ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล.

ระบบบริหารการให้บริการคอมพิวเตอร์

Computer Service Management System

โดย

นายอดิศักดิ์ สุกุล

รหัส 44067095

H001996

อาจารย์ที่ปรึกษา

ดร.จันทร์บูรณ์ สถิตวิริยวงศ์

วัน เดือน ปี 24 มี.ค. 2550
เลขทะเบียน 01996
เลขเรียกหนังสือ วพ ว1295 2545
"ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล."

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน

หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

ภาคเรียนที่ 2 ปีการศึกษา 2545

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ	ระบบบริหารการให้บริการคอมพิวเตอร์
นักศึกษา	นายอดิศักดิ์ สุกุล
อาจารย์ที่ปรึกษา	ดร.จันทร์บูรณ์ สถิตวิริยวงศ์
ระดับการศึกษา	วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2545

บทคัดย่อ

ในปัจจุบันคอมพิวเตอร์ได้เข้ามามีบทบาทเป็นอย่างมาก จึงเกิดการให้บริการเครื่องคอมพิวเตอร์เพื่อใช้งานโปรแกรมต่างๆ รวมถึงอินเตอร์เน็ต เกิดขึ้นในสถานศึกษา หน่วยงานราชการและเอกชน เพื่อตอบสนองความต้องการของผู้ใช้ จึงควรมีการพัฒนาระบบงานขึ้นนี้เพื่อบริหารจัดการเครื่องคอมพิวเตอร์สำหรับให้บริการ เพื่อให้สามารถจัดสรรเครื่องคอมพิวเตอร์และเวลาในการให้บริการกับผู้ใช้แต่ละประเภท เก็บสถิติข้อมูลการเข้ารับบริการของผู้ใช้ ทำรายงานสถานะและวิเคราะห์ข้อมูลเชิงสถิติ เพื่อเป็นข้อมูลในการตัดสินใจให้รองรับการขยายตัวขององค์กรได้อย่างมีประสิทธิภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title	Computer Service Management System
Student	Mr. Adisak Sukul
Advisor	Dr. Chanboon Sathitwiriyawong
Level of Study	Master of Science in Information Technology
Major	Information Science
Academic Year	2002

ABSTRACT

Presently, computers have played an important role which is why businesses, government, and educational institutions have made them widely available to service the use of many software applications and the internet. Development of a system to manage these service computers should be done in order to allocate the appropriate number of clients and time usage for various types of user groups, record statistics of computer service usage, report status and analyze statistical data to support the decision of scaling services effectively within the organization.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I

บทคัดย่อภาษาอังกฤษ.....II

สารบัญ.....III-IV

บทที่

1. บทนำ.....1
2. ความรู้พื้นฐานในการพัฒนาระบบบริหารการให้บริการคอมพิวเตอร์.....3
   - 2.1 Windows Registry.....3
   - 2.2 การติดตั้งสื่อสารด้วย TCP/IP บน Visual Basic.....7
   - 2.3 พื้นฐานเกี่ยวกับฟังก์ชันวินโดวส์ API-32 บิต.....10
   - 2.4 การจัดการโพรเซสของวินโดวส์.....17
   - 2.5 ทฤษฎีพื้นฐานโครงสร้างของโพรโตคอล TCP/IP และ UDP.....18
3. การออกแบบโครงสร้างของระบบ.....25
   - 3.1 การออกแบบการทำงานของ Server Program และระบบโดยรวม.....25
   - 3.2 การออกแบบการทำงานของ Client.....33
   - 3.3 การออกแบบฐานข้อมูล.....34
   - 3.4 การออกแบบการติดต่อสื่อสารระหว่าง Server Program และ Client Program.....37
4. การพัฒนาระบบบริหารการให้บริการคอมพิวเตอร์.....43
   - 4.1 การพัฒนาส่วนการทำงานของ Server Program.....43
   - 4.2 การพัฒนาส่วนการทำงานของ Client Program.....51
   - 4.3 การพัฒนาส่วนการสร้างรายงาน.....52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

5. สรุปผลการทดลอง.....54
5.1 ผลการทดสอบระบบ.....54
5.2 การทดสอบการใช้ทรัพยากรเครือข่ายของระบบ.....55
5.3 แนวทางในการพัฒนาระบบเพิ่มเติมให้มีความสามารถมากขึ้น.....57
5.4 แนวทางในการพัฒนาระบบเพิ่มเติมให้มีความสามารถมากขึ้น.....57
บรรณานุกรม.....58
ประวัติผู้เขียน..... 59

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

สถานศึกษาและหน่วยงานต่างๆ ทั้งภาครัฐและเอกชนในปัจจุบัน การมีเครื่องคอมพิวเตอร์เพื่อให้บริการใช้งาน Internet โปรแกรมสำเร็จรูปต่างๆ รวมถึงห้องปฏิบัติการคอมพิวเตอร์ มีความจำเป็นอย่างมาก ซึ่งเมื่อมีเครื่องคอมพิวเตอร์เพื่อให้บริการจำนวนมาก จะก่อให้เกิดภาระงานกับผู้ดูแลระบบเป็นอย่างมาก เกี่ยวกับการควบคุมสิทธิการใช้เครื่อง เวลาในการใช้เครื่อง การควบคุมการใช้งานโปรแกรมที่ไม่ได้รับอนุญาต การเก็บสถิติ รวมไปถึงการนำข้อมูลการให้บริการมาสรุปเป็นรายงาน เพื่อวัดประสิทธิภาพของการให้บริการเครื่องคอมพิวเตอร์นั้น ซึ่งงานเหล่านี้เป็นงานที่จำเป็นและจะทำให้ผู้ดูแลระบบต้องใช้เวลาสูงทั้งยังเกิดความล่าช้า และความผิดพลาดได้ง่าย

ดังนั้นจึงมีแนวคิดในการพัฒนาระบบบริหารการให้บริการเครื่องคอมพิวเตอร์ เพื่อเป็นเครื่องมือที่เหมาะสมในการบริหารการให้บริการเครื่องคอมพิวเตอร์ โดยสามารถลดภาระงานของผู้ดูแลระบบได้อย่างมาก และยังทำงานได้อย่างรวดเร็วและถูกต้องกว่าใช้คนทำ และระบบยังสามารถสรุปผลรายงานที่ละเอียด ถูกต้องและสามารถตรวจสอบได้ให้กับผู้บริหาร จึงส่งผลให้ประสิทธิภาพการให้บริการเครื่องคอมพิวเตอร์ของหน่วยงานนั้นๆสูงขึ้นด้วย

1.1 วัตถุประสงค์ของการพัฒนาโครงการ

เพื่อพัฒนาเครื่องมือในการให้บริการใช้เครื่องคอมพิวเตอร์ โดยเป็นซอฟท์แวร์ที่ทำหน้าที่ควบคุมสิทธิการใช้เครื่อง เวลาในการใช้เครื่อง การควบคุมการใช้งานโปรแกรมที่ไม่ได้รับอนุญาต การเก็บสถิติ รวมไปถึงการนำข้อมูลการให้บริการมาสรุปเป็นรายงาน เพื่อให้การให้บริการมีความรวดเร็ว ลดข้อผิดพลาดและสามารถตรวจสอบได้

1.2 แนวทางการพัฒนา

โปรแกรมทำงานในลักษณะ Client/Server โดยเครื่องที่ทำงานเป็น Server คือเครื่องของผู้ดูแลระบบที่มีความสามารถควบคุม และตรวจสอบการทำงานต่างๆบนเครื่อง Client ที่มีโปรแกรม Agent ทำงานอยู่ โดยAgent จะทำหน้าที่ควบคุมการใช้งานของผู้ใช้เครื่อง เช่น การ lock หน้าจอ การจับเวลา การปิดกั้นอินพุตจากคีย์บอร์ดและเมาส์ เพื่อป้องกันการลักลอบใช้เครื่องโดยไม่ได้รับอนุญาต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของระบบส่วนใหญ่จะอยู่ที่โปรแกรมฝั่ง Server ที่สามารถควบคุมการใช้งานบนเครื่อง Client ที่มี Agent ทำงานอยู่ และทำหน้าที่เก็บข้อมูลสถานะต่างๆของการให้บริการ รวมถึงความสามารถในการรักษาความปลอดภัยในการป้องกันการใช้งานเครื่อง Client โดยไม่ได้รับอนุญาต

1.3 เครื่องมือที่ใช้ในการพัฒนา

เครื่องมือที่ใช้ในการพัฒนาระบบบริหารการให้บริการคอมพิวเตอร์ประกอบไปด้วย

- Microsoft Visual Basic 6.0 ใช้ในการพัฒนาโปรแกรม Server และ โปรแกรม Client โดยใช้งานผ่าน API ของ Microsoft Windows
- เครื่อง PC Computer เชื่อมต่อกันผ่านเครือข่าย Ethernet 10/100 โดยรองรับ TCP และ UDP
- ใช้ระบบปฏิบัติการ Microsoft Windows 2000 Server หรือ Professional สำหรับโปรแกรม Server และ Windows 98/Me สำหรับโปรแกรม Client
- ใช้ Microsoft Access ในการพัฒนาฐานข้อมูล

1.4 ขอบเขตการพัฒนา

ระบบใช้งานบนเครื่อง PC Computer เชื่อมต่อกันผ่านเครือข่าย Ethernet ความเร็ว 100Mbps โดยใช้ระบบปฏิบัติการ Microsoft Windows 2000 Server หรือ Professional สำหรับโปรแกรม Server และ Windows 98/Me/2000Pro/XP สำหรับโปรแกรม Client

การติดต่อสื่อสารระหว่างเครื่อง PC Computer อยู่บนเครือข่าย Local Area Network เดียวกัน โดยไม่มีการปิดกั้น port การสื่อสาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ความรู้พื้นฐานในการพัฒนาระบบบริหารการให้บริการคอมพิวเตอร์

2.1 Windows Registry

Registry คือฐานข้อมูลกลางของระบบปฏิบัติการ Windows ที่ใช้เก็บข้อมูลการกำหนดค่าต่างๆทั้งในส่วนของฮาร์ดแวร์ ซอฟต์แวร์, การปรับหน้าจอ รวมถึง Configuration ทั้งหลายของ Windows ซึ่งจำเป็นต่อการบริหารการทำงานของคอมพิวเตอร์ทั้งเครื่อง

ข้อมูลใน Registry อาจแบ่งได้เป็น 4 ส่วนใหญ่ๆ คือข้อมูลเกี่ยวกับฮาร์ดแวร์ที่มีอยู่ในเครื่องคอมพิวเตอร์ ข้อมูลการทำงานของซอฟต์แวร์แต่ละตัวซึ่งรวมถึงซอฟต์แวร์แบบ 16 บิตด้วย ข้อมูลการกำหนดค่าและการทำงานต่างๆ ของ Windows และสุดท้ายข้อมูลการกำหนดของผู้ใช้แต่ละคน ทั้ง 4 ส่วนนี้ถูกจัดเก็บไว้ในไฟล์ 2 ไฟล์คือ System.dat และ User.dat ภายใน Directory หลักของ Windows โดยไฟล์ System.dat ทำหน้าที่เก็บข้อมูลฮาร์ดแวร์ , ซอฟต์แวร์ ส่วนไฟล์ User.dat เก็บข้อมูลการกำหนดค่าสำหรับผู้ใช้แต่ละคน เช่น หน้าตาของ Desktop ลักษณะรูป Cursor ที่ใช้เป็นต้น ไฟล์ .dat เหล่านี้เก็บข้อมูลผสมกันทั้งที่เป็น Binary และ String ไม่ใช่เท็กซ์ไฟล์เหมือนอย่างไฟล์ .INI อีกต่อไป อีกส่วนที่สำคัญมากและถือเป็นจุดเด่นหนึ่งของ Registry ก็คือข้อมูลบางอย่างใน Registry จะถูกปรับแก้ตลอดเวลาในขณะที่เปิดเครื่องอยู่หรือเปลี่ยนแปลงได้ทุกครั้งที่เปิดเครื่องใหม่ เช่นข้อมูลของการ์ดต่างๆและอุปกรณ์ต่อพ่วง เป็นต้น และเพื่อให้สามารถแก้ไขได้อย่างรวดเร็ว ในระหว่างการทำงานนั้นข้อมูลดังกล่าวจึงถูกโหลดเข้ามาอยู่ในหน่วยความจำหลัก โดย Windows จะดำเนินการตรวจสอบอุปกรณ์ทุกชิ้นที่มีอยู่ในระบบและปรับปรุงค่าต่างๆ ทุกครั้งที่เปิดเครื่องใหม่เพื่อให้ใช้งานอุปกรณ์เหล่านี้ได้ตามคุณสมบัติของระบบปปฏิบัติการแบบ Plug and Play (PnP) แต่ด้วยความจำเป็น เพื่อให้โปรแกรมเก่าๆยังคงทำงานได้(Backward Compatibility) ทำให้ยังคงมีไฟล์ WIN.INI และ SYSTEM.INI อยู่เพื่อให้ซอฟต์แวร์รุ่นเก่าสามารถกำหนดค่าบางอย่างให้กับ Windows ได้เช่นเดิม ซึ่งหากเปิดดูก็จะพบว่ามี เนื้อหาน้อยลงมาก รวมทั้งค่าไฟล์ AUTOEXEC.BAT และ CONFIG.SYS ก็ยังมีความจำเป็นต่อการกำหนดสภาพแวดล้อมบางอย่างของ DOS ใน Windows ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้างโดยทั่วไปภายใน Registry ประกอบด้วยส่วนหลักๆที่เรียกว่า คีย์ (key) ซึ่งมีความสัมพันธ์ต่อกันในลักษณะของต้นไม้แบบลำดับชั้น (Hierarchical Tree) เริ่มจากคีย์ที่เป็น Root (root key) แล้วแตกลงไปในคีย์ย่อย (subkey หรือ Hive) ต่างๆ ส่วนที่ใช้เก็บข้อมูลจริงนั้นเป็น Ertry ภายในคีย์เรียกว่า Value Entry หรือสั้นๆว่า Entry โดยแต่ละคีย์ใน Registry สามารถจัดเก็บ Entry ไว้ได้ไม่จำกัดจำนวน ไม่ว่าจะเป็นคีย์ในระดับใดก็ตาม
ซึ่งข้อมูลที่เก็บนั้นแบ่งแยกออกได้ 3 ประเภทคือ

1. ข้อมูลแบบ DWORD (ค่าตัวเลขแบบ Double word ขนาด 32 บิต)
2. ข้อมูลแบบ String (ข้อความ)
3. ข้อมูลแบบ Binary

การที่แบ่งแยกเป็น System.dat กับ User.dat นี้ เพื่อให้เกิดความชัดเจนในการจัดการมากยิ่งขึ้น โดย System.dat จะเก็บข้อมูลเฉพาะส่วนที่เกี่ยวข้องกับคอมพิวเตอร์เครื่องนั้นๆ ทั้งซอฟต์แวร์และฮาร์ดแวร์ ส่วนไฟล์ User.dat ก็จะเก็บข้อมูลส่วนที่กำหนดแตกต่างกันสำหรับผู้ใช้แต่ละคน เช่น Icon บน Desktop ข้อดีของการเก็บข้อมูลแยกกันสามารถสรุปได้ดังนี้

1. ผู้ใช้แต่ละคนสามารถกำหนดลักษณะหน้าจอของตัวเองบนจอคอมพิวเตอร์เครื่องเดียวกัน เลือก Icon บน Desktop เลือก Wallpaper ตามที่ตังเองต้องการได้

2. ในระบบเครือข่ายสามารถกำหนดให้ผู้ใช้เห็นหน้าตา Desktop ของตัวเองได้เสมอไม่ว่าจะเข้าสู่ระบบจากคอมพิวเตอร์เครื่องใดก็ตาม โดยกำหนดให้ใช้ไฟล์ User.dat ที่เครื่อง Server และการปรับแก้ก็สามารถสั่งจากเครื่องที่กำลังใช้งานอยู่ได้ด้วย ในขณะที่ไฟล์ System.dat ต้องมีอยู่ในเครื่องคอมพิวเตอร์แต่ละเครื่องเสมอ

2.1.1 คีย์และคำใน Registry ที่สำคัญ

HKEY_CURRENT_USER และ HKEY_USERS

HKEY_CURRENT เป็นศูนย์รวมของ Configuration ต่างๆที่เกี่ยวข้องกับผู้ใช้ ในระบบที่การกำหนดผู้ใช้ไว้หลายคนก็จะมีคีย์ย่อยของผู้ใช้แต่ละคน ส่วนในคีย์ HKEY-_CURRENT_USER จะเป็นคอนฟิกูเรชั่นของผู้ใช้คนปัจจุบัน ซึ่งการให้แก้ไขค่าในคีย์ HKEY-_CURRENT_USER จะมีผลต่อค่าในคีย์ย่อยของผู้ใช้คนนั้น ๆ ภายในคีย์ HKEY_USER ด้วย เนื่องจาก KEY_CURRENT_USER เป็นเพียงจุดที่ Registry ยกขึ้นมาเพื่อความสะดวกในการเข้าถึงค่า Configuration ต่างๆของผู้ใช้คนปัจจุบันเท่านั้น แท้ที่จริงก็ชี้ไปที่ข้อมูลชุดเดียวกันนั้นเอง คีย์ย่อยที่อยู่ภายใต้คีย์ทั้งสองมีดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Control Pane\ Desktop\CursorBlinkRate กำหนดอัตราการกระพริบของ caret สำหรับป้อนตัวอักษร หน่วยเป็น millisec (1/1000วินาที) ค่าของ Entry นี้สามารถกำหนดได้จากแอพเล็ต Keyboard ใน Control Panel

Control Pane\ Desktop\MenuShowDelay ค่าของ Entry นี้กำหนดช่วงเวลาที่ถ่วงไว้ก่อนแสดงเมนูย่อยออกมา(เช่นเมื่อเลื่อนเมาส์ไปที่เมนู Programs ของเมนูปุ่ม Start แล้วจะปรากฏเมนูย่อยของเมนู Programs ออกมา) มีหน่วยเป็นวินาที

Control Pane\ Desktop\DoubleClickHight และ DunleClickWidth ผู้ใช้มือใหม่ที่ไม่คุ้นเคยกับเมาส์อาจเคยเจอเหตุการณ์แบบนี้ก็ได้ คือขณะที่ ต้องการ Double Click แต่ด้วยความไม่ถนัดมือการกดปุ่มแต่ละครั้งกลับทำให้เมาส์ขยับไปด้วยในระหว่างที่ครั้งแรกกลับครั้งที่สอง ผลที่ได้จึงเป็นการลากเมาส์ในระยะทางสั้นๆแทนการดับ Double Click ที่ต้องการ Entry ทั้งสองนี้จะช่วยให้ Windows ตัดสินใจแทน ว่าเหตุการณ์แบบนี้เป็นการ Double Click หรือเปล่า โดย จะต้องกำหนดระยะเผื่อการขยับของเมาส์ในระหว่างการขยับของเมาส์ในระหว่าง Double Click ไว้เป็นตัวเลขแบบค่าสตริงในหน่วยพิกเซลให้มากขึ้น หารการ Double Click คลิกของ ยังไม่นิ่งพอ

Control Pane\ Desktop\Desktop\ คีย์นี้ก็เก็บ Entry ที่นอกเหนือจากคีย์ย่อย Colors ในคีย์ Appearance โดยเป็นค่าเกี่ยวกับขนาดของวินโดว์, ขนาดของตัวอักษร,ระยะห่างระหว่างไอคอน และอีกหลายๆอย่างที่ไม่ใช่ค่าเกี่ยวกับสี อีก Entry หนึ่งที่ Windows ไม่ได้สร้างไว้ให้หลังการติดตั้งเสร็จใหม่ๆก็คือ MinAnmiate ซึ่งใช้กำหนดวิธีการแสดงการ Minimze และ Maximize ของวินโดวส์ ว่าจะให้ Animation แบบค่อยๆย่อ ค่อยๆขยายหรือไม่ ถ้าค่านี้ไม่มีหรือมีค่าเป็น 1 จะแสดง Animation ในการย่อ-ขยาย แต่ถ้าเป็น 0 จะไม่แสดง

Control Pane\ Desktop\Desktop\Keyboard คีย์ นี้เก็บ Entry ที่เกี่ยวข้องกับคีย์บอร์ดไว้ 2 Entry คือ KeyboardDelay กำหนดช่วงเวลาหลังการกดปุ่มบนคีย์บอร์ดค้างไว้ก่อนถือเป็นการพิมพ์ตัวอักษรซ้ำและ Keyboardspeed กำหนดอัตราเร็วในการพิมพ์ซ้ำตัวอักษรเมื่อกดปุ่มค้างไว้ ค่าสูงสุดที่กำหนดได้8nv 31

Control Pane\ Desktop\Keyboard layout\toggle คีย์นี้เก็บ Entry ที่เกี่ยวข้องกับการสลับภาษาของคีย์บอร์ด ปกติสามารถแก้ไขได้จาก Tab Language ใน Sheet Keyboard Properties โดยค่า Defailt ของคีย์จะหมายถึงปุ่มที่ใช้สลับภาษา ค่าที่กำหนดได้คือ 1,2และ 3 ซึ่งสอดคล้องกับตัวเลือก Left Alt+Shift,Ctrl+Shift และ None ใน Tab Language ตามลำดับ

Control Pane\ Desktop\RemoteAccess|Profile คีย์นี้จะมีคีย์ย่อยที่เก็บ คอนฟิกูเรชั่นของการติดต่อทางโมเด็มแต่ละรายการที่ สร้างขึ้นมาในโฟลเดอร์ Dial-Up networking

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

HKEY_LOCAL_MACHINE

เป็นคีย์ที่เก็บ Configuration ของเครื่องทั้งในส่วนฮาร์ดแวร์และซอฟต์แวร์ ไม่ขึ้นกับผู้ใช้แต่ละคน มีคีย์ย่อยที่น่าสนใจดังต่อไปนี้

Software

คีย์นี้มีโครงสร้างเดียวกับคีย์ Software ใน HKEY_USER ตามที่ได้กล่าวไปแล้วว่าซอฟต์แวร์ต่างๆ จะใช้คีย์ในลักษณะของข้อมูลที่เป็นส่วนรวม ไม่ขึ้นอยู่กับผู้ใช้ผู้ใดผู้หนึ่ง ยกเว้นคีย์ย่อย Classes ที่บรรจุข้อมูลการกำหนดประเภทของ Object ต่างๆ ภายในเครื่องไว้เช่น ไฟล์, โฟลเดอร์, ประเภทของไฟล์ และ Object แบบ OLE เป็นต้น ซึ่งเป็นส่วนหนึ่งก็คือ File Association ที่เราสามารถกำหนดได้จาก Tab file Types ใน Folder Options นั่นเอง และคีย์นี้จะถูกดึงออกมาเป็นคีย์หลักที่ชื่อว่า HKEY_CLASSES_ROOT เพื่ออำนวยความสะดวกในการเข้าถึงคีย์ต่างๆ ที่อยู่ภายใน

