้อง ถึงแม้ว่าในเครื่องเซิร์ฟเวอร์ที่ให้บริการจะมีการทำงานอยู่หลายโปรเซสที่แตกต่างกันก็ตาม หรือมีผู้ใช้บริการเข้ามาใช้งานพร้อมกันใช้งานจำนวนมากและหลายแอปพลิเคชันในเวลาเดียวกัน ในชั้น Host-to-Host หรือ Transport layer ของ TCP/IP นี้ จะมีโปรโตคอลทำงานอยู่ 2 โปรโตคอลที่แตกต่างกัน คือ โปรโตคอล TCP และโปรโตคอล UDP (User Datagram Protocol) ในการส่งผ่านข้อมูลลงไปที่ยื่นถัดๆ ไป เราจะเห็นว่าโปรโตคอล TCP และ UDP จะถูกผนึกเข้าไปในโปรโตคอล IP อีกทีหนึ่งและส่งต่อไปยังเครือข่ายอินเทอร์เน็ตต่อไป



รูปที่ 2.5 การเรียกใช้โปรโตคอล TCP และ UDP

ตัวโปรโตคอล TCP และโปรโตคอล UDP จะมีแอฟพลิเคชันเฉพาะเพื่อเรียกใช้งาน แยกกันคือแอฟพลิเคชันที่ใช้โปรโตคอล FTP , Telnet , HTTP และ SMTP จะมีการส่งผ่านข้อมูล โดยเรียกใช้โปรโตคอล TCP ส่วนแอฟพลิเคชันที่ใช้งานโปรโตคอล SNMP และ DHCP จะส่งผ่านข้อมูลโดยเรียกใช้โปรโตคอล UDP และสำหรับโปรโตคอล DNS นั้นจะสามารถเรียกใช้งานได้ทั้ง TCP และ UDP ดังรูปซึ่งเหตุผลที่มีการเรียกใช้โปรโตคอล TCP และ UDP แตกต่างกัน ก็เนื่องมาจากวิธีการทำงานของทั้งสองโปรโตคอลต่างกันนั่นเอง

2.4 ความรู้พื้นฐานเกี่ยวกับ Visual Basic

เนื่องจากภาษา Visual Basic เป็นเครื่องมือสำหรับเขียนโปรแกรมที่ได้รับความนิยมทั่วโลก รวมทั้งในเมืองไทย และเป็นภาษาที่เข้าใจได้ไม่ยากนัก ซึ่งความเป็นมาของภาษา Basic นั้นได้ถูกสร้างขึ้นในปี 1963 โดย Horn Keneny และ Thomas Kutz ที่มหาวิทยาลัย Dartmouth สร้างขึ้นเพื่อใช้ในการสอนการเขียนโปรแกรม โดยเน้นภาษาง่ายต่อการเข้าใจและการใช้งาน โดยที่เราสามารถทำการสร้างแอฟพลิเคชันที่สามารถใช้งานกับระบบ Windows ได้ง่ายขึ้นไม่เหมือนกับในอดีตที่ต้องเขียนโปรแกรมที่ซับซ้อนโดยที่บริษัทไมโครซอฟท์ได้สังเกตเห็นว่าถ้าสร้างแอฟพลิเคชันยากก็เหมือนกับว่ามา Windows ทางอ้อมด้วยนั่นเอง ทำให้มีการสร้างการเขียนโปรแกรมแบบ Visual Programming กำเนิดขึ้นมา รูปแบบนี้ก็คือ การเขียนโปรแกรมพร้อมกับการเห็นผลลัพธ์ที่เกิดขึ้น Visual Basic เป็นหนึ่งใน Visual Programming ที่ไมโครซอฟท์สร้างขึ้นมา และด้วยความเรียบง่ายของภาษาและการเขียนโปรแกรมที่รวดเร็ว ทำให้ได้รับความนิยมสูงในปัจจุบัน

2.4.1 จุดเด่นของ Visual Basic

Visual Basic สร้างแอปพลิเคชันได้อย่างรวดเร็ว Visual Basic ได้รับการวางตัวให้เป็นเครื่องมือที่ช่วยให้การสร้างแอปพลิเคชันง่ายและรวดเร็วเพื่อลดเวลาการสร้างแอปพลิเคชันให้สั้นที่สุด ซึ่งรูปแบบนี้เรียกว่า Rapid Application Development หรือ RAD ทั้งนี้เพราะการขจัดงานที่ต้องทำซ้ำ ๆ ซาก ๆ ของโปรแกรมออกไปขจัดสิ่งที่ไม่จำเป็นเกี่ยวกับการควบคุมฮาร์ดแวร์และการจัดการภายในของ Windows ออกไป เหลือเฉพาะ โฟกัสของปัญหาของงานจริง ๆ แล้วเขียนโปรแกรมจัดการกับปัญหานั้น ๆ ส่วนที่เหลือให้ Visual Basic จัดการ การเขียนโปรแกรมด้วย Visual Basic นั้นง่ายต่อการเรียนรู้ ใน Visual Basic นั้นถ้าลองศึกษาภาษาที่ใช้มันจะเห็นว่าง่ายเพราะอ่านแล้้งเข้าใจได้ไม่ยาก ใกล้เคียงกับภาษาปกติจึงทำให้เราเรียนรู้ได้ง่ายกว่าภาษาอื่น ๆ นอกจากนั้น Visual Basic ยังมีเครื่องมือที่ช่วยในการเขียนโปรแกรมที่สะดวกไม่ยุ่งยากเพราะเครื่องมือนี้ช่วยให้เราไม่ต้องจำหลักไวยากรณ์ที่ยุ่งยาก สามารถตรวจสอบได้อย่างอัตโนมัติว่าโปรแกรมนั้นถูกต้องตามหลักไวยากรณ์หรือไม่ และมีการแยกส่วนต่าง ๆ ของโปรแกรมไว้อย่างเป็นระเบียบ

2.4.2 การเขียนโปรแกรมเพื่อสร้างแอปพลิเคชันจาก Visual Basic

ในการเขียนโปรแกรมเพื่อสร้างแอปพลิเคชันจาก Visual Basic นั้นจะมีวิธีการสร้างที่แตกต่างจากโปรแกรมอื่นที่เราเคยเรียนรู้มาเพราะ Visual Basic จะทำงานแบบ Event-Driven ซึ่งก็คือการเขียนโปรแกรมแบบ “ ถ้าเหตุการณ์เกิดขึ้น เราจะดำเนินการอย่างไร ” กล่าวคือ ถ้าเหตุการณ์นี้เกิดขึ้น เราจะจัดการกับเหตุการณ์นี้อย่างไร เป็นโปรแกรมที่รองรับเหตุการณ์ Visual Basic เป็นโปรแกรมที่สนับสนุนการเขียนโปรแกรมแบบ Event-Driven โดยมีเครื่องมือต่าง ๆ ช่วยในการจัดการกับเหตุการณ์ต่าง ๆ ที่จะเกิดขึ้น

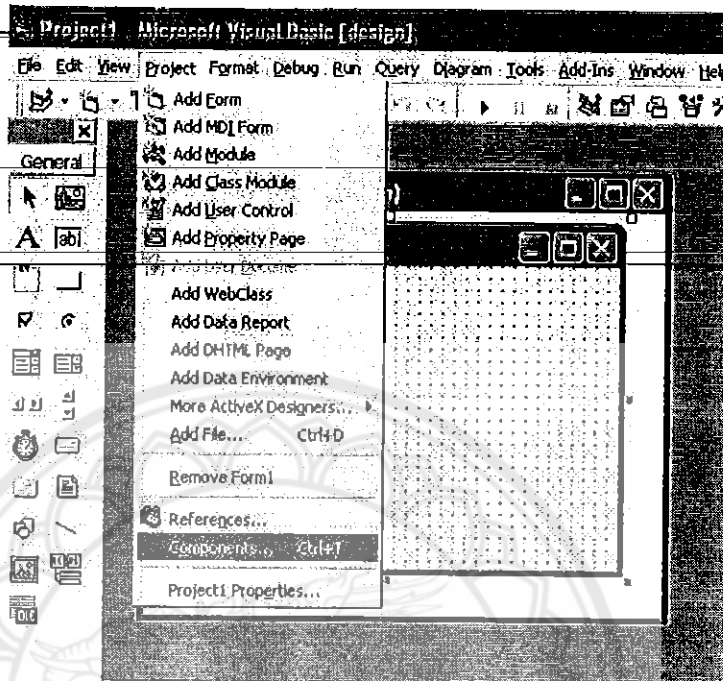
การเขียนโปรแกรมบนระบบ Windows ให้สามารถติดต่อสื่อสารบนระบบเครือข่าย ถ้าเป็นเมื่อก่อนบนระบบปฏิบัติการ DOS นั้นคงจะยากทีเดียว แต่ปัจจุบันนี้เครื่องคอมพิวเตอร์ก็ใช้ระบบปฏิบัติการ Windows กันแล้วเนื่องจากมีเครื่องมือ (Tool) ที่ช่วยให้เราสามารถเขียนโปรแกรมติดต่อผ่านระบบเครือข่าย ที่เราจะกล่าวถึงก็คือ MS Winsock Control 6 ซึ่งเป็นเครื่องมือที่อำนวยความสะดวกสำหรับการเขียนโปรแกรมติดต่อผ่านระบบเครือข่าย-โดย โพรโทคอล TCP/IP

2.4.3 MS Winsock Control 6 *

คอนโทรลที่ชื่อว่า MS Winsock Control 6 ที่มากับ Visual Basic เป็นเครื่องมือที่มีประโยชน์ในการพัฒนา โปรแกรมบน Internet ให้กับคุณ เมื่อเปรียบเทียบกับคอนโทรล ActiveX ตัวอื่นๆ Winsock จะเป็นคอนโทรลที่ถูกนำมาใช้งานมากคอนโทรลหนึ่ง แต่อย่างไรก็ตามถ้าต้องการที่จะคอนโทรลขึ้นมาเอง และไม่ต้องการใช้คอนโทรลของผู้พัฒนาคนอื่น ซึ่งอาจมีข้อผิดพลาดที่คุณไม่ต้องการ คุณก็จำเป็นต้องมีพื้นฐานที่เกี่ยวกับ Network Protocols และ หลักในการติดต่อสื่อสาร แต่นี้ คุณก็สามารถที่จะสร้าง โปรแกรมบนอินเทอร์เน็ตได้ตามที่ต้องการ

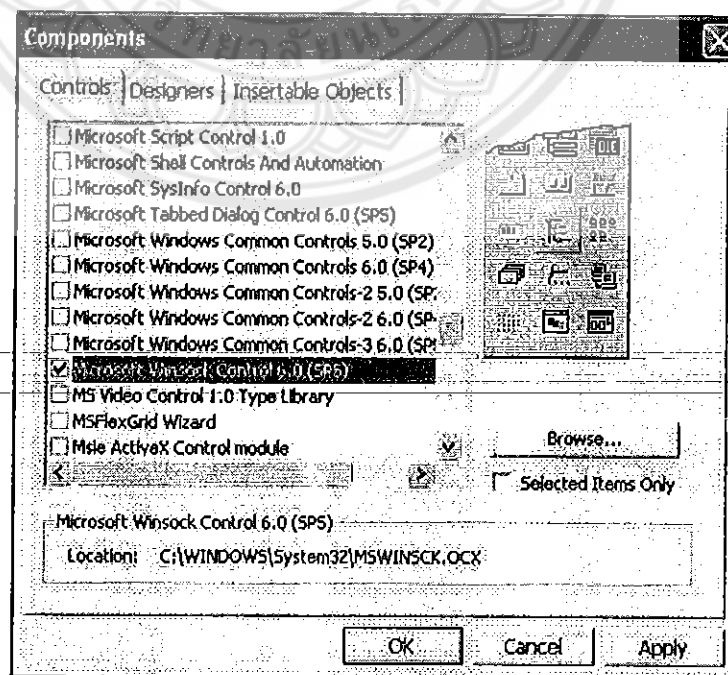
การเลือกใช้ Components MS Winsock Control 6

1. เลือก Menu → Project → Components

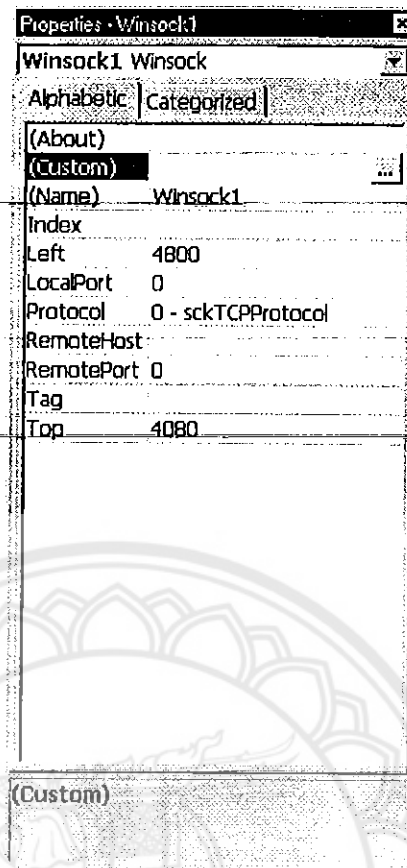


รูปที่ 2.6 Components MS Winsock Control 6

2. เลือก Microsoft Winsock Control 6.0



รูปที่ 2.7 Microsoft Winsock Control 6.0



รูปที่ 2.8 Properties Winsock Dialog

Winsock Procedure

ในส่วนของ Windows Code Object → Winsock มี Procedure สำหรับกระทำติดต่อกับสื่อสาร ทั้งทางฝ่าย เซิร์ฟเวอร์ และ ไคลเอนต์

- Close คือ เหตุการณ์เมื่อมีหยุดหรือยกเลิกการติดต่อกับสื่อสารของฝ่าย
- เซิร์ฟเวอร์ หรือไคลเอนต์ โดย Function Winsock.Close ซึ่งจะเราสามารถจะใช้ ตรวจสอบฝ่ายตรงข้ามว่ามีการติดต่อยู่หรือไม่โดยอาจจะใส่ Message เตือนเป็นต้น
- Connect เป็นเหตุการณ์ที่ฝ่าย ไคลเอนต์ มีการส่งสัญญาณติดต่อกับมายัง
- Sever ส่งผลให้ Procedure นี้ของฝ่าย เซิร์ฟเวอร์ ก็เลยทำงานขึ้นมา
- ConnectionRequest เป็นเหตุการณ์เมื่อฝ่าย ไคลเอนต์ ส่งสัญญาณติดต่อกับมายัง เซิร์ฟเวอร์ Procedure ส่วนนี้ก็จะทำงานพร้อมกับค่า RequestID As Long ซึ่งเป็นหมาย ที่ Gen ขึ้นมาในระบบค่านั้นจะไม่เหมือนเดิม โดยจะให้ฝ่าย เซิร์ฟเวอร์ รั บรู้ว่าใช้ ID จากคอนโทรลตัวใด
- DataArrival เหตุการณ์นี้เกิดขึ้นเมื่อมีการส่งข้อมูลระหว่าง เซิร์ฟเวอร์ และ ไคลเอนต์ Porcedure นี้ก็จะทำงานขึ้นมา พร้อมกับค่าจำนวน bytesTotal As Long ที่รับเข้ามา

- Error เหตุการณ์ที่เกิดความผิดพลาดระหว่างการติดต่อสื่อสารระหว่าง เซิร์ฟเวอร์ และ ไคลเอนต์ โดยจะส่งค่า Number As Integer มาให้ว่าเป็นหมายเลขใดพร้อมทั้งรายละเอียดของการผิดพลาดในเหตุการณ์นั้นๆ คือ Description As String
- SendProgress จะเกิดขึ้นในขณะที่มีการส่งข้อมูลอยู่เหตุการณ์นี้จะทำงานเมื่อส่งข้อมูลหมดแล้วก็จะส่งผลทำให้เกิด Event SendComplete
- SendComplete เหตุการณ์เมื่อมีการส่งข้อมูลออกไปยังฝ่ายตรงข้ามเสร็จเรียบร้อยแล้ว

Winsock Properties & Events

Accept (requestID) คือการตกลงกันระหว่าง เซิร์ฟเวอร์ และ ไคลเอนต์ ในการเลือกหมายเลข ID Control ให้ตรงกันเพื่อสามารถสื่อสารได้ถูกต้อง

Close เป็นการส่งสัญญาณยกเลิกการติดต่อระหว่างกัน จะเป็นฝ่าย เซิร์ฟเวอร์ หรือ ไคลเอนต์ ก็ได้ ที่จะใช้ Function นี้ จากนั้นจะทำให้ Procedure close ในฝ่ายตรงข้ามทำงาน

Connect เป็นการส่งสัญญาณว่าตอนนี้ทำการติดต่อเรียบร้อยแล้ว ซึ่งจะส่งผลให้ Procedure ฝ่ายตรงข้ามทำงาน

Getdata เป็นการรับข้อมูลเมื่อฝ่ายตรงข้ามส่งมาโดยประโยคคำสั่งนี้จะอยู่ในส่วนของ Procedure DataArrival เนื่องจากเป็นเหตุการณ์ที่การกระทำขณะเมื่อฝ่ายตรงข้ามส่งข้อมูลเข้ามา

Listen การกระทำที่จะคอยตรวจสอบสัญญาณที่ส่งไปว่าฝ่ายตรงข้ามตอบรับการร้องขอการติดต่อ

LocalHostName คำสั่งนี้จะส่งชื่อของ Computer name ของเครื่องนั้นๆ

```
Winsock1.LocalHostName = "ComputerName"
```

LocalIP คำสั่งนี้จะทำการส่งหมายเลข IP Address

```
Winsock1.LocalIP = "192.168.0.1"
```

LocalPort คำสั่งที่จะส่งค่าของหมายเลขในการติดต่อ TCP/IP ของเครื่องนั้นๆ

```
Winsock1.LocalPort = "7001"
```

RemoteHost กำหนดหรือคืนค่าชื่อ Computer name ของเครื่องที่จะทำการติดต่อ

```
Winsock1.RemoteHost = "ComputerName"
```

RemoteHostIP กำหนดหมายเลข IP Address ของเครื่องที่จะทำการติดต่อ

```
Winsock1.RemoteHostIP = "192.168.0.1"
```

RemoteHostPort กำหนดหมายเลข Port ที่จะใช้ในการติดต่อระหว่างกัน

```
Winsock1.RemoteHostIP = "7001"
```

SocketHandle จะคืนค่าของช่องทางที่ใช้ในการติดต่อระหว่างกันซึ่งสามารถเรียกดูได้ดังนี้

```
Winsock1.SocketHandle
```


State จะคืนค่าของสถานะของ Socket ขณะที่ใช้ติดต่อกันอยู่โดยอาจจะใช้ตรวจสอบสถานะ โดยค่าคงที่เหล่านี้เช่น sockClosed (มีค่า=0) Socket ปิดการใช้งาน, sockOpen (มีค่า= 1) Socket เปิดใช้งาน หรือ sockError (มีค่า = 9) Socket มีความผิดพลาดเกิดขึ้น เป็นต้น

Basic Winsock Process Connect

การเชื่อมต่อกัน (Connection) ของ ไคลเอนต์-เซิร์ฟเวอร์

เหตุการณ์ที่ต้องทำในฐานะที่เป็น ฝ่าย เซิร์ฟเวอร์ คือการตรวจสอบสัญญาณจากฝั่ง ไคลเอนต์ ในที่นี้ใช้ Event Click ของปุ่มชื่อ cmdListen

```
Private Sub cmdListen_Click()
```

```
    Winsock1.LocalPort = "7001"
```

```
    Winsock1.Listen
```

```
End Sub
```

