สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เกมเศรษฐีระบบออนไลน์

MONOPOLY GAME ONLINE

T 036148

พรทวี อำนรรฆกิตติกุล
พีรพงศ์ จุฑารัตน์จรัส
อนุรักษ์ รัตนวิฬาร์

ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต
ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2542

เลขหมู่
เลขทะเบียน 36148
วัน, เดือน, ปี 11 ก.ค. 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น และห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MONOPOLY GAME ONLINE

PHONTAWEE AMNUKKITTIKUL
PEERAPONG CHUTARATJARUS
ANURAK RATTANAVILA

A SPECIAL PROJECT SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF BACHELOR OF SCIENCE
DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCES
FACULTY OF SCIENCE
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
ACADEMIC YEAR 1999

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปัญหาพิเศษ	เกมเศรษฐีระบบออนไลน์	
	MONOPOLY GAME ONLINE	
ชื่อนักศึกษา	นายพรทวี อำนรรฆกิตติกุล	39054642
	นายพีรพงศ์ จุฑารัตน์จรัส	39054646
	นายอนุรักษ์ รัตนวิฬาร์	39054684
ภาควิชา	คณิตศาสตร์และวิทยาการคอมพิวเตอร์	
สาขาวิชา	วิทยาการคอมพิวเตอร์	
อาจารย์ที่ปรึกษา	อาจารย์สมศรี บัณฑิตวิไล	

ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้นับปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร วิทยาศาสตรบัณฑิต สาขาวิทยาการคอมพิวเตอร์ ประจำปีการศึกษา 2542

คณะกรรมการสอบ		ลายมือชื่อ
ประธานกรรมการ	อาจารย์นันทิกา เบญจเทพานันท์	
กรรมการ	อาจารย์สิริลักษณ์ อนันต์สถิตย์สิน	
กรรมการและอาจารย์ที่ปรึกษา	อาจารย์สมศรี บัณฑิตวิไล	

(อาจารย์ไพรบูลย์ พันธรักษ์พงษ์)

หัวหน้าภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์

ลิขสิทธิ์ของภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปัญหาพิเศษ	เกมเศรษฐีระบบออนไลน์	
ชื่อนักศึกษา	นายพรทวี อำนรรฆกิตติกุล	39054642
	นายพีรพงศ์ จุฑารัตน์จรัส	39054646
	นายอนุรักษ์ รัตนวิพาร์	39054684
ภาควิชา	คณิตศาสตร์และวิทยาการคอมพิวเตอร์	
สาขาวิชา	วิทยาการคอมพิวเตอร์	
ปีการศึกษา	2542	
อาจารย์ที่ปรึกษา	อาจารย์สมศรี บัณฑิตวิไล	

บทคัดย่อ

ในปัจจุบันเป็นที่ยอมรับกันอย่างกว้างขวางว่า ระบบคอมพิวเตอร์ และระบบการสื่อสารข้อมูลนั้น นอกจากจะสามารถเพิ่มประสิทธิภาพ และอำนวยความสะดวกในการทำงานทุกระบบงานแล้ว คอมพิวเตอร์ยังถูกนำมาใช้ในแง่ของการให้ความบันเทิงกับผู้ใช้งานในแบบต่าง ๆ เช่น การชมภาพยนตร์ การฟังเพลง ฯลฯ การเล่นเกมก็เช่นกัน เป็นความบันเทิงอย่างหนึ่งที่คอมพิวเตอร์สามารถตอบสนองความต้องการของผู้ใช้งานที่ต้องการผ่อนคลายความตึงเครียด ด้วยเหตุนี้ คอมพิวเตอร์ในปัจจุบัน จึงถูกนำมาใช้เป็นเครื่องมือที่ให้ความสนุกสนานเพลิดเพลินแก่ผู้ใช้งานนอกเหนือจากการใช้งานแบบปกติอีกด้วย

ดังนั้น เกมเศรษฐีระบบออนไลน์ ซึ่งมีพื้นฐานการเล่นมาจาก แบบแผนและกฎกติกาของเกมเศรษฐีแบบกระดาน ซึ่งเป็นที่นิยมมากจึงได้ถูกพัฒนาขึ้นมาเป็นเกมบนระบบคอมพิวเตอร์ที่สามารถเล่นได้ทั้งแบบคนเดียว(คนกับคอมพิวเตอร์) แบบหลายคนบนคอมพิวเตอร์เครื่องเดียว (คนกับคน) และแบบหลายคนผ่านระบบเครือข่าย ซึ่งแบบสุดท้ายนี้กำลังเป็นที่นิยมอย่างมากในปัจจุบัน เพราะการเล่นเกมผ่านระบบเครือข่าย (โดยการสื่อสารผ่าน โปรโตคอล TCP/IP) ทำให้ผู้เล่นแต่ละคน สามารถเล่นเกมร่วมกันได้ถึงแม้ว่าจะอยู่ห่างไกลกัน เพียงแค่ผู้เล่นแต่ละคนต้องเชื่อมต่ออยู่บนระบบเครือข่ายที่สื่อสารกันด้วยโปรโตคอล TCP/IP เท่านั้น ผู้เล่นทุกคนก็สามารถเล่นเกมร่วมกันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Special Project Title	MONOPOLY GAME ONLINE
Students	Mr.Phontawee Amnukkitikul 39054642
	Mr.Peerapong Chutaratjarus 39054646
	Mr.Anurak Rattanavila 39054684
Degree	Bachelor's Degree of Science
Department	Mathematics and Computer Sciences, Faculty of Science
Programme	Computer Sciences
Academic Year	1999
Special Project Advisor	Lecturer Somsri Banditvilai

ABSTRACT

Nowadays, it is widely accepted that Computer System and Data Communication can increase the efficiency and convenience to every working system. Besides, computer is used to entertain users in various ways such as movies, music, etc (computer multimedia). Games are also the example of the entertainment which computer can provide. So computer are used for entertainment reason, in addition to the normal use.

Monopoly Game Online, which based on the rules and regulations of original monopoly game, is developed to be a computer game. Players can choose to play alone (with computer), together on the same computer, or link together through network. The last style is quite popular. By networking (through Protocal TCP/IP), each player can join together even they are separated by distance. They are only connected by Protocal TCP/P, and then the game can start.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ในการทำปัญหาพิเศษเรื่องเกมเศรษฐีระบบออนไลน์สามารถสำเร็จลุล่วงไปด้วยดี คณะผู้จัดทำต้องขอบพระคุณ อาจารย์กรกช ประชุมรักษ์ และอาจารย์สมศรี บัณฑิตวิไล อาจารย์ผู้รับผิดชอบปัญหาพิเศษนี้ที่กรุณาให้คำแนะนำและเป็นที่ปรึกษาในการแก้ปัญหาต่าง ๆ รวมทั้งเป็นผู้ตรวจสอบความถูกต้องของปัญหาพิเศษฉบับนี้

นอกจากนี้คณะผู้จัดทำต้องขอขอบพระคุณ บิดา มารดา และญาติพี่น้อง ที่ได้ให้ความสนับสนุนทางด้านกำลังใจและทุนทรัพย์ จนการทำปัญหาพิเศษครั้งนี้สำเร็จด้วยดี รวมทั้งเพื่อน ๆ และรุ่นพี่ทุกคนที่ให้ความช่วยเหลือในด้านต่าง ๆ เกี่ยวกับปัญหาพิเศษไว้ ณ ที่นี้

คณะผู้จัดทำ

มีนาคม 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I
บทคัดย่อภาษาอังกฤษ.....II
กิตติกรรมประกาศ.....III
สารบัญ.....IV
สารบัญภาพ.....VII
บทที่ 1 บทนำ.....1
1.1 ความเป็นมาและความสำคัญของปัญหา.....1
1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา.....1
1.3 สมมติฐานของการศึกษา.....1
1.4 ขอบเขตการศึกษา.....2
1.5 ขั้นตอนการศึกษา.....2
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง.....3
2.1 ทฤษฎีการเขียนโปรแกรมด้วยภาษา C/C++.....3
2.2 ทฤษฎีการเขียนโปรแกรมด้วย Microsoft Visual C++ 6.0.....4
2.3 ทฤษฎีการเชื่อมต่อผ่านระบบเครือข่าย.....7
2.4 ทฤษฎีอื่น ๆ ที่ได้แนวคิดมาจากปัญหาพิเศษก่อน ๆ.....13
บทที่ 3 วิธีดำเนินการปัญหาพิเศษ.....14
3.1 เทคนิควิธีการต่าง ๆ ของ Visual C++ ที่นำมาใช้ในปัญหาพิเศษ.....15
3.1.1 การใช้งานแอปวิซาร์ด.....15
3.1.2 การใช้งานไดอะล็อกอิดิตเตอร์.....17
3.1.3 วิธีการสร้างไดอะล็อกเพิ่มในการใช้งาน.....19
3.1.4 การใช้งานคลาสวิซาร์ด.....21
3.2 วิธีการนำเอารูปภาพแสดงในโปรแกรม.....23
3.3 วิธีการเชื่อมต่อทางระบบเครือข่ายที่ใช้.....28
3.4 วิธีการแสดงตัวเบี้ยในเกม.....31
3.5 การออกแบบโครงสร้างข้อมูลที่ใช้ในเกม.....32
3.6 วิธีการใช้เสียงประกอบในเกม.....37
3.6.1 การเล่นเสียง Wave.....37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

3.6.2 การเล่นเสียง MIDI Sequencer 38
3.7 เครื่องมือที่ใช้ในการเขียนโปรแกรม..... 39
3.7.1 เครื่องมือสำหรับทำภาพ และเสียงประกอบเกม..... 39
3.7.1.1 Adobe Photoshop 5.5..... 39
3.7.1.2 3D Studio MAX Release 3..... 39
3.7.1.3 Adobe Premiere 5.1..... 40
3.7.1.4 Ulead GIF Animator Version 4.0..... 41
3.7.2 Microsoft Visual C++ 6.0..... 42
3.8 Flow Chart of Monopoly Game Online..... 43
บทที่ 4 บทผลการทดลองหรือการวิเคราะห์ข้อมูล..... 44
4.1 โครงสร้างของ Algorithm ของเกม Monopoly..... 44
4.2 การทำงานของเกมในส่วนของการเล่นกับคอมพิวเตอร์..... 44
4.3 การออกแบบให้เกมสามารถเล่นบนระบบเครือข่ายคอมพิวเตอร์..... 44
4.4 การออกแบบกราฟฟิก และเสียงประกอบเกม..... 45
บทที่ 5 บทการวิจารณ์หรืออภิปรายผล..... 46
บทที่ 6 สรุปผลการศึกษาและข้อเสนอแนะ..... 48
6.1 สรุปผลการศึกษา..... 48
6.1.1 การศึกษารวบรวมข้อมูล..... 48
6.1.2 การวิเคราะห์และการออกแบบเกม..... 48
6.1.3 การพัฒนาโปรแกรมคอมพิวเตอร์..... 48
6.1.4 คุณสมบัติของเกมมีดังต่อไปนี้..... 48
6.2 ข้อเสนอแนะ..... 49
ภาคผนวก ก. กฎกติกาและส่วนประกอบของเกม..... 51
1 ระบบที่ใช้..... 52
2 กฎกติกาเบื้องต้น..... 52
3 ส่วนประกอบของเกม..... 52
3.1 กระดาน..... 52
3.2 ตัวเบี้ยหรือตัวเดิน..... 53
3.3 สิ่งปลูกสร้าง..... 53
3.4 โฉนดที่ดิน..... 54

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

3.5 บัตร Community และบัตร Chance 54
4 หน้าจอแสดงผล และปุ่มต่างๆที่ใช้ในการเล่น 55
4.1 หน้าจอหลัก 55
4.2 หน้าจอจัดการทรัพย์สินส่วนตัว 56
4.3 หน้าจอเสนอข้อตกลง 56
4.4 หน้าจอพิจารณาข้อตกลงที่เสนอมา 57
4.5 หน้าจอซื้อที่ดิน 58
4.6 หน้าจอเมื่อไม่มีเงินพอชำระหนี้ 58
4.7 หน้าจอเลือกหนทางออกจากคุก 59
5 วิธีการเล่น 59
5.1 คอมพิวเตอร์เครื่องเดียว 60
5.2 คอมพิวเตอร์หลายเครื่องโดยเล่นผ่านโปรโตคอล TCP/IP 60
5.2.1 ผู้เล่นหลัก 61
5.2.2 ผู้เล่นรอง 61
ภาคผนวก ข. กฎกติกาการเล่นเกม 62

บรรณานุกรม 66

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

รูปที่		หน้า
2.1	Flow Chart Connection-Oriented data transfer with Socket	8
2.2	Flow Chart Connectionless data transfer with Sockets	9
3.1	แสดง Chart แสดงขั้นตอนการทำปัญหาพิเศษ	14
3.2	แสดงการสร้างโปรเจคใหม่	15
3.3	แสดงการเลือกโปรแกรมแบบ Dialog based	16
3.4	แสดงการเลือก 3D controls และ Windows Sockets	16
3.5	แสดงโปรเจ็คที่สร้างจากแอปวิซาร์ดแบบ Dialog based	17
3.6	แสดงการกำหนดค่าไอดีใหม่ให้กับคอนโทรล	18
3.7	ตัวอย่างการใช้ไดอะล็อกอิดิเตอร์ปรับแต่งสร้างไดอะล็อก	18
3.8	แสดงการถามว่าจะสร้างคลาสให้กับรีซอร์สใหม่หรือไม่	19
3.9	แสดงการตั้งชื่อให้กับคลาสของรีซอร์ดใหม่	19
3.10	แสดงรูปไดอะล็อกที่จะแสดงถัดไป	20
3.11	แสดงการแมปแมสเสจให้กับปุ่ม	21
3.12	แสดงการใช้คลาสวิซาร์ดในกระบวนการ DDX และ DDV	22
3.13	แสดงการเลือกว่าใช้ตัวแปรทำอะไรและตั้งชื่อให้กับตัวแปร	23
3.14	แสดงการกำหนดชื่อ และประเภทให้กับคลาสใหม่	24
3.15	แสดงการเลือก Import Bitmap เข้ามาเป็นทรัพยากร	26
3.16	แสดงการเลือกรูปภาพที่จะ Import เข้ามา	27
3.17	แสดงการกำหนดไอดีเพื่อใช้เรียกแทนชื่อในโปรแกรม	27
3.18	แสดง step 2 of 4 ตามข้อ 1	29
3.19	แสดงการสร้างคลาสใหม่	29
3.20	แสดงการเลือก Base Class เป็น CSocket	30
3.21	แสดงรูป Object Model ของโปรแกรม	37
3.22	โปรแกรม Adobe Photoshop 5.5 ขณะกำลังตกแต่งและออกแบบภาพ	39
3.23	โปรแกรม 3D Studio MAX Release 3 ขณะทำการ Render ภาพ เป็นไฟล์นามสกุล avi	40
3.24	โปรแกรม Adobe Premiere 5.1 ขณะทำการตัดต่อเสียงเข้ากับภาพ	41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ **หน้า**

3.25 โปรแกรม Ulead GIF Animator Version 4.0 ขณะจะทำการ Export เป็นนามสกุล avi 41
3.26 Flow Chart of MONOPOLY GAME ONLINE 43
7.1 กระดานเกม 53
7.2 ตัวเบี้ยหรือตัวเดิน 53
7.3 บ้าน จำนวน 1, 2, 3 และ 4 หลังตามลำดับ 54
7.4 โรงแรม 54
7.5 ตัวอย่างโฉนดที่ดิน 54
7.6 ตัวอย่างบัตร Community และบัตร Chance 55
7.7 หน้าจอหลัก 55
7.8 หน้าจอจัดการกับทรัพย์สินส่วนตัว 56
7.9 หน้าจอเสนอข้อตกลง 57
7.10 หน้าจอพิจารณาข้อตกลงที่เสนอมา 57
7.11 หน้าจอซื้อที่ดิน 58
7.12 หน้าจอเมื่อไม่มีเงินพอชำระหนี้ 58
7.13 หน้าจอเลือกหนทางการออกจากคุก 59
7.14 หน้าเมนูเริ่มต้นโปรแกรมเกม 59
7.15 เมนู Single Computer 60
7.16 เมนู TCP/IP 60

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เนื่องจากปัจจุบันคอมพิวเตอร์ได้เข้ามาเป็นส่วนหนึ่งของชีวิตประจำวันเพิ่มมากขึ้น และเกมคอมพิวเตอร์ก็เป็นที่นิยมเล่นกันแพร่หลาย และเกมประเภทที่เล่นได้พร้อมกันทีละหลายคน เป็นประเภทของเกมที่ได้รับความนิยมมาก เพราะได้เล่นกับผู้เล่นอื่น ๆ โดยที่ไม่จำเป็นต้องอยู่ที่เดียวกัน ทางผู้จัดทำเห็นว่าเป็นเรื่องที่น่าสนใจจึงนำเกมเศรษฐีซึ่งเป็นเกมที่หลาย ๆ คนรู้จักและนิยมเล่นกันมาพัฒนาเป็นเกมคอมพิวเตอร์และสามารถเล่นผ่านระบบออนไลน์ได้โดยผ่านโปรโตคอล TCP/IP นำมาเสนอเป็นปัญหาพิเศษนี้

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

- เพื่อสามารถพัฒนาการเขียนโปรแกรมเกมโดยใช้โปรแกรม Microsoft Visual C++ 6.0
- เพื่อศึกษาการเขียนโปรแกรมเชิงวัตถุ
- ศึกษาการเชื่อมต่อของระบบเครือข่ายเพื่อนำมาพัฒนาเกมให้สามารถเล่นผ่านระบบเครือข่ายได้
- สามารถใช้โปรแกรมสำเร็จรูปต่าง ๆ เพื่อการออกแบบกราฟฟิก ภาพเคลื่อนไหวและเสียงประกอบเกมเพื่อเพิ่มอรรถรสในการเล่นเกมได้

1.3 สมมติฐานของการศึกษา

- ใช้โปรแกรม Microsoft Visual C++ 6.0 ในการเขียนโปรแกรมทั้งหมดของเกมนี้
- ใช้ทฤษฎีภาษา C/C++ เป็นพื้นฐานในการเขียนโปรแกรมด้วย Microsoft Visual C++ 6.0
- วิธีการใช้คลาส CSocket ของ Microsoft Foundation Class นำมาเขียนโปรแกรมในการส่งผ่านข้อมูลผ่านระบบเครือข่าย ด้วยโปรโตคอล TCP/IP
- นำเอาโปรแกรมสำเร็จรูปต่างๆ เช่น Adobe Photoshop 5.5, 3D Studio MAX, Adobe Premiere, Ulead GIF Animator เป็นต้น ในการจัดการกับการทำภาพกราฟิก ภาพเคลื่อนไหว และเสียงประกอบเกม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ขอบเขตการศึกษา

- เป็นเกมที่เล่นบนระบบปฏิบัติการ Windows 95/98
- เป็นเกมที่สามารถเล่นพร้อมกันได้ไม่เกิน 4 คน ผ่านระบบเครือข่าย ที่เชื่อมต่อกันด้วยโปรโตคอล TCP/IP
- ผู้เล่นสามารถเลือกเล่นกับคอมพิวเตอร์ได้
- มีภาพกราฟฟิก ภาพเคลื่อนไหว และเสียงประกอบเกม
- สามารถใส่ชื่อผู้เล่น และเลือกลักษณะของตัวเบี้ยเดินได้
- ผู้เล่นสามารถติดต่อโดยการพิมพ์ข้อความส่งให้กันได้

1.5 ขั้นตอนการศึกษา

2 มิ.ย. – 15 มิ.ย. ศึกษารูปแบบและแนวทางของเกม
16 มิ.ย. – 31 ส.ค. ออกแบบและวางโครงร่างเกม
1 ก.ย. – 30 ก.ย. ศึกษาและทดลองโปรแกรม Visual C++
1 ต.ย. – 31 ต.ค. ออกแบบภาพ Animation และรูปภาพประกอบ
1 พ.ย. – 15 พ.ย. ศึกษาวิธีเชื่อมต่อคอมพิวเตอร์ทาง Network
16 พ.ย. – 15 ม.ค. เขียนโปรแกรม
16 ม.ค. – 15 ก.พ. ทดสอบและแก้ไขข้อผิดพลาด
16 ก.พ. – 25 ก.พ. สรุปโครงงานและจัดทำเอกสารประกอบปัญหาพิเศษ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีการเขียนโปรแกรมด้วยภาษา C/C++

ในส่วนนี้สำคัญมากต่อการจัดทำปัญหาพิเศษนี้ เพราะการเขียนโปรแกรมทุกส่วนในปัญหาพิเศษนี้ใช้ Microsoft Visual C++ 6.0 ในการพัฒนาโปรแกรม ดังนั้นการศึกษาการเขียนโปรแกรมด้วยภาษา C/C++ จึงเป็นส่วนสำคัญ โดยสามารถที่จะศึกษาและเรียนรู้ทฤษฎี ตัวอย่างโจทย์ปัญหาต่าง ๆ ได้จากตำราภาษา C/C++ ทั่วไป ทั้งตำราภาษาไทยและภาษาอื่น ๆ ก็มีให้ศึกษาอย่างแพร่หลาย

ภาษา C เป็นภาษาที่เป็นโครงสร้าง และใช้ได้กับงานทั่วไป คำสั่งของภาษา C จะประกอบด้วยพจน์ซึ่งจะมีลักษณะเหมือนกับนิพจน์ทางพีชคณิต และมีส่วนขยายเป็นคำหลัก ในภาษาอังกฤษ เช่น if else for do และ while ดังนั้นภาษา C จึงมีส่วนคล้ายกับภาษาระดับสูงที่มีลักษณะเป็นโครงสร้างภาษาอื่น ๆ เช่น ปาสกาล และ ฟอร์แทรน 77 แต่ภาษา C ก็มีคุณสมบัติพิเศษเพิ่มขึ้นมา นั่นคือมันสามารถใช้งานในระดับต่ำได้ ดังนั้นมันจึงเปรียบเหมือนสะพานเชื่อมภาษาเครื่องเข้ากับภาษาระดับสูง จากความอ่อนตัวนี้เองทำให้ภาษา C สามารถใช้กับงานด้านโปรแกรมระบบ เช่น เขียนโปรแกรมระบบปฏิบัติการ หรือใช้กับงานทั่ว ๆ ไป เช่น เขียนโปรแกรมเพื่อออกใบเสร็จให้กับลูกค้า เป็นต้น

เนื่องจากภาษา C มีตัวดำเนินการจำนวนมาก ดังนั้นเราสามารถเขียนโปรแกรมภาษา C ให้มีความกระชับได้ ภาษา C มีคำสั่งน้อยมาก ดังนั้นในการใช้งานจริงจะต้องมีการติดตั้งไลบรารีฟังก์ชัน เพื่อเพิ่มเติมจากคำสั่งพื้นฐานเหล่านั้น ยิ่งไปกว่านั้นภาษา C ยังยอมให้ผู้ใช้สร้างไลบรารีฟังก์ชันของตัวเองได้ ดังนั้นผู้ใช้สามารถที่จะเพิ่มเติมคุณสมบัติ และความสามารถของภาษาได้โดยง่าย

โดยทั่วไปเราจะพบตัวแปลโปรแกรมภาษา C ในคอมพิวเตอร์ทุกขนาด และตัวแปลคำสั่งภาษา C ก็เริ่มมีมากขึ้น ตัวแปลโปรแกรมภาษา C โดยทั่วไปจะมีขนาดเล็ก และจะสร้างโปรแกรมประยุกต์ที่มีขนาดเล็ก และมีประสิทธิภาพสูง เมื่อเปรียบเทียบกับโปรแกรมประยุกต์ที่ได้จากตัวแปลโปรแกรมของภาษาระดับสูงอื่น ๆ ส่วนตัวแปลคำสั่งของภาษา C นั้น จะมีประสิทธิภาพด้อยกว่า แต่จะใช้พัฒนาโปรแกรมได้ง่ายกว่า ดังนั้นนักเขียนโปรแกรมหลายคนจึงเริ่มต้นพัฒนาโปรแกรมโดยใช้ตัวแปลคำสั่ง และเมื่อโปรแกรมไม่มีข้อผิดพลาดแล้ว จึงเปลี่ยนมาใช้ตัวแปลโปรแกรม

คุณสมบัติที่สำคัญอีกประการหนึ่งของภาษา C ก็คือ โปรแกรมภาษา C สามารถย้ายไปทำงานในเครื่องอื่นได้ง่ายกว่าภาษาระดับสูงอื่น ๆ ที่เป็นเช่นนี้เพราะภาษา C ได้แยกส่วนที่ขึ้นอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กับเครื่องคอมพิวเตอร์ ไปเป็นไลบรารีฟังก์ชัน ดังนั้นโปรแกรมภาษา C ทุกๆโปรแกรม ก็ทำงานโดยเรียกใช้ฟังก์ชันจากไลบรารีฟังก์ชัน ซึ่งเขียนขึ้นมาสำหรับคอมพิวเตอร์แต่ละเครื่อง ฟังก์ชันที่อยู่ในไลบรารีฟังก์ชันจะเป็นฟังก์ชันมาตรฐาน และมีวิธีการเรียกใช้งานแบบเดียวกัน ดังนั้นโปรแกรมภาษา C ทั้งหมดจึงสามารถนำมาใช้งานบนเครื่องคอมพิวเตอร์ที่แตกต่างกันได้ โดยแก้ไขโปรแกรมเพียงเล็กน้อย หรืออาจจะไม่ต้องแก้ไขเลย

ภาษา C++ ถูกสร้างขึ้นมาบนพื้นฐานของภาษา C ซึ่งเป็นภาษาสำหรับการเขียนโปรแกรมเชิงวัตถุ โดยมีการใช้งานคลาส เทมเพลต และส่วนประกอบอื่น ๆ เพิ่มเติมเข้ามานอกเหนือจากที่มีในภาษา C แต่ภาษา C++ ก็ไม่ใช่ภาษาสำหรับการเขียนโปรแกรมเชิงวัตถุที่สมบูรณ์แบบหรือมีคุณสมบัติครบทุกประการของวิธีการเชิงวัตถุ ดังนั้นจึงมีความจำเป็นที่จะต้องศึกษาการเขียนโปรแกรมเชิงวัตถุด้วย โดยการเขียนโปรแกรมเชิงวัตถุเป็นวิธีในการพัฒนาและจัดระเบียบการเขียนโปรแกรมอีกรูปแบบหนึ่ง ที่มององค์ประกอบต่างๆให้เป็นวัตถุ โดยแต่ละวัตถุก็จะมีคุณสมบัติและวิธีการใช้คุณสมบัติที่แตกต่างกันไป แต่สามารถนำมาใช้ร่วมกันได้ง่าย นอกจากนั้นก็ยังมีคุณสมบัติอื่น ๆ อีก ที่จะต้องศึกษา อีกทั้งการเขียนโปรแกรมด้วย Microsoft Visual C++ 6.0 มีความจำเป็นที่จะต้องใช้ประโยชน์จาก Microsoft Foundation Class ซึ่งเป็นไลบรารีที่มีโครงสร้างเป็นแบบเชิงวัตถุ การเข้าใจกับการเขียนโปรแกรมเชิงวัตถุ ก็จะทำให้การทำความเข้าใจและใช้งาน Microsoft Foundation Class ได้ง่ายขึ้นด้วย

สิ่งหนึ่งที่มีความสำคัญในการใช้ทฤษฎีของภาษา C/C++ คือการออกแบบโครงสร้างข้อมูลที่ใช้ในการเขียนโปรแกรม เราจำเป็นจะต้องมีการศึกษาถึงโครงสร้างข้อมูลแบบต่าง ๆ ในภาษา C/C++ ให้ดีเสียก่อนจึงจะทำการออกแบบได้อย่างมีประสิทธิภาพ มีการใช้งานที่ง่าย และง่ายต่อการตรวจสอบแก้ไขปัญหา โดยเฉพาะการเขียนคลาสขึ้นมาใช้งาน ที่มีความเหมาะสมมากในการรวบรวมข้อมูลต่าง ๆ ที่เกี่ยวข้องกัน หรืออยู่ในคุณสมบัติของการกระทำการหนึ่ง ๆ มารวมอยู่ในคลาสเดียวกัน

