

การพัฒนาเกมออนไลน์โดยใช้สถาปัตยกรรมเครือข่ายแบบผสม

DEVELOPMENT OF AN ONLINE GAME BASED ON A HYBRID
NETWORK ARCHITECTURE



H003482

โดย

สิทธิกร คุณวิโรจน์พานิช

SITTIKORN KHUNVIROJPANICH

อาจารย์ที่ปรึกษา

รศ.ดร.นพพร โชติกกำธร

วัน เดือน ปี.....	04 S.ศ. 2550
เลขทะเบียน.....	H003482
เลขเรียกหนังสือ.....	อา. ๘721ก 2549
"ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล."	

611940944
11717742

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ภายในห้องสมุดนี้ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
แม้ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**DEVELOPMENT OF AN ONLINE GAME BASED ON A HYBRID
NETWORK ARCHITECTURE**



A SYSTEM DEVELOPMENT PROJECT

OF THE REQUIREMENT FOR THE DEGREE OF

MASTER OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY

FACULTY OF INFORMATION TECNOLGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานี้เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
2/ 2006
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COPYRIGHT 2007

FACULTY OF INFORMATION TECHNOLOGY

เอกสารนี้เป็นทรัพย์สินทางปัญญาของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง การคัดลอกหรือการนำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาตจากสถาบันฯ ถือว่าผิดกฎหมาย
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อ	การพัฒนาเกมออนไลน์โดยใช้สถาปัตยกรรมเครือข่ายแบบผสม
นักศึกษา	นายสิทธิกร คุณวิโรจน์พานิช
รหัสนักศึกษา	47066415
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2549
อาจารย์ที่ปรึกษา	รศ.ดร.นพพร โชติกำจร

บทคัดย่อ

รายงานโครงการพิเศษฉบับนี้เสนอผลการพัฒนาระบบเกมออนไลน์โดยใช้สถาปัตยกรรมเครือข่ายแบบผสม อันเป็นการนำสถาปัตยกรรมเครือข่ายแบบไคลเอนท์และเซิร์ฟเวอร์(Client and Server) และสถาปัตยกรรมเครือข่ายแบบเพียร์-ทู-เพียร์(Peer-to-Peer) มาประยุกต์ใช้พัฒนาเกมออนไลน์แบบมีผู้เล่นจำนวนมาก โดยมีจุดประสงค์เพื่อลดการทำงานในฝั่งเซิร์ฟเวอร์จากการแบ่งส่วนการประมวลผลบางอย่างให้ไคลเอนท์ประมวลผลด้วยตัวเอง ซึ่งมีผลทำให้เซิร์ฟเวอร์สามารถรองรับจำนวนผู้เล่นได้มากขึ้น ระบบแบ่งรูปแบบการเชื่อมต่อออกเป็น 2 ส่วน อันได้แก่การติดต่อระหว่างไคลเอนท์และเซิร์ฟเวอร์ เป็นส่วนที่เกี่ยวข้องกับข้อมูลของผู้เล่นและการรักษาความปลอดภัย และส่วนที่เป็นการติดต่อแบบเพียร์-ทู-เพียร์ เป็นส่วนที่ไคลเอนท์ทำการสื่อสารถึงกันเองโดยไม่ผ่านเซิร์ฟเวอร์ โดยอาศัยไคลเอนท์ตัวใดตัวหนึ่งเปิดบริการเป็น โฮสต์ขึ้นมาเพื่อใช้เป็นตัวกลางแลกเปลี่ยนและกระจายข้อมูลระหว่างเพียร์ที่เชื่อมต่อเข้ามา ซึ่งบริการของโฮสต์ที่สนับสนุน ได้แก่ การพูดคุยระหว่างเพียร์(Chat) การเคลื่อนที่ตัวละครของผู้เล่นที่เป็นเพียร์ และการใช้ข้อมูลต่างๆ ระหว่างเพียร์ร่วมกัน โดยโครงการพิเศษนี้พัฒนาโดยใช้โคเรคเอ็กซ์(DirectX) จัดการเกี่ยวกับการแสดงผลและการติดต่อกับผู้ใช้(Input) และวินซ็อก(Winsock) จัดการเกี่ยวกับระบบเน็ตเวิร์ค

Title	Development of an Online Game based on a Hybrid Network Architecture
Student	Mr. Sittikorn Khunvirojpanich
Student ID.	47066415
Degree	Master of Science
Programme	Information Science
Academic Year	2006
Advisor	Assoc. Prof. Dr. Nopporn Chotikakamthorn

ABSTRACT

This project document represents the result of an online game development using hybrid network architecture. The salient feature of the proposed method is the use of the peer-to-peer architecture to process some information by itself. It can decrease high processing on the server side so that the server can support more players than an online game system using the client-server architecture. The hybrid network architecture separates the connections into 2 parts, the client-server and peer-to-peer. The client-server connection provides game security and guarantee user data. The peer-to-peer connection allows other clients to share the common environment of a host machine. It can exchange data without association on the server side. The host provides services such as chatting, character movement and sharing files among peers. This project was developed using DirectX and Winsock API to provide the graphics display, interaction and network connection respectively.

กิตติกรรมประกาศ

โครงการพัฒนาระบบงานนี้สำเร็จได้ ด้วยคำแนะนำ คำปรึกษาและความเอาใจใส่จาก
รศ.ดร.นพพร โชติกคำธร ซึ่งเป็นอาจารย์ที่ปรึกษา ข้าพเจ้ารู้สึกทราบดีซึ่งในความอนุเคราะห์ และ
ขอขอบพระคุณเป็นอย่างสูง

ขอกราบขอบพระคุณอาจารย์คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้า
เจ้าคุณทหารลาดกระบัง ทุกๆ ท่านที่ได้ให้ความรู้แก่ข้าพเจ้า

ขอขอบคุณ บัณฑิตวิทยาลัย และคณะเทคโนโลยีสารสนเทศ ที่ได้ให้ความช่วยเหลือใน
เรื่องต่างๆ ขอขอบคุณเพื่อนๆ ในรุ่น พี่ๆ ทุกคนที่คอยเป็นกำลังใจในการทำงานตั้งแต่ต้น

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้าที่เป็น
กำลังใจ และให้การสนับสนุนในทุกเรื่องๆ ทำให้ข้าพเจ้าสามารถทำโครงการพัฒนาระบบงานฉบับ
นี้สำเร็จลุล่วงด้วยดี

สิทธิกร คุณวิโรจน์พานิช

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VIII
สารบัญรูป.....	IX
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 ความมุ่งหมาย และวัตถุประสงค์ของการศึกษา.....	2
1.3 สมมุติฐานของการศึกษา.....	2
1.4 ทฤษฎีหรือแนวความคิดที่ใช้ในโครงการพิเศษ.....	3
1.5 การเปรียบเทียบระหว่างวิธีที่นำเสนอกับวิธีการแบบพื้นฐาน.....	3
1.6 ขอบเขตของโครงการพิเศษ.....	4
1.7 ขั้นตอนของการศึกษา.....	4
1.8 ประโยชน์ที่คาดว่าจะได้รับ.....	5
บทที่ 2 หลักการและทฤษฎีที่เกี่ยวข้อง.....	6
2.1 ลักษณะของเกมออนไลน์.....	6
2.2 สถาปัตยกรรมเน็ตเวิร์คที่ใช้พัฒนาเกม.....	7
2.2.1 สถาปัตยกรรมเน็ตเวิร์คแบบเพียร์-ทู-เพียร์.....	7
2.2.2 สถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนท์และเซิร์ฟเวอร์.....	9
2.3 ความแตกต่างของเกมที่พัฒนาโดยใช้สถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนท์ และเซิร์ฟเวอร์กับเพียร์-ทู-เพียร์.....	10
2.3.1 การทำงานของสถาปัตยกรรมเน็ตเวิร์คแบบเพียร์-ทู-เพียร์.....	10
2.3.2 การทำงานของสถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนท์และเซิร์ฟเวอร์.....	11
2.3.3 ข้อเด่นและข้อด้อย.....	12
2.4 วิเคราะห์เกมออนไลน์ที่ได้รับความนิยมในประเทศไทย.....	13
2.4.1 เกมเรอ์กนาร์เรือคอนไลน์.....	13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

สารบัญ (ต่อ)

	หน้า
2.4.2 เกมยุทธภพครบสลัง	14
2.4.3 เกมปิงปอง	15
2.4.4 เกมวอร์ครีฟ	16
2.5 วิเคราะห์ข้อจำกัดในการออกแบบเกมออนไลน์ในปัจจุบัน	17
2.6 การเข้ารหัสและถอดรหัสลับแบบกุญแจส่วนตัว	18
2.7 การเข้ารหัสและถอดรหัสลับแบบกุญแจสาธารณะ	18
บทที่ 3 เกมออนไลน์	20
3.1 ภาษาที่ใช้พัฒนาเกมออนไลน์	20
3.2 เครื่องมือสำหรับพัฒนาเกมออนไลน์	20
3.2.1 ซอร์ฟแวร์ชุดคำสั่งโคเรคเอ็กซ์	21
3.2.1.1 โคเรคเอ็กซ์กราฟฟิก	21
3.2.1.2 โคเรคอินพุต	22
3.2.1.3 โคเรคซาวน์	22
3.2.1.4 โคเรคมิวสิก	22
3.2.1.5 โคเรคเพล	22
3.2.2 ซอร์ฟแวร์ชุดคำสั่งสำหรับช่วยพัฒนาเกม	23
3.2.2.1 ซอร์ฟแวร์ซีดีเอ็กซ์	23
3.2.2.2 ซอร์ฟแวร์ฮ็อก	24
3.3 สถาปัตยกรรมเน็ตเวิร์คแบบผสม	24
3.3.1 การทำงานส่วนที่เป็นไคลเอนท์และเซิร์ฟเวอร์	24
3.3.2 การทำงานส่วนที่เป็นเพียร์-ทู-เพียร์	24
บทที่ 4 การวิเคราะห์และออกแบบระบบ	26
4.1 การออกแบบเกม	26
4.1.1 ลักษณะของเกม	26
4.1.2 การบังคับตัวละคร	26
4.1.3 การทำตัวเองเป็นโฮสต์	27

สารบัญ (ต่อ)

หน้า

4.1.4 การร่วมกลุ่มกับผู้เล่นที่เป็นโฮสต์อื่น	27
4.1.5 การลงทะเบียนสินค้ากับระบบเซิร์ฟเวอร์	27
4.1.6 การซื้อ-ขายสินค้าภายในเกม	28
4.1.7 การตรวจสอบความถูกต้องของข้อมูลสินค้าของโฮสต์.....	29
4.1.8 การพูดคุย.....	29
4.2 การออกแบบการทำงานของโปรแกรม.....	29
4.2.1 การเข้าใช้ระบบ	29
4.2.2 การสร้างโฮสต์.....	30
4.2.3 การเข้าร่วมโฮสต์ของผู้เล่นคนอื่น	31
4.2.4 การลงทะเบียนสินค้าของผู้เล่น.....	31
4.2.5 การซื้อสินค้าระหว่างผู้เล่นกับเซิร์ฟเวอร์	32
4.2.6 การซื้อสินค้าระหว่างโฮสต์และเพียร์.....	33
4.2.7 การตรวจสอบความถูกต้องของข้อมูลโฮสต์.....	33
4.2.8 การพูดคุยระหว่างผู้เล่น	34
4.2.9 การอัปเดตข้อมูลตัวละครของผู้เล่น.....	35
4.3. การออกแบบโปรโตคอลที่ใช้พัฒนาโปรแกรมในสถาปัตยกรรมเน็ตเวิร์คแบบผสม ..	36
4.3.1 โปรโตคอลที่ใช้สำหรับสถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนท์และเซิร์ฟเวอร์ .	36
4.3.2 โปรโตคอลที่ใช้สำหรับสถาปัตยกรรมเน็ตเวิร์คแบบเพียร์-ทู-เพียร์.....	38
4.3.3 โปรโตคอลที่ใช้สำหรับสถาปัตยกรรมเน็ตเวิร์คแบบผสม.....	39
4.4 การออกแบบฐานข้อมูล	41
4.5 ฟังก์ชันและรายละเอียดที่ใช้พัฒนา	42
4.5.1 ฟังก์ชันที่ใช้ทำการเชื่อมต่อผ่านเน็ตเวิร์ค.....	42
4.5.1.1 ฟังก์ชันทางฝั่งเซิร์ฟเวอร์	43
4.5.1.2 ฟังก์ชันทางฝั่งไคลเอนท์และฟังก์ชันที่ใช้ติดต่อระหว่างเพียร์	43
4.5.2 ฟังก์ชันต้นแบบที่ใช้ติดต่อกับโปรโตคอล.....	45
4.5.3 ฟังก์ชันที่ใช้ทำการเข้ารหัสข้อมูล	45
4.5.4 ฟังก์ชันที่ใช้ในการแสดงผล	46

สารบัญ (ต่อ)

	หน้า
บทที่ 5 การพัฒนาระบบ	48
5.1 เครื่องมือในการพัฒนาระบบ	48
5.2 หน้าจอการทำงานของโปรแกรม	48
5.2.1 หน้าจอการเข้าระบบ.....	48
5.2.2 หน้าจอลงทะเบียนสินค้าที่ผู้เล่นสร้างขึ้นเอง	50
5.2.3 หน้าจอซื้อสินค้าจากเซิร์ฟเวอร์.....	52
5.2.4 หน้าจอการสร้างโฮสต์.....	54
5.2.5 หน้าจอการเข้าร่วมโฮสต์.....	55
5.2.6 หน้าจอการซื้อสินค้าระหว่างเพียร์และ โฮสต์.....	57
5.2.7 หน้าจอแสดงการตรวจสอบความถูกต้องของข้อมูลโฮสต์	58
5.2.8 หน้าจอแสดงการพูดคุย	59
บทที่ 6 สรุปผลและข้อเสนอแนะ.....	60
6.1 สรุปผล.....	60
6.2 ข้อเสนอแนะ	60
บรรณานุกรม.....	62
ภาคผนวก.....	63
ภาคผนวก ก.	64
ประวัติผู้เขียน	67

สารบัญตาราง

ตารางที่	หน้า
4.1 ตาราง Login	41
4.2 ตาราง Item	41
4.3 ตาราง UserItem.....	42



สารบัญรูป

รูปที่	หน้า
2.1	สถาปัตยกรรมเน็ตเวิร์คแบบเพียร์-ทู-เพียร์..... 7
2.2	สถาปัตยกรรมเน็ตเวิร์คแบบเพียร์-ทู-เพียร์ ชนิดโมเด็มต่อโมเด็ม..... 8
2.3	สถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนท์และเซิร์ฟเวอร์..... 9
2.4	ลักษณะรูปแบบการร้องขอและการตอบกลับ..... 10
2.5	ตัวอย่างภาพเกมแร็คนารีคออนไลน์..... 14
2.6	ตัวอย่างภาพเกมยุทธภพครบสติง..... 15
2.7	ตัวอย่างภาพเกมปิงป่า..... 16
2.8	ตัวอย่างภาพเกมวอร์ครีฟ..... 17
3.1	สถาปัตยกรรมของไครคเพล..... 23
3.2	การทำงานของเพียร์-ทู-เพียร์ในสถาปัตยกรรมเน็ตเวิร์คแบบผสมรูปแบบ 1 ต่อ 1..... 25
3.3	การทำงานของเพียร์-ทู-เพียร์ในสถาปัตยกรรมเน็ตเวิร์คแบบผสมรูปแบบสร้างโฮสต์..... 25
4.1	แสดงการดำเนินการในขั้นตอนเข้าใช้ระบบ..... 30
4.2	แสดงการสร้างโฮสต์ของผู้เล่น..... 30
4.3	แสดงการเข้าร่วมโฮสต์ของผู้เล่น..... 31
4.4	แสดงการลงทะเบียนสินค้าที่ผู้เล่นสร้างขึ้น..... 31
4.5	แสดงการซื้อขายระหว่างผู้เล่นกับเซิร์ฟเวอร์..... 32
4.6	แสดงการซื้อขายระหว่างโฮสต์และเพียร์..... 33
4.7	แสดงการตรวจสอบความถูกต้องของข้อมูลโฮสต์..... 34
4.8	แสดงการทำงานการพูดคุยระหว่างผู้เล่น..... 35
4.9	แสดงการอัปเดตข้อมูลตัวละครที่ผู้เล่นบังคับ..... 36
5.1	หน้าจอการเข้าใช้ระบบ..... 48
5.2	หน้าจอหลังจากทำการเข้าสู่ระบบ..... 49
5.3	หน้าจอการใส่ข้อมูลของสินค้าเพื่อทำการลงทะเบียน..... 50
5.4	หน้าจอยืนยันการลงทะเบียนสินค้า..... 51
5.5	หน้าจอแสดงรายการสินค้าที่ซื้อกับเซิร์ฟเวอร์..... 52
5.6	หน้าจอแสดงเมื่อผู้เล่นทำการซื้อสินค้าเสร็จสิ้น..... 53
5.7	หน้าจอแสดงความเป็นโฮสต์..... 54

สารบัญรูป (ต่อ)

รูปที่	หน้า
5.8	หน้าจอแสดงการเลือกโฮสต์ที่ต้องการเข้าร่วม..... 55
5.9	หน้าจอแสดงการเข้าร่วมโฮสต์กับผู้เล่นคนอื่น..... 56
5.10	หน้าจอแสดงรายการสินค้าที่โฮสต์ขาย 57
5.11	หน้าจอแสดงการตรวจสอบข้อมูลโฮสต์..... 58
5.12	หน้าจอแสดงการพูดคุยของผู้เล่น 59



บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เกมคอมพิวเตอร์มีการพัฒนามาเป็นเวลานาน โดยในระยะแรกเกมถูกสร้างมาเพื่อใช้ฝึกทักษะของมนุษย์บางอย่าง เช่น การคำนวณทางคณิตศาสตร์ ซึ่งสามารถทำให้สิ่งที่ดูหน้าเบื่อกลับกลายเป็นความสนุกสนานได้ เมื่อเกมคอมพิวเตอร์เริ่มได้รับความนิยมมากขึ้น จึงได้มีการพัฒนาโดยมีการเพิ่มลูกเล่นต่างๆ เข้าไปมากมาย เช่น เพิ่มการแสดงผลรูปภาพที่มีสีสันสวยงาม มีเสียงเพลงบรรเลงประกอบ และอุปกรณ์คอมพิวเตอร์ได้มีการพัฒนาจนมีความสามารถมากกว่าเดิมมาก ปัจจุบันผู้พัฒนาเกมคอมพิวเตอร์ส่วนใหญ่มักพัฒนาเกมของตน โดยมีการแสดงผลในรูปแบบ 3 มิติ ซึ่งมีความสมจริงมากกว่าเกมในรูปแบบ 2 มิติ เกมมีการพัฒนารูปแบบการเล่นต่างๆ มากมาย เช่น เกมเสริมสร้างทักษะของเด็กก่อนวัยเรียน เกมเสริมสร้างความรู้ในแขนงวิชาต่างๆ จนถึงเกมที่ใช้เล่นเพื่อความสนุกสนาน ช่วยผ่อนคลายความเครียดจากการทำงานและเรียนหนังสือ ส่วนใหญ่เกมคอมพิวเตอร์จะถูกพัฒนาให้เล่นได้เพียง 1 คน โดยมีเนื้อเรื่องให้ผู้เล่นรับบทเป็นตัวละครในเกมและเล่นไปจนจบเกม แต่ด้วยลักษณะโดยทั่วไปของมนุษย์จะมีความต้องการทางด้านสังคม จึงได้มีการพัฒนาเกมออกมาในรูปแบบที่สามารถเล่นได้พร้อมๆ กันหลายคน และเรียกว่า “เน็ตเวิร์คเกม (Network Game)” เริ่มแรกเกมประเภทนี้ถูกพัฒนาให้สามารถเล่นผ่านระบบแลน(LAN) เท่านั้น และสามารถรองรับผู้เล่นได้ 2 – 8 คน ต่อมาเมื่ออุปกรณ์ด้านเน็ตเวิร์คมีการพัฒนาความสามารถสูงขึ้น ทำให้สามารถเล่นได้พร้อมกันได้เป็น 16 - 32 คนหรือมากกว่านี้ โดยเกมใช้สถาปัตยกรรมเน็ตเวิร์คแบบเพียร์-ทู-เพียร์(Peer-to-Peer architecture) และเกมจะถูกออกแบบมาให้เล่นแบบรอบเดียวจบ

อย่างไรก็ตามจำนวนผู้เล่นที่สามารถเล่นได้ยังคงถูกจำกัดตามความสามารถและทรัพยากรของเครื่องคอมพิวเตอร์ที่ใช้เล่นเกม และเมื่ออินเทอร์เน็ต(Internet) เริ่มได้รับความนิยมและแพร่หลายขึ้นในกลุ่มผู้ใช้คอมพิวเตอร์ตามบ้าน จึงได้มีการพัฒนาให้เกมสามารถเล่นผ่านอินเทอร์เน็ตได้ เรียกว่า “เกมออนไลน์(Game Online)” ในระยะแรกเกมที่เล่นผ่านอินเทอร์เน็ตเป็นเกมที่ถูกดัดแปลงและพัฒนาจากเน็ตเวิร์คเกมที่เล่นผ่านระบบแลน ด้วยการเพิ่มโปรโตคอลให้เกมสนับสนุนเข้าไป ซึ่งโปรโตคอลที่ใช้เล่นผ่านระบบอินเทอร์เน็ต คือ TCP/IP นั่นเอง ต่อมาจึงมีผู้คิดค้นเกมรูปแบบใหม่ที่ต้องการความสามารถในการรองรับผู้เล่นได้เป็นจำนวนมากๆ ได้ถึงระดับหลายพันคน โดยเกมพัฒนาบนสถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนท์และเซิร์ฟเวอร์(Client and server architecture) และให้ชื่อเกมออนไลน์ประเภทนี้ว่า “เกมออนไลน์แบบมีผู้เล่นจำนวนมาก(Massive Multiplayer Online Game : MMOG)” ซึ่งเป็นที่นิยมอย่างมากในปัจจุบัน ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในขั้นตอนการพัฒนาเกมออนไลน์แบบมีผู้เล่นจำนวนมาก(ซึ่งต่อไปในโครงการพิเศษฉบับนี้ จะขอเรียกว่า “เกมออนไลน์” เนื่องจากเกมที่เล่นผ่านระบบเน็ตเวิร์คในปัจจุบันเกือบทั้งหมดจะเป็น เกมในรูปแบบนี้) ปัญหาใหญ่ที่มักพบ คือ ความสามารถของตัวฮาร์ดแวร์(Hardware) ที่ใช้เป็น ตัวเครื่องเซิร์ฟเวอร์ จะต้องมีประสิทธิภาพสูง เนื่องจากตัวเซิร์ฟเวอร์มีหน้าที่ดูแลและจัดการการ เชื่อมต่อกับไคลเอนต์ทั้งหมด และมีหน้าที่ประมวลผลคำสั่งที่ไคลเอนต์ส่งมาให้พร้อมทั้งเป็นตัว เก็บข้อมูลต่างๆ ของผู้เล่นไว้ทั้งหมดอีกด้วย จะเห็นได้ว่าผู้ที่ต้องการพัฒนาเกมออนไลน์นั้นนอกจาก จะต้องออกแบบเกมให้เป็นที่ยอมรับของผู้เล่นแล้ว ยังต้องพิจารณาถึงต้นทุนที่ต้องเสียไปกับตัว ฮาร์ดแวร์ที่มีราคาสูงมากอีกด้วย จึงทำให้ผู้พัฒนาเกมในประเทศไทยหลายรายที่ไม่กล้าเสี่ยงลงทุน กับการพัฒนาเกมออนไลน์ในระบบนี้ เนื่องจากเป็นระบบเกมออนไลน์ที่ใหญ่มากนั่นเอง

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

โครงการพิเศษฉบับนี้มุ่งหวังเพื่อทำการศึกษางานของโปรแกรมและระบบเกมออนไลน์ ที่ใช้สถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนต์และเซิร์ฟเวอร์ และศึกษากันออกแบบระบบการทำงาน ใหม่โดยนำเอาสถาปัตยกรรมแบบเพียร์-ทู-เพียร์ มาผสมผสานประยุกต์เข้ากับสถาปัตยกรรมแบบ ไคลเอนต์และเซิร์ฟเวอร์ ซึ่งเป็นสถาปัตยกรรมเน็ตเวิร์คที่นิยมใช้พัฒนาเกมออนไลน์ในปัจจุบัน โดยมีจุดประสงค์หลัก คือ ใช้สถาปัตยกรรมแบบเพียร์-ทู-เพียร์ เข้ามามีส่วนช่วยประมวลผลคำสั่ง บางอย่างในตัวไคลเอนต์เอง เพื่อลดภาระหน้าที่การประมวลผลในฝั่งเซิร์ฟเวอร์ ซึ่งมีผลช่วยลดภาระ การทำงานของเซิร์ฟเวอร์ลงได้ เมื่อเปรียบเทียบกับการทำงานของระบบโปรแกรมและเกม ออนไลน์ที่พัฒนาโดยใช้สถาปัตยกรรมแบบไคลเอนต์และเซิร์ฟเวอร์เพียงอย่างเดียว

1.3 สมมุติฐานของการศึกษา

จากการนำสถาปัตยกรรมแบบเพียร์-ทู-เพียร์เข้ามาช่วยประมวลผลคำสั่งบางส่วน ทำให้ระบบ เซิร์ฟเวอร์ทำงานลดลง และทำให้ระบบเกมทางฝั่งไคลเอนต์ทำงานได้เร็วขึ้น เนื่องจากไม่ต้องรอ การประมวลผลและตอบกลับจากเซิร์ฟเวอร์ และมีผลทำให้เซิร์ฟเวอร์มีการตอบสนองได้รวดเร็ว ยิ่งขึ้น สามารถรองรับจำนวนผู้เล่นได้จำนวนมากขึ้น แต่ทั้งนี้ยังมีปัจจัยอื่นๆ ที่ต้องพิจารณา ประกอบด้วย เช่น การประมวลผลที่แบ่งให้ไคลเอนต์ทำงานด้วยตัวเองนั้น ถ้าโดยปกติผู้เล่นไม่ได้ ใช้งานเลย หรือมีการใช้งานน้อยนั้นแทบจะไม่ช่วยลดงานให้กับเซิร์ฟเวอร์เลย ดังนั้นการแบ่งส่วน การประมวลผลที่มีการเรียกใช้งานบ่อยๆ จะสามารถช่วยลดภาระให้กับเซิร์ฟเวอร์ได้มาก เช่น การ เคลื่อนที่ของตัวละครไปยังตำแหน่งต่างๆ ภายในเกม การพูดคุยกันระหว่างตัวละคร เป็นต้น การ ทำงานส่วนนี้มีการเรียกใช้งานบ่อยในเกมออนไลน์ นอกจากนี้การปรับให้ใช้เพียร์-ทู-เพียร์มาช่วย ได้นั้น จำเป็นตัวคูเรื่องความถูกต้องภายในเกมอีกด้วย เช่น การเดินของตัวละครไม่ควรที่จะเดิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทะลุกำแพงได้ ดังนั้นจึงควรมีวิธีการป้องกันให้อยู่ภายใต้เงื่อนไขกฎเกณฑ์ของเกมด้วย โดยการประมวลผลที่ไคลเอนต์นั้นสามารถทำการดัดแปลงได้ง่ายกว่าการส่งคำสั่งไปประมวลผลที่เซิร์ฟเวอร์ และการประมวลผลบางอย่างที่มีผลกระทบต่อผลประโยชน์ของผู้เล่นต่างๆ เช่น การซื้อ ขาย แลกเปลี่ยนสิ่งของต่างๆ ภายในเกม จำเป็นต้องมีระบบรักษาความปลอดภัยของข้อมูลด้วย โดยหน้าที่นี้การประมวลผลในส่วนนี้ควรให้เป็นความรับผิดชอบของเซิร์ฟเวอร์มากกว่า แต่สามารถนำเอาเพียร์-ทู-เพียร์เข้ามาใช้ได้ โดยต้องมีการนำระบบรักษาความปลอดภัยอื่นๆ เข้ามาใช้ เช่น การเข้ารหัสแพ็คเกจข้อมูล(Packet) ที่ใช้ส่งระหว่างเพียร์ด้วยกันเอง เพื่อให้ยากต่อการปลอมแปลง

1.4 ทฤษฎีหรือแนวความคิดที่ใช้ในโครงการพิเศษ

การลดภาระให้กับเซิร์ฟเวอร์โดยใช้สถาปัตยกรรมแบบเพียร์-ทู-เพียร์มาช่วยนั้น ทำโดยใช้ความสามารถด้านการสร้างโฮสต์(Host) ของสถาปัตยกรรมแบบเพียร์-ทู-เพียร์ ซึ่งโฮสต์จะทำหน้าที่เป็นตัวกลางจัดการเพียร์อื่นในกลุ่มผู้เล่นที่อยู่ในสภาพแวดล้อมที่เหมือนกัน ในส่วนของการสร้างโฮสต์ของกลุ่มนั้นจะต้องมีผู้เล่นเป็นคนสร้างขึ้น ซึ่งในโครงการพิเศษนี้ผู้ที่เป็นคนสร้างโฮสต์หมายถึง เจ้าของบ้านหรือเจ้าของร้าน และผู้เล่นคนอื่นที่ต้องการเข้าไปเยี่ยมชมผู้เล่นคนอื่นๆ ก็จะทำการขอร่วมกลุ่มหรือขอเข้าไปภายในบ้านของผู้เล่นที่เป็นโฮสต์

ผู้เล่นที่อยู่ในกลุ่มเดียวกันสามารถทำกิจกรรมบางอย่างร่วมกันได้ เช่น การพูดคุย สามารถมองเห็นผู้เล่นคนอื่นๆ ในกลุ่มเดียวกันได้ ในขณะที่ผู้เล่นที่อยู่ต่างกลุ่มกัน จะไม่สามารถทำกิจกรรมใดๆ ต่อกันได้ ซึ่งการประมวลผลเกือบทุกอย่างที่อยู่ภายในโฮสต์จะอยู่ที่ไคลเอนต์

1.5 การเปรียบเทียบระหว่างวิธีที่นำเสนอกับวิธีการแบบพื้นฐาน

วิธีการสร้างโฮสต์แบ่งผู้เล่นออกเป็นกลุ่มแล้วประมวลผลเกมกันภายในกลุ่มนั้น มีข้อดีตรงที่เซิร์ฟเวอร์ไม่ต้องทำหน้าที่ดูแลกลุ่มที่ผู้เล่นสร้างขึ้น เซิร์ฟเวอร์มีหน้าที่ดูแลข้อมูลของผู้เล่น และไม่จำเป็นต้องเข้าไปมีส่วนเกี่ยวข้องกับผู้เล่นในกลุ่มแต่ละกลุ่ม ในกรณีที่ผู้เล่นกลุ่มนั้นไม่ได้ต้องการทำสิ่งที่จำเป็นต้องใช้ระบบรักษาความปลอดภัยของข้อมูล แต่เมื่อใดที่ต้องการเกี่ยวกับระบบรักษาความปลอดภัยของข้อมูลจะต้องทำผ่านเซิร์ฟเวอร์ เช่น การซื้อขายสินค้า ผู้เล่นต้องการความปลอดภัยและความน่าเชื่อถือ ซึ่งต้องทำการร้องขอให้เซิร์ฟเวอร์ทำหน้าที่เป็นตัวกลางในการซื้อขายให้