คีย์ย่อยที่น่าสนใจภายใน Software นี้ ก็จะเป็นคีย์ของ Windows เป็นคีย์อื่นๆ จะขึ้นอยู่กับว่ามีซอฟต์แวร์ใดถูกติดตั้งไว้ในเครื่องนั้นบ้าง

Microsoft\Windows\CurrentVersion\App Paths เป็นตำแหน่งที่ Window ใช้ในการค้นหาโปรแกรมเมื่อมีการป้อนคำสั่งลงในช่อง Open ของ Dialogue Box Run โดยไม่ได้ระบุ Path (ลักษณะคล้ายๆ ตัวแปร Path ของ Dos) แต่ละคีย์ย่อย เก็บ Path เต็มของไฟล์โปรแกรมไว้ที่ค่า Default และเก็บ Path ที่เป็น Directory ทำงานของโปรแกรมไว้ที่ Entry Path

Microsoft\Windows\CurrentVersion\controls Folder ภายในคีย์ย่อยที่เก็บรายละเอียดต่างๆ เกี่ยวกับส่วนเพิ่มเติม Shell (Shell Extension) ของ Applet ใน Control Panel เช่นการเพิ่ม Tab สำหรับกำหนดค่า แต่ละ Applet ใน Control Panel ก็จะต้องมาลงทะเบียนไว้ที่นี่ด้วยเช่นกัน

Microsoft\Windows\CurrentVersion\explorer\xxx\NameSpace xxx ที่เขียนไว้นั้นสื่อถึงพวก Visual โฟลเดอร์ต่างๆ เช่น Desktop , My Computer เป็นต้นภายในคีย์ของ Visual โฟลเดอร์เหล่านี้จะมีคีย์ย่อยชื่อ NameSpace

คีย์ NameSpace คือตำแหน่งที่ จะได้พบกับคีย์ย่อยที่มีชื่อเป็นหมายเลข CLSID (ปิดหัวท้ายด้วยเครื่องหมายปีกกา) ของ Object ใน Visual โฟลเดอร์นั้นอีกทีหนึ่ง (เมื่อพิจารณาจากโครงสร้าง NameSpace) จากรูปข้อมูลใน Registry บอกให้เรารู้ว่าโฟลเดอร์ Dial-Up Networking เป็น Object ที่อยู่ภายใต้โฟลเดอร์ My Computer นั้นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Microsoft\Windows\CurrentVersion\Run... คีย์ Run ใช้เก็บชื่อ Path เต็มของไฟล์โปรแกรมซึ่งจะถูกรันทุกครั้งที่ Log On เข้าสู่ Windows คีย์อื่นๆในระดับเดียวกันได้แก่ RunOnce,Runservice และ RunserviceOnce คีย์ RunOnce ถูกใช้ในการติดตั้งซอฟต์แวร์ที่ต้องมีการ Restart Windows แล้วจึงติดตั้งต่อจนเสร็จ วิธีรันโปรแกรมติดตั้งต่อหลังจากบู๊ตเครื่องใหม่ทำได้โดยการบันทึก Path เต็มของโปรแกรมที่จะรันไว้ในคีย์นี้ เมื่อรันเรียบร้อยแล้ว

Microsoft\Windows\CurrentVersion\Uninstall ภายในคีย์นี้ จะพบคีย์ย่อยที่เก็บข้อมูลสำหรับใช้ยกเลิกการติดตั้งซอฟต์แวร์ต่างๆ และจะถูกแสดงเป็นรายการใน Tab Install/Uninstall ของ Applet Add/Remove Programs

HKEY_DYN_DATA

เป็นคีย์ที่ข้อมูลถูกเก็บไว้ในหน่วยความจำอยู่ตลอดเวลา ข้อมูลส่วนใหญ่จะถูกสร้างขึ้นในขณะบู๊ตเครื่องและสามารถถูกแก้ไขได้ตลอดเวลาที่ใช้งานคอมพิวเตอร์ ทีคีย์ย่อยที่หน้าสนใจดังนี้

Config Manager เป็นข้อมูลเกี่ยวกับคอนฟิกูเรชั่นของฮาร์ดแวร์ที่ติดตั้งและกำลังใช้งานอยู่ภายในคีย์นี้ถูกสร้างขึ้นเมื่อบู๊ตเครื่องและสามารถเปลี่ยนแปลงแก้ไขได้ตลอดเวลา คีย์นี้เองที่ทำให้สามารถติดตั้งอุปกรณ์ฮาร์ดแวร์ต่างๆ ได้อย่างสะดวกโดยใช้ สมบัติ Plug and Play ของอุปกรณ์นั้น

PerStats เก็บข้อมูลประสิทธิภาพการทำงานในแต่ละส่วนของคอมพิวเตอร์ ซึ่งจะถูกนำไปแสดงโดยโปรแกรม System Monitor [1]

2.2 การติดตั้งสื่อสารด้วย TCP/IP บน Visual Basic

การทำงานเกี่ยวกับเครือข่ายคอมพิวเตอร์นั้นเป็นเรื่องข้อนข้างซับซ้อนจึงทำให้มีการสร้างข้อกำหนดต่างๆขึ้นมาเพื่อให้วิธีการสื่อสารระหว่างกันเป็นไปได้โดยราบรื่นแล้งก็ทำให้เกิดเป็นโพรโตคอล(Protocol) มากมายหลายชนิดมาใช้งานร่วมกับเครือข่าย ซึ่งมีการแบ่ง Protocol ต่างๆ ออกเป็นระดับชั้น เพื่อให้ง่ายต่อการใช้งาน และง่ายต่อการทำความเข้าใจ
นานวันเข้าโพรโตคอลก็มีเพิ่มมากขึ้นจนกลายเป็นความยุ่งยากต่าง บริษัทต่างก็มีผลิตภัณฑ์ที่สนับสนุนโพรโตคอลต่างๆไม่เหมือนกัน ทำให้การใช้งานร่วมกันทำได้ยากขึ้น ทำให้ต้องหาวิธีใช้งานโพรโตคอลต่างๆ ร่วมกันโดยใช้มาตรฐานอันเดียวกันที่ทุกคนยอมรับ ซึ่ง Winsock ก็เป็นหนึ่งในมาตรฐานเหล่านั้น

ระบบปฏิบัติการยูนิกซ์จะมีความสามารถอย่างหนึ่งที่เรียกว่า Socket ซึ่งก็คือ การที่โปรแกรมต่างๆ สามารถสื่อสารข้อมูลระหว่างกันได้ โดยไม่จำเป็นว่าโปรแกรมนั้นจะต้องรันอยู่บนคอมพิวเตอร์เครื่องเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Socket เองก็มาจากแนวความคิด Client/Server โดยส่วนที่ทำหน้าที่เป็นเซิร์ฟเวอร์ จะสร้าง Socket (ช่องเสียบ หรือรูเสียบ) จะมีจำนวนเท่าใดก็แล้วแต่ จากนั้นเมื่อมี Client ที่ต้องการสื่อสารด้วยก็จะติดต่อมาที่ Socket ที่ Server ได้เตรียมไว้ให้ เมื่อเชื่อมต่อกันเสร็จแล้วก็พร้อมจะแลกเปลี่ยนข้อมูลระหว่างกัน โดยเป็นการสื่อสารแบบสองทางชนิดเต็มรูปแบบ (2 Way Full-Duplex)

2.2.1 การใช้งาน Winsock กับระบบปฏิบัติการตระกูล Windows

ระบบปฏิบัติการตระกูล Windows อันได้แก่ Windows 95/98 และ Windows NT ก็ได้เตรียมการทำงานในลักษณะ Socket โดยตั้งชื่อว่า Winsock โดยมีหลักการคล้ายคลึงกับ Socket ในยูนิกซ์ แม้จะไม่ Compatible กัน 100% ก็ตาม

Winsock เป็น API(Application Programming Interface) หรือชุดฟังก์ชันมาตราฐาน ซึ่งออกแบบมาเพื่อทำให้การเขียนโปรแกรมกับอินเตอร์เน็ตเป็นเรื่องง่าย อีกทั้ง Winsock ยังถูกกำหนดให้เป็น API ที่มีลักษณะเปิดกว้าง และสามารถ Port ไปยัง Socket ของยูนิกซ์ได้อย่างไม่ยากนัก

2.2.2 โหมดการทำงานของ Winsock Control

การทำงานของ Winsock Control กับโพรโตคอล TCP/IP นี้แบ่งการทำงานออกเป็น 2 โหมดคือ TCP กับ UDP

TCP (Transmission Control Protocol) เป็นการทำงานในลักษณะ Connection-Based เปรียบเหมือนกับการทำงานของโทรศัพท์ที่จะมีการตรวจสอบการทำงานจากทั้งสองฝั่งมีการโต้ตอบระหว่างกัน ทำให้ได้การสื่อสารที่มีประสิทธิภาพ TCP จะเหมาะกับการส่งข้อมูลที่ต้องการการทำงานที่ถูกต้องแม่นยำของข้อมูล เช่นการส่งภาพหรือเสียงผ่านเครือข่าย ซึ่งจะมีการตรวจสอบความถูกต้องของข้อมูลที่มีการส่งถึงกัน ทำให้มีการใช้ทรัพยากรของระบบค่อนข้างสูง

UDP (User Datagram Protocol) เป็นการทำงานในลักษณะตรงกันข้ามกับ TCP นั่นคือมีการทำงานในลักษณะ Connectionless เปรียบเหมือนการกระจายเสียงวิทยุ UDP จะเหมาะกับการส่งข้อมูลที่ไม่สำคัญนัก เช่นส่งข้อมูลแสดงสถานะการทำงานซึ่งก็มักจะเป็นข้อมูลขนาดเล็ก ซึ่งทำให้ UDP กินทรัพยากรของระบบน้อยกว่า TCP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Property ที่สำคัญของ Winsock Control

Protocol เป็นการเลือกโพรโตคอลสำหรับการทำงาน

LocalPort เป็นการกำหนดหมายเลขพอร์ตของคอมพิวเตอร์ที่จะใช้งานกับ Winsock

RemoteHost เป็นการกำหนดชื่อของคอมพิวเตอร์ที่เราจะติดต่อด้วย อาจจะเป็น IP Address หรือเป็นชื่อคอมพิวเตอร์ที่เป็นชื่อที่ง่ายต่อการจดจำ

RemotePort เป็นการกำหนดหมายเลขพอร์ตของคอมพิวเตอร์ที่เราจะติดต่อด้วย

ByteReceive เป็นจำนวนข้อมูลที่รับเข้ามาเก็บใน Buffer จากเมธอด GetData

Method ที่สำคัญของ Winsock Control

Listen เป็นเมธอดที่ใช้สร้าง Socket แล้ว รอการ Connect ทำให้คอมพิวเตอร์เครื่องอื่นๆ สามารถติดต่อเข้ามาได้

Connect เป็นเมธอดที่ใช้สร้างการติดต่อแบบ Socket ไปยังคอมพิวเตอร์เครื่องอื่น โดยจะต้องระบุ Socket Address (IP Address กับหมายเลขพอร์ตที่กำหนดให้ใช้กับ Socket)

Accept Request เป็นเมธอดที่ใช้รับ Request จากคอมพิวเตอร์ที่ติดต่อเข้ามา

SendData เป็นเมธอดที่ใช้ส่งข้อมูลไปยังคอมพิวเตอร์เครื่องอื่นที่เราติดต่อแบบ Socket

GetData เป็นเมธอดที่ใช้รับข้อมูลจากบัฟเฟอร์เข้ามาเก็บในตัวแปรที่เรากำหนดให้ โดยสามารถกำหนดชนิดตัวแปร และความยาวของข้อมูลที่จะนำมาเก็บได้

Close เป็นเมธอดที่ใช้ยกเลิกการติดต่อ

Event ที่สำคัญของ Winsock Control

ConnectionRequest เป็นเหตุการณ์ที่เกิดเมื่อ คอมพิวเตอร์เครื่องอื่น มีการ Request เข้ามา ซึ่งจะมีการกำหนด ID ให้กับแต่ละค่าของ Request ที่เข้ามา

Data Arrival เป็นเหตุการณ์ที่เกิดเมื่อมีข้อมูลชุดใหม่เข้ามาเก็บใน Buffer ซึ่งเราสามารถตรวจสอบขนาดข้อมูลได้จาก Property ByteReceive

SendProgress เป็นเหตุการณ์ที่เกิดขณะที่มีการส่งข้อมูลระหว่างกัน ซึ่งจะมีพารามิเตอร์แสดงจำนวนข้อมูลที่ส่งมาแล้ว และข้อมูลที่ยังคงเหลืออยู่

SendComplete เป็นเหตุการณ์ที่เกิดเมื่อการส่งข้อมูลเสร็จสิ้นสมบูรณ์

Error เป็นเหตุการณ์ที่เกิดเมื่อมีความผิดพลาดเกิดขึ้น ซึ่งจะแสดงหมายเลขของความผิดพลาด, คำอธิบาย และรายละเอียดอื่นๆ สำหรับการจัดการกับข้อผิดพลาดที่เกิดขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Dllz32.dll	ไลบรารีสำหรับการสนับสนุนการขยายขนาดของไฟล์ที่ถูกบิดอัด อย่างเช่น ในการสร้างแผ่น setup นั้นไฟล์ต่างๆจะถูกบีบอัดด้วยโปรแกรม compress.exe ซึ่งโดยปกติไฟล์ที่ถูกบีบอัดก็จะมีนามสกุลตัวสุดท้ายเป็นเครื่องหมาย underscore(_) โดยในขณะทำการติดตั้ง ไฟล์เหล่านี้ก็ถูกขยายขนาดอัตโนมัติโดยวินโดวส์ ซึ่งจะอาศัยฟังก์ชันวินโดวส์ API จากไฟล์ไลบรารีนั่นเอง
Kernel32.dll	ไลบรารีที่ประกอบด้วยฟังก์ชันสำหรับการติดต่อในระดับล่าง (low-level) กับระบบปฏิบัติการ การจัดการงาน การควบคุมทรัพยากรของระบบ และการจัดการระบบการใช้งานหน่วยความจำเป็นต้น
User32.dll	ไลบรารีที่ประกอบด้วยฟังก์ชันสำหรับการจัดการกับเมนู Cursor, Caret การจัดการกับเวลาของระบบ การจัดการกับการส่ง message ระหว่าง Application หรือระบบปฏิบัติการ เป็นต้น
Netapi32.dll	ไลบรารีที่ประกอบด้วยฟังก์ชันสำหรับการจัดการด้านระบบเครือข่าย
Odbc32.dll	ไลบรารีที่ประกอบด้วยฟังก์ชันสำหรับการควบคุมการติดต่อสื่อสารระหว่างระบบฐานข้อมูล ที่รูปแบบต่างๆกัน โดยอาศัยเทคโนโลยีที่เรียกว่า Open Database (ODBC) ซึ่งจะช่วยให้การแลกเปลี่ยนข้อมูลระหว่างฐานข้อมูลที่ถูกจัดการโดยเซิร์ฟเวอร์ต่างๆกัน เป็นไปได้โดยสะดวกและง่ายมากขึ้น
Winmm.dll	ไลบรารีที่ประกอบด้วยฟังก์ชันสำหรับการจัดการด้านมัลติมีเดีย เช่นการจัดการหรือควบคุมการเล่นไฟล์ .wav .mid หรือแม้กระทั้งการจัดเก็บ CD-Audio เป็นต้น

2.3.2 การเขียนฟังก์ชัน API ด้วย Visual Basic

เนื่องจากฟังก์ชันวินโดว์ API ส่วนใหญ่จะถูกเขียนาขึ้นมาเพื่อสนับสนุนภาษา C หรือ VISUAL C++ ดังนั้นชนิดของข้อมูลบางส่วนจึงไม่ COMPSTIBLE กับชนิดข้อมูลของ VISUAL BASIC 6.0 ซึ่งในการส่งผ่านตัวแปรให้กับพารามิเตอร์ของฟังก์ชันวินโดว์ API จึงต้องมีการตรวจสอบความเข้ากันได้ของชนิดข้อมูลเสียก่อนทุกครั้ง และความหมายของชนิดข้อมูลเสียก่อนทุกครั้ง และความหมายของชนิดข้อมูลสำหรับภาษา C หรือ VISUAL C++ มีดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 แสดงความหมายของชนิดข้อมูลสำหรับภาษา C หรือ VISUAL C++

ชนิดข้อมูล	รายละเอียด
BOOL	ข้อมูล ชนิด BOOLEAN ขนาด 32 บิต โดยที่ 0 หมายถึง FALSE และค่าไม่เท่ากับ 0 หมายถึง TRUE
BYTE	ข้อมูลชนิดไบต์ ขนาด 8 บิต
CHAR	ข้อมูลชนิดตัวเลขจำนวนเต็ม UNSIGNED ขนาด 8 บิต
FARPROC	ข้อมูลชนิด POINTER ไปยัง PROCEDURE หรือฟังก์ชันขนาด 32 บิต
HANDLE	ข้อมูลชนิดเลขจำนวนเต็ม UNSIGNED ขนาด 32 บิตเพื่อเป็น HANDLE ไปยัง OBJECT
INT	ข้อมูลชนิดเลขจำนวนเต็ม SIGNED ขนาด 32 บิต
SHORT	ข้อมูลชนิดเลขจำนวนเต็ม SIGNED ขนาด 16 บิต
LONG	ข้อมูลชนิดเลขจำนวนเต็ม SIGNED ขนาด 32 บิต
LPINT	ข้อมูลชนิด POINTER ขนาด 32 บิตชี้ไปยังข้อมูลชนิดเลขจำนวนเต็ม SIGNED ขนาด 32 บิต
LPSTR	ข้อมูลชนิด POINTER ขนาด 32 บิตชี้ไปยังข้อมูลชนิดสตริงชนิด แอสกี
UINT	ข้อมูลชนิดตัวเลขจำนวนเต็ม UNSIGNED ขนาด 32 บิต
WORD	ข้อมูลชนิดตัวเลขจำนวนเต็ม UNSIGNED ขนาด 16 บิต
DWORD	ข้อมูลชนิดตัวเลขจำนวนเต็ม UNSIGNED ขนาด 32 บิต
FLAGS	ข้อมูลชนิด FLAG ขนาด 32 บิต
FLOAT	ข้อมูลชนิดเลขจำนวนจริง ขนาด 32 บิต
DOUBLE	ข้อมูลชนิดเลขจำนวนจริง ขนาด 32 บิต
LPUNKNOWN	POINTERขนาด 32 บิต ไปยัง INTERFACE UNKNOWN OLE 2.0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตัวอย่างข้างต้นจะเห็นว่าชนิดข้อมูลของ ภาษา C หรือ VISUAL C++ นั้นมีความหลากหลายมากโดยเฉพาะชนิดชนิดตัวเลขเพราะภาษาC สนับสนุนทั้งข้อมูลตัวเลขที่มีเครื่องหมายและไม่มีเครื่องหมาย แต่สำหรับ VISUAL BASIC 6.0 จะสนับสนุนเฉพาะข้อมูลตัวเลขที่มีเครื่องหมายเท่านั้น ยกเว้นข้อมูลชนิด BYTE ของ VISUAL BASIC 6.0 ที่สนับสนุนข้อมูลชนิดที่ไม่มีเครื่องหมาย ดังนั้นเพื่อให้สามารถเข้าใจถึงการเปรียบเทียบชนิดข้อมูลระหว่างภาษาซี และ VISUAL 6.0 ได้ง่ายขึ้น ก็สามารถพิจารณาได้ดังนี้

1. ข้อมูลชนิดตัวเลข ขนาด 8 บิต ข้อมูลชนิดตัวเลขขนาด 8 บิต สำหรับภาษาซีนั้นมีอยู่ด้วยกัน 2 ประเภทดังนี้ CHAR และ BYTE ซึ่งจะ COMPATIBLE กับข้อมูลชนิด BYTE ของ VISUAL BASIC 6.0 โดยจะต้องมีการประกาศให้กับ พารามิเตอร์ดังนี้

ตารางที่ 2.3 แสดงความหมายของชนิดข้อมูลสำหรับภาษา C หรือ VISUAL C++

C,C++	VISUAL BASIC 6.0
CHAR POSVAL	BYVAL POSVAL AS BYTE
BYTE POSVAL	BYVAL POSVAL AS BYTE

สำหรับข้อมูลชนิด CHAR นั้นสามารถเก็บค่าได้ตั้งแต่ –128-127 แต่ข้อมูลชนิด BYTE ของ VISUAL BASIC 6.0 เป็นข้อมูลชนิดเลขจำนวนเต็ม UNSIGNED ดังนั้นถ้าหากค่าของข้อมูลชนิด CHAR มีค่าเป็นลบ ต้องทำการเขียนโค๊ด เพื่อแปลงค่าของข้อมูลเอาเอง ส่วนข้อมูลชนิด BYTE นั้น COMPATIBLE กับข้อมูลชนิด BYTE ของ VISUAL BASIC 6.0 แต่เนื่องจากวินโดว์ 32 บิต เน้นหนักด้านการระบบปฏิบัติการระบบ 32 บิต ดังนั้น ข้อมูลชนิด CHAR และ BYTE มักจะไม่มีการใช้งานในฟังก์ชันวินโดว์ API

2. ข้อมูลชนิดตัวเลขขนาด 16 บิต สำหรับภาษาซีนั้นมีอยู่ด้วยกัน 2 ประเภทคือ SHORT และ WORD ซึ่งจะ COMPATIBLE กับข้อมูลชนิด INTEGER ของ VISUAL BASIC 6.0 โดยจะต้องมีการประกาศให้กับพารามิเตอร์ดังนี้

ตารางที่ 2.4 แสดงการประกาศพารามิเตอร์ของข้อมูลชนิดตัวเลขขนาด 16 บิต

C, C++	VISUAL BASIC 6.0
SHORT POSVAL	BYVAL POSVAL AS INTEGER
WORD POSVAL	BYVAL POSVAL AS INTEGER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับข้อมูลชนิด SHORT นั้นเป็นข้อมูลชนิดตัวเลข SIGNED ขนาด 16 บิต ซึ่งจะเหมือนกับข้อมูลชนิด INTEGER ของ VISUAL BASIC6.0 ทุกประการ แต่สำหรับข้อมูลชนิด WORD นั้นเป็นข้อมูลชนิดตัวเลข UNSIGNED ขนาด 16 บิต ซึ่งสามารถจัดเก็บค่าตัวเลขได้ ตั้งแต่ 0 ถึง 65535 ดังนั้น จึงไม่เหมือนกับข้อมูลชนิด INTEGER ของ VISUAL BASIC 6.0 แต่เนื่องจากขนาดของข้อมูลเท่ากัน จึงมีความ COMPATIBLE กัน และในการส่งค่าตัวเลขหรืออ่านค่ากลับจากฟังก์ชันวินโดว์ API สำหรับข้อมูล ชนิดตัวเลข UNSIGNED ขนาด 16 บิต ที่มีค่าตั้งแต่ 32768 ถึง 65535 จึงต้องอาศัยเทคนิคในการแปลงข้อมูลตัวเลข