2.2 ทฤษฎีการเขียนโปรแกรมด้วย Microsoft Visual C++ 6.0

เราได้ใช้ Microsoft Visual C++ 6.0 ในการเขียนโปรแกรม และพัฒนาโปรแกรม ดังนั้นเราจึงต้องเขียนโปรแกรมบนระบบปฏิบัติการ Windows 95 หรือ Windows 98 ซึ่งเป็นระบบปฏิบัติการแบบกราฟฟิกที่อำนวยความสะดวกในการใช้งานได้เป็นอย่างมาก ทำให้ง่ายต่อการใช้งานและการติดต่อกับผู้ใช้ เช่น ปุ่มกด, ช่องรับข้อความ, เช็คบ็อกซ์ เป็นต้น เราจึงควรที่จะต้องศึกษาวิธีการสร้าง และควบคุมระบบ Interface ของวินโดวส์ เช่น การศึกษาวิธีการสร้างวินโดวส์, การสร้างปุ่มกด, เท็กซ์บ็อกซ์ และลิสต์บ็อกซ์ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Visual C++ เป็นโปรแกรมประเภท Visual นั่นคือรูปแบบการพัฒนาโปรแกรมจะอยู่ในลักษณะของ WYSIWYG (What you see is what you get) Microsoft Visual C++ ได้รับการพัฒนาให้มีความยืดหยุ่น และมีประสิทธิภาพสูงขึ้นมาจากภาษา C++ และได้สนับสนุนการพัฒนาโปรแกรมในหลาย ๆ ด้าน ไม่ว่าจะเป็นการสร้างโปรแกรมทั่ว ๆ ไป, การสร้างโปรแกรมจัดการฐานข้อมูล, การสร้างโปรแกรมบนระบบเครือข่าย หรือการสร้างโปรแกรมมัลติมิเดียอย่างครบครัน และได้พัฒนามาจนถึงเวอร์ชัน 6.0 ในปัจจุบัน โดยเป็นเครื่องมือตัวหนึ่งที่ใช้ในการพัฒนาโปรแกรมในชุด Microsoft Visual Studio 98 (ซึ่งประกอบไปด้วย Microsoft Visual C++ 6.0, Microsoft Visual Basic 6.0, Microsoft Visual Foxpro 6.0, Microsoft Visual InterDev 6.0, Microsoft Visual J++ 6.0, Microsoft Visual SourceSafe 6.0 และ Microsoft Developer Network Library)

Visual C++ ได้รับการพัฒนาขึ้นมาจาก Microsoft C/C++ ให้เป็น IDE (Intergrated Development Environtment คือเป็นโปรแกรมซึ่งมีไว้ใช้สำหรับเพิ่มความสะดวกในการสร้างและแก้ไขโปรเจคให้ง่ายขึ้น โดยมีเครื่องอำนวยความสะดวกในการพัฒนาโปรแกรมต่าง ๆ ให้เรียกใช้ภายใน IDE เช่น โปรแกรมที่ใช้ในการดีบัก, คอมไพเลอร์ และลิงค์เกอร์ เป็นต้น) โดยมี MFC (Microsoft Foundation Class) เป็นไลบรารีที่จะช่วยอำนวยความสะดวกในการพัฒนาโปรแกรมบนวินโดวส์ โดยในยุคที่ MFC ยังไม่ถือกำเนิดนั้นเราใช้ SDK (Software Development Kit) และคอมไพลเลอร์ภาษา C เช่น Microsoft C++, Borland C++ ช่วยในการเขียนโปรแกรม ซึ่งโค้ดที่เขียนขึ้นด้วย SDK นั้นค่อนข้างจะซับซ้อนและสร้างความลำบากในการศึกษาทำความเข้าใจ จึงได้มีการสร้างคลาสชุดหนึ่งขึ้น เขียนโดยใช้โครงสร้างของ OOP (Objected Oriented Programming) ด้วยภาษา C++ ชื่อว่า MFC โดยโค้ดที่เขียนโดยใช้ MFC นั้นจะมีขนาดเล็กและไม่ซับซ้อน ง่ายกว่าการใช้ SDK ซึ่งในโปรเจคของเรานั้นได้ใช้ MFC ในการพัฒนาโปรแกรมด้วย โดย Microsoft Visual C++ นั้นพร้อมรองรับการพัฒนาข้ามแพลตฟอร์ม หลายแพลตฟอร์มด้วย (Multiplatform) และมีวิซาร์ด (Wizard) เป็นเครื่องมือให้ใช้ในการเขียนโปรแกรมด้วยกัน 2 อย่างคือ

แอปวิซาร์ด (AppWizard) เป็นอุปกรณ์ตัวหนึ่งที่ใช้สำหรับสร้างซอร์สโค้ดโปรแกรมต้นแบบซึ่งจะช่วยอำนวยความสะดวกให้เราอย่างมาก โดย แอปวิซาร์ดจะถามเกี่ยวกับรูปแบบของโปรแกรม, ตัวเลือกต่าง ๆ ที่ต้องการ แล้วแอปวิซาร์ดก็จะสร้างซอร์ดโค้ดโปรแกรมขึ้นมาโดยใช้คลาสจาก MFC เป็นหลัก และสามารถนำซอร์ดโค้ดนั้นไปใช้เพื่อเพิ่มเติมส่วนต่าง ๆ ที่ต้องการเพิ่มที่ไม่สามารถสร้างจากแอปวิซาร์ดได้ทันที โดยแอปวิซาร์ดมีตัวเลือกรูปแบบของโปรแกรมหลักๆอยู่ 3 แบบคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) โปรแกรมแบบ SDI (Single Document Interface) เป็นการสร้างโปรแกรมในลักษณะของ Notepad
2) โปรแกรมแบบ MDI (Multiple Document Interface) เป็นการสร้างโปรแกรมที่มีหลาย ๆ หน้าต่างคล้าย ๆ กับ Microsoft Word
3) โปรแกรมแบบ Dialog based เป็นการสร้างโปรแกรมในลักษณะแบบไดอะล็อก เช่น Scandisk, Defrag ใน Windows 95

คลาสวิซาร์ด (ClassWizard) เป็นอุปกรณ์ที่จะช่วยอำนวยความสะดวกในการจัดการคลาส, การแมปแมสเสจ, การสร้างตัวแปรต่าง ๆ ในกระบวนการ DDX (Dialog Data Exchange) ที่เป็นกระบวนการแลกเปลี่ยนข้อมูลในไดอะล็อก และกระบวนการ DDV (Dialog Data Validation) ที่เป็นกระบวนการที่จะใช้ตรวจสอบข้อมูลในไดอะล็อก การใช้คลาสวิซาร์ดแมปแมสเสจให้กับคอนโทรลนั้นจะทำได้ง่ายกว่าการเขียนโค้ดด้วยตัวเองมาก อีกทั้งยังรวดเร็วกว่าด้วย โดยการเรียกใช้คลาสวิซาร์ดจะสามารถใช้ได้กับโค้ดที่สร้างขึ้นโดยแอปวิซาร์ดเท่านั้น

อีกส่วนหนึ่งที่น่าสนใจใน Microsoft Visual C++ คือ **ดีบักเกอร์** (Debugger) ใช้สำหรับช่วยให้สามารถแก้ไขข้อผิดพลาดทางตรรกะได้ ดีบักเกอร์นี้จะทำให้สามารถจับการทำงานของโปรแกรมได้ทุกขั้นตอน ในลักษณะที่ให้โปรแกรมรัน และหยุดที่ตำแหน่ง หรือสถานการณ์ที่เราเลือกได้ เราสามารถเซตจุดหยุดโปรแกรม (breakpoint) ไว้ที่บรรทัดใดบรรทัดหนึ่งของโค้ด และให้แอปพริเคชั่นของเรารันจนถึงบรรทัดนั้น หรือสามารถให้แอปพริเคชั่นหยุดทำงานไว้ก่อน เมื่อได้รับแมสเสจของวินโดวส์ที่กำหนดไว้ หรือเมื่อเกิดข้อผิดพลาดที่กำหนดไว้ และถ้าเราสนใจค่าของตัวแปรที่เกิดขึ้น เราก็สามารถให้ดีบักเกอร์หยุดไว้ให้แอปพริเคชั่นเปลี่ยนค่าตัวแปรได้เช่นกัน โดยดีบักเกอร์ยังสามารถเริ่มรันสำหรับโปรแกรมใดที่ทำงานผิดพลาดอยู่ โดยที่โปรแกรมนั้นจะได้ทำการดีบักข้อมูลอยู่หรือไม่ก็ตาม หรือ Visual C++ จะรันอยู่ก่อนหรือไม่ก็ได้ ดีบักเกอร์จะรันขึ้นในขณะที่โปรแกรมยังทำงานอยู่ ซึ่งสิ่งนี้เรียกว่าการดีบักแบบ Just-In-Time ซึ่งทำให้เราสามารถวิเคราะห์โปรแกรมที่ยังทำงานอยู่ได้ แทนที่จะต้องไปตรวจสอบหลังจากที่มันเลิกทำงานไปแล้ว โดยที่ดีบักแบบ Just-In-Time ทำให้เป็นไปได้ที่จะหา และแก้ไขปัญหาที่เกิดขึ้นในโปรแกรม และทำให้มันยังคงรันต่อไปได้

โดยในที่นี้เราจะพูดถึงทฤษฎีและวิธีการใช้งานต่าง ๆ ของ Microsoft Visual C++ 6.0 ที่เราใช้ในปัญหาพิเศษนี้เท่านั้น ซึ่งจะกล่าวถึงต่อไปในส่วนเครื่องมือและวิธีการในบทที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ทฤษฎีการเชื่อมต่อผ่านระบบเครือข่าย

Socket คือ Abstract Class ที่สามารถเชื่อมต่อกับคอมพิวเตอร์และรับ-ส่งข้อมูลผ่าน network ได้ 2 ทิศทาง (Bidirectional) โดยผ่าน Socket Socket ประกอบไปด้วย 3 attribute คือ

- Network address (หรือคือ IP ใน TCP/IP)
- Port Number คือ หมายเลขของ Process ที่ใช้ในการแลกเปลี่ยนข้อมูลผ่าน Socket
- Socket Type คือ รูปแบบของ Protocol ที่ใช้ในการแลกเปลี่ยนข้อมูล (Data Stream หรือ Datagram)

ดังนั้น IP address จึงเป็นตัวแทนของ network node port number จะเป็นตัวแทนของแต่ละ Process ที่รับส่งผ่าน Socket บนคอมพิวเตอร์ และ socket type จะหมายถึงรูปแบบของการแลกเปลี่ยนข้อมูลผ่านระบบ network (Connection Oriented หรือ Connectionless Service)

Connection-Oriented Service

Connection-Oriented data exchanged คือรูปแบบของการแลกเปลี่ยนข้อมูลที่ทั้งฝั่งของผู้รับ (receiver) และผู้ส่ง (sender) จำเป็นต้องสร้างการเชื่อมต่อกันก่อน (establish connection) ที่จะมีการแลกเปลี่ยนข้อมูลกัน ในโปรโตคอล TCP/IP TCP โปรโตคอลได้จัดหาบริการแบบ Connection-Oriented ที่จะรับส่งข้อมูลกันในแบบ Byte Stream

TCP มีประโยชน์อย่างมากสำหรับ application ที่ต้องการแลกเปลี่ยนข้อมูลขนาดใหญ่ในเวลาจำกัด สำหรับใน socket แล้ว socket จะใช้โปรโตคอล TCP ได้โดยการ set attribute ของ socket เป็น Stream socket

Connectionless Service

Connection-Oriented data exchanged คือรูปแบบของการแลกเปลี่ยนข้อมูลที่ทั้งฝั่งของผู้รับ (receiver) และผู้ส่ง (sender) ไม่จำเป็นต้องสร้างการเชื่อมต่อกันก่อน (establish connection) ที่จะมีการแลกเปลี่ยนข้อมูลกัน ในโปรโตคอล TCP/IP มีโปรโตคอลอีกตัวหนึ่งที่ชื่อว่า User Datagram Protocol (UDP) โปรโตคอล ที่สามารถจัดหาบริการแบบ Connectionless ที่จะรับส่งข้อมูลกันในแบบ datagram ซึ่งแตกต่างกับ TCP UDP จะไม่การันตีว่า datagram จะถูกส่งไปถึงผู้รับหรือไม่ แต่ UDP จะรับประกันได้ว่า datagram ได้ถูกส่งออกไปตามคำสั่งของผู้ส่งแน่นอน

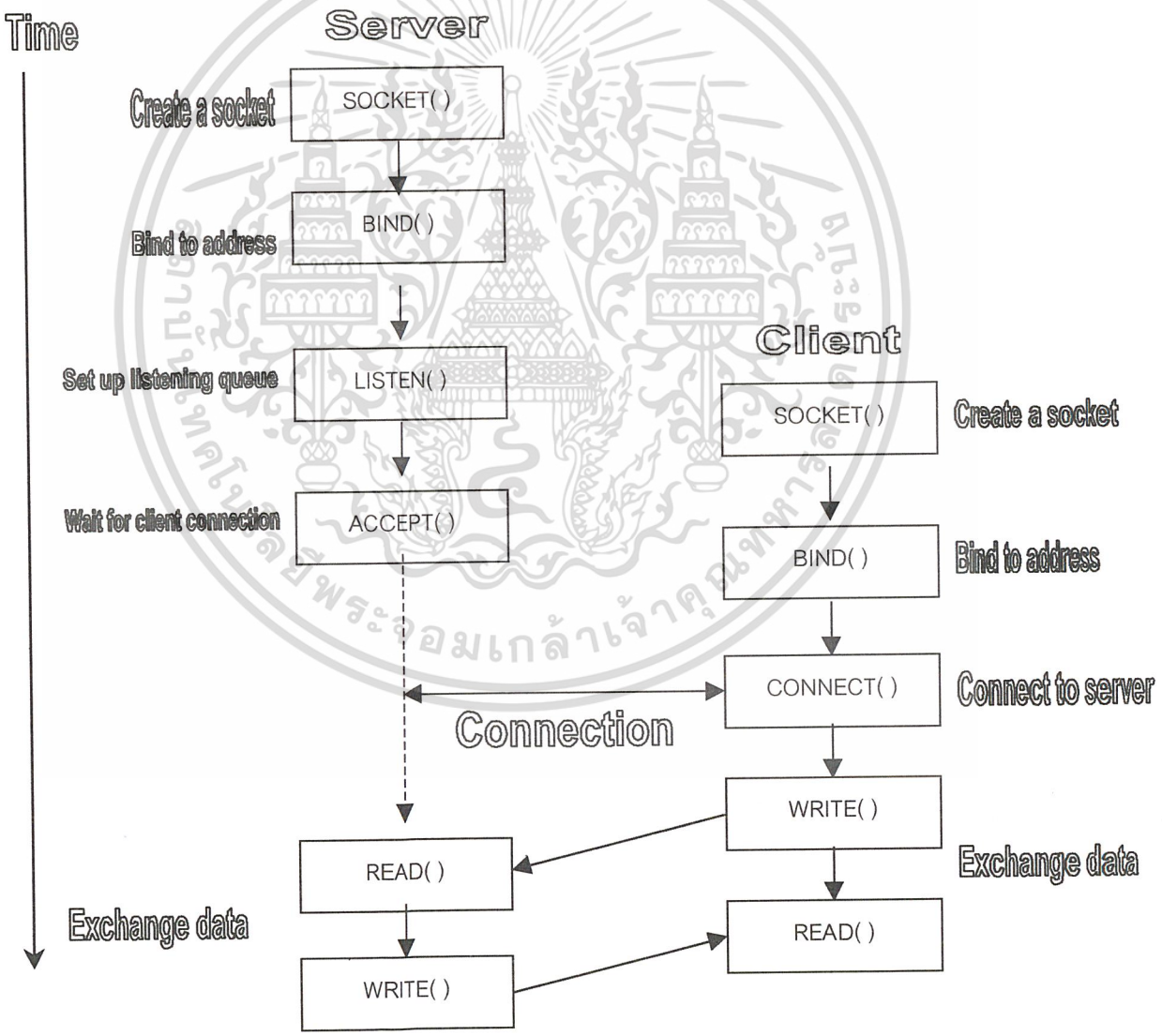
UDP มีประโยชน์อย่างมากสำหรับ application ที่ต้องการแลกเปลี่ยนข้อมูลขนาดเล็ก สำหรับใน socket แล้ว socket จะใช้โปรโตคอล UDP ได้โดยการ set attribute ของ socket เป็น Datagram socket

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Socket and Client-Server Model

เมื่อมีการสื่อสารกันระหว่าง Client กับ Server เกิดขึ้น ทั้ง 2 ฝั่งจำเป็นจะต้องใช้ Socket ของแต่ละฝั่งในการสื่อสารกัน ในฝั่ง Server Application จะต้องทำการเฝ้าฟัง (Listen) อยู่บน IP Address ของตัวเอง เพื่อรอให้ Client เข้ามาสร้างการ connection กับ Server ทางฝั่ง Client จะต้องทำการเปิด port และ bind IP Address ของตนเองก่อน แล้วจึงพยายามที่จะ connect มาที่ Server (โดย Client จะติดต่อผ่าน IP Address) ทันทีที่การเชื่อมต่อสำเร็จ ทั้ง Client และ Server จึงจะสามารถแลกเปลี่ยนข้อมูลกันได้

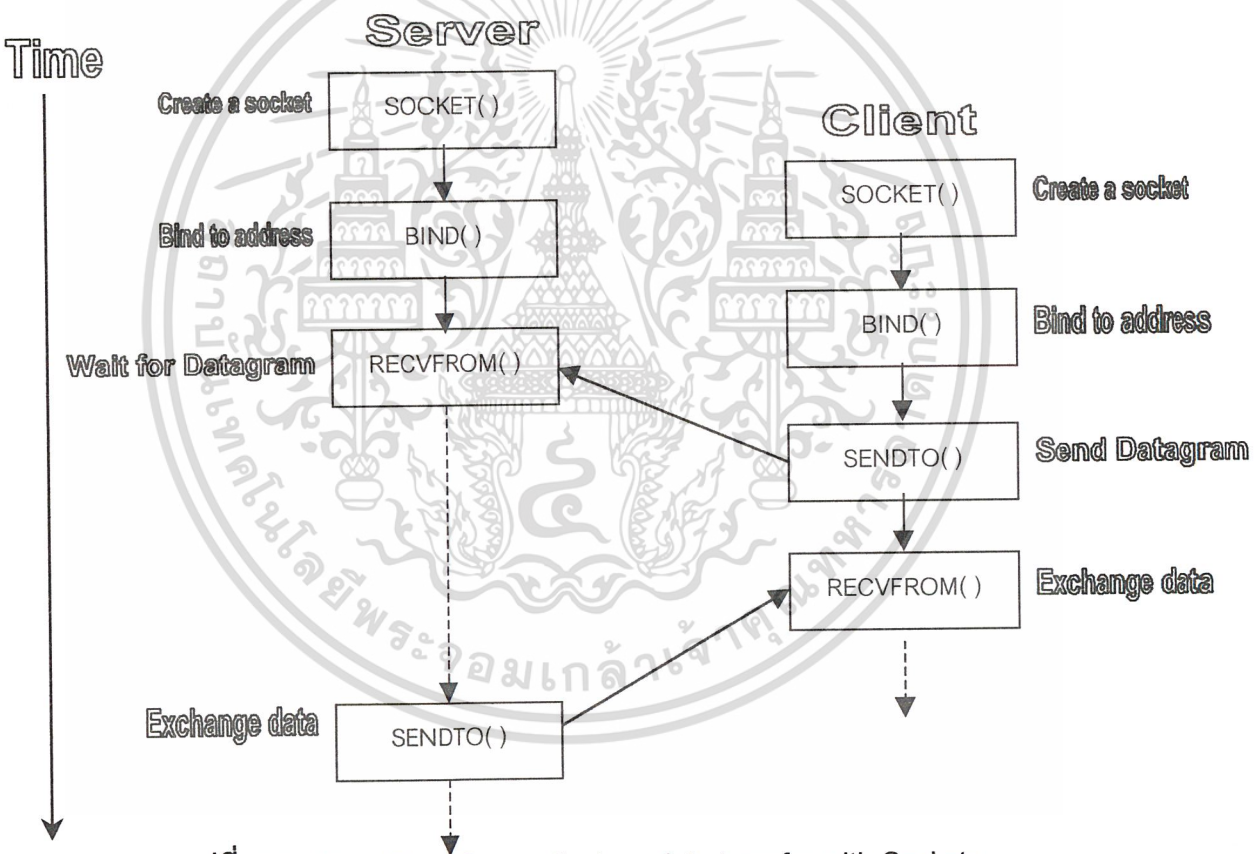
ลำดับของเหตุการณ์ในการสื่อสารกันของ Client-Server ผ่าน Socket จะขึ้นอยู่กับว่า ทั้ง Client และ Server นั้นติดต่อกันในรูปแบบใด (Connection-Oriented หรือ Connectionless)



รูปที่ 2.1 Flow Chart Connection-Oriented data transfer with Socket

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังรูปจะแสดงถึงลำดับของเหตุการณ์ของการติดต่อแบบ Connection-Oriented โดยใช้ Socket แต่ละขั้นตอนที่แสดงอยู่จะแสดงชื่อของ Socket Function ที่กำลังถูกเรียกใช้งาน โดยที่การติดต่อแบบนี้จะเริ่มต้นโดย Server ก่อน โดยที่ Server จะต้องสร้าง Socket ขึ้นมาจากนั้นจึงทำการกำหนดที่อยู่ของตนเอง (Bind Socket)โดยกำหนด IP Address และกำหนด Port Number (หมายเลขของแต่ละ Process เพื่อที่ว่าเมื่อมี Client อื่นมาติดต่ออีก Server ก็จะทำการกำหนด port No. ของ process ใหม่สำหรับ Client นั้น เพื่อให้ Server สามารถจัดการ Client ได้ทีละหลาย ๆ Client) จากนั้น Server ก็จะทำการเฝ้าฟัง (Listen) เพื่อรอคอยให้ Client เข้ามาติดต่อ เมื่อ Client เข้ามาติดต่อ (Connect) server ก็จะยอมรับการติดต่อนั้น (Accept) แล้วจึงสร้างการเชื่อมต่อขึ้นมา ดังนั้นทั้ง 2 ฝั่งจึงจะสามารถแลกเปลี่ยนข้อมูลกันได้



รูปที่ 2.2 Flow Chart Connectionless data transfer with Sockets

สำหรับการสื่อสารกันในแบบ Connectionless นั้น Server จะเพียงแต่รอให้ Datagram ส่งมาถึง Server เท่านั้น Server ไม่จำเป็นต้องเฝ้าฟัง (Listen) เพื่อรอให้ Client มาติดต่อก่อน แล้วจึงรอรับ Datagram จาก Client ได้ ดังนั้นทาง Client ไม่จำเป็นต้องรอให้การเชื่อมต่อทั้ง 2 ฝั่งสำเร็จ แต่ Client สามารถส่งข้อมูล Datagram ไปให้ Server ได้เลยทั้ง Client และ Server จะใช้งาน Socket และรับส่ง Datagram โดยใช้งานฟังก์ชั่น sendto และ recvfrom

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการส่งและรับ Datagram ตามลำดับ เมื่อ Server ได้รับ Datagram โดยการเรียก recvfrom ฟังก์ชัน recvfrom จะ return Network Address(IP) ของ Client ที่ส่ง Datagram ให้กับ Server ด้วยเหตุนี้ Server จึงจะสามารถส่ง Acknowledge กลับไปบอก Client ได้ว่าการส่ง Datagram นั้นสำเร็จแล้ว

Type of the Windows Sockets Function

เช่นเดียวกับ Abstract class อื่น ๆ ที่จะถูกออกแบบมาให้มี Function การทำงานหลาย ๆ แบบ Windows Sockets จะประกอบไปด้วยฟังก์ชั่นการทำงานหลัก ๆ ด้วยกัน 3 แบบ

- Database Function ฟังก์ชั่นชนิดนี้จะช่วยให้เราสามารถเรียกดูข้อมูลแบบต่าง ๆได้(เช่น Host name, Service name, Network Address of Host ฯลฯ)ฟังก์ชั่นเหล่านี้จะมีการใช้งานอยู่บน Berkeley Socket ด้วยเช่นกัน
- Socket Function ฟังก์ชั่นชนิดนี้จะเป็นฟังก์ชั่นการทำงานหลัก ๆ ที่จะถูกนำมาใช้ในการสร้าง Connection ระหว่าง Client และ Server การส่งข้อมูล, การรับข้อมูล และการยกเลิกการใช้งาน Socket ฟังก์ชั่นเหล่านี้จะมีเหมือนกับฟังก์ชั่นของ Berkeley Socket
- Windows Sockets Extension Function ฟังก์ชั่นชนิดนี้จะมีเฉพาะบน Windows Sockets เท่านั้น Extension Function ส่วนใหญ่ที่ถูกเพิ่มเข้ามานั้นจะถูกใช้งานร่วมกับ Socket Function

Data Structure

Windows Sockets จะประกาศ member data เหล่านี้ไว้ใน Class ของ Windows Sockets

- fd_set จะเก็บข้อมูลที่เกี่ยวกับ Socket
- hostent จะเก็บข้อมูลที่เกี่ยวกับ Host
- in_addr จะแสดง Internet address
- ip_mreg จะเป็น Argument สำหรับ IP_ADD_MEMBERSHIP และ IP_DROP_MEMBERSHIP
- linger จะถูกใช้งานในการควบคุมการทำงานของ Socket หลังจากที่ Application ได้ Close Socket ไปแล้ว
- netent จะเก็บข้อมูลที่เกี่ยวกับ network
- protoent จะเก็บข้อมูลที่เกี่ยวกับ Protocol เช่น TCP หรือ UDP
- servent จะเก็บข้อมูลที่เกี่ยวกับ service และ port number ของ service นั้น ๆ
- sockaddr เป็น General Structure สำหรับใช้ในการเก็บ Endpoint Address ที่จะไป connect ผ่าน socket

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- sockaddr_in จะถูกใช้ในการเก็บที่อยู่สำหรับ Internet address family ของ Endpoint
- sockproto จะเก็บข้อมูลที่เกี่ยวกับ โปรโตคอล สำหรับ SOCK_RAW
- timeval เป็น structure ที่เกี่ยวกับเวลา ถูกใช้เป็น Argument สำหรับเลือก Function
- WSAData จะเก็บข้อมูลที่เกี่ยวกับ Windows Sockets Implementation

แน่นอนว่ามี Member data เพียงไม่กี่ชนิดที่จะถูกใช้งานโดย Windows Sockets Application โดย Application ส่วนใหญ่จะใช้งาน sockaddr_in, hostent, protoent, servent, และ WSAData Structure เท่านั้น

Socket Function

Socket function เป็นอีกหนึ่งฟังก์ชั่นที่ Application จะเรียกใช้งานในการที่จะสร้าง Connection ระหว่าง Socket และแลกเปลี่ยนข้อมูลกัน ดังแสดงในตารางรายชื่อของ Socket ฟังก์ชั่นใน Windows Sockets ซึ่งฟังก์ชั่นเหล่านี้จะคล้าย ๆ กับ ฟังก์ชั่นของ Berkeley Socket

จะพบว่ามีฟังก์ชั่นที่ช่วยในการสร้าง connection และแลกเปลี่ยนข้อมูล เช่น htonl, htons, ntohl และ ntohs ซึ่งเป็นฟังก์ชั่นที่ใช้ในการเปลี่ยนแปลงข้อมูลระหว่าง Network Byte Order และ Host Byte Order

Byte Order คือ วิธีการที่ใช้ในการเก็บข้อมูล (เช่น ข้อมูลชนิด Integer) ลงบนหน่วยความจำ ซึ่งในหน่วยความจำนั้น เนื้อที่สำหรับที่อยู่หนึ่ง Address จะสามารถเก็บได้เพียง 1 byte ซึ่งถ้าข้อมูลมีขนาดหลาย byte ก็จะต้องมองลำดับของการเก็บข้อมูลเหล่านั้น ลงบนหน่วยความจำอีกด้วย ซึ่งรูปแบบของการเก็บข้อมูลลงหน่วยความจำมีอยู่ด้วยกัน 2 แบบคือ