ส่วนเกมออนไลน์โดยทั่วไปนั้น เซิร์ฟเวอร์จะเป็นตัวกลางให้กับผู้เล่นทุกคน เซิร์ฟเวอร์จะเป็นโฮสต์เพียงหนึ่งเดียวเท่านั้น แต่สามารถแยกการทำงานออกเป็นหลายๆ ส่วนซึ่งเปรียบเสมือนทำการสร้างโฮสต์ไว้หลายโฮสต์ เรียกว่า “การแบ่งเกมเซสชัน(Game Session)” โดยทั่วไปจะแบ่งตามจำนวนฉากที่มีในเกม หรือแบ่งตามขอบเขตพื้นที่ในเกมที่เป็นเกมในรูปแบบ 3 มิติ การ

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประมวลผลทุกอย่างจากไคลเอนท์จะถูกส่งมาให้เซิร์ฟเวอร์ทำการประมวลผลแล้วส่งผลกลับ การทำงานในลักษณะนี้จะเห็นได้ว่าเซิร์ฟเวอร์ต้องทำงานหนัก เครื่องที่ใช้เป็นตัวเซิร์ฟเวอร์จึงต้องเป็นเครื่องที่มีประสิทธิภาพการประมวลผลที่สูง เพื่อให้สามารถรองรับผู้เล่นที่มีจำนวนมากได้ นอกจากนี้อาจต้องทำการใช้เครื่องเซิร์ฟเวอร์จำนวนหลายเครื่องเพื่อให้การทำงานมีประสิทธิภาพ ซึ่งต่างจากวิธีที่ได้นำเสนอใน โครงการงานพิเศษนี้ที่สามารถนำเครื่องที่มีประสิทธิภาพต่ำกว่ามาทำหน้าที่เป็นเซิร์ฟเวอร์ได้

1.6 ขอบเขตของโครงการงานพิเศษ

โครงการงานพิเศษนี้ได้ทำการพัฒนาเกมออนไลน์ที่ใช้ทั้งสถาปัตยกรรมแบบเพียร์-ทู-เพียร์ และสถาปัตยกรรมแบบไคลเอนท์และเซิร์ฟเวอร์ โดยเซิร์ฟเวอร์จะมีหน้าที่หลักๆ คือ เก็บข้อมูลของผู้เล่น ทำหน้าที่รักษาความปลอดภัยของข้อมูลผู้เล่นรวมทั้งการร้องขอเข้าระบบ(Login) และการซื้อขายสิ่งของระหว่างผู้เล่น

ในส่วนของเกมที่จะนำเสนอในโครงการงานพิเศษนี้ มีความสามารถดังนี้

1. ผู้เล่นทุกคนจะมีบ้านเป็นของตัวเอง ซึ่งผู้เล่นที่เป็นเจ้าของบ้านเท่านั้นที่จะทำหน้าที่เป็นโฮสต์ในบ้านของตนเองได้ เมื่อใดก็ตามที่เจ้าของบ้านไม่อยู่หรือกำลังอยู่บ้านของผู้เล่นคนอื่น ผู้เล่นคนอื่นจะไม่สามารถเข้าไปยังบ้านของผู้เล่นคนนั้นได้
2. ผู้เล่นสามารถสร้างสินค้าขึ้นมาเองได้โดยเป็น โมเดล 3 มิติ ซึ่งได้ทำการแปลงเป็นไฟล์ .X ซึ่งเป็นไฟล์ที่โคเร็กเอ็กซ์รองรับ และสามารถขายสินค้าของตนให้กับผู้เล่นคนอื่นได้ แต่ต้องนำสินค้านั้นลงทะเบียนกับเซิร์ฟเวอร์ก่อน โดยสินค้าไม่มีการระบุรูปร่างลักษณะ ขนาด และหมวดหมู่ โดยให้ถือว่าสินค้าทุกชนิดเป็นเพียงของตกแต่งบ้านเท่านั้น ผู้เล่นไม่สามารถตกแต่งกับสินค้าที่ได้ลงทะเบียนไว้แล้วได้ ผู้ที่เป็นคนสร้างสินค้าจะสามารถขายสินค้าของตนได้ไม่จำกัดจำนวน สำหรับสินค้าที่ไม่ได้สร้างด้วยตัวผู้เล่นเองจะไม่สามารถทำการขายให้กับผู้เล่นคนอื่นได้
3. ผู้เล่นสามารถทำการซื้ออุปกรณ์ตกแต่งบ้านของตนเองได้ สามารถซื้อสินค้าจากเซิร์ฟเวอร์หรือจากผู้เล่นที่เป็นโฮสต์ในขณะนั้นได้เท่านั้น ไม่สามารถทำการซื้อขายสินค้าจากผู้เล่นคนอื่นๆ ที่ไม่ใช่โฮสต์ได้ สำหรับการซื้อสินค้าจากเซิร์ฟเวอร์ผู้เล่นจะต้องอยู่ที่บ้านของตัวเองเท่านั้น
4. มีระบบเข้ารหัสข้อมูล(Cryptography) และผู้เล่นสามารถตรวจสอบความถูกต้องของข้อมูลสินค้าภายในโฮสต์ได้ โดยมีเซิร์ฟเวอร์เป็นผู้ช่วยเหลือในการตรวจสอบความถูกต้อง

1.7 ขั้นตอนของการศึกษา

1. ศึกษากระบวนการออนไลน์ที่มีการพัฒนาในปัจจุบัน

2. กำหนดแนวทาง วัตถุประสงค์ ขอบเขตของโครงการและวิธีการที่จะศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ออกแบบและประยุกต์การใช้สถาปัตยกรรมเน็ตเวิร์กแบบเพียร์-ทู-เพียร์ เพื่อมาใช้ในการพัฒนาเกมออนไลน์ที่ใช้สถาปัตยกรรมเน็ตเวิร์กแบบไคลเอนท์และเซิร์ฟเวอร์
4. ออกแบบเกม วิธีการสื่อสารภายในเกม โดยใช้สถาปัตยกรรมเน็ตเวิร์กแบบผสม
5. พัฒนาเกมไคลเอนท์ และเซิร์ฟเวอร์ตามที่ได้ออกแบบไว้
6. ทดสอบการทำงาน
7. สรุปผลและข้อเสนอแนะ

1.8 ประโยชน์ที่คาดว่าจะได้รับ

1. เป็นแนวทางในการพัฒนาระบบเซิร์ฟเวอร์อีกรูปแบบหนึ่ง ซึ่งจะช่วยลดภาระการทำงานของเซิร์ฟเวอร์ที่ใช้สถาปัตยกรรมแบบไคลเอนท์และเซิร์ฟเวอร์
2. เป็นแนวทางในการพัฒนาโปรแกรมและเกมออนไลน์รูปแบบใหม่ที่ใช้สถาปัตยกรรมเครือข่ายแบบผสม



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

หลักการและทฤษฎีที่เกี่ยวข้อง

ในหัวข้อนี้จะกล่าวถึงลักษณะของเกมออนไลน์ว่ามีลักษณะอย่างไร สามารถพัฒนาเกมโดยใช้สถาปัตยกรรมเน็ตเวิร์คอะไรได้บ้าง มีความแตกต่างกันอย่างไรบ้างระหว่างเกมที่พัฒนาโดยใช้สถาปัตยกรรมที่ต่างกัน วิเคราะห์เกมออนไลน์ในปัจจุบันว่าทำไมถึงเป็นที่นิยมและเกมยังมีข้อจำกัดอะไรบ้าง และการเข้ารหัสข้อมูลและถอดรหัสข้อมูลแบบใช้กุญแจสาธารณะและลายเซ็นดิจิทัล

2.1 ลักษณะของเกมออนไลน์

เกมออนไลน์เป็นเกมที่มีการเชื่อมต่อกับผู้เล่นคนอื่น ซึ่งอาจจะติดต่อกันผ่านแลน ผ่านโมเด็ม หรือผ่านระบบอินเทอร์เน็ต โดยลักษณะเด่นของเกมออนไลน์คือความสามารถในการโต้ตอบกับผู้เล่นคนอื่น ๆ ได้ เช่น การพูดคุย การแลกเปลี่ยนสิ่งของซึ่งกันและกัน การช่วยเหลือกันในรูปแบบต่างๆ และด้วยความสามารถนี้ทำให้เกมออนไลน์ได้รับความนิยมสูงมาก เนื่องจากมนุษย์มักชอบการเข้าร่วมสังคม และมักชอบการแข่งขัน

เกมออนไลน์สามารถแบ่งออกเป็น 2 ประเภท คือ แบบเล่นเป็นฉาก เมื่อเล่นจบก็จบเกม และแบบที่สามารถเล่นต่อเนื่อง โดยไม่มีการจบเกมจึงสามารถเล่นได้เรื่อยๆ โดยเกมที่เล่นเป็นฉากจะเป็นเกมที่เล่นผ่านระบบแลน ใช้สถาปัตยกรรมเน็ตเวิร์คแบบพีียร์-ทู-พีียร์ และมีการกำหนดจำนวนผู้เล่นที่สามารถเข้าไปเล่นได้ ซึ่งจำนวนผู้เล่นที่สามารถเข้าไปเล่นได้สูงสุด ผู้พัฒนาจะเป็นผู้กำหนดไว้โดยดูจากขนาดของฉากที่เล่นเป็นหลัก เกมในกลุ่มนี้มักเป็นเกมที่เล่นผ่านระบบแลน ตัวอย่างเกมที่มีรูปแบบการเล่นเช่นนี้ ได้แก่ เกมเคา์เตอร์สไตรค์(Counter Strike) ซึ่งเป็นเกมประเภทยิงต่อสู้กัน โดยนำเสนอการเล่นโดยใช้มุมมองสายตาคอนเล่น(First person shooting) ตัวเกมสามารถรองรับผู้เล่นได้สูงสุด 32 คน เกมวอร์คราฟ เป็นเกมแนววางแผนการรบ สามารถรองรับผู้เล่นได้สูงสุด 8 - 10 คน และเกมประเภทนี้ส่วนใหญ่จะมีส่วนที่สามารถเล่นคนเดียวได้โดยไม่จำเป็นต้องทำการเชื่อมต่อเพื่อเล่นกับผู้เล่นคนอื่น โดยเล่นตามเนื้อเรื่องของเกม

ส่วนเกมออนไลน์ที่เล่นได้ต่อเนื่อง มักเป็นเกมที่ผู้เล่นจะต้องทำการสร้างตัวละคร และเล่นผ่านระบบอินเทอร์เน็ต ใช้สถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนท์และเซิร์ฟเวอร์ เป็นเกมที่ผู้เล่นจะต้องพัฒนาตัวละครของตนไปเรื่อยๆ โดยการต่อสู้กับศัตรูหรือผู้เล่นคนอื่น ๆ ภายในเกม เกมจะมีการสอดแทรกความสามารถทางสังคมเข้าไปด้วย เช่น การพูดคุยระหว่างกัน การแลกเปลี่ยนซื้อขายสิ่ง

ค่ากัน และการตั้งกลุ่มหรือตั้งสมาคมเพื่อทำกิจกรรมร่วมกันเป็นต้น โดยจำนวนของผู้เล่นที่สามารถเข้ามามีส่วนในการเล่น ไม่มีการกำหนดตายตัว ขึ้นอยู่กับความสามารถของเครื่องเซิร์ฟเวอร์ที่ให้บริการเกมนั้นๆ

ส่วนใหญ่มักจะสามารถรองรับจำนวนผู้เล่นได้เป็นจำนวนมากกว่าหลายพันคน เกมออนไลน์ประเภทนี้ถ้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งยังมีให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

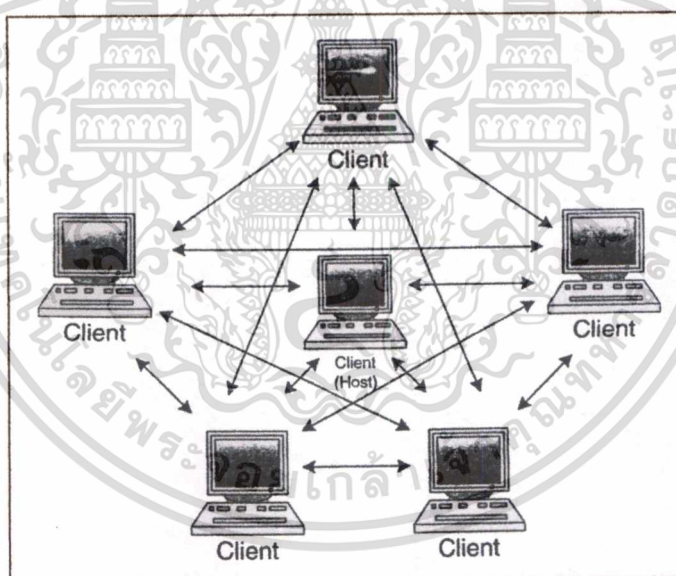
จำเป็นต้องมีการเชื่อมต่อไปยังเครื่องที่ให้บริการเกมก่อนที่จะเริ่มเล่นเกมทุกครั้ง ไม่สามารถเล่นคนเดียวได้ ตัวอย่างเกมที่มีลักษณะนี้ได้แก่ เกมเร็นน่าร็อคออนไลน์

2.2 สถาปัตยกรรมเน็ตเวิร์กที่ใช้พัฒนาเกม

สถาปัตยกรรมเน็ตเวิร์กมี 2 ประเภท คือ สถาปัตยกรรมเน็ตเวิร์กแบบเพียร์-ทู-เพียร์ และ สถาปัตยกรรมเน็ตเวิร์กแบบไคลเอนท์และเซิร์ฟเวอร์ ทั้ง 2 สถาปัตยกรรมนี้สามารถใช้พัฒนาเกมออนไลน์ได้ทั้งสิ้น แต่แตกต่างกันที่รูปแบบการเล่นของเกม

2.2.1 สถาปัตยกรรมเน็ตเวิร์กแบบเพียร์-ทู-เพียร์

สถาปัตยกรรมนี้ไม่จำเป็นต้องมีตัวทำหน้าที่เป็นเซิร์ฟเวอร์ ไคลเอนท์ทุกตัวจะมีการเชื่อมต่อเข้าหากันและทำการส่งข้อมูลที่อัปเดตไปยังไคลเอนท์ตัวอื่นๆ ที่อยู่ในกลุ่มเดียวกัน จึงมีข้อจำกัดในเรื่องจำนวนการเชื่อมต่อของผู้เล่น โดยความสามารถในการเชื่อมต่อจะได้มากหรือน้อยขึ้นอยู่กับทรัพยากรของเครื่องคอมพิวเตอร์ที่เชื่อมต่อเข้าด้วยกัน ดังแสดงในรูปที่ 2.1



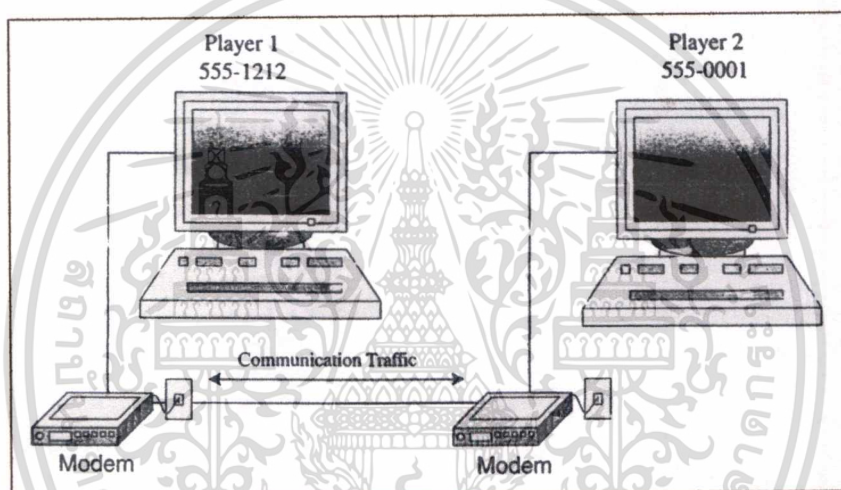
รูปที่ 2.1 สถาปัตยกรรมเน็ตเวิร์กแบบเพียร์-ทู-เพียร์

เกมที่พัฒนาโดยใช้สถาปัตยกรรมแบบเพียร์-ทู-เพียร์ เป็นเกมที่เล่นผ่านระบบแลน(LAN) เป็นส่วนใหญ่ ซึ่งผู้พัฒนามักกำหนดจำนวนผู้เล่นไว้ว่าสามารถเข้าไปเล่นพร้อมๆ กันได้ไม่เกินที่กำหนด แล้วแต่ลักษณะของตัวเกมและผู้พัฒนาจะกำหนด โดยทั่วไปจะกำหนดตามขนาดของฉากหรือแผนที่ที่ใช้เป็นฉากสนามของเกม ก่อนที่ผู้เล่นจะเริ่มเล่นเกมได้นั้นจะต้องมีผู้เล่นคนใดคนหนึ่งสร้างโฮสต์ขึ้นมาก่อนแล้วให้ผู้เล่นคนอื่นเข้ามาาร่วมเล่นด้วย โดยทั่วไปผู้พัฒนาเกมจะแสดงผลตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โฮสต์ในเกมให้เห็นเป็นห้อง ซึ่งมีชื่อห้องกำกับแยกความแตกต่างระหว่างห้องให้ผู้เล่นสามารถเลือกเข้าไปเล่นในห้องที่ต้องการได้

นอกจากจะเล่นผ่านระบบแลนแล้ว เกมที่ใช้สถาปัตยกรรมแบบเพียร์-ทู-เพียร์ ยังสามารถเล่นผ่านโมเด็มต่อโมเด็มได้ด้วย ซึ่งจะเล่นได้ 2 คนเท่านั้น โดยการเชื่อมต่อจะเป็นลักษณะการโทรหา กัน โดยจะต้องมีผู้เล่นคนหนึ่งเป็นผู้โทรไปหาผู้เล่นอีกคนหนึ่ง และผู้เล่นที่รอรับการโทรเข้ามา จากผู้เล่นอีกคนก็จะต้องกำหนดให้เกมจัดการเปิดโมเด็มรับสัญญาณเอาไว้ก่อน เมื่อผู้เล่นอีกฝ่าย โทรเข้ามาตัวเกมจะทำการเชื่อมต่อผู้เล่นทั้ง 2 เข้าด้วยกันเปรียบได้กับทั้ง 2 เครื่องได้ทำการต่อผ่าน พอร์ตอนุกรมเข้าด้วยกัน และสามารถที่จะเล่นได้ยาวนานเท่าไรก็ได้จนกว่าจะมีผู้เล่นคนใดคนหนึ่ง ออกจากเกมหรือตัดการเชื่อมต่อ ซึ่งแสดงในรูปที่ 2.2



รูปที่ 2.2 สถาปัตยกรรมเน็ตเวิร์คแบบเพียร์-ทู-เพียร์ ชนิด โมเด็มต่อโมเด็ม

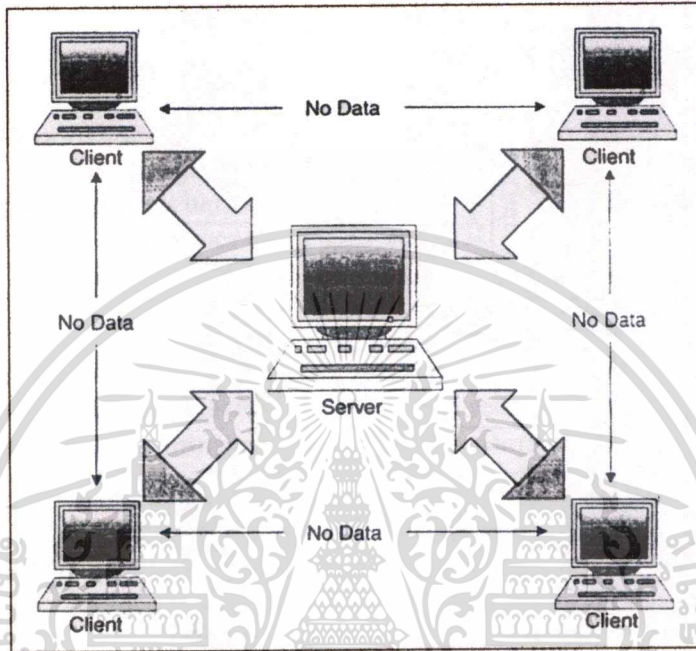
ปัจจุบันเกมที่พัฒนาจากสถาปัตยกรรมแบบเพียร์-ทู-เพียร์ ได้มีการพัฒนาโดยสามารถทำการ เล่นผ่านอินเทอร์เน็ตได้ ซึ่งมีข้อดีกว่าการเล่นผ่านแลนและการเชื่อมต่อผ่าน โมเด็มกับ โมเด็มตรงที่ สามารถเล่นได้พร้อมกันได้มากกว่า 2 คน แต่จำนวนผู้เล่นสูงสุดยังคงถูกจำกัดไว้ตามฉากหรือแผนที่ที่โฮสต์เลือกเล่น การทำให้เกิดการเล่นในรูปแบบนี้ได้จำเป็นต้องมีตัวเซิร์ฟเวอร์คอยดูแลจัดการ เกี่ยวกับการเชื่อมต่อระหว่างไคลเอนท์กับไคลเอนท์ด้วยกัน โดยเซิร์ฟเวอร์ในระบบเกมนี้จะทำ หน้าที่ดูแลผู้เล่นที่ยังไม่ได้สร้างห้องหรือยังไม่ได้เข้าร่วมเล่นเกมกับผู้เล่นคนอื่น และดูแลข้อมูลของ ผู้เล่น ซึ่งส่วนใหญ่ผู้เล่นจะต้องทำการเข้าระบบเพื่อเชื่อมต่อกับเซิร์ฟเวอร์ก่อน

เกมออนไลน์ประเภทนี้เป็นที่นิยมมาก ในช่วงปีพ.ศ. 2540 จนถึงปัจจุบัน เกมใหม่ๆ ที่ถูกพัฒนา ออกมามักจะใส่ฟังก์ชันการเล่นผ่านระบบเครือข่ายโดยใช้สถาปัตยกรรมแบบเพียร์-ทู-เพียร์ ทำให้ เกมมีความสามารถในการเล่นแบบหลายคนเข้ามาด้วยเกือบทุกเกม ซึ่งเป็นเรื่องปกติของเกม ที่พัฒนาออกสู่ตลาดในปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

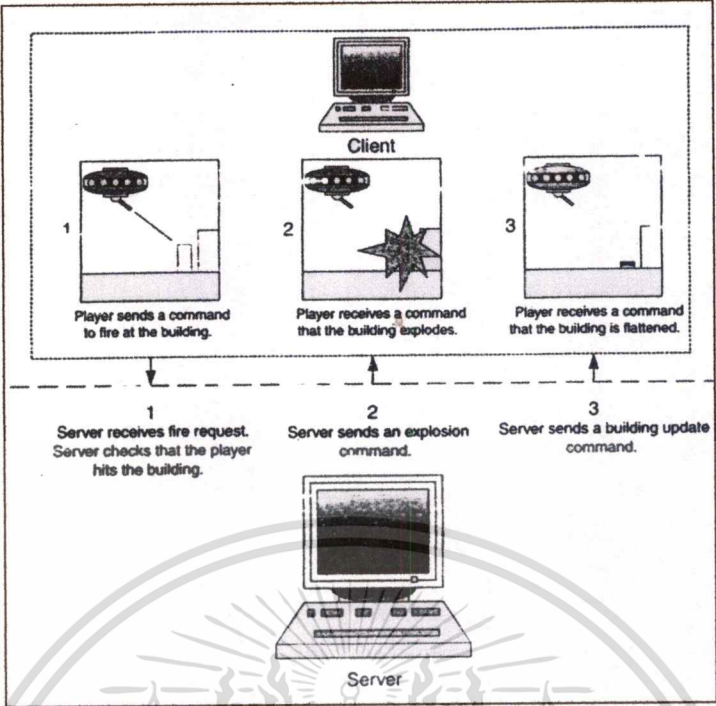
2.2.2 สถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนต์และเซิร์ฟเวอร์

สถาปัตยกรรมแบบไคลเอนต์และเซิร์ฟเวอร์จะมีตัวเซิร์ฟเวอร์คอยรับการติดต่อสื่อสารจากไคลเอนต์ โดยไคลเอนต์จะทำการติดต่อกับเซิร์ฟเวอร์เท่านั้น ไม่มีการติดต่อถึงกันระหว่างไคลเอนต์เลย ดังแสดงในรูปที่ 2.3



รูปที่ 2.3 สถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนต์และเซิร์ฟเวอร์

การติดต่อสื่อสารกันระหว่างไคลเอนต์และเซิร์ฟเวอร์นี้จะอยู่ในรูปของการร้องขอและการตอบกลับ(Request and Response) โดยเครื่องไคลเอนต์จะทำการร้องขอไปยังเซิร์ฟเวอร์ เมื่อเซิร์ฟเวอร์ได้รับการร้องขอจากไคลเอนต์จะประมวลผลสิ่งที่ไคลเอนต์ต้องการและส่งผลที่ได้และตอบกลับเป็นคำตอบ โดยมีรูปแบบการทำงานแยกส่วนของการแสดงผลและประมวลผลออกจากกัน ซึ่งเซิร์ฟเวอร์ไม่ได้เน้นการแสดงผลใดๆ แต่จะเป็นผู้ประมวลผลเกม แต่ไคลเอนต์จะทำหน้าที่เป็นผู้รับผิดชอบการแสดงผลทั้งหมด และแทบไม่ได้ทำการประมวลผลเกม เซิร์ฟเวอร์จะคอยจัดการประมวลผลที่ไคลเอนต์ส่งมาร้องขอ ซึ่งมีจำนวนมาก ดังนั้นตัวเซิร์ฟเวอร์จึงต้องเป็นเครื่องคอมพิวเตอร์ที่มีประสิทธิภาพสูง และต้องมีช่องทางการสื่อสารเพื่อให้รองรับการเชื่อมต่อจากไคลเอนต์ที่มีจำนวนมากและรวดเร็วได้ด้วย ในขณะที่ประสิทธิภาพของไคลเอนต์จะต้องการเพียงแต่ให้สามารถแสดงผลของเกมได้อย่างถูกต้องเหมาะสมเท่านั้น ดังแสดงในรูปที่ 2.4



รูปที่ 2.4 ลักษณะรูปแบบการร้องขอและการตอบกลับ

เกมที่ใช้สถาปัตยกรรมนี้สามารถรองรับผู้เล่นพร้อมๆ กันได้เป็นจำนวนมากหลายพันคน ปัจจุบันเกมออนไลน์ที่ใช้สถาปัตยกรรมนี้ มักเรียกว่า MMOG : Massive Multiplayer Online Game เนื่องจากเป็นเกมที่สามารถพบเจอกับผู้เล่นเป็นจำนวนมาก และผู้เล่นสามารถแสดงการกระทำให้ผู้เล่นคนอื่นเห็นได้ เกมประเภทนี้จึงมีลักษณะทางสังคมเกิดขึ้นภายในเกม และผู้พัฒนาเกมประเภทนี้ส่วนใหญ่จะออกแบบเกมให้มีลักษณะทางด้านสังคมเพิ่มเติมเข้าไปนอกเหนือการเล่นเกมตามปกติ จนพูดได้ว่าเป็นเหมือนโลกอีกใบหนึ่งของผู้เล่นได้เลย

2.3 ความแตกต่างของเกมที่พัฒนาโดยใช้สถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนท์และเซิร์ฟเวอร์กับเพียร์-ทู-เพียร์

การเลือกใช้สถาปัตยกรรมเน็ตเวิร์คมาใช้พัฒนาเกม จำเป็นจะต้องทราบถึงลักษณะของเกมด้วยว่าเกมมีรูปแบบออกมาในลักษณะอย่างไร และเลือกใช้สถาปัตยกรรมที่เหมาะสมที่จะพัฒนาเกมนั้น ในหัวข้อนี้จะกล่าวถึงลักษณะการทำงานของสถาปัตยกรรมเน็ตเวิร์คทั้ง 2 แบบ ข้อเด่นและข้อด้อยของสถาปัตยกรรมแต่ละตัว

2.3.1 การทำงานของสถาปัตยกรรมเน็ตเวิร์คแบบเพียร์-ทู-เพียร์

การทำงานของสถาปัตยกรรมเน็ตเวิร์คแบบเพียร์-ทู-เพียร์ เริ่มต้นจะต้องมีเพียร์คนใดคนหนึ่งทำหน้าที่เป็นโฮสต์ แล้วคอยการเชื่อมต่อจากเพียร์คนอื่นๆ นอกจากนี้ผู้ที่เป็โฮสต์ยังมีหน้าที่กระจายข้อมูลของผู้ที่เข้าร่วมกลุ่มว่ามีใครบ้างให้เพียร์ทุกเพียร์รู้จักซึ่งกันและกัน สำหรับโปรแกรมที่ใช้

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการศึกษาเท่านั้น เมื่อผู้ใดเห็นประโยชน์ในการนำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาตให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สถาปัตยกรรมแบบเน็ตเวิร์คแบบเพียร์-ทู-เพียร์ โดยทั่วไปจะมีเซิร์ฟเวอร์กลางคอยทำหน้าที่กระจายข้อมูลของเพียร์ให้สามารถทำการเชื่อมต่อและทำงานได้ แต่ในเกมจะต้องทำการตั้งกลุ่มให้พร้อมก่อนแล้วผู้ที่ป็นโฮสต์จะทำการเริ่มเกม เกมถึงจะเริ่มดำเนินการไปได้ และเมื่อเกมเริ่มต้นขึ้นแล้วผู้เล่นคนอื่นๆ ที่ไม่ได้อยู่ในกลุ่มจะไม่สามารถเข้ามาเล่นได้ และผู้เล่นที่อยู่ภายในกลุ่มแล้วออกจากเกมก็จะไม่สามารถกลับเข้ามาเล่นในกลุ่มเดิมได้อีก

การรับ-ส่งข้อมูลเกมในสถาปัตยกรรมเน็ตเวิร์คแบบเพียร์-ทู-เพียร์ โดยปกติจะทำการส่งข้อมูลที่มีการเปลี่ยนแปลงของตัวเองให้กับเพียร์ที่เหลือทั้งหมด ไม่ส่งไปที่เพียร์ใดเพียร์หนึ่ง และเพียร์ที่เป็นผู้รับจะมีข้อดคลงระหว่างกันว่าข้อมูลที่รับมาจากเพียร์ตัวอื่น ให้ถือว่าเป็นข้อมูลที่ถูกต้องและไม่มีการตรวจสอบความถูกต้องของข้อมูล ผู้รับเมื่อได้รับข้อมูลแล้วมักจะไม่มีการส่งสัญญาณกลับไปหาผู้ส่งเพื่อบอกว่าได้รับข้อมูลแล้ว เนื่องจากการเพิ่มความคับคั่งให้กับเครือข่าย ในกรณีที่ข้อมูลที่ส่งไปให้เพียร์ตัวอื่นแล้วเกิดการสูญหายระหว่างทางผู้ส่งจะไม่ส่งซ้ำ แต่จะรอจนข้อมูลมีการเปลี่ยนแปลงแล้วทำการส่งใหม่ ซึ่งผู้รับเมื่อได้ข้อมูลใหม่ก็ถือว่าเป็นข้อมูลระหว่างผู้รับและผู้ส่งตรงกัน