เหตุการณ์ในฐานะไคลเอนต์ ก็จะตอบรับ โดยส่งสัญญาณไปให้กับ เซิร์ฟเวอร์

```
Private Sub cmdConnect_Click()
```

```
    Winsock1.RemoteHost = "192.168.0.1"
```

```
    Winsock1.RemotePort = "7001" ' กำหนดให้ตรงกันทั้ง 2 ฝ่าย
```

```
    Winsock1.Connect
```

```
End Sub
```

เมื่อคลิกปุ่ม cmdConnect ตอนนี้ทาง Winsock Control ฝั่ง Sever จะเกิด Event Connect ซึ่งใน Event นี้เราอาจจะใส่ Message เตือนก็ได้ดังนี้

```
Private Sub Winsock1_Connect()
```

```
    MsgBox "ตอบรับการติดต่อกลับมาแล้ว", vbExclamation
```

```
End Sub
```

ถ้า ข้อความ ขึ้นก็เป็นอันว่าทั้ง เซิร์ฟเวอร์ กับ ไคลเอนต์ สามารถติดต่อสื่อสารกันได้แล้ว การส่ง Data ของ ไคลเอนต์-เซิร์ฟเวอร์

```
Private Sub cmdSend_Click()
```

```
    Winsock1.SendData "Hi! How are you ? "
```

```
End Sub
```

เมื่อฝ่าย เซิร์ฟเวอร์ ส่งData ออกไปที่ Winsock control ฝ่าย ไคลเอนต์ จะเกิด Event DataArrival เราก็จะรับข้อมูลนั้นด้วยคำสั่ง Getdata โดยเขียนโปรแกรมไว้ใน Event นี้

```
Sub Winsock1_DataArrival(ByVal bytesTotal As Long)
```

```
    Dim StrValue As String
```

```
Winsock1.GetData StrValue
```

```
End Sub
```

ถ้าหากต้องการยกเลิกการติดต่อก็ใช้คำสั่ง Close เมื่อ เซิร์ฟเวอร์ หรือ ไคลเอนต์ ใช้คำสั่งนี้ จะทำให้เกิด Event Close ขึ้นกับ Winsock control ของฝั่งนั้น ดังเช่นเราสร้างปุ่ม Close ไว้

```
Private Sub cmdClose_Click()
```

```
Winsock1.SendData "ยกเลิกการติดต่อแล้วครับ"
```

```
Winsock1.Close
```

```
End Sub
```

เมื่อฝั่ง เซิร์ฟเวอร์ มีการคลิกปุ่มเกิดขึ้นก็จะทำให้ Winsock Control ฝั่ง ไคลเอนต์ เกิด Event Close โดยอาจจะเขียน code ข้อความ เตือนไว้ดังต่อไปนี้

```
Private Sub Winsock1_Close()
```

```
MsgBox "ยกเลิกการติดต่อแล้วครับ", vbExclamation
```

```
End Sub
```



บทที่ 3

การศึกษาและพัฒนาโปรแกรม

จากการที่ได้ศึกษารูปแบบการติดต่อสื่อสารและหลักการพัฒนาโปรแกรมผ่านระบบเครือข่ายภายในบทนี้จะเป็นการออกแบบและพัฒนาโครงงาน โดยมีรายละเอียดดังต่อไปนี้

3.1 เครื่องมือที่ใช้ในการพัฒนาโปรแกรม

- ภาษาที่ใช้พัฒนาโปรแกรม คือ Microsoft Visual Basic 6
- Components ที่ใช้ในการสื่อสาร คือ Microsoft Winsock Control
- Library ที่ใช้ คือ CDO Library ของ Microsoft Exchange
- โพรโทคอลที่ใช้ในการติดต่อสื่อสาร คือ TCP / IP
- Smtip เซิร์ฟเวอร์ ที่ใช้ในการส่งอีเมลล์ คือ exchange.nu.ac.th
- ฐานข้อมูลที่ใช้ในการเก็บข้อมูลของยูสเซอร์ คือ Microsoft Access 97

3.2 การออกแบบระบบ

โปรแกรมจะแบ่งออกเป็น 2 ส่วนใหญ่ ๆ คือ

1. ส่วนของ เซิร์ฟเวอร์ (Server)
2. ส่วนของ ไคลเอนต์ (Client)

3.2.1 เซิร์ฟเวอร์ (Server)

เซิร์ฟเวอร์จะเป็นเครื่องที่ให้บริการต่างๆแก่เครื่องที่เป็นไคลเอนต์ตามแต่ที่ไคลเอนต์ขอบริการมา โดยการรับค่าต่าง ๆ ที่ส่งมาจากไคลเอนต์ตามแต่แต่ละส่วนที่ไคลเอนต์ขอบริการมาแล้วจึงส่งสิ่งที่ไคลเอนต์ต้องการกลับไปยังไคลเอนต์

3.2.2 ไคลเอนต์ (Client)

ไคลเอนต์จะเป็นเครื่องคอมพิวเตอร์ที่จะร้องขอบริการ หรือข้อมูลจากเครื่องเซิร์ฟเวอร์ ในการส่งและรับข้อมูลผ่านนั้นจะผ่านทางหมายเลขพอร์ตที่กำหนด

3.3 ส่วนอธิบายโปรโตคอลของการติดต่อสื่อสาร

ส่วนที่ 1 ไคลเอนต์ --> เซิร์ฟเวอร์

- | | | |
|----------|---|-----------------------------------|
| cmdLogin | : | ยูสเซอร์เนม , พาสเวิร์ด |
| psDecode | : | ถอดรหัสพาสเวิร์ด |
| cmdMsgID | : | ส่งไอดีไปขอข้อความมาใส่ไว้ในลิสต์ |

cmdMDesc	:	ส่งพาร์ทไอดีพร้อมชื่อไฟล์ไปขอรายละเอียดของไฟล์มาก่อน
cmdMFile	:	ส่งพาร์ทพร้อมชื่อไฟล์ไปขอข้อความ
cmdMRead	:	บอกเซิร์ฟเวอร์ว่าได้รับข้อความแล้วให้เปลี่ยนสถานะข้อความ เป็น "r"
cmdMDele	:	ส่งลบข้อความ
cmdStatu	:	ส่งว่าออนไลน์หรือออฟไลน์
reqLName	:	ขอชื่อ , ไอดี มาใส่ในลิสต์
reqMSend	:	ส่งข้อความไปให้เซิร์ฟเวอร์
fileList	:	แสดงชื่อไฟล์ในระดับที่ขอไป
fileDLoa	:	ส่งชื่อไฟล์ที่ต้องการจะดาวน์โหลด
fileULoa	:	ส่งชื่อไฟล์ที่ต้องการจะอัปโหลด
fileAcpt	:	รับไฟล์
fileCanc	:	ยกเลิกการรับไฟล์

ส่วนที่ 2 เซิร์ฟเวอร์ --> ไคลเอนต์

cmdLogin	:	ถ้าพาสเวิร์ดถูกต้องจะส่งระดับ , สถานะ , ไอดี , ชื่อยูสเซอร์
psEncode	:	เข้ารหัสพาสเวิร์ด
cmdMList	:	ส่งรายละเอียดของข้อความและจำนวนข้อความจากไฟล์ Message.ini
cmdMOpn1	:	ส่งรายละเอียดของไฟล์ที่ขอ
cmdMOpn2	:	ส่งเนื้อหาของข้อความ
cmdMDele	:	บอกว่าลบข้อความแล้ว
fileName	:	ส่งชื่อไฟล์ไปให้ไคลเอนต์
fileRequ	:	ส่งชื่อไฟล์และขนาดของไฟล์
fileCont	:	เริ่มส่งไฟล์ , ส่งไฟล์ต่อ
fileStop	:	หยุดการส่งไฟล์
fileAcpt	:	รับไฟล์
fileDelS	:	ลบไฟล์เรียบร้อยแล้ว
chatMsgx	:	บอกคนเข้า-ออก
chatText	:	สนทนา

3.4 โครงสร้างและหลักการทำงานของโปรแกรม

โปรแกรมได้แบ่งการทำงานออกเป็น 7 ส่วนใหญ่ ๆ ดังนี้

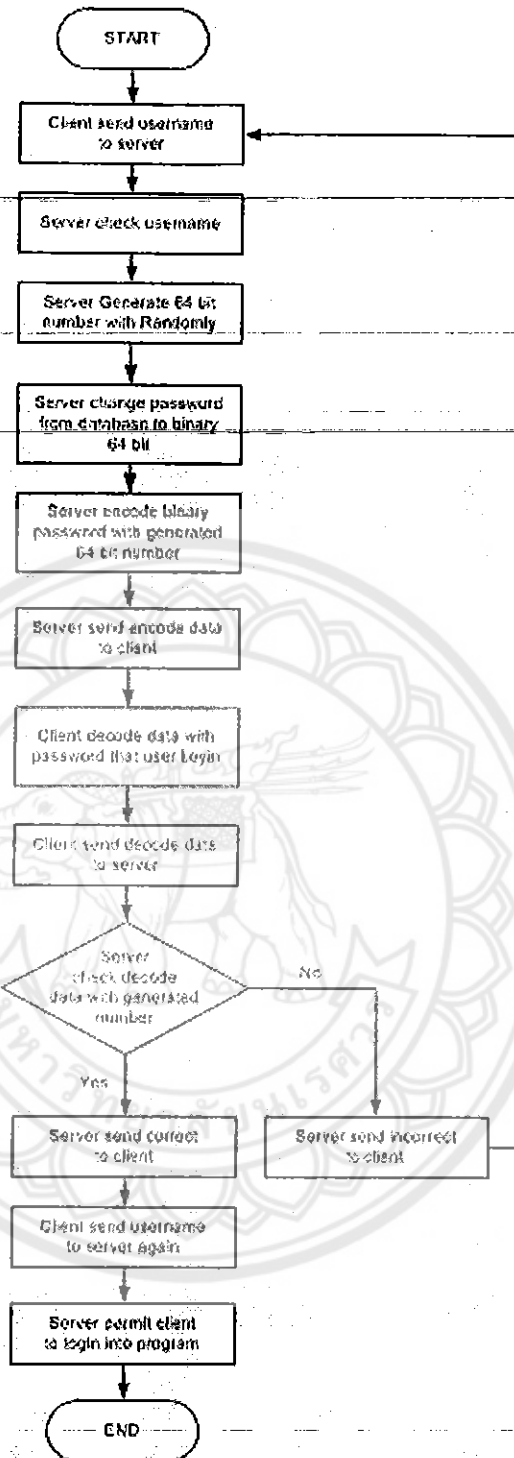
ส่วนที่ 1	ล็อกอิน (Login) / ล็อกออฟ (Logoff)
ส่วนที่ 2	ข้อความ (message)
ส่วนที่ 3	อัปโหลด (Upload) / ดาวน์โหลด (Download)
ส่วนที่ 4	ส่งอีเมล (Send e-mail) / สมุดรายชื่อ (Address Book)
ส่วนที่ 5	ค้นหาไฟล์หรือเอกสาร (Search file)
ส่วนที่ 6	การสนทนา (Chat)
ส่วนที่ 7	ฐานข้อมูลยูสเซอร์ (User Database)

3.4.1 ล็อกอิน (Login) / ล็อกออฟ (Logoff)

ส่วนของการล็อกอิน ของไคลเอนต์จะเป็นการส่งยูสเซอร์เนมไปตรวจสอบกับข้อมูลที่มีอยู่ที่เซิร์ฟเวอร์หลังจากนั้นเซิร์ฟเวอร์จะทำการสุ่มตัวเลขไบนารีขนาด 64 บิตมาเข้ารหัสกับพาสเวิร์ดเพื่อตรวจสอบว่าไคลเอนต์ที่ติดต่อเข้ามานั้นเป็นยูสเซอร์จริงหรือไม่แล้วส่งตัวเลขที่ทำการเข้ารหัสแล้วกลับไปยังไคลเอนต์ จากนั้นไคลเอนต์จะทำการถอดรหัสตัวเลขที่ได้แล้วส่งกลับไปยังเซิร์ฟเวอร์ สามารถแสดงการทำงานในส่วนของการล็อกอิน ได้ดังนี้

หลักการทำงานของโปรโตคอลในส่วนของการการล็อกอิน

```
ไคลเอนต์ --> เซิร์ฟเวอร์      : cmdLoin ยูสเซอร์เนม
เซิร์ฟเวอร์ --> ไคลเอนต์      : psEncode พาสเวิร์ดที่ถูกเข้ารหัสกับ Randomly
                               Generated
ไคลเอนต์ --> เซิร์ฟเวอร์      : psDecode Randomly Generated
เซิร์ฟเวอร์ --> ไคลเอนต์      : pass1111 (ถ้าพาสเวิร์ดถูก)
ไคลเอนต์ --> เซิร์ฟเวอร์      : UserName UserName
เซิร์ฟเวอร์ --> ไคลเอนต์      : cmdLogin 1 | เลเวล | สถานะ | ไอดี
เซิร์ฟเวอร์ --> ไคลเอนต์      : pass2222 (ถ้าพาสเวิร์ดผิด)
```

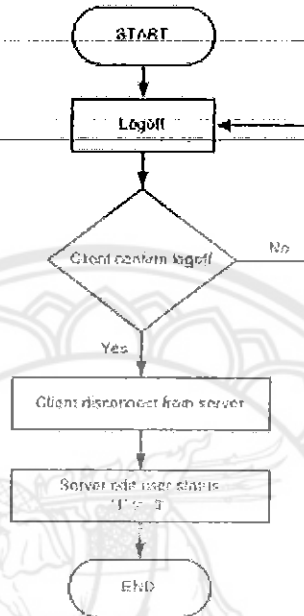


รูปที่ 3.1 แผนภาพแสดงการทำงานของการทำงานของกรล็อกอิน

ส่วนการล็อกออกจะเป็นการยกเลิกการติดต่อกับเซิร์ฟเวอร์สามารถแสดงการทำงานของกรล็อกออกได้ดังนี้

หลักการทำงานของโปรโตคอลในส่วนของการลือกออฟ

ไคลเอนต์ --> เซิร์ฟเวอร์	:	close connection
เซิร์ฟเวอร์ --> ไคลเอนต์	:	chatConsชื่อผู้ใช้ ชื่อผู้ใช้
เซิร์ฟเวอร์ --> ไคลเอนต์	:	chatMsgxชื่อผู้ใช้ has ลือกออฟ



รูปที่ 3.2 แผนภาพแสดงการทำงานของการทำงานของการลือกออฟ

3.4.2 ข้อความ (message)

ในส่วนนี้เป็นส่วนของโปรแกรมที่ใช้ในการส่งข้อความระหว่างกันของไคลเอนต์โดยจะแบ่งการทำงานออกเป็น ส่วนของการรับ – ส่งข้อความดังนี้

ส่วนของการรับข้อความ

- แสดงจำนวนข้อความที่ถูกส่งเข้ามาใหม่
- แสดงข้อความและจำนวนข้อความทั้งหมด
- แสดงรายละเอียดของข้อความ (ชื่อผู้ส่ง, หัวข้อ, เนื้อหาของข้อความ, วันเวลาส่ง)
- จะทำการแจ้งเตือนเมื่อได้รับข้อความใหม่

สามารถแสดงการทำงานในส่วนของข้อความได้ดังนี้

หลักการทำงานของโปรโตคอลในส่วนของการร้องขอข้อความทั้งหมด

ไคลเอนต์ --> เซิร์ฟเวอร์	:	cmdMsgID ไอดีของผู้ใช้
เซิร์ฟเวอร์ --> ไคลเอนต์	:	cmdMListสถานะ วันที่ เวลา ชื่อไฟล์ ข้อความ ชื่อหัวเรื่อง จำนวนข้อความ

หลักการทํางานของโปรโตคอลในส่วนของการอ่านข้อความ

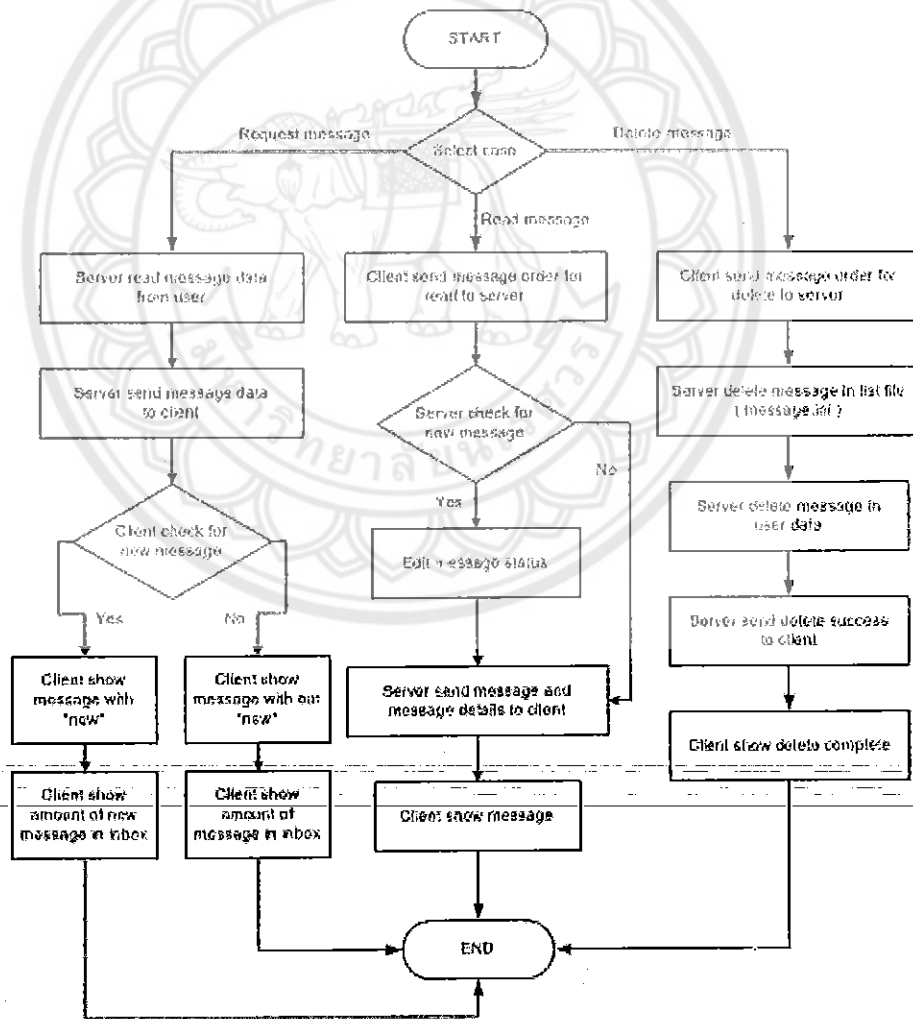
- ไคลเอนต์ --> เซิร์ฟเวอร์ : cmdMDesc ไอดี \ ชื่อไฟล์
- เซิร์ฟเวอร์ --> ไคลเอนต์ : cmdMOpn1 ผู้ส่ง | หัวเรื่อง | วัน | เวลา
- ~~ไคลเอนต์ --> เซิร์ฟเวอร์ : cmdMFile ไอดี \ ชื่อไฟล์~~
- เซิร์ฟเวอร์ --> ไคลเอนต์ : cmdMOpn2 ข้อความ

