



ภาควิชาวิศวกรรม

คณะครุศาสตร์อุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองปริญญาโท

ชื่อหัวข้อ ระบบบันทึกเวลาโดยใช้ลายนิ้วมือ

Time Attention by Fingerprint Scanner

ชื่อนักศึกษา	1. นางสาววิรินทร์นิภา ขจัดมาน	รหัสประจำตัว	48035258
	2. นายกิตติศักดิ์ ยิ่งยง	รหัสประจำตัว	48035261
	3. นางสาวภชนันท์ สิงห์ดารา	รหัสประจำตัว	48035282
	4. นายอัมพวัฒน์ พัทธพัฒน์ชัย	รหัสประจำตัว	48035311

หลักสูตร ครุศาสตร์อุตสาหกรรมบัณฑิต สาขาวิชา วิศวกรรมโทรคมนาคม

อาจารย์ที่ปรึกษา อ.สุรพงษ์ สิริพงษ์ดี

อาจารย์ที่ปรึกษาร่วม อ.สุระชัย พิมพ์สาลี

คณะกรรมการสอบปริญญาโท	ลายมือชื่อ
1. อ.ปิยะ ศุภวาราสวัฒน์	
2. อ.สุรพงษ์ สิริพงษ์ดี	
3. อ.สุระชัย พิมพ์สาลี	
4. อ.พิชญ์สินี มะโน	
5. อ.อำพล ทองระอา	

วัน/เดือน/ปีที่สอบ วันวันศุกร์ที่ 11 เดือนพฤษภาคม พ.ศ. 2550 เวลา 16.00 น.

สถานที่สอบ ห้อง ค.310 คณะครุศาสตร์อุตสาหกรรม สจล.

ภาควิชารับรองแล้ว

ลงนาม.....

(รศ.สุรสิทธิ์ รัตรี)

หัวหน้าภาควิชาวิศวกรรม

วันที่ 30 เดือน พ.ค. พ.ศ. 50



<BT402582>

ระบบบันทึกเวลาโดยใช้ลายนิ้วมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาบัตร

ระบบบันทึกเวลาโดยใช้ลายนิ้วมือ

TIME ATTENTION BY FINGERPRINT SCANNER



วรินทร์นิภา ขจัดมาน
กิตติศักดิ์ ยิ่งยง
ภัชฉันทน์ สิงห์ดารา
อัมพวัฒน์ พิชรพัฒน์ชัย

รฟ.
๑๓๒๘๖
๑๕๕๑

เลขหมู่.....
เลขทะเบียน..... 75194
วัน,เดือน,ปี..... ๒4 ต.ค. 2550

b. 1๙81๖211
i.....

ปริญญาบัตรฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรครุศาสตรบัณฑิต
สาขาวิชาวิศวกรรมโทรคมนาคม
ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์

เรื่อง ระบบบันทึกเวลาโดยใช้ลายนิ้วมือ

Time Attention by Fingerprint Scanner

วัตถุประสงค์

1. เพื่อศึกษาหลักการทำงานเครื่องสแกนลายนิ้วมือและวิธีการทำงานระบบบันทึกเวลา
2. เพื่อออกแบบระบบบันทึกเวลาเพื่อใช้ร่วมกับเครื่องสแกนลายนิ้วมือ
3. เพื่อสร้างระบบบันทึกเวลาเพื่อใช้ร่วมกับเครื่องสแกนลายนิ้วมือ
4. เพื่อทดสอบระบบบันทึกเวลาเพื่อใช้ร่วมกับเครื่องสแกนลายนิ้วมือ
5. เพื่อนำระบบบันทึกเวลาไปใช้ร่วมกับเครื่องสแกนลายนิ้วมือ

ประโยชน์ที่คาดว่าจะได้รับ

1. ได้ความรู้เกี่ยวกับหลักการทำงานเครื่องสแกนลายนิ้วมือและวิธีการทำงานระบบบันทึกเวลา
2. ได้แบบของระบบบันทึกเวลาเพื่อใช้ร่วมกับเครื่องสแกนลายนิ้วมือ
3. ได้สร้างระบบบันทึกเวลาเพื่อใช้ร่วมกับเครื่องสแกนลายนิ้วมือ
4. ได้ผลการทดสอบระบบบันทึกเวลาเพื่อใช้ร่วมกับเครื่องสแกนลายนิ้วมือ
5. ได้นำระบบบันทึกเวลาไปใช้ร่วมกับเครื่องสแกนลายนิ้วมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ	ระบบบันทึกเวลาโดยใช้ลายนิ้วมือ
นักศึกษา	นางสาววรินทร์นิภา ขจัดมาน
	นายกิตติศักดิ์ ยิ่งยง
	นางสาวภัชชนันท์ สิงห์ดารา
	นายอัมพัทธ์พนธ์ พัชรพัฒน์ชัย
อาจารย์ที่ปรึกษา	อาจารย์สุรพงษ์ สิริพงศ์ดี
อาจารย์ที่ปรึกษาร่วม	อาจารย์สุระชัย พิมพ์สาส์
หลักสูตร	ครุศาสตร์อุตสาหกรรมบัณฑิต
สาขาวิชา	วิศวกรรมโทรคมนาคม
ปีการศึกษา	2549

บทคัดย่อ

บริษัทยาพันธ์ฉบับนี้ได้อธิบายถึงหลักการทำงานและการออกแบบของระบบบันทึกเวลาเข้า-ออกของพนักงานโดยการใช้ลายนิ้วมือ โดยอาศัยเครื่องสแกนลายนิ้วมือในการบันทึกเวลาเข้า-ออกของพนักงาน แบ่งออกเป็นส่วนต่างๆด้วยกัน คือ การบันทึกเวลาเข้า-ออกของพนักงาน การตรวจสอบเวลาเข้า-ออกของพนักงาน การจัดเก็บข้อมูลของพนักงาน และการจัดการข้อมูลลายนิ้วมือ ลักษณะการทำงานของระบบบันทึกเวลาโดยการใช้ลายนิ้วมือ เมื่อพนักงานทำการบันทึกเวลาโดยทำการสแกนลายนิ้วมือ ระบบจะทำการประมวลผลข้อมูลลายนิ้วมือเพื่อค้นหาข้อมูลของพนักงานและทำการจัดเก็บข้อมูลเวลาเข้า-ออกของพนักงานลงในฐานข้อมูล

การตรวจสอบข้อมูลโดยใช้ลายนิ้วมือ จะมีความถูกต้องแม่นยำมากกว่าการลงลายมือชื่อหรือการใช้บัตรพนักงาน ซึ่งจะช่วยลดปัญหาความผิดพลาดในการจัดเก็บข้อมูลและการทุจริตภายในองค์กร

II

Thesis Title	Time Attention by Fingerprint Scanner
Students	Miss.Warunnipar Kajudmarn Mr.Kitisak Yingyong Mr.Ampawat Phatcharaphattanachai Miss.Patchanan Singdara
Advisor	Mr.Surapong Siripongdee
Co-Advisor	Mr.Surachai Pimsalee
Education Level	Bachelor of Science in Industrial Education
Program in	Telecommunication Engineering
Academic Year	2006

ABSTRACT

This thesis is about the operation principles and design of the time attention by fingerprint scanner for the staff to check in and out by printing fingers. The time attention by fingerprint scanner consists of several parts such as time check in and out, time monitoring, general information of the staff and fingerprint information. The working system of the time attention fingerprint scanner is to begin when the staff scan their fingerprint then the system will search for the general information of the staff for keeping the record

The information checking fingerprint is more accuracy card. If also reduce the mistake in keeping the records of corruption in the organization

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สามารถสำเร็จลุล่วงได้ด้วยดีนั้น เนื่องมาจากความร่วมมือร่วมใจในการทำงานของสมาชิกภายในกลุ่มทุกท่าน คณะผู้จัดทำขอขอบพระคุณ อาจารย์สุรพงษ์ สิริพงศ์ดี และ อาจารย์สุรชัย พิมพ์สาลี อาจารย์ประจำภาควิชาครุศาสตร์วิศวกรรมท่านได้กรุณาให้คำปรึกษาและคำแนะนำในการแก้ไขปัญหาต่างๆ ตลอดจนข้อมูลและอุปกรณ์ที่เป็นประโยชน์ต่อการทำโครงการครั้งนี้ ขอขอบคุณห้องสมุดคณะครุศาสตร์อุตสาหกรรม สำนักหอสมุดกลาง ที่อำนวยความสะดวกในการค้นคว้าข้อมูล และขอขอบคุณเพื่อนๆทุกคนที่ให้คำแนะนำต่างๆในการทำโครงการ

ขอกราบขอบพระคุณบิดา มารดา และผู้ที่มีพระคุณสำหรับพวกเราที่ได้ให้การสนับสนุนทุกๆ สิ่งทุกอย่างอย่างทางด้านการศึกษาจนถึงปัจจุบัน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VIII
สารบัญรูป	X
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 จุดมุ่งหมายของโครงการ	1
1.3 สมมุติฐานของโครงการ	1
1.4 ขีดความสามารถของโครงการ	1
1.5 ขั้นตอนการทำโครงการ	2
1.6 เนื้อหาโดยสังเขป	2
บทที่ 2 ทฤษฎีและหลักการ	3
2.1 กล่าวนำ	3
2.2 พื้นฐานของลายนิ้วมือ	3
2.3 ความรู้เบื้องต้นเกี่ยวกับลายนิ้วมือ	4
2.3.1 คำจำกัดความที่ใช้ในการศึกษาเรื่องลายนิ้ว	4
2.3.2 ระบบ AFVS (Automated Fingerprint Verification System)	6
2.3.3 วิธีการแยกแยะลักษณะของลายนิ้วมือโดยอาศัยทฤษฎี Minutiae	7
2.3.4 การใช้เทคโนโลยี Biologon	7
2.3.5 การใช้เทคโนโลยี Automated Fingerprint Identification System	8
2.3.6 เครื่องตรวจสอบลายนิ้วมือ โดยใช้ทฤษฎีไบโอเมตริก	9
2.4 การอ่านลายนิ้วมือด้วยเทคโนโลยีแสง	9
2.4.1 ที่มาของการอ่านลายนิ้วมือด้วยเทคโนโลยีแสง	9
2.4.2 การอ่านลายนิ้วมือด้วยเทคโนโลยีแสง	10

สารบัญ (ต่อ)

เรื่อง	หน้า
2.5 ความรู้พื้นฐานของระบบฐานข้อมูล	13
2.6 องค์ประกอบของระบบฐานข้อมูล	14
2.7 การออกแบบฐานข้อมูล	16
2.7.1 เป้าหมายของการออกแบบข้อมูล	16
2.7.2 พื้นฐานหลักการออกแบบฐานข้อมูล	17
2.8 การเขียน Visual Basic ติดต่อ Microsoft Access	20
2.8.1 Open Database Connectivity (ODBC)	20
2.8.2 การติดตั้ง ODBC Data Source	20
2.9 หลักการใช้งานเครื่องสแกนลายนิ้วมือ	21
2.9.1 การเตรียมขั้นตอนการใช้งานเครื่องสแกนลายนิ้วมือ	21
2.9.2 ส่วนประมวลผลลายนิ้วมือ - เริ่มการใช้งาน	21
2.9.3 ส่วนประมวลผลลายนิ้วมือ - จบการใช้งาน	21
2.9.4 ส่วนประมวลผลลายนิ้วมือ - Context	23
2.9.5 ส่วนประมวลผลลายนิ้วมือ - ข้อมูลลายนิ้วมือ	23
2.9.6 ส่วนประมวลผลลายนิ้วมือ - เครื่องอ่านลายนิ้วมือลายนิ้วมือ	26
2.9.7 ส่วนประมวลผลลายนิ้วมือ - การจัดการ Message	27
2.10 สรุปลักษณะการใช้งานของเครื่องสแกน U.are.U 4000	28
2.10.1 สรุปลักษณะทั้งหมดใน IDWORKS Integrator - ส่วนการประมวลผลลายนิ้วมือ	28
2.10.2 สรุปลักษณะทั้งหมดใน IDWORKS Integrator - ส่วนการลงทะเบียนลายนิ้วมือ	29
2.10.3 สรุปลักษณะทั้งหมดใน IDWORKS Integrator - ส่วนข้อมูลเครื่องอ่านลายนิ้วมือ	30
บทที่ 3 การออกแบบ การสร้าง และการทำงาน	31
3.1 กล่าวนำ	31
3.2 การออกแบบ	31
3.2.1 ส่วนของเครื่องสแกนลายนิ้วมือ	31
3.2.2 ส่วนของการเตรียมเครื่องคอมพิวเตอร์ที่ใช้	33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