จากการทำงานจะเห็นว่าเครือข่ายจะมีความคับคั่งจากการส่งข้อมูลสูงมาก ผู้พัฒนาเกมจึงต้องกำหนดจำนวนสูงสุดของผู้เล่นไว้ให้เหมาะสมด้วย ในโปรแกรมบางโปรแกรมที่ใช้สถาปัตยกรรมนี้จึงมักจะต้องออกแบบให้โปรแกรมมีการตรวจสอบข้อมูลที่รับมาว่าครบหรือไม่และในกรณีข้อมูลเกิดสูญหายก็จำเป็นต้องสร้างกลไกตรวจสอบเพิ่มเข้ามาด้วย อย่างเช่น โปรแกรมประเภทให้บริการแชร์ไฟล์ข้อมูลเป็นต้น

2.3.2 การทำงานของสถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนท์และเซิร์ฟเวอร์

การทำงานของสถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนท์และเซิร์ฟเวอร์ จะทำงานโดยมีเซิร์ฟเวอร์เป็นตัวกลางทำหน้าที่ประมวลผลข้อมูลที่ได้รับการร้องขอจากตัวไคลเอนท์แล้วส่งผลที่ได้กลับไปให้ไคลเอนท์ ซึ่งตัวอย่างโปรแกรมที่มีการทำงานในลักษณะนี้ ได้แก่ โปรแกรมเว็บเซิร์ฟเวอร์ โดยโปรแกรมประเภทนี้จะทำการประมวลคำสั่งที่ได้รับจากไคลเอนท์มาประมวลผล เช่น คำสั่งขอไฟล์หน้าเว็บ ในกรณีที่หน้าเว็บที่ไคลเอนท์ทำการร้องขอเป็นไฟล์สคริปต์(Script) เซิร์ฟเวอร์จะทำการประมวลผลของสคริปต์ก่อนแล้วส่งผลสุดท้ายที่ได้จากการประมวลผลสคริปต์นั้นกลับไปให้ไคลเอนท์ การทำงานในลักษณะนี้มีข้อดีตรงที่มีความรวดเร็วกว่าส่งสคริปต์ให้ไคลเอนท์เอาไปประมวลผลเอง ไคลเอนท์ไม่จำเป็นต้องรู้จักสคริปต์ที่ใช้สร้างหน้าเว็บ และระบบปลอดภัยจากการลอกเลียนแบบรูปแบบการทำงานตามสคริปต์ด้วย และการที่ไคลเอนท์ทำการประมวลผลสคริปต์ด้วยตัวเองแล้วช้านั้นเป็นผลเนื่องจากการทำงานปกติของโปรแกรมมักจะทำงานช้าเมื่อมีการเรียกใช้งานในครั้งแรก เพราะตัวโปรแกรมจะต้องทำการโหลดข้อมูลที่ใช้ประมวลผลไปไว้ในแรม(Ram) ก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และต้องทำการแปลภาษาสคริปต์ต่างๆ ให้อยู่ในรูปคำสั่งที่หน่วยประมวลผลกลาง(CPU) เข้าใจ แล้วจึงประมวลผลได้ผลลัพธ์ที่ต้องการออกมา

จะเห็นว่าการทำงานของเว็บเซิร์ฟเวอร์ที่มีผู้เข้ามาติดต่อขอข้อมูลนั้นมีมาก การให้เซิร์ฟเวอร์เป็นผู้ประมวลผลจึงทำให้เกิดความรวดเร็วโดยรวมมากกว่าการส่งข้อมูลไปให้ไคลเอนท์ประมวลผลด้วยตัวเอง นอกจากนี้การเปลี่ยนแปลงของข้อมูลที่เซิร์ฟเวอร์ยังมีผลดีหลายอย่างด้วยกัน เช่น ทำให้ความผิดพลาดของข้อมูลที่ผู้ร้องขอได้ไปมีจำนวนน้อยลง ข้อมูลมีความทันสมัยอยู่ตลอด การอัปเดตข้อมูลสามารถทำได้สะดวกเนื่องจากอัปเดตข้อมูลที่ฝั่งเซิร์ฟเวอร์เพียงอย่างเดียว และสามารถจัดการกับข้อมูลที่มีขนาดใหญ่ได้อย่างมีประสิทธิภาพมาก เช่น ระบบฐานข้อมูลของบริษัท

ในส่วนของเกมออนไลน์ที่ใช้สถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนท์และเซิร์ฟเวอร์ ส่วนมากจะเป็นเกมที่ต้องการความสามารถการรองรับผู้เล่นจำนวนมากๆ ซึ่งไม่สามารถใช้สถาปัตยกรรมแบบเพียร์-ทู-เพียร์มาช่วยจัดการได้ และเนื่องด้วยการประมวลผลทั้งหมดอยู่ที่เซิร์ฟเวอร์ ดังนั้นการกระจายข้อมูลของผู้เล่นคนหนึ่งให้ผู้เล่นคนอื่นๆ ทราบก็สามารถทำได้รวดเร็ว ช่วยลดความคับคั่งของเครือข่ายได้ นอกจากนี้สถาปัตยกรรมนี้ยังสามารถทำการรักษาความปลอดภัยของข้อมูลได้ดีกว่าสถาปัตยกรรมแบบเพียร์-ทู-เพียร์มาก แต่เกมจะมีส่วนที่แตกต่างจากตัวโปรแกรมที่ใช้สถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนท์และเซิร์ฟเวอร์ตรงที่การอัปเดตข้อมูลจะต้องทำการอัปเดตที่ไคลเอนท์ด้วย โดยไคลเอนท์จะมีการตรวจสอบกับเซิร์ฟเวอร์ว่ามีเวอร์ชันที่ตรงกันหรือไม่ ถ้าไม่ตรงเซิร์ฟเวอร์จะต้องให้บริการอัปเดตข้อมูลให้กับไคลเอนท์ได้

2.3.3 ข้อเด่นและข้อด้อย

จากลักษณะการทำงานของสถาปัตยกรรมเน็ตเวิร์คทั้ง 2 แบบ สามารถสรุปข้อเด่นและข้อด้อยของทั้ง 2 สถาปัตยกรรมได้ดังนี้

- สถาปัตยกรรมเน็ตเวิร์คแบบเพียร์-ทู-เพียร์ สามารถพัฒนาได้ง่ายกว่าสถาปัตยกรรมไคลเอนท์และเซิร์ฟเวอร์ เพราะความซับซ้อนมีน้อยกว่า ไม่ต้องแยกพัฒนาเป็น 2 ส่วน คือ ส่วนที่เป็นเซิร์ฟเวอร์ และส่วนที่เป็นไคลเอนท์

- สถาปัตยกรรมเน็ตเวิร์คแบบเพียร์-ทู-เพียร์ ทำงานบนพื้นฐานในมุมมองของผู้มองโลกในแง่ดี คือ คิดเสมอว่าข้อมูลที่รับมาจากเพียร์อื่นเป็นข้อมูลที่มีความถูกต้องแน่นอน แต่สำหรับสถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนท์และเซิร์ฟเวอร์นั้น เซิร์ฟเวอร์สามารถสร้างกลไกตรวจสอบข้อมูลที่ส่งมาจากไคลเอนท์ว่ามีความถูกต้องหรือไม่ จึงทำให้สถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนท์และเซิร์ฟเวอร์สามารถเชื่อมั่นได้ว่าข้อมูลมีความถูกต้องและปลอดภัย

- สถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนท์และเซิร์ฟเวอร์สามารถรองรับจำนวนผู้เข้ามาติดต่อได้เป็นจำนวนมากหลายพันคน แต่สถาปัตยกรรมเน็ตเวิร์คแบบเพียร์-ทู-เพียร์ สามารถรองรับจำนวนผู้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เข้ามาติดต่อกันได้เป็นจำนวนจำกัด เนื่องการทำงานของสถาปัตยกรรมเน็ตเวิร์กแบบเพียร์-ทู-เพียร์ จะเป็นลักษณะส่งข้อมูลกระจายไปทุกเพียร์ที่ติดต่อด้วยตัวของมันเอง ทำให้เครือข่ายเกิดความคับคั่งสูง ผิดกับสถาปัตยกรรมเน็ตเวิร์กแบบไคลเอนท์และเซิร์ฟเวอร์ที่เซิร์ฟเวอร์จะเป็นผู้ประมวลผลและกระจายข้อมูลเพียงตัวเดียว และข้อมูลที่เปลี่ยนแปลงจากไคลเอนท์ก็จะถูกส่งไปให้เซิร์ฟเวอร์เพียงตัวเดียวเช่นกัน

2.4 วิเคราะห์เกมออนไลน์ที่ได้รับความนิยมในประเทศไทย

ผู้ให้บริการเกมออนไลน์ในประเทศไทยในขณะนี้มืออยู่ด้วยกันหลายบริษัท ซึ่งเกมออนไลน์ในประเทศไทยเกือบทั้งหมดล้วนเป็นเกมที่พัฒนาโดยผู้พัฒนาต่างประเทศทั้งสิ้น และเกมที่ได้รับความนิยมอย่างมาก ได้แก่ เกมเร็นนาร์โรคออนไลน์(Ragnarok Online) เกมยูทธรภพครบสถึง(Yulgang Online) เกมปิงย่า(Pangya) เกมวอร์คราฟ(Warcraft)

2.4.1 เกมเร็นนาร์โรคออนไลน์

เกมเร็นนาร์โรคออนไลน์เป็นเกมที่พัฒนาโดยบริษัทกราวิตี้(Gravity) เกมมีรูปแบบการเล่นถึงแอ็คชั่นกึ่งแฟนตาซี(Action Fantasy) และแสดงออกมาด้วยรูปแบบตัวละครที่น่ารัก การเล่นไม่ยากทำความเข้าใจได้ง่าย ผู้เล่นสามารถทำการพัฒนาตัวละครของตัวเองได้หลากหลายแบบ และมีอาชีพหลายอาชีพ เช่น อัศวิน นักบวช พ่อค้า นักรบ และโจร ซึ่งแต่ละอาชีพก็มีลักษณะเด่นเป็นของตัวเอง ทำให้ผู้เล่นมีแนวทางการเลือกเล่นได้มากมายทำให้ไม่เบื่อ

นอกจากนี้เกมเร็นนาร์โรคออนไลน์ยังมีฟังก์ชันทางด้านสังคมหลายอย่าง เช่น การพูดคุยติดต่อดสื่อสารกัน สามารถทำการแลกเปลี่ยนซื้อขายระหว่างผู้เล่นด้วยกัน มีการตั้งทีมเพื่อร่วมไปผจญภัย สามารถตั้งสมาคมภายในเกมได้ และยังสามารถจัดงานแต่งงานระหว่างผู้เล่นด้วยกันได้ ตัวเกมจะเน้นการเล่นเป็นกลุ่มเป็นทีม มีการพึ่งพาอาศัยกัน ช่วยเหลือซึ่งกันและกันในแต่ละอาชีพได้ และด้วยความเป็นเกมที่มีฟังก์ชันทางสังคมคล้ายกับสังคมของมนุษย์มาก ถึงกับมีผู้เล่นบางคนถึงกับคิดว่าเป็นเรื่องที่เกิดขึ้นในชีวิตของตนเองจริงๆ

ตัวเกมถูกพัฒนาโดยใช้สถาปัตยกรรมแบบไคลเอนท์และเซิร์ฟเวอร์ ข้อมูลมีการเก็บอยู่ที่เซิร์ฟเวอร์ การเข้าเล่นเกมจะมีการตรวจสอบตัวตนของผู้เล่น การประมวลผลต่างๆ จะอยู่ที่เซิร์ฟเวอร์ทั้งหมดยกเว้นการประมวลผลทางด้านกราฟฟิคที่จะถูกประมวลผลจากไคลเอนท์

ปัจจุบันกลุ่มผู้เล่นชาวไทยให้ความสนใจกับเกมเร็นนาร์โรคออนไลน์เป็นอย่างมาก ปัจจุบันมียอดการลงทะเบียนเล่นเกมนี้มากกว่า 5 ล้านคน



รูปที่ 2.5 ตัวอย่างภาพเกมเร็นน่าร็อคออนไลน์

2.4.2 เกมยุทธภพครบสสิ่ง

เกมยุทธภพครบสสิ่งเป็นเกมที่พัฒนาโดยบริษัทเอ็มเกม(MGAME) เกมมีลักษณะการเล่นถึงแอ็คชั่นถึงแฟนตาซีเช่นเดียวกับเร็นน่าร็อคออนไลน์ และมีการแสดงผลเป็น 3 มิติทั้งเกม ไม่ว่าจะเป็นตัวละคร ฉากแผนที่ อาคารบ้านเรือนและของประดับต่าง ๆ หรือแม้กระทั่งตัวศัตรู ก็แสดงออกมาในรูปแบบ 3 มิติทั้งหมด ตัวละครมีรูปแบบน่ารัก มีอาชีพหลายอาชีพ แต่ละอาชีพก็มีลักษณะเด่นเป็นของตัวเอง

ตัวเกมค่อนข้างเน้นการเล่นเป็นทีมมากกว่าเกมเร็นน่าร็อค โดยศัตรูในเกมค่อนข้างเก่ง ในส่วนของฟังก์ชันทางสังคมนั้นค่อนข้างมีน้อยกว่าเกมเร็นน่าร็อค เนื่องจากเป็นเกมที่สร้างโดยอ้างอิงมาจากการ์ตูน ผู้เล่นสามารถเข้าสำนักวิชาต่างๆ เพื่อฝึกวิชาได้ ทำให้ผู้เล่นแต่ละคนมีความแตกต่างกันได้หลากหลายรูปแบบ จุดเด่นของตัวเกมนี้นี้คือ ความน่ารักของตัวละคร และเครื่องแต่งกาย อาวุธต่างๆ ที่สามารถเปลี่ยนแปลงได้ตามสิ่งของที่ผู้เล่นใส่ให้กับตัวละคร

ตัวเกมถูกพัฒนาโดยใช้สถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนท์และเซิร์ฟเวอร์เช่นเดียวกับเกมเร็นน่าร็อค ปัจจุบันมีผู้เล่นให้ความสนใจและมียอดลงทะเบียนมากกว่า 1.5 ล้านคน



รูปที่ 2.6 ตัวอย่างภาพเกมยุทธภพครบสถึง

2.4.3 เกมปิงปอง

เกมปิงปองเป็นเกมที่พัฒนาโดยบริษัทเอ็นทรีวี(NTREEV) และฮันบิตซอฟท์(HanbiSoft) ตัวเกมเป็นแนวกีฬาตีกอล์ฟ เป็นหนึ่งในเกมออนไลน์เพียงไม่กี่เกมที่มีรูปแบบเป็นการเล่นเป็นการแข่งขันทางด้านกีฬา จุดเด่นของเกมอยู่ที่ความน่ารักของตัวละคร ฉากมีสีสันสดใส การแสดงผลอยู่ในรูปแบบ 3 มิติ ตัวเกมมีการเพิ่มลูกเล่นต่างๆ ในการแข่งขันตีกอล์ฟหลายหลายรูปแบบ โดยใช้วิธีการกดปุ่มเหมือนเป็นการใส่คำสั่ง เพื่อเพิ่มความสุขให้กับผู้เล่น และเพิ่มความหลากหลายรูปแบบในการตีกอล์ฟ

เกมปิงปองเป็นเกมที่เน้นการแข่งขันระหว่างผู้เล่นเป็นหลัก มีฉากสนามที่ใช้ดีหลายสนามและหลากหลายรูปแบบ ภายในเกมมีการจัดอันดับความสามารถของผู้เล่น ซึ่งเหมือนผู้เล่นได้เข้าไปแข่งขันกอล์ฟกันจริงๆ ภายในเกมยังมีการจัดกิจกรรมต่างๆ จากผู้ให้บริการเกมมากมายหลายอย่าง นอกจากนี้ตัวเกมยังมีระบบการซื้อสินค้าเพื่อแต่งตัวให้กับตัวละครได้มากมาย เช่น หมวก เสื้อผ้า ลักษณะต่างๆ ทางเกมหลากหลายรูปแบบ ไม้ตีกอล์ฟลักษณะต่างๆ รองเท้า และแม้แต่ผู้ช่วยตีกอล์ฟ หรือที่เรียกว่าแคนดี้ ก็สามารถทำการเปลี่ยนได้ด้วย ซึ่งสิ่งของที่ใส่แต่งตัวนั้นบางชนิดจะมีค่าความสามารถพิเศษเพื่อช่วยเพิ่มความสามารถให้กับตัวละครได้อีกด้วย

เกมถูกพัฒนาโดยใช้สถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนท์และเซิร์ฟเวอร์เช่นเดียวกับเกมเร็นนาร์โรค และเกมยุทธภพครบสถึง และมีผู้เล่นให้ความสนใจเป็นจำนวนมากเช่นกัน



รูปที่ 2.7 ตัวอย่างภาพเกมปิงปอง

2.4.4 เกมเวิร์ดคราฟ

เกมเวิร์ดคราฟเป็นเกมที่พัฒนาโดยบริษัทบริซซาร์ด(Blizzard) เป็นเกมออนไลน์ที่แตกต่างจากเกมออนไลน์ที่ได้กล่าวไว้ด้านบน ซึ่งพื้นฐานของเกมนี้เป็นเกมสำหรับเล่นคนเดียว ตัวเกมจะมีเนื้อเรื่องให้ผู้เล่นได้ติดตามเป็นตอนๆ ปัจจุบันเกมนี้ได้ออกมาเป็นภาคที่ 3 และมีการเพิ่มลูกเล่นใหม่ๆ เข้าไปในเกมทุกๆ ภาค โดยในภาค 3 ได้รับความนิยมอย่างมาก เนื่องจากผู้พัฒนาได้เพิ่มตัวละครพิเศษเข้าไปในเกม ซึ่งเป็นตัวละครที่มีความสามารถเหนือกว่าตัวละครอื่นๆ ทั่วๆ ไป เรียกว่า “ฮีโร่” นอกจากนี้ผู้เล่นยังสามารถซื้อของเก็บไว้ให้ตัวฮีโร่ใช้ได้อีกด้วย ตัวเกมผู้พัฒนายังได้ใส่ระบบการจับกลุ่มเล่นพร้อมๆ กันได้ โดยส่วนใหญ่จะเล่นผ่านแลน สามารถรองรับการเล่นได้พร้อมกันสูงสุด 8 คน และผู้พัฒนายังได้พัฒนาตัวสร้างฉากติดมากับตัวเกมด้วย ซึ่งเป็นเครื่องมือสร้างฉากที่มีประสิทธิภาพสูง และมีผู้เล่นให้ความสนใจอยากทดสอบฝีมือการสร้างฉากของตัวเอง แล้วทำการส่งไปเก็บไว้ที่เซิร์ฟเวอร์ของผู้พัฒนา ผู้เล่นคนอื่นๆ ก็สามารถดาวน์โหลดเอาฉากที่สร้างโดยผู้เล่นคนอื่นทำไว้มาเล่นได้ โดยบางฉากได้รับความนิยมสูงมากจนถึงกับมีการสร้างหน้าเว็บ และมีทีมพัฒนาการสร้างฉากออกมาอย่างต่อเนื่องเลขที่เดียว โดยกลุ่มผู้พัฒนาจากที่วนนี้ใช้ชื่อว่า ดอตเอออลสตาร์(DotA Allstars)

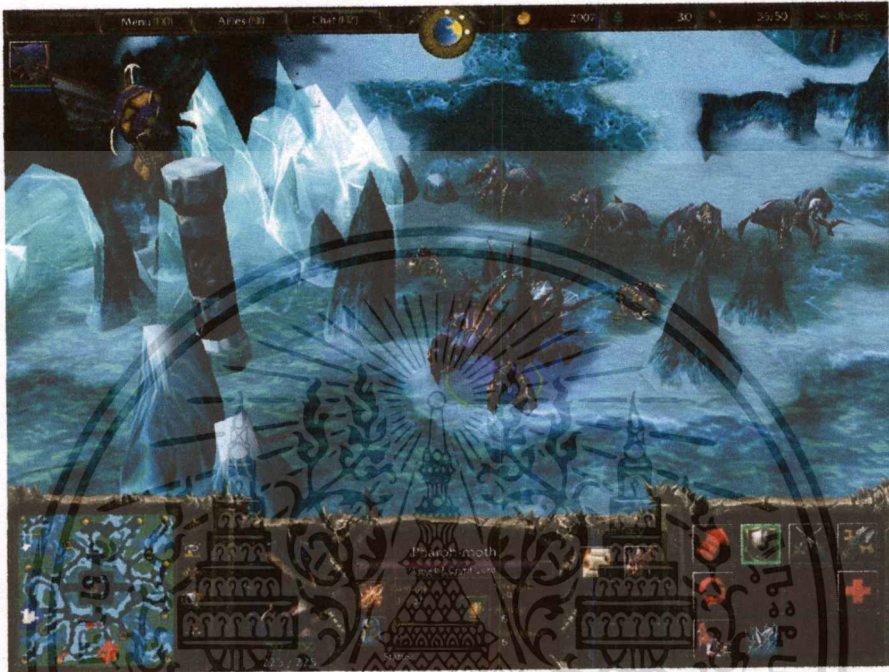
เนื่องจากเป็นเกมที่มีความนิยมเป็นอย่างมากจากผู้เล่นทั่วโลก ทำให้ผู้พัฒนาได้ทำการเพิ่มฟังก์ชันการเล่นออนไลน์ผ่านระบบอินเทอร์เน็ตเข้าไป และผู้พัฒนาได้ทำการพัฒนาตัวเซิร์ฟเวอร์เพื่อใช้เป็นที่พบปะกันของผู้เล่นทั่วโลก ทำให้เกมนี้เป็นเกมที่พัฒนาโดยใช้สถาปัตยกรรม 2 แบบ คือ สถาปัตยกรรมแบบไคลเอนท์และเซิร์ฟเวอร์สำหรับเล่นผ่านระบบอินเทอร์เน็ตและเชื่อมต่อไป

เอกสารนี้เป็นเอกสารทบทวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ยังเซิร์ฟเวอร์ที่ผู้พัฒนาได้ติดตั้งไว้ในจุดต่างๆ ทั่วโลก และสถาปัตยกรรมแบบเพียร์-ทู-เพียร์สำหรับเล่นผ่านแลน

ตัวเกมมีลักษณะเป็นเกมแนววางแผนการรบ โดยผู้เล่นจะมีฐานเป็นของตัวเอง และหน้าที่ของผู้เล่นคือการรักษาฐานของตัวเอง และทำลายฐานของผู้เล่นทีมตรงข้าม



รูปที่ 2.8 ตัวอย่างภาพเกมวอร์ครีฟ

2.5 วิเคราะห์ข้อจำกัดในการออกแบบเกมออนไลน์ในปัจจุบัน

เกมออนไลน์ส่วนใหญ่จะถูกพัฒนาโดยใช้สถาปัตยกรรมเน็ตเวิร์กแบบไคลเอนท์และเซิร์ฟเวอร์เป็นหลัก สิ่งที่เป็นข้อจำกัดที่เด่นชัดที่สุด คือ การลงทุนสำหรับการพัฒนา เนื่องจากมีความซับซ้อนในการพัฒนาสูง นอกจากนี้ยังมีส่วนของอุปกรณ์ที่ต้องลงทุนลงไปสูงมาก จึงเป็นที่สังเกตได้ว่าผู้พัฒนาเกมในประเทศไทย ไม่ค่อยกล้าที่จะเสี่ยงพัฒนาเกมออนไลน์ซักเท่าไร

และด้วยเกมถูกพัฒนาโดยสถาปัตยกรรมแบบไคลเอนท์และเซิร์ฟเวอร์ ข้อมูลต่างๆ ของเกมส่วนใหญ่จึงเป็นข้อมูลที่ต้องมีการแชร์กันระหว่างผู้เล่นทุกคน เช่น ฉากแผนที่ในเกมที่เป็นระบบ 3 มิติแบบสมจริง ผู้เล่นจะไม่สามารถยืนอยู่ที่จุดเดียวกันพร้อมๆ กันมากกว่า 1 คน ทำให้ในส่วนของการออกแบบเกมมีข้อจำกัดบางประการที่ทำให้ไม่สามารถพัฒนาเกมเพื่อให้รองรับกับจำนวนผู้เล่นที่มีจำนวนมากได้อย่างเพียงพอ เช่น การกำหนดภาระกิจภายในเกมโดยให้ผู้เล่นทำการล่าสัตว์บางชนิดเพื่อเก็บของมาแลกเปลี่ยนเงิน ซึ่งถ้าผู้เล่นมีจำนวนมากๆ ก็ไม่สามารถรองรับผู้เล่นให้ทำการภารกิจได้ครบทุกคน หรือในบางเกมจะมีการกำหนดให้ผู้เล่นจับกลุ่มแล้วทำการภารกิจโดยเข้าไปยังสถานที่ทำการภารกิจได้ต้องมีเงื่อนไขให้รวมกลุ่มให้ได้ 4 คน ซึ่งจะสามารถเข้าได้ทีละกลุ่มเท่านั้น ที่เป็น

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้ในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เช่นนี้ เนื่องจากภาระกิจนี้ไม่สามารถรองรับผู้เล่นได้เกิน 4 คนนั่นเอง และฉากที่แสดงก็เป็นฉากที่ต้องมีการแชร์ข้อมูลให้กับผู้เล่นทุกคนที่เข้าไปเล่น ซึ่งถือว่าเป็นข้อจำกัดเช่นกัน

2.6 การเข้ารหัสและถอดรหัสลับแบบกุญแจส่วนตัว(Private Key Cryptography, Secret Key Cryptography)

การเข้ารหัสแบบกุญแจส่วนตัวหรือสมมาตร เป็นการเข้าและถอดรหัสโดยใช้กุญแจตัวเดียวกันหรือต่างกัน โดยที่การใช้กุญแจต่างกันจะสามารถคำนวณหากุญแจอันหนึ่งได้จากกุญแจอีกอัน จึงเรียกว่าเป็นแบบสมมาตร ซึ่งถ้ากุญแจรั่วไหลจะทำให้ระบบที่ใช้การเข้ารหัสแบบนี้ไม่ปลอดภัยทันที ดังนั้นกุญแจแบบส่วนตัวจะต้องเก็บรักษาไว้อย่างดี ตัวที่เป็นมาตรฐานของการเข้ารหัสในระบบนี้ได้แก่ Data Encryption Standard (DES) คิดค้นโดย IBM DES ใช้ key 56 bits + 8 parity bits = 64 bits เป็น block cipher เข้ารหัสเป็น block ขนาด 64 bits การเข้ารหัสจะแบ่ง block ออกเป็นสองซีกซ้าย-ขวาแล้วเข้ารหัสด้วย key จากนั้นทำการสลับด้านซ้าย-ขวา สร้าง key ใหม่จาก key เดิมมาเข้ารหัส ซ้ำโดยทำถึง 16 รอบ แต่เนื่องจาก key มีขนาดเล็กทำให้ไม่ปลอดภัย ต่อมาจึงได้มีการคิดค้นโดยปรับจาก DES เดิมเป็น Triple-DES คือใช้ key สามอันเข้า DES สามครั้งต่อ block ทำให้มีความปลอดภัยขึ้นมาก และ IDEA(International Data Encryption Algorithm) ที่มีการทำงานเหมือน DES แต่ที่เป็น IDEA เพราะ DES ถูกจดเป็นมาตรฐานในสหรัฐอเมริกา ซึ่งมีกฎหมายการใช้งานควบคุมอยู่ ทำให้ประเทศอื่นใช้งานไม่สะดวก IDEA พัฒนาในยุโรป และ key ขนาด 128 bits ซึ่งมีความปลอดภัยกว่า DES มาก

2.7 การเข้ารหัสและถอดรหัสลับแบบกุญแจสาธารณะ(Public Key Cryptography)

การรหัสแบบกุญแจสาธารณะหรือสมมาตร เป็นการเข้าและถอดรหัสด้วยกุญแจที่ต่างกัน ด้วยการใช้คู่กุญแจ คือ กุญแจสาธารณะ (Public Key) และกุญแจส่วนตัว (Private Key) โดยที่ไม่สามารถจะคำนวณหาค่าของกุญแจที่ใช้ในการถอดรหัสจากค่ากุญแจที่ใช้เข้ารหัสลับได้ หรืออาจจะต้องใช้เวลานานมากในการคำนวณหา โดยที่ข้อมูลจากผู้ส่งจะถูกเข้ารหัสด้วยกุญแจส่วนตัวและจะถูกถอดรหัสได้โดยการใช้กุญแจสาธารณะที่เป็นคู่ของมันเท่านั้น ซึ่งการเข้ารหัสแบบนี้สามารถเปิดเผยกุญแจสาธารณะได้ แต่กุญแจส่วนตัวต้องเก็บไว้กับตัว ห้ามให้คนอื่นรู้ การเข้ารหัสแบบกุญแจสมมาตรช่วยรักษาความปลอดภัยของข้อมูลได้เป็นอย่างดี นำไปสู่การเข้ารหัสและถอดรหัสลับแบบใช้กุญแจสาธารณะด้วยวิธีการเข้ารหัสแบบ RSA ซึ่งมาจากชื่อคนคิด 3 คน คือ Ron Rivest, Adi Shamir, และ Leonard Adleman ในปี 1978 ด้วยหลักการทางคณิตศาสตร์ของจำนวนเฉพาะ (Prime number) และการแยกตัวประกอบของจำนวนที่มีขนาดใหญ่ โดยใช้ mod ในการคำนวณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อให้หาค่าจำนวนเศษที่เหลือจากการหาร ซึ่งการเข้ารหัสและถอดรหัสแบบ RSA เป็นระบบการเข้ารหัสแบบกุญแจสาธารณะที่รู้จักกันแพร่หลายที่สุด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