หลักการทํางานของโปรโตคอลในส่วนของการแก้ไขสถานะ

- ไคลเอนต์ --> เซิร์ฟเวอร์ : cmdMRead ไอดี | อันดับของข้อความ

หลักการทํางานของโปรโตคอลในส่วนของการลบข้อความ

- ไคลเอนต์ --> เซิร์ฟเวอร์ : cmdMDele ไอดี \ ชื่อไฟล์
- เซิร์ฟเวอร์ --> ไคลเอนต์ : cmdMDeleSuccess



รูปที่ 3.3 แผนภาพแสดงการทํางานของข้อความ

ส่วนของการส่งข้อความ

- แสดงรายชื่อของผู้ที่สามารถส่งข้อความไปให้ได้
- โปรแกรมจะทำการลงชื่อผู้ส่งและวันเวลาส่งให้โดยอัตโนมัติ
- สามารถส่งข้อความได้มากกว่า 1 คนพร้อมๆกัน

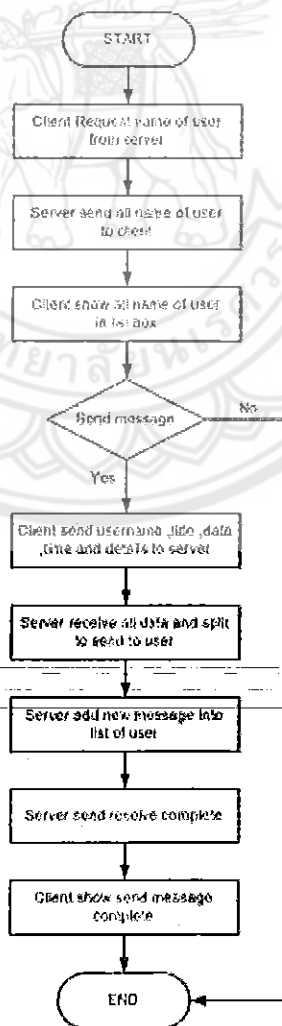
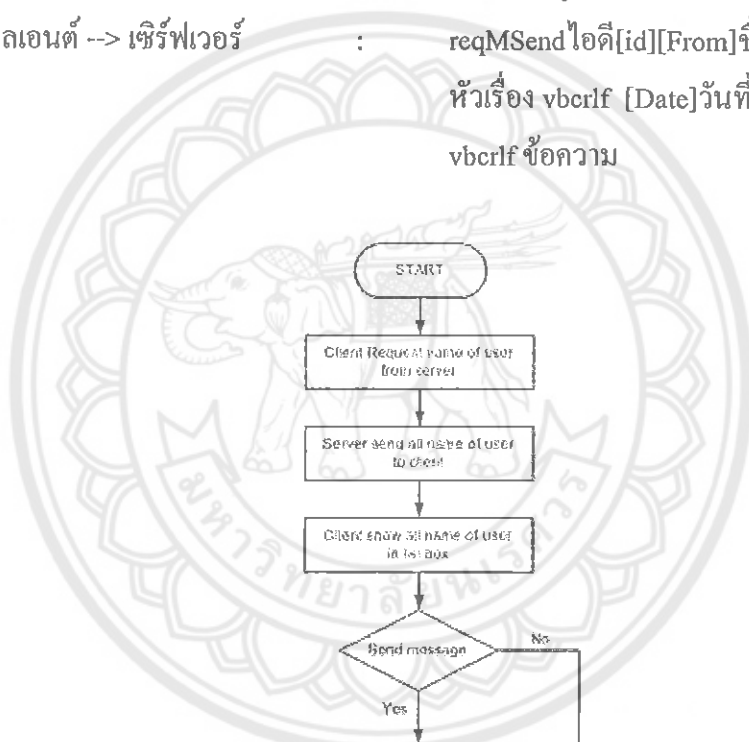
สามารถแสดงการทำงานในส่วนของการส่งข้อความ ได้ดังนี้

หลักการการทำงานของโปรโตคอลในส่วนของการส่งข้อความ

ไคลเอนต์ --> เซิร์ฟเวอร์ : reqLName

เซิร์ฟเวอร์ --> ไคลเอนต์ : reqLNameจำนวนผู้ใช้ทั้งหมด# ไอดี / ชื่อผู้ใช้ | ไอดี / ชื่อผู้ใช้ |

ไคลเอนต์ --> เซิร์ฟเวอร์ : reqMSend ไอดี[id][From]ชื่อผู้ส่ง vbCrLf [Title] หัวเรื่อง vbCrLf [Date]วันที่ vbCrLf [Time]เวลา vbCrLf ข้อความ



รูปที่ 3.4 แผนภาพแสดงการทำงานของการทำงานของการส่งข้อความ

3.4.3 อัปโหลด (Upload) / ดาวน์โหลด (Download)

ภายในส่วนของ อัปโหลด / ดาวน์โหลด นั้นจะเป็นที่ที่ผู้สเซอร์สามารถที่จะ ดาวน์โหลด ไฟล์หรือเอกสาร จากเครื่องเซิร์ฟเวอร์ได้ซึ่งจะสามารถดาวน์โหลด อะไรได้บ้างนั้นจะขึ้นอยู่กับ รัศมีของผู้สเซอร์แต่ละคน นอกจากนี้ยังสามารถที่จะอัปโหลดไฟล์หรือเอกสาร ไปยังเซิร์ฟเวอร์ ได้อีกด้วยสามารถแสดงการทำงานของอัปโหลด / ดาวน์โหลด ได้ดังนี้

หลักการการทำงานของโปรโตคอลในส่วนของการแสดงรายชื่อไฟล์

ไคลเอนต์ --> เซิร์ฟเวอร์	:	fileListเลขของไฟล์เคอร์
เซิร์ฟเวอร์ --> ไคลเอนต์	:	fileNameจำนวนไฟล์ ชื่อไฟล์ ชื่อไฟล์

หลักการการทำงานของโปรโตคอลในส่วนของการดาวน์โหลดไฟล์

ไคลเอนต์ --> เซิร์ฟเวอร์	:	fileDLoaเลขของไฟล์เคอร์ ชื่อไฟล์
เซิร์ฟเวอร์ --> ไคลเอนต์	:	fileRequชื่อไฟล์ ขนาด
ไคลเอนต์ --> เซิร์ฟเวอร์	:	fileAcptชื่อไฟล์
ไคลเอนต์ --> เซิร์ฟเวอร์	:	fileCanc
เซิร์ฟเวอร์ --> ไคลเอนต์	:	fileContไฟล์
ไคลเอนต์ --> เซิร์ฟเวอร์	:	fileAcpt
เซิร์ฟเวอร์ --> ไคลเอนต์	:	fileStop

หลักการการทำงานของโปรโตคอลในส่วนของการอัปโหลดไฟล์

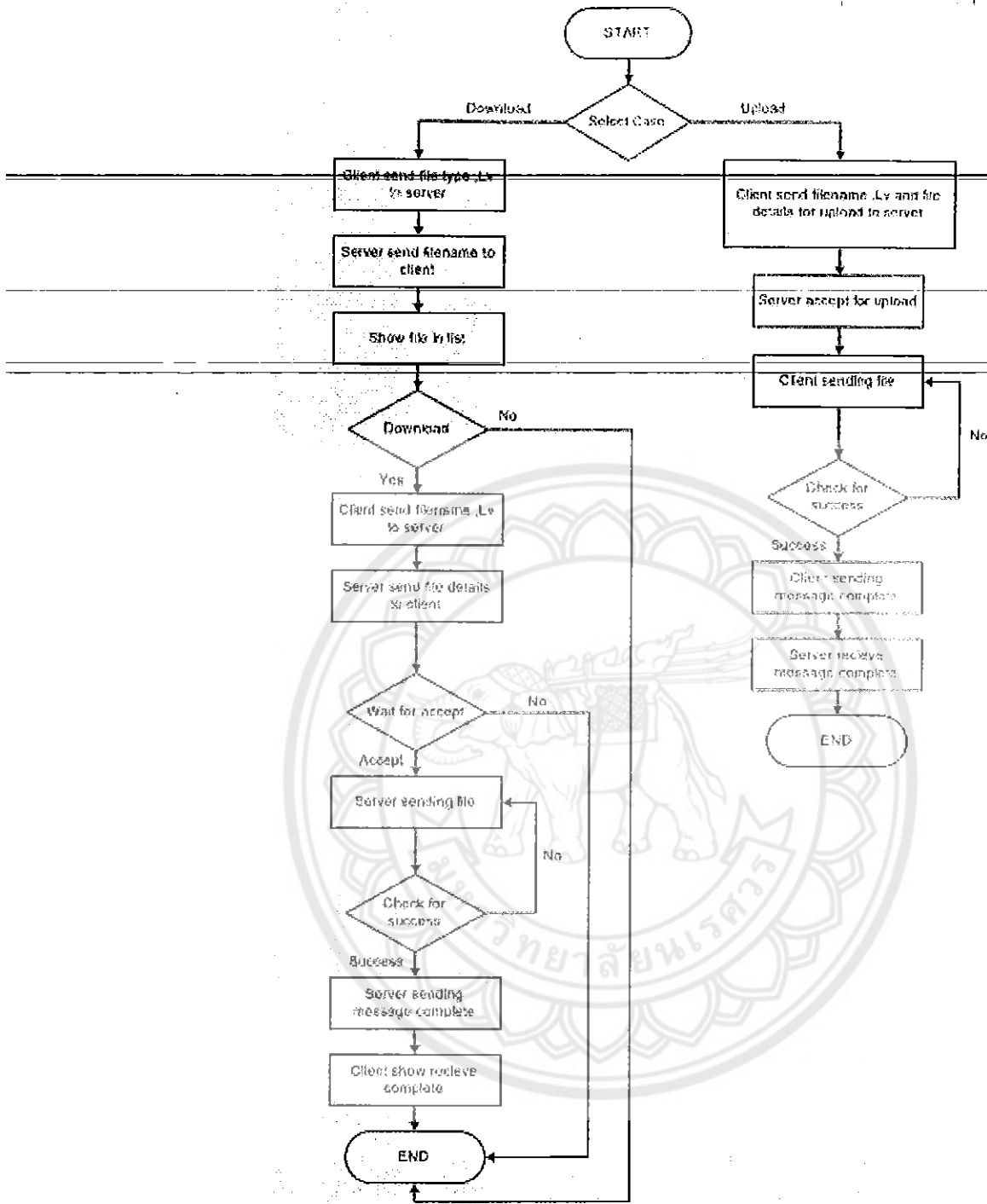
ไคลเอนต์ --> เซิร์ฟเวอร์	:	fileULoaเลขของไฟล์เคอร์ ชื่อไฟล์
เซิร์ฟเวอร์ --> ไคลเอนต์	:	fileAcpt
ไคลเอนต์ --> เซิร์ฟเวอร์	:	fileContไฟล์
เซิร์ฟเวอร์ --> ไคลเอนต์	:	fileAcpt
ไคลเอนต์ --> เซิร์ฟเวอร์	:	fileStop

หลักการการทำงานของโปรโตคอลในส่วนของการลบไฟล์

ไคลเอนต์ --> เซิร์ฟเวอร์	:	fileDeleเลขของไฟล์เคอร์ ชื่อไฟล์
เซิร์ฟเวอร์ --> ไคลเอนต์	:	fileDeIS

4941994

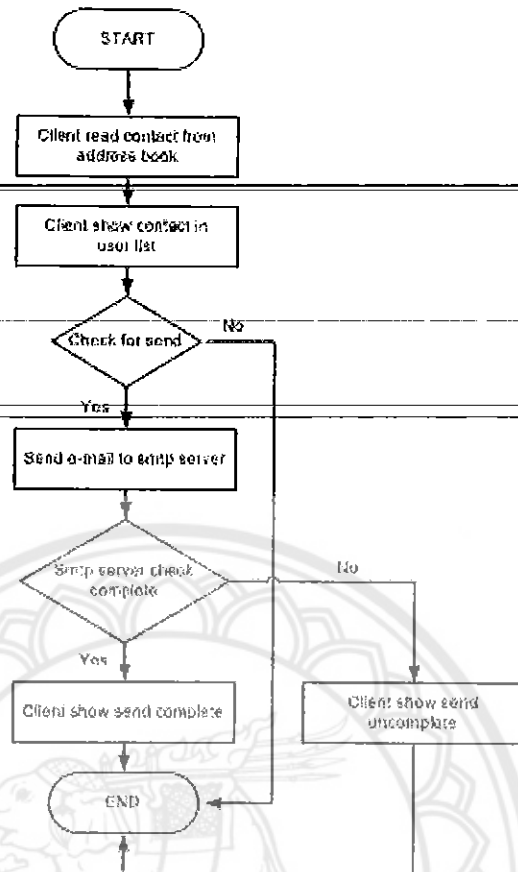
ร/ส.
064891
2547



รูปที่ 3.5 แผนภาพแสดงการทำงานของการทำงานของการอัปโหลด / ดาวน์โหลด

3.4.4 ส่งอีเมล (Send e-mail) / สมุดรายชื่อ (Address Book)

ภายในส่วนของการส่งอีเมลนั้นจะเป็นที่ที่ยูสเซอร์สามารถที่จะส่งอีเมลไปยังผู้รับที่อยู่ภายนอกได้ สามารถแสดงการทำงานของการทำงานของการส่งอีเมลได้ดังนี้

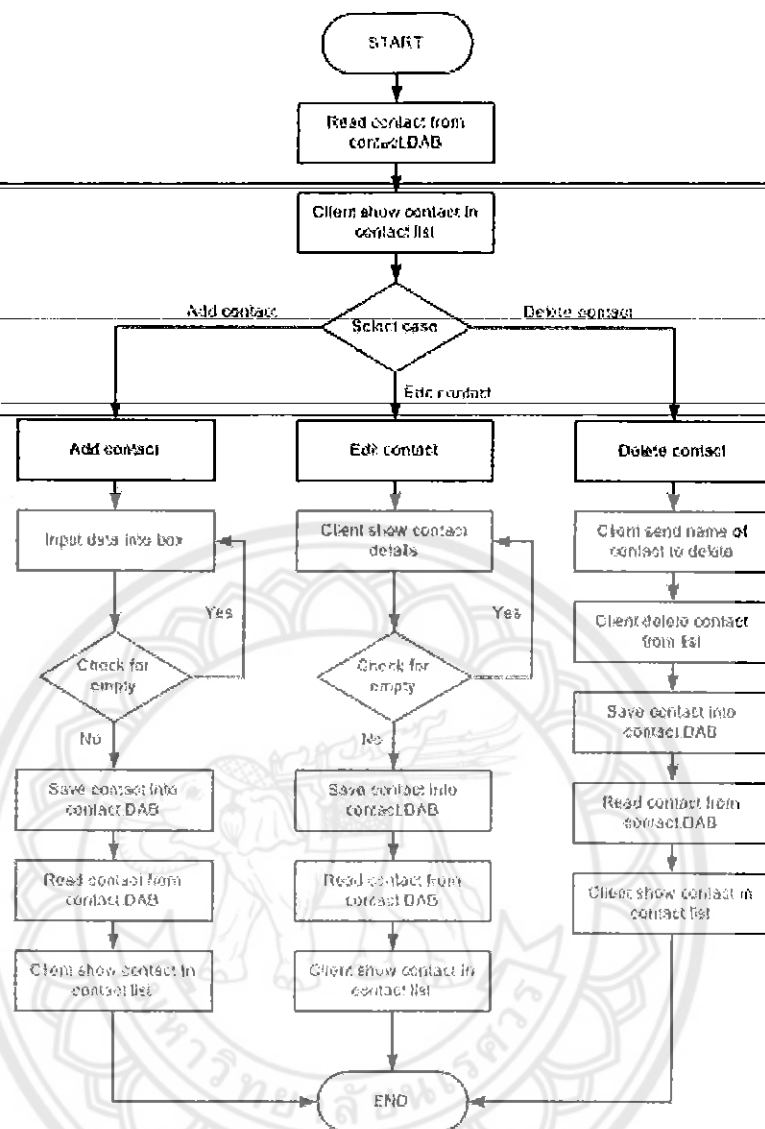


รูปที่ 3.6 แผนภาพแสดงการทำงานของการทำงานของการส่งอีเมลล์

ส่วนของสมุดรายชื่อ

- ยูสเซอร์สามารถที่จะเพิ่มรายชื่อในส่วนของ สมุดรายชื่อ ได้
- ยูสเซอร์สามารถที่จะลบรายชื่อในส่วนของ สมุดรายชื่อ ได้
- ยูสเซอร์สามารถที่จะแก้ไขรายชื่อในส่วนของ สมุดรายชื่อ ได้
- ยูสเซอร์สามารถ Export และ Import สมุดรายชื่อ file ได้

สามารถแสดงการทำงานของ Addresss Book ได้ดังนี้



รูปที่ 3.7 แผนภาพแสดงการทำงานของสมุดรายชื่อ

3.4.5 ค้นหาไฟล์หรือเอกสาร (Search file)

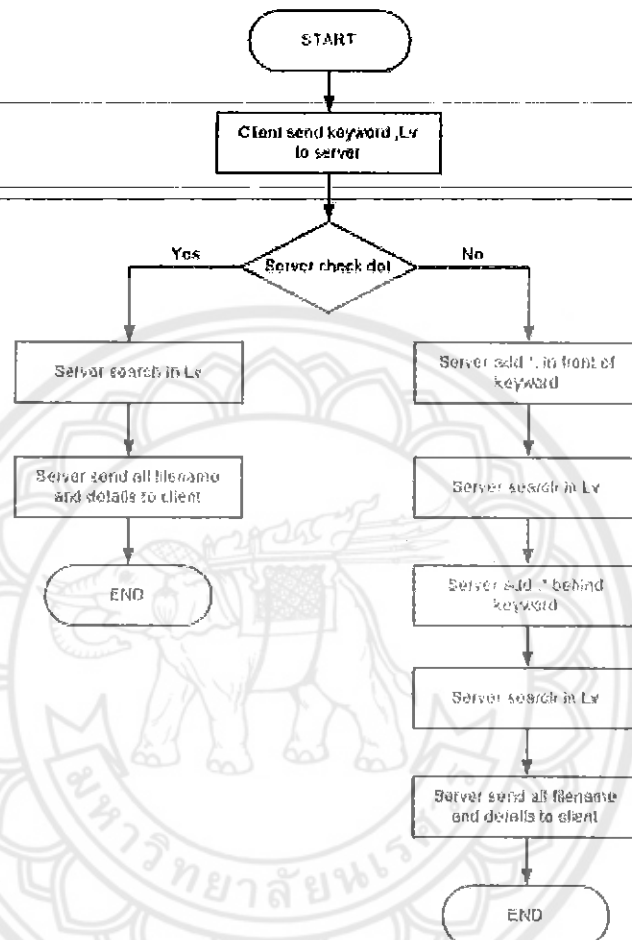
ภายในส่วนของการค้นหานี้ จะเป็นการค้นหาไฟล์หรือเอกสารต่างๆจากเซิร์ฟเวอร์โดย การหาจาก Keyword ภายในชื่อของ ไฟล์ที่ต้องการค้นหาแล้วแสดงออกมาและเมื่อค้นหาได้แล้วก็สามารถที่จะดาวน์โหลด ไฟล์หรือเอกสารนั้นมาจากเซิร์ฟเวอร์ได้ทันที สามารถแสดงการทำงานของ การค้นหาไฟล์ได้ดังนี้