1. Little-Endian Order ซึ่งทำงานอยู่บน Intel 80x86 และ Pentium Processor, ในลำดับของ byte ที่ต่ำกว่า(Low-Order Byte) จะถูกเก็บในหน่วยความจำในตำแหน่งที่อยู่ต่ำสุด (Lowest Address) และในลำดับของ byte ที่สูงสุด (highest-Order Byte) จะถูกเก็บในหน่วยความจำตำแหน่งที่อยู่สูงสุด (Highest Address)
2. Big-Endian Order ซึ่งทำงานอยู่บน Motorola processor จะกลับกันกับวิธีแรก คือ ในลำดับของ byte ที่อยู่สูงสุด (Highest-Order Byte) จะถูกเก็บในหน่วยความจำที่ตำแหน่งที่อยู่ต่ำสุด (Lowest Address)

บนระบบ network, TCP/IP Protocol จะใช้วิธี Big-Endian Byte Order ในการรับส่งข้อมูลผ่านระบบเครือข่าย Application ควรจะใช้งานฟังก์ชั่น (เช่น ntohl หรือ ntohs) เพื่อที่จะเปลี่ยนแปลงข้อมูลจาก network byte order ไปเป็น host byte order

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Socket Function in Windows Sockets

Function	Description
accept	ทำหน้าที่ในการส่ง Acknowledges ตอบกลับไปบอก Client, สร้าง Socket, และเปิด connection
bind	กำหนด local name ให้ Socket สำหรับ TCP/IP ชื่อประกอบด้วย 3 ส่วน: IP Address, Protocol Type และ Port Number
closesocket	ปิด Socket(Blocks ถ้า SO_LINGER ถูกเซต)
connect	สร้างการเชื่อมต่อระหว่าง Client กับ Server
getpeername	เรียกข้อมูล name of peer connection เพื่อที่จะ assign Socket
getsockname	เรียกข้อมูล ชื่อของ specified socket
getsockopt	เรียกข้อมูล option ที่จัดการกับ specified socket
htonl	แปลงข้อมูลขนาด 32 bit จาก host byte order ไปเป็น network byte order
htons	แปลงข้อมูลขนาด 16 bit จาก host byte order ไปเป็น network byte order
inet_addr	แปลงข้อมูลประเภท Character string ที่แสดง หมายเลข Internet Standard dot Notation ของ IP Address(เช่น 140.90.23.100)ไปเป็นค่า Internet Address
inet_ntoa	แปลงข้อมูลจาก Internet Address Value ไปเป็น ASCII String in Dot Notation เช่น 143.90.23.100
ioctlsocket	ควบคุม โหมดการทำงานของ Socket
listen	เฝ้าฟังการเข้ามาติดต่อของ specified socket
ntohl	แปลงข้อมูลขนาด 32 bit จาก network byte order ไปเป็น host byte order
ntohs	แปลงข้อมูลขนาด 16 bit จาก network byte order ไปเป็น host byte order
recv*	รับข้อมูลจาก socket ที่ได้ทำการสร้าง connection แล้ว
recvfrom*	รับข้อมูลจาก socket ที่ได้ทำการสร้าง connection แล้ว หรือ จาก socket ที่ไม่ได้ทำ การ connect(unconnected socket)
select*	เลือกที่จะ view status ของ socket 1 socket หรือ มากกว่านั้น
send*	ส่งข้อมูลไปยัง socket ที่ได้ทำการสร้าง connection แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Function	Description
sendto*	ส่งข้อมูลไปยัง socket ที่ได้ทำการสร้าง connection แล้ว หรือ จาก socket ที่ไม่ได้ทำ การ connect(unconnected socket)
setsockopt	ทำการเซต option สำหรับ specified socket
shutdown	ยกเลิกการส่งหรือรับข้อมูลบน specified connection
socket	สร้าง socket และจะ return socket descriptor ไปให้ object

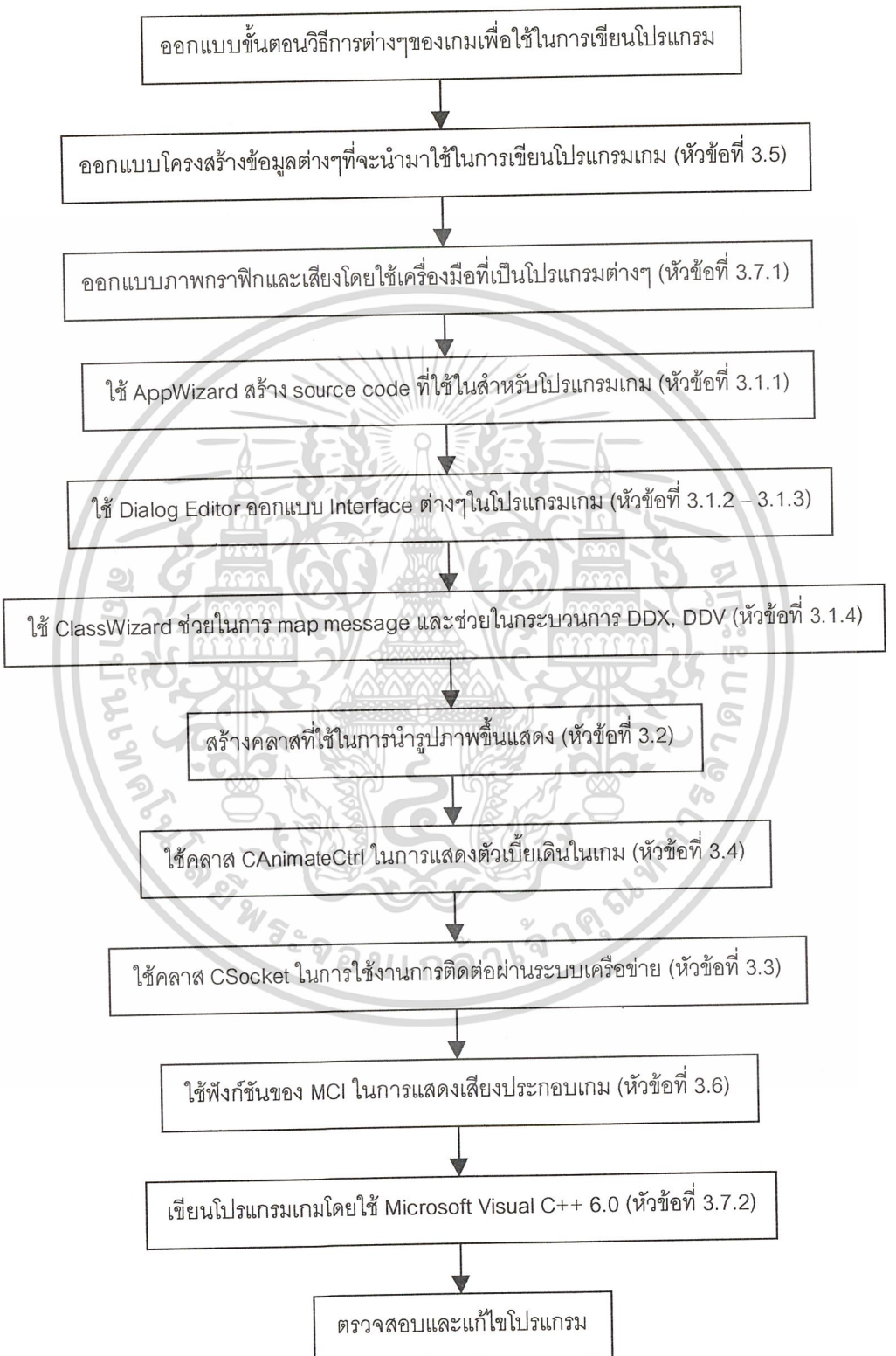
*ฟังก์ชั่นเหล่านี้จะถูก block สำหรับ blocking socket

2.4 ทฤษฎีอื่น ๆที่ได้แนวคิดมาจากปัญหาพิเศษก่อน ๆ

ผู้จัดทำได้ศึกษาปัญหาพิเศษเรื่อง โปรแกรมการเล่นเกมผ่านอินเตอร์เน็ต ของนักศึกษาภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ปีการศึกษา 2541 โดยได้แนวคิดต่าง ๆ ที่นำมาเป็นแนวทางในการคิดทำให้เกิดแนวความคิดต่าง ๆ มากมายที่จะนำมาใช้ในปัญหาพิเศษ ซึ่งทำให้ได้แนวความคิดต่าง ๆ เช่น คุณสมบัติเบื้องต้นของ Microsoft Visual C++ 6.0, การออกแบบโปรแกรมเกม ที่เป็นการออกแบบโครงสร้างข้อมูล และอัลกอริทึมของเกมอย่างคร่าวๆ, เทคนิคการเชื่อมต่อผ่านเครือข่ายด้วยโปรโตคอล TCP/IP โดยใช้งานคลาส CSocket ของ Microsoft Foundation Class ที่ทำให้ได้แนวคิดและวิธีการเบื้องต้น ซึ่งทางผู้จัดทำก็ได้ทำการเชื่อมต่อผ่านเครือข่ายโดยใช้งานจากคลาส CSocket นี้เช่นกัน, เทคนิคการนำรูปภาพ bitmap มาแสดงในโปรแกรม Microsoft Visual C++ 6.0 ทำให้ได้วิธีการใช้งานที่ง่ายเนื่องจากกำหนดให้การใช้งานอยู่ภายในคลาส, เทคนิคการทำภาพเคลื่อนไหว, วิธีการใช้งานฟังก์ชันที่ใช้ในการแมปแมสเสจ WM_TIMER ทำให้เกิดการทำงานตามเวลาที่เรากำหนดไว้ได้ เช่น ให้ทำงาน ก. ทุก ๆ 1 มิลลิวินาทีเป็นต้น, วิธีการใช้งานไลบรารี winmm.lib ในการนำเอาไฟล์เสียงสกุล wav มาแสดงได้ และได้แนวทางเพื่อเป็นตัวอย่างในการเขียนรายงานในการทำปัญหาพิเศษ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
วิธีดำเนินการปัญหาพิเศษ



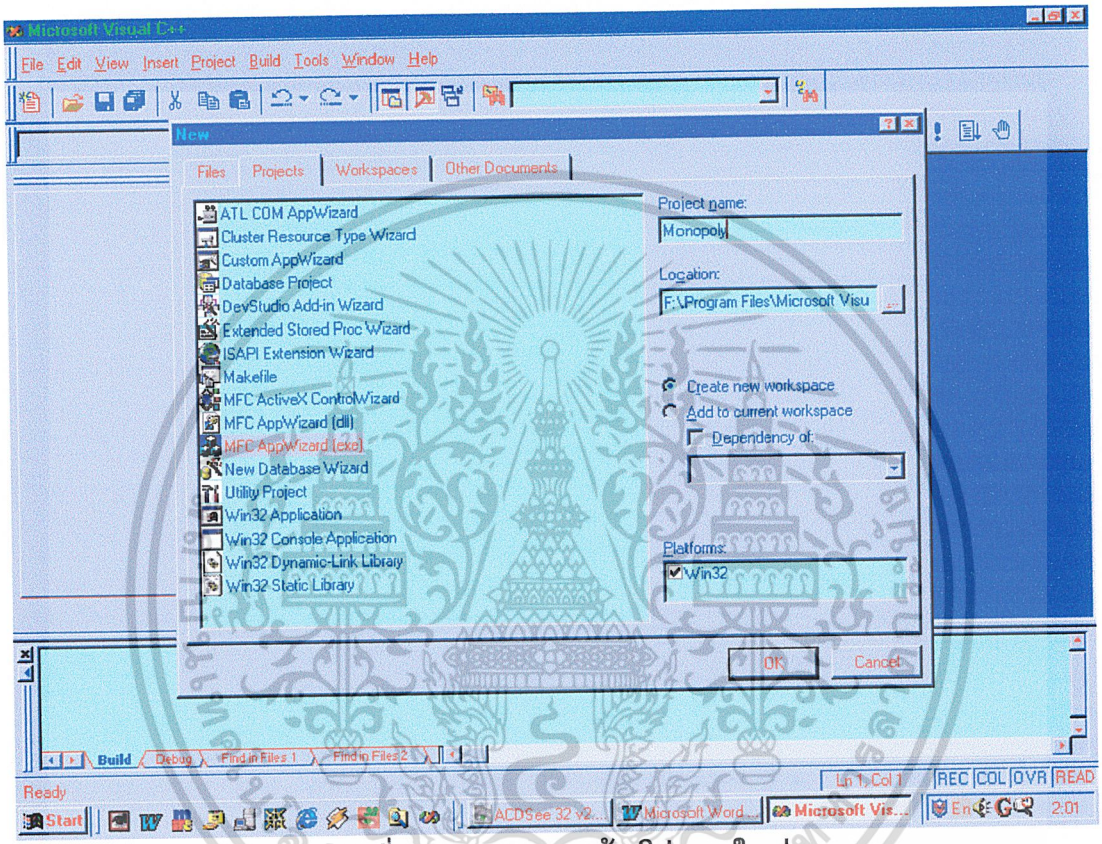
รูปที่ 3.1 แสดง Chart แสดงขั้นตอนการทำปัญหาพิเศษ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1 เทคนิควิธีการต่าง ๆของ Visual C++ที่นำมาใช้ในปัญหาพิเศษ

3.1.1 การใช้งานแอปวิซาร์ด

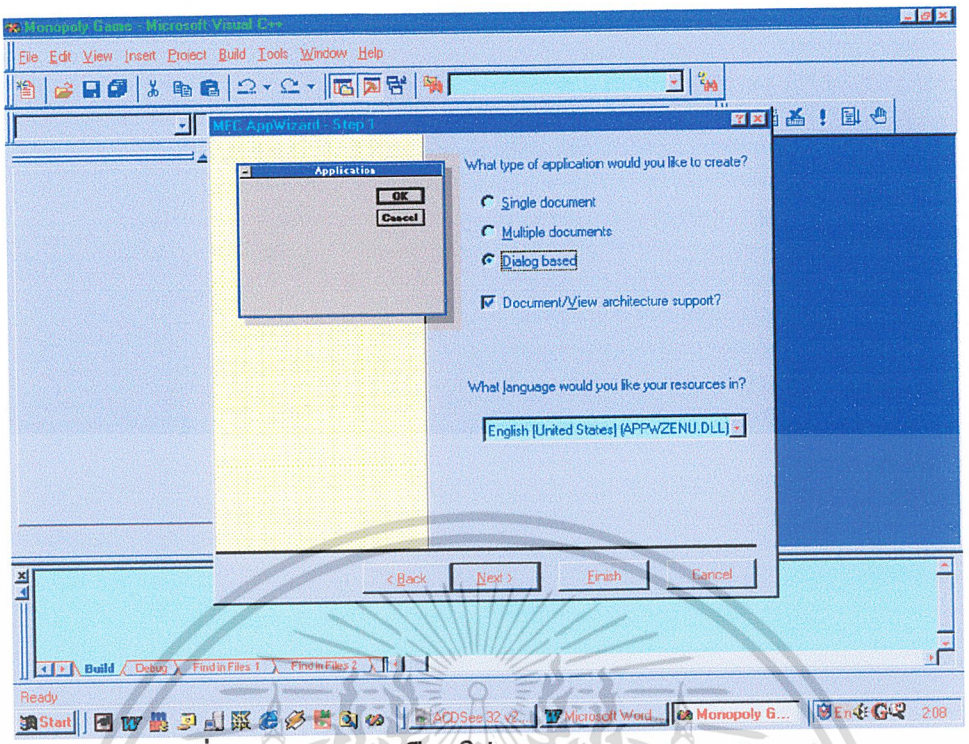
ให้เราเปิดโปรแกรม Visual C++ 6.0 แล้วทำการสร้างโปรเจ็คใหม่โดยการเลือกคำสั่ง File→ New แล้วเลือกแท็ป Projects จาก นั้นเลือก MFC AppWizard (exe) แล้วใส่ชื่อของโปรเจ็ค ในที่นี้เราใช้ชื่อว่า Monopoly และคลิ๊ก Ok ดังรูป



รูปที่ 3.2 แสดงการสร้างโปรเจคใหม่

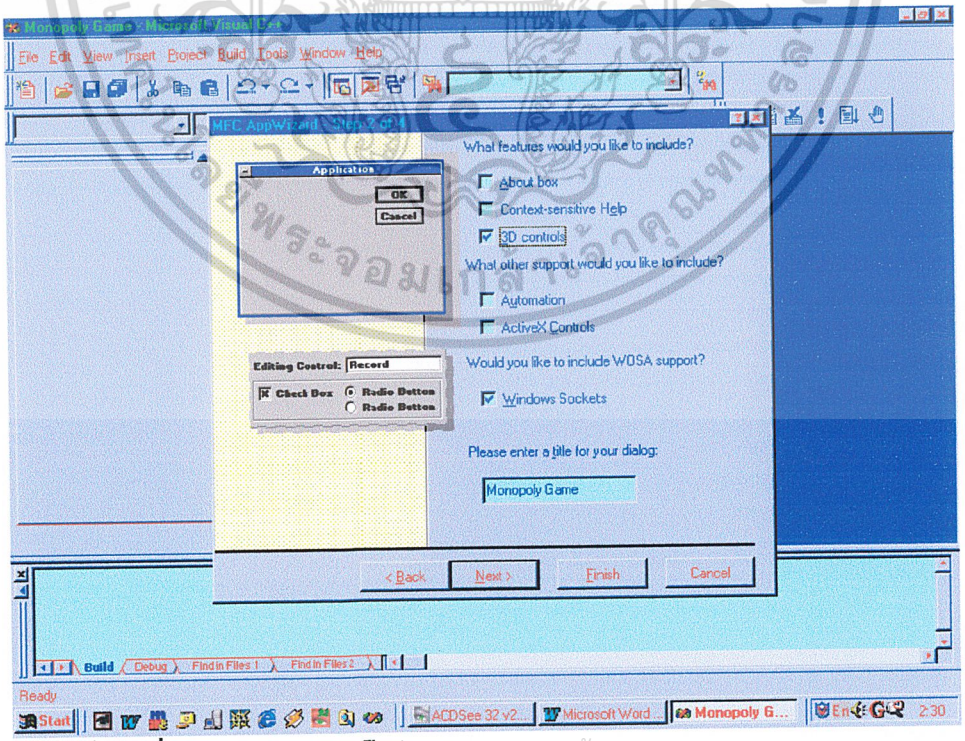
จากนั้นให้เราเลือกแบบของโปรแกรมที่เราต้องการสร้าง ในที่นี้เราเลือกแบบ Dialog based แสดงดังรูปข้างล่าง แล้วคลิ๊ก Next >

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 แสดงการเลือกโปรแกรมแบบ Dialog based

ต่อมาให้เราเลือก 3D controls เพื่อให้โปรแกรมดูสวยงามมีมิติ และเลือก Windows Sockets เพื่อใช้การติดต่อด้วย TCP/IP ด้วยคลาส CSocket ได้ ดังรูป



รูปที่ 3.4 แสดงการเลือก 3D controls และ Windows Sockets

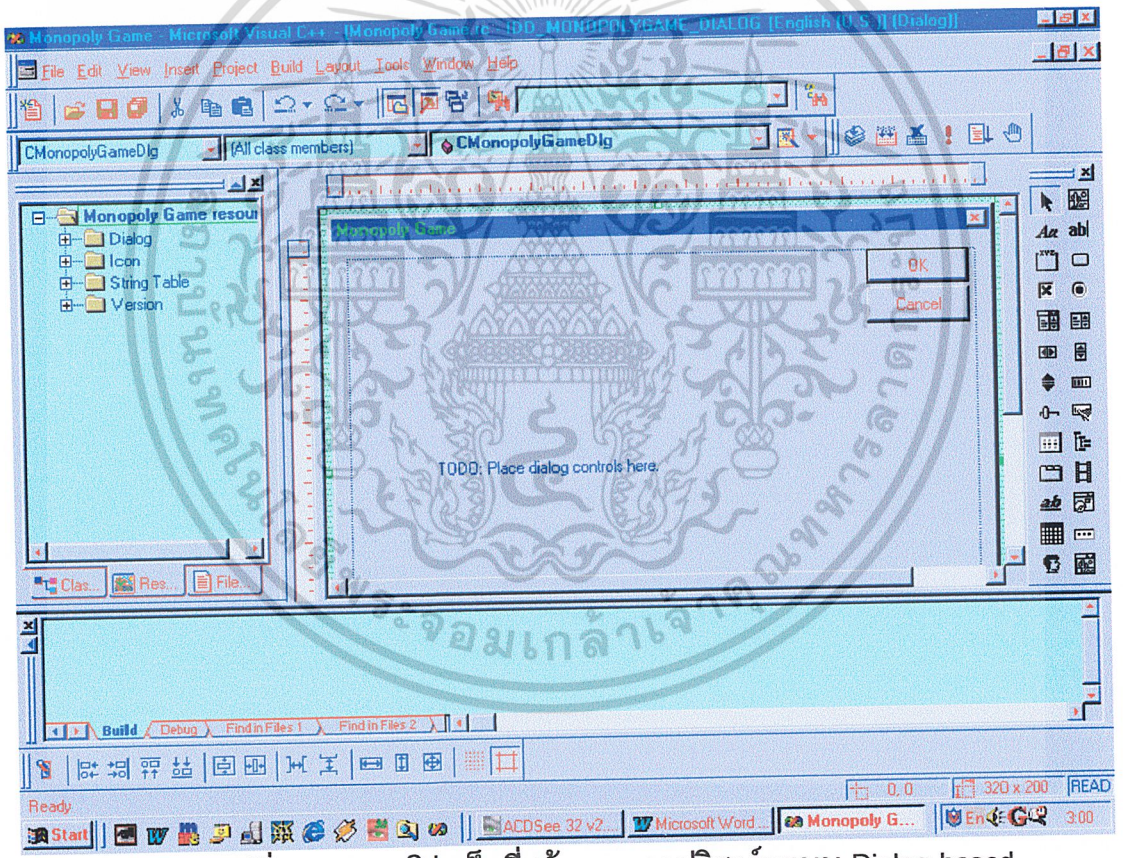
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

จากนั้นจะมีการให้เลือกว่าจะเอาคำอธิบายในซอร์สโค้ดหรือไม่ และการเลือกรูปแบบการลิงค์ (Link) กับไลบรารี MFC แบบใดซึ่งจะมี 2 ตัวเลือกคือ

- As a Shared DLL นั่นคือเมื่อเรานำโปรแกรมไปรันบนวินโดวส์ในเครื่องอื่น ในระบบนั้นจะต้องมีไฟล์ MFC42.DLL และ MSVCRT.DLL อยู่ด้วย ซึ่งถ้าไม่มีไฟล์ .DLL ทั้งสองตัวนี้ โปรแกรมก็จะไม่สามารถทำงานได้ ข้อดีของการลิงค์แบบนี้คือไฟล์ .EXE ที่ได้จะมีขนาดเล็ก
- As a Statically linked library นั่นคือเมื่อเรานำโปรแกรมไปรันเครื่องอื่น เราไม่จำเป็นต้องนำไฟล์ .DLL ไปด้วย แต่ไฟล์ .EXE ที่ได้จะมีขนาดใหญ่กว่าเล็กน้อย

ซึ่งในที่นี้เราเลือกให้มีคำอธิบาย และลิงค์แบบ “As a Shared DLL” แล้วกดปุ่ม Next> เพื่อทำการสร้างโปรเจ็คต่อไปจนเสร็จก็จะได้ดังรูป



รูปที่ 3.5 แสดงโปรเจ็คที่สร้างจากแอปวิซาร์ดแบบ Dialog based

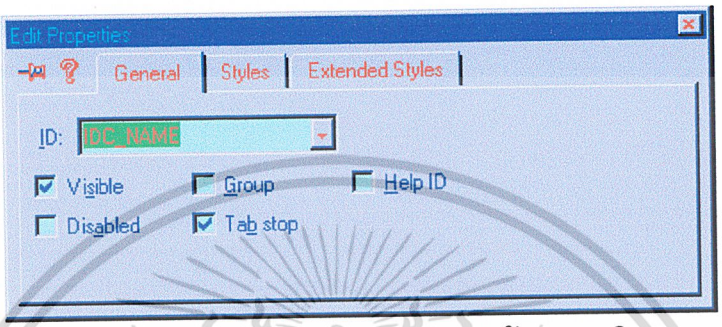
3.1.2 การใช้งานไดอะล็อกอิดิตเตอร์

เราก็ทำการปรับแต่งไดอะล็อกให้ได้เป็นไดอะล็อกของโปรแกรมโดยการหารูปภาพ หรือสร้างรูปภาพเพื่อจะนำมาเป็นพื้นของไดอะล็อก ซึ่งการปรับแต่ง ตกแต่ง สร้างรูปต่างๆในโปรเจ็ค เราจะทำการสร้างด้วยโปรแกรม Adobe Photoshop ส่วนการคัดเลือก เปลี่ยนฟอร์แมตของรูปภาพให้เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

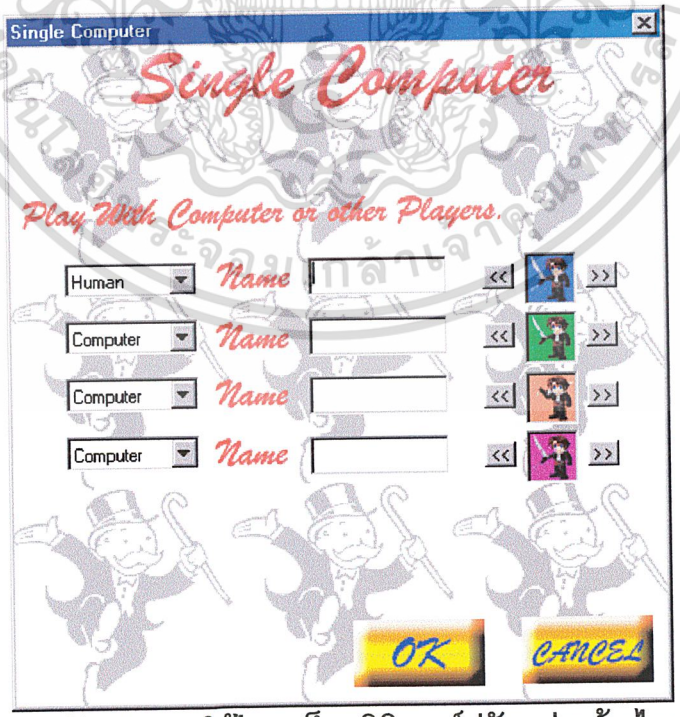
36148

.BMP(เพื่อให้ได้ภาพที่สามารถนำมาแสดงได้จากการอิดิตคอนโทรลจากอิดิตเตอร์ได้โดยตรง) โดยใช้โปรแกรม ACDSee 32 จัดการจากนั้นเราก็สามารถนำเอาคอนโทรลต่างๆจากคอนโทรลทูลบาร์มาใส่ลงในไดอะล็อกของเราได้ เช่น อิดิตบ็อกซ์คอนโทรลในการรับข้อมูลชื่อ เป็นต้น โดยเราจะทำการกำหนดค่าไอดี (ID) ใหม่ให้กับคอนโทรลเพื่อความง่ายในการจดจำ และสอดคล้องกับหน้าที่ของมัน โดยการกำหนดที่ Properties ดังรูป



รูปที่ 3.6 แสดงการกำหนดค่าไอดีใหม่ให้กับคอนโทรล

โดยเราสามารถกำหนดลักษณะต่างๆของคอนโทรลได้ เช่น ในแท็ป Styles และ Extended Styles ดังรูปข้างต้น เมื่อเราวางคอนโทรลต่างๆ จะได้ไดอะล็อกดังตัวอย่างในรูปข้างล่างนี้

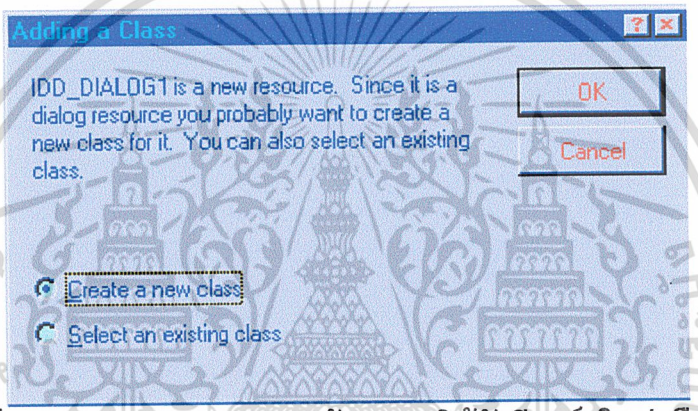


รูปที่ 3.7 ตัวอย่างการใช้ไดอะล็อกอิดิเตอร์ปรับแต่งสร้างไดอะล็อก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