เรื่อง	หน้า
3.2.3 ส่วนของโปรแกรมที่ใช้	33
3.2.4 ส่วนของการตรวจสอบลายนิ้วมือ	41
3.2.5 ส่วนของฐานข้อมูล	42
บทที่ 4 การทดลองและผลการทดลอง	50
4.1 กล่าวนำ	50
4.2 การทดลองการทำงานของระบบบันทึกเวลาโดยใช้ลายนิ้วมือ	50
4.3 การทดสอบเครื่องสแกนลายนิ้วมือ DigitalPersona U.are.U 4000	50
4.3.1 การทดลอง	50
4.3.2 ผลการทดลอง	53
4.4 การเข้าสู่ระบบบันทึกเวลาโดยใช้ลายนิ้วมือของเจ้าหน้าที่	43
4.4.1 การทดลอง	43
4.4.2 ผลการทดลอง	54
4.5 การเข้าสู่ระบบบันทึกและแก้ไขข้อมูลของพนักงาน	54
4.5.1 การทดลอง	54
4.5.2 ผลการทดลอง	55
4.6 การกำหนดกะการทำงานของพนักงาน	55
4.6.1 การทดลอง	55
4.6.2 ผลการทดลอง	56
4.7 การตรวจเช็คการลงเวลาเข้า-ออก ของพนักงาน	56
4.7.1 การทดลอง	56
4.7.2 ผลการทดลอง	58
4.8 การตรวจเช็คการมาสายและการขาดงานของพนักงาน	59
4.8.1 การทดลอง	59
4.8.2 ผลการทดลอง	61
4.9 การกำหนดวันหยุดของบริษัท	61
4.9.1 การทดลอง	61
4.9.2 ผลการทดลอง	62

สารบัญ (ต่อ)

เรื่อง	หน้า
บทที่ 5 บทสรุป	62
5.1 สรุป	62
5.2 ปัญหาและวิธีการแก้ไข	62
5.3 แนวทางการพัฒนา	62
บรรณานุกรม	64
ภาคผนวก ก เครื่องต้นแบบ	65
ภาคผนวก ข รายการอุปกรณ์	67
ภาคผนวก ค รายละเอียดและคุณสมบัติของอุปกรณ์	69
ภาคผนวก ง แผนผังการทำงานของโปรแกรม	101
ภาคผนวก จ แผนผังความสัมพันธ์ของเอนทิตี	111
ภาคผนวก ฉ ต้นฉบับของโปรแกรม	113
ภาคผนวก ช คู่มือการใช้งานระบบบันทึกเวลาโดยการใช้ลายนิ้วมือ	129
ประวัติผู้แต่ง	136

สารบัญตาราง

ตารางที่	หน้า
2.1 ฟังก์ชันการเริ่มและเลิกใช้งานไลบรารีของส่วนการประมวลผลลายนิ้วมือ	28
2.2 ฟังก์ชันการสร้างและเปิด context และการ Activate ของส่วนการประมวลผลลายนิ้วมือ	28
2.3 ฟังก์ชันการควบคุมเครื่องอ่านลายนิ้วมือของส่วนการประมวลผลลายนิ้วมือ	29
2.4 ฟังก์ชันการเพิ่มและลบลายนิ้วมือของส่วนการประมวลผลลายนิ้วมือ	29
2.5 ฟังก์ชันการเริ่มและเลิกใช้งานไลบรารีของส่วนการลงทะเบียนลายนิ้วมือ	29
2.6 ฟังก์ชันการสร้างและเปิด context และการ Activate ของส่วนการลงทะเบียนลายนิ้วมือ	30
2.7 ฟังก์ชันการลงทะเบียนลายนิ้วมือของส่วนการลงทะเบียนลายนิ้วมือ	30
2.8 ฟังก์ชันการเริ่มและเลิกใช้งานไลบรารีของส่วนข้อมูลเครื่องอ่านลายนิ้วมือ	30
2.9 ฟังก์ชันการตั้งข้อมูลเครื่องอ่านลายนิ้วมือของส่วนข้อมูลเครื่องอ่านลายนิ้วมือ	31
3.1 สรุปรายชื่อตารางทั้งหมดในโปรแกรมระบบลงเวลาโดยใช้ลายนิ้วมือ	43
3.2 ตารางรายละเอียดของ IdiFingerprint	43
3.3 ตารางรายละเอียดของ tbl_employee	44
3.4 ตารางรายละเอียดของ tbl_employee	45
3.5 ตารางรายละเอียดของ tbl_position	45
3.6 ตารางรายละเอียดของ tbl_section	46
3.7 ตารางรายละเอียดของ tbl_holiday	46
3.8 ตารางรายละเอียดของ tbl_work_la	47
3.9 ตารางรายละเอียดของ tbl_work_stop	48
3.10 ตารางรายละเอียดของ tbl_wokrlog	48
4.1 ผลการบันทึกวันที่และเวลาการเข้าทำงานของพนักงาน	58
4.1 (ต่อ) ผลการบันทึกวันที่และเวลาการเข้าทำงานของพนักงาน	59
4.2 ผลการบันทึกวันที่และเวลาการออกจากทำงานของพนักงาน	59
ข.1 รายการอุปกรณ์ของระบบส่งงานโดยการยืนยันลายนิ้วมือ	68
ค.1 คำ MsgID กับคำ MsgParam ของฟังก์ชัน IDI_IDENTEVENTMSG	81
ค.2 คำ MsgID กับคำ MsgParam ของฟังก์ชัน IDI_SENSOREVENTMSG	84
ค.3 ค่าคงที่สำหรับ IDI_RCODE	87

สารบัญตาราง (ต่อ)

ตารางที่	หน้า
ค.4 ค่าคงที่สำหรับ IDI_IDENTEVENTTYPE	88
ค.5 ค่าคงที่สำหรับ IDI_SENSORTYPE	88
ค.6 ค่าคงที่สำหรับ IDI_SENSOREVENTTYPE	88
ค.7 ค่าคงที่ทั่วไปของส่วนข้อมูลเครื่องอ่านลายนิ้วมือ	94
ค.8 ค่าคงที่สำหรับ ISL_RCODE ของส่วนข้อมูลเครื่องอ่านลายนิ้วมือ	94
ค.9 ค่าคงที่สำหรับ ISL_DEVICETYPE ของส่วนข้อมูลเครื่องอ่านลายนิ้วมือ	94
ค.10 ค่าคงที่ทั่วไปของส่วนช่วยการบันทึกลายนิ้วมือ	100
ค.11 ค่าคงที่สำหรับ ISL_RCODE ของส่วนช่วยการบันทึกลายนิ้วมือ	100
ค.12 ค่าคงที่สำหรับ ISL_DEVICETYPE ของส่วนช่วยการบันทึกลายนิ้วมือ	100
ช.1 การแก้ไขปัญหาเบื้องต้น	134
ช.2 ข้อมูลจำเพาะ	135

สารบัญรูป

รูปที่	หน้า
2.1 ลักษณะรูปขยายของลายนิ้วมือ	3
2.2 แบบเส้นโค้ง (Arch)	5
2.3 แบบลูปหรือมัดหวาย (Loop)	5
2.4 แบบก้นหอย	6
2.5 ลายมือและรูปขยายในส่วนของนิ้วมือ	10
2.6 ระบบการอ่านลายนิ้วมือแบบใช้หลักการกระจายของแสง	11
2.7 ผลการอ่านลายนิ้วมือที่ใช้หลักการของการกระจายแสง	12
2.8 การจัดตั้งอุปกรณ์สำหรับการอ่านลายนิ้วมือที่ใช้หลักการสะท้อนกลับหมดของแสง	12
2.9 ผลการอ่านลายนิ้วมือที่ใช้หลักการสะท้อนกลับหมดของแสงและหลักการของแสงที่หายไปอย่างรวดเร็ว	13
2.10 ตัวอย่างฐานข้อมูล	14
2.11 องค์ประกอบของระบบฐานข้อมูล	15
2.12 ขั้นตอนการออกแบบฐานข้อมูล	17
2.13 การเรียกฟังก์ชัน <code>idi_InitModule</code> เป็นการเริ่มใช้งาน	22
2.14 การเรียกฟังก์ชัน <code>idi_FinalizeModule</code> เป็นการจบการใช้งาน	22
2.15 การเรียกใช้งานของ Context	23
2.16 การปิด Context	24
2.17 การเพิ่มข้อมูลเข้าไปในฐานข้อมูลชั่วคราว	24
2.18 ข้อมูลประเภท <code>FP_FPINFO</code>	25
2.19 การใช้ฟังก์ชันคัดลอกข้อมูลลายนิ้วมือ	26
2.20 การรับข้อมูลจากเครื่องอ่านลายนิ้วมือ	27
2.21 การทำงานของเครื่องอ่านลายนิ้วมือ	27
3.1 ผังการทำงานของระบบบันทึกเวลาโดยใช้ลายนิ้วมือ	32
3.2 เครื่องสแกนลายนิ้วมือรุ่น DigitalPersona U.are.U 4000	33
3.3 φόρμ การ Login เข้าสู่ระบบ	34
3.4 φόρμ หลักของระบบบันทึกเวลาโดยใช้ลายนิ้วมือ	35
3.5 φόρμ บันทึกเวลาเข้า-ออกของพนักงาน	35
3.6 φόρμ บันทึกข้อมูลและลายนิ้วมือของพนักงาน	36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า	
3.7	ฟอร์ม การตรวจสอบเวลาเข้า-ออกของพนักงาน	36
3.8	ฟอร์ม การตรวจสอบการขาดงานหรือมาสายของพนักงาน	37
3.9	ฟอร์ม กำหนดเวลาการทำงานของพนักงาน	38
3.10	ฟอร์ม กำหนดวันหยุดของพนักงาน	38
3.11	ฟอร์ม ตารางตรวจสอบการเข้าทำงานและออกของพนักงานเฉพาะบุคคล	39
3.12	ฟอร์ม ตารางตรวจสอบการเข้าทำงานและออกของพนักงานทั้งหมด	39
3.13	ฟอร์ม ตารางตรวจสอบการขาดงานของพนักงานเฉพาะบุคคล	40
3.14	ฟอร์ม ตารางตรวจสอบการขาดงานของพนักงานทั้งหมด	40
3.15	ฟอร์ม ตารางตรวจสอบการหยุดงานของพนักงานเฉพาะบุคคล	41
3.16	ฟอร์ม ตารางตรวจสอบการหยุดงานของพนักงานทั้งหมด	41
3.17	ฟอร์ม ลงทะเบียนลายนิ้วมือ	42
3.18	รายละเอียดข้อมูลในตาราง IdiFingerprint	43
3.19	รายละเอียดข้อมูลในตาราง tbl_employee	44
3.20	รายละเอียดข้อมูลในตาราง tbl_department	45
3.21	รายละเอียดข้อมูลในตาราง tbl_position	46
3.22	รายละเอียดข้อมูลในตาราง tbl_section	46
3.23	รายละเอียดข้อมูลในตาราง tbl_holiday	47
3.24	รายละเอียดข้อมูลในตาราง tbl_work_late	47
3.25	รายละเอียดข้อมูลในตาราง tbl_work_stop	48
3.26	รายละเอียดข้อมูลในตาราง tbl_wokrlog	49
3.27	แสดงการเชื่อมโยงความสัมพันธ์ระหว่างตาราง (Query)	49
4.1	หน้าต่างการลงทะเบียนตัวเครื่องสแกนกับโปรแกรม	51
4.2	การสแกนลายนิ้วมือเพื่อเก็บข้อมูลลายนิ้วมือ	51
4.3	การสแกนลายนิ้วมือเพื่อเก็บข้อมูลลายนิ้วมือ	52
4.4	การบันทึกข้อมูลลายนิ้วมือลงในฐานข้อมูลจำลอง	52
4.5	ข้อมูลที่บันทึกในฐานข้อมูล	53
4.6	การสแกนลายนิ้วมือ	53
4.7	หน้าต่าง Time Attention By Fingerprint	54