หลักการการทำงานของโปรโตคอลในส่วนของการค้นหาไฟล์

ไคลเอนต์ --> เซิร์ฟเวอร์ : SearFileกำหนดการค้นหาระดับของเลเวล

เซิร์ฟเวอร์ --> ไคลเอนต์ : SearFile จำนวนไฟล์ | ชื่อไฟล์ เกล็ดไฟล์เดอร์ |
ชื่อไฟล์ เกล็ดไฟล์เดอร์ |

เซิร์ฟเวอร์ --> ไคลเอนต์ : SearFileFinish



รูปที่ 3.8 แผนภาพแสดงการทำงานของการทำงานของการค้นหาไฟล์

3.4.6 การสนทนา (Chat)

การสนทนาเป็นห้องสนทนาสามารถคุยกันได้ทั้งหมด ภายในส่วนนี้จะประกอบไปด้วยที่สำหรับพิมพ์ข้อความ, ที่สำหรับแสดงบทสนทนาและส่วนที่แสดงรายชื่อของผู้ที่อยู่ในห้องสนทนา และสามารถรองรับการสนทนาได้ตามที่ตั้งค่าไว้ที่เซิร์ฟเวอร์ สามารถแสดงการทำงานของการทำงานของการสนทนาได้ดังนี้

หลักการการทำงานของโปรโตคอลในส่วนของการล็อกอินเข้าห้องสนทนา

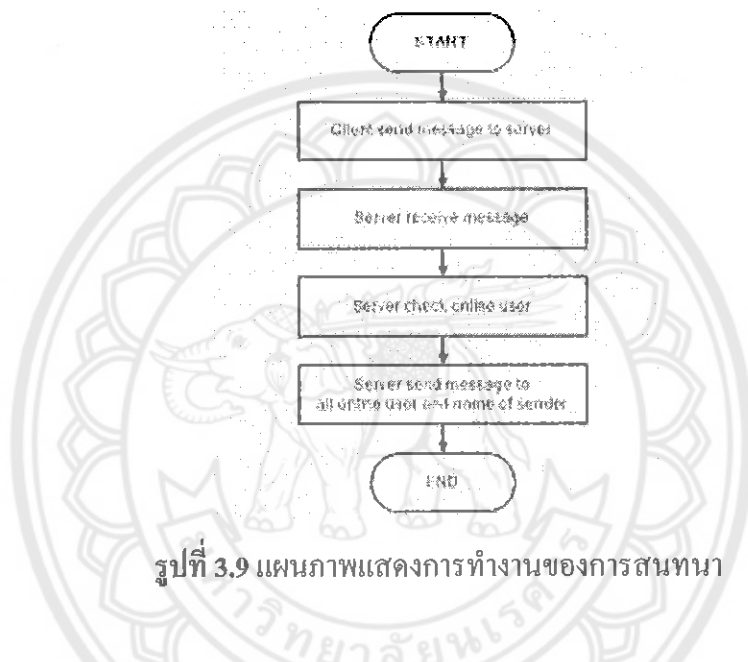
ไคลเอนต์ --> เซิร์ฟเวอร์ : chatNameชื่อผู้ใช้
เซิร์ฟเวอร์ --> ไคลเอนต์ : chatConsชื่อผู้ใช้ | ชื่อผู้ใช้
เซิร์ฟเวอร์ --> ไคลเอนต์ : chatMsgxชื่อผู้ใช้ใหม่ | has ล็อกอิน

หลักการทำงานของโปรโตคอลในส่วนของการสนทนา

ไคลเอนต์ --> เซิร์ฟเวอร์ : chatTexข้อความ
 เซิร์ฟเวอร์ --> ไคลเอนต์ : chatTex(ชื่อผู้ส่งข้อความ | ข้อความ)

หลักการทำงานของโปรโตคอลในส่วนของการล๊อคออฟออกจากห้องสนทนา

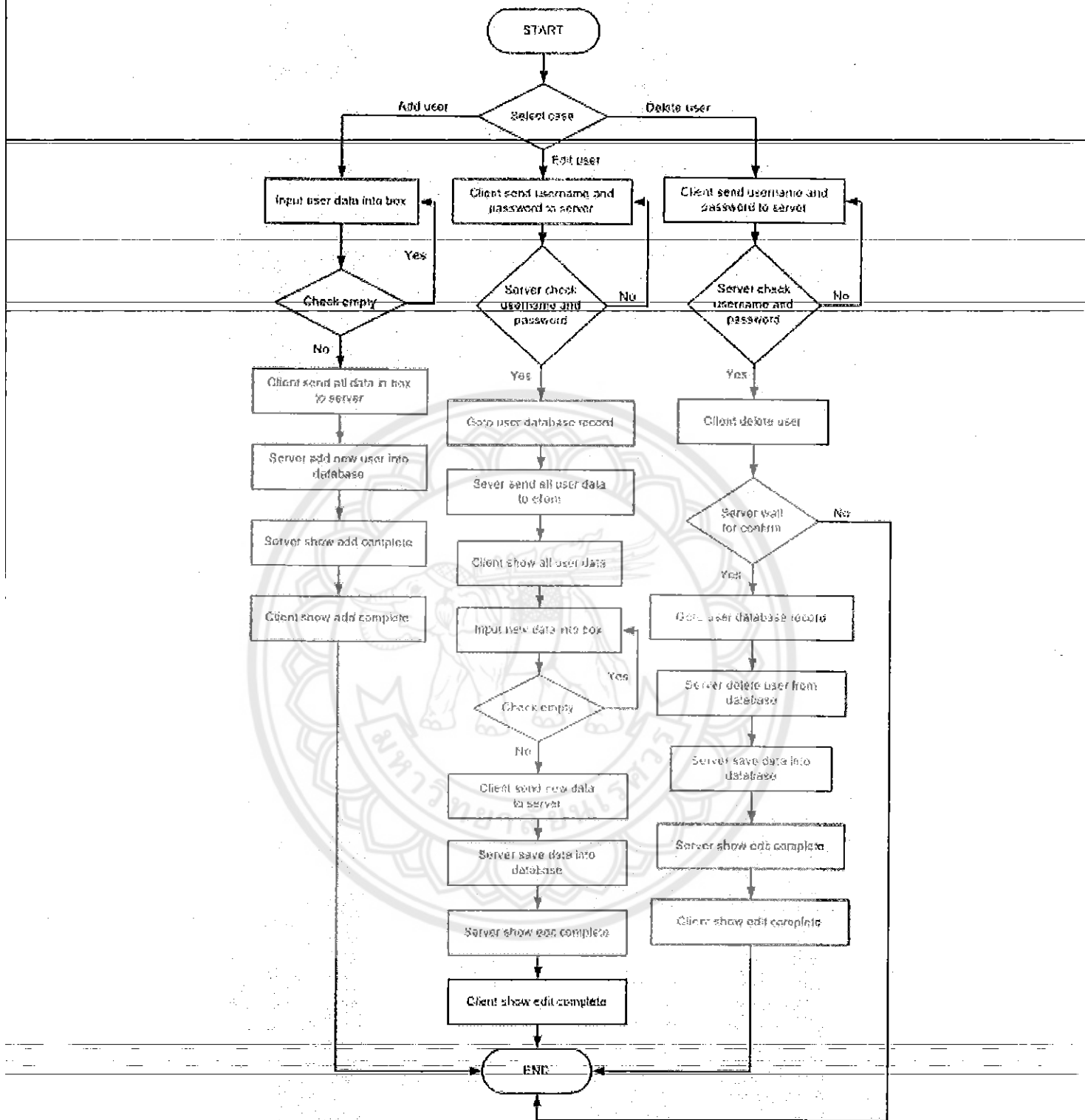
ไคลเอนต์ --> เซิร์ฟเวอร์ : close connection
 เซิร์ฟเวอร์ --> ไคลเอนต์ : chatCons(ชื่อผู้ใช้ | ชื่อผู้ใช้)
 เซิร์ฟเวอร์ --> ไคลเอนต์ : chatMsgx(ชื่อผู้ใช้ | has ล๊อคออฟ)



รูปที่ 3.9 แผนภาพแสดงการทำงานของการทำงานของการสนทนา

3.4.7 การจัดการฐานข้อมูลยูสเซอร์ (User Database)

ภายในส่วนนี้จะเป็นส่วนเฉพาะของผู้ดูแลระบบเท่านั้นเป็นส่วนที่ใช้สำหรับการเพิ่ม, ลบ และ แก้ไขข้อมูลต่าง ๆ ของยูสเซอร์ สามารถแสดงการทำงาน ได้ดังนี้



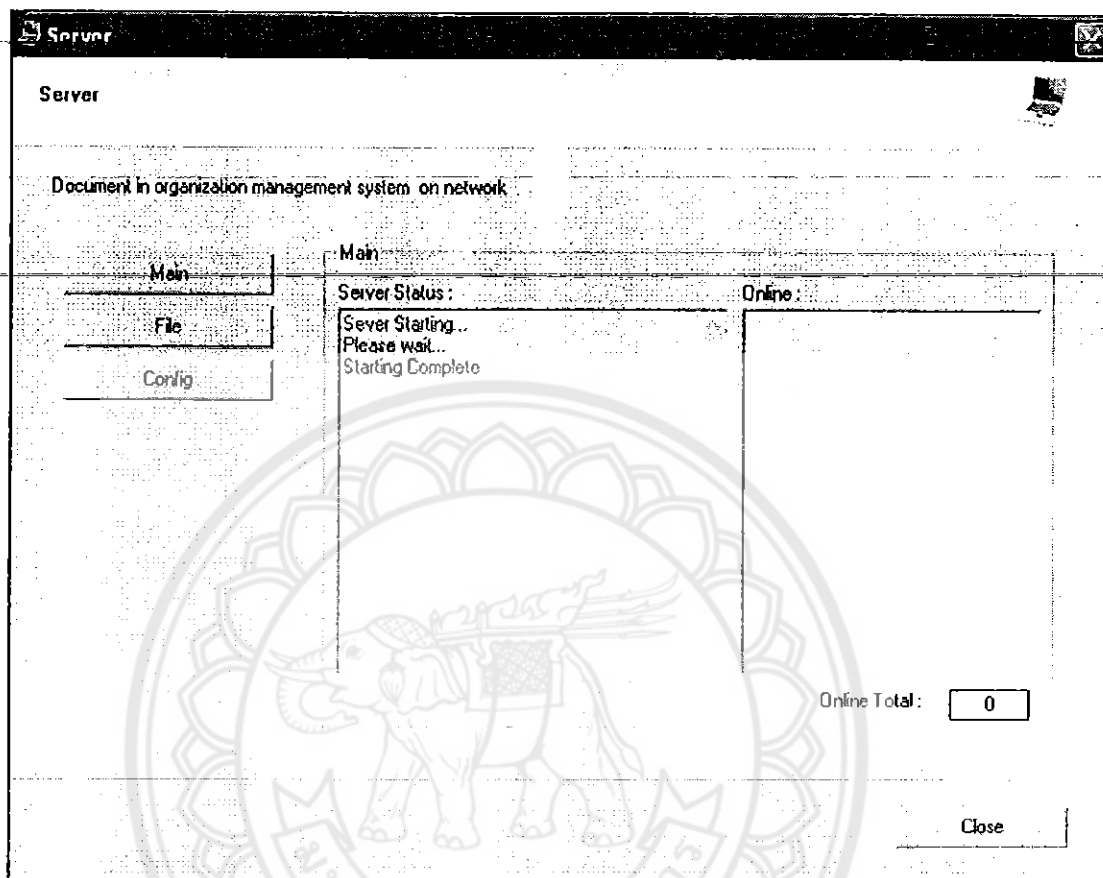
รูปที่ 3.10 แผนภาพแสดงการทำงานของฐานข้อมูลยูสเซอร์

3.5 การออกแบบในส่วนของอินเตอร์เฟซที่ติดต่อกับผู้ใช้ในส่วนต่างๆ ของโปรแกรม

สำหรับการออกแบบในส่วนของอินเตอร์เฟซที่ติดต่อกับผู้ใช้แบ่งออกเป็นส่วนของ เซิร์ฟเวอร์ และ ไคลเอนต์ สามารถแสดงผลของการออกแบบอินเตอร์เฟซที่ติดต่อกับผู้ใช้ ได้ดังนี้

3.5.1 ส่วนของเซิร์ฟเวอร์

ผลของการออกแบบอินเทอร์เน็ตเฟสที่ติดต่อกับผู้ใช้ในส่วนของเซิร์ฟเวอร์ แสดงได้ดังรูป 3.11

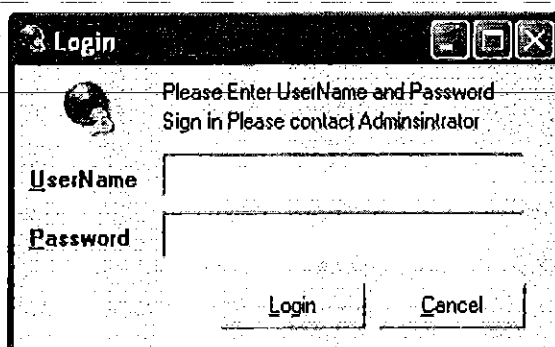


รูปที่ 3.11 อินเทอร์เน็ตของเซิร์ฟเวอร์

3.5.2 ส่วนของไคลเอนต์

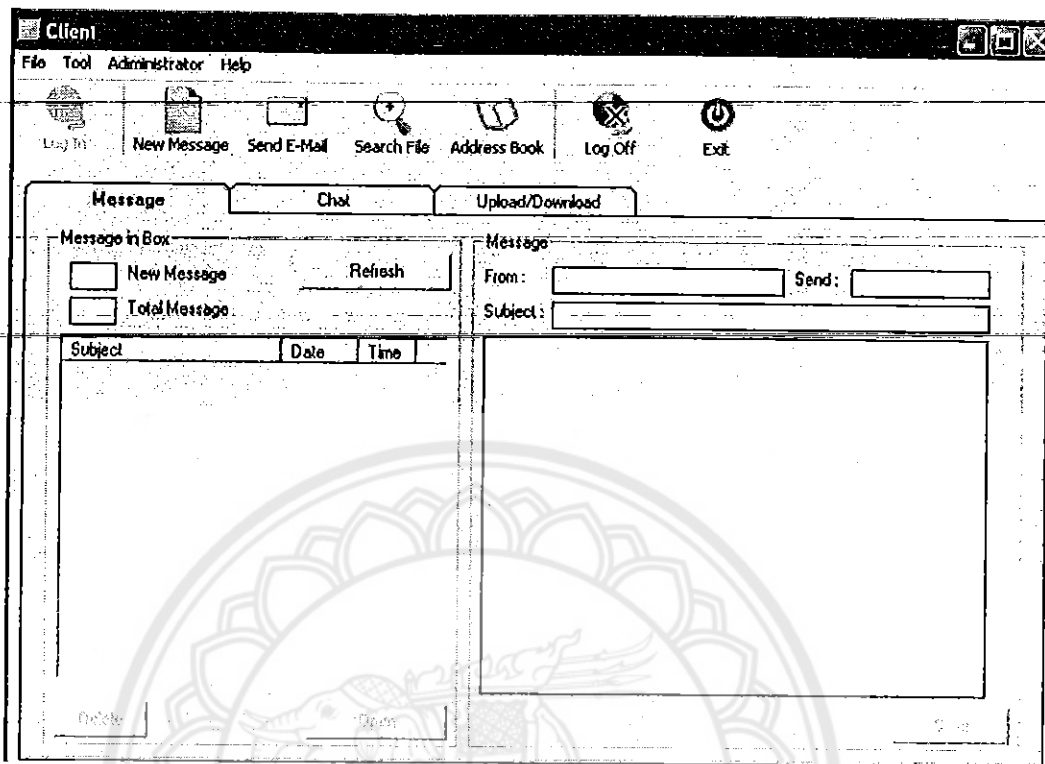
ภายในส่วนของไคลเอนต์ นี้จะแบ่งออกเป็นส่วนต่าง ๆ แสดงผลดังนี้

ผลของการออกแบบอินเทอร์เน็ตเฟสที่ติดต่อกับผู้ใช้ในส่วนของไคลเอนต์ แสดงได้ดังรูป 3.12



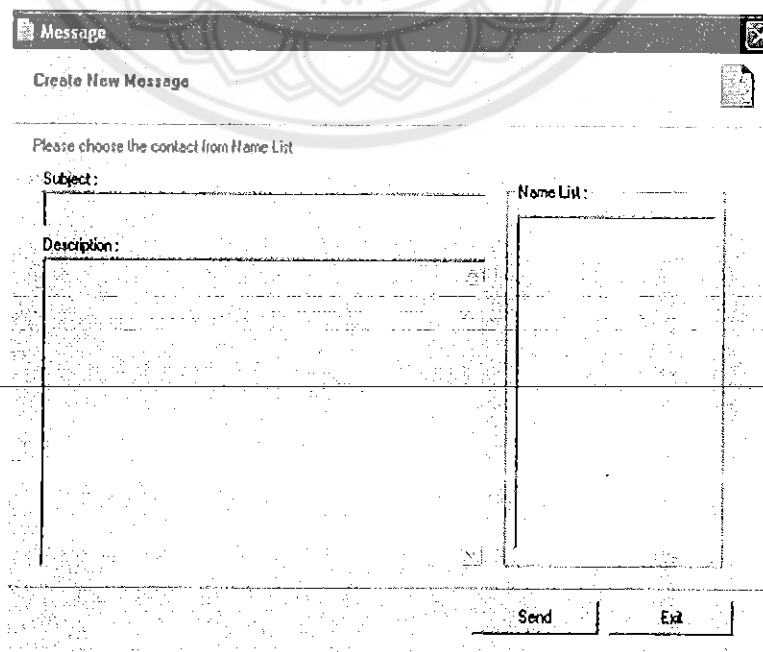
รูปที่ 3.12 อินเทอร์เน็ตของไคลเอนต์

ผลของการออกแบบอินเตอร์เฟซที่ติดต่อกับผู้ใช้ในส่วนของการข้อความ แสดงได้ดังรูป 3.13



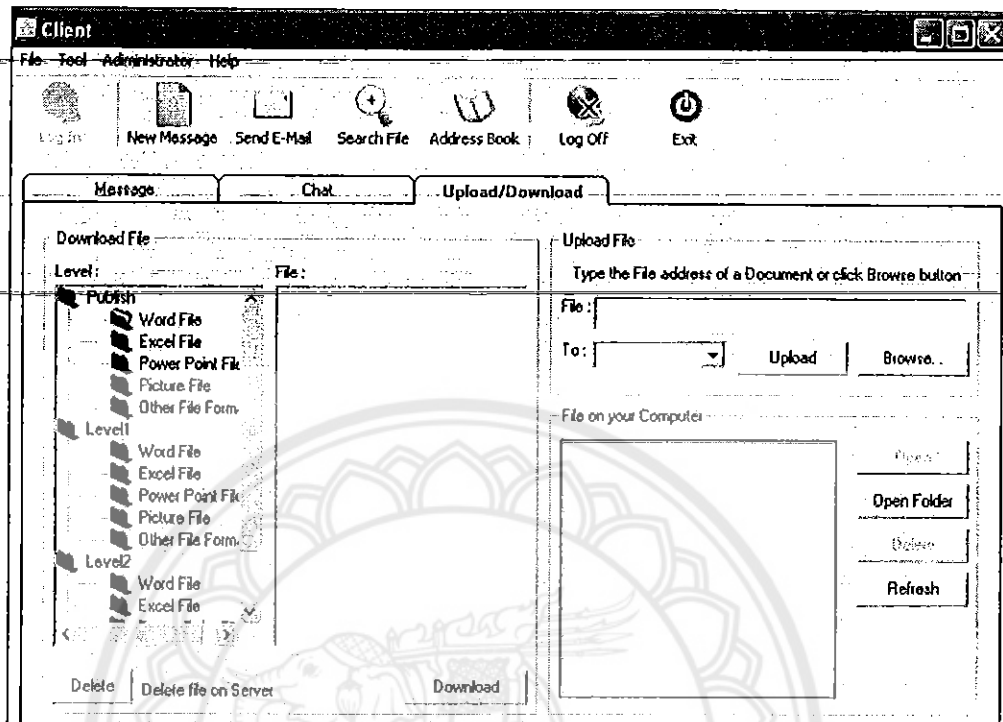
รูปที่ 3.13 อินเตอร์เฟซของข้อความ

ผลของการออกแบบอินเตอร์เฟซที่ติดต่อกับผู้ใช้ในส่วนของการข้อความใหม่ แสดงได้ดังรูป 3.14