3. ข้อมูลชนิดตัวเลข ขนาด 32 บิต สำหรับภาษาซีนั้นมีอยู่ด้วยกัน 5 ประเภทดังนี้ BOOL DWORD INT UINT LONG ซึ่ง จะ COMPATIBLE กับข้อมูลชนิด LONG ของ VISUAL BASIC 6.0 โดยจะ ต้องมีการประกาศให้กับพารามิเตอร์ดังนี้

ตารางที่ 2.5 แสดงการประกาศพารามิเตอร์ของข้อมูลชนิดตัวเลขขนาด 32 บิต

C,C++	VISUAL BASIC 6.0
BOOL POSVAL	BYVAL POSVAL AS LONG
DWORD POSVAL	BYVAL POSVAL AS LONG
INT POSVAL	BYVAL POSVAL AS LONG
LONG POSVAL	BYVAL POSVAL AS LONG
UINT POSVAL	BYVAL POSVAL AS LONG

ข้อมูลชนิด INT และ LONG นั้นเป็นข้อมูลชนิดตัวเลข SIGNED ขนาด 32 บิต ซึ่งจะเหมือนกับข้อมูลชนิด LONG ของ VISUAL BASIC 6.00 ทุกประการ แต่สำหรับข้อมูลชนิด DWORD และ UINT นั้นเป็นข้อมูลชนิดตัวเลข UNSIGNED ขนาด 32 บิต และสามารถจัดเก็บค่าตัวเลขได้ตั้งแต่ 0 ถึง 4294967295 ดังนั้นจึงไม่เหมือนกับข้อมูลชนิด LONG ของ VISUAL BASIC 6.0 แต่เนื่องจากขนาดของข้อมูลเท่ากัน จึงมีความ COMPATIBLE กัน และในการส่งค่าตัวเลขหรืออ่านค่ากลับจากฟังก์ชันวินโดว์ API สำหรับข้อมูลชนิดตัวเลข UNSIGNED ขนาด 32 บิต ที่มีค่าตั้งแต่ 2147483647 ถึง 4294967295 นั้นจึงต้องมีเทคนิคในการแปลงข้อมูล

4. ข้อมูลชนิด บูลีน นั้นสำหรับ VISUAL BASIC 6.0 จะมีอยู่เพียง 2 ความหมาย ดังนี้ 0 หมายถึง FALSE ส่วนค่าอื่นๆที่ไม่เท่ากับ 0 หรือโดยปรกติดค่าของ –1 จะหมายถึง TRUE แต่สำหรับภาษาซีมีข้อแตกต่าง กับ VISUAL BASIC 6.0 ตรงที่ค่าของ TRUE นั้นจะเท่ากับ 1 ส่วนค่าของ FALSE นั้นจะเท่ากับ 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ข้อมูลชนิดตัวเลขจำนวนจริงขนาด 32 บิต สำหรับภาษาซีนั้น มีอยู่ด้วยกันเพียงประการเดียวคือ FLOAT ซึ่งจะเหมือนกับข้อมูลชนิด SINGLE ของ VISUAL BASIC 6.0 ทุกประการ ดังนั้นจึงไม่จำเป็นต้องมีการแปลงข้อมูลที่ได้

6. ข้อมูลชนิดตัวเลขจำนวนจริงขนาด 64 บิต สำหรับภาษาซีนั้นมีอยู่ด้วยกันเพียงประการเดียวคือ DOUBLE ซึ่งจะเหมือนกับข้อมูลชนิด DOUBLE ของ VISUAL BASIC 6.0 ทุกประการ ดังนั้นจึงไม่จำเป็นต้องมีการเปลี่ยนแปลงข้อมูลที่ได้ ซึ่งสามารถประกาศให้กับพารามิเตอร์

7. ข้อมูลชนิด HANDLE สำหรับวินโดว์ จะมีขนาดเท่ากับ 32 บิต สำหรับใช้กำหนดหรืออ้างอิง ชนิดของ ออปเจค ของวินโดว์ ดังนั้นถ้าหากมีการกำหนดตัวแปร สำหรับ รับหรือส่งข้อมูลชนิด HANDLE นี้ ต้องใช้ข้อมูลชนิด LONG ของ VISUAL BASIC

8. ข้อมูลชนิดสตริง ที่สนับสนุนโดย C หรือ C++ นั้น เป็น NULL-TERMINATED ซึ่งแตกต่างจากสตริงของ VISUAL BASIC 6.0 โดยสิ้นเชิง ดังนั้น จึงไม่สามารถส่งผ่านสตริง ของ VISUAL BASIC 6.0 ให้อยู่ในรูปแบบของ NULL-TERMINATED

2.3.3 การประกาศฟังก์ชันวินโดวส์ API

ก่อนที่จะสามารถใช้งานฟังก์ชันวินโดวส์ API ในการเขียนโปรแกรม จะต้องทำการประกาศฟังก์ชันนั้นเสียก่อน เพื่อเป็นการกำหนด Prototype ในการติดต่อระหว่างฟังก์ชันวินโดวส์ API กับVisual Basic 6.0 สำหรับการประกาศฟังก์ชันวินโดวส์ API ซึ่งมีอยู่ด้วยกัน 2 รูปแบบ คือ การประกาศ Procedure ประเภท Sub routine และ การ Procedure ประเภทฟังก์ชัน

รูปแบบที่ 1

[Public l Private] Declare Sub name Lip “lipname”[Alias”aliasname”][(arglist])]

รูปแบบที่2

[Public l private] Declare Function name Lip “lipname”[Alias”aliasname”][(arglist)][As Type]

ตารางที่ 2.6 แสดงการประกาศฟังก์ชันวินโดวส์ API

Public	กำหนดให้ Procedure ที่ถูกประกาศสามารถถูกเรียกใช้ได้ จาก Module หรือ Form อื่นใน Project ปัจจุบัน แต่การประกาศ Procedure ด้วย Public จะต้องกระทำใน Module มาตรฐาน (.bas)เท่านั้น
Private	กำหนดให้ Procedure ที่ถูกประกาศสามารถถูกเรียกใช้ได้ เฉพาะจาก Module หรือ Form ที่ประกาศ Procedure นี้เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Sub	กำหนดให้ Procedure เป็นชนิด Routine ซึ่งจะไม่มีการส่งค่ากลับ
Function	กำหนดให้ Procedure เป็นชนิดฟังก์ชัน ซึ่งจะมีค่าส่งกลับ ดังนั้นจึงสามารถนำค่าที่ได้จากฟังก์ชันไปใช้ในลักษณะของตัวแปรหนึ่งๆ ในนิพจน์ (expression)เช่น นิพจน์การคำนวณ
Name	กำหนดชื่อของ Procedure สำหรับ Visual Basic 6.0 ยอมให้ชื่อของ Procedure ความยาวทั้งหมดเพียง 40 ตัวอักษรเท่านั้น
Lip	กำหนดให้สตริงที่ตาม (“libname”) จะบ่งบอกถึงชื่อของไฟล์ไลบรารี .dll ซึ่งใช้ในการประกาศฟังก์ชันวินโดวส์ API จะต้องมีการกำหนดชื่อของไฟล์ไลบรารี .dll ทุกครั้ง
Alias	กำหนดให้ Procedure ชื่อ “aliasname” สามารถถูกเรียกใช้ได้โดยใช้ชื่ออื่นๆ ตามที่กำหนด โดย name ทั้งนี้เนื่องจากในบางกรณีชื่อของฟังก์ชันวินโดวส์ API นั้นอาจจะซ้ำกับประโยคหรือคำสั่งของ Visual Basic 6.0 หรืออาจจะประกอบด้วยตัวอักษรที่ Visual Basic 6.0 ไม่ยอมรับในการกำหนดเป็นชื่อของ Procedure
Arglist	รายการของพารามิเตอร์ที่จะส่งให้กับ Procedure
ByVal	กำหนดให้พารามิเตอร์มีการส่งค่าแบบ passed by value แต่ถ้าหากนำไปใช้ประกาศฟังก์ชันวินโดวส์ API แต่ก็จะมีความหมายตามแต่ชนิดข้อมูลของพารามิเตอร์ตามที่เคยกล่าวรายละเอียดมาแล้วข้างต้น เช่น Byte StrVal$ หมายถึง ASCIIZ
ByRef	กำหนดให้พารามิเตอร์มีการส่งค่าแบบ pass by reference
ParamArray	ใช้กำหนดให้พารามิเตอร์ดังกล่าวเป็น array ของข้อมูลชนิด variant ซึ่งจะต้องเป็นพารามิเตอร์ตัวสุดท้ายของ Procedure เท่านั้น โดยทั่วไปส่วนนี้มักไม่สามารถใช้ได้กับฟังก์ชันวินโดวส์ API มาตรฐาน ยกเว้นฟังก์ชันถูกออกแบบเฉพาะกับ Visual Basic 6.0 เท่านั้น