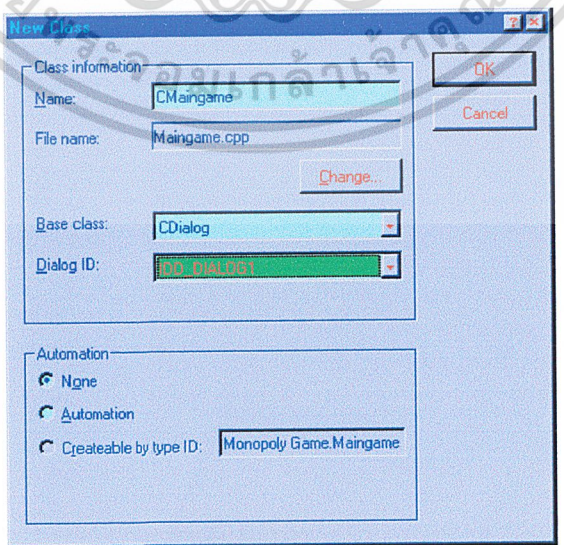
3.1.3 วิธีการสร้างไดอะล็อกเพิ่มในการใช้งาน

เมื่อออกแบบหน้าจอหนึ่ง และมีการส่งแมสเสจให้กับฟังก์ชันเพื่อไปยังหน้าจอถัดไปได้แล้ว เราก็จะทำการออกแบบหน้าจอถัดมาโดยการ Insert→Resource แล้วเลือกไดอะล็อก และทำการ New ก็จะได้ไดอะล็อกเปล่าๆ เราก็จะทำการออกแบบไดอะล็อกที่ 2 ด้วยวิธีเช่นเดียวกับการสร้างไดอะล็อกแรก และการสร้างไดอะล็อกที่ 2 นั้นเราจะต้องทำการสร้างคลาสของไดอะล็อกใหม่นี้โดยใช้คลาสวิซาร์ด เพื่อใช้คลาสเป็นตัวอ้างถึงไดอะล็อกนี้ โดยเชื่อมต่อทำด้วยการสร้าง Object จากฟังก์ชันของคลาสของไดอะล็อกแรกที่รับแมสเสจที่ให้มีการไปยังหน้าจอไดอะล็อกถัดไปได้ (โดยเมื่อเราเรียกใช้คลาสวิซาร์ด ก็จะมีการถามทันทีว่าจะสร้างคลาสของไดอะล็อกใหม่หรือไม่) แสดงได้ดังรูป



รูปที่ 3.8 แสดงการถามว่าจะสร้างคลาสให้กับรีซอร์สใหม่หรือไม่

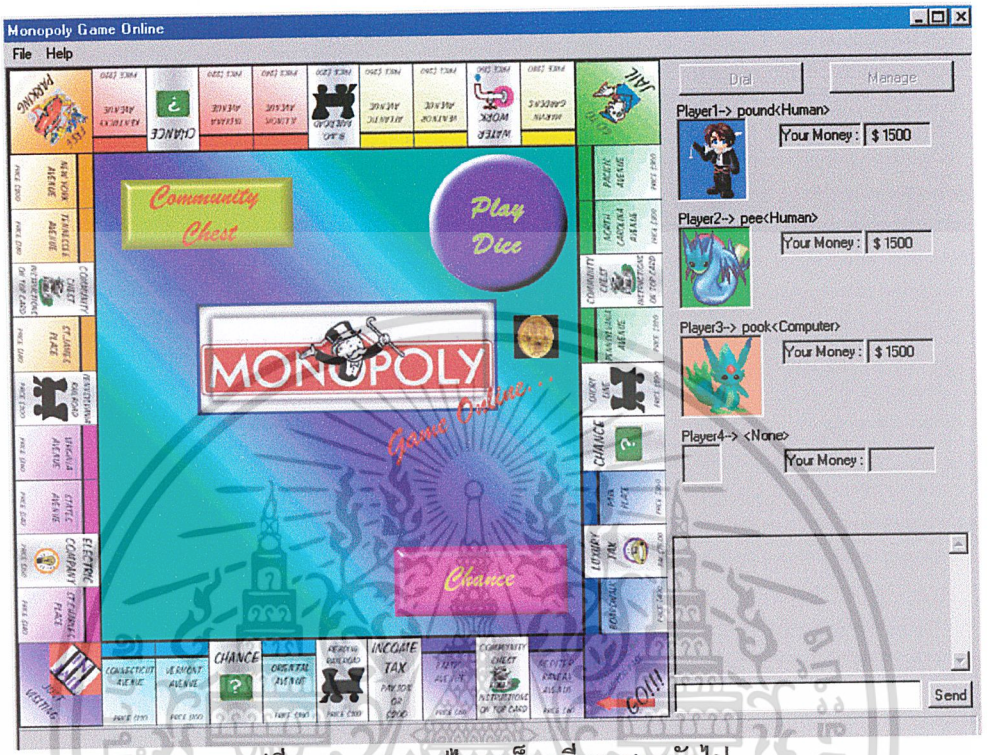
จากนั้นเมื่อเราตกลงจะสร้างคลาสใหม่ ก็จะมีการให้ใส่ชื่อคลาสที่เราจะสร้างดังรูป



รูปที่ 3.9 แสดงการตั้งชื่อให้กับคลาสของรีซอร์ดใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยเราสามารถเปลี่ยนค่า Dialog ID ดังแสดงดังรูปข้างต้นได้จาก Properties ของไดอะล็อก ให้สอดคล้องกับหน้าที่ของมัน เพื่อให้ง่ายต่อการจำ และนำไปใช้



รูปที่ 3.10 แสดงรูปไดอะล็อกที่จะแสดงถัดไป

จากนั้นเราจะทำการเชื่อมต่อไดอะล็อกโดยการสร้างวัตถุจากคลาสไดอะล็อกใหม่นี้ในส่วนโค้ดโปรแกรมของไดอะล็อกแรก โดยทำการสร้างวัตถุเพื่อนำมาใช้ในส่วนที่มีการแมปแมสเสจที่ต้องการให้มีการเปลี่ยนสถานะจากไดอะล็อกแรกมายังใหม่นี้ เช่น ต้องการแมปกับแมสเสจการคลิ๊กเมาส์ปุ่มซ้ายในบริเวณที่ต้องการก็จะทำการสร้างวัตถุไว้ที่ส่วนนั้นดังตัวอย่าง

```
void CSinglecomDlg::OnLButtonDown(UINT nFlags, CPoint point)
{
        // TODO: Add your message handler code here and/or call default
        if(point.x>=221&&point.x<=305&&point.y>=373&&point.y<=416){ //
กำหนดบริเวณปุ่ม Ok
            CMaingameDlg main;
            main.DoModal(); //ส่วนที่ต้องการให้มีการเรียกแสดงไดอะล็อกถัดไป
        }
        if(point.x>=330&&point.x<=414&&point.y>=372&&point.y<=415){ //
กำหนดบริเวณปุ่ม Cancel
```

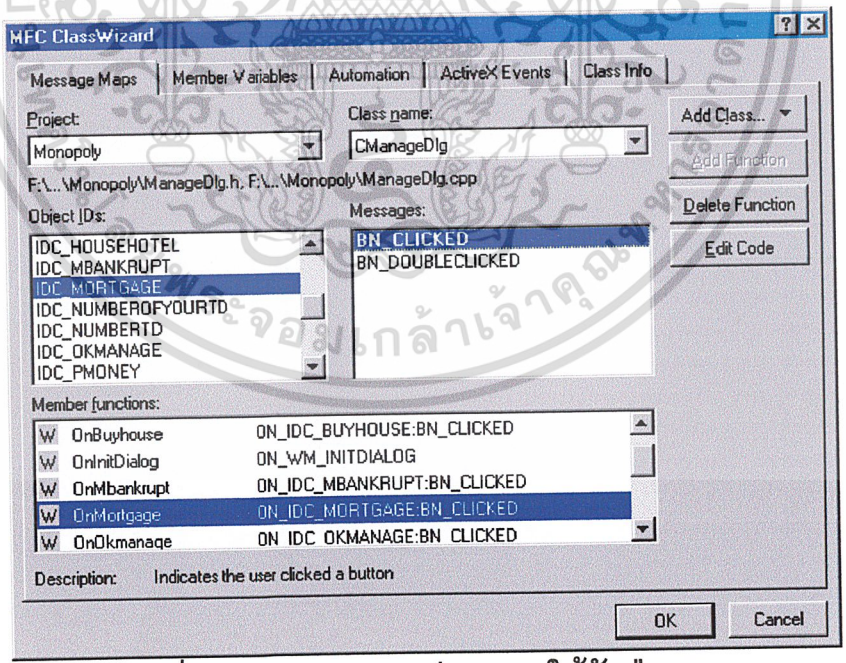
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                CSinglecomDlg::OnCancel(); //ส่วนที่ต้องการให้มีการยกเลิก(ทำลาย
ไดอะล็อกนี้)
            }
            CDialog::OnLButtonDown(nFlags, point);
}
```

และต้องมีการประกาศ #include "MaingameDlg.h" ที่เป็นไฟล์ส่วนหัวของไดอะล็อกใหม่ไว้ที่ไฟล์ส่วนต้นของไฟล์ SinglecomDlg.cpp ที่เป็นไฟล์ของไดอะล็อกแรกที่มีการเรียกใช้ไดอะล็อกใหม่

3.1.4 การใช้งานคลาสวิซาร์ด

การใช้งานคลาสวิซาร์ดในการแมปแมสเสจให้กับฟังก์ชันต่าง ๆ ทำได้โดยการคลิ๊กที่ View->ClassWizard แล้วคลิ๊กที่แท็ป Message Maps เพื่อทำการเลือกคลาสในโปรเจ็คที่ต้องการให้มีการแมปแมสเสจให้กับฟังก์ชันและนำมาใช้งาน ดังรูปตัวอย่างแสดงการแมปแมสเสจ BN_CLICKED ที่เป็นการแมปแมสเสจการกดปุ่มครั้งเดียวด้วยเมาส์ปุ่มซ้าย ให้กับปุ่ม IDC_MORTGAGE ดังนี้

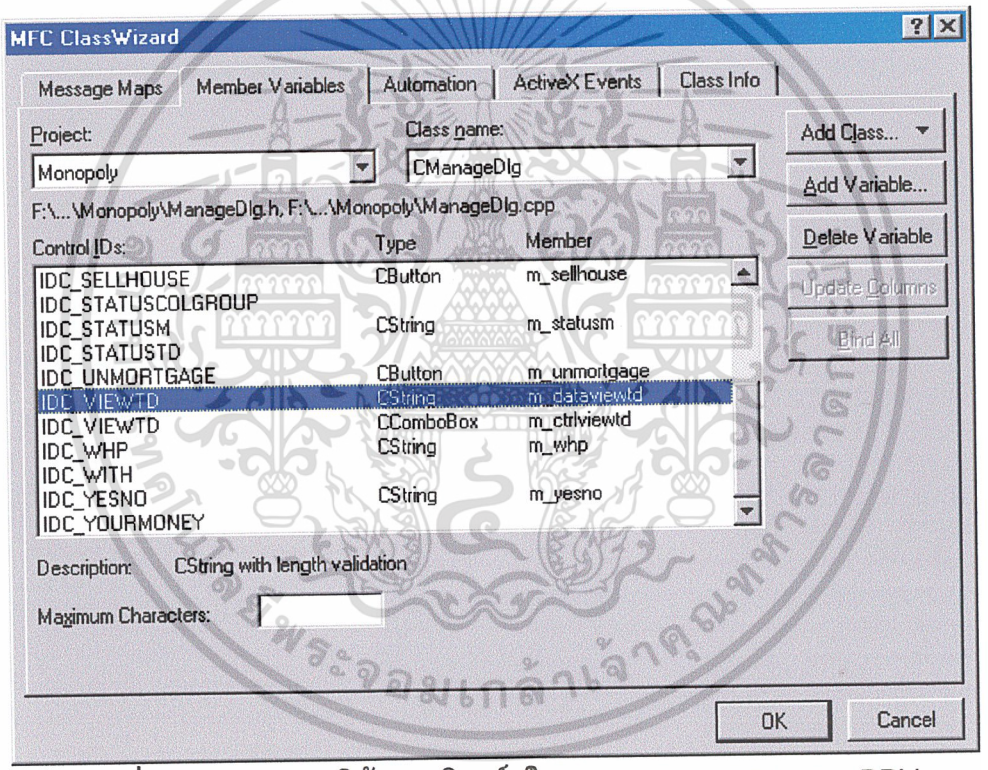


รูปที่ 3.11 แสดงการแมปแมสเสจให้กับปุ่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

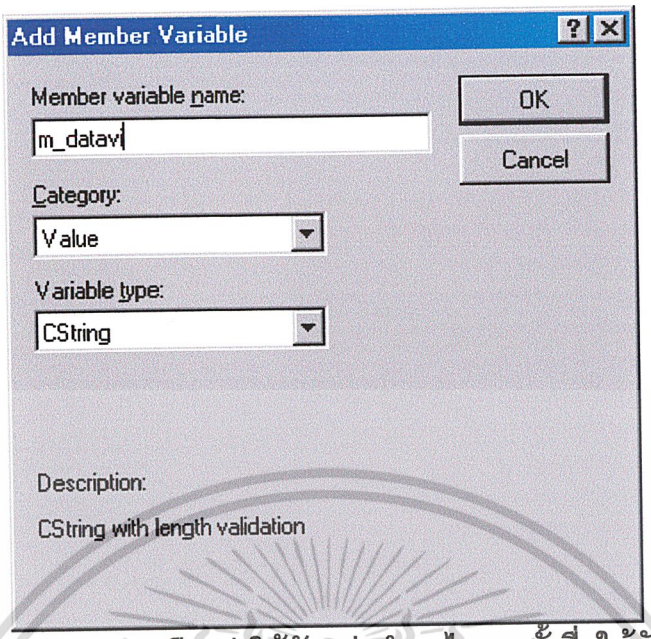
โดยเราจะทำการเลือกค่า Object IDs ที่เราต้องการ แล้วจึงเลือกแมปแมสเสจในช่อง Messages แล้วคลิ๊ก Add Function ก็จะขึ้นไดอะล็อกให้เราตั้งชื่อให้กับฟังก์ชันนั้นๆได้ จากนั้นเราก็สามารถกดปุ่ม Edit Code เข้าไปจัดการเขียนโค้ดต่างๆ เพื่อแมปกับแมสเสจที่เราแมปไว้ได้

เรายังใช้คลาสวิซาร์ดช่วยในกระบวนการ DDX(Dialog Data Exchange) และ DDV (Dialog Data Validation) รูปข้างล่างเป็นตัวอย่างการใช้ DDX และ DDV ในการจัดการคอมโบบ็อก IDC_VIEWTD โดยการเลือกไปที่แท็ป Member Variables ในคลาสวิซาร์ด แล้วทำการเลือก Control IDs ในตัวอย่างนี้เราเลือก IDC_VIEWTD แล้วกด Add Variable เพื่อทำการเลือกว่าตัวแปรดังกล่าวจะใช้เก็บค่าให้กับคอนโทรลหรือใช้เป็นตัวคอนโทรล ในตัวอย่างเราจะเลือกทั้ง 2 แบบ ดังรูป



รูปที่ 3.12 แสดงการใช้คลาสวิซาร์ดในกระบวนการ DDX และ DDV

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.13 แสดงการเลือกว่าใช้ตัวแปรทำอะไรและตั้งชื่อให้กับตัวแปร

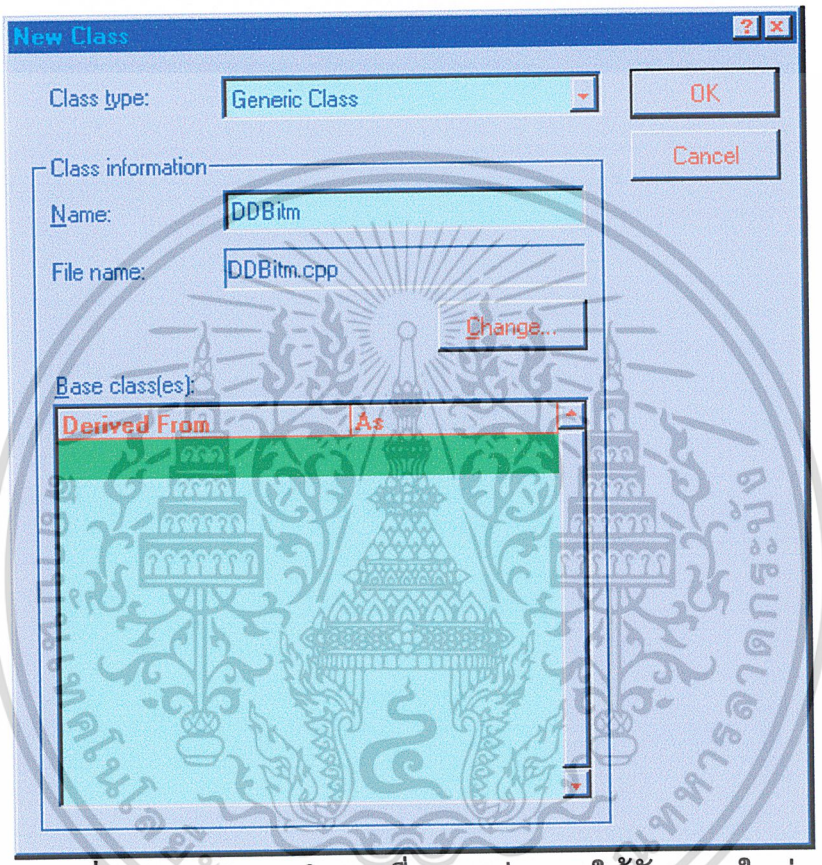
โดยในตัวอย่างเราจะกำหนดให้กับตัวแปรของ IDC_VIEWTD เป็น m_dataviewtd สำหรับเก็บค่าให้กับคอนโทรล และ m_ctrlviewtd สำหรับใช้คอนโทรลให้กับคอนโทรล ในตัวอย่างนี้เป็นคอนโทรลคอมโบบ็อก โดยเราจะใช้ m_dataviewtd ในการรับและส่งค่าให้กับคอนโทรลได้ โดยใช้ฟังก์ชัน UpdateData(TRUE) ในการรับค่ามาจากคอนโทรล เช่น ถ้าคอมโบบ็อกขณะนั้นเลือกที่ไอเท็มใด ก็จะได้ค่าไอเท็มนั้นมาเก็บไว้ในตัวแปรตามประเภทของตัวแปรนั้นๆ(ถ้าเป็นประเภท CString ก็จะได้ค่าเป็นข้อความ) เป็นต้น โดยเราจะต้องมีการเรียกใช้ฟังก์ชัน UpdateData(TRUE) ทุกครั้งก่อนที่จะรับเอาค่านั้นๆไปใช้ ในทางกลับกันเราสามารถกำหนดค่าเพื่อส่งไปให้กับคอนโทรลนั้นๆได้ โดยใช้ฟังก์ชัน UpdateData(FALSE) หลังจากกำหนดค่าให้กับตัวแปรแล้ว ส่วนกระบวนการ DDV คือการตรวจสอบค่าของตัวแปรไม่ให้เกินค่าที่เรากำหนดไว้ โดยกำหนดค่าไว้ที่คลาสวิซาร์ดแท็ป Member Variables ในช่อง Maximum Characters ของตัวแปรที่ใช้ในการเก็บค่าเท่านั้น ส่วนตัวแปร m_ctrlviewtd ก็สามารถที่จะใช้ฟังก์ชันต่างๆในการคอนโทรลให้กับคอนโทรลได้ เช่น ใช้ฟังก์ชัน AddString(LPCTSTR lpszString) โดยที่ lpszString ก็คือค่าของสตริงที่จะเพิ่มเข้าไปให้กับไอเท็มในคอมโบบ็อกตามลำดับ ซึ่งใช้วิธีเรียกฟังก์ชันผ่านตัวแปรเช่น m_ctrlviewtd.AddString("Test"); เป็นต้น

3.2 วิธีการนำเอารูปภาพแสดงในโปรแกรม

สร้างคลาสใหม่เพื่อนำมาสืบทอดจากคลาส CBitmap ของ Microsoft Foundation Class ให้เป็นคลาสที่ใช้ในการรวมฟังก์ชันต่างๆในการนำรูป bitmap ขึ้นแสดง ทำให้สะดวกในการเรียกใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

งาน เพียงแต่สร้างวัตถุจากคลาสแล้วใช้ฟังก์ชันนั้นๆเรียกใช้รูปที่เรานำมาเป็นทรัพยากรของโปรเจ็คเมื่อต้องการแสดงรูปและทำลายวัตถุนั้นเสียเพื่อคืนพื้นที่หน่วยความจำเมื่อสิ้นสุดการใช้งาน โดยการเลือกที่เมนูของ Microsoft Visual C++ โดยเลือก Insert→New Class แล้วกำหนดชนิดของคลาส (Class Type) เป็น Generic Class และเราจะกำหนดชื่อของคลาสนี้เป็น DDBitmap แสดงดังรูป



รูปที่ 3.14 แสดงการกำหนดชื่อ และประเภทให้กับคลาสใหม่

เมื่อเราได้คลาสใหม่ เราก็จะทำการเพิ่มเติมในไฟล์ DDBitmap.h ของคลาสใหม่ดังนี้

```
class CDDBitmap:public CBitmap
{
public:
        CDDBitmap();
        ~CDDBitmap();
public:
        BOOL DrawOnDC(CDC* pDC,int x,int y);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        BOOL StretchOnDC(CDC* pDC,int x,int y,int newWidth,int newHeight);
};
class DDBitmap
{
public:
        DDBitmap();
        virtual ~DDBitmap();
};
```

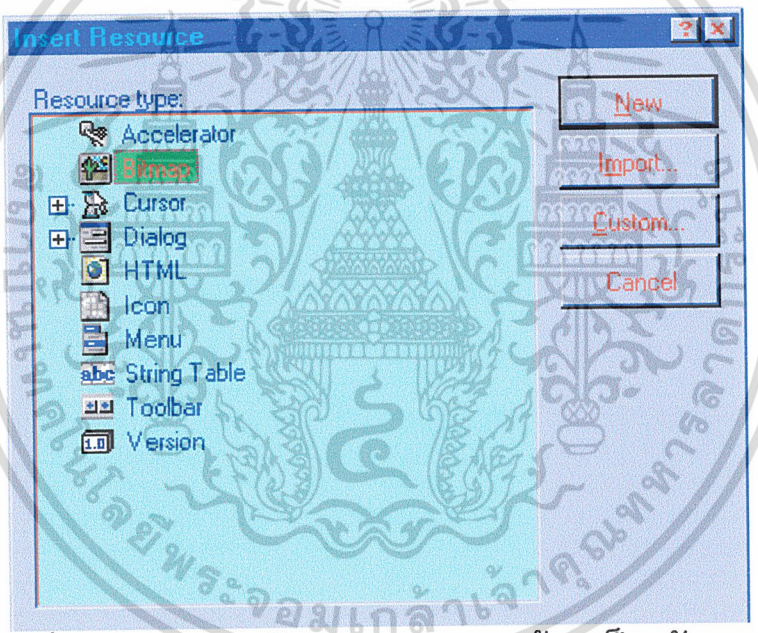
แล้วทำการเพิ่มเติมในไฟล์ DDBitmap.cpp ของคลาสใหม่ดังนี้

```
//CDDbitmap
CDDBitmap::CDDBitmap()
{
}
CDDBitmap::~CDDBitmap()
{
}
BOOL CDDBitmap::DrawOnDC(CDC* pDC,int x,int y)
{
        ASSERT_VALID(pDC);
        BITMAP bm;
        GetObject(sizeof(bm),&bm);
        CDC dcBM;
        dcBM.CreateCompatibleDC(pDC);
        CBitmap* pOldBM = dcBM.SelectObject(this);
        BOOL Result = pDC-
>BitBlt(x,y,bm.bmWidth,bm.bmHeight,&dcBM,0,0,SRCCOPY);
        dcBM.SelectObject(pOldBM);
        return Result;
};
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DDBitmap::DDBitmap()
{
}
DDBitmap::~DDBitmap()
{
}
```

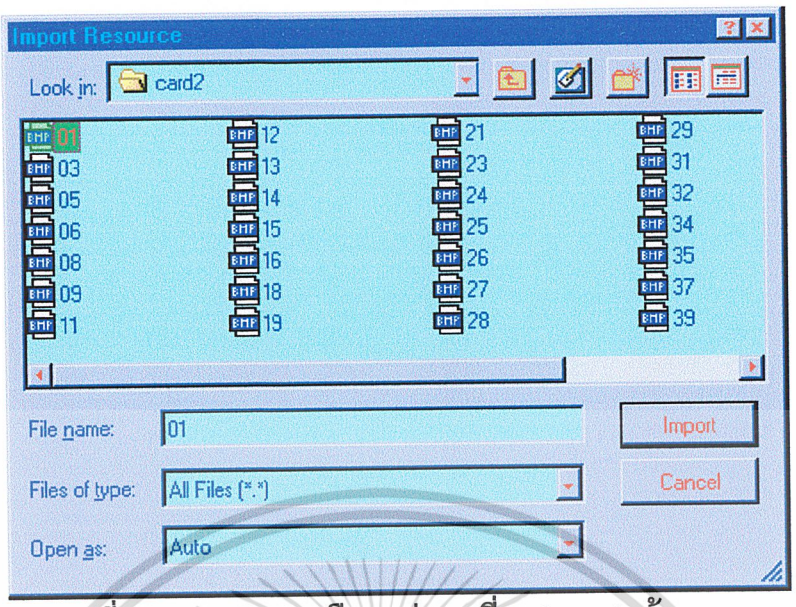
จึงนำเอารูปภาพ bitmap เข้ามาเป็นทรัพยากรของโปรเจ็ค โดยเลือกเมนู Insert→Resource แล้วเลือกชนิดของทรัพยากรเป็น bitmap แล้วคลิ๊กที่ปุ่ม Import และเลือกรูปภาพที่ต้องการนำมาใช้แสดงดังรูป



รูปที่ 3.15 **แสดงการเลือก** Import Bitmap **เข้ามาเป็นทรัพยากร**

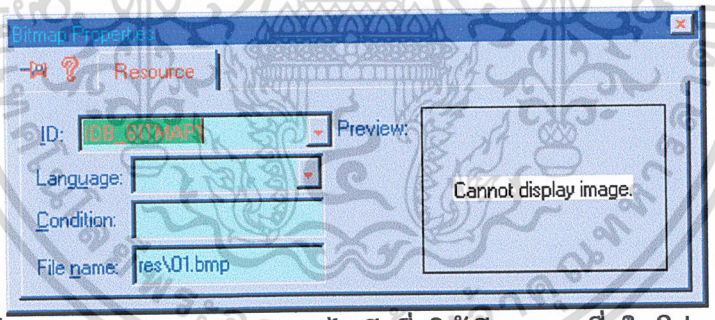
โดยเราสามารถที่จะเลือกรูป bitmap ใดๆก็ได้ในสกุล bmp เพื่อนำเข้ามาเป็นทรัพยากรของโปรแกรมได้ ดังรูปข้างล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.16 แสดงการเลือกรูปภาพที่จะ Import เข้ามา

เราสามารถกำหนดค่าไอดีให้กับรูปภาพที่นำเข้ามาเป็นทรัพยากรของโปรแกรมได้ตามต้องการเพื่อง่ายต่อการเข้าใจและนำไปใช้ โดยเลือก Properties ของรูปภาพขึ้นมากำหนดค่าไอดีได้ แสดงดังรูป



รูปที่ 3.17 แสดงการกำหนดค่าไอดีเพื่อใช้เรียกแทนชื่อในโปรแกรม

ก่อนที่เราจะใช้งานคลาส DDBitmap ที่เราสร้างขึ้นได้นั้นจะต้องเพิ่ม #include "DDBITMAP.H" เข้าไปที่ส่วนต้นของคลาสที่เราจะนำคลาส DDBitmap ไปใช้งาน ซึ่งในที่นี้คลาส CViewbankDlg ของเรามีการใช้งานคลาส DDBitmap จึงต้องมีการเพิ่ม #include "DDBITMAP.H" เข้าไปที่ไฟล์ ViewbankDlg.cpp ด้วย

การเรียกภาพขึ้นมาแสดงของคลาส DDBitmap นั้นทำได้โดยพิมพ์คำสั่งต่อไปนี้ลงไปในฟังก์ชันที่จะเรียกแสดงรูปภาพ ดังนี้