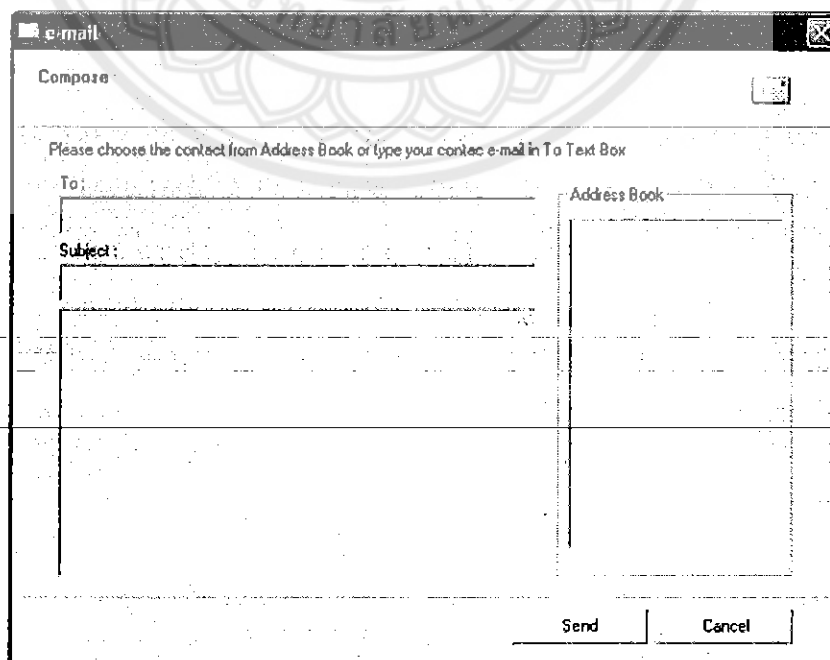
รูปที่ 3.14 อินเตอร์เฟซของข้อความใหม่

ผลของการออกแบบอินเตอร์เฟซที่ติดต่อกับผู้ใช้ในส่วนของอัปโหลด / ดาวน์โหลด แสดง
ได้ดังรูป 3.15



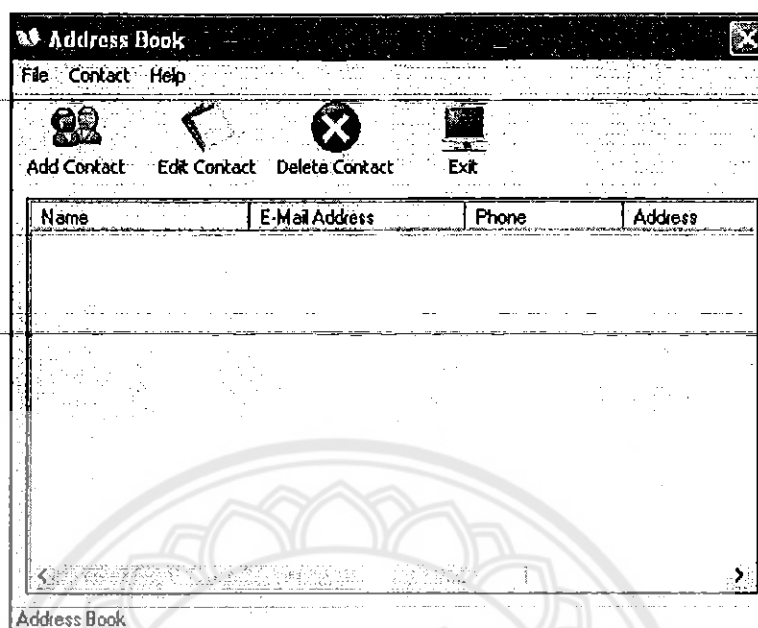
รูปที่ 3.15 อินเตอร์เฟซของอัปโหลด / ดาวน์โหลด

ผลของการออกแบบอินเตอร์เฟซที่ติดต่อกับผู้ใช้ในส่วนของส่งอีเมล แสดงได้ดังรูป 3.16



รูปที่ 3.16 อินเตอร์เฟซของการส่งอีเมล

ผลของการออกแบบอินเตอร์เฟซที่ติดต่อกับผู้ใช้ในส่วนของสมุดรายชื่อ แสดงได้ดังรูป 3.17



รูปที่ 3.17 อินเตอร์เฟซของสมุดรายชื่อ

ผลของการออกแบบอินเตอร์เฟซที่ติดต่อกับผู้ใช้ในส่วนของสมุดรายชื่อ (เพิ่มรายชื่อ) แสดงได้ดังรูป 3.18

รูปที่ 3.18 อินเตอร์เฟซของการเพิ่มรายชื่อ

ผลของการออกแบบอินเตอร์เฟซที่ติดต่อกับผู้ใช้ในส่วนของสมุดรายชื่อ (แก้ไขรายชื่อ)
แสดงได้ดังรูป 3.19

Eulii Contact

Edit name and e-mail address about contact

Name:

E-Mail Address:

Edit home address and phone about contact

Address:

Phone:

OK Cancel

รูปที่ 3.19 อินเตอร์เฟซของการแก้ไขรายชื่อ

ผลของการออกแบบอินเตอร์เฟซที่ติดต่อกับผู้ใช้ในส่วนของการสนทนา แสดงได้ดังรูป 3.20

Eulii Client

File Tool Address Book Help

Log On New Message Send E-Mail Search File Address Book Log Off Exit

Message Chat Upload/Download

Send

Online

รูปที่ 3.20 อินเตอร์เฟซของการสนทนา

ผลของการออกแบบอินเตอร์เฟซที่ติดต่อกับผู้ใช้ในส่วนของการจัดการฐานข้อมูลยูสเซอร์ (เพิ่มยูสเซอร์ใหม่) แสดงได้ดังรูป 3.21

The screenshot shows a window titled "User Database" with a sub-header "Administrator only". The main content area is titled "Create a new user: User information". It contains five text input fields: "Name:", "Address:", "Phone Number:", "E-Mail Address:", and "Work Section:". At the bottom, there are three buttons: "Menu" on the left, and "Next" and "Exit" on the right.

รูปที่ 3.21 อินเตอร์เฟซการจัดการฐานข้อมูลยูสเซอร์ (เพิ่มยูสเซอร์ใหม่)

ผลของการออกแบบอินเตอร์เฟซที่ติดต่อกับผู้ใช้ในส่วนของการจัดการฐานข้อมูลยูสเซอร์ (แก้ไขข้อมูลยูสเซอร์) แสดงได้ดังรูป 3.22

The screenshot shows a window titled "User Database" with a sub-header "Administrator only". The main content area is titled "Edit user information from database". It contains five text input fields: "Name:", "Address:", "Phone Number:", "E-Mail Address:", and "Work Section:". At the bottom, there are three buttons: "Menu" on the left, and "Next" and "Exit" on the right.

รูปที่ 3.22 อินเตอร์เฟซการจัดการฐานข้อมูลยูสเซอร์ (แก้ไขข้อมูลยูสเซอร์)

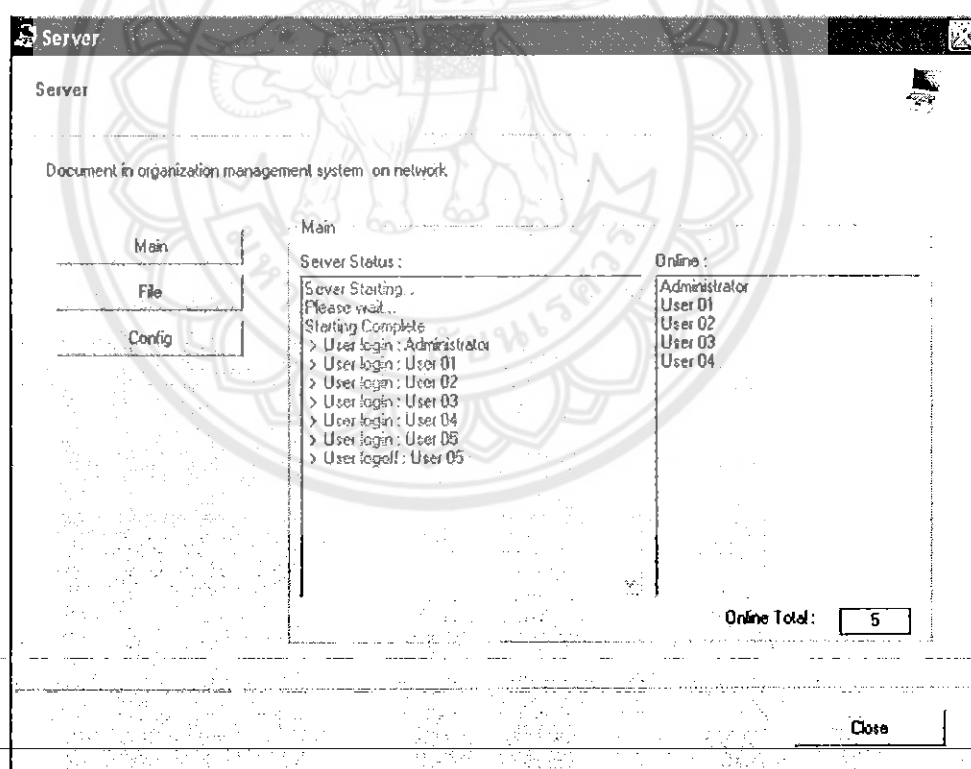
บทที่ 4

การทดสอบและวิเคราะห์การทำงาน

ในบทนี้จะกล่าวถึงการทดสอบและวิเคราะห์การใช้งานจริงของโปรแกรม เพื่อให้ทราบว่าโปรแกรมที่ได้พัฒนาขึ้นมาสามารถทำงานได้ตามที่ต้องการหรือไม่ โดยการทดสอบนั้นสามารถแบ่งการทดสอบโปรแกรมเป็น 2 ส่วนใหญ่ ๆ ได้ดังนี้

4.1 ส่วนของเซิร์ฟเวอร์ (Server)

ในส่วนนี้จะเป็นการรันเซิร์ฟเวอร์ โดยเซิร์ฟเวอร์จะเป็นส่วนให้บริการต่างๆแก่เครื่องที่เป็นไคลเอนต์ตามแต่ที่ไคลเอนต์ขอบริการมา และจะแสดงสถานะของยูสเซอร์ที่อยู่ภายในระบบทั้งหมดว่าขณะนั้นมีใครบ้างที่กำลังติดต่อกับระบบบ้าง แสดงผลได้ดังรูป 4.1 ซึ่งเป็นการรันเซิร์ฟเวอร์แล้วมียูสเซอร์อยู่ในระบบ



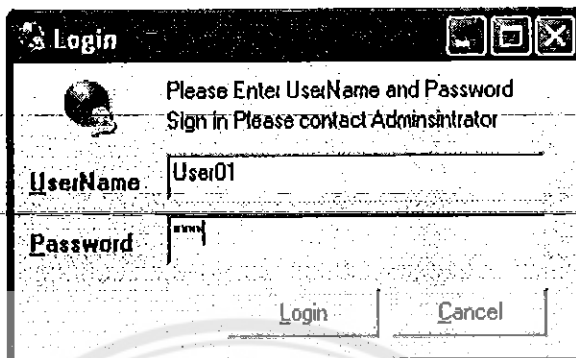
รูปที่ 4.1 ภาพแสดงเซิร์ฟเวอร์

4.2 ส่วนของไคลเอนต์ (Client)

ในส่วนนี้จะเป็นการรันไคลเอนต์โดยไคลเอนต์จะเป็นเครื่องคอมพิวเตอร์ที่จะร้องขอบริการ หรือ ข้อมูลจากเครื่องเซิร์ฟเวอร์ ซึ่งจะแบ่งการทดสอบเป็นส่วนต่าง ๆ ดังนี้

4.2.1 ล็อกอิน (Login)

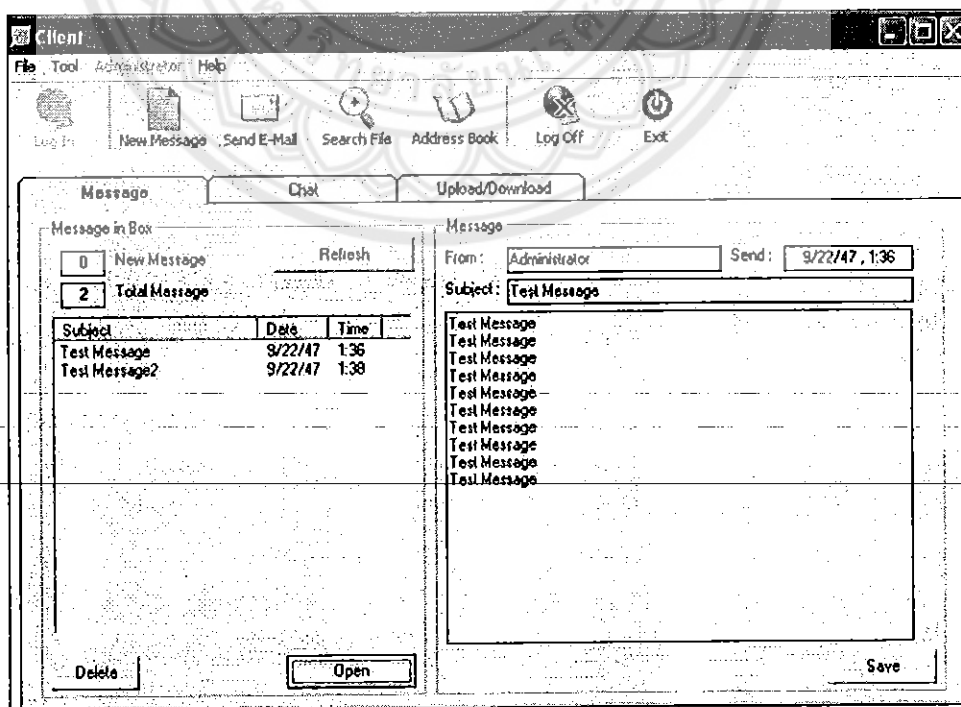
เมื่อสั่งรัน โคลเอนต์แล้วก็ต้องทำการล็อกอินเข้าระบบเสียก่อนและเมื่อกดที่ล็อกอินก็จะพบกับหน้าต่างล็อกอิน ดังรูป 4.2



รูปที่ 4.2 ภาพแสดงการล็อกอิน

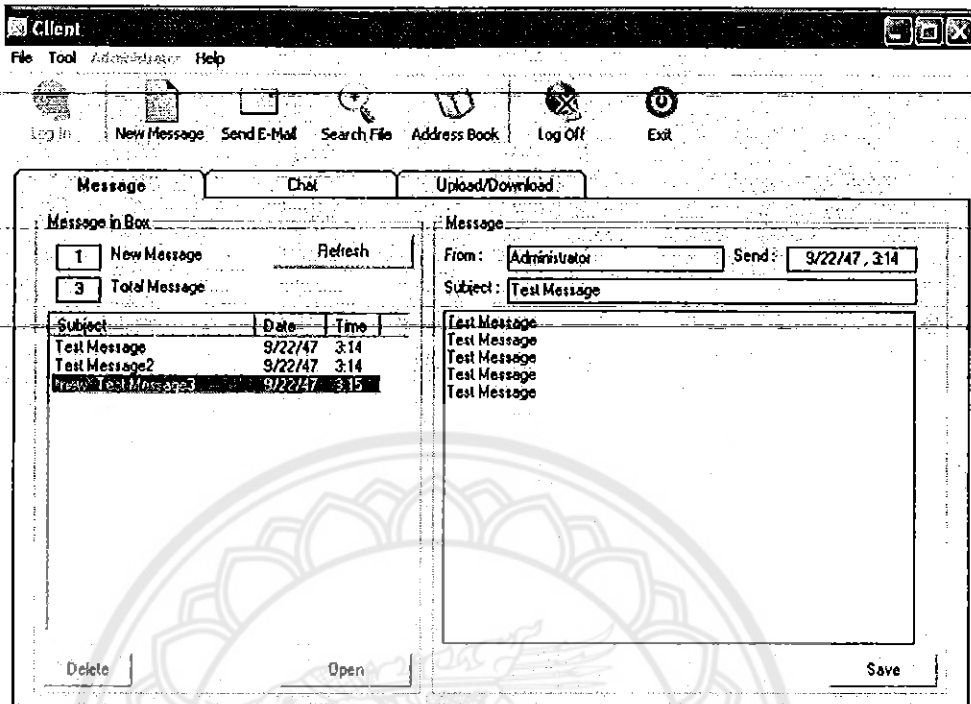
4.2.2 ข้อความ (Message)

ในส่วน of ข้อความนั้นเมื่อทำการล็อกอินเข้ามาแล้วหน้าต่างในส่วนนี้ก็จะแสดงว่ามีข้อความอยู่ใน In Box เท่าใดมีข้อความเข้ามาใหม่หรือไม่และเมื่อเลือกที่ข้อความแล้วเลือก Open ก็จะเป็นการเปิดข้อความนั้นอ่านแล้วก็จะมีส่วนที่แสดงข้อความของข้อความนั้นรวมทั้งแสดง ชื่อผู้ส่ง, หัวข้อ, วันและเวลาส่ง ดังรูป 4.3