ตัวอย่างการประกาศ Function Windows API

```
Public Declare Funtion Sendmessage Lip “User32.dll” Alias_
        “SenmessageA” (ByVal hwnd As Long,ByVal wMsg As Long,_
        ByVal wParam As Long,lParam As Long ) As Long
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล.

การประกาศฟังก์ชันวินโดวส์ Sendmessage ตามตัวอย่างข้างบน กำหนดให้เป็นฟังก์ชันที่มีการส่งค่ากลับเป็นข้อมูลชนิด long integer (4byte) โดยที่เป็นฟังก์ชัน ชื่อ Sendmessage ซึ่งชื่อฟังชันเดิมที่ถูกจัดเก็บในไฟล์ไลบรารีได้แก่ SendmessageA และฟังก์ชันของไฟล์ไลบรารี User32.dll โดยที่ฟังก์ชันนี้จะประกอบด้วยพารามิเตอร์ทั้งหมด 4 ตัว ดังมีรายละเอียดดังต่อไปนี้

Hwnd หมายเลข handle ของออบเจ็กต์ ซึ่งเป็นข้อมูลชนิด long

WMsg หมายเลข message ของวินโดวส์ ซึ่งเป็นข้อมูลชนิด long

WParam ข้อมูลการทำงานของฟังก์ชัน ซึ่งเป็นข้อมูลชนิด long

LParam ข้อมูลสำหรับการทำงานของฟังก์ชัน ซึ่งเป็นข้อมูลชนิด long

2.4 การจัดการ Process ของ Windows

เราใช้ Applicationmmที่มีมากับ Visual basic 6.0 ในการสร้างหน้าต่างในการควบคุม process ต่างๆ ของ ระบบปฏิบัติการ windows

Process ของวินโดว์ ที่สามารถจัดการได้ คือ จัดการกับเหตุการณ์เปลี่ยนแปลงที่เกิดขึ้น บนเครื่อง ทั้ง เครื่องที่เป็นตัวควบคุม และเครื่องที่เป็นตัวถูกควบคุม โดยสามารถจัดการได้ดังนี้

- อ่าน และ Detect process ที่ running บนเครื่อง ที่ถูกควบคุมอยู่ หรือ เครื่อง client ได้
- เริ่ม process บนเครื่องที่เครื่องที่ทำงานเป็นเครื่องควบคุม หรือ เครื่อง server ได้
- หยุด process การทำงานทั้งหมด หรือหยุดเพียงบางส่วน ของเครื่องที่เป็นเครื่องควบคุม และ เครื่องที่ถูกควบคุมได้
- อ่าน modules ต่างๆ ที่ใช้จากการ running process
- ใช้ในการเลือกปิดบาง process ที่ไม่ต้องการใช้งาน หรือ ต้องการให้ process นั้นหยุด run ในกรณีใดๆ
- ตรวจสอบ properties ของอุปกรณ์ที่ใช้งานอยู่ ทั้งส่วนที่เป็นเครื่องควบคุม และเครื่องที่ถูกควบคุม
- ในส่วนการให้บริการ สามารถกำหนดให้มีการ เริ่ม หรือ หยุด process การให้บริการได้

เราสามารถใช้ component ที่มีอยู่ใน Visual Basic ในการจัดการกับ windows process ได้อย่างสะดวกและรวดเร็ว สามารถใช้ component ที่มีอยู่เหล่านี้ ในการควบคุมได้ทั้งเครื่องที่เป็นเครื่องควบคุม และ เครื่องที่ถูกควบคุม ในเครื่องที่เป็นตัวควบคุม เราสามารถเริ่ม และ หยุด process อีกทั้งยังสามารถทำงานตามประเภทคำสั่งที่ระบุไว้ได้ สามารถแสดง ชื่อ ของ modules ต่างๆที่ component นั้นมีการโหลดมาใช้ สามารถ ตั้งเวลาเริ่ม processs ได้ แสดงจำนวน ผู้ที่เข้ามา join ใน

เอกสารนี้มีลิขสิทธิ์ ห้ามนำไปใช้เพื่อการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบได้ และมีการเรียงลำดับตาม threads ที่เข้าทำงาน เป็นต้น ในส่วนของเครื่องที่เป็นตัวถูกควบคุม เราไม่สามารถ เริ่ม หรือ หยุด process ได้โดย ตรง แต่สามารถ query process ที่มีอยู่แล้วโดย รับมาเป็นข้อมูลได้

การทำงานกับ process เรากำหนดคำสั่งให้อยู่ใน classes ต่างๆ ที่เราสร้างขึ้น โดยสร้าง process component ลงใน classes ทำให้ class ของเราสามารถ access ตัว process ของ เครื่องควบคุมได้ อีกทั้งเริ่ม และ หยุด processใหม่ มีบาง Components ในโปรแกรม สามารถ เรียก Process Component ได้เลยในกรณี ที่จำเป็นต้อง เริ่ม process ใหม่ หรือ แก้ไข process เดิมที่มีอยู่แล้ว

ข้อจำกัด

ในด้านความปลอดภัย เราต้องการให้มีความแม่นยำสูง ในการ access และให้ตรงกับเครื่องที่เป็น target ก่อนที่จะมีการ run code คำสั่ง ในการสั่งให้ทำงาน process หรือรับข้อมูล จากการ process ทุกครั้งต้องมีการตรวจสอบ authorization ก่อน ว่าสามารถ access process นั้นๆก่อน มิฉะนั้นจะเกิด error ในการ access process [2]

2.5 ทฤษฎีพื้นฐานโครงสร้างของโพรโตคอล TCP/IP และ UDP

ทฤษฎีพื้นฐานที่ใช้ในโครงงานได้แก่ โปรแกรมอำนวยการสอนภายในห้อง LAB Computer (Close View Learning), โครงสร้างของโพรโตคอล TCP/IP มีรายละเอียดดังนี้

2.5.1 โครงสร้างของโพรโตคอล TCP/IP

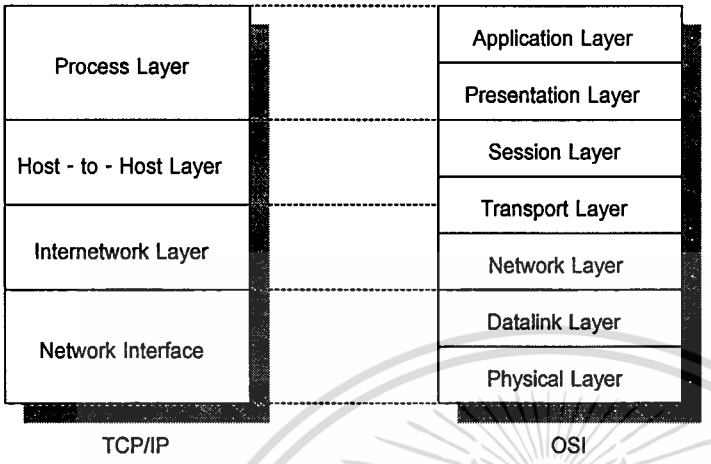
โพรโตคอล TCP/IP มีการจัดกลไกการทำงานเป็นชั้นหรือ Layer เรียงต่อกัน โดยในแต่ละ Layer จะมีการทำงานเทียบได้กับ OSI model มาตรฐาน แต่บาง Layer ของโพรโตคอล TCP/IP จะทำงานเทียบกับ OSI หลาย Layer ปนกัน ซึ่งในแต่ละ Layer ของโพรโตคอล TCP/IP จะประกอบด้วย

- Process Layer
- Host -- to – Host Layer
- Internetwork Layer
- Network Interface Layer

โดยเมื่อเทียบกับมาตรฐาน OSI model ซึ่งเราจะเห็นว่าบางกลไกของโพรโตคอล TCP/IP เทียบได้กับมาตรฐาน OSI model สองชั้น หรือบางกลไกก็จะทำงานคาบเกี่ยวกันระหว่างบางชั้นของ OSI model ตัวอย่างเช่น กลไกการทำงานของโพรโตคอล TCP/IP ในส่วน Network Interface Layer เมื่อเทียบกับมาตรฐาน OSI model จะเทียบได้กับ Data Link Layer และ Physical Layer 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชั้นรวมกัน เป็นต้น ในแต่ละกลไกของโพรโตคอล TCP/IP จะโพรโตคอลอื่น ๆ ในชุดของ TCP/IP ร่วมทำงานอยู่ด้วย ซึ่งจะกล่าวโดยละเอียดต่อไป



รูปที่ 2.1 แสดง TCP/IP stack เปรียบเทียบกับมาตรฐาน OSI

Process Layer

การแสดงลำดับชั้นการทำงานของโพรโตคอล TCP/IP เทียบกับมาตรฐาน OSI model นั้น ในชั้นบนสุดเรียกว่า Process Layer ทำงาน 2 หน้าที่เทียบได้กับ Application Layer และ Presentation Layer ในชั้นนี้จะรองรับการทำงานของแอพพลิชันต่าง ๆ ที่ทำงานเป็นโปรเซ็ส อยู่ในเครื่องเซิร์ฟเวอร์ที่ให้บริการและเครื่องที่ขอใช้บริการ หรือไคลเอนต์ (client) ซึ่งจะติดต่อกันผ่านโพรโตคอลเฉพาะแอพพลิเคชันอีกทีหนึ่ง ตัวอย่างเช่น เมื่อผู้ใช้งานอินเตอร์เน็ตต้องการโอนถ่ายไฟล์หรือ download ข้อมูลจากเครื่องเซิร์ฟเวอร์ที่ให้บริการ โดยอาจจะเรียกใช้โปรแกรม ftp client ทั่วไป เช่น โปรแกรม WS_ftp ติดต่อกับโปรเซ็ส ftp ที่กำลังให้บริการอยู่ที่เครื่องเซิร์ฟเวอร์ จากนั้นตัวโปรเซ็ส ftp ก็จะเรียกใช้โพรโตคอล FTP (File Transfer Protocol) เพื่อทำการโอนถ่ายไฟล์นี้ หรือถ้าผู้ใช้ต้องการเรียกใช้งานคอมพิวเตอร์ที่อยู่ห่างไกลออกไปด้วยการใช้โปรแกรม telnet ที่เครื่องเซิร์ฟเวอร์ให้บริการ ตัวโปรเซ็ส Telnet ที่ทำงานอยู่ก็จะเรียกใช้โพรโตคอล Telnet เพื่อติดต่อกัน หรือในกรณีที่มีการเรียกใช้โปรแกรม web browser เช่น Netscape Navigator เพื่อเรียกดูเว็บไซท์ CNN ที่เครื่องซึ่งให้บริการเว็บของ CNN ก็จะมีโปรเซ็ส HTTP (HyperText Transfer Protocol) ทำงานอยู่และจะติดต่อกับผู้ใช้ผ่านโพรโตคอล HTTP เป็นต้น

การทำงานของแอพพลิเคชันต่าง ๆ จะอยู่ที่ Process Layer นี้ และมีการติดต่อกันตามแต่ละโพรโตคอลเฉพาะแล้วแต่แอพพลิเคชันที่ใช้งาน จากการที่ Process Layer ของ TCP/IP รองรับให้โพรโตคอลอื่นทำงานได้หลายโปรเซ็สและหลายโพรโตคอลได้พร้อมกันนั้น ทำให้ผู้ใช้สามารถเปิดโปรแกรมใช้งานได้หลาย ๆ โปรแกรมพร้อมกัน เช่น เปิดโปรแกรม Internet Explorer เพื่อเรียก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดูเว็บเพจ พร้อมกับใช้งานโปรแกรม Outlook Express เพื่อรับส่งอีเมล์ไปพร้อมกันได้โดยไม่ต้องรอให้ทำงานอย่างหนึ่งอย่างใดเสร็จก่อน หรือในปัจจุบันมีการพัฒนาโปรแกรม web browser ให้สามารถเรียกใช้งานโพรโตคอลอื่น ๆ ได้มากขึ้น ทำให้เราสามารถใช้โปรแกรม web browser โอนถ่ายไฟล์ข้อมูลที่ใช้โพรโตคอล FTP ได้โดยไม่ต้องไปหาโปรแกรมอื่นมาใช้

โพรโตคอลหลัก ๆ ที่ทำงานใน Process Layer ซึ่งผู้ใช้มักจะคุ้นเคยกันดีได้แก่ FTP(File Transfer Protocol), Telnet, HTTP(HyperText Transfer Protocol), SMT(Simple Mail Transfer protocol) นอกจากนี้ยังมีโพรโตคอลอื่นที่อยู่เบื้องหลัง ซึ่งทำงานโดยที่ผู้ใช้ไม่ได้มีการใช้งานโดยตรง เช่น

- โพรโตคอล DNS (Domain Name System) ที่ทำหน้าที่แปลงข้อมูลชื่อ domin name หรือชื่อเว็บไซท์ทั้งหลายให้เป็นหมายเลข IP address

- โพรโตคอล DHCP (Dynamic Host Configuration Protocol) ทำหน้าที่แจกจ่ายข้อมูลพารามิเตอร์ของเครือข่ายให้กับเครื่องลูกข่ายที่เชื่อมต่ออยู่

Host – To – Host Layer

การทำงานที่ชั้นของ Host – to – Host Layer นี้จะมีบทบาทในการจัดการต่อจาก Process Layer บางครั้งเรามักเรียกชั้น Host – to – Host ว่าเป็น Transport Layer ซึ่งไม่ใช่ชั้นของ Transport Layer ในมาตรฐาน OSI model การทำงานของ Host – to Host Layer นี้จะมีการสร้าง connection หรือการเชื่อมต่อกันระหว่างแอพพลิเคชันกับ Host – to – Host Layer โดยจุดที่เชื่อมกันเพื่อรับส่งข้อมูลที่เรียกว่า port หรือ socket (คำว่า port ในที่นี้ไม่ได้หมายถึง port ทางฮาร์ดแวร์) และในแต่ละแอพพลิเคชันก็จะสร้างการเชื่อมต่อผ่าน port ได้พร้อมกันหลายแอพพลิเคชัน ซึ่งการใช้งาน port ของแต่ละแอพลิเคชันที่อยู่ในชั้น Process Layer จะแตกต่างกันตามหมายเลขที่กำหนดไว้ และแต่ละโพรโตคอลจะมีการใช้งาน port หมายเลขต่าง ๆ ไม่ซ้ำกัน

เมื่อแอพพลิเคชั่นทำงานผ่านโพรโตคอลในชั้น Process Layer จะมีการส่งผ่านข้อมูลไปยัง Host – to – Host Layer ที่ชั้นนี้จะมีการเชื่อมต่อผ่าน port ที่กำหนด ทำให้การรับส่งข้อมูลในแต่ละโพรโตคอลทำได้ถูกต้อง ถึงแม้ว่าในเครื่องเซิร์ฟเวอร์ที่ให้บริการจะมีการทำงานอยู่หลายโปรเซ็สที่แตกต่างกันก็ตาม หรือมีผู้ใช้บริการเข้ามาใช้งานพร้อมกันจำนวนมากและหลายแอพลิเคชันในเวลาเดียวกัน ในชั้น Host – to – Host หรือ Transport Layer ของ TCP/IP นี้ จะมีโพรโตคอลทำงานอยู่ 2 โพรโตคอลที่แตกต่างกัน คือ โพรโตคอล TCP และโพรโตคอล UDP (User Datagram Protocol) ในการส่งผ่านข้อมูลลงไปที่ชั้นถัด ๆ ไป เราจะเห็นว่าโพรโตคอล TCP และ UDP จะถูกผนึกเข้าไปในโพรโตคอล IP อีกทีหนึ่งและส่งต่อไปยังเครือข่ายอินเตอร์เน็ตต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวโพรโตคอล TCP และโพรโตคอล UDP จะมีแอพพลิเคชันเฉพาะเพื่อเรียกใช้งานแยกกัน คือ แอพพลิเคชันที่ใช้โพรโตคอล FTP, Internet, HTTP และ SMTP จะมีการส่งผ่านข้อมูลโดยเรียกใช้โพรโตคอล TCP ส่วนแอพพลิเคชันที่ใช้โพรโตคอล SNMP และ DHCP จะส่งผ่านข้อมูลโดยเรียกใช้โพรโตคอล UDP และสำหรับโพรโตคอล DNS นั้น จะสามารถเรียกใช้งานได้ทั้ง TCP และ UDP

Internetwork Layer

ในระดับล่างต่อมาในชั้น Internetwork Layer มีหน้าที่ส่งผ่านข้อมูลในระหว่างเครือข่าย โดยมีโพรโตคอลที่ทำงานเป็นกลไกสำคัญในการส่งผ่านข้อมูลไปยังเครือข่ายใด ๆ บนอินเตอร์เน็ต คือ โพรโตคอล IP (Internet Protocol) นอกจากนี้ในชั้น Internetwork Layer ยังมีโพรโตคอลทำงานอยู่ด้วยอีก 2 ชนิดคือ โพรโตคอล Internet Control Message Protocol (ICMP) และโพรโตคอล Address Resolution Protocol (ARP) แต่ที่ใช้ในการทำโครงงานนี้ คือ โพรโตคอล IP

Network Interface Layer

เนื่องจากในด้านกายภาพของเครือข่ายนั้น มีหลายวิธีการและหลายรูปแบบในการเชื่อมต่อระบบให้เป็นเครือข่าย แต่อย่างไรก็ตามในเครือข่ายอินเตอร์เน็ตนี้ ข้อมูลหรือ IP datagram จะถูกถ่ายทอดและส่งผ่านไปยังปลายทางโดยไม่คำนึงถึงรูปแบบการเชื่อมต่อทางกายภาพ ไม่ว่าจะเป็นการใช้เครือข่ายใยแก้วนำแสงหรือเครือข่ายสาย Unshielded Twist Pair (UTP) เชื่อมต่อเป็นแบบเครือข่าย Ethernet ธรรมดาหรือเครือข่าย Token Ring, ATM, ISDN ฯลฯ ก็ตาม

การทำงานระดับล่างสุดต่อจาก Internetwork Layer จะเป็นการแปลงข้อมูล IP datagram ให้อยู่ในรูปที่เหมาะสม และแปลงเป็นสัญญาณไฟฟ้าส่งไปยังเครือข่ายต่อไป ซึ่งในชั้น Network Interface Layer นี้เมื่อเทียบกับมาตรฐาน OSI model แล้วจะเป็นการรวม 2 Layer เข้าด้วยกันคือ Data link laye และ Physical Layer กล่าวโดยสรุปคือ การทำงานในชั้นต่าง ๆ ตามโครงสร้างของโพรโตคอล TCP/IP

2.5.2 โพรโตคอล TCP