```
CClientDC dc(this);
CDDBitmap DDB;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DDB.LoadBitmap(PictureID);
DDB.DrawOnDC(&dc,x,y);
DDB.DeleteObject();
```

โดยที่ x คือ ตำแหน่งของหน้าจอโปรแกรมในแกน x (pixels)

y คือ ตำแหน่งของหน้าจอโปรแกรมในแกน y (pixels)

(จุดซ้ายบนสุดของหน้าจอโปรแกรมคือ (0,0))

PictureID คือ ค่าไอดีของรูปภาพที่เป็นทรัพยากรของโปรแกรม

3.3 วิธีการเชื่อมต่อทางระบบเครือข่ายที่ใช้

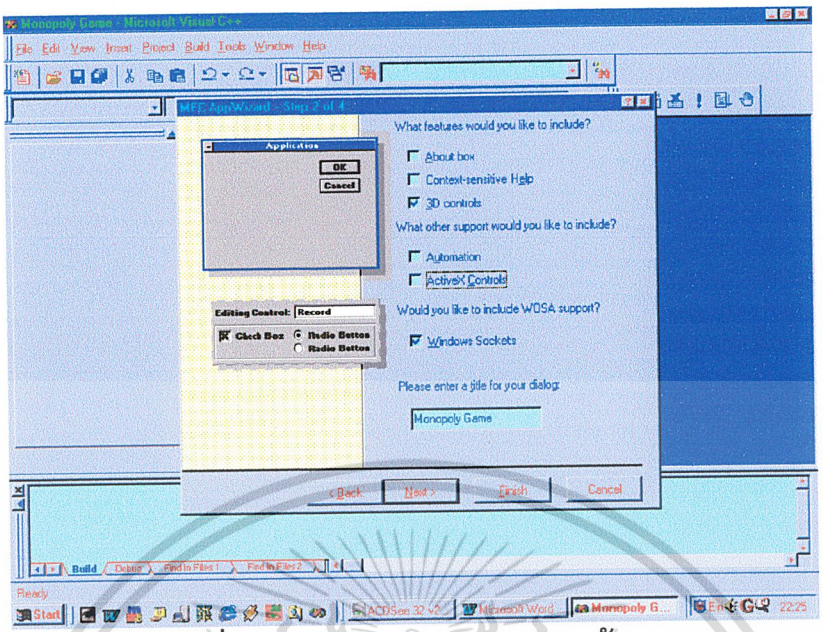
โดยในปัญหาพิเศษนี้เราจะใช้ Class CSocket ของ Microsoft Foundation Class ของ Microsoft Visual C++ 6 มีวิธีการที่เป็นแนวคิดอย่างคร่าวๆ ดังต่อไปนี้

1. เมื่อเปิดโปรแกรม ผู้เล่นจะเลือกว่าจะเป็นผู้เล่นหลัก (Server) หรือเป็นแบบผู้ไปขอเชื่อมต่อ (Client)
   - ถ้าเป็นแบบผู้เล่นหลักจะทำการสร้าง (Create) และเปิดพอร์ต (Port) เพื่อรอการเชื่อมต่อจากผู้เล่นอื่น
   - ถ้าเป็นแบบผู้ไปขอเชื่อมต่อจะต้องใส่ IP Address ของผู้เล่นที่จะไปเชื่อมต่อและทำการเชื่อมต่อ (Connect)
2. เมื่อผู้เล่นหลักกดเริ่มเกมส์ ผู้เล่นที่มาเชื่อมต่อจะทำการส่งข้อมูลของเกมส์ที่มีการเปลี่ยนแปลงให้กับผู้เล่นหลักทุกๆครั้งที่มีการกระทำการใดๆที่จะต้องมีการเปลี่ยนแปลงข้อมูลต่างๆของเกมส์ หรือให้มีการแสดงผลหน้าจอ
3. เมื่อผู้เล่นหลักได้รับข้อมูลจากผู้เล่นอื่นๆ จะทำการ Update หน้าจอ หรือข้อมูลของเกมของตนเอง และจะทำการส่งข้อมูลที่ได้รับมาไปให้กับผู้เล่นทุกคน
4. เมื่อผู้เล่นได้รับข้อมูลจากผู้เล่นหลักก็จะ Update หน้าจอของตนเอง

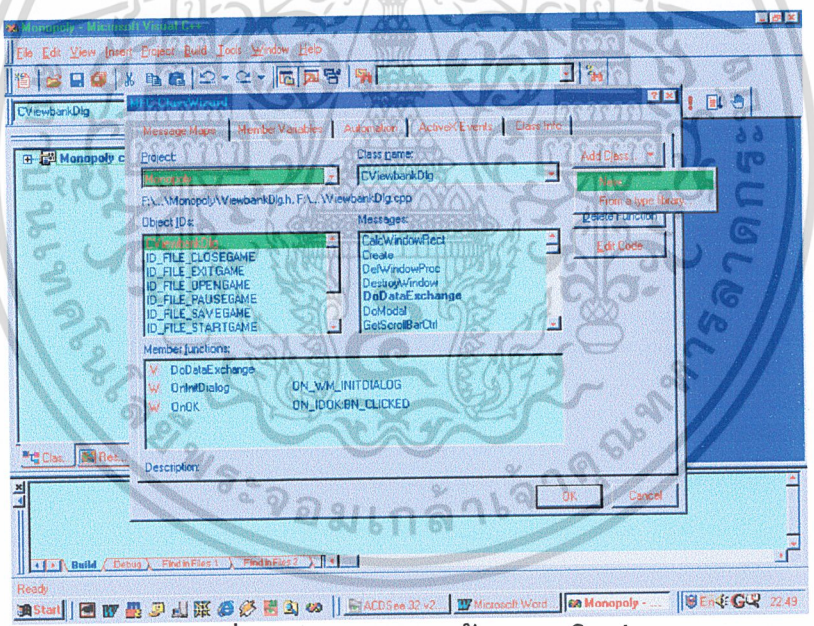
ส่วนขั้นตอนในการเชื่อมต่อจะเชื่อมต่อโดยผ่านทางโปรโตคอล (Protocal) TCP/IP โดย Microsoft Visual C++ จะใช้ Class CSocket ซึ่งมีขั้นตอนในการทำดังนี้

1. ใน Step ที่ 2 จาก 4 ของการสร้างโปรเจ็คใหม่โดยวิธี MFC แบบ Dialog based จะมีเช็คบล็อคคำว่าWindows Sockets ให้ทำการเช็คแล้วสร้างโปรเจ็คต่อไป แสดงดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



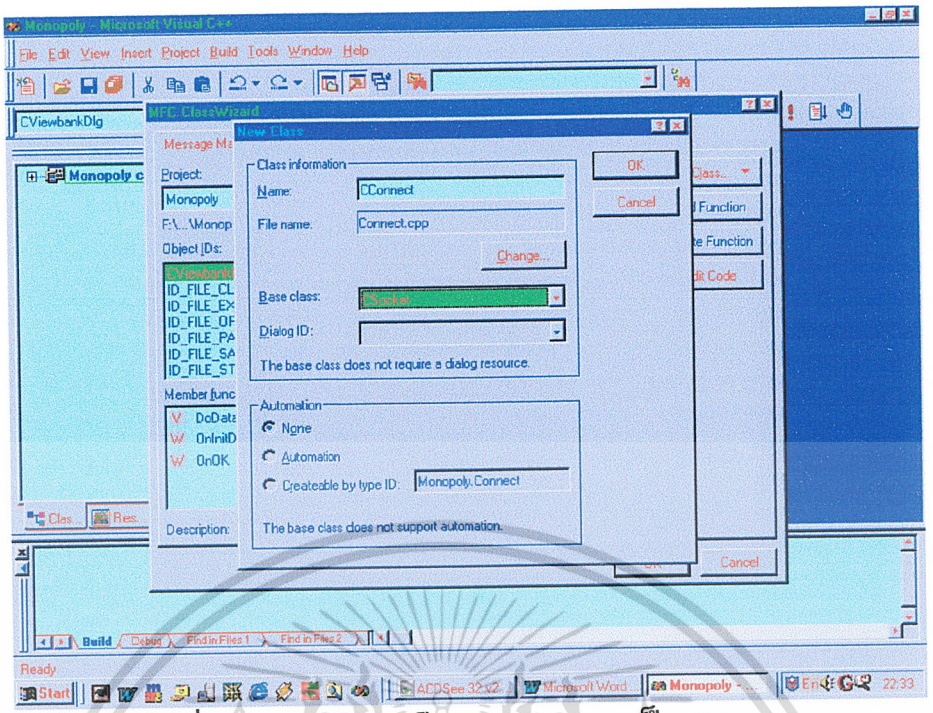
รูปที่ 3.18 แสดง Step 2 of 4 ตามข้อ 1.



รูปที่ 3.19 แสดงการสร้างคลาสใหม่

2. ใช้คลาสวิซาร์ด สร้างคลาสใหม่ 3 คลาส โดยมีคลาสพื้นฐานเป็น CSocket โดยที่แบ่งคลาสออกเป็น 2 อย่างคือ คลาสของเซิฟเวอร์ 2 คลาสโดยในที่นี้จะตั้งชื่อเป็น CServer และ CListenSock และคลาสของตัวเชื่อมต่อตั้งชื่อเป็น CConnect

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.20 แสดงการเลือก Base Class เป็น CSocket

3. ใช้คลาสวิซาร์ดในการเพิ่มฟังก์ชัน OnConnect, OnClose และ OnReceive ให้กับคลาส CConnect
4. ใช้คลาสวิซาร์ดในการเพิ่มฟังก์ชัน OnClose และ OnReceive ให้กับคลาส CServer และฟังก์ชัน OnAccept และ OnClose ให้กับคลาส CListenSock
5. #include “Server.h” และ #include “Connect.h” ไว้ที่ .h ของโปรแกรมหลัก
6. ทำการ Map Message จากคลาสที่สร้างไปยังโปรแกรมหลัก ในกรณีที่มีการ Notify มายังฟังก์ชันที่เราสร้างในข้อ 3, 4 ไปยังโปรแกรมหลัก โดยใช้คำสั่ง SendMessage ตัวอย่างคือ ((CmainFrame *) AfxGetMainWnd())->GetActiveView()->SendMessage (WM_CLIDIS,0,(LPRAM) this);

   โดยประกาศตัวแปร #define WM_CLIDIS WM_USER + 1 ไว้ที่ .h ของโปรแกรม ในที่นี้คือ Connect.h และต้องหาฟังก์ชันมารับส่วนที่ Map มาโดยเขียนไว้ในบริเวณ BEGIN_MESSAGE_MAP ของโปรแกรมหลักเช่น ON_MESSAGE(WM_CLIDIS, OnClientDisconnected)

   โดยประกาศฟังก์ชัน OnClientDisconnected ไว้ที่ .h ของโปรแกรมหลักเขียนได้ดังนี้

   afx_msg long OnClientDisconnected(UINT wParam,LPARAM lParam);
7. ตั้งตัวแปรตามคลาส CServer หรือ CConnect เมื่อผู้เล่นเลือกเล่นในแต่ละแบบโดยที่ถ้าผู้เล่นเลือกเล่นแบบเป็นเซิฟเวอร์ให้ทำการ Creat และ Listen

   CListenSock m_Listen; (ตั้งชื่อตัวแปรชื่อ m_Listen ไว้ที่ .h ของโปรแกรมหลัก)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารนี้ทุกครั้งที่มีการนำไปใช้

M_Listen.Create(3200);

(m_Listen.Listen())

โดยที่ 3200 คือหมายเลขพอร์ตที่ใช้ในการเชื่อมต่อ

ในขณะที่ถ้าผู้เล่นเลือกเล่นแบบเป็นผู้เชื่อมต่อให้ทำการ Connect

CConnect m_Connect;

m_Connect.Create();

m_Connect.Connect(SerIP,3200);

โดยที่ SerIP แทน IP Address ของ Server ที่ต้องการเชื่อมต่อ

8. ใช้คำสั่ง Send ในการส่งข้อมูลไปหาเซิฟเวอร์ หรือส่งจากเซิฟเวอร์ไปยังผู้เล่นอื่น

m_Listen.Send(dataM1, 7);

ในที่นี้ dataM1 คือข้อมูลที่ต้องการจะส่งไปและ 7 คือขนาดของข้อมูล

และใช้คำสั่ง Receive ในฟังก์ชันที่ Map มาจากฟังก์ชัน OnReceive ในข้อ 3 และ 4

Cserver * pSock = (CServer *) lParam;

pSock->Receive(dataMRec, 7);

ในที่นี้ dataMRec เป็นตัวแปรที่มารับข้อมูลและ 7 คือขนาดของข้อมูล(นับเป็นจำนวนของ char)

ขนาดของข้อมูลที่ส่งมาต้องเท่ากับขนาดของตัวแปรที่นำมารับข้อมูล

3.4 วิธีการแสดงตัวเบี้ยในเกม

การแสดงตัวเบี้ยเดินในเกมนี้ เราใช้ตัวเบี้ยเดินจากไฟล์สกุล avi โดยมีเทคนิคในการแสดงดังต่อไปนี้

1. ทำการประกาศตัวแปรของคลาส CAnimateCtrl (ซึ่งเป็นคลาสของ Microsort Foundation Class)เป็นพอยเตอร์ไปยังคลาส โดยประกาศไว้ที่ไฟล์ส่วนหัวของไดอะล็อกที่ต้องการจะแสดงตัวเบี้ยเดิน เพื่อที่จะสามารถอ้างถึงได้สะดวกในการจองพื้นที่หน่วยความจำให้ในการนำมาแสดงไฟล์ avi และการคืนค่าพื้นที่หน่วยความจำเพื่อหยุดแสดงภาพนั้น เช่น ประกาศ CAnimateCtrl* animate; เป็นแบบ public ไว้ที่ไฟล์ CMaingameDlg.h ของไดอะล็อกส่วนตารางเกม เป็นต้น
2. ทำการจองพื้นที่หน่วยความจำให้กับตัวแปรพอยเตอร์นั้น ในตัวอย่างข้อ 1. เช่น

animate = new CAnimateCtrl;

แล้วทำการแสดงภาพได้โดยใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

animate->Create(WS_CHILD|WS_VISIBLE|ACS_AUTOPLAY,CRect(CPoint (0,0),CSize(40,40)),this,110); // แสดงแบบ autoplay ที่จุด(0,0) ขนาด 40*40 pixels ที่ไดอะล็อกนี้ มีค่า ID = 110(ID ของคอนโทรล)

an1->Open(IDR_BLUE1); // เปิดทรัพยากร IDR_BLUE1 ที่เก็บไฟล์ avi ไว้ โดยสามารถนำเอาไฟล์ avi เข้ามาเป็นทรัพยากรของโปรแกรมได้เช่นเดียวกับวิธีของภาพสกุล bmp

3. เมื่อต้องการไม่แสดงภาพที่จุดนั้นอีกก็ทำการคืนค่าให้กับหน่วยความจำ ในตัวอย่าง เช่น

animate.Close(); // ทำการปิดการเล่นไฟล์

delete animate; // ทำการคืนค่าหน่วยความจำ

เนื่องจากภาพยังคงแสดงค้างอยู่ที่ไดอะล็อกให้ทำการวาดวินโดว์ใหม่ดังนี้

RedrawWindow(NULL,NULL,RDW_ERASENOW|RDW_UPDATENOW|RDW_INVALIDATE);

ภาพก็จะหายไป (วิธีการวาดวินโดว์ใหม่นี้จะนำมาใช้ลบการแสดงภาพ bitmap ก็ได้)

4. จากนั้นก็สามารถทำการแสดงที่จุดใหม่ในตารางเกมได้โดยทำตามข้อ 2. โดยเปลี่ยนค่า CPoint ในฟังก์ชัน Create และทำการลบตามข้อ 3. ทำเช่นนี้วนไปเรื่อยๆ หากยังต้องการแสดงตัวเบี้ยเดินในตารางเกมอยู่

3.5 การออกแบบโครงสร้างข้อมูลที่ใช้ในเกม

เราทำการออกแบบโครงสร้างข้อมูลที่จะใช้ในโปรแกรมเกม โดยมีโครงสร้างข้อมูลอยู่ 2 อย่างที่สำคัญและน่าสนใจมาแสดงให้ทราบถึงโครงสร้างการออกแบบดังนี้

1. โครงสร้างข้อมูลที่จะนำมาเก็บข้อมูลของผู้เล่นแต่ละคน เราทำการออกแบบเป็นคลาส ซึ่งวิธีการที่จะเพิ่มคลาสเข้าไปในโปรเจ็คสามารถทำได้โดยเข้าไปในแท็ป Massage Maps ของคลาสวิซาร์ด แล้วเลือก Add Class ที่ New จากนั้นก็ทำการใส่ชื่อคลาสใหม่ได้ โดยในคลาสที่เราจะสร้างนี้จะไม่เลือก Based Class เนื่องจากเราต้องการสร้างมันขึ้นมาใหม่ ไม่ได้ต้องการสืบทอดคุณสมบัติจากคลาสอื่นๆที่เป็น Based Class มาใช้ เราก็จะได้ไฟล์สกุล cpp และ h ของคลาสนั้นๆสร้างคลาสใช้งานได้ โดยการประกาศค่าสมาชิกข้อมูลและฟังก์ชัน จะอยู่ที่ไฟล์ส่วนหัว ส่วนให้นิยามการทำงานของฟังก์ชันต่างๆจะอยู่ที่ไฟล์สกุล cpp โดยคลาสของข้อมูลผู้เล่นแต่ละคนจะมีรายละเอียดดังต่อไปนี้

// PlayData.h

class CPlayData

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
public:
        CPlayData(); // ฟังก์ชันคอนสตรักเตอร์ของคลาส
        virtual ~CPlayData(); // ฟังก์ชันเดสตรักเตอร์ของคลาส
public:
        char idplayer; // ใช้เก็บสถานะของผู้เล่นว่าเป็น computer, human หรือ none
        char playname[13]; // ใช้เก็บชื่อของผู้เล่น
        char idtoken; // ใช้เก็บว่าผู้เล่นเลือกใช้ตัวเดินอะไร เช่น หมา, แมว, นก เป็นต้น
        bool idtitled[40]; // ใช้เก็บสถานะของโฉนดแต่ละใบว่าผู้เล่นคนนี้เป็นเจ้าของ
        int idstay; // ใช้เก็บค่าตำแหน่งของผู้เล่นในตารางเกม
        int money; // ใช้เก็บค่าเงินของผู้เล่น
        int njcard; // ใช้เก็บจำนวนการ์ดออกจากคุกฟรี
public:
        void addPlayData(char IdPlayer,char *PlayName,char IdToken); // เป็น
        ฟังก์ชันที่ใช้กำหนดค่าเริ่มต้นให้กับผู้เล่นแต่ละคน ทำให้มีความสะดวกในการใช้
        งาน
};
// PlayData.cpp
CPlayData::CPlayData()
{ // ไม่มีการโค้ดใดๆเนื่องจากเราไม่ได้ใช้การกำหนดค่าเริ่มต้นใดๆโดยใช้ประโยชน์จาก
คอนสตรักเตอร์
}
CPlayData::~CPlayData()
{
}
void CPlayData::addPlayData(char IdPlayer,char *PlayName,char IdToken)
{
// การโค้ดโปรแกรมเพื่อกำหนดค่าเริ่มต้นให้กับผู้เล่นแต่ละคน
        idplayer = IdPlayer; // กำหนดรับสถานะผู้เล่นเข้ามา
        for(int i=0;i<13;i++){
                playname[i] = PlayName[i]; // กำหนดรับค่าชื่อเข้ามา
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        }
        idtoken = IdToken; // กำหนดตัวเบี้ยเดิน
        //
        for(int j=0;j<40;j++){
                idtitled[j] = false; // กำหนดให้ยังไม่มีโฉนดใดๆ
        }
        njcard = 0; // กำหนดว่ายังไม่มีการ์ดออกจากคุกฟรี
}
```

โดยเวลานำไปใช้งานก็จะทำการประกาศ #include "PlayData.h" ไว้ที่ไฟล์ส่วนหัวของส่วนที่จะเอาไปใช้งาน จากนั้นใช้ทำการประกาศตัวแปรขึ้นมาใช้งานที่ไฟล์ส่วนหัวนั้นเช่นกันเพื่อประกาศเป็น public และนำมาใช้ได้ทุกส่วนในไฟล์สกุล cpp ของส่วนที่จะนำไปใช้งานนั้น โดยเราจะทำการประกาศเป็นพอยเตอร์ชี้ไปที่วัตถุของคลาส เพื่อความสะดวกในการใช้งาน โดยเฉพาะในส่วนของการส่งผ่านค่า เราเพียงประกาศตัวแปรตัวอื่นที่เป็นพอยเตอร์ชี้ไปที่วัตถุของคลาส แล้วให้ค่าตำแหน่งในหน่วยความจำแก่ตัวแปรนั้นไป ก็สามารถเข้าถึงข้อมูลของเราได้ ดังตัวอย่างการประกาศเพื่อนำมาใช้งานข้างล่าง

```
CPlayData* m_p1data; // ของผู้เล่นคนที่ 1
CPlayData* m_p2data; // ของผู้เล่นคนที่ 2
CPlayData* m_p3data; // ของผู้เล่นคนที่ 3
CPlayData* m_p4data; // ของผู้เล่นคนที่ 4
```

ก่อนการใช้งานจริงเราต้องมีการจองค่าพื้นที่หน่วยความจำให้กับมันเสียก่อน ในไฟล์สกุล cpp ของส่วนที่จะนำไปใช้งาน เช่น m_p1data = new CPlayData(); เป็นต้น เพื่อความสะดวกเราจะทำการกระทำส่วนนี้ไว้ที่คอนสตรักเตอร์ฟังก์ชันของไดอะล็อกที่จะนำไปใช้งานข้อมูลนี้ โดยเวลานำไปใช้งานเพื่อเข้าถึงข้อมูลต่างๆภายในวัตถุของคลาสนั้นๆทำได้โดย เช่น ต้องการข้อมูลจำนวนเงินของผู้เล่นคนแรก ใช้ int money = m_p1data.money; เป็นต้น

2. โครงสร้างข้อมูลของโฉนด โดยเราจะทำการสร้างคลาสของโฉนดหนึ่งใบ เพื่อเก็บข้อมูลต่าง ๆ เกี่ยวกับโฉนด แล้วในการใช้งานจริงเราจะประกาศการใช้งานเป็นอะเรย์ของคลาสโฉนด เช่น

   CtitleDeed titleded[40]; // ก็จะได้ที่เก็บข้อมูลของโฉนดจำนวน 40 ใบ เป็นต้น

   โดยคลาสข้อมูลของโฉนดมีรายละเอียดดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

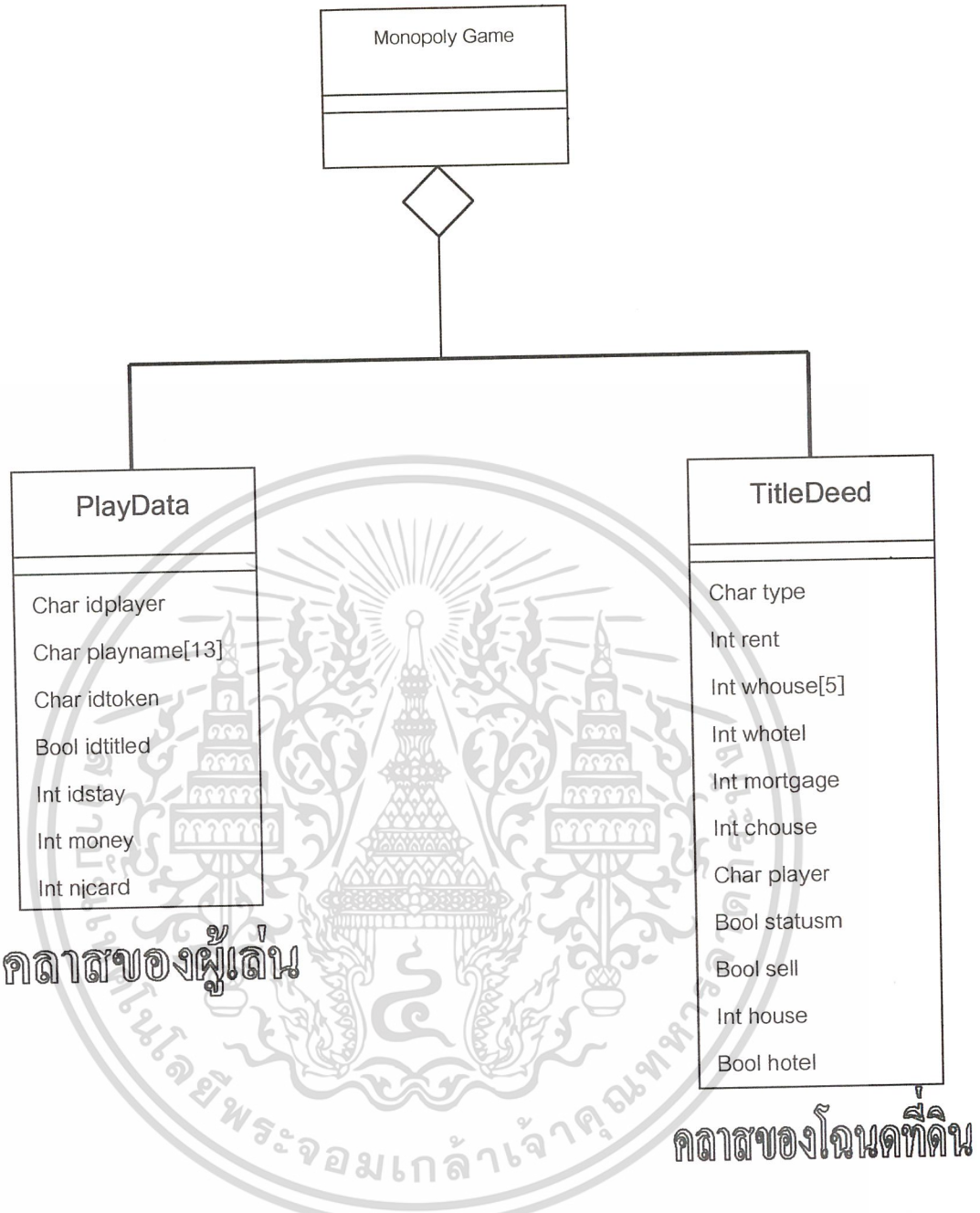
```
// TitleDeed.h
class CTitleDeed
{
public:
        CTitleDeed();
        virtual ~CTitleDeed();
public:
        char type; // เก็บประเภทของโฉนดเช่น โฉนดที่ดิน, โฉนดรถไฟ, โฉนด
        สาธารณูปโภค เป็นต้น
        int rent; // เก็บค่าเช่าที่ของโฉนดที่ดิน
        int whouse[5]; // เก็บค่าเช่าที่เมื่อมีบ้านจำนวนต่างๆ
        int whotel; // เก็บค่าเช่าที่เมื่อมีโรงแรม
        int mortgage; // เก็บราคาจำนองของที่ มีราคาเท่ากับครึ่งหนึ่งของราคาที่
        int chouse; // เก็บราคาซื้อบ้านของโฉนด
        //status of TitleDeed
        char player; // เก็บว่าโฉนดนี้ใครเป็นเจ้าของ
        bool statusm; // เก็บสถานะการจำนองที่ว่าจำนองหรือไม่
        bool sell; // เก็บสถานะว่าสามารถซื้อที่ได้หรือไม่
        int house; // เก็บจำนวนบ้านที่มีอยู่ในที่นั้นๆ
        bool hotel; // เก็บว่าที่นั้นๆมีโรงแรมหรือไม่
public:
        // ฟังก์ชันกำหนดค่าเริ่มต้นให้กับโฉนด
        void setData(char Type='N',int Mortgage=0,int Rent=0,int
        WHouse1=0,int WHouse2=0,int WHouse3=0,int WHouse4=0,int
        WHotel=0,int CHouse=0);
};
// TitleDeed.cpp
CTitleDeed::CTitleDeed()
{
}
CTitleDeed::~CTitleDeed()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
}
void CTitleDeed::setData(char Type,int Mortgage,int Rent,int WHouse1,int
WHouse2,int WHouse3,int WHouse4,int WHotel,int CHouse)
{
        type = Type; // กำหนดประเภทให้กับโฉนดนั้นๆ
        rent = Rent; // กำหนดค่าเช่าที่ดิน
        whouse[0] = 0;
        whouse[1] = WHouse1; // กำหนดค่าเช่าเมื่อมีบ้าน 1 หลัง
        whouse[2] = WHouse2; // กำหนดค่าเช่าเมื่อมีบ้าน 2 หลัง
        whouse[3] = WHouse3; // กำหนดค่าเช่าเมื่อมีบ้าน 3 หลัง
        whouse[4] = WHouse4; // กำหนดค่าเช่าเมื่อมีบ้าน 4 หลัง
        whotel = WHotel; // กำหนดค่าเช่าเมื่อมีโรงแรม
        mortgage = Mortgage; // กำหนดค่าจำนองที่
        chouse = CHouse; // กำหนดราคาซื้อบ้านของที่นั้นๆ
        // initial data
        statusm = false; // กำหนดว่าที่ยังไม่ได้จำนอง
        player = 'n'; // กำหนดว่ายังไม่มีใครเป็นเจ้าของ
        sell = true; // กำหนดว่าที่นั้นสามารถซื้อขายกันได้
        house = 0; // กำหนดว่าที่นั้นยังไม่มีบ้าน
        hotel = false; // กำหนดว่าที่นั้นยังไม่มีโรงแรม
}
```