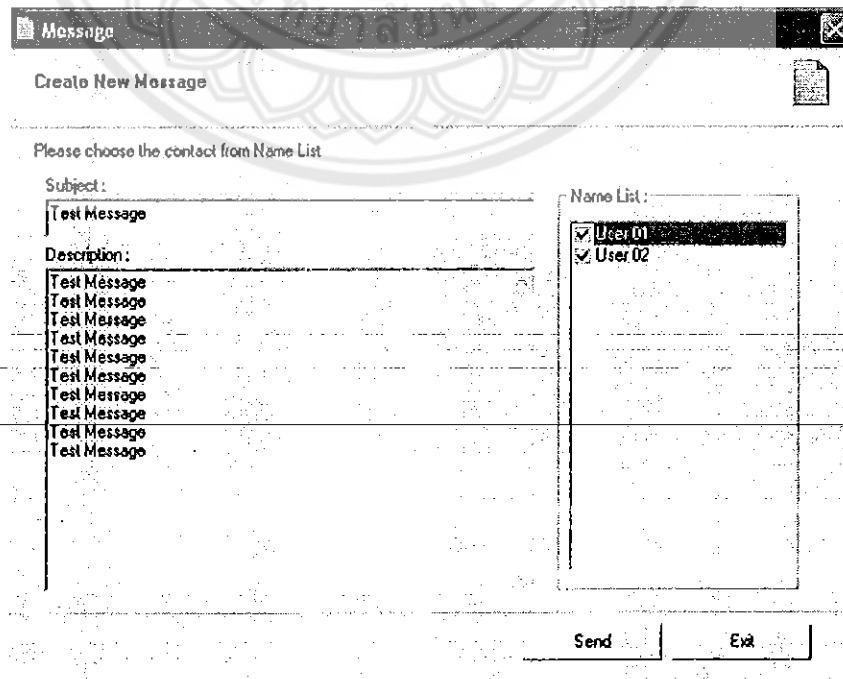
รูปที่ 4.3 ภาพแสดงข้อความ

และเมื่อมีข้อความใหม่เข้ามา ก็จะแสดงบอกในช่อง New Message ดังรูป 4.4



รูปที่ 4.4 ภาพแสดงข้อความใหม่

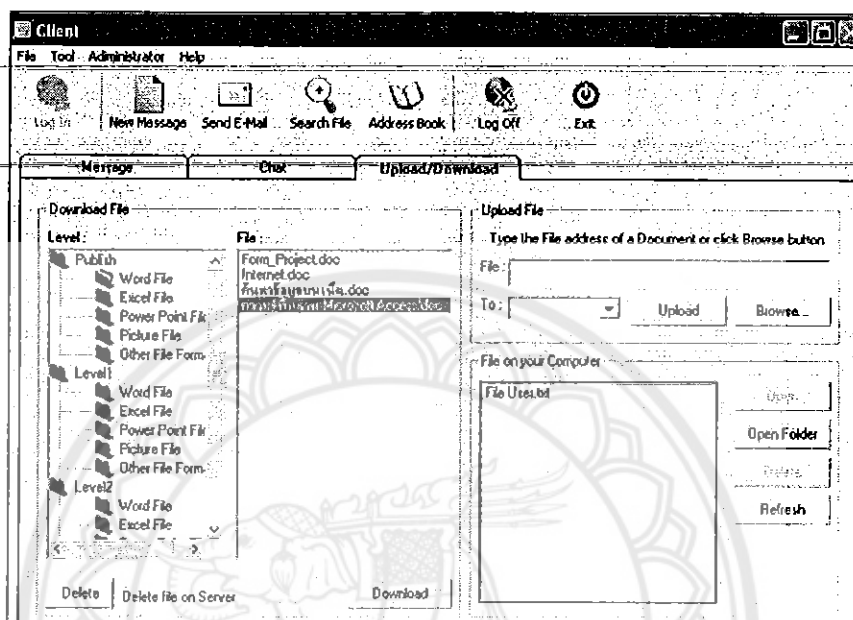
และหากต้องการที่จะส่งข้อความไปให้คนอื่น ๆ ก็จะสามารถเลือกได้ที่ Tool Bar -> New Message ก็จะมีส่วนที่ให้พิมพ์ หัวข้อ, ข้อความที่จะส่ง, และรายชื่อของผู้รับ



รูปที่ 4.5 ภาพแสดงการส่งข้อความ

4.2.3 อัปโหลด (Upload) / ดาวน์โหลด (Download)

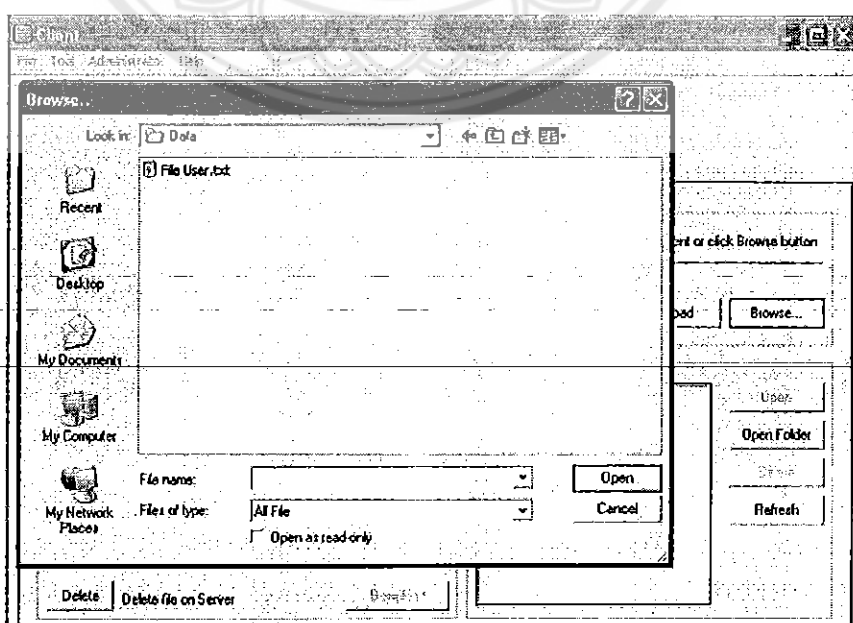
ในส่วนของ อัปโหลด / ดาวน์โหลด นั้นจะแสดงให้เห็นถึงว่ายูสเซอร์ที่ล็อกอินเข้ามานั้นสามารถเข้าถึงข้อมูลได้มากน้อยแค่ไหนนั้นขึ้นอยู่กับระดับของตนเอง โดยถ้าต้องการดาวน์โหลดไฟล์ก็ให้เลือกระดับแล้วเลือกไฟล์ที่ต้องการก็จะสามารถดาวน์โหลด ได้ ดังรูป 4.6



รูปที่ 4.6 ภาพแสดงการดาวน์โหลด

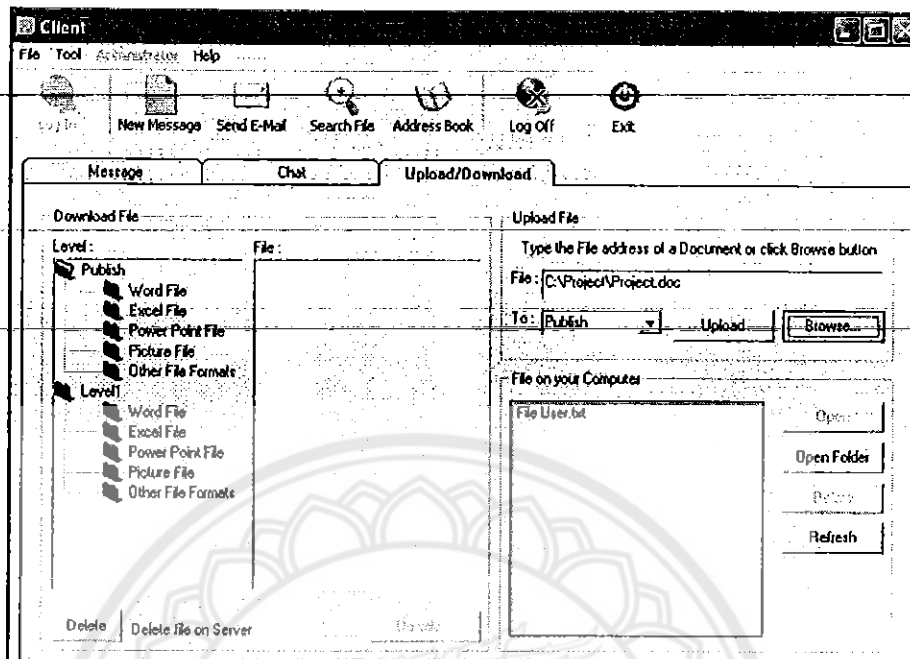
หากต้องการอัปโหลดไฟล์ ก็สามารถที่จะเลือกไฟล์ที่ต้องการอัปโหลดได้ที่ปุ่ม Browse ดัง

รูป 4.7



รูปที่ 4.7 ภาพแสดงการอัปโหลด(1)

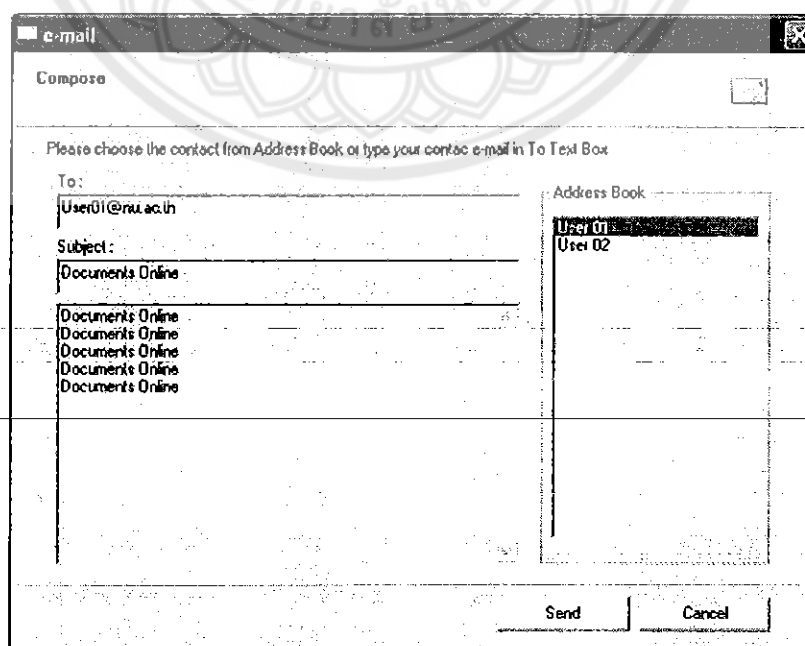
จากนั้นเลือกกระดบที่ต้องการแล้วทำการอัปโหลด ดังรูป 4.8



รูปที่ 4.8 ภาพแสดงการอัปโหลด(2)

4.2.4 ส่งอีเมล (Send e-mail)

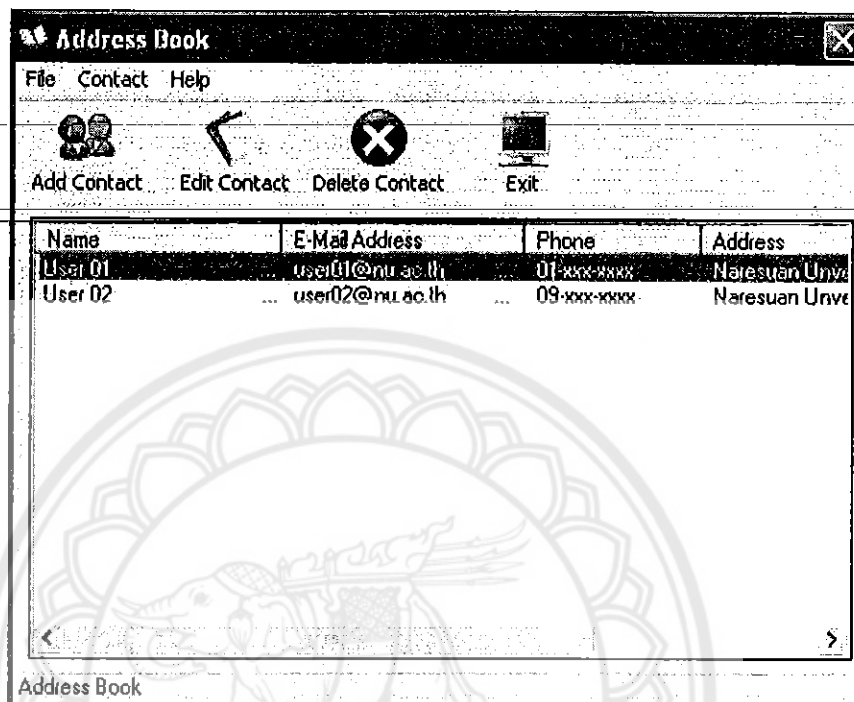
หากต้องการส่งอีเมลให้เลือกที่ Tool Bar แล้วเลือกที่ Send E-Mail ซึ่งภายในส่วนนี้จะเป็นการส่งอีเมลไปยังอีเมลของผู้รับ ดังรูป 4.9



รูปที่ 4.9 ภาพแสดงการส่งอีเมล

4.2.5 สมุดรายชื่อ (Address Book)

ในส่วนนี้จะเป็นส่วนที่เก็บรายชื่ออีเมลล์ของยูสเซอร์ แต่ละคนสามารถเพิ่มเติมและแก้ไขได้ ซึ่งจะประกอบด้วยรายละเอียดต่าง ๆ ดังรูป 4.10



รูปที่ 4.10 ภาพแสดงสมุดรายชื่อ

หากต้องการเพิ่มรายชื่อลงในสมุดรายชื่อ ก็สามารถทำได้โดยการเลือกที่ Add Contact ก็จะพบกับหน้าต่าง Add Contact ดังรูป 4.11

The screenshot shows a dialog box titled "New Contact" with a close button. It contains two sections for entering contact information:

Enter name and e-mail address about contact

*Name: User 09

*E-Mail Address: user09@nu.ac.th

Enter home address and phone about contact

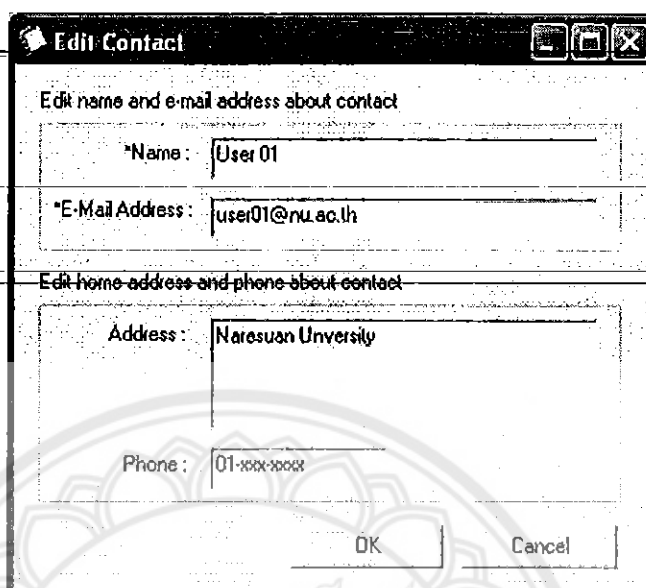
Address: Naresuan University

Phone: 07-xxx-xxxx

At the bottom, there are "OK" and "Cancel" buttons.

รูปที่ 4.11 ภาพแสดงการเพิ่มรายชื่อลงในสมุดรายชื่อ

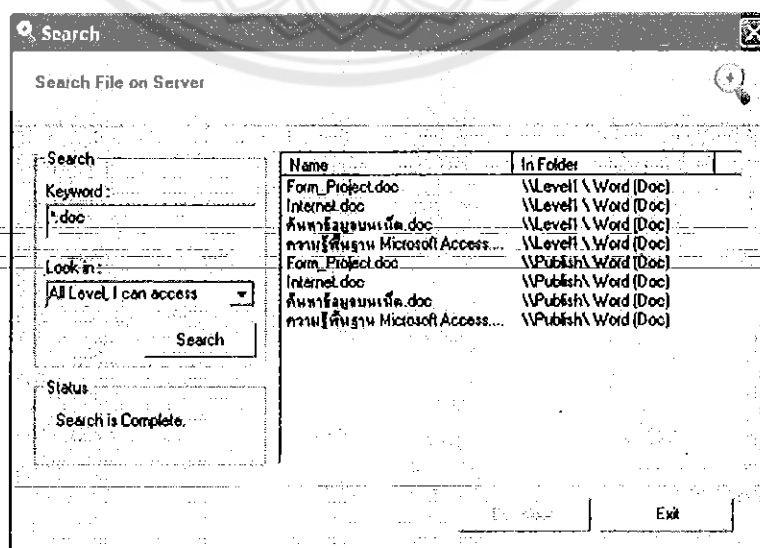
หากต้องการแก้ไขรายชื่อใน สมุดรายชื่อ ก็สามารถทำได้โดยการเลือกที่ Edit Contact ก็จะมีพบกับหน้าต่าง Edit Contact ดังรูป 4.12



รูปที่ 4.12 ภาพแสดงการแก้ไขรายชื่อในสมุดรายชื่อ

4.2.6 ค้นหาไฟล์หรือเอกสาร (Search file)

ส่วนนี้เป็นส่วนของการค้นหาไฟล์หรือเอกสารซึ่งถูกจัดเก็บอยู่ภายในเซิร์ฟเวอร์อย่างเป็นระเบียบ หากต้องการค้นหาที่เลือกที่ Tool Bar แล้วเลือก Search File ก็จะมีพบกับหน้าต่างดังรูป 4.13 ซึ่งการค้นหาสามารถค้นหาได้โดยการใส่ keyword ลงไปแล้วเลือกระดับของเอกสารที่ต้องการค้นหา

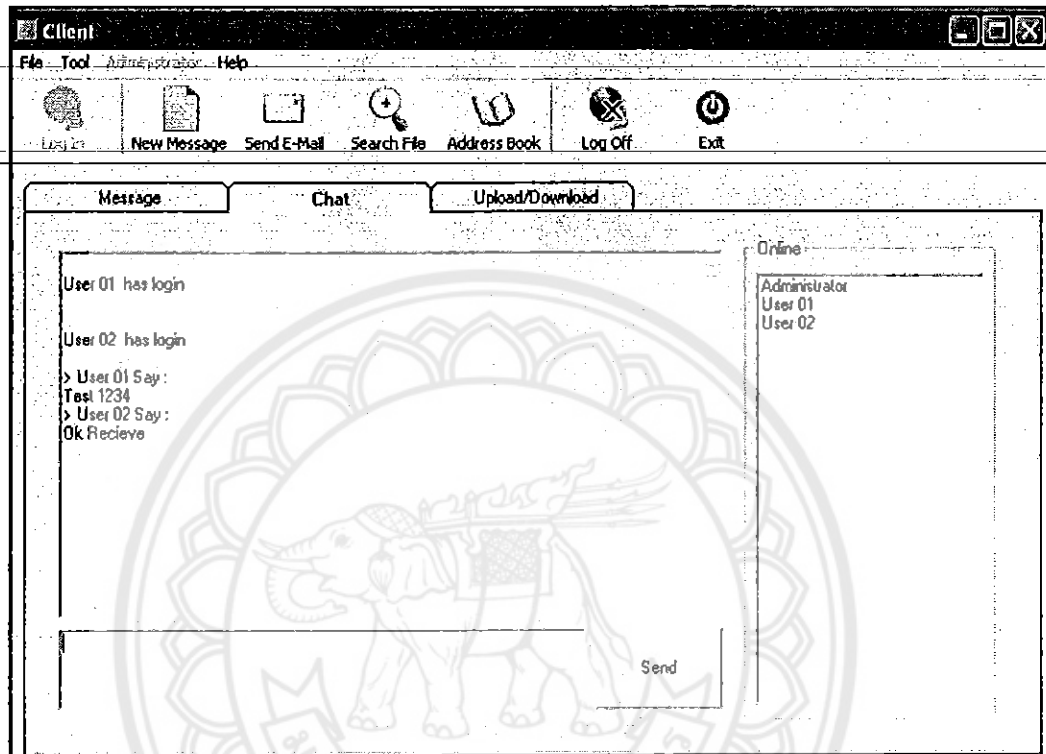


รูปที่ 4.13 ภาพแสดงการค้นหาไฟล์

4.2.7 การสนทนา (Chat)

