

ระบบป้องกันความปลอดภัยตรวจสอบการเข้า-ออกด้วยลายนิ้วมือ

Security Building Automation Control System by Fingerprint



นายสมเกียรติ

เกิดหิรัญ

นายโกยิต

อรุณจิตต์

นายอาทิตย์

วาริเจริญชัย

ร.พ.
๕๖๖๕

เลขหมู่.....๒๐๔๓
เลขทะเบียน.....42520
วัน, เดือน, ปี 24 พ.ค. 2545

b.....
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมการวัดคุม

ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

๒๕๔๕

Security Building Automation Control System by Fingerprint

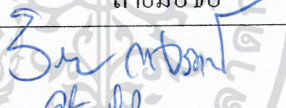



A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIRE FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN INSTRUMENTATION ENGINEERING
DEPARTMENT OF INDUSTRIAL INSTRUMENTATION TECHNOLOGY
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองปริญญาโท

หัวข้อปริญญาโท ระบบป้องกันความปลอดภัยตรวจสอบการเข้า-ออกด้วยลายนิ้วมือ
Security Building Automation Control System by Fingerprint
นักศึกษาผู้จัดทำ นายสมเกียรติ เกิดหิรัญ รหัสประจำตัว 41012102
นายโกยิต อรุณจิตต์ รหัสประจำตัว 41013402
นายอาทิตย์ วารีเจริญชัย รหัสประจำตัว 41013438
ปริญญา วิศวกรรมศาสตรบัณฑิต
สาขาวิชา วิศวกรรมการวัดคุม
ปีการศึกษา 2543

อาจารย์ผู้ควบคุมปริญญาโท	ลายมือชื่อ
ผศ.วิริยะ กองรัตน์	
อาจารย์อาจันต์ น่วมสำราญ	

วัน/เดือน/ปี ที่สอบ วันจันทร์ที่ 9 เมษายน พ.ศ. 2544

สถานที่สอบ ณ.ห้องสอบปริญญาโท ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม

ภาควิชารับรองแล้ว



(ผศ.ประสิทธิ์ จุลเสรีวงศ์)

หัวหน้าภาควิชาฯ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	ระบบป้องกันความปลอดภัยตรวจสอบการเข้า-ออกด้วยลายนิ้วมือ Security Building Automation Control System by Fingerprint	
นักศึกษาผู้จัดทำ	นายสมเกียรติ	เกิดศิริณู
	นายโกษิต	อรุณจิตต์
	นายอาทิตย์	วาริเจริญชัย
อาจารย์ที่ปรึกษา	ผศ.วิริยะ	กองรัตน์
	อาจารย์อาจินต์	น่วมสำราญ
ปีการศึกษา	2543	

บทคัดย่อ

ปัจจุบัน ระบบป้องกันความปลอดภัยโดยใช้ บัตรแถบแม่เหล็ก รหัสผ่านที่เป็นตัวเลข หรือตัวอักษรที่นิยมใช้กันอย่างแพร่หลายแต่การป้องกันด้วยวิธีนี้ยังไม่อาจสร้างความมั่นใจได้มากนัก เมื่อเทียบกับการตรวจสอบทางกายภาพ ซึ่งเป็นลักษณะเฉพาะทางกายภาพของแต่ละบุคคลที่ยากต่อการปลอมแปลง

ดังนั้นปริญญานิพนธ์ฉบับนี้จึงได้นำเอาเทคโนโลยีการจดจำตรวจสอบยืนยันตัวบุคคลด้วยลายนิ้วมือนำมาออกแบบและประยุกต์ใช้ เพื่อเป็นแนวทาง ในการออกแบบระบบรักษาความปลอดภัยในโรงงานหรือสำนักงานที่ต้องการระบบรักษาความปลอดภัยที่มีความน่าเชื่อถือ ปริญญานิพนธ์นี้จะเป็นการออกแบบระบบ โดยนำระบบการสื่อสารข้อมูลทางเน็ตเวิร์คมาเป็นพื้นฐานการออกแบบ เพื่อต้องการให้เป็นระบบป้องกันความปลอดภัยตรวจสอบการเข้า - ออกด้วยลายนิ้วมือ

Thesis Title	Security Building Automation Control System by Fingerprint	
Authers	Mr. Somkeat	Keadhirun
	Mr. Kosit	Arunjit
	Mr. Artit	Wareecharearnchai
Thesis Advisor	Asst. Prof. Viriya	Kongrut
	Mr. Arjin	Numsomran
Year	2000	

ABSTRACT

Nowadays the security system used the ID card number alphabet to pass security code to be popular. But this prevention can not be firmly believe for user as compare with verification by Biometrics method that particularity individual so difficulty to fake

So in this thesis use the particularity individual technology to attest person by fingerprint, The design and application is the new trend to design security systems in the factory and office for make inspect high. This Thesis design make under communicating network.

กิตติกรรมประกาศ

ปริญญาบัตรฉบับนี้สำเร็จลุล่วงไปด้วยดีเพราะได้รับความอนุเคราะห์และคำแนะนำจากท่าน ผศ.วิริยะ กองรัตน์ และ อาจารย์อานินต์ น่วมสำราญ ที่ได้กรุณาให้โอกาสได้จัดทำปริญญาบัตรเรื่องนี้และพร้อมทั้งยังกรุณาเอื้อเฟื้ออุปกรณ์เครื่องมือต่างๆ ในการทำปริญญาบัตรเล่มนี้จนสำเร็จลุล่วงไปด้วยดีผู้วิจัยรู้สึกซาบซึ้งและขอขอบพระคุณเป็นอย่างสูง

ขอขอบพระคุณ อาจารย์ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรมทุกท่าน ตลอดจนเพื่อนๆ ที่คอยให้กำลังใจด้วยดีตลอดมา

และที่ลืมเสียมิได้คือ ขอกราบขอบพระคุณคุณพ่อและคุณแม่ อันเป็นที่รักยิ่ง ที่สนับสนุนและเป็นแรงบันดาลใจในการทำปริญญาบัตรฉบับนี้

คุณค่าและประโยชน์อันพึงมีจากปริญญาบัตรฉบับนี้ ผู้วิจัยขอบแต่ผู้มีพระคุณทุกท่าน

คณะผู้จัดทำ

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญภาพ.....	VI
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและเหตุจูงใจของ โครงการงานวิจัยปริญญาโท.....	1
1.2 วัตถุประสงค์ของปริญญาโท.....	1
1.3 ขอบเขตของปริญญาโท.....	1
1.4 วิธีการดำเนินงาน.....	1
บทที่ 2 ทฤษฎีที่ใช้ในการทำโครงการงานวิจัยปริญญาโท.....	2
2.1 ลักษณะทั่วไปของระบบเครือข่าย.....	2
2.2 การสื่อสารบนระบบเครือข่าย.....	3
2.3 การแบ่งปันการใช้การแจกจ่าย.....	5
2.4 ความปลอดภัยบนระบบเครือข่าย.....	6
2.5 การขนส่งข้อมูลบนเครือข่าย.....	10
2.6 ชุดโปรโตคอลที่ซีพี/ไอพี (TCP/IP).....	12
2.7 โปรโตคอลในชั้นอินเทอร์เน็ต.....	14
2.8 ทฤษฎีการทำงานของโปรแกรม Visual Basic.....	16
2.9 ระบบ ไคลเอนต์ / เซิร์ฟเวอร์.....	22
2.10 การสร้างฐานข้อมูลด้วยโปรแกรม Access.....	27
2.11 รูปแบบการติดต่อฐานข้อมูลด้วย Visual Basic.....	30
บทที่ 3 โปรแกรมการจัดการตรวจสอบลายนิ้วมือ.....	33
3.1 ลักษณะ โปรแกรมการจัดการตรวจสอบลายนิ้วมือ.....	33
3.2 ลักษณะ โปรแกรมตรวจสอบลายนิ้วมือ.....	33

สารบัญ (ต่อ)

	หน้า
บทที่ 4 โครงสร้างโดยรวมของโปรแกรม.....	37
4.1 โครงสร้างการออกแบบ โปรแกรมระบบจัดการฐานข้อมูล.....	37
4.2 ลักษณะ โครงสร้างการออกแบบทางระบบเน็ตเวิร์ก.....	42
บทที่ 5 ขั้นตอนการใช้งานของโปรแกรมและผลการทดลอง.....	44
5.1 จัดเก็บข้อมูลต่างๆที่ได้ทำการออกแบบเก็บไว้เป็นฐานข้อมูลที่เครื่องเซิร์ฟเวอร์.....	44
5.2 ทำการรัน โปรแกรมบริการข้อมูลที่เครื่อง เซิร์ฟเวอร์.....	48
5.3 ทำการรัน โปรแกรมที่เครื่องควบคุมทาง โคลเอนต์.....	49
5.4 ขั้นตอนการรับข้อมูลจากผู้ใช้.....	50
5.5 แสดงรายงาน.....	54
บทที่ 6 สรุปผลการวิจัยเสนอแนะ.....	57
บทสรุป.....	57
ข้อเสนอแนะ.....	57
ภาคผนวก.....	58
1. แสดงสอร์ซโค้ด โปรแกรมการจัดการเก็บข้อมูลประวัติผู้ใช้ที่เครื่องเซิร์ฟเวอร์....	58
2. แสดงสอร์ซโค้ด โปรแกรมการจัดการเก็บข้อมูลลายนิ้วมือ.....	71
3. แสดงสอร์ซโค้ด โปรแกรมตรวจสอบการร้องขอข้อมูลจากเครื่องควบคุม.....	76
4. แสดงสอร์ซโค้ด โปรแกรมเครื่องควบคุมที่เชื่อมต่อและร้องขอข้อมูลไปยังเครื่องเซิร์ฟเวอร์.....	84
บรรณานุกรม.....	94

สารบัญภาพ

ภาพที่	หน้า
2.1 การแบ่งปันฮาร์ดดิสก์	6
2.2 การต่อเครื่องพิมพ์บนระบบเครือข่าย	6
2.3 กลยุทธ์ในการรักษาความปลอดภัย	7
2.4 แสดงส่วนประกอบของแพ็กเก็ต	10
2.5 แสดงการแบ่งระดับการทำงานของ ทีซีพี / ไอพี โดยอ้างอิงจากมาตรฐาน OSI	13
2.6 แสดงรูปแบบของอินเทอร์เน็ตแอดเดรส	15
2.7 แสดงคอนโทรล เทเบิ้ล ที่ใช้ในการแสดงข้อความ	18
2.8 แสดงคอนโทรลปุ่มคำสั่ง Command Button	18
2.9 แสดงคอนโทรล Frame สำหรับใช้ในการจัดกลุ่มคอนโทรลอื่นๆ	18
2.10 แสดงคอนโทรล ComboBox สำหรับแสดงรายการที่ต้องการเลือก	19
2.11 แสดงคอนโทรล ListBox สำหรับใช้แสดงข้อความหลายๆข้อความ	19
2.12 แสดงคอนโทรล Image and PictureBox สำหรับแสดงรูป	19
2.13 แสดงคอนโทรล OptionButton สำหรับใช้ในการเลือก	20
2.14 แสดงคอนโทรล Mscomm สำหรับใช้ในการติดต่อกับอุปกรณ์ภายนอก	20
2.15 แสดงระบบ ไคลเอนต์ / เซิร์ฟเวอร์	23
2.16 แสดงรูปแบบการให้บริการฐานข้อมูล	25
2.17 แสดงการให้บริการฐานข้อมูลแบบกระจาย	26
2.18 แสดงการให้บริการฐานข้อมูลขนาดใหญ่	26
2.19 แสดงการแปลงฐานข้อมูลต่างๆเป็นเท็กซ์ไฟล์	29
2.20 แสดงการร่วมกันใช้เพิ่มข้อมูลของโปรแกรม	29
2.21 แสดงตัวอย่างการสร้างตาราง	30
3.1 แสดงฟอร์มของโปรแกรมการเก็บข้อมูลลายนิ้วมือ	34
3.2 แสดงฟอร์มลักษณะการจับเก็บข้อมูลลายนิ้วมือ	34
4.1 แสดงโครงสร้างการทำงานของโปรแกรม	37
4.2 แสดงความสัมพันธ์ของตาราง	38
4.3 แสดงรายละเอียดข้อมูลในตาราง TblMember	39
4.4 แสดงรายละเอียดการจัดเก็บข้อมูลในตาราง TblPicture	39
4.5 แสดงรายละเอียดการจัดเก็บข้อมูลในตาราง TblRoomData	39

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

	หน้า
4.6 แสดงการบันทึกข้อมูลกำหนดการใช้ห้อง TblRoomfix	40
4.7 แสดงการบันทึกข้อมูลเหตุการณ์ของผู้ใช้ห้อง TblCheck	40
4.8 แสดงฟอร์มหลักของโปรแกรมการจัดการระบบความปลอดภัย.....	41
4.9 แสดงโครงสร้างของโปรแกรมควบคุมทางเน็ตเวิร์ก.....	42
4.10 แสดง Flow Chart การทำงานของโปรแกรม.....	43
5.1 แสดงฟอร์มเมนูหลักที่ใช้ในการจัดเก็บข้อมูล.....	44
5.2 แสดงฟอร์มเก็บข้อมูลค่าน้ำหน้าชื่อ.....	45
5.3 แสดงฟอร์มเก็บข้อมูลระดับผู้ใช้ที่เกี่ยวข้อง.....	45
5.4 แสดงฟอร์มเก็บข้อมูลสาขาวิชา.....	46
5.5 แสดงฟอร์มเก็บข้อมูลจำนวนห้องที่ต้องการควบคุม.....	46
5.6 แสดงฟอร์มเก็บประวัติผู้ใช้.....	47
5.7 แสดงฟอร์มกำหนดห้องให้กับผู้ใช้.....	47
5.8 แสดงฟอร์มเก็บข้อมูลลายนิ้วมือ.....	48
5.9 แสดงฟอร์มเปิดการเชื่อมต่อให้บริการข้อมูลที่เครื่องเซิร์ฟเวอร์.....	48
5.10 แสดงฟอร์มเมื่อเปิดการเชื่อมต่อเสร็จสมบูรณ์.....	49
5.11 แสดงฟอร์มเปิดการเชื่อมต่อไปที่เซิร์ฟเวอร์.....	49
5.12 แสดงฟอร์มหลักเมื่อมีการเชื่อมต่อเสร็จสมบูรณ์และรอรับข้อมูลจากผู้ใช้.....	50
5.13 แสดงฟอร์มลักษณะขั้นตอนรับข้อมูลรหัสประจำตัวจากผู้ใช้.....	51
5.14 แสดงฟอร์มลักษณะขั้นตอนการเลือกลายนิ้วมือ.....	52
5.15 แสดงฟอร์มลักษณะขั้นตอนรับข้อมูลลายนิ้วมือจากผู้ใช้.....	52
5.16 แสดงผลการตรวจสอบข้อมูลลายนิ้วมือกับฐานข้อมูลที่เครื่องเซิร์ฟเวอร์ที่ไม่ ถูกต้อง.....	53
5.17 แสดงผลการตรวจสอบข้อมูลลายนิ้วมือกับฐานข้อมูลที่เครื่องเซิร์ฟเวอร์ ที่ถูกต้อง.....	54
5.18 แสดงฟอร์มเมนูสำหรับแสดงรายงาน.....	55
5.19 แสดงฟอร์มสำหรับแสดงรายงานข้อมูลทั้งหมดที่เก็บไว้ในฐานข้อมูล.....	55
5.20 แสดงฟอร์มสำหรับแสดงรายงานข้อมูลแต่ละห้องที่มีผู้ใช้เข้าใช้งาน.....	56

บทที่ 1

บทนำ

1.1 ความเป็นมาและเหตุผลใจของโครงการวิจัยปริญญาโท

เนื่องจากวิศวกรนอกจากจะมีความรู้ทักษะทางการเรียนจากภาควิชาแล้ววิศวกรจะต้องมีความรู้ที่จะนำเอาอุปกรณ์ต่างๆที่มีอยู่มาประยุกต์ใช้งานได้อย่างมีประสิทธิภาพ

เทคโนโลยีการจดจำลักษณะทางกายภาพได้ถูกพัฒนามาโดยตลอด เพื่อนำมาใช้ในเชิงพาณิชย์มากขึ้น ด้วยเหตุนี้การจดจำลายนิ้วมือจึงได้ถูกนำมาประยุกต์ใช้งาน ในระบบที่ต้องการความปลอดภัย

1.2 วัตถุประสงค์ของโครงการวิจัยปริญญาโท

เพื่อสามารถที่จะนำเอาเทคโนโลยีการจดจำลายนิ้วมือ มาประยุกต์ใช้เป็นระบบตรวจสอบยืนยันตัวตน ให้เป็นระบบป้องกันความปลอดภัยกับห้องที่ต้องการควบคุมตรวจสอบการใช้ของบุคคลที่เกี่ยวข้อง

1.3 ขอบเขตของโครงการวิจัยปริญญาโท

1. ศึกษาและออกแบบระบบตรวจสอบการเข้า-ออกด้วยลายนิ้วมือผ่านทางระบบเน็ตเวิร์ก โดยใช้โปรแกรม Visual Basic ในการออกแบบจัดการกับระบบฐานข้อมูลในคอมพิวเตอร์ผ่านระบบเน็ตเวิร์ก
2. สามารถกำหนดการเข้าใช้ห้องให้กับบุคคลต่างๆได้
3. สามารถตรวจสอบการเข้าใช้ห้องของแต่ละบุคคลได้

1.4 ขั้นตอนการศึกษา

1. ทำความเข้าใจลักษณะการทำงานของเครื่องสแกนลายนิ้วมือกับ Visual Basic
2. ศึกษาทำความเข้าใจการติดต่อกับระบบของฐานข้อมูลในคอมพิวเตอร์
3. เขียนโปรแกรมทดสอบการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีที่ใช้ในการทำโครงการวิจัยปริญญาโท

2.1 ลักษณะทั่วไปของระบบเครือข่าย

2.2.1 ความหมายและความสำคัญของ Local Area Network (LAN)

โพลคอลลแอเรียเน็ตเวิร์ก (Local Area Network) หรือ LAN มีความหมายตามคำนิยามของ IEEE หรือ ISO คือ “เครือข่ายที่เป็นไปได้สำหรับการส่งข้อมูลที่มีความเร็วขนาดกลางไปจนกระทั่งความเร็วที่ถูกจำกัดขอบเขตอยู่ในสำนักงานอาคารหรืออาณาเขตหนึ่งๆ” นอกจากนี้สำหรับเครือข่ายอื่นๆที่นอกเหนือจาก LAN แล้วเครือข่ายที่มีขอบเขตกว้างขวางทั่วประเทศจะเรียกว่า WAN (Wide Area Network) และเครือข่ายที่มีขนาดระหว่าง LAN กับ WAN จะถูกเรียกว่า MAN (Metropolitan Area Network)

สำหรับโครงสร้างของ LAN นั้นจะเป็นเครือข่ายที่เชื่อมต่อโฮสต์คอมพิวเตอร์ เวิร์ดโปรเซสเซอร์ ไมโครคอมพิวเตอร์ หรือโทรสาร เป็นต้น ที่อยู่กระจายภายในอาคาร หรืออาณาเขตเดียวกันของสำนักงานหรือโรงงาน ด้วยความเร็วในการส่งประมาณ 1 Mbps-100 Mbps โดยมีจุดมุ่งหมายเพื่อใช้ในการริสรัด (Resource) เช่น ข่าวสารหรืออุปกรณ์ร่วมกัน

LAN จะมีพื้นที่เป้าหมายค่อนข้างแคบนอกจากนี้รูปแบบการติดตั้งโดยทั่วไปจะคล้ายคลึงกันและมีเสถียรภาพ LAN มีบทบาทเพื่อเพิ่มประสิทธิภาพของงานในสำนักงานของแต่ละองค์กรเพิ่มความก้าวหน้าของการผลิตและลดค่าใช้จ่ายโดยมีการทำงานในลักษณะของสำนักงานอัตโนมัติ : OA (Office Automation) โรงงานอัตโนมัติ : FA (Factory Automation) กล่าวคือ การนำอุปกรณ์แต่ละชนิดมาใช้ร่วมกัน

ดังนั้นเราอาจจะกล่าวได้ว่า LAN นั้นมีบทบาทสำหรับการสื่อสารข่าวสารที่ก้าวหน้าเป็นอิสระ ด้วยการเชื่อมต่อเทอร์มินัลของแต่ละแผนกภายในองค์กร

สำหรับผลลัพธ์ของการนำเอา LAN มาใช้นั้น

1. การใช้แหล่งข่าวสารร่วมเทอร์มินัลใดๆ ที่มีการต่อเข้าเครือข่าย สามารถติดต่อสื่อสารกันได้อย่างอิสระ เมื่อเป็นเช่นนี้ข่าวสารรวมทั้งหมดที่ถูกเก็บรักษาไว้ที่อุปกรณ์ไฟลั่วม สามารถถูกเรียกออกมาใช้ร่วมกันด้วยฟังก์ชันสื่อสารความเร็วสูงของ LAN ซึ่งจะทำให้การใช้ข่าวสารร่วมกันทั้งองค์กร จึงอาจกล่าวได้ว่าเป็นระบบไฟล์อิเล็กทรอนิกส์หรือระบบค้นหาดาต้าเบสนั่นเอง

2. ความรวดเร็วในการส่งข่าวสาร การส่งเอกสารข่าวสาร โดยใช้คนนำส่งแต่ละแผนกนั้นสามารถเปลี่ยนเป็นการส่งจากเทอร์มินัลของแผนกหนึ่งไปยังอีกแผนกหนึ่งหรือการส่งข่าวสารเดียวกันโดยใช้อิเล็กทรอนิกส์เมล์และสามารถกระทำด้วยความรวดเร็ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ความประหยัดเนื่องจากใช้อุปกรณ์ร่วมกัน อุปกรณ์ราคาแพงบางชนิด ไม่จำเป็นต้องใช้ในลักษณะส่วนตัว เช่น เครื่องพิมพ์เลเซอร์ ดังนั้นหากนำเอาฟังก์ชันการสื่อสารความเร็วสูงของ LAN มาใช้งานแล้วเทอร์มินัล ทั้งหมดสามารถใช้อุปกรณ์ราคาแพงร่วมกันได้ จึงทำให้ประหยัดค่าใช้จ่าย

2.2 การสื่อสารบนระบบเครือข่าย

2.2.1 การสื่อสารแบบ Real - Time

การสื่อสารแบบ Real - Time บนระบบเครือข่ายเกิดขึ้นได้หลายวิธี ดังตัวอย่างเช่น ห้องสนทนาบนเครือข่ายอินเทอร์เน็ต (Internet Chat Room) เป็นนวัตกรรมยอดเยี่ยมล่าสุด ที่ทำให้ผู้คนแสดงความคิดเห็นกันได้อย่างกว้างขวางตั้งแต่เนื้อหาด้านธรรมะจนถึงเรื่องหยาบโลน ประหนึ่งว่าพวกเขาได้นั่งรวมอยู่ในห้องเดียวกัน ไม่ว่าผู้คนเหล่านั้นจะอยู่ห่างไกลกันเพียงใดก็ตาม สิ่งที่ต้องทำก็คือ เชื่อมต่อสายโทรศัพท์เข้ามายังห้องประชุมเสมือนบนเครือข่ายอินเทอร์เน็ตในเวลาเดียวกัน เพื่อจะได้พูดคุยกันแบบ Real - Time

เช่นเดียวกับระบบเครือข่ายที่ใช้ในธุรกิจ ซึ่งสนับสนุนให้มีการสื่อสารแบบ Real - Time ตัวอย่างเช่น เมื่อพนักงานที่อยู่ต่างสำนักงานกันร่วมมือกันทำรายงาน โดยใช้ซอฟต์แวร์ หรือการประชุมจากระยะทางไกล (Videoconference) ก็อาศัยระบบเครือข่ายเพื่อโต้ตอบกันแบบ Real - Time ในกรณีของซอฟต์แวร์อย่างเช่น Microsoft NetMeeting อยู่ในระบบเครือข่ายที่อนุญาตให้กลุ่มสมาชิกแบ่งปันการติดตั้งโปรแกรมประยุกต์บนเครื่องคอมพิวเตอร์ภายในกลุ่ม และเช่นเดียวกัน เมื่อประธานกรรมการบริษัท (CEO) ส่งคำปราศรัยไปยังเครื่องคอมพิวเตอร์แบบตั้งโต๊ะทั่วทั้งองค์กรในเวลาอาหารกลางวัน ก็ใช้การสื่อสารของระบบแบบ Real - Time เมื่อผู้ร่วมงานของคุณเข้าร่วมหลักสูตรการฝึกอบรมผ่านระบบเครือข่ายของบริษัท เขาก็กำลังทำงานกับเอกสารการฝึกอบรมแบบ Real - Time แม้แต่พนักงานในหลายหน่วยงานที่ Access เครือข่ายอินเทอร์เน็ต เพื่อบันทึกเวลาการลาพักผ่อน ในฐานะข้อมูลที่ตั้งอยู่บางแห่งบนระบบเครือข่ายขององค์กร ก็สามารถเรียกว่าเป็นการสื่อสารแบบ Real - Time ได้เช่นกัน แม้ว่าจะเป็นการสื่อสารระหว่างคนกับการบันทึกบนฮาร์ดดิสก์ แต่ก็สามารถปรับปรุงฐานข้อมูลการลาพักผ่อนของพนักงานให้ทันสมัยได้ในทันทีทันใด หรือเร็วที่สุดเท่าที่จะเป็นไปได้ ระบบอาจล่าช้าเพียงเล็กน้อยถ้ากำลังบันทึกข้อมูลอยู่ หรือมีการเชื่อมต่อเข้ามาเป็นจำนวนมากอย่างต่อเนื่อง

คุณสามารถจินตนาการ ได้ว่า ระบบเครือข่ายแบบ Real - Time ต้องอาศัยเทคโนโลยีมากมายตั้งแต่ระบบปฏิบัติการเครือข่าย ไปจนถึงซอฟต์แวร์ของเครื่องเซิร์ฟเวอร์ที่มีประสิทธิภาพ ซอฟต์แวร์ที่ทำให้เครื่องไครเอนต์แสดงภาพและเสียง ระบบฐานข้อมูล และโปรแกรมประยุกต์อื่นที่สนับสนุนเครือข่ายอินเทอร์เน็ต รวมทั้งต้องอาศัยความสามารถในการสื่อสารสมัยใหม่และเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการส่งสัญญาณข้อมูลกับโปรโตคอลที่ทำให้สัญญาณเรียบ และจัดส่งข่าวสารไปยังเครื่องไคลเอนต์เครื่องเดียวหรือหลายเครื่องในเวลาเดียวกัน

2.2.2 การสื่อสารแบบ Asynchronous

ถึงแม้ว่าระบบเครือข่ายแบบ Real – Time เริ่มที่จะได้รับความนิยมเพิ่มขึ้น แต่ความสามารถในการโต้ตอบผ่านการสื่อสารแบบ Asynchronous ก็ทำให้ผู้ที่ไม่สมหวังจากการพ่วงต่อสายโทรศัพท์หรือผู้ที่วิ่งขึ้นลงไปตามห้องต่างๆ เพื่อทิ้งข้อความไว้บนโต๊ะทำงานของผู้อื่น ต้องตระหนักในคุณค่าของการประหยัดเวลาและความสะดวกสบายที่เสนอให้ การสื่อสารด้วยระบบเครือข่ายแบบ Asynchronous ได้แก่ การส่งจดหมายอิเล็กทรอนิกส์หรืออีเมลไปยังผู้ร่วมงาน ในระหว่างที่รอการตอบรับคุณก็ทำงานได้อย่างต่อเนื่อง คุณอาจคาดหวังว่าอาจได้รับคำตอบในทันทีทันใดแต่เมื่อคุณไม่ได้รับคำตอบ อารมณ์ โมโหของคุณก็จะรุนแรงน้อยกว่าที่ผู้อื่นทำให้เกิด นอกจากนี้ผู้ร่วมงานของคุณก็มีเครื่องมือที่ช่วยในการตอบรับได้ตามต้องการ ซึ่งเหมาะกับตารางเวลาการทำงานด้วย และอาจให้ผู้อื่นแจ้งกำหนดเวลาในการรวบรวมข่าวสาร เพื่อขบคิดในเรื่องที่ต้องใช้ความคิดร่วมกันหรือเลื่อนการเผชิญสถานการณ์ที่คาดว่าจะทำความตกลงได้ดีกว่าในภายหลังได้

ระบบเครือข่ายแบบ Asynchronous เข้ามีบทบาทในขณะที่คุณใช้โปรแกรมการจัดการเอกสารการกำหนดแนวทางการใช้งบประมาณ หรือทำรายงานให้กับบุคคลต่างๆ แล้วให้พวกเขาเหล่านั้นวิจารณ์ ซึ่งแต่ละคนก็สามารถทำงานกับเอกสารนั้นได้อย่างอิสระ เมื่อเสร็จก็สามารถส่งกลับมาที่คุณโดยระบบอิเล็กทรอนิกส์ได้ ด้วยความช่วยเหลือของระบบเครือข่ายคุณสามารถก้าวหน้าไปได้อีกขั้น โดยการส่งเอกสารที่แก้ไขปรับปรุงแล้วผ่านระบบเครือข่ายไปยังเครื่องพิมพ์ที่ได้รับการแบ่งปันการใช้ไว้แล้ว แม้ว่าขั้นตอนในการโต้ตอบของอุปกรณ์เหล่านี้ไม่ได้เป็นแบบ Asynchronous แต่ถ้าหากเครื่องพิมพ์ยังไม่ว่าง งานของคุณจะเข้าคิวตามลำดับการรับข้อมูลในการพิมพ์

ระบบเครือข่ายแบบ Asynchronous เหมือนกับระบบเครือข่ายแบบ Real – Time คือต้องอาศัยฮาร์ดแวร์ระบบสายเคเบิลและซอฟต์แวร์ ตั้งแต่ระบบปฏิบัติการของเครื่องไคลเอนต์ซึ่งได้รับการออกแบบมาให้ใช้บนระบบเครือข่าย และแปลงข่าวสารให้ส่งผ่านสายของระบบเครือข่ายได้ นอกจากนี้ข่าวสารที่เดินทางไปและกลับระหว่างบุคคลต่างๆ ทำให้ระบบเครือข่ายต้องการเทคโนโลยีที่ไม่เพียงแลกเปลี่ยนไฟล์จากที่หนึ่งไปยังอีกที่หนึ่งเท่านั้น แต่ยังคงมีการรวบรวมรายการเพื่อความสะดวกต่อการใช้ เช่น จัดให้มีเครื่องเซิร์ฟเวอร์บริการไคลเอนต์ที่มีการติดตามชื่อของผู้ใช้ และแอคแคสของเครือข่ายในเครื่องคอมพิวเตอร์เหล่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 การแบ่งปันการใช้และการแจกจ่าย

ระบบเครือข่ายยังสนับสนุนให้มีการแบ่งปันการใช้ที่แตกต่างกัน 2 ประเภท คือ การแบ่งปันการใช้ไฟล์ (File Sharing) และการแบ่งปันการใช้ทรัพยากร (Resource Sharing)

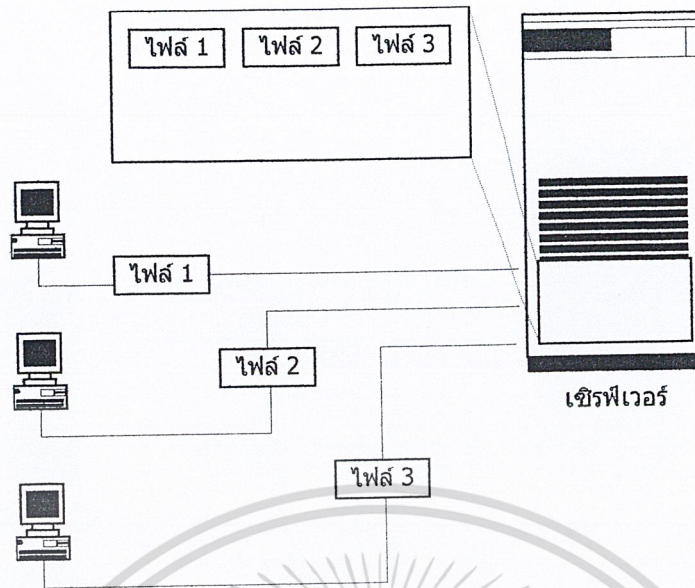
2.3.1 การแบ่งปันการใช้ไฟล์

การแบ่งปันการใช้ไฟล์ จะครอบคลุมเนื้อหาทุกอย่างที่ได้รับการตั้งชื่อ และบันทึกไว้ใช้ในอนาคตไฟล์เอกสาร เช่น จดหมายและรายงาน เป็นตัวอย่างที่เห็นได้อย่างเด่นชัด แต่ก็มีไฟล์แบบอื่นได้แก่ รูปภาพ เว็บเพจ บันทึกฐานข้อมูล ตารางข้อมูล สไลด์นำเสนอ และการรวบรวมข้อมูลแบบอื่น ถ้าสามารถจัดเก็บไฟล์ไว้บนระบบเครือข่าย และเชื่อมต่อกับคอมพิวเตอร์เครื่องหนึ่งไปยังเครื่องอื่น หรือการ Access โดยเครื่องคอมพิวเตอร์ใดๆ ที่ได้รับสิทธิในการใช้ก็นับว่าเป็นความเสมอภาคสำหรับการแบ่งปันการใช้ไฟล์ ข่าวสารที่บรรจุอยู่ในไฟล์อาจจะมีให้สำหรับทุกคน หรือจำกัดให้มีเฉพาะกลุ่มขนาดเล็ก หรือแม้แต่ใส่เข้าไปในระบบเครือข่ายเพื่อบุคคลเพียงผู้เดียว ซึ่งไม่ว่าจะเป็นเช่นใด การใช้ระบบเครือข่ายก็เพื่อที่จะสื่อสารกัน แบ่งปันการใช้และกระจายข่าวสารระหว่างเครื่องคอมพิวเตอร์ในรูปแบบของไฟล์