โพรโตคอล TCP (Transmission Control Protocol) เป็นโพรโตคอลที่มีการรับส่งข้อมูลแบบ stream oriented protocol หมายความว่า การรับส่งข้อมูลจะไม่คำนึงถึงปริมาณข้อมูลที่จะส่งไป แต่จะแบ่งข้อมูลเป็นส่วนย่อย ๆ ก่อน แล้วจึงส่งไปยังปลายทางอย่างต่อเนื่องเป็นลำดับข้อมูล ในกรณีที่ข้อมูลส่วนใดส่วนหนึ่งสูญหายไป ก็จะส่งข้อมูลส่วนนั้นใหม่อีกครั้ง สำหรับปลายทางก็

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะทำหน้าที่จัดเรียงส่วนของข้อมูล datagram ดังนั้นแอพพลิเคชันหรือโปรเซ็สใดที่อาศัยการส่งผ่านข้อมูลด้วยโพรโตคอล TCP จะต้องใช้หน่วยความจำและขนาดของช่องสัญญาณ (bandwidth) มากกว่า UDP

การติดต่อระหว่างกันจะต้องเป็นแบบ connection – oriented คือต้องมีการสร้างการติดต่อกันเป็น session ทั้ง 2 ด้านเสียก่อน แล้วจึงจะรับส่งข้อมูลไปได้พร้อมกัน (full duplex) เหมือนกับการใช้โทรศัพท์ติดต่อกัน เมื่อผู้ติดต่อต้นทางเรียกให้ฝ่ายตรงข้ามรับสายแล้วจึงเริ่มการสนทนา เช่น พูดคำว่า “สวัสดี” หรือ “ฮัลโหล” กันก่อนเพื่อให้แน่ใจว่าฝ่ายตรงข้ามพร้อมจะติดต่อด้วย จากนั้นจึงเริ่มต้นติดต่อกัน และเมื่อต้องการจะเลิกการติดต่อก็จะมีการพูดคำว่า “สวัสดี” ให้ฝ่ายตรงข้ามทราบว่าจะเลิกการติดต่อและวางสายไป ซึ่งในระหว่างการติดต่อกันนั้น แม้ว่าฝ่ายใดฝ่ายหนึ่งหรือทั้งสองฝ่ายจะเงียบไป คือไม่พูดอะไรเป็นเวลานาน ๆ แต่การเชื่อมโยงระหว่างทั้งสองด้านยังคงมีอยู่ไม่ขาดไปจนกว่าฝ่ายใดฝ่ายหนึ่งจะวางสาย เช่นเดียวกับการติดต่อกันด้วยกลไกโพรโตคอล TCP เมื่อแอพพลิเคชันต้องการส่งผ่านข้อมูลจะใช้โพรโตคอลที่เหมาะสมในชั้น Process Layer ติดต่อไปและมีการสร้างช่องส่งข้อมูลผ่าน port ที่กำหนดเพื่อส่งผ่านข้อมูลไปยังโพรโตคอล TCP

ในระหว่างการรับส่งข้อมูลนี้ โพรโตคอล TCP จะเพิ่มขบวนการสอบทานข้อมูล เพื่อให้ข้อมูลมีความถูกต้องไม่ผิดพลาดไปจากเดิม โดยการส่งสัญญาณสอบทานข้อมูล (acknowledgment) และส่งข้อมูลให้ใหม่อีกครั้ง ถ้าปลายทางไม่ได้รับหรือเกิดความผิดพลาดขึ้น

ความน่าเชื่อถือของการส่งผ่านข้อมูลโดยโพรโตคอล TCP จะมีมากกว่า แต่ก็ต้องอาศัยทรัพยากรของระบบมากกว่าในการทำงานเช่นกัน

2.5.3 โพรโตคอล UDP

ใน Host – to – Host Layer นอกจากจะมีโพรโตคอล TCP ทำงานแล้ว ก็ยังมีโพรโตคอล UDP (User Datagram Protocol) ในการส่งข้อมูลแต่ละครั้งและไม่มีการส่งข้อมูลใหม่อีกในกรณีที่เกิดความผิดพลาดของการส่งข้อมูล เมื่อเป็นเช่นนี้แอพพลิเคชันหรือโปรเซ็สใดที่ต้องอาศัยโพรโตคอล UDP จะเป็นแบบที่ทั้งสองด้านไม่จำเป็นต้องอาศัยการสร้างช่องทางเชื่อมต่อกัน (connectionless) ระหว่างเครื่องเซิร์ฟเวอร์ให้บริการกับเครื่องที่ขอใช้บริการ โดยไม่ต้องแจ้งให้ฝ่ายรับข้อมูลเตรียมรับข้อมูลเหมือนโพรโตคอล TCP และไม่มีการตรวจสอบความถูกต้องครบถ้วนในการรับส่งข้อมูลนั้น ๆ ด้วย เนื่องจากโพรโตคอล UDP ไม่มีสัญญาณสอบทานข้อมูล (acknowledgement) ในการส่งข้อมูลแต่ละครั้ง และไม่มีการส่งข้อมูลใหม่อีกในกรณีที่เกิดความผิดพลาดของการส่งข้อมูล เมื่อเป็นเช่นนี้ Application หรือ Process ใดที่ต้องอาศัยโพรโตคอล UDP ในการส่งผ่านข้อมูลก็อาจจะต้องสร้างขบวนการตรวจสอบข้อมูลขึ้นมาเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างขั้นตอนกลไกการทำงานโดยใช้โพรโตคอล UDP มีดังต่อไปนี้

1. ในชั้นของ Process Layer เมื่อโปรแกรมควบคุมอุปกรณ์เครือข่าย เช่น โปรแกรม Network Management ต้องการส่งข้อมูลไปยังอุปกรณ์ที่ต้องการ Application นั้นจะติดต่อผ่านโพรโตคอล SNMP ในชั้น Process Layer
2. โพรโตคอล SNMP จะติดต่อกับโพรโตคอล UDP ในชั้นถัดไป เพื่อขอติดต่อผ่าน port ที่กำหนด
3. โพรโตคอล SNMP เตรียมข้อมูลที่จะส่ง รวมทั้งที่อยู่ปลายทาง
4. โพรโตคอล SNMP ส่งผ่านข้อมูลให้โพรโตคอล UDP ที่อยู่ในชั้น Host – to – Host Layer
5. โพรโตคอล UDP ทำหน้าที่ผนึกข้อมูลหรือ datagram นั้น ไปกับโพรโตคอล IP ในชั้นถัดลงไป เพื่อส่งข้อมูลออกจากเครื่อง

ซึ่งจะเห็นว่ามีกลไกที่ต่างจากการส่งข้อมูลด้วยโพรโตคอล TCP ซึ่งจะต้องมีการติดต่อกันก่อน และทั้งสองฝ่ายรับทราบการรับส่งข้อมูลของช่องการส่งข้อมูลนั้น

2.5.4 โพรโตคอล IP

โพรโตคอล IP ทำหน้าที่ให้บริการส่งผ่านข้อมูลที่มาจาก Host -- to -- Host Layer เพื่อส่งข้ามไปยังเครือข่ายใด ๆ ได้อย่างถูกต้อง แม้ว่าจะมีเครือข่ายเชื่อมต่อกันอยู่ในอินเตอร์เน็ตเป็นล้าน ๆ เครือข่ายก็ตาม เนื่องจากโพรโตคอล IP มีข้อมูลตำแหน่ง IP ปลายทางที่จะส่งข้อมูลไปให้โดยทำงานร่วมกับอุปกรณ์ router เพื่อส่งข้อมูลข้ามเครือข่ายออกไปได้ ตัวโพรโตคอล IP จะทำงานแบบ packet switching คือมีการส่งข้อมูลผ่านสวิทซ์ (switch) ไปยังปลายทาง โดยข้อมูลจะเดินทางไปยังเครือข่ายต่าง ๆ ผ่านสวิทซ์นี้ไปเรื่อง ๆ จนกว่าจะถึงปลายทาง ตัววงจรผ่านหรือ switch นี้อาจเป็น gateway หรือ router ในระบบเครือข่ายก็ได้ ซึ่งในข้อมูลของโพรโตคอล IP จะมีข้อมูลของหมายเลข IP ที่จะส่งข้อมูลไปและเมื่อถึงเครือข่ายปลายทางแล้ว จะมีกลไกแปลงหมายเลข IPให้เป็นหมายเลขฮาร์ดแวร์ประจำเครื่องที่ถูกต้องอีกทีหนึ่งด้วยโพรโตคอล ARP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.5 โพรโตคอล ARP

โพรโตคอล ARP (Address Resolution Protocol) ถูกเรียกใช้งานโดยโพรโตคอล IP เพื่อช่วยแปลงหมายเลข IP ไปเป็นหมายเลขฮาร์ดแวร์ปลายทาง ตัวอย่างเช่น เว็บเซิร์ฟเวอร์เครื่องหนึ่งเชื่อมต่ออยู่ในเครือข่ายอินเตอร์เน็ต และในการเชื่อมต่อนี้ต้องอาศัย Network Interface Card (NIC) หรือ LAN card ติดตั้งอยู่ที่ LAN card นี้เองจะมีหมายเลขเฉพาะประจำฮาร์ดแวร์ที่ไม่ซ้ำกับใคร เพื่อใช้อ้างอิงการส่งข้อมูลในเครือข่าย แต่เมื่อมาใช้งานในโพรโตคอล TCP/IP ก็จะต้องมีการกำหนดหมายเลข IP address ประจำตัวเพื่อใช้อ้างอิงกัน และโพรโตคอล ARP จะทำหน้าที่แปลงค่าหมายเลข IP ให้เป็นหมายเลขฮาร์ดแวร์จริงให้ในระดับการทำงานที่ Internetwork Layer นี้ ซึ่งกลไกการแปลงนี้เรียกว่า address resolution [3]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบโครงสร้างระบบ

ระบบบริหารการให้บริการเครื่องคอมพิวเตอร์ ประกอบไปด้วยโปรแกรม Server ทำหน้าที่ส่วนใหญ่คือเป็นศูนย์กลางในการควบคุมการให้บริการ เก็บสิทธิการใช้งาน, เวลาในการใช้เครื่อง, การควบคุมการใช้งานโปรแกรมที่ไม่ได้รับอนุญาต, การเก็บสถิติการให้บริการและเหตุการณ์ที่เกิดขึ้น รวมไปถึงการนำข้อมูลการให้บริการมาสรุปเป็นรายงาน

3.1 การออกแบบการทำงานของ Server Program และระบบโดยรวม

โปรแกรม Server มีการให้บริการ Web Server เพื่อให้โปรแกรม Client นำ html ที่ได้รับไปแสดงผลในการ lock หน้าจอเครื่อง เพื่อให้ผู้ดูแลระบบสามารถแก้ไปเปลี่ยนแปลงลักษณะและข้อมูลที่ใช้ในการแสดงบนหน้าจอได้ง่าย หรืออาจแยกการให้บริการ HTTP Service นี้ออกไปอยู่บนเครื่องคอมพิวเตอร์อื่นที่ทำหน้าที่เป็น Web Server ที่มีอยู่ให้บริการแทนได้

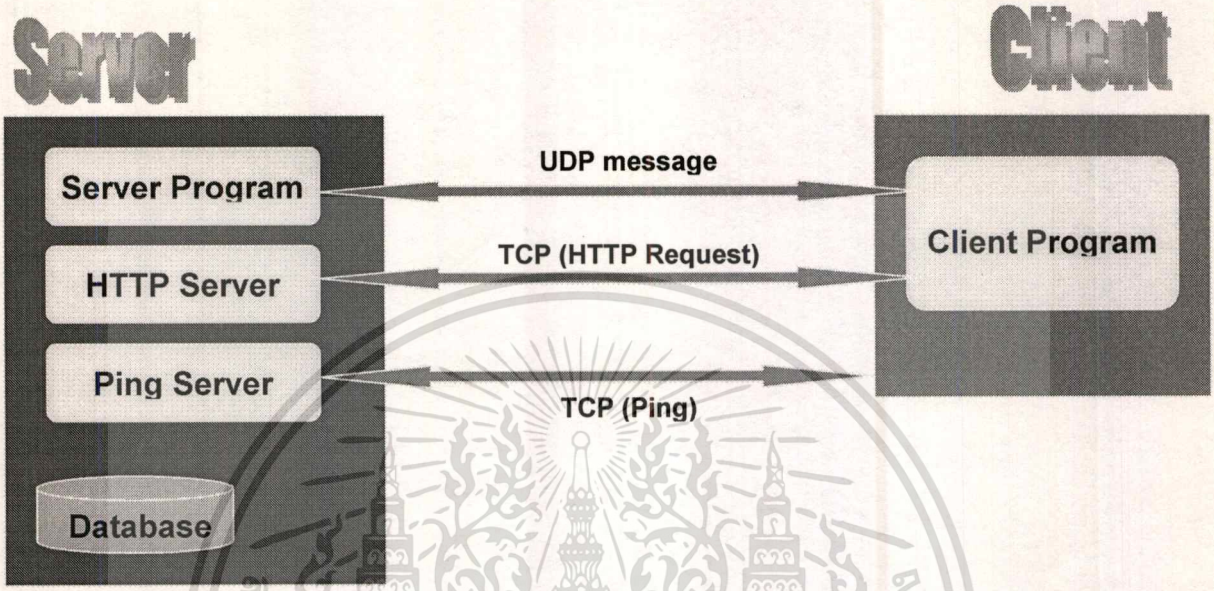
ในโปรแกรม Server มีการติดต่อกับเครื่อง Client โดยการ ping เพื่อตรวจสอบเครื่อง Client ในกรณีที่เครื่องถูกเปิดใช้งานอยู่ แต่โปรแกรม Client ไม่ได้ทำงานอยู่เพื่อให้ผู้ดูแลระบบทราบถึงการลักลอบใช้งาน

นอกจากที่กล่าวมา โปรแกรม Server ยังมีการทำงานเป็น Time Server ที่มีการส่งข้อมูลวันที่ และ เวลาปัจจุบันของเครื่อง Server ให้กับโปรแกรม Client ทุกๆเครื่องเพื่อ Syncronize เวลาให้ตรงกับเครื่อง Server เสมอ เพื่อป้องการลักลอบใช้งานเกินเวลาบนเครื่อง Client โดยการแก้ไขนาฬิกาของเครื่อง Clientให้ย้อนหลัง

การเก็บข้อมูลของโปรแกรม Server เก็บโดยใช้ Database เป็นหลัก โดยเก็บข้อมูลสถานะการทำงานของโปรแกรม Server, สถานะการทำงานแต่ละของ Client ว่าอยู่ในขั้นตอนใด เพื่อป้องกันกรณีที่เครื่อง Server Crash จะสามารถ restart ขึ้นมาใหม่ได้โดยที่สถานะการให้บริการยังคงอยู่ รวมไปถึงการเก็บข้อมูลของสมาชิกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม Server ทำงานติดต่อกับ โปรแกรม Client ผ่าน UDP โดยใช้ text string ที่กำหนดรูปแบบขึ้นโดยเฉพาะ โดยสามารถออกแบบภาพรวมของระบบได้ดังนี้



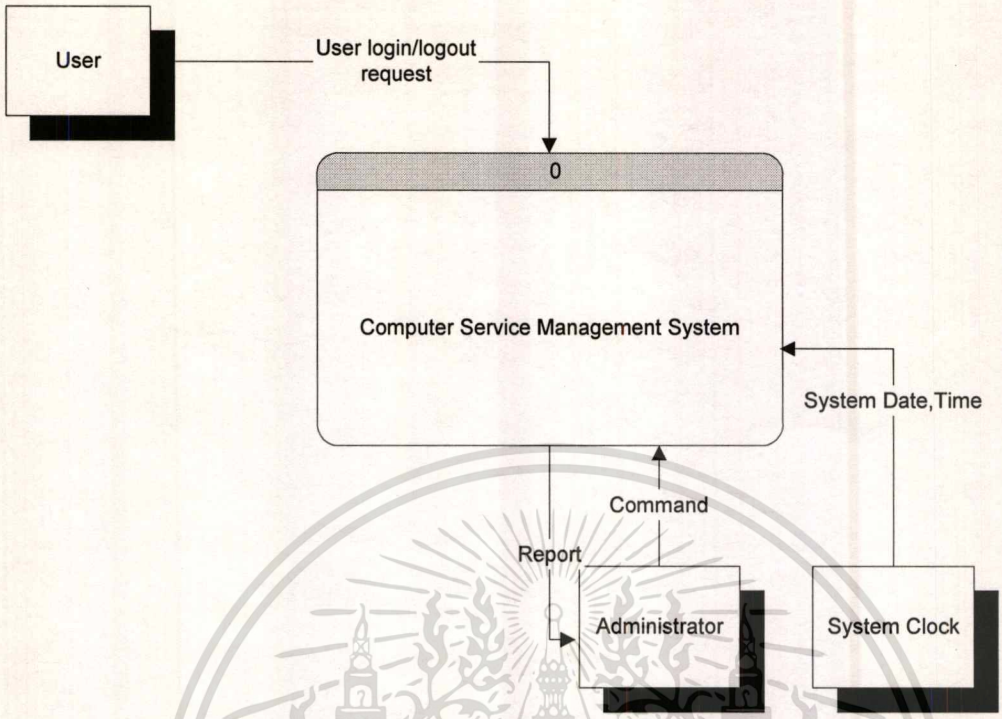
รูปที่ 3.1 แสดงโครงสร้างของระบบบริหารการให้บริการคอมพิวเตอร์

ในขั้นตอนของการออกแบบระบบการให้บริการเครื่องคอมพิวเตอร์นั้น วัตถุประสงค์ที่สำคัญก็คือการตรวจสอบสิทธิในการใช้งานของผู้ขอใช้เครื่องคอมพิวเตอร์ จึงต้องมีการป้องกันในแนวบังคับไม่ให้ผู้ที่ไม่ได้รับอนุญาติใช้งานเครื่องคอมพิวเตอร์ได้ ซึ่งเป็นหน้าที่ของโปรแกรม Client ที่ต้องมีการทำงานแบบ lock หน้าจอของเครื่องคอมพิวเตอร์ และป้องกันไม่ให้ผู้ใช้เข้าไปใช้งาน Windows หรือ Application อื่นๆได้โดยตรง โดยโปรแกรม Client ไม่ได้ทำงานอยู่

ฉะนั้นการออกแบบโครงสร้างของระบบจึงเน้นให้ความสำคัญไปที่ความมีเสถียรภาพและการรักษาความปลอดภัยของทั้งโปรแกรม Server และโปรแกรม Client เป็นอันดับแรก ซึ่งทำให้ต้องแลกมาด้วยประสิทธิภาพการทำงานที่อาจลดลง และมี overhead ที่สูงมากยิ่งขึ้น เช่น ในขั้นตอนของการติดต่อสื่อสารผ่านเครือข่ายจึงต้องมีการตรวจสอบอย่างรอบคอบ ซึ่งส่งผลให้ message ที่ติดต่อกันผ่านเครือข่ายมีขนาดใหญ่ขึ้นด้วย

จากภาพรวมของระบบสามารถนำออกมาเขียน Context Diagram ได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



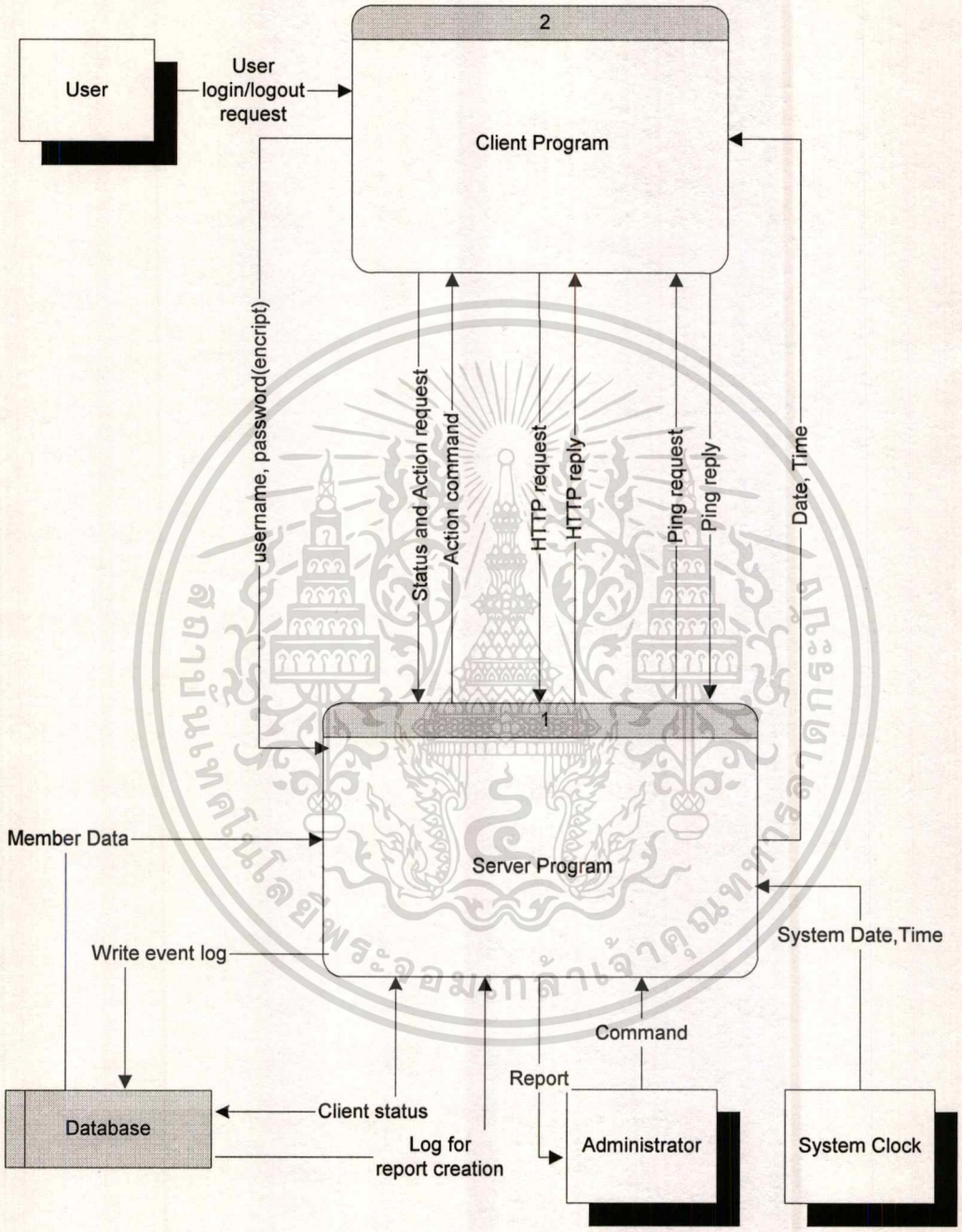
รูปที่ 3.2 แสดง Context Diagram ของระบบบริหารการให้บริการคอมพิวเตอร์

จากรูปที่ 3.2 สามารถอธิบายได้ว่าระบบบริหารให้บริการคอมพิวเตอร์นั้นติดต่อรับข้อมูลจากภายนอกระบบสองแหล่งด้วยกันคือ

- User ผู้ใช้ที่ร้องขอการใช้บริการ ทำการติดต่อโดยใส่ username และ password ที่เครื่อง Client ที่ต้องการใช้
- Administrator ผู้ดูแลระบบทั้งรับและส่งข้อมูลกับระบบ เนื่องจากมีหน้าที่ในการควบคุมการทำงาน ข้อมูลที่ส่งเข้าระบบ เช่น การกำหนดค่าเริ่มต้นของระบบ คำสั่งการให้อนุญาตผู้ใช้ให้ใช้งานหรือหยุดใช้งานเครื่อง เป็นต้น และการรับข้อมูลจากระบบ เช่น การร้องขอการใช้งานจากผู้ใช้ผ่านระบบ และการที่ระบบแจ้งเตือนแบบต่างๆ

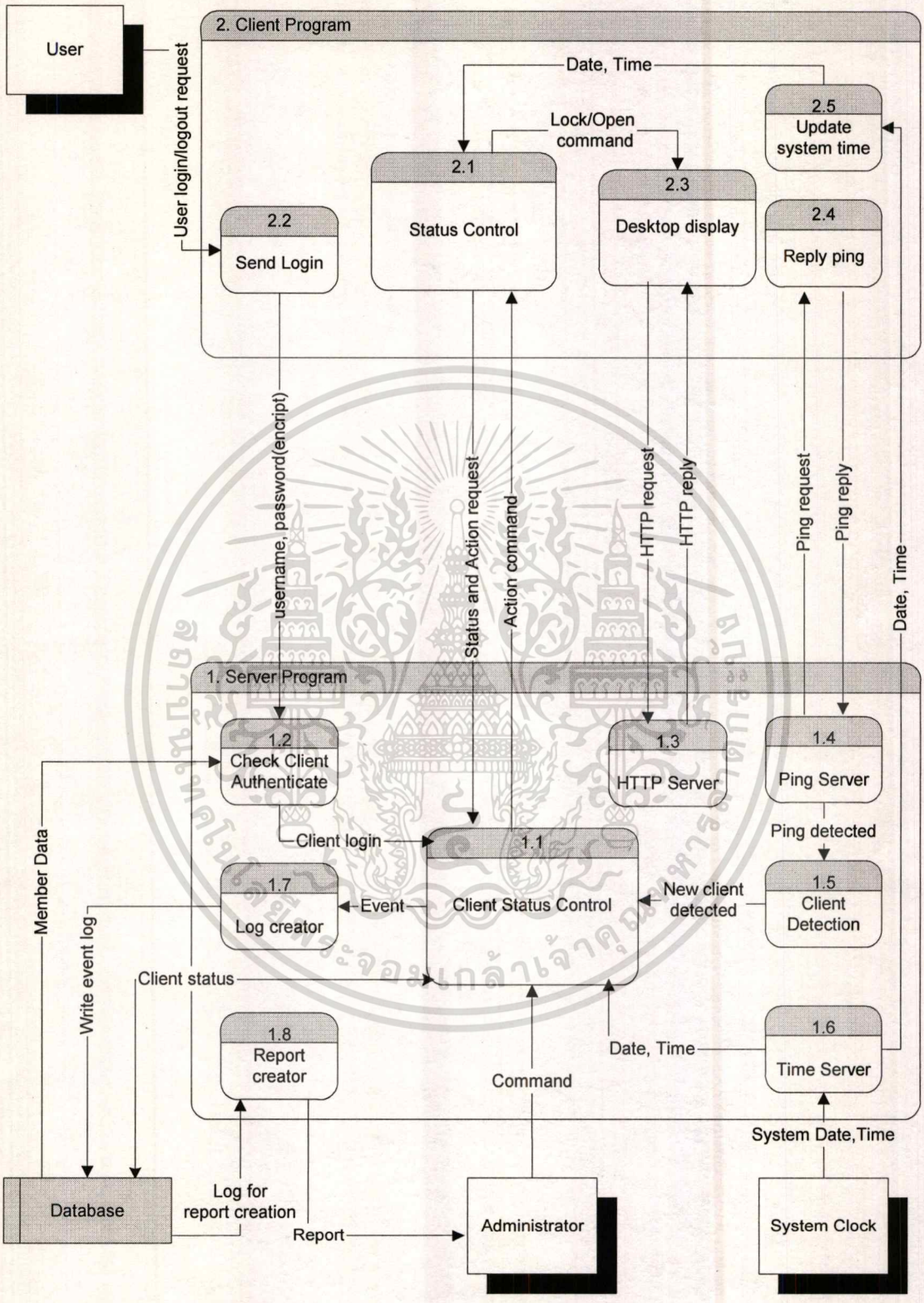
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถเขียนเป็น Data Flow Diagram (DFD level 1) ได้ดังนี้



รูปที่ 3.3 แสดง DFD level 1 ของระบบบริหารการให้บริการคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

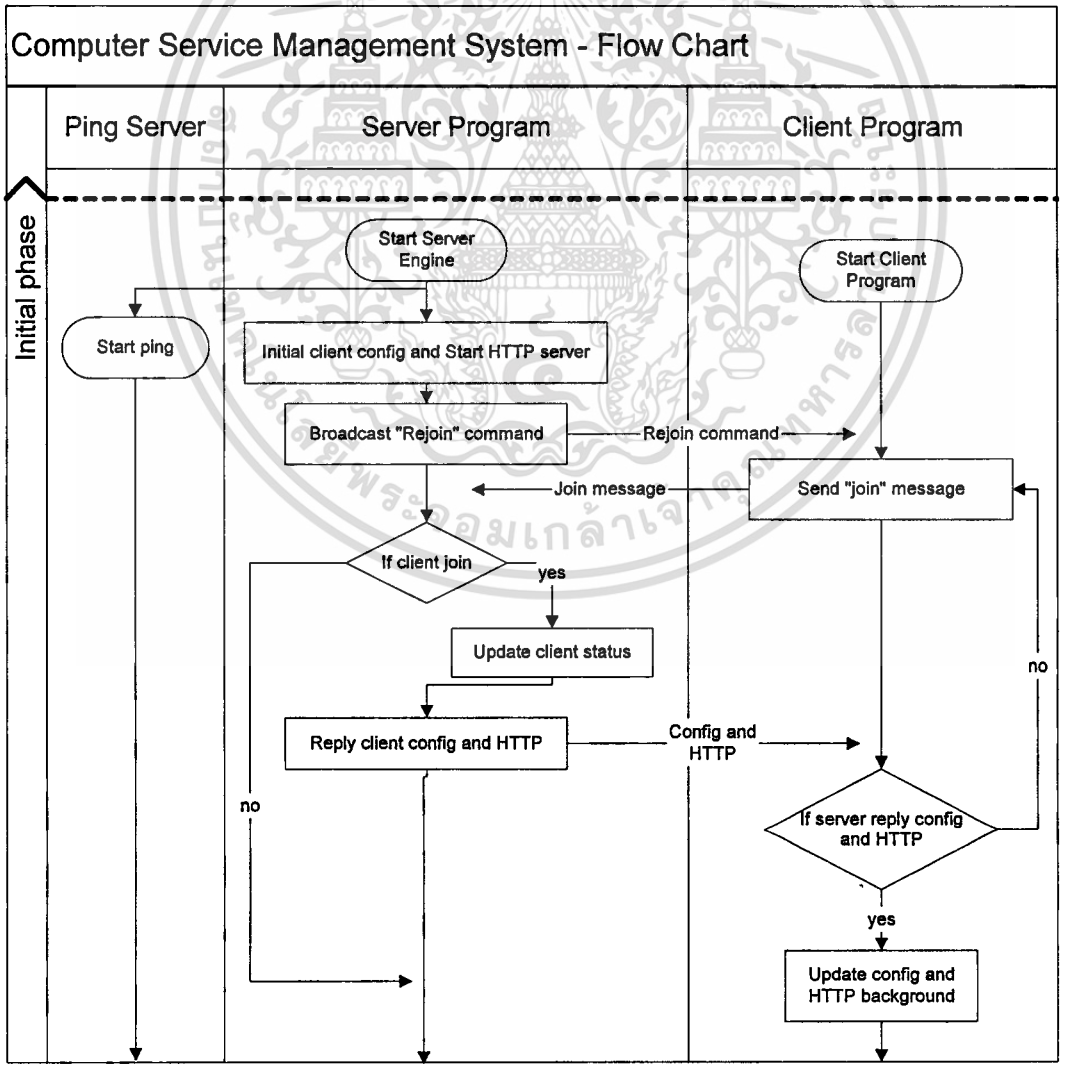


รูปที่ 3.4 แสดง DFD level 2 ของระบบบริหารการให้บริการคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

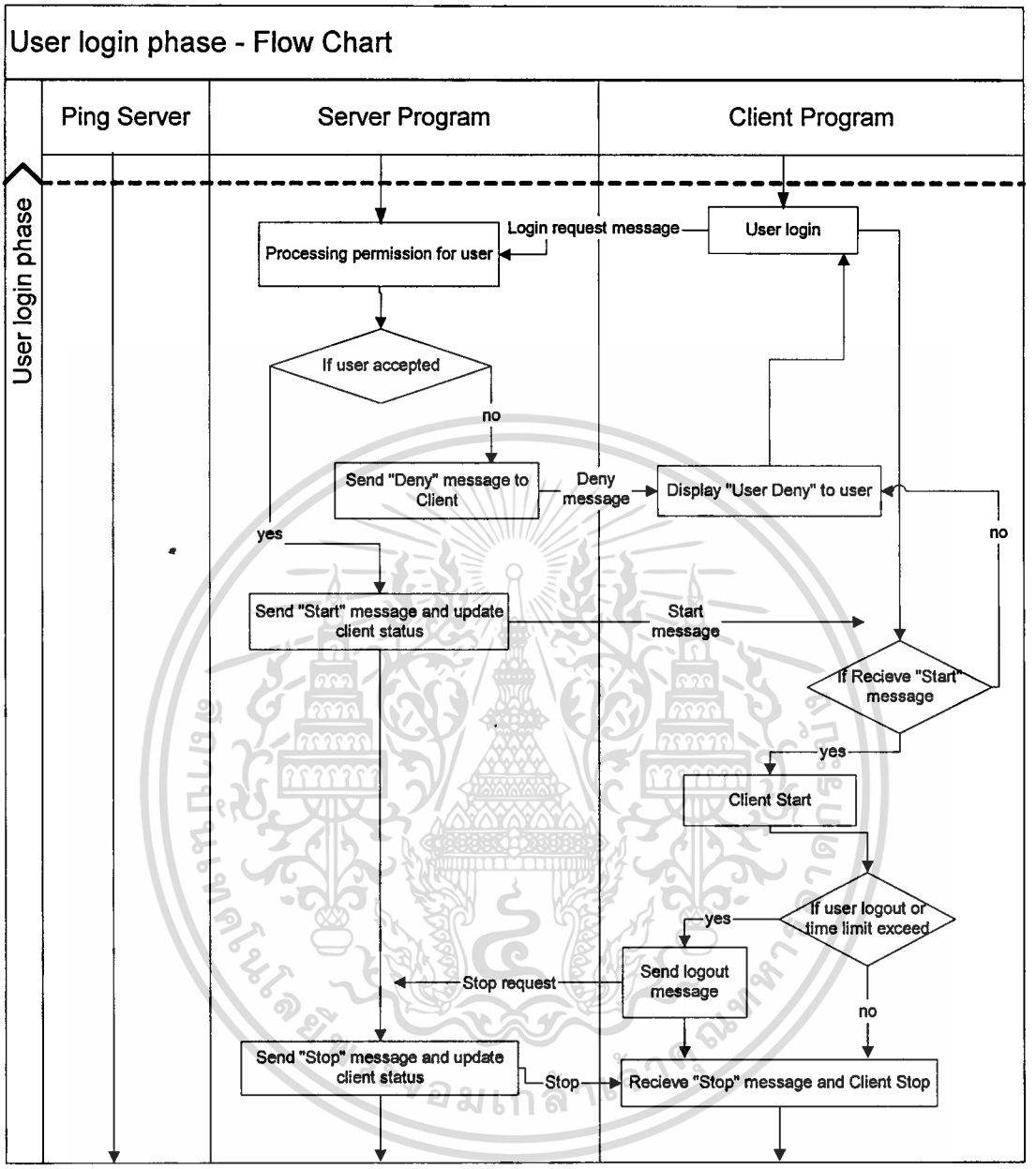
การอธิบายลำดับขั้นตอนการทำงานของระบบได้โดยแบ่งการทำงานออกเป็น Phase ต่างๆ ได้แก่

- Initial phase คือขั้นตอนเริ่มต้นเมื่อระบบเริ่มเปิดใช้งาน
- User Login phase ขั้นตอนปกติในการทำงานที่ User ขอเข้าใช้บริการผ่านทางหน้าจอของ Client Program
- Client Detection phase ขั้นตอนที่ตรวจสอบหา Client ที่เพิ่งเข้ามาสู่ระบบ โดยขั้นตอนนี้จะทำงานอยู่ตลอดเวลาหลังจาก Initial phase เสร็จ
- Time Synchronization phase ขั้นตอนในการปรับเวลาให้เครื่อง Client ทุกๆเครื่องมีนาฬิกาตรงกันกันเครื่อง Server โดยขั้นตอนนี้จะทำงานอยู่ตลอดเวลาหลังจาก Initial phase เสร็จ
- End phase ขั้นตอนการปิดระบบ



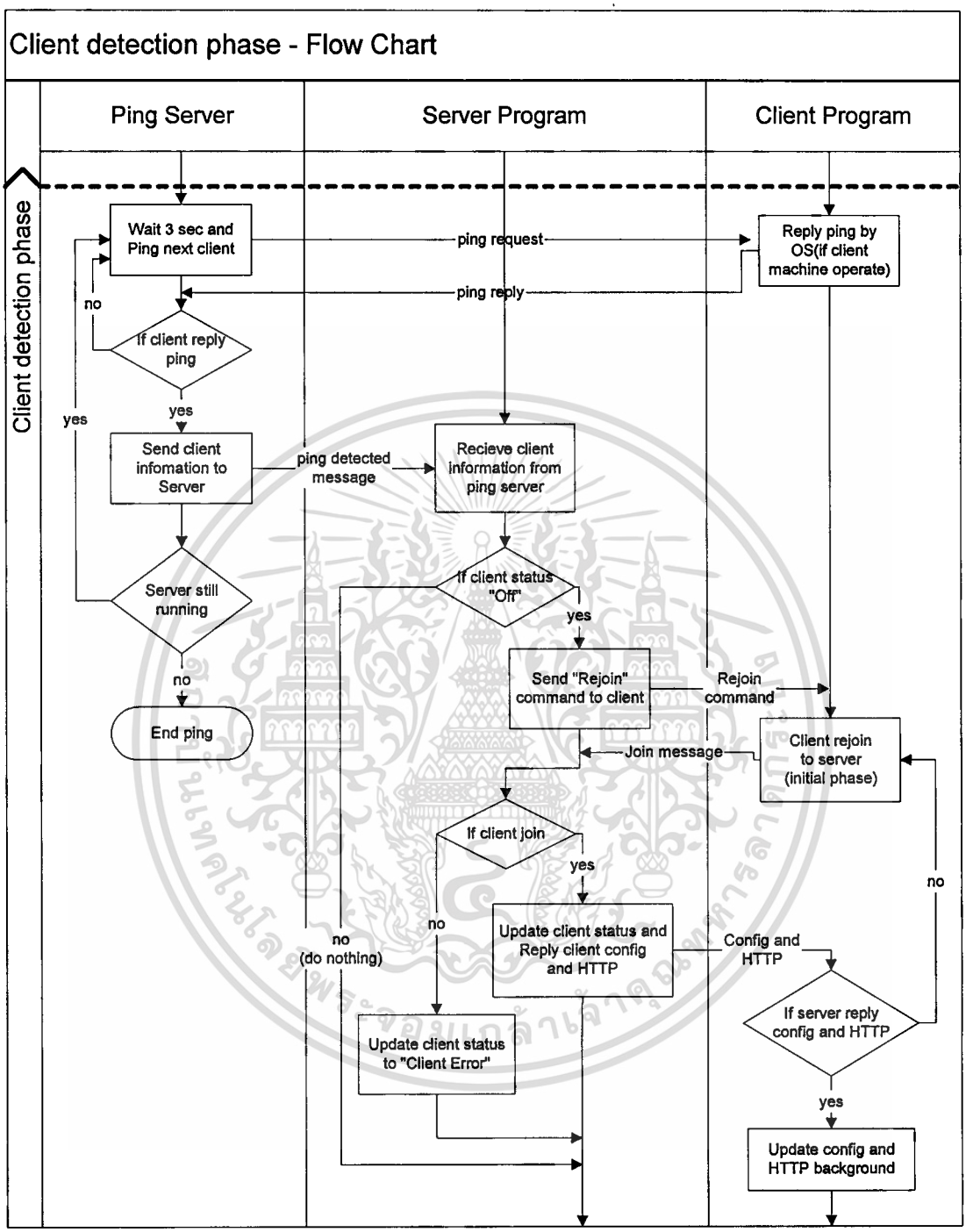
รูปที่ 3.5 แสดง Flow Chart ของระบบบริหารการให้บริการคอมพิวเตอร์ในขั้นตอน Initial phase

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



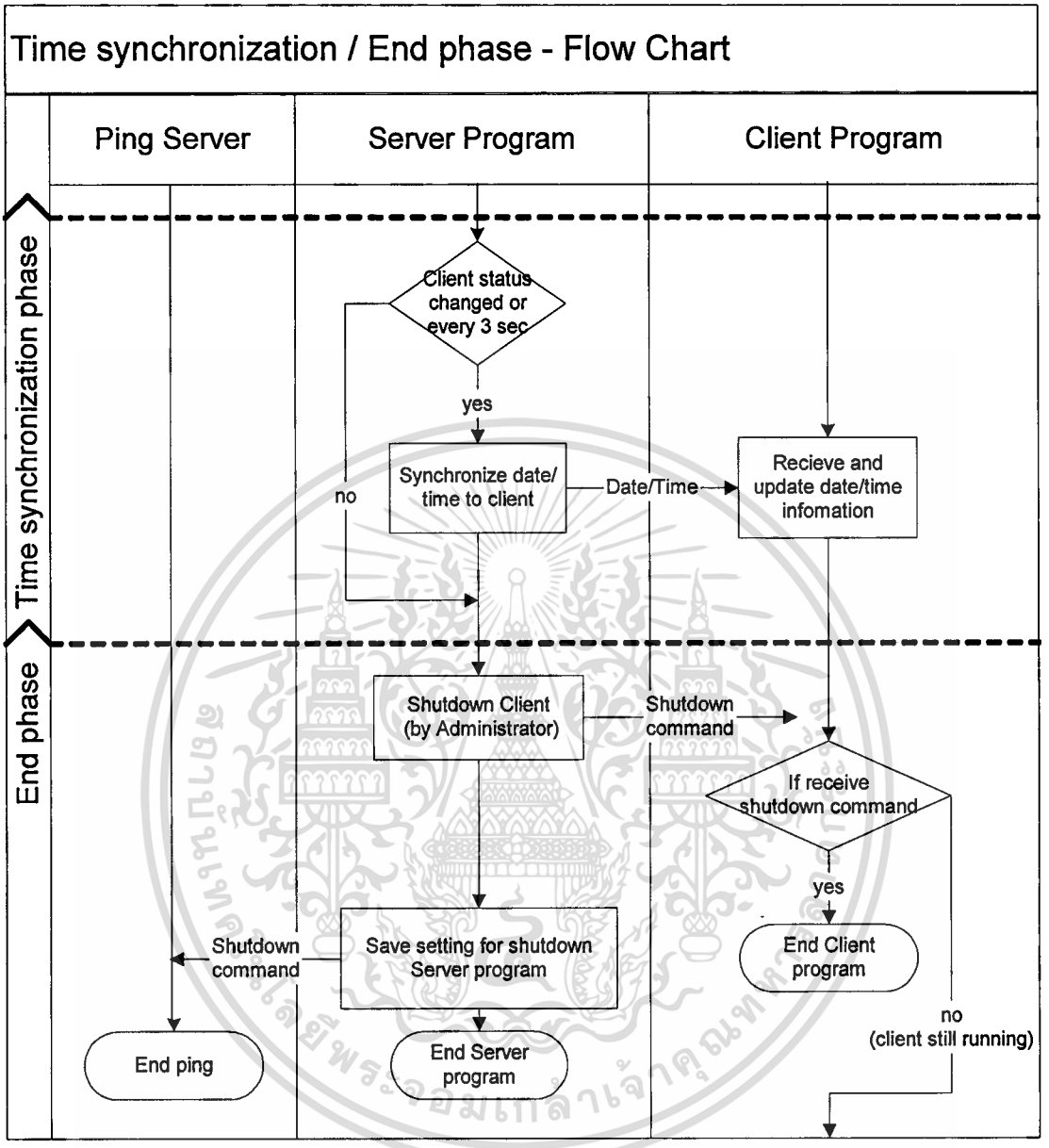
รูปที่ 3.6 แสดง Flow Chart ของระบบบริหารการให้บริการคอมพิวเตอร์ในขั้นตอน User login phase

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 แสดง Flow Chart ของระบบบริหารการให้บริการคอมพิวเตอร์ในขั้นตอน Client detection phase

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 แสดง Flow Chart ของระบบบริหารการให้บริการคอมพิวเตอร์ในขั้นตอน Time sychronize/End phase phase