ส่วนนี้จะเป็นส่วนของการสนทนา ซึ่งยูสเซอร์ทุกคนสามารถเข้ามาใช้ได้โดยการเลือกที่ Chat ซึ่งจะมีรายชื่อของผู้ที่อยู่ในห้องสนทนาแสดงอยู่ และมีส่วนที่แสดงข้อความในการสนทนา

ผังรูป 4.14



รูปที่ 4.14 ภาพแสดงการสนทนา

จากการทดสอบโปรแกรมในแต่ส่วนข้างต้น ผลที่ได้เป็นที่น่าพอใจเพราะโปรแกรมสามารถทำงานได้ตามที่ออกแบบไว้

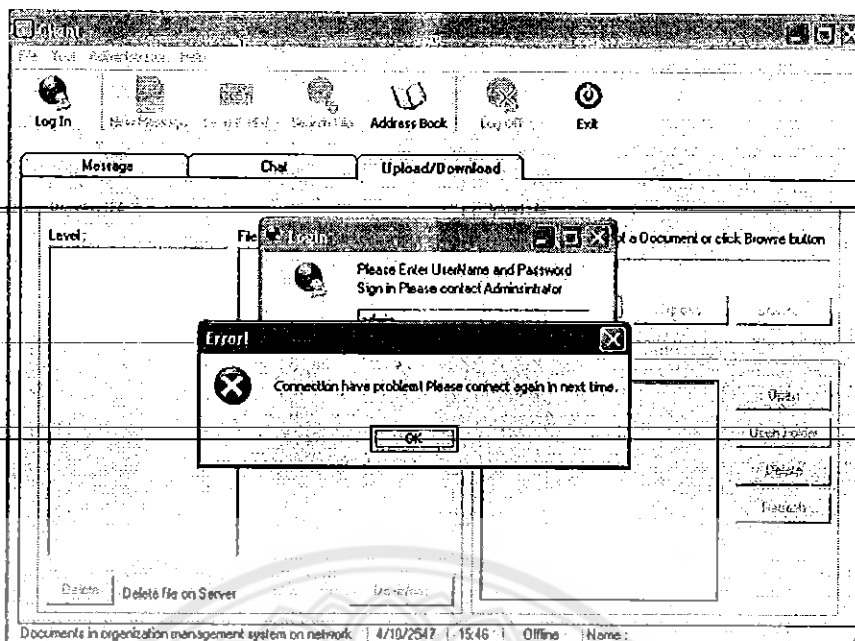
4.3 ปัญหาที่พบขณะทำการทดสอบและวิธีแก้ไข

กรณีที่ 1 เมื่อทำการรันโปรแกรมส่วนไคลเอนต์ แล้วเมื่อล็อกอิน มีข้อความแสดงขึ้นดังรูป

4.15 แสดงว่าสาเหตุมาจาก เซิร์ฟเวอร์ ยังไม่ได้รับ หรือ IP Address ของ ไคลเอนต์ ไม่ตรงกันกับเซิร์ฟเวอร์

การแก้ไข

1. ตรวจสอบว่าเซิร์ฟเวอร์รันหรือยังหากเซิร์ฟเวอร์ยังไม่ได้รับก็ให้ทำการรันเสียก่อน
2. ตรวจสอบว่า IP ในโปรแกรมตรงกับ IP ของเซิร์ฟเวอร์หรือไม่ถ้าไม่ตรงก็เปลี่ยนให้ตรงกับ เซิร์ฟเวอร์ แล้วรัน โปรแกรมใหม่อีกครั้ง



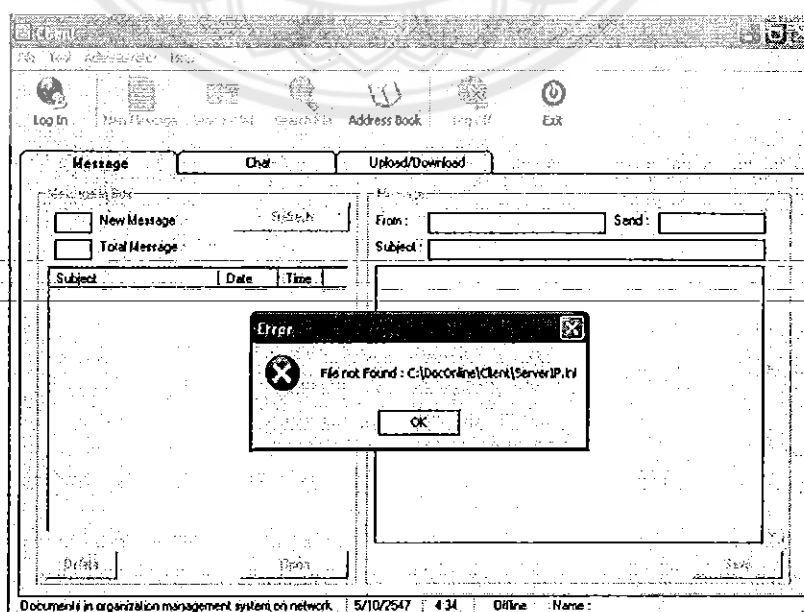
รูปที่ 4.15 Error connection

กรณีที่ 2

เมื่อทำการรัน โปรแกรมส่วนไคลเอนต์แล้วเมื่อล็อกอินมีข้อความแสดงขึ้นดังรูป 4.16 แสดงว่าสาเหตุมาจากโปรแกรมหาไฟล์ ServerIP.ini ไม่พบหรือไฟล์ ServerIP.ini อยู่ผิดที่

การแก้ไข

ตรวจสอบดูว่าในไฟล์เดอร์ ของโปรแกรมมีไฟล์ ServerIP.ini หรือไม่ถ้าไม่มีก็ให้ทำการคัดลอกไฟล์มาจากเครื่องที่มี หรือถ้ามีแต่อยู่ผิดที่ก็ให้ย้ายมาไว้ในที่ที่ถูกต้อง



รูปที่ 4.16 Not found ServerIP.ini

กรณีที่ 3 ขณะที่รัน โปรแกรมส่วนไคลเอนต์อยู่แล้วเซิร์ฟเวอร์เกิดปิดตัวลงกระทันหันโดยที่ ยูสเซอร์ไม่ทราบเมื่อจะล็อกอินเข้าไปใหม่ก็จะมีข้อความแสดงขึ้นเหมือนกับกรณี ที่ 2 ดังรูป 4.16

การแก้ไข รอให้เซิร์ฟเวอร์รันเสียก่อนแล้วจึงคอยล็อกอินเข้าไปใหม่อีกครั้ง

กรณีที่ 4 เมื่อจะทำการรัน โปรแกรมส่วนไคลเอนต์แล้วไม่สามารถรันได้เกิดข้อความว่าหา ไฟล์ DLL ไม่พบอาจเกิดจากการลงโปรแกรมไม่สมบูรณ์ หรือไฟล์ DLL ถูกลบไป

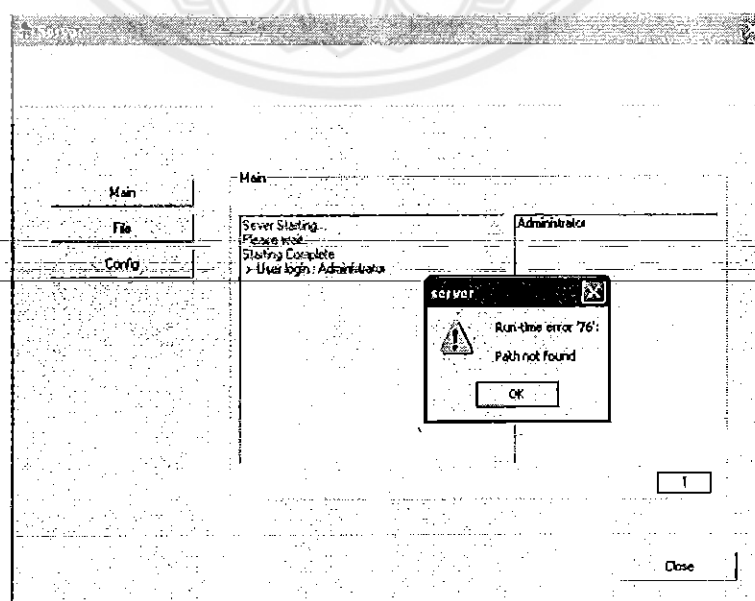
การแก้ไข หากเกิดจากการลงโปรแกรมไม่สมบูรณ์ก็ให้ทำการลงโปรแกรมใหม่อีกครั้งหรือ จะคัดลอกไฟล์ DLL ที่ขาดไปมาจากเครื่องอื่นก็ได้แต่ถ้าไฟล์ DLL ถูกลบไปก็ให้ คัดลอก ไฟล์ DLL ที่หายไปไปมาจากเครื่องอื่น

กรณีที่ 5 เมื่อทำการส่งข้อความแล้วผู้รับไม่ได้รับข้อความแสดงว่าสาเหตุมาจากไฟล์ Message.ini ที่อยู่ภายในเซิร์ฟเวอร์ ติดสถานะ Read only file

การแก้ไข แก้ไขไฟล์ Message.ini จากสถานะ Read only file

กรณีที่ 6 ขณะที่รัน โปรแกรมส่วนเซิร์ฟเวอร์อยู่แล้วมีข้อความแสดงขึ้นมา ดังรูป 4.17 แสดง ว่าสาเหตุมาจากชื่อพาร์ทของไฟล์เคอร์ต่าง ๆ ภายในเซิร์ฟเวอร์ไม่ถูกต้อง

การแก้ไข ตรวจสอบดูว่าชื่อพาร์ทของไฟล์เคอร์ต่าง ๆ ภายในเซิร์ฟเวอร์ถูกต้องหรือไม่ ถ้าไม่ ถูกต้องก็ให้แก้ไขให้เรียบร้อย



รูปที่ 4.17 Path not found

บทที่ 5

สรุปผลการดำเนินงานและข้อเสนอแนะ

ในบทนี้จะกล่าวถึงการสรุปผลการดำเนินงานที่ได้จากศึกษาและดำเนินงานเพื่อการจัดทำโปรแกรมการจัดการระบบเอกสารภายในองค์กรผ่านเครือข่ายงานได้ โปรแกรมที่ผู้พัฒนาคิดว่ามีประสิทธิภาพเพียงพอต่อการใช้งานภายในองค์กร ซึ่งจะช่วยให้การทำงานเป็นไปอย่างรวดเร็วและมีประสิทธิภาพมากยิ่งขึ้น

5.1 สรุปผลการดำเนินโครงการ

จากการดำเนินงานจัดทำโปรแกรมการจัดการระบบเอกสารภายในองค์กรผ่านเครือข่าย มีจุดประสงค์เพื่อให้การทำงานภายในองค์กรเป็นไปอย่างรวดเร็ว, ปลอดภัยและมีประสิทธิภาพมากยิ่งขึ้น ได้แก่ ช่วยให้การสื่อสารภายในองค์กรมีความสะดวกรวดเร็วมากยิ่งขึ้น รักษาความลับทางเอกสารภายในองค์กร และลดปริมาณการใช้กระดาษในการจัดเก็บเอกสาร ซึ่งจะช่วยให้ประหยัดค่าใช้จ่ายภายในองค์กร ซึ่งได้ผลการดำเนินงานดังนี้

- การติดต่อสื่อสารกันภายในองค์กรมีความรวดเร็วมากขึ้น เนื่องจากการติดต่อกันนั้นสามารถติดต่อกันผ่านทาง โปรแกรมได้ทันที และยังสามารถส่งงานผ่านทาง โปรแกรมได้อีกด้วย
- ความปลอดภัยของข้อมูลมีมากขึ้นเนื่องจากการแบ่งระดับการเข้าถึงข้อมูลแตกต่างกันตามแต่ละหน้าที่ของบุคคล
- สามารถลดปริมาณการใช้กระดาษให้น้อยลงได้และการค้นหาเอกสารต่าง ๆ นั้นสามารถทำได้อย่างสะดวกรวดเร็วมากยิ่งขึ้น เนื่องจากการจัดเก็บที่เป็นระเบียบ และสามารถเรียกใช้เอกสารได้ทันที

5.2 ประโยชน์ที่ได้รับจากการทำโครงการ

- ได้รับความรู้ความเข้าใจในเรื่องของการสื่ออกอิน โดยผ่านระบบความปลอดภัยด้วยการเข้ารหัสและถอดรหัส
- ได้รับความรู้ความเข้าใจในเรื่องของการส่งอีเมลล์ผ่านsmtp เซิร์ฟเวอร์
- ได้รับความรู้ความเข้าใจในเรื่องของการเชื่อมต่อกับฐานข้อมูล
- ได้รับความรู้ความเข้าใจในเรื่องของการติดต่อสื่อสารผ่าน โปรโตคอล TCP / IP

- ได้รับความรู้ความเข้าใจในเรื่องของการพัฒนาโปรแกรมด้วย Microsoft Visual Basic
- ได้รับความรู้ความเข้าใจในเรื่องของการพัฒนาโปรแกรมผ่านระบบเครือข่าย โดยการใช้ Microsoft Winsock Control

5.3 ข้อเสนอแนะ

ผู้จัดทำโครงการมีข้อเสนอแนะในการทำโครงการดังต่อไปนี้

- จากการศึกษาโครงการและพัฒนาโปรแกรมการจัดการระบบเอกสารภายในองค์กรผ่านเครือข่าย พบว่าสามารถนำไปปรับใช้กับองค์กรต่าง ๆ ที่มีลักษณะคล้ายคลึงกันได้
- โปรแกรมการจัดการระบบเอกสารภายในองค์กรผ่านเครือข่ายที่ได้พัฒนาขึ้นมาแล้วยังคงมีข้อจำกัดอยู่บางประการหากต้องการให้โปรแกรมมีประสิทธิภาพมากยิ่งขึ้นก็ต้องทำการแก้ไขข้อจำกัดต่าง ๆ เหล่านี้เสียก่อน ซึ่งข้อจำกัดต่าง ๆ มีดังนี้
 - โปรแกรมถูกออกแบบมาสำหรับการสื่อสารภายในองค์กรเท่านั้น
 - โปรแกรมเหมาะสำหรับองค์กรขนาดเล็กที่มีการจัดเก็บเอกสารไม่ซับซ้อนมากนัก
 - โปรแกรมไม่สามารถสนทนากันแบบ 1-1 ได้
 - โปรแกรมสามารถส่งอีเมลล์ได้เฉพาะ โดเมนของ nu.ac.th เท่านั้น
 - ขณะที่ เซิร์ฟเวอร์ กำลังทำการรับ-ส่งไฟล์เซิร์ฟเวอร์จะไม่สามารถทำงานอย่างอื่นได้จนกว่าการรับ-ส่งจะเสร็จสิ้น
- เนื่องจากโปรแกรมการจัดการระบบเอกสารภายในองค์กรผ่านเครือข่ายที่ได้พัฒนาขึ้นมานี้อาจจะยังเป็นของใหม่สำหรับองค์กรจึงควรมีการอบรมและสาธิตวิธีการใช้ให้กับผู้ใช้อีกเพื่อให้ผู้ใช้มีคุณภาพมากยิ่งขึ้นและสามารถนำไปใช้ได้อย่างมีประสิทธิภาพสูงสุด

5.4 แนวทางการพัฒนาโปรแกรมในอนาคต

- พัฒนาระบบการจัดการเอกสารให้มีการควบคุมการไหลเวียนของเอกสารให้ถูกต้องตามแบบขององค์กร
- พัฒนาระบบความปลอดภัยของระบบโดยปรับปรุง โปรโตคอลในการล็อกอิน และปรับปรุงกลไกในการเข้าถึงเอกสาร โดยอาจใช้การเข้ารหัสและถอดรหัสเอกสาร รวมทั้งการใช้ลายเซ็นดิจิทัลเข้ามาช่วยในการยืนยันตัวตนและป้องกันการปฏิเสธความรับผิดชอบของผู้เซ็นเอกสาร

- พัฒนาระบบโครงสร้างของโปรแกรมให้สามารถรองรับผู้ใช้ได้มากขึ้นเพื่อพัฒนาไปใช้กับองค์กรที่มีขนาดใหญ่ขึ้น
- พัฒนาระบบการสนทนาให้สามารถเลือกคุยกันแบบตัวต่อตัวได้
- พัฒนาระบบติดต่อฐานข้อมูลให้สามารถรองรับฐานข้อมูลแบบอื่นได้
- พัฒนาระบบให้สามารถแบ่งระดับการเข้าถึงข้อมูลให้เหมาะสมกับองค์กร
- พัฒนาระบบของการค้นหาเซิร์ฟเวอร์อัตโนมัติกรณีเปลี่ยนเซิร์ฟเวอร์ใหม่หรือมีการเคลื่อนย้ายเซิร์ฟเวอร์
- พัฒนาระบบให้มีการเก็บล็อกไฟล์เพื่อแจ้งสถานะของการทำงานของเซิร์ฟเวอร์ไว้ตรวจสอบว่าเซิร์ฟเวอร์ทำงานอะไรบ้างหรือตรวจสอบในกรณีเซิร์ฟเวอร์ทำงานผิดพลาด



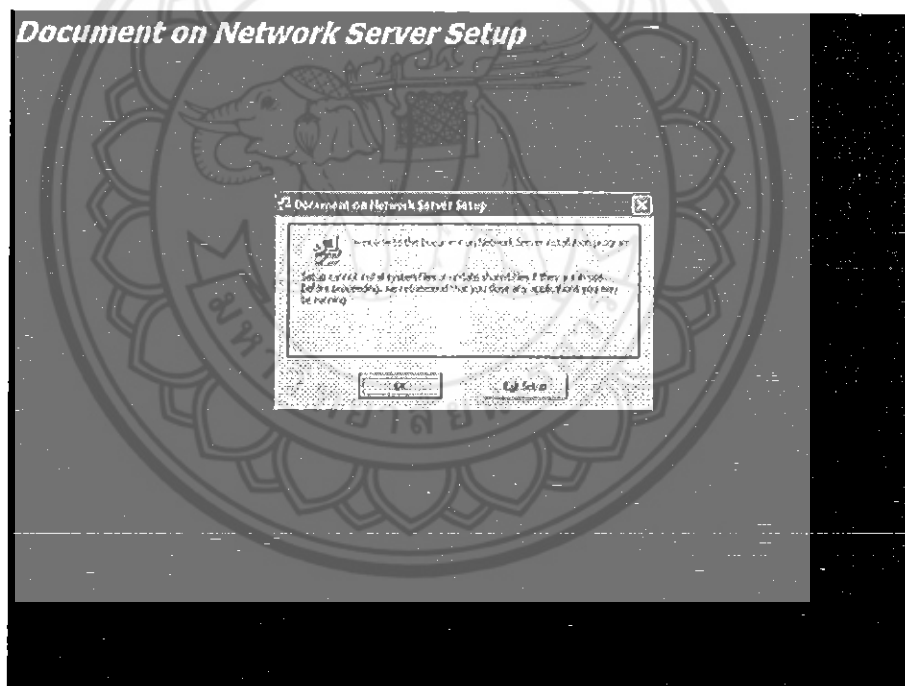
ภาคผนวก

ในส่วนนี้จะเป็นการอธิบายถึงขั้นตอนต่างๆ ในการติดตั้งโปรแกรมการจัดการระบบเอกสารภายในองค์กรผ่านเครือข่าย ในการอธิบายขั้นตอนการติดตั้งโปรแกรมจะแบ่งเป็น 2 ส่วน คือ การติดตั้งโปรแกรมในส่วนของเซิร์ฟเวอร์ (Server) และ การติดตั้งโปรแกรมในส่วนของไคลเอนต์ (Client) ซึ่งมีรายละเอียดดังนี้

ขั้นตอนการติดตั้งโปรแกรมในส่วนของเซิร์ฟเวอร์ (Server)

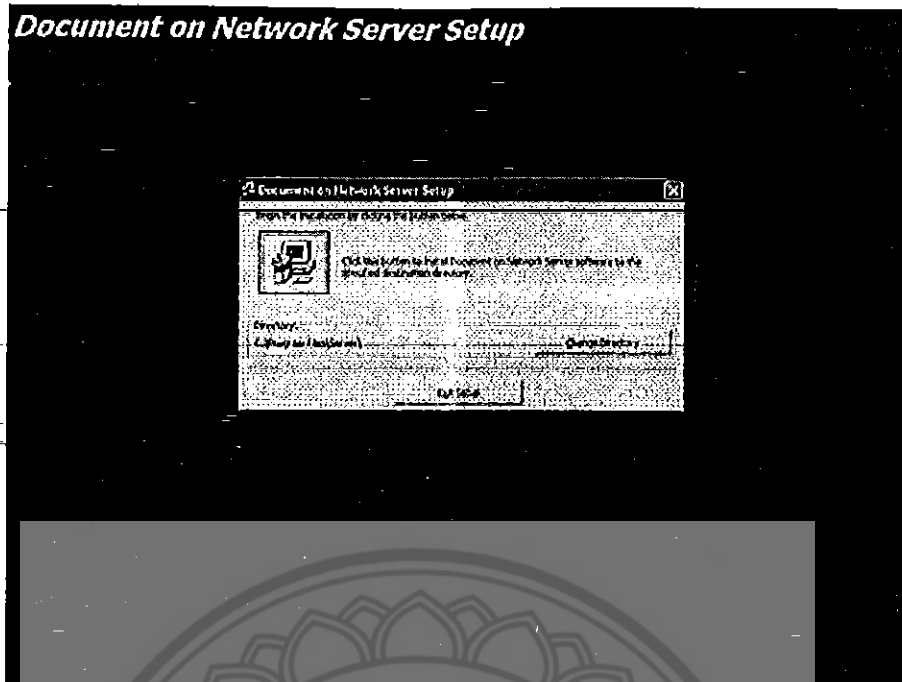
เมื่อทำการรันตัวเซตอัพ (Setup) ของโปรแกรมจะเป็นการเข้าสู่หน้าจอการติดตั้งโปรแกรม

ผังรูป 6.1



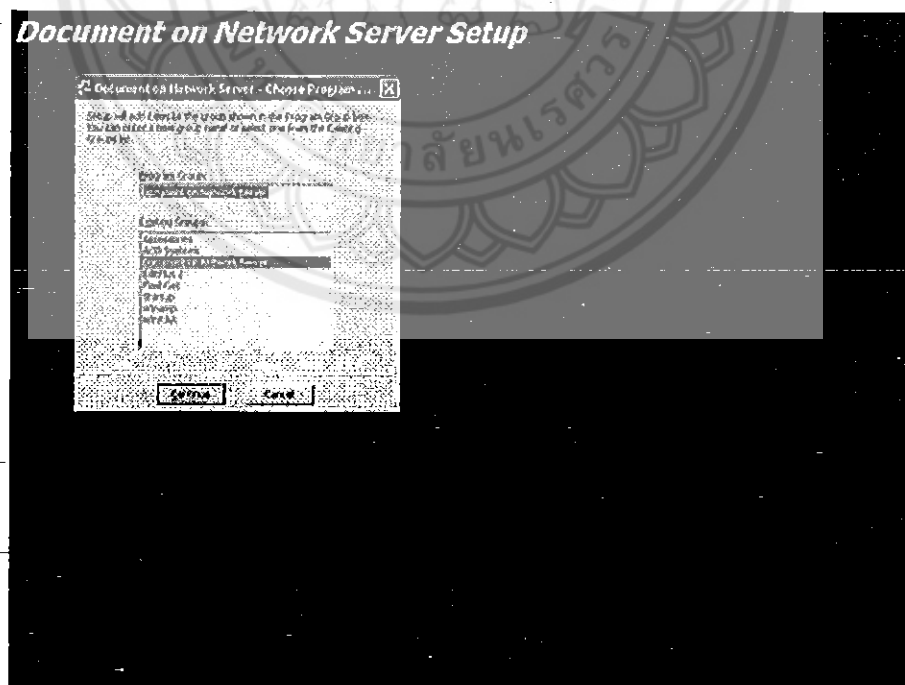
รูปที่ 6.1 การติดตั้งเซิร์ฟเวอร์(1)

เมื่อต้องการติดตั้งโปรแกรมก็ให้เลือกที่ OK จากนั้นจะเป็นการเข้าสู่หน้าจอเพื่อเลือกปลายทางที่ต้องการติดตั้งโปรแกรม ดังรูป 6.2



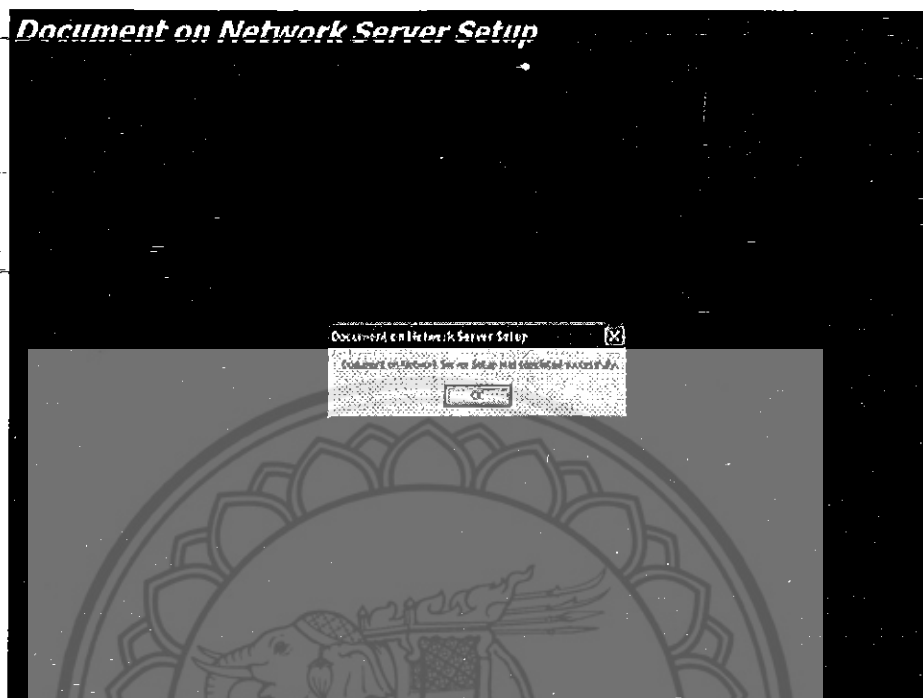
รูปที่ 6.2 การติดตั้งเซิร์ฟเวอร์(2)

เมื่อเลือกปลายทางที่ต้องการติดตั้งโปรแกรมได้แล้วก็เลือกที่ไอคอนตามรูป 6.2 ก็จะพบกับหน้าจอตั้งรูป 6.3 ซึ่งจะเป็นการติดตั้งโปรแกรมลงบนสตาร์ทเมนู (Start Menu)



รูปที่ 6.3 การติดตั้งเซิร์ฟเวอร์(3)

จากนั้นระบบก็จะทำการติดตั้งโปรแกรมลงในเครื่อง และเมื่อติดตั้งเสร็จก็จะพบกับหน้าจอ
ดังรูป 6.4

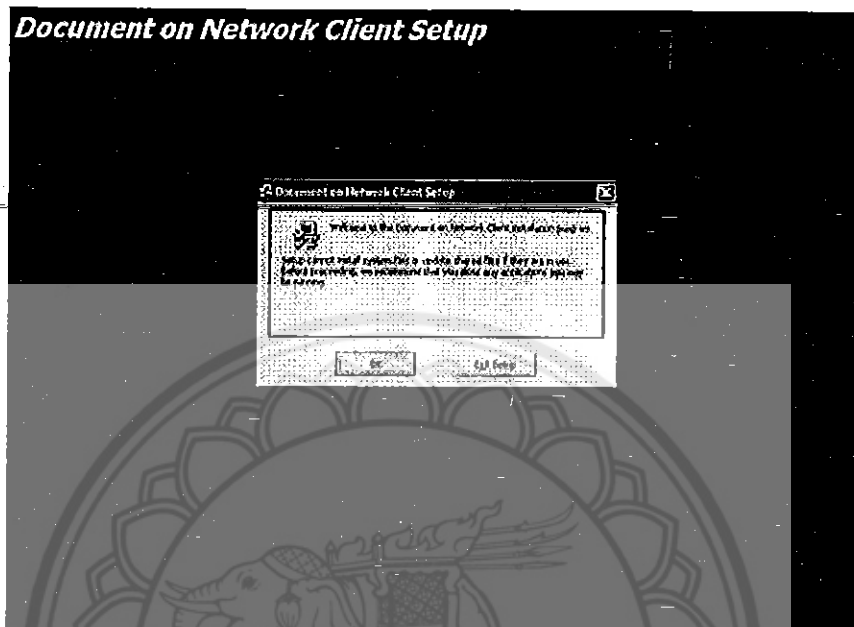


รูปที่ 6.4 การติดตั้งเซิร์ฟเวอร์(4)

เมื่อทำการติดตั้งโปรแกรมเป็นที่เรียบร้อยแล้วก็ให้เลือกที่ Start --> Program --> Document on Network Server --> Setup ซึ่งจะเป็นขั้นตอนสุดท้ายของการติดตั้ง โปรแกรมเป็นอันเสร็จสิ้นการติดตั้งโปรแกรมในส่วนของเซิร์ฟเวอร์

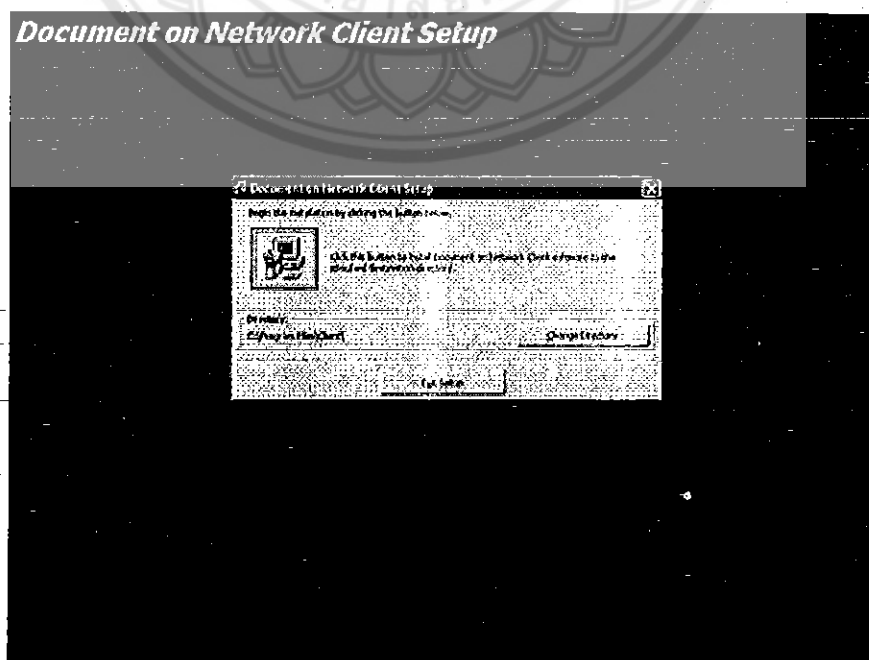
ขั้นตอนการติดตั้งโปรแกรมในส่วนของไคลเอนต์ (Client)

เมื่อทำการรันตัวเซตอัพ (Setup) ของโปรแกรมจะเป็นการเข้าสู่หน้าจอการติดตั้งโปรแกรม
 ดังรูป 6.5



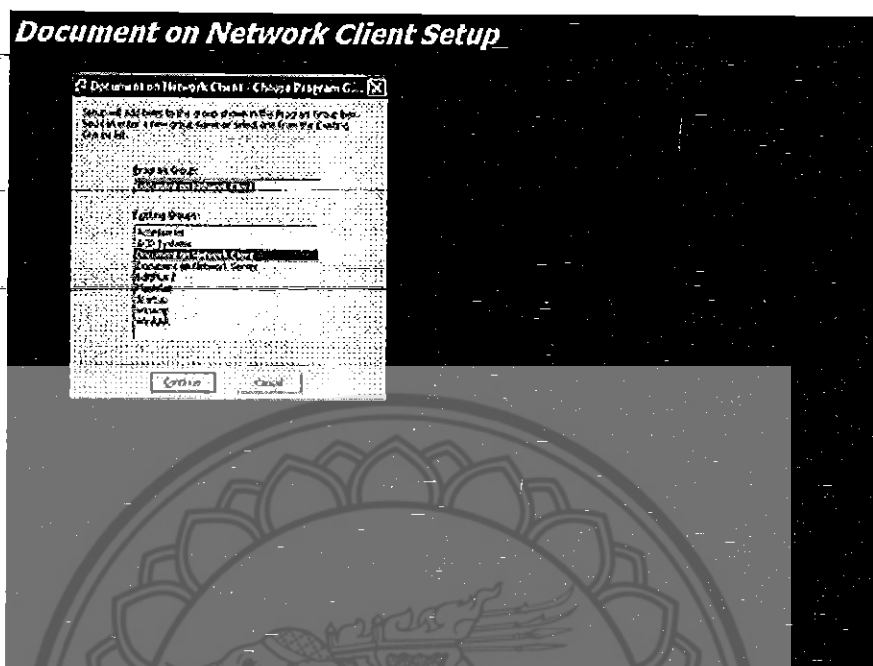
รูปที่ 6.5 การติดตั้ง ไคลเอนต์(1)

เมื่อต้องการติดตั้งโปรแกรมก็ให้เลือกที่ OK จากนั้นจะเป็นการเข้าสู่หน้าจอเพื่อเลือก
 ปลายทางที่ต้องการติดตั้ง โปรแกรม ดังรูป 6.6



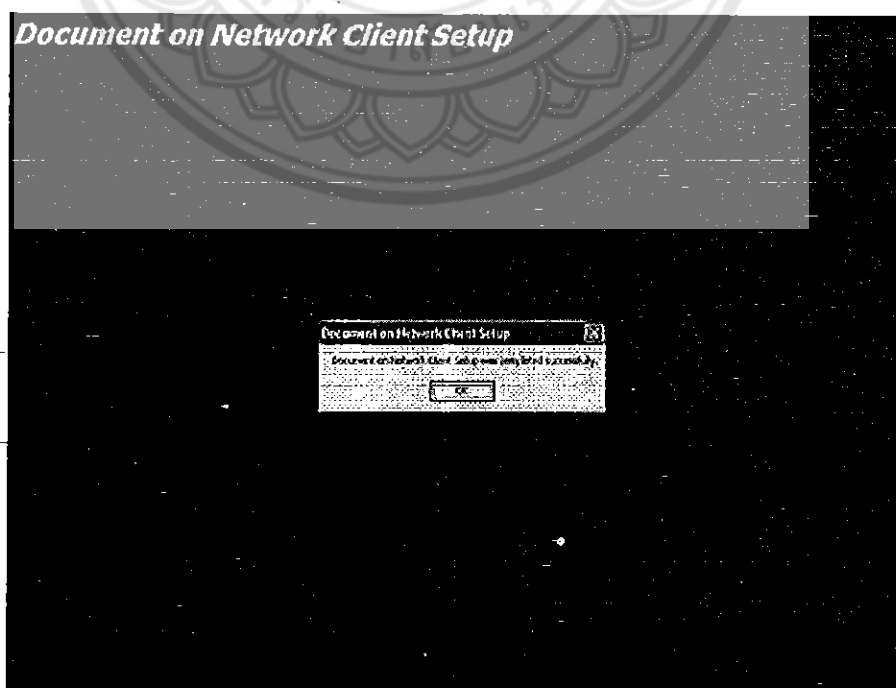
รูปที่ 6.6 การติดตั้ง ไคลเอนต์(2)

เมื่อเลือกปลายทางที่ต้องการติดตั้งโปรแกรมได้แล้วก็เลือกที่ไอคอนตามรูป 6.6 ก็จะพบกับหน้าจอดังรูป 6.7 ซึ่งจะเป็นการติดตั้งโปรแกรมลงบนสตาร์ทเมนู (Start Menu)



รูปที่ 6.7 การติดตั้ง ไคลเอนต์(3)

จากนั้นระบบก็จะทำการติดตั้ง โปรแกรมลงในเครื่อง และเมื่อติดตั้งเสร็จก็จะพบกับหน้าจอดังรูป 6.8



รูปที่ 6.8 การติดตั้ง ไคลเอนต์(4)

เมื่อทำการติดตั้ง โปรแกรมเป็นที่เรียบร้อยแล้วก็ให้เลือกที่ Start --> Program --> Document on Network Client --> Setup ซึ่งจะเป็นขั้นตอนสุดท้ายของการติดตั้ง โปรแกรมเป็นอันเสร็จสิ้นการติดตั้งโปรแกรมในส่วนของไคลเอนต์



มหาวิทยาลัยนเรศวร

เอกสารอ้างอิง

- [1] กิตติ ภัคดีวัฒนกุล , จำลอง ทรูตสาหะ. **Visual Basic 6 ฉบับโปรแกรมเมอร์**. กรุงเทพฯ: ไทยเจริญการพิมพ์. 2542.
- [2] กิตติ ภัคดีวัฒนกุล , จำลอง ทรูตสาหะ. **Visual Basic 6 ฉบับฐานข้อมูล**. กรุงเทพฯ: ไทยเจริญการพิมพ์. 2544.
- [3] สัจจะ จรัสรุ่งระวีวร. **คู่มือพัฒนาแอปพลิเคชันด้วย-Visual Basic 6.0**. กรุงเทพฯ: คำนสุทธการพิมพ์. 2542.
- [4] สัจจะ จรัสรุ่งระวีวร. **คู่มือการเขียนโปรแกรมและใช้งาน Visual Basic 6.0**. กรุงเทพฯ: คำนสุทธการพิมพ์. 2542.



ประวัติผู้เขียนโครงการ



ชื่อ นายธีรวิชญ์ ทรัพย์พันธุ์
ภูมิลำเนา 35/1 ตำบลท่าอิฐ อำเภอเมือง จังหวัดอุตรดิตถ์
ประวัติการศึกษา

- จบระดับมัธยมศึกษาตอนปลายจาก โรงเรียนอุตรดิตถ์

- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 5

สาขาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
มหาวิทยาลัยนเรศวร

e-mail : teerawit_p@hotmail.com



ชื่อ นายเทพรัตน์ มุลกัณษา
ภูมิลำเนา 167/48 หมู่ 7 ตำบลท่าโพธิ์ อำเภอเมือง จังหวัด
พิษณุโลก

ประวัติการศึกษา

- จบระดับมัธยมศึกษาตอนปลายจาก โรงเรียนจุฬารัตน์ราช
วิทยาลัย

- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 5

สาขาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
มหาวิทยาลัยนเรศวร

e-mail : Moonkanta@hotmail.com