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.8 การบันทึกและแก้ไขข้อมูลพนักงาน	55
4.9 หน้าต่าง Work Schedule	56
4.10 หน้าต่างการลงเวลาเข้า ของพนักงาน	57
4.11 หน้าต่างการลงเวลาเข้า กรณีลงเวลาเกินกว่าเวลาที่กำหนดไว้	57
4.12 หน้าต่างการสแกนลายนิ้วมือลงเวลาออก	58
4.13 หน้าต่างรายงานสรุปพนักงานที่มาสาย	60
4.14 หน้าต่างรายงานสรุปพนักงานที่ขาดงาน	60
4.15 หน้าต่างแสดงข้อมูลวันหยุดของบริษัท	61
ก.1 ด้านหน้าของเครื่องสแกนลายนิ้วมือ	66
ก.2 ด้านข้างของเครื่องสแกนลายนิ้วมือ	66
ค.1 รายละเอียดฟังก์ชันของ idi_InitModule	70
ค.2 รายละเอียดฟังก์ชันของ idi_FinalizeModule	70
ค.3 รายละเอียดฟังก์ชันของ idi_CreateContext	71
ค.4 รายละเอียดฟังก์ชันของ idi_CloseContext	71
ค.4 (ต่อ) รายละเอียดฟังก์ชันของ idi_CloseContext	72
ค.5 รายละเอียดฟังก์ชันของ idi_ActivateSensor	72
ค.5 (ต่อ) รายละเอียดฟังก์ชันของ idi_ActivateSensor	73
ค.6 รายละเอียดฟังก์ชันของ idi_DeactivateSensor	73
ค.7 รายละเอียดฟังก์ชันของ idi_StartCapturing	74
ค.8 รายละเอียดฟังก์ชันของ idi_StopCapturing	75
ค.9 รายละเอียดฟังก์ชันของ idi_RegisterSensorEventProc	75
ค.9 (ต่อ) รายละเอียดฟังก์ชันของ idi_RegisterSensorEventProc	76
ค.10 รายละเอียดฟังก์ชันของ idi_UnregisterSensorEventProc	76
ค.10 (ต่อ) รายละเอียดฟังก์ชันของ idi_UnregisterSensorEventProc	77
ค.11 รายละเอียดฟังก์ชันของ idi_RegisterIdentEventProc	77
ค.12 รายละเอียดฟังก์ชันของ idi_UnregisterIdentEventProc	78
ค.13 รายละเอียดฟังก์ชันของ idi_fp_Add	79
ค.14 รายละเอียดฟังก์ชันของ idi_fp_Add	79

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
ค.14 (ต่อ) รายละเอียดฟังก์ชันของ idi_fp_Add	80
ค.15 ประเภทข้อมูลของ IDI_CONTEXT	80
ค.16 ประเภทข้อมูลของ IDI_IDENTEVENTMSG	81
ค.17 ประเภทข้อมูลของ IDI_IDENTEVENTTYPE	81
ค.18 ประเภทข้อมูลของ IDI_IDENTEVENTPARAM	82
ค.19 ประเภทข้อมูลของ IDI_REFPARAM	82
ค.20 ประเภทข้อมูลของ IDI_IDENTEVENTPROC	83
ค.21 ประเภทข้อมูลของ IDI_IDENTEVENTMATCHEDSTRUCT	83
ค.22 ประเภทข้อมูลของ IDI_SENSOREVENTMSG	84
ค.23 ประเภทข้อมูลของ IDI_SENSOREVENTPROC	85
ค.24 ประเภทข้อมูลของ IDI_SENSOREVENTIMAGEACQUIREDSTRUCT	85
ค.24 (ต่อ) ประเภทข้อมูลของ IDI_SENSOREVENTIMAGEACQUIREDSTRUCT	86
ค.25 ประเภทข้อมูลของ FP_FPINFO	86
ค.25 (ต่อ) ประเภทข้อมูลของ FP_FPINFO	87
ค.26 รายละเอียดฟังก์ชันของ isl_IsModuleInited	89
ค.27 รายละเอียดฟังก์ชันของ isl_IsModuleInited	89
ค.28 รายละเอียดฟังก์ชันของ isl_FinalizeModule	90
ค.29 รายละเอียดฟังก์ชันของ isl_GetDeviceCount	90
ค.30 รายละเอียดฟังก์ชันของ isl_GetDeviceList	91
ค.31 ประเภทข้อมูลของ ISL_RCODE	91
ค.32 ประเภทข้อมูลของ ISL_DEVICETYPE	92
ค.33 ประเภทข้อมูลของ SL_DEVICEID	92
ค.34 ประเภทข้อมูลของ ISL_CHAR	92
ค.35 ประเภทข้อมูลของ ISL_SENSORINFO	93
ค.36 รายละเอียดฟังก์ชันของ fpe_InitModule	95
ค.37 รายละเอียดฟังก์ชันของ fpe_FinalizeModule	95
ค.38 รายละเอียดฟังก์ชันของ fpe_CreateContext	96
ค.39 รายละเอียดฟังก์ชันของ fpe_CloseContext	96

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
ค.40 รายละเอียดฟังก์ชันของ fpe_ActivateSensor	97
ค.41 รายละเอียดฟังก์ชันของ fpe_StartEnrolment	98
ค.42 ประเภทข้อมูลของ FPE_RCODE	99
ค.43 ประเภทข้อมูลของ FPE_CONTEXT	99
ค.44 ประเภทข้อมูลของ FPE_SENSORTYPE	99
ง.1 แผนผังการเข้า Login	102
ง.2 แผนผังระบบส่งงาน	103
ง.3 แผนผังการตรวจสอบการส่งงานของนักศึกษา	104
ง.4 แผนผังการเพิ่มงานของอาจารย์	105
ง.5 แผนผังการแก้ไขงานของอาจารย์	106
ง.6 แผนผังการเพิ่มข้อมูลของนักศึกษา	107
ง.7 แผนผังการแก้ไขข้อมูลนักศึกษา	108
ง.8 แผนผังการบันทึกข้อมูลการสแกนลายนิ้วมือ	109
ง.9 แผนผังการแก้ไขข้อมูลลายนิ้วมือ	110
จ.1 แผนผังความสัมพันธ์ของเอนทิตีของระบบส่งงานโดยการยืนยันลายนิ้วมือ	112
ช.1 ส่วนประกอบและปุ่มควบคุมของเครื่องสแกนลายนิ้วมือ	131
ช.2 สำหรับ login เข้าสู่ระบบเป็นเวลาโดยใช้ลายนิ้วมือในส่วนของผู้ใช้หน้า	132
ช.3 หน้าต่างสำหรับบันทึกเวลาโดยใช้ลายนิ้วมือในส่วนของผู้ใช้พนักงาน	132
ช.4 ส่วนประกอบและปุ่มการใช้งานของระบบบันทึกเวลาโดยใช้ลายนิ้วมือในส่วนของผู้ใช้พนักงาน	133

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบัน การทำงานของพนักงานและข้าราชการในการมาทำงานทุกวัน นั้นจะต้องลงเวลาในการเข้างานและออกจากงาน โดยการใช้บัตรตอกลงเวลาหรือการเซ็นชื่อเข้าทำงานแต่พบว่าเมื่อพนักงานมาทำงานสายหรือมาไม่ตรงเวลาก็จะฝากเพื่อนมาตอกบัตรให้หรือเซ็นชื่อให้กัน ทำให้เกิดการทุจริตขึ้นในองค์กร อาจจะทำให้เกิดความเสียหายให้กับบริษัทหรือหน่วยงานได้

1.2 จุดมุ่งหมายของโครงการ

คณะผู้จัดทำได้สร้างระบบบันทึกเวลาเพื่อใช้ร่วมกับเครื่องสแกนลายนิ้วมือ เพื่อนำไปใช้ในลงเวลาเข้าออกของพนักงาน เพื่อความสะดวกในการตรวจสอบการมาทำงานของพนักงาน

1.3 สมมุติฐานของการจัดทำโครงการ

เมื่อผ่านการทดลองใช้ระบบบันทึกเวลาโดยใช้ลายนิ้วมือ โดยระบบสามารถใช้งานได้จริง มีความสะดวกและถูกต้องแม่นยำ

1.4 ขีดความสามารถของโครงการ

โครงการนี้มีขีดความสามารถดังนี้

1. สามารถบันทึกข้อมูลลายนิ้วมือของพนักงานคนละ 2 ลายนิ้วมือ จำนวน 10,000 คน
2. ใช้จอคอมพิวเตอร์ในการแสดงรายละเอียดของพนักงาน เมื่อทำการสแกนลายนิ้วมือ
3. สามารถตรวจสอบรายละเอียดการเข้าทำงานของผู้ใช้ เช่น จำนวนวันเข้าทำงาน จำนวนวันมาสาย วันขาด วันลา
4. ตั้งค่าการทำงานเบื้องต้นของระบบสแกนลายนิ้วมือ เช่น กำหนดวันหยุดของบริษัท เวลาเข้า-ออก ข้อมูลบริษัท ช่วงกะการทำงาน
5. มีรายงานแสดงข้อมูล เข้า-ออก โดยกำหนดช่วงวันที่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 ขั้นตอนของการทำโครงการ

โครงการนี้ประกอบไปด้วยซอฟต์แวร์และฮาร์ดแวร์ซึ่งการทำงานในระยะแรกเริ่มต้น โดยการศึกษาวิธีการใช้โปรแกรม และออกแบบฐานข้อมูลของพนักงาน และทำการเขียนโปรแกรมเพื่อติดต่อกับเครื่องสแกนลายนิ้วมือ

1.6 เนื้อหาโดยสังเขป

เนื้อหาในปฏิญานิพนธ์ฉบับนี้แบ่งออกเป็นบทต่างๆ เพื่อสะดวกต่อการศึกษาและทำความเข้าใจในแต่ละบทจะประกอบด้วยเนื้อหาต่อไปนี้

บทที่ 1 กล่าวถึงความเป็นมาและความสำคัญของปฏิญานิพนธ์ ชี้ความสามารถของโครงการและเนื้อหาในบทต่างๆ โดยสังเขป

บทที่ 2 ประกอบด้วยทฤษฎีต่างๆ ที่เกี่ยวกับระบบฐานข้อมูล โครงสร้างของเครื่องสแกนลายนิ้วมือ ทฤษฎีลายนิ้วมือ วิธีการใช้งานเครื่องสแกนลายนิ้วมือ และทฤษฎีและหลักการต่างๆ ที่เกี่ยวข้องกับระบบสงงานโดยการยืนยันลายนิ้วมือ

บทที่ 3 กล่าวถึงเนื้อหาที่เกี่ยวกับ แผนผังการทำงานของโครงการ การออกแบบส่วนประกอบต่างๆ ของโปรแกรม เพื่อให้ทำงานได้อย่างมีประสิทธิภาพ

บทที่ 4 การทดลองและผลการทดลองในบทนี้จะกล่าวถึงการทดลองและผลการทดลองโดยการแบ่งผลการทดลองออกเป็นส่วนๆ ตามการออกแบบและการสร้าง พร้อมทั้งบันทึกผลการทดลองในแต่ละส่วน

บทที่ 5 เป็นการสรุปผลการจัดทำโครงการ ปัญหาที่เกิดขึ้นและแนวทางในการแก้ไข รวมทั้งแนวทางการพัฒนา

ภาคผนวก ก เครื่องต้นแบบ การติดตั้ง การเชื่อมต่อกับอุปกรณ์อื่นๆ ขณะใช้งานจริง

ภาคผนวก ข รายการอุปกรณ์ที่ใช้งานในโครงการ

ภาคผนวก ค รายละเอียดและคุณสมบัติของอุปกรณ์

ภาคผนวก ง แผนผังการทำงานของโปรแกรม

ภาคผนวก จ แผนผังความสัมพันธ์ของเอนทิตี

ภาคผนวก ฉ รหัสต้นฉบับของโปรแกรม

ภาคผนวก ช คู่มือการใช้งานระบบบันทึกเวลาโดยการใช้ลายนิ้วมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