3.2 การออกแบบการทำงานของ Client

หน้าที่หลักของโปรแกรม Client ได้แก่

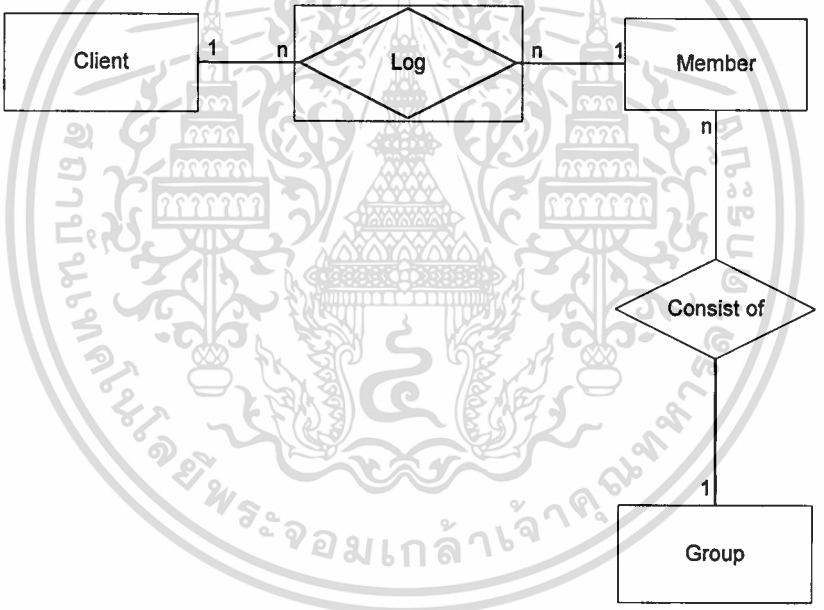
- Lock หน้าจอ ป้องกันไม่ให้ผู้ใช้เรียกใช้งาน Application อื่นๆ จนกว่าจะได้รับอนุญาต
- ป้องกันไม่ให้ผู้ใช้สามารถปิดการทำงานของตัวโปรแกรม Client เองได้
- แสดงวิธีการเข้าใช้ระบบ และข้อมูลประชาสำพันธ์บนหน้าจอเครื่องให้บริการในระหว่างที่มีการ lock หน้าจออยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากหน้าที่การทำงานของระบบจะเห็นได้ว่าสิ่งที่สำคัญมากอย่างหนึ่งก็คือการรักษาความปลอดภัย เพื่อมิให้ผู้ที่ไม่ได้รับอนุญาตสามารถใช้งานเครื่องได้

3.3 การออกแบบฐานข้อมูล

จากการออกแบบโปรแกรม Server ที่มีหน้าที่เก็บรวบรวมข้อมูลสถานะการทำงานของเครื่อง Client ที่ให้บริการทั้งหมด เก็บข้อมูลผู้ใช้ และกลุ่มของผู้ใช้ โดยเวลาการใช้บริการที่ผู้ใช้สามารถใช้งานได้นั้นขึ้นอยู่กับสิทธิที่กลุ่มได้รับ สามารถออกแบบการจัดเก็บข้อมูลในฐานข้อมูลโดยการเขียน Entity Relationship Diagram (ER Diagram) ได้ดังนี้



รูปที่ 3.9 แสดง ER Diagram ของระบบการให้บริการคอมพิวเตอร์

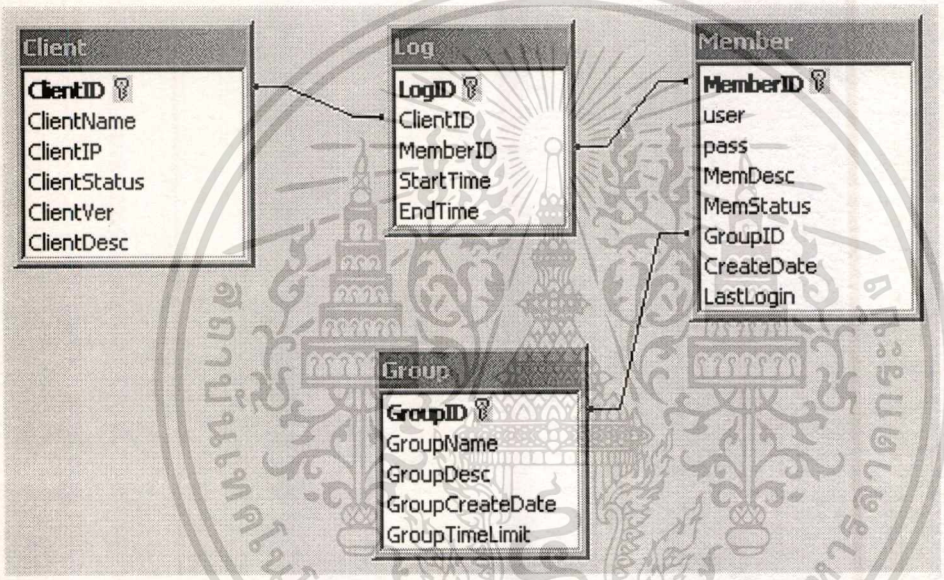
จาก Entity Relationship Diagram ดังกล่าวสามารถอธิบายได้ว่า Client ของระบบการให้บริการคอมพิวเตอร์นั้น คือเครื่องคอมพิวเตอร์ที่ Server Program รู้จักและนำมาให้บริการ โดยจะเก็บสถานะต่างๆของเครื่องคอมพิวเตอร์ โดยเก็บข้อมูล IP address และ Software Version ซึ่งหมายความว่า Client ที่รู้จักและให้บริการนั้น สามารถเปลี่ยนไปใช้เครื่องคอมพิวเตอร์เครื่องอื่นมาทดแทนได้ ไม่ผูกติดกับเครื่องคอมพิวเตอร์ โดยใช้ Client Program ที่ถูกต้อง และกำหนด IP address ที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Server Program รู้จัก ซึ่งประโยชน์ที่ได้รับคือความสะดวกในกรให้บริการที่สามารถนำเครื่องคอมพิวเตอร์มาเปลี่ยนใช้แทนกันได้ในกรณีที่เครื่องเดิมเสีย หรือ การ Upgrade เครื่องคอมพิวเตอร์

สำหรับการใช้งาน Member หนึ่งคนสามารถใช้งานได้ครั้งละ 1 เครื่อง Client และเครื่อง Client ก็สามารถถูกใช้งานได้ครั้งละ 1 Member โดยใน Group คือกลุ่มของ Member ที่สามารถจำกัดเวลาการใช้งานของ Member ที่อยู่ในกลุ่มได้

จาก Entity Relationship สามารถนำมาออกแบบความสัมพันธ์ของการจัดเก็บข้อมูลในฐานข้อมูลได้ดังนี้



รูปที่ 3.10 แสดง ความสัมพันธ์ของการจัดเก็บข้อมูลในฐานข้อมูล

โดยสามารถแสดงรายละเอียดของตารางการจัดเก็บข้อมูลได้ดังนี้

ตารางที่ 3.1 ตาราง Client เก็บรายละเอียดของเครื่องคอมพิวเตอร์ที่ให้บริการ

Client Table - Field Name	Field Type	Field Length	Description
ClientID - **PK**	Text	10	Client ID
ClientName	Text	20	Client Computer Name
ClientIP	Text	15	Client IP Address
ClientStatus	Text	5	Client Status
ClientVersion	Text	10	Version of Client program
ClientDesc	Text	50	Client Description

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.2 ตาราง Member เก็บรายละเอียดของสมาชิก

Member Table - Field Name	Field Type	Field Length	Description
MemberID - **PK**	Text	10	Member ID
user	Text	20	Username
pass	Text	20	Password
MemDesc	Text	50	Member Description
MemStatus	Text	5	Member Status
GroupID	Text	10	Group ID
CreateDate	Date/Time	10	Member Create Date
LastLogin	Date/Time	10	Member Last Login

ตารางที่ 3.3 ตาราง Group เก็บรายละเอียดกลุ่มของสมาชิก

Group Table - Field Name	Field Type	Field Length	Description
GroupID - **PK**	Text	10	Group ID
GroupName	Text	20	Group Name
GroupDesc	Text	50	Group Description
GroupCreateDate	Date/Time	10	Group CreateDate
GroupTimeLimit	Integer	10	Group Time Limit (minute)

ตารางที่ 3.4 ตาราง Log เก็บรายละเอียดการใช้บริการเครื่องคอมพิวเตอร์ของสมาชิก

Log Table - Field Name	Field Type	Field Length	Description
LogID - **PK**	Text	10	Log ID
ClientID	Text	10	Client ID
MemberID	Text	10	Member ID
StartTime	Date/Time	10	Start Time
EndTime	Date/Time	10	End Time

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การออกแบบการติดต่อสื่อสารระหว่าง Server Program, Client Program และ Ping Server

ในขั้นตอนการติดต่อสื่อสารระหว่าง Server Program กับ Client Program ใช้ Message ที่กำหนดขึ้นเอง โดยใช้ UDP (User Datagram Protocol)

โดยสามารถแสดงรายละเอียดของ Message ที่ใช้ติดต่อสื่อสารกัน โดยกำหนดให้ตัวอย่าง IP Address 10.66.11.1 เป็นของเครื่อง Server และ 10.66.11.222 เป็นของเครื่อง Client

ตารางที่ 3.5 แสดงรายละเอียดของ Message SVR_REJOIN

SVR_REJOIN	
from	Server Program
To	All Client (Initial phase) / Target Client (Client detection phase)
Protocol	UDP
Use Phase	Initial phase, Client detection phase
Description	ใช้เมื่อเริ่ม Start Server Program หรือเมื่อต้องการให้ทุกๆ Client ตอบรับ
Parameter	Time : Server IP: **SVR_REJOIN** : All Client
Example	14:27:07 : 10.66.11.1 : SVR_REJOIN , IP_ALL

ตารางที่ 3.6 แสดงรายละเอียดของ Message CLI_JOIN

CLI_JOIN	
from	Client Program
To	Server Program
Protocol	UDP
Use Phase	Initial phase, Client detection phase
Description	Client ใช้ในการ Join เข้าหา Server
Parameter	Time : Client IP : **CLI_JOIN** , Client Name
Example	14:29:14 : 10.66.11.222 : CLI_JOIN , Adisak

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.7 แสดงรายละเอียดของ Message SVR_CONFIG

SVR_CONFIG	
from	Server Program
To	Target Client
Protocol	UDP
Use Phase	Initial phase
Description	Server Program ใช้ส่งค่าเริ่มต้นต่างๆให้กับ Client Program หลังจาก Client ได้ส่ง Message Join กลับมา ในขั้นตอน Initial phase ซึ่งค่าต่างๆที่ส่งนั้นประกอบไปด้วย Server IP เพื่อให้ Client Program ทราบว่าตนถูกลงทะเบียนไว้กับ Server เครื่องใด (ในกรณีที่มีการทำงานของ Server Program หลายเครื่อง เช่นการแบ่งกลุ่มการให้บริการมากกว่าหนึ่งกลุ่มบนเครือข่ายเดียวกัน) HTTP Path เพื่อให้ Client Program ทราบว่าจะสามารถเรียกใช้เอกสาร HTTP จากที่ใดมาเป็น Background ในการ Lock หน้าจอเครื่อง Client ซึ่ง HTTP Path นี้สามารถกำหนดเป็นที่อยู่บน Web Server ของ Server Program เอง หรือ Web Server ที่ให้บริการบนเครื่องอื่นๆที่ Client สามารถเข้าถึง รวมทั้งเอกสาร Web Page ที่ให้บริการอยู่บน Internet ในขณะนั้นได้
Parameter	Time : Server IP: **SVR_CONFIG** , Target Client , HTTP Path
Example	14:27:07 : 10.66.11.1 : SVR_CONFIG , 10.66.11.222 , http://server/index.html

ตารางที่ 3.9 แสดงรายละเอียดของ Message CLI_LOGIN

CLI_LOGIN	
From	Client Program
To	Server Program
Protocol	UDP
Use Phase	User Login phase
Description	Client ใช้ในการ Login โดย Member ที่มี Username และ Password ถูกต้อง เพื่อขอใช้บริการเครื่อง
Parameter	Time : Client IP : **CLI_LOGIN** , Server IP , Member Username , Password
Example	14:29:14 : 10.66.11.222 : CLI_LOGIN , 10.66.11.1 , user123 , pass123

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.10 แสดงรายละเอียดของ Message SVR_DENY

SVR_DENY	
From	Server Program
To	Client Program
Protocol	UDP
Use Phase	User Login phase
Description	Server Program ใช้ตอบปฏิเสธกลับการขอ Login ของ Client Program ในกรณีที่ Username หรือ Password ไม่ถูกต้อง
Parameter	Time : Server IP : **SVR_DENY** , Client IP , Error Message
Example	14:29:14 : 10.66.11.1 : SVR_DENY , 10.66.11.222 , Invalid login (Username or Password) Try again

ตารางที่ 3.11 แสดงรายละเอียดของ Message SVR_START

SVR_START	
From	Server Program
To	Client Program
Protocol	UDP
Use Phase	User Login phase
Description	Server Program ใช้ตอบยอมรับการขอ Login ของ Client Program โดยส่ง Start Message นี้เพื่อสั่งให้ Client Program Unlock หน้าจอและยอมให้ผู้ใช้สามารถใช้งานเครื่องคอมพิวเตอร์นั้นได้ ในแง่ของ Client Program จะมีเพียงทางเดียวเท่านั้นที่ จะ Unlock หน้าจอของเครื่องคอมพิวเตอร์ คือการที่ Client Program ได้รับ Message SVR_START นี้
Parameter	Time : Server IP : **SVR_START** , Client IP , Start Time , Time Limit (ถ้ามี จะถูกกำหนดอยู่ในตารางข้อมูลของ Group ของ Member ที่ Login) หน่วยเป็นนาที
Example	14:35:14 : 10.66.11.1 : SVR_START , 10.66.11.222 , 14:35 , 7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.12 แสดงรายละเอียดของ Message CLI_STOP

CLI_STOP	
From	Client Program
To	Server Program
Protocol	UDP
Use Phase	User Login phase
Description	Client Program ใช้ Message CLI_STOP นี้เพื่อเป็นการขออนุญาตจาก Server Program ในการหยุดให้บริการ และทำการ Lock หน้าจอเครื่อง (ซึ่ง Client Program ต้องรอ Message SVR_STOP ตอบกลับจาก Server Program เท่านั้น) โดยการขอหยุดให้บริการนี้เกิดจาก 2 กรณีคือ ผู้ใช้กดปุ่ม Logout ที่ Client Program หรือ ครบตามเวลาที่กำหนดตามที่ส่งมากับ SVR_START Message โดย Client Program ตรวจสอบการครบกำหนดเวลาการใช้งานนี้ก่อน แล้วจึงขอหยุดให้บริการ ที่เป็นเช่นนี้เพื่อเป็นการลดภาระการตรวจสอบเวลาที่ครบกำหนดจาก Server program ให้มาอยู่ที่ Client Program แทน
Parameter	Time : Client IP : **CLI_STOP** , Server IP
Example	14:35:14 : 10.66.11.222 : CLI_STOP , 10.66.11.1

ตารางที่ 3.13 แสดงรายละเอียดของ Message SVR_STOP

SVR_STOP	
From	Server Program
To	Client Program
Protocol	UDP
Use Phase	User Login phase
Description	Server Program ใช้ตอบยอมรับการขอ หยุดให้บริการของ Client Program อาจเกิดต่อเนื่องจากการที่ Server Program ได้รับ Message CLI_STOP ขอหยุดให้บริการจาก Client Program หรือ การที่ Administrator สั่งหยุดการให้บริการจาก Server Program ก็ได้
Parameter	Time : Server IP : **SVR_STOP** , Client IP
Example	14:35:14 : 10.66.11.1 : SVR_STOP , 10.66.11.222

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.14 แสดงรายละเอียดของ Message PING Request/Reply

PING Request/Reply	
From	Ping Server
To	Client Machine
Protocol	TCP
Use Phase	Client detection phase
Description	Ping Server ทำการ Ping ไปยังเครื่อง Client ทุกๆเครื่องที่ลงทะเบียนไว้ โดยมีการหน่วงเวลาระหว่างการวน Ping ทุกๆเครื่อง ซึ่งถ้าได้รับ Ping Reply จากเครื่อง Client ใด ก็หมายความว่าเครื่อง Client นั้นเปิดใช้งานอยู่ Ping Server จะทำการส่ง Message บอกยัง Server Program เพื่อให้ตรวจสอบเครื่อง Client นั้นต่อไป
Parameter	Ping ClientIP
Example	Ping 10.66.11.222

ตารางที่ 3.15 แสดงรายละเอียดของ Message PING_DETECT

PING_DETECT	
From	Ping Server
To	Server Program
Protocol	TCP
Use Phase	Client detection phase
Description	เมื่อ Ping Server ได้รับ Ping Reply จาก Client เครื่องใด Ping Server จะส่ง PING_DETECT Message ไปบอกยัง Server Program ว่าเครื่อง Client นั้นเปิดใช้งานอยู่ เพื่อที่ Server Program จะได้ทำการตรวจสอบต่อไปว่าเครื่อง Client นั้นมีการทำงานของ Client Program อยู่หรือไม่ ถ้าไม่มี Server Program จะสรุปได้ต่อไปว่าเครื่อง Client นั้นกำลังถูกลักลอบใช้งาน
Parameter	Time : Ping Server IP : **PING_DETECT** , Server IP , Client IP (ที่ตอบ Ping Reply กลับมา)
Example	14:35:14 : 10.66.11.1 : PING_DETECT , 10.66.11.1 , 10.66.11.222