เกมออนไลน์

3.1 ภาษาที่ใช้พัฒนาเกมออนไลน์

โครงงานพิเศษนี้จะใช้ภาษาไมโครซอฟท์ฟิวลชีพลัสพลัส(Microsoft Visual C++) ซึ่งเป็นภาษาที่ถูกพัฒนาโดยบริษัทไมโครซอฟท์(Microsoft) เป็นภาษาที่มีความยืดหยุ่นสูงสามารถสร้างโปรแกรมได้หลากหลายรูปแบบ โปรแกรมยังมีความสามารถสร้างฟอร์มได้เหมือนไมโครซอฟท์ฟิวลเบสิก(Microsoft Visual Basic) ด้วย ซึ่งเรียกว่า “เอ็มเอฟซี(MFC)” ภายในบรรจุคลาสต่างๆ ไว้อำนวยความสะดวกในการเขียนโปรแกรมมากมาย การสร้างโปรแกรมบนระบบปฏิบัติการวินโดวส์ทำได้ง่าย แต่การพัฒนาโปรแกรมด้วยภาษานี้ยังทำได้ยากกว่าไมโครซอฟท์ฟิวลเบสิก เนื่องจากความยืดหยุ่นที่มีมาก ดังนั้นจึงมีผู้พัฒนาโดยใช้ภาษานี้น้อยกว่า แต่ด้วยความเป็นภาษาที่มีประสิทธิภาพสูง และมีการทำงานที่รวดเร็วกว่าภาษาอื่นมาก จึงเป็นภาษาที่โคเรคเอ็กซ์(DirectX) ออกมารองรับตั้งแต่เวอร์ชันแรกๆ จนกระทั่งปัจจุบัน และโคเรคเอ็กซ์ยังมีการอัปเดตสำหรับภาษานี้บ่อยที่สุดด้วย

ภาษาซีเป็นภาษาที่นักพัฒนาเกมส่วนใหญ่เลือกใช้ โดยสามารถพัฒนาโปรแกรมและเกมสำหรับเครื่องคอมพิวเตอร์ โปรแกรมและเกมสำหรับมือถือที่เป็นระบบปฏิบัติการซิมเบียน(Symbian) หรือแม้กระทั่งพัฒนาเกมบนเครื่องเล่นเกมตามบ้าน เช่น เครื่องเล่นเกมเอ็กซ์บ็อก(Xbox) ของบริษัทไมโครซอฟท์ นอกจากนี้ภาษาซียังสามารถพัฒนาโปรแกรมได้ทุกประเภทอีกด้วย ไม่ว่าจะเป็นโปรแกรมมัลติมีเดียทั้งหลาย เช่น โปรแกรมดูหนัง ฟังเพลง โปรแกรมคำนวณต่างๆ โปรแกรมเกี่ยวกับกราฟฟิก หรือแม้แต่โปรแกรมควบคุมเครื่องจักรก็สามารถพัฒนาได้ด้วยภาษานี้

จุดเด่นอีกประการหนึ่งของไมโครซอฟท์ฟิวลชีพลัสพลัสก็คือ ความสามารถในการเพิ่มชุดคำสั่งเข้าไปได้ โดยตัวโปรแกรมยอมให้ผู้พัฒนากำหนดได้ว่าต้องการจะเพิ่มชุดคำสั่งอะไรเข้าไปเพิ่มจากของเดิมที่มีอยู่แล้ว ด้วยเหตุนี้เองจึงทำให้ไมโครซอฟท์ฟิวลชีพลัสพลัสเป็นภาษาที่มีความยืดหยุ่นสูง

3.2 เครื่องมือสำหรับพัฒนาเกมออนไลน์

หัวข้อนี้จะกล่าวถึงเครื่องมือที่ช่วยอำนวยความสะดวก ความรวดเร็วและช่วยเพิ่มประสิทธิภาพให้กับผู้พัฒนาเกม โดยแบ่งเป็น 2 หัวข้อ คือ ซอร์ฟแวร์ชุดคำสั่งโคเรคเอ็กซ์ เป็นซอร์ฟแวร์ที่มีความสำคัญมากสำหรับใช้พัฒนาเกม ซึ่งทำให้ผู้พัฒนาสามารถควบคุมอุปกรณ์ได้โดยตรงทำให้ตัว

เกมที่พัฒนามีการตอบสนองกับผู้เล่น ได้อย่างรวดเร็ว และซอร์ฟแวร์ชุดคำสั่งสำหรับช่วยพัฒนาเกม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นซอฟต์แวร์ที่เก็บฟังก์ชันการทำงานที่มักมีการเรียกใช้บ่อยในการสร้างเกม ทำให้การพัฒนาเกมทำได้สะดวกรวดเร็วขึ้น

3.2.1 ซอฟต์แวร์ชุดคำสั่งไคลเรคเอ็กซ์

ไคลเรคเอ็กซ์เดิมทีถูกเรียกว่า “ชุดซอฟต์แวร์พัฒนาเกม(Game SDK)” เป็นเอพีไอ(API) สำหรับจัดการทรัพยากรในการพัฒนาเกมบนวินโดวส์ แต่ก่อนไคลเรคเอ็กซ์จะถูกผูกติดไปกับตัวติดตั้งเกมที่ผู้พัฒนาเกมใช้ไคลเรคเอ็กซ์พัฒนาเกม ปัจจุบันไคลเรคเอ็กซ์ได้ถูกรวมเข้าไว้ในตัวระบบปฏิบัติการวินโดวส์แล้ว แต่ถึงแม้ว่าไคลเรคเอ็กซ์จะได้อยู่รวมเข้าไปอยู่ในระบบปฏิบัติการแล้วก็ตาม ผู้พัฒนาเกมก็ยังคงมีการผูกตัวไคลเรคเอ็กซ์ไปกับตัวติดตั้งเกมที่ออกมาใหม่อยู่เรื่อยๆ สาเหตุมาจากตัวชุดคำสั่งไคลเรคเอ็กซ์มีการพัฒนาออกมาบ่อยครั้ง

บริษัทไมโครซอฟท์ซึ่งเป็นผู้พัฒนาไคลเรคเอ็กซ์โดยในตอนแรกตั้งใจออกแบบให้ไคลเรคเอ็กซ์ถูกนำมาใช้ในอุตสาหกรรมพัฒนาเกมเท่านั้น แต่ในปัจจุบันมีการนำไคลเรคเอ็กซ์ไปใช้พัฒนากับซอฟต์แวร์ต่างๆ มากมาย เนื่องจากไคลเรคเอ็กซ์มีชุดคำสั่งที่สามารถประมวลผลภาพ 3 มิติ ได้อย่างรวดเร็วและมีคุณภาพ โดยใช้ร่วมกับความสามารถของการ์ดแสดงผล 3 มิติ ได้เป็นอย่างดีโดยที่มีพื้นมาจากโอเพ่นจีแอล(OpenGL) ทำให้เกิดชุดคำสั่งไคลเรคทรีดี(Direct3D) ต่อมาไคลเรคเอ็กซ์จึงได้ถูกพัฒนาให้รองรับงานด้านมัลติมีเดียได้ดีมากยิ่งขึ้น และเริ่มเข้ามาใช้ครั้งแรกกับระบบปฏิบัติการวินโดวส์เวอร์ชัน 95

ปัจจุบันไคลเรคเอ็กซ์ถูกนำมาใช้ในอุตสาหกรรมพัฒนาซอฟต์แวร์หลากหลายอย่างมาก เช่น เกมคอมพิวเตอร์ ซอฟต์แวร์ประเภทมัลติมีเดียต่างๆ ไม่ว่าจะเป็นซอฟต์แวร์สำหรับฟังเพลง ดูหนัง เป็นต้น และนอกจากนี้ไคลเรคเอ็กซ์ยังได้ถูกพัฒนาฝังไว้กับเครื่องเล่นเกมตามบ้าน อย่างเครื่องเล่นเกมเอ็กซ์บ็อกซ์ซึ่งเป็นผลิตภัณฑ์ของบริษัทไมโครซอฟท์เช่นกัน

ไคลเรคเอ็กซ์ประกอบไปด้วยเอพีไอต่างๆ มากมาย ซึ่งทำหน้าที่ต่างกัน โดยเอพีไอเหล่านี้จะอยู่ในรูปของออบเจกต์คอม(COM-Compliant) ดังนี้

3.2.1.1 ไคลเรคเอ็กซ์กราฟฟิกส์(DirectX Graphics)

ประกอบไปด้วยเอพีไอ 2 ตัว คือ

- ไคลเรคครอว์(DirectDraw)

เป็นตัวที่ใช้จัดการวาดภาพลงบนหน้าจอ ไคลเรคครอว์สามารถทำงานกับข้อมูลที่อยู่ในการ์ดแสดงผลได้โดยตรง ทำให้มีความเร็วในการแสดงผลออกทางหน้าจอ โดยไคลเรคครอว์มีฟังก์ชันในการจัดการกับรูปภาพที่ต้องการแสดงหลายอย่าง เช่น การทำให้ภาพโปร่งใส การตัดรูปภาพเฉพาะบางส่วนของภาพไปแสดงผลหน้าจอ การตัดฉากหลังของภาพออกโดยใช้คัลเลอร์คีย์(Color key) เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ไคเรคทีรีดี(Direct3D)

เป็นตัวที่ใช้จัดการการประมวลผลภาพในรูปแบบ 3 มิติ จำเป็นต้องใช้ร่วมกับไคเรครอว์ในการแสดงผล เนื่องจากไคเรคทีรีดีจะทำการประมวลผลภาพ 3 มิติว่าจะออกมาเป็นรูปลักษณะอย่างไร แสงและเงาของภาพที่ได้ออกมาเป็นอย่างไรเท่านั้น เมื่อต้องการแสดงผลก็จะทำการส่งผลของรูปภาพที่ได้คำนวณไว้แล้วให้ไคเรครอว์ทำการวาดออกทางหน้าจออีกทอดหนึ่ง

3.2.1.2 ไคเรคอินพุต(DirectInput)

จัดการรับค่าอินพุตจากอุปกรณ์ต่างๆ ที่เชื่อมต่ออยู่ เช่น คีย์บอร์ด, เมาส์ และจอยสติ๊ก เป็นต้น ตัวไคเรคอินพุตสามารถรับค่าจากอุปกรณ์ได้โดยตรงไม่ต้องผ่านระบบปฏิบัติการวินโดวส์ ทำให้มีความรวดเร็วกว่ารับค่าอินพุตผ่านระบบปฏิบัติการมาก มีผลให้ตัวโปรแกรมหรือเกมที่สร้างสามารถตอบสนองการควบคุมจากผู้ใช้งานได้ดี

3.2.1.3 ไคเรคซาวน์(DirectSound)

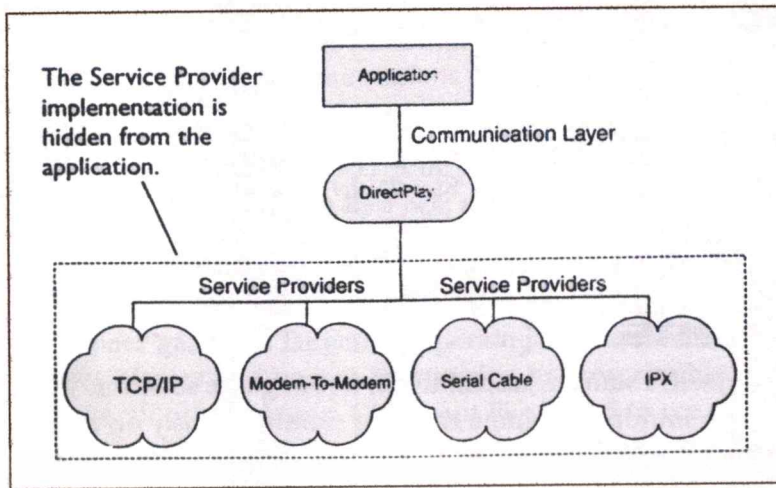
จัดการควบคุมและบันทึก เสียงในรูปแบบเวฟ(WAV) ซึ่งส่วนใหญ่จะใช้กับเสียงที่เป็นเสียงเอฟเฟ็คต่างๆ เนื่องจากไคเรคซาวน์สามารถเพิ่มลูกเล่นกับเสียงได้ เช่น การให้เสียงออกลำโพงด้านซ้ายหรือด้านขวา หรือการเพิ่มลดความดังของเสียง

3.2.1.4 ไคเรคมิวสิก(DirectMusic)

จัดการควบคุมการเล่นเสียงเช่นเดียวกับไคเรคซาวน์ แต่ไม่มีความสามารถด้านเพิ่มลูกเล่นให้เสียงและไคเรคมิวสิก ยังรองรับกับไฟล์เสียงดนตรีอย่างมีดี(MIDI) จึงทำให้ไคเรคมิวสิก มักถูกนำมาใช้เล่นเสียงทั่วๆ ไป หรือใช้เป็นเสียงของประกอบฉาก

3.2.1.5 ไคเรคเพล(DirectPlay)

จัดการด้านการสื่อสารผ่านระบบเน็ตเวิร์คต่างๆ เช่น แลน, โมเด็ม, อินเทอร์เน็ต และการเชื่อมต่อผ่านพอร์ตอนุกรมต่างๆ มีส่วนสำคัญที่จะช่วยให้การพัฒนาเกมในรูปแบบเล่นได้หลายคนทำได้ง่ายขึ้นมาก โดยผู้พัฒนาไม่จำเป็นต้องศึกษาถึงระดับซ็อกเก็ต(Socket) เพราะไคเรคเพลจะครอบการทำงานอยู่บนซ็อกเก็ตอีกทีหนึ่ง ไคเรคเพลสามารถรองรับการพัฒนาโดยใช้ได้ทั้งสถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนท์และเซิร์ฟเวอร์ หรือสถาปัตยกรรมเน็ตเวิร์คแบบเพียร์-ทู-เพียร์ได้ นอกจากนี้ไคเรคเพลยังทำงานครอบการเชื่อมต่อเน็ตเวิร์คอีกทีหนึ่ง ทำให้ผู้พัฒนาสามารถพัฒนาโปรแกรมให้สามารถรองรับการเชื่อมต่อไ่ว่ลงหน้าได้สะดวก แต่ในชุดพัฒนาเรคเอ็กซ์ตัวปัจจุบันไคเรคเพลได้ถูกถอดออกไปแล้ว สถาปัตยกรรมกรรมของไคเรคเพลแสดงดังรูปที่ 3.1



รูปที่ 3.1 สถาปัตยกรรมของไดเรคเพล

3.2.2 ซอร์ฟแวร์ชุดคำสั่งสำหรับช่วยพัฒนาเกม

ซอร์ฟแวร์ประเภทนี้มักเป็นซอร์ฟแวร์ที่ครอบคลุมการทำงานของไดเรคเอ็กซ์ หรือ โอเพ่นจีแอลอีท ทอดหนึ่ง ภายในซอร์ฟแวร์มีฟังก์ชันต่างๆ ที่มักต้องใช้อยู่ในขั้นตอนการพัฒนาเกม เช่น ฟังก์ชัน การวาดภาพ ฟังก์ชันการตัดภาพ หรืออาจเป็นฟังก์ชันประเภทควบคุมการรับอินพุตต่างๆ เป็นต้น การที่มีซอร์ฟแวร์ประเภทนี้เกิดขึ้น เนื่องมาจากในการพัฒนาเกม ถึงแม้ว่าจะมีชุดคำสั่งไดเรคเอ็กซ์ มาช่วยครอบคลุมการทำงานการควบคุมอุปกรณ์ให้ทำได้แล้วนั้น ก็ยังไม่ความซับซ้อนอยู่แน่นอน เช่น การวาดรูปหน้าจอก็จำเป็นต้องเรียกคำสั่งของไดเรคเอ็กซ์หลายคำสั่ง จึงได้มีผู้พัฒนานำชุดคำสั่ง ไดเรคเอ็กซ์มารวมไว้แล้วสร้างเป็นฟังก์ชันวาดรูป ผู้พัฒนา ก็เพียงเรียกคำสั่งวาดรูปเพียงคำสั่งเดียวก็ สามารถทำงานได้เหมือนกัน จะเห็นว่าซอร์ฟแวร์ประเภทนี้สามารถช่วยเพิ่มความสะดวก รวดเร็ว ในการพัฒนาได้มาก

ปัจจุบันมีผู้พัฒนาซอร์ฟแวร์ชุดคำสั่งสำหรับช่วยพัฒนาเกมหลายตัว ซึ่งแต่ละตัวก็จะมีจุดเด่น จุดด้อยต่างกัน ส่วนใหญ่ผู้พัฒนามักจะเจาะจงว่าพัฒนาโดยมีความสามารถทางด้านอะไรบ้าง เช่น รองรับกับเกมในรูปแบบ 2 มิติ หรือรองรับกับเกมในรูปแบบ 3 มิติ และผู้พัฒนาบางคนจะพัฒนา ออกมาให้ใช้ฟรีๆ หรือพัฒนาออกมาขายก็มีเช่นกัน โดยเรียกซอร์ฟแวร์ชุดคำสั่งสำหรับช่วยพัฒนา เกมนี้ว่า “เกมเอนจิน(Game Engine)”

3.2.2.1 ซอร์ฟแวร์ซิดีเอ็กซ์(CDX)

ซอร์ฟแวร์ซิดีเอ็กซ์ เป็นซอร์ฟแวร์ที่พัฒนาบนไดเรคเอ็กซ์ ซอร์ฟแวร์สามารถรองรับการทำงานได้ทั้งแบบ 2 มิติ และ 3 มิติ นอกจากนี้ยังรองรับการทำงานกับการเล่นเสียงด้วย โดย ซอร์ฟแวร์ถูกออกแบบมาให้สามารถใช้งานร่วมกับโปรแกรมไมโครซอฟท์วิสวลซีพลัสพลัส (Microsoft Visual C++) บอร์แลนด์ซีพลัสพลัส(Borland C++) และซิดีเอ็กซ์ยังมีข้อดีอีกก็คือเป็น

โอเพ่นซอร์ซ(Open Source) คือ ผู้พัฒนาซอร์ฟแวร์มีการปล่อยตัวโค้ดโปรแกรมให้สามารถนำมา แยกสารเป็นเอกสารที่ส่งมอบไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พัฒนาต่อไป ชอร์ฟแวร์ตัวนี้มีผู้สนใจและใช้เป็นจำนวนมาก เนื่องจากเป็นชอร์ฟแวร์ที่มีประสิทธิภาพและสามารถดาวน์โหลดได้ฟรี จึงมีผู้พัฒนาออกตัวเทมเพลต(Template) ซึ่งตัวเทมเพลตนี้รองรับกับภาษาไมโครชอร์ฟวิซวลซีพลัสพลัส โดยตัวเทมเพลตสามารถกำหนดรูปแบบของการแสดงผลเกมที่กำลังจะพัฒนาขึ้นใหม่ได้ โดยมีตัวเลือกให้ผู้พัฒนาสามารถเลือกได้อย่างสะดวก เช่น การแสดงผลแบบเต็มจอ หรือเป็นแบบวินโดว์ และยังกำหนดความละเอียดของการแสดงผลเกมได้อีกด้วย แต่น่าเสียดายที่ปัจจุบันชอร์ฟแวร์ซีดีเอ็กซ์ไม่ได้มีการพัฒนาต่ออีกแล้ว

3.2.2.2 ชอร์ฟแวร์ออร์ก(Orge)

ชอร์ฟแวร์ออร์ก(Orge) เป็นเกมเอนจินที่มีคุณภาพที่ดีอีกตัวหนึ่ง ชอร์ฟแวร์ตัวนี้เน้นการรองรับการทำงานในแบบ 3 มิติ เป็นชอร์ฟแวร์ที่สามารถดาวน์โหลดมาใช้ได้ฟรี มีฟังก์ชันการทำงานมากมาย ชอร์ฟแวร์มีการอัปเดตอยู่เรื่อยๆ เพื่อรองรับความสามารถใหม่ๆ ที่ไคเรคเอ็กซ์ได้เพิ่มเข้ามา การทำงานของชอร์ฟแวร์มีประสิทธิภาพที่ค่อนข้างดี คือ มีการแสดงผลที่รวดเร็วพอสมควร จึงเป็นชอร์ฟแวร์ที่น่าเลือกใช้อีกตัวหนึ่งสำหรับพัฒนาเกมรูปแบบ 3 มิติ

3.3 สถาปัตยกรรมเน็ตเวิร์คแบบผสม

สถาปัตยกรรมเน็ตเวิร์คแบบผสมเป็นการรวมเอาสถาปัตยกรรมเน็ตเวิร์คทั้งแบบไคลเอนท์และเซิร์ฟเวอร์ และแบบเพียร์-ทู-เพียร์ เข้าด้วยกัน การทำงานโดยใช้สถาปัตยกรรมเน็ตเวิร์คแบบผสมนี้จะมีลักษณะ นำเอาสถาปัตยกรรมแบบเพียร์-ทู-เพียร์เข้ามาใช้เพื่อทำงานบางอย่างที่ไม่จำเป็นต้องทำผ่านเซิร์ฟเวอร์ รูปแบบการทำงานจะแบ่งออกเป็น 2 ส่วน คือ ส่วนที่เป็นไคลเอนท์และเซิร์ฟเวอร์ และส่วนที่เป็นเพียร์-ทู-เพียร์

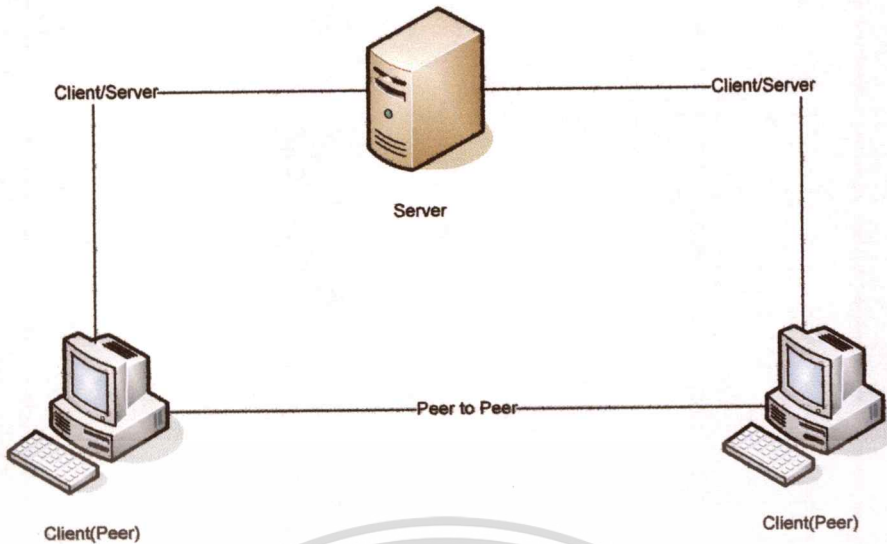
3.3.1 การทำงานส่วนที่เป็นไคลเอนท์และเซิร์ฟเวอร์

รูปแบบการทำงานจะเป็นเหมือนสถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนท์และเซิร์ฟเวอร์ทั่วไป โดยเซิร์ฟเวอร์ ทำหน้าที่จัดการประมวลผลข้อมูลและจัดเก็บข้อมูลสำหรับในบางระบบ ซึ่งระบบที่สำคัญและต้องการความสามารถในการตรวจสอบความถูกต้องหรือต้องการความปลอดภัยจะอยู่ที่เซิร์ฟเวอร์ เช่น ระบบการเข้าใช้ ระบบฐานข้อมูล เป็นต้น

3.3.2 การทำงานส่วนที่เป็นเพียร์-ทู-เพียร์

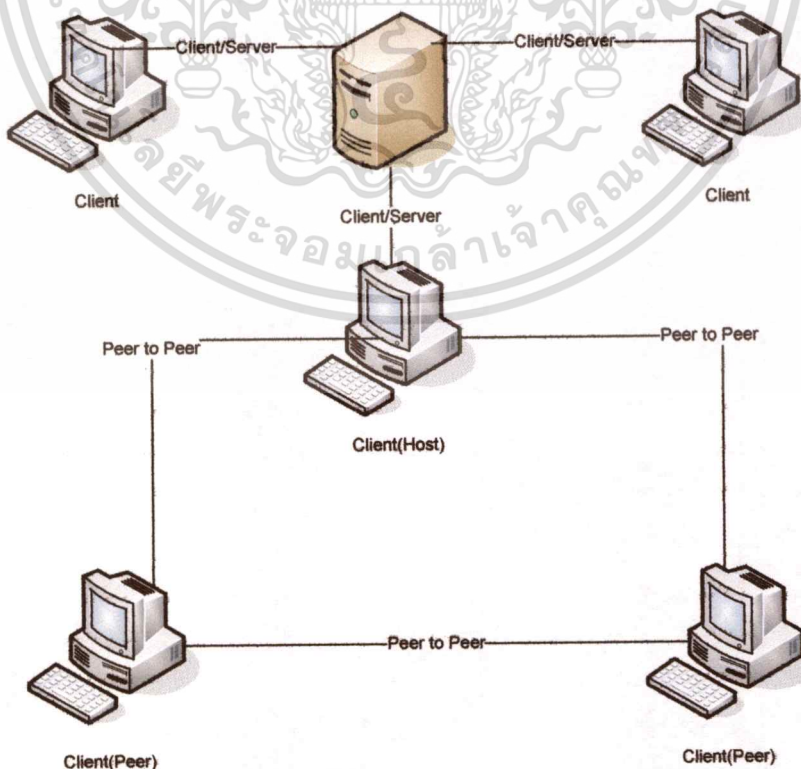
รูปแบบที่ใช้เป็นลักษณะเพียร์-ทู-เพียร์ โดยเพียร์-ทู-เพียร์จะนำมาใช้งานในส่วนที่ไม่จำเป็นต้องยุ่งเกี่ยวกับเซิร์ฟเวอร์ อย่างเช่น การส่งไฟล์ การส่งสตรีมมิ่งผ่านเน็ตเวิร์คต่างๆ เนื่องจากข้อมูลพวกนี้มักมีขนาดใหญ่ ซึ่งการส่งผ่านเซิร์ฟเวอร์จะทำให้เครือข่ายบนเซิร์ฟเวอร์มีความคับคั่งสูง โดยทั่วไปการทำงานที่เป็นเพียร์-ทู-เพียร์จะเป็นการติดต่อระหว่างไคลเอนท์ด้วยกันเป็นแบบ 1 ต่อ 1 โดยจะมีไคลเอนท์ตัวหนึ่งทำหน้าที่เปิดรับการเชื่อมต่อเอาไว้ รูปแบบการเชื่อมต่อแบบนี้แสดงได้ดังรูปที่ 3.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 การทำงานของเพียร์-ทู-เพียร์ในสถาปัตยกรรมเน็ตเวิร์คแบบผสมรูปแบบ 1 ต่อ 1

นอกจากนี้เพียร์-ทู-เพียร์ ยังมีความสามารถในการเชื่อมต่อไคลเอนต์หลายตัวได้ด้วย โดยใช้วิธีการสร้างโฮสต์มาช่วย ซึ่งจะทำให้การติดต่อระหว่างไคลเอนต์สามารถทำได้พร้อมกันหลายคน และการสร้างโฮสต์จะต้องมีการบอกให้เซิร์ฟเวอร์รู้ เนื่องจากว่าเซิร์ฟเวอร์จะเป็นผู้ช่วยกระจายข้อมูลของโฮสต์ให้กับไคลเอนต์ที่ยังไม่ได้ทำการเชื่อมต่อโฮสต์ ได้ทราบว่าขณะนี้โฮสต์อะไรที่ออนไลน์อยู่ ซึ่งแสดงได้ดังรูปที่ 3.3



รูปที่ 3.3 การทำงานของเพียร์-ทู-เพียร์ในสถาปัตยกรรมเน็ตเวิร์คแบบผสมรูปแบบสร้างโฮสต์

เอกสารนี้เป็นเอกสารของกรมส่งเสริมการค้าระหว่างประเทศ กระทรวงพาณิชย์ เพื่อใช้ในการศึกษาเท่านั้น ไม่ควรเผยแพร่หรือใช้เพื่อประโยชน์ทางการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การวิเคราะห์และออกแบบระบบ

4.1 การออกแบบเกม

การออกแบบเกมเป็นการกำหนดขอบเขตของรูปแบบการเล่นภายในเกม ซึ่งจะบอกว่าเกมนี้สามารถทำอะไรได้บ้าง ตัวเกมมีลักษณะเป็นอย่างไร เงื่อนไขต่างๆ ที่ใช้ในเกมนี้อาจจะต้องมีในส่วนนี้ด้วย

4.1.1 ลักษณะของเกม

เกมมีลักษณะเป็นบ้าน โดยผู้เล่นแต่ละคนจะมีบ้านเป็นของตัวเองภายในบ้านจะมีการเพิ่มสิ่งของเข้าไปได้โดยการซื้อสินค้า หรือสร้างขึ้นมาเอง โดยบ้านของแต่ละคนจะสามารถมีสินค้าได้สูงสุดไม่เกิน 6 ชิ้น ผู้เล่นสามารถทำการเชื่อมต่อเข้าไปยังบ้านคนอื่นได้ สามารถทำการพูดคุยกับคนในบ้านอื่นได้ และสามารถทำการซื้อสินค้าของผู้เล่นคนอื่นได้

ผู้เล่นสามารถทำการสร้างโฮสต์เพื่อให้ผู้เล่นคนอื่นเข้ามาภายในบ้านของตัวเองได้ โดยเมื่อผู้เล่นเป็นโฮสต์แล้ว ผู้เล่นจะมีสิทธิในการขายสินค้าของตัวเองได้ นอกจากนี้ผู้เล่นสามารถที่จะเข้าร่วมโฮสต์ของผู้เล่นคนอื่นได้โดยเมื่อผู้เล่นเข้าร่วมโฮสต์กับใครจะมีสิทธิในการซื้อสินค้าของผู้เล่นที่เป็นโฮสต์ที่เข้าร่วมได้

การแสดงผลจะเป็นการแสดงผลในรูปแบบ 3 มิติ โดยใช้มุมมองเฉียงและฉากเลื่อนตามการเคลื่อนที่ของตัวละครของผู้เล่น และตัวละครแต่ละตัวจะสามารถเคลื่อนที่ได้ในบริเวณที่เป็นบ้านเท่านั้น จะไม่สามารถเคลื่อนที่ออกนอกบริเวณบ้านได้

เกมไม่ได้ตรวจสอบการซ้อนทับกันของตัวละคร และสินค้าที่อยู่ภายในบ้าน ดังนั้นผู้เล่นสามารถทำการเคลื่อนที่ตัวละครทับในตำแหน่งที่ผู้เล่นคนอื่นยืนอยู่ได้ และผู้เล่นสามารถเดินซ้อนตำแหน่งที่วางสินค้าอยู่ภายในบ้านได้

4.1.2 การบังคับตัวละคร

ผู้เล่นสามารถทำการเคลื่อนที่ตัวละครไปยังตำแหน่งต่างๆ ภายในบ้านได้ โดยมีลักษณะการควบคุมตัวละครดังนี้

- กดปุ่มลูกศร ไปข้างหน้า จะทำให้ตัวละครเคลื่อนที่ไปข้างหน้า
- กดปุ่มลูกศร ไปข้างหลัง จะทำให้ตัวละครเคลื่อนที่ถอยหลัง
- กดปุ่มลูกศร ไปทางซ้าย จะทำการหมุนตัวละครไปทางซ้าย

- กดปุ่มลูกศร ไปทางขวา จะทำการหมุนตัวละครไปทางขวา

เอกสารนี้เป็นเอกสารต้นฉบับที่จัดทำขึ้นเพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.3 การทำตัวเองเป็นโฮสต์