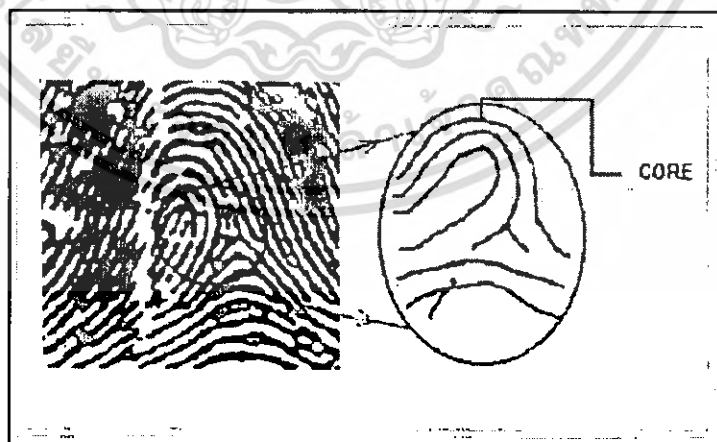
ทฤษฎีและหลักการ

2.1 กล่าวนำ

การสแกนลายนิ้วมือเข้าทำงานนั้นมีประโยชน์อย่างมาก สำหรับการลงเวลาเข้าและออกทำงานในปัจจุบัน โดยเนื้อหาของปริศยานิพนธ์ในบทนี้จะเป็นทฤษฎีและหลักการที่จะนำมาใช้ประกอบการสร้างโครงการเครื่องบันทึกเวลาโดยใช้ลายนิ้วมือประกอบด้วยเนื้อหาดังต่อไปนี้คือ พื้นฐานของลายนิ้วมือ ความรู้เบื้องต้นเกี่ยวกับลายนิ้วมือ ความรู้พื้นฐานของระบบฐานข้อมูล การอ่านลายนิ้วมือด้วยเทคโนโลยีแสงซึ่งเนื้อหาต่างๆ เหล่านี้ล้วนแต่เป็นประโยชน์สำหรับผู้สนใจเป็นอย่างยิ่ง

2.2 พื้นฐานของลายนิ้วมือ

จุดประสงค์ของงานปริศยานิพนธ์นี้ เพื่อแสดงถึงความจำเป็นที่ต้องมีการประมวลผลลายนิ้วมือเบื้องต้นเพื่อใช้ในระบบเปรียบเทียบลายนิ้วมืออัตโนมัติ โดยได้อธิบายถึงลักษณะและรูปแบบของลายนิ้วมือ ทฤษฎีที่เกี่ยวข้องกับลายนิ้วมือทั้งการตรวจเปรียบเทียบโดยใช้บุคคลและการตรวจเปรียบเทียบโดยใช้เครื่องจักร อธิบายถึงการทำงานของเครื่องตรวจเปรียบเทียบลายนิ้วมือทั้งหมดนี้ก็เพื่อให้เห็นถึงภาพรวมของกระบวนการตรวจเปรียบเทียบลายนิ้วมือและส่วนที่สามารถออกแบบให้ทำงานอย่างอัตโนมัติมากขึ้น เช่น การนับเส้นของลายนิ้วมือ เป็นต้น ดังรูปที่ 2.1



รูปที่ 2.1 ลักษณะรูปร่างของลายนิ้วมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการตรวจสอบลายนิ้วมือนี้ได้มุ่งเน้นการพัฒนาขั้นตอนเปรียบเทียบลายนิ้วมือโดยใช้ลายพิมพ์นิ้วมือของบุคคลทั่วไปมาแก้ไขให้อยู่ในรูปขาวดำ (Binary) พร้อมทั้ง ปรับการแก้ไขภาพ จนชัดเจน (Image Enhancement) และเก็บไว้ในฐานข้อมูล เพื่อใช้เป็นแบบการ ตรวจสอบ โดยเก็บค่าที่ได้จากการนับเส้นของลายนิ้วมือโดยอัตโนมัติ ภายหลังกระบวนการทำให้ภาพกลับมาเหมือนเดิม (Prepossessing) และเก็บเฉพาะลายเส้นที่อยู่ระหว่างสันดอน (Delta) ถึงจุดใจกลาง (Core) จากนั้นกำหนดให้เป็นชื่อแฟ้มหนึ่งๆ แตกต่างกันไปตามกลุ่มแต่ละแบบ โดยมีทั้งหมด 12 แบบมาตรฐาน ซึ่งมีข้อมูลเป็นจำนวนมากเก็บบันทึกเอาไว้เปรียบเทียบเช่นนำไปใช้เปรียบเทียบในการพิสูจน์ลายนิ้วมือของบุคคลคนหนึ่งต่อลายนิ้วมือที่มีอยู่เป็นจำนวนมาก

2.3 ความรู้เบื้องต้นเกี่ยวกับลายนิ้วมือ

ที่ผิวหนังบริเวณปลายนิ้วมือ ฝ่ามือ ฝ่าเท้า ของมนุษย์จะประกอบด้วยลายเส้นสองชนิด ชนิดหนึ่งเป็นเส้นนูน อีกชนิดหนึ่งเป็นเส้นร่อง ซึ่งจะสลับกันไปตลอดและจุดลักษณะสำคัญพิเศษ

- เส้นนูน (Ridges) คือการเกิดของรอยนูนที่อยู่สูงขึ้นมาจากผิวหนังส่วนนอก
- เส้นร่อง (Furrows) คือ รอยลึกที่อยู่ต่ำกว่าระดับของเส้นนูน
- จุดลักษณะสำคัญพิเศษ (Characteristics) เป็นตำหนิบนลายนิ้วมือ เช่น จุดเส้นแตก เส้นขาด

เส้นทะเลสาบ เป็นต้น

ซึ่งลักษณะต่างๆ เหล่านี้จะประกอบกันขึ้นเป็นลายนิ้วมือ ซึ่งเป็นเอกลักษณ์เฉพาะของแต่ละบุคคล โดยลายนิ้วมือของแต่ละบุคคลจะไม่เหมือนกัน ที่เป็นเช่นนี้เพราะลายนิ้วมือไม่มีรูปแบบที่แน่นอนตายตัว ไม่มีกฎเกณฑ์ว่าลายนิ้วมือของคนหนึ่งคนใดเป็นเช่นใด ลายนิ้วมือไม่ขึ้นกับอะไร ทั้งสิ้นไม่ขึ้นกับกรรมพันธุ์หญิงหรือชาย จึงมีการใช้ลายนิ้วมือมาเป็นเครื่องมือช่วยในการให้ พนักงานสอบสวนสามารถเข้าถึงตัวผู้กระทำความผิดและสามารถใช้ยืนยันตัวบุคคลผู้กระทำผิดได้

วิธีการพิมพ์ลายนิ้วมือจะทำโดยการใช้หมึกสีดำทาบนนิ้วและกดนิ้วมือไปบนกระดาษขาว จะได้ลายเส้นสีดำสลับกับสีขาว โดยส่วนเส้นสีดำที่ปรากฏจะเป็นส่วนของเส้นนูนส่วนสีขาวจะเป็นเส้นร่องเนื่องจากไม่ถูกกับน้ำหมึกและไม่ถูกกับกระดาษขาว ถ้าเป็นการถ่ายรูปหรือสแกนภาพเข้าไปต้องทำให้เป็นภาพก่อนจึงจะได้ลายเส้นสีดำกับสีขาวตามต้องการ

2.3.1 คำจำกัดความที่ใช้ในการศึกษาเรื่องลายนิ้วมือ

ในการศึกษาเรื่องการจำแนกลายนิ้วมือ มีคำจำกัดความควรรู้จักอยู่ 4 ข้อด้วยกัน ดังนี้

1. เส้นขอบ (Type Line) หมายถึง การที่เส้นคู่ขนานคู่ในสุด ซึ่งได้เดินขนานกัน มาพอสมควรแล้วแยกตัวออกจากกันเพื่อที่จะได้โอบล้อม หรือพยายามโอบล้อมบริเวณลายนิ้วมือที่อยู่ ภายใน โดยเส้นขอบไม่จำเป็นต้องเป็นเส้นยาวราบเรียบติดกันตลอดไป โดยจะถือว่าเส้นที่ขาดหัววน ลงไปนั้นเป็นที่เดินหรือไหลต่อเนื่องกันไป เสมือนว่าเส้นขอบนั้นไม่ได้ขาด

2. สันดอน (Delta) หมายถึง ลายเส้นในลายนิ้วมือ ซึ่งอยู่ตรงหน้าและใกล้ที่สุดกับ กึ่งกลางของปากทางแยกของเส้นขอบ

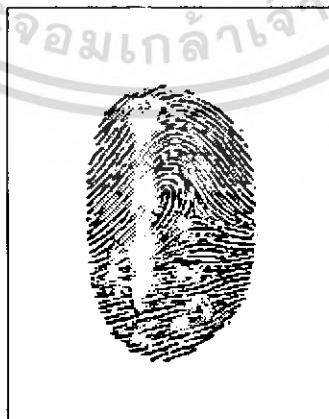
3. จุดใจกลาง (Core) หมายถึง จุดใดจุดหนึ่งบนปลายเส้นหรือบนบ่าหรือไหล่ของ การเลี้ยววกกลับ บริเวณลายนิ้วมือที่อยู่ภายใน (Pattern Area) คือ พื้นที่บริเวณภายในของลายนิ้วมือที่ถูก เส้นขอบโอบล้อม โดยในการจำแนกลายนิ้วมือและรูปแบบของลายนิ้วมือ จะมีกลุ่มใหญ่ๆ 3 กลุ่ม คือ แบบเส้นโค้ง แบบรูปหรือมัดหวาย และแบบก้นหอย

กลุ่มที่ 1 แบบเส้นโค้ง (Arch) ดังรูปที่ 2.2



รูปที่ 2.2 แบบเส้นโค้ง (Arch)

กลุ่มที่ 2 แบบรูปหรือมัดหวาย (Loop) ดังรูปที่ 2.3



รูปที่ 2.3 แบบรูปหรือมัดหวาย (Loop)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กลุ่มที่ 3 แบบก้นหอย (Whorl) ดังรูปที่ 2.4



รูปที่ 2.4 แบบก้นหอย

2.3.2 ระบบ AFVS (Automated Fingerprint Verification System) โดยใช้ทฤษฎี

Minutiae

ในระยะแรกเริ่มของการนำลายพิมพ์นิ้วมือมาใช้ประโยชน์ ทำโดยการนำลายพิมพ์นิ้วมือ ที่ได้ไปทำการให้คำรหัสลายพิมพ์นิ้วมือซึ่งเป็นระบบเฮนรี (Henry System) ต่อมาได้มีการปรับปรุงให้ดีขึ้นตามระบบของหน่วยงานข่าวกรองแห่งชาติ ซึ่งกระบวนการทั้งหมดก็ยังคงเป็นเพียง การจัดกลุ่มลายนิ้วมือของบุคคลจำนวนมากไว้ด้วยกันแต่เมื่อเก็บสะสมไว้เป็นจำนวน หลายๆ ล้านคนก็เริ่มเกิดความยุ่งยากในการตรวจสอบค้นหา เพราะปริมาณงานที่มีเข้ามามากใน วันหนึ่งๆ ทำให้ระบบการตรวจสอบด้วยมือคนเริ่มเกิดความผิดพลาดมากขึ้นๆ คุณภาพของงานไม่คงที่ แนนอน โดยเฉพาะอย่างยิ่งการนำเอาลายนิ้วมือแฝงที่ได้มาจากสถานที่เกิดเหตุเข้าตรวจสอบกับ ฐานข้อมูลซึ่งมีจำนวนนิ้วมือนับล้านนิ้วโดยระบบมือคนจึงเป็นเรื่องที่เป็นไปไม่ได้