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.16 แสดงรายละเอียดของ Message SVR_DATETIME

SVR_DATETIME	
from	Server Program
To	All Client Program
Protocol	UDP
Use Phase	Time synchonyzation phase
Description	Server Program ใช้ส่งค่า Date, Time ปัจจุบันของเครื่อง Server ให้กับ Client Program ทุกๆเครื่อง เพื่อทำการ Update ให้ Date, Time ตรงกันกับเครื่อง Server เสมอ โดยจะมีการหน่วงเวลา แล้วส่ง SVR_DATETIME นี้ให้ Client ทุกเครื่องเสมอๆ เพื่อป้องกันการทำงานผิดพลาดและการลักลอบใช้งานเกินเวลา
Parameter	Time : Server IP: **SVR_DATETIME** , All Client , Date Time
Example	14:28:07 : 10.66.11.1 : SVR_DATETIME , IP_ALL , 18/02/2546 14:28:07

ตารางที่ 3.17 แสดงรายละเอียดของ Message SVR_ENDPING

SVR_ENDPING	
From	Server Program
To	Ping Server
Protocol	TCP
Use Phase	End phase
Description	Server Program ใช้ Message SVR_ENDPING หลังจาก Administrator สั่งปิด Server Program เพื่อบอกให้ Ping Server ปิดเช่นเดียวกัน
Parameter	Time : Server IP: **SVR_ENDPING** , Ping Server IP
Example	14:28:07 : 10.66.11.1 : SVR_ENDPING , 10.66.11.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การพัฒนาระบบบริหารการให้บริการคอมพิวเตอร์

ในขั้นตอนการพัฒนานั้น ระบบบริหารการให้บริการคอมพิวเตอร์นั้น มีการพัฒนาส่วนต่างๆของโปรแกรม โดยใช้เครื่องมือในการพัฒนาดังนี้

1. เครื่องคอมพิวเตอร์ PC ติดตั้งระบบปฏิบัติการ Windows 2000/XP Professional เพื่อใช้ทดสอบส่วน Server Program และ Ping Server
2. เครื่องคอมพิวเตอร์ PC ติดตั้งระบบปฏิบัติการ Windows ME เพื่อใช้ทดสอบส่วน Client Program
3. เครือข่าย LAN ที่สามารถใช้งาน Protocol TCP และ UDP
4. Microsoft Visual Basic 6.0 และ Component ต่างๆที่จำเป็น
5. Microsoft Access 2000 สำหรับออกแบบฐานข้อมูลและใช้เป็น RDBMS ของระบบ
6. Microsoft Word 2000 และ Visio 2000 สำหรับการจัดทำเอกสารประกอบการพัฒนาระบบ
7. Microsoft PowerPoint 2000 ในการจัดทำและนำเสนอ Presentation

โดยระบบบริหารการให้บริการคอมพิวเตอร์นี้ประกอบไปด้วย 2 ส่วนการทำงานหลักคือ Server Program และ Client Program

4.1 การพัฒนาส่วนการทำงานของ Server Program

หน้าที่การทำงานส่วนใหญ่ของระบบจะทำงานที่ส่วน Server Program นี้โดยมีหน้าที่หลักก็คือการตรวจสอบสิทธิในการใช้งานของผู้ขอใช้เครื่องคอมพิวเตอร์ จึงต้องมีการป้องกันในแนวบังคับไม่ให้ผู้ที่ไม่ได้รับอนุญาติใช้งานเครื่องคอมพิวเตอร์ได้ ซึ่งเป็นหน้าที่ของโปรแกรม Client ที่ต้องมีการทำงานแบบ lock หน้าจอของเครื่องคอมพิวเตอร์ และป้องกันไม่ให้ผู้ใช้เข้าไปใช้งาน Windows หรือ Application อื่นๆได้โดยตรง โดยโปรแกรม Client ไม่ได้ทำงานอยู่

ฉะนั้นการออกแบบโครงสร้างของระบบจึงเน้นให้ความสำคัญไปที่ความมีเสถียรภาพและการรักษาความปลอดภัยของทั้งโปรแกรม Server และโปรแกรม Client เป็นอันดับแรก ซึ่งทำให้ต้องแลกมาด้วยโครงสร้างของระบบที่ซับซ้อนขึ้น ประสิทธิภาพการทำงานที่อาจลดลง และมี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

overhead ที่สูงมากยิ่งขึ้น เช่น ในขั้นตอนของการติดต่อสื่อสารผ่านเครือข่ายจึงต้องมีการตรวจสอบอย่างรอบคอบ ซึ่งส่งผลให้ message ที่ติดต่อกันผ่านเครือข่ายมีขนาดใหญ่ขึ้นด้วย

ส่วนการทำงานต่างๆที่ช่วยเพิ่มความสามารถของระบบและเพิ่มประสิทธิภาพในการรักษาความปลอดภัยได้แก่

HTTP Server

เป็น Web Server ที่ให้บริการ Web Page เพื่อให้ Client Program นำหน้า Web Page นี้ไปเป็น Background ของเครื่อง Client ซึ่งมีประโยชน์คือ

- ผู้ดูแลระบบสามารถเปลี่ยนแปลงรูปแบบหน้าจอของเครื่อง Client ได้โดยง่าย เพื่อความสวยงาม และสามารถแสดงข้อมูลที่ต้องการประชาสัมพันธ์ หรือกฎระเบียบวิธีใช้บริการ ซึ่งสามารถเปลี่ยนแปลงแก้ไขได้สะดวก

- เมื่อมีการใช้ Client จำนวนมากขึ้น ระบบสามารถบอก Client ให้ใช้บริการจาก Web Server เครื่องอื่นได้ เพื่อลดภาระงานของเครื่อง Server

Ping Server

คือส่วนการทำงานที่เพิ่มขึ้นเพื่อให้ระบบมีความรัดกุมในเรื่องการรักษาความปลอดภัยมากขึ้น โดยหน้าที่หลักของ Ping Server คือการตรวจหาเครื่องคอมพิวเตอร์ที่ลงทะเบียนไว้ว่าเป็น Client ของระบบ ว่ามีการเปิดเครื่องใช้งานหรือไม่ ถ้าเปิด Ping Server จะส่ง message บอกไปยัง Server Program ให้ตรวจสอบต่อไปว่าเครื่องคอมพิวเตอร์ที่เปิดใช้งานอยู่นั้นได้มีการควบคุมโดย Client Program หรือไม่ หากไม่มีส่วนการทำงานนี้อยู่ เครื่องคอมพิวเตอร์อาจถูกลักลอบใช้งานโดยที่ระบบไม่รู้เลยถ้า Client program บนเครื่องคอมพิวเตอร์นั้นไม่ได้ทำงานอยู่ อาจเนื่องจากการ Boot เครื่องที่ไม่สมบูรณ์ หรือความพยายามในการปิด Client Program ของผู้ใช้ ซึ่งอาจเกิดขึ้นได้ ในกรณีเช่นนี้ Ping Server ก็จะสามารถเตือนผู้ดูแลระบบได้เพื่อให้ผู้ดูแลระบบทราบและตรวจสอบต่อไป

Time Server

ในส่วนการทำงานนี้ Time Server จะทำการ Synchronize วันที่และเวลาให้เครื่อง Client ทุกๆเครื่องมีวันที่ละเวลาตรงกันกับเครื่อง Server เสมอ เพื่อให้การติดต่อกันของ Server Program และ Client Program มีความหมายถูกต้องเนื่องจากมีนาฬิกาที่ตรงกันเสมอ โดยมีประโยชน์สำคัญ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อีกอย่างหนึ่งคือเพื่อป้องกันการลักลอบใช้งานเครื่อง Client เกินเวลา โดยการปรับนาฬิกาที่เครื่อง Client ให้ถอยหลัง

หากย้อนกลับไปมองว่าเหตุใดจึงต้องมีการ Synchronizeเวลากันนั้น ก็เพราะเนื่องมากจากการที่ผู้ใช้บริการใช้งานเครื่องได้ในแบบจำกัดเวลานั้น Client Program จะเป็นผู้ตรวจสอบเวลาเองไม่ใช้ที่ Server Program และสาเหตุที่เป็นเช่นนี้ก็เนื่องจากความพยายามในการแบ่งภาระงานออกจาก Server Program ให้มากที่สุด เพราะถ้าหากมีการใช้เครื่อง Client ในระบบเป็นจำนวนมากแล้วหากตรวจสอบเวลาที่ครบกำหนดให้บริการที่ Server Program จะทำให้ Server ภาระงานที่สูงตามความละเอียดของหน่วยเวลาที่ใช้ในการตรวจสอบ ยกตัวอย่างเช่น

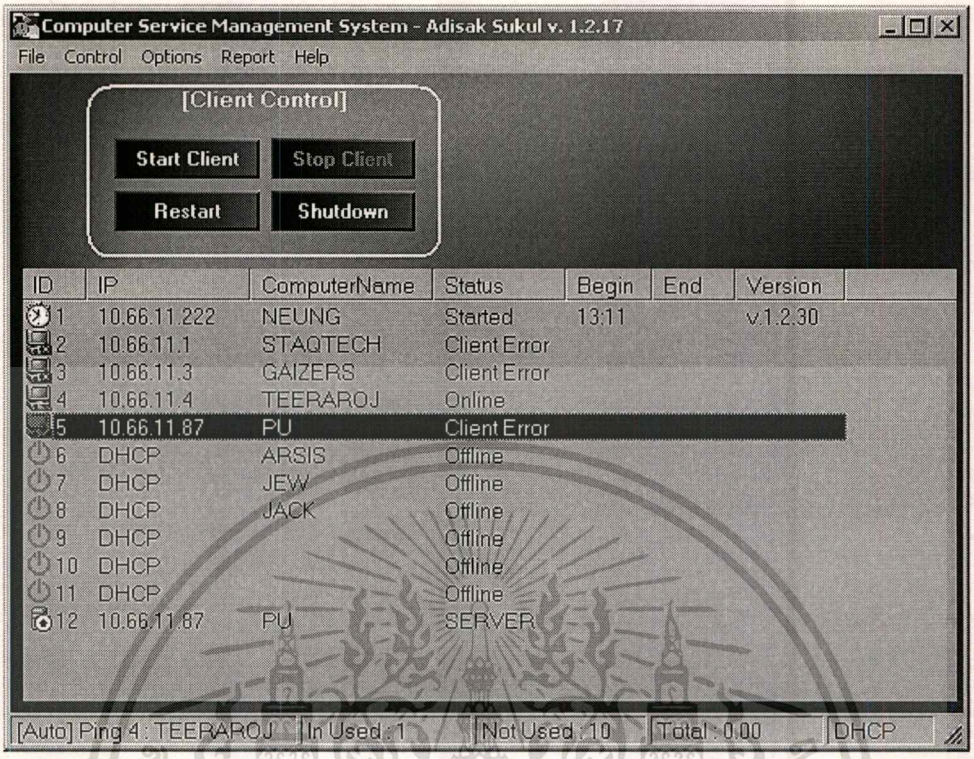
มี Client 200เครื่อง ความละเอียดของเวลาที่ต้องการในการตรวจสอบ คือ ทุกๆ 5 วินาที หรือ 12 ครั้งต่อนาที ภาระงานของ Server Program ที่จะต้องทำงานตรวจสอบการหมดเวลาให้บริการนี้คือ

12 x 200 = 2,400 ครั้ง ต่อนาที

ด้วยเหตุนี้ผู้พัฒนาจึงออกแบบระบบให้ Client Program เป็นผู้ตรวจสอบการหมดเวลาให้บริการของเครื่อง Client เอง และเพื่อความปลอดภัย จึงมีส่วนการทำงาน Time Server นี้ เพื่อ Sychronize ให้เวลาของ Server กับ Client ตรงกันเสมอ

หน้าที่การทำงานหลักของระบบคือการควบคุมดูแล Client ที่ลงทะเบียนไว้ในระบบ ฉนั้น Server Program จึงมี Interface หลักที่การแสดงรายการ และสถานะของ Client ดังรูปที่ 4.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.1 แสดงหน้าจอหลักในการใช้งาน Server Program

Server Program จะแสดงรายการ และสถานะของเครื่อง Client โดยสถานะที่แสดงมีดังนี้

ตารางที่ 4.1 แสดงรายการ และสถานะของเครื่อง Client

Status	Description
Offline	เครื่อง Client ปิดอยู่
Online	เครื่อง Client เปิดอยู่และพร้อมใช้งาน โดย Client Program ทำการ Lock หน้าจอเครื่องอยู่
Started	ผู้ใช้กำลังใช้งานเครื่องอยู่ โดยจะมีเวลาเริ่มต้นของการใช้บริการแสดง
Client Error	เครื่องเปิดอยู่แต่ Client Program ไม่ได้ทำงานอย่างถูกต้อง สถานะนี้จะเกดขึ้นเมื่อ Ping Server ตรวจสอบเจอว่าเครื่อง Client นั้นเปิด แต่ Server Program ไม่สามารถติดต่อกับ Client Program บนเครื่องได้
Server	แสดงถึงเครื่องที่เป็น Server จะอยู่ในตำแหน่งท้ายสุดของ List เพื่อความสะดวกในการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดย Status Bar ด้านล่างจะแสดงถึงการทำงานของ Ping Server ว่าปัจจุบันกำลังส่ง Ping Message ไปที่เครื่อง Client ใดอยู่ รวมทั้งแสดงจำนวนของเครื่องรวมที่ถูกใช้งานอยู่ และเวลารวมในการใช้งานของเครื่องเหล่านั้น

สำหรับ Menu ในการใช้งาน ถูกแบ่งให้เข้าใจได้ง่าย โดยแบ่งเป็น

รูปที่ 4.2 แสดง File Menu

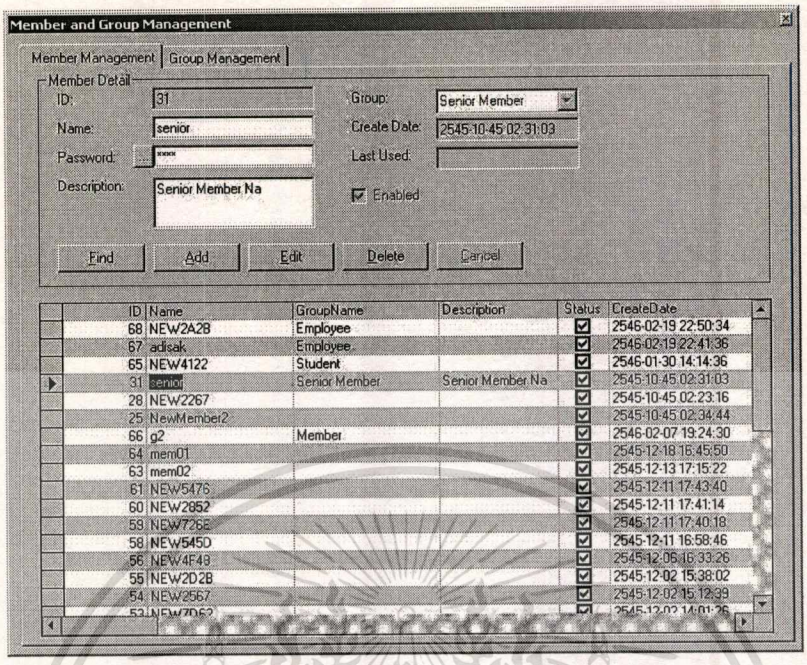
File Menu ประกอบไปด้วย

- Login คือการเข้าใช้โปรแกรมของผู้ดูแลระบบ โดยจะมีการถาม Username และ Password
- Logout การออกจากการใช้ของผู้ดูแลระบบ
- Exit การออกจากโปรแกรม

รูปที่ 4.3 แสดง Member Management Menu ที่อยู่ภายใต้ Control Menu

Member Management Menu ซึ่งอยู่ภายใต้หัวข้อ Control Menu โดยจะประกอบไปด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



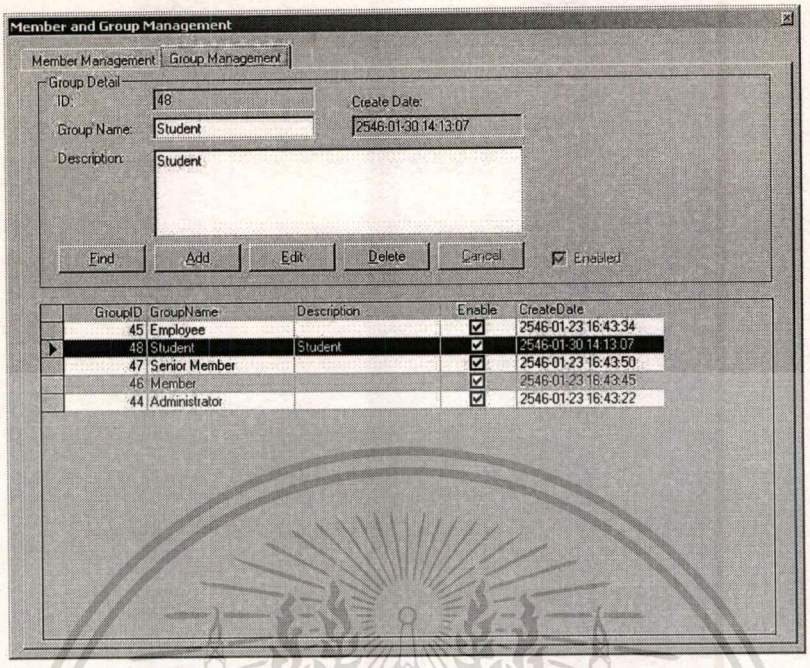
รูปที่ 4.4 แสดง Member Management

Member Management เป็นหน้าจอในการจัดการกับ Member โดยแสดงข้อมูลรายละเอียดต่างๆของ Member รวมไปถึง Group ที่ Member นั้นอยู่, วันที่ Member ถูก Add เข้าสู่ระบบ และการใช้งานครั้งสุดท้ายของ Member นั้น

โดยผู้ดูแลระบบสามารถ

- Find ค้นหารายการ Member ได้
- Add เพิ่มรายการ Member
- Edit แก้ไขรายการ Member
- Delete ลบรายการ Member
- Enable/Disable Member ได้ เพื่อความสะดวกแทนการลบและสร้างใหม่
- เลือก Group ให้กับแต่ละ Member ได้

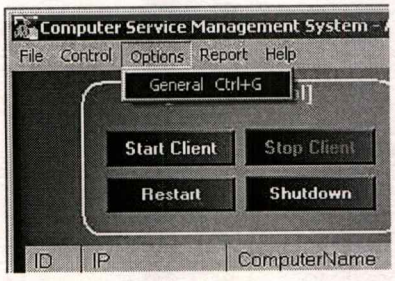
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 แสดง Group Management

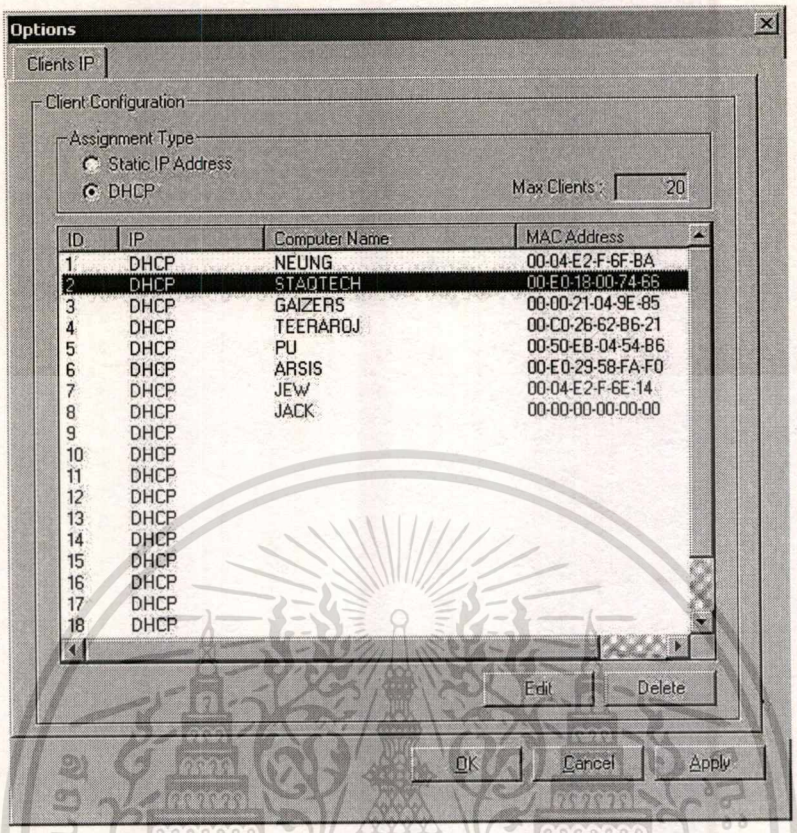
Group Management เป็นหน้าจอในการจัดการกับ Group ของ Member โดยแสดงข้อมูลรายละเอียดต่างๆของ Group รวมถึงวันที่ Group ถูก Add เข้าสู่ โดยผู้ดูแลระบบสามารถ

- Find ค้นหารายการของ Group
- Add เพิ่มรายการ Group
- Edit แก้ไขรายการ Group
- Delete ลบรายการ Group
- Enable/Disable Group ได้ เพื่อความสะดวกแทนการลบและสร้างใหม่



รูปที่ 4.6 แสดง General Menu ซึ่งอยู่ภายใต้ Options Menu

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 แสดง Client IP Menu ซึ่งอยู่ภายใต้ Options Menu

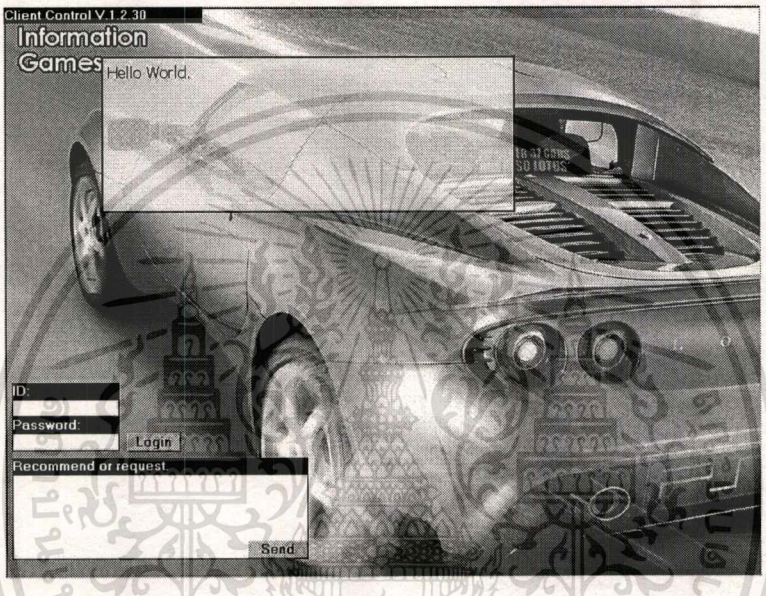
General Menu (Client IP) ซึ่งอยู่ภายใต้ Options Menu เป็นหน้าจอที่ใช้ในการลงทะเบียนเครื่อง Client เข้าสู่ระบบ โดยสามารถเลือกการลงทะเบียนได้ 2 ลักษณะ คือ

- Static IP Address ใช้สำหรับลงทะเบียน Client ที่ใช้ Static IP Address โดยใส่ชื่อเครื่อง หรือ IP Address ของเครื่อง จากนั้น Server Program จะจดจำเครื่อง Client ด้วย IP Address ต่อไป
- DHCP ใช้สำหรับลงทะเบียน Client ที่รับการแจก IP Address จาก DHCP Server ซึ่ง IP Address ของ Client จะเปลี่ยนแปลงไปทุกครั้งที่เปิดเครื่อง วิธีการลงทะเบียนทำโดยใส่ชื่อเครื่องของ Client ลงไป โดยที่ Server Program จะทำการหา IP Address เอง และจดจำ Client นั้นด้วยชื่อเครื่องเพียงอย่างเดียว

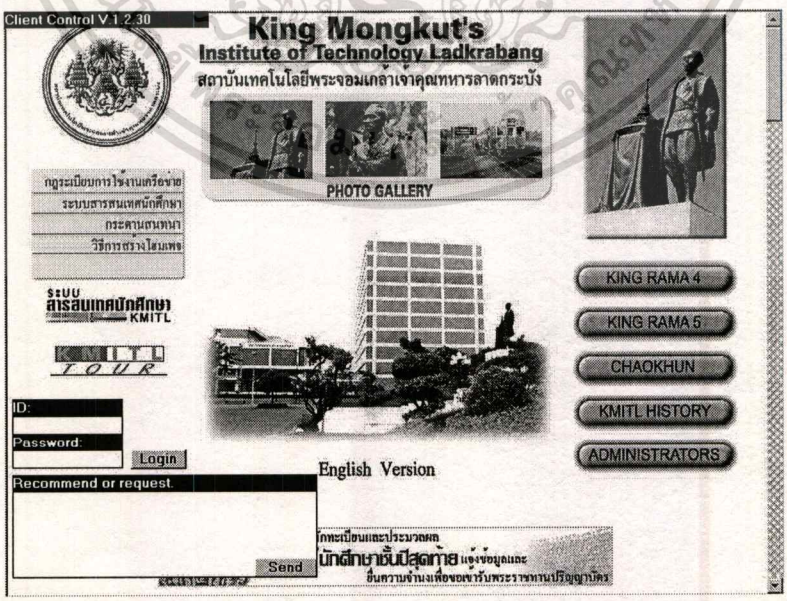
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การพัฒนาส่วนการทำงานของ Client Program

Client Program นั้นมีหน้าที่การทำงานหลักคือการ Lockหน้าจอเครื่องจึงถูกออกแบบมาให้ทำงานอยู่ด้านบนที่สุดเพื่อให้ทับ Application ในเครื่อง และทำการยกเลิก Hot Key บางประเภทของ Windows เพื่อป้องกันการปิด Client Program โดยใช้ HTTP Document ที่ได้รับจาก Server Program



รูปที่ 4.8 แสดงหน้าจอการทำงานของ Client โดยใช้รูปภาพ



รูปที่ 4.9 แสดงหน้าจอการทำงานของ Client โดยใช้ Web Page มาใช้เป็น Background

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.9 แสดงให้เห็นว่าสามารถใช้ Web Page มาทำเป็นภาพหน้าจอของเครื่อง Client ได้ เพื่อให้ผู้ใช้สามารถอ่านข่าวประชาสัมพันธ์หรือกฎระเบียบข้อบังคับของการใช้งานระบบได้

หน้าจอของ Client Program จะประกอบไปด้วยช่องรับ Input ได้แก่

- Username เพื่อให้ผู้ใช้ใส่ Username ที่ใช้ในการ Login
- Password เพื่อให้ผู้ใช้ใส่ Password ที่ใช้ในการ Login