ส่วนโครงสร้างข้อมูลอื่น ๆ เช่น เป็นบูลีน ไว้เก็บสถานะต่างๆของเกม, จำนวนเต็มไว้นับจำนวนครั้งที่ได้แต้มลูกเต๋าซ้ำกัน, อะเรย์ของจำนวนเต็ม ไว้เก็บค่าจุดเป็นพิกเซลของตำแหน่งต่าง ๆ ในตารางเกม เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.21 แสดงรูป Object Model ของโปรแกรม

3.6 วิธีการใช้เสียงประกอบในเกม

3.6.1 การเล่นเสียง Wave

เนื่องจากไฟล์เสียงของ Wave จะใช้เนื้อที่ในการเก็บข้อมูลมาก จึงเหมาะสมกับการเล่นไฟล์เสียงที่สั้นๆ แต่ไฟล์ wav สามารถที่จะเก็บข้อมูลเสียงได้ทุกแบบ เช่น เสียงคนพูด, เสียงฟ้าผ่า, เสียงดนตรี เป็นต้น

การเล่นเสียงจากไฟล์สกุล wav เราจะใช้ฟังก์ชัน sndPlaySound หรือ PlaySound ก็ได้ โดยมีรูปแบบการใช้งานดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น และไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

sndPlaySound(“Path_filename”,SND_SYNC);

เช่น sndPlaySound((“C:\\smile.wav”,SND_SYNC); เป็นต้น หรือ

PlaySound(“Path_filename”,0,SND_SYNC);

เช่น PlaySound(“res//happy.wav”,0,SND_SYNC);

แต่เราจะต้องทำการประกาศ #include “mmsystem.h” เข้าไปที่ไฟล์ส่วนต้นที่เรามีการเรียกใช้งานด้วย(*.cpp) จากนั้นเราจะต้องกำหนดไลบรารีที่เราจะทำการลิงก์ ไม่เช่นนั้นลิงก์เกอร์จะมองหาไลบรารีของเสียงไม่พบ ให้เราไปที่เมนู แล้วเลือก Projects->Settings จากนั้นไปที่แท็ป Link และใส่ชื่อ ไลบรารี winmm.lib ลงในช่อง Object/Library Modules

3.6.2 **การเล่นเสียง** MIDI Sequencer

ไฟล์เสียงของ MIDI จะมีขนาดเล็กกว่าไฟล์ของ Wave มากหลายเท่า เนื่องจากจะมีการใช้ตารางเสียงในการ์ดเสียงเวลาที่จะเล่นเสียงนั้นๆโดยจะเก็บเสียงในลักษณะเสียงต่างๆ เช่น เสียงกลอง, เสียงแตร เป็นต้น ดังนั้นไฟล์เสียง mid จึงเก็บข้อมูลได้เฉพาะเสียงทำนองดนตรี

เราสามารถใช้งาน MCI(Media Control interface) Window ในการเล่นไฟล์สกุล mid หรือ avi ที่มีเสียงได้ โดยมีรูปแบบการใช้งานดังต่อไปนี้

HWND hMCI = MCIWndCreate(m_hWnd, AfxGetApp()-> m_hInstance, รูปแบบการแสดงผล, "Path_filename”);

MCIWndPlay(hMCI); เช่น

HWND hMCI = MCIWndCreate(m_hWnd, AfxGetApp()->m_hInstance, MCIWNDF_NOPLAYBAR,"res//mind.mid");

MCIWndPlay(hMCI); หรือ

HWND hMCI = MCIWndCreate(m_hWnd, AfxGetApp()->m_hInstance, MCIWNDF_NOPLAYBAR, "C:\\running.avi");

MCIWndPlay(hMCI); เป็นต้น

โดยเราจะต้องทำการประกาศ #include “vfw.h” ไว้ที่ไฟล์ cpp ส่วนต้นที่มีการนำเสียง mid หรือ avi แบบมีเสียงไปใช้งาน จากนั้นทำการลิงก์โปรแกรมกับ vfw32.lib เพื่อที่จะใช้งานฟังก์ชันของ MCIWnd(MCI Window) API(Application Programming Interface) โดยการเลือกเมนูไปที่ Project->Setting แล้วเลือกแท็ป Link และใส่ในช่อง Object/Library Modules

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7 เครื่องมือที่ใช้ในการเขียนโปรแกรมเกม

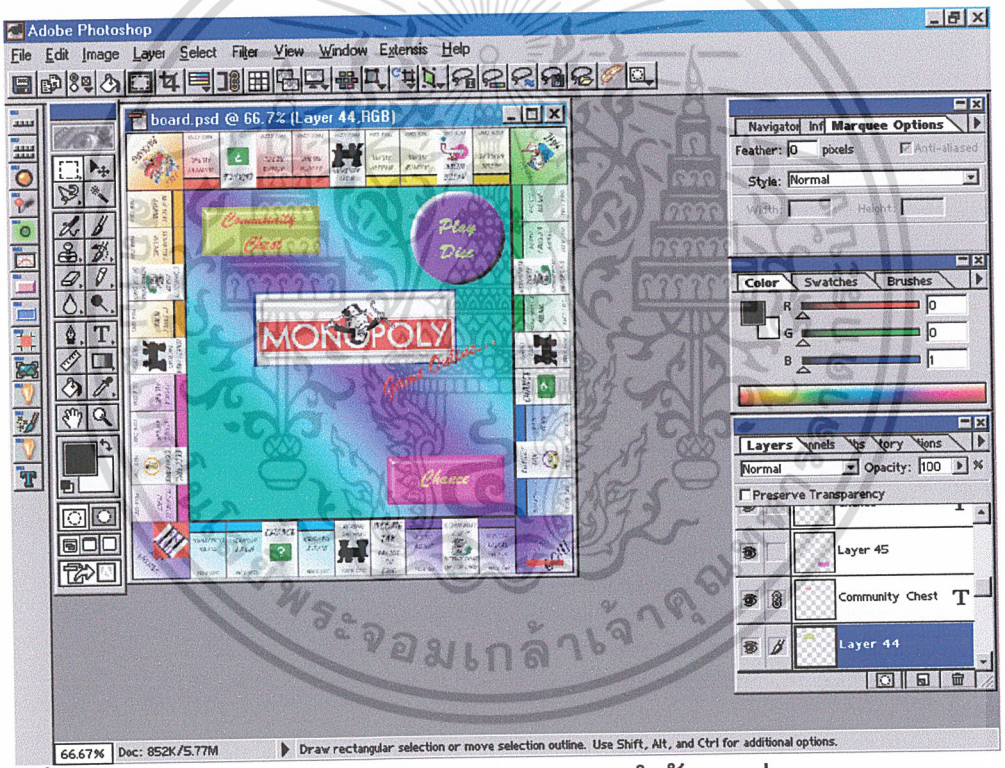
ในบทนี้จะกล่าวถึงเครื่องมือที่ใช้ในการเขียนโปรแกรมเกม โดยจะแบ่งเป็น 2 ส่วนใหญ่คือ

1. เครื่องมือสำหรับทำภาพ และเสียงประกอบเกม
2. เครื่องมือที่ใช้เขียนโปรแกรม ซึ่งโปรแกรมที่ใช้คือ Microsoft Visual C++ 6.0

3.7.1 เครื่องมือสำหรับทำภาพ และเสียงประกอบเกม

3.7.1.1 Adobe Photoshop 5.5

โปรแกรม Adobe Photoshop 5.5 ใช้ในการตกแต่งและออกแบบภาพต่าง ๆ เพื่อใช้ประกอบในเกม และใช้ Plug-in Extensis Photo Tools 3.03 เสริมเพื่อความสะดวกในการตกแต่งภาพ โดยภาพที่ออกแบบจะเก็บไว้ในนามสกุล bmp เพื่อให้โปรแกรม Microsoft Visual C++ 6.0 สามารถเรียกใช้ได้ โดยใช้ Microsoft Foundation Class หรือเรียกย่อว่า MFC



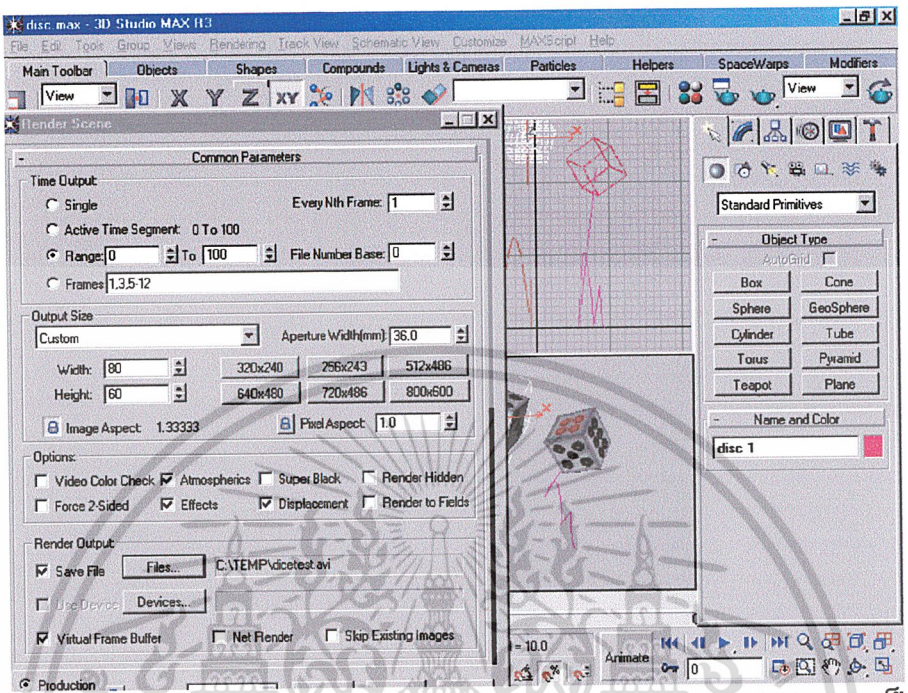
รูปที่ 3.22 โปรแกรม Adobe Photoshop 5.5 **ขณะกำลังตกแต่งและออกแบบภาพ**

3.7.1.2 3D Studio MAX Release 3

โปรแกรม 3D Studio MAX Release 3 ใช้สร้างภาพ 3 มิติ โดยมีขั้นตอนการสร้างภาพ คือเริ่มต้นด้วยการสร้างวัตถุ 3 มิติ แล้วทำการวางภาพลงบนพื้นผิววัตถุ 3 มิตินั้น นอกจากนี้ยังสามารถกำหนดเส้นทางการเคลื่อนที่พร้อมกำหนดการหมุนของวัตถุ 3 มิตินั้น เมื่อทำการออกแบบวัตถุนั้นเสร็จแล้ว ก็ทำการ Render โดยเลือก Render Output เป็นนามสกุล avi กำหนดขนาดของภาพตามความเหมาะสม จะได้ภาพเป็นไฟล์นามสกุล avi เพื่อให้โปรแกรม Microsoft

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Visual C++ 6.0 สามารถเรียกใช้ได้ โดยใช้ Media Control Interface หรือเรียกย่อว่า MCI ภาพเคลื่อนไหวในโปรแกรมเกมคือ ภาพการทอดลูกเต๋า

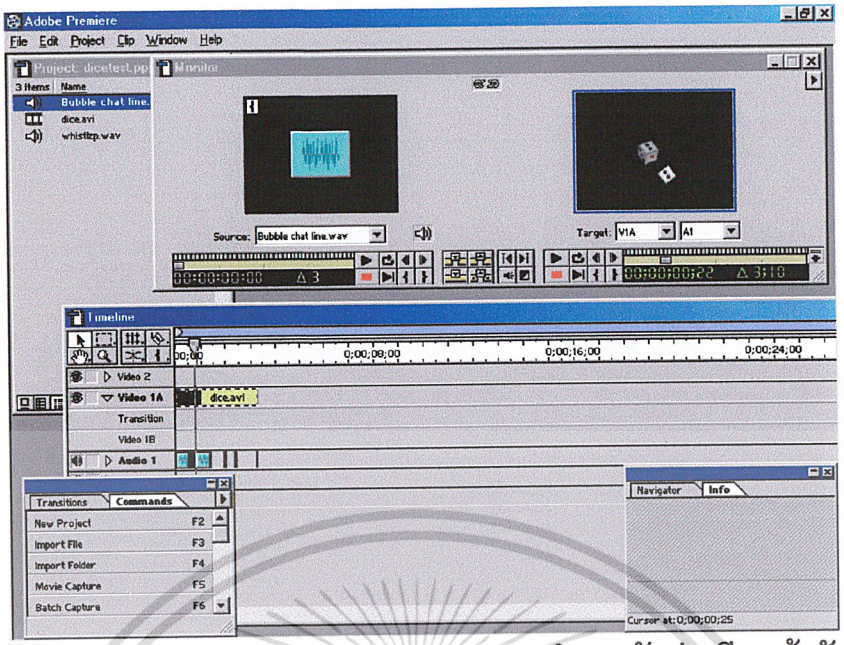


รูปที่ 3.23 โปรแกรม 3D Studio MAX Release 3 ขณะทำการ Render ภาพเป็นไฟล์นามสกุล avi

3.7.1.3 Adobe Premiere 5.1

ใช้เพื่อตัดต่อเสียงประกอบกับภาพเคลื่อนไหวให้เกิดความสัมพันธ์กัน ในโปรแกรมเกมคือ ภาพการทอดลูกเต๋าที่มีเสียงประกอบขณะลูกเต๋ากระดอน ภาพที่ตัดต่อประกอบเสียงแล้วจะ Export เป็น Movie ได้ไฟล์นามสกุล avi เพื่อให้โปรแกรม Microsoft Visual C++ 6.0 สามารถเรียกใช้ได้ โดยใช้ Media Control Interface หรือเรียกย่อว่า MCI

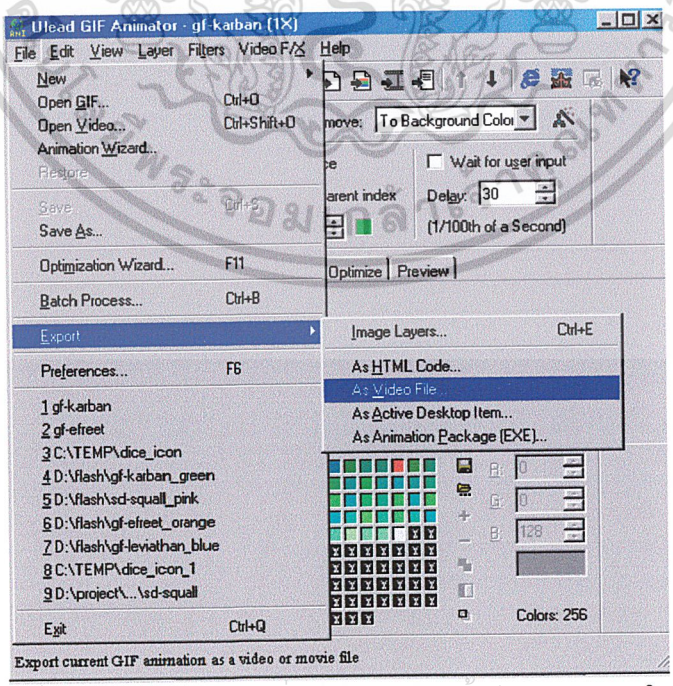
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.24 โปรแกรม Adobe Premiere 5.1 ขณะทำการตัดต่อเสียงเข้ากับภาพ

3.7.1.4 Ulead GIF Animator Version 4.0

โปรแกรม Ulead GIF Animator Version 4.0 ใช้สร้างภาพเคลื่อนไหวโดยแปลงภาพเคลื่อนไหวจากนามสกุล gif แปลงเป็นนามสกุล avi เพื่อให้โปรแกรม Microsoft Visual C++ 6.0 สามารถเรียกใช้ได้ โดยใช้ Media Control Interface หรือเรียกย่อ ว่า MCI ภาพเคลื่อนไหวที่ใช้ในเกมนี้คือ ภาพตัวเบี้ยหรือตัวเดินในเกม



รูปที่ 3.25 โปรแกรม Ulead GIF Animator Version 4.0 ขณะจะทำการ Export เป็นนามสกุล avi

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

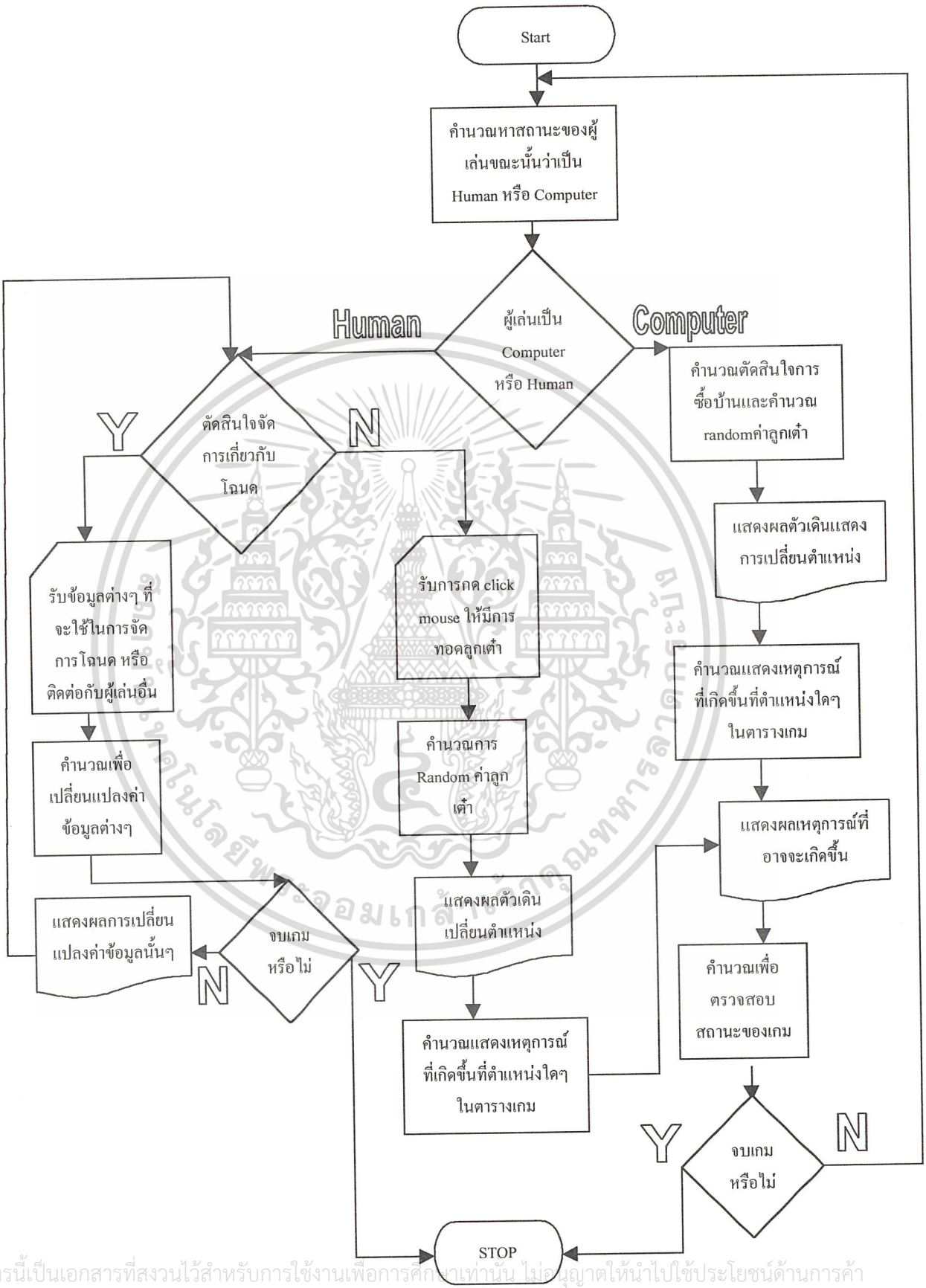
3.7.2 Microsoft Visual C++ 6.0

Visual C++ 6.0 เป็นผลิตภัณฑ์ล่าสุดและดีที่สุดของ Visual C/C++ คอมไพเลอร์ และมีอะไรที่มากกว่าแค่เป็นเพียงคอมไพเลอร์ มันยังประกอบไปด้วย Microsoft Foundation Class ที่มีการใช้งานที่ง่ายและรวดเร็วสำหรับพัฒนาโปรแกรมบนวินโดว์, resource editors ที่ไว้ใช้สำหรับออกแบบไดอะล็อกบ็อก, เมนู, ทูลบาร์, รูปภาพ (ไม่เกิน 256 สี)และส่วนประกอบอื่นๆที่ วินโดว์ปัจจุบันมี ซึ่งเป็นสภาพแวดล้อมในการพัฒนาแบบรวมที่ดีที่เรียกว่า Developer Studio ที่จะแสดงให้เห็นการเปลี่ยนแปลงที่สามารถเห็นได้ในขณะที่พัฒนาโปรแกรม และรวมเครื่องมือในการดีบักที่ทำให้ตรวจสอบแก้ไขได้ทุกจุดที่เกิดบักช่วงเวลาที่รันโปรแกรม นี่เป็นเพียงแค่ส่วนเล็กน้อยเท่านั้นของลักษณะทั้งหมดของ Visual C++ 6.0 ที่ช่วยทำให้การพัฒนาโปรแกรมบนวินโดว์เป็นไปด้วยความรวดเร็ว

โดยรายละเอียดอื่นๆเกี่ยวกับ Visual C++ 6.0 สามารถหารายละเอียดเพิ่มเติมได้ในบทที่ 2 หัวข้อที่ 2.2 และบทที่ 3 หัวข้อที่ 3.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.8 Flow Chart of Monopoly Game Online



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.26 Flow Chat of MONOPOLY GAME ONLINE

บทที่ 4
บทผลการทดลองหรือการวิเคราะห์ข้อมูล

จากการศึกษา รวบรวมข้อมูล และวิเคราะห์ถึงปัญหาที่เกิดขึ้นในระหว่างการทำงานในปัญหาพิเศษนี้ พบว่าเราสามารถสรุปผลออกมาได้เป็น 4 ข้อใหญ่ ๆ ดังนี้

4.1 โครงสร้างของ Algorithm ของเกม Monopoly

เนื่องจากเกมเศรษฐีนี้มีพื้นฐานมาจากเกมเศรษฐีแบบกระดาน ซึ่งมีกติกา และข้อปลีกย่อยมากมาย (ไม่ว่าจะเป็น เงื่อนไขของการซื้อที่ดิน เงื่อนไขการเสียค่าเช่าที่ดิน และ กระบวนการแลกเปลี่ยนโฉนด ฯลฯ) อันจะทำให้โครงสร้างของ Algorithm ของเกมมีความซับซ้อนมากขึ้น เพื่อที่จะทำให้เกม ครอบคลุมเงื่อนไขทั้งหมดที่มีอยู่ ด้วยเหตุนี้การออกแบบ Algorithm ของเกมจึงต้องกระทำด้วยความรอบคอบ และต้องคอยตรวจสอบผลการทำงานของเกมไปทุกระยะ เพื่อตรวจหาข้อผิดพลาดที่อาจจะเกิดขึ้น

4.2 การทำงานของเกมในส่วนของการเล่นกับคอมพิวเตอร์

การที่เราจะออกแบบ Algorithm ของเกมโดยให้คอมพิวเตอร์สามารถเล่นกับคนได้นั้น มีความสำคัญอย่างยิ่ง เนื่องจากมีเงื่อนไขมากมายภายในเกมนี้ การที่เราจะพัฒนาให้คอมพิวเตอร์มีความสามารถประพฤติตนอย่างชาญฉลาด (Artificial Intelligent : AI) ให้สามารถรู้จักที่จะเอาชนะอย่างมีเหตุผลกับผู้เล่นที่เป็นมนุษย์นั้น จำเป็นที่จะต้องค้นหาวิธีเล่นทั้งหมดที่เป็นไปได้ของเกมนี้ (Search Tree) ภายใต้ข้อกำหนดและกติกาของเกม ซึ่งมีขนาดใหญ่มาก และเกินขอบเขตของเวลาที่มีอยู่ ด้วยเหตุนี้ การพัฒนา Algorithm ของคอมพิวเตอร์ให้สามารถเล่นกับมนุษย์ได้นั้น จึงเป็นไปในลักษณะของการตรวจสอบเงื่อนไขของการเล่นของคอมพิวเตอร์เองให้ตอบสนองกับผู้เล่นโดยอยู่ภายใต้กติกาของเกม ดังนั้นในการออกแบบ Algorithm ของคอมพิวเตอร์ให้สามารถเล่นกับมนุษย์ได้นั้น จึงเป็นการตรวจสอบเงื่อนไขและใช้วิธีการสุ่มให้คอมพิวเตอร์ตัดสินใจที่จะกระทำการต่างๆภายในเกมและโต้ตอบกับผู้เล่นได้ ซึ่งทำให้ Algorithm ลดความซับซ้อนลงไปได้มาก

4.3 การออกแบบให้เกมสามารถเล่นบนระบบเครือข่ายคอมพิวเตอร์

ในส่วนของการทำงานของเกมที่สามารถเล่นกับผู้เล่นคนอื่น ๆ ได้ โดยผ่านระบบเครือข่ายนั้น จะใช้หลักการทำงานตามแบบของ Client-Server โดยทั้ง Client และ Server จะติดต่อกันผ่าน Window Socket ซึ่งเป็น Interface ที่ถูกใช้บนระบบปฏิบัติการ Window ที่สามารถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จัดการ การรับส่งข้อมูลระหว่างคอมพิวเตอร์ 2 เครื่องที่เชื่อมต่อกันได้อย่างมีประสิทธิภาพ โดยรับส่งผ่านโปรโตคอล TCP/IP ซึ่งเป็นโปรโตคอลหลักของระบบเครือข่าย ด้วยเหตุนี้เกมเศรษฐีระบบออนไลน์ จึงสามารถเล่นร่วมกันได้กับผู้เล่นคนอื่น ๆ ที่ใดก็ได้ผ่านระบบเครือข่ายที่เชื่อมต่อด้วยโปรโตคอล TCP/IP ซึ่งเป็นข้อดีที่จะทำให้เกมนี้จะเป็นที่นิยมต่อไปได้ในอนาคต อันเนื่องมาจากเกมมีความง่ายที่จะติดตั้ง และมีข้อดีที่สามารถเล่นกับผู้เล่นคนอื่น ๆ ได้ผ่านระบบเครือข่ายที่สื่อสารกันผ่านโปรโตคอล TCP/IP ซึ่งเป็นโปรโตคอลที่กำลังได้รับความนิยมและมีการใช้งานอยู่บนระบบเครือข่ายส่วนใหญ่ในปัจจุบัน