ผู้เล่นสามารถสร้างตัวเองเป็นโฮสต์ได้โดยการกดปุ่ม H เมื่อผู้เล่นเข้าระบบเกมแล้วและยังไม่ได้ทำการรวมกลุ่มโฮสต์ของผู้เล่นคนอื่น

4.1.4 การรวมกลุ่มกับผู้เล่นที่เป็นโฮสต์อื่น

ผู้เล่นสามารถเข้าร่วมกลุ่มโฮสต์ของผู้เล่นคนอื่นได้ เมื่อผู้เล่นเข้าระบบเกมและไม่ได้ประกาศตัวเองเป็นโฮสต์ โดยมีขั้นตอนดังนี้

1. กดปุ่ม J จะปรากฏรายชื่อของโฮสต์และข้อมูลโฮสต์ที่ออนไลน์อยู่ในขณะนั้น
2. เลือกโฮสต์ที่ต้องการเข้าร่วมและกดปุ่ม Enter เกมจะทำการเข้าร่วมกลุ่มให้อัตโนมัติ

4.1.5 การลงทะเบียนสินค้ากับระบบเซิร์ฟเวอร์

ผู้เล่นสามารถสร้างสินค้าเพื่อใช้ขายในเกมได้ โดยสินค้าไม่มีการระบุชนิด รูปแบบ และขนาดของสินค้า ระบบเกมสามารถรองรับตัวข้อมูลสินค้าที่เป็นโมเดล 3 มิติ และไฟล์จะต้องเป็นไฟล์ในรูปแบบ .X เท่านั้น โดยสามารถทำการแปลงโมเดล 3 มิติจาก โปรแกรม 3 มิติ เช่น 3dsMax ได้โดยใช้ปลั๊ก-อิน(Plug-in) เช่น Panda Exporter และ kilowatt X file Exporter เป็นต้น ระบบเกมไม่ได้มีการตรวจสอบการวางสินค้า และขอบเขตของโมเดลสินค้า ดังนั้นในขั้นตอนการลงทะเบียนสินค้า ระบบเกมจะให้ผู้เล่นทำการใส่ข้อมูลของสินค้าเอง ซึ่งค่าที่เหมาะสมอาจต้องหาผ่านโปรแกรมที่สามารถปรับตัวโมเดลได้ โดยในโครงการพิเศษนี้ไม่ได้พัฒนาโปรแกรมที่ใช้ทดสอบปรับขนาด

เงื่อนไขการลงทะเบียนสินค้า

1. ผู้เล่นสามารถลงทะเบียนสินค้าของตัวเองได้จากภายในเกม
2. ผู้เล่นจะต้องไม่เป็นโฮสต์และเพียร์ภายในบ้านของผู้เล่นคนอื่น

ลำดับการลงทะเบียนสินค้า

1. ผู้เล่นทำการแปลงไฟล์จากโปรแกรม 3 มิติ
2. ทำการก๊อปปี้ไฟล์ทั้ง .X และไฟล์รูปที่เป็นเท็กเจอร์ของโมเดล ไปไว้ที่ Data/Import โดยสร้างไดเรกทอรีใหม่ ให้ตั้งชื่อไดเรกทอรี ชื่อไฟล์ .X เหมือนกันและให้ตั้งชื่อตามชื่อที่จะทำการลงทะเบียนสินค้ากับระบบเกม

3. ผู้เล่นทำการเข้าระบบเกม
4. ทำการกดปุ่ม R บนคีย์บอร์ด จะปรากฏหน้าจอให้ใส่ข้อมูลสินค้า
5. ทำการใส่ข้อมูลสินค้า โดยชื่อสินค้าจะต้องตรงกับชื่อที่ตั้งไว้ที่หัวข้อที่ 2
6. กดปุ่ม Enter เมื่อใส่ข้อมูลเรียบร้อยแล้ว จะปรากฏหน้าจอแสดงรายละเอียดสินค้าที่ผู้เล่นต้องการลงทะเบียน

7. เลือก Yes ถ้าข้อมูลถูกต้องและต้องการลงทะเบียน หรือเลือก No ถ้าต้องการยกเลิกการลงทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.6 การซื้อ-ขายสินค้าภายในเกม

ระบบซื้อขายสินค้าจะเป็นการซื้อขายอุปกรณ์ตกแต่งบ้านต่างๆ ซึ่งอาจทำการซื้อสินค้าจากเซิร์ฟเวอร์หรือซื้อจากผู้เล่นคนอื่นได้ โดยกรณีที่ผู้เล่นทำการซื้อสินค้ากับเซิร์ฟเวอร์ ผู้เล่นจะเห็นสินค้าหลังซื้อทันที สำหรับการซื้อสินค้าผ่านผู้เล่นคนอื่นเมื่อซื้อแล้วจะมีข้อความแสดงว่าสำเร็จหรือไม่ ในกรณีที่สำเร็จระบบเกมจะบันทึกสินค้าในบ้านของผู้เล่น ผู้เล่นสามารถดูได้โดยการเข้าระบบเพื่อทำการเข้าบ้านของตัวเอง

เงื่อนไขสำหรับผู้เล่นที่ต้องการซื้อสินค้ากับเซิร์ฟเวอร์

1. ผู้เล่นสามารถซื้อสินค้ากับเซิร์ฟเวอร์ได้ภายในเกมเท่านั้น
2. ผู้เล่นจะต้องอยู่บ้านของตัวเองเท่านั้น และต้องไม่เป็นโฮสต์
3. ผู้เล่นไม่สามารถซื้อสินค้าซ้ำกับสินค้าที่มีอยู่แล้วได้
4. เกมจะทำการวางสินค้าให้อัดโนมิต ในตำแหน่งที่กำหนดไว้แล้วภายในบ้านของผู้เล่น 6 จุด

ลำดับการซื้อสินค้ากับเซิร์ฟเวอร์

1. ผู้เล่นทำการเข้าระบบเกม
2. กดปุ่ม T จะปรากฏหน้าจอสินค้าที่เซิร์ฟเวอร์มีขายในขณะนั้น ในกรณีที่ไม่มีสินค้าขายจะมีข้อความเตือนว่าเซิร์ฟเวอร์ไม่มีสินค้าสำหรับขาย
3. ทำการเลือกสินค้าที่ต้องการและกดปุ่ม Enter จะขึ้นหน้าต่างแสดงรายละเอียดสินค้าที่เลือก
4. เลือก Yes เมื่อต้องการซื้อสินค้า หรือเลือก No เมื่อต้องการยกเลิกการซื้อสินค้า

เงื่อนไขการซื้อสินค้ากับผู้เล่นที่เป็นโฮสต์

1. ผู้เล่นสามารถซื้อสินค้าจากภายในเกมได้เท่านั้น
2. ผู้เล่นจะต้องทำการร่วมโฮสต์กับผู้เล่นที่เป็นเจ้าของโฮสต์ที่ต้องการซื้อสินค้า
3. ผู้เล่นไม่สามารถซื้อสินค้าซ้ำกับสินค้าที่มีอยู่แล้วได้
4. ผู้เล่นไม่สามารถกำหนดตำแหน่งว่าสินค้าได้ ระบบเกมจะเป็นผู้กำหนดให้อัดโนมิต

ลำดับการซื้อสินค้ากับผู้เล่นที่เป็นโฮสต์

1. ผู้เล่นทำการเข้าระบบเกม
2. ทำการร่วมโฮสต์ที่ต้องการซื้อสินค้า
3. ในขณะที่อยู่ในบ้านของโฮสต์กดปุ่ม T จะขึ้นรายละเอียดสินค้าทั้งหมดภายในบ้าน โดยสินค้าที่สามารถซื้อได้จะมีบอกไว้ว่า CanTrade เป็นตัวอักษรสีเขียว
4. เลือกสินค้าที่ต้องการซื้อจากรายการ และกดปุ่ม Enter จะขึ้นหน้าจอแสดงรายละเอียดสินค้า
5. เลือก Yes เมื่อต้องการซื้อสินค้า หรือเลือก No เมื่อต้องการยกเลิกการซื้อสินค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.7 การตรวจสอบความถูกต้องของข้อมูลสินค้าของโฮสต์

ผู้เล่นสามารถทำการตรวจสอบข้อมูลสินค้าของโฮสต์กับเซิร์ฟเวอร์ได้ โดยทำการร้องขอข้อมูลของโฮสต์ที่ผู้เล่นเข้าร่วมอยู่ในขณะนั้น โดยการส่งการร้องขอนี้ผ่านโฮสต์ไปยังเซิร์ฟเวอร์ โดยสามารถทำได้ด้วยการกดปุ่ม V เมื่อผู้เล่นได้เข้าร่วมกับโฮสต์ที่ต้องการตรวจสอบแล้ว

4.1.8 การพูดคุย

ผู้เล่นสามารถทำการพูดคุยกับผู้เล่นคนอื่นได้ โดยเกมมีการพูดคุย 4 รูปแบบ เป็นการพูดคุยแบบมีกำหนดครีมีความห่างระหว่างตัวละครผู้เล่น 3 แบบ และเป็นการพูดคุยแบบไม่มีกำหนดครีมีความห่าง 1 แบบ ซึ่งมีรายละเอียดดังนี้

1. การกระชับ มีรัศมีการส่งข้อความถึงตัวละครผู้เล่นคนอื่นในแนวรัศมีวงกลมรอบตัวละครผู้พูดโดยมีรัศมี 50 หน่วยในระบบ 3 มิติ สำหรับตัวละครผู้เล่นที่ไม่ได้อยู่ในรัศมีของตัวละครของผู้พูดจะไม่สามารถรับข้อความ
2. การพูดธรรมดา จะมีรัศมีกว้างกว่าแบบกระชับ โดยมีรัศมีที่ 200 หน่วยในระบบ 3 มิติ ซึ่งผู้เล่นที่อยู่ในรัศมีรอบตัวผู้พูดจะได้รับข้อความ ส่วนผู้เล่นที่อยู่ไกลเกินขอบเขตรัศมีจะไม่สามารถรับข้อความ
3. การตะโกน จะมีรัศมีกว้างที่สุดครอบคลุมทั้งบ้าน ซึ่งมีขนาด 500 หน่วยในระบบ 3 มิติ ผู้เล่นทุกคนจะได้รับข้อความจากผู้เล่นที่พูดแบบตะโกน
4. การพูดแบบไม่กำหนดระยะ ผู้เล่นทุกคนจะได้รับข้อความเหมือนกันหมดไม่ว่าจะอยู่ส่วนไหนของบ้านก็ตาม

นอกจากการพูดคุยที่มีการตรวจสอบระยะตามหัวข้อ 1 – 3 แล้ว การพูดแบบกำหนดระยะ จะมีการตรวจสอบการหันหน้าของตัวละครของผู้พูดด้วย ถ้าตัวละครของผู้พูดหันหน้าไปหาตัวละครของผู้เล่นคนใด ผู้เล่นคนนั้นจะทราบได้ว่ามีคนพูดด้วยโดยข้อความที่ส่งไปจะมีชื่อของผู้ส่งกำกับไว้ด้วย

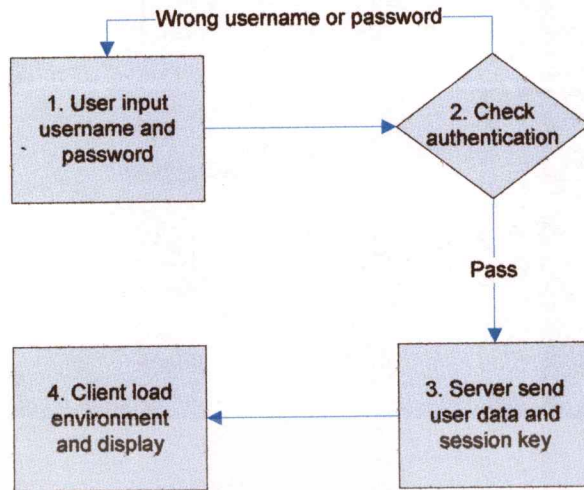
4.2 การออกแบบการทำงานของโปรแกรม

โครงงานพิเศษนี้เป็นโครงงานที่แสดงการนำสถาปัตยกรรมเน็ตเวิร์ค 2 แบบมาใช้ร่วมกัน ในหัวข้อนี้จะแสดงรายละเอียดของการทำงานแต่ละส่วน และการนำสถาปัตยกรรมมาใช้ในส่วนต่างของเกมที่นำมาใช้เป็นตัวศึกษา โดยมีการนำเสนอด้วยแผนผังการดำเนินงาน(Flowchart)

4.2.1 การเข้าใช้ระบบ

การเข้าระบบจะเป็นส่วนแรกที่ผู้เล่นจะต้องผ่านเพื่อเข้าใช้งานโปรแกรม โดยการทำงานในส่วนนี้ทั้งหมดจะใช้สถาปัตยกรรมแบบไคลเอนท์และเซิร์ฟเวอร์ โดยมีลำดับการทำงานแสดงดังรูป

เอกสาร 4.1 เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



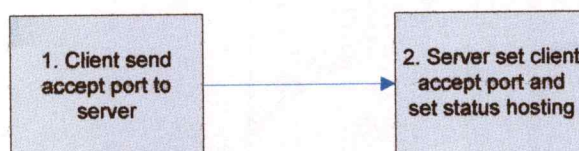
รูปที่ 4.1 แสดงการดำเนินการในขั้นตอนเข้าใช้ระบบ

รายละเอียดการทำงาน

1. ผู้เล่นจะทำการใส่ชื่อบัญชีระบบ(username) และรหัสเข้าใช้ระบบ(password) และส่งให้เซิร์ฟเวอร์
2. เซิร์ฟเวอร์ตรวจสอบข้อมูลของผู้เล่นกับฐานข้อมูล ถ้าชื่อบัญชีและรหัสผ่านไม่เข้าคู่กัน เซิร์ฟเวอร์จะส่งข้อผิดพลาดกลับมาให้ แต่ถ้าชื่อบัญชีและรหัสผ่านเข้าคู่กันจะส่งข้อมูลกลับไปให้ไคลเอนท์
3. เมื่อผ่านการตรวจสอบข้อมูลเซิร์ฟเวอร์จะทำการสร้างคีย์แบบสมมาตรที่ใช้สำหรับเข้ารหัสข้อมูล(encryption) และส่งกลับมาพร้อมกับข้อมูลของผู้เล่นตามชื่อบัญชี คีย์สมมาตรที่เซิร์ฟเวอร์สร้างขึ้นใช้เพื่อเอาไว้ใช้ติดต่อกับเซิร์ฟเวอร์ในภายหลัง เมื่อผู้เล่นไปเข้าร่วมกับโฮสต์ของผู้เล่นคนอื่นแล้ว
4. ไคลเอนท์ที่ได้รับข้อมูลแล้วจะทำการสร้างสภาพแวดล้อมของผู้เล่นตามข้อมูลที่ได้รับมาจากเซิร์ฟเวอร์

4.2.2 การสร้างโฮสต์

ผู้เล่นสามารถสร้างโฮสต์ได้หลังจากเข้าสู่ระบบและอยู่ที่บ้านของตัวเอง การทำงานใช้สถาปัตยกรรมเน็ตเวิร์กแบบไคลเอนท์และเซิร์ฟเวอร์ มีลำดับการทำงานแสดงดังรูปที่ 4.2



รูปที่ 4.2 แสดงการสร้างโฮสต์ของผู้เล่น

รายละเอียดการทำงาน

1. ไคลเอนต์ทำการสร้างช่องทางการเชื่อมต่อ เมื่อสำเร็จจะทำการส่งหมายเลขพอร์ตที่เปิดการเชื่อมต่อไปที่เซิร์ฟเวอร์
2. เซิร์ฟเวอร์บันทึกข้อมูลหมายเลขพอร์ตที่ไคลเอนต์เปิดและทำการเปลี่ยนสเตตัสของไคลเอนต์เป็นโฮสต์

4.2.3 การเข้าร่วมโฮสต์ของผู้เล่นคนอื่น

ผู้เล่นสามารถทำการเข้าร่วมโฮสต์ของผู้เล่นคนอื่นได้โดยทำการขอลิสต์ผู้ที่ เป็นโฮสต์จากเซิร์ฟเวอร์ การทำงานเป็นรูปแบบสถาปัตยกรรมเน็ตเวิร์คแบบผสม มีลำดับการทำงานแสดงดังรูปที่

4.3



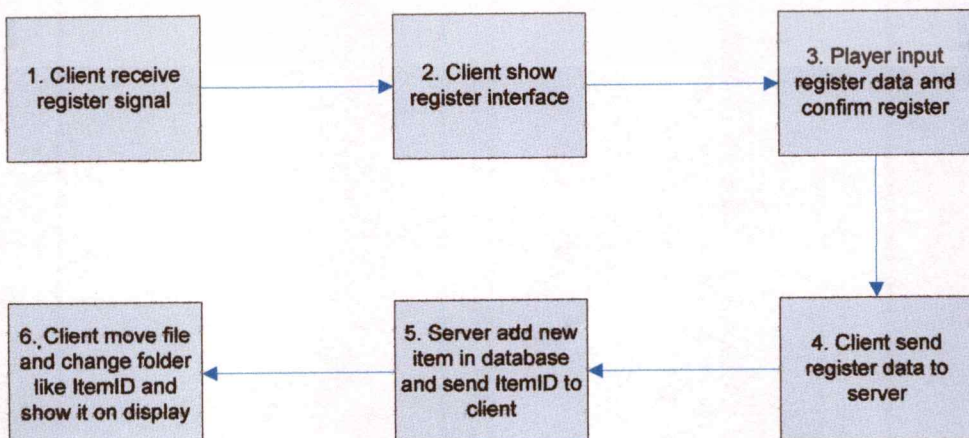
รูปที่ 4.3 แสดงการเข้าร่วมโฮสต์ของผู้เล่น

รายละเอียดการทำงาน

1. ไคลเอนต์ทำการร้องขอรายละเอียดโฮสต์จากเซิร์ฟเวอร์
2. เซิร์ฟเวอร์ส่งข้อมูลโฮสต์ที่ออนไลน์อยู่กลับไปให้ไคลเอนต์
3. ไคลเอนต์ทำการเชื่อมต่อเข้าไปยังโฮสต์ที่ผู้เล่นเลือก

4.2.4 การลงทะเบียนสินค้าของผู้เล่น

ผู้เล่นสามารถทำการลงทะเบียนสินค้าที่สร้างขึ้นกับเซิร์ฟเวอร์ได้ การทำงานจะใช้สถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนต์และเซิร์ฟเวอร์ มีลำดับการทำงานแสดงดังรูปที่ 4.4



รูปที่ 4.4 แสดงการลงทะเบียนสินค้าที่ผู้เล่นสร้างขึ้น

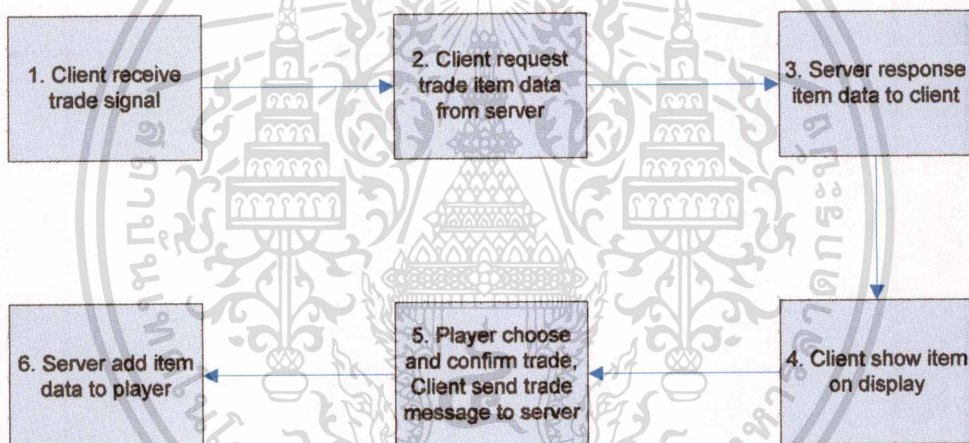
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดการทำงาน

1. ผู้เล่นทำการกดปุ่มเมื่อต้องการลงทะเบียนสินค้า
2. ไคลเอนท์แสดงหน้าจอใส่ข้อมูลสินค้า
3. ผู้เล่นกรอกข้อมูลและยืนยันข้อมูลการลงทะเบียน
4. ไคลเอนท์ส่งข้อมูลสินค้าไปให้เซิร์ฟเวอร์
5. เซิร์ฟเวอร์บันทึกข้อมูลสินค้าและส่งหมายเลขสินค้ากลับ
6. ไคลเอนท์ทำการย้ายไฟล์ข้อมูลสินค้าและเปลี่ยนชื่อไดเรกทอรีตามหมายเลขสินค้า ทำการโหลดและแสดงผลบนหน้าจอ

4.2.5 การซื้อสินค้าระหว่างผู้เล่นกับเซิร์ฟเวอร์

ผู้เล่นสามารถซื้อสินค้าตกแต่งบ้านกับเซิร์ฟเวอร์ได้ การทำงานจะใช้สถาปัตยกรรมเน็ตเวิร์คแบบไคลเอนท์และเซิร์ฟเวอร์ มีลำดับการทำงานแสดงดังรูปที่ 4.5



รูปที่ 4.5 แสดงการซื้อสินค้าระหว่างผู้เล่นกับเซิร์ฟเวอร์

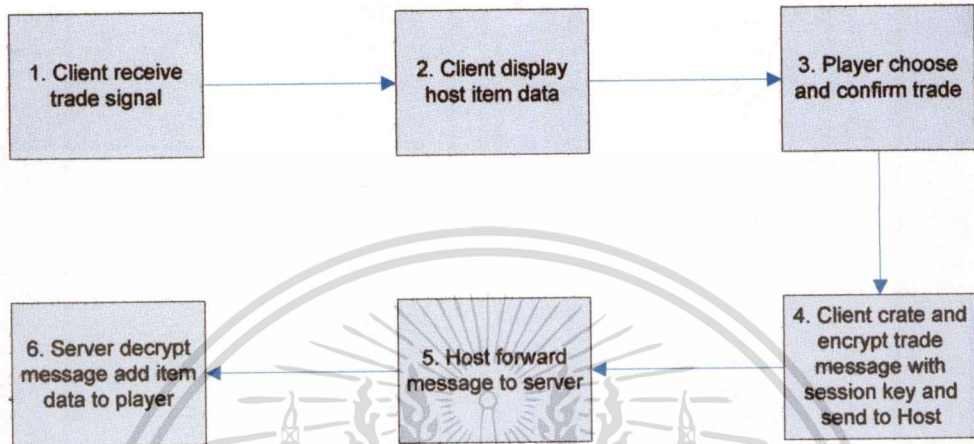
รายละเอียดการทำงาน

1. ผู้เล่นทำการกดปุ่มเพื่อซื้อสินค้ากับเซิร์ฟเวอร์
2. ไคลเอนท์ทำการขอข้อมูลสินค้าที่เซิร์ฟเวอร์ขาย
3. เซิร์ฟเวอร์ส่งข้อมูลสินค้ากลับมาที่ไคลเอนท์
4. ไคลเอนท์แสดงหน้าจอรายละเอียดสินค้า
5. ผู้เล่นเลือกสินค้าและยืนยันการซื้อสินค้า ไคลเอนท์ทำการส่งข้อมูลการซื้อสินค้าไปเซิร์ฟเวอร์
6. เซิร์ฟเวอร์ทำการตรวจสอบรายการซื้อและบันทึกการซื้อสินค้าลงฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.6 การซื้อสินค้าระหว่างโฮสต์และเพียร์

ผู้เล่นสามารถทำการซื้อสินค้ากับผู้เล่นด้วยกันได้โดยผู้ขายจะต้องเป็น โฮสต์ในขณะนั้น โดย การซื้อสินค้าจะต้องส่งการซื้อขายไปให้เซิร์ฟเวอร์ทำการตรวจสอบ การทำงานจะใช้สถาปัตยกรรม เน็ตเวิร์คแบบผสม มีลำดับการทำงานแสดงดังรูปที่ 4.6



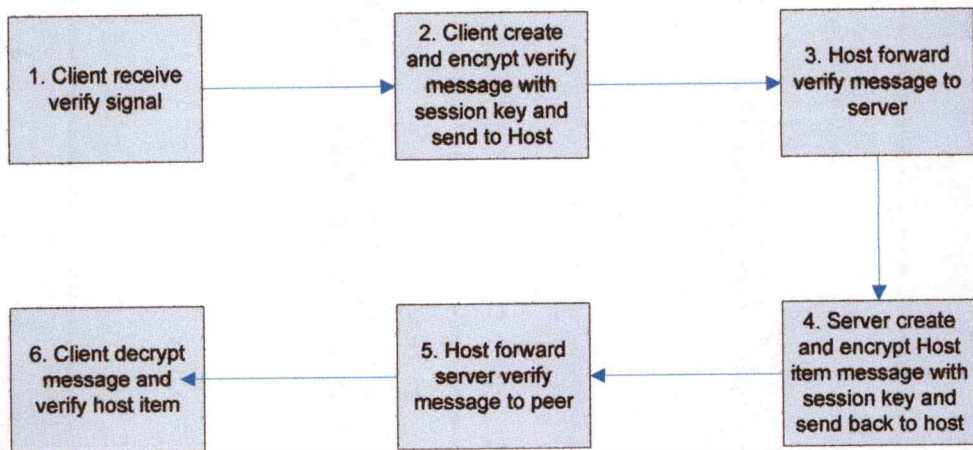
รูปที่ 4.6 แสดงการซื้อสินค้าระหว่างโฮสต์และเพียร์

รายละเอียดการทำงาน

1. ผู้เล่นทำการกดปุ่มซื้อสินค้า ในขณะที่เป็นเพียร์ที่เข้าไปโฮสต์ของผู้เล่นคนอื่น
2. โคลเอนท์แสดงข้อมูลสินค้าของโฮสต์นั้นๆ
3. ผู้เล่นทำการเลือกสินค้าและยืนยันการซื้อสินค้า
4. โคลเอนท์ทำการสร้างข้อความซื้อสินค้า และเข้ารหัสสินค้าด้วยคีย์สมมาตรที่ได้รับมาจากเซิร์ฟเวอร์ในขั้นตอนเข้าระบบ และส่งให้โฮสต์
5. โฮสต์ทำการส่งต่อข้อความไปให้เซิร์ฟเวอร์
6. เซิร์ฟเวอร์ถอดรหัส, ตรวจสอบสิทธิการซื้อขายสินค้า และบันทึกลงฐานข้อมูล

4.2.7 การตรวจสอบความถูกต้องของข้อมูลโฮสต์

ผู้เล่นที่เป็นเพียร์มีสิทธิ์ที่จะตรวจสอบความถูกต้องของข้อมูลสินค้าของโฮสต์ได้ เพื่อป้องกันการดัดแปลงข้อมูล โดยการร้องขอข้อมูลของโฮสต์จากเซิร์ฟเวอร์มาเปรียบเทียบข้อมูลที่มีอยู่ มีลำดับการทำงานแสดงดังรูปที่ 4.7



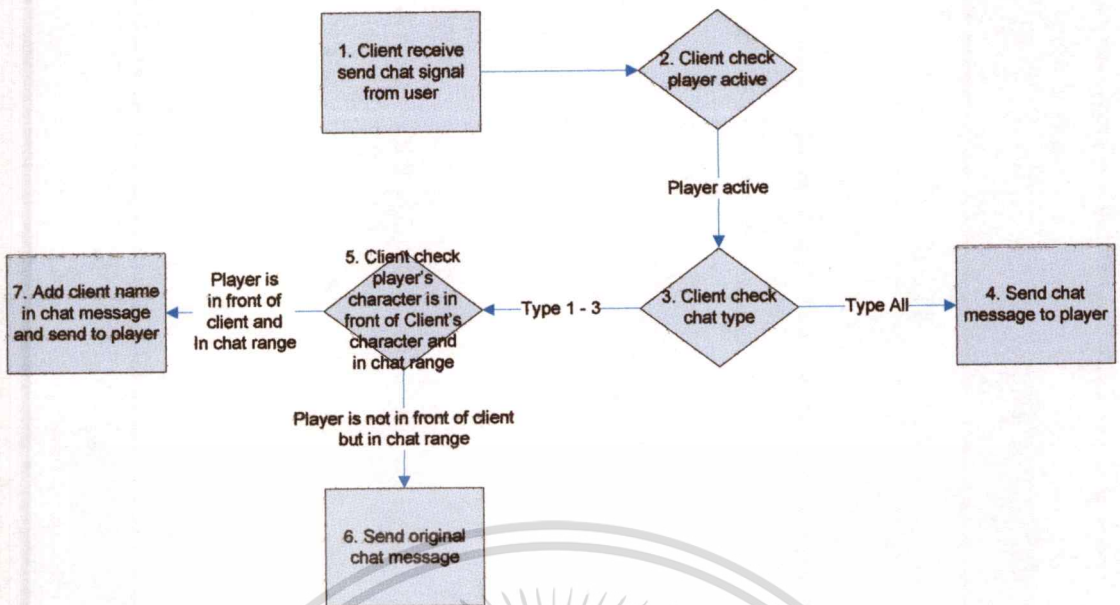
รูปที่ 4.7 แสดงการตรวจสอบความถูกต้องของข้อมูลโฮสต์

รายละเอียดการทำงาน

1. ผู้เล่นทำการกดปุ่มตรวจสอบสินค้าของโฮสต์
2. ไคลเอนท์ทำการสร้างและเข้ารหัสข้อความตรวจสอบด้วยคีย์สมมาตรและส่งไปให้โฮสต์
3. โฮสต์ทำการส่งข้อความตรวจสอบต่อไปยังเซิร์ฟเวอร์
4. เซิร์ฟเวอร์ถอดรหัสและทราบว่าไคลเอนท์ต้องการตรวจสอบข้อมูลโฮสต์ จะทำการค้นหาข้อมูลและสร้างข้อความและเข้ารหัสด้วยคีย์สมมาตรของไคลเอนท์ และส่งกลับไปที่โฮสต์
5. โฮสต์ส่งต่อข้อความตรวจสอบกลับไปที่ไคลเอนท์
6. ไคลเอนท์ถอดรหัสด้วยคีย์สมมาตรและตรวจสอบข้อมูลสินค้าของโฮสต์

4.2.8 การพูดคุยระหว่างผู้เล่น

ผู้เล่นสามารถทำการพูดคุยกับผู้เล่นคนอื่นได้ มีรูปแบบทั้งหมด 4 รูปแบบและมีการตรวจสอบการหันหน้าของตัวละครของผู้เล่น โดยยึดหลักการที่ว่าเมื่อตัวละครหันหน้าไปหาใคร หมายถึงต้องการพูดคุยกับตัวละครนั้น การทำงานจะใช้สถาปัตยกรรมเน็ตเวิร์คแบบเพียร์-ทู-เพียร์ มีลำดับการทำงานแสดงดังรูปที่ 4.8



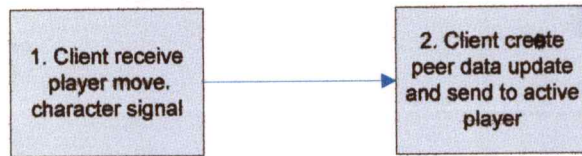
รูปที่ 4.8 แสดงการทำงานการพูดคุยระหว่างผู้เล่น