- Comment เพื่อให้ผู้ใช้แสดงความคิดเห็น หรือส่งข้อความให้ผู้ดูแลระบบทราบ

4.3 การพัฒนาส่วนการสร้างรายงาน

ในการนำระบบ Software เข้ามาใช้อำนวยความสะดวกในการบริหารการให้บริการคอมพิวเตอร์นั้น ประโยชน์สำคัญอีกอย่างหนึ่งคือการที่ผู้ให้บริการสามารถที่จะนำข้อมูลประวัติของการใช้บริการมาสร้างเป็นรายงานเพื่อให้ทราบถึงผลลัพธ์และประสิทธิภาพของการให้บริการ รวมทั้งสามารถนำมาวิเคราะห์ความเหมาะสมระหว่างจะนวนเครื่องคอมพิวเตอร์และจำนวนผู้ใช้งาระบบเพื่อปรับปรุงประสิทธิภาพการให้บริการให้ดีขึ้นได้อย่างถูกต้อง และมีข้อมูลยืนยันที่น่าเชื่อถือมากขึ้น

โดยในขั้นตอนการสร้างรายงานนั้น ส่วนการทำงาน Report Creator จะอ่านข้อมูลการใช้งานจากตาราง Log ซึ่งรวบรวมเหตุการณ์ต่างๆที่เกิดขึ้นในระบบเก็บอยู่ในฐานข้อมูลอยู่ก่อนแล้ว ดังนั้นในอนาคตหากพัฒนาเพิ่มรูปแบบของรายงานให้มากขึ้น Report Creator ก็จะสามารถสืบค้นข้อมูลใน Log เพื่อสร้างเป็นรายงานในรูปแบบใดก็ได้

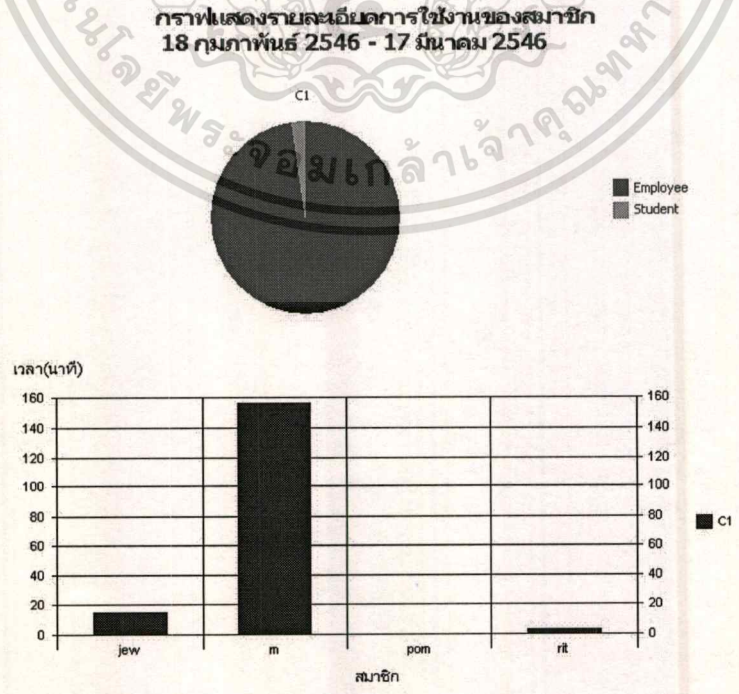
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Adisak Sukul 44067095
Computer Service Management System Report
21 มกราคม 2546 - 24 มกราคม 2546

Page : 1 / 3

Date	ID	Member	เวลาเริ่มต้น	เวลาสิ้นสุด	เวลาที่ใช้	ส่วนลด	จำนวนเงิน
รายงานประจำวันที่ 21 มกราคม 2546							
2546-01-21		Coupon_Time				0.00	99.00
2546-01-21		Coupon_Time				0.00	99.00
2546-01-21		Coupon_Time				0.00	99.00
2546-01-21		Coupon_Time				0.00	99.00
2546-01-21	06	Member1	16:04	17:09	01:05	0.00	20.00
2546-01-21	19	Member5	16:05	19:49	02:55	0.00	0.00
2546-01-21	18	Member8	16:05	16:40	00:35	0.00	10.00
2546-01-21	19	Member1	16:05	16:40	00:35	0.00	10.00
2546-01-21	17	Member5	16:05	16:40	00:35	0.00	10.00
2546-01-21	07	Member8	16:17	17:54	01:36	0.00	35.00
2546-01-21	14	Member1	16:20	18:16	01:56	0.00	40.00
2546-01-21	08	Member1	16:20	18:50	02:30	0.00	50.00
2546-01-21	20	Member5	16:36	17:06	00:30	0.00	10.00
2546-01-21	17	Member8	17:03	18:08	01:05	0.00	20.00
2546-01-21	06	Member1	17:22	19:20	01:57	0.00	40.00
2546-01-21	20	Member1	17:22	19:17	01:55	0.00	40.00
2546-01-21	17	Member5	19:05	19:45	00:40	0.00	0.00
2546-01-21	06	Member1	19:49	19:57	00:08	0.00	5.00
2546-01-21	07	Member5	19:49	19:57	00:08	0.00	5.00
2546-01-21	08	Member8	19:49	19:57	00:08	0.00	5.00
2546-01-21	17	Member1	22:18	22:26	00:07	0.00	0.00
		ส่วนลดโดยรวมทั้งสิ้น				0.00	
		รวมเวลาการใช้งาน Member				00003:42	ชั่วโมง
		จำนวนเงินรวมประจำวันที่ 21-ม.ค.-46					696.00

รูปที่ 4.10 แสดงรายงานการใช้บริการ



รูปที่ 4.11 แสดงกราฟการใช้บริการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการทดลอง

จากการพัฒนาระบบบริหารการให้บริการคอมพิวเตอร์ซึ่งใช้ในการควบคุมการให้บริการใช้เครื่องคอมพิวเตอร์ของผู้ใช้ โดยได้ทำการทดสอบโดยใช้เครื่องคอมพิวเตอร์ 1 เครื่องติดตั้ง Server Program และเครื่องคอมพิวเตอร์ 4 เครื่อง ติดตั้ง Client Program โดยเครื่องคอมพิวเตอร์ทั้ง 5 เครื่องเชื่อมต่อกันด้วย Switching ในระบบ LAN และได้ทำการสร้าง Member จำนวน 10 Member และ Group ขึ้น 3 Group เพื่อทำการทดสอบระบบ จากนั้นได้ทำการลงทะเบียน Client โดยใช้รูปแบบการจดจำ Client แบบ DHCP

5.1 ผลการทดสอบระบบ

- เมื่อเริ่มต้นเปิดระบบ โดยเปิด Server Program และเปิดเครื่อง Client เพียง 3 เครื่องและทำการ Remove Client Program ออกจากเครื่อง Client เครื่องหนึ่ง แล้ว Server Program แสดงสถานะของเครื่อง Client ได้อย่างถูกต้องทั้งหมด คือ Online 2 เครื่อง, Offline 1 เครื่อง และ Client Error อีกหนึ่งเครื่อง

- ทดลอง Login จากเครื่อง Client โดย Member ที่สร้างขึ้น ระบบก็สามารถทำงานได้อย่างถูกต้อง โดยสถานะการทำงานที่แสดงบนหน้าจอของ Server Program ก็เปลี่ยนแปลงได้อย่างถูกต้อง

- ผู้ใช้สามารถ Logout เพื่อยุติการใช้บริการ และผู้ดูแลสามารถสั่งให้ยุติการใช้บริการ ระบบก็ทำงานได้อย่างถูกต้อง

- ผู้ดูแลสามารถสั่ง Shutdown หรือ Restart เครื่อง Client แต่ละเครื่องโดยทำงานได้ถูกต้อง

- ทดสอบการรับและส่ง Message ระหว่าง Server Program กับ Client Program โดยใช้ Debug Program ซึ่งสร้างขึ้นเพื่อดักจับ Message ที่ติดต่อกันผ่านเครือข่าย ก็ได้ผลลัพธ์ถูกต้องตามที่ออกแบบระบบ

- ทดสอบการปิดเครื่อง Server โดยใช้วิธีการตัดไฟเพื่อให้เครื่อง Server ปิดในทันที เพื่อทดสอบกรณีที่ Server Program ไม่สามารถ Save Setting ต่างๆก่อนปิดโปรแกรม หลังจากเปิดเครื่อง Server ขึ้นมาใหม่ Server Program ก็ส่ง Rejoin Message ให้กับทุกๆ Client และ Client ก็สามารถ Join กลับเข้ามาสู่ระบบได้อย่างถูกต้อง

- ระบบสามารถสร้างรายงานการใช้ได้ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 การทดสอบการใช้ทรัพยากรเครือข่ายของระบบ

โครงสร้างการทำงานของระบบนั้น ถูกออกแบบมาเพื่อให้สามารถมีเครื่อง Server หลายเครื่องทำงานบนระบบเครือข่ายเดียวกันได้ เพื่อเพิ่มความสะดวกในการให้บริการและสามารถแยกส่วนการควบคุมเครื่องคอมพิวเตอร์ได้เป็นหลายกลุ่ม แต่ในขณะเดียวกันก็มีข้อเสียเนื่องจากระบบมีการติดต่อกันโดยใช้ Broadcast Message จึงทำให้เกิดภาระงานที่มากขึ้นบนระบบเครือข่ายตามไปด้วย

เนื่องจากในการทำงานของระบบบริหารการให้บริการเครื่องคอมพิวเตอร์นี้ ใช้การส่ง Message ติดต่อกันผ่านเครือข่ายอยู่ตลอดเวลาที่ระบบทำงาน ผู้พัฒนาจึงทำการทดสอบการใช้ทรัพยากรเครือข่ายของระบบ ในกรณีการทำงานปกติ เพื่อให้สามารถประมาณการใช้ทรัพยากรของเครือข่ายก่อนติดตั้งระบบได้ โดยทดสอบกับ LAN ความเร็ว 100Mbps โดยกำหนดให้ระบบเปิดให้บริการวันละ 8 ชั่วโมงโดยระบบมีการเริ่มต้น 1 ครั้งและปิด 1 ครั้ง โดย User แต่ละคนมีการ Login เข้าใช้งานเฉลี่ยชั่วโมงละ 1 ครั้ง

ตารางที่ 5.1 แสดงขนาดของ Message โดยเฉลี่ยที่ส่งผ่านเครือข่าย

	ขนาด(bytes)	ขนาด(bits)	วันละ(ครั้ง)	ขนาด(bits)/วัน	ขนาด(bits)/วินาที
CLI_LOGIN	69	552	8	4416	0.153333333
SVR_DENY	20	160	3	480	0.016666667
SVR_START	61	488	8	3904	0.135555556
CLI_STOP	48	384	8	3072	0.106666667
SVR_STOP	48	384	8	3072	0.106666667
PING Request/Reply	32	256	480	122880	4.266666667
PING_DETECT	64	512	8	4096	0.142222222
				รวม	4.774444444

Message ที่ส่งแบบ Broadcast

SVR_DATETIME	72	576	480	276480	9.6

Message ที่ส่งทุกๆ 1 วัน

SVR_REJOIN	46	368	1	368	0.012777778
CLI_JOIN	46	368	1	368	0.012777778
SVR_CONFIG	79	632	1	632	0.021944444
				รวม	0.0475

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

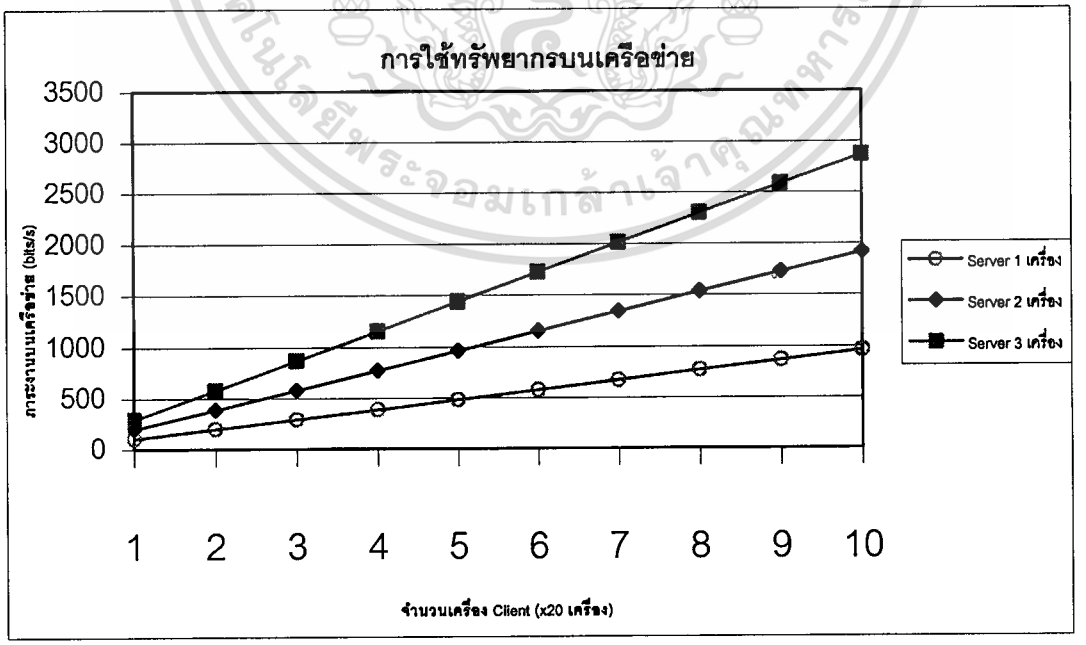
จากตาราง มี message 3 กลุ่มด้วยกัน คือ

- message ที่ต้องส่งทุกๆครั้งที่มีการเข้าใช้งาน
- message ประเภท Broadcast ส่งเพียงครั้งเดียว Client ทุกเครื่องสามารถนำไปใช้งานได้ทันที จึงประหยัดทรัพยากรเครือข่ายในกรณีที่มีเครื่อง Client จำนวนมาก
- message สำหรับเริ่มต้นระบบ ส่งเพียงครั้งแรกที่ระบบเริ่มต้นเท่านั้น

โดยเมื่อประมาณการขนาดของ message ที่ใช้ในหน่วยเวลาแล้ว สามารถจำลองการใช้งานเมื่อมี Client จำนวนต่างกัน ตั้งแต่ 20 เครื่อง ไปจนถึง 200 เครื่องในเครือข่ายเดียวกัน และจำลองการใช้ Server Program ตั้งแต่ 1 เครื่อง ถึง 3 เครื่องได้ขนาดของ message มีหน่วยเป็น bits/s ดังนี้

ตารางที่ 5.2 แสดงรายการ และสถานะของเครื่อง Client

จำนวน Client	20	40	60	80	100	120	140	160	180	200
Server 1 เครื่อง	105.05	200.45	295.85	391.25	486.65	582.05	677.45	772.85	868.25	963.65
Server 2 เครื่อง	200.5	391.3	582.1	772.9	963.7	1154.5	1345.3	1536.1	1726.9	1917.7
Server 3 เครื่อง	295.95	582.15	868.35	1154.55	1440.75	1726.95	2013.15	2299.35	2585.55	2871.75



รูปที่ 5.1 แสดงกราฟการใช้ทรัพยากรเครือข่ายของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากกราฟ สามารถสรุปได้ว่า ระบบใช้ทรัพยากรน้อยมากเมื่อเทียบกับขนาดของเครือข่าย 100 Mbps เช่น ที่ Server 1 เครื่อง Client 20 เครื่อง ระบบใช้ทรัพยากรเครือข่ายเพียง 105.05 bit/s

การใช้เครื่อง Server หลายเครื่องในระบบเครือข่ายเดียวกันนั้นสามารถทำได้ โดยทำให้เกิดความสะดวกมากขึ้นในแง่ของการแบ่งส่วนการควบคุม แต่มีข้อเสียคือภาระงานที่เพิ่มขึ้นดังที่แสดงให้เห็นในกราฟ แต่หากเทียบกับขนาด Bandwidth ก็ยังนับว่าระบบบริหารการให้บริการคอมพิวเตอร์นี้ใช้ทรัพยากรเครือข่ายน้อยมาก

5.3 ปัญหาในการพัฒนาระบบ

- Version ของระบบปฏิบัติการ Windows แต่ละ Version มีรายละเอียดที่ต่างกัน เมื่อพัฒนาระบบบน Version หนึ่งแล้วนำไปใช้บนอีก Version หนึ่งจะเกิดการทำงานผิดพลาด และคำสั่งบางส่วนไม่สามารถใช้กับ Windows บาง Version ได้ ซึ่งผู้พัฒนาระบบทดสอบระบบโดยใช้ Windows 2000 ทำงานเป็นเครื่อง Sever และใช้ WindowsME เป็นเครื่อง Client

- ในการส่ง UDP Message ในเครือข่ายมีการ Disable UDP Message เอาไว้ หรือมีการเชื่อมต่อระหว่างเครือข่ายโดยใช้ Network Address Translator ทำให้ไม่สามารถใช้งานได้

5.4 แนวทางในการพัฒนาระบบเพิ่มเติมให้มีความสามารถมากขึ้น

- เพิ่มเติมรูปแบบการใช้บริการเครื่อง เช่น การให้บริการแบบจำกัดเวลาล่วงหน้า หรือการให้บริการแบบมีการเก็บเงินค่าบริการ ซึ่งเป็นการขยายกลุ่มองค์กรผู้ใช้ระบบนี้ให้มีมากยิ่งขึ้น นอกจากนั้นอาจเพิ่มความสามารถการกำหนดราคาค่าใช้บริการให้สามารถกำหนดราคาให้ละเอียดยิ่งขึ้นโดยอาจกำหนดราคาตามช่วงเวลาได้ และอาจเพิ่มรูปแบบการให้บริการแบบการพิมพ์ Coupon ที่มี Username และ Password รวมอยู่ด้วยแล้ว และจำกัดเวลาการใช้งานในตัว Username นั้นแล้ว ซึ่งจะสะดวกกับผู้ใช้บางประเภทซึ่งไม่ต้องผูกติดกับ Member ที่แน่นอน

- อาจเพิ่มเติมการ Capture หน้าจอของเครื่อง Client มาแสดงบนที่ Server Program เพื่อให้ผู้ดูแลระบบทราบว่าผู้ใช้กำลังใช้งานอะไรอยู่ในขณะนั้น โดยอาจแสดงเป็นภาพเล็กๆลักษณะ Thumbnails ของ Client ทั้งหมด จากนั้นผู้ดูแลระบบสามารถเลือกหน้าจอที่ต้องการมองภาพใหญ่หรือต้องการเข้าควบคุม Mouse และ Keyboard ของเครื่อง Client เลยได้ จะเป็นประโยชน์ในการควบคุมการใช้งาน ซึ่งมีข้อเสียคืออาจทำให้การเข้าใช้งานเครื่องขาดความเป็นส่วนตัว และต้องใช้การส่งข้อมูลบนเครือข่ายจำนวนมาก เนื่องจากการดูภาพหน้าจอต้องใช้การส่งข้อมูลขนาดใหญ่ถึงแม่จะมีการบีดอัดแล้วก็ตาม

- สามารถทำงานร่วมกับระบบ Inventory Management จะได้ประโยชน์มาก เนื่องจากการทำงานเป็นลักษณะเก็บรายการของเครื่อง Client อยู่แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] Roman Steven. Jan 2000. **Win32 API Programming With Visual Basic** . : O'rerilly.

[2] Craigg John Clark . 1998 . **Microsoft Visual Basic 6.0 Developer Workshop**. :Microsoft Press.

[3] PARALLEL DATA LAB. 2002. **TCP and UDP message**. [Online]. Avaliable: http://www.pdl.cmu.edu/mailinglists/ips/mail/msg0131.html.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

นาย อดิศักดิ์ สุกุล

เกิดเมื่อวันที่ 26 สิงหาคม 2522

สถานที่เกิด กรุงเทพมหานคร

ประวัติการศึกษา

2539 –2542	วิทยาศาสตร์บัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าลาดกระบัง
2544 – ปัจจุบัน	วิทยาศาสตร์มหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ แขนงวิชา วิทยาการสารสนเทศ คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าลาดกระบัง

ประวัติการทำงาน

2542 – 2544	Mweb Thailand Co., Ltd.– Programmer
2544 – ปัจจุบัน	STAQ Technologies Co., Ltd. – Project Manager, Founder

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้