4.4 การออกแบบกราฟฟิก และเสียงประกอบเกม

การทำงานของเกมนั้น ส่วนกราฟฟิก และเสียงประกอบเกมนับว่ามีความสำคัญที่จะช่วยเพิ่มอรรถรสของการเล่น ทำให้ผู้เล่นไม่รู้สึกเบื่อหน่าย สำหรับเกมเศรษฐีนี้ได้ใช้เครื่องมือหลายอย่างในการพัฒนา และสร้างภาพกราฟฟิกแบบต่าง ๆ รวมทั้ง การสร้างภาพเคลื่อนไหว ที่เกิดขึ้นภายในเกม เครื่องมือที่ใช้ในการสร้างภาพกราฟฟิก ได้แก่ โปรแกรมตกแต่งรูปภาพ Adobe Photoshop โปรแกรม Adobe Premiere โปรแกรม GIF Animation และโปรแกรม 3D Studio MAX

โดยโปรแกรม 3D Studio MAX นั้น สามารถสร้างภาพกราฟฟิกได้เสมือนจริงในแบบ 3 มิติ อีกทั้งยังสามารถกำหนดการเคลื่อนไหวของภาพให้สมจริง ด้วยเหตุนี้ การทำงานของเกมในส่วนกราฟฟิก จึงใช้ 3D Studio MAX ในการสร้างภาพลักษณะ 3 มิติ และในการทำภาพเคลื่อนไหวในลักษณะ 2 มิตินั้นจะใช้โปรแกรม GIF Animation ในการออกแบบคือเป็นโปรแกรมที่นำเอาภาพหลาย ๆ ภาพมาแสดงในช่วงเวลาที่ต่อกันทำให้เกิดภาพเคลื่อนไหวแล้วทำการ Export ภาพเคลื่อนไหวนั้นออกมาเป็นไฟล์นามสกุล avi การออกแบบภาพหรือตกแต่งภาพจะใช้โปรแกรม Adobe Photoshop ในการทำงาน นอกจากนี้ยังใช้โปรแกรม Adobe Premiere ในการตัดต่อเสียงเข้ากับภาพเคลื่อนไหว แล้ว Export เป็นไฟล์นามสกุล avi ส่วนของเสียงประกอบเกมอื่น ๆ เช่น เพลง เสียงคลิกจะอยู่ในส่วนของการเขียนโปรแกรมเรียกใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทการวิจารณ์หรืออภิปรายผล

จากการได้ทำการพัฒนาโปรแกรมเกมเศรษฐีผ่านระบบเครือข่าย ผู้จัดทำพบว่า ในส่วนที่มีความสำคัญในการทำให้งานสำเร็จลุล่วงมาได้อย่างหนึ่งคือ การออกแบบโครงสร้างของโปรแกรม ซึ่งผู้จัดทำนั้นได้ทำการคิดในการออกแบบโครงสร้างให้ละเอียดถี่ถ้วน และคิดถึงเหตุผลความเป็นไปได้ที่การออกแบบนั้นจะสามารถเป็นการออกแบบของการพัฒนาโปรแกรมได้ จึงค่อยตัดสินใจลงมือทำการเขียนและพัฒนาโปรแกรม หากมีการออกแบบที่ไม่ดี ผู้จัดทำคิดว่าอาจจะมีปัญหาตามมาหลายอย่าง เช่น การตรวจสอบแก้ไขอาจจะทำได้ยาก, อาจจะมีการล้มเหลวระหว่างการทำงานเนื่องจากการออกแบบเดิมนั้นไม่เพียงพอต่อการพัฒนาโปรแกรมต่อไป ทำให้จะต้องมีการออกแบบโปรแกรมใหม่หรือออกแบบเพิ่มเติม, อาจจะเกิดข้อผิดพลาดในการโปรแกรมได้ง่าย เนื่องจากการออกแบบที่ไม่ดี จะทำให้โครงสร้างโปรแกรมดูมีความไม่เป็นระเบียบ ตรวจสอบแก้ไขเพิ่มเติมได้ยาก เป็นต้น ในการพัฒนาโปรแกรมเกมเศรษฐีนั้น จำเป็นจะต้องมีเงื่อนไขในการตัดสินใจมากพอสมควร เนื่องมาจาก กฎและกติกาที่ผู้พัฒนาโปรแกรมเลือกนำมาใช้ในการเขียนโปรแกรมนั้น มีความซับซ้อนบ้างในการคิดหาเงื่อนไขในการตัดสินใจกำหนดความเป็นไปของเกมเพื่อให้เกมมีความถูกต้องตามกฎและกติกาที่ตั้งไว้ ส่วนปัญหาอีกอย่างหนึ่งที่ผู้พัฒนาโปรแกรมเห็นว่ามีความสำคัญในการเขียนโปรแกรม คือผู้เขียนโปรแกรมควรที่จะศึกษาถึงทฤษฎีและโครงสร้างของภาษานั้นๆให้มากพอสมควร เพราะหากไม่มีการศึกษาที่ดีพอ อาจจะทำให้โปรแกรมที่พัฒนามีความยากในการเขียนโปรแกรมและมีความซับซ้อนมากขึ้นได้ เช่น ในการเลือกการใช้โครงสร้างข้อมูลในการเขียนโปรแกรม หากมีการศึกษามาไม่ครอบคลุมเนื้อหาของภาษานั้นๆ ก็อาจจะทำการเลือกใช้แต่โครงสร้างข้อมูลที่ไม่เหมาะสมกับลักษณะของข้อมูลที่ต้องการใช้งานทั้งๆที่จริงแล้วอาจจะมีโครงสร้างข้อมูลที่เหมาะสมที่จะจัดเก็บข้อมูลลักษณะนั้นมากกว่า เป็นต้น อีกเหตุผลหนึ่งที่สนับสนุนความคิดนี้และเกิดขึ้นจริงกับผู้จัดทำเช่น ผู้จัดทำเป็นกลุ่มของผู้เริ่มต้นลองใช้ Visual C++ 6.0 ทำให้เกิดปัญหามากมายพอสมควรเหมือนกัน โดยเฉพาะในเรื่องการแสดงผลภาพและเสียงของเกมที่ผู้จัดทำไม่ทราบว่าจะใช้วิธีใดจึงจะเหมาะสม แต่ผู้จัดทำก็แก้ไขให้เพียงแต่มีการแสดงผลที่สื่อความหมายให้กับผู้ใช้ได้ เป็นต้น ในการออกแบบภาพกราฟิกและเสียงนั้นมีความจำเป็นที่จะต้องใช้โปรแกรมสำเร็จรูปในการออกแบบ เช่น Adobe Photoshop, 3D Studio Max เป็นต้น เนื่องจากโปรแกรมเหล่านี้มีประสิทธิภาพในการใช้งานที่ดี และเป็นที่นิยมของผู้ใช้งานทั่วไป อีกทั้งยังมีความสะดวกรวดเร็วในการทำงานซึ่งดีกับผู้จัดทำเป็นอย่างมาก เนื่องจากต้องจัดทำภาพประกอบมากพอสมควร ในส่วนของการติดต่อสื่อสารผ่านระบบเครือข่ายก็มีปัญหาพอสมควร เนื่องจากผู้จัดทำยังรู้ไม่ครอบคลุมวิธีการใช้งานคลาส CSocket ของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Microsoft Foundation Class แต่ก็สามารถที่จะแก้ไขปัญหาจนโปรแกรมสำเร็จลุล่วงมาได้ ดังนั้น ในการพัฒนาโปรแกรม ผู้พัฒนาโปรแกรมควรที่มีความพร้อมทั้งทางด้านทฤษฎีและปฏิบัติจนเกิดความรู้ความชำนาญ เพื่อที่ความผิดพลาด และปัญหาต่างๆที่เกิดขึ้นในการพัฒนาโปรแกรมแต่ละครั้งจะได้เป็นประสบการณ์ที่ดีที่จะทำให้ผู้พัฒนาโปรแกรมเหล่านั้น เป็นผู้เชี่ยวชาญในการพัฒนาโปรแกรม และเป็นทรัพยากรที่มีค่าในสังคมต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
สรุปผลการศึกษาและข้อเสนอแนะ

6.1 สรุปผลการศึกษา

6.1.1 การศึกษารวบรวมข้อมูล

การศึกษารวบรวมข้อมูลทำโดยการศึกษาความเป็นไปได้ของเกมว่าสามารถที่จะพัฒนาด้วยการ เขียนโปรแกรมให้เป็นเกมบนเครื่องคอมพิวเตอร์ได้หรือไม่ เมื่อตัดสินใจเลือกแล้ว ก็ทำการศึกษารวบรวบข้อมูลต่างๆของเกมนั้นๆ เช่น กฎและกติกา เป็นต้น อีกทั้งยังต้องศึกษาค้นคว้าในเรื่องวิธีการต่างๆที่ใช้ในการเขียนโปรแกรมเกม ทั้งหนังสือ หรือศึกษาจากตัวเกมจริงที่มีอยู่ทั่วไป เพื่อที่จะนำเอาข้อมูลต่างๆที่ได้มานั้นมาเป็นเครื่องมือช่วยในการติดสินใจและคิดค้นต่างๆ เช่น การออกแบบหน้าตาเกม, การออกแบบโครงสร้างและวิธีการของเกม, การเลือกเครื่องมือต่างๆที่จะใช้ในการพัฒนาโปรแกรมเกม เป็นต้น

6.1.2 การวิเคราะห์และการออกแบบเกม

โดยการนำข้อมูลต่างๆที่รวบรวมมาประกอบกับความรู้ความสามารถที่มีอยู่ที่ได้มาจากการศึกษาและประสบการณ์ มาเป็นเครื่องมือในการช่วยในการวิเคราะห์หารูปแบบของเกมที่สามารถจะเป็นไปได้ จากข้อมูลและความรู้ต่างๆที่มีอยู่ แล้วทำการวางโครงสร้างและออกแบบหน้าตา รวมทั้งวิธีการต่างๆของเกมและการกำหนดขอบเขตเพื่อให้สามารถทำงานให้เสร็จตามระยะเวลาที่กำหนดได้ จึงทำการแบ่งงานและมอบหมายหน้าที่ให้แต่ละคนในกลุ่มจัดการ

6.1.3 การพัฒนาโปรแกรมคอมพิวเตอร์

พัฒนาด้วย Microsoft Visual C++ 6.0 ในชุดของ Microsoft Visual Studio 6.0 มาพัฒนาในส่วนของการเขียนโปรแกรมเกมทั้งหมด รวมทั้งในส่วนการติดต่อสื่อสารผ่านระบบเครือข่าย โดยใช้คลาส CSocket ของ Microsoft Foundation Class ของ Visual C++ 6.0 และการสร้างภาพประกอบและเสียงประกอบเกมได้ใช้โปรแกรมต่างๆมาช่วย เช่น Adobe Photoshop, 3D Studio Max เป็นต้น

6.1.4 คุณสมบัติของเกมมีดังต่อไปนี้

- เป็นเกมที่เล่นบนระบบปฏิบัติการ Windows 95/98
- เป็นเกมที่สามารถเล่นพร้อมกันได้ไม่เกิน 4 คน ผ่านระบบเครือข่ายที่เชื่อมต่อกันด้วยโปรโตคอล TCP/IP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ผู้เล่นสามารถเลือกเล่นกับคอมพิวเตอร์ได้
- มีภาพกราฟิก ภาพเคลื่อนไหว และเสียงประกอบเกม
- สามารถใส่ชื่อผู้เล่น และเลือกลักษณะของตัวเบี้ยได้
- ผู้เล่นสามารถติดต่อโดยการพิมพ์ข้อความส่งให้กันได้

6.2 ข้อเสนอแนะ

ข้อเสนอแนะต่างๆ มีดังต่อไปนี้

- การแสดงภาพและภาพเคลื่อนไหวต่างๆในเกมสามารถพัฒนาให้มีความสวยงามและดูสมจริงขึ้นได้ โดยในปัญหาพิเศษนี้ผู้จัดทำได้ใช้การแสดงภาพ bitmap ในการแสดงภาพต่างๆในโปรแกรม แล้วทำการลบภาพนั้นเมื่อไม่ต้องการแสดงภาพเหล่านั้นด้วยฟังก์ชันการวาดวินโดว์ใหม่ ทำให้สังเกตเห็นการเปลี่ยนแปลงจากการวาดวินโดว์ใหม่นี้ทั้งไดอะล็อก ทำให้ดูไม่สวยงามและอาจเป็นที่รำคาญสายตาของผู้ใช้งานโปรแกรมได้ ซึ่งสามารถแก้ไขปัญหาเหล่านี้ได้โดยการใช้เก็บทุก bit ของภาพที่เป็นพื้นไว้ในบัฟเฟอร์ก่อนการแสดงภาพทุกครั้ง และเมื่อไม่ต้องการแสดงภาพเหล่านี้ก็สามารถทำได้โดยการเรียกภาพพื้นหลังจากบัฟเฟอร์ขึ้นมาแสดงได้ ส่วนการแสดงการเปลี่ยนตำแหน่งของตัวเบี้ยเดินในตารางเกมที่ปัญหาพิเศษนี้ใช้ไฟล์ avi ในการแสดงการเคลื่อนไหว แต่การเคลื่อนไหวจะแสดงได้เมื่ออยู่กับที่เท่านั้นและต้องมีการวาดวินโดว์ใหม่เมื่อไม่ต้องการแสดงตัวเบี้ยเดินที่จุดใดๆอีกซึ่งจะทำให้เกิดปัญหาเช่นเดียวกับที่กล่าวมาแล้ว อีกทั้งยังติดภาพพื้นหลังของไฟล์ avi (สามารถดูภาพตัวอย่างได้ในภาคผนวก หัวข้อ 3.2) ทำให้ดูไม่สมจริงขัดกับภาพพื้นหลังของเกมที่เป็นตารางเกม วิธีแก้ไขก็สามารถทำได้เช่นเดียวกับที่กล่าวมาแล้วคือทำการเก็บภาพพื้นหลังไว้ในบัฟเฟอร์ก่อนการแสดงภาพทุกครั้งและเมื่อไม่ต้องการแสดงก็นำภาพจากบัฟเฟอร์มาแสดงที่ตำแหน่งเดิมของภาพพื้นนั้น แต่ต้องมีการออกแบบภาพแต่ละเฟรมในภาพเคลื่อนไหวที่ต้องการนำมาแสดงเป็นตัวเบี้ยเดิน (เช่นเดียวกับภาพแต่ละเฟรมในการทำภาพเคลื่อนไหวของภาพยนต์การ์ตูนต่างๆ) นำมาแสดงทีละภาพเริ่มจากภาพแรก และเมื่อต้องการแสดงภาพถัดไปก็ให้เปลี่ยนจุดบนจอภาพที่จะใช้แสดงผลด้วย ทำเช่นนี้จนหมดภาพทุกภาพ แล้ววนแสดงใหม่โดยเริ่มจากภาพแรก แต่จุดการแสดงผลก็ยังคงต้องเปลี่ยนไปตามทิศทางการเคลื่อนไหว ทำให้ได้ภาพตัวเบี้ยเคลื่อนไหวไปยังจุดที่ต้องการที่สวยงามและดูสมจริงได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สามารถที่จะเพิ่มเติม แก้ไขดัดแปลง เงื่อนไข กฎและกติกาของเกม (โดยสามารถดูกฎและกติกาต่างๆของเกมของปัญหาพิเศษนี้ได้ที่บท 2 หัวข้อ 2.3) ให้ดูมีความเหมาะสมมากยิ่งขึ้นได้ตามความต้องการ
- ปัจจุบันเทคโนโลยีทางด้านคอมพิวเตอร์ ระบบการสื่อสารข้อมูล และอินเตอร์เน็ตกำลังพัฒนาและก้าวหน้าอย่างรวดเร็ว ประเทศไทยยังขาดแคลนผู้ที่มีความรู้ความสามารถอย่างแท้จริงในการเผชิญหน้ากับสภาวะการเปลี่ยนแปลงอย่างรวดเร็วนี้มาก ดังนั้นถ้าคนไทยสามารถพัฒนาและเรียนรู้ในเรื่องต่างๆในทุกๆด้านให้เข้าใจและนำไปใช้ได้ ก็จะมีประโยชน์ต่อการช่วยประเทศให้รับมือกับความเปลี่ยนแปลงต่าง ๆ ได้ การพัฒนาโปรแกรมเกมก็เป็นอีกด้านหนึ่งที่เมืองไทยมีการพัฒนาในด้านนี้อยู่น้อยมาก จึงเป็นสิ่งที่น่าสนใจที่คนไทยจะหันมาให้ความสนใจในการพัฒนาในด้านนี้บ้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

ส่วนประกอบของเกมและวิธีการเล่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.
ส่วนประกอบของเกมและวิธีการเล่น

1. ระบบที่ใช้

- Microsoft Windows 95/98
- Pentium 66 MHz
- Sound Card
- Network Card กรณีเล่นผ่านระบบเครือข่าย

2. กฎกติกาเบื้องต้น

ผู้เล่นคนแรกทำการทอดลูกเต๋า แล้วเดินไปตามช่องให้เท่ากับจำนวนแต้มที่ทอดลูกเต๋าได้ ถ้าผู้เล่นไปหยุด ณ ที่ดินที่ยังว่าง ผู้เล่นมีสิทธิ์จะซื้อที่ดินได้จากธนาคารตามราคาที่ระบุไว้ในช่องนั้น แล้วได้โฉนดที่ดินมาถือเป็นกรรมสิทธิ์ ถ้าที่ดินนั้นมีเจ้าของแล้วผู้ที่เดินไปตกช่องที่ดินนั้นจะต้องจ่ายค่าเช่าที่ดินให้แก่เจ้าของโฉนดนั้น เจ้าของที่ดินจะมีสิทธิ์ซื้อสิ่งปลูกสร้างได้ก็ต่อเมื่อเป็นเจ้าของที่ดินในกลุ่มสีเดียวกันทั้งหมด

ถ้าผู้เล่นตกลงช่องที่ไม่ใช่ที่ดิน คือ ช่อง COMMUNITY CHEST, INCOME TAX, CHANCE, GO TO JAIL, LUXURY TAX, หรือผ่านช่อง GO ผู้เล่นจะต้องทำตามเงื่อนไขของแต่ละช่องกำหนดไว้ โดยคอมพิวเตอร์จะเป็นผู้จัดการให้ทำตามเงื่อนไขนั้น

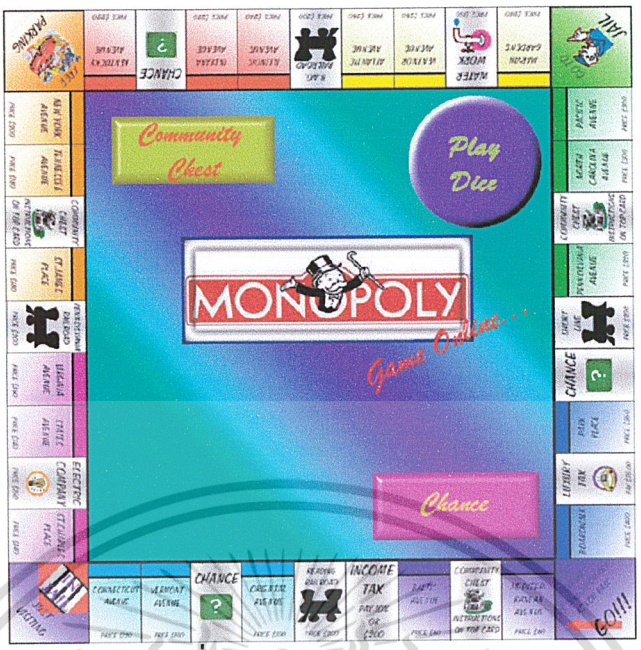
ผู้เล่นสามารถขอซื้อที่ดินได้จากผู้เล่นอื่น และสามารถจัดการกับที่ดินของตัวเองได้เช่น ซื้อสิ่งปลูกสร้าง ได้ก่อนการทอดลูกเต๋า เพราะหลังจากทำการทอดลูกเต๋าเสร็จแล้ว หากไม่ติดเงื่อนไขต้องทำการทอดลูกเต๋าใหม่ จะเป็นการเปลี่ยนผู้เล่นทันที

3. ส่วนประกอบของเกม

3.1 กระดาน

เป็นส่วนที่แสดงตำแหน่งที่ดินต่าง ๆ รวมถึงสิ่งปลูกสร้างได้แก่ บ้านและโรงแรม และแสดงตำแหน่งของตัวเบี้ยของผู้เล่นแต่ละคน

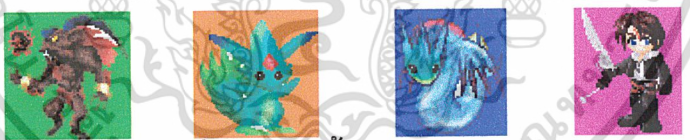
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.1 กระดานเกม

3.2 ตัวเบี้ยหรือตัวเดิน

ตัวเบี้ยประจำผู้เล่นแต่ละคน ผู้เล่นสามารถเลือกตัวเบี้ยได้ก่อนการเล่นเกมโดยผู้เล่นอาจเลือกตัวเบี้ยเหมือนกันได้ แต่จะแตกต่างกันที่สีพื้น คือจะมีสีเขียว, สีส้ม, สีน้ำเงิน และสีชมพู ตัวเบี้ยจะเป็นลักษณะภาพเคลื่อนไหว



รูปที่ 7.2 ตัวเบี้ยหรือตัวเดิน

3.3 สิ่งปลูกสร้าง

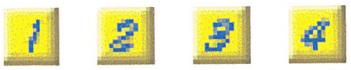
สิ่งปลูกสร้างบนที่ดิน ได้แก่บ้านและโรงแรม

การซื้อบ้าน จะซื้อบ้านมาไว้ในที่ดินได้ก็ต่อเมื่อเป็นเจ้าของโฉนดที่ดินในกลุ่มสีเดียวกันทั้งหมด และถ้าจะซื้อบ้านหลังที่ 2 จะต้องมีบ้านหลังที่ 1 อยู่ในที่ดินในกลุ่มสีเดียวกันทั้งหมดก่อน การซื้อบ้านหลังที่ 3 และ 4 ก็เช่นเดียวกันจะต้องมีบ้านหลังที่ 2 และ 3 ก่อน (ตามลำดับ) ในที่ดินกลุ่มสีเดียวกันทั้งหมดก่อน ราคาบ้านขึ้นอยู่ตามโฉนดของแต่ละที่ดิน

การซื้อโรงแรม จะต้องมีบ้าน 4 หลังบนที่ดินกลุ่มสีเดียวกันทั้งหมดก่อน จึงจะทำการซื้อโรงแรมได้ ราคาโรงแรมขึ้นอยู่ตามโฉนดของแต่ละที่ดิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การขายบ้านและโรงแรม ถ้ามีโรงแรมจะต้องทำการขายโรงแรมก่อนจึงจะขายบ้านได้ ถ้าขายโรงแรมแล้วจะได้บ้านคืนมา 4 หลัง การขายบ้านและโรงแรมจะต้องทำการขายในลักษณะย้อนกลับกับการซื้อ เช่น ถ้ามีบ้านอยู่ 4 หลังในที่ดินกลุ่มสีเดียวกัน ถ้าเราต้องการขายบ้านหลังที่ 3 บนที่ดินนั้น จะต้องขายบ้านหลังที่ 4 ในกลุ่มสีเดียวทั้งหมดก่อนจึงจะขายบ้านหลังที่ 3 ได้ ราคาที่ขายสิ่งปลูกสร้างจะเท่ากับครึ่งหนึ่งของราคาที่ซื้อมา



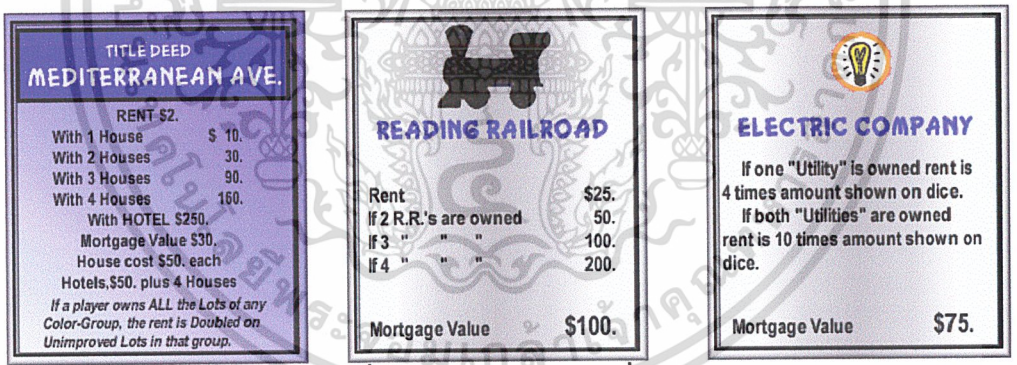
รูปที่ 7.3 บ้าน จำนวน 1, 2, 3 และ 4 หลังตามลำดับ



รูปที่ 7.4 โรงแรม

3.4 โฉนดที่ดิน

โฉนดที่ดินจะแสดงรายละเอียดของที่ดินไว้ว่า จะเก็บค่าเช่าเท่าใดหากผู้เล่นอื่นเดินมาตกที่ดินนั้น ราคาบ้าน ราคาโรงแรม และราคาจำนองที่ดิน

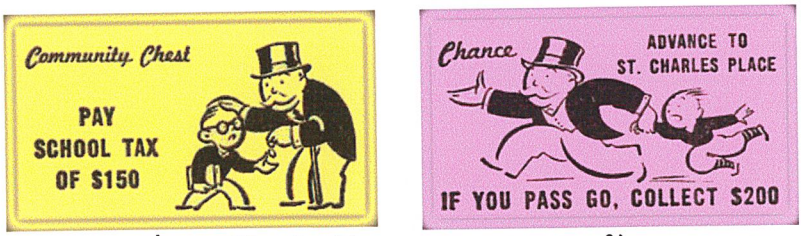


รูปที่ 7.5 ตัวอย่างโฉนดที่ดิน

3.5 บัตร COMMUNITY และบัตร CHANCE

เมื่อผู้เล่นเดินตกช่อง COMMUNITY และ CHANCE คอมพิวเตอร์จะสุ่มหยิบบัตรให้และผู้เล่นต้องทำตามเงื่อนไขในบัตรที่สุ่มได้นั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



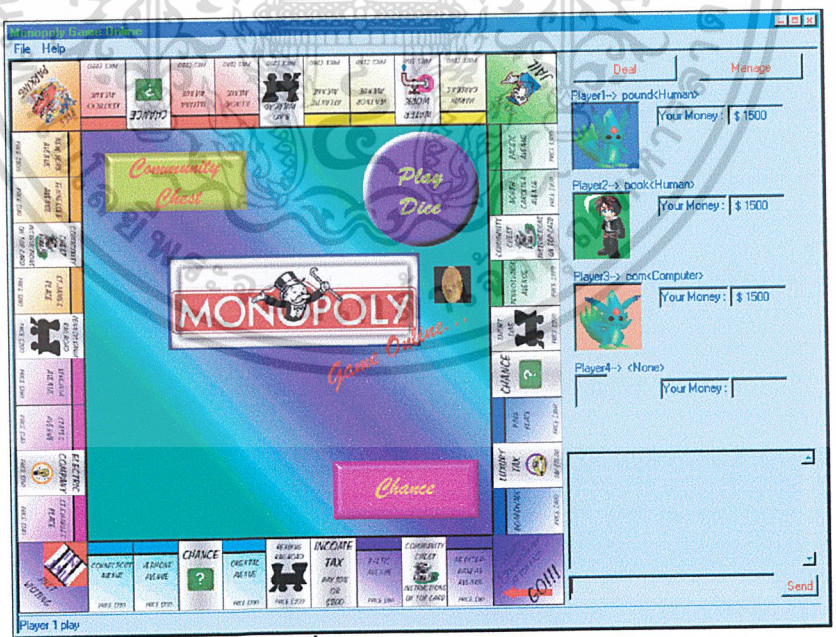
รูปที่ 7.6 ตัวอย่างบัตร Community และบัตร Chance