รายละเอียดการทำงาน

1. ผู้เล่นต้องการส่งข้อความหาผู้เล่นคนอื่น
2. ไคลเอนท์ทำการวนลูปรตรวจสอบผู้เล่นทั้งหมดยกเว้นตัวเอง หาผู้เล่นที่ยังอยู่ในกลุ่ม
3. ไคลเอนท์ทำการตรวจสอบชนิดการส่งข้อความที่ผู้เล่นเลือกส่ง
4. กรณีที่เป็นส่งทั้งหมดแบบไม่กำหนดขอบเขต ไคลเอนท์ทำการส่งข้อความตามที่ผู้ส่งพิมพ์
5. กรณีเป็นส่งแบบกำหนดขอบเขต ไคลเอนท์จะตรวจสอบขอบเขตของผู้เล่นที่จะส่ง และตรวจสอบว่าอยู่ด้านหน้าหรือไม่
6. กรณีตัวละครผู้เล่นที่จะส่งข้อความไปหาไม่ได้อยู่ด้านหน้าตัวละครผู้ส่ง แต่อยู่ในรัศมีการส่ง จะทำการส่งข้อความตามที่ผู้ส่งพิมพ์
7. กรณีตัวละครผู้เล่นที่จะส่งข้อความอยู่ด้านหน้าและอยู่ในรัศมีการส่งจะทำการเพิ่มชื่อผู้ส่งเข้าไปในข้อความ และทำการส่งไปให้ผู้เล่น

4.2.9 การอัปเดตข้อมูลตัวละครของผู้เล่น

ผู้เล่นสามารถทำการเคลื่อนที่ตัวละครของตัวเองได้ จึงต้องมีการอัปเดตข้อมูลของตัวละครให้ผู้เล่นคนอื่นได้ทราบตำแหน่งปัจจุบันของตัวละครที่ผู้เล่นบังคับอยู่ด้วย ซึ่งจะเป็นการกระจายข้อมูลให้กับผู้เล่นทุกคนที่อยู่ในกลุ่มเดียวกัน การทำงานจะใช้สถาปัตยกรรมเน็ตเวิร์กแบบเพียร์-ทู-เพียร์ มีลำดับการทำงานแสดงดังรูปที่ 4.9



รูปที่ 4.9 แสดงการอัปเดตข้อมูลตัวละครที่ผู้เล่นบังคับ

รายละเอียดการทำงาน

1. ผู้เล่นทำการเคลื่อนที่ตัวละคร
2. ไคลเอนต์ทำการสร้างข้อความอัปเดตตัวละคร และส่งไปยังผู้เล่นที่อยู่ในกลุ่มทั้งหมด

4.3 การออกแบบโปรโตคอลที่ใช้พัฒนาโปรแกรมในสถาปัตยกรรมเน็ตเวิร์กแบบผสม

โปรโตคอลในระดับแอปพลิเคชันเลเยอร์ที่ใช้ในโครงงานพิเศษนี้แบ่งส่วนการใช้งานได้เป็น 3 ส่วนตามสถาปัตยกรรมเน็ตเวิร์กที่ใช้พัฒนา

4.3.1 โปรโตคอลที่ใช้สำหรับสถาปัตยกรรมเน็ตเวิร์กแบบไคลเอนต์และเซิร์ฟเวอร์

เป็นโปรโตคอลที่ใช้ติดต่อระหว่างไคลเอนต์และเซิร์ฟเวอร์โดยในโครงงานพิเศษนี้โปรโตคอลที่ใช้จะเกี่ยวข้องกับการดำเนินการต่างๆ ที่ร้องขอไปที่เซิร์ฟเวอร์ เช่น การเข้าใช้ระบบ การลงทะเบียนสินค้าใหม่ และการซื้อสินค้ากับเซิร์ฟเวอร์

รายละเอียดของโปรโตคอล

1. MessageHeader

เป็นโปรโตคอลที่เก็บชนิดของโปรโตคอลและขนาดของโปรโตคอล โดยมีรายละเอียดดังนี้

- dwMessageType เก็บชนิดของโปรโตคอล
- dwMessageSize เก็บขนาดของโปรโตคอล

2. MessageLogin

เป็นโปรโตคอลที่ใช้สำหรับเข้าระบบ โดยมีรายละเอียดดังนี้

- msgHeader เก็บ MessageHeader
- Username เก็บชื่อบัญชีการเข้าใช้ระบบ
- Password เก็บรหัสการเข้าใช้ระบบ

3. MessageResult

เป็นโปรโตคอลที่ส่งผลการเข้าใช้ระบบ, ข้อมูลผู้ใช้ และคีย์สมมาตรที่ใช้เข้ารหัส กลับไปที่ไคลเอนต์ โดยมีรายละเอียดดังนี้

- msgHeader เก็บ MessageHeader

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Result เก็บผลการเข้าใช้ระบบซึ่งมี 3 แบบ คือ SUCCEED, WRONG และ ALREADY ONLINE
- UserID เก็บหมายเลขอ้างอิงผู้เข้าใช้ระบบ
- CountUserItem เก็บจำนวนสินค้าที่ผู้เล่นระบบมีอยู่ในปัจจุบัน
- SymmetricKey เก็บคีย์สมมาตรของผู้เล่น
- SymmetricIV เก็บข้อมูลที่ใช้ร่วมกับคีย์สมมาตรเพื่อใช้เข้ารหัส

4. MessageError

เป็นโปรโตคอลที่ใช้บอกความผิดพลาดต่างๆ โดยมีรายละเอียดดังนี้

- msgHeader เก็บ MessageHeader
- cError เก็บข้อความผิดพลาด
- iError เก็บหมายเลขผิดพลาด

5. MessageSync

เป็นโปรโตคอลสำหรับส่งข้อมูลสินค้าของผู้เล่น โดยมีรายละเอียดดังนี้

- msgHeader เก็บ MessageHeader
- CountUserItem เก็บจำนวนสินค้าที่ผู้เล่นมี
- Item เก็บรายละเอียดสินค้าที่ผู้เล่นมี

6. MessageStatus

เป็นโปรโตคอลบอกสถานะต่างๆ ของไคลเอนท์ โดยมีรายละเอียดดังนี้

- msgHeader เก็บ MessageHeader
- UserID เก็บหมายเลขอ้างอิงตัวผู้เล่น
- StatusCode เก็บหมายเลขสเตตัสที่ต้องการแจ้งให้เซิร์ฟเวอร์

7. MessageServerItemData

เป็นโปรโตคอลที่ใช้ส่งข้อมูลสินค้าของเซิร์ฟเวอร์ โดยมีรายละเอียดดังนี้

- msgHeader เก็บ MessageHeader
- TotalItemSet เก็บจำนวนชุดของสินค้าทั้งหมดที่มี
- CurrentItemSet เก็บหมายเลขชุดสินค้าที่ใช้ส่ง
- TradeItemData เก็บข้อมูลสินค้า

8. MessageTradeItemServer

เป็นโปรโตคอลที่ทำการซื้อสินค้าจากเซิร์ฟเวอร์ โดยมีรายละเอียดดังนี้

- msgHeader เก็บ MessageHeader

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- UserID เก็บหมายเลขอ้างอิงผู้เล่น
- ItemID เก็บหมายเลขสินค้าที่ต้องการซื้อ

9. MessageRegisterItem

เป็นโปรโตคอลที่ใช้ลงทะเบียนสินค้ากับเซิร์ฟเวอร์ โดยมีรายละเอียดดังนี้

- msgHeader เก็บ MessageHeader
- UserID เก็บหมายเลขอ้างอิงผู้เล่น
- ItemName เก็บชื่อสินค้าที่ต้องการลงทะเบียน
- PosY เก็บตำแหน่งสินค้าในแนวแกน Y ในระบบ 3 มิติ
- DegreeY เก็บค่าองศาของโมเดลสินค้าในแนวแกน Y ในระบบ 3 มิติ
- Scale เก็บขนาดของโมเดลสินค้า
- Price เก็บราคาของสินค้าที่ต้องการลงทะเบียน

10. MessageRegisterItemID

เป็นโปรโตคอลที่บอกหมายเลขสินค้าที่ได้ลงทะเบียนไป โดยมีรายละเอียดดังนี้

- msgHeader เก็บ MessageHeader
- ItemID เก็บหมายเลขสินค้าที่ได้ลงทะเบียนไป

11. MessageHostPort

เป็นโปรโตคอลที่บอกหมายเลขพอร์ตที่โฮสต์เปิดเพื่อรับการเชื่อมต่อ โดยมีรายละเอียดดังนี้

- msgHeader เก็บ MessageHeader
- Port เก็บหมายเลขพอร์ตที่โฮสต์เปิดเพื่อรับการเชื่อมต่อ

12. MessageHostList

เป็นโปรโตคอลที่เก็บรายละเอียดของโฮสต์ที่ออนไลน์อยู่ในปัจจุบัน โดยมีรายละเอียดดังนี้

- msgHeader เก็บ MessageHeader
- MaxHost เก็บจำนวนโฮสต์ที่ออนไลน์อยู่
- Host เก็บข้อมูลของโฮสต์ที่ออนไลน์อยู่

4.3.2 โปรโตคอลที่ใช้สำหรับสถาปัตยกรรมเน็ตเวิร์กแบบเพียร์-ทู-เพียร์

เป็นโปรโตคอลที่ใช้ติดต่อระหว่างโฮสต์กับเพียร์โดยในโครงงานพิเศษนี้โปรโตคอลที่ใช้จะเกี่ยวข้องกับการดำเนินการต่างๆ เมื่อผู้เล่นทำการสร้างโฮสต์และมีผู้เล่นคนอื่นมาร่วมเข้าโฮสต์ด้วย เช่น การพูดคุยภายในบ้าน การอัพเดทสถานะของตัวละครที่เคลื่อนที่

รายละเอียดของโปรโตคอล

1. MessageSync

เป็นโปรโตคอลสำหรับส่งข้อมูลสินค้าของผู้โฮสต์ โดยมีรายละเอียดดังนี้

- msgHeader เก็บ MessageHeader
- CountUserItem เก็บจำนวนสินค้าที่ผู้เล่นมี
- Item เก็บรายละเอียดสินค้าที่ผู้เล่นมี

2. MessageNewPeer

เป็นโปรโตคอลที่ใช้ออกเฝ้าระวังว่ามีเพียร์ใหม่เข้ามาในกลุ่ม โดยมีรายละเอียดดังนี้

- msgHeader เก็บ MessageHeader
- ip เก็บหมายเลข ip ที่ใช้ติดต่อ
- Port เก็บหมายเลขพอร์ตที่เปิดการเชื่อมต่ออยู่

3. MessagePeerReady

เป็นโปรโตคอลที่ใช้ออกสถานะพร้อมของเพียร์ที่เข้าใหม่ โดยมีรายละเอียดดังนี้

- msgHeader เก็บ MessageHeader
- PlayerID เก็บหมายเลขอ้างอิงเพียร์ของผู้เล่นที่ติดต่อเข้ามา
- AcceptPort เก็บหมายเลขพอร์ตที่เปิดให้ทำการเชื่อมต่อเข้ามา

4. MessagePeerData

เป็นโปรโตคอลที่ใช้ส่งข้อมูลอ็อปเทคของตัวละครของเพียร์ในกลุ่ม โดยมีรายละเอียดดังนี้

- msgHeader เก็บ MessageHeader
- PlayerID เก็บหมายเลขอ้างอิงเพียร์ของผู้เล่น
- PlayerName เก็บชื่อของผู้เล่น
- vecCurPos เก็บตำแหน่งปัจจุบันของตัวละครของผู้เล่น
- iFrame เก็บท่าทางปัจจุบันของตัวละครของผู้เล่น
- fDegrees เก็บองศาที่ตัวละครของผู้เล่นหันหน้าอยู่

5. MessageChat

เป็นโปรโตคอลที่ใช้ส่งข้อความสื่อสารระหว่างเพียร์ในกลุ่ม โดยมีรายละเอียดดังนี้

- msgHeader เก็บ MessageHeader
- ChatData เก็บข้อความที่ใช้ส่ง

4.3.3 โปรโตคอลที่ใช้สำหรับสถาปัตยกรรมเน็ตเวิร์คแบบผสม

เป็นโปรโตคอลที่ใช้ติดต่อระหว่างเซิร์ฟเวอร์กับเพียร์ที่อยู่ในโฮสต์ของผู้เล่นคนอื่น ซึ่ง

โปรโตคอลจะเกี่ยวข้องกับการดำเนินการต่างๆ ที่เพียร์ต้องการให้เซิร์ฟเวอร์ช่วยจัดการ โดยการส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพียร์จะทำการส่งข้อมูลผ่านโสตค์อีกต่อหนึ่ง ในโครงงานพิเศษนี้การดำเนินการที่เกี่ยวข้องกับโปรโตคอลนี้ เช่น การซื้อสินค้าระหว่างโสตค์และเพียร์ และการตรวจสอบข้อมูลสินค้าภายในโสตค์

รายละเอียดของโปรโตคอล

1. MessageRequestServer

เป็นโปรโตคอลที่เพียร์ทำการร้องขอให้เซิร์ฟเวอร์ช่วยตรวจสอบ หรือทำงานบางอย่าง โดยมีรายละเอียดดังนี้

- msgHeader เก็บ MessageHeader
- UserRequest เก็บหมายเลขของเพียร์ที่ร้องขอการทำงานของเซิร์ฟเวอร์
- RequestType เก็บค่าที่เพียร์ต้องการร้องขอให้เซิร์ฟเวอร์ทำ

2. MessageTradeItemHost

เป็นโปรโตคอลที่เพียร์ทำการซื้อสินค้าจากโสตค์ โดยส่งไปให้เซิร์ฟเวอร์ทำการตรวจสอบและบันทึกข้อมูล โดยมีรายละเอียดดังนี้

- msgHeader เก็บ MessageHeader
- ItemID เก็บหมายเลขสินค้าที่เพียร์ต้องการซื้อสินค้ากับโสตค์

3. MessageServerResponse

เป็นโปรโตคอลที่เซิร์ฟเวอร์ทำการตอบกลับไปหาเพียร์ในกลุ่มโสตค์ โดยมีรายละเอียดดังนี้

- msgHeader เก็บ MessageHeader
- ResponseUser เก็บหมายเลขเพียร์ที่เซิร์ฟเวอร์ต้องการตอบกลับ
- ResponseType เก็บค่าที่เซิร์ฟเวอร์ต้องการตอบกลับไปหาเพียร์

4. MessageVerifyHost

เป็นโปรโตคอลที่เซิร์ฟเวอร์ส่งข้อมูลของสินค้าของโสตค์ที่เพียร์ร้องขอเพื่อตรวจสอบ โดยมีรายละเอียดดังนี้

- msgServerResponse เก็บ MessageServerResponse
- CountHostItem เก็บจำนวนสินค้าที่โสตค์มีทั้งหมด
- Item เก็บรายละเอียดสินค้าที่โสตค์มี

5. MessageResultTradeItemHost

เป็นโปรโตคอลที่เซิร์ฟเวอร์ตอบกลับผลการซื้อสินค้าของเพียร์กับโสตค์ โดยมีรายละเอียดดังนี้

- msgServerResponse เก็บ MessageServerResponse
- Result เก็บผลการซื้อสินค้าของเพียร์และโสตค์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 การออกแบบฐานข้อมูล

ในการออกแบบฐานข้อมูลจะมีตารางอยู่ทั้งหมด 3 ตาราง คือ ตาราง Login, ตาราง Item และ ตาราง UserItem โดยตาราง login จะทำการเก็บข้อมูลที่ใช้เข้าระบบ และเก็บสถานะการออนไลน์ และเก็บจำนวนเงินที่ผู้เล่นมีในขณะนั้น และเก็บข้อมูลกฎแถมมาคร โดยเซิร์ฟเวอร์ได้ทำการสร้างให้ตอนที่ผู้เล่นทำการเข้าระบบ ตาราง Item จะเก็บข้อมูลของสินค้าในระบบว่ามีสินค้าอะไรบ้าง ตาราง UserItem จะเก็บรายการสินค้าที่ผู้เล่นได้ทำการซื้อไว้หรือเป็นเจ้าของ โดยรายละเอียดของข้อมูลในตารางมีดังนี้

ตารางที่ 4.1 ตาราง Login

ชื่อฟิลด์ข้อมูล	รายละเอียด	รูปแบบการ จัดเก็บ	ตัวอย่าง รูปแบบ	ค่าของข้อมูล	ข้อมูลเพิ่มเติม
ID	เก็บหมายเลขสมาชิก	Integer	1234	N/A	เป็นคีย์หลัก
Username	ชื่อสมาชิก	Varchar(20)	xxxxxxxx	N/A	ข้อมูลห้ามซ้ำ
Password	รหัสการเข้าใช้ระบบ	Varchar(20)	xxxxxxxx	N/A	
Money	จำนวนเงินที่มี	Integer	1234	N/A	
IsOnline	สถานะการออนไลน์	Char(1)	x	'Y', 'N'	
SymKey	เก็บกฎแถมแบบสมมาตร	Varchar(50)	xxxxxxxx	N/A	
SymIV	เก็บค่าที่ใช้ร่วมกับ SymKey เพื่อเข้ารหัสข้อมูล	Varchar(50)	xxxxxxxx	N/A	

ตารางที่ 4.2 ตาราง Item

ชื่อฟิลด์ข้อมูล	รายละเอียด	รูปแบบการ จัดเก็บ	ตัวอย่าง รูปแบบ	ค่าของข้อมูล	ข้อมูลเพิ่มเติม
ItemID	เก็บหมายเลขสินค้า	Integer	1234	N/A	เป็นคีย์หลัก

ตารางที่ 4.2 (ต่อ)

ชื่อฟิลด์ข้อมูล	รายละเอียด	รูปแบบการจัดเก็บ	ตัวอย่างรูปแบบ	ค่าของข้อมูล	ข้อมูลเพิ่มเติม
OwnerID	เก็บหมายเลขเจ้าของสินค้า	Integer	1234	N/A	
ItemName	เก็บชื่อสินค้า	Varchar(100)	xxxxxxxx	N/A	
Price	เก็บราคาสินค้า	Integer	1234	N/A	
PosY	เก็บตำแหน่งแกน Y ของสินค้า	Float	123.45	N/A	
DegreeY	เก็บองศาที่วางสินค้า	Float	123.45	N/A	
Scale	เก็บขนาดของสินค้า	Float	123.45	N/A	

ตารางที่ 4.3 ตาราง UserItem

ชื่อฟิลด์ข้อมูล	รายละเอียด	รูปแบบการจัดเก็บ	ตัวอย่างรูปแบบ	ค่าของข้อมูล	ข้อมูลเพิ่มเติม
ID	เก็บหมายเลขสมาชิก	Integer	1234	N/A	เป็นคีย์หลัก
ItemID	เก็บหมายเลขสินค้า	Integer	xxxxxxxx	N/A	เป็นคีย์หลัก

4.5 ฟังก์ชันและรายละเอียดที่ใช้พัฒนา

การทำงานภายในเกมแบ่งได้เป็น 4 ส่วน คือ ส่วนที่เป็นการเชื่อมต่อผ่านเน็ตเวิร์ค ส่วนที่เป็นการติดต่อกับโปรโตคอล ส่วนที่เป็นการเข้ารหัสข้อมูล และส่วนที่ใช้เพื่อแสดงผล

4.5.1 ฟังก์ชันที่ใช้ทำการเชื่อมต่อผ่านเน็ตเวิร์ค

ประกอบด้วยฟังก์ชันทางฝั่งเซิร์ฟเวอร์ และฟังก์ชันทางฝั่งไคลเอนท์ มีรายละเอียดดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.1.1 ฟังก์ชันทางฝั่งเซิร์ฟเวอร์

เนื่องจากเซิร์ฟเวอร์ได้พัฒนาโดยใน Visual C++ .Net 2003 และใช้ MFC ในการพัฒนา ซึ่งใน MFC มีคลาสในการติดต่อสื่อสารผ่าน Socket อยู่แล้ว จึงได้ทำการ Inherit ฟังก์ชันต่างๆ เหล่านั้นมาใช้ โดยคลาสมีชื่อว่า CAsyncSocket และมีรายละเอียดการทำงานของฟังก์ชันดังนี้

ฟังก์ชัน Create

เป็นฟังก์ชันที่ใช้สร้างออบเจ็กต์ Socket โดยสามารถกำหนดหมายเลขพอร์ตที่ต้องการเปิดเพื่อให้รอการติดต่อกับไคลเอนท์

ฟังก์ชัน Listen

เป็นฟังก์ชันที่ทำการเปิดพอร์ตเพื่อรอรับการติดต่อจากไคลเอนท์ตามหมายเลขที่กำหนดในฟังก์ชัน Create

ฟังก์ชัน Accept

เป็นฟังก์ชันที่ทำการตอบรับการเชื่อมต่อกับไคลเอนท์ โดยมีฟังก์ชัน OnAccept เป็นฟังก์ชันที่ส่งสัญญาณว่ามีไคลเอนท์ทำการติดต่อเข้ามา และต้องมีออบเจ็กต์ Socket มารับข้อมูลของไคลเอนท์ที่เข้ามาติดต่อด้วย

ฟังก์ชัน OnClose

เป็นฟังก์ชันที่ส่งสัญญาณว่ามีไคลเอนท์ทำการตัดการเชื่อมต่อมาที่เซิร์ฟเวอร์ ซึ่งจะต้องทำการเพิ่มโค้ดส่วนการตรวจสอบเข้าไปว่าไคลเอนท์ตัวไหนที่ทำการตัดการเชื่อมต่อ

ฟังก์ชัน OnReceive

เป็นฟังก์ชันที่ส่งสัญญาณบอกว่ามีไคลเอนท์ทำการส่งเมสเสจมาที่เซิร์ฟเวอร์ ต้องทำการเพิ่มโค้ดส่วนของ การตอบกลับกลับไปหาไคลเอนท์ โดยมีฟังก์ชัน Recv เป็นฟังก์ชันในการรับเมสเสจ

ฟังก์ชัน Recv

เป็นฟังก์ชันที่ใช้รับเมสเสจจาก Socket ปลายทาง โดยข้อมูลที่รับมาจะอยู่ในรูปของอาร์เรย์ตัวอักษร

ฟังก์ชัน Send

เป็นฟังก์ชันที่ใช้ส่งเมสเสจไปยัง Socket ปลายทาง โดยข้อมูลที่ส่งต้องอยู่ในรูปของอาร์เรย์ตัวอักษร

4.5.1.2 ฟังก์ชันทางฝั่งไคลเอนท์และฟังก์ชันที่ใช้ติดต่อระหว่างเพียร์

เนื่องจากไคลเอนท์พัฒนาโดยใช้ Win32 ไม่มีคลาสที่ครอบการทำงาน ของ Socket จึงต้องพัฒนาตัว Socket เองโดยใช้ Window Socket API มาพัฒนา ซึ่งฟังก์ชันที่นำมาใช้พัฒนามีรายละเอียดดังนี้

เอกสารนี้จัดทำขึ้นเพื่อใช้ในการเรียนการสอน ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน WSASStartup

เป็นฟังก์ชันที่ใช้เรียกเพื่อบอกให้รู้ว่า จะทำการใช้ฟังก์ชันต่างๆ ของ Window Socket API

ฟังก์ชัน WSACreateEvent

เป็นฟังก์ชันที่ใช้สร้างตัวรับเหตุการณ์ของ socket เช่น การรับ การส่ง การติดต่

ฟังก์ชัน WSAEventSelect

เป็นฟังก์ชันที่ใช้กำหนดเหตุการณ์ของ socket ที่จะรับ

ฟังก์ชัน WSAWaitForMultipleEvents

เป็นฟังก์ชันที่ใช้รอการเกิดเหตุการณ์ที่กำหนดในฟังก์ชัน WSAEventSelect

ฟังก์ชัน WSAEnumNetworkEvents

เป็นฟังก์ชันที่ทำการแยกเหตุการณ์ว่าเกิดเหตุการณ์ใดที่เลือกไว้ในฟังก์ชัน WSAEventSelect

ฟังก์ชัน DisconnectPeer

เป็นฟังก์ชันที่ทำการตัดการเชื่อมต่อของเพียร์

ฟังก์ชัน BindAndListenP2PSocket

เป็นฟังก์ชันที่ใช้สำหรับกำหนดหมายเลขพอร์ตและ socket ที่ต้องการเปิดเพื่อรอรับการติดต่อจากเพียร์อื่นและทำการเปิดพอร์ตรอรับการติดต่อจากเพียร์

ฟังก์ชัน AcceptSocket

เป็นฟังก์ชันที่ทำการตอบรับการเชื่อมต่อโดยมี socket มารับข้อมูลของเพียร์ที่เข้ามาติดต่ด้วย

ฟังก์ชัน Connect

เป็นฟังก์ชันที่ใช้ทำการเชื่อมต่อไปยังเซิร์ฟเวอร์หรือโฮสต์ โดยต้องระบุหมายเลขพอร์ตที่ต้องการติดต่อ

ฟังก์ชัน Recv

เป็นฟังก์ชันที่ใช้สำหรับรับเมสเสจที่ส่งมาจากเซิร์ฟเวอร์ โดยข้อมูลอยู่ในรูปแบบอาร์เรย์ตัวอักษร

ฟังก์ชัน Send

เป็นฟังก์ชันที่ใช้สำหรับส่งเมสเสจไปยังเซิร์ฟเวอร์ โดยข้อมูลอยู่ในรูปแบบอาร์เรย์ตัวอักษร

4.5.2 ฟังก์ชันต้นแบบที่ใช้ติดต่อกับโปรโตคอล

ฟังก์ชันที่ใช้ติดต่อกับโปรโตคอลนี้จะเป็นฟังก์ชันพื้นฐานที่ใช้สื่อสารด้วยโปรโตคอลที่ได้ออกแบบไว้ ซึ่งในการใช้งานเมื่อนำไปประยุกต์ใช้จริงจะต้องมีการพัฒนาเพิ่มเติม โดยยึดจากการออกแบบโปรแกรมที่พัฒนา โดยฟังก์ชันมีรายละเอียดดังนี้

ฟังก์ชัน `int SendStatusMessage(SOCKET skSend, StatusCode code, int UserID)`

เป็นฟังก์ชันที่ใช้แจ้งสถานะของตัวไคลเอนท์ให้เซิร์ฟเวอร์ทราบ เช่น การแจ้งสถานะการสร้างโฮสต์ การขอข้อมูลโฮสต์ที่ออนไลน์ การขอข้อมูลต่างๆ ของเซิร์ฟเวอร์ เป็นต้น

ฟังก์ชัน `int SendLoginMessage(SOCKET skSend, char* Username, char* Password)`

เป็นฟังก์ชันที่ใช้ส่งข้อมูลการเข้าใช้ระบบจากไคลเอนท์ให้กับเซิร์ฟเวอร์ เพื่อตรวจสอบสิทธิการเข้าใช้ระบบ

ฟังก์ชัน `int SendHostPortMessage(SOCKET skSend, int Port)`

เป็นฟังก์ชันที่ใช้บอกสถานะของผู้ส่งที่ทำตัวเป็นโฮสต์ให้เซิร์ฟเวอร์ทราบว่าพร้อมรับการเชื่อมต่อจากเพียร์อื่นๆ แล้ว

ฟังก์ชัน `int SendPeerReadyMessage(SOCKET skSend, int PlayerID, int Port)`

เป็นฟังก์ชันที่ใช้ส่งสถานะของผู้ส่งที่เป็นเพียร์ให้กับโฮสต์ทราบว่าพร้อมที่จะเริ่มทำการรับส่งข้อมูลภายในกลุ่ม และพร้อมที่จะให้เพียร์อื่นๆ เข้ามาเชื่อมต่อด้วย

ฟังก์ชัน `int SendNewPeerMessage(SOCKET skSend, char* PeerIP, int Port)`

เป็นฟังก์ชันที่แจ้งสถานะให้เพียร์อื่นๆ ในกลุ่มรับทราบการเข้ากลุ่มของเพียร์ตัวใหม่ ซึ่งจะมีการส่งให้กับเพียร์อื่นๆ หลังจากที่โฮสต์ได้รับ `MessagePeerReady` จากเพียร์ใหม่แล้ว

ฟังก์ชัน `int SendChatMessage(SOCKET skSend, char* Playername, char* ChatMessage)`

เป็นฟังก์ชันที่ใช้ส่งข้อความระหว่างผู้เล่นที่อยู่กลุ่มโฮสต์เดียวกัน

ฟังก์ชัน `int SendRequestServerMessage(SOCKET skSend, char* RequestType, int UserID)`

เป็นฟังก์ชันที่ใช้ส่งต่อการร้องขอให้เซิร์ฟเวอร์ช่วยประมวลผลจากเพียร์ที่อยู่ในกลุ่มโฮสต์

4.5.3 ฟังก์ชันที่ใช้ทำการเข้ารหัสข้อมูล

ฟังก์ชันที่ใช้เข้ารหัสข้อมูลจะใช้นิยามปิดกั้นแบบเน็ตเวิร์คแบบไคลเอนท์และเซิร์ฟเวอร์ โดยมีการเข้ารหัสแบบกุญแจสาธารณะ ในส่วนของสถาปัตยกรรมเน็ตเวิร์คแบบเพียร์-ทู-เพียร์ จะมีการเข้ารหัสในส่วนที่ทำการติดต่อกับเซิร์ฟเวอร์ ซึ่งใช้การเข้ารหัสแบบกุญแจสมมาตร โดยใน Visual Studio .Net 2003 มีคลาสของ .Net Framework ที่ใช้การเข้ารหัสและถอดรหัสข้อมูลได้ จึงได้ใช้ฟังก์ชันของคลาสมาดำเนินการพัฒนาการเข้ารหัสข้อมูล โดยมีรายละเอียดดังนี้

คลาสที่ได้นำเอามาใช้

RSACryptoServiceProvider เป็นคลาสที่ทำหน้าที่เข้ารหัสและถอดรหัสข้อมูลแบบกุญแจสาธารณะ

RijndaelManaged เป็นคลาสที่ทำหน้าที่เข้ารหัสและถอดรหัสข้อมูลแบบกุญแจส่วนตัว

ฟังก์ชันที่พัฒนาใน โครงการงานพิเศษ

ฟังก์ชัน **EncryptData**

เป็นฟังก์ชันเข้ารหัสแบบกุญแจสาธารณะ โดยรับข้อมูล Plaintext และทำการเข้ารหัสข้อมูลได้เป็น Ciphertext

ฟังก์ชัน **DecryptData**

เป็นฟังก์ชันถอดรหัสแบบกุญแจสาธารณะ โดยรับข้อมูล Ciphertext และทำการถอดรหัสข้อมูลได้เป็น Plaintext