จาก ARPANET จนถึงปัจจุบัน การแบ่งปันการใช้ไฟล์เป็นฟังก์ชันที่สำคัญในระบบเครือข่าย และเป็นแรงผลักดันอยู่เบื้องหลังของการพัฒนาเทคโนโลยีในการสื่อสารระหว่างเครื่องคอมพิวเตอร์ให้เป็นอย่างดีราบเรียบที่สุดเท่าที่จะเป็นไปได้ แม้แต่การสื่อสารระหว่างแพลตฟอร์มที่แตกต่างกัน เช่น เครื่องแมคอินทอช กับเครื่องของวินโดวส์ หรือ อินเทล ส่วนความสามารถในการแลกเปลี่ยนข้อมูลคือผลของ โปรโตคอล โครงสร้างระบบเครือข่ายเทคโนโลยีการสื่อสาร

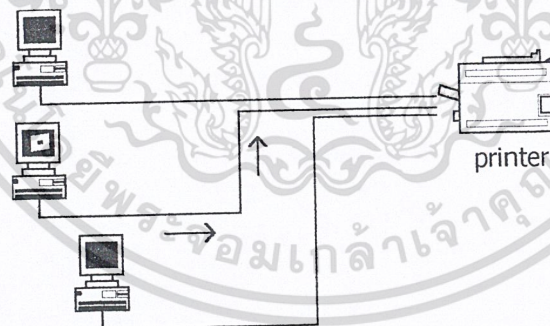
2.3.2 การแบ่งปันการใช้ทรัพยากร

นอกเหนือจากการแบ่งการใช้ไฟล์แล้ว ระบบเครือข่ายยังจัดให้มีประโยชน์อีกอย่างหนึ่งคือการแบ่งปันการใช้ทรัพยากร ซึ่งโดยทั่วไปเข้าใจกันว่าเป็นส่วนของฮาร์ดแวร์ ที่ติดตั้งไว้สำหรับใช้บนระบบเครือข่าย ตัวอย่างเช่น ฮาร์ดไดรฟ์ของระบบเครือข่ายที่ทำการแบ่งปันแล้ว จะเป็นทรัพยากรบนระบบเครือข่ายที่มีไว้สำหรับจัดเก็บข้อมูลของบุคคล ซึ่งได้รับการจัดสรรเนื้อที่ว่างบนไดรฟ์ หรือสามารถใช้เป็นที่อยู่ชั่วคราวสำหรับการโยกย้ายไฟล์จากคอมพิวเตอร์เครื่องหนึ่ง ไปยังอีกเครื่องหนึ่งดังรูป



รูปที่ 2.1 การแบ่งปันการใช้งานฮาร์ดดิสก์

เช่นเดียวกัน เครื่องพิมพ์บนระบบเครือข่ายก็เป็นทรัพยากรที่อาจถูกใช้ หรือแบ่งปันการ
ใช้กับผู้ที่กำหนดตำแหน่งของเครื่องพิมพ์บนเครือข่าย ไปยังคำสั่งการพิมพ์ในโปรแกรมการ
จัดการเอกสาร ตารางข้อมูล ฐานข้อมูล หรือโปรแกรมประยุกต์อื่นๆ



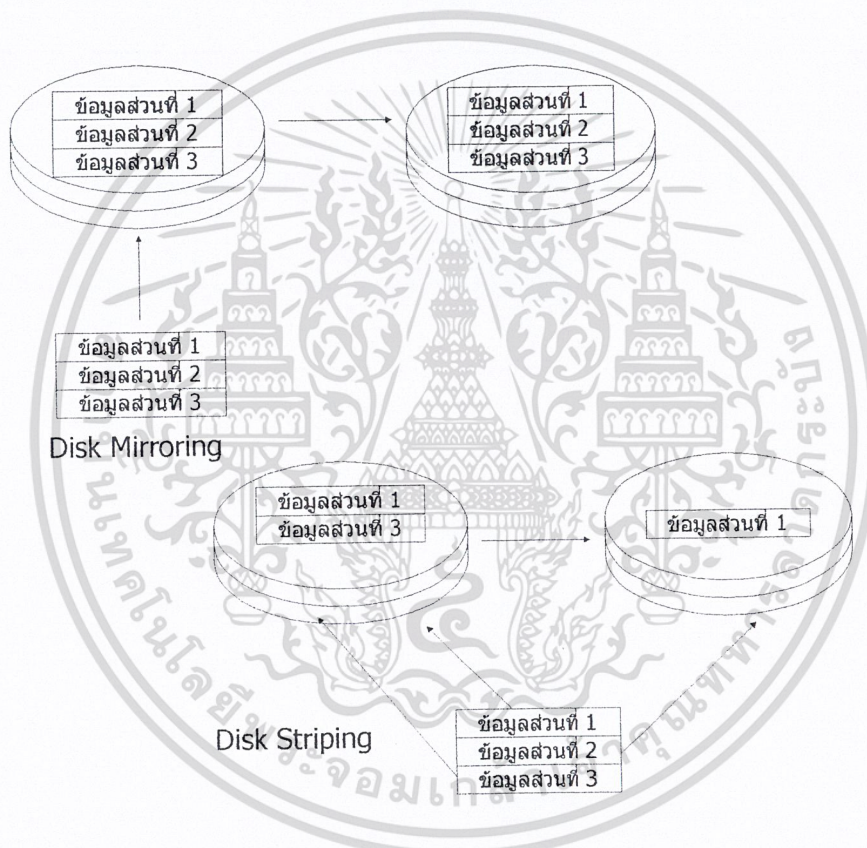
รูปที่ 2.2 การต่อเครื่องพิมพ์บนระบบเครือข่าย

2.4 ความปลอดภัยบนระบบเครือข่าย

ความปลอดภัย (Security) จะครอบคลุมขอบเขตของเนื้อหาเป็นจำนวนมาก บ้างก็เป็น
พื้นฐานของซอฟต์แวร์ และเกี่ยวข้องกับการรักษาความสมบูรณ์ของระบบเครือข่าย ทรัพยากรของ
ระบบเครือข่าย และอุปกรณ์ระบบเครือข่าย

ความปลอดภัยในที่นี้หมายถึง ความมั่นใจว่า โปรแกรมจะสามารถ Access หน่วยความจำ
ที่ไม่แน่ใจว่าถูกใช้โดยระบบปฏิบัติการหรือไม่ และคือกรณีที่โปรแกรมหนึ่งๆ ไม่สามารถเข้าไปยุ่ง
เฝ้าระวังใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือทำลายข้อมูลของโปรแกรมอื่น เรื่องอื่นที่เกี่ยวข้องคือการเอาใจใส่ระบบเครือข่ายและข้อมูลในระบบเครือข่ายให้อยู่ในสภาพดี โดยวัดจากการป้องกันอัคคีภัยและการสำรองข้อมูลตามระยะเวลาเกณฑ์อีกอย่างหนึ่งของความปลอดภัย ได้แก่ กลยุทธ์ในการรักษาความปลอดภัยหลายระดับที่เรียกว่า RAID (Redundant Array of Inexpensive Disks) ซึ่งสนใจการป้องกันข้อมูลที่เก็บอยู่ในดิสก์ โดยการจัดวางตำแหน่งของข้อมูลที่สำคัญมาจัดเก็บไว้ในระบบเครือข่าย RAID ซึ่งถูกใช้บ่อยครั้งในการอธิบายเกี่ยวกับความสมบูรณ์ของระบบเครือข่ายโดยมีพื้นฐานการป้องกันข้อมูลโดยวิธีที่เรียกว่า Mirroring (ทำข้อมูลซ้ำ) หรือแบ่งการเก็บข้อมูลไว้บนดิสก์หลายตัวดังรูปที่ 2.3 เพื่อให้มั่นใจว่าข้อมูลทั้งหมดจะไม่สูญหาย หากดิสก์ตัวใดตัวหนึ่งเกิดการเสียหายขึ้น



รูปที่ 2.3 กลยุทธ์ในการรักษาความปลอดภัย

อย่างไรก็ตามความปลอดภัยส่วนใหญ่จะคำนึงถึงการป้องกันผู้ใช้ที่ไม่ได้รับสิทธิใช้ Access เข้ามายังระบบเครือข่ายหรือทรัพยากรของระบบเครือข่าย นี้คือการรักษาความปลอดภัยที่ระบบเครือข่ายต้องมี และ ดำเนินการ โดยใช้รหัสผ่าน (Password) , การกำหนดสิทธิการใช้ (Permission) , Firewall , เครื่อง Proxy Server และกลยุทธ์อื่นๆ ที่ช่วยในการระบุผู้ใช้ที่ถูกต้อง และจำกัดหรือกั้นกรองผู้ใช้ที่ไม่มีสิทธิในการใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1 การ Access เข้าไปยังระบบเครือข่าย

เกือบจะทุกคนทราบดีอยู่แล้วว่า ผู้ที่ได้รับสิทธิในการที่จะ Access เข้าไปยังการบริการของระบบเครือข่ายหรือทรัพยากรของระบบเครือข่ายได้โดยเริ่มติดต่อกับระบบเครือข่ายด้วยการ ล็อกออน (log on) ซึ่งกระบวนการล็อกออนจะต้องให้ข้อมูลชื่อผู้ใช้หรือ Username (ชื่อที่คุณคุ้นเคยและแจกจ่ายให้กับระบบเครือข่าย) และรหัสผ่านหรือ Password (รหัสส่วนตัวที่ไม่จำเป็นต้องแสดงให้เห็นหรือเป็นชื่อพิสูจน์ตัวตนต่อระบบเครือข่าย) กับระบบเครือข่าย ด้วยเหตุที่ชื่อผู้ใช้อาจจะถูกแจกจ่ายอย่างอิสระรหัสผ่านจึงเป็นการป้องกันขั้นแรกในการเข้ามาโดยที่ไม่ได้รับสิทธิการใช้แม้ว่าจะเป็นไปได้ที่จะเจาะเข้าไปหรือขโมย แต่รหัสผ่านก็มีความสำคัญมากจนกระทั่งไม่แสดงให้เห็นบนจอภาพแม้แต่ในขณะที่พิมพ์ โดยทั่วไปจะถูกจัดเก็บด้วยการเข้ารหัส (Encrypt) ให้ไม่สามารถอ่านได้ เพื่อช่วยรักษาความปลอดภัย

2.4.2 การ Access เข้าไปยังทรัพยากร

ถึงแม้ว่าผู้ใช้ทุกคนที่ล็อกออนเข้ามาในระบบเครือข่าย จะเป็นผู้ที่ได้รับสิทธิในระบบเครือข่ายอย่างแท้จริง แต่ก็ยังมีคำถามอยู่ว่าจะสามารถเชื่อมต่อ ไปยังไฟล์อะไร แหล่งทรัพยากรใด และแม้แต่เซิร์ฟเวอร์เครื่องใดได้บ้าง? ระบบเครือข่ายต้องสามารถแยกแยะผู้ใช้เพื่อพิจารณาว่า นาย ก. สามารถที่จะ Access ไฟล์เงินเดือนได้หรือไม่ หรือ นาย ข. สามารถที่จะแก้ไขไฟล์ราคาสินค้าของบริษัทได้หรือไม่ เป็นต้น

เพื่อควบคุมการ Access ของผู้ใช้ที่ได้รับสิทธิในการใช้ระบบเครือข่าย ระบบเครือข่ายจะต้องอาศัยอุปกรณ์อย่างใดอย่างหนึ่งใน 2 อย่างนี้ คือการใช้รหัสผ่านเข้าไปยังทรัพยากร และการกำหนดสิทธิการใช้ (Access Permission) ด้วยการกำหนดรหัสผ่านให้กับแหล่งทรัพยากร เช่น โฟลเดอร์หรือเครื่องพิมพ์ทำให้เจ้าของหรือผู้จัดการแหล่งทรัพยากรนั้นสามารถควบคุมการ Access โดยจำกัดการแจกจ่ายรหัสผ่านให้กับผู้ที่ได้รับการเลือกสรรว่าต้องการทรัพยากรนั้น และนอกเหนือจากการกำหนดรหัสผ่าน เจ้าของแหล่งทรัพยากรนั้นๆ ก็อาจจะต้องอาศัยการกำหนดสิทธิการใช้ ซึ่งทำให้ผู้ใช้สามารถ (Read) หรือ (Write) หรือทำทุกอย่าง (Full) กับข้อมูลที่ Access เข้าไปในบางกรณีคนที่รู้รหัสผ่านได้รับการอนุญาตให้ Access เข้าไปยังแหล่งทรัพยากร ผู้ใช้ก็อาจจะได้รับอนุญาตให้ Access เพียงอย่างเดียว เฉพาะบุคคลใช้แต่ละคนอาจจะมีสิทธิในการ Access ที่แตกต่างกัน ตัวอย่างเช่น Joe อาจจะได้รับสิทธิในการอ่าน โฟลเดอร์เอกสารสำคัญเพียงอย่างเดียว แต่ Jean อาจได้รับการอนุญาตให้อ่านและแก้ไขปรับปรุงเอกสารนั้น

2.4.3 ควบคุมการ Access จากภายนอกองค์กร

ถ้าการใช้ระบบเครือข่ายถูกจำกัด โดยการเชื่อมต่อระหว่างคอมพิวเตอร์ด้วยสายเคเบิล การใช้รหัสผ่าน และการควบคุมการ Access ระบบเครือข่ายก็จะเป็นผู้ชนะด้านความปลอดภัย อย่างไรก็ตามในปัจจุบันการ Access ระบบเครือข่ายมักจะมีความต้องการมาจากภายนอกโดยใช้โมเด็มเอกสารนี้เป็นเอกสารที่ส่งมาให้อำนาจบริการใช้งาน ซึ่งผู้ใช้ทุกคนสามารถเข้าถึงได้โดยไม่ต้องทำการติดต่อผ่านสายโทรศัพท์ ตัวอย่างเช่น ถ้ามีผู้ใช้หนึ่งคนติดต่อสื่อสารมาจากบ้านอาจจะต้องการทำธุรกรรมไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เชื่อมต่อบ่อยครั้งในการรับและส่งอีเมล บุคคลแปลกหน้าที่ต้องการค้นหาข้อมูลบนเว็บไซต์บริษัทหรือบุคคลกึ่งภายนอกเช่น ผู้แทนจำหน่าย หรือหุ้นส่วนธุรกิจ อาจมีความต้องการใช้ทรัพยากรเช่นเดียวกับผู้ที่เกี่ยวข้องกับธุรกิจที่ทำกับองค์กร

ในกรณีแรก ผู้ใช้ระบบเครือข่ายที่มีสิทธิในการใช้ จะเชื่อมต่อด้วยการหมุน โทรศัพท์เข้ามาจากภายนอก การรักษาความปลอดภัยสามารถทำได้โดยติดตั้งให้เครื่องเซิร์ฟเวอร์เครื่องหนึ่งหรือมากกว่า ทำหน้าที่เสมือน “ ประตูเข้า ” ผู้ระบบเครือข่ายซึ่งทำให้ผู้ใช้ Access เข้ามายังระบบเครือข่ายได้ จัดเครื่องเซิร์ฟเวอร์เหล่านี้ให้มีวิธีพิเศษสำหรับตรวจสอบว่าผู้ที่เรียกเข้ามามีสิทธิในการใช้ วิธีหนึ่งที่ก่อให้เกิดกระบวนการนี้เรียกว่า กลไกการเรียกกลับ หรือcallback เครื่องเซิร์ฟเวอร์ซึ่งจะตรวจสอบชื่อและรหัสผ่านของผู้เรียกเข้ามา จากนั้นก็ยกเลิกการเชื่อมต่อแล้วจึงเรียกกลับไปยังหมายเลขที่กำหนดไว้ล่วงหน้าในกรณีที่ผู้ใช้เรียกมาจากสถานที่ต่างๆ ระบบเครือข่ายจะเรียกกลับไปยังหมายเลขที่ผู้แจ้งให้ระบบเครือข่ายทราบในการเรียกครั้งแรก ในการขยายความปลอดภัยในสถานการณ์เช่นนี้ การส่งสัญญาณข้อมูลรวมทั้งรหัสผ่านของผู้ใช้ อาจจะต้องเข้ารหัสข้อมูลสำหรับส่งผ่านสายโทรศัพท์

ในกรณีบุคคลแปลกหน้า Access เข้ามายังเว็บไซต์บนอินเทอร์เน็ตขององค์กร วิธีป้องกันที่แนะนำให้ใช้เรียกว่า Firewall ซึ่งมีลักษณะทางกายภาพเหมือนกำแพงป้องกันไฟที่มีวัตถุประสงค์ในการป้องกันไม่ให้ไฟลุกลามข้ามกำแพงจากด้านหนึ่งไปยังอีกด้านหนึ่งของอาคาร Firewall ของระบบเครือข่ายก็เป็นรูปแบบการคิดขวาง ระหว่างระบบเครือข่ายขององค์กรกับเครือข่ายอินเทอร์เน็ต โดยที่อาจจะเป็นฮาร์ดแวร์ ซอฟต์แวร์ หรือใช้ทั้งสองอย่างร่วมกัน ทั้งยังช่วยตรวจสอบข้อมูลข่าวสารที่เดินทางเข้ามายังระบบเครือข่าย และจะปฏิเสธข้อมูลข่าวสารที่ไม่ได้รับอนุญาตในขณะที่รับส่งข้อมูลข่าวสารที่ได้รับอนุญาตเข้ามาได้ ด้วยวิธี Firewall จะอนุญาตให้การสื่อสารเป็นไปอย่างอิสระ ในขณะที่แยกระบบเครือข่ายออกจากระบบเครือข่ายอินเทอร์เน็ตอย่างเด็ดขาด

อย่างไรก็ตาม จะไม่มีการส่งข้อมูลจากระบบเครือข่ายภายในองค์กรกับระบบเครือข่ายอินเทอร์เน็ตผ่านจากภายนอกสู่ภายในมากไปกว่านั้นผู้ใช้ระบบเครือข่ายยังสามารถ Access เข้าไปยังระบบเครือข่ายอินเทอร์เน็ตจากระบบเครือข่ายภายในองค์กรได้ ในกรณีนี้เพื่อให้มั่นใจในความปลอดภัย องค์กรอาจจะอาศัยรูปแบบของ Firewall ที่เรียกว่า Proxy server ที่กั้นการส่งข้อมูลในทางตรงกันข้าม โดยติดตั้งไว้สำหรับการเชื่อมต่อระบบเครือข่ายอินเทอร์เน็ต ตัวอย่างเช่น เครื่องคอมพิวเตอร์ในระบบเครือข่ายเชื่อมต่อไปยังเว็บไซต์บนเครือข่ายอินเทอร์เน็ตโดยอาศัยการร้องขอไปยังเครื่อง Proxy server จากนั้นเครื่อง Proxy server ก็จะถ่ายทอดคำร้อง ไปยังเว็บไซต์ และถ่ายทอดการตอบรับกลับไปยังเครื่องคอมพิวเตอร์ที่ทำการร้องขอ ในขณะที่ Firewall เลือกลงเลือกสัญญาณข้อมูลที่ผ่านเข้ามายังระบบเครือข่ายเครื่อง Proxy server ก็จัดให้ผู้ใช้ระบบเครือข่ายมีการ Access เข้าไปยังเครือข่ายอินเทอร์เน็ตในขณะที่แยกโลกภายนอกไว้ต่างหาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 การขนส่งข้อมูลบนระบบเครือข่าย

ในบางครั้งอาจจะมีการเคลื่อนย้ายข้อมูลบนระบบเครือข่าย

2.5.1 แพ็กเก็ตข้อมูล

ระบบเครือข่ายทำงานได้อย่างรวดเร็วมาก เมื่อเปรียบเทียบกับความเร็วในการทำงานของมนุษย์ ระบบเครือข่ายเปรียบเสมือน ทางด่วนที่เร็วเกือบเท่าความคิด แต่มีขีดจำกัดสูงสุดของปริมาณข้อมูลที่จะส่ง โดยการใช้แบนด์วิดธ์อย่างจำกัด (เป็นอัตราการขนส่งข้อมูล) ดังนั้นการที่จะทำให้การส่งข้อมูลเป็นไปได้อย่างรวดเร็วก็ต้องทำการแบ่งข้อมูลที่จะทำการส่งเป็นชิ้นเล็กๆ เรียกว่า แพ็กเก็ต (Packet)

ส่วนประกอบของแพ็กเก็ต

แพ็กเก็ตเป็นก้อนของข้อมูลและ ประกอบด้วยข่าวสารที่บอกว่า แพ็กเก็ตนั้นมาจากไหน จะถูกส่งไปที่ไหน จะตรวจสอบความผิดพลาดได้อย่างไร และจะประกอบเข้าด้วยกันให้เป็นไฟล์หรือข้อมูลเดิมได้อย่างไร ดังนั้นเพื่อที่จะบรรจุข่าวสารที่กล่าวมาข้างต้น แพ็กเก็ตจึงประกอบด้วย 3 ส่วนใหญ่ๆ คือ ส่วนหัว (Header) ส่วนข้อมูล (Data หรือ Body) และส่วนต่อท้าย (Trailer)



รูปที่ 2.4 แสดงส่วนประกอบของแพ็กเก็ต

1. ส่วนหัว (Header) เป็นส่วนที่อยู่หน้าก่อนข้อมูลในแพ็กเก็ต ซึ่งจะประกอบด้วยข่าวสารเกี่ยวกับที่อยู่ของแหล่งข้อมูล จุดหมายปลายทาง รวมทั้งข้อมูลการควบคุมและกำหนดจังหวะเวลา เพื่อให้มั่นใจว่าแพ็กเก็ตนั้นจะถูกส่งอย่างเหมาะสม
2. ส่วนข้อมูล (Data หรือ Body) บรรจุส่วนของข้อมูลจริงที่จะทำการส่ง ส่วนนี้ของแพ็กเก็ตโดยปกติจะมีขนาดระหว่าง 512 ไบต์ ถึง 4 กิโลไบต์ ขึ้นอยู่กับระบบเครือข่าย
3. ส่วนต่อท้าย (Trailer) เป็นส่วนของแพ็กเก็ตที่บรรจุข่าวสาร ที่เกี่ยวกับการตรวจสอบข้อผิดพลาดในการส่งข้อมูล ซึ่งจะทำให้คอมพิวเตอร์ฝ่ายผู้รับทราบว่าข้อมูล ได้มาถึงอย่างสมบูรณ์

2.5.2 การตรวจสอบความผิดพลาด

การพิสูจน์ความผิดพลาดในการส่งข้อมูล ระบบเครือข่ายโดยทั่วไปจะต้องอาศัยกระบวนการการตรวจสอบความผิดพลาด ที่เรียกว่า Cyclical Redundancy Check (CRC) ซึ่งเครื่องคอมพิวเตอร์ผู้ส่งจะทำการคำนวณค่าๆ หนึ่ง บนพื้นฐานของข้อมูลในแพ็กเก็ตและเครื่องคอมพิวเตอร์ผู้รับก็จะทำการคำนวณค่าๆ นั้นเมื่อได้รับแพ็กเก็ต ถ้าได้ผลลัพธ์เหมือนกันก็ให้สันนิษฐานว่าไม่มีความผิดพลาดเกิดขึ้นในระหว่างการส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.3 แพ็กเก็ตกับการ์ดเชื่อมต่อระบบเครือข่าย

ระบบเครือข่ายจะจัดการกับข้อมูลที่วิ่งผ่านอย่างรวดเร็วเพื่อให้ได้แพ็กเก็ตที่เรียบร้อย คือได้รับการกำหนดแอดแตรสส่งออกไปและ กำหนดแอดแตรสปลายทางของผู้รับ โดยการ์ดเชื่อมต่อระบบเครือข่ายของเครื่องคอมพิวเตอร์ที่เชื่อมต่อในระบบเครือข่าย

เริ่มต้นที่ข้อมูลจะเดินทางภายในเครื่องคอมพิวเตอร์ระหว่างอุปกรณ์ต่างๆ กับโปรเซสเซอร์ผ่านเส้นทางที่เรียกว่า “บัส” ซึ่งประกอบด้วยหลายช่องทาง (สายไฟ) ที่วางเรียงกัน เช่นเดียวกับถนนซึ่งมีหลายเลน เครื่องคอมพิวเตอร์จะใช้พื้นฐานของไมโครโปรเซสเซอร์ ซึ่งโดยปกติจะมีขนาด 16 หรือ 32 บิต (และในอนาคตอันใกล้จะมีขนาด 64 บิต) แทนอักขระข้อความ หรือแอดแตรสของหน่วยความจำ การเดินทางของข้อมูลที่วางเรียงติดต่อกันภายในเครื่องคอมพิวเตอร์ในลักษณะนี้เรียกว่าการส่งสัญญาณข้อมูลขนาน (Parallel Transmission) เนื่องจากมีการแลกเปลี่ยนบิตเป็นกลุ่มพร้อมกัน และคงที่คุณทราบคืออยู่แล้วว่าถ้าจำนวนบิตในแต่ละกลุ่มยิ่งมาก เครื่องคอมพิวเตอร์ก็สามารถเคลื่อนย้ายข้อมูลได้ปริมาณมากในครั้งหนึ่งๆ ดังนั้นข้อมูลจะเดินทางจากที่หนึ่งไปยังอีกที่หนึ่งในเครื่องคอมพิวเตอร์ได้อย่างรวดเร็วขึ้น

อย่างไรก็ตามเมื่อข้อมูลเคลื่อนย้ายออกจากเครื่องคอมพิวเตอร์ไปยังสายเคเบิลระบบเครือข่ายสัญญาณจะต้องได้รับการเปลี่ยนแปลง ถึงแม้ว่าบิตต่างๆ ที่จะทำการส่งจะเป็นตัวแทนของข้อมูล แต่กลุ่มของบิตเหล่านั้นไม่สามารถเดินทางไปพร้อมกันแบบขนานได้ สายเคเบิลระบบเครือข่ายต้องการให้ข้อมูลเดินทางในลักษณะ บิตต่อบิต ที่เรียกว่า การส่งสัญญาณข้อมูลแบบอนุกรม (Serial Transmission) ซึ่งแตกต่างจากการเดินทางภายในเครื่องคอมพิวเตอร์ ผลก็คือถนนที่มีหลายเลนก็ต้องปรับให้มีเลนเดียว โดยที่ จำกัดความเร็วสูงสุดเพื่อให้เกิดความแน่นอน แต่นั่นก็เป็นการบังคับช่องทางมากขึ้น

อย่างไรก็ตามบิตเป็นสิ่งที่ไม่มีความรู้สึกนึกคิด จึงไม่สามารถตัดสินใจในการเดินทางด้วยตัวเอง และสามารถตกลงใจออกจากบัสในเครื่องคอมพิวเตอร์ เพื่อขึ้นไปติดอยู่ในแพ็กเก็ตระบบเครือข่ายได้ จึงต้องอาศัยการ์ดเชื่อมต่อระบบเครือข่าย ซึ่งเป็นแผงวงจรที่ทำหน้าที่เปลี่ยนรูปแบบการส่งสัญญาณข้อมูลของเครื่องคอมพิวเตอร์จากแบบขนานให้เป็นแบบอนุกรมและร้องขอข้อมูลจากเครื่องคอมพิวเตอร์ นอกจากนี้ยังทำหน้าที่กำหนดแอดแตรส จัดหีบห่อให้กับข้อมูลเคลื่อนย้ายแพ็กเก็ตบนระบบเครือข่าย และประสานในเรื่องของ ขนาด ความเร็ว และช่วงระยะเวลาของการส่งสัญญาณข้อมูล เพื่อให้มั่นใจว่ามีการแลกเปลี่ยนข้อมูลอย่างสมบูรณ์

ฝ่ายเครื่องผู้รับการ์ดเชื่อมต่อระบบเครือข่ายก็จะทำงานในทางกลับกัน โดยเฟ้าระบบเครือข่ายเพื่อมองหาและจับแพ็กเก็ตที่ส่งมายังแอดแตรสของตนเอง และแพ็กเก็ตที่ออกอากาศบนระบบเครือข่าย การ์ดระบบเครือข่ายจะถอดข่าวสารที่เป็นแอดแตรส และข่าวสารที่เกี่ยวข้อง มีความสัมพันธ์กับการเดินทางของข้อมูลออกจากแพ็กเก็ต สุดท้ายจึงจะส่งข้อมูลต่อไปยัง

เครื่องคอมพิวเตอร์ เพื่อให้โปรแกรมประยุกต์ที่ร้องขอหรือได้รับการออกแบบให้นำข้อมูลนั้นไปใช้ กระบวนการเหล่านี้จะถูกดำเนินการโดยการ์ดเลน

2.6 ชุดโปรโตคอล ทีซีพี / ไอพี (TCP / IP)

โปรโตคอล ทีซีพี / ไอพี (TCP / IP – Transmission Controller Protocol / Internet Protocol) เป็นกลุ่มโปรโตคอลที่พัฒนาขึ้นเพื่อให้คอมพิวเตอร์ใช้ทรัพยากรและบริการฟังก์ชันพื้นฐานสำหรับการใช้งานบนระบบสื่อสารข้อมูลคอมพิวเตอร์ได้ ข้อมูลที่ใช้ ทีซีพี/ไอพี นำส่งจะถูกแบ่งออกเป็นข้อมูลย่อยหลายๆ ส่วนเพื่อทยอยส่งไปตามลำดับเพื่อให้เหมาะสมกับระบบโครงข่ายในชั้นถัดไปที่อาจจะไม่สามารถส่งข้อมูลขนาดใหญ่ได้ทันที และเมื่อส่งไปถึงปลายทางก็จะรวบรวมข้อมูลนั้นกลับเป็นข้อมูลชุดเดิมอีกครั้ง

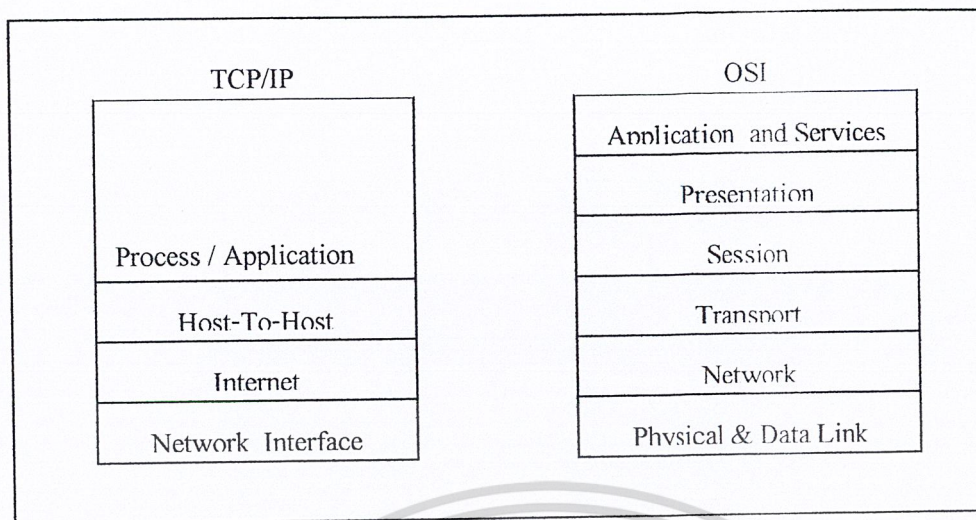
2.6.1 ชั้นต่าง ๆ ของ ทีซีพี / ไอพี (TCP / IP Layer)

การติดต่อสื่อสารของ ทีซีพี / ไอพี ถูกกำหนดให้มีการทำงานเป็นระดับชั้น (Layer) เพื่อให้มีการทำงานเป็นอิสระต่อกัน ในแต่ละระดับชั้นและเพื่อให้มีขั้นตอนการทำงานในการแลกเปลี่ยนข้อมูลระหว่างคอมพิวเตอร์เป็นไปอย่างถูกต้องดังนี้

- กำหนดรูปแบบข้อมูล
- จัดเตรียมชุดข้อมูล
- กำหนดเส้นทางการส่งข้อมูล
- กำหนดอัตราความเร็วในการส่งข้อมูล
- ทำการส่งข้อมูลผ่านตัวกลาง
- รวบรวมและจัดลำดับชุดข้อมูลที่ส่งมา
- ตรวจสอบว่ามีชุดข้อมูลซ้ำหรือไม่
- ตอบกลับไปให้ผู้ส่งรู้ว่าได้รับข้อมูลแล้ว
- ส่งผ่านข้อมูลไปให้ชั้นการทำงานถัดไป