4. หน้าจอแสดงผล และปุ่มต่าง ๆ ที่ใช้ในการเล่น

4.1 หน้าจอหลัก

เป็นหน้าจอที่แสดงว่าผู้เล่นแต่ละคนเป็นใครและมีทรัพย์สินอยู่เท่าใด บอกสถานะผู้เล่นว่าใครเป็นผู้กำลังเล่นอยู่ หน้าจอนี้จะมีปุ่ม

- ปุ่ม Play Dice อยู่บริเวณกระดานเล่น ปุ่มนี้เพื่อให้ผู้เล่นทำการทอดลูกเต๋า
- ปุ่ม Deal เพื่อให้ผู้เล่นติดต่อเสนอข้อตกลงการซื้อขายกับผู้เล่นอื่น
- ปุ่ม Manage เพื่อให้ผู้เล่นจัดการกับทรัพย์สินส่วนตัว เช่น การจำนอง-ไถ่จำนองที่ดิน การซื้อ-ขายสิ่งปลูกสร้าง
- ปุ่ม Send เพื่อส่งข้อความเพื่อคุยกับผู้เล่นอื่น ๆ



รูปที่ 7.7 หน้าจอหลัก

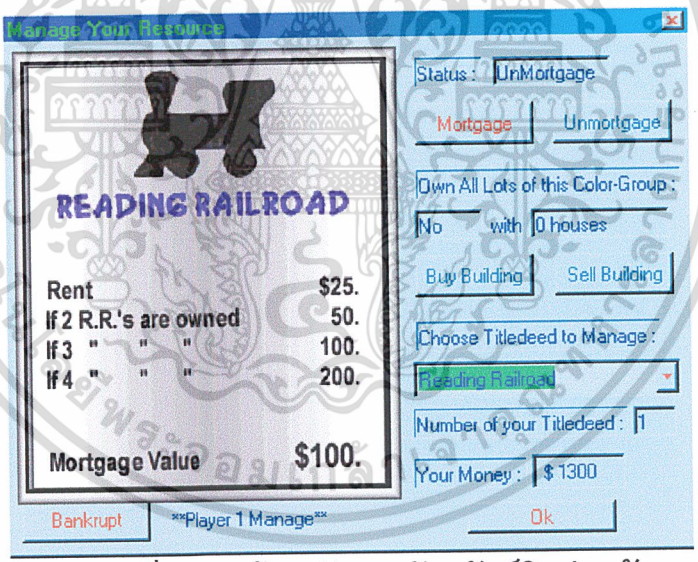
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 หน้าจอจัดการทรัพย์สินส่วนตัว

เมื่อคลิกปุ่ม Manage ในหน้าจอหลักจะปรากฏหน้าจอจัดการข้อมูลส่วนตัวขึ้นเพื่อให้ผู้เล่นสามารถจัดการกับทรัพย์สินส่วนตัวได้ หน้าจอนี้จะประกอบด้วยปุ่มต่าง ๆ คือ

- ปุ่ม Mortgage เลือกเมื่อต้องการจำนองที่ดิน
- ปุ่ม Unmortgage เลือกเมื่อต้องการไถ่จำนองที่ดิน
- ปุ่ม Buy Building เลือกเมื่อต้องการซื้อสิ่งปลูกสร้าง ได้แก่บ้าน หรือโรงแรม บนที่ดินที่เลือก
- ปุ่ม Sell Building เลือกเมื่อต้องการขายสิ่งปลูกสร้างต่าง ๆ บนที่ดิน
- ปุ่ม Bankrupt เลือกเมื่อต้องการล้มละลาย หรือออกจากเกม
- ปุ่ม Ok เลือกเมื่อจัดการกับทรัพย์สินเสร็จแล้วกลับสู่หน้าจอหลัก

โดยปุ่มต่าง ๆ ถ้าไม่อยู่ในเงื่อนไขที่จะกระทำได้ จะไม่สามารถคลิกเลือกได้ เช่น ที่ดินที่จำนองไว้แล้วจะไม่สามารถเลือก Mortgage ได้อีก หรือ ถ้าเจ้าของที่ดินไม่ได้เป็นเจ้าของโฉนดในกลุ่มสีเดียวกันจะไม่สามารถสร้างสิ่งปลูกสร้างได้

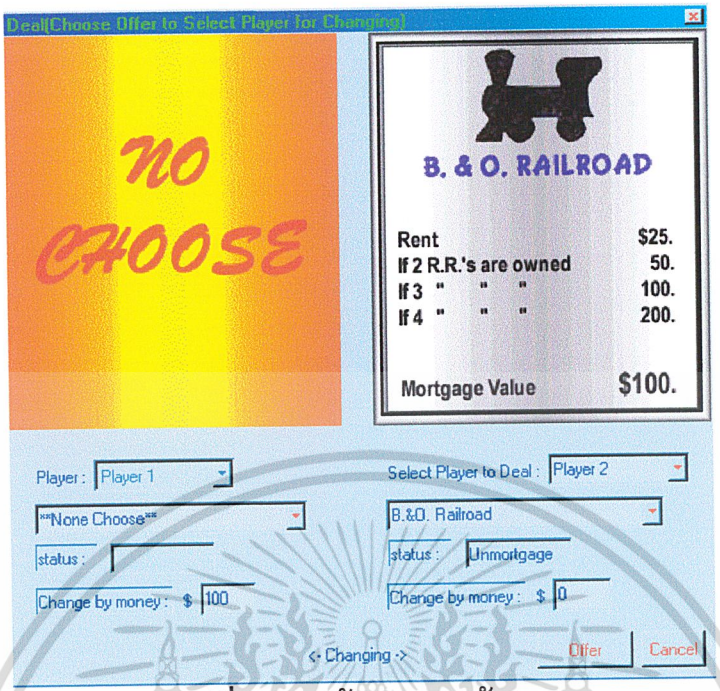


รูปที่ 7.8 หน้าจอจัดการกับทรัพย์สินส่วนตัว

4.3 หน้าจอเสนอข้อตกลง

เมื่อผู้เล่นคลิกปุ่ม Deal ในหน้าจอหลัก จะปรากฏหน้าจอเสนอข้อตกลงเพื่อให้ผู้เล่นเสนอข้อตกลงกับผู้เล่นอื่น เช่น การขอซื้อที่ดิน การขายที่ดิน เมื่อผู้เล่นตัดสินใจที่จะเสนอข้อตกลงใดแล้วให้คลิกปุ่ม offer เพื่อยื่นเสนอข้อตกลง หากไม่ต้องการยื่นข้อเสนอหรือต้องการกลับหน้าจอหลักให้คลิก Cancel

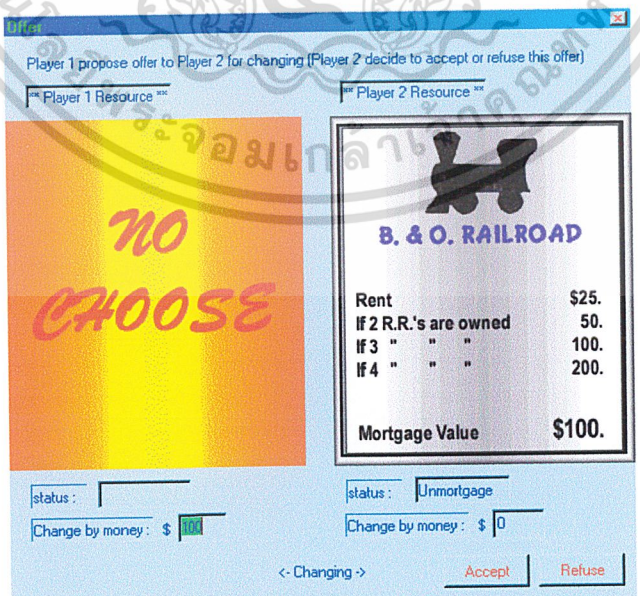
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.9 หน้าจอเสนอข้อตกลง

4.4 **หน้าจอพิจารณาข้อตกลงที่เสนอมา**

เมื่อมีผู้เล่นยื่นข้อเสนอ จะปรากฏหน้าจอพิจารณาข้อเสนอให้ผู้มีสิทธิตอบรับข้อเสนอนั้น เพื่อให้พิจารณาข้อเสนอนั้น ๆ ถ้ายอมรับข้อเสนอคลิกปุ่ม Accept ถ้าปฏิเสธข้อเสนอคลิกปุ่ม Refuse

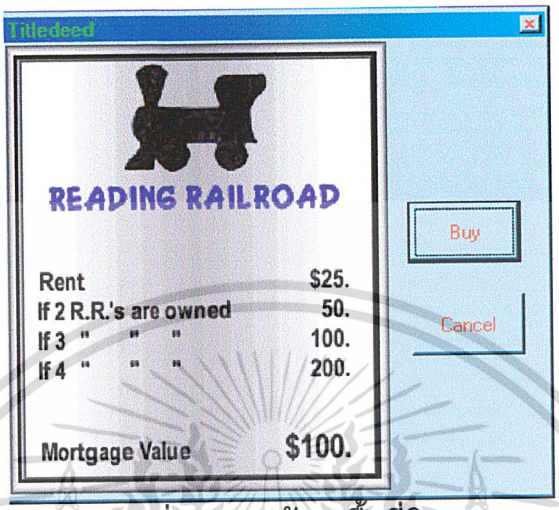


รูปที่ 7.10 หน้าจอพิจารณาข้อตกลงที่เสนอมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 หน้าจอซื้อที่ดิน

เมื่อผู้เล่นเดินตกลงในที่ดินที่ยังไม่มีเจ้าของ จะปรากฏหน้าจอขึ้นเพื่อให้พิจารณาว่าจะซื้อที่ดินนี้หรือไม่ ถ้าต้องการซื้อให้คลิกปุ่ม Buy ถ้าไม่ต้องการซื้อหรือมีเงินไม่พอคลิกปุ่ม Cancel

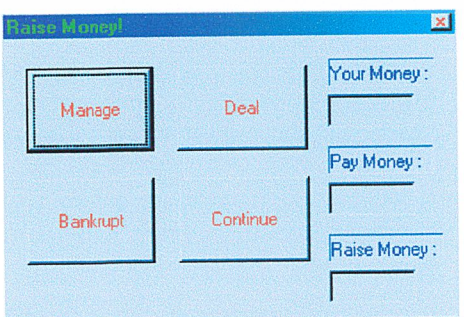


รูปที่ 7.11 หน้าจอซื้อที่ดิน

4.6 หน้าจอเมื่อไม่มีเงินพอชำระหนี้

เมื่อผู้เล่นไม่มีเงินพอจ่ายค่าเช่าที่ หรือไม่พอชำระหนี้จะปรากฏหน้าจอนี้ขึ้นเพื่อให้ผู้เล่นเลือกจัดการโดยปุ่มต่าง ๆ คือ

- ปุ่ม Manage เลือกเพื่อไปจัดการกับทรัพย์สินส่วนตัวเพื่อหาเงิน เช่นจำนองที่ หรือขายสิ่งปลูกสร้าง
- ปุ่ม Deal เพื่อติดต่อเสนอข้อตกลงกับผู้เล่นอื่นเพื่อหาเงินเพิ่ม
- ปุ่ม Bankrupt ล้มละลาย
- ปุ่ม Continue ถ้าผู้เล่นสามารถหาเงินได้พอชำระหนี้ จะสามารถคลิกปุ่ม Continue ได้



รูปที่ 7.12 หน้าจอเมื่อไม่มีเงินพอชำระหนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.7 หน้าจอเลือกหนทางออกจากคุก

ถ้าผู้เล่นติดคุกอยู่เมื่อถึงตาเล่นจะมีหนทางให้เลือกออกจากคุกคือเลือกปุ่มต่าง ๆดังนี้คือ

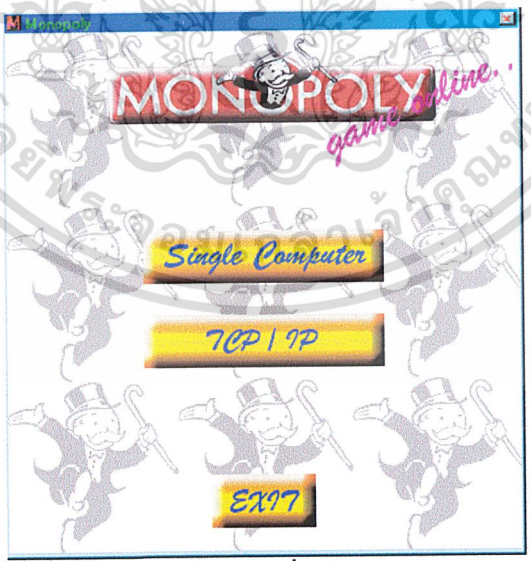
- ปุ่ม Pay $50 เลือกจ่ายเงิน $50 เพื่อออกจากคุก
- ปุ่ม Throw Dice เลือกที่จะทอดลูกเต๋า หากได้หน้าลูกเต๋าทั้ง 2 ลูกเหมือนกันก็สามารถออกจากคุกได้
- ปุ่ม Use Card เลือกใช้บัตร Community Chest หรือบัตร Chance ที่มีข้อความ Get Out Of Jail Free ถ้าผู้เล่นไม่มีบัตรจะไม่สามารถเลือกได้



รูปที่ 7.13 หน้าจอเลือกหนทางการออกจากคุก

5. วิธีการเล่น

เมื่อเข้าโปรแกรมเกมเศรษฐีระบบออนไลน์ จะปรากฏเมนูให้เลือกว่าจะเล่นลักษณะใด คือถ้าเล่นคอมพิวเตอร์เครื่องเดียว เลือกปุ่ม Single Computer หรือเลือกเล่นแบบผู้เล่นหลายคนผ่าน TCP/IP เลือกปุ่ม TCP/IP ต้องการออกจากโปรแกรมเกมคลิกปุ่ม EXIT

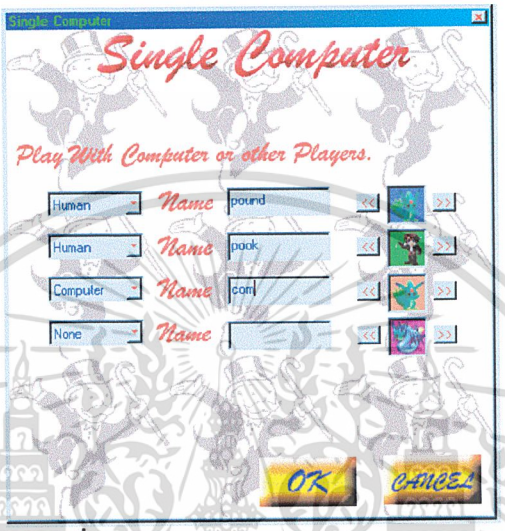


รูปที่ 7.14 หน้าเมนูเริ่มต้นโปรแกรมเกม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1 คอมพิวเตอร์เครื่องเดียว

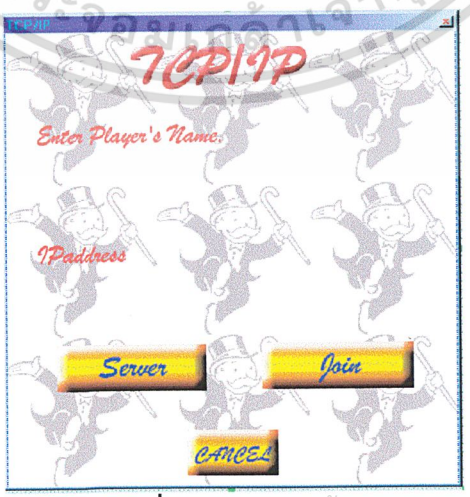
เมื่อเลือกปุ่ม Single Computer แล้ว จะปรากฏเมนูของ Single Computer ขึ้น ผู้เล่นสามารถเลือกเล่นได้ไม่เกิน 4 คน โดยจะเป็นผู้เล่นเล่นทั้งหมดหรือจะเลือกให้คอมพิวเตอร์เล่นด้วยก็ได้ ผู้เล่นสามารถตั้งชื่อและเลือกตัวเบี้ยที่จะเดินเองได้ เมื่อเลือกเสร็จแล้วคลิกที่ปุ่ม OK เพื่อเริ่มเล่นเกม หรือคลิกปุ่ม CANCEL เพื่อกลับสู่เมนูเริ่มต้น



รูปที่ 7.15 **เมนู** Single Computer

5.2 คอมพิวเตอร์หลายเครื่องโดยเล่นผ่านโปรโตคอล TCP/IP

เมื่อเลือกปุ่ม TCP/IP ในหน้าจอเมนูเริ่มต้น จะปรากฏเมนู TCP/IP ผู้เล่นสามารถเล่นผ่านโปรโตคอล TCP/IP ได้ไม่เกิน 4 คน หากต้องการยกเลิกการเล่นแบบ TCP/IP ให้คลิกปุ่ม CANCEL เพื่อกลับสู่หน้า เมนูเริ่มต้น



รูปที่ 7.16 **เมนู** TCP/IP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยแบ่งผู้เล่นเป็น 2 ประเภทคือ

5.2.1 ผู้เล่นหลัก

ผู้เล่นหลักมีเพียง 1 คนเท่านั้น โดยเมื่อเข้ามาเมนู TCP/IP แล้วให้ผู้เล่นกรอกชื่อและเลือกตัวเบี้ยที่จะเดิน แล้วคลิกปุ่ม Server จะปรากฏหน้าเมนู Server ขึ้น ที่หน้าจอเมนู Server ผู้เล่นหลักจะเข้ามารอผู้เล่นรอง หากมีผู้เล่นรองเข้ามาจะปรากฏรายชื่อผู้เล่นเกมขึ้น ผู้เล่นสามารถพิมพ์ข้อความคุยกันได้ เมื่อผู้เล่นทั้งหมดพร้อมที่จะเล่นแล้ว ให้ผู้เล่นหลักเลือกเมนู File Start Game

5.2.2 ผู้เล่นรอง

ผู้เล่นรองจะมีได้ตั้งแต่ 1 ถึง 3 คน โดยผู้เล่นรองเมื่อเข้าเมนู TCP/IP แล้วให้ใส่ชื่อ เลือกตัวเบี้ย แล้วป้อนค่า IP Address ของผู้เล่นหลัก จากนั้นคลิกที่ปุ่ม Join ผู้เล่นจะเข้าสู่หน้าจอหลักและปรากฏรายชื่อผู้ที่เล่นเกมด้วยกัน ผู้เล่นสามารถพิมพ์ข้อความคุยกันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

กฎกติกาการเล่นเกม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.
กฎกติกาการเล่นเกม

เรามีความจำเป็นที่จะต้องมีการศึกษาและอ้างอิงกฎ กติกา และวิธีการต่าง ๆ ของเกม ที่ทำให้เกมสามารถดำเนินไปได้จนกระทั่งจบเกม โดยกฎและกติกาต่าง ๆ ที่ผู้จัดทำศึกษาและนำมาอ้างอิง แล้วทำการดัดแปลงกฎและกติกาบางส่วนเพื่อให้มีความเหมาะสมมีดังต่อไปนี้

ก่อนทำการเล่น

- ตัวเบี้ย หรือตัวเดินของผู้เล่นทุกคนตั้งที่จุด GO
- ผู้เล่นจะได้รับเงินเริ่มต้นคนละ $1,500

นายธนาคารและธนาคาร คอมพิวเตอร์เป็นส่วนทำการจัดการ

วิธีการเล่น เมื่อเริ่มเล่น ผู้เล่นคนแรกทำการทอดลูกเต๋า แล้วเดินตัวเบี้ยตามที่ได้แต้ม หากที่ดินที่ผู้เล่นเดินไปตกยังไม่มีเจ้าของผู้เล่นสามารถซื้อโฉนดที่ดินมาเป็นกรรมสิทธิของตนได้ หากตกช่องที่มีเจ้าของจะต้องจ่ายค่าเช่าที่ตามอัตราที่กำหนดไว้ในโฉนดแก่เจ้าของโฉนด หรือหากผู้เล่นไม่ได้เดินไปตกในช่องที่ดิน ได้แก่ช่อง GO, COMMUNITY CHEST, INCOME TAX, CHANCE, IN JAIL, FREE PARKING, LUXURY TAX, GO TO JAIL ผู้เล่นจะต้องทำตามเงื่อนไขในแต่ละช่องที่กำหนดไว้

การซื้อโฉนดที่ดิน สามารถทำได้ 2 ลักษณะคือ

- เมื่อเดินตกลงในช่องที่โฉนดที่ดินไม่มีเจ้าของ จะมีสิทธิ์เลือกที่จะซื้อหรือไม่ซื้อที่นั้นได้
- สามารถเสนอราคาขอซื้อโฉนดที่ดินแก่เจ้าของโฉนดที่ดินนั้นได้

การจ่ายค่าเช่าที่ เมื่อตกลงในช่องที่มีเจ้าของโฉนดที่ดินจะต้องทำการจ่ายค่าเช่าที่ตามอัตราที่กำหนด

โฉนดที่ดิน ได้กำหนดราคาต่าง ๆ ไว้ ทั้งการเช่าที่ การซื้อบ้าน เป็นต้น โดยมีรายละเอียดเพิ่มเติมคือ

- ถ้าผู้เล่นใดเป็นเจ้าของโฉนดที่ทั้งหมดในกลุ่มสีเดียวกัน จะได้รับค่าเช่าที่เป็น 2 เท่าจากอัตราเดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ช่อง GO เมื่อตัวเบี้ยของผู้เล่นเดินผ่าน GO หรือตกที่ช่อง GO ผู้เล่นจะได้รับเงิน $200 จากธนาคาร

ช่อง "CHANCE" และ "COMMUNITY CHEST" เมื่อตกที่ช่อง CHANCE หรือ COMMUNITY CHEST คอมพิวเตอร์จะทำการสุ่มบัตรให้แล้วผู้เล่นจะต้องทำตามเงื่อนไขที่ได้รับ

ถ้าได้รับ "GET OUT OF JAIL FREE" จะสามารถเก็บไว้ใช้ออกจากคุกได้เมื่อติดคุก

ช่อง INCOME TAX เมื่อผู้เล่นตกช่อง INCOME TAX ผู้เล่นจะต้องทำการจ่ายเงิน $200 แก่ธนาคาร

ช่อง LUXURY TAX เมื่อผู้เล่นตกช่อง LUXURY TAX ผู้เล่นจะต้องทำการจ่ายเงิน $75 แก่ธนาคาร

ช่อง IN JAIL ถ้าผู้เล่นเดินตกลงในช่อง IN JAIL ผู้เล่นไม่ต้องติดคุก ถือเป็นช่องพัก

เงื่อนการติดคุก

- เมื่อตกลงในช่อง " GO TO JAIL" ผู้เล่นจะต้องกลับไปช่อง IN JAIL
- ได้บัตร Chance หรือ Community Chest สั่งว่า "GO TO JAIL" ผู้เล่นจะต้องกลับไปช่อง IN JAIL
- ทอดลูกเต๋าได้หน้าเหมือนกัน (ลูกเต๋าทั้ง 2 ลูกขึ้นแต้มเท่ากัน) 3 ครั้งติดต่อกัน ผู้เล่นจะต้องกลับไปช่อง IN JAIL

เงื่อนไขการออกจากคุก

- จ่ายเงิน $50 แก่ธนาคาร
- ทอดลูกเต๋า ถ้าทอดลูกเต๋าได้หน้าเหมือนกัน (ลูกเต๋าทั้ง 2 ลูกขึ้นแต้มเท่ากัน)
- ใช้บัตร "GET OUT OF JAIL FREE"
- ถ้าทอดลูกเต๋า 3 ครั้ง (3 รอบ) ถ้ายังไม่ได้ออกจากคุก รอบต่อไป(รอบที่ 4) จะต้องจ่ายเงิน $50 แก่ธนาคาร

โดยเมื่อผู้เล่นหลังจากผ่านเงื่อนไขการออกจากคุกแล้ว ผู้เล่นจะสามารถเดินตามแต้มลูกเต๋าที่ทอดได้ในตาถัดไป

ช่อง FREE PARKING ผู้เล่นเดิมมาตกที่ช่องนี้ไม่ต้องเสียค่าเช่าอะไร ถือเป็นช่องพัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การซื้อบ้าน การซื้อบ้านในโฉนดที่ดินใดๆจะกระทำได้ก็ต่อเมื่อเป็นเจ้าของโฉนดที่ดินอื่นๆที่มีสีเดียวกับโฉนดที่ดินนั้นๆทั้งหมดเสียก่อน โดยจะเรียกโฉนดที่ดินที่มีสีเดียวกันนั้นว่า ที่ดินในกลุ่มสีเดียวกัน และถ้าจะซื้อบ้านหลังที่ 2 จะต้องมีบ้านหลังที่ 1 อยู่ในที่ดินทุกโฉนดในกลุ่มสีเดียวกันทั้งหมดก่อน การซื้อบ้านหลังที่ 3 และ 4 ก็เช่นเดียวกันจะต้องมีบ้านหลังที่ 2 และ 3 (ตามลำดับ) ในที่ดินกลุ่มสีเดียวกันทั้งหมดก่อน ราคาบ้านขึ้นอยู่ตามโฉนดของแต่ละที่ดิน

การซื้อโรงแรม จะต้องมีบ้าน 4 หลังบนที่ดินกลุ่มสีเดียวกันทั้งหมดก่อน จึงจะทำการซื้อโรงแรมได้ ราคาโรงแรมขึ้นอยู่ตามโฉนดของแต่ละที่ดิน

การขายบ้านและโรงแรม ถ้ามีโรงแรมจะต้องทำการขายโรงแรมก่อนจึงจะขายบ้านได้ ถ้าขายโรงแรมแล้วจะได้บ้านคืนมา 4 หลัง การขายบ้านและโรงแรมจะต้องทำการขายในลักษณะย้อนกลับกับการซื้อ เช่น ถ้ามีบ้านอยู่ 4 หลังในที่ดินกลุ่มสีเดียวกัน ถ้าเราต้องการจะขายบ้านจะต้องขายบ้านหลังที่ 4 ในกลุ่มสีเดียวกันทั้งหมดก่อนจึงจะขายบ้านหลังที่ 3 ได้ ถ้าจะทำการขายบ้านหลังที่ 2 และ 1 ก็เช่นเดียวกันคือต้องทำการขายบ้านหลังที่ 3 และ 2 (ตามลำดับ) ในที่ดินกลุ่มสีเดียวกันให้หมดก่อน การขายบ้านจะได้เงิน เป็นครึ่งหนึ่งจากราคาที่ซื้อมา

การจำนองโฉนดที่ดิน

- การจำนองโฉนด จะทำได้ก็ต่อเมื่อโฉนดที่ดินที่เป็นเจ้าของอยู่ไม่มีบ้านหรือโรงแรมอยู่ ถ้ามีอยู่จะต้องทำการขายให้หมดก่อน จึงจะทำการจำนองได้ การจำนองจะได้ค่าจำนองตามอัตราที่กำหนดไว้ในโฉนดของแต่ละที่
- การไถ่จำนองจะต้องจ่ายเงินเป็นจำนวนเท่ากับ ค่าจำนอง + 10%ของค่าจำนอง
- ที่ที่ติดจำนองอยู่ เจ้าของที่สามารถเก็บค่าเช่าได้ตามปกติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

Byron S.Gottfried. 2541. **ทฤษฎีและตัวอย่างโจทย์ การเขียนโปรแกรมด้วย ภาษาซี.** แปลโดย ศรัณย์ อินทโกสุม. กรุงเทพฯ : แมคกรอ-ฮิล

John Hubbard. 2540. **ทฤษฎีและตัวอย่างโจทย์ การเขียนโปรแกรมด้วยภาษา C++.** แปลโดย ศรัณย์ อินทโกสุม. กรุงเทพ ฯ : แมคกรอ-ฮิล

Jonathan Bates, Timothy Tompkins. 1998. **Using Visual C++ 6.** พิมพ์ครั้งที่ 1. United States of America : Que Corporation

Kate Gregory, Paul Robichaux, Brady Merkel, Markus Pope. 1997. **Building Internet Applications with Visual C++.** United States of America : Que Corporation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้