กรมตำรวจในเมืองใหญ่ ต่างก็ทราบถึงปัญหานี้และพยายามมองหาวิธีการในการพิสูจน์ ตัวบุคคลในระบบใหม่ จนกระทั่งเมื่อปี พ.ศ.2503 (ค.ศ.1960) แพร์รี ซูบรอน ตำรวจชาวฝรั่งเศสได้คิดค้นทฤษฎี Minutiae คือ เส้นแยกเส้นแตก เส้นขาดบนลายนิ้วมือโดยกำหนดจุดไหลเวียนของ ลายเส้นไว้พร้อมทั้งความสัมพันธ์ของจุดใกล้เคียง ซึ่งเป็นเอกลักษณ์เฉพาะตัวคนโดยไม่มีกร ซ้ำกับบุคคลอื่น ได้สร้างระบบฐานข้อมูลขึ้นทดลองการใช้ทฤษฎี ต่อมาในปี พ.ศ. 2510 (ค.ศ.1967) หน่วยงานรัฐบาลสหรัฐอเมริกาได้แก่ U.S.National Bureau of Standards ร่วมกับหน่วยงานข่าวกรองแห่งชาติของสหรัฐอเมริกาค่อยๆ พัฒนาทฤษฎีของฝรั่งเศสมาจนเป็นที่รู้จักแพร่หลายใช้กันอยู่เรียกว่า The Minutiae Based AFIS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.3 วิธีการแยกแยะลักษณะของลายนิ้วมือโดยอาศัยทฤษฎี Minutiae

เป็นวิธีการกำหนดจุดตำแหน่งซึ่งเป็นลักษณะโดดเด่นของแต่ละลายนิ้วมือ เนื่องจากในการอ่านเส้นลายนิ้วมือนั้นส่วนใหญ่จะเป็นเส้นเดี่ยว เริ่มที่ด้านหนึ่งของนิ้วมือลากผ่านไปออกที่ปลาย อีกด้านของนิ้วมือแต่จะมีบางลายเส้นที่ไม่ได้เริ่มต้นหรือสิ้นสุดที่ข้างใดข้างหนึ่งของนิ้วมือ หรือเป็น เส้นสั้นๆ หรือไม่ได้เป็นเส้นเดี่ยวตลอดลายกล่าวคือเป็นเส้นแยก เส้นแตก เส้นขาด หรือจุดใจกลาง ของลายเส้นในลายนิ้วมือแต่ละนิ้ว ซึ่งจุดที่กำหนดขึ้นนี้คล้ายจุดลูกน้ำมีหัวและหางแสดงการไหลเวียนของลายเส้นนิ้วมือ เรียกว่า Minutiae และต้องตั้งอยู่บนแกน X (เส้นนอน) และแกน Y (เส้นตั้ง)

ตำแหน่งของจุด Minutiae ของแต่ละนิ้ว เมื่อลากเส้นตรงเข้าหากันจะแสดงถึงความสัมพันธ์ของจุดใกล้เคียงและสามารถวัดได้เป็นองศาของมุมต่างๆ รอบๆ จุดใจกลาง เสมือนหนึ่งสร้างแผนที่ ลายเส้นบนนิ้วมือ ซึ่งลายนิ้วมือของแต่ละนิ้วและของแต่ละคนจะไม่มีซ้ำกันและสามารถกำหนดจุดได้ถึง 150 จุดบนหนึ่งนิ้วมือคน

2.3.4 การใช้เทคโนโลยี Biologon

การดำเนินธุรกิจในปัจจุบัน ได้พัฒนาเข้าสู่ยุคของการทำงานโดยใช้ข้อมูลข่าวสาร (Information Age) ในการตัดสินใจ ข้อมูลทางธุรกิจของแต่ละองค์กรถือเป็นกลยุทธ์ที่มีความสำคัญและมีความหมายในการแข่งขันทางธุรกิจการรักษาข้อมูลหรือความลับขององค์กรจึงเป็นสิ่งที่ไม่ควรละเลยเพื่อป้องกันความเสียหายที่อาจจะเกิดขึ้นได้โดยไม่รู้ตัวดังนั้นระบบการเข้ารหัสผ่าน (Password) ที่ใช้กันอย่างแพร่หลายในปัจจุบันซึ่งสามารถขโมยหรือใช้แทนกันได้ทำให้เกิดปัญหาความลับรั่วไหลหรือการโจรกรรมข้อมูลและความเสียหายจากการใช้งานโดยผู้อื่นที่ไม่มีสิทธิ์ถูกต้องอาจไม่เพียงพอ ต่อการรักษาความปลอดภัยของข้อมูลอันมีค่าขององค์กรได้นอกจากนี้จากการสำรวจความเสียหายจากการโจรกรรมข้อมูลเกิดจากการกระทำของบุคคลภายในองค์กรแทบทั้งสิ้นจึงได้เกิดนวัตกรรมใหม่ ซึ่งใช้เทคโนโลยีชีวภาคซึ่งนำมาใช้งานร่วมกับการควบคุมตรวจสอบการผ่านเข้า - ออกนั้น เป็นการใช้สิ่งที่เป็นเอกลักษณ์เฉพาะของบุคคล เช่น ลายนิ้วมือ ดวงตา เป็นต้น เป็นสิ่งที่ไม่สามารถสร้างใหม่ ลอกเลียนแบบ ทำหายหรือถูกขโมยไปได้ตลอดชีวิตมาทำการยืนยันยืนยันความเป็นบุคคลนั้นๆ สำหรับการผ่านเข้าสู่ระบบซึ่งมีการใช้งานที่ง่ายแต่ให้ความมั่นใจในระดับสูงซึ่งเป็นเทคโนโลยีที่ได้รับการพัฒนาและนำมาใช้งานอย่างแพร่หลายในด้านระบบการผ่านเข้า - ออก เพื่อให้เกิดความปลอดภัยสูงสุด

ดังนั้น Biologon ก็คือ เครื่องอ่านลายนิ้วมือเพื่อการผ่านเข้าสู่การใช้งานในเครื่องหรือในระบบเครือข่ายซึ่งให้ความแม่นยำถูกต้องและมีความน่าเชื่อถือในด้านบุคคลที่ใช้งาน ซึ่งจะเป็นผู้ที่มีสิทธิ์ใช้เท่านั้น เนื่องจาก Biologon จะใช้เทคโนโลยีชีวภาคในการตรวจสอบหรืออนุญาตให้ผ่าน เข้าสู่การใช้งาน บนเครื่องคอมพิวเตอร์โดยเฉพาะ หรือในระบบเครือข่ายของ Biologon เป็นเครื่องตรวจสอบการผ่านเข้าสู่ระบบคอมพิวเตอร์โดยเฉพาะมีความสะดวกในการใช้งาน เพียงแค่สัมผัสปลายนิ้วบนเลนส์เพื่ออ่านลายนิ้วมือ โดยที่ไม่ต้องจำรหัสผ่านซึ่งมีโอกาส ในการถูกนำไปใช้งานโดยบุคคลอื่น หรือจำรหัสผ่านของตนเองไม่ได้ ซึ่งเป็นความยุ่งยากในการใช้งานอีกต่อไป นอกจากนี้ลายนิ้วมือเป็นลักษณะเฉพาะบุคคลที่ไม่มีใครเหมือนและทำหาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือลืมไม่ได้ เป็นสิ่งที่คุณมีอยู่ตลอดเวลา และ คุณสมบัติ พิเศษของ Biologon คือมีความสามารถในการทำงานกับเครื่องคอมพิวเตอร์แบบเครื่องเดียว (Standalone) เป็นการกำหนดสิทธิ์การใช้งานเพื่อป้องกันการลักลอบนำไปใช้งานโดยไม่ได้รับอนุญาตจากเจ้าของเครื่อง โดยผู้ที่เป็นเจ้าของเครื่องนั้นหรือไฟล์นั้นๆ เท่านั้นที่มีสิทธิ์ใช้งานได้ และ Biologon ใช้งานกับเครื่องคอมพิวเตอร์ในระบบเครือข่ายโดยกำหนดสิทธิ์ผู้ใช้ได้หลายระดับ เช่น กำหนดสิทธิ์เป็นกลุ่มผู้ใช้ เฉพาะผู้ที่อยู่ในกลุ่มเดียวกันเท่านั้นที่ใช้งาน ข้อมูล ได้ กำหนดสิทธิ์เป็นรายบุคคลเฉพาะบุคคลนั้นเท่านั้นที่ใช้งานข้อมูลได้รวมไปถึงการกำหนดสิทธิ์ใช้งานภายในข้อมูล เช่น การอ่านข้อมูลสามารถกำหนดให้เปิดเฉพาะเจาะจงเป็นไฟล์ได้ หรือกำหนดให้ใช้งานร่วมกันได้โดยการเปิดอ่านได้อย่างเดียวหรือกำหนดให้อ่านแล้วแก้ไขและบันทึกข้อมูลได้ เป็นต้น ทั้งนี้ขึ้นอยู่กับระดับของความปลอดภัยและสิทธิ์ของผู้ใช้งานที่ต้องการ นอกจากนี้ยังมี คุณสมบัติพิเศษ ในการทำงานจากระยะไกลคือสามารถเข้าในระบบได้โดยที่ไม่ต้องอยู่ในองค์กร ทำให้สะดวกในการทำงานนอกสถานที่โดยมีรูปแบบให้เลือกใช้ได้ตามความพอใจทั้งเครื่องแบบ ตั้งโต๊ะ และคอมพิวเตอร์แบบพกพา

2.3.5 การใช้เทคโนโลยี Automated Fingerprint Identification System (AFIS)

ก่อนที่คอมพิวเตอร์จะเป็นที่แพร่หลายนั้น การตรวจสอบลายนิ้วมือจะทำโดยผู้เชี่ยวชาญเฉพาะ โดยการตรวจสอบทีละ 1 ลายนิ้วมือจนเมื่อปี พ.ศ. 2512 บริษัท NEC ได้พัฒนาระบบ Automated Fingerprint Identification System (AFIS) ซึ่งเป็นระบบที่นำเอาคอมพิวเตอร์มาช่วยในการเก็บข้อมูลลายพิมพ์ลายนิ้วมือที่ได้มาหรือเปรียบเทียบกับข้อมูลที่อยู่ในฐานข้อมูลเดิมมาใช้ในประเทศ ญี่ปุ่น จากนั้นร่วมราว 13 ปีบริษัท NEC ก็ได้ศึกษาและเลียนแบบการตรวจสอบโดยผู้เชี่ยวชาญ และนำเอาหลักการดังกล่าวมาออกแบบ ปรากฏว่า ระบบดังกล่าวสามารถใช้งานได้ดี และมีประสิทธิภาพ ตามที่คาดคิด และหลังจากนั้นบริษัท NEC ได้เริ่มติดตั้งระบบ AFIS ในที่ต่างๆ ทั่วโลก รวมทั้งประเทศไทยได้ไม่นานระบบการตรวจสอบลายพิมพ์นิ้วมือ ก็ได้เข้ามามีบทบาทในทางธุรกิจอย่างมาก โดยนำมาประยุกต์ใช้กับธุรกิจที่ต้องการระดับความปลอดภัยสูง

ระบบ AFIS จะค้นหา จุดสำคัญ บนนิ้วมือและหา ความสัมพันธ์ ระหว่างจุดต่างๆ เหล่านั้นเป็นจุดสำคัญจะหาได้จากตำแหน่งลายเส้นที่เป็นจุดปลาย หรือจุดแยกโดยอาศัยหลักการความสัมพันธ์ (Relation) นั้นเอง

การใช้ AFIS ใช้หลักค้นหาจุดสำคัญ แทนการจำภาพลายนิ้วมือทั้งหมด เนื่องจากการจำภาพลายนิ้วมือทั้งหมดเป็นการจำที่ค่อนข้างยาก และลายนิ้วมือเดียวแต่มาพิมพ์ซ้ำสอง อาจให้ผลไม่เหมือนกัน ดังนั้นการใช้เทคโนโลยีมาพิสูจน์ลายนิ้วมือ ก็จะใช้วิธีหาจุดสำคัญบนลายนิ้ว ซึ่งมี 2 จุดนั่นเอง จุดแรกคือ จุดปลาย เนื่องจากเส้นลายนิ้วมือคนเราจะประกอบด้วยเส้นลายนิ้วมือเต็มไปหมด จะมีบางจุดบางเส้นที่วิ่งมาและหยุดไปเลย นั่นคือจุดที่สำคัญจุดหนึ่ง จุดที่สอง คือจุดแยกหรือ ตัว Y เพราะแค่ 1 เส้นแต่แยกออกเป็น 2 เส้น นอกจากจะหาจุดสำคัญบนลายนิ้วมือแล้วก็จะสร้างความสัมพันธ์ของแต่ละจุดเข้าด้วยกัน