เปรียบเทียบกับโมเดลอ้างอิงการเชื่อมต่อระบบเปิด (Open System Interconnection Reference Model : OSI-RM) ได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 แสดงการแบ่งระดับการทำงานของ ทีซีพี / ไอพี โดยอ้างอิงจากมาตรฐาน OSI

● **ชั้นเชื่อมต่อระบบโครงข่าย (Network Interface Layer)**

ชั้นนี้จะทำหน้าที่ในการสื่อสารข้อมูลทางกายภาพในระดับสัญญาณนำส่ง (สัญญาณทางลอจิก “ 0 ” หรือ “ 1 ”) ตัวอย่างของระบบที่ทำงานในชั้นนี้ เช่น ระบบโครงข่าย อีเทอร์เน็ตหรือระบบโครงข่ายแบบ Token Ring โดยจัดข้อมูลเป็นกลุ่มที่เรียกว่าเฟรม (Frame) โดยในส่วนของเฟรมจะบรรจุด้วยตำแหน่งต้นทางและปลายทาง ข่าวดสารที่ใช้ในการควบคุม และส่วนท้ายที่ใช้ในการตรวจสอบข้อผิดพลาดติดต่อดระดับล่างสุด โดยผ่านระบบสื่อสัญญาณต่างๆ ไปถึงอุปกรณ์เชื่อมต่อปลายทาง

● **ชั้นอินเทอร์เน็ต (Internet Layer)**

ชั้นนี้จะมีอินเทอร์เน็ตโปรโตคอล (Internet Protocol) หรือ IP คอยทำหน้าที่ส่งข้อมูลที่เรียกว่า “ ดาต้าแกรม ” (Datagram) โดยในชั้นนี้ IP จะทำงานโดยไม่มีการเชื่อมต่อก่อน (Connectionless) ดาต้าแกรมจะถูกจัดเส้นทางเป็นอิสระต่อกัน ไม่มีการรับประกันความถูกต้อง ความน่าเชื่อถือ หรือแม้แต่การจัดเรียงลำดับดาต้าแกรมให้อยู่ในลำดับที่ถูกต้อง

ข้อมูลจะส่งเข้าไปในระบบโครงข่ายโดยมี Router เป็นตัวจัดเส้นทางว่าข้อมูลจะไปถึงปลายทางอาจต้องผ่านตัวจัดเส้นทาง (Router) ของหลายโครงข่าย จึงต้องมีการผนวกหมายเลขเครื่องต้นทางและปลายทางเข้าไปในชุดข้อมูลเพื่อให้ Router รู้ข้อมูลที่ผ่านมาต้องการจะไปไหน ถ้าไม่ใช่หมายเลขโครงข่ายของตัวเองก็จะส่งต่อไปยังโครงข่ายอื่นแต่ถ้าใช่ก็จะส่งไปให้สมาชิกทั้งหมดของเครือข่าย

- **ชั้นโฮสต์ทูโฮสต์ (Host-to-Host Layer) – TCP และ UDP**

โปรโตคอลที่ทำงานในชั้นนี้มี 2 แบบ

- แบบที่เรียกว่า TCP จะทำงานแบบมีการเชื่อมต่อก่อน (Connection Orient) มีการส่งข้อมูลโดยรับประกันความน่าเชื่อถือและเรียงลำดับข้อที่ถูกต้อง โปรโตคอล TCP จะทำการเพิ่มส่วนหัวเรื่องสร้างเป็นเซ็กเมนต์

- แบบที่เรียกว่า UDP (Users Datagram Protocol) ซึ่งการทำงานไม่มีการเชื่อมต่อ ก่อน (Connectionless) และ ไม่มีการรับประกันความถูกต้องของข้อมูล

- **ชั้นโปรแกรมประยุกต์ (Application Layer)**

ชุดโปรโตคอล ทีซีพี / ไอพี จะมีโปรโตคอลในชั้น โปรแกรมประยุกต์ให้ใช้งานอยู่มาก ที่นิยมใช้กันมากและจัดเป็นบริการพื้นฐานของ ทีซีพี / ไอพี เช่น การล็อกอินระยะไกล (Remote Login) การส่งผ่านแฟ้มข้อมูล (Files Transfer) การส่งไปรษณีย์อิเล็กทรอนิกส์ (Electronic Mail) บริการเวิลด์ไวด์เว็บ (World Wide Web) เน็ตเวิร์ก ไฟล์ซิสเต็ม (NFS – Network File System) การพิมพ์ระยะไกล (Remote Printing) เป็นต้น

2.7 โปรโตคอลในชั้นอินเทอร์เน็ต

ทีซีพี / ไอพี เป็นโปรโตคอลที่ใช้สื่อสารข้อมูลกันในระบบโครงข่ายคอมพิวเตอร์โดยคอมพิวเตอร์แต่ละตัวจะส่งข้อมูลเข้าไปในระบบโครงข่าย จากนั้นจะใช้อุปกรณ์เชื่อมต่อระบบโครงข่ายที่เรียกว่าเราเตอร์ (router) ในการสื่อสารข้อมูลเข้าหากันซึ่งจะอาศัยการทำงานในชั้นอินเทอร์เน็ตของชุดโปรโตคอล ทีซีพี / ไอพี

2.7.1 อินเทอร์เน็ตโปรโตคอล (Internet Protocol)

อินเทอร์เน็ตโปรโตคอล หรือ ไอพี เป็นโปรโตคอลหลักที่ทำหน้าที่ส่งข้อมูลในชั้นอินเทอร์เน็ตนี้ ข้อมูลในส่วนของการทำงานในชั้นนี้จะเรียกว่า คาด้าแกรม (Datagram) ชั้นนี้ไม่มีการรับประกันว่าข้อมูลที่ผ่านจากไอพีนี้ขึ้นไปนั้นจะถูกต้องอาจมีการสูญหาย การซ้ำซ้อนของข้อมูล การไม่เรียงลำดับของข้อมูล ซึ่งจะเป็นหน้าที่ของโปรโตคอลชั้นบนต่อไปเป็นผู้จัดการ ไอพี จะไม่มีการตรวจสอบข้อผิดพลาดของข้อมูล จะมีเฉพาะผลรวมตรวจสอบส่วนหัวเท่านั้น ไม่มีการส่งซ้ำ (re-transmission) ไม่มีการควบคุมการรับส่ง (flow control)

ไอพี มีหน้าที่หลักคือรับข้อมูลจาก ทีซีพี หรือ ยูดีพี แล้วสร้างเป็นคาด้าแกรมจากนั้นจึงทำการค้นหาเส้นทาง ที่จะให้นำส่งคาด้าแกรมตัวนี้ไปยังแอดเดรสปลายทางต่อไป ซึ่ง คาด้าแกรมแต่ละตัวจะถูกจัดเส้นทางเป็นอิสระต่อกัน ในการหาเส้นทางนั้น ไอพี จำเป็นจะต้องมีความสามารถในการแลกเปลี่ยนข่าวสารที่ใช้กับโปรโตคอลตัวอื่นด้วยเช่นอาร์ไอพี

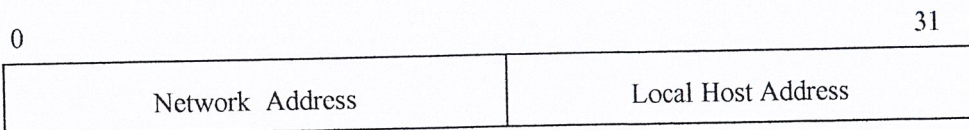
(RIP : Routing Information Protocol) จีจีพี (GGP : Galteway - Gateway Protocol) อีจีพี (EGP : External Gateway Protocol) เป็นต้น

การส่งข้อมูลของ ไอพี จำเป็นจะต้องบอกอินเตอร์เน็ตแอดเดรส ของเครื่องปลายทาง งานหลักของไอพี คือ ค้นหาเส้นทางเพื่อที่จะส่งคาต้าแกรมไปยังปลายทาง ในการที่จะให้อุปกรณ์ ระหว่างทาง (Intermedia system) ส่งคาต้าแกรมจะต้องมีข้อมูลในส่วนหัวของคาต้าแกรมเป็น ตัวบอก ในส่วนนี้จะบอกถึงอินเตอร์เน็ตแอดเดรสต้นทางและ ปลายทางขนาด 32 บิต โปรโตคอล นัมเบอร์ (Protocol Number) และผลรวมตรวจสอบตำแหน่งต้นทางและปลายทางใช้เพื่อบอกให้ ทั้ง 2 ฝ่ายทราบว่าข้อมูลจากที่ใด และจะไปไปที่ใด โปรโตคอลนัมเบอร์ใช้บอก ไอพี ว่าจะส่ง คาต้าแกรมไปยังชั้น ทีซีพี หรืออื่นๆ ผลรวมตรวจสอบใช้เพื่อตรวจสอบว่าข้อมูลของส่วนหัวไม่มี ข้อผิดพลาดระหว่างการส่งแฟล็ก (Flag) และแฟร็กเมนต์ออฟเซต (Fragment Offset) ใช้เพื่อ แบ่งคาต้าแกรมเป็นขนาดเล็กๆ เนื่องจากอาจจะส่งข้อมูลขนาดใหญ่เกินกว่าที่จะให้ระบบโครงข่าย นั้นส่งได้ทั้งหมดภายในการส่งครั้งเดียว เวลาคงอยู่เป็นตัวเลขที่ใช้บอกถึงช่วงเวลาที คาต้าแกรม ตัวนี้จะคงอยู่ในระบบโครงข่ายได้ คำนี้อาจลดลงเรื่อย ๆ เมื่อมีการประมวลผลคาต้าแกรมในแต่ละ ระบบ เมื่อค่านี้เป็น “ 0 ” คาต้าแกรมนี้ก็จะถูกกำจัดทิ้งไป เพื่อป้องกันการส่งข้อมูลวนในเน็ตเวิร์ก

2.7.2 ชุดโปรโตคอลในชั้นอินเตอร์เน็ต

อินเตอร์เน็ตแอดเดรสเป็นตัวเลขขนาด 32 บิตใช้บ่งบอกเลขประจำตัวของโฮสต์นั้น ๆ เพื่อความสะดวกในการอ่าน จึงมักจะเขียนอินเตอร์เน็ตแอดเดรสเป็นเลขฐานสิบจำนวน 4 ชุด แต่ละ ชุดจะแบ่งด้วยเครื่องหมายจุด (dot) เรียกกันว่า dotted decimal notation ค่า 32 บิตนี้จะถูกแบ่งออกเป็นค่า 8 บิตต่อ หนึ่งฟิลด์ เรียกกันว่า อ็อกเตท (octet) ซึ่งจะมีทั้งหมด 4 ฟิลด์ และกำหนดค่า ในแต่ละฟิลด์ ด้วยค่าเลขฐานสิบ เช่น อินเตอร์เน็ตแอดเดรสของระบบหนึ่งเป็น 1010 0001 1111 0110 0000 1010 0001 0101 จะให้ค่าในแต่ละอ็อกเตทคือ 161 246 10 21 ซึ่งเมื่อเขียนอยู่ในรูป dotted decimal notation จะได้เป็น 161.246.10.21

อินเตอร์เน็ตแอดเดรสขนาด 32 บิตนี้จะถูกมองเป็น 2 ส่วน ส่วนแรกเป็น หมายเลข เน็ตเวิร์ก (Network Number) ซึ่งเป็น โครงข่ายที่โฮสต์คอมพิวเตอร์นั้นเชื่อมต่ออยู่ และส่วนที่เหลือ เป็น หมายเลขท้องถิ่น (Local Host Address) หรือหมายเลขโฮสต์ของตัวเอง (Host Address)



รูปที่ 2.6 แสดงรูปแบบของอินเตอร์เน็ตแอดเดรส

ซึ่งจากอินเตอร์เน็ตแอดเดรสขนาด 32 บิตนี้ จะแบ่งระบบโครงข่ายออกเป็น 5 คลาสใหญ่ดังนี้

1. ระบบโครงข่ายใน Class A
2. ระบบโครงข่ายใน Class B
3. ระบบโครงข่ายใน Class C
4. ระบบโครงข่ายใน Class D
5. ระบบโครงข่ายใน Class E

2.8 ทฤษฎีการใช้งานของโปรแกรม Microsoft Visual Basic

2.8.1 รายละเอียดของส่วนประกอบต่าง ๆ ใน Visual Basic

- เมนูบาร์ (Menu Bar)

เป็นคำสั่งที่เราสามารถใช้งานได้ทั้งหมดใน VB

- ทูลบาร์ (Toolbar)

เป็นส่วนที่รวบรวมเอาไอคอนเล็กๆ มากมายเอาไว้ซึ่งแต่ละไอคอนเหล่านี้เปรียบเสมือนคำสั่งหนึ่ง ๆ ของเมนู นั่นคือไอคอนเหล่านี้ได้ถูกออกแบบมาเพื่อให้การใช้งานคำสั่งต่าง ๆ ทำได้รวดเร็วและมีลักษณะที่สื่อความหมายกับผู้ใช้มากขึ้น

- ทูลบ็อก (Toolbox)

เป็นแถบเครื่องมือที่ประกอบไปด้วยไอคอนต่าง ๆ ซึ่งเรียกว่า “คอนโทรล” (Control) โดยจะนำมาใช้ร่วมกับฟอร์มเพื่อสร้างจอภาพของโปรเจกต์ แต่ละคอนโทรลจะใช้เป็นเครื่องมือในการสร้างส่วนติดต่อกับผู้ใช้ (User Interface) โดยการคลิกเลือกที่คอนโทรลนั้นแล้ว นำมาวางลงบนฟอร์มด้วยการลากแล้ววาง (Drag and Drop) หรือ โดยการดับเบิลคลิกที่ไอคอนในทูลบ็อกซก็ได้

- หน้าต่างโปรเจกต์ (Project Explorer Window)

เป็นหน้าต่างที่รวบรวมรายชื่อของฟอร์ม โมดูลไฟล์ (Custom Control) โมดูล Class (Class Module) หรือไฟล์ทรัพยากร (Resource File) สำหรับการสร้างแอปพลิเคชันหนึ่ง ๆ ซึ่งการรวมเอาไฟล์เหล่านี้เข้าด้วยกันเพื่อสร้างแอปพลิเคชันภายใต้ VB เรียกว่า โปรเจกต์ (Project)

- หน้าต่างคุณสมบัติ (Properties Window)

เป็นหน้าต่างที่รวบรวมคุณสมบัติทั้งหมดของฟอร์มหรือคอนโทรลเอาไว้ ซึ่งคุณสมบัติทั้งหมดที่ปรากฏในหน้าต่างนี้ จะเป็นคุณสมบัติที่ผู้อ่านสามารถกำหนดค่าได้ในขณะกำลังออกแบบเมื่อผู้อ่านทำการแก้ไขคุณสมบัติต่าง ๆ ในหน้าต่างคุณสมบัตินี้ก็จะส่งผลต่อคอนโทรลตัวนั้นทันที ซึ่งบางคุณสมบัติสามารถแสดงให้เห็นการเปลี่ยนแปลงคุณลักษณะได้ทันที ส่วนคุณสมบัติบางอย่างจะแสดงผลให้เห็น ก็ต่อเมื่อผู้อ่านมีการรันแอปพลิเคชันเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ฟอรัม (Form)

เป็นหน้าต่างที่ใช้ในการออกแบบส่วนติดต่อกับผู้ใช้ของแอปพลิเคชันของเราเนื่องจาก Visual Basic ได้รับการออกแบบให้ผู้ใช้ นำมาออกแบบ แอปพลิเคชัน ในลักษณะ ของการสื่อสารด้วยรูป (Visual) ดังนั้นการสื่อสารทำงานต่างๆ ระหว่างแอปพลิเคชันกับผู้ใช้จะต้องกระทำผ่านฟอรัม โดยในแอปพลิเคชันหนึ่งสามารถมีได้หลายฟอรัม และภายในฟอรัมก็就会被ใช้ในการบรรจุคอนโทรลต่างๆ ฟอรัมจึงทำหน้าที่เป็นตัวบรรจุ (Container) และฟอรัมยังเป็นออบเจกต์ (Object) ตัวหนึ่งของ VB ที่อนุญาตให้ผู้ใช้แก้ไขคุณสมบัติได้ในขณะออกแบบจากหน้าต่างคุณสมบัติและสามารถควบคุมพฤติกรรมต่างๆ ของฟอรัมด้วยวิธีเดียวกับออบเจกต์อื่น ๆ

- หน้าต่างโค้ดอิดิเตอร์ (Code Editor Window)

เป็นหน้าต่างที่ใช้ในการพิมพ์คำสั่งโปรแกรมเข้าไปในแอปพลิเคชัน

- หน้าต่างฟอรัมเลย์เอาท์ (Form Layout Window)

เป็นหน้าต่างแสดงตำแหน่งฟอรัมบนจอ ช่วยให้เรากำหนดตำแหน่งของฟอรัมบนหน้าจอได้สะดวกขึ้น โดยการคลิกเมาส์ลากฟอรัมที่แสดงบนหน้าต่าง ให้เคลื่อนไปยังตำแหน่งที่เราต้องการ

- คอนโทรลของ VB

คอนโทรลเป็นเครื่องมืออย่างหนึ่งที่จะช่วยให้การสร้างแอปพลิเคชันด้วย VB ทำได้ง่ายและรวดเร็วยิ่งขึ้น แต่เมื่อได้นำไปวางบนฟอรัมแล้ว จะเรียกว่าออบเจกต์ (Object) ซึ่งจริงๆ แล้วคอนโทรลก็คือไฟล์ไคโนมิกลิงคไลบรารีทั่ว ๆ ไปของวินโดวส์ ไฟล์เหล่านี้จะมีนามสกุล .VBX หรือ .OCX ซึ่งเราเรียกว่า Custom Control ซึ่งไฟล์เหล่านี้จะถูกจัดเก็บในไดเรกทอรีย่อย System ของ Window

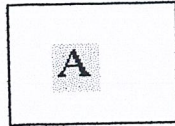
คอนโทรลพื้นฐานใน VB จะเป็นคอนโทรลติดต่อกับผู้ใช้ในการทำงาน พื้นฐานต่าง ๆ เช่น การแสดงข้อความและรับข้อความจากผู้ใช้ การแสดงรูปภาพ การแสดงตัวเลือก เป็นต้น ซึ่งจะกล่าวถึงเฉพาะคอนโทรลที่ได้นำมาใช้งานดังต่อไปนี้

- เท็กซ์บ็อกซ์ (Text Box)

เป็นคอนโทรลที่ผู้ใช้สามารถเติม แก้ไขข้อความในเท็กซ์บ็อกซ์นี้ได้โดยผ่านคีย์บอร์ดเข้าสู่โปรแกรม หรือแสดงข้อมูลต่างๆ ในโปรแกรมของเราได้

- เลเบล (Label)

เป็นคอนโทรลที่ใช้ในการแสดงข้อความตามที่เรต้องการ เพื่อแสดงข้อมูลบางอย่างแก่ผู้ใช้โดยผู้ใช้ไม่สามารถแก้ไขข้อความนั้นได้ในตอนรันโปรแกรม การแก้ไขข้อความในเลเบลจะต้องทำในขณะออกแบบ หรือในคำสั่งโปรแกรมตอนรันเท่านั้น



รูปที่ 2.7 แสดงคอนโทรล เลเบล ที่ใช้ในการแสดงข้อความ

- ปุ่มคำสั่ง (Command Button)

เป็นคอนโทรล ที่ใช้ในการทำงานอย่างหนึ่งตามที่เราต้องการ โดยเราสามารถกำหนดสิ่งที่เราต้องการให้ทำงานในอีเวนต์ Click คอนโทรลนี้จะอยู่ในทูลบ็อกซ์ ดังรูป การกำหนดข้อความที่แสดงบนคอนโทรล

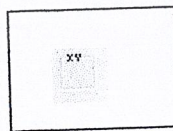
ในการกำหนดข้อความที่ต้องการแสดงผลบนคอนโทรล ให้เรากำหนดที่คุณสมบัติ Caption ของคอนโทรลในหน้าต่างคุณสมบัติ หรือในคำสั่งโปรแกรมก็ได้



รูปที่ 2.8 แสดงคอนโทรลปุ่มคำสั่ง Command Button

- เฟรม (Frame)

เป็นคอนโทรลที่ใช้ในการจัดกลุ่มคอนโทรลที่ต้องใช้ร่วมกัน ทำงานอย่างเดียวกันไว้ด้วยกันซึ่งโดยทั่วไป คอนโทรลเฟรมเป็นคอนโทรลที่ช่วยเพิ่มความเรียบร้อย และความสวยงามของโปรแกรม



รูปที่ 2.9 แสดงคอนโทรล Frame สำหรับใช้ในการจัดกลุ่มคอนโทรลอื่นๆ

- คอมโบบ็อกซ์ (ComboBox)

จะเป็นคอนโทรลที่ใช้ในการแสดงรายการที่ต้องการให้ผู้เลือกใช้รายการ หรือสามารถแก้ไขรายการที่เลือกได้คอนโทรลนี้จะใช้พื้นที่ในการวางคอนโทรลน้อยกว่าคอมโบบ็อกซ์ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้เฉพาะเอกสารที่สงวนไว้เท่านั้นไปรวมกันหน้าไปใช้ประโยชน์ด้านการค้า คอนโทรลนี้จะเหมือนกับคอนโทรลที่เก็บบ็อกซ์ และลิสต์บ็อกซ์รวมกัน ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 แสดงคอนโทรล ComboBox สำหรับแสดงรายการที่ต้องการเลือก

- คอนโทรลลิสต์บ็อก (ListBox)

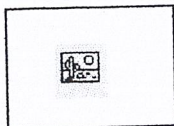
คอนโทรลลิสต์บ็อก (ListBox) เป็นคอนโทรลที่มีลักษณะการใช้งานในรูปแบบการแสดงผลข้อความหลาย ๆ บรรทัดภายในคอนโทรล ListBox และถ้ามีข้อความจำนวนมากจนไม่สามารถแสดงผลได้ในหน้าจอเดียว Visual Basic ก็จะเพิ่มแถบเลื่อน (Scroll Bar) ให้กับคอนโทรลโดยอัตโนมัติโดยแถบเลื่อนขึ้นได้ทั้งแนวตั้งและแนวนอนทั้งนี้ขึ้นอยู่กับวิธีการแสดงผลของ ListBox



รูปที่ 2.11 แสดงคอนโทรล ListBox สำหรับใช้แสดงข้อความหลายๆข้อความ

- พิกเจอร์บ็อกซ์ (Image and PictureBox)

พิกเจอร์บ็อกซ์คอนโทรล ทำให้สามารถวางสารสนเทศทางกราฟฟิกไว้ในตำแหน่งเฉพาะบนฟอร์ม พิกเจอร์บ็อกซ์ จะมีความยืดหยุ่นมากกว่า และด้วยเหตุนี้จึงต้องการหน่วยความจำและหน่วยประมวลผลในปริมาณที่มากกว่า พิกเจอร์บ็อกซ์ เหมาะอย่างยิ่งสำหรับสภาพแวดล้อมไคโนมิก เช่น การวาดกราฟฟิกเข้าไปที่จอภาพโดยตรงในขณะที่โปรแกรมกำลังทำงาน หรือการทำไอคอนให้เคลื่อนไหวได้ ด้วยการเคลื่อนย้ายไอคอนไปรอบ ๆ จอภาพ อิมเมจคอนโทรลเหมาะสำหรับสถานการณ์สถิตินั้นคือไม่มีการเปลี่ยนแปลงภาพที่สร้างหรือวางเอาไว้บนจอ

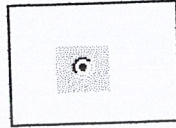


รูปที่ 2.12 แสดงคอนโทรล Image and PictureBox สำหรับแสดงรูปภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- คอนโทรล (OptionButton)

คอนโทรล (OptionButton) เป็นคอนโทรลที่ถูกออกแบบมาให้ใช้ในลักษณะของการเลือกอย่างใดอย่างหนึ่งเท่านั้นในกลุ่ม OptionButton ที่อยู่บน Frame , Form หรือ PictureBox เดียวกัน เป็นต้น



รูปที่ 2.13 แสดงคอนโทรล OptionButton สำหรับใช้ในการเลือก

- คอนโทรล (MSComm)

คอนโทรล MSComm (Communication) เป็นคอนโทรลตัวหนึ่งที่ทำหน้าที่เสมือนช่วยติดต่อกับพอร์ตอนุกรม (Serial Port) โดยใช้ฟังก์ชันในวินโดวส์ API ซึ่งวินโดวส์จะทำหน้าที่โดยตรงในการส่งหรือรับข้อมูลผ่านทางพอร์ตอนุกรมโดยอาศัยไดรเวอร์ Comm.driv อีกทอดหนึ่งข้อมูลที่ถูกรวบรวมในรูปแบบมาตรฐานการสื่อสารแล้วก็สามารถทำการรับ-ส่ง ได้



รูปที่ 2.14 แสดงคอนโทรล MSComm สำหรับใช้ในการติดต่อกับอุปกรณ์ภายนอก

2.8.2 คุณสมบัติของตัวคอนโทรล (Property)

เป็นสิ่งที่ใช้บรรยายลักษณะต่าง ๆ ของคอนโทรล ทำให้คอนโทรลสามารถใช้งานที่หลายรูปแบบมากขึ้น และมีลักษณะแปลกตาไม่ซ้ำกัน การกำหนดค่าคุณสมบัติอาจทำในขณะที่ออกแบบผ่านทางหน้าต่างคุณสมบัติหรือกำหนดขณะรันแอปพลิเคชัน โดยเขียนคำสั่ง โปรแกรมให้ตั้งคุณสมบัติขณะโปรแกรมทำงานคุณสมบัติที่สำคัญได้แก่

BackColor ใช้กำหนดสีส่วนหลัง (Background) ของออบเจกต์

Caption ใช้กำหนดข้อความที่แสดงบนไตเติลบาร์ของออบเจกต์

ControlBox เป็นคุณสมบัติที่บอกว่าให้มีเมนูที่ใช้ย่อ ขยาย หรือปิดหน้าต่างอยู่ ที่มุมบนด้านซ้ายของหน้าต่างหรือไม่ ถ้าเป็น True จะมีการแสดงปุ่มเมนู ถ้าเป็น false จะไม่มีการแสดงปุ่มเมนู

ForeColor คุณสมบัติที่ใช้กำหนดสีของข้อความหรือกราฟิกบนออบเจกต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Height , width เป็นคุณสมบัติที่ใช้กำหนดความสูงและความกว้างของออบเจกต์ซึ่งวัดอยู่ในหน่วยที่เรียกว่า ทวิพ (ทวิพ = 1 / 1440 นิ้ว)

Name เป็นคุณสมบัติที่ใช้กำหนดชื่อของฟอร์มในรหัสโปรแกรมซึ่งเปลี่ยนแปลงได้เฉพาะในขณะออกแบบเท่านั้น

Visible เป็นคุณสมบัติที่กำหนดว่าจะให้เห็นออบเจกต์หรือไม่ในขณะรันโปรแกรม

Hide เป็นเมธอดที่ใช้ซ่อนออบเจกต์ไว้ชั่วคราว

Show เป็นเมธอดที่ใช้ในการแสดงออบเจกต์ที่ต้องการ

LoadPicture เป็นเมธอดที่ใช้ตั้งคุณสมบัติ Picture ของออบเจกต์

Setfocus เป็นเมธอดที่ใช้ในการโฟกัสมายังออบเจกต์นี้เพื่อให้ผู้ใช้สามารถทำงานด้วยได้

FontBold , FontItalic , FontName , Fontsize , FontUnderline เป็นคุณสมบัติที่กำหนดรูปแบบของข้อความที่แสดง ในคอนโทรลว่าจะให้ใช้ตัวอักษรแบบใด (FontName) ขนาดเท่าใด (Font Size) เป็นตัวหนา (FontBold) ตัวเอียง (FontItalic) หรือขีดเส้นใต้ (FontUnderline)

MaxLength เป็นคุณสมบัติที่กำหนดจำนวนอักขระสูงสุดของข้อความ โดย ถ้ากำหนดเป็น “ 0 ” จะรับข้อความได้ไม่จำกัด

Multiline เป็นคุณสมบัติที่กำหนดให้แท็กบ็อกซ์แสดงข้อความในรูปแบบหลายบรรทัดได้

ScrollBars เป็นคุณสมบัติที่กำหนดรูปแบบของข้อความที่แสดงในคอนโทรลว่าจะให้แสดงสกรอลบาร์ด้วยหรือไม่อย่างไร เมื่อข้อความยาวเกินความกว้างหรือความยาวของคอนโทรล

SelLength เป็นคุณสมบัติที่บอกว่ายาวของข้อความที่เลือกอยู่ในขณะนั้นมีค่าเท่าไร

SelStart เป็นคุณสมบัติที่บอกว่ายาวตำแหน่งแรกของข้อความที่เลือกอยู่ในขณะนั้นอยู่ในตำแหน่งที่เท่าไร ถ้าไม่มีการเลือกข้อความอยู่ จะบอกตำแหน่งที่จะแทรกข้อความอยู่ในขณะนั้น

SelText เป็นคุณสมบัติที่บอกว่ายาวข้อความส่วนที่เลือกอยู่ในขณะนั้นคืออะไร

Text เป็นคุณสมบัติที่เก็บข้อความของแท็กบ็อกซ์ในปัจจุบัน และผู้ใช้ยังสามารถโปรแกรมดัดแปลงแก้ไขข้อความที่แสดงออกมาได้

2.8.3 เมธอด (Method)

เป็นโปรแกรมย่อยประเภทหนึ่ง ซึ่งเป็นสมาชิกของคอนโทรลนั้น ๆ เหมือนกับคุณสมบัติของคอนโทรล เมื่อเรียกใช้เมธอดจะเป็นการสั่งให้คอนโทรลทำงานให้ ซึ่งอาจจะมีการเปลี่ยนแปลงค่าคุณสมบัติต่าง ๆ ของคอนโทรลนั้น

Cls เป็นเมธอดที่ใช้เคลียร์สิ่งที่วาดบนออบเจกต์

Hide เป็นเมธอดที่ใช้ซ่อนออบเจกต์ไว้ชั่วคราว

Show เป็นเมธอดที่ใช้ในการแสดงออบเจกต์ที่ต้องการ

LoadPicture เป็นเมธอดที่ใช้ตั้งคุณสมบัติ Picture ของออบเจกต์นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SetFocus เป็นเมธอดที่ใช้ในการโฟกัสมายังออบเจกต์นี้เพื่อให้ผู้ใช้สามารถทำงานด้วยได้

2.8.4 อีเวนต์ (Event)

เป็นการตอบสนองต่อเหตุการณ์ภายนอกของคอนโทรล และเป็น โปรแกรมย่อยที่ทำงานทันทีที่เกิดเหตุการณ์นั้นขึ้นมาอีเวนต์ที่สำคัญได้แก่

Click เป็นอีเวนต์ที่เกิดขึ้นเมื่อมีการคลิกเมาส์บนออบเจกต์ อีเวนต์ Click ก็จะทำการรันคำสั่งที่เขียน

Change เป็นอีเวนต์ที่เกิดขึ้นเมื่อมีการเปลี่ยนแปลงคุณสมบัติ Text ของ เท็กซ์บ็อกซ์ไม่ว่าจะเป็นจากผู้ใช้อื่นหรือความเข้าไปใหม่ หรือ จากการตั้งค่าคุณสมบัติ Text ใหม่ ในโปรแกรม

LostFocus เป็นอีเวนต์ที่เกิดขึ้นเมื่อผู้ใช้ตั้งระยะเคลื่อนออกไปจากเท็กซ์บ็อกซ์หรือใช้เมาส์เลือกออบเจกต์อื่น ๆ

SetFocus เป็นวิธีการที่มีประโยชน์ที่สุดในการชี้ความสนใจไปยังเท็กซ์บ็อกซ์ที่สำคัญที่สุดในหลายๆ เท็กซ์บ็อกซ์ โดยเคอร์เซอร์จะ ไปอยู่ในเท็กซ์บ็อกซ์ที่เรากำหนด

2.9 ระบบ ไคลเอนต์ / เซิร์ฟเวอร์

ระบบ ไคลเอนต์ / เซิร์ฟเวอร์ เป็นสถาปัตยกรรมทางด้านระบบคอมพิวเตอร์แบบ Distribute แบบหนึ่งที่น่าเอาเครื่องคอมพิวเตอร์มากกว่า 1 เครื่องมาเชื่อมต่อกันด้วยระบบเครือข่าย (Computer Network) โดยมีจุดประสงค์เพื่อต้องการให้เกิดการใช้ข้อมูลร่วมกันระหว่างเครื่องคอมพิวเตอร์ในเครือข่ายนั้น ภายใต้คุณสมบัติต่างๆ ดังนี้

- เครื่องคอมพิวเตอร์ที่เชื่อมต่อกันเป็นระบบ จะต้องประกอบด้วยเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็น ไคลเอนต์ และ เซิร์ฟเวอร์

ฐานข้อมูลจะถูกจัดเก็บอยู่ในคอมพิวเตอร์ที่ทำหน้าที่ เซิร์ฟเวอร์ โปรแกรมที่ ทำหน้าที่ในการเรียกใช้ข้อมูลจากฐานข้อมูล จะทำงานอยู่บนเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็น ไคลเอนต์ ภายในระบบคอมพิวเตอร์แบบ ไคลเอนต์ / เซิร์ฟเวอร์ โดยทั่วไป มักจะประกอบด้วย เครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็น เซิร์ฟเวอร์ 1 เครื่องที่เชื่อมต่อกับเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็น ไคลเอนต์ มากกว่า 1 เครื่อง ซึ่งจะมีจำนวนเท่าใดขึ้นอยู่กับปริมาณงานภายในระบบ เครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็น เซิร์ฟเวอร์ จะใช้จัดเก็บฐานข้อมูล และ โปรแกรม DBMS ที่มีหน้าที่ในการรับคำสั่งจาก ไคลเอนต์ มาประมวลผล และ นำข้อมูลข่าวสารที่ ไคลเอนต์ ต้องการ ส่งกลับไปยัง ไคลเอนต์ ดังนั้น จึงมักเรียกส่วน เซิร์ฟเวอร์ นี้ว่า ส่วนของ โปรแกรม Backend ส่วนเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็น ไคลเอนต์ แต่ละเครื่อง จะใช้สำหรับ Run โปรแกรมที่ทำหน้าที่ในการส่งคำสั่งสำหรับเรียกใช้ข้อมูลจากฐานข้อมูลไปยัง เซิร์ฟเวอร์ เพื่อให้ เซิร์ฟเวอร์ ส่งข้อมูลข่าวสารที่ต้องการกลับมาและนำไปแสดงผล ให้กับผู้ใช้ ต่อไปดังนั้นจึงมักเรียกส่วน ไคลเอนต์ นี้ว่า ส่วนของ

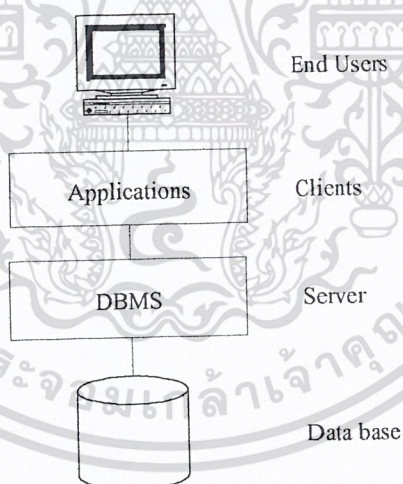
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในบางระบบ อาจมีการใช้ เซิร์ฟเวอร์ มากกว่า 1 ตัว ซึ่งอาจเนื่องมาจาก มีการใช้ฐานข้อมูลหลายฐานข้อมูลและจำเป็นที่จะต้องแยกเก็บใน เซิร์ฟเวอร์ ที่ต่างเครื่องกัน ในกรณีนี้ ไคลเอนต์ จะต้องสามารถที่จะเชื่อมต่อกับ เซิร์ฟเวอร์ แต่ละ เซิร์ฟเวอร์ ได้ แต่ในการเชื่อมต่อของไคลเอนต์ ไปยัง เซิร์ฟเวอร์ ในแต่ละครั้ง จะเกิดขึ้น ได้เพียง เซิร์ฟเวอร์ เดียว ดังนั้น ผู้ใช้ของระบบ ในลักษณะนี้ จึงต้องทราบถึง เซิร์ฟเวอร์ ที่จัดเก็บข้อมูลที่ตนต้องการ แต่ก็ยังมีบางกรณี ที่ ไคลเอนต์สามารถใช้งานหลายๆ เซิร์ฟเวอร์ ในแบบต่อเนื่องได้ ซึ่งมักเกิดกับฐานข้อมูล ที่มีการแยกข้อมูลในฐานข้อมูลออกเป็นหลายๆ และจัดเก็บลงในแต่ละ เซิร์ฟเวอร์ แต่การใช้งานในกรณีนี้จะไม่จัดเป็นระบบคอมพิวเตอร์ในแบบ ไคลเอนต์ / เซิร์ฟเวอร์ เนื่องจาก ระบบมีการทำงานที่ตรงข้ามกับระบบคอมพิวเตอร์ในแบบ ไคลเอนต์ / เซิร์ฟเวอร์ กล่าวคือแต่ละ เซิร์ฟเวอร์ จะมอง ไคลเอนต์ เป็น เซิร์ฟเวอร์ เดียวแทน เนื่องจาก จะต้องทยอยส่งข้อมูล ไปให้กับ ไคลเอนต์ แทน

2.9.1 ระบบไคลเอนต์ / เซิร์ฟเวอร์ กับฐานข้อมูล

ด้วยมุมมองตามสถาปัตยกรรม ANSI / SPARC ของระบบฐานข้อมูลอาจสามารถแบ่งระบบฐานข้อมูลออกเป็น 2 ส่วนคือ ส่วนของ เซิร์ฟเวอร์ ซึ่งโดยทั่วไปมักเรียกว่า โปรแกรม Backend และส่วนของ ไคลเอนต์ ซึ่งโดยทั่วไปมักเรียกว่า โปรแกรม Front-end ดังรูป



รูปที่ 2.15 แสดงระบบ ไคลเอนต์ / เซิร์ฟเวอร์

ส่วน เซิร์ฟเวอร์ ในแง่ของฐานข้อมูล ได้แก่ โปรแกรม DBMS ที่ทำหน้าที่สนับสนุนการทำงานต่างๆ ของฐานข้อมูล เช่น การสร้างและกำหนดโครงสร้างของฐานข้อมูล การจัดการข้อมูลภายในฐานข้อมูล การรักษาความปลอดภัยให้กับข้อมูล การรักษาความถูกต้องของข้อมูล ฯลฯ เป็นต้น

ส่วนไคลเอนต์ ในแง่ของฐานข้อมูล ได้แก่โปรแกรมต่างๆ ที่ทำงานร่วมกับโปรแกรม DBMS เพื่อใช้งานฐานข้อมูล โปรแกรมเหล่านี้ อาจพัฒนาขึ้นด้วยภาษาคอมพิวเตอร์ภาษาใดภาษาเอกจากเป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนึ่ง เช่น ภาษา C , COBOL , FOCUS ฯลฯ เป็นต้น หรืออาจพัฒนาขึ้นด้วย Tool ของตัว DBMS เอง เช่น Tool ที่ใช้ในการสร้างรายงานต่างๆ ภาษา Query Language ตารางที่ใช้แสดงข้อมูล ในรูป Spreadsheet ภาษาที่ใช้ในการสร้าง Form สำหรับจัดการกับข้อมูลหรือที่เรียกว่า 4GL รูปภาพ กราฟฟิคที่ใช้แสดงข้อมูลทางสถิติ หรือในบาง Tool อาจมีส่วนที่ใช้ในการสร้าง โปรแกรมขึ้นใช้งาน หรือที่เรียกว่า CASE (Computer-aided Software Engineering) ฯลฯ เป็นต้น

ทั้ง 2 ส่วนของระบบฐานข้อมูลนี้ จะถูกนำไปใช้บนเครื่องคอมพิวเตอร์ที่ต่างกัน เพื่อสนับสนุนการประมวลผลในแบบกระจาย (Distributed Processing) เช่น ระบบคอมพิวเตอร์ในแบบ ไคลเอนต์ / เซิร์ฟเวอร์ ที่ซึ่งเครื่องคอมพิวเตอร์แต่ละเครื่องถูกเชื่อมโยงกันด้วยระบบเครือข่าย ซึ่งส่งผลให้ข้อมูลชุดเดียวกันสามารถส่งถ่ายไปมาระหว่างเครื่องคอมพิวเตอร์เครื่องต่างๆ ที่เชื่อมต่อกันอยู่ภายในเครือข่ายนั้นได้

2.9.2 การเขียนโปรแกรมกับระบบ ไคลเอนต์ / เซิร์ฟเวอร์

สิ่งที่โปรแกรมเมอร์ จะต้องคำนึงถึงในการพัฒนาโปรแกรมขึ้นใช้ในระบบคอมพิวเตอร์แบบ ไคลเอนต์ / เซิร์ฟเวอร์ ได้แก่ การมอง เซิร์ฟเวอร์ เป็นเครื่องมือในการเข้าถึงข้อมูล และการมองข้อมูลในระดับ Record เนื่องจาก Request ที่ ไคลเอนต์ ส่งไปให้กับ เซิร์ฟเวอร์ เป็นคำสั่งที่ต้องการได้ผลลัพธ์ในระดับ Record และจะต้องอยู่ในรูปของคำสั่งในแบบ Non-rocedural เช่น คำสั่ง SQL ดังนั้น จึงไม่มีการใช้คำสั่งที่ก่อให้เกิดการทำซ้ำ (Loop) หรือเป็นคำสั่งที่ใช้ในการแก้ไข (Update) หรือลบ (Delete) ข้อมูล แต่เมื่อต้องการลดจำนวนของการติดต่อระหว่าง ไคลเอนต์ และ เซิร์ฟเวอร์ ให้นำเอา Stored Procedure มาใช้แทนคำสั่ง SQL (Stored Procedure) เป็นคำสั่งที่เรียกใช้ข้อมูลจากฐานข้อมูลเช่นเดียวกับ SQL แต่สามารถนำมาเอาคำสั่งมารวมเป็นชุดคำสั่งได้ รวมทั้งจะถูกจัดเก็บและทำงานอยู่บน เซิร์ฟเวอร์ และจะถูกเรียกใช้โดย ไคลเอนต์

สำหรับข้อดีของการใช้ Stored Procedure ประกอบด้วย

- Stored Procedure สามารถปิดบังความซับซ้อนทางด้านการใช้งานส่งผลให้ ผู้ใช้สามารถเรียกใช้ข้อมูลภายในฐานข้อมูลได้ง่าย ซึ่งเป็นไปตามคุณสมบัติทางด้าน Data Independence
- แต่ละ Stored Procedure สามารถเรียกใช้ร่วมกันหลายๆ ไคลเอนต์
- ผู้ใช้สามารถตรวจสอบความถูกต้องของ Stored Procedure ก่อนนำไปใช้งาน ซึ่งต่างจากคำสั่ง SQL ที่จะไม่สามารถตรวจสอบความถูกต้องก่อนนำไปใช้งานได้ เนื่องจาก เป็นคำสั่งที่อยู่ในรูปของ Request ที่ ไคลเอนต์ ส่งไปยัง เซิร์ฟเวอร์ ให้ทำงาน ณ ขณะนั้น
- Stored Procedure สามารถกำหนดความปลอดภัยได้ดีกว่าคำสั่ง SQL เนื่องจาก สามารถกำหนดผู้ที่มีสิทธิ์ในการใช้ Stored Procedure นั้น ได้โดยตรง

แต่อย่างไรข้อเสียของการใช้ Stored Procedure ได้แก่ ความแตกต่างในการใช้งาน

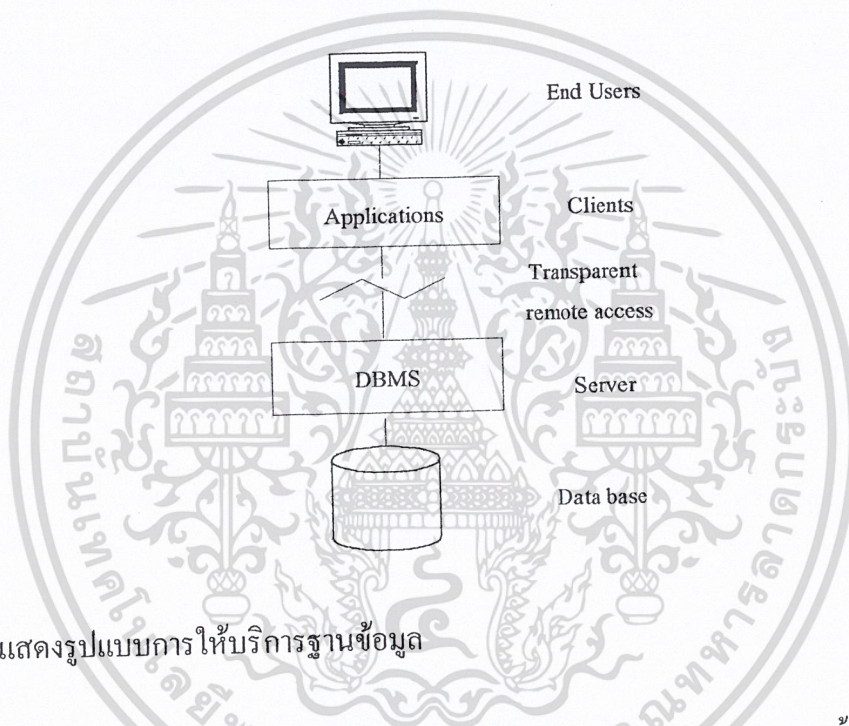
Stored Procedure ของแต่ละผลิตภัณฑ์ เนื่องจากยังไม่มีกำหนดมาตรฐานของ Stored Procedure

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9.3 ระบบ ไคลเอนต์ / เซิร์ฟเวอร์ ที่สนับสนุนการประมวลผลแบบกระจาย

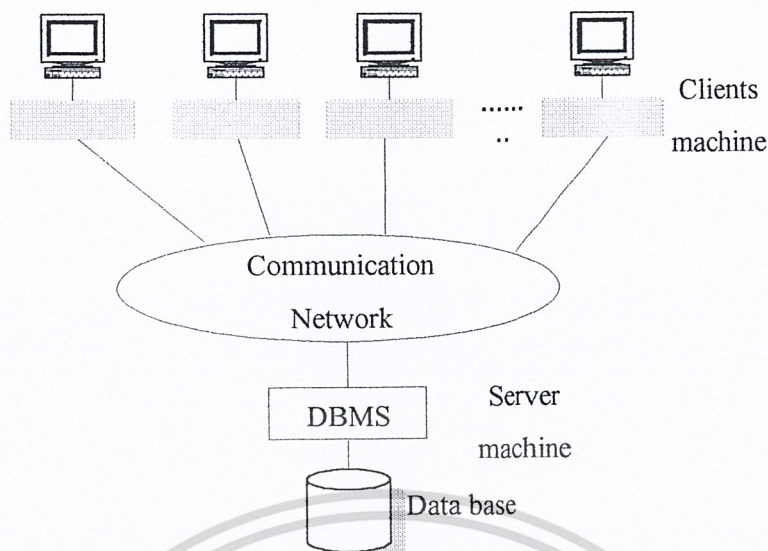
เนื่องจากการประมวลผลแบบกระจาย เป็นการนำเอาคอมพิวเตอร์หลายๆ เครื่อง มาเชื่อมต่อกันด้วยระบบเครือข่าย ด้วยจุดประสงค์ ที่ต้องการใช้ข้อมูลร่วมกัน ดังนั้นระบบฐานข้อมูลที่นำมาใช้ในการประมวลผลแบบกระจายจึงต้องแบ่งออกเป็น ส่วน เซิร์ฟเวอร์ (โปรแกรม Backend) และ ส่วน ไคลเอนต์ (โปรแกรม Front-end) ไว้ในคอมพิวเตอร์ที่ต่างเครื่องกัน เพื่อให้คอมพิวเตอร์แต่ละเครื่องที่ทำหน้าที่เป็น ไคลเอนต์ สามารถเรียกใช้ข้อมูลจากฐานข้อมูลที่เก็บอยู่ในเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็น เซิร์ฟเวอร์ ได้ร่วมกัน ดังรูป



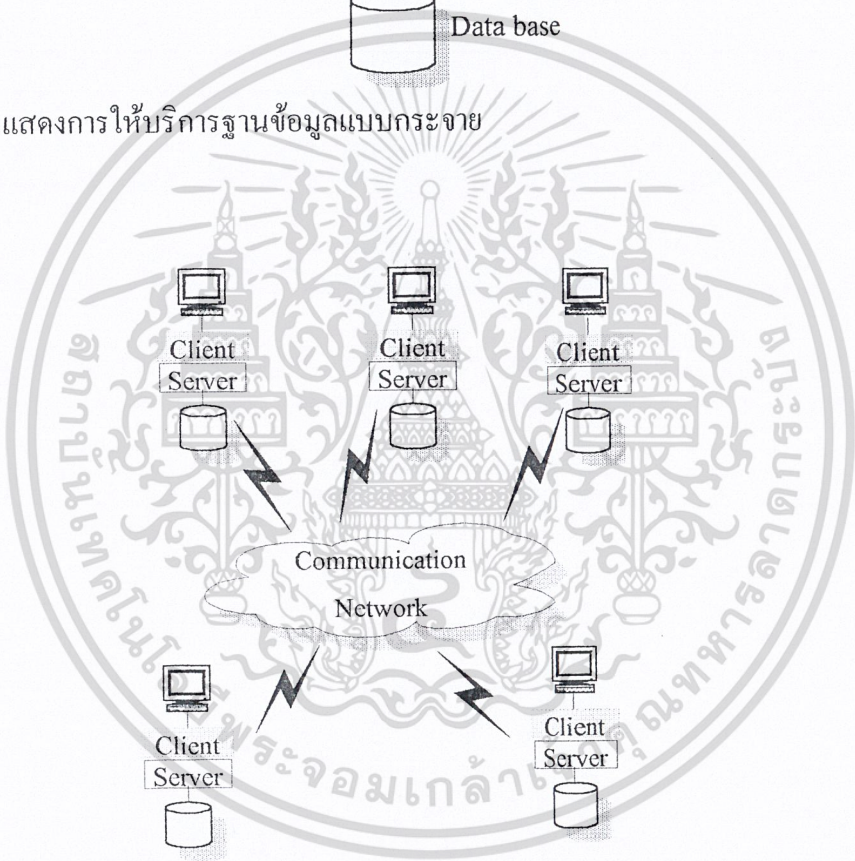
รูปที่ 2.16 แสดงรูปแบบการให้บริการฐานข้อมูล

- การนำเอาระบบไคลเอนต์ / เซิร์ฟเวอร์ มาใช้ในการประมวลผลแบบกระจายนี้จะส่งผลให้
1. การประมวลผลของ ไคลเอนต์ / เซิร์ฟเวอร์ จะอยู่ในรูปแบบขนาน ดังนั้นเวลาในการตอบสนอง (Response Time) ของระบบที่มีต่อการทำงานจะใช้เวลาน้อยลง รวมทั้งปริมาณผลลัพธ์ที่ได้จากการประมวล (Throughput) จะได้ปริมาณมากขึ้นด้วย
 2. ประสิทธิภาพการทำงานของ DBMS จะดีขึ้น โดยเฉพาะอย่างยิ่ง ในกรณีที่คอมพิวเตอร์ที่ใช้เป็น เซิร์ฟเวอร์ มีความสามารถที่สนับสนุนฟังก์ชันการทำงานต่างๆ ของ DBMS
 3. การติดต่อกับผู้ใช้ (User Interface) จะมีความรวดเร็ว และมีประสิทธิภาพมากขึ้น เนื่องจากหน้าที่หลักของคอมพิวเตอร์ที่ทำหน้าที่เป็น ไคลเอนต์ ได้แก่ การแสดงผล
 4. คอมพิวเตอร์ที่ทำหน้าที่เป็น ไคลเอนต์ สามารถใช้ข้อมูลร่วมกันได้ซึ่งสามารถแสดงด้วยแผนภาพดังรูปที่ 2.17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.17 แสดงการให้บริการฐานข้อมูลแบบกระจาย



รูปที่ 2.18 รูปแสดงการให้บริการฐานข้อมูลขนาดใหญ่

ในกรณีที่ต้องค์กรที่มีขนาดใหญ่ ที่มีการจัดเก็บข้อมูลไว้หลายๆ ที่ เช่น ธนาคาร ที่มีการแยกจัดเก็บข้อมูลตามสาขาต่างๆ อาจมีการกำหนดให้คอมพิวเตอร์ที่ทำหน้าที่เป็น ไคลเอนต์ มีการจัดเก็บข้อมูลของตนเองไว้และในขณะเดียวกัน คอมพิวเตอร์ที่ทำหน้าที่เป็น เซิร์ฟเวอร์ จะคอยส่ง ข้อมูลให้กับคอมพิวเตอร์เครื่องอื่น และในขณะเดียวกัน ก็สามารถทำหน้าที่เป็น ไคลเอนต์ เพื่อเรียกใช้ข้อมูลจากฐานข้อมูลในคอมพิวเตอร์เครื่องอื่นได้เช่นเดียวกัน สำหรับรูปแบบของระบบในลักษณะนี้เรียกว่า Entire Database System ดังรูปที่ 2.18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกรณีที่มีการจัดเก็บฐานข้อมูลใน เซิร์ฟเวอร์ หลายๆ ตัว ไคลเอนต์ จะต้องมีความสามารถเพิ่มเติมในการเชื่อมต่อ ไปยัง เซิร์ฟเวอร์ แต่ละตัว เพื่อที่จะเรียกใช้ข้อมูลในฐานข้อมูลของแต่ละ เซิร์ฟเวอร์ ได้ แต่การเรียกใช้ข้อมูลในฐานข้อมูลของแต่ละ เซิร์ฟเวอร์ นี้ จะเกิดขึ้นได้เพียง เซิร์ฟเวอร์ เดียว ในแต่ละครั้งของการติดต่อระหว่าง ไคลเอนต์ / เซิร์ฟเวอร์

สรุป

ระบบ ไคลเอนต์ / เซิร์ฟเวอร์ เป็นสถาปัตยกรรมทางด้านระบบคอมพิวเตอร์แบบหนึ่ง ที่นำมาใช้สนับสนุนการทำงานของระบบคอมพิวเตอร์ที่มีการประมวลผลแบบกระจายเนื่องจาก ระบบ ไคลเอนต์ / เซิร์ฟเวอร์ จะแบ่งการทำงานของคอมพิวเตอร์ออกเป็น 2 ส่วนคือ ส่วนของ เซิร์ฟเวอร์ และส่วนของ ไคลเอนต์ ในระบบ ไคลเอนต์ / เซิร์ฟเวอร์ โดยทั่วไป จะประกอบด้วย คอมพิวเตอร์ที่ทำหน้าที่เป็น เซิร์ฟเวอร์ 1 เครื่องที่เชื่อมต่ออยู่กับคอมพิวเตอร์ที่ทำหน้าที่เป็น ไคลเอนต์ มากกว่า 1 เครื่องผ่านทางระบบเครือข่ายคอมพิวเตอร์ โดยส่วนของ เซิร์ฟเวอร์ จะมีหน้าที่หลักในการจัดเก็บข้อมูล และคอยจัดหาข้อมูลตามความต้องการของ ไคลเอนต์ ในขณะที่ส่วนของ ไคลเอนต์ จะมีหน้าที่นำเอาข้อมูลที่รับจาก เซิร์ฟเวอร์ ไปประมวลผลหรือแสดงผล ซึ่งโดยวิธีนี้ จะทำให้ เซิร์ฟเวอร์ สามารถให้บริการแก่ไคลเอนต์ ได้มากขึ้น

2.10 การสร้างฐานข้อมูลด้วยโปรแกรม Access

โปรแกรม Access เป็น โปรแกรมหนึ่งในกลุ่มของ Microsoft Office ซึ่งจะช่วยงานด้าน “ฐานข้อมูล” (Database) นั่นคือการนำมาเก็บรายละเอียดของข้อมูลต่างๆ ให้เป็นหมวดหมู่ จากนั้น จึงนำข้อมูลเหล่านั้นมาสร้าง รายงาน (Record) , แบบฟอร์ม (Form) , ค้นหา ข้อมูล (Query) ในฐานข้อมูลตามเงื่อนไข และอื่นๆ อีกมากมาย

2.10.1 ความหมายของ Database

Database หรือ ฐานข้อมูล หมายถึง ข้อมูลที่ถูกรวบรวมไว้อย่างมีระเบียบแบบแผน โดยข้อมูลเหล่านี้จะต้องมีความสัมพันธ์กันด้วย ตัวอย่างของ Database เช่น

- ข้อมูลในสมุดโทรศัพท์ขององค์กรโทรศัพท์ ซึ่งจะมีชื่อ , ที่อยู่ และหมายเลขโทรศัพท์
- ข้อมูลของนักศึกษาใน โรงเรียน หรือมหาวิทยาลัย โดยจะมีชื่อ , นามสกุล , ที่อยู่ ฯลฯ รวบรวมไว้

2.10.2 การสร้างฐานข้อมูล

เมื่อเราต้องการสร้างฐานข้อมูลในเครื่องคอมพิวเตอร์เราจะสามารถสร้างได้จาก 2 วิธีดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. สร้างจากโปรแกรมสำเร็จรูป

ในกรณีที่เราต้องการความสะดวกรวดเร็ว ในการสร้างฐานข้อมูลแล้ว ก็สามารถสร้างจากโปรแกรมสำเร็จรูปที่ถูกรออกแบบมาเพื่อทำงานทางด้านฐานข้อมูล โดยเฉพาะซึ่งโปรแกรมสำเร็จรูปนี้ จะมีมาตั้งแต่ยุคแรกๆ เลยทีเดียว ที่เรารู้จักกันดีได้แก่ Dbase , Foxpro ซึ่งจะสามารถบันทึกแต่เพียงตัวหนังสือเท่านั้น จนในปัจจุบันนี้จะมีโปรแกรมที่จัดการกับฐานข้อมูลที่เราู้จักกันดีได้แก่ Microsoft Access ของบริษัทไมโครซอฟท์ที่มีความสามารถบันทึกข้อมูลแปลกๆ ที่ได้กล่าวไปแล้วนั้น นอกจากนี้แล้วยังมีอีกหลายๆ โปรแกรมที่สร้างมาเพื่อฐานข้อมูล โดยเฉพาะคือ Oracle หรือ power Builder

การสร้างฐานข้อมูล โดยโปรแกรมสำเร็จรูปนี้ จะทำให้เราสร้างฐานข้อมูลได้รวดเร็ว แต่จะมีข้อจำกัดอยู่มาก เมื่อต้องการสร้างโปรแกรม เพื่อทำงานกับฐานข้อมูล และเพื่อป้องกันปัญหาเหล่านี้ เราจะใช้โปรแกรมสำเร็จรูปสร้างฐานข้อมูลแต่เพียงอย่างเดียว และใช้วิธีเขียนโปรแกรมอีกภาษาหนึ่งเพื่อเปิดดูฐานข้อมูลนี้ที่อีกทีหนึ่ง

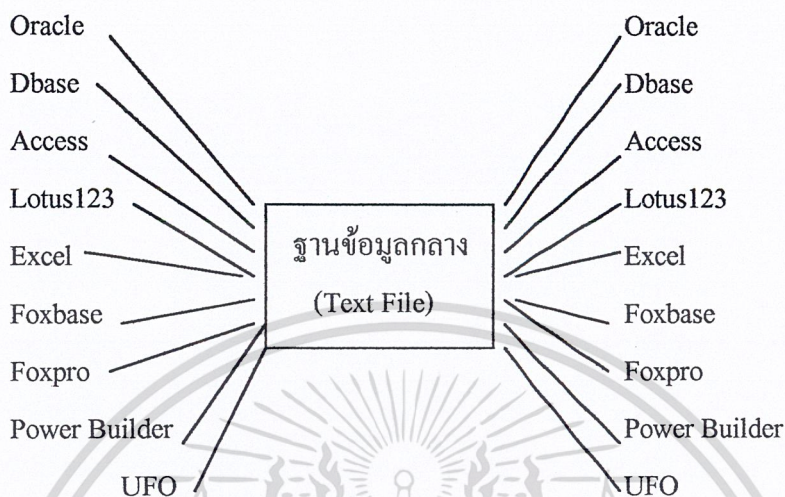
นอกจากนี้แล้ว การใช้โปรแกรมสำเร็จรูปเพื่อสร้างฐานข้อมูลจะช่วยให้การพัฒนาโปรแกรมทำงานได้เร็วขึ้น ให้มีการสร้างฐานข้อมูลด้วยโปรแกรมสำเร็จรูปไปก่อน โดยไม่ต้องรอนกว่าโปรแกรมที่พัฒนาสำเร็จ แล้วค่อยเปลี่ยนจากฐานข้อมูลที่สร้างแล้วนี้ ให้รับกับโปรแกรมที่พัฒนานี้ โปรแกรมสำเร็จรูปบางโปรแกรมก็สามารถสร้างฐานข้อมูลได้เช่นเดียวกัน เช่นในโปรแกรมทางด้าน Spread Sheet เช่น Microsoft Excel หรือ Lotus 123 เป็นต้น เนื่องจากเป็นโปรแกรม ที่มีพื้นที่ทำงานเป็นตาราง ที่เราสามารถบันทึกข้อมูลนี้ลงในตารางได้

2. การสร้างฐานข้อมูลเอง

ถ้าหากเราต้องการสร้างโปรแกรม และควบคุมการทำงานของฐานข้อมูลได้ง่ายแล้วก็ควรที่จะสร้างฐานข้อมูลของเราเอง ซึ่งตามปกติฐานข้อมูลที่สามารถสร้างขึ้นได้เองนี้ จะมีอยู่ 3 ชนิดคือ Sequential , Random และ Binary ซึ่งแต่ละชนิด จะมีคุณสมบัติที่แตกต่างกันออกไปแล้วแต่เราจะเลือก และเราจะกล่าวอีกทีหนึ่งในหนังสือเล่มนี้ สำหรับภาษาที่เราจะสร้างฐานข้อมูลของเราเองนี้ จะเป็นภาษา Visual Basic ของบริษัทไมโครซอฟท์

2.10.3 การย้ายฐานข้อมูล

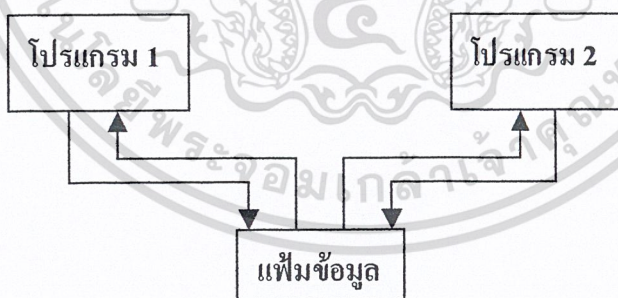
ในการย้ายฐานข้อมูลไปมานี้ บางครั้งจะให้มีการย้ายข้อมูลกลางอันหนึ่ง ซึ่งสามารถติดต่อกับฐานข้อมูลทุกชนิดได้ ฐานข้อมูลกลางนี้ก็คือ ฐานข้อมูลที่เป็นเท็กซ์ไฟล์ธรรมดาๆ นี้เอง ซึ่งฐานข้อมูลเท็กซ์ไฟล์นี้จะเป็นที่ยอมรับของฐานข้อมูลเกือบทุกๆ ชนิดของข้อมูลเลย ดังนั้นถ้าเราต้องการย้ายฐานข้อมูลจากโปรแกรมที่พิศดาร มีคนใช้น้อยๆ แล้วก็ควรแปลงฐานข้อมูลเป็นฐานข้อมูลเท็กซ์ไฟล์ก่อน จากนั้นจึงค่อยๆ ย้ายเข้าไปในโปรแกรมที่เราต้องการ



รูปที่ 2.19 การแปลงฐานข้อมูลต่างๆ เป็นเท็กซ์ไฟล์

2.10.4 การทำงานของโปรแกรมร่วมกับฐานข้อมูล

ฐานข้อมูลนั้นจะอยู่คนละส่วนกับโปรแกรม ดังนั้นการพัฒนาทั้งสองส่วนนี้จึงแยกกัน พัฒนาได้ฐานข้อมูลจึงทำงานได้อิสระ ไม่ยึดติดกับโปรแกรมหนึ่งโปรแกรมใด คือฐานข้อมูลหนึ่งๆ อาจจะทำงานร่วมกับโปรแกรมหลายโปรแกรมก็ได้ ดังแสดงในรูป



รูปที่ 2.20 แสดงการร่วมการใช้เพิ่มข้อมูลของโปรแกรม

2.10.5 สัพท์ต่างๆ ที่ควรทราบเกี่ยวกับ Database สำหรับโปรแกรม Access Field (ฟิลด์)

หมายถึงรายละเอียดย่อยๆ เช่น ชื่อคน , นับเป็น 1 Field , หมายเลขโทรศัพท์ก็นับเป็น 1 Field เช่นกัน (ใน Relational Database จะใช้คำว่า Attribute แทน ซึ่งมีความหมายที่ตรงมากกว่า)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Record (เรคคอร์ด)

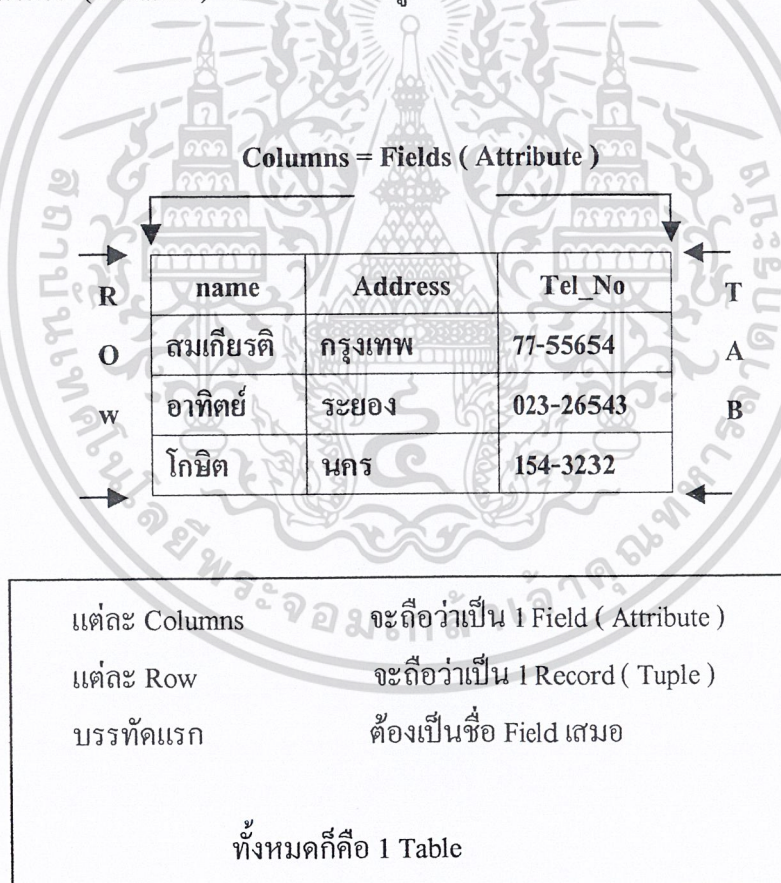
ชื่อคนนับเป็น 1 Field , ที่อยู่นับเป็น 1 Field และหมายเลขโทรศัพท์ก็นับเป็น 1 Field ถ้า นำ 3 Field นี้มารวมกันจะนับเป็น 1 Record ดังนั้นรายละเอียดของคน 1 คนในสมุดโทรศัพท์ ก็คือ 1 Record นั่นเอง (ใน Relational Database จะใช้คำว่า Tuple แทนคำว่า Record)

Table (ตาราง)

หมายถึงรายละเอียดของทุกคนที่อยู่ในสมุดโทรศัพท์ 1 เล่มคือ 1 Table

Relational Database

เป็น Database ชนิดหนึ่ง ซึ่ง Access นำมาใช้เป็นพื้นฐานในการจัดการฐานข้อมูล ของ โปรแกรม โดยใน Database หนึ่งนั้นอาจประกอบไปด้วย Table หลายๆ Table และในแต่ละ Table ก็จะมีความสัมพันธ์ (Relation) ซึ่งกันและกันอยู่



รูปที่ 2.21 แสดงตัวอย่างการสร้างตาราง

2.11 รูปแบบการติดต่อกับฐานข้อมูลด้วย Visual Basic

ในการติดต่อกับฐานข้อมูลโดยปกติแล้ว VB จะเชื่อมโยงผ่านทาง Database Engine ที่เรียกว่า JET Engine จึงอาจกล่าวได้อีกนัยหนึ่งว่า JET Engine เป็นไดรเวอร์ชนิดหนึ่ง ซึ่งทำหน้าที่ การค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นตัวเชื่อมโยงให้ VB สามารถติดต่อกับฐานข้อมูลได้นั่นเอง โดยที่ฐานข้อมูลหลัก (Default) ที่ VB รู้จักเป็นอย่างดีคือ MS Access แต่ Visual Basic สามารถติดต่อกับฐานข้อมูลได้ทุกชนิดเช่นกัน โดยอาศัยเทคโนโลยีหลายๆ อย่าง

ถ้าสำหรับฐานข้อมูลที่น่าเสนอ จะใช้ฐานข้อมูลของ MS Access 2000 Thai Edition เป็นหลักการติดต่อกับฐานข้อมูลใน Visual Basic จะแยกออกเป็น 4 ประเภทใหญ่ๆ คือ

2.11.1 แบบที่ 1 ติดต่อกับโดยอาศัยคอนโทรลด้านฐานข้อมูล

มีศัพท์เรียกคอนโทรลกลุ่มนี้โดยเฉพาะว่า Bound Control ส่วนใหญ่แล้วก็คือ กลุ่มของคอนโทรลมาตรฐานที่ใช้กันโดยทั่วไปนั่นเอง เช่น คอนโทรล TextBox , Picture , Image , ListBox , comboBox เป็นต้น โดยใช้คอนโทรล Data (Data Control) เป็นตัวเชื่อมโยงระหว่างฐานข้อมูล กับกลุ่ม Bound Controls

คุณสามารถตรวจสอบได้ว่า คอนโทรลตัวใดบ้างถูกจัดอยู่ในกลุ่มของ Bound Controls โดยการตรวจสอบว่าคอนโทรลตัวนั้นมีคุณสมบัติที่ขึ้นต้นด้วยคำว่า Data... เช่น DataField , DataFormat , dataSource หรือ ไม่ ถ้ามีหมายถึง คอนโทรลตัวดังกล่าว ถูกจัดอยู่ในกลุ่มของ Bound Controls ด้วยเช่นกัน

ยังมีคอนโทรลอีกกลุ่มหนึ่งเรียกว่า ActiveX Bound Controls หมายถึง กลุ่มของคอนโทรลที่มีคุณสมบัติที่ขึ้นต้นด้วยคำว่า Data... เช่นกัน แต่มีข้อแตกต่างจากคอนโทรลในกลุ่ม Bound Controls ก็คือ ActiveX Bound Controls อาจมาจากผู้ผลิตรายอื่นๆ (Thirds Party) ซึ่งเป็นคอนโทรลที่ไม่ได้อยู่ในแถบเครื่องมือมาตรฐานของ Visual Basic โดยที่คุณต้องเพิ่มเติมคอนโทรลกลุ่มนี้เข้ามาในแถบเครื่องมือเอง ดังนั้นจึงเรียกคอนโทรลกลุ่มนี้ว่า ActiveX Bound Controls เช่น คอนโทรล DBGrid , คอนโทรล Dblist หรือคอนโทรล DBCombo ฯลฯ เป็นต้น

2.11.2 แบบที่ 2 ติดต่อกับโดยใช้ออบเจกต์ Data Access Object (DAO)

ถือว่าเป็นวิธีที่ล้ำสมัยแล้ว โดยมีแนวคิดในการติดต่อหรือเข้าถึงข้อมูลในฐานข้อมูลผ่านทางองค์ประกอบต่างๆ ในฐานข้อมูล เช่น ฟیلด์ (Field) , เร็คคอร์ด (Record) , ความสัมพันธ์ระหว่างตาราง (Relation) เป็นต้น โดยจะแทนแต่ละองค์ประกอบเหล่านั้นด้วยออบเจกต์ (Object) และควบคุมออบเจกต์ต่างๆ เหล่านี้โดยการเขียนโค้ด

แม้จะทำงานได้ดีกว่า อีซระกว่า แต่มีความยุ่งยากในการเขียนโปรแกรมด้วยเช่นกัน อีกทั้งเป็นเทคโนโลยีที่เก่าแล้วคือ เน้นเฉพาะระบบฐานข้อมูลที่เป็นตาราง (โดยเฉพาะ Access รุ่นเก่าๆ) แต่ว่าการเก็บข้อมูลในปัจจุบัน ถูกจัดเก็บในสภาพแวดล้อมแตกต่างกันมากมาย เช่น รูปภาพ (Image) , ข้อความ (Text) และรูปแบบอื่นๆ อีกมากมายทำให้ต้องสร้างออบเจกต์ใหม่ๆ ขึ้นมาเรื่อยๆ แต่นั่นไม่ใช่ที่ทำได้ง่าย และกลายเป็นข้อความที่สำคัญของ DAO ตัวอย่างออบเจกต์ในกลุ่มนี้ เช่น ออบเจกต์ RecordSet , ออบเจกต์ TableDef , คอนเล็กชัน Fields เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11.3 แบบที่ 3 ติดต่อผ่าน ODBC โดยตรง (ODBC Direct)

เป็นการติดต่อกับฐานข้อมูลแบบ 32 บิตที่สนับสนุนมาตรฐาน ODBC (Open DataBase Connectivity) ที่ JET Engine (กลไกการติดต่อกับฐานข้อมูลของไมโครซอฟต์ ซึ่งเน้นที่ Access) ไม่สามารถจัดเก็บได้ เช่นฐานข้อมูลของ Oracle , ฐานข้อมูลของ Microsoft SQL Server เป็นต้น ซึ่งเป็นการติดต่อเฉพาะฐานข้อมูลที่มีการเก็บข้อมูลในรูปแบบตารางเท่านั้น

ไมโครซอฟต์เองได้สร้างออบเจกต์ขึ้นมาอีกชุดหนึ่งที่ชื่อว่า Remote Data Object หรือเรียกสั้นๆ ว่า RDO เพื่อใช้สำหรับติดต่อกับฐานข้อมูลแบบเครือข่าย เช่น Oracle , SQL Server , DB2 ฯลฯ) ซึ่งเป็นไปตามสถาปัตยกรรม n-Tier (n เทียร์) เช่น ไคลเอนต์ / เซิร์ฟเวอร์ (2-Tier ทุเทียร์) หรือ Middle Tier (3-Tier ทรีเทียร์) โดยอาศัยมาตรฐาน ODBC ในการเชื่อมโยง ที่เกี่ยวข้องกับ Visual Basic คือ

- คอนโทรล Remote Data (Remote Data Control – RDC)
- ออบเจกต์ Remote Data (Remote Data Object – RDO)

2.11.4 แบบที่ 4 เข้าถึงข้อมูลโดยอาศัยเทคโนโลยี OLEDB

เป็นรูปแบบการติดต่อกับฐานข้อมูลผ่านทางกลุ่มออบเจกต์ในโมเดล ADO ซึ่งใช้ OLEDB Provider เป็นกลไกในการเข้าถึงข้อมูลในฐานข้อมูลแทน JET Engine โดยเปลี่ยนจากมุมมองการติดต่อกับฐานข้อมูลแบบเดิมๆ ที่ต้องกำหนดชนิดของฐานข้อมูลมาเป็นมองที่รูปแบบของการเชื่อมต่อ (Connection) เข้ากับฐานข้อมูล

OLEDB เป็นเทคโนโลยีล่าสุดที่ไมโครซอฟต์วางตำแหน่งให้เป็นยุคต่อไปในการเข้าถึงข้อมูลในฐานข้อมูลที่มีรูปแบบการเก็บข้อมูลสารพัดรูปแบบซึ่งไม่ได้จำกัดตัวเองเฉพาะตาราง โดย OLEDB เป็นส่วนหนึ่งของสถาปัตยกรรม Universal Data Access (UDA) ของไมโครซอฟต์

ใน Visual Basic สามารถใช้ OLEDB ได้ 2 แบบ โดยการใช้

1. คอนโทรล ADO Data (ADO Data Control) ร่วมกับกลุ่มของ ActiveX Bound Controls ซึ่งสังเกตจากชื่อของคอนโทรล DataGrid , Datalist , Hierarchical FlexGrid เป็นต้น
2. ชุดออบเจกต์ ActiveX Data (ActiveX Data Object – ADO) โดยการเขียนโปรแกรมควบคุม

บทที่ 3

โปรแกรมการจัดการตรวจสอบลายนิ้วมือ

3.1 ลักษณะโปรแกรมการจัดการตรวจสอบลายนิ้วมือ

โปรแกรมการตรวจสอบลายนิ้วมือได้ถูกพัฒนาอย่างต่อเนื่องจนสามารถนำมาใช้งานได้อย่างสะดวกในการเข้าถึงโปรแกรมการจัดการด้วย Visual Basic โดยการพัฒนาให้เป็นไฟล์ ไคโนมิกส์ลิงค์ไอบารี ทั่วๆ ไปที่มีนามสกุลเป็น *.OCX เป็น Tool ช่วยให้การสร้างแอปพลิเคชันด้วย Visual Basic ทำได้ง่ายและรวดเร็วยิ่งขึ้น

Event ที่พัฒนาจะนำไปใช้ดังต่อไปนี้

1. เก็บข้อมูลลายนิ้วมือ
2. ตรวจสอบข้อมูลลายนิ้วมือ
3. นำภาพลายนิ้วมือแสดงทาง Dialog

3.2 ลักษณะโปรแกรมตรวจสอบลายนิ้วมือ

ลักษณะการจัดเก็บของเครื่องสแกนลายนิ้วมือ

1. เครื่องสแกนลายนิ้วมือ นั้นต้องการรายละเอียด เฉพาะของลายนิ้วมือของแต่ละบุคคล เพื่อนำไปสร้างและแสดงผลที่ Image Window

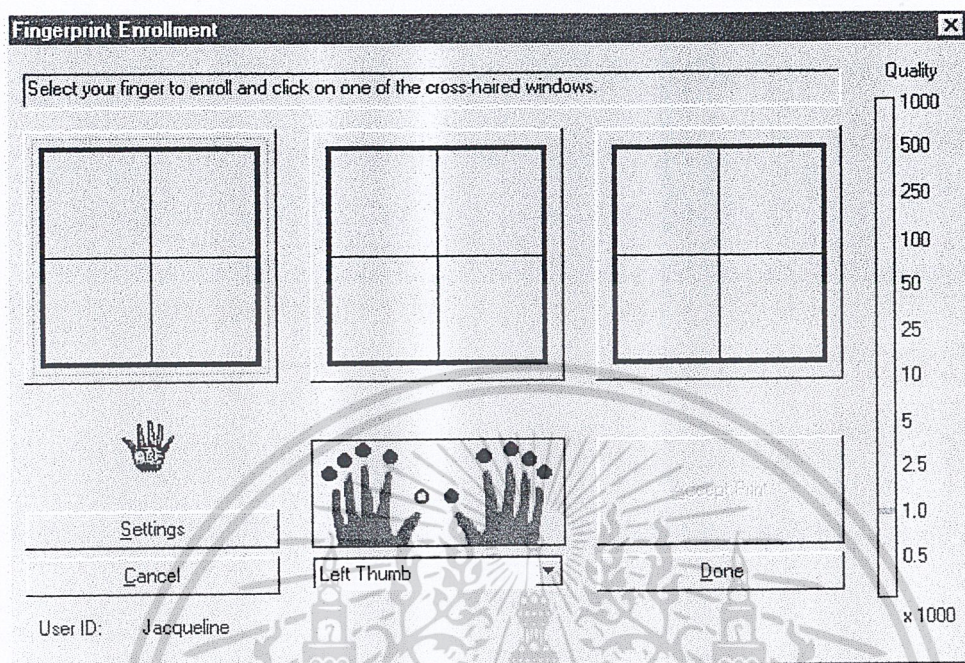
เครื่องสแกนลายนิ้วมือจะระบุลายนิ้วมือของแต่ละบุคคล โดยเริ่มจากการแสดงลักษณะของมือ ที่ภาพด้านข้างของ ไดอะล็อก โดยที่รูปมือจะแสดงรายละเอียดเฉพาะของนิ้วมือแต่ละนิ้วของแต่ละบุคคล และจะแสดงสถานะของการบันทึกลายนิ้วมือในช่วงวงกลมเหนือรูปมือ โดยที่จะแสดงสถานะด้วยสี

สีดำ : ลายนิ้วมือไม่สามารถบันทึก

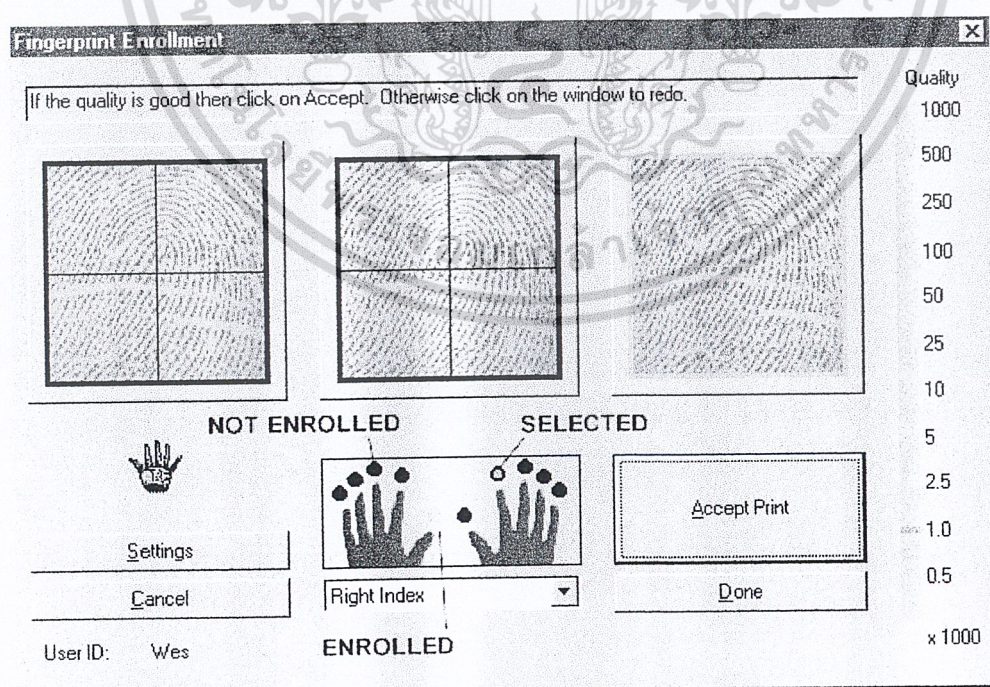
สีเขียว : ลายนิ้วมือที่สามารถบันทึก

สีแดง : ลายนิ้วมือที่ถูกเลือกเพื่อที่จะทำการบันทึก

ภาพตัวอย่างเพื่อความเข้าใจ



รูปที่ 3.1 แสดงฟอร์มของ โปรแกรมการเก็บข้อมูลลายนิ้วมือ



รูปที่ 3.2 แสดงฟอร์มลักษณะการจัดเก็บข้อมูลลายนิ้วมือ
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. คลิกเมาส์ปุ่มซ้ายมือที่บริเวณปลายนิ้วมือหรือบริเวณวงกลมเหนือปลายนิ้วมือที่ต้องการทำการบันทึกที่รูปมือ หรืออาจจะเลือกจาก Combo Box ก็ได้

3. Image Window ทั้ง 3 รูป ใช้สำหรับแสดงภาพลายนิ้วมือที่ได้มาจากการสแกนลายนิ้วมือด้วยเครื่องสแกนลายนิ้วมือ

4. คลิกเมาส์ที่บริเวณ Image Window แล้วเคาะ Space Bar เพื่อเริ่มทำการบันทึกลายนิ้วมือ จากนั้นนำนิ้วมือที่ต้องการสแกนวางลงบนเครื่องสแกนลายนิ้วมือ กดนิ้วมือให้สัมผัสกับหน้าจอเครื่องสแกนให้มากที่สุด

5. วางนิ้วมือให้อยู่กับที่บนตัวสแกน 1 ถึง 2 วินาที เพื่อให้ Software ได้ทำการตรวจสอบ จากนั้นที่ Image Window จะมีการเปลี่ยนแปลงของสีเพื่อแสดงสถานะของการตรวจสอบลายนิ้วมือแบบอัตโนมัติ

สีดำ : ไม่พบลายนิ้วมือ

สีแดง : ลายนิ้วมือมีการเคลื่อนที่

สีเหลือง : พิมพ์นิ้วมือเรียบร้อยแล้ว

6. ทำแบบเดิมกับ Image Window เพื่อให้ได้ลายนิ้วมืออีก 2 ภาพ โดยใช้นิ้วมือเดียวกับครั้งแรก

7. ลายนิ้วมือที่มีคุณภาพที่ดีที่สุดจาก 3 Image Window จะถูกเลือกเพื่อทำการบันทึก โดยที่ลายนิ้วมือที่ดีที่สุดที่ถูกเลือกเพื่อทำการบันทึกจะมีการเปลี่ยนแปลงเป็นสีเขียวที่บริเวณ Image Window

8. แถบแสดงคุณภาพลายนิ้วมือ (Quality Bar) ต้องมีค่ามากกว่าหรือเท่ากับค่าความปลอดภัย (Security Level) ที่ตั้งไว้ ซึ่งระดับคุณภาพแบ่งเป็นดังนี้

คุณภาพระดับ 1 : 1 ใน 10000 ของค่าความปลอดภัย (ในการตรวจสอบ 10000 ครั้ง ผิดพลาด 1 ครั้ง)

คุณภาพระดับ 2 : 1 ใน 100000 ของค่าความปลอดภัย (ในการตรวจสอบ 100000 ครั้ง ผิดพลาด 1 ครั้ง)

คุณภาพระดับ 3 : 1 ใน 1000000 ของค่าความปลอดภัย (ในการตรวจสอบ 1000000 ครั้ง ผิดพลาด 1 ครั้ง)

การสแกนเพื่อให้ได้คุณภาพของลายนิ้วมือที่ดีที่สุด

- การสแกนลายนิ้วมือต้องให้จุดกึ่งกลางของลายนิ้วมือ อยู่ใกล้บริเวณจุด กึ่งกลางของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ควรตรวจสอบว่านิ้วมือของผู้ใช้แห้งเกินไปหรือเปล่า เวลาสแกนลายนิ้วมืออาจทำให้ลายนิ้วมือที่ปรากฏบน Image Window ขาดหายหรือเลื่อนราง

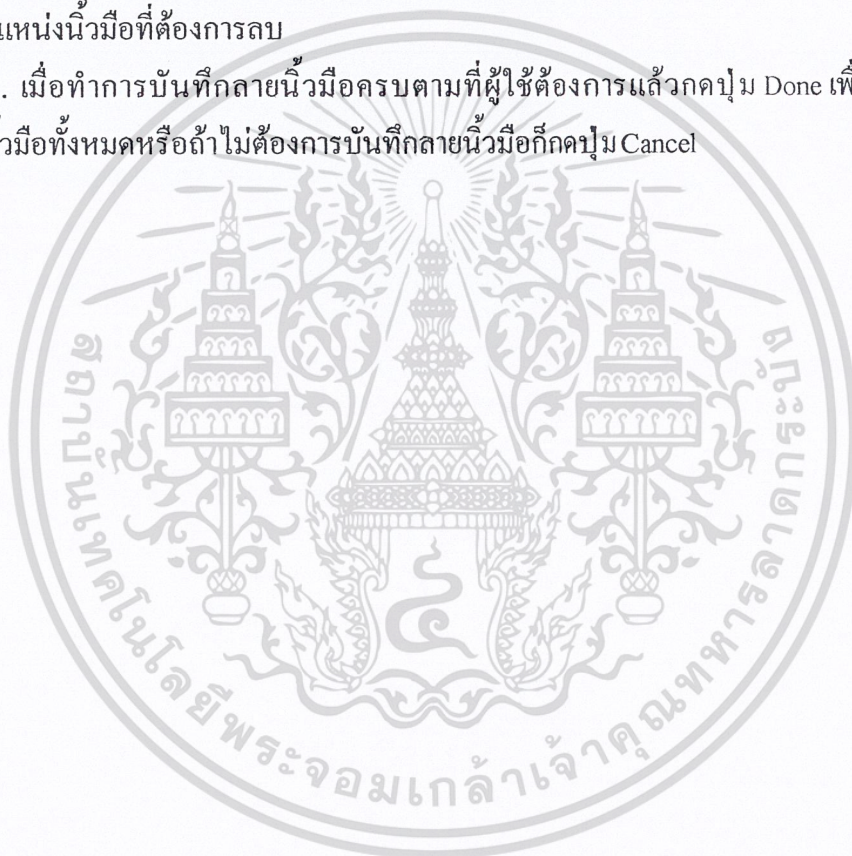
- ถ้า Image Window ปรากฏคำว่า “ Bloom ” แสดงว่านิ้วมือของผู้ใช้มีความชื้นมากเกินไปหรือออกแรงกดมากเกินไป ควรทำให้มือแห้งก่อนหรือออกแรงกดให้น้อยลง

- ควรตรวจสอบทำความสะอาดบริเวณตัวสแกนก่อนใช้งาน

9. เมื่อระดับคุณภาพของลายนิ้วมือผ่านการยอมรับ (ปุ่ม Accept Print จะสามารถกดได้) และผู้ใช้พอใจกับคุณภาพลายนิ้วมือแล้ว กดปุ่ม Accept Print เพื่อทำการบันทึกลายนิ้วมือ

10. การลบการบันทึกลายนิ้วมือทำได้โดยการคลิกเมาส์ปุ่มขวาที่บริเวณวงกลมเหนือรูปมือ ในตำแหน่งนิ้วมือที่ต้องการลบ

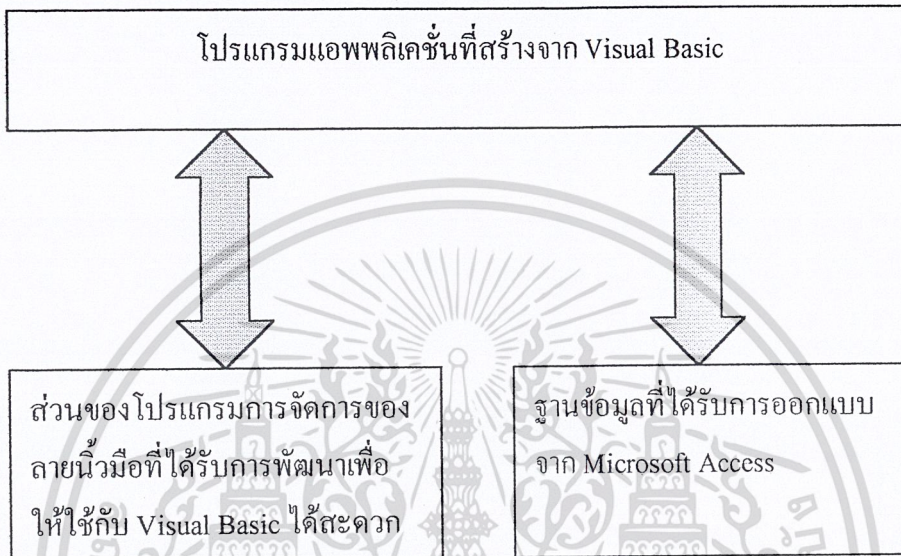
11. เมื่อทำการบันทึกลายนิ้วมือครบตามที่ผู้ใช้ต้องการแล้วกดปุ่ม Done เพื่อทำการบันทึกลายนิ้วมือทั้งหมดหรือถ้าไม่ต้องการบันทึกลายนิ้วมือก็กดปุ่ม Cancel



บทที่ 4

โครงสร้างโดยรวมของโปรแกรม

4.1 โครงสร้างการออกแบบโปรแกรมระบบจัดการฐานข้อมูล

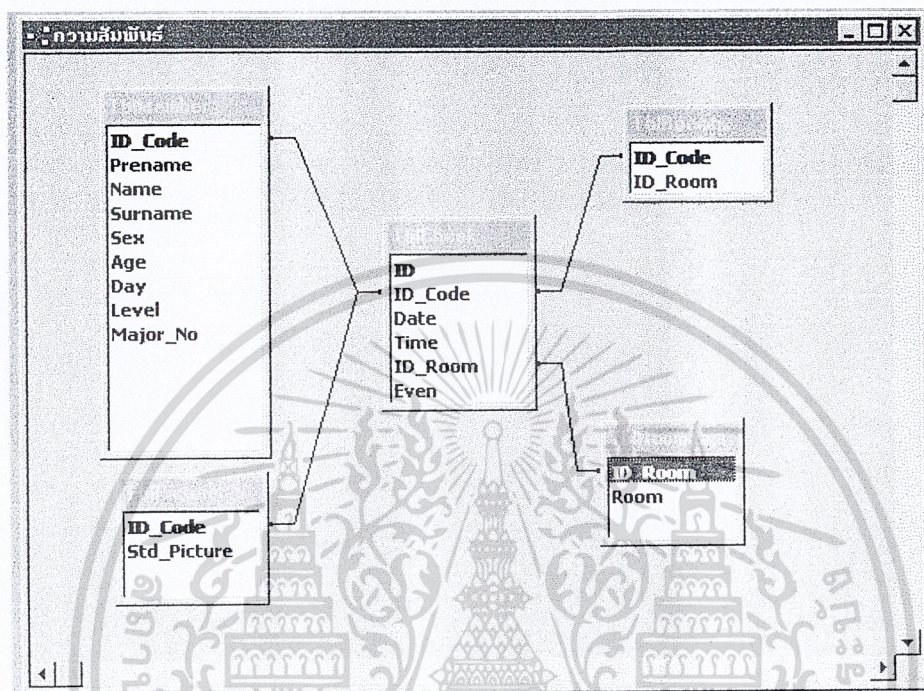


รูปที่ 4.1 แสดงโครงสร้างการทำงานของโปรแกรม

โปรแกรมการจัดการของลายนิ้วมือเป็นโปรแกรมที่ได้รับการพัฒนาจนสามารถใช้โปรแกรม Visual Basic พัฒนาได้ง่ายในการเข้าถึงข้อมูล Visual Basic จะมองโปรแกรมส่วนนี้เป็นเช่นเดียวกับ Tool ทั่วไป โดยจะพัฒนามาเป็น ไฟล์ไดนามิกส์ลิงก์ที่มีนามสกุลเป็น *.OCX ดังนั้น Visual Basic จึงสามารถควบคุมโปรแกรมการจัดการลายนิ้วมือได้อย่างสะดวก โดยรูปแบบการควบคุมด้วย Visual Basic จะมีเหตุการณ์อยู่ดังนี้

1. เก็บข้อมูลลายนิ้วมือซ้าย
2. เก็บลายนิ้วมือขวา
3. เปรียบเทียบลายนิ้วมือซ้าย
4. เปรียบเทียบลายนิ้วมือขวา
5. นำลายนิ้วมือมาแสดงที่ Dialog

ในการออกแบบโปรแกรมระบบตรวจสอบการเข้า-ออกนั้นจะทำการออกแบบฐานข้อมูลด้วย MicroSoft Access เนื่องจากเป็นโปรแกรมที่เข้าใจง่ายและมีประสิทธิภาพเพียงพอกับการนำมาใช้งานด้านฐานข้อมูลลักษณะฐานข้อมูลที่ออกแบบจะมีโครงสร้างดังรูป



รูปที่ 4.2 แสดงความสัมพันธ์ของตาราง

จะมี โครงสร้างตารางอยู่ 5 ตารางซึ่งตารางทั้งหมดจะมีความสัมพันธ์กันดังรูป การจัดการความสัมพันธ์ดังกล่าวเพื่อลดความซ้ำซ้อนของข้อมูลที่ใช้ในการจัดเก็บ และการเข้าถึงข้อมูล ทำได้ง่ายและสะดวกขึ้น

ตาราง TblMember จัดว่าเป็นตารางหลักสำหรับจัดเก็บประวัติของผู้ใช้ โดยจะจัดเก็บข้อมูลเบื้องต้นไว้ดังนี้

1. รหัสประจำตัวผู้ใช้ (ID_Code) จะเป็นตัวกำหนดการเข้า-ออกของแต่ละบุคคล
2. คำนามหน้าชื่อ (Purname)
3. ชื่อ (Name)
4. นามสกุล (Surname)
5. เพศ (Sex)
6. อายุ (Age)
7. ระดับผู้ใช้ห้อง (Level)
8. ภาควิชาที่เกี่ยวข้อง (Major_No)

ID_Code	Prenome	Name	Surname	Sex	Age	Day	Level	Major_No
+ 41012100	นาย	วัชรพล	ประกอบกิจ	ชาย	26	28/3/44	เจ้าหน้าที่	เครื่องกล
+ 41012102	นาย	สนเกียรติ	เกิดเจริญ	ชาย	24	23/3/44	นักศึกษา	วัดคุณ
+ 41012103	นาย	กิตติโชค	พรหมรัตน์	ชาย	23	23/3/44	นักศึกษา	วัดคุณ
+ 41012105	นาย	กนกพร	การกิจ	ชาย	20	2/2/2001	นักศึกษา	ไฟฟ้า
+ 41012106	นาย	กิตติ	พรหม	ชาย	25	25/3/44	นักศึกษา	วัดคุณ

รูปที่ 4.3 แสดงรายละเอียดข้อมูลในตาราง TblMember

ตาราง TblPicture เป็นตารางสำหรับจัดเก็บรูปภาพและจะจัดเก็บพร้อมรหัสของแต่ละบุคคลแต่ที่จริงการเก็บรูปภาพจะไม่ได้เก็บข้อมูลของภาพแต่จะเป็นการเก็บที่อยู่ของรูปภาพดังรูป

ID_Code	Std_Picture
+ 123456789	e:\picture\un.jpg
+ 41012102	e:\picture\un.jpg
+ 41012103	e:\nung.jpg
+ 41012105	e:\nung.jpg

รูปที่ 4.4 แสดงรายละเอียดการจัดเก็บข้อมูลในตาราง TblPicture

ตารางจัดเก็บข้อมูลห้อง (TblRoomData) จะเก็บข้อมูลห้องที่ต้องการควบคุมพร้อมกับกำหนดรหัสให้กับห้องที่ต้องการควบคุมมีรายละเอียดดังรูป

ID_Room	Room
+ 101	PLC
+ 102	CONTROL
+ 103	Power Eletronic
+ 104	Computer
+ 105	CNC
+ 106	Process
+ 107	ห้อง I-103
+ 108	ห้องปฏิบัติการ 1
+ 203	ระบบควบคุม

รวมมี: 14 | 10 | จาก 10

รูปที่ 4.5 แสดงรายละเอียดการจัดเก็บข้อมูลในตาราง TblRoomData

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางบันทึกข้อมูลที่กำหนดการใช้ห้องของแต่ละบุคคล (TblRoomFix) โดยจะจัดการจัดเก็บของแต่ละบุคคลด้วยรหัสประจำตัวคงรูป

ID_Code	ID_Room
+ 41012102	101102103104105106107108109200
+ 41012103	101102103104
+ 41012105	102104105106107
+ 41012106	101102103108

รวม: 4 จาก 4

รูปที่ 4.6 แสดงตารางกำหนดการใช้ห้อง

รหัสของห้องที่ถูกเลือกจะถูกจัดเรียงอยู่ใน Record เดียวกันและในการแยกแยะว่าผู้ใช้สามารถใช้ห้องอะไรได้บ้างใช้เทคนิคจากการเขียน โปรแกรม

ตารางเช็คการใช้ห้อง (TblCheck) ตารางนี้มีไว้สำหรับบันทึกการใช้งานของแต่ละบุคคลจะบันทึกข้อมูลดังนี้

1. รหัสประจำตัว
2. วันที่เข้าใช้ห้อง
3. เวลา
4. ชื่อห้อง
5. เหตุการณ์ที่เกิดขึ้น

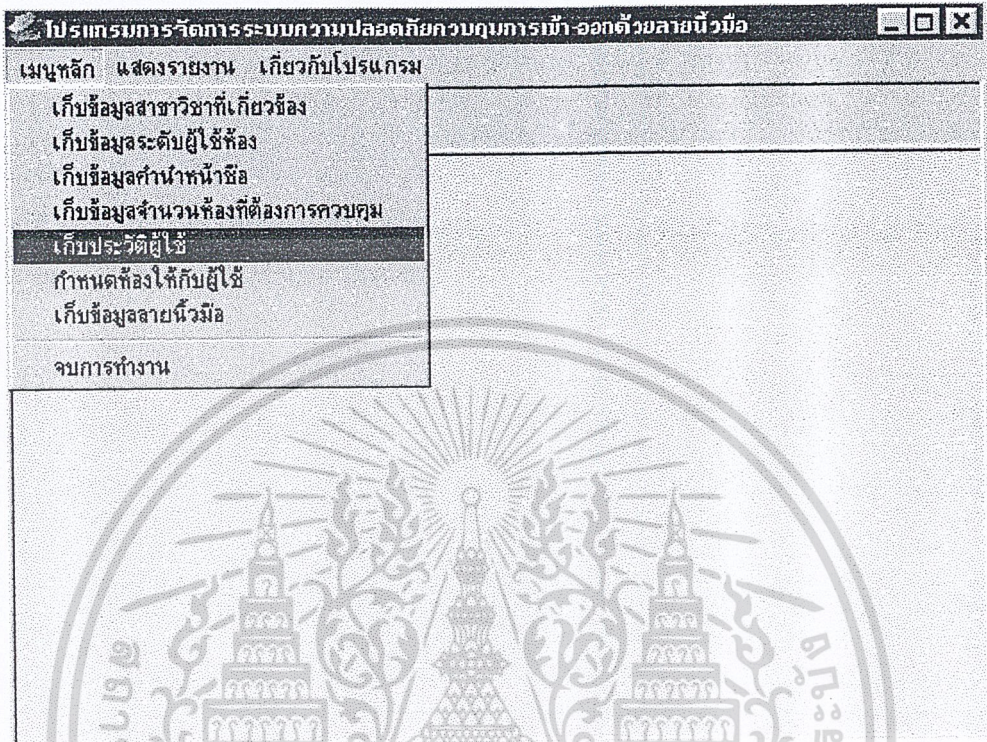
ID	ID_Code	Date	Time	ID_Room	Even
1	41012102	1 มกราคม 2001	16:00:00	101	ok
2	41012102	2 เมษายน 2001	16:00:00	102	ok
3	41012103	22 กุมภาพันธ์	7:00:00	101	ok

* (AutoNu
รวม: 1 จาก 3

รูปที่ 4.7 ตารางแสดงการบันทึกข้อมูลเหตุการณ์ของผู้ใช้ห้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

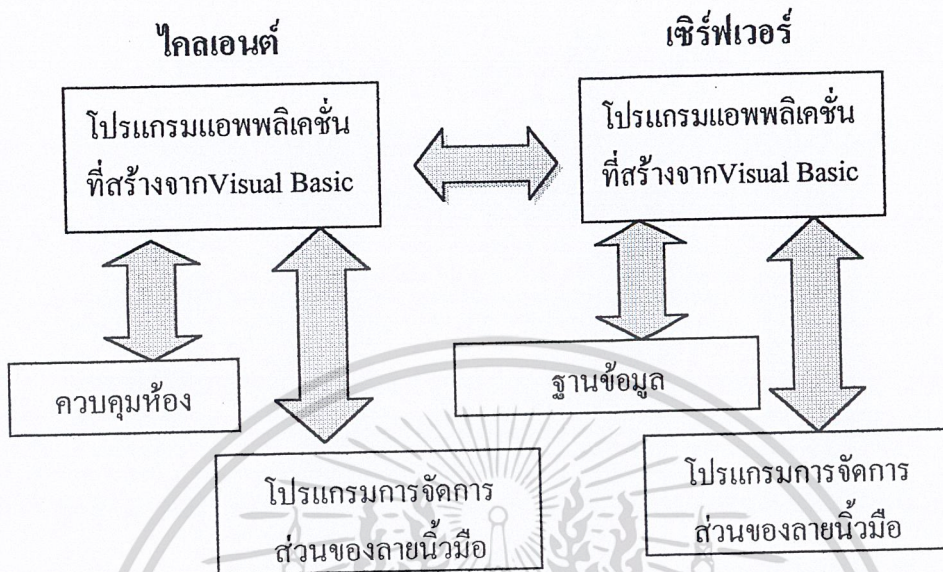
ฐานข้อมูลที่สร้างขึ้นจะใช้ Visaul Basic ในการเข้าถึงฐานข้อมูลโดยมีรูปแบบการทำงานของโปรแกรมจะมีเมนูหลักใช้ในการจัดเก็บข้อมูลต่างๆดังรูป



รูปที่ 4.8 แสดงฟอร์มหลักของโปรแกรมการจัดการระบบความปลอดภัย

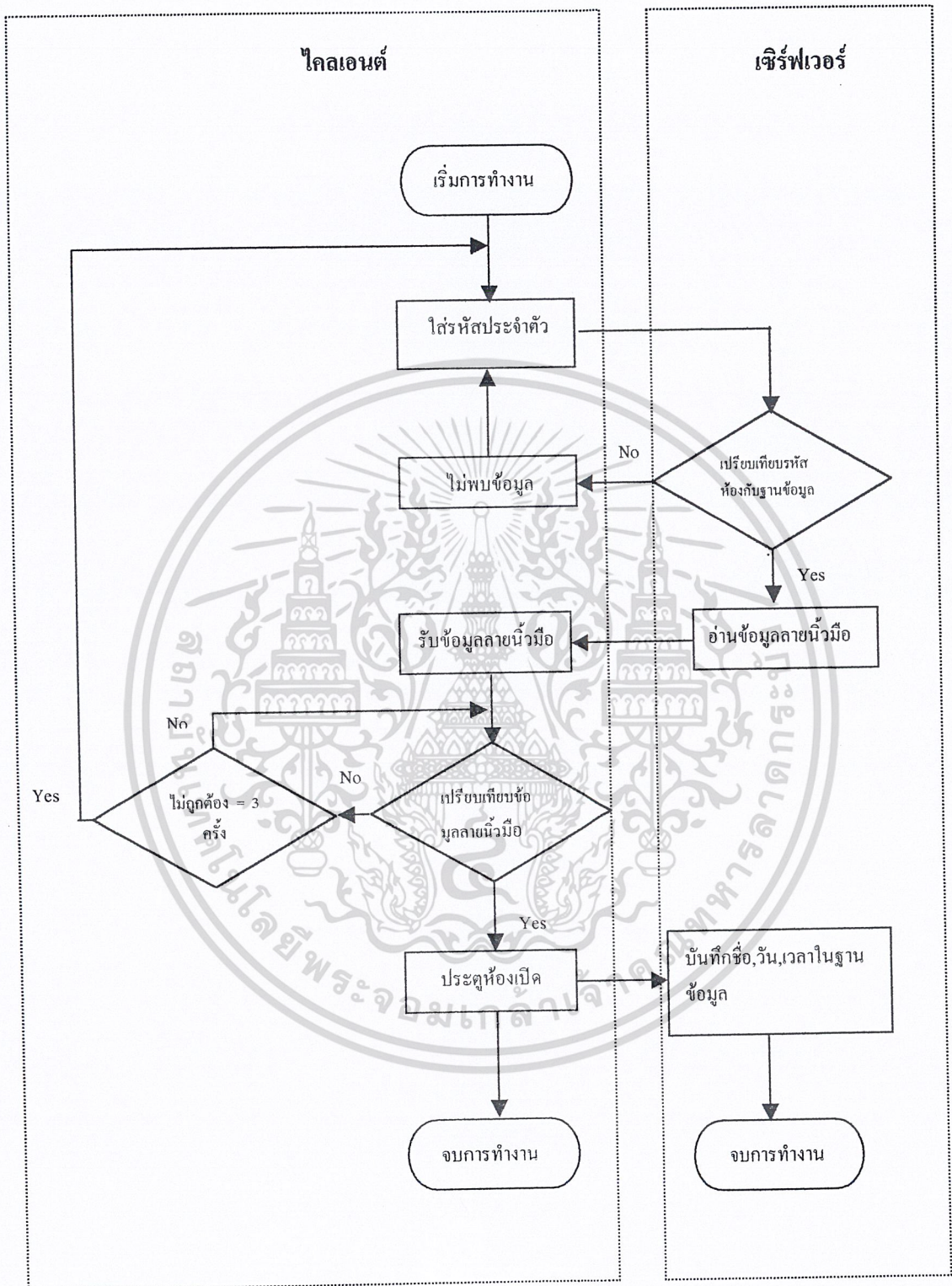
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ลักษณะโครงสร้างการออกแบบทางระบบเน็ตเวิร์ก



รูปที่ 4.9 แสดงโครงสร้างของโปรแกรมควบคุมทางเน็ตเวิร์ก

ส่วนของโปรแกรมแอปพลิเคชันทาง ไคลเอนต์ จะทำการรับข้อมูลรหัสประจำตัว และลายนิ้วมือส่งข้อมูลให้กับ เซิร์ฟเวอร์ เพื่อทำการตรวจสอบตามเงื่อนไขที่กำหนดให้ลักษณะโปรแกรมที่ทำงาน



รูปที่ 4.10 แสดง Flow Chart การทำงานของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

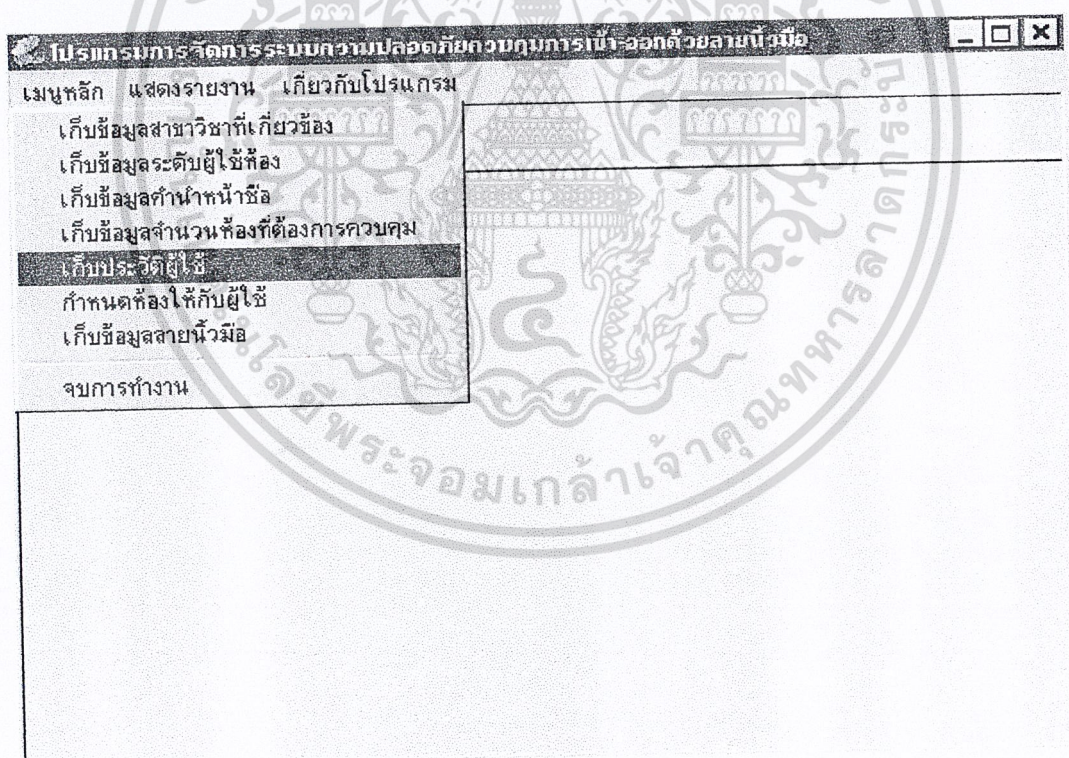
บทที่ 5

ขั้นตอนการใช้งานของโปรแกรมและผลการทดลอง

5.1 จัดเก็บข้อมูลต่างๆที่ได้ทำการออกแบบเก็บไว้เป็นฐานข้อมูลที่เครื่อง เซิร์ฟเวอร์

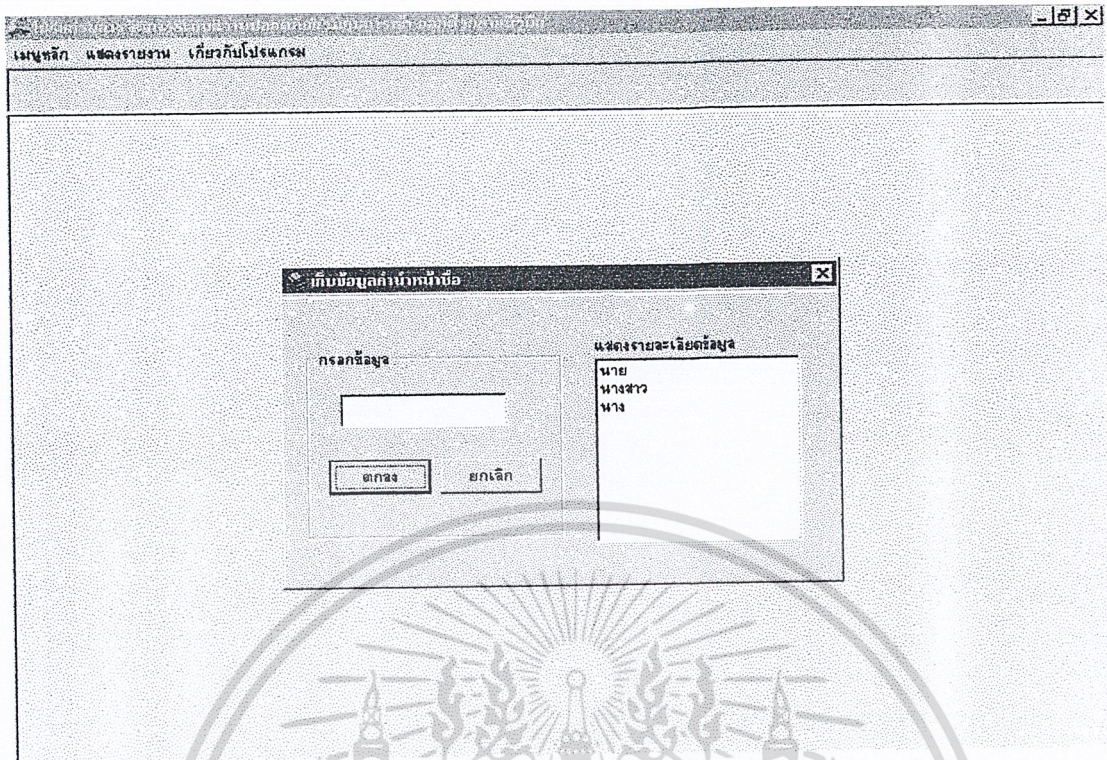
เมนูหลัก ใช้สำหรับเก็บฐานข้อมูลต่างๆ ของผู้ใช้

1. เก็บข้อมูลสาขาวิชาที่เกี่ยวข้อง
2. เก็บข้อมูลระดับของผู้ใช้
3. เก็บข้อมูลค่านำหน้าชื่อ
4. เก็บข้อมูลจำนวนห้องที่ต้องการควบคุม
5. เก็บข้อมูลประวัติของผู้ใช้ เช่น ชื่อ , นามสกุล , รหัสประจำตัว , เพศ และ อายุ
6. กำหนดห้องให้กับผู้ใช้ที่เกี่ยวข้อง
7. เก็บข้อมูลลายนิ้วมือ

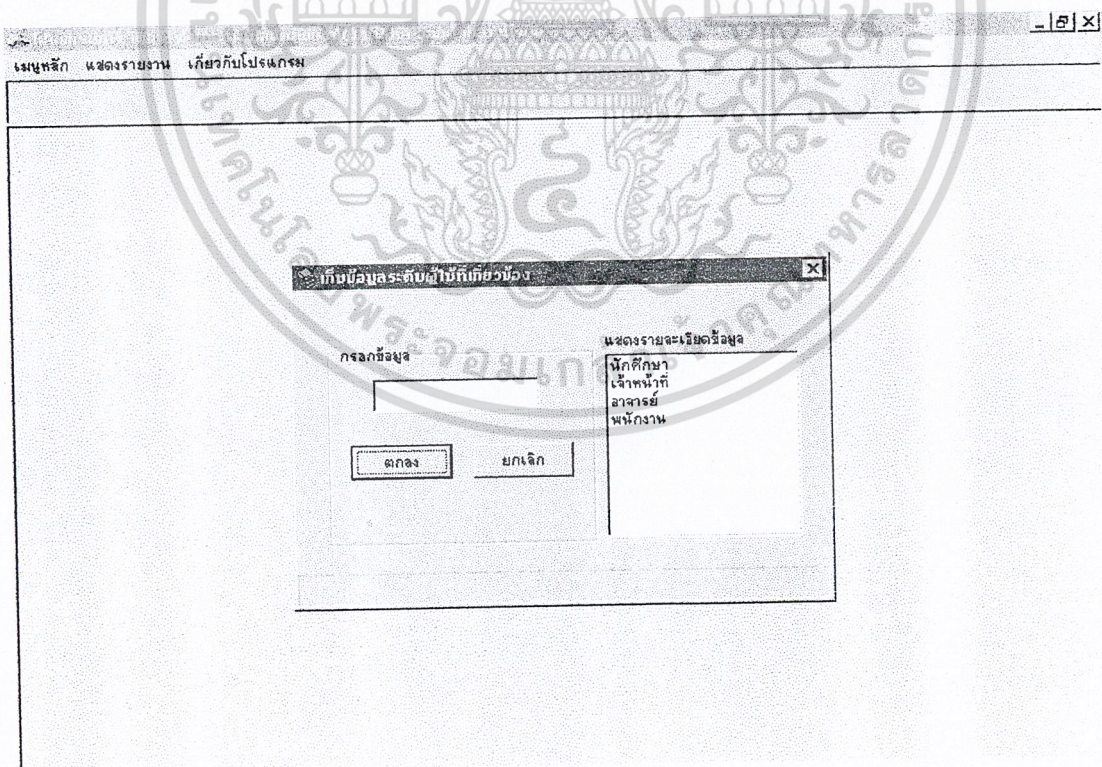


รูปที่ 5.1 แสดงฟอร์มเมนูหลักที่ใช้ในการจัดเก็บข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

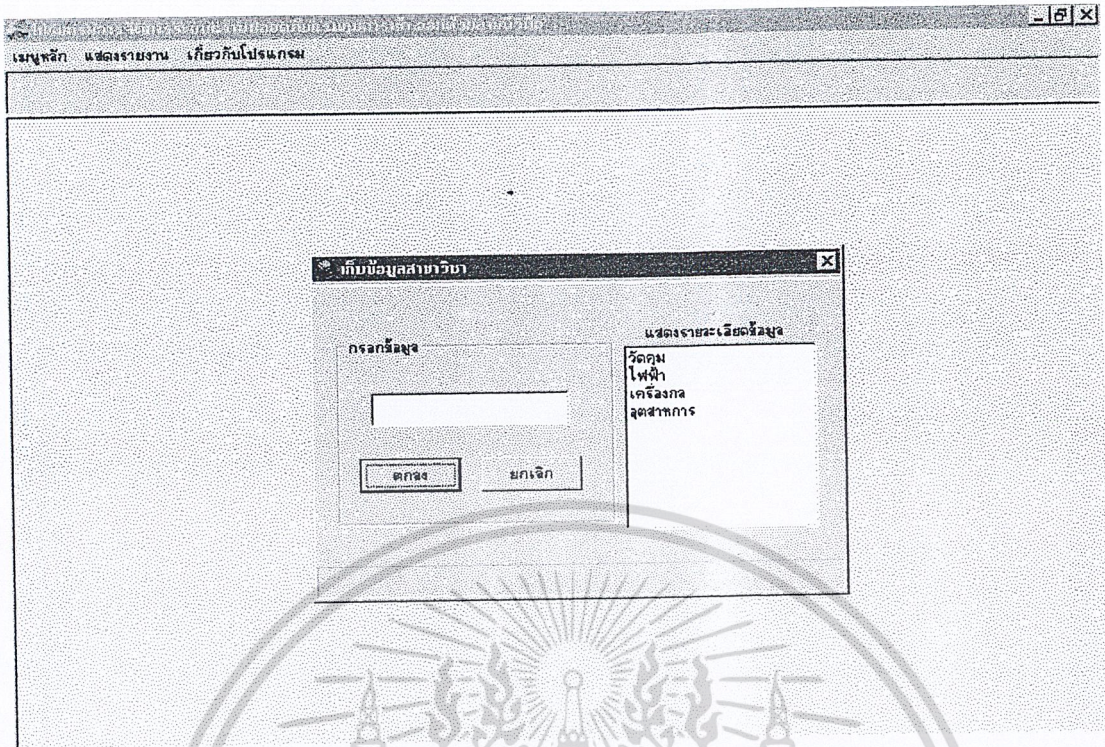


รูปที่ 5.2 แสดงฟอร์มเก็บข้อมูลค่านำหน้าชื่อ

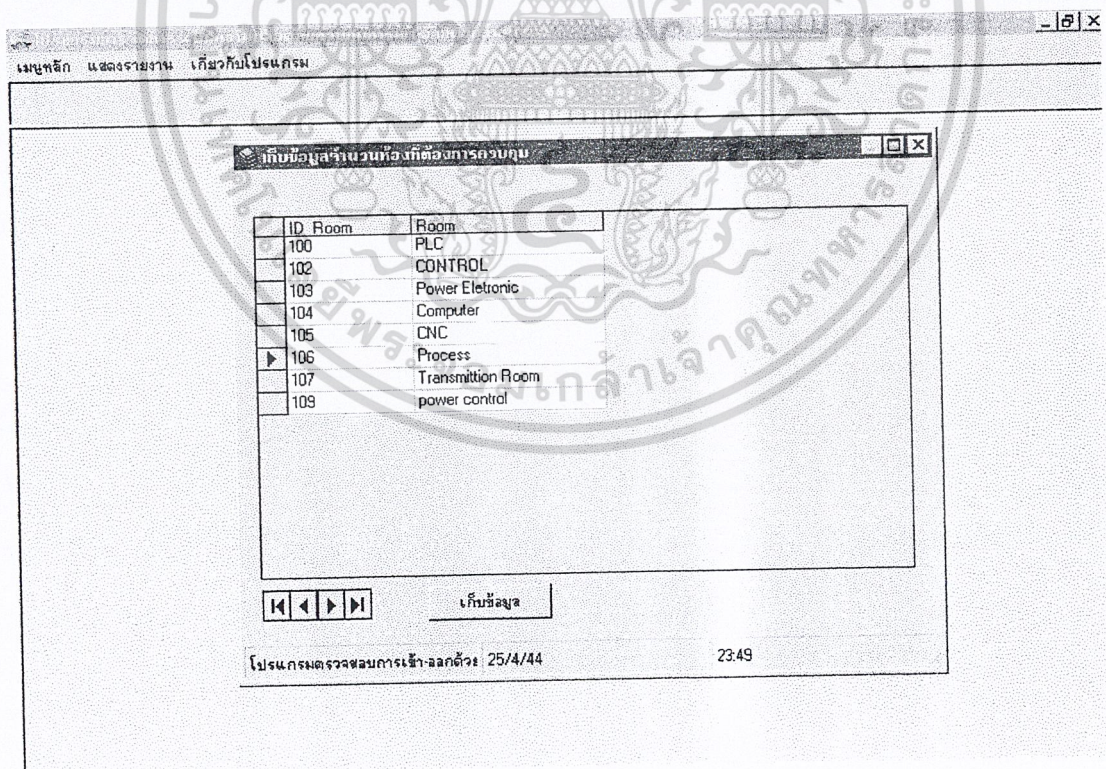


รูปที่ 5.3 แสดงฟอร์มเก็บข้อมูลระดับผู้ใช้ที่เกี่ยวข้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.4 แสดงฟอร์มเก็บข้อมูลสาขาวิชา



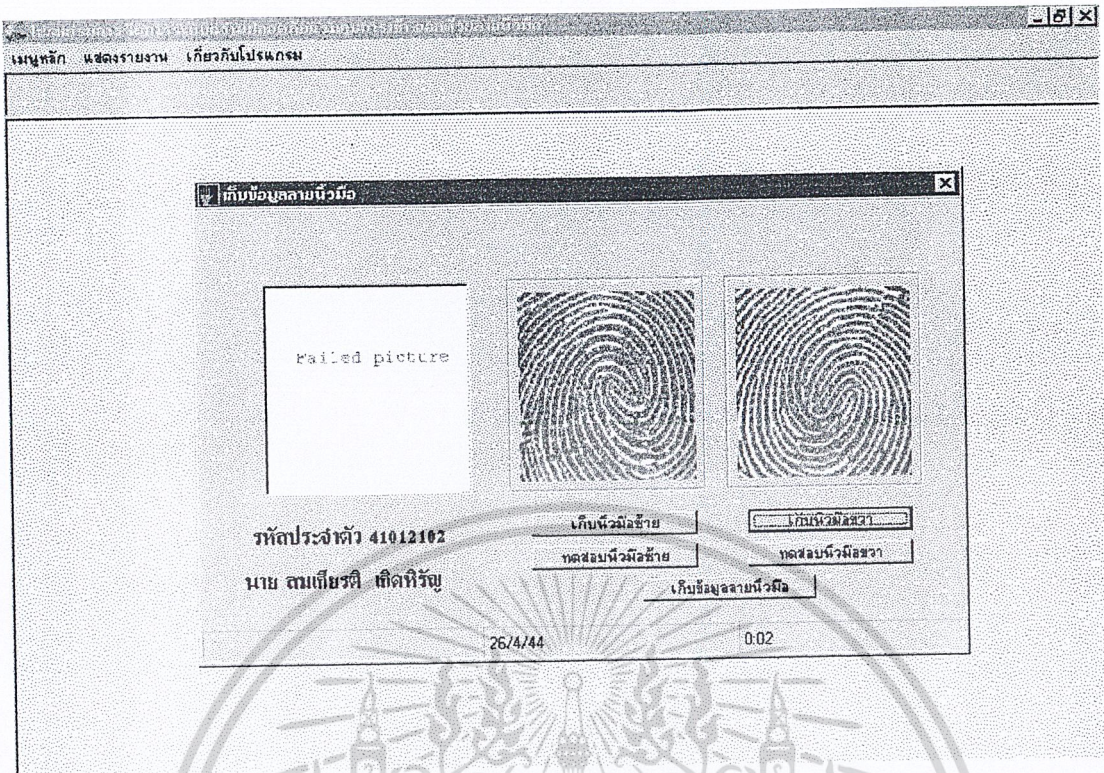
รูปที่ 5.5 แสดงฟอร์มเก็บข้อมูลจำนวนห้องที่ต้องการควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 5.6 แสดงฟอร์มเก็บประวัติผู้ใช้

รูปที่ 5.7 แสดงฟอร์มกำหนดห้องให้กับผู้ใช้

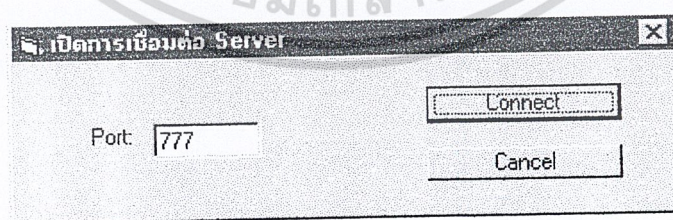
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.8 แสดงฟอร์มเก็บข้อมูลลายนิ้วมือ

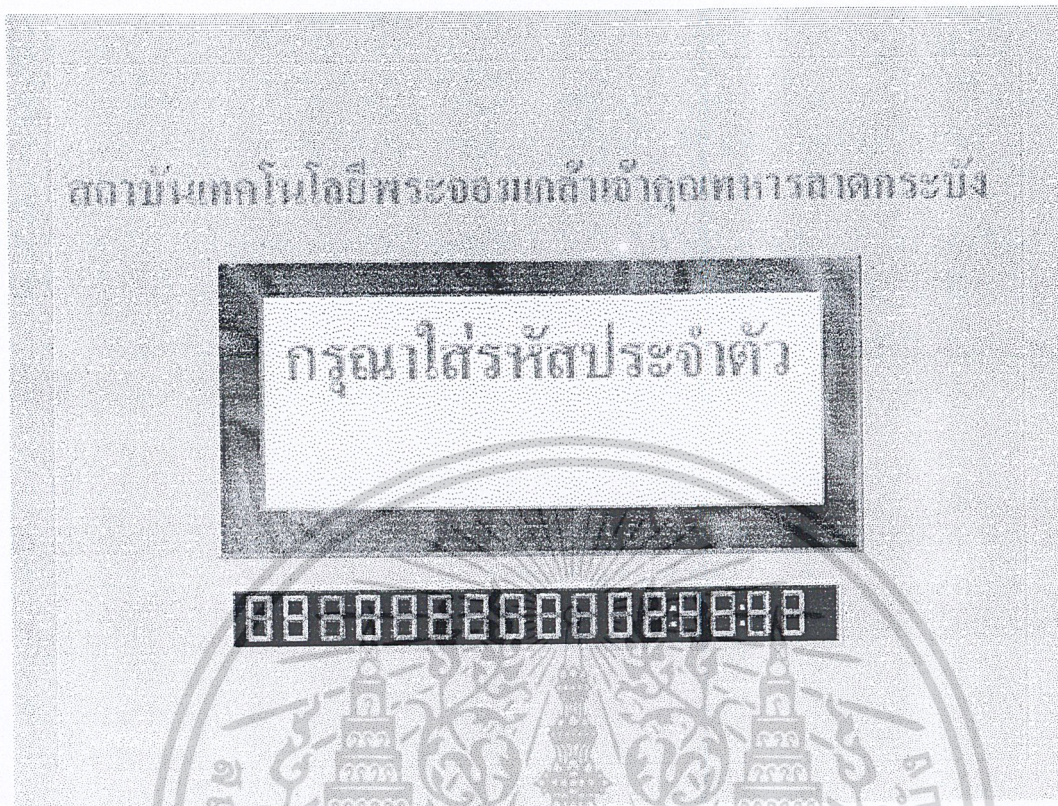
5.2 ทำการรันโปรแกรมบริการข้อมูลที่เครื่อง เซิร์ฟเวอร์

โปรแกรมส่วนนี้จะตรวจสอบการร้องขอข้อมูลจากเครื่องควบคุม (ไคลเอนต์) เมื่อมีการร้องขอข้อมูลโปรแกรมจะทำการตรวจสอบว่าเครื่องใดทำการร้องขอและจะทำการให้บริการข้อมูลตามขั้นตอนของโปรแกรมที่ได้ออกแบบไว้ โดยจะต้องกำหนดพอร์ตสำหรับให้บริการข้อมูล



รูปที่ 5.9 แสดงฟอร์มเปิดการเชื่อมต่อให้บริการข้อมูลที่เครื่องเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.13 แสดงฟอร์มลักษณะขั้นตอนรับข้อมูลรหัสประจำตัวจากผู้ใ้

เป็นฟอร์มรับข้อมูลรหัสประจำตัวของผู้ที่ต้องการเข้าใช้ห้อง เมื่อใ้รหัสประจำตัวเสร็จเรียบร้อยเครื่องควบคุม (ไมโครเอนต์) จะทำการส่งรหัสประจำตัวนี้ไปยังเครื่องเซิร์ฟเวอร์เพื่อทำการค้นหารหัสประจำตัวเมื่อพบรหัสประจำตัวก็จะทำการส่งข้อมูลลายนิ้วมือของผู้ใ้ที่มีรหัสประจำตัวดังกล่าวกลับมายังเครื่องควบคุม (ไมโครเอนต์) ถ้าไม่พบรหัสประจำตัวก็จะส่งข้อมูลให้เครื่องควบคุม (ไมโครเอนต์) แสดงข้อความไม่พบข้อมูล

เมื่อเครื่อง เซิร์ฟเวอร์ พบรหัสประจำตัวจะส่งข้อมูลลายนิ้วมือมายังเครื่องควบคุม (ไมโครเอนต์) จากนั้นเครื่องควบคุม (ไมโครเอนต์) จะแสดงหน้าต่างเพื่อให้เลือกลายนิ้วมือซ้ายหรือขวา เพื่อทำการตรวจสอบลายนิ้วมือต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อผู้ใช้เลือกลายนิ้วมือที่ต้องการตรวจสอบแล้ว เครื่องควบคุม (ไคลเอนต์) จะแสดงฟอร์มเพื่อให้ผู้ใช้ทำการการสแกนลายนิ้วมือ

เมื่อผู้ใช้ได้สแกนลายนิ้วมือเรียบร้อยแล้วเครื่องควบคุม (ไคลเอนต์) จะทำการเปรียบเทียบลายนิ้วมือของผู้ใช้กับลายนิ้วมือที่ได้รับจากเครื่องเซิร์ฟเวอร์ ซึ่งเป็นลายนิ้วมือที่เก็บไว้ที่ฐานข้อมูล จากรูปที่ 5.16 แสดงการ ำอนุญาตให้เข้าใช้ห้องเนื่องจากลายนิ้วมือไม่ถูกต้องซึ่งเครื่อง (ไคลเอนต์) จะอนุญาตให้สแกนลายนิ้วมือผิดได้ 3 ครั้ง ถ้าสแกนลายนิ้วมือผิดครบ 3 ครั้ง เครื่อง ไคลเอนต์ จะแสดงฟอร์มให้ใส่รหัสใหม่



รูปที่ 5.16 แสดงผลการตรวจสอบข้อมูลลายนิ้วมือกับฐานข้อมูลที่เครื่องเซิร์ฟเวอร์ที่ไม่ถูกต้อง

ถ้าเครื่องควบคุม (ไคลเอนต์) เปรียบเทียบลายนิ้วมือแล้วลายนิ้วมือถูกต้องก็จะแสดงฟอร์มอนุญาตให้เข้าใช้ห้องได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.17 แสดงผลการตรวจสอบข้อมูลลายนิ้วมือกับฐานข้อมูลเครื่องเซิร์ฟเวอร์ที่ถูกต้อง

5.4 แสดงรายงาน

5.4.1 แสดงข้อมูลห้องที่กำหนดให้กับผู้ใช้

โปรแกรมที่ได้ออกแบบเราสามารถแสดงรายละเอียดข้อมูลห้องที่ได้กำหนดให้กับผู้ใช้งานว่ามีจำนวนห้องใดบ้างที่สามารถเข้าใช้งานได้

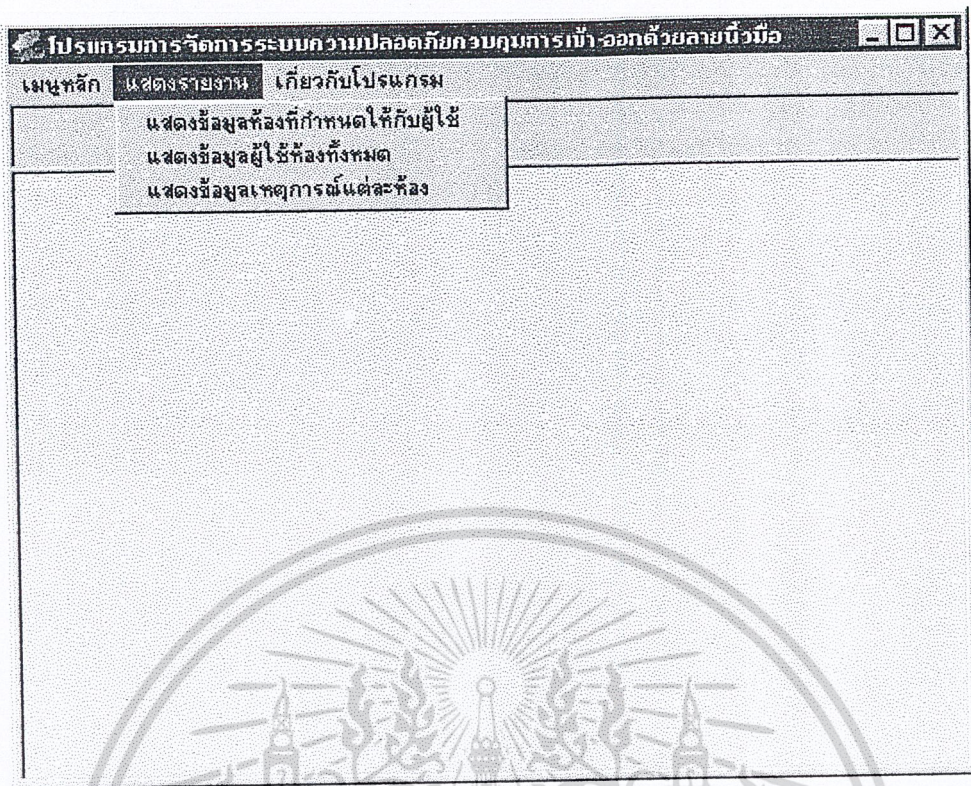
5.4.2 แสดงข้อมูลทั้งหมดในฐานข้อมูล

เมื่อต้องการแสดงรายละเอียดของข้อมูลทั้งหมดสามารถแสดงข้อมูลทั้งหมด ได้ดังรูปที่ 5.19

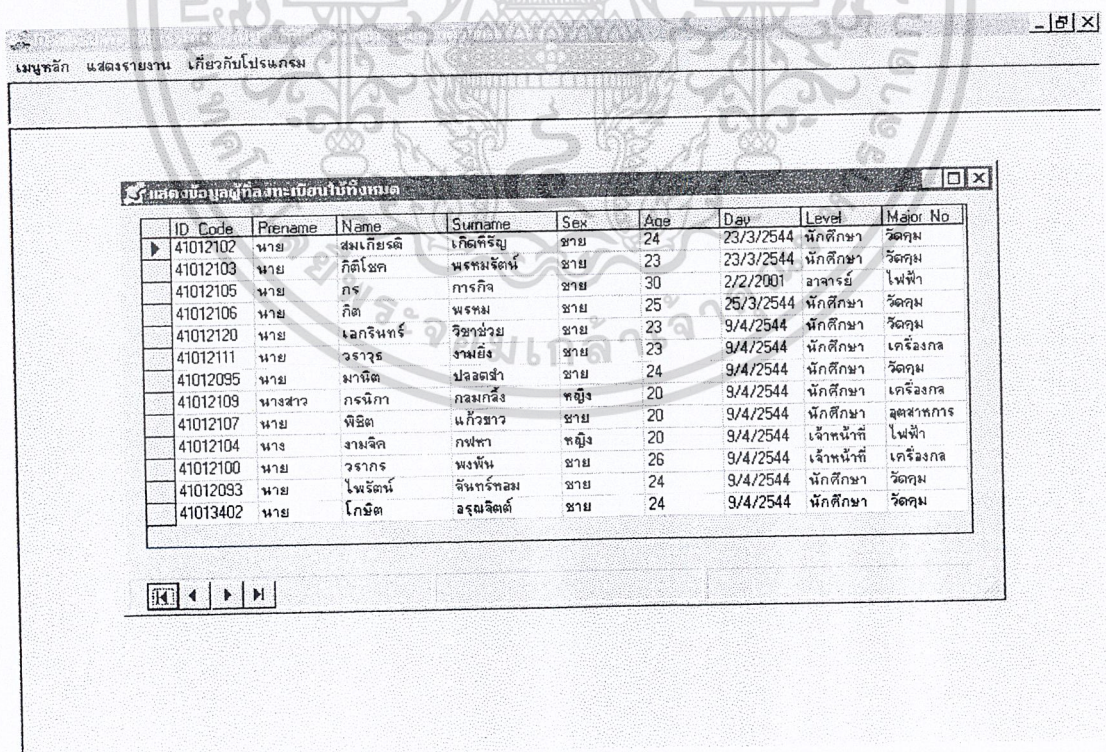
5.4.1 แสดงรายละเอียดการเข้าใช้ห้องแต่ละห้อง

เมื่อระบบมีการเชื่อมต่อเข้าด้วยกันเป็นระบบเน็ตเวิร์กทำให้ระบบทำงานตามขั้นตอนของโปรแกรมที่ออกแบบ ดังนั้นเราสามารถตรวจสอบการเข้าใช้ห้องของแต่ละห้องได้จากโปรแกรมที่ออกแบบดังรูปที่ 5.20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.18 แสดงฟอร์มเมนูสำหรับแสดงรายงาน



รูปที่ 5.19 แสดงฟอร์มสำหรับแสดงรายงานข้อมูลทั้งหมดที่เก็บไว้ในฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมนูหลัก แสดงรายงาน เกี่ยวกับโปรแกรม

ตรวจสอบข้อมูลห้อง

เมนู

แสดงรายละเอียดผู้ใช้ห้อง

ID Code	Prename	Name	Surname	Level	Major No	ID Room	Room	Date	Time	Even
▶ 41012102	นาย	สมเกียรติ	เกิดเจริญ	นักศึกษา	วัดชุม	100	PLC	6/4/2544	16:25:38	OK
41012102	นาย	สมเกียรติ	เกิดเจริญ	นักศึกษา	วัดชุม	100	PLC	8/4/2544	20:00:47	OK
41012111	นาย	วราวุธ	งามยิ่ง	นักศึกษา	เครื่องกล	100	PLC	9/4/2544	8:14:25	OK
41012100	นาย	วราภกร	พงษ์พันธ์	เจ้าหน้าที่	เครื่องกล	100	PLC	9/4/2544	9:20:41	OK
41012093	นาย	ไพรัตน์	จันทร์หอม	นักศึกษา	วัดชุม	100	PLC	9/4/2544	9:28:39	OK
41012095	นาย	มาหิด	ปลอตธา	นักศึกษา	วัดชุม	100	PLC	9/4/2544	9:29:27	OK
41012102	นาย	สมเกียรติ	เกิดเจริญ	นักศึกษา	วัดชุม	100	PLC	9/4/2544	9:43:30	OK
41013402	นาย	โกวิท	ลฤมจิตต์	นักศึกษา	วัดชุม	100	PLC	9/4/2544	12:57:41	OK
41013402	นาย	โกวิท	ลฤมจิตต์	นักศึกษา	วัดชุม	100	PLC	9/4/2544	12:58:30	OK
41012102	นาย	สมเกียรติ	เกิดเจริญ	นักศึกษา	วัดชุม	100	PLC	18/4/2544	15:18:07	OK
41012102	นาย	สมเกียรติ	เกิดเจริญ	นักศึกษา	วัดชุม	100	PLC	6/4/2544	16:36:38	OK
41012102	นาย	สมเกียรติ	เกิดเจริญ	นักศึกษา	วัดชุม	100	PLC	6/4/2544	16:43:07	OK
41012102	นาย	สมเกียรติ	เกิดเจริญ	นักศึกษา	วัดชุม	100	PLC	6/4/2544	16:45:45	OK
41012102	นาย	สมเกียรติ	เกิดเจริญ	นักศึกษา	วัดชุม	100	PLC	6/4/2544	20:50:42	OK

โปรแกรมตรวจสอบการเข้าออกด้วยลายนิ้วมือ

26/4/44

รูปที่ 5.20 แสดงฟอร์มสำหรับแสดงรายงานข้อมูลแต่ละห้องที่มีผู้ใช้เข้าใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

สรุปผลการวิจัยและเสนอแนะ

บทสรุป

ระบบการตรวจสอบการเข้า-ออกด้วยลายนิ้วมือ จะถูกออกแบบให้มีการสื่อสารข้อมูล โดยใช้ระบบเน็ตเวิร์กเพื่อให้บริการข้อมูล เก็บข้อมูลโดยคอมพิวเตอร์ที่ทำหน้าที่เป็น เซิร์ฟเวอร์ ซึ่งจะทำการจัดเก็บข้อมูลประวัติ, ข้อมูลลายนิ้วมือของผู้ใช้ ทำการกำหนดห้องให้กับผู้ใช้แต่ละคน ส่วนเครื่อง ไคลเอนต์ จะทำหน้าที่รับข้อมูลรหัสประจำตัว ข้อมูลลายนิ้วมือจากผู้ใช้เพื่อทำการอ่าน เปรียบเทียบข้อมูลลายนิ้วมือที่เก็บไว้ที่เครื่อง เซิร์ฟเวอร์และทำการตรวจสอบห้อง ถ้าข้อมูลลายนิ้วมือตรงกับของผู้ใช้และผู้ใช้สามารถเข้าห้องนี้ได้ก็จะอนุญาตให้สามารถเข้าใช้ห้องได้ และที่เครื่องเซิร์ฟเวอร์ ก็จะทำการบันทึกข้อมูลต่างๆ ของผู้ใช้รวมทั้งเวลาที่เข้าใช้ห้อง ซึ่งสามารถแสดงเป็นรายงานการเข้าใช้ห้องได้

ข้อเสนอแนะ

จากโครงการเรื่องนี้ สามารถที่จะนำไปเป็นแนวทางในการพัฒนาหรือประยุกต์เพื่อใช้เป็นระบบรักษาความปลอดภัยของข้อมูล กับระบบเครือข่ายขนาดใหญ่ เช่น เครือข่าย อินเทอร์เน็ต หรืออาจจะประยุกต์เพื่อใช้ควบคุมอุปกรณ์ต่างๆ ผ่านทางระบบเครือข่าย LAN หรือ เครือข่ายอินเทอร์เน็ตเช่น การเปิด-ปิดประตู เป็นต้น

ภาคผนวก

แสดงสอร์ซโค้ดของโปรแกรม

1. แสดงสอร์ซโค้ดโปรแกรมการจัดการเก็บข้อมูลประวัติผู้ใช้ที่เครื่องเซิร์ฟเวอร์

Private NewRecord As Boolean

Private Update As Boolean

Private MemberCn As ADODB.Connection

Private MemberRs As ADODB.Recordset

Private PictureCn As ADODB.Connection

Private PictureRs As ADODB.Recordset

Private pic As String

Private ds As Long

Dim Picturepath As String

Dim unloadpicture As Boolean

Dim n, nx, DS(100, 1)

Dim n1, n1x, D1S(100, 1)

Dim n2, n2x, D2S(100, 1)

Private Sub btnAdd_Click()

On Error GoTo handleError

Call Enablemove(False)

Call EnableOperationButton(False)

Call EnableSaveCancelButton(True)

Call ClearData

Update = False

Frame1.Enabled = True

Exit Sub

handleError:

MsgBox Error(Err.Number)

End Sub

Private Sub btnCancel_Click()

On Error GoTo handleError

Call ClearData

Call DisplayFields

Call Enablemove(True)

Call EnableSaveCancelButton(False)

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Call EnableOperationButton(True)
Frame1.Enabled = False
Exit Sub
handleError:
    MsgBox Error(Err.Number)
End Sub

```

```

Private Sub btnDelete_Click()
On Error GoTo handleError
If MsgBox("ต้องการลบข้อมูลเรคคอร์ดนี้,แน่ใจหรือไม่?", [vbYesNo]) = vbYes Then
MemberRs.Delete
PictureRs.Delete
Call btnMoveFirst_Click
End If
Exit Sub
handleError:
    MsgBox Error(Err.Number)
End Sub

```

```

Private Sub btnEdit_Click()
On Error GoTo handleError
Update = True
Call Enablemove(False)
Call EnableOperationButton(False)
Call EnableSaveCancelButton(True)
Frame1.Enabled = True
Exit Sub
handleError:
    MsgBox Error(Err.Number)
End Sub

```

```

Private Sub btnExit_Click()
MemberRs.Close
MemberCn.Close
PictureRs.Close
PictureCn.Close
MDIForm1.Enabled = True
Unload Me
End Sub

```

```
Private Sub btnMoveFirst_Click()
```

```
On Error GoTo handleError
```

```
MemberRs.MoveFirst
```

```
Call DisplayFields
```

```
btnMoveLast.Enabled = True
```

```
btnMoveNext.Enabled = True
```

```
btnMoveFirst.Enabled = False
```

```
btnMovePrevious.Enabled = False
```

```
Exit Sub
```

```
handleError:
```

```
MsgBox Error(Err.Number)
```

```
End Sub
```

```
Private Sub btnMoveLast_Click()
```

```
On Error GoTo handleError
```

```
MemberRs.MoveLast
```

```
Call DisplayFields
```

```
btnMoveLast.Enabled = False
```

```
btnMoveNext.Enabled = False
```

```
btnMoveFirst.Enabled = True
```

```
btnMovePrevious.Enabled = True
```

```
Exit Sub
```

```
handleError:
```

```
MsgBox Error(Err.Number)
```

```
End Sub
```

```
Private Sub btnMoveNext_Click()
```

```
On Error GoTo handleError
```

```
MemberRs.Movenext
```

```
If MemberRs.EOF = False Then
```

```
Call DisplayFields
```

```
btnMoveFirst.Enabled = True
```

```
btnMovePrevious.Enabled = True
```

```
Else
```

```
btnMoveFirst.Enabled = True
```

```
btnMovePrevious.Enabled = True
```

```
btnMoveLast.Enabled = False
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

btnMoveNext.Enabled = False
Call ClearData
End If
Exit Sub
handleError:
    MsgBox Error(Err.Number)
End Sub

```

```
Private Sub btnMovePrevious_Click()
```

```
On Error GoTo handleError
```

```
MemberRs.MovePrevious
```

```
If MemberRs.BOF = False Then
```

```
Call DisplayFields
```

```
btnMoveLast.Enabled = True
```

```
btnMoveNext.Enabled = True
```

```
Else
```

```
btnMoveFirst.Enabled = False
```

```
btnMovePrevious.Enabled = False
```

```
End If
```

```
Exit Sub
```

```
handleError:
```

```
    MsgBox Error(Err.Number)
```

```
End Sub
```

```
Private Sub btnSave_Click()
```

```
On Error GoTo handleError
```

```
Call chakEOF
```

```
If Update = True Then
```

```
DoUpdate
```

```
Exit Sub
```

```
End If
```

```
If textID_Code.Text = "" Then
```

```
MsgBox "กรุณาป้อนรหัส"
```

```
ElseIf cmbPrenome.Text = "" Then
```

```
MsgBox "กรุณาใส่ค่านำหน้าชื่อ"
```

```
ElseIf txtName.Text = "" Then
```

```
MsgBox "กรุณาใส่ชื่อ"
```

```
ElseIf txtSurName.Text = "" Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MsgBox "กรุณาใส่นามสกุล"
ElseIf txtAge.Text = "" Then
MsgBox "กรุณาใส่อายุ"
ElseIf cmbSex.Text = "" Then
MsgBox "กรุณากำหนดเพศ"
ElseIf cmbLevel.Text = "" Then
MsgBox "กรุณาใส่สถานะภาพผู้ใช้"
ElseIf cmbMajor.Text = "" Then
MsgBox "กรุณาใส่ข้อมูลภาควิชา"
Else
Dosearch1
If ds = 1 Then
DoUpdate
ElseIf ds = 3 Then
MemberRs.AddNew
PictureRs.AddNew
PictureRs.Fields("ID_Code") = txtID_Code.Text
MemberRs.Fields("ID_Code") = txtID_Code.Text
MemberRs.Fields("Prenome") = cmbPrenome.Text
MemberRs.Fields("Name").Value = txtName.Text
MemberRs.Fields("Surname").Value = txtSurName.Text
MemberRs.Fields("Age").Value = txtAge.Text
MemberRs.Fields("Sex").Value = cmbSex.Text
MemberRs.Fields("Day").Value = cmbDate.Text
MemberRs.Fields("Level").Value = cmbLevel.Text
MemberRs.Fields("Major_No").Value = cmbMajor.Text
If Text1.Text = "" Then
MsgBox " เพื่อความสมบูรณ์ของข้อมูลท่านควรจะมีรูปภาพด้วยครับ", , "เตือนก่อนการบันทึก"
PictureRs.Fields("std_Picture") = "e:\PictureProject\Failed picture.jpg"
End If
If Text1.Text <> "" Then
PictureRs.Fields("std_Picture") = Text1.Text
End If
MemberRs.Update
PictureRs.Update
MemberRs.Requery

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

PictureRs.Requery
Frame1.Enabled = False
Call Enablemove(True)
Call EnableSaveCancelButton(False)
Call EnableOperationButton(True)
MsgBox "บันทึกข้อมูลเรียบร้อยแล้ว"
ElseIf ds = 2 Then
Frame1.Enabled = False
Call Enablemove(True)
Call EnableSaveCancelButton(False)
Call EnableOperationButton(True)
End If
End If
Exit Sub
handleError:
MsgBox Error(Err.Number)
End Sub
Private Sub btnSearch_Click()
Frame1.Enabled = True
dloSeach.Show
End Sub
Private Sub cmdPicture_Click()
frmPicture.Show
End Sub
Private Sub Form_Load()
Dim Cnnstring As String
Dim picstring As String
On Error GoTo handleError:
Set MemberCn = New ADODB.Connection
Cnnstring = "Provider=Microsoft.Jet.OLEDB.4.0 ;" & _
" Data Source = e:\data\Instrument_Data.mdb;"
Set PictureCn = New ADODB.Connection
picstring = "Provider=Microsoft.Jet.OLEDB.4.0 ;" & _
" Data Source = e:\data\Instr;"
MemberCn.CommandTimeout = 25
PictureCn.CommandTimeout = 25

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MemberCn.Open Cnnstring
PictureCn.Open picstring
Set MemberRs = New ADODB.Recordset
MemberCn.CursorLocation = adUseClient
MemberRs.Open "TblMember", MemberCn, adOpenStatic, adLockPessimistic, adCmdTable
Set PictureRs = New ADODB.Recordset
PictureRs.Open "TblPicture", PictureCn, adOpenDynamic, adLockPessimistic, adCmdTable

Call Prenom_no
Call Sex_no
Call Date_no
Call ReadMajor_no
Call ReadLevel_no
Call EnableSaveCancelButton(False)
MemberRs.MoveFirst
PictureRs.MoveFirst
Call DisplayFields
btnMoveFirst.Enabled = False
btnMovePrevious.Enabled = False
Exit Sub
handleError:
MsgBox Error(Err.Number)
End Sub

Private Sub DisplayFields()
On Error GoTo handleError
textID_Code.Text = MemberRs.Fields("ID_Code").Value
cmbPrenom.Text = MemberRs.Fields("Prenom").Value
txtName.Text = MemberRs.Fields("Name").Value
txtSurName.Text = MemberRs.Fields("Surname").Value
txtAge.Text = MemberRs.Fields("Age").Value
cmbSex.Text = MemberRs.Fields("Sex").Value
cmbDate.Text = MemberRs.Fields("Day").Value
cmbLevel.Text = MemberRs.Fields("Level").Value
cmbMajor.Text = MemberRs.Fields("Major_No").Value
Call DoSearchPicture
If unloadpicture = False Then
pic = Picturepath

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

DoPicture
End If
Exit Sub
handleError:
    MsgBox Error(Err.Number)
End Sub

```

```
Private Sub ReadLevel_no()
```

```
On Error GoTo Lm
```

```
Open "e:\data\Leveldata.txt" For Input As #1
```

```
nx = 0
```

```
Do While Not EOF(1)
```

```
Input #1, f1$: nx = nx + 1
```

```
DS(nx, 1) = f1$
```

```
cmbLevel.AddItem DS(nx, 1)
```

```
Loop
```

```
Close
```

```
Exit Sub
```

```
Lm:
```

```
MsgBox "ไม่มีข้อมูลระดับผู้ใช้งานต้องจัดเก็บข้อมูลใหม่"
```

```
Close
```

```
End Sub
```

```
Private Sub EnableOperationButton(EnableFlag As Boolean)
```

```
btnAdd.Visible = EnableFlag
```

```
btnEdit.Visible = EnableFlag
```

```
btnDelete.Visible = EnableFlag
```

```
btnSearch.Visible = EnableFlag
```

```
btnExit.Visible = EnableFlag
```

```
End Sub
```

```
Private Sub EnableSaveCancelButton(EnableFlag As Boolean)
```

```
btnSave.Visible = EnableFlag
```

```
btnCancel.Visible = EnableFlag
```

```
cmdPicture.Enabled = EnableFlag
```

```
End Sub
```

```
Private Sub ClearData()
```

```
On Error GoTo handleError
```

```
textID_Code.Text = ""
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

cmbPrename.Text = ""
txtName.Text = ""
txtSurName.Text = ""
txtAge.Text = ""
cmbSex.Text = ""
cmbDate.Text = ""
cmbLevel.Text = ""
cmbMajor.Text = ""
imgMember.Picture = LoadPicture()
Exit Sub

handleError:
MsgBox Error(Err.Number)
End Sub

```

```

Sub DoPicture()
On Error GoTo handleError
imgMember.Visible = False
imgMember.Stretch = False
imgMember.Picture = LoadPicture(pic)
DoPic 2055, 1935, imgMember.Width, imgMember.Height, w, h
imgMember.Width = w
imgMember.Height = h
imgMember.Visible = True
imgMember.Stretch = True
Exit Sub

handleError:
imgMember.Visible = False
imgMember.Stretch = False
imgMember.Picture = LoadPicture("e:\PictureProject\Failed picture.jpg")
DoPic 2055, 1935, imgMember.Width, imgMember.Height, w, h
imgMember.Width = w
imgMember.Height = h
imgMember.Visible = True
imgMember.Stretch = True
End Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Sub Dosearch1()
Dim oldPosition As Variant
Dim search As String
Dim criteria As String
Dim name As String
Dim lastname As String
On Error GoTo handleError

search = textID_Code.Text

With MemberRs
    oldPosition = .Bookmark
    .MoveFirst
    criteria = "ID_Code LIKE '" & search & "'"
    .Find criteria, , adSearchForward
    If Not MemberRs.EOF Then
        name = MemberRs.Fields(2).Value
        lastname = MemberRs.Fields(3).Value
        If MsgBox("พบข้อมูลเดิมต้องการแก้ไขข้อมูลของคุณ " & name & " " & lastname & " หรือ
ไม่", vbYesNo, "ยืนยันการบันทึก") = vbYes Then
            ds = 1
            .Bookmark = oldPosition
        Else
            ds = 2
        End If
    Else
        ds = 3
    End If
End With

Exit Sub

handleError:
MsgBox Error(Err.Number)

End Sub

```

```

Sub DoUpdate()
On Error GoTo handleError

PictureRs.Fields("ID_Code") = textID_Code.Text

MemberRs.Fields("ID_Code") = textID_Code.Text

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MemberRs.Fields("Prenome") = cmbPrenome.Text
MemberRs.Fields("Name").Value = txtName.Text
MemberRs.Fields("Surname").Value = txtSurName.Text
MemberRs.Fields("Age").Value = txtAge.Text
MemberRs.Fields("Sex").Value = cmbSex.Text
MemberRs.Fields("Day").Value = cmbDate.Text
MemberRs.Fields("Level").Value = cmbLevel.Text
MemberRs.Fields("Major_No").Value = cmbMajor.Text
If Text1.Text <> "" Then
PictureRs.Fields("std_Picture") = Text1.Text
End If
MemberRs.Update
PictureRs.Update
MsgBox "แก้ไขข้อมูลเรียบร้อยแล้ว"
Call Enablemove(True)
Call EnableSaveCancelButton(False)
Call EnableOperationButton(True)
Exit Sub
handleError:
MsgBox "บันทึกการแก้ไขข้อมูลผิดพลาด"
End Sub

```

```

Private Sub ReadMajor_no()
On Error GoTo Ln
Open "e:\data\Sectiondata.txt" For Input As #1
nx = 0
Do While Not EOF(1)
Input #1, fl$: nx = nx + 1
DIS(nx, 1) = fl$
cmbMajor.AddItem DIS(nx, 1)
Loop
Close
Exit Sub
Ln:
MsgBox "ไม่มีข้อมูลภาควิชาท่านต้องจัดเก็บข้อมูลใหม่"
Close

```

```
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Private Sub Sex_no()
    cmbSex.AddItem "ชาย"
    cmbSex.AddItem "หญิง"
End Sub

Private Sub Date_no()
    dtmD = Now
    cmbDate.AddItem Format(dtmD, "d/m/yyyy")
End Sub

```

```

Private Sub Preamble_no()
On Error GoTo Ln
    Open "e:\data\Prenamedata.txt" For Input As #1
    nx = 0
    Do While Not EOF(1)
        Input #1, f1$: nx = nx + 1
        D2$(nx, 1) = f1$
        cmbPreamble.AddItem D2$(nx, 1)
    Loop
    Close
Exit Sub
Ln:
    MsgBox "ไม่มีข้อมูลค่านำหน้าชื่อท่านต้องจัดเก็บข้อมูลใหม่"
Close
End Sub

```

```

Sub DoSearch()
    Dim oldPosition As Variant
    Dim search As String
    Dim criteria As String

    search = dloSeach.txtID_Code.Text

    With MemberRs
        oldPosition = .Bookmark
        .MoveFirst
        criteria = "ID_Code LIKE " & search & ""
        .Find criteria, , adSearchForward

        If Not MemberRs.EOF Then
            Call DisplayFields

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Else
    MsgBox "ไม่พบข้อมูล"
    .Bookmark = oldPosition
    Frame1.Enabled = False
End If

End With

End Sub

```

```

Private Sub chakEOF()

```

```

    If MemberRs.EOF = True Then

```

```

        MemberRs.MovePrevious

```

```

    End If

```

```

    If PictureRs.EOF = True Then

```

```

        PictureRs.MovePrevious

```

```

    End If

```

```

End Sub

```

```

Private Sub Enablemove(EnableFlag As Boolean)

```

```

    btnMoveFirst.Enabled = EnableFlag

```

```

    btnMoveLast.Enabled = EnableFlag

```

```

    btnMoveNext.Enabled = EnableFlag

```

```

    btnMovePrevious.Enabled = EnableFlag

```

```

End Sub

```

```

Private Sub DoSearchPicture()

```

```

    Dim search As String

```

```

    Dim criteria As String

```

```

    On Error GoTo handleError

```

```

    search = textID_Code.Text

```

```

    With PictureRs

```

```

        .MoveFirst

```

```

        criteria = "ID_Code LIKE '" & search & "'"

```

```

        .Find criteria, , adSearchForward

```

```

        If Not PictureRs.EOF Then

```

```

            Picturepath = PictureRs.Fields("Std_Picture")

```

```

            unloadpicture = False

```

```

        End If

```

```

    End With

```

```

Exit Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

handleError:
    unloadpicture = True
    imgMember.Picture = LoadPicture()
    MsgBox "ไม่มีข้อมูลรูปภาพในฐานข้อมูลเพื่อความสมบูรณ์ควรจะจัดเก็บไว้"

```

End Sub

```
Private Sub Form_Unload(Cancel As Integer)
```

```
    MDIForm1.Enabled = True
```

```
    Unload Me
```

End Sub

```
Private Sub textID_Code_KeyUp(KeyCode As Integer, Shift As Integer)
```

```
    If KeyCode = 13 Then
```

```
        If Not IsNumeric(textID_Code.Text) Then
```

```
            MsgBox "กรุณาใส่ค่าตัวเลขเท่านั้น"
```

```
        End If
```

```
    End If
```

End Sub

2. แสดงผลลัพธ์โค้ดโปรแกรมการจัดการเก็บข้อมูลลายนิ้วมือ

```
Private PictureCn As ADODB.Connection
```

```
Private PictureRs As ADODB.Recordset
```

```
Private MemberCn As ADODB.Connection
```

```
Private MemberRs As ADODB.Recordset
```

```
Private chake As Boolean
```

```
Dim Pathpicture As String
```

```
Dim Pathfinger As String
```

```
Dim ID_code As String
```

```
Private Sub cmdCompareL_Click()
```

```
    finger1.currentName = ID_code
```

```
    If finger1.compareL = True Then
```

```
        MsgBox "ลายนิ้วมือถูกต้อง"
```

```
    Else
```

```
        MsgBox "ลายนิ้วมือไม่ถูกต้อง"
```

```
    End If
```

End Sub

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Private Sub cmdCompareR_Click()
    finger2.currentName = ID_code
    If finger2.compareR = True Then
        MsgBox "ลายนิ้วมือถูกต้อง"
    Else
        MsgBox "ลายนิ้วมือ ไม่ถูกต้อง"
    End If
End Sub

```

```

Private Sub cmdSaveL_Click()
    finger1.currentName = ID_code
    finger1.saveL
    finger1.DisplayImgL
End Sub

```

```

Private Sub cmdSaveR_Click()
    finger2.currentName = ID_code
    finger2.saveR
    finger2.DisplayImgR
End Sub

```

```

Private Sub Command1_Click()
    dloFingerSeach.Show
    frmFingerdata.Enabled = False
End Sub

```

```

Private Sub Form_Load()
    Dim Cnnpic As String
    Dim Cnnmem As String
    On Error GoTo handleError
        Set PictureCn = New ADODB.Connection
        Cnnpic = "Provider=Microsoft.Jet.OLEDB.4.0 ;" & _
            " Data Source = e:\data\Instrument_Data.mdb;"
        Set MemberCn = New ADODB.Connection
        Cnnmem = "Provider=Microsoft.Jet.OLEDB.4.0 ;" & _
            " Data Source = e:\data\Instrument_Data.mdb;"
        PictureCn.CommandTimeout = 25
        MemberCn.CommandTimeout = 25
        PictureCn.Open Cnnpic

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MemberCn.Open Cnnpic
Set PictureRs = New ADODB.Recordset
PictureCn.CursorLocation = adUseClient
PictureRs.Open "TblPicture", PictureCn, adOpenStatic, adLockPessimistic, adCmdTable
Set MemberRs = New ADODB.Recordset
MemberCn.CursorLocation = adUseClient
MemberRs.Open "TblMember", MemberCn, adOpenStatic, adLockPessimistic, adCmdTable
PictureRs.MoveFirst
MemberRs.MoveFirst
Call EnableOperationButton(False)
Exit Sub

```

handleError:

```
MsgBox Error(Err.Number)
```

End Sub

```
Sub DisplaySeachMember_no()
```

```
Dim search As String
```

```
Dim criteria As String
```

```
Dim Prenom As String
```

```
Dim name As String
```

```
Dim lastname As String
```

```
Dim ID As String
```

```
On Error GoTo handleError
```

```
frmFingerdata.Enabled = True
```

```
ID_code = dloFingerSeach.txtID_Code.Text
```

```
search = ID_code
```

```
With MemberRs
```

```
.MoveFirst
```

```
criteria = "ID_Code LIKE " & search & ""
```

```
.Find criteria, , adSearchForward
```

```
If Not MemberRs.EOF Then
```

```
ID = MemberRs.Fields("ID_Code").Value
```

```
Label1.Caption = " รหัสประจำตัว " & ID & ""
```

```
Prenom = MemberRs.Fields("Prenom").Value
```

```
name = MemberRs.Fields("Name").Value
```

```
lastname = MemberRs.Fields("Surname").Value
```

```
Label2.Caption = "" & Prenom & " " & name & " " & lastname & ""
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Call DisplayFinger
Call EnableOperationButton(True)
Call DisplaySeachPicture_no
chake = True
Else
MsgBox "ไม่มีประวัติในฐานข้อมูลกรุณาเก็บประวัติก่อน"
End If

```

```
End With
```

```
Exit Sub
```

```
handleError:
```

```
MsgBox Error(Err.Number)
```

```
End Sub
```

```
Private Sub DisplaySeachPicture_no()
```

```
On Error GoTo handleError
```

```
Dim search1 As String
```

```
Dim criteria As String
```

```
search1 = ID_code
```

```
With PictureRs
```

```
.MoveFirst
```

```
criteria = "ID_Code LIKE " & search1 & ""
```

```
.Find criteria, , adSearchForward
```

```
If Not PictureRs.EOF Then
```

```
Pathpicture = PictureRs.Fields("std_picture").Value
```

```
Call Displaypicture_no
```

```
Else
```

```
MsgBox "ไม่พบข้อมูล"
```

```
End If
```

```
End With
```

```
Exit Sub
```

```
handleError:
```

```
MsgBox Error(Err.Number)
```

```
End Sub
```

```
Private Sub Displaypicture_no()
```

```
On Error GoTo handleError
```

```
imgPicture.Visible = False
```

```
imgPicture.Stretch = False
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีก้ารนำไปใช้

```

imgPicture.Picture = LoadPicture(Pathpicture)
DoPic 2055, 1935, imgPicture.Width, imgPicture.Height, w, h
imgPicture.Width = w
imgPicture.Height = h
imgPicture.Visible = True
imgPicture.Stretch = True
Exit Sub

```

handleError:

```

imgPicture.Visible = False
imgPicture.Stretch = False
imgPicture.Picture = LoadPicture("e:\PictureProject\Failed picture.jpg")
DoPic 2055, 1935, imgPicture.Width, imgPicture.Height, w, h
imgPicture.Width = w
imgPicture.Height = h
imgPicture.Visible = True
imgPicture.Stretch = True

```

End Sub

Private Sub EnableOperationButton(EnableFlag As Boolean)

```

cmdSaveL.Enabled = EnableFlag
cmdSaveR.Enabled = EnableFlag
cmdCompareL.Enabled = EnableFlag
cmdCompareR.Enabled = EnableFlag

```

End Sub

Private Sub DisplayFinger()

```

finger1.currentName = ID_code
finger2.currentName = ID_code
finger1.DisplayImgL
finger2.DisplayImgR

```

End Sub

Private Sub Form_Unload(Cancel As Integer)

```

PictureRs.Close
PictureCn.Close
MemberRs.Close
MemberCn.Close
MDIForm1.Enabled = True

```

Unload Me

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

End Sub

Private Sub mnuExit_Click()

 Unload Me

End Sub

Private Sub txtID_Code_KeyUp(KeyCode As Integer, Shift As Integer)

 cmdSeve.Enabled = True

 Call EnableOperationButton(False)

End Sub

3. แสดงฟอร์มโค้ดโปรแกรมตรวจสอบการร้องขอข้อมูลจากเครื่องควบคุม

Private RoomfixCn As ADODB.Connection

Private RoomfixRs As ADODB.Recordset

Private CheckmemberCn As ADODB.Connection

Private CheckmemberRs As ADODB.Recordset

Dim PathID_Code As String

Dim IP_No As String

Dim IP_com As String

Dim m As Long

Dim n As Long

Dim RoomID As String

Dim isolateID As String

Dim D2\$(10000)

Dim D1\$(10000)

Dim cs As Long

Dim ds As Long

Dim qs As Long

Dim bs As Long

Dim checkL As Boolean

Dim M1\$(10000)

Dim M2\$(10000)

Dim Cr As Long

Dim Crr As Long

Dim Rr As Long

Dim Rrr As Long

เอกสาร Dim checkR As Boolean ใช้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Dim check As Boolean

Dim finger_Check As Boolean

Private Sub bStart_Click()

IPDaemon1.LocalPort = tPort

IPDaemon1.Listening = True

bStart.Enabled = False

frmserverShow.Show

Me.Hide

End Sub

Private Sub Form_Load()

Dim Cnnfix As String

Dim Cnncheck As String

On Error GoTo handleError

Set RoomfixCn = New ADODB.Connection

Cnnfix = "Provider=Microsoft.Jet.OLEDB.4.0 ;" & _
 " Data Source = e:\data\Instrument_Data.mdb;"

Set CheckmemberCn = New ADODB.Connection

Cnncheck = "Provider=Microsoft.Jet.OLEDB.4.0 ;" & _
 " Data Source = e:\data\Instrument_Data.mdb;"

CheckmemberCn.ConnectionTimeout = 25

RoomfixCn.CommandTimeout = 25

RoomfixCn.Open Cnnfix

CheckmemberCn.Open Cnncheck

Set CheckmemberRs = New ADODB.Recordset

CheckmemberCn.CursorLocation = adUseClient

CheckmemberRs.Open "Tblcheck", CheckmemberCn, adOpenStatic, adLockPessimistic, adCmdTable

Set RoomfixRs = New ADODB.Recordset

RoomfixCn.CursorLocation = adUseClient

RoomfixRs.Open "TblRoomfix", RoomfixCn, adOpenStatic, adLockPessimistic, adCmdTable

CheckmemberRs.MoveFirst

RoomfixRs.MoveFirst

Exit Sub

handleError:

MsgBox Error(Err.Number)

End Sub

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub IPDaemon1_DataIn(ConnectionId As Integer, Text As String, EOL As Boolean)
```

```
    If Text = "11111" Then
```

```
        Call SendL
```

```
    Exit Sub
```

```
    End If
```

```
    If Text = "22222" Then
```

```
        Call SendLL
```

```
    Exit Sub
```

```
    End If
```

```
    If Text = "33333" Then
```

```
        Call sendR
```

```
    Exit Sub
```

```
    End If
```

```
    If Text = "44444" Then
```

```
        Call sendRR
```

```
    Exit Sub
```

```
    End If
```

```
    IP_No = IPDaemon1.RemoteHost(ConnectionId)
```

```
    Call SplitIP
```

```
    If Text = "L" Then
```

```
        qs = 0
```

```
        bs = 0
```

```
        check = True
```

```
        Call leadL
```

```
    Exit Sub
```

```
    End If
```

```
    If Text = "R" Then
```

```
        Crr = 0
```

```
        Rrr = 0
```

```
        check = False
```

```
        Call leadR
```

```
    Exit Sub
```

```
    End If
```

```
    If Text = "OK_Open" Then บันทึกความเรียบร้อย
```

```
    Call Add_Check
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Exit Sub
End If
'and echo it back
IPDaemon1.DataToSend(ConnectionId) = Text
PathID_Code = Text
Call update_no
Call DoSearchID_Code
End Sub

```

```

Private Sub update_no()
Dim Cnnfix As String
Set RoomfixCn = New ADODB.Connection
    Cnnfix = "Provider=Microsoft.Jet.OLEDB.4.0;" & _
        " Data Source = e:\data\Instrument_Data.mdb;"
RoomfixCn.CommandTimeout = 25
RoomfixCn.Open Cnnfix
Set RoomfixRs = New ADODB.Recordset
RoomfixCn.CursorLocation = adUseClient
RoomfixRs.Open "TblRoomfix", RoomfixCn, adOpenStatic, adLockPessimistic, adCmdTable
RoomfixRs.MoveFirst
End Sub

```

```

Private Sub SendL()
If cs < qs Then GoTo 1
IPDaemon1.DataToSend(1) = D2S(qs)
checkL = True
qs = qs + 1
Exit Sub
1:
IPDaemon1.DataToSend(1) = "Stop"
End Sub

```

```

Private Sub leadL()
m_pathname = "h:\\"
m_currentName = PathID_Code
Debug.Print compareL
IPDaemon1.DataToSend(1) = "ReadyL"
End Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub SendLL()
```

```
    If ds < bs Then GoTo 1
```

```
        IPDaemon1.DataToSend(1) = D1$(bs)
```

```
        bs = bs + 1
```

```
    Exit Sub
```

```
1:
```

```
    IPDaemon1.DataToSend(1) = "alreadyL"
```

```
End Sub
```

```
Private Sub sendR()
```

```
    If Cr < Crr Then GoTo 1
```

```
    IPDaemon1.DataToSend(1) = M1$(Crr)
```

```
    Crr = Crr + 1
```

```
    Exit Sub
```

```
1:
```

```
    IPDaemon1.DataToSend(1) = "Stop1"
```

```
End Sub
```

```
Private Sub sendRR()
```

```
    If Rr < Rrr Then GoTo 1
```

```
    IPDaemon1.DataToSend(1) = M2$(Rrr)
```

```
    Rrr = Rrr + 1
```

```
    Exit Sub
```

```
1:
```

```
    IPDaemon1.DataToSend(1) = "alreadyR"
```

```
End Sub
```

```
Sub ShutDown()
```

```
On Error GoTo handleError
```

```
    IPDaemon1.DataToSend(1) = "ShutDown"
```

```
    Exit Sub
```

```
handleError:
```

```
    MsgBox Error(Err.Number)
```

```
End Sub
```

```
Private Sub leadR()
```

```
    m_pathname = "h:\\"
```

```
    m_currentName = PathID_Code
```

```
    compareR
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
IPDaemon1.DataToSend(1) = "ReadyR"
```

```
End Sub
```

```
Private Sub DoSearchID_Code()
```

```
Dim search As String
```

```
Dim criteria As String
```

```
On Error GoTo handleError
```

```
search = PathID_Code
```

```
With RoomfixRs
```

```
.MoveFirst
```

```
criteria = "ID_Code LIKE " & search & ""
```

```
.Find criteria, , adSearchForward
```

```
If Not RoomfixRs.EOF Then
```

```
RoomID = RoomfixRs.Fields("ID_Room").Value
```

```
Call isolateID_Room
```

```
Else
```

```
IPDaemon1.DataToSend(1) = "ErrorID_Code"
```

```
End If
```

```
End With
```

```
Exit Sub
```

```
handleError:
```

```
MsgBox Error(Err.Number)
```

```
End Sub
```

```
Private Sub Add_Check()
```

```
CheckmemberRs.AddNew
```

```
CheckmemberRs.Fields("ID_Code") = PathID_Code
```

```
CheckmemberRs.Fields("Date") = Format(Now, "Long Date")
```

```
CheckmemberRs.Fields("Time") = Format(Now, "tttt")
```

```
CheckmemberRs.Fields("ID_Room") = IP_com
```

```
CheckmemberRs.Fields("Even") = "OK"
```

```
CheckmemberRs.Update
```

```
CheckmemberRs.Requery
```

```
RoomfixRs.Close
```

```
RoomfixCn.Close
```

```
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Private Sub SplitIP()

IP_com = Mid\$(IP_No, 12, 3)

End Sub

Private Sub isolateID_Room()

isolateID = "Show me the ID_Room "

1:

m = m + 1

If isolateID <> "" Then

isolateID = Mid\$(RoomID, 1 + n, 3)

If IP_com = isolateID Then

IPDaemon1.DataToSend(1) = "ID_OK"

Exit Sub

End If

n = n + 3

GoTo 1

Else

IPDaemon1.DataToSend(1) = "No_ID"

End If

End Sub

Public Property Get pathname() As Variant

pathname = Text1.Text

End Property

Public Function compareL() As Boolean

Dim currenfile As String: Dim a, l2

l2 = DBI_InitDriver(l, D)

Call DBI_DLLInit(l, D)

Dim SerialNumber(1) As Long

Dim DriverVersion(1) As Long

Dim RecognitionVersion(1) As Long

Dim hhh As Date

l2 = DBI_GetInfo(DBI_BIOMOUSE_SERIAL, SerialNumber(0))

l2 = DBI_GetInfo(DBI_DRIVER_VERSION, DriverVersion(0))

l2 = DBI_GetInfo(DBI_RECOGNITION_VERSION, RecognitionVersion(0))

Dim DITptr(10) As Long: Dim Dits() As Byte: Dim Ddts() As Byte

ReDim Dits(DBI_sizeofDIT(), 10): ReDim Ddts(DBI_sizeofDDT(), 10)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

For I = 0 To 9
    DITptr(I) = DBI_GetByteAddress(Dits(0, I))
Next
Dim finger1(57600) As Byte, Buffer1(57600) As Byte, Buffer2(57600) As Byte
Dim intN As Long
intN = DBI_sizeofDIT()
cs = intN
Open m_pathname + m_currentName + ".DITL" For Binary As #1
    check = False
    For intN = 0 To intN
        Get #1, , Dits(intN, 4)
        D2$(intN) = Dits(intN, 4)
    Next intN
Close #1
intN = DBI_sizeofDDT()
ds = intN
Open m_pathname + m_currentName + ".DDTL" For Binary As #1
check = True
    For intN = 0 To intN
        Get #1, , Ddts(intN, 4)
        D1$(intN) = Ddts(intN, 4)
    Next intN
Close #1
End Function

```

```

Public Function compareR() As Boolean

```

```

    Dim currenfile As String: Dim a, l2
    l2 = DBI_InitDriver(l, D)
    Call DBI_DLLInit(l, D)
    Dim SerialNumber(1) As Long
    Dim DriverVersion(1) As Long
    Dim RecognitionVersion(1) As Long
    Dim hhh As Date
    l2 = DBI_GetInfo(DBI_BIOMOUSE_SERIAL, SerialNumber(0))
    l2 = DBI_GetInfo(DBI_DRIVER_VERSION, DriverVersion(0))
    l2 = DBI_GetInfo(DBI_RECOGNITION_VERSION, RecognitionVersion(0))

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Dim DITptr(10) As Long: Dim Dits() As Byte: Dim Ddts() As Byte
ReDim Dits(DBI_sizeofDIT(), 10): ReDim Ddts(DBI_sizeofDDT(), 10)
For I = 0 To 9
    DITptr(I) = DBI_GetByteAddress(Dits(0, I))
Next
Dim finger1(57600) As Byte, Buffer1(57600) As Byte, Buffer2(57600) As Byte
Dim intN As Long
intN = DBI_sizeofDIT()
Cr = intN
Open m_pathname + m_currentName + ".DITR" For Binary As #1
    For intN = 0 To intN
        Get #1, , Dits(intN, 5)
        M1$(intN) = Dits(intN, 5)
    Next intN
Close #1
intN = DBI_sizeofDDT()
Rr = intN
Open m_pathname + m_currentName + ".DDTR" For Binary As #1
    For intN = 0 To intN
        Get #1, , Ddts(intN, 5)
        M2$(intN) = Ddts(intN, 5)
    Next intN
Close #1
End Function

```

4. แสดงสตริงโค้ดโปรแกรมเครื่องควบคุมที่เชื่อมต่อและร้องขอข้อมูลไปยังเครื่องเซิร์ฟเวอร์

```

Dim ns As Long
Dim ms As Long
Dim finger$(10000, 1)
Dim fing$(10000, 1)
Dim FingerRS(10000, 1)
Dim FingerRR$(10000, 1)
Dim TextData As String
Dim CheckL As Boolean
Dim CheckR As Boolean

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Dim selection As Boolean

Dim UnloadShow As Boolean

Dim Moon As Boolean

Dim Cnn As Boolean

Dim Gnn As Boolean

Dim Bm As Long

Dim Bmm As Long

Dim DataL As String

Dim DataR As String

Dim ID_CodeMember As String

Dim gs As Long

Private Sub bConnect_Click()

 dlgConneCSever.Show

 frmLinkSever.Enabled = False

End Sub

Sub KeyPress()

 If gs <= 3 Then

 If Cnn = True Then

 Debug.Print compareL

 Exit Sub

 End If

 If Gnn = True Then

 compareR

 Exit Sub

 End If

 End If

End Sub

Private Sub Form_Load()

 Image1.Visible = False

 Image1.Stretch = False

 Image1.Picture = LoadPicture("D:\client\picture\logo.jpg")

 DoPic Me.Width - 100, Me.Height - 100, Image1.Width, Image1.Height, w, h

 Image1.Width = w

 Image1.Height = h

 Image1.Visible = True

 Image1.Stretch = True

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

End Sub


---


Sub DosendID_code()
    IPPort1.DataToSend = frmID_code.Label1.Caption
    ID_CodeMember = frmID_code.Label1.Caption
End Sub


---


Private Sub Form_Resize()
    Image1.Move 100, CoolBar1.Height + 50, Me.Width - 510, Me.Height - StatusBar1.Height - 1200
    CoolBar1.Move 0, 0, Me.Width
End Sub


---


Sub ConnectSever()
    IPPort1.RemoteHost = dlgConnecSever.txtServer
    IPPort1.RemotePort = dlgConnecSever.txtPort
    IPPort1.Connected = True
End Sub


---


Sub Splash()
    If UnloadShow = True Then
        frmSplash.Show
    End If
End Sub


---


Sub OK_Open()
    IPPort1.DataToSend = "OK_Open"
End Sub


---


Private Sub IPPort1_Connected(StatusCode As Integer, Description As String)
    If StatusCode = 0 Then
        UnloadShow = True
    Else
        UnloadShow = False
        frmConnectOK.Show
    End If
End Sub


---


Private Sub IPPort1_DataIn(Text As String, EOL As Boolean)
    If Text = "ShutDown" Then
        End
    Exit Sub
    End If
    If ID_CodeMember = Text Then

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Unload frmID_code
Exit Sub
End If
If Text = "No_ID" Then
frmShow1.Show
Exit Sub
End If
If Text = "ID_OK" Then
Seclecfinger.Show
Exit Sub
End If
If Text = "alreadyL" Then
Debug.Print compareL
Exit Sub
End If
If Text = "alreadyR" Then
compareR
Exit Sub
End If
If Text = "ErrorID_Code" Then
frmFailed.Show
Exit Sub
End If
If selection = True Then
If Text = "ReadyL" Then
IPPort1.DataToSend = "11111"
CheckL = True
Exit Sub
End If
If Text <> "Stop" Then
If CheckL = True Then
DataL = Text
Call moveL
Exit Sub
End If
End If

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

If Text = "Stop" Then
  IPPort1.DataToSend = "22222"
  CheckL = False
  Exit Sub
End If
If CheckL = False Then
  DataL = Text
  Call moveLL
  Exit Sub
End If
Exit Sub
End If
If selection = False Then
  If Text = "ReadyR" Then
    IPPort1.DataToSend = "33333"
    CheckR = True
    Exit Sub
  End If
  If Text <> "Stop1" Then
    If CheckR = True Then
      DataR = Text
      Call moveR
      Exit Sub
    End If
  End If
  If Text = "Stop1" Then
    CheckR = False
    IPPort1.DataToSend = "44444"
    Exit Sub
  End If
  If CheckR = False Then
    DataR = Text
    Call moveRR
    Exit Sub
  End If
Exit Sub

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

End If

End Sub

Sub compareLLL()

gs = 1

ns = 0

ms = 0

selection = True

IPPort1.DataToSend = "L"

End Sub

Sub compareRRR()

gs = 1

Bm = 0

Bmm = 0

selection = False

IPPort1.DataToSend = "R"

End Sub

Private Sub moveL()

finger\$(ns, 1) = DataL

ns = ns + 1

CheckL = True

selection = True

IPPort1.DataToSend = "11111"

End Sub

Private Sub moveLL()

fing\$(ms, 1) = DataL

ms = ms + 1

CheckL = False

selection = True

IPPort1.DataToSend = "22222"

End Sub

Private Sub moveR()

FingerR\$(Bm, 1) = DataR

Bm = Bm + 1

CheckR = True

selection = False

IPPort1.DataToSend = "33333"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

End Sub

Private Sub moveRR()

FingerRR\$(Bmm, 1) = DataR

Bmm = Bmm + 1

CheckR = False

selection = False

IPPort1.DataToSend = "44444"

End Sub

Private Sub IPPort1_Disconnected(StatusCode As Integer, Description As String)

IPPort1.Connected = False

End Sub

Private Sub IPPort1_Error(ErrorCode As Integer, Description As String)

Debug.Print "IPPort Error " & ErrorCode & ": " & Description

End Sub

Public Function compareL() As Boolean

Dim currenfile As String: Dim a, l2

l2 = DBI_InitDriver(l, D)

Call DBI_DLLInit(l, D)

Dim SerialNumber(1) As Long

Dim DriverVersion(1) As Long

Dim RecognitionVersion(1) As Long

Dim hhh As Date

l2 = DBI_GetInfo(DBI_BIOMOUSE_SERIAL, SerialNumber(0))

l2 = DBI_GetInfo(DBI_DRIVER_VERSION, DriverVersion(0))

l2 = DBI_GetInfo(DBI_RECOGNITION_VERSION, RecognitionVersion(0))

Dim DITptr(10) As Long: Dim Dits() As Byte: Dim Ddts() As Byte

ReDim Dits(DBI_sizeofDIT(), 10): ReDim Ddts(DBI_sizeofDDT(), 10)

For i = 0 To 9

DITptr(i) = DBI_GetByteAddress(Dits(0, i))

Next

Dim finger1(57600) As Byte, Buffer1(57600) As Byte, Buffer2(57600) As Byte

Dim intN As Long

For j = 0 To 656

Dits(j, 4) = finger\$(j, 1)

Next j

For j = 0 To 5664

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Ddts(j, 4) = fing$(j, 1)
Next j
l2 = DBI_UnCompress_DIT(Dits(0, 4), Ddts(0, 4))
currentfile = ID_CodeMember
Dim aa As Integer: Dim BB As Integer:
Dim UserName() As Byte
BB = Len(currentfile)
ReDim UserName(BB + 1)
For aa = 0 To BB - 1
UserName(aa) = Asc(Mid(currentfile, aa + 1, 1))
Next aa
UserName(aa + 1) = 0
MaxSecurity(0) = FA_1_IN_100000
Dim UserTitle(20) As Byte
UserTitle(0) = Asc("U"): UserTitle(1) = Asc("s"): UserTitle(2) = Asc("e"): UserTitle(3) = Asc("r")
UserTitle(4) = Asc("T"): UserTitle(5) = Asc("i"): UserTitle(6) = Asc("t"): UserTitle(7) = Asc("l")
UserTitle(8) = Asc("e"): UserTitle(9) = Asc("."): UserTitle(10) = 0
Dim AutoOt(1) As Long: Dim AutoSt(1) As Long: Dim AutoDt(1) As Long:
Dim InteractiveAutoDetect(1) As Boolean: Dim Security(1) As Long:
AutoOt(0) = 3000: AutoSt(0) = 3000: AutoDt(0) = 2: InteractiveAutoDetect(0) = 1: Security(0) =
FA_1_IN_100000: Indicator(0) = 0
l2 = DBI_AutoCapture_Dlg(finger1(0), UserName(0), 3000, 3000, 3, 1, 0)
b = DBI_sizeofDDT()
ReDim CurDDT(b) As Byte
l2 = DBI_CharacterizeImage(finger1(0), Buffer1(0), Buffer2(0), CurDDT(0), q)
l2 = DBI_Compare(CurDDT(0), Ddts(0, 4))
Dim Scores(3) As Long
Scores(0) = l2
l2 = DBI_Multi_Dim(Scores(0), 1, FA_1_IN_10000)
If l2 = 1 Then
compareL = True
frmSplash1.Show
Else
compareL = False
frmSplash2.Show
gs = gs + 1

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    Cnn = True
End If
jin:
    Call DBI_CloseDriver
End Function

```

```

Public Function compareR() As Boolean

```

```

    Dim currentfile As String: Dim a, l2
    l2 = DBI_InitDriver(l, D)
    Call DBI_DLLInit(l, D)
    Dim SerialNumber(1) As Long
    Dim DriverVersion(1) As Long
    Dim RecognitionVersion(1) As Long
    Dim hhh As Date
    l2 = DBI_GetInfo(DBI_BIOMOUSE_SERIAL, SerialNumber(0))
    l2 = DBI_GetInfo(DBI_DRIVER_VERSION, DriverVersion(0))
    l2 = DBI_GetInfo(DBI_RECOGNITION_VERSION, RecognitionVersion(0))
    Dim DITptr(10) As Long: Dim Dits() As Byte: Dim Ddts() As Byte
    ReDim Dits(DBI_sizeofDIT(), 10): ReDim Ddts(DBI_sizeofDDT(), 10)
    For i = 0 To 9
        DITptr(i) = DBI_GetByteAddress(Dits(0, i))
    Next
    Dim finger1(57600) As Byte, Buffer1(57600) As Byte, Buffer2(57600) As Byte
    Dim intN As Long
    intN = DBI_sizeofDIT()
    For j = 0 To intN
        Dits(j, 5) = FingerRS(j, 1)
    Next j
    intN = DBI_sizeofDDT()
    For j = 0 To intN
        Ddts(j, 5) = FingerRRS(j, 1)
    Next j
    l2 = DBI_UnCompress_DIT(Dits(0, 5), Ddts(0, 5))
    currentfile = ID_CodeMember
    Dim aa As Integer: Dim BB As Integer:
    Dim UserName() As Byte
    BB = Len(currentfile)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ReDim UserName(BB + 1)
For aa = 0 To BB - 1
  UserName(aa) = Asc(Mid(currentfile, aa + 1, 1))
Next aa

UserName(aa + 1) = 0

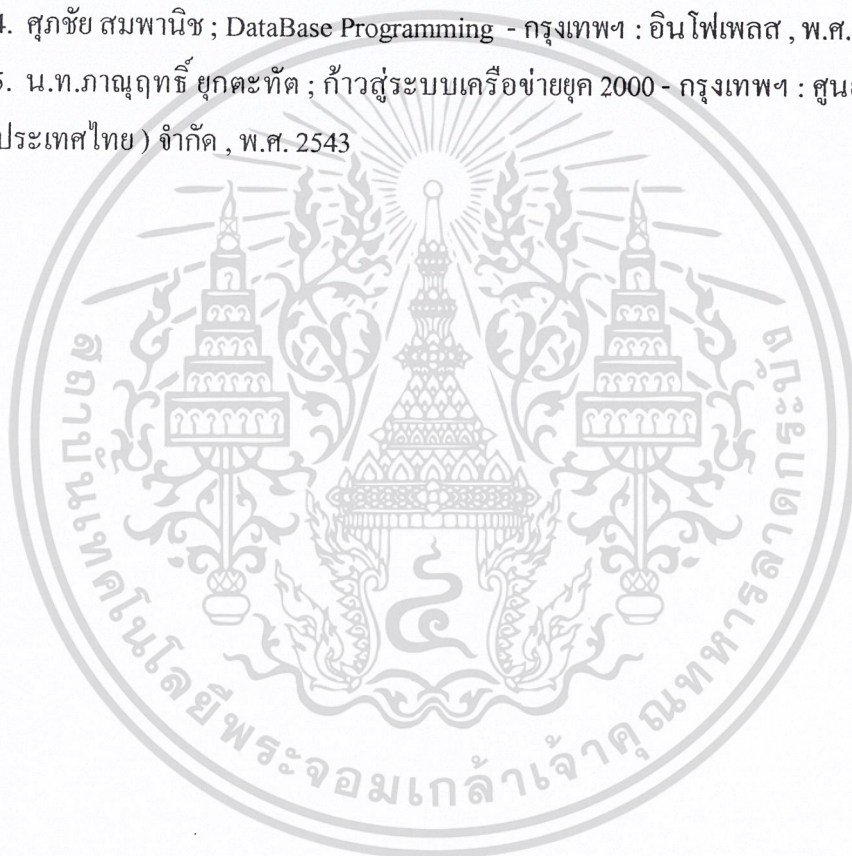
Dim MaxSecurity(1) As Long: Dim Indicator(1) As Integer
MaxSecurity(0) = FA_1_IN_1000000
Dim UserTitle(20) As Byte
UserTitle(0) = Asc("U"): UserTitle(1) = Asc("s"): UserTitle(2) = Asc("e"): UserTitle(3) = Asc("r")
UserTitle(4) = Asc("T"): UserTitle(5) = Asc("i"): UserTitle(6) = Asc("t"): UserTitle(7) = Asc("I")
UserTitle(8) = Asc("e"): UserTitle(9) = Asc("."): UserTitle(10) = 0
Dim AutoOt(1) As Long: Dim AutoSt(1) As Long: Dim AutoDt(1) As Long:
Dim InteractiveAutoDetect(1) As Boolean: Dim Security(1) As Long:
AutoOt(0) = 3000: AutoSt(0) = 3000: AutoDt(0) = 2: InteractiveAutoDetect(0) = 1: Security(0) =
FA_1_IN_100000: Indicator(0) = 0
I2 = DBI_AutoCapture_Dlg(finger1(0), UserName(0), 3000, 3000, 3, 1, 0)
b = DBI_sizeofDDT()
ReDim CurDDT(b) As Byte
I2 = DBI_CharacterizeImage(finger1(0), Buffer1(0), Buffer2(0), CurDDT(0), q)
I2 = DBI_Compare(CurDDT(0), Ddts(0, 5))
Dim Scores(3) As Long
Scores(0) = I2
I2 = DBI_Multi_Dim(Scores(0), 1, FA_1_IN_100000)
If I2 = 1 Then
  compareR = True
  frmSplash1.Show
Else
  compareR = False
  frmSplash2.Show
  gs = gs + 1
  Gnn = True
End If
jin:
  Call DBI_CloseDriver
End Function

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. สมศักดิ์ ศรีจักรเกียรติ; Visual Basic 6 - กรุงเทพฯ : บิบัติโอไฟล์ พับลิชชิง , พ.ศ. 2542
2. กิตติ ภัคดีวัฒนกุล , จำลองครุอุตสาหกรรม ; Visual Basic 6 ฉบับฐานข้อมูล - กรุงเทพฯ : ไทยเจริญการพิมพ์ , พ.ศ. 2543
3. ธนพล ฉันทวิชัย ; การออกแบบและสร้างฐานข้อมูลด้วย Visual Basic 6.0 - กรุงเทพฯ : ซีเอ็ดดูเคชั่น , พ.ศ. 2543
4. ศุภชัย สมพานิช ; DataBase Programming - กรุงเทพฯ : อินโฟเพลส , พ.ศ. 2543
5. น.ท.ภาณุฤทธิ์ ยุทธะทัต ; ก้าวสู่ระบบเครือข่ายยุค 2000 - กรุงเทพฯ : ศูนย์การศึกษาทางไกล (ประเทศไทย) จำกัด , พ.ศ. 2543



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้