ฟังก์ชัน **GenSymmetricKey**

เป็นฟังก์ชันสร้างกุญแจสำหรับเข้ารหัสและถอดรหัสแบบกุญแจส่วนตัว

ฟังก์ชัน **GenSymmetricIV**

เป็นฟังก์ชันสร้างตัวช่วยเข้ารหัสและถอดรหัสแบบกุญแจส่วนตัว

ฟังก์ชัน **SymmetricEncryptData**

เป็นฟังก์ชันเข้ารหัสแบบกุญแจส่วนตัว โดยรับข้อมูล Plaintext และทำการเข้ารหัสข้อมูลได้เป็น Ciphertext

ฟังก์ชัน **SymmetricDecryptData**

เป็นฟังก์ชันถอดรหัสแบบกุญแจส่วนตัว โดยรับข้อมูล Ciphertext และทำการถอดรหัสข้อมูลได้เป็น Plaintext

4.5.4 ฟังก์ชันที่ใช้ในการแสดงผล

ฟังก์ชันกลุ่มนี้จะเป็นการแสดงผลภาพทั้งแบบ 2 มิติ และ 3 มิติ โดยการนำฟังก์ชันของโคเร็ก เอ็กซ์มาพัฒนา โดยมีรายละเอียดฟังก์ชันที่พัฒนาขึ้นดังนี้

ฟังก์ชัน **hrInit3D**

เป็นฟังก์ชันที่เตรียมการใช้งานการแสดงผล 3 มิติ

ฟังก์ชัน **hrLoadXFile**

เป็นฟังก์ชันโหลดข้อมูลโมเดลในไฟล์ .X เพื่อนำไปแสดงผล

ฟังก์ชัน vSetupDefaultView

เป็นฟังก์ชันกำหนดมุมมองในการแสดงผล 3 มิติ โดยเริ่มต้นโดยหันมุมมองไปยังจุด (0, 0, 0)

ฟังก์ชัน vUpdateAbsoluteView

เป็นฟังก์ชันที่ใช้กำหนดมุมมองตามตำแหน่งที่ต้องการ

ฟังก์ชัน vInitVertexBuffer

เป็นฟังก์ชันเตรียมความพร้อมการใช้งาน vertex เพื่อใช้แสดงผล texture ในระบบ 3 มิติ

ฟังก์ชัน vInitVertex

เป็นฟังก์ชันที่เตรียมความพร้อมก่อนทำการวาด vertex ในระบบ 3 มิติ

ฟังก์ชัน vBeginScene

เป็นฟังก์ชันที่เริ่มต้นการวาดภาพการแสดงผลต่างๆ ในระบบ 3 มิติ

ฟังก์ชัน vRenderObjectScene

เป็นฟังก์ชันที่ใช้วาดตัวโมเดล 3 มิติ

ฟังก์ชัน vRenderVertexScene

เป็นฟังก์ชันที่ใช้วาด vertex

ฟังก์ชัน vEndScene

เป็นฟังก์ชันจบการวาดภาพแสดงผล

ฟังก์ชัน InitSprite

เป็นฟังก์ชันที่ใช้สร้างอ็อบเจกต์ที่ใช้แสดงภาพ 2 มิติ

ฟังก์ชัน hrLoadSpriteTexture

เป็นฟังก์ชันที่ใช้โหลด texture และภาพ 2 มิติต่างๆ

ฟังก์ชัน vDrawSprite

เป็นฟังก์ชันที่ใช้วาด texture และภาพ 2 มิติ ในระบบ 2 มิติ

ฟังก์ชัน hrInitFont

เป็นฟังก์ชันที่ใช้เตรียมการใช้งานการวาดตัวอักษร

ฟังก์ชัน vDrawText

เป็นฟังก์ชันวาดตัวอักษรที่กำหนด แสดงผลบนหน้าจอ

บทที่ 5

การพัฒนาระบบ

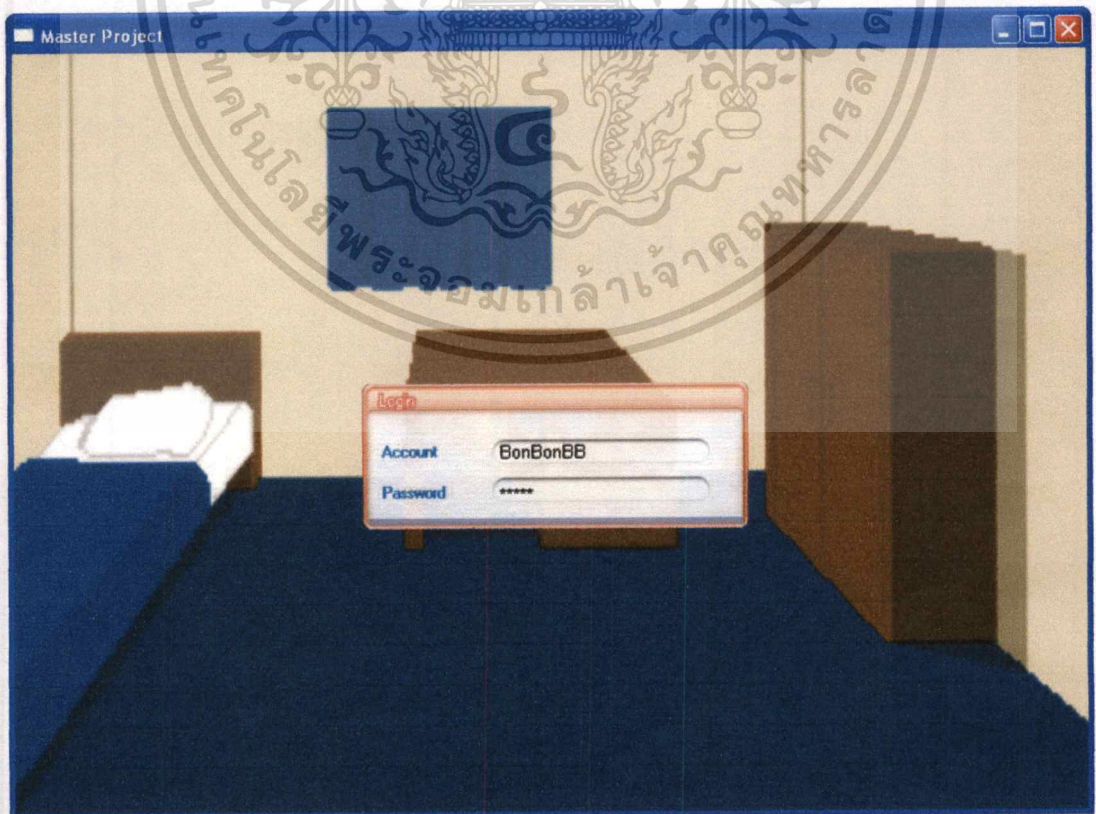
5.1 เครื่องมือในการพัฒนาระบบ

ในการพัฒนาระบบได้ใช้เครื่องมือในการเขียนโปรแกรม Microsoft Visual Studio .NET โดยใช้ Visual C++ .NET และใช้ไคลเอนต์ในการแสดงและประมวลผล 3 มิติ และมีการเชื่อมโยงกับ MySQL ในการดูแลและจัดการฐานข้อมูล

5.2 หน้าจอการทำงานของโปรแกรม

5.2.1 หน้าจอการเข้าระบบ

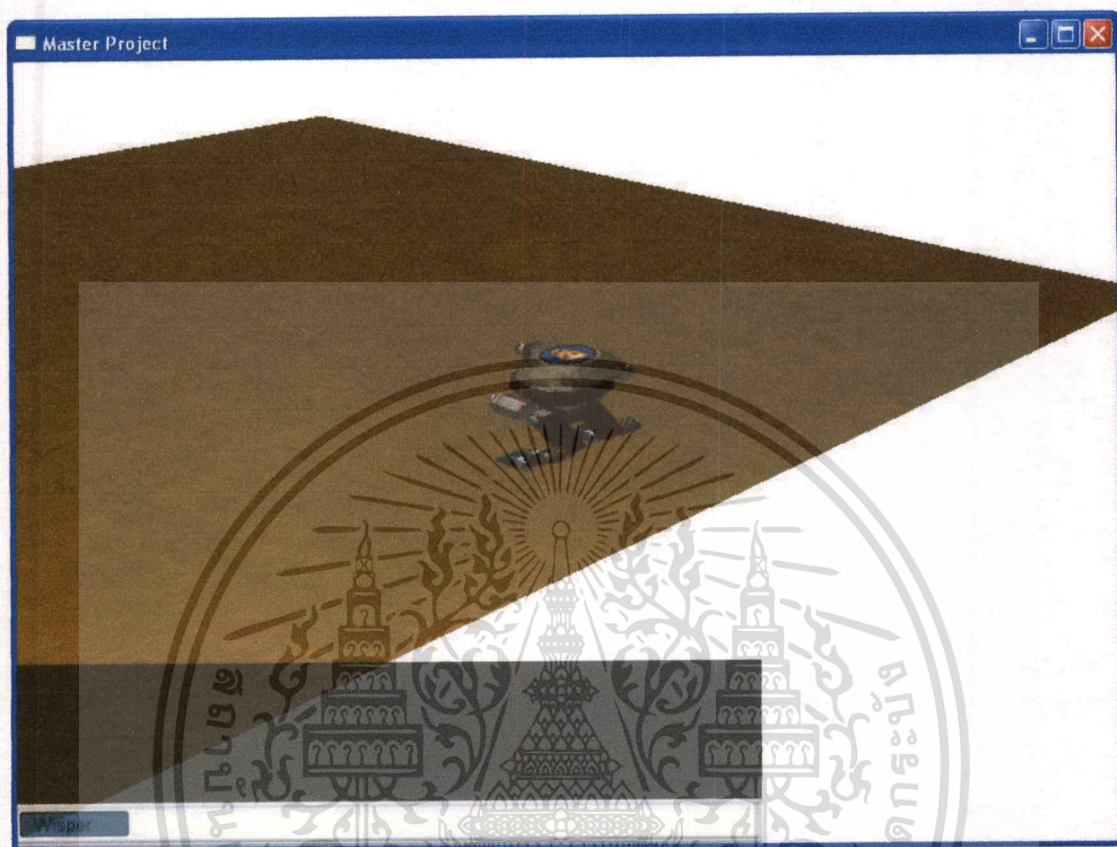
การทำงานจริงผู้เล่นจะต้องทำการสมัครสมาชิกรับระบบก่อนโดยการลงทะเบียนผ่านเว็บไซต์ ในโครงการพิเศษนี้ไม่ได้จัดทำตัวลงทะเบียนผ่านเว็บไซต์ ดังนั้นจึงนำเสนอแต่ตัวโปรแกรม หลังจากที่ผู้เล่นได้ทำการสมัครสมาชิกไว้เรียบร้อยแล้ว ซึ่งหน้าจอแรกหลังจากทำการเปิดตัวโปรแกรมเป็นดังรูปที่ 5.1



รูปที่ 5.1 หน้าจอการเข้าใช้ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผู้เล่นทำการใส่บัญชีเข้าใช้ระบบและรหัสผ่าน และทำการกด Enter ถ้าใส่ถูกต้องจะเข้าสู่บ้าน
ของตัวเอง แสดงในรูป 5.2

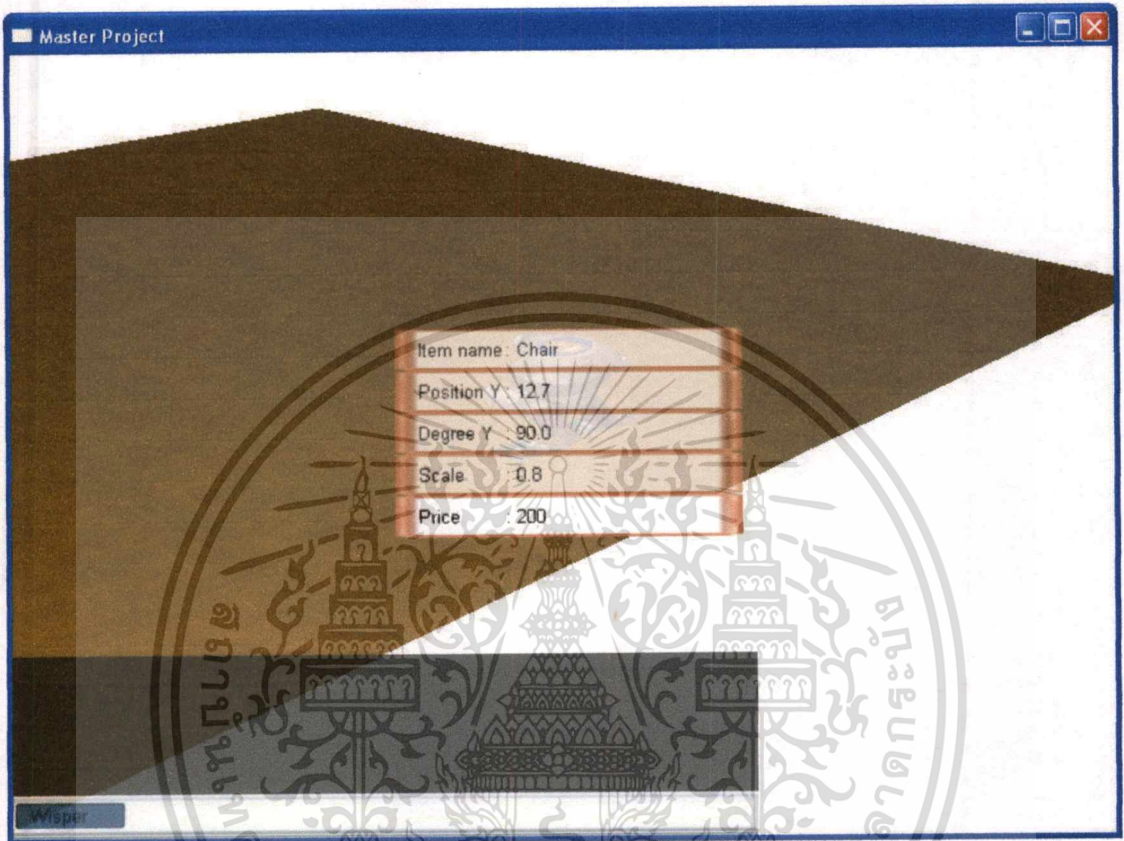


รูปที่ 5.2 หน้าจอหลังจากทำการเข้าสู่ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.2 หน้าจอลงทะเบียนสินค้าที่ผู้เล่นสร้างขึ้นเอง

ผู้เล่นสามารถทำการลงทะเบียนสินค้าที่ตัวเองสร้างขึ้นมาได้ โดยทำการกดปุ่ม R เพื่อทำการลงทะเบียนสินค้า ซึ่งจะมีหน้าให้กรอกข้อมูลสินค้าขึ้นมา ดังแสดงได้ดังรูป 5.3



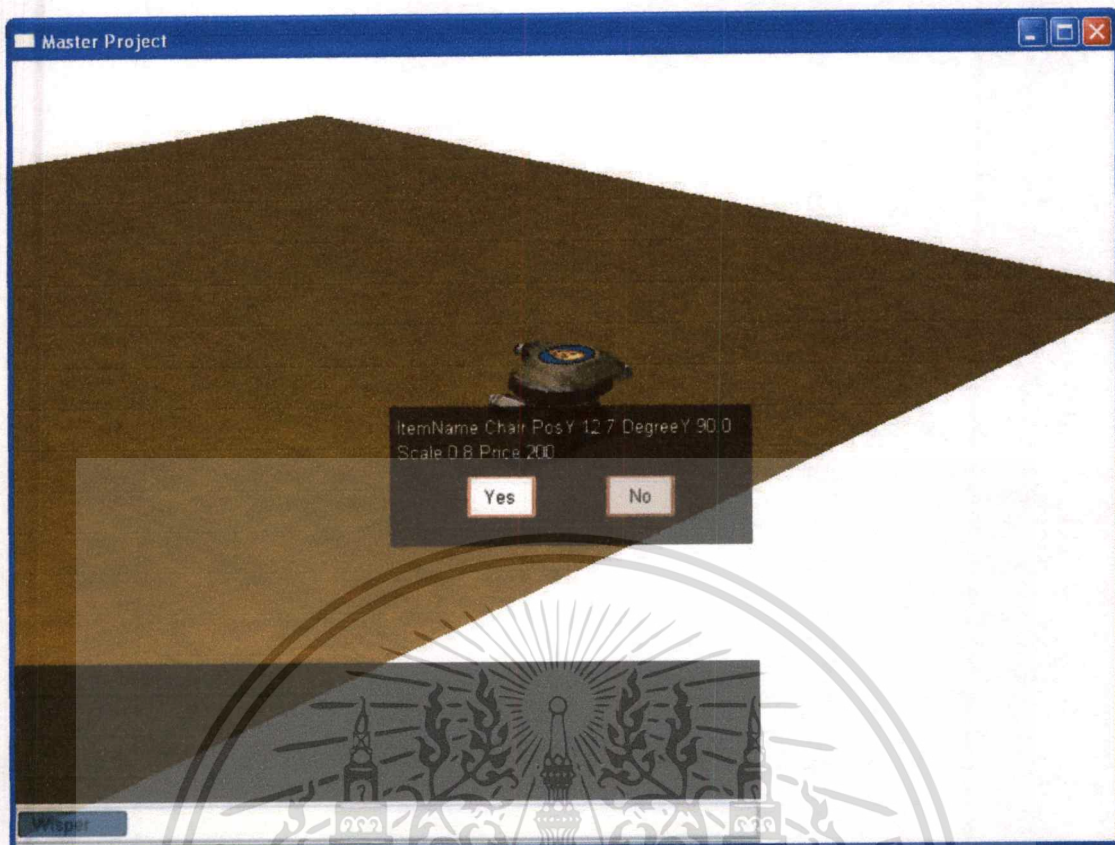
รูปที่ 5.3 หน้าจอการใส่ข้อมูลของสินค้าเพื่อทำการลงทะเบียน

ผู้เล่นทำการกรอกรายละเอียด โดยมีรายละเอียดดังนี้

- Item name : ชื่อสินค้าที่ต้องการลงทะเบียน
- Position Y : ระยะความสูงที่จะวางสินค้า เนื่องจากตัวโมเดลที่แปลงออกมา จุดเริ่มต้นของตัวโมเดลมักจะอยู่ที่จุดกึ่งกลางของตัวโมเดล ดังนั้นจึงต้องมีการกำหนดตำแหน่งความสูงที่จะว่าโมเดลด้วย
- Degree Y : มุมที่ต้องการให้โมเดลหันหน้า
- Scale : ขนาดของสินค้า โดยคิดเป็นจำนวนเท่าของขนาดจริงของโมเดล
- Price : ราคาสินค้าที่ต้องการขายให้กับผู้เล่นคนอื่น

เมื่อผู้เล่นทำการกรอกรายละเอียดสินค้าเสร็จแล้วให้กดปุ่ม Enter จะปรากฏหน้าจอขึ้นยืนยันข้อมูล ดังแสดงในรูป 5.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

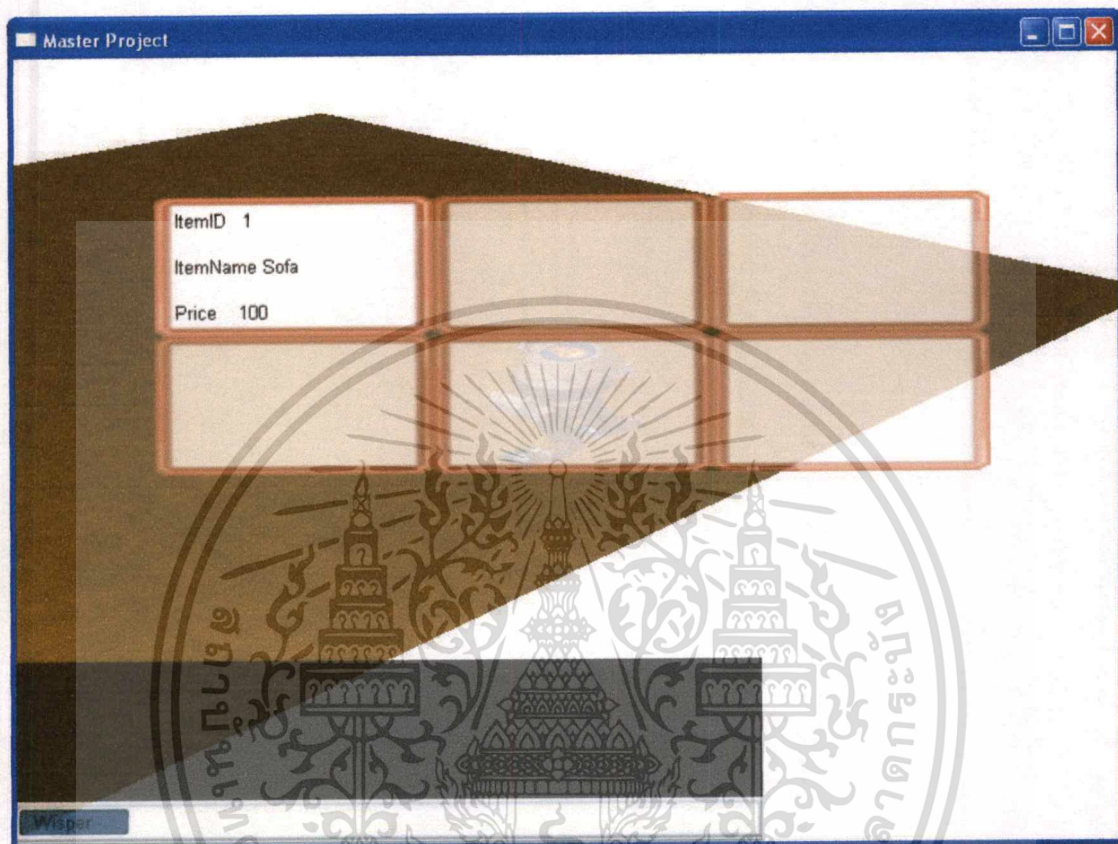


รูปที่ 5.4 หน้าจอยืนยันการลงทะเบียนสินค้า

ถ้าข้อมูลถูกต้องให้เลือก Yes และกดปุ่ม Enter รอซักครู่จะเห็นสินค้าที่ตัวเองได้ทำการลงทะเบียนไว้กับระบบ

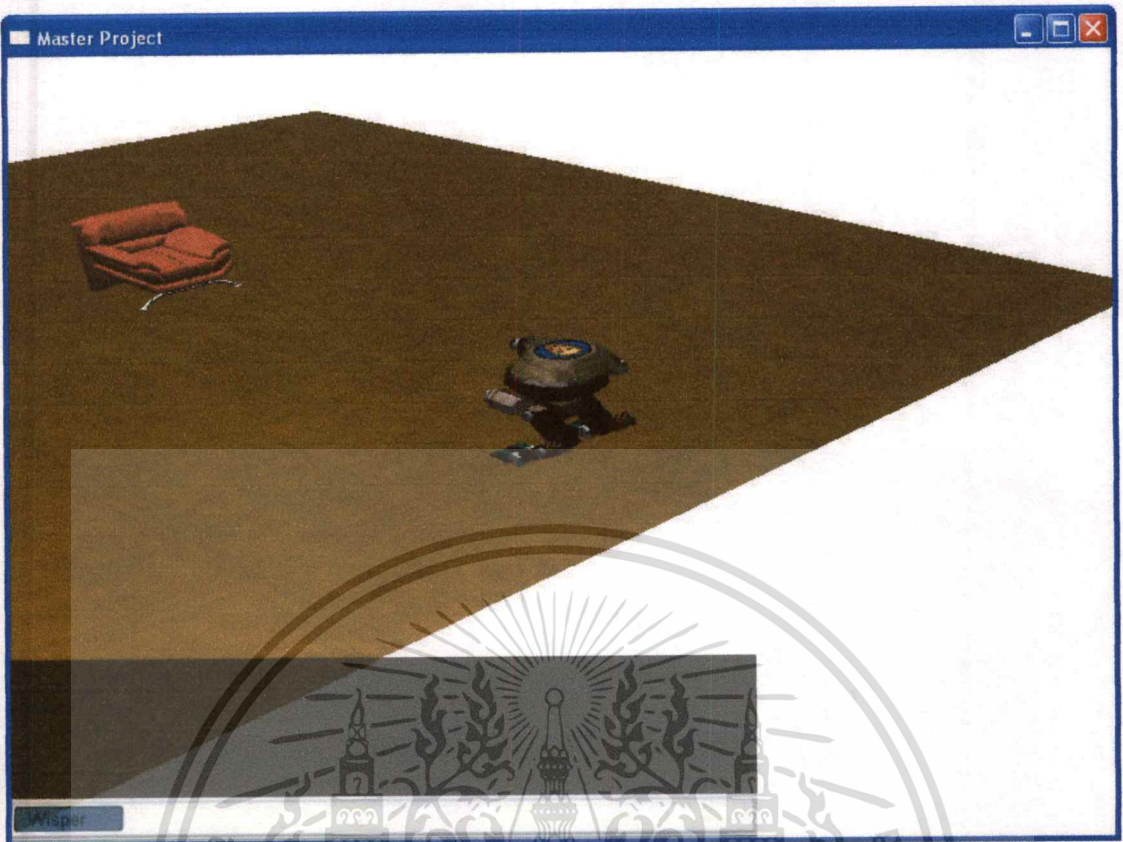
5.2.3 หน้าจอซื้อสินค้าจากเซิร์ฟเวอร์

ผู้เล่นสามารถทำการซื้อสินค้าจากเซิร์ฟเวอร์ได้โดยการกดปุ่ม T เมื่ออยู่ในบ้านของตัวเอง และ ยังไม่ได้ทำตัวเป็นโฮสต์ ซึ่งจะมีรายการสินค้าให้เลือก ดังแสดงในรูป 5.5



รูปที่ 5.5 หน้าจอแสดงรายการสินค้าที่ซื้อกับเซิร์ฟเวอร์

เมื่อผู้เล่นทำการเลือกสินค้า จะมีหน้าจอแสดงรายละเอียดสินค้าและยืนยันการซื้อโดยเลือกที่ Yes และทำการรอซักรูจะปรากฏสินค้าที่ผู้เล่นเลือกซื้อในบ้านของตัวเอง ดังแสดงในรูป 5.6

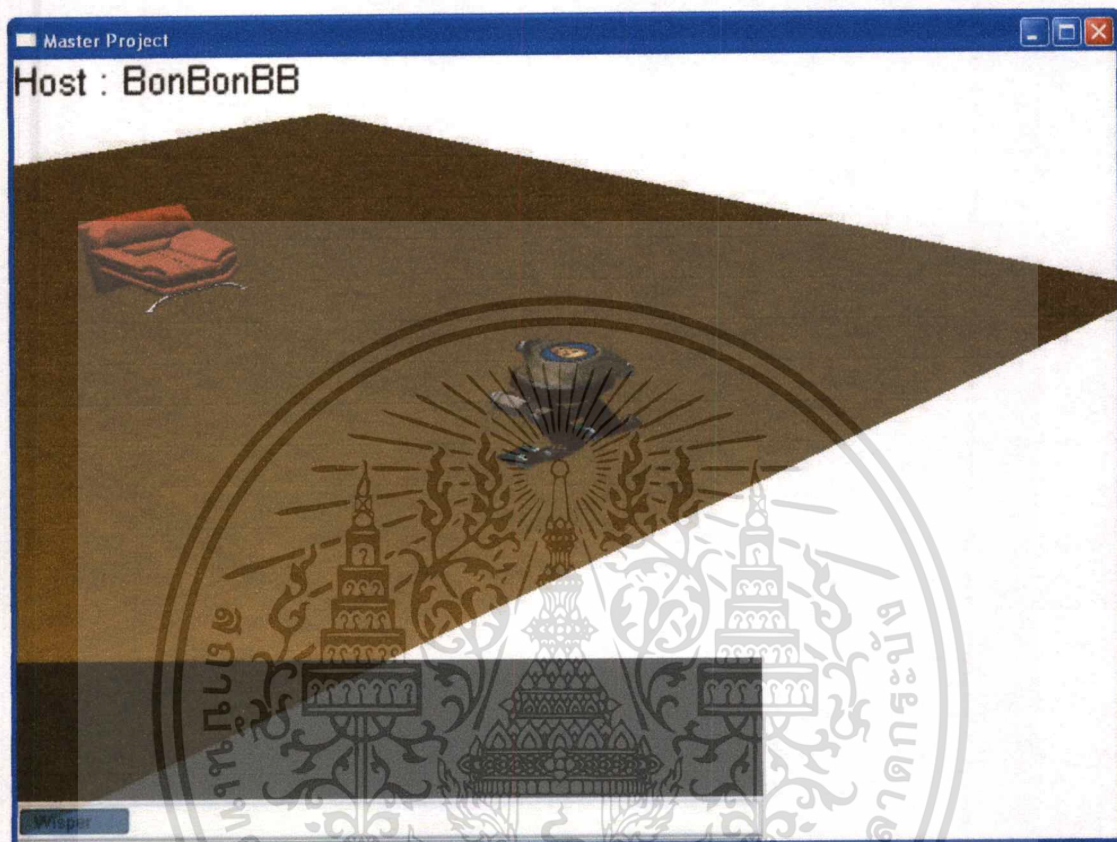


รูปที่ 5.6 หน้าจอแสดงเมื่อผู้เล่นทำการซื้อสินค้าเสร็จสิ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.4 หน้าจอการสร้างโฮสต์

ผู้เล่นสามารถทำการสร้างโฮสต์ได้โดยการกดปุ่ม H และจะปรากฏคำว่า Host ที่หน้าจอ ซึ่งแสดงได้ดังรูป 5.7

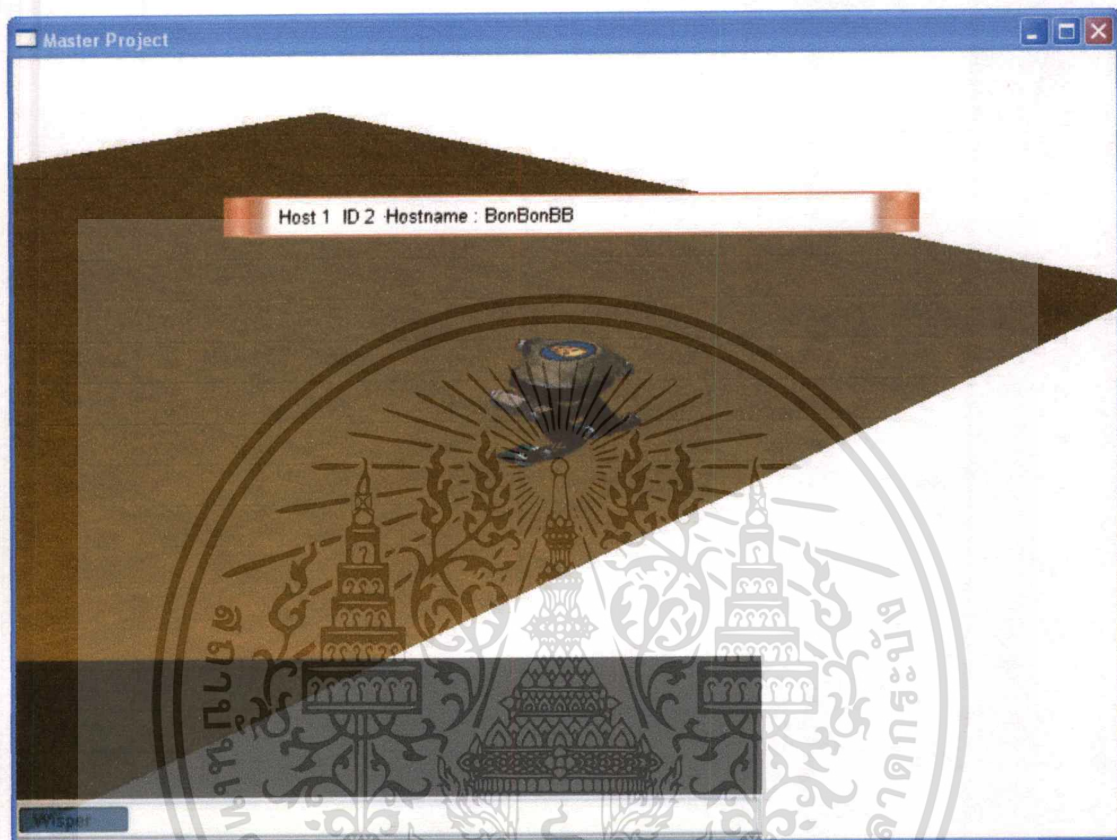


รูปที่ 5.7 หน้าจอแสดงความเป็นโฮสต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