2.3.6 เครื่องตรวจสอบลายนิ้วมือ โดยใช้ทฤษฎีไบโอเมตริก

ในบรรดาเทคโนโลยีไบโอเมตริกที่ใช้กันในปัจจุบัน การตรวจสอบลายนิ้วมือจะเป็นวิธีการที่ถูกเลือกใช้มากที่สุด (และเป็นวิธีการที่ถูกนำมาใช้ในบริษัทยาพันธ์นี้ด้วย) ส่วนหนึ่งก็เพราะ การตรวจสอบลายนิ้วมือเป็นรูปแบบที่ถูกนำมาใช้มาเป็นเวลานาน โดยมีหลักฐานเชื่อมโยงที่แสดงว่า การตรวจสอบลายนิ้วมือถูกเริ่มใช้มาตั้งแต่ในยุคสมัยของชาวจีนโบราณเมื่อหลายพันปีก่อน การใช้งานที่มีมาอย่างยาวนานนี้เอง ทำให้เกิดข้อสรุปที่ชัดเจนว่าลายนิ้วมือเป็นลักษณะเฉพาะที่แตกต่างไปในแต่ละคน และลายนิ้วมือของแต่ละคนก็ไม่มีเปลี่ยนแปลงแม้ว่าวันเวลาจะผ่านไป คุณสมบัติ ทั้ง 2 ข้อนี้เป็นประเด็นสำคัญที่ทำให้ลายนิ้วมือเหมาะสำหรับการเป็นกุญแจผ่านระบบรักษาความปลอดภัยที่ไม่มีใครสามารถลอกเลียนได้นอกจากนั้นในแง่ของความเหมาะสมและความสะดวกในการใช้งาน รวมทั้งราคาของเครื่องตรวจสอบลายนิ้วมือก็มีแนวโน้มลดลงเรื่อยๆ ตามเวลา และการแข่งขันของตลาดก็เป็นอีกเหตุผลที่ทำให้ระบบไบโอเมตริกในรูปแบบนี้มีความโดดเด่นตามไปด้วย

โครงสร้างของระบบตรวจสอบลายนิ้วมือแบบอิเล็กทรอนิกส์ ประกอบด้วยส่วนมีหลักๆ อยู่ 2 ส่วนด้วยกันคือ ฮาร์ดแวร์และซอฟต์แวร์ โดยหน้าที่หลักของฮาร์ดแวร์ก็คือ การจับภาพลายนิ้วมือและแปลงภาพที่ได้ไปอยู่ในรูปข้อมูลแบบดิจิทัลเสียก่อน จากนั้นซอฟต์แวร์ก็จะมารับหน้าที่ต่อใน การตรวจสอบและตีความข้อมูล ตามระดับของการยอมรับที่ได้ถูกกำหนดไว้

ขั้นตอนสำคัญในการเริ่มต้นใช้งานระบบไบโอเมตริกก็คือการลงทะเบียน การลงทะเบียนที่ว่านี้หมายถึงการที่ผู้ใช้แต่ละรายจะต้องส่งตัวอย่างข้อมูลไปเก็บไว้ในฐานข้อมูลกลางของระบบก่อน เพื่อใช้เป็นข้อมูลอ้างอิง หลังจากลงทะเบียนแล้วเมื่อผู้ใช้ระบบในครั้งต่อไป ระบบก็จะ เปรียบเทียบว่าข้อมูลที่ได้อัปกับข้อมูลที่อยู่ในฐานข้อมูลกลางตรงกันหรือไม่ ซึ่งข้อมูลทั้งคู่จะต้องตรงกันเท่านั้น ผู้ใช้จึงผ่านเข้าไปยังระบบได้

2.4 การอ่านลายนิ้วมือด้วยเทคโนโลยีแสง

2.4.1 ที่มาของการอ่านลายนิ้วมือด้วยเทคโนโลยีแสง

ในปัจจุบันนี้วิธีการบันทึกข้อมูลลายนิ้วมือที่ใช้กันอยู่ในประเทศไทยยังเป็นระบบเก่าอยู่ คือ ต้องมีการประทับลายนิ้วมือด้วยหมึก จากนั้นจึงกดนิ้วมือนั้นๆ ลงบนแบบฟอร์มบันทึกใหม่ซึ่งเป็นการสิ้นเปลืองเสียเวลา และมีหมึกเปื้อนที่นิ้วมือนอกจากนี้แล้วถ้าต้องการนำข้อมูลเก็บบันทึกลงในหน่วยความจำคอมพิวเตอร์ จะต้องนำแบบฟอร์มไปอ่านด้วยเครื่องสแกนอีกครั้งหนึ่งซึ่งเป็นการเพิ่ม ขั้นตอนในการทำงานวิธีการดังกล่าวนี้ยังใช้ได้อย่างแพร่หลายในงานต่างๆ เช่น งานระบบโรงรับจำนำหน่วยงานราชการ ศาล งานนิติเวช และ งานข้อมูลอาชญากรรมของสำนักงานตำรวจแห่งชาติ เป็นต้น ดังนั้นวิธีการที่จะลดขั้นตอนต่างๆ เหล่านี้ลงและเพื่อให้การทำงานมีความรวดเร็วในการอ่านและบันทึก ข้อมูลนิ้วมือแบบเก่าจึงถูกแทนที่ด้วยเครื่องอ่านลายนิ้วมือเพื่อช่วยให้การเก็บบันทึกข้อมูลลายนิ้วมือ ทำได้ง่ายและรวดเร็วขึ้น นอกจากนี้ แล้วเครื่องอ่านลายนิ้วมือยังสามารถนำไปประยุกต์ใช้งาน ทางด้านระบบรักษาความปลอดภัยในสถานที่

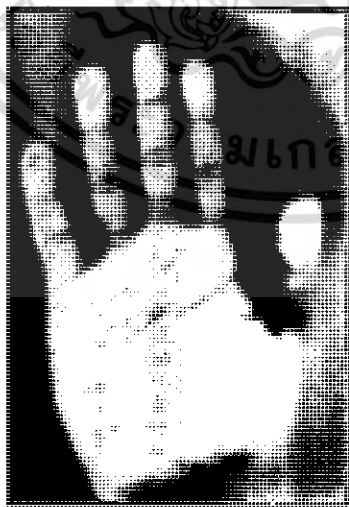
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2 การอ่านลายนิ้วมือโดยใช้เทคโนโลยีแสง

การอ่านลายนิ้วมือโดยใช้เทคนิคทางแสงนั้นสามารถแบ่งออกได้เป็น 3 แบบตามหลักการที่ใช้ คือ แบบที่ใช้หลักการอ่านโดยตรง (Direct Technique) แบบที่ใช้หลักการของการกระจายแสง (Scattering Light Technique) แบบผสมผสานที่อาศัยหลักการของการสะท้อนกลับหมดของแสงและหลักการของแสงที่หายไประยะเร็ว (Total Internal Reflection and Evanescent Wave Technique)

2.4.2.1 แบบที่ใช้หลักการอ่านโดยตรง

เครื่องมือที่ทำได้ง่ายในปัจจุบันที่อาศัยหลักการนี้และสามารถนำมาใช้ในการอ่านลายนิ้วมือได้ คือ เครื่องสแกนเนอร์สำหรับคอมพิวเตอร์ ดังในรูปที่ 2.5 แสดงผลของการอ่านลายเส้นบนมือของผู้เขียนซึ่งจะเห็นว่าข้อมูลที่ได้อาจไม่มีความคมชัด ลักษณะของลายนิ้วมือที่สังเกตจากอีกด้านหนึ่งของกระจก นอกจากนี้แล้ว ข้อเสียอีกอย่างหนึ่งของการใช้เครื่องสแกนเนอร์ก็คือจะสามารถอ่านลายนิ้วมือที่ได้จากรูปถ่ายได้จึงไม่เหมาะในการนำมาใช้กับระบบที่ต้องการความปลอดภัยสูงโดยคุณภาพของลายนิ้วมือจากวิธีการอ่านโดยตรงนี้สามารถปรับปรุงได้โดยใช้อุปกรณ์รับภาพ ที่ถูกออกแบบมาเป็นพิเศษมาอ่านค่าอุณหภูมิจากนิ้วมืออย่างไรก็ตามอุปกรณ์ตัวนี้จะใช้ได้กับนิ้วมือที่มีความกว้าง 1.4 เซนติเมตร เท่านั้น จึงไม่เหมาะกับระบบรักษาความปลอดภัย หรือ งานทะเบียนราษฎรนอกจากนี้แล้วนิ้วมือของผู้ถูกอ่านจะต้องเลื่อนอย่างช้าๆ เพื่อให้อุปกรณ์ที่ทำขึ้นพิเศษนี้สามารถรับข้อมูลเข้ามาได้ อุปกรณ์อีกชนิดหนึ่งที่ทำขึ้นสามารถนำมาอ่านลายนิ้วมือที่ใช้หลักการตรวจวัดค่าเก็บประจุ (Capacitive Sensor) โดยที่ค่าประจุบนพื้นผิวของอุปกรณ์นี้จะเปลี่ยนแปลงสอดคล้องกับการใช้งานกับนิ้วมือที่มีขนาดเล็กอยู่ ข้อดีของอุปกรณ์พิเศษทั้งสองชนิดนี้คือจะอ่านลายนิ้วมือจากนิ้วมือจริงๆ ได้เท่านั้น ดังรูปที่ 2.5

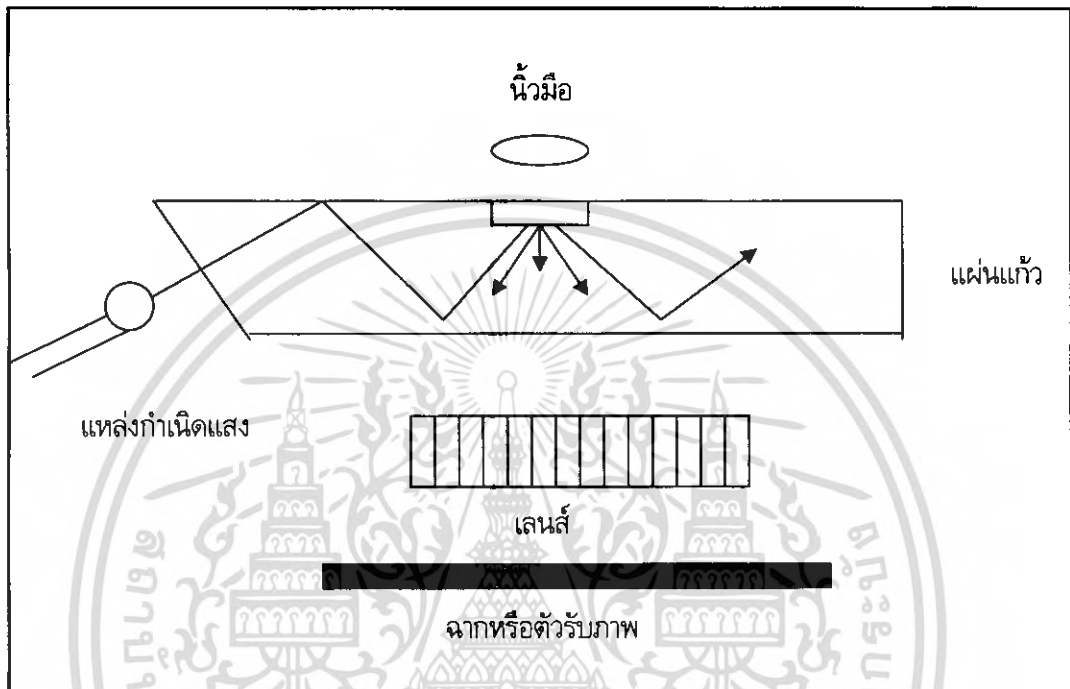


รูปที่ 2.5 ลายมือและรูปขยายในส่วนของนิ้วมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2.2 แบบที่ใช้หลักการกระจายแสง

วิธีการนี้จะเป็นวิธีการตรวจจับแสงที่กระจายไปสู่ตัวรับแสงเนื่องจากร่องรอยบนนิ้วมือ ระบบการอ่านลายนิ้วมือแบบใช้หลักการกระจายแสงแสดงได้ดังรูปที่ 2.6



รูปที่ 2.6 ระบบการอ่านลายนิ้วมือแบบใช้หลักการกระจายของแสง

ในที่นี้แสงจากแหล่งกำเนิดแสง ซึ่งอาจจะเป็นแสงเลเซอร์หรือไดโอดเปล่งแสง (Light Emitting Diode LED) ก็ได้ จะถูกส่งเข้าไปยังแผ่นแก้วโดยการสะท้อนกลับไปกลับมาภายใน แผ่นแก้ว เมื่อมีนิ้วมือมาสัมผัสอยู่บนแผ่นแก้วจะทำให้แสงบางส่วนที่สะท้อนที่ผิวสัมผัสกระจายไปสู่ตัวรับภาพที่วางอยู่ใต้แผ่นแก้ว เลนส์ที่ใช้จะทำหน้าที่ส่งภาพจากพื้นผิวของแผ่นแก้วที่มีนิ้วมือสัมผัสอยู่ไปยังตัวรับภาพด้วยกำลังขยายที่กำหนด

รูปที่ 2.7 แสดงผลการอ่านลายนิ้วมือที่ได้ซึ่งจะเห็นว่า คุณภาพความคมชัดของลายนิ้วมือดีกว่าแบบอ่านโดยตรงที่ใช้เครื่องสแกนเนอร์ ข้อดีอีกอย่างหนึ่งก็คือ เราสามารถใช้แผ่นแก้วที่มีขนาดใหญ่ได้ทำให้วิธีการนี้เหมาะสมกับการอ่านลายนิ้วมือที่มีขนาดใหญ่ นอกจากนี้แล้ววิธีการนี้จะอ่านลายนิ้วมือจากนิ้วจริงๆ เท่านั้น จะไม่สามารถใช้อ่านลายนิ้วมือจากรูปถ่ายได้ ดังนั้นจึงเหมาะกับระบบรักษาความปลอดภัยด้วย



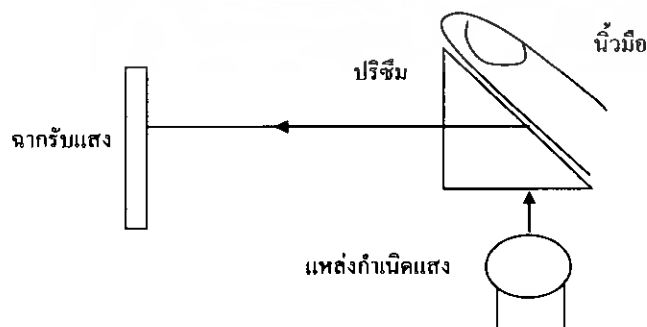
รูปที่ 2.7 ผลการอ่านลายนิ้วมือที่ใช้หลักการของการกระจายแสง

2.4.2.3 แบบผสมผสานที่อาศัยหลักการสะท้อนกลับหมดของแสงและหลักการ ของแสงที่หายไป

อย่างรวดเร็ว

รูปการจัดตั้งอุปกรณ์สำหรับอ่านลายนิ้วมือที่ใช้หลักการสะท้อนกลับหมดของแสงและ หลักการของแสงที่หายไปอย่างรวดเร็วแสดงได้ดังรูปที่ 2.8 อุปกรณ์ที่จำเป็นต้องใช้มีเพียงแคंपริซึม แหล่งกำเนิดแสงซึ่งอาจเป็นเลเซอร์หรือหลอดไดโอดเปล่งแสงก็ได้ และตัวรับภาพซึ่งอาจเป็นฉากรับภาพ ก็ได้ นอกจากนี้จะเห็นว่าการจัดตั้งอุปกรณ์และจำนวนอุปกรณ์ที่ใช้จะง่ายกว่าการสร้างเครื่อง อ่านลายนิ้วมือแบบใช้หลักการกระจายของแสง

จากรูปที่ 2.8 ในเวลาปกติเมื่อไม่มีนิ้วมือวางอยู่บนปริซึมแสงจากแหล่งกำเนิดแสงจะ ตกกระทบผิวปริซึมทางด้านที่วางนิ้วมือซึ่งจะทำให้แสงตกกระทบทั้งหมดสะท้อนไปยังตัวรับภาพ ในทางกลับกันเมื่อนิ้วมือวางอยู่บนปริซึมแสงบางส่วนจะเล็ดลอดออกไปทางด้านผิวของปริซึมที่มีนิ้วมือสัมผัสอยู่ ปริมาณของแสงที่เล็ดลอดออกไปจะขึ้นอยู่กับแรงกด และความลึก ของส่วนต่างๆ บนนิ้วมือ ดังนั้นภาพที่รับได้จะแสดงลักษณะของลายนิ้วมือที่สัมผัสอยู่บนพื้นผิวปริซึมข้อมูลที่ได้จากตัวรับภาพนี้สามารถถูกส่งไปบันทึกลงบนคอมพิวเตอร์เพื่อการประมวลผลต่อไปได้



รูปที่ 2.8 การจัดตั้งอุปกรณ์สำหรับการอ่านลายนิ้วมือที่ใช้หลักการสะท้อนกลับหมดของแสง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



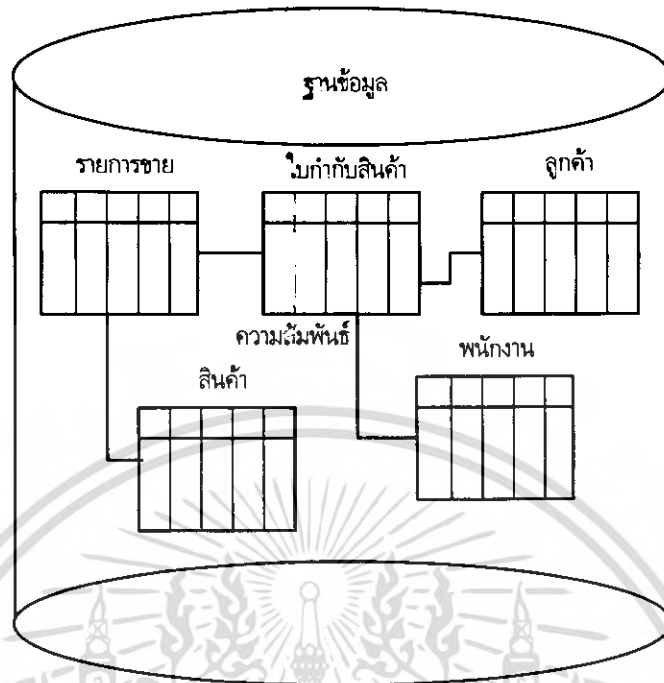
รูปที่ 2.9 ผลการอ่านลายนิ้วมือที่ใช้หลักการสะท้อนกลับหมดของแสงและหลักการของแสงที่หายไปอย่างรวดเร็ว

ผลของลายนิ้วมือที่อ่านได้จากวิธีการนี้จะมีความชัดเจนมากดังแสดงได้ดังรูปที่ 2.9 ข้อดี อีกอย่างหนึ่งก็คือวิธีการนี้สามารถใช้กับนิ้วมือที่มีขนาดใหญ่ได้เพียงแค่ออกใช้ขนาดของปริซึม ที่เหมาะสมเท่านั้น ความละเอียดของลายนิ้วมือที่อ่านได้ขึ้นอยู่กับประสิทธิภาพของตัวรับภาพ ข้อมูล ที่ได้จากตัวรับภาพเหล่านั้นสามารถเชื่อมต่อเข้ากับคอมพิวเตอร์เพื่อการเก็บบันทึกและการประมวลผลนอกจากนี้แล้ววิธีการนี้ไม่สามารถใช้อ่านลายนิ้วมือ จากรูปถ่ายได้จึงเหมาะอย่างยิ่งกับระบบรักษาความปลอดภัย

จะเห็นว่าการนำเทคโนโลยีแสง มาใช้ในการอ่านลายนิ้วมือนั้นไม่ใช่เรื่องยากและซับซ้อนอะไร คุณภาพของลายนิ้วมือที่อ่านได้จะขึ้นอยู่กับคุณภาพของแสงจากแหล่งกำเนิดแสง พื้นผิวที่นิ้วมือ สัมผัส และประสิทธิภาพของตัวรับภาพ ดังนั้นจึงควรที่จะศึกษาและสร้างเครื่องอ่านลายนิ้วมือเชิงแสงขึ้นใช้เองเพื่อลดการนำเข้า นอกจากนี้ยังเป็นการส่งเสริมการวิจัยและพัฒนาทางด้านเทคโนโลยีแสง ในประเทศ และเป็นการพัฒนาบุคลากรภายในประเทศให้มีความรู้ความเข้าใจในสาขาเทคโนโลยีนี้มากขึ้น

2.5 ความรู้พื้นฐานของระบบฐานข้อมูล

ฐานข้อมูล (Database) คือที่อยู่ของข้อมูลที่มีความสัมพันธ์กัน หรืออาจจะเปรียบเทียบเป็นคลังของข้อมูลก็ได้ ข้อมูลเหล่านี้จะถูกจัดเก็บร่วมกันอย่างมีระบบและรูปแบบ ทำให้ง่ายต่อการประมวลผลและการจัดการฐานข้อมูลที่มีอยู่ซึ่งเรียกว่า Database Management System หรือ DBMS สำหรับฐานข้อมูลที่มีความนิยมมากที่สุดในปัจจุบันจะเป็นแบบ Relational Database ซึ่งจะจัดเก็บข้อมูลอยู่ในรูปของตาราง (Table) โดยที่ข้อมูลในแต่ละตารางจะมีความสัมพันธ์กันและกัน ดังรูปที่ 2.10

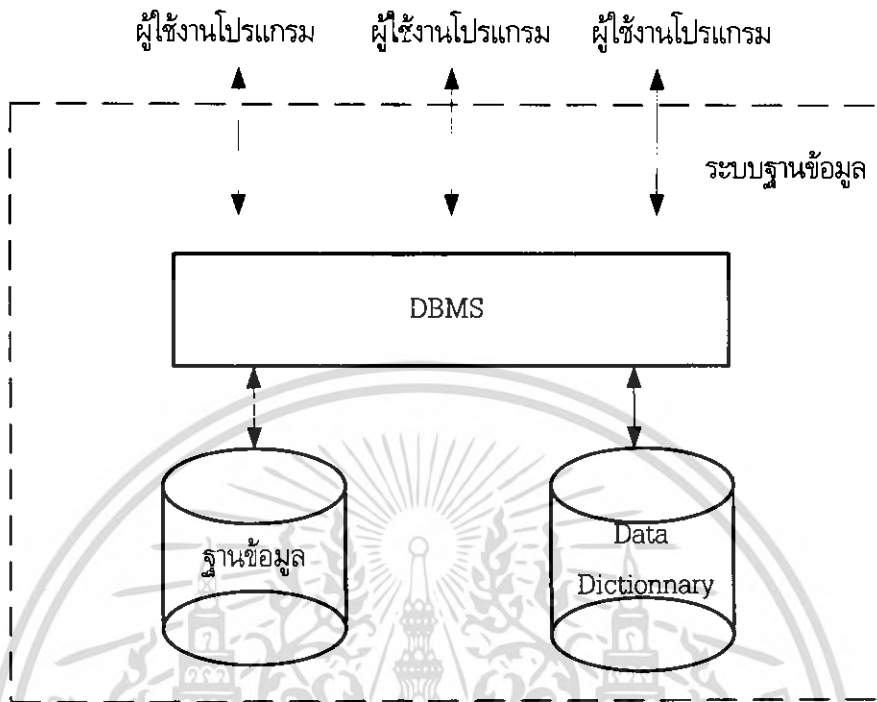


รูปที่ 2.10 ตัวอย่างฐานข้อมูล

2.6 องค์ประกอบของระบบฐานข้อมูล

ระบบฐานข้อมูล (Database System) จะประกอบไปด้วยฐานข้อมูล (Database) ระบบจัดการฐานข้อมูล (Database Management System หรือ DBMS) และ Data Dictionary ดังในรูปที่ 2.11 โดยที่ฐานข้อมูลจะเป็นที่จัดเก็บข้อมูลต่างๆ ที่เกี่ยวข้องไว้ด้วยกัน มี DBMS ทำหน้าที่จัดการกับฐานข้อมูลดังกล่าว และโครงสร้างของฐานข้อมูลจะถูกจัดเก็บไว้ใน Data Dictionary

ถึงตอนนี้หลายคนอาจจะสับสนระหว่าง "ฐานข้อมูล" และ "ระบบฐานข้อมูล" ว่ามีความแตกต่างกันหรือไม่อย่างไร สำหรับฐานข้อมูลจะเป็นเพียงส่วนหนึ่งของระบบฐานข้อมูลเท่านั้น ดังนั้นหากพูดถึงระบบฐานข้อมูลจะต้องประกอบไปด้วย 3 ส่วน คือ ฐานข้อมูล DBMS และ Data Dictionary ดังในรูปที่ 2.11 สำหรับฐานข้อมูลนั้นจะประกอบไปด้วย ตารางและความสัมพันธ์ระหว่างตาราง ดังในรูปที่ 2.10 และในส่วนหนึ่งของระบบฐานข้อมูล



รูปที่ 2.11 องค์ประกอบของระบบฐานข้อมูล

สำหรับ DBMS นับว่าเป็นส่วนสำคัญในระบบฐานข้อมูลเป็นอย่างยิ่ง เปรียบเสมือนผู้จัดการฐานข้อมูลทำหน้าที่เป็นตัวกลางระหว่างผู้ใช้กับฐานข้อมูลโดยที่ DBMS จะรับคำสั่งจากผู้ใช้งานหรือจากโปรแกรมต่างๆ หลังจากนั้นจะทำการประมวลผลกับฐานข้อมูลโดยอาศัยโครงสร้างที่จัดเก็บไว้ใน Data Dictionary (โครงสร้างของฐานข้อมูลเหล่านี้จะเรียกว่า Meta Data) และทำหน้าที่ส่งผลลัพธ์ที่ได้กลับคืนไปยังผู้ใช้งาน หรือโปรแกรมโดยที่ผู้ใช้งานไม่จำเป็นต้องรู้เลยว่า DBMS จัดเก็บข้อมูลอย่างไรมีกลไกในการเข้าถึงหรือค้นหาข้อมูลอย่างไร ขอเพียงรู้คำสั่งที่ต้องการสั่งงานเพื่อให้ได้ผลลัพธ์ที่ต้องการเท่านั้น ที่เหลือจะเป็นหน้าที่ของ DBMS ในการดึงข้อมูลหรือการประมวลผลต่างๆ ดังนั้น สำหรับผู้ใช้งานจะรู้สึกว่าการใช้งาน DBMS ทำได้อย่างง่ายดาย เพราะ DBMS จะซ่อนความยุ่งยากในการเข้าถึงข้อมูลได้เอง สำหรับ DBMS ที่ได้รับความนิยมสูงสุดในปัจจุบันจะเรียกว่า RDBMS (Relational DBMS) ซึ่ง RDBMS นี้จะมีให้เลือกใช้งานมากมายทั้งแบบใช้งานคนเดียวหรือหลายคนพร้อมๆ กัน เช่น MS-Access, FoxPro, Paradox เป็นต้น จนถึงในระดับ Server ที่เรียกว่า Database Server เช่น SQL Server, Oracle, Informix, Sybase เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 การออกแบบฐานข้อมูล

การออกแบบระบบฐานข้อมูลสามารถแบ่งออกได้เป็น 2 ระดับ ระดับแรกเรียกว่า การออกแบบระดับสารสนเทศ (Information-Level Design) คือส่วนของการศึกษาวิเคราะห์รวบรวมความต้องการของผู้ใช้เอาไว้โดยที่การออกแบบในระดับนี้มีเป้าหมาย เพื่อให้การใช้งานเป็นไปตามความต้องการของผู้ใช้มากที่สุด โดยในจุดนี้เรายังจะไม่ให้ความสำคัญกับชนิดและยี่ห้อของ DBMS ที่จะใช้โดยเราจะเริ่มศึกษาเกี่ยวกับ DBMS ที่จะใช้ในครั้งที่ 2 ซึ่งเรียกว่าการออกแบบระดับกายภาพ (Physical-Level Design) อันเป็นระดับที่เราเริ่มให้ความสำคัญต่อประสิทธิภาพของระบบ ซึ่งแน่นอนว่าผู้ที่ทำหน้าที่ดูแลการออกแบบควรจะได้ศึกษาความต้องการ และเป้าหมายการใช้งานจากกลุ่มผู้ใช้ให้ถ่องแท้เสียก่อนว่าต้องการเน้นประสิทธิภาพในงานไหน

การออกแบบใน 2 ระดับนี้ก็ต้อยอมรับว่ามันมีความสำคัญเท่าเทียมกันเพราะการออกแบบในระดับข่าวสารที่ไม่ดีย่อมจะมีผลกระทบต่อประสิทธิภาพของระบบด้วย

2.7.1 เป้าหมายของการออกแบบข้อมูล

การออกแบบฐานข้อมูลมีเป้าหมายที่ต้องการสร้างประสิทธิภาพของการใช้งานให้แก่ผู้ใช้ ผู้ออกแบบก็จะต้องรวบรวมเอกภพเกณฑ์ และข้อบังคับต่างๆ เอาไว้ด้วย ซึ่งเราสามารถสรุปรายละเอียดต่างๆ ที่จำเป็นต้องทราบในส่วนนี้ได้ดังนี้

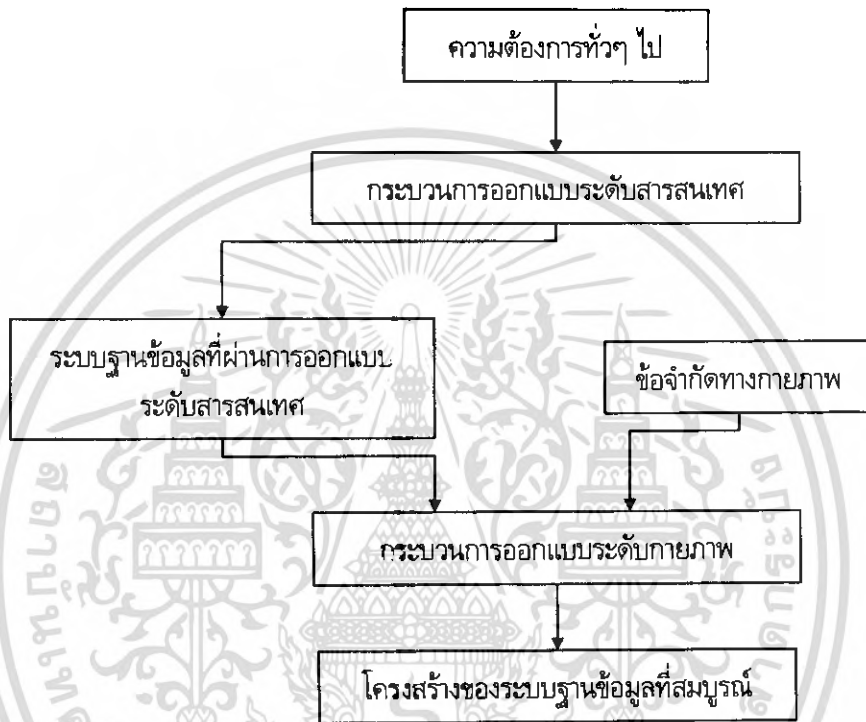
1. ลักษณะของรายงานทั้งหมด
2. การค้นหาข้อมูลในทุกลักษณะ
3. เอาต์พุตที่ต้องการส่งไปให้แผนกอื่น หรือระบบอื่น
4. การประมวลและแก้ไขข้อมูลทั้งหมด
5. การคำนวณทุกอย่าง
6. กฎเกณฑ์ข้อบังคับต่างๆ (เช่น ห้ามไม่ให้ลูกค้าคนใดใช้บริการของพนักงานขายที่ไม่มีหมายเลขปรากฏอยู่ในตารางของพนักงานขาย)
7. การตั้งชื่อพ้อง (Synonym) ต่างๆ (เช่น ในแต่ละแผนกหรือผู้ใช้แต่ละคนอาจจะเรียกชื่อของ Attribute ตัวเดียวกันแตกต่างกันไป เช่น Attribute ที่แสดงค่ารหัสสินค้าอาจจะถูกเรียกว่ารหัสชิ้นส่วนในแผนกหนึ่ง ในขณะที่อีกแผนกหนึ่งเรียกว่าเบอร์สินค้า เป็นต้น)

ข้อมูลเหล่านี้ จะถูกนำมาประมวลในการออกแบบฐานข้อมูลในระดับสารสนเทศ และเมื่อถึงเวลาของการออกแบบในระดับกายภาพแล้ว เรายังต้องอาศัยข้อมูลอื่นๆ ที่เกี่ยวข้องกับการทำงานของระบบในระดับกายภาพอีก เช่น

1. จำนวนของแต่ละ Entity (เช่น การประมาณการว่ามีลูกค้ากี่คนมีพนักงานขายกี่คน เป็นต้น)
2. ความถี่ในการพิมพ์รายงาน
3. ความยาวของรายงานแต่ละชิ้น
4. กฎเกณฑ์ในการควบคุมความปลอดภัยในการใช้ข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลต่างๆ เหล่านี้จะถูกนำมาวิเคราะห์พร้อมกับระบบฐานข้อมูลที่ออกแบบในระดับสารสนเทศไว้แล้ว เพื่อการออกแบบในระดับกายภาพ โดยในระดับนี้เราจะเริ่มคำนึงถึงความสามารถของ DBMS ที่กำลังใช้อยู่ด้วยแล้ว เพื่อให้ผลการทำงานของระบบที่ออกแบบมานี้สมบูรณ์แบบ และมีประสิทธิภาพสูงสุด โดยสรุปแล้วขั้นตอนของการออกแบบระบบฐานข้อมูลก็จะเป็นไปตามที่ปรากฏในรูปที่ 2.12



รูปที่ 2.12 ขั้นตอนการออกแบบฐานข้อมูล

2.7.2 พื้นฐานหลักการของการออกแบบฐานข้อมูล

ความต้องการของกฎข้อบังคับต่างๆ ซึ่งผู้ใช้ระบบจะเริ่มกระบวนการออกแบบได้นั้นต้องมี 5 ขั้นตอนดังนี้

1. เปลี่ยนรูปแบบของความต้องการให้อยู่ในรูปลักษณะของรีเลชัน
2. นอร์มัลไลซ์รีเลชัน
3. การกำหนดฟิลด์ที่จะเป็นคีย์ต่างๆ และคุณสมบัติของคีย์แต่ละตัว
4. พิจารณาข้อจำกัดและกฎเกณฑ์อื่นๆ
5. นำผลที่ได้จากการออกแบบในขั้นตอนแรกมาผนวกกัน

ขั้นตอนที่ 1 การเปลี่ยนรูปแบบของความต้องการให้อยู่ในรูปลักษณะของรีเลชัน

การสร้างรีเลชัน คือ การวิเคราะห์ว่าฐานข้อมูลควรมีรีเลชันอะไรบ้างและในแต่ละตัวนั้นควรมีฟิลด์อะไรบ้างเป็นคีย์บางครั้งที่การออกแบบในส่วนนี้ออกจะตรงไปตรงมา และง่ายจนน่าตกใจ เช่นถ้าเราต้องการเก็บข้อมูลเกี่ยวกับพนักงาน และแผนกที่พนักงานเหล่านี้สังกัดอยู่ ก็ให้เห็นได้อย่างชัดเจนว่า ฐานข้อมูลควรจะประกอบไปด้วยรีเลชัน 2 ตัว คือ แผนกและพนักงาน

ในกรณีที่ความต้องการของผู้ใช้ไม่สามารถวิเคราะห์ได้ง่าย เราก็นำหลักทฤษฎีมาช่วยในการปฏิบัติ ดังนี้

1. สร้างรีเลชันขึ้นมาใช้สำหรับ Entity แต่ละตัว เช่น เมื่อวิเคราะห์ความต้องการของผู้ใช้แล้ว ได้ผลว่าจะใช้เฉพาะข้อมูลของแผนก และพนักงานเท่านั้นเราก็สร้างรีเลชันขึ้นมา 2 ตัว คือ รีเลชันแผนกและรีเลชันพนักงาน แต่ถ้าหากผู้ใช้เกิดระบุว่าจะต้องมีการใช้ข้อมูลเกี่ยวกับลูกค้า ผู้ขายสินค้า และการสั่งซื้อสินค้าด้วยแล้ว เราก็จำเป็นต้องเพิ่มรีเลชันเข้าไปอีก 3 ตัว
2. พิจารณาวรีเลชันแต่ละตัวควรