

ระบบควบคุมคอมพิวเตอร์ช่วยในการเรียนการสอน

A Remote Control System for E - Learning

นางสาวจิรรัตน์	เอี่ยมสอาด	รหัส	47360045
นายชินภัทร	คำแน่น	รหัส	47361894
นายเรืองฤทธิ์	บุญรอด	รหัส	47362314

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ..... 25 พ.ค. 2553
เลขทะเบียน..... 1.5008748
เลขเรียกหนังสือ..... 6.5.215
มหาวิทยาลัยนเรศวร

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2550



ใบรับรองโครงการวิศวกรรม

หัวข้อโครงการ	ระบบควบคุมคอมพิวเตอร์ช่วยในการเรียนการสอน		
ผู้ดำเนินโครงการ	นางสาวจิรารัตน์ เอี่ยมสอาด	รหัส	47360045
	นายชินภัทร คำแน่น	รหัส	47361894
	นายเรืองฤทธิ์ บุญรอด	รหัส	47362314
อาจารย์ที่ปรึกษา	อาจารย์เศรษฐา ตั้งคำวานิช		
สาขาวิชา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2550		

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเกษตรศาสตร์ อนุมัติให้โครงการฉบับนี้เป็นส่วนหนึ่งของ
การศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์
คณะกรรมการสอบ โครงการวิศวกรรม

ได้รับ..... ส่ง..... ประธานกรรมการ
(อาจารย์เศรษฐา ตั้งคำวานิช)

..... กรรมการ
(ดร.ไพศาล มณีสว่าง)

..... กรรมการ
(อาจารย์ภาณุพงศ์ สอนคม)

หัวข้อโครงการ	ระบบควบคุมคอมพิวเตอร์ช่วยในการเรียนการสอน		
ผู้ดำเนินโครงการ	นางสาวจิรารัตน์	เอี่ยมสอาด	รหัส 47360045
	นายชินภัทร	กำแน่น	รหัส 47361894
	นายเรืองฤทธิ์	บุญรอด	รหัส 47362314
อาจารย์ที่ปรึกษา	อาจารย์เสรษฐา ตั้งค้ำวานิช		
สาขาวิชา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2550		

บทคัดย่อ

โครงการนี้ได้พัฒนาซอฟต์แวร์คอมพิวเตอร์ระดับเครือข่าย (Computer Network) เพื่อช่วยในการควบคุมระบบคอมพิวเตอร์ในการเรียนการสอน ซอฟต์แวร์แบ่งเป็นสองส่วนคือ ส่วนที่ติดตั้งและทำงานบนคอมพิวเตอร์แม่ข่าย (Server) และส่วนที่ติดตั้งและทำงานบนคอมพิวเตอร์ลูกข่าย (Client) ซอฟต์แวร์ถูกพัฒนาด้วยภาษาจาวา ให้มีความสามารถในการสนทนาและจัดกลุ่มสนทนา, การส่งไฟล์จากเครื่อง Server ไปยัง Client และส่งจาก Client ไปยัง Server, การส่งไฟล์แบบกระจาย (Broadcast) และการนำความสามารถของระบบ VNC (Virtual Network Computing) ที่มีความสามารถให้ Client สามารถควบคุมหน้าจอของ Server (Remote Access) มาพัฒนาต่อให้เครื่อง Client สามารถเห็นหน้าจอของ Server เพื่อให้การเรียนการสอนในห้องเรียนสะดวกมากขึ้น

Project Title A Remote Control System for E-Learning.
Name Miss Jirarat Ieamsaard ID. 47360045
Mr. Chinnapat Kamnan ID. 47361894
Mr. Ruengrit Bunrod ID. 47362314
Project Advisor Mr. Settha Tungkawanit
Major Computer Engineering.
Department Electrical and Computer Engineering.
Academic Year 2007

.....

ABSTRACT

This project was developed computer software on computer network level to be useful for computer system control in learning. The software was divided into two parts: an installed part working on server and an installed part working on client. The software was developed by using Java and designed to be able to chat and group chatting, send files from server to client and send files from client to server, send files through broadcasting, and bring ability of VNC system (Virtual Network Computing) which was allowed clients to control a monitor of server (Remote Access) to further develop in client's computer to see the monitor of server in class learning comfortably.

กิตติกรรมประกาศ

ขอขอบพระคุณ อาจารย์เศรษฐา ตั้งคำวานิช อาจารย์ที่ปรึกษาโครงการนี้ ที่คอยให้ คำปรึกษา ความช่วยเหลือตลอดจนคำแนะนำและแนวทางต่างๆ ในการทำโครงการนี้ และสุดท้าย ขอขอบพระคุณอาจารย์ทุกท่านและเพื่อนๆ ทุกคนที่ยังไม่ได้เอ่ยนามที่คอยให้การสนับสนุน ผู้ดำเนินโครงการ ให้สามารถทำโครงการนี้จนสำเร็จลุล่วงไปได้ด้วยดี



สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญตาราง	ฉ
สารบัญรูป	ช
บทที่ 1 บทนำ.....	1
1.1 ที่มาและความสำคัญของโครงการ.....	1
1.2 วัตถุประสงค์ของโครงการ.....	2
1.3 ขอบข่ายของโครงการ.....	2
1.4 ขั้นตอนการดำเนินงาน.....	2
1.5 แผนการดำเนินงาน.....	3
1.6 ผลที่คาดว่าจะได้รับ.....	3
1.7 งบประมาณของโครงการ.....	3
บทที่ 2 หลักการและทฤษฎีที่เกี่ยวข้อง	4
2.1 VNC (Virtual Network Computing).....	4
2.2 Network Programming.....	5
2.3 TCP.....	6
2.4 UDP.....	7
2.5 Ports.....	8
2.6 Socket.....	9
2.7 Scope of Multicast IP.....	10
2.8 Stream.....	11
บทที่ 3 วิธีดำเนินงาน	13
3.1 VNC (Virtual Network Computing).....	13

สารบัญ (ต่อ)

	หน้า
3.2 การออกแบบโปรแกรม.....	15
3.3 สถาปัตยกรรมของระบบ (System Architecture).....	19
3.4 โปรแกรมการสร้างห้องสนทนา (Chat Room).....	21
3.5 โปรแกรมการรับส่งไฟล์ (File Server).....	29
3.6 โปรแกรมการรับส่งไฟล์แบบกระจาย (Broadcast).....	34
3.7 โปรแกรมดูหน้าจอเครื่องแม่ข่าย (RCL VNC).....	39
บทที่ 4 ผลการทดลอง	44
4.1 การทดสอบการทำงานของ Chat Room.....	44
4.2 การทดสอบการทำงานรับส่งไฟล์ (File Server).....	53
4.3 การทดสอบการทำงานของ (Broadcast)	55
4.4 การทดสอบการทำงานของ VNC.....	59
บทที่ 5 บทสรุป	61
5.1 วิเคราะห์ผลการทดลอง.....	61
5.2 ปัญหาและการแก้ไข	64
5.3 สรุปผลการทดลอง.....	65
5.4 ข้อเสนอแนะ	65
เอกสารอ้างอิง	66
ประวัติผู้เขียนโครงการ	67

สารบัญตาราง

ตารางที่	หน้า
2.1 ตัวอย่าง Well Known Ports[2].....	9
2.2 TTL Scoping[2].....	10
2.3 Administrative Scoping.....	11
3.1 การเปลี่ยนแปลงที่มีผลต่อระบบเชิงธุรกิจของซอฟต์แวร์ช่วยควบคุมการเรียนการสอนของ เครื่องแม่ข่าย.....	16
3.2 การเปลี่ยนแปลงที่มีผลต่อการระบบเชิงธุรกิจของซอฟต์แวร์ช่วยควบคุมการเรียนการสอน ของเครื่องลูกข่าย.....	16
3.3 Business Event ของ Chat Room Server.....	24
3.4 Business Event ของ Chat Room Client.....	24
3.5 Business Event ของ File Server – Server.....	31
3.6 Business Event ของ File Server – Client.....	31
3.7 Business Event ของ Broadcast – Server.....	36
3.8 Business Event ของ Broadcast – Client.....	36
3.9 Business Event ของ RCL VNC – Server.....	41
3.10 Business Event ของ RCL VNC – Client.....	41
5.1 ผลการทดสอบซอฟต์แวร์ตาม Specification ของซอฟต์แวร์แต่ละส่วน.....	61
5.2 ปัญหา Multicast Socket ที่เกิดขึ้น.....	65

สารบัญรูป

รูปที่	หน้า
2.1 แบบจำลอง VNC	4
2.2 การใช้ข้อมูลร่วมกันของ thread ที่อยู่ในโปรเซสเดียวกัน [1]	5
2.3 โครงสร้างของ TCP Header [2].....	6
2.4 UDP Header[3]	7
2.5 ปัญหา Multicast Socket ที่เกิดขึ้น	8
3.1 แผนผังระบบงานเชิงธุรกิจ ของซอฟต์แวร์ช่วยควบคุมการเรียนการสอนของเครื่องแม่ข่าย (RCL Server).....	15
3.2 แผนผังระบบงานเชิงธุรกิจของซอฟต์แวร์ช่วยควบคุมการเรียนการสอนของเครื่องลูกข่าย (RCL Client)	15
3.3 แผนผังปฏิสัมพันธ์ตามลำดับ (Sequence diagram) การใช้งานพื้นฐานของ RCL – Server	18
3.4 Sequence Diagram การใช้งานพื้นฐานของ RCL Client.....	18
3.5 System Architecture ของ Server – Client RCL System	19
3.6 Context Diagram ของ Chat Room Server	23
3.7 Context Diagram ของ Chat Room Client	23
3.8 Sequence Diagram ของการใช้ Chat Room Server เบื้องต้น	26
3.9 Sequence Diagram ของการใช้ Chat Room Client เบื้องต้น	27
3.10 System Architecture ของ Server-Client Chat Room system	28
3.11 Context Diagram ของ File Server – Server	30
3.12 Context Diagram ของ File Server – Client.....	30
3.13 Sequence Diagram ของการใช้งานFile Server – Server.....	32
3.14 Sequence Diagram ของการใช้งาน File Server – Client	33
3.15 System Architecture ของ Server – Client File Server System.....	33
3.16 Context Diagram ของBroad cast – Server	35
3.17 Context Diagram ของBroad cast – Client.....	36
3.18 Sequence Diagram การใช้งานเบื้องต้นของ Broadcast – Server	37
3.19 Sequence Diagram การทำงานพื้นฐานของ Broadcast – Client	38
3.20 System Architecture ของ Server – Client Broadcast System	38

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.21 Context Diagram ของ RCL VNC – Serve	40
3.22 Context Diagram ของ RCL VNC – Client	40
3.23 Sequence Diagram การใช้งานเบื้องต้นของ RCL VNC – Server.....	42
3.24 Sequence Diagram การทำงานพื้นฐานของ RCL VNC – Client	42
3.25 System Architecture ของ Server – Client RCL VNC system.....	43
4.1 กำหนดของ Teacher ในการสร้างห้องสนทนา.....	44
4.2 การใส่จำนวนห้อง (Total Group) ในการสร้างห้องสนทนา	45
4.3 เลือกการตอบรับ Client เป็นแบบ Manual Acception	45
4.4 เลือกตอบรับหรือปฏิเสธการขอเข้าร่วมสนทนาของ Client	45
4.5 เลือกตอบรับหรือปฏิเสธการขอเข้าร่วมสนทนาของ Client	46
4.6 เรียก Client ที่ต้องการเข้าสู่ห้องสนทนา	46
4.7 เลือก Client ที่ต้องการเรียกเข้าในห้องสนทนา.....	47
4.8 การลบ Client ออกจากห้องสนทนา	47
4.9 การส่งไฟล์ในห้องสนทนา	48
4.10 ไฟล์ที่ส่งในห้องสนทนา.....	48
4.11 การ Broadcast ในห้องสนทนา.....	49
4.12 ไฟล์ที่ Broadcast ในห้องสนทนา	49
4.13 การเพิ่มห้องสนทนา	50
4.14 การลบห้องสนทนา.....	50
4.15 การปิดห้องสนทนา.....	51
4.16 เลือกการทำงาน Chat Room ของ Client.....	51
4.17 ใส่ชื่อของ Client.....	51
4.18 เลือกห้องสนทนาของ Client	52
4.19 ห้องสนทนาของ Client	52
4.20 รายชื่อสมาชิกที่อยู่ภายในสนทนาของ Client	52
4.21 เปิดการทำงาน Receiver File	53
4.22 เลือกที่เก็บไฟล์	53
4.23 ไฟล์ที่ซอฟต์แวร์รับและเขียนเก็บไว้ใน Directory ที่ผู้ใช้เลือกไว้.....	54

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.24 เลือกการทำงาน Send File	54
4.25 เลือกการทำงาน Send File	54
4.26 ไฟล์ที่ Client ส่งไปและถูกเก็บไว้ใน Server	55
4.27 เลือกการทำงาน Broadcast	55
4.28 เลือกไฟล์ที่ต้องการ Broadcast.....	56
4.29 เพิ่มไฟล์ในรายการสำหรับ Broadcast	56
4.30 ลบไฟล์ในรายการสำหรับ Broadcast	57
4.31 ไฟล์ที่ Broadcast และถูกเก็บไว้ที่ Client	57
4.32 สถานะและความคืบหน้าของการส่งไฟล์	57
4.33 ไฟล์ที่รับจากการ Broadcast และเขียนเก็บไว้ที่ Client	58
4.34 Directory ที่สร้างเพื่อเก็บไฟล์ที่รับมาจากการ Broadcast	58
4.35 เลือกการทำงาน VNC.....	59

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของโครงการ

เนื่องด้วยการศึกษาในปัจจุบันนี้ คอมพิวเตอร์เข้ามามีบทบาทในชั้นเรียนเป็นอย่างมาก เช่น เป็นสื่อในการเรียนการสอน เป็นเครื่องมือที่ใช้ในการศึกษาค้นคว้า และช่วยในการดูแลและควบคุมระบบการศึกษาของห้องเรียน ซึ่งระบบที่ช่วยในการดูแลและควบคุมระบบการศึกษาในชั้นเรียนนั้นมีบทบาทมาก เช่น ช่วยสังเกตว่าคอมพิวเตอร์แต่ละเครื่องในระบบกำลังทำอะไรอยู่ ส่ง file งานต่างๆ จากเครื่องแม่ไปยังแต่ละเครื่องในระบบ และช่วยในการจัดการสนทนา ระหว่างเครื่องแม่และเครื่องอื่นๆ ภายในระบบ

และจากการที่ผู้จัดทำได้มีประสบการณ์การเรียนภายในห้องที่มีระบบที่ช่วยในการดูแลนั้นพบว่า ระบบที่ใช้อยู่ยังขาดความสามารถบางอย่างที่จะช่วยในการเรียนการสอนได้ คือ การทำงานเป็นกลุ่ม เมื่อมีการจัดกลุ่มใหม่สมาชิกแต่ละคนต้องย้ายจากเครื่องประจำของตน เพื่อไปสนทนากับสมาชิกภายในกลุ่ม และบางกรณี ต้องมีการใช้ file ที่อยู่ในเครื่องเก่าของแต่ละคน จะทำให้เกิดความวุ่นวายขึ้น และระบบที่ใช้อยู่ไม่สามารถช่วยจัดการในเรื่องนี้ได้ ผู้จัดทำจึงเล็งเห็นว่าควรพัฒนาให้มีระบบที่ช่วยจัดการดูแลในส่วนนี้คือ จัดให้มีการสนทนาแบบเป็นกลุ่มได้โดยได้รับการยินยอมจากเครื่องแม่ เพื่อที่จะทำงานเป็นกลุ่มได้โดยไม่ต้องมีการย้ายจากเครื่องประจำ และลดความวุ่นวายดังกล่าว และในการส่งงานบางครั้ง อาจจะต้องให้เครื่องลูกเข้าไปเครื่องแม่ เพื่อเอางานของตัวเองไปวางไว้ในโฟลเดอร์ที่จัดไว้ นั้นเนื่องด้วยข้อจำกัดของ Window XP จะอนุญาตให้เครื่องต่าง ๆ เข้าไปยังเครื่องแม่พร้อมกันได้สูงสุด 10 เครื่อง หากต้องการส่งงานพร้อมกันมากกว่า 10 เครื่อง อาจจะทำให้เกิดความล่าช้าได้ และถ้าเป็นการทำงานหรือการสอบนั้น การจะส่งงานต้องส่งให้ทันเวลา หากเกิดเหตุการณ์ที่ต้องรอกันเพื่อจะเข้าไปส่งงาน ก็อาจทำให้ไม่สามารถส่งงานได้ทันเวลา ระบบที่ใช้อยู่ไม่สามารถช่วยจัดการในเรื่องนี้ได้ ทางผู้จัดทำจึงพัฒนาระบบที่จัดการดูแลในส่วนนี้คือ จัดให้มีตัวช่วยที่จะส่งงานโดยไม่ต้องให้เข้าไปวางงานในเครื่องแม่ เพื่อช่วยลดเวลาในการส่งและปัญหาที่เกิดจากการส่งงานไม่ทัน

ผู้จัดทำได้พัฒนาระบบช่วยการเรียนการสอนด้วยภาษาจาวา (Java) โดยมีเพิ่มความสามารถจาก open source ที่มีชื่อว่า UltraVnc ซึ่งเป็น open source ที่สามารถให้เครื่อง client ดูหน้าจอของ server ส่งไฟล์ และสนทนากันในระบบได้

1.2 วัตถุประสงค์ของโครงการ

1. เพื่อศึกษาหลักการทำงานของระบบช่วยในการเรียนการสอน
2. เพื่อพัฒนาระบบช่วยในการเรียนการสอนที่สามารถจัดการปัญหาที่พบได้

1.3 ขอบข่ายของโครงการ

1. ประยุกต์ใช้ภาษา Java เพื่อพัฒนา open source UltraVnc เพื่อพัฒนาระบบช่วยในการเรียนการสอน
2. ประยุกต์การเขียนโปรแกรมผ่านระบบเครือข่าย เพื่อพัฒนาระบบช่วยในการเรียนการสอน

1.4 ขั้นตอนการดำเนินงาน

1. ศึกษาทฤษฎีและหลักการที่เกี่ยวข้องกับโครงการ
 - ศึกษาการทำงานของ VNC
 - เลือก Tool เพื่อนำมาใช้พัฒนาโปรแกรม และศึกษาวิธีการใช้งาน
 - ศึกษารูปแบบการเขียนโปรแกรมภาษาจาวา
2. ออกแบบโปรแกรม
3. เขียนโปรแกรม
4. ทดสอบและปรับปรุงโปรแกรมให้สมบูรณ์และเหมาะสม
5. เขียนรายงานและจัดทำรูปเล่ม

1.5 แผนการดำเนินงาน

กิจกรรม	ปี	ปี 2550										
	2549	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.
ศึกษาทฤษฎีและ หลักการ ที่เกี่ยวข้อง กับโครงการ		←→										
ออกแบบโปรแกรม		←→										
เขียนโปรแกรม					←→							
ทดสอบ และปรับปรุงโปรแกรม ให้สมบูรณ์และ เหมาะสม						←→						
เขียนรายงาน และจัดทำ รูปเล่ม										←→		

1.6 ผลที่คาดว่าจะได้รับ

1. ได้เห็นถึงหลักการทำงานของระบบช่วยในการเรียนการสอน
2. ได้ซอฟต์แวร์ ที่มีความสามารถในการควบคุมระบบคอมพิวเตอร์เพื่อช่วยในการเรียนการสอนได้

1.7 งบประมาณของโครงการ

1. ค่าถ่ายเอกสาร	700 บาท
2. ค่ากระดาษ	500 บาท
3. ค่าหมึกพิมพ์	850 บาท
4. ค่าเช่าเล่มโครงการ	950 บาท
รวมเป็นเงิน	3,000 บาท (สามพันบาทถ้วน)

หมายเหตุ ขอดำเนินการทุกรายการ

บทที่ 2

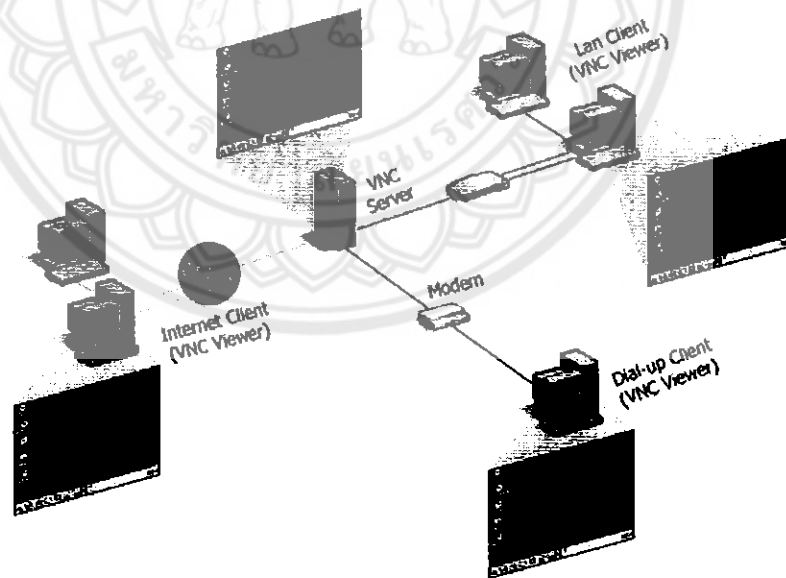
หลักการและทฤษฎีที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงหลักการและทฤษฎีต่างๆ ที่นำมาประยุกต์ใช้เพื่อทำโครงการนี้ ซึ่งทฤษฎีและหลักการที่ได้ใช้ ได้แก่ VNC (Virtual Network Computing) Network Programming, TCP, UDP, Port, Socket, Scope of Multicast IP และหลักการการทำงานของ Stream

2.1 VNC (Virtual Network Computing)

VNC เป็น graphical (GUI) desktop sharing system ที่ใช้ RFB (Remote Frame Buffer) Protocol เพื่อควบคุมคอมพิวเตอร์อื่นจากระยะไกล โดยส่ง การกระทำ ของ mouse และ keyboard จากเครื่องหนึ่งไปยังเครื่องต่างๆ

VNC มี 2 ส่วน client และ server ซึ่ง server เป็นโปรแกรมที่อยู่บนเครื่องที่ต้องการจะ shares หน้าจอของตัวเอง client (หรือ viewer) เป็น program ที่จะเข้าไปดูและ ควบคุมการทำงานของ server



รูปที่ 2.1 แบบจำลอง VNC

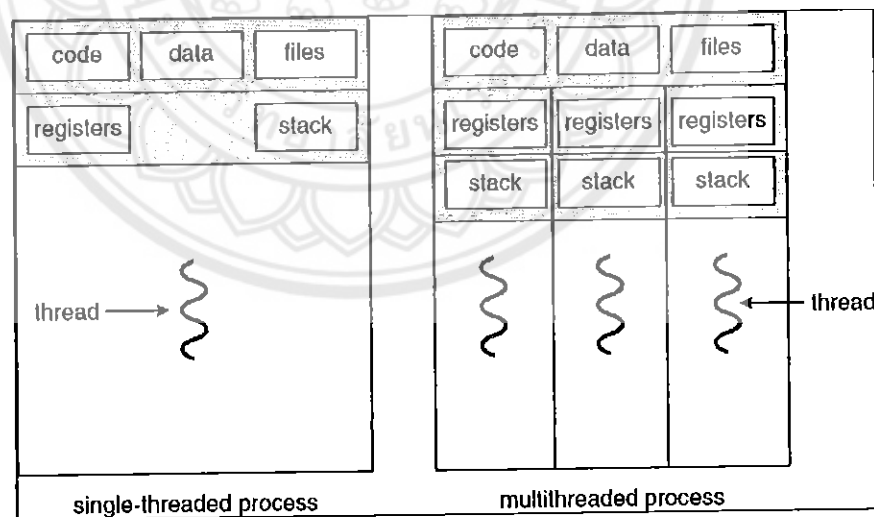
2.2 Network Programming

Network Programming คือ การเขียนโปรแกรม เพื่อให้คอมพิวเตอร์ในเครือข่ายสามารถติดต่อกันได้ด้วย protocol โดยที่ protocol คือกฎเกณฑ์ที่กำหนดขั้นตอนในการติดต่อ แลกเปลี่ยนข้อมูลระหว่างกัน รวมทั้งรูปแบบของข้อมูลที่ส่งไปมา

ในการนำส่งข้อมูลจากเครื่องหนึ่ง ไปสู่อีกเครื่องหนึ่ง ระบบเครือข่ายจะส่งข้อมูลนั้นออกไปเป็นส่วนย่อย ๆ ทีละส่วนเรียกว่า packet ด้านเครื่องที่เป็นผู้ส่งจะส่ง packet ผ่านชั้น (layer) ของโปรแกรมหลายชั้น เพื่อทำการเพิ่มเติมและปรับเปลี่ยนข้อมูลให้ packet นั้นสามารถเดินทางผ่านตัวกลางไปสู่ผู้รับได้ ซึ่งอาจอยู่ในรูปสัญญาณไฟฟ้า แสง หรือคลื่นแม่เหล็กไฟฟ้า ขึ้นกับตัวกลาง เมื่อไปถึงเครื่องที่เป็นผู้รับแล้วฮาร์ดแวร์ที่เป็น protocol ชั้นล่างสุด ได้ข้อมูลจากตัวกลางมาแล้ว ก็จะส่งข้อมูลนั้นผ่านชั้นของโปรแกรมหลายชั้น เพื่อทำการเปลี่ยนคืนกลับเป็น packet ดั้งเดิม

2.2.1 Thread

Thread เป็นส่วนประกอบของโปรเซส ถ้าในหนึ่งโปรเซสมี thread ควบคุมเพียง thread เดียว จะเรียกว่า Single Thread แต่ถ้าในหนึ่งโปรเซสประกอบด้วย thread หลายๆ ตัว จะเรียกว่า Multithreaded ซึ่ง thread ที่อยู่ในโปรเซสเดียวกันจะสามารถใช้โค้ด, ข้อมูล และทรัพยากรของระบบร่วมกันได้



รูปที่ 2.2 การใช้ข้อมูลร่วมกันของ thread ที่อยู่ในโปรเซสเดียวกัน [1]

ภายใน Thread แต่ละตัวจะประกอบไปด้วย

1. Thread ID เป็นหมายเลขของ Thread ในโปรเซส
2. Program Counter ใช้นับจำนวนคำสั่ง เพื่อระบุตำแหน่งของคำสั่งที่กำลังจะประมวลผลในลำดับต่อไป

3. Register Set ใช้เก็บค่าที่ทำงานอยู่
4. Stack ใช้เก็บประวัติการประมวลผล

2.2.2 Thread ในภาษาจาวา

การสร้าง thread ในภาษาจาวามี 2 ลักษณะ คือ

1. การสร้างโดยใช้ Thread Class จะมี method สำหรับควบคุมการทำงานของ thread คือ start(), yield(), sleep(), suspend(), resume(), stop(), run() เป็นต้น เพื่อกำหนดสถานะและควบคุมการทำงานของ thread

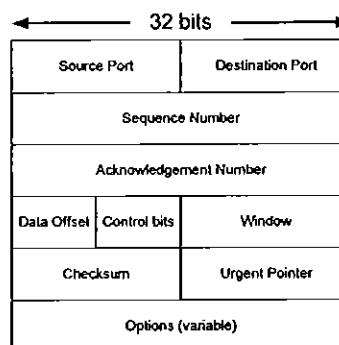
2. การสร้างโดยใช้ Runnable Interface จะมีเพียง method สำหรับควบคุมการทำงานของ thread เพียง method เดียวเท่านั้น นั่นคือ run() และ class ที่ได้ implement Runnable Interface ไปใช้งานนั้น จำเป็นต้องมี method run() เสมอ เพื่อระบุขั้นตอนการทำงานให้กับ thread เมื่ออยู่ในสถานะ running

เนื่องจากภาษาจาวาไม่สามารถ extends class ได้มากกว่า 1 class จึงจำเป็นต้องมีการสร้าง thread ในลักษณะที่ 2 ขึ้น เพื่อให้ class ที่ไม่สามารถ extends class อื่นได้ สามารถเรียกใช้งาน thread ได้

2.3 TCP

TCP (Transmission Control Protocol) คือ การติดต่อที่มีการกำหนดช่วงการสื่อสารตลอดระยะเวลาการสื่อสาร (connection-oriented) ซึ่งจะยอมให้มีการส่งข้อมูลเป็นแบบ Byte stream โดยไม่มีข้อผิดพลาด ข้อมูลที่มีปริมาณมากจะถูกแบ่งออกเป็นส่วนเล็กๆ เรียกว่า message ซึ่งจะถูกส่งไปยังผู้รับผ่านทางชั้นสื่อสารของอินเทอร์เน็ต ทางฝ่ายผู้รับจะนำ message มาเรียงต่อกันตามลำดับเป็นข้อมูลตัวเดิม TCP มีความสามารถในการควบคุมการไหลของข้อมูล (flow control) เพื่อป้องกันไม่ให้ผู้ส่ง ส่งข้อมูลเร็วเกินกว่าที่ผู้รับจะทำงานได้ TCP มีกลไกควบคุมการรับส่งข้อมูลให้มีความถูกต้อง (reliable) และมีการสื่อสารอย่างเป็นทางการ (connection-oriented)

TCP Header



รูปที่ 2.3 โครงสร้างของ TCP Header[2]

มีรายละเอียด ดังนี้

- Source Port : พอร์ตต้นทางที่ส่ง datagram นี้
- Destination Port : พอร์ตปลายทางที่จะเป็นผู้รับ datagram
- Sequence Number : ฟิวด์ที่ระบุหมายเลขลำดับอ้างอิงในการสื่อสารข้อมูลแต่ละครั้ง เพื่อใช้ในการแยกแยะว่าเป็นข้อมูลของชุดใด และนำมาจัดลำดับได้ถูกต้อง
- Acknowledgment Number : ทำหน้าที่เช่นเดียวกับ Sequence Number แต่จะใช้ในการตอบรับ

2.4 UDP

UDP (User Datagram Protocol) คือ การติดต่อแบบไม่ต่อเนื่อง (connectionless) มีการตรวจสอบความถูกต้องของข้อมูลแต่จะไม่มีการแจ้งกลับไปยังผู้ส่ง จึงถือได้ว่าไม่มีการตรวจสอบความถูกต้องของข้อมูล UDP มีข้อดีในด้านความรวดเร็วในการส่งข้อมูล ใช้ทรัพยากรน้อยกว่า TCP แต่อาจจะมีข้อมูลบาง packets สูญหายไประหว่างทาง และ packets ที่ไปถึงจุดหมายอาจจะไม่เป็นที่ตามลำดับการส่งออกไปได้ UDP มีประโยชน์ในการส่งข้อมูลที่ไม่มีความสำคัญถึงขั้นทุก packets จะสูญหายไม่ได้ แต่มีประสิทธิภาพความเร็วในการส่งข้อมูลสามารถส่งข้อมูลได้ทั้งแบบ unicast, multicast และ broadcast จึงนิยมใช้ในระบบผู้ให้และผู้ให้บริการ (client/server system) ซึ่งมีการสื่อสารแบบ ถาม/ตอบ (request/reply) โดยการส่งข้อมูลของ UDP นั้นจะเป็นการส่งครั้งละ 1 ชุดข้อมูล เรียกว่า UDP datagram ซึ่งจะไม่มีความสัมพันธ์กันระหว่าง datagram และจะไม่มีกลไกการตรวจสอบความสำเร็จในการรับส่งข้อมูล

กลไกการตรวจสอบโดย checksum ของ UDP นั้นเพื่อเป็นการป้องกันข้อมูลที่จะถูกแก้ไข หรือมีความผิดพลาดระหว่างการส่ง และหากเกิดเหตุการณ์ดังกล่าว ปลายทางจะรู้ว่ามีการผิดพลาดเกิดขึ้น แต่มันจะเป็นการตรวจสอบเพียงฝ่ายเดียวเท่านั้น โดยในข้อกำหนดของ UDP หากพบว่า Checksum Error ก็ให้ผู้รับปลายทางทำการทิ้งข้อมูลนั้น แต่จะไม่มีการแจ้งกลับไปยังผู้ส่งแต่อย่างใด การรับส่งข้อมูลแต่ละครั้งหากเกิดข้อผิดพลาดในระดับ IP เช่น ส่งไม่ถึง หมดเวลา ผู้ส่งจะได้รับ Error Message จากระดับ IP เป็น ICMP Error Message แต่เมื่อข้อมูลส่งถึงปลายทางถูกต้อง แต่เกิดข้อผิดพลาดในส่วนของ UDP เอง จะไม่มีการยืนยัน หรือแจ้งให้ผู้ส่งทราบ

UDP Header

Source Port	Destination Port
UDP Length	Checksum
Data (bytes)	

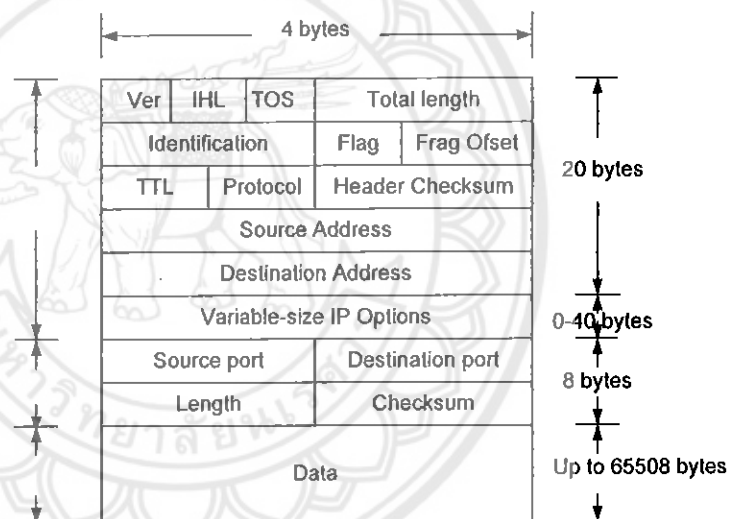
รูปที่ 2.4 UDP Header[3]

มีรายละเอียดดังนี้

- Source Port : พอร์ตต้นทางที่ส่ง datagram นี้
- Destination Port : พอร์ตปลายทางที่จะเป็นผู้รับ datagram
- UDP Length : ความยาวของ datagram ทั้งส่วน Header และ data นั้นหมายความว่าค่าที่น้อยที่สุดในฟิลด์นี้คือ 8 ซึ่งเป็นขนาดของ Header
- Checksum : เป็นตัวตรวจสอบความถูกต้องของ UDP datagram และจะนำข้อมูลบางส่วนใน IP Header มาคำนวณด้วย

UDP datagram

UDP datagram มีโครงสร้างดังนี้



รูปที่ 2.5 โครงสร้างของ UDP datagram[2]

2.5 Ports

สำหรับพวก Application ในชั้น layer สูงๆ ที่ใช้ TCP (Transmission Control Protocol) หรือ UDP (User Datagram Protocol) จะมีหมายเลข Port หมายเลขของ Port จะเป็นเลข 16 bit เริ่มตั้งแต่ 0 ถึง 65535 หมายเลข Port ใช้สำหรับตัดสินว่า service ใดที่ต้องการเรียกใช้ ในทางทฤษฎีหมายเลข Port แต่ละหมายเลขถูกเลือกสำหรับ service ใดๆ ขึ้นอยู่กับ OS (operating system) ที่ใช้ ไม่จำเป็นต้องเหมือนกัน แต่ได้มีการกำหนดขึ้นให้ใช้ค่อนข้างเป็นมาตรฐานเพื่อให้มีการติดต่อการส่งข้อมูลที่ดีขึ้น ทาง Internet Assigned Numbers Authority (IANA) เป็นหน่วยงานกลางในการประสานการเลือกใช้ Port ว่า Port หมายเลขใดควรเหมาะสมสำหรับ Service ใด และได้กำหนดใน

Request For Comments (RFC') 1700 ตัวอย่างเช่น เลือกใช้ TCP Port หมายเลข 23 กับ Service Telnet และเลือกใช้ UDP Port หมายเลข 69 สำหรับ Service Trivial File transfer Protocol (TFTP)

หมายเลข Port ถูกจัดแบ่งเป็น 2 ประเภท ตามที่ได้กำหนดใน RFC' 1700 คือ well known Ports และ Registered Ports

- Well Known Ports คือจะเป็น Port ที่ระบบส่วนใหญ่ กำหนดให้ใช้โดย Privileged User (ผู้ใช้ที่มีสิทธิพิเศษ) โดย port เหล่านี้ ใช้สำหรับการติดต่อระหว่างเครื่องที่มีระบบเวลาที่ยาวนาน วัตถุประสงค์เพื่อให้ service แก่ผู้ใช้ (ที่ไม่รู้จักหรือคุ้นเคย) แลกหน้า จึงจำเป็นต้องกำหนด Port ติดต่อกับ Service นั้นๆ
- Registered Ports จะเป็น Port หมายเลข 1024 ขึ้นไป ซึ่ง IANA ไม่ได้กำหนดไว้

ตารางที่ 2.1 ตัวอย่าง Well Known Ports[2]

Protocol	Description	Protocol: Port
FTP	File transfer	tcp : 21
TELNET	Remote console access	tcp : 23
SMTP	Mail transfer	tcp : 25
SNMP	Network management	udp : 161
DNS	Domain name system	udp : 53, tcp : 53
HTTP	Hypertext transfer	Tcp : 80

การใช้ Port ในการติดต่อด้วย TCP

มี 2 วิธี คือ

1. Passive Ports คือ การติดต่อที่ Application process สั่งให้ TCP รอหมายเลข Port สำหรับการร้องขอการติดต่อจาก Source Host เมื่อ TCP ได้รับการร้องขอแล้วจึงทำการเลือกหมายเลข Port ให้
2. Active Ports คือ การให้ Application process เป็นฝ่ายเลือกหมายเลข Port ให้เลย

2.6 Socket

Socket คือ กลุ่มของหมายเลข Port และ หมายเลข IP ในการใช้งานจะต้องมี Socket สำหรับต้นทาง และ Socket สำหรับปลายทาง Socket ใช้ในการเขียน โปรแกรมส่งข้อมูลผ่านระบบเครือข่าย โดยที่ socket ซ่อนรายละเอียดเกี่ยวกับ packet รวมทั้งวิธีการติดต่อกับระบบเครือข่ายไป

จากผู้เขียนโปรแกรมแล้ว ให้ผู้เขียนโปรแกรม สามารถเขียนโปรแกรมส่งและรับข้อมูลผ่านระบบเครือข่ายได้เหมือนกับการเขียนและอ่านข้อมูลจาก streams

ใน package java.net มีคลาส Socket สำหรับสร้าง socket ซึ่งมีกลไกสำหรับ

- สร้างการติดต่อไปสู่เครื่องเป้าหมายที่ port หนึ่ง โดยเครื่องเป้าหมายอาจเป็นผู้รับหรือส่งก็ได้
- สร้าง streams สำหรับอ่านหรือเขียนข้อมูลไปที่เครื่องเป้าหมาย
- ปิดการติดต่อไปสู่เครื่องเป้าหมาย

คลาส Socket มี constructors หลายตัว เช่น

```
public Socket (String, int) throws UnknownHostException, IOException;
```

```
public Socket (InetAddress, int) throws IOException;
```

2.7 Scope of Multicast IP

การกำหนดขอบข่ายของ Multicast IP (การส่ง packet เดียวที่ระบุ IP แบบเป็นกลุ่มไว้ ซึ่งจะทำให้ packet นั้นกระจายออกไปทุกๆเครื่องในกลุ่มที่ต้องการ) ทำได้ 2 วิธี คือ กำหนด TTL (Time to live) และ Administrative Scoping

TTL (Time to live)

คือ อายุของ packet ที่ทำการส่งผ่านระบบเน็ตเวิร์ค โดยจำนวน TTL จะลดลงทีละ 1 ทุกครั้งที่ข้อมูลวิ่งผ่าน router ถ้าหากค่า TTL เหลือ 0 ก่อนที่จะถึงจุดหมาย ข้อมูลชุดนั้นจะไม่ถูกส่งต่อ

ตารางที่ 2.2 TTL Scoping[2]

TTL	Scoping Restriction
= 0	Local host
= 1	Local subnet
<= 32	Site (organization)
<= 64	Region
<= 128	Continent
> 128 (<= 255)	Unrestricted

Administrative Scoping

Administrative Scoping คือการกำหนดขอบข่ายด้วยการระบุ IP โดย administrative

ตารางที่ 2.3 Administrative Scoping

IP Range	Scoping Restriction
224.0.0.0 – 224.0.0.255	สำหรับโปรโตคอลของการ routing
224.0.1.0 – 224.0.1.255	สำหรับโปรโตคอลของแอปพลิเคชัน
224.0.2.0 – 238.255.255.255	สำหรับ companies และ application(global)
239.192.0.0 – 239.251.255.255	สำหรับใช้ภายในองค์กร
239.255.0.0 – 239.255.255.255	Local-scope

2.8 Stream

แนวความคิดของภาษาจาวานั้น ในการอ่านเข้ามาหรือส่งค่าออกไปของ stream ไม่ว่าจะ เป็นอุปกรณ์ภายนอกชนิดใดหรือข้อมูลแบบใดก็ตาม ลักษณะหรือรูปแบบของลำดับการเขียน โปรแกรมโดยทั่วไปนั้นจะเป็นแบบเดียวกันเสมอนั้นก็คือ การเปิด stream หรือสะพานเชื่อมโยง ข้อมูลเพื่อทำการถ่ายโอนข้อมูล จากนั้นก็ตรวจสอบว่าข้อมูลดังกล่าวได้ถูกเปิดหรืออ่าน หมดแล้วหรือยัง ถ้ายังก็จะทำการอ่านจนกว่าจะหมด แล้วจึงทำการปิดสะพานหรือ stream นั้น

และเมื่อใดก็ตามที่มีการทำงานเกี่ยวข้องกับ I/O สิ่งที่เราควรระลึกถึงอยู่เสมอก็คือ ข้อมูลที่ เกี่ยวข้องนั้นจะถูกแบ่งออกเป็นสองกลุ่มใหญ่ๆ คือ ข้อมูลที่อยู่ในรูปแบบที่เครื่องอ่านออกเท่านั้น (Machine-formatted data) และข้อมูลที่เป็นภาษามนุษย์ (Human-readable data) ซึ่งสองชนิดของ stream ที่ภาษาจาวาใช้สำหรับข้อมูลสองชนิดดังกล่าวคือ byte stream และ character stream

การจัดการข้อมูลด้วย Stream

คลาส stream ถูกกำหนดเอาไว้ใน Package java.io ที่ยังมีคลาสอื่น ๆ ที่รองรับการทำงานกับ stream อีก ซึ่งเหมือนกับการเขียนโปรแกรม ว่าถ้าต้องการใช้คลาสที่กำหนดใน package ก็ให้ทำการ Import ไว้ที่ด้านบนสุดของโปรแกรม ดังนี้

```
import java.io.*;
```

การอ่าน stream

การอ่าน stream ของออบเจกต์ที่สร้างจากคลาส InputStream และ Reader นั้นใช้เมธอด เดียวกันนั่นคือ เมธอด read() ซึ่งเป็นการอ่านข้อมูลที่ละไบต์ถัดไปจาก stream โดยจะคืนค่าเป็น จำนวนเต็มบวกที่มีค่าตั้งแต่ 0 – 255 และจะคืนค่าเป็น -1 ก็ต่อเมื่อมาถึงไบต์สุดท้ายของ stream (นั่น คือไม่มีไบต์ถัดไป) แต่ถ้าหากเกิด error ขึ้นก็จะทำการแสดงค่า error จาก IOException หรือที่ เรียกว่าเป็นการ Throw Exception

การเขียน Stream

เมื่อเราได้ stream หนึ่ง ๆ ที่อ่านได้มาจากการอ่านที่ผ่านมาหรือถูกกำหนดค่าด้วยโปรแกรมเอง การนำค่าของ stream นั้น ๆ จะมีการต้องเขียน stream โดยเมธอดที่ใช้คือ write ()

เมธอด write () นี้จะทำการแปลงค่า int ที่อ่านมาจาก stream ให้เป็นตัวอักษรทันที นอกจากนั้นทั้งสามรูปแบบด้านบนนั้นมีวิธีใช้ที่คล้ายกับเมธอด read () มาก เนื่องจากบางครั้งเมื่อ stream ถูกอ่านก็จะถูกนำมาเขียนทันที ดังนั้น Stream เมื่อถูกอ่านมาอย่างไรก็อาจจะต้องถูกเขียนออกมาด้วยวิธีเดียวกัน เป็นต้น

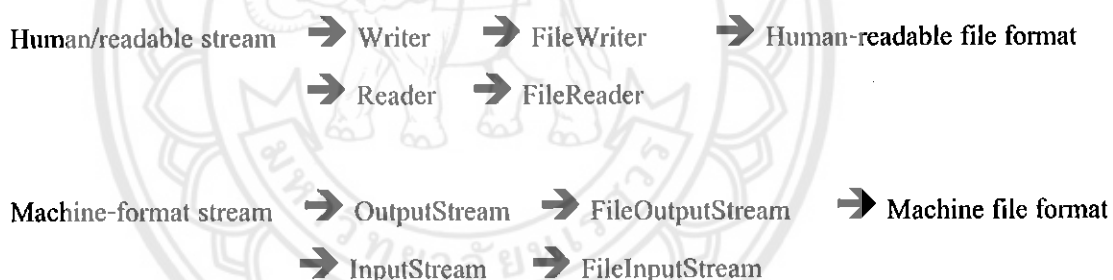
การทำงานกับแฟ้มข้อมูล

File Stream

File Stream ประกอบด้วยคลาสย่อยดังนี้

- FileReader , FileWriter >> ถูกใช้สำหรับการสร้าง Stream เพื่อการอ่านและเขียนลงไฟล์แบบใช้ภาษามนุษย์
- FileInputStream , FileOutputStream >> ถูกใช้สำหรับการสร้าง Stream เพื่อการอ่านและเขียนลงไฟล์แบบภาษาเครื่อง

อธิบายที่มาของ File Stream ทั้ง 4 ชนิด ดังนี้



การสร้าง File Stream

สร้าง stream ขึ้นมาใช้งาน โดยการสร้าง object ขึ้นมาจากคลาส File ซึ่ง Constructor ของมันจะใช้พารามิเตอร์ String ของชื่อไฟล์ที่ต้องการสร้าง FileStream

การอ่าน File Stream

ไม่ว่าจะเป็นการอ่านแบบใช้ Reader หรือ InputStream ต่างก็ให้ค่าผลลัพธ์จากการอ่านเหมือน ๆ กัน การสร้าง File Stream สำหรับการอ่านนั้น ชื่อไฟล์ที่ระบุจะต้องมีอยู่จริงเนื่องจากเราต้องการอ่านข้อมูลขึ้นมาและที่สำคัญจะต้องไม่ลืมว่า การใช้คลาส File ต่าง ๆ นี้จะต้องอยู่ภายใต้การดักจับ Error คือ IOException เนื่องจากถ้าไฟล์ที่ต้องการเปิดอ่านนั้นไม่มีอยู่จริง IOException จะทำหน้าที่จับความผิดพลาดนี้และแสดงผลออกทางหน้าจอว่าเกิดความผิดพลาดของโปรแกรมขึ้นจากสาเหตุใด (ไม่มีไฟล์อยู่, มีไฟล์อยู่แต่ไม่สามารถเปิดได้ เป็นต้น)[5]

บทที่ 3

วิธีการดำเนินงาน

ในบทนี้จะเป็นกล่าวถึงขั้นตอนการทำงาน เพื่อให้ได้มาซึ่งซอฟต์แวร์ช่วยควบคุมการเรียนการสอน (RCL System) โดยการทำงานนั้นจะเริ่มจากการกำหนดความสามารถของซอฟต์แวร์ (โดยการกำหนดความสามารถของซอฟต์แวร์นั้น จะกำหนดตามที่มาและความสำคัญเพื่อที่จะสามารถแก้ปัญหาดังที่กล่าวมาได้) เพื่อที่จะสามารถกำหนดขอบเขตของการพัฒนาซอฟต์แวร์และเมื่อได้ขอบเขตของการพัฒนาซอฟต์แวร์แล้วจะเป็นการออกแบบสถาปัตยกรรมของระบบ (System Architecture) และแยกส่วนต่างๆของซอฟต์แวร์เข้าสู่กระบวนการพัฒนาซอฟต์แวร์และนำกลับมาพร้อมกับส่วนอื่นๆเพื่อให้กลับกลายเป็นซอฟต์แวร์ที่ออกแบบไว้

จากการออกแบบซอฟต์แวร์นั้น ซอฟต์แวร์ช่วยควบคุมการเรียนการสอนภายในห้องเรียนนั้นจะถูกแบ่งแยกออกเป็นซอฟต์แวร์ 2 ฟังก์ชันที่มีการติดต่อกันแบบ เครื่องแม่ข่าย – เครื่องลูกข่าย (Server – Client) ซึ่งซอฟต์แวร์แต่ละฟังก์ชันก็จะประกอบด้วยซอฟต์แวร์ย่อยที่มีความสามารถต่างๆ ซึ่ง ซอฟต์แวร์ย่อยนี้การทำงานก็จะสอดคล้องกับซอฟต์แวร์ย่อยของอีกฟังก์ชัน

ซอฟต์แวร์แต่ละฟังก์ชันจะประกอบไปด้วยซอฟต์แวร์ย่อยคือ 1.ซอฟต์แวร์การสร้างห้องสนทนา (Chat Room), 2. ซอฟต์แวร์การรับส่งไฟล์ (File Server) 3. ซอฟต์แวร์การรับส่งไฟล์แบบกระจาย (Broadcast File) และ 4. ซอฟต์แวร์การดูหน้าจอเครื่องแม่ข่าย (RCL VNC)

3.1 การกำหนดความสามารถของซอฟต์แวร์

3.1.1 ซอฟต์แวร์ช่วยควบคุมการเรียนการสอนภายในห้องเรียนของเครื่องแม่ข่าย

1. การทำงานของซอฟต์แวร์ (Functional)

1. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์การสร้างห้องสนทนาได้
2. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์การรับส่งไฟล์ได้
3. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์การส่งไฟล์แบบกระจายได้
4. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์การดูหน้าจอเครื่องแม่ข่ายได้

2. คุณสมบัติของซอฟต์แวร์ (Non Functional)

1. ซอฟต์แวร์สามารถแสดงรายละเอียดของ Client ที่เปิดใช้ซอฟต์แวร์ช่วยควบคุมการเรียนการสอนของเครื่องลูกข่ายได้
2. ซอฟต์แวร์มีข้อความซึ่งบอกถึงรายละเอียดของซอฟต์แวร์ย่อยต่างๆ

3. ซอฟต์แวร์สามารถส่งคำสั่งต่างๆ ไปยังซอฟต์แวร์ช่วยควบคุมการเรียนการสอนของเครื่องลูกข่ายได้

3. ข้อกำหนดของซอฟต์แวร์ (Specification)

1. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์การสร้างห้องสนทนาเพื่อใช้ในการติดต่อสื่อสารกับเครื่องอื่นๆ ได้
2. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์การรับไฟล์จากซอฟต์แวร์ ของฝั่ง Client ได้โดยที่ลูกข่ายไม่ต้อง Access เข้ามาใน Server ได้
3. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์การส่งไฟล์แบบกระจาย เพื่อส่งไฟล์ที่ต้องการไปยังเครื่อง Client ที่ติดต่อได้
4. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์การดูหน้าจอเครื่องแม่ข่ายได้
5. ซอฟต์แวร์มีส่วนแสดงให้รู้ถึงรายละเอียดของเครื่อง Client ที่ติดต่อเข้ามา
6. ซอฟต์แวร์มีข้อความที่บอกถึงรายละเอียดของซอฟต์แวร์ย่อยแต่ละส่วน
7. ซอฟต์แวร์สามารถส่งคำสั่งต่างๆ ที่ใช้ในการติดต่อกับเครื่อง Client ได้

3.1.2 ซอฟต์แวร์ช่วยควบคุมการเรียนการสอนภายในห้องเรียนของเครื่องลูกข่าย

1. การทำงานของซอฟต์แวร์ (Functional)

1. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์การสร้างห้องสนทนาเพื่อใช้ในการสนทนาได้
2. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์การรับส่งไฟล์เพื่อส่งไฟล์ไปยังเครื่อง Server ได้
3. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์เพื่อ รับ ไฟล์ที่ Server ส่งแบบกระจายได้
4. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์การดูหน้าจอเครื่องแม่ข่ายได้

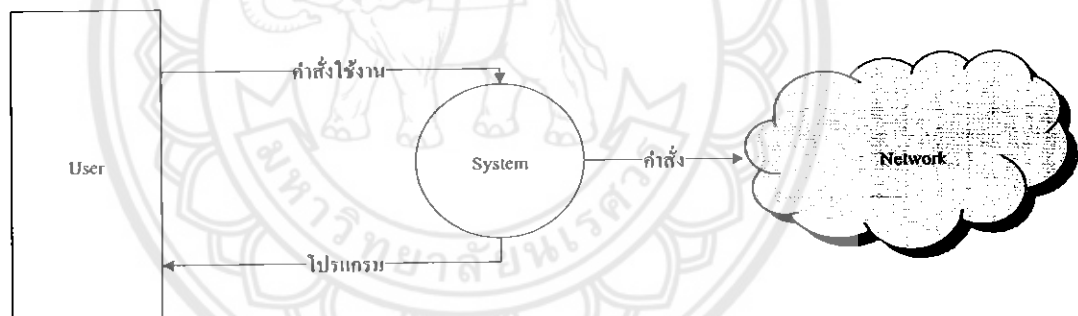
2. คุณสมบัติของซอฟต์แวร์ (Non Functional)

1. ซอฟต์แวร์สามารถค้นหาเครื่องแม่ข่ายได้เองโดยอัตโนมัติ
2. ซอฟต์แวร์สามารถติดต่อรับคำสั่งต่างๆจากเครื่องแม่ข่ายได้
3. ซอฟต์แวร์มีข้อความที่บอกถึงรายละเอียดของ โปรแกรมย่อยแต่ละส่วน

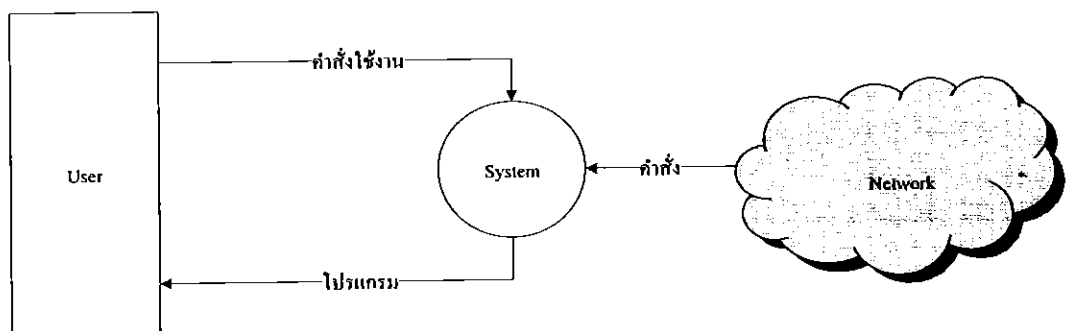
3. ข้อกำหนดของซอฟต์แวร์ (Specification)

1. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์การสร้างห้องสนทนาเพื่อใช้ในการติดต่อสื่อสารกับเครื่องอื่นๆ ได้
2. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์การรับส่งไฟล์ เพื่อส่งไฟล์ที่ต้องการไปยังเครื่อง Server ได้อย่างถูกต้อง
3. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์เพื่อ รับไฟล์ที่เครื่องแม่ข่ายส่งไฟล์แบบกระจายมาได้
4. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์การดูหน้าจอเครื่องแม่ข่ายได้
5. ซอฟต์แวร์สามารถค้นหาเครื่อง Server เพื่อติดต่อได้เองโดยอัตโนมัติ
6. ซอฟต์แวร์สามารถรองรับคำสั่งต่างๆจากซอฟต์แวร์ ของเครื่อง Server ได้
7. ซอฟต์แวร์มีส่วนที่แสดงรายละเอียดของซอฟต์แวร์ย่อยแต่ละส่วน

3.2 การออกแบบโปรแกรม



รูปที่ 3.1 แผนผังระบบงานเชิงธุรกิจ ของซอฟต์แวร์ช่วยควบคุมการเรียนการสอนของเครื่องแม่ข่าย (RCL Server)



รูปที่ 3.2 แผนผังระบบงานเชิงธุรกิจของซอฟต์แวร์ช่วยควบคุมการเรียนการสอนของเครื่องลูกข่าย (RCL Client)

ตารางที่ 3.1 การเปลี่ยนแปลงที่มีผลต่อระบบเชิงธุรกิจของซอฟต์แวร์ช่วยควบคุมการเรียนการสอนของเครื่องแม่ข่าย

เหตุการณ์	การไหล
1. ผู้ใช้ เลือกคำสั่งเรียกใช้งาน Chat Room	(เข้า) คำสั่งใช้งาน
2. ซอฟต์แวร์แสดงโปรแกรม Chat Room	(ออก) โปรแกรม
3. ผู้ใช้ เลือกคำสั่งเรียกใช้งาน File Server	(เข้า) คำสั่งใช้งาน
4. ซอฟต์แวร์ แสดงโปรแกรม File Server	(ออก) โปรแกรม
5. ผู้ใช้ เลือกคำสั่งเรียกใช้งาน Broadcast	(เข้า) คำสั่งใช้งาน
6. ซอฟต์แวร์ แสดงโปรแกรม Broadcast	(ออก) โปรแกรม
7. ผู้ใช้ เลือกคำสั่งเรียกใช้งาน RCL VNC	(เข้า) คำสั่งใช้งาน
8. ซอฟต์แวร์ แสดงสถานะของโปรแกรม RCL VNC	(ออก)สถานะ
9. ซอฟต์แวร์ ส่งคำสั่งโปรแกรม RCL VNC ไปยัง Network	(ออก) คำสั่งใช้งาน

ตารางที่ 3.2 การเปลี่ยนแปลงที่มีผลต่อระบบเชิงธุรกิจของซอฟต์แวร์ช่วยควบคุมการเรียนการสอนของเครื่องลูกข่าย

เหตุการณ์	การไหล
1. ผู้ใช้ เลือกคำสั่งเรียกใช้งาน Chat Room	(เข้า) คำสั่งใช้งาน
2. ซอฟต์แวร์ แสดงโปรแกรม Chat Room	(ออก) โปรแกรม
3. ผู้ใช้ เลือกคำสั่งเรียกใช้งาน File Server	(เข้า) คำสั่งใช้งาน
4. ซอฟต์แวร์ แสดงโปรแกรม File Server	(ออก) โปรแกรม
5. ซอฟต์แวร์ รับคำสั่งเรียกใช้งาน RCL VNC	(เข้า) คำสั่งใช้งาน
6. ซอฟต์แวร์ แสดงโปรแกรม RCL VNC	(ออก) โปรแกรม

3.2.1 กรณีใช้งานเชิงธุรกิจ (Business Use Case)

ชื่อ : การใช้งานซอฟต์แวร์ ช่วยควบคุมการเรียนการสอนฝั่ง Server

คำอธิบาย : ผู้ใช้ซอฟต์แวร์ช่วยควบคุมการเรียนการสอน เลือกใช้คำสั่งการใช้งานต่างๆ

เงื่อนไขเริ่มต้น : ผู้ใช้เปิดโปรแกรมช่วยควบคุมการเรียนการสอนในห้องเรียน

การกระทำพื้นฐาน

1. ซอฟต์แวร์รอการใช้งาน
2. ผู้ใช้เลือกใช้คำสั่งเรียกใช้งาน ซอฟต์แวร์ย่อยอื่นๆ
3. ซอฟต์แวร์เปิดโปรแกรมตามที่ผู้ใช้เลือกใช้งาน
4. กลับไป 1.
5. จบการทำงาน

ชื่อ : : การใช้งานซอฟต์แวร์ ช่วยควบคุมการเรียนการสอนฝั่ง Client

คำอธิบาย : ผู้ใช้ซอฟต์แวร์ช่วยควบคุมการเรียนการสอนเลือกใช้คำสั่งเรียกใช้งานต่างๆ

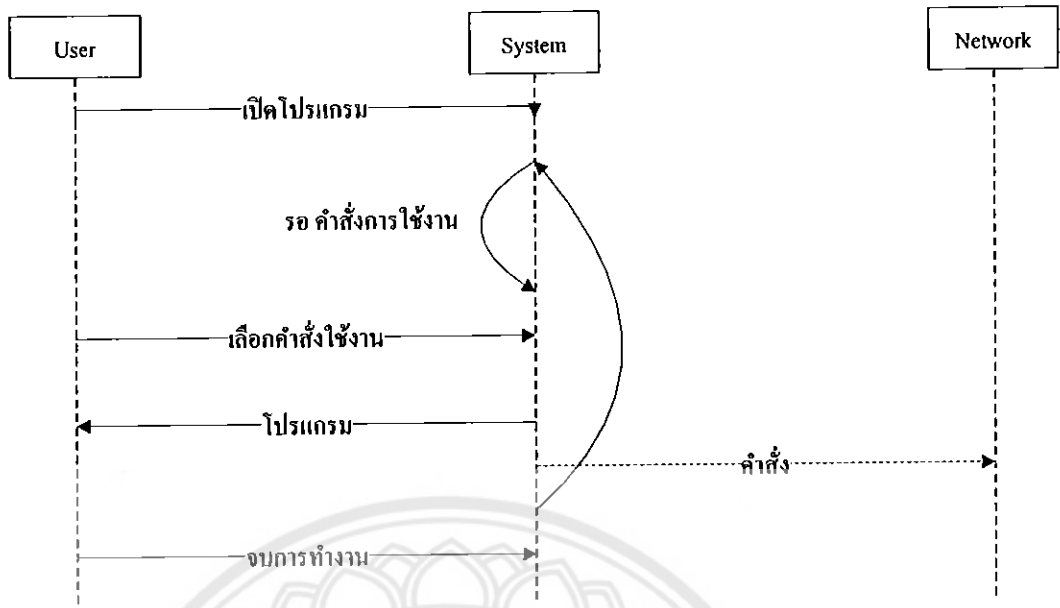
เงื่อนไขเริ่มต้น : ผู้ใช้เปิดโปรแกรมช่วยควบคุมการเรียนการสอนในห้องเรียนฝั่ง Client

การกระทำพื้นฐาน

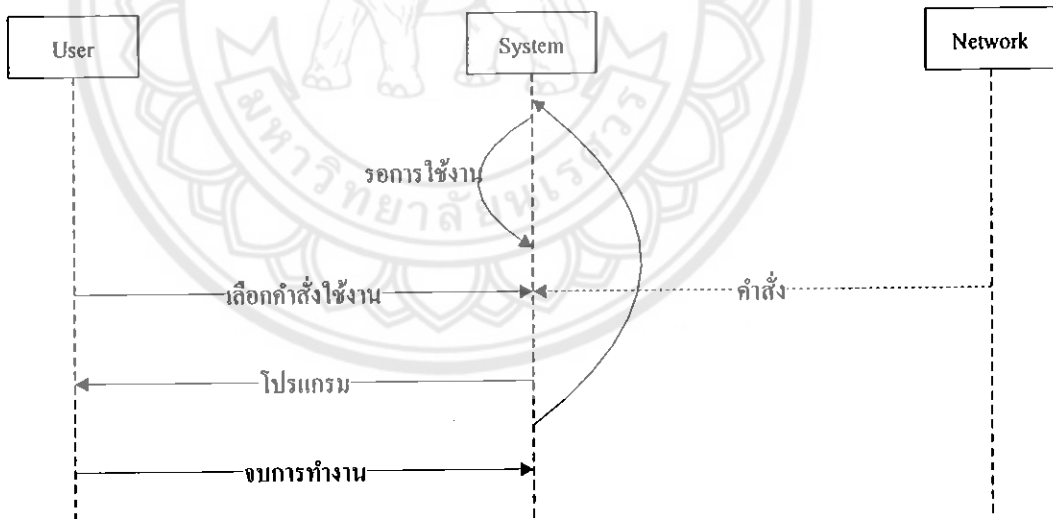
1. ซอฟต์แวร์รอการใช้งานจากผู้ใช้และรอคำสั่งจาก Server
2. ผู้ใช้เลือกใช้คำสั่งเรียกใช้งาน ซอฟต์แวร์ย่อยอื่นๆ หรือรับคำสั่งจาก เรียกใช้งาน RCL

VNC จาก Network

3. ซอฟต์แวร์เปิดโปรแกรมตามที่ผู้ใช้เลือกใช้งาน
4. กลับไป 1.
5. จบการทำงาน

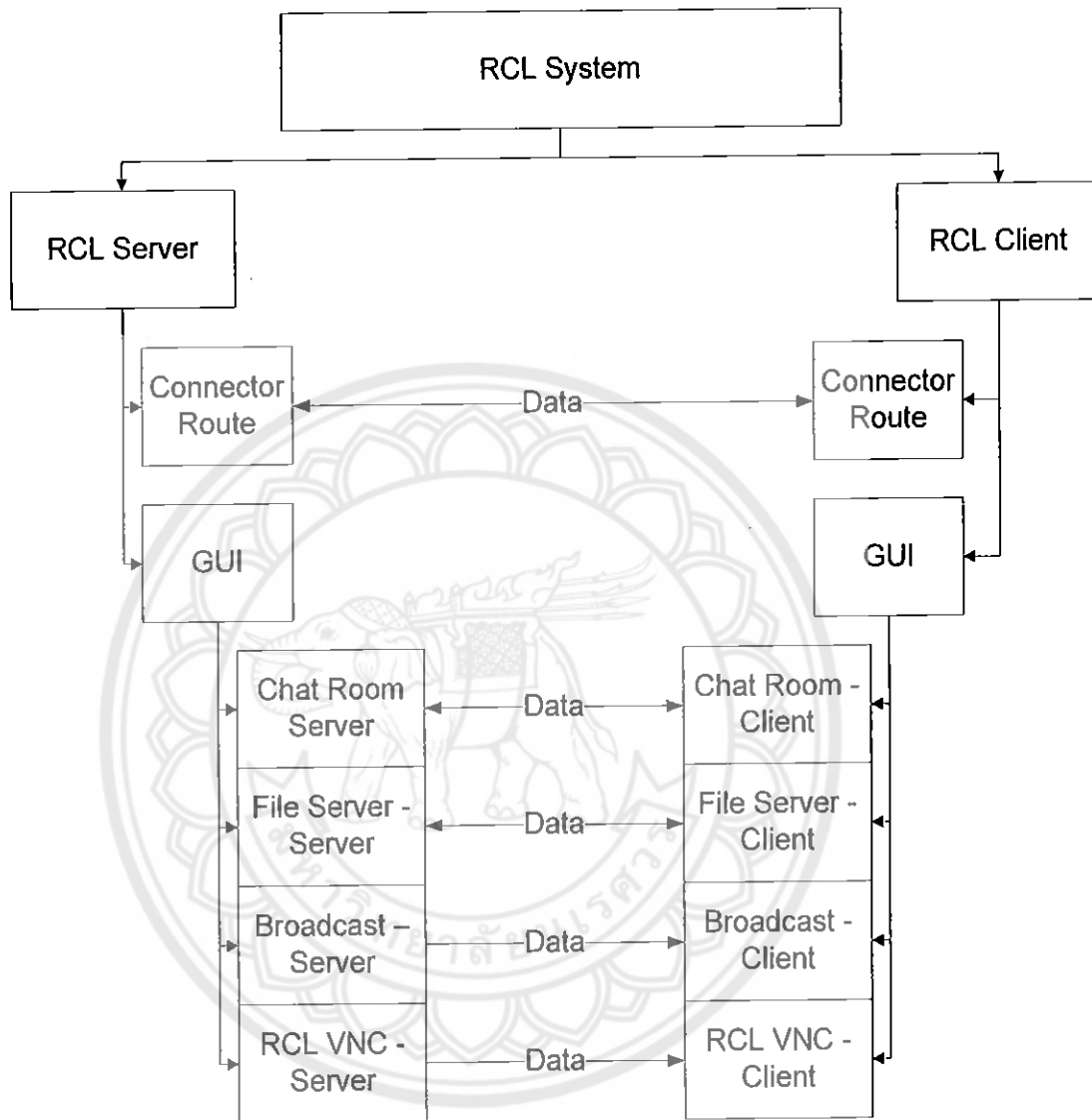


รูปที่ 3.3 แผนผังปฏิสัมพันธ์ตามลำดับ (Sequence diagram) การใช้งานพื้นฐานของ RCL – Server



รูปที่ 3.4 Sequence Diagram การใช้งานพื้นฐานของ RCL Client

3.3 สถาปัตยกรรมของระบบ (System Architecture)



รูปที่ 3.5 System Architecture ของ Server – Client RCL System

ซอฟต์แวร์ช่วยควบคุมการเรียนการสอนนั้น สามารถแบ่งตามหน้าที่การทำงานได้เป็น 2 ส่วนหลักๆ คือ

1. ส่วนของช่องทางการติดต่อ (Connector Route) เป็น ส่วนที่ใช้ในการแลกเปลี่ยนข้อมูลระหว่าง Server และ Client ซึ่ง การติดต่อนี้ จะใช้ Protocol 2 ชนิดคือ TCP และ UDP แต่โดยส่วนมากแล้วจะใช้ UDP ในการส่งคำสั่งต่างๆ ไปยัง Client ส่วน TCP จะใช้ในกรณีที่ต้องการการส่งข้อมูลที่ต้องการความแน่นอนและถูกต้อง
2. ส่วนต่อต่อกับผู้ใช้ (GUI) เป็นส่วนที่ติดต่อกับผู้ใช้และคอยเรียกใช้ซอฟต์แวร์ย่อยตามที่ผู้ใช้เรียกใช้คำสั่ง

ซอฟต์แวร์ช่วยควบคุมการเรียนการสอนในห้องเรียนนั้น จะเป็นซอฟต์แวร์ที่มีซอฟต์แวร์ย่อยๆ 4 อย่างตามที่ได้กำหนดไว้ คือ

1. ซอฟต์แวร์การสร้างห้องสนทนา (Chat Room) เป็นซอฟต์แวร์ที่ใช้สำหรับสร้างห้องสนทนาเพื่อติดต่อพูดคุยกับผู้ใช้อื่นๆ ที่อยู่ภายในห้องสนทนาเดียวกัน
2. ซอฟต์แวร์การรับส่งไฟล์ (File Server) เป็นซอฟต์แวร์ที่ใช้สำหรับให้เครื่องลูกข่ายสามารถส่งไฟล์ที่ต้องการเข้าสู่เครื่องของแม่ข่าย โดยที่ไม่ต้องทำการ Access เข้ามาภายในเครื่องของแม่ข่าย
3. ซอฟต์แวร์การรับส่งไฟล์แบบกระจาย (Broad Cast) เป็นซอฟต์แวร์ที่ใช้สำหรับการส่งไฟล์แบบกระจายไปยังเครื่องลูกข่ายที่ติดต่อ
4. ซอฟต์แวร์ดูหน้าจอเครื่องแม่ข่าย (RCL VNC) เป็นซอฟต์แวร์ที่ใช้สำหรับให้เครื่องลูกข่ายสามารถมองเห็นหน้าจอของเครื่องแม่ข่ายที่หน้าจอของตัวเองได้

3.4 โปรแกรมการสร้างห้องสนทนา (Chat Room)

เป็น โปรแกรมที่ใช้สำหรับการสร้างห้องสนทนาเพื่อใช้ในการติดต่อกันระหว่างเครื่องต่างๆภายในเครือข่าย โดยที่ภายใน 1 ห้องสนทนานั้นสามารถมีผู้สนทนาได้ไม่จำกัดจำนวน โปรแกรมการสร้างห้องสนทนานี้จะแบ่งออกเป็น 2 ฝั่ง คือ ฝั่งแม่ข่าย (Chat Room Server) และฝั่งลูกข่าย (Chat Room Client)

โปรแกรมฝั่ง Server นั้น จะใช้สำหรับสร้างห้องสนทนาตามจำนวนที่ต้องการและรอรับการติดต่อขอเข้าร่วมสนทนาของโปรแกรมฝั่งลูกข่าย หากเครื่อง Server ไม่ได้สร้างห้องสำหรับการสนทนา เครื่อง Client จะไม่สามารถใช้ Chat Room Client ได้

3.4.1 การกำหนดความสามารถของซอฟต์แวร์

3.4.1.1 โปรแกรมการสร้างห้องสนทนาฝั่งแม่ข่าย (Chat Room Server)

1. ความสามารถของซอฟต์แวร์ (Functional)

- 1.1 ซอฟต์แวร์สามารถสร้างห้องสนทนาได้ตามจำนวนที่ต้องการ
- 1.2 ซอฟต์แวร์สามารถรับส่งข้อความสนทนาได้

2. คุณสมบัติของซอฟต์แวร์ (Non – Functional)

- 2.1 ซอฟต์แวร์สามารถรับชื่อผู้ใช้เพื่อใช้ในการสนทนาได้
- 2.2 ซอฟต์แวร์สามารถตอบรับหรือปฏิเสธการติดต่อเพื่อเข้าห้องสนทนาของซอฟต์แวร์ฝั่ง Client ได้
- 2.3 ซอฟต์แวร์สามารถแสดงถึงรายชื่อสมาชิกที่อยู่ภายในห้องสนทนาได้
- 2.4 ซอฟต์แวร์สามารถเรียก Client เข้าห้องสนทนาตามต้องการได้
- 2.5 ซอฟต์แวร์สามารถกำหนดให้สมาชิกภายในห้องสนทนาที่ต้องการออกจากห้องสนทนาได้
- 2.6 ซอฟต์แวร์สามารถส่งไฟล์ที่ต้องการไปยังสมาชิกที่ต้องการได้
- 2.7 ซอฟต์แวร์สามารถกระจายส่งไฟล์ที่ต้องการแบบกระจายไปยังสมาชิกทุกคนภายในห้องสนทนาได้
- 2.8 ซอฟต์แวร์สามารถเพิ่มและลดจำนวนห้องสนทนาได้

3. ข้อกำหนดของซอฟต์แวร์ (Specification)

- 3.1 ซอฟต์แวร์สามารถกำหนดชื่อผู้ใช้ตามที่ผู้ใช้ต้องการเพื่อใช้ในการสนทนาได้
- 3.2 ซอฟต์แวร์สามารถสร้างห้องสนทนาตามจำนวนที่ผู้ใช้ต้องการได้
- 3.3 ซอฟต์แวร์สามารถตอบรับหรือปฏิเสธการขอเข้าร่วมสนทนาของ Client ตามความต้องการของผู้ใช้ได้

- 3.4 ซอฟต์แวร์สามารถแสดงรายชื่อของสมาชิกภายในแต่ละห้องสนทนาได้
- 3.5 ซอฟต์แวร์สามารถเรียก Client ที่ต้องการเข้าสู่ห้องสนทนาได้
- 3.6 ซอฟต์แวร์สามารถกำหนดให้สมาชิกภายในห้องสนทนาที่ต้องการออกจากห้องสนทนาได้ตามต้องการ
- 3.7 ซอฟต์แวร์สามารถส่งไฟล์ที่ต้องการไปยังสมาชิกภายในห้องสนทนาได้ตามต้องการ
- 3.8 ซอฟต์แวร์สามารถส่งไฟล์แบบกระจายไปยังสมาชิกทุกคนภายในแต่ละห้องสนทนาได้
- 3.9 ซอฟต์แวร์สามารถเพิ่มห้องสนทนา หรือ ปิดห้องสนทนาตามต้องการได้
- 3.10 ซอฟต์แวร์สามารถรับส่งข้อความสนทนาได้

3.4.1.2 ซอฟต์แวร์การสร้างห้องสนทนาของฝั่งลูกข่าย (Chat Room Client)

1. ความสามารถของซอฟต์แวร์ (Functional)

- 1.1 ซอฟต์แวร์สามารถค้นหา Chat Room Server เพื่อขอเข้าร่วมสนทนาได้
- 1.2 ซอฟต์แวร์สามารถสร้างห้องสนทนาได้
- 1.3 ซอฟต์แวร์สามารถรับส่งข้อความสนทนาได้

2. คุณสมบัติของซอฟต์แวร์ (Non Functional)

- 2.1 ซอฟต์แวร์สามารถแสดงรายชื่อของสมาชิกที่อยู่ภายในห้องสนทนาได้
- 2.2 ซอฟต์แวร์สามารถกำหนดชื่อผู้ใช้เพื่อใช้ในการสนทนาตามที่ผู้ใช้ต้องการได้
- 2.3 ซอฟต์แวร์สามารถตอบรับคำสั่งจาก Chat Room Server ได้

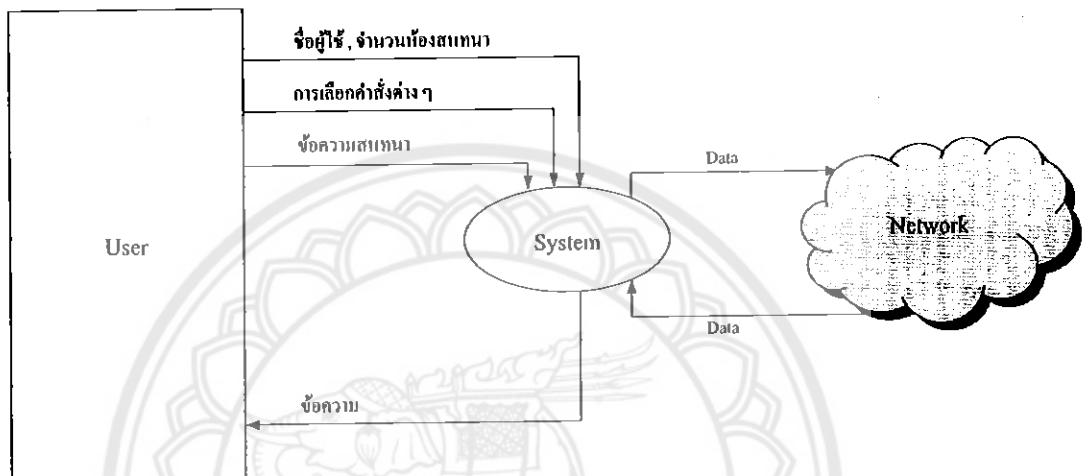
3. ข้อกำหนดของซอฟต์แวร์ (Specification)

- 3.1 ซอฟต์แวร์สามารถค้นหา Chat Room Server เพื่อติดต่อขอเข้าร่วมสนทนาได้โดยอัตโนมัติ
- 3.2 ซอฟต์แวร์สามารถสร้างห้องสนทนาเพื่อสนทนากับห้องที่ต้องการได้
- 3.3 ซอฟต์แวร์สามารถรับส่งข้อความสนทนาได้

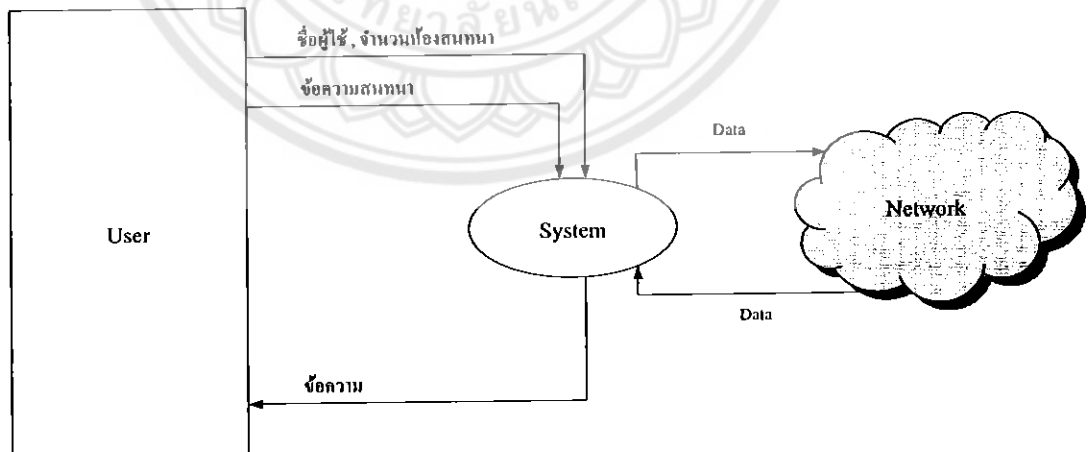
3.4 ซอฟต์แวร์สามารถแสดงรายชื่อสมาชิกที่อยู่ภายในห้องสนทนาได้

3.5 ซอฟต์แวร์สามารถตอบรับคำสั่งที่มาจาก Chat Room Server ได้

3.4.2 การออกแบบโปรแกรม



รูปที่ 3.6 Context Diagram ของ Chat Room Server



รูปที่ 3.7 Context Diagram ของ Chat Room Client

ตารางที่ 3.3 Business Event ของ Chat Room Server

เหตุการณ์	การไหล
1. ผู้ใช้กำหนดชื่อที่ใช้ในการสนทนา	(เข้า) ชื่อผู้ใช้
2. ผู้ใช้กำหนดจำนวนของห้องสนทนา	(เข้า) จำนวนห้องสนทนา
3. ผู้ใช้เลือกใช้คำสั่งต่างๆ	(เข้า) คำสั่งการใช้งาน
4. ผู้ใช้พิมพ์ข้อความสนทนา	(เข้า) ข้อความ
5. ซอฟต์แวร์ส่งข้อความ, คำสั่ง ไปยัง Network	(ออก) Data gram packet
6. ซอฟต์แวร์รับข้อความ, คำสั่ง จาก Network	(เข้า) Data gram packet
7. ซอฟต์แวร์แสดงข้อความ	(ออก) ข้อความ

ตารางที่ 3.4 Business Event ของ Chat Room Client

เหตุการณ์	การไหล
1. ผู้ใช้กำหนดชื่อผู้ใช้ในการสนทนา	(เข้า) ชื่อผู้ใช้
2. ผู้ใช้กำหนดหมายเลขห้องที่ต้องการ	(เข้า) หมายเลขห้องสนทนา
3. ผู้ใช้พิมพ์ข้อความสนทนา	(เข้า) ข้อความ
4. ซอฟต์แวร์ส่งข้อความ, คำสั่ง ไปยัง Network	(ออก) Data gram packet
5. ซอฟต์แวร์รับข้อความ, คำสั่ง จาก Network	(เข้า) Data gram packet
6. ซอฟต์แวร์แสดงข้อความ	(ออก) ข้อความ

3.4.2.1 Business Use Case

ชื่อ: การใช้งานโปรแกรม Chat Room Server

คำอธิบาย: ผู้ใช้ใช้โปรแกรม Chat Room Server เพื่อ สร้างห้องสนทนาสำหรับสนทนากับ Client

เงื่อนไขเริ่มต้น: ผู้ใช้เปิดโปรแกรม ช่วยควบคุมการเรียนการสอน และเรียกใช้คำสั่งเปิดโปรแกรม Chat Room ที่เครื่อง Server

การกระทำพื้นฐาน

1. ซอฟต์แวร์ถามชื่อผู้ใช้
2. ผู้ใช้กำหนดชื่อเพื่อใช้ในการสนทนาตามที่ต้องการ
3. ซอฟต์แวร์ถามจำนวนห้องสนทนาที่ต้องการ
4. ผู้ใช้กำหนดจำนวนห้องสนทนาตามต้องการ
5. ซอฟต์แวร์สร้างห้องสนทนาตามด้วยจำนวนกลุ่มที่ต้องการ
6. ซอฟต์แวร์สร้าง Server Chat Room เพื่อรอการติดต่อของ Client
7. ซอฟต์แวร์รอการใช้งานอื่นๆ
8. จบการทำงาน

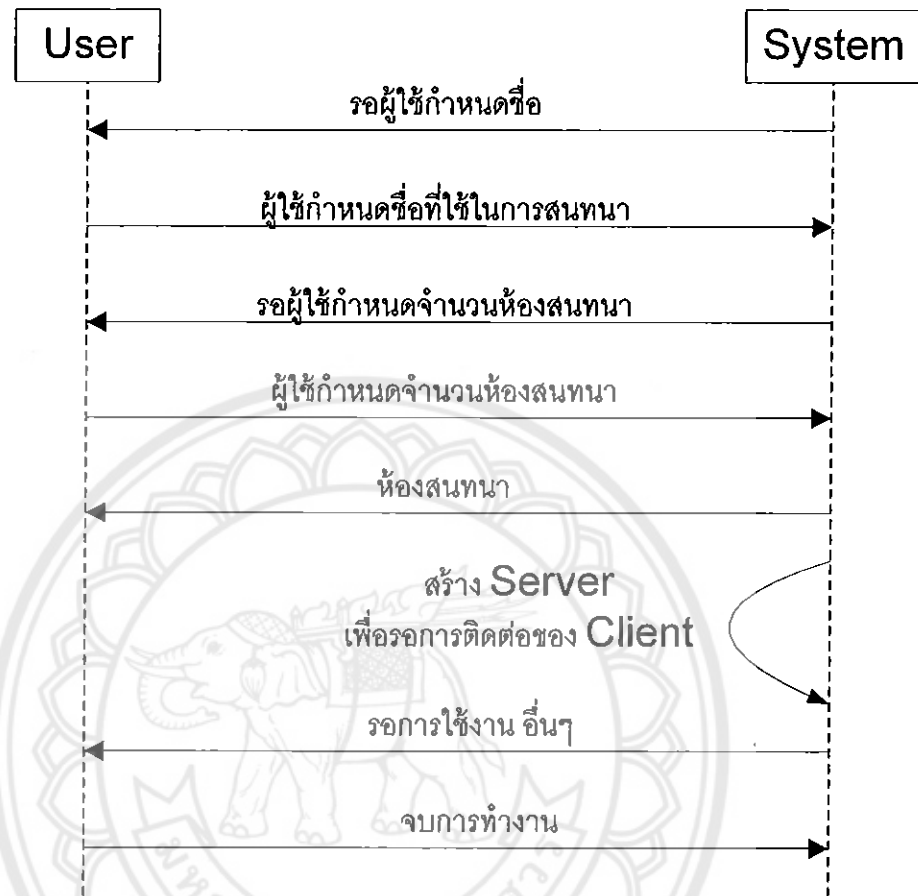
ชื่อ : การใช้งาน โปรแกรม Chat Room Client

คำอธิบาย: ผู้ใช้ใช้โปรแกรม Chat Room Client เพื่อสนทนากับผู้อื่น

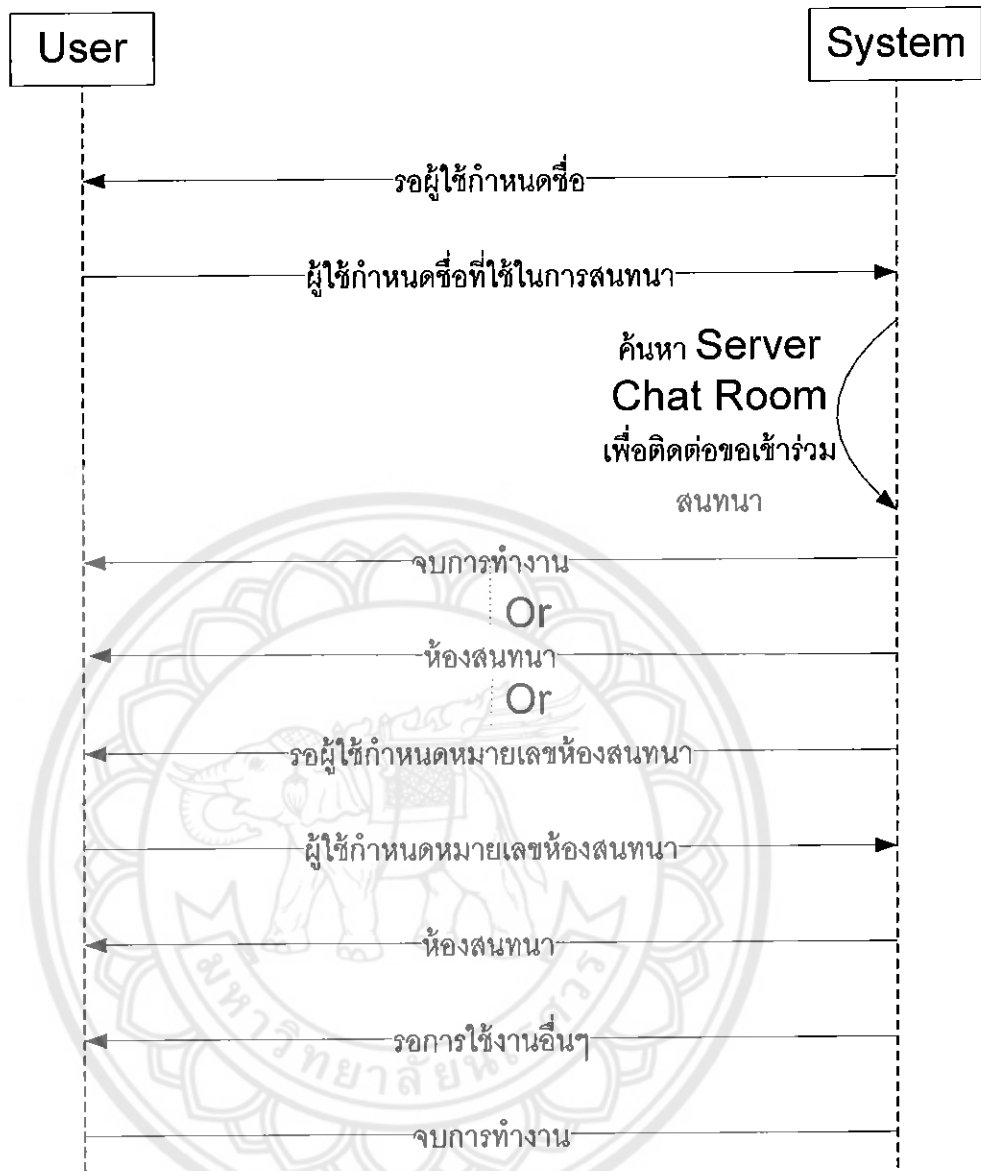
เงื่อนไขเริ่มต้น: ผู้ใช้เปิดโปรแกรม ช่วยควบคุมการเรียนการสอน และเรียกใช้คำสั่งเปิดโปรแกรม Chat Room ที่เครื่อง Client

การกระทำพื้นฐาน

1. ซอฟต์แวร์ถามชื่อผู้ใช้งาน
2. ผู้ใช้กำหนดชื่อเพื่อใช้ในการสนทนาตามที่ต้องการ
3. ซอฟต์แวร์ทำการค้นหา Server Chat Room เพื่อขอการเข้าร่วมสนทนา
4. ถ้า ไม่สามารถค้นหา Server Chat Room จบการทำงาน
5. ถ้า Server เป็นกำหนดหมายเลขห้องให้ ซอฟต์แวร์ทำการสร้างห้องสนทนาตามหมายเลขกลุ่มที่ถูกกำหนดให้
6. ถ้า Server ไม่ได้กำหนดหมายเลขกลุ่มให้ ซอฟต์แวร์ถามหมายเลขกลุ่มที่ผู้ใช้ต้องการ
7. ผู้ใช้ใส่หมายเลขกลุ่มที่ต้องการ
8. ซอฟต์แวร์สร้างห้องสนทนาตามหมายเลขห้องที่ต้องการ
9. ซอฟต์แวร์รอการใช้งานอื่น
10. จบการทำงาน

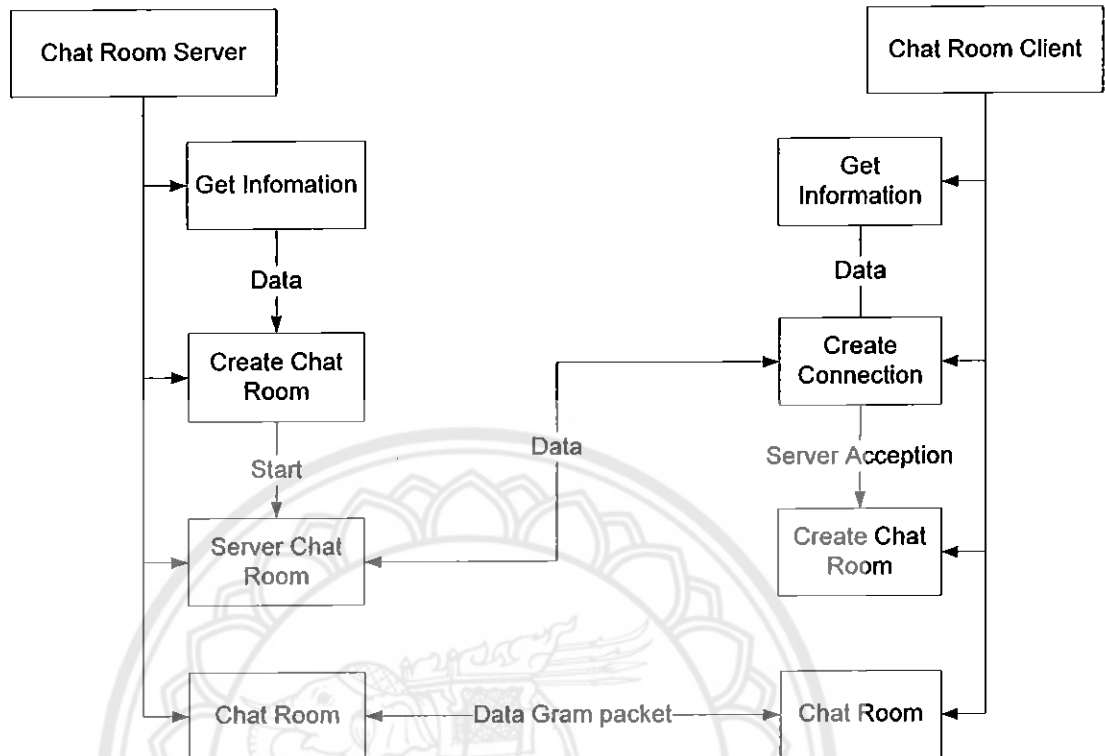


รูปที่ 3.8 Sequence Diagram ของการใช้ Chat Room Server เบื้องต้น



รูปที่ 3.9 Sequence Diagram ของการใช้ Chat Room Client เบื้องต้น

3.4.3 สถาปัตยกรรมของระบบ



รูปที่ 3.10 System Architecture ของ Server-Client Chat Room system

โปรแกรมการสร้างห้องสนทนานั้น มีการออกแบบตามการทำงานได้เป็น 5 ส่วนหลักๆ คือ

1. ส่วนรับข้อมูลพื้น (Get Information) เป็นส่วนที่ใช้รับข้อมูลเพื่อใช้ในการสร้างห้องสนทนา เช่น ชื่อผู้ใช้ จำนวนห้องสนทนาที่ต้องการสร้าง หมายเลขห้องสนทนาที่ต้องการเข้าร่วม
2. ส่วนสร้างห้องสนทนา (Create Chat Room) เป็นส่วนที่นำข้อมูลเริ่มต้นมาทำการสร้างหน้าต่างห้องสนทนา และสร้างช่องทางในการติดต่อสื่อสารระหว่างกลุ่มแต่ละกลุ่ม
3. ส่วนรอรับการติดต่อ (Server Chat Room) เป็นส่วนที่คอยรับการติดต่อเข้าร่วมสนทนาของ Client และส่งการตัดสินใจในการยอมรับการขอเข้าร่วมสนทนากลับไปยัง Client
4. ส่วนขอการติดต่อ (Create Connection) เป็นส่วนที่ทำการค้นหา Server Chat Room เพื่อติดต่อขอเข้าร่วมการสนทนา
5. ส่วนห้องสนทนา (Chat Room) เป็นส่วนหลักในการใช้ในการติดต่อกับผู้ใช้ (GUI) ทำหน้าที่เป็นตัวรับและแสดงข้อความของการสนทนา

3.5 โปรแกรมการรับส่งไฟล์ (File Server)

เป็นโปรแกรมที่ใช้สำหรับการส่งไฟล์ จากเครื่องลูกข่าย ไปยังเครื่องแม่ข่าย โดยที่ ลูกข่าย ไม่จำเป็นต้องรู้ IP Address ของเครื่องแม่ข่ายเพื่อทำการ Access เข้าไปยังเครื่องแม่ข่ายโดยตรง

ซอฟต์แวร์นั้นแบ่งออกเป็น 2 ฟังก์ชัน คือ File Server – Server และ File Server – Client โดยที่ โปรแกรม File Server – Server จะทำการสร้าง Server เพื่อรับการติดต่อเพื่อส่งไฟล์จาก โปรแกรม File Server – Client

3.5.1 การกำหนดความสามารถของซอฟต์แวร์

3.5.1.1 โปรแกรมการรับส่งไฟล์ฝั่งแม่ข่าย (File Server - Server)

1. ความสามารถของซอฟต์แวร์ (Functional)

1.1 ซอฟต์แวร์สามารถรับและเขียนไฟล์ได้

2. คุณสมบัติของซอฟต์แวร์ (Non-Functional)

2.1 ซอฟต์แวร์สามารถเขียนไฟล์ลง Directory ที่ต้องการได้

2.2 ซอฟต์แวร์สามารถรับการติดต่อจาก File Server – Client ได้ พร้อมๆกัน

3. ข้อกำหนดของซอฟต์แวร์ (Specification)

3.1 ซอฟต์แวร์สามารถเลือก Directory ที่ต้องการเพื่อใช้ในการเก็บ ไฟล์จาก File Server – Client ได้ตามต้องการ

3.2 ซอฟต์แวร์สามารถรับและเขียนไฟล์ได้อย่างถูกต้อง

3.3 ซอฟต์แวร์สามารถรับการติดต่อจาก File Server – Client ได้ พร้อมๆกัน

3.5.1.2 โปรแกรมการรับส่งไฟล์ฝั่งลูกข่าย (File Server – Client)

1. ความสามารถของซอฟต์แวร์ (Functional)

1.1 ซอฟต์แวร์สามารถส่งไฟล์ที่ต้องการได้

2. คุณสมบัติของซอฟต์แวร์ (Non – Functional)

2.1 ซอฟต์แวร์สามารถค้นหา Server ที่รอการติดต่อได้เองโดยอัตโนมัติ

2.2 ซอฟต์แวร์สามารถแสดงให้เห็นถึงสถานะหรือความก้าวหน้าในการส่งไฟล์ได้

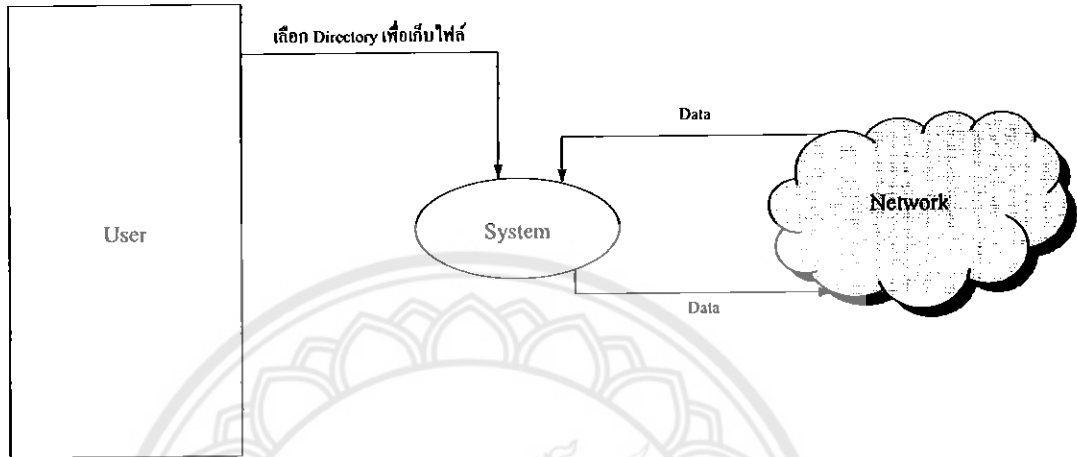
3. ข้อกำหนดของซอฟต์แวร์ (Specification)

3.1 ซอฟต์แวร์สามารถส่งไฟล์ที่ต้องการได้อย่างถูกต้อง

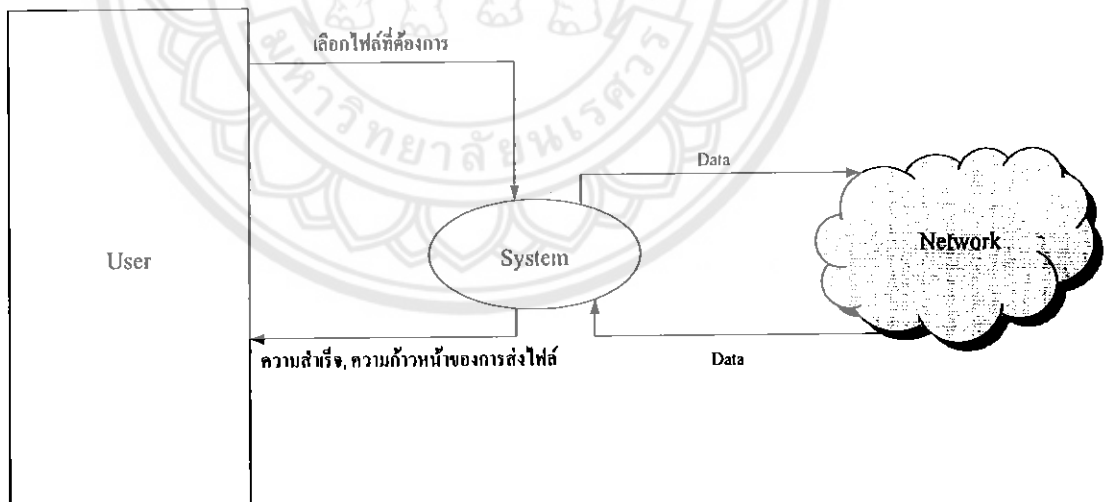
3.2 ซอฟต์แวร์สามารถค้นหาและติดต่อไปยัง Server ของ File Server ได้เอง

3.3 ซอฟต์แวร์สามารถแสดงข้อความเพื่อแสดงให้เห็นถึงสถานะของการติดต่อและส่งไฟล์ได้

3.5.2 การออกแบบโปรแกรม



รูปที่ 3.11 Context Diagram ของ File Server – Server



รูปที่ 3.12 Context Diagram ของ File Server – Client

ตารางที่ 3.5 Business Event ของ File Server – Server

เหตุการณ์	การไหล
1. ผู้ใช้เลือก Directory ที่ต้องการ	(เข้า) การเลือกคำสั่ง
2. ซอฟต์แวร์รับ ข้อมูลจาก Network	(เข้า) ข้อมูล
3. ซอฟต์แวร์ส่ง ข้อมูลไปยัง Network	(ออก) ข้อมูล

ตารางที่ 3.6 Business Event ของ File Server – Client

เหตุการณ์	การไหล
1. ผู้ใช้เลือก ไฟล์ที่ต้องการ	(เข้า) การเลือกคำสั่ง
2. ซอฟต์แวร์ส่งข้อมูล ไปยัง Network	(ออก) ข้อมูล
3. ซอฟต์แวร์รับข้อมูลจาก Network	(เข้า) ข้อมูล
4. ซอฟต์แวร์แสดงข้อความสถานะของการทำงาน	(ออก) ข้อความ

3.5.2.1 Business Use Case

ชื่อ : การใช้งานโปรแกรม File Server – Server

คำอธิบาย : ผู้ใช้ใช้งาน File Server – Server เพื่อรับการส่งไฟล์จาก Client

เงื่อนไขเริ่มต้น : ผู้ใช้เปิดโปรแกรม ช่วยควบคุมการเรียนการสอน และเรียกใช้คำสั่งเปิดโปรแกรม

การรับส่งไฟล์ที่เครื่อง Server

การกระทำพื้นฐาน

1. ซอฟต์แวร์รอการเลือก Directory ที่ต้องการเก็บไฟล์จาก Client
2. ผู้ใช้เลือก Directory ที่ต้องการ
3. ซอฟต์แวร์สร้าง Server เพื่อรอการติดต่อจาก Client
4. จบการทำงาน

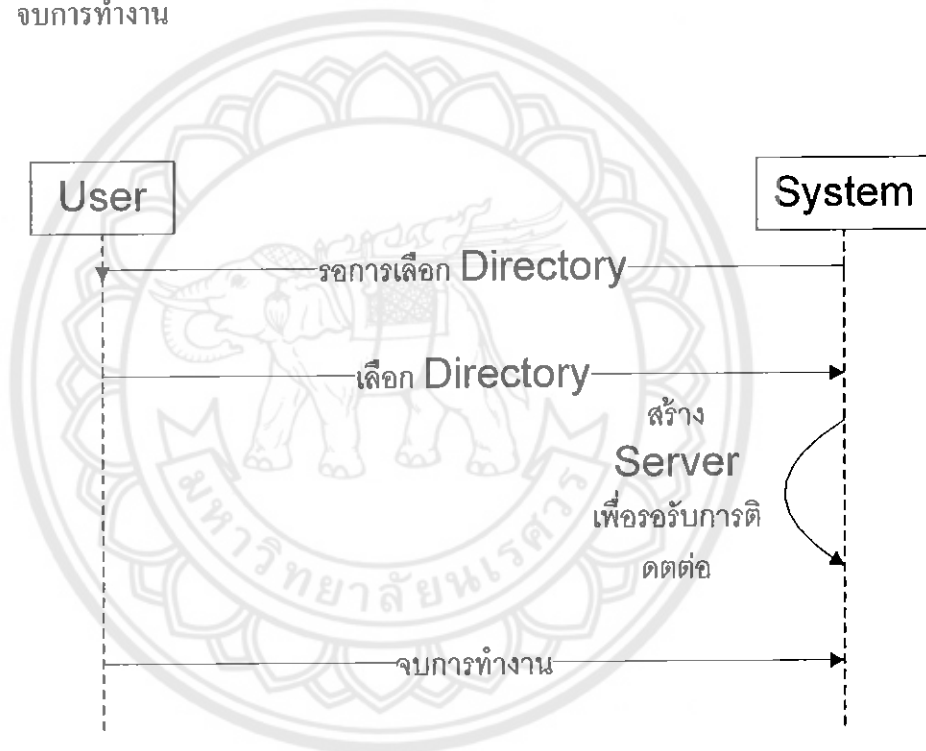
ชื่อ : การใช้งานโปรแกรม File Server – Client

คำอธิบาย : ผู้ใช้ใช้งาน File Server – Client เพื่อส่งไฟล์ที่ต้องการไปยัง Server

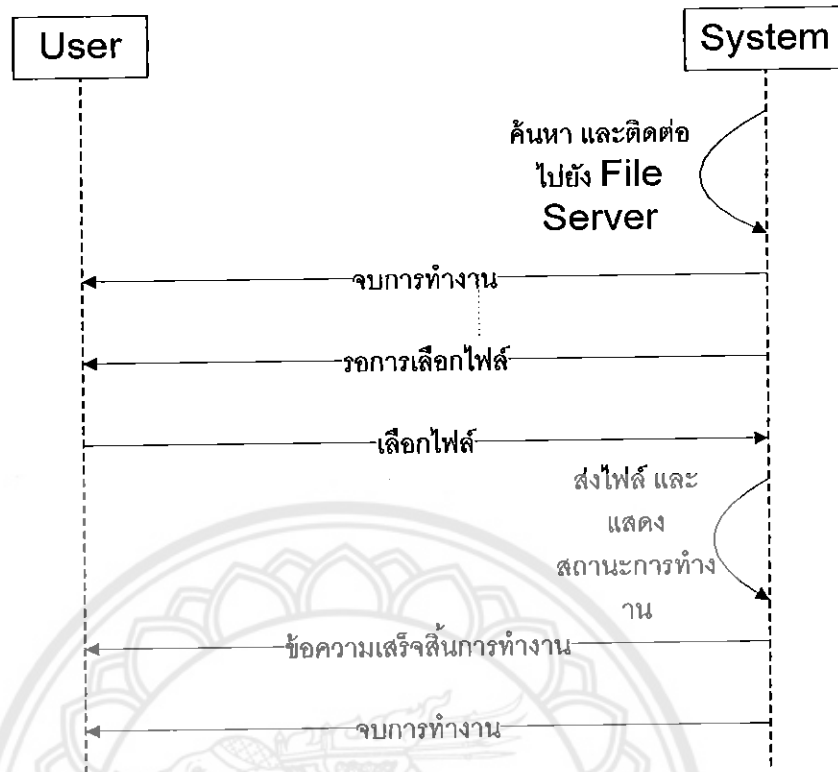
เงื่อนไขเริ่มต้น : ผู้ใช้เปิดโปรแกรม ช่วยควบคุมการเรียนการสอน และเรียกใช้คำสั่งเปิดโปรแกรม การรับส่งไฟล์ที่เครื่อง Client

การกระทำพื้นฐาน

1. ซอฟต์แวร์รอผู้ใช้เลือกไฟล์ที่ต้องการส่งไปยัง Server
2. ผู้ใช้เลือกไฟล์ที่ต้องการ
3. ซอฟต์แวร์แสดงสถานะของการทำงาน
4. ซอฟต์แวร์แสดงข้อความเสร็จสิ้นการทำงาน
5. จบการทำงาน

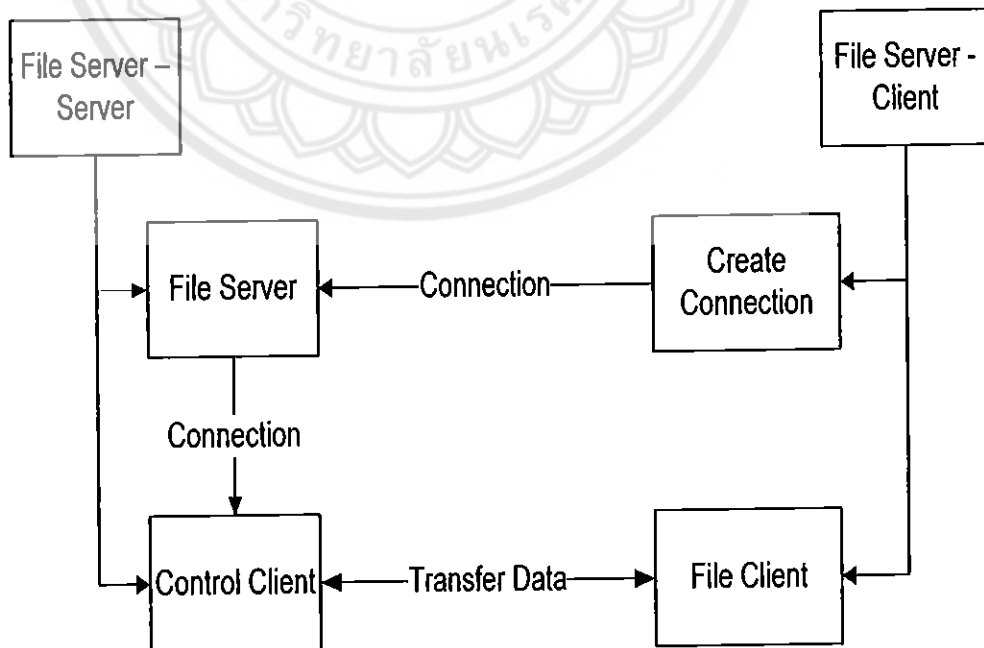


รูปที่ 3.13 Sequence Diagram ของการใช้งาน File Server - Server



รูปที่ 3.14 Sequence Diagram ของการใช้งาน File Server - Client

3.5.3 สถาปัตยกรรมของระบบ



รูปที่ 3.15 System Architecture ของ Server - Client File Server System

โปรแกรมการรับส่งไฟล์ จะถูกออกแบบตามหน้าที่การทำงาน ซึ่งแบ่งออกเป็น 4 อย่างที่สำคัญๆ คือ

1. File Server เป็นตัวที่รอรับการติดต่อ (Connection) จาก Client (Create Connection) และนำ Connection นั้น ไปให้ส่วน Control Client ทำงาน และตัวเอกก็จะกลับมารอการ Connection ของ Client
2. Create Connection เป็นส่วนที่จะตรวจสอบหา File Server และทำการ Connection ไปยัง Server เพื่อที่จะทำการส่งไฟล์ที่ต้องการ
3. Control Client เป็นส่วนที่รับ Connection แต่ละตัวมาจาก File Server และเตรียมการรับไฟล์จาก Client โดย Control Client นี้จะทำงานแบบ Thread เพื่อที่จะได้สามารถรองรับการติดต่อของ Client ได้พร้อมๆ กัน
4. File Client เป็นส่วนที่ทำหน้าที่ในการอ่านและ ส่งไฟล์ที่ต้องการ ไปยัง Server และยังทำหน้าที่แสดงสถานะของการทำงาน

3.6 โปรแกรมการรับส่งไฟล์แบบกระจาย (Broadcast)

เป็นซอฟต์แวร์ที่ใช้สำหรับการรับส่งไฟล์โดยการกระจาย (Broadcast) เพื่อกระจายไฟล์จากเครื่อง Server ไปยังเครื่อง Client ทุกเครื่องที่อยู่ในการเชื่อมต่อในขณะนั้น โดยฝั่ง Server จะเลือกไฟล์และกระจายไฟล์ไปยังเครื่องลูกข่าย เมื่อเครื่องลูกข่ายรับไฟล์แล้วจะเก็บไฟล์ไว้ในโฟลเดอร์ Broadcast file from teacher ซึ่งลูกข่ายสร้างไว้เพื่อรอรับไฟล์จากเครื่องแม่ข่าย

3.6.1 การกำหนดความสามารถของซอฟต์แวร์

3.6.1.1 ซอฟต์แวร์การรับส่งไฟล์แบบกระจายฝั่งแม่ข่าย (Broadcast Server)

1. ความสามารถของซอฟต์แวร์ (Functional)

1.1 ซอฟต์แวร์สามารถกระจายไฟล์ที่ผู้ใช้เลือกไปยังเครื่องลูกข่ายได้

2. คุณสมบัติของซอฟต์แวร์ (Non – Functional)

2.1 ซอฟต์แวร์สามารถเลือกไฟล์ที่ต้องการกระจายไฟล์ได้

2.2 ซอฟต์แวร์สามารถแสดงจำนวนไฟล์ที่เลือกเพื่อกระจายไฟล์ได้

2.3 ซอฟต์แวร์สามารถเพิ่มไฟล์ที่ต้องการกระจายได้

2.4 ซอฟต์แวร์สามารถลบไฟล์ที่ต้องการกระจายได้

2.5 ซอฟต์แวร์สามารถแสดงความถี่หน้าของการกระจายไฟล์ได้

3. ข้อกำหนดของซอฟต์แวร์ (Specification)

3.1 ซอฟต์แวร์สามารถเพิ่มรายชื่อไฟล์ที่ผู้ใช้เลือกลงในรายการไฟล์ที่ต้องการกระจายไฟล์ได้

3.2 ซอฟต์แวร์สามารถลบรายชื่อไฟล์ที่ผู้ใช้เลือกในรายการไฟล์ที่ต้องการกระจายไฟล์ได้

3.3 ซอฟต์แวร์สามารถกระจายไฟล์ตามที่ใช้ต้องการได้

3.4 ซอฟต์แวร์สามารถบอกถึงสถานะและความคืบหน้าของการทำงานได้

3.6.1.2 ซอฟต์แวร์การรับส่งไฟล์แบบกระจายฝั่งลูกข่าย (Broadcast Client)

1. ความสามารถของซอฟต์แวร์ (Functional)

1.1 ซอฟต์แวร์สามารถรับและเขียนไฟล์จากการ Broadcast จาก Broadcast Server ได้

2. คุณสมบัติของซอฟต์แวร์ (Non Functional)

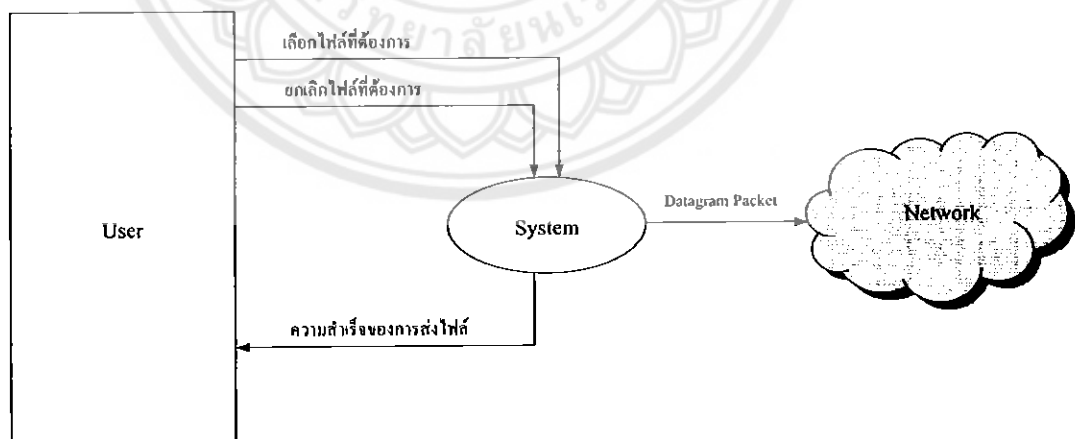
2.1 ซอฟต์แวร์รับและเขียนไฟล์ทุกไฟล์ลง Directory เดียวกันได้

3. ข้อกำหนดของซอฟต์แวร์ (Specification)

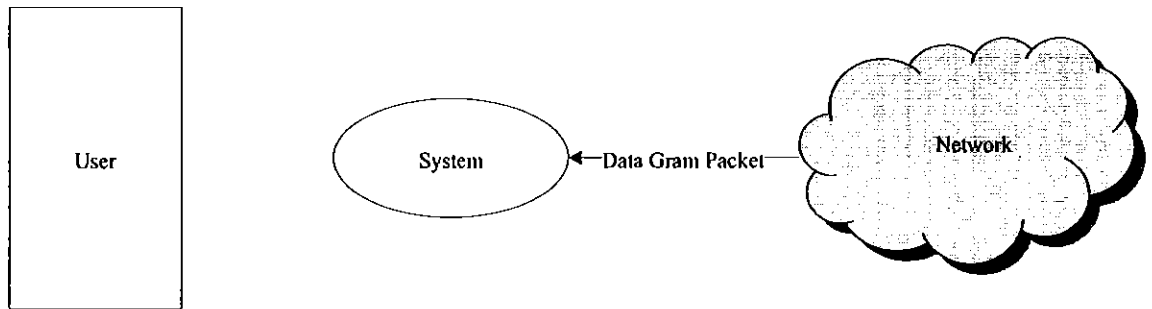
3.1 ซอฟต์แวร์สามารถรับและเขียนไฟล์ที่ Server ทำการ Broadcast มาได้

3.2 ซอฟต์แวร์สามารถสร้าง Directory เพื่อใช้ในการเก็บไฟล์ที่รับมาจากการ Broadcast ไว้ที่เดียวกัน

3.6.2 การออกแบบโปรแกรม



รูปที่ 3.16 Context Diagram ของ Broadcast – Server



รูปที่ 3.17 Context Diagram ของ Broadcast – Client

ตารางที่ 3.7 Business Event ของ Broadcast - Server

เหตุการณ์	การไหล
1. ผู้ใช้เลือกไฟล์ที่ต้องการส่ง	(เข้า) คำสั่งการใช้งาน
2. ผู้ใช้ยกเลิกไฟล์ที่เลือก	(เข้า) คำสั่งการใช้งาน
3. ซอฟต์แวร์ Broadcast Data ไปยัง Network	(ออก) Datagram packet
4. ซอฟต์แวร์แสดงสถานะความคืบหน้าของการกระจายไฟล์	(ออก) สถานะ การทำงาน

ตารางที่ 3.8 Business Event ของ Broadcast – Client

เหตุการณ์	การไหล
1. ซอฟต์แวร์รับ Datagram Packet จาก Network	(เข้า) Datagram Packet

3.4.2.1 Business Use Case

ชื่อ: การใช้งานโปรแกรม Broadcast – Server

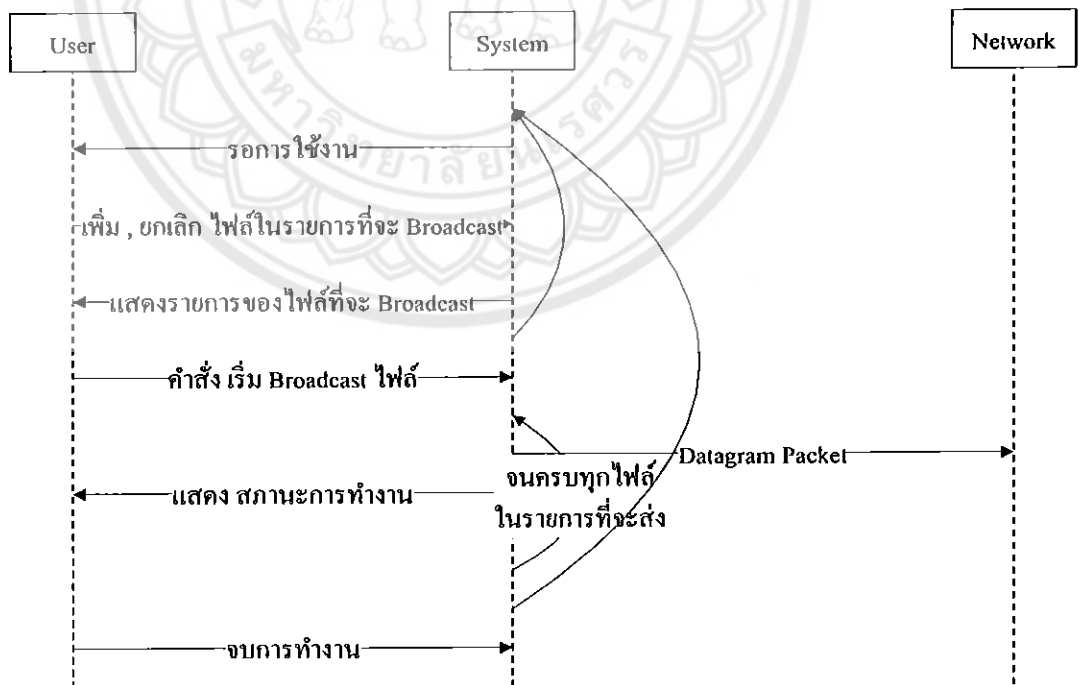
คำอธิบาย : ผู้ใช้ใช้โปรแกรม Broadcast เพื่อกระจายไฟล์ไปยังเครื่องลูกข่ายทุกเครื่อง

เงื่อนไขเริ่มต้น : ผู้ใช้เปิดโปรแกรม ช่วยควบคุมการเรียนการสอน และเรียกใช้คำสั่งเปิด โปรแกรม

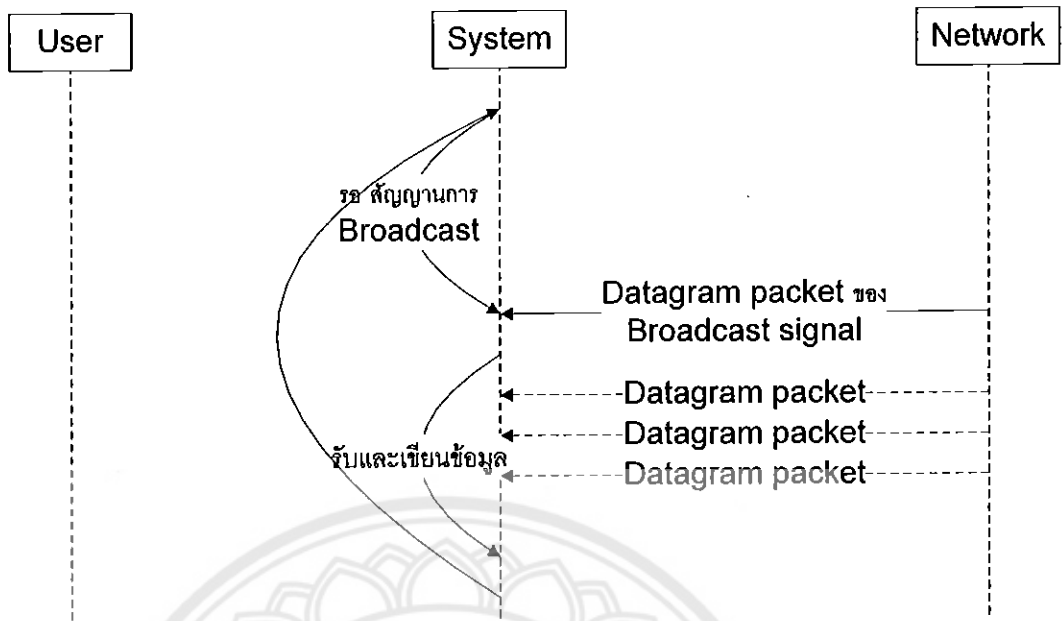
Broadcast ที่เครื่อง Server

การกระทำพื้นฐาน

1. ซอฟต์แวร์รอการใช้งาน
2. ผู้ใช้เลือกไฟล์ที่ต้องการกระจายไฟล์
3. ซอฟต์แวร์เพิ่มรายชื่อไฟล์ที่ผู้ใช้เลือกไว้ในรายการไฟล์ที่จะทำการกระจายไฟล์
4. ผู้ใช้กำหนดให้ซอฟต์แวร์เริ่มกระจายไฟล์
5. ซอฟต์แวร์กระจายไฟล์ที่เลือกไปยังเครื่องลูกข่าย
6. ซอฟต์แวร์แสดงความคืบหน้าของการกระจายไฟล์
7. กลับไป 1.
8. จบการทำงาน

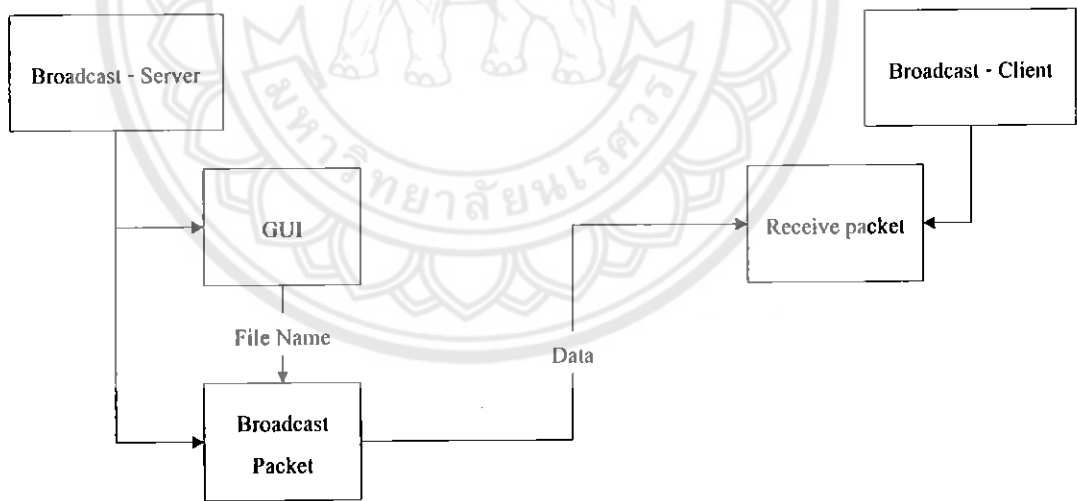


รูปที่ 3.18 Sequence Diagram การใช้งานเบื้องต้นของ Broadcast – Server



รูปที่ 3.19 Sequence Diagram การทำงานพื้นฐานของ Broadcast – Client

3.6.3 สถาปัตยกรรมของระบบ



รูปที่ 3.20 System Architecture ของ Server – Client Broadcast System

ซอฟต์แวร์ Broadcast สามารถแยกตามหน้าที่การทำงานออกได้เป็น 3 ส่วนหลักๆ คือ

1. GUI เป็นส่วนที่ใช้ในการติดต่อกับผู้ใช้งานเพื่อให้ผู้ใช้งานสามารถเรียกใช้คำสั่งใช้งานได้ง่าย และเป็นส่วนที่แสดงรายชื่อของไฟล์ที่จะทำการส่ง
2. Broadcast Packet เป็นส่วนที่รับชื่อไฟล์มาจากส่วนของ GUI ทีละชื่อ แล้วทำการกระจายสัญญาณการส่งไฟล์แบบกระจายไปยัง Network และทำการเปิดอ่านข้อมูลของไฟล์เพื่อทำการกระจายข้อมูลนั้นๆ ไปยัง Network
3. Receive packet เป็นส่วนที่รองรับ Datagram packet จาก Network และนำข้อมูลต่างๆ ที่ได้เขียนลงไฟล์เพื่อให้ได้ไฟล์ที่ถูกส่งมา

3.7 โปรแกรมดูหน้าจอเครื่องแม่ข่าย (RCL VNC)

RCL VNC (Remote Control System for E-Learning Virtual Network Computing) เป็นโปรแกรมที่ใช้สำหรับการให้เครื่องที่เปิดโปรแกรมช่วยควบคุมการเรียนการสอนฝั่ง Client สามารถที่จะมองเห็น หน้าจอของเครื่อง Server ซึ่งสามารถแบ่งออกได้เป็น 2 ฝั่งการทำงานคือ RCL VNC – Server ทำหน้าที่ในการควบคุม RCL VNC – Client ในการเปิดปิดโปรแกรม

โปรแกรมนี้ถูกพัฒนาขึ้น โดยการนำ Source Code ของ Java Viewer ซึ่งอยู่ใน Open Source Ultra VNC Version 1.0.2 มาพัฒนาต่อเพื่อใช้ใน โปรแกรม RCL VNC – Client

3.7.1 การกำหนดความสามารถของซอฟต์แวร์

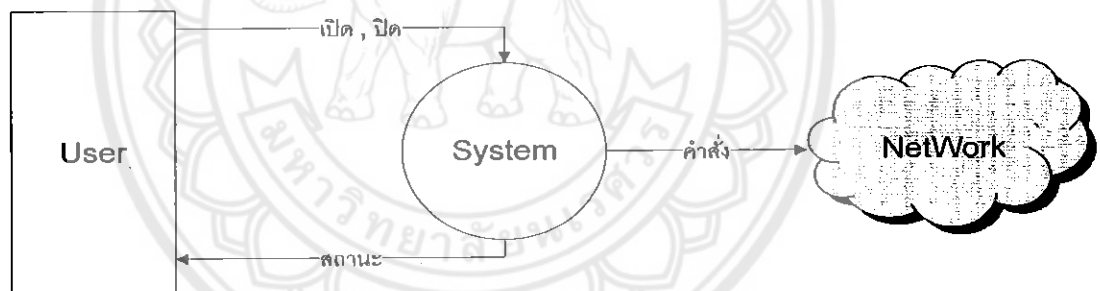
3.7.1.1 โปรแกรมดูหน้าจอเครื่องแม่ข่ายฝั่งแม่ข่าย (RCL VNC - Server)

1. ความสามารถของซอฟต์แวร์ (Functional)
 1. ซอฟต์แวร์สามารถให้เครื่อง Client เห็นหน้าจอของเครื่อง Server ได้
2. คุณสมบัติของซอฟต์แวร์ (Non – Functional)
 1. ซอฟต์แวร์สามารถควบคุมให้เครื่อง Client เปิด – ปิด โปรแกรม RCL VNC ได้ตามความต้องการ
 2. ซอฟต์แวร์สามารถแสดงให้ผู้ใช้งานรู้ถึงสถานะการเรียกให้โปรแกรม RCL VNC
3. ข้อกำหนดของซอฟต์แวร์ (Specification)
 1. ซอฟต์แวร์สามารถให้เครื่อง Client เห็นหน้าจอของเครื่อง Server ตามที่ต้องการ
 2. ซอฟต์แวร์สามารถควบคุมให้ Client ที่เปิดโปรแกรม ช่วยควบคุมการเรียนการสอน เปิด – ปิด โปรแกรม RCL VNC ได้ตามที่ต้องการ
 3. ซอฟต์แวร์สามารถแสดงให้ผู้ใช้งานรู้ได้ถึงสถานะการให้คำสั่งให้ Client เปิด หรือ ปิดโปรแกรม RCL VNC

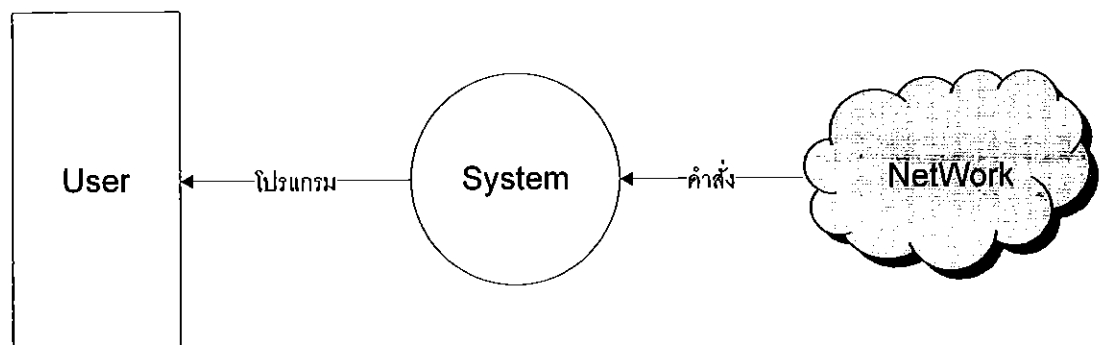
3.7.1.2 โปรแกรมดูหน้าจอเครื่องแม่ฝั่งลูกข่าย (RCL VNC – Client)

1. ความสามารถของซอฟต์แวร์ (Functional)
 1. ซอฟต์แวร์สามารถดูหน้าจอของเครื่อง Server ได้
2. คุณสมบัติของซอฟต์แวร์ (Non Functional)
 1. ซอฟต์แวร์ไม่สามารถเปิด โปรแกรม RCL VNC – Client ได้เองตามความต้องการ
3. ข้อกำหนดของซอฟต์แวร์ (Specification)
 1. ซอฟต์แวร์สามารถดูหน้าจอของเครื่อง Server ได้ โดยที่การเปิด – ปิด โปรแกรมนั้น ขึ้นอยู่กับความต้องการของผู้ใช้งาน RCL VNC – Server
 2. ซอฟต์แวร์ไม่สามารถเปิดได้โดยผู้ใช้งาน RCL VNC – Client

3.7.2 การออกแบบโปรแกรม



รูปที่ 3.21 Context Diagram ของ RCL VNC – Server



รูปที่ 3.22 Context Diagram ของ RCL VNC – Client

ตารางที่ 3.9 Business Event ของ RCL VNC – Server

เหตุการณ์	การไหล
1. ผู้ใช้เลือกใช้คำสั่ง เปิด , ปิด โปรแกรม	(เข้า) คำสั่งการใช้งาน
2. ซอฟต์แวร์ส่งคำสั่งใช้งานไปยัง Network	(ออก) คำสั่งการใช้งาน
3. ซอฟต์แวร์แสดงสถานะการเรียกใช้โปรแกรม	(ออก) สถานะการใช้งาน

ตารางที่ 3.10 Business Event ของ RCL VNC – Client

เหตุการณ์	การไหล
1. ซอฟต์แวร์รับคำสั่งการใช้งานจาก Network	(เข้า) คำสั่งการใช้งาน
2. ซอฟต์แวร์เปิด – ปิด การใช้งาน	(ออก)สถานะของ โปรแกรม

3.7.2.1 Business Use Case

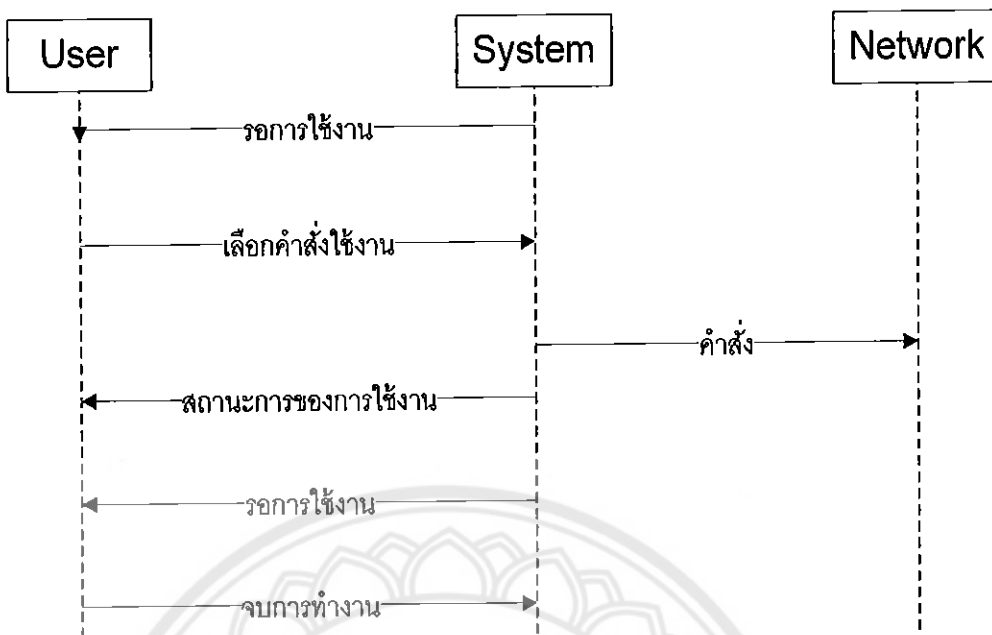
ชื่อ : การใช้งาน RCL VNC – Server

คำอธิบาย : ผู้ใช้ซอฟต์แวร์ช่วยควบคุมการเรียนการสอนฝั่ง Server ต้องการ ใช้ Client เห็นหน้าจอของเครื่อง Server ที่หน้าจอของตนเอง

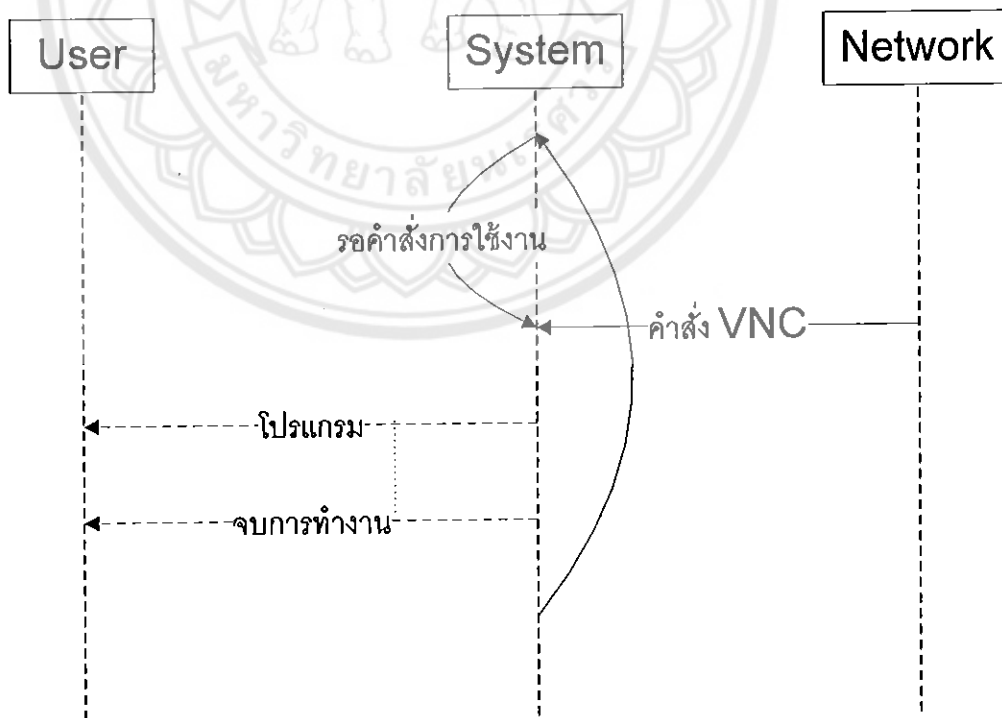
เงื่อนไขเริ่มต้น : ผู้ใช้เปิดโปรแกรมช่วยควบคุมการเรียนการสอน และเลือกส่วนควบคุมเกี่ยวกับ VNC

การกระทำเบื้องต้น

1. ซอฟต์แวร์รอการใช้งาน
2. ผู้ใช้เลือกใช้คำสั่งการใช้งาน
3. ซอฟต์แวร์ส่งคำสั่งใช้งานไปยัง Network
4. ซอฟต์แวร์แสดงสถานะของการใช้งาน
5. ซอฟต์แวร์รอการใช้งานคำสั่งอื่น
6. จบการทำงาน

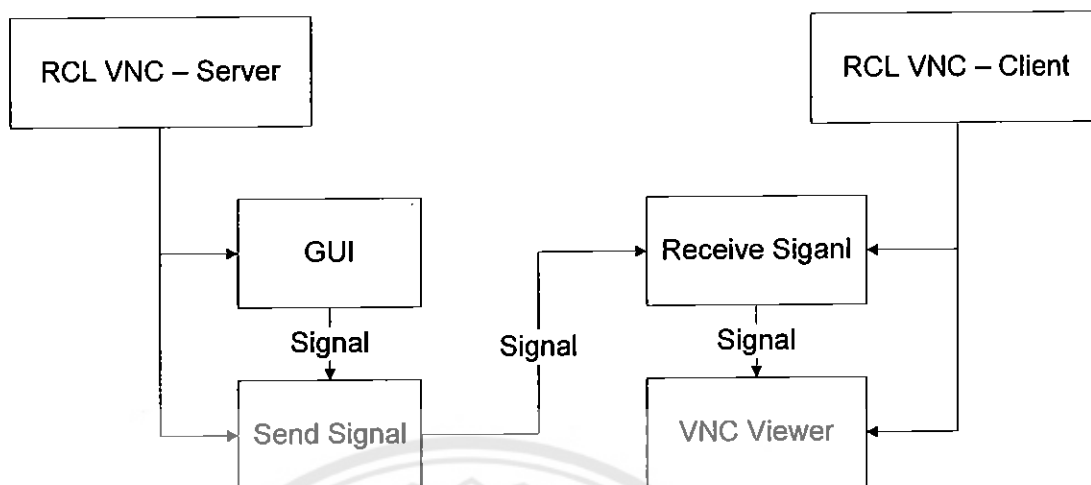


รูปที่ 3.23 Sequence Diagram การใช้งานเบื้องต้นของ RCL VNC – Server



รูปที่ 3.24 Sequence Diagram การทำงานพื้นฐานของ RCL VNC – Client

3.7.3 สถาปัตยกรรมของระบบ



รูปที่ 3.25 System Architecture ของ Server – Client RCL VNC system

ซอฟต์แวร์ RCL VNC สามารถแบ่งตามหน้าที่การทำงานได้เป็น 4 ส่วนหลักๆ คือ

1. GUI (Graphic User Interface) เป็นส่วนที่ใช้สำหรับให้ผู้ใช้สามารถเลือกใช้คำสั่งต่างๆ ได้โดยง่ายและ ส่งคำสั่งที่เลือกไปยังส่วนการส่งข้อมูลคำสั่งต่อไป
2. Send Signal เป็นส่วนที่ส่งรหัสคำสั่งที่ถูกผู้ใช้เลือกไปยังช่องทางการเชื่อมต่อระหว่าง Server – Client
3. Receive Signal เป็นส่วนที่รองรับคำสั่งของช่องทางการเชื่อมต่อระหว่าง Server – Client และเมื่อรับคำสั่งมาก็จะส่งคำสั่งที่ได้ให้กับส่วนของ VNC Viewer
4. VNC Viewer เป็นส่วนที่ใช้เรียกโปรแกรม VNC Viewer เพื่อดูหน้าจอของ Server หรือปิดการใช้งาน VNC Viewer ในการดูหน้าจอ

บทที่ 4

ผลการทดลอง

ในบทนี้จะกล่าวถึงการทดสอบและการแสดงผลของซอฟต์แวร์ระบบควบคุมคอมพิวเตอร์ช่วยในการเรียนการสอน โดยทดสอบตามข้อกำหนดของซอฟต์แวร์ (Specification) การทดสอบแบ่งออกเป็น 4 ส่วน คือ การทดสอบการทำงานของ โปรแกรมห้องสนทนา (Chat Room) โปรแกรมการรับส่งไฟล์ (File Server) โปรแกรมการกระจายไฟล์ (Broadcast) และ ทดสอบการทำงานของโปรแกรม VNC

4.1 การทดสอบการทำงานของ Chat Room

การทดสอบการทำงานของ Chat Room ของซอฟต์แวร์ระบบควบคุมคอมพิวเตอร์ช่วยในการเรียนการสอนนั้นแบ่งการทดสอบเป็น 2 ส่วนคือ การทำงานของ Server และ Client ดังนี้

4.1.1 การทดสอบ Server

การทดสอบการทำงานของ Server เป็นการทดสอบการสร้างห้องสนทนา การจัดการการสนทนา แบบ Auto Acceptance คือ Client สามารถเข้าห้องสนทนาได้เองโดยไม่ต้องรอการอนุญาตจาก Server และ Manual Acceptance คือ Server จะสามารถ ยอมรับและเลือกห้องสนทนาให้ Client หรือไม่ยอมรับให้ Client เข้าห้องสนทนาได้ การทดสอบการเพิ่ม ลบ และปิดห้องสนทนา การทดสอบการเพิ่ม ลบ Client ที่อยู่ในห้องสนทนา ทดสอบการส่งไฟล์ และทดสอบการกระจายไฟล์

ข้อกำหนดของซอฟต์แวร์ (Specification)

1. ซอฟต์แวร์สามารถกำหนดชื่อผู้ใช้ตามที่ผู้ใช้ต้องการเพื่อใช้ในการ สนทนาได้



PLEASE INSERT NAME		OK	CANCEL
Teacher			

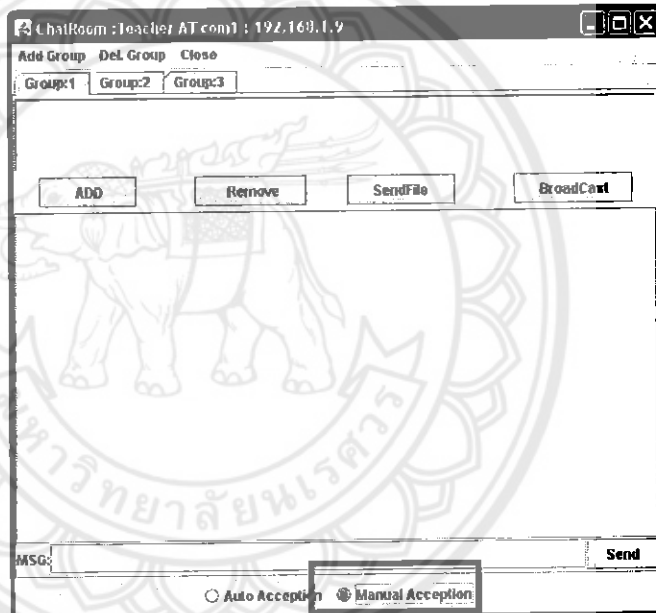
รูปที่ 4.1 กำหนดของ Teacher ในการสร้างห้องสนทนา

2. ซอฟต์แวร์สามารถสร้างห้องสนทนาตามจำนวนที่ผู้ใช้ต้องการได้



รูปที่ 4.2 การใส่จำนวนห้อง (Total Group) ในการสร้างห้องสนทนา

3. ซอฟต์แวร์สามารถตอบรับหรือปฏิเสธการขอเข้าร่วมสนทนาของ Client ตามความต้องการของผู้ใช้ได้

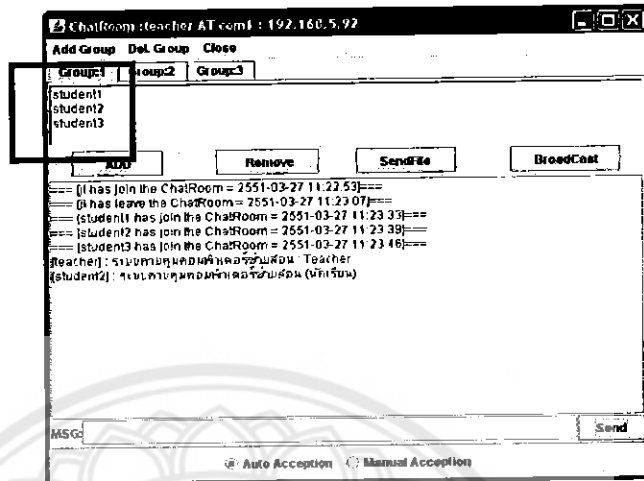


รูปที่ 4.3 เลือกการตอบรับ Client เป็นแบบ Manual Acceptation



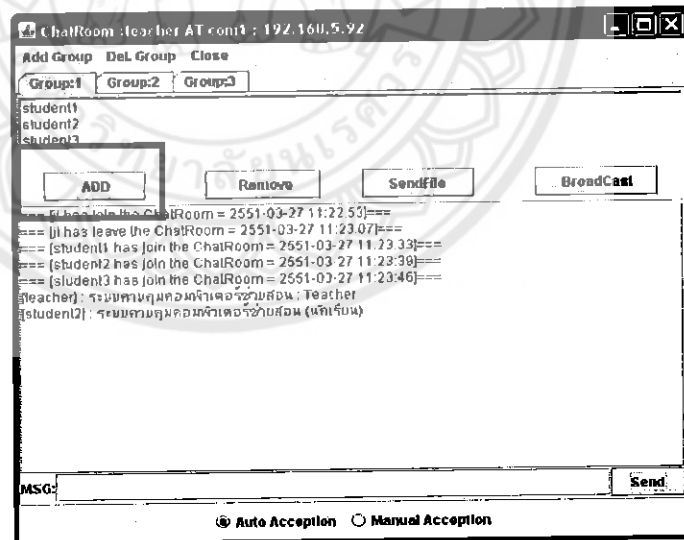
รูปที่ 4.4 เลือกตอบรับหรือปฏิเสธการขอเข้าร่วมสนทนาของ Client

4. ซอฟต์แวร์สามารถแสดงรายชื่อของสมาชิกภายในแต่ละห้องสนทนาได้

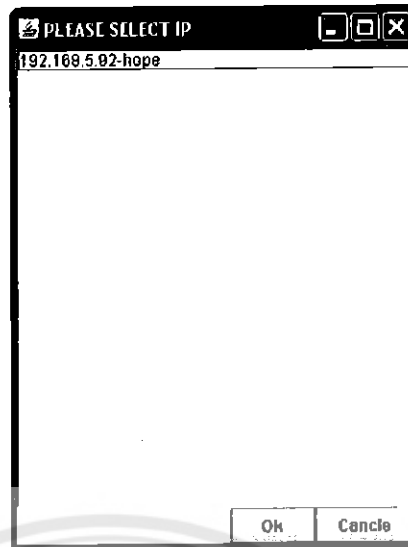


รูปที่ 4.5 แสดงรายชื่อของสมาชิกภายในห้องสนทนา

5. ซอฟต์แวร์สามารถเรียก Client ที่ต้องการเข้าสู่ห้องสนทนาได้

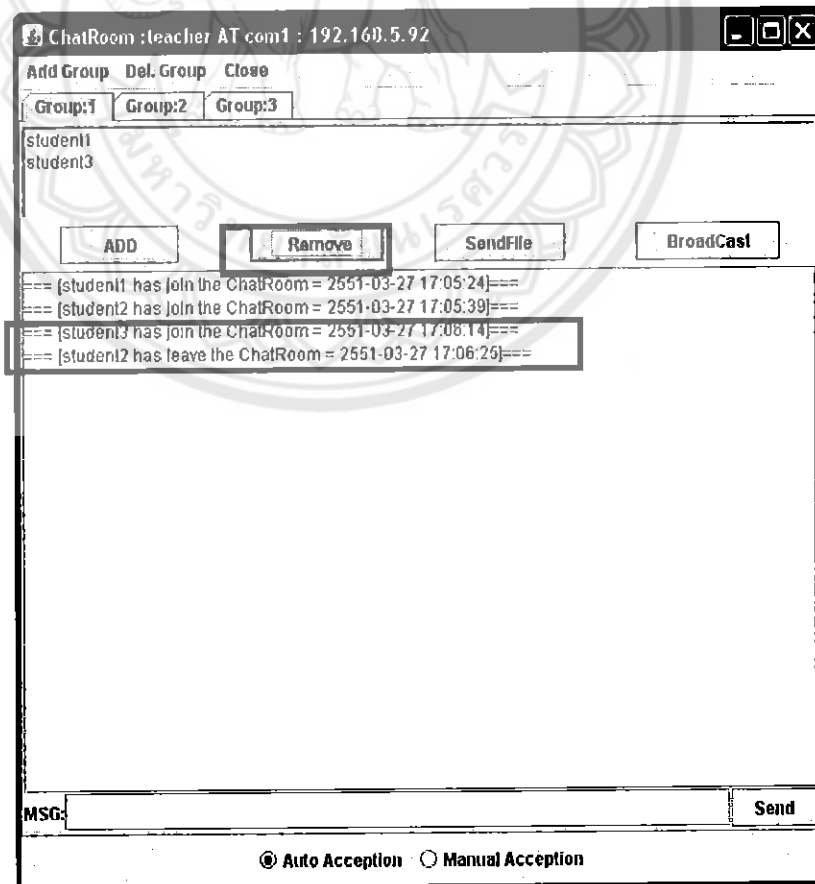


รูปที่ 4.6 เรียก Client ที่ต้องการเข้าสู่ห้องสนทนา



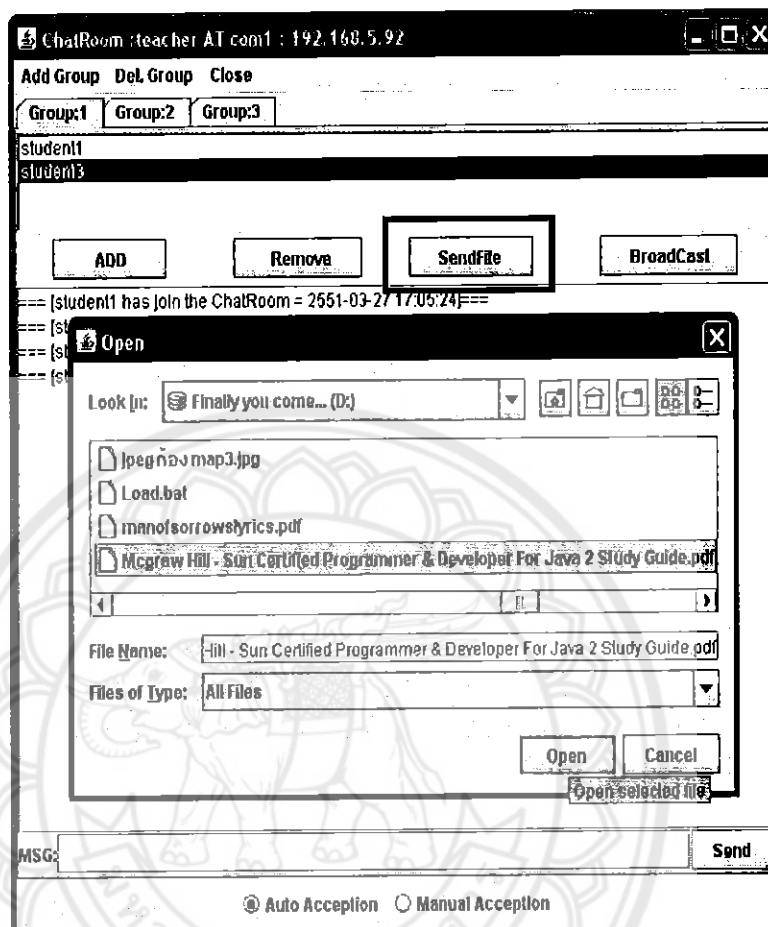
รูปที่ 4.7 เลือก Client ที่ต้องการเรียกเข้าในห้องสนทนา

6. ซอฟต์แวร์สามารถกำหนดให้สมาชิกภายในห้องสนทนาที่ต้องการ ออกจากห้องสนทนาได้ตามต้องการ

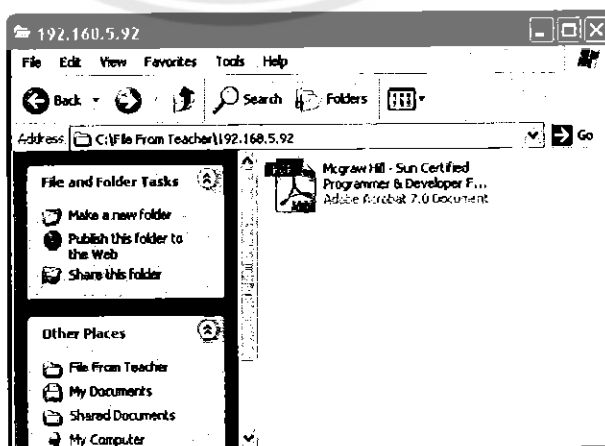


รูปที่ 4.8 การลบ Client ออกจากห้องสนทนา

7. ซอฟต์แวร์สามารถส่งไฟล์ที่ต้องการไปยังสมาชิกภายในห้องสนทนาได้ตามต้องการ

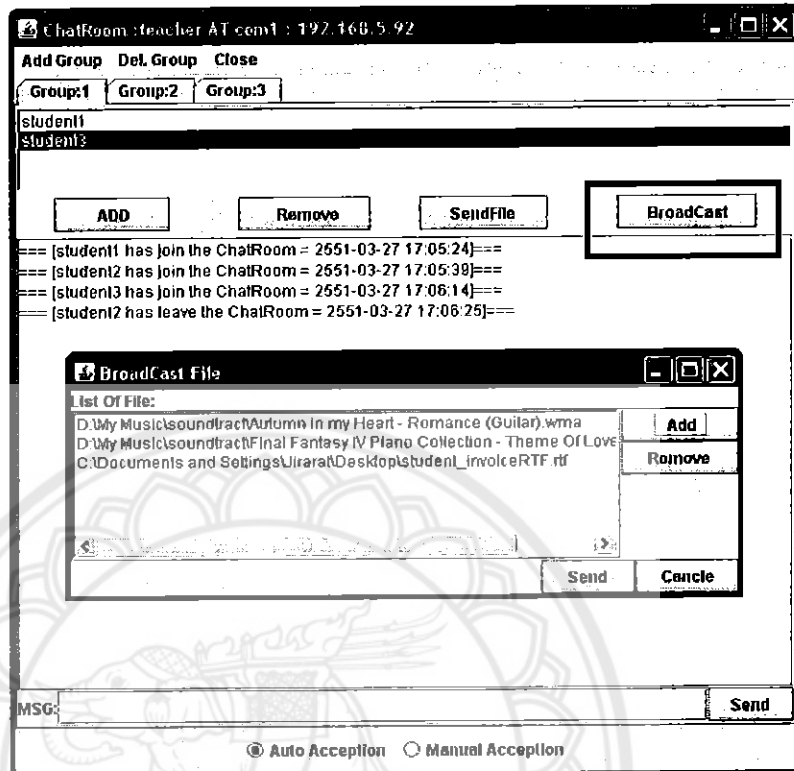


รูปที่ 4.9 การส่งไฟล์ในห้องสนทนา

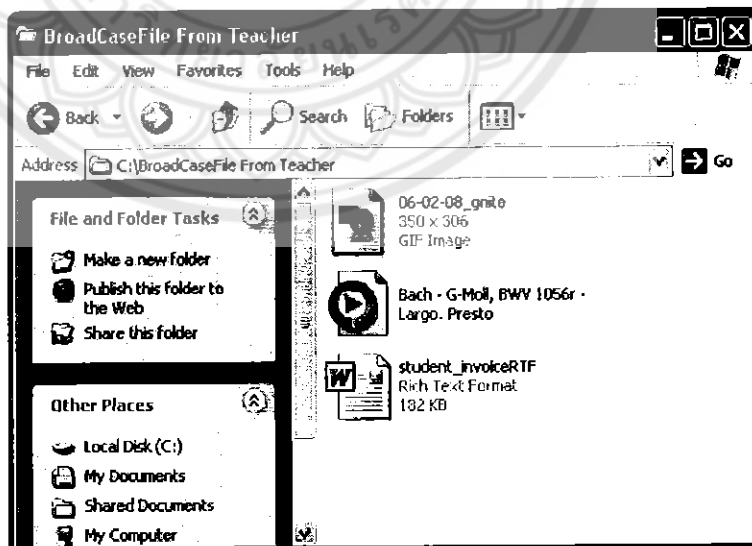


รูปที่ 4.10 ไฟล์ที่ส่งในห้องสนทนา

8. ซอฟต์แวร์สามารถส่งไฟล์แบบกระจายไปยังสมาชิกทุกคนภายในแต่ละห้องสนทนาได้

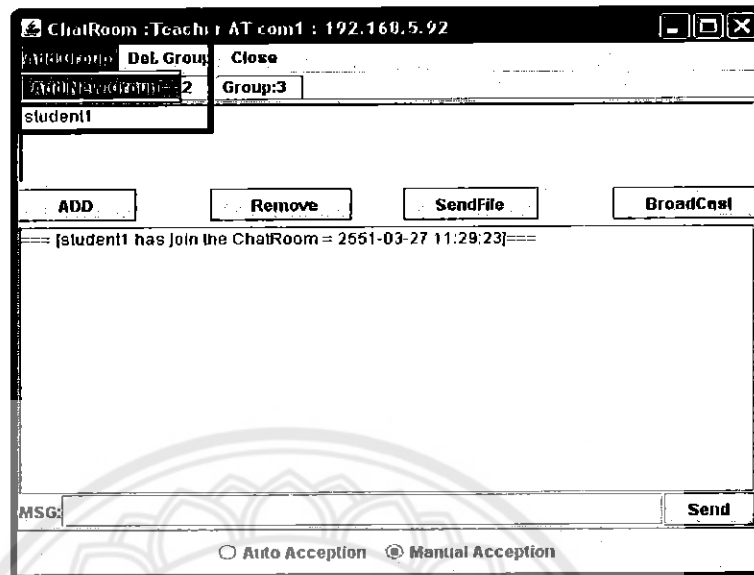


รูปที่ 4.11 การ Broadcast ในห้องสนทนา

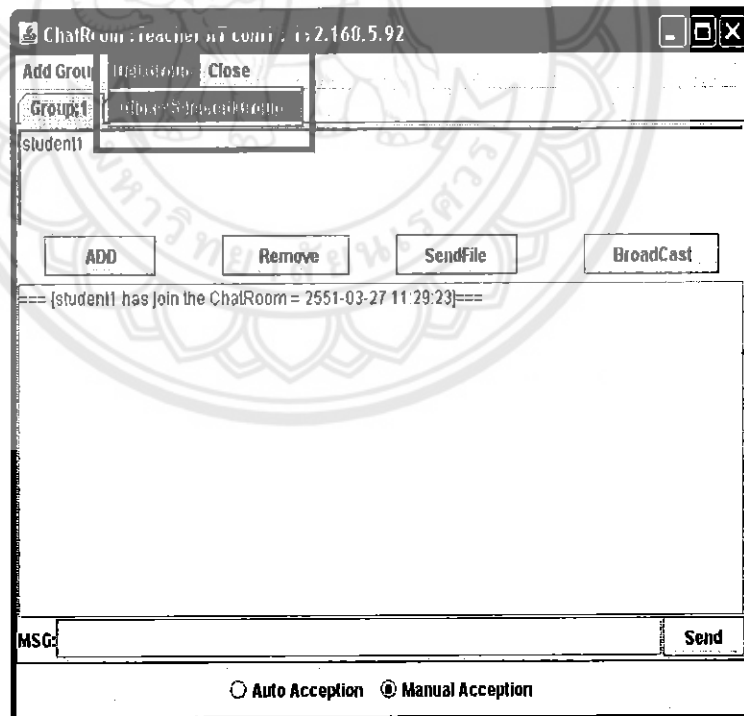


รูปที่ 4.12 ไฟล์ที่ Broadcast ในห้องสนทนา

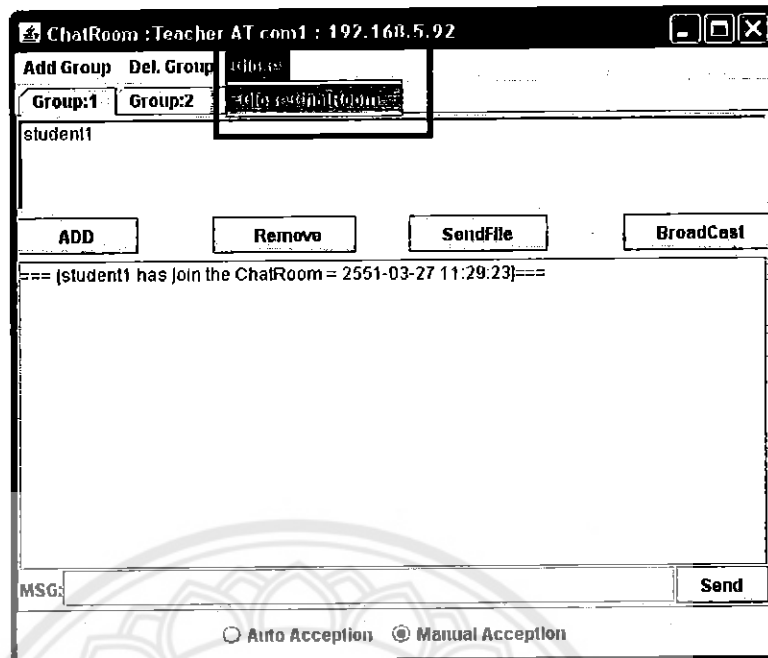
9. ซอฟต์แวร์สามารถเพิ่มห้องสนทนา ลบ และปิดห้องสนทนาตามต้องการได้



รูปที่ 4.13 การเพิ่มห้องสนทนา



รูปที่ 4.14 การลบห้องสนทนา



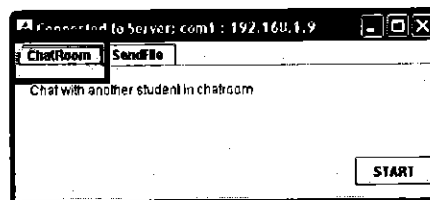
รูปที่ 4.15 การปิดห้องสนทนา

4.1.2 การทดสอบ Client

การทดสอบการทำงานของ Client เป็นการทดสอบการทำงานของ Client ในการเข้าห้อง และการสนทนาภายในห้อง และการส่งไฟล์ไปที่เครื่อง Server โดยทดสอบการทำงานตามข้อกำหนดของซอฟต์แวร์ (Specification) ดังนี้

ข้อกำหนดของซอฟต์แวร์ (Specification)

1. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์การสร้างห้องสนทนาเพื่อใช้ในการติดต่อสื่อสารกับเครื่องอื่นๆได้



รูปที่ 4.16 เลือกการทำงาน Chat Room ของ Client

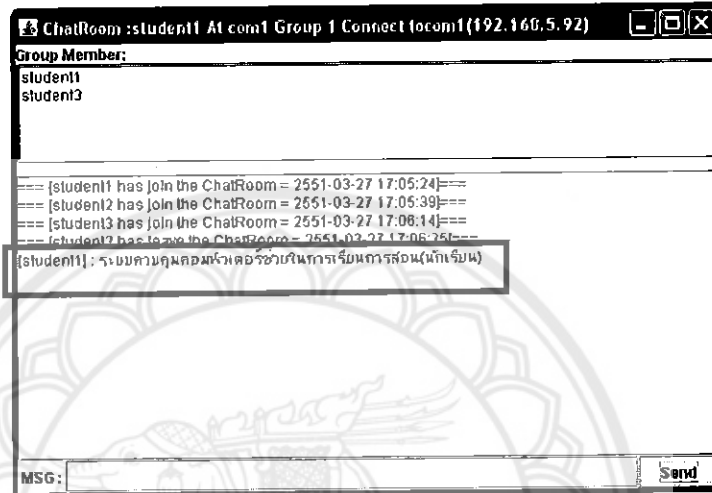


รูปที่ 4.17 ใส่ชื่อของ Client



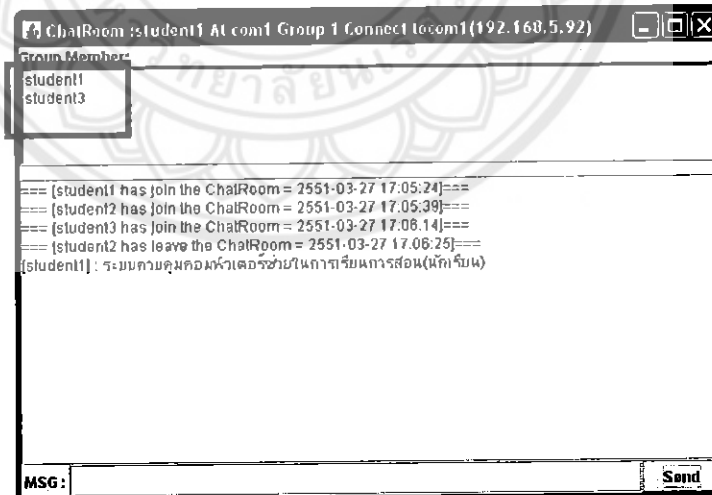
รูปที่ 4.18 เลือกห้องสนทนาของ Client

2. ซอฟต์แวร์สามารถรับส่งข้อความสนทนาได้



รูปที่ 4.19 ห้องสนทนาของ Client

3. ซอฟต์แวร์สามารถแสดงรายชื่อสมาชิกที่อยู่ภายในห้องสนทนาได้



รูปที่ 4.20 รายชื่อสมาชิกที่อยู่ภายในสนทนาของ Client

4.2 โปรแกรมการรับส่งไฟล์ (File Server)

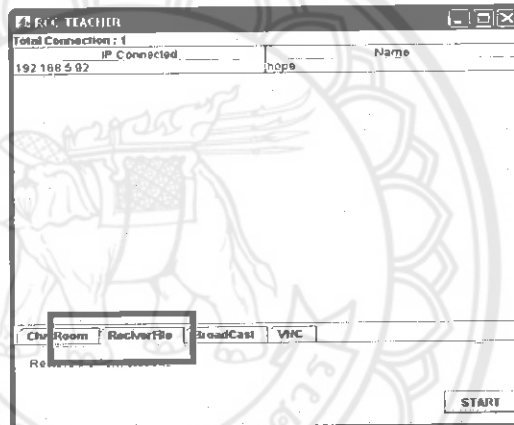
การทดสอบแบ่งออกเป็น 2 ฟังก์ชัน คือ File Server – Server และ File Server – Client โดยที่โปรแกรม File Server – Server จะทำการสร้าง Server เพื่อรอรับการติดต่อเพื่อส่งไฟล์จากโปรแกรม File Server – Client

4.2.1 การทดสอบ Server

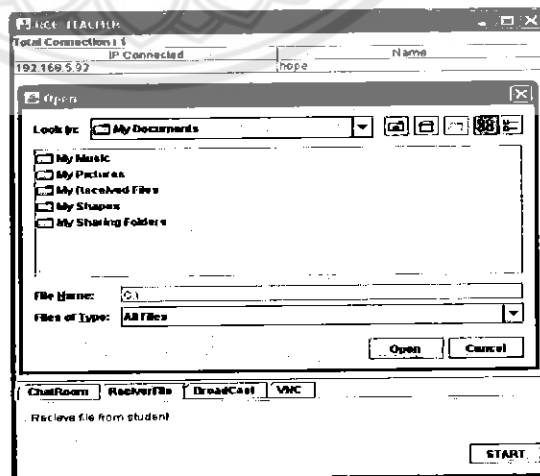
การทดสอบการทำงานของ Server เป็นการทดสอบการรับไฟล์จาก Client โดยจะต้องเลือกที่จัดเก็บไฟล์ไว้รอรับ ไฟล์ที่ Client ส่งมา

ข้อกำหนดของซอฟต์แวร์ (Specification)

1. ซอฟต์แวร์สามารถเลือก Directory ที่ต้องการเพื่อใช้ในการเก็บไฟล์จาก File Server – Client ได้ตามต้องการ

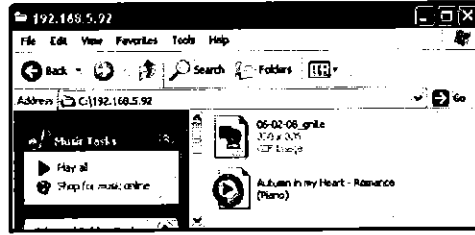


รูปที่ 4.21 เปิดการทำงาน Receiver File



รูปที่ 4.22 เลือกที่เก็บไฟล์

2. ซอฟต์แวร์สามารถรับและเขียนไฟล์ได้อย่างถูกต้อง

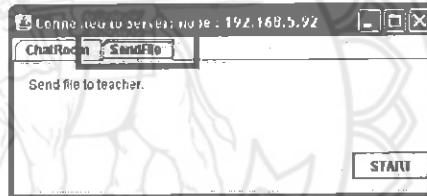


รูปที่ 4.23 ไฟล์ที่ซอฟต์แวร์รับและเขียนเก็บไว้ใน Directory ที่ผู้ใช้เลือกไว้

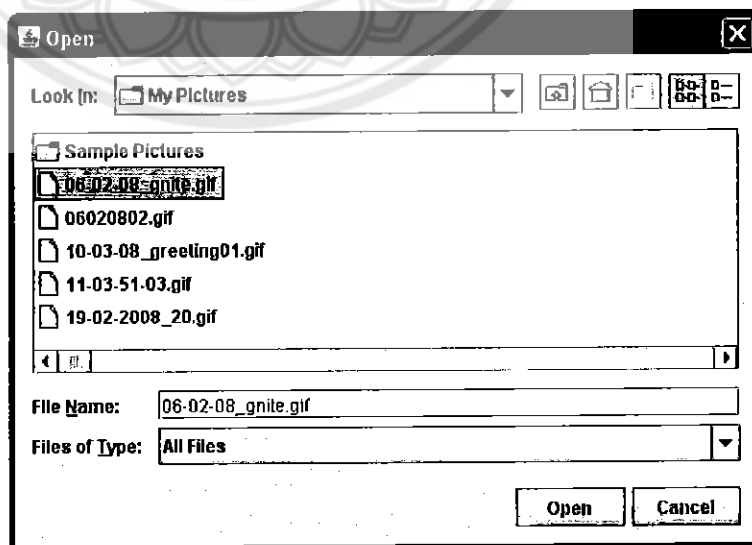
4.2.2 การทดสอบ Client

การทดสอบการทำงานของ Client เป็นการทดสอบการส่งไฟล์ไปยัง Server โดยไฟล์จะถูกเก็บไว้ในโฟลเดอร์ที่ Server สร้างไว้รอ

1. ซอฟต์แวร์สามารถส่งไฟล์ที่ต้องการได้อย่างถูกต้อง



รูปที่ 4.24 เลือกการทำงาน Send File



รูปที่ 4.25 เลือกการทำงาน Send File



รูปที่ 4.26 ไฟล์ที่ Client ส่งไปและถูกเก็บไว้ใน Server

4.3 การทดสอบการทำงานของ Broadcast

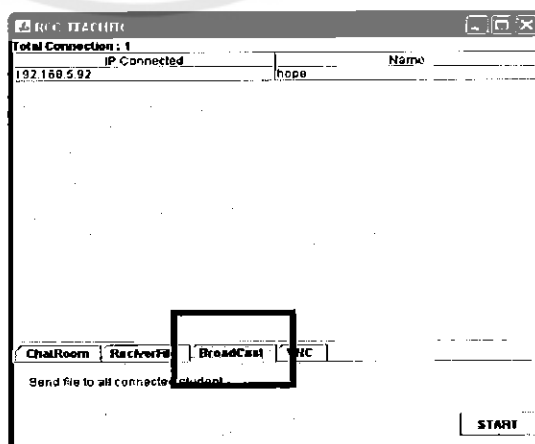
การทดสอบการทำงานของ Broadcast ของซอฟต์แวร์ระบบควบคุมคอมพิวเตอร์ช่วยในการเรียนการสอนนั้นเป็นการทดสอบการกระจายไฟล์ แบ่งการทดสอบเป็น 2 ส่วนคือ การทำงานของ server และ client ดังนี้

4.3.1 การทดสอบ Server

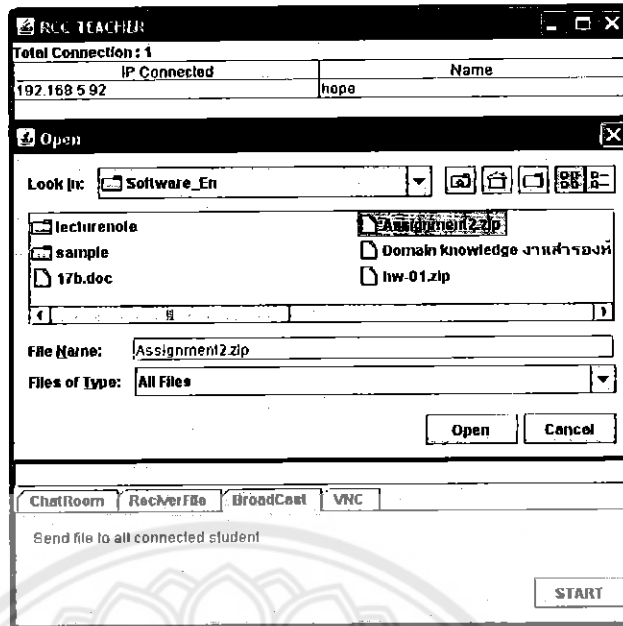
การทดสอบการทำงานของ Server เป็นการทดสอบการกระจายไฟล์ไปยัง Client โดย Server จะเลือกไฟล์ที่ต้องการ (สามารถเลือกได้หลายไฟล์) แล้วกระจายไฟล์ไปยัง Client ทั้งหมดที่อยู่ใน Connection ในขณะนั้น

ข้อกำหนดของซอฟต์แวร์ (Specification)

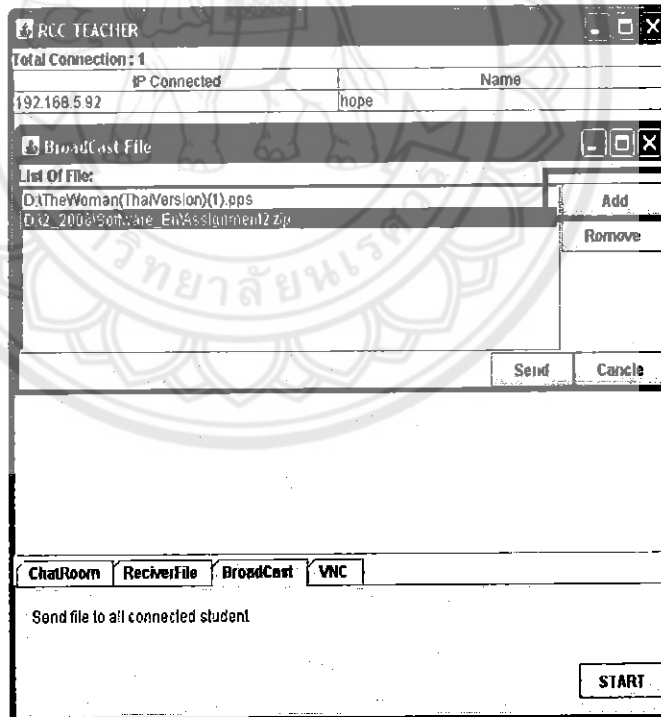
1. ซอฟต์แวร์สามารถเพิ่มรายชื่อไฟล์ที่ผู้ใช้เลือกลงในรายการไฟล์ที่ต้องการกระจายไฟล์ได้



รูปที่ 4.27 เลือกการทำงาน Broadcast

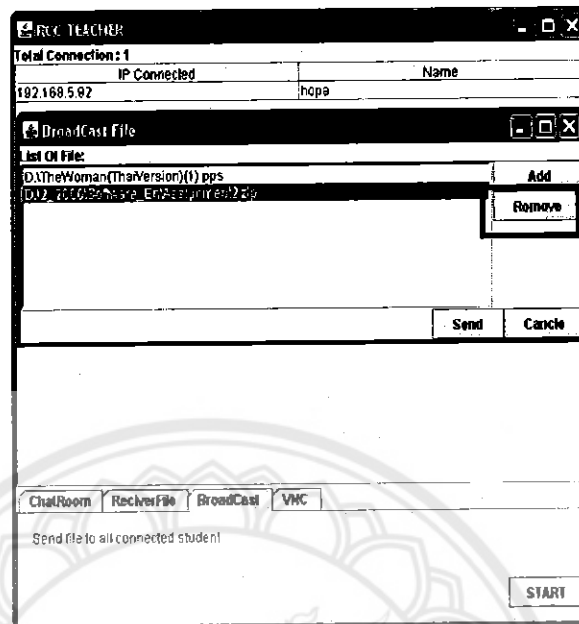


รูปที่ 4.28 เลือกไฟล์ที่ต้องการ Broadcast



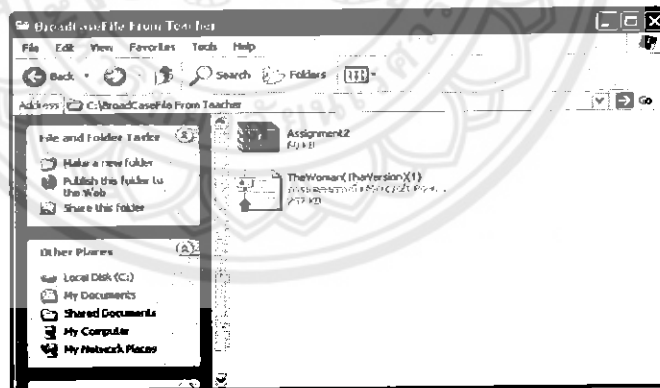
รูปที่ 4.29 เพิ่มไฟล์ในรายการสำหรับ Broadcast

2. ซอฟต์แวร์สามารถลบรายชื่อไฟล์ที่ผู้ใช้เลือกในรายการไฟล์ที่ต้องการกระจายไฟล์ได้



รูปที่ 4.30 ลบไฟล์ในรายการสำหรับ Broadcast

3. ซอฟต์แวร์สามารถกระจายไฟล์ตามที่ผู้ใช้ต้องการได้



รูปที่ 4.31 ไฟล์ที่ Broadcast และถูกเก็บไว้ที่ Client

4. ซอฟต์แวร์สามารถบอกถึงสถานะและความคืบหน้าของการทำงานได้



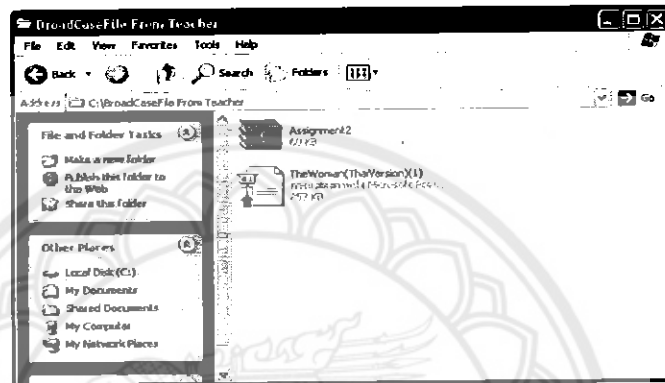
รูปที่ 4.32 สถานะและความคืบหน้าของการส่งไฟล์

4.3.2 การทดสอบ Client

การทดสอบการทำงานของ Client เป็นการตรวจสอบว่า Client ได้รับไฟล์ที่ Server ทำการ Broadcast และเก็บไฟล์นั้นไว้ใน Client โดยทดสอบตามข้อกำหนดของซอฟต์แวร์ (Specification) ดังนี้

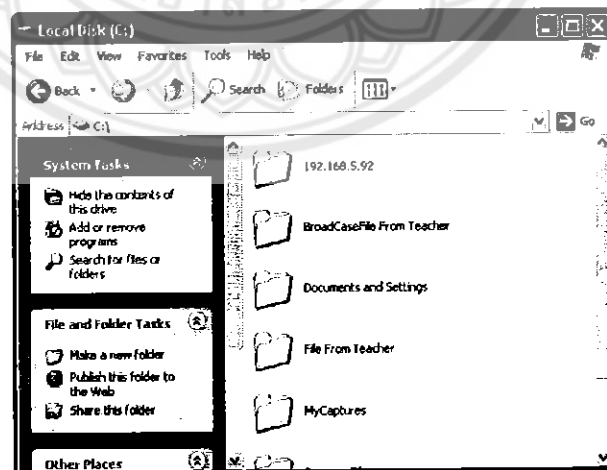
ข้อกำหนดของซอฟต์แวร์ (Specification)

1 ซอฟต์แวร์สามารถรับและเขียนไฟล์ที่ Server ทำการ Broadcast มาได้



รูปที่ 4.33 ไฟล์ที่รับจากการ Broadcast และเขียนเก็บไว้ใน Client

2 ซอฟต์แวร์สามารถสร้าง Directory เพื่อใช้ในการเก็บไฟล์ที่รับมาจากการ Broadcast ไว้ที่เดียวกัน



รูปที่ 4.34 Directory ที่สร้างเพื่อเก็บไฟล์ที่รับมาจากการ Broadcast

4.4 การทดสอบการทำงานของ VNC

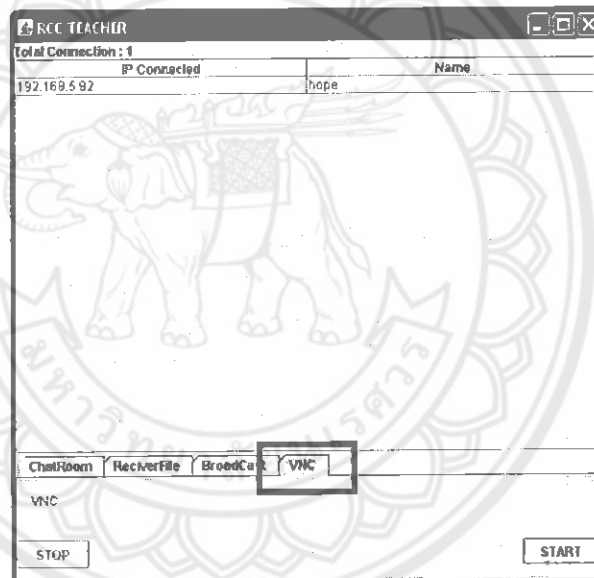
การทดสอบการทำงานของ VNC ของซอฟต์แวร์ระบบควบคุมคอมพิวเตอร์ช่วยในการเรียนการสอนนั้นเป็นการทดสอบการใช้งาน VNC คือ ให้ Client สามารถดูหน้าจอของ Server ได้ แบ่งการทดสอบเป็น 2 ส่วนคือ การทำงานของ server และ client ดังนี้

4.4.1 การทดสอบ Server

การทดสอบการทำงานของ Server เป็นการทดสอบว่า Server สามารถใช้งาน VNC ได้ โดยทดสอบตามข้อกำหนดของซอฟต์แวร์ (Specification) ดังนี้

ข้อกำหนดของซอฟต์แวร์ (Specification)

1. ซอฟต์แวร์สามารถให้เครื่อง Client เห็นหน้าจอของเครื่อง Server ตามที่ต้องการ



รูปที่ 4.35 เลือกการทำงาน VNC

2. ซอฟต์แวร์สามารถควบคุมให้ Client ที่เปิดโปรแกรม ช่วยควบคุมการเรียนการสอน เปิด-ปิด โปรแกรม RCL VNC ได้ตามที่ต้องการ
3. ซอฟต์แวร์สามารถแสดงให้ผู้ใช้รู้ได้ถึงสถานะการใช้คำสั่งให้ Client เปิด หรือ ปิด โปรแกรม RCL VNC

4.4.2 การทดสอบ Client

การทดสอบการทำงานของ Server เป็นการทดสอบว่า Client สามารถเห็นหน้าจอของ Server ตามข้อกำหนดของซอฟต์แวร์ (Specification) ดังนี้

ข้อกำหนดของซอฟต์แวร์(Specification)

1. ซอฟต์แวร์สามารถดูหน้าจอของเครื่อง Server ได้ โดยที่การเปิด - ปิดโปรแกรมนั้นขึ้นอยู่กับความต้องการของผู้ใช้งาน RCL VNC - Server



บทที่ 5

บทสรุป

ซอฟต์แวร์ช่วยในการเรียนการสอนสามารถนำไปใช้งานกับระบบ Network และสามารถใช้งานได้จริงตาม ข้อกำหนด (Specification) ของซอฟต์แวร์ที่กำหนดไว้

5.1 วิเคราะห์ผลการทดลอง

จากการทดลองในบทที่ 4 ซึ่งเป็นการทดลองใช้งานตาม Specification ของซอฟต์แวร์ที่ทำการกำหนดไว้ในระบบ Network จริง ได้ผลดังตารางต่อไปนี้

ตารางที่ 5.1 ผลการทดสอบซอฟต์แวร์ตาม Specification ของซอฟต์แวร์แต่ละส่วน

ซอฟต์แวร์	Specification	การทำงาน
RCL – Server	1. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์การสร้างห้องสนทนาเพื่อใช้ในการติดต่อสื่อสารกับเครื่องอื่นๆ ได้	✓
	2. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์การรับไฟล์จากซอฟต์แวร์ ของฝั่งClient ได้	✓
	3. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์การส่งไฟล์แบบกระจาย เพื่อส่งไฟล์ที่ต้องการ ไปยังเครื่อง Client ที่ติดต่อได้	✓
	4. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์การดูหน้าจอเครื่องแม่ข่ายได้	✓
	5. ซอฟต์แวร์มีส่วนแสดงให้รู้ถึงรายละเอียดของเครื่อง Client ที่ติดต่อเข้ามา	✓
	6. ซอฟต์แวร์มีข้อความที่บอกถึงรายละเอียดของซอฟต์แวร์ย่อยแต่ละส่วน	✓
	7. ซอฟต์แวร์สามารถส่งคำสั่งต่างๆที่ใช้ในการติดต่อกับเครื่อง Client ได้	✓
RCL – Client	1. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์การสร้างห้องสนทนาเพื่อใช้ในการติดต่อสื่อสารกับเครื่องอื่นๆ ได้	✓

	2. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์การรับส่งไฟล์ เพื่อส่งไฟล์ที่ต้องการไปยังเครื่อง Server ได้อย่างถูกต้อง	✓
	3. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์เพื่อ รับไฟล์ที่เครื่องแม่ข่ายส่งไฟล์แบบกระจายมาได้	✓
	4. ซอฟต์แวร์สามารถเรียกใช้ซอฟต์แวร์การดูหน้าจอเครื่องแม่ข่ายได้	✓
	5. ซอฟต์แวร์สามารถค้นหาเครื่อง Server เพื่อติดต่อได้เองโดยอัตโนมัติ	✓
	6. ซอฟต์แวร์สามารถรองรับคำสั่งต่างๆจากซอฟต์แวร์ ของเครื่อง Server ได้	✓
	7. ซอฟต์แวร์มีส่วนที่แสดงรายละเอียดของซอฟต์แวร์ย่อยแต่ละส่วน	✓
Chat Room- Server	1. ซอฟต์แวร์สามารถกำหนดชื่อผู้ใช้ตามที่ผู้ใช้ต้องการเพื่อใช้ในการ สนทนาได้	✓
	2. ซอฟต์แวร์สามารถสร้างห้องสนทนาตามจำนวนที่ผู้ใช้ต้องการได้	✓
	3. ซอฟต์แวร์สามารถตอบรับหรือปฏิเสธการขอเข้าร่วมสนทนาของ Client ตามความต้องการของผู้ใช้ได้	✓
	4. ซอฟต์แวร์สามารถแสดงรายชื่อของสมาชิกภายในแต่ละห้องสนทนาได้	✓
	5. ซอฟต์แวร์สามารถเรียก Client ที่ต้องการเข้าสู่ห้องสนทนาได้	✓
	6. ซอฟต์แวร์สามารถกำหนดให้สมาชิกภายในห้องสนทนาที่ต้องการ ออกจากห้องสนทนาได้ตามต้องการ	✓
	7. ซอฟต์แวร์สามารถส่งไฟล์ที่ต้องการไปยังสมาชิกภายในห้องสนทนาได้ตามต้องการ	✓
	8. ซอฟต์แวร์สามารถส่งไฟล์แบบกระจายไปยังสมาชิกทุกคนภายในแต่ละห้องสนทนาได้	✓
	9. ซอฟต์แวร์สามารถเพิ่มห้องสนทนา หรือ ปิดห้องสนทนาตามต้องการได้	✓
	10. ซอฟต์แวร์สามารถรับส่งข้อความสนทนาได้	

Chat Room Client	1. ซอฟต์แวร์สามารถค้นหา Chat Room Server เพื่อติดต่อขอเข้าร่วมสนทนาได้โดยอัตโนมัติ	✓
	2. ซอฟต์แวร์สามารถสร้างห้องสนทนาเพื่อสนทนากับห้องที่ต้องการได้	✓
	3. ซอฟต์แวร์สามารถรับส่งข้อความสนทนาได้	
	4. ซอฟต์แวร์สามารถแสดงรายชื่อของสมาชิกที่อยู่ภายในห้องสนทนาได้	✓
	5. ซอฟต์แวร์สามารถตอบรับคำสั่งที่มาจาก Chat Room Server ได้	✓
File Server – Server	1. ซอฟต์แวร์สามารถเลือก Directory ที่ต้องการเพื่อใช้ในการเก็บไฟล์จาก File Server – Client ได้ตามต้องการ	✓
	2. ซอฟต์แวร์สามารถรับและเขียนไฟล์ได้อย่างถูกต้อง	
	3. ซอฟต์แวร์สามารถรับการติดต่อจาก File Server – Client ได้พร้อมๆกัน	✓
File Server – Client	1. ซอฟต์แวร์สามารถส่งไฟล์ที่ต้องการได้อย่างถูกต้อง	✓
	2. ซอฟต์แวร์สามารถค้นหาและติดต่อไปยัง Server ของ File Server ได้เอง	✓
	3. ซอฟต์แวร์สามารถแสดงข้อความเพื่อแสดงให้เห็นถึงสถานะของการติดต่อและส่งไฟล์ได้	✓
Broadcast – Server	1. ซอฟต์แวร์สามารถเพิ่มรายชื่อไฟล์ที่ผู้ใช้เลือกลงในรายการไฟล์ที่ต้องการกระจายไฟล์ได้	✓
	2. ซอฟต์แวร์สามารถลบรายชื่อไฟล์ที่ผู้ใช้เลือกในรายการไฟล์ที่ต้องการกระจายไฟล์ได้	✓
	3. ซอฟต์แวร์สามารถกระจายไฟล์ตามที่ผู้ใช้ต้องการได้	✓
	4. ซอฟต์แวร์สามารถบอกถึงสถานะและความคืบหน้าของการทำงานได้	✓
Broadcast – Client	1. ซอฟต์แวร์สามารถรับและเขียนไฟล์ที่ Server ทำการ Broadcast มาได้	✓
	2. ซอฟต์แวร์สามารถสร้าง Directory เพื่อใช้ในการเก็บไฟล์ที่รับมาจากการ Broadcast ไว้ที่เดียวกัน	✓

RCL VNC – Server	<ol style="list-style-type: none"> 1. ซอฟต์แวร์สามารถให้เครื่อง Client เห็นหน้าจอของเครื่อง Server ตามที่ต้องการ 2. ซอฟต์แวร์สามารถควบคุมให้ Client ที่เปิด โปรแกรม ช่วยควบคุมการเรียนการสอน เปิด – ปิด โปรแกรม RCL VNC ได้ตามที่ต้องการ 	✓
	<ol style="list-style-type: none"> 3. ซอฟต์แวร์สามารถแสดงให้ผู้ใช้รู้ได้ถึงสถานะการใช้คำสั่งให้ Client เปิด หรือ ปิด โปรแกรม RCL VNC 	✓
RCL VNC – Client	<ol style="list-style-type: none"> 1. ซอฟต์แวร์สามารถดูหน้าจอของเครื่อง Server ได้ โดยที่การเปิด – ปิด โปรแกรม นั้น ขึ้นอยู่กับความต้องการของผู้ใช้งาน RCL VNC – Server 	✓
	<ol style="list-style-type: none"> 2. ซอฟต์แวร์ไม่สามารถเปิดได้โดยผู้ใช้งาน RCL VNC – Client 	✓

จากตาราง 5.1 เครื่องหมายถูกหมายถึง ซอฟต์แวร์สามารถใช้งานได้ตรงกับ Specification ที่กำหนดไว้ได้ ยกเว้น Chat Room Server – Spec.10 , Chat Room Client – Spec.3 , File Server – Server Spec.2 ซึ่งทำงานไม่ตรงกับ Specification ที่กำหนดไว้ คือ มีข้อจำกัดในเรื่อง การรองรับภาษาของเครื่อง คือ เครื่องที่ไม่รองรับภาษาไทยจะไม่สามารถรับไฟล์ที่มีชื่อเป็นภาษาไทยและข้อความสนทนาที่เป็นภาษาไทยได้

5.2 ปัญหาและการแก้ไข

1. ความรู้เรื่อง Network Programming ที่ใช้ในการเขียนซอฟต์แวร์

การแก้ไข : ศึกษาข้อมูลจากทาง Internet และฝึกเขียนซอฟต์แวร์อื่นๆที่มีตัวอย่างเพื่อให้เกิดความรู้ความเข้าใจยิ่งขึ้น

2. Multicast Socket ของ จาวา ไม่สามารถทำงานได้โดยไม่รู้สาเหตุ โดยปัญหานี้เกิดขึ้นระหว่างการทดลอง การใช้งานซอฟต์แวร์กับเครื่องอื่นๆ คือ ซอฟต์แวร์ในส่วนที่มีการใช้งาน Multicast Socket นั้นไม่สามารถทำงานได้กับเครื่องอื่นๆบางเครื่อง โดยที่ทั้ง หมาดใช้ Java version เดียวกัน ซึ่งสามารถแสดงเป็นตารางดังนี้

ตารางที่ 5.2 ปัญหา Multicast Socket ที่เกิดขึ้น

เครื่องที่	OS	Network	การใช้งานในส่วน Multicast Socket
1	Window XP SP2	ไม่ต่อ	ไม่สามารถใช้งานได้
1	Window XP SP2	ต่อ	สามารถใช้งานได้
2	Window XP SP2	ไม่ต่อ	ไม่สามารถใช้งานได้
2	Window XP SP2	ต่อ	ไม่สามารถใช้งานได้
3	Window XP SP2	ไม่ต่อ	สามารถใช้งานได้
3	Window XP SP2	ต่อ	สามารถใช้งานได้

การแก้ไข: ไม่สามารถแก้ไขได้เพราะไม่ทราบสาเหตุ

5.3 สรุปผลการทดลอง

ซอฟต์แวร์ช่วยควบคุมการเรียนการสอน เป็นซอฟต์แวร์ที่สามารถนำไปใช้งานภายในระบบ Network ได้จริง โดยการทำงานแบ่งออกเป็น 2 ฟังก์ชัน คือ ฟังก์ชัน Server และฟังก์ชัน Client โดย Server สามารถติดต่อส่งการควบคุมการใช้งานไปยัง Client ได้ และการทำงานฟังก์ชัน Client นั้น หาก Server ไม่อนุญาตให้ใช้งาน Client ก็จะไม่สามารถใช้งานได้

5.4 ข้อเสนอแนะ

1. การสร้าง Connection ระหว่าง Student กับ Teacher สามารถพัฒนาให้มี Option ที่เลือกได้ว่าจะ Automatic Search (Broadcast signal) หรือ Fixed IP ของ Teacher ในโปรแกรม Student
2. ซอฟต์แวร์นี้สามารถนำไปพัฒนาต่อเพื่อหา กระบวนการ Broadcast ไฟล์ ที่มีความสัมพันธ์กันระหว่าง ขนาดของไฟล์ ขนาดของ Datagram Packet และระยะเวลาห่างของการส่ง แต่ละ Packet และตัวแปรอื่นๆ เพื่อที่จะได้ ซอฟต์แวร์ในการ Broadcast ไฟล์ที่มีประสิทธิภาพที่สูงขึ้น

เอกสารอ้างอิง

- [1] Abraham Silberschatz, Peter Baer Galvin and Greg Gagne. **Operating System Concepts**. 6th Edition. New York: John Wiley and Sons Inc. 2003.
- [2] Chard Darby, John Griffin, Pascal den Haan, Peter den Haan, Alexander V. Konstantinou, Sing Li, Sean MacLean, Glenn E. Mitchell, Joel Peach, Peter Wansch, William Wright. **Beginning Java Networking**. : Wrox Press Ltd. 2001.
- [3] ศุภามน วาณิชย์ก่อกุล, สิทธิกาณจน์ ปิยมพรชัย, ชวลิต ทินกรสูติบุตร, เลอศักดิ์ ลิ้มวิวัฒน์กุล. “ความรู้พื้นฐานเกี่ยวกับ โปรโตคอล TCP/IP” . [Online]. Available: <http://www.thaicert.nectec.or.th/paper/basic/tcp-ip.php>. 2548
- [4] วีระศักดิ์ ชิ่งถาวร. **Java Programming Volume I**. กรุงเทพมหานคร : ซีเอ็ดดูเกชั่น. 2546.
- [5] นพปฎล กุลจรรยาวิวัฒน์. “Port”. [Online]. Available: <http://www.thaicert.nectec.or.th/paper/basic/port.php>. 2544

ประวัติผู้เขียนโครงการ



ชื่อ นางสาวจिरารัตน์ เอี่ยมสะอาด
ภูมิลำเนา 44/1 หมู่ 7 ต.บางขาว อ.พยุหะคีรี
 จังหวัดนครสวรรค์ 61300

ประวัติการศึกษา

- จบการศึกษาระดับมัธยมศึกษาจาก โรงเรียนโกรกพระ
- ปัจจุบันกำลังศึกษาอยู่ชั้นปีที่ 4
 สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
 มหาวิทยาลัยนเรศวร

E-mail jirarati@hotmail.com



ชื่อ นายชินภัทร คำแน่น
ภูมิลำเนา 230/2 หมู่ 13 ต.ประทาย อ.ประทาย จ.นครราชสีมา 30180

ประวัติการศึกษา

- จบการศึกษาระดับมัธยมศึกษาจาก
 โรงเรียนเบญจมะมหาราช
- ปัจจุบันกำลังศึกษาอยู่ชั้นปีที่ 4
 สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
 มหาวิทยาลัยนเรศวร

E-mail pm_chin@hotmail.com



ชื่อ นายเรืองฤทธิ์ บุญรอด
ภูมิลำเนา 26/1 หมู่ 14 ต.ท่าโรง อ.วิเชียรบุรี จ.เพชรบูรณ์ 67130

ประวัติการศึกษา

- จบการศึกษาระดับมัธยมศึกษาจาก
 โรงเรียนศิลปอนุสรณ์
- ปัจจุบันกำลังศึกษาอยู่ชั้นปีที่ 4
 สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
 มหาวิทยาลัยนเรศวร

E-mail ritchie_rit@hotmail.com