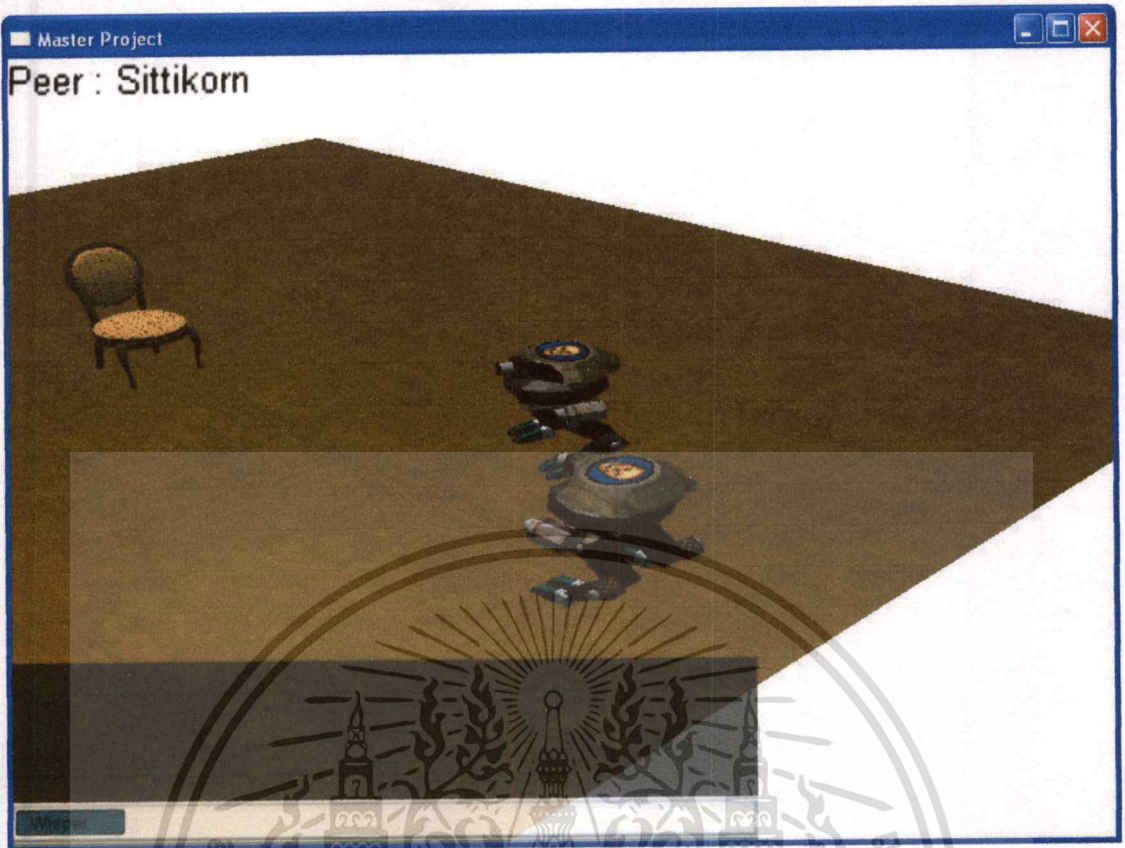
5.2.5 หน้าจอการเข้าร่วมโฮสต์

ผู้เล่นสามารถทำการเข้าร่วมโฮสต์กับผู้เล่นคนอื่นๆ ได้โดยการกดปุ่ม J และจะปรากฏรายการโฮสต์ที่ออนไลน์อยู่ในปัจจุบัน ดังแสดงในรูป 5.8



รูปที่ 5.8 หน้าจอแสดงการเลือกโฮสต์ที่ต้องการเข้าร่วม

เมื่อผู้เล่นทำการเลือกโฮสต์ผู้เล่นจะเข้าไปอยู่ในบ้านของโฮสต์นั้น ดังแสดงในรูป 5.9

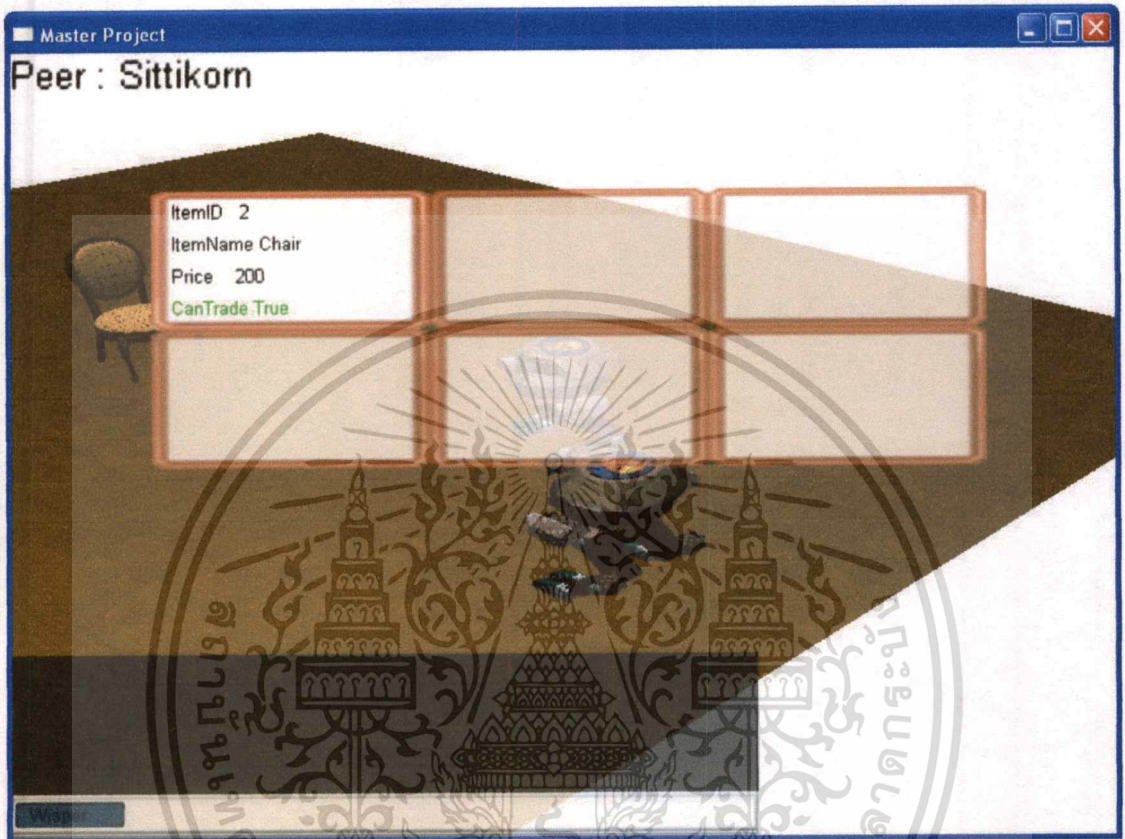


รูปที่ 5.9 หน้าจอแสดงการเข้าร่วมโฮสต์กับผู้เล่นคนอื่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.6 หน้าจอการซื้อขายระหว่างเพียร์และโฮสต์

ผู้เล่นที่เป็นเพียร์สามารถทำการซื้อสินค้าจากโฮสต์ได้โดยการกดปุ่ม T จะแสดงรายการสินค้าที่โฮสต์มีรวมทั้งบอกว่าสินค้าไหนที่สามารถทำการซื้อได้ ดังแสดงในรูป 5.10

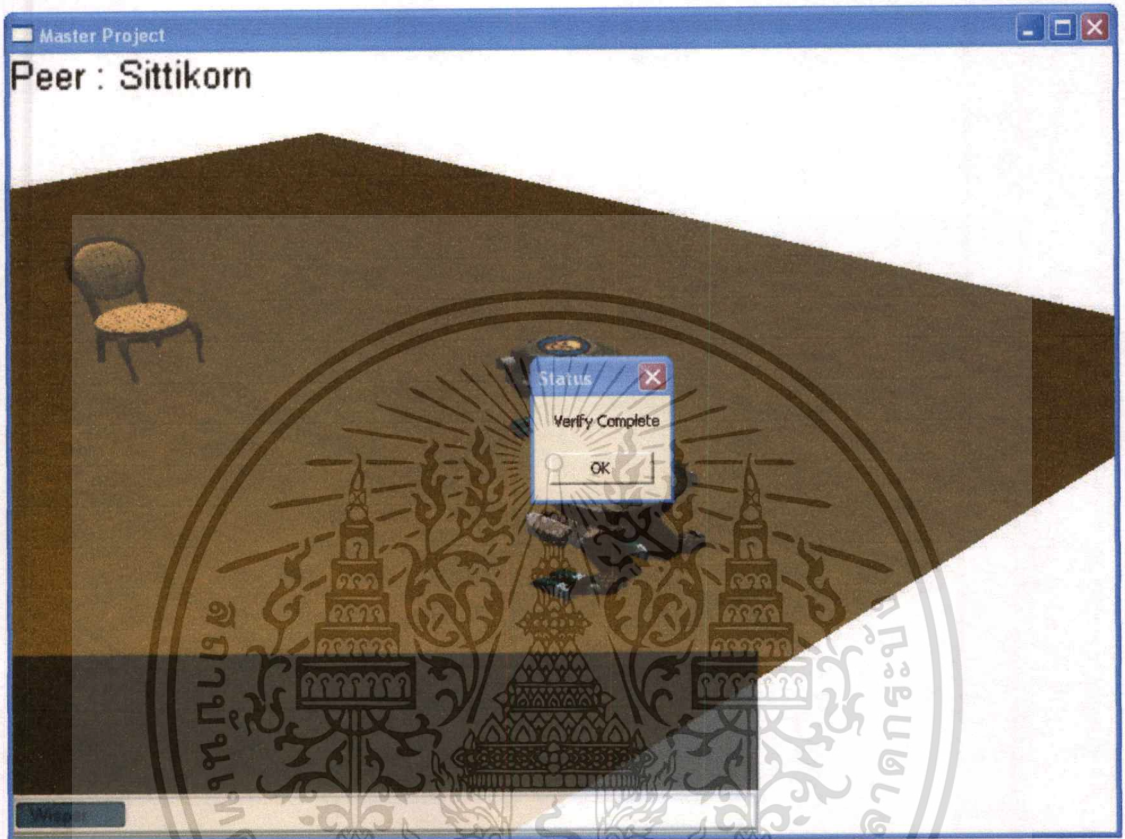


รูปที่ 5.10 หน้าจอแสดงรายการสินค้าที่โฮสต์ขาย

เมื่อทำการเลือกซื้อแล้วจะปรากฏหน้าจอขึ้นขึ้นให้เลือก Yes และถ้าการซื้อขายสำเร็จจะมีกล่องข้อความแจ้งว่าสำเร็จ

5.2.7 หน้าจอแสดงการตรวจสอบความถูกต้องของข้อมูลโฮสต์

ผู้เล่นที่เป็นเพียร์สามารถทำการตรวจสอบความถูกต้องของข้อมูลโฮสต์ได้ โดยการกดปุ่ม V ซึ่งเมื่อระบบตรวจสอบเสร็จจะปรากฏกล่องข้อความ ดังแสดงในรูปที่ 5.11



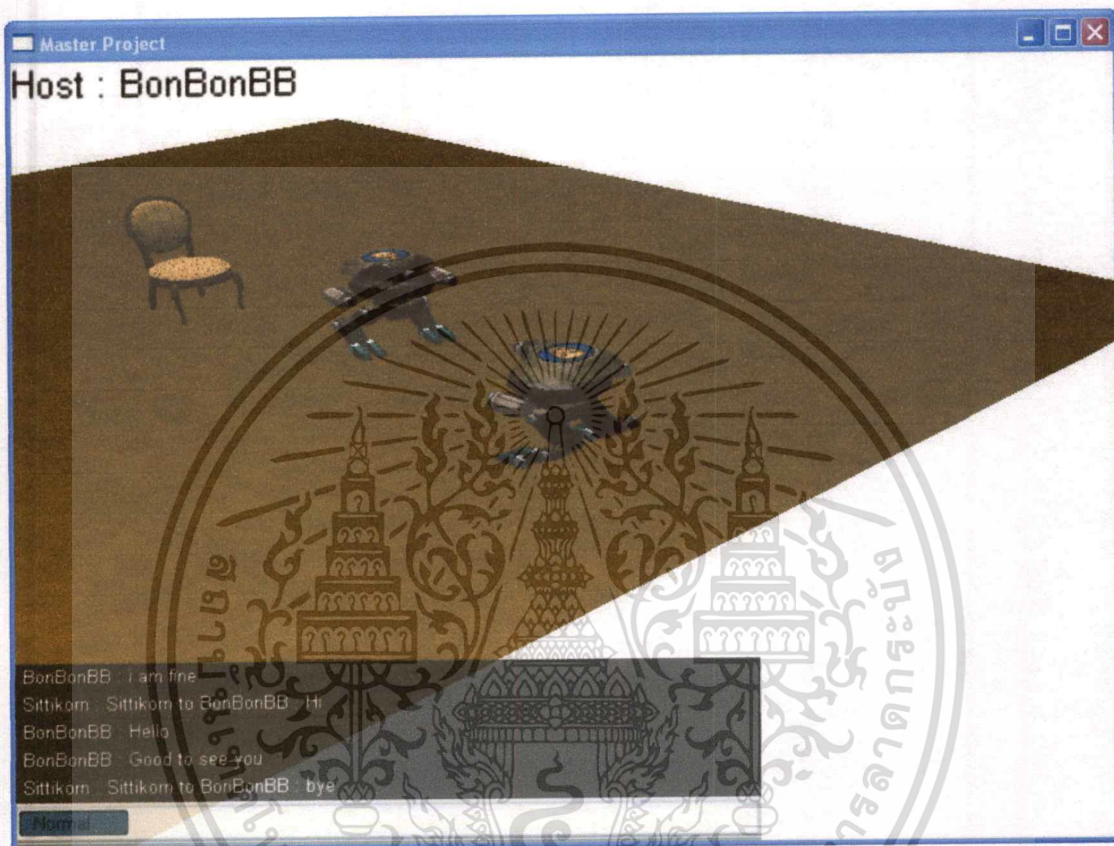
รูปที่ 5.11 หน้าจอแสดงการตรวจสอบข้อมูลโฮสต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.8 หน้าจอแสดงการพูดคุย

ผู้เล่นสามารถทำการพูดคุยถึงกันได้ โดยการกดปุ่ม Enter และพิมพ์ข้อความที่ต้องการส่ง เมื่อต้องการส่งให้ทำการกดปุ่ม Enter อีกครั้งหนึ่ง ซึ่งตัวอย่างการแสดงผลหน้าจอการพูดคุยเป็นดังรูป

5.12



รูปที่ 5.12 หน้าจอแสดงการพูดคุยของผู้เล่น

ในการพูดคุยกับผู้เล่นจะมีการตรวจจับการหันหน้าของตัวละครด้วย ซึ่งเมื่อหันหน้าไปยังตัวละครผู้เล่นคนอื่น ผู้เล่นคนนั้นจะทราบได้ว่ามีคนพูดด้วยโดยข้อความที่ส่งจะมีชื่อของผู้ส่งแนบมาด้วย

บทที่ 6

สรุปผลและข้อเสนอแนะ

เนื่องจากระบบเกมออนไลน์จำเป็นต้องมีความถูกต้องของข้อมูลเวลาที่ทำกรจัดเก็บข้อมูล และรับ-ส่งข้อมูล จึงเป็นเหตุผลที่ว่าสถาปัตยกรรมแบบไคลเอนท์และเซิร์ฟเวอร์เป็นสถาปัตยกรรมหลักและสำคัญที่ต้องมีคู่กับการพัฒนาเกมออนไลน์ แต่เนื่องด้วยปกติแล้วเกมออนไลน์เป็นเกมที่มุ่งเน้นการจำลองความสามารถทางด้านสังคมของมนุษย์ มาไว้ในเกม ซึ่งการพูดคุยติดต่อกันในเกมนั้นจำเป็นต้องมี แต่ข้อมูลของการพูดคุยไม่ได้มีความสำคัญอะไรมากนัก เมื่อเทียบกับข้อมูลส่วนอื่นๆ ของเกม นอกจากนี้การต่างๆ ทั้งหมดยังคงไปอยู่ทางฝั่งเซิร์ฟเวอร์แค่ฝั่งเดียว การนำเพียร์-ทู-เพียร์มาใช้เพื่อลดภาระบางอย่างจึงเป็นแนวคิดที่ดี เพื่อใช้ลดภาระของเซิร์ฟเวอร์

6.1 สรุปผล

ในการพัฒนาระบบงานนี้ ได้ใช้เครื่องมือในการพัฒนา คือ Visual Studio .Net 2003 ใช้ฐานข้อมูล MySQL Server และได้ใช้ชุดพัฒนาซอฟต์แวร์และเกม DirectX โดยสามารถจำลองการทำงานของเกมออนไลน์ขนาดเล็ก ที่ใช้สถาปัตยกรรมเครือข่ายแบบไคลเอนท์และเซิร์ฟเวอร์ และสถาปัตยกรรมเครือข่ายแบบเพียร์-ทู-เพียร์ได้ ซึ่งได้ทำการแยกส่วนของการประมวลผลบางอย่างให้ไคลเอนท์สามารถทำการประมวลผลเองได้ พบว่าสามารถช่วยลดการทำงานของเซิร์ฟเวอร์ได้ โดยที่ไม่มีผลกระทบต่อระบบโดยรวมของเกมออนไลน์ นอกจากนี้ยังสามารถนำมาใช้พัฒนาโปรแกรมเน็ตเวิร์คโดยแยกส่วนที่ไคลเอนท์สามารถทำงานเองได้อย่างเช่นการส่งไฟล์ประเภทต่างๆ วิดีโอสตริมมิ่ง ได้โดยไม่ต้องทำผ่านเซิร์ฟเวอร์ แต่อย่างไรก็ตามการทำงานที่อยู่ฝั่งไคลเอนท์ทำให้ด้านความปลอดภัยนั้นลดลง จึงจำเป็นต้องทำอย่างที่จะต้องเพิ่มส่วนที่เป็นการเข้ารหัสข้อมูลเข้าไป ในโปรแกรมด้วย ซึ่งจะทำให้โปรแกรมมีประสิทธิภาพและน่าเชื่อถือมากยิ่งขึ้น

6.2 ข้อเสนอแนะ

สำหรับโครงการพัฒนาระบบงานนี้ ยังสามารถพัฒนาต่อได้อีก ดังนี้

1. พัฒนาส่วนของการโอนไฟล์ข้อมูลระหว่างเพียร์ เช่น รูปภาพต่างๆ หรือไฟล์ข้อมูลต่างๆ
2. พัฒนาวิธีโอคอนเฟอเรนซ์ติดต่อกันระหว่างเพียร์ หรือการส่งสตริมมิ่งต่างๆ ถึงกันได้
3. ออกแบบส่วนของการรักษาความปลอดภัยในฝั่งของไคลเอนท์ที่จำลองตัวเองเป็นโฮสต์และเพียร์ ให้มีความปลอดภัยมากยิ่งขึ้น การปลอมแปลงทำได้ยากขึ้น
4. พัฒนาตัวเอนจินสำหรับพัฒนาเกมออนไลน์โดยใช้สถาปัตยกรรมเครือข่ายแบบผสม

เพื่อให้การพัฒนาเกมออนไลน์สามารถทำได้รวดเร็วมากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการศึกษาเท่านั้น ไม่ควรนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. พัฒนาโปรโตคอลให้เป็นมาตรฐาน โดยออกแบบให้แมตเซตมีขนาดเล็กถึงแต่ยังคงมีความปลอดภัยในการสื่อสาร



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- Andrew and Kirmse, C. 1997. **Security in Online Games**. [Online]. Available:
<http://www.gamasutra.com/features/19970707/security.htm>.
- Bobic, N. 2000. **Advanced Collision Detection Techniques**. [Online]. Available:
http://www.gamasutra.com/features/20000330/bobic_pfv.htm.
- Federal Information Processing Standards Publication 180-1. 1997. **Secure Hash Standard**.
 [Online]. Available: <http://www.itl.nist.gov/fipspubs/fip180-2.htm>.
- Federal Information Processing Standards Publication 186. 1994. **Digital Signature Standard**.
 [Online]. Available: <http://www.itl.nist.gov/fipspubs/fip180-2.htm>.
- Knutsson, B and Hopkins, B. ET.AL. 2004. "Peer-to-Peer Support for Massively Multiplayer Games." **IEEE Computer and Communication Societies**. 23(1): 107.
- Yan, J. 2003. "Security Design in Online Games." **IEEE Computer Security Applications Conference**. 19: 286-295.



ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

ฟังก์ชันต้นแบบสำหรับพัฒนาโปรแกรมที่ใช้สถาปัตยกรรมเครือข่าย

แบบผสม

ฟังก์ชันต้นแบบที่ใช้สำหรับพัฒนาโปรแกรมที่ใช้สถาปัตยกรรมเครือข่ายแบบผสมนี้ เป็นการนำฟังก์ชันในหัวข้อ 4.5.2 มาขยายรายละเอียด ซึ่งจะสามารถนำไปใช้พัฒนาต่อไปในอนาคตได้ ดังนี้

รายละเอียดฟังก์ชัน

1. ฟังก์ชัน SendStatusMessage

รายละเอียดของฟังก์ชัน

เป็นฟังก์ชันที่ใช้แจ้งสถานะของตัวไคลเอนท์ให้เซิร์ฟเวอร์ทราบ

รูปแบบการใช้งาน

```
int SendStatusMessage(SOCKET skSend, StatusCode code, int UserID)
```

การคืนค่าของฟังก์ชัน

จำนวนข้อมูลหน่วยเป็นไบต์ ที่ส่งผ่าน skSend

พารามิเตอร์ของฟังก์ชัน

skSend เป็นซ็อกเก็ตปลายทางที่ต้องการส่งข้อมูลสถานะ

code เป็นค่าไค้ดของสถานะ ถูกประกาศไว้ใน Enum StatusCode

UserID เป็นหมายเลขอ้างอิงผู้เล่นในเกมที่ต้องการแจ้งสถานะกับเซิร์ฟเวอร์

2. ฟังก์ชัน SendLoginMessage

รายละเอียดของฟังก์ชัน

เป็นฟังก์ชันที่ใช้ส่งข้อมูลการเข้าใช้ระบบจากไคลเอนท์ให้กับเซิร์ฟเวอร์ เพื่อตรวจสอบสิทธิ์

การเข้าใช้ระบบ

รูปแบบการใช้งาน

```
int SendLoginMessage(SOCKET skSend, char* Username, char* Password)
```

การคืนค่าของฟังก์ชัน

จำนวนข้อมูลหน่วยเป็นไบต์ ที่ส่งผ่าน skSend

พารามิเตอร์ของฟังก์ชัน

skSend เป็นซ็อกเก็ตปลายทางที่ต้องการส่งข้อมูลการเข้าใช้ระบบ

Username เป็นชื่อบัญชีการเข้าใช้ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Password เป็นรหัสผ่านการใช้งานระบบ

3. ฟังก์ชัน SendHostPortMessage

รายละเอียดของฟังก์ชัน

เป็นฟังก์ชันที่ใช้บอกสถานะของผู้ส่งที่ทำตัวเป็นโฮสต์ให้เซิร์ฟเวอร์ทราบว่าพร้อมรับการเชื่อมต่อจากเพียร์อื่นๆ แล้ว

รูปแบบการใช้งาน

```
int SendHostPortMessage(SOCKET skSend, int Port)
```

การคืนค่าของฟังก์ชัน

จำนวนข้อมูลหน่วยเป็น ไบต์ ที่ส่งผ่าน skSend

พารามิเตอร์ของฟังก์ชัน

skSend เป็นซ็อกเก็ตที่เชื่อมต่อกับเซิร์ฟเวอร์ ที่ต้องการส่งข้อมูลโฮสต์ไปให้

Port เป็นหมายเลขพอร์ตที่ไคลเอนต์ทำการเปิดเพื่อรอรับการเชื่อมต่อของเพียร์อื่น

4. ฟังก์ชัน SendPeerReadyMessage

รายละเอียดของฟังก์ชัน

เป็นฟังก์ชันที่ใช้ส่งสถานะของผู้ส่งที่เป็นเพียร์ให้กับโฮสต์ทราบว่าพร้อมที่จะเริ่มทำการรับส่งข้อมูลภายในกลุ่ม และพร้อมที่จะให้เพียร์อื่นๆ เข้ามาเชื่อมต่อด้วย

รูปแบบการใช้งาน

```
int SendPeerReadyMessage(SOCKET skSend, int PlayerID, int Port)
```

การคืนค่าของฟังก์ชัน

จำนวนข้อมูลหน่วยเป็น ไบต์ ที่ส่งผ่าน skSend

พารามิเตอร์ของฟังก์ชัน

skSend เป็นซ็อกเก็ตที่เชื่อมต่อกับโฮสต์ ที่ต้องการส่งข้อมูลสถานะเพียร์ไปให้

PlayerID เป็นหมายเลขอ้างอิงผู้เล่นที่เป็นเพียร์

Port เป็นหมายเลขพอร์ตที่เพียร์ทำการเปิดเพื่อรอรับการเชื่อมต่อของเพียร์อื่นในกลุ่ม

5. ฟังก์ชัน SendNewPeerMessage

รายละเอียดของฟังก์ชัน

เป็นฟังก์ชันที่แจ้งสถานะให้เพียร์อื่นๆ ในกลุ่มรับทราบการเข้ากลุ่มของเพียร์ตัวใหม่ ซึ่งจะมีการส่งให้กับเพียร์อื่นๆ หลังจากที่โฮสต์ได้รับ MessagePeerReady จากเพียร์ใหม่แล้ว

รูปแบบการใช้งาน

```
int SendNewPeerMessage(SOCKET skSend, char* PeerIP, int Port)
```

การคืนค่าของฟังก์ชัน

จำนวนข้อมูลหน่วยเป็นไบต์ ที่ส่งผ่าน skSend

พารามิเตอร์ของฟังก์ชัน

skSend เป็นซ็อกเก็ตที่เชื่อมต่อกับเพียร์ที่อยู่ในกลุ่ม

PeerIP เป็นหมายเลข IP หรือชื่อเครื่องของเพียร์ตัวใหม่ที่เชื่อมต่อเข้ากลุ่ม

Port เป็นหมายเลขพอร์ตที่เพียร์ตัวใหม่เปิดเพื่อรอรับการเชื่อมต่อของเพียร์อื่นในกลุ่ม

6. ฟังก์ชัน SendChatMessage

รายละเอียดของฟังก์ชัน

เป็นฟังก์ชันที่ใช้ส่งข้อความระหว่างผู้เล่นที่อยู่กลุ่มโฮสต์เดียวกัน

รูปแบบการใช้งาน

```
int SendChatMessage(SOCKET skSend, char* Playername, char* ChatMessage)
```

การคืนค่าของฟังก์ชัน

จำนวนข้อมูลหน่วยเป็นไบต์ ที่ส่งผ่าน skSend

พารามิเตอร์ของฟังก์ชัน

skSend เป็นซ็อกเก็ตที่เชื่อมต่อกับเพียร์ที่อยู่ในกลุ่ม

Playername เป็นชื่อของผู้เล่นที่ต้องการส่งข้อความ

ChatMessage เป็นข้อความที่ต้องการส่งไปยังเพียร์อื่นในกลุ่ม

7. ฟังก์ชัน SendRequestServerMessage

รายละเอียดของฟังก์ชัน

เป็นฟังก์ชันที่ใช้ส่งต่อการร้องขอให้เซิร์ฟเวอร์ช่วยประมวลผลจากเพียร์ที่อยู่ในกลุ่มโฮสต์

รูปแบบการใช้งาน

```
int SendRequestServerMessage(SOCKET skSend, char* RequestType, int UserID)
```

การคืนค่าของฟังก์ชัน

จำนวนข้อมูลหน่วยเป็นไบต์ ที่ส่งผ่าน skSend

พารามิเตอร์ของฟังก์ชัน

skSend เป็นซ็อกเก็ตที่เชื่อมต่อกับโฮสต์ของกลุ่ม ซึ่งจะเป็นผู้ที่ส่งต่อไปเซิร์ฟเวอร์

RequestType เป็นข้อความที่ทำการเข้ารหัสแล้ว โดยเก็บรายละเอียดการร้องขอที่เพียร์ต้องการ

UserID เป็นหมายเลขอ้างอิงของผู้เล่นที่เป็นเพียร์ ซึ่งต้องการร้องขอให้เซิร์ฟเวอร์ช่วยทำงานให้

ประวัติผู้เขียน

นายสิทธิกร คุณวิโรจน์พานิช เกิดเมื่อวันที่ 31 ตุลาคม พ.ศ. 2524 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาปริญญาตรีวิทยาศาสตร์บัณฑิต สาขาวิทยาการคอมพิวเตอร์ จากภาควิชาคณิตศาสตร์ และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ในปีการศึกษา 2546 และได้เข้าทำงานในตำแหน่งโปรแกรมเมอร์ บริษัทไซเบอร์พลานेटอินเทอร์แอคทีฟมหาชนจำกัด ทำหน้าที่พัฒนาเกมบนมือถือยี่ห้อต่างๆ และพัฒนาระบบควาน์โพลคอนเทนต์ต่างๆ สำหรับมือถือผ่านระบบเว็บ(wap) ทั้งที่เป็นแบบให้ฟรีควาน์โพลและแบบคิดค่าบริการ โดยทำการเชื่อมต่อกับระบบคิดค่าบริการกับผู้ให้บริการเครือข่ายมือถือ ในประเทศไทย เช่น ดีแท็ก และเอไอเอส ต่อมาได้เข้าศึกษาต่อในระดับปริญญาโท หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาเทคโนโลยีสารสนเทศ ภาควิชาวิทยาการสารสนเทศ คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ในปีการศึกษา 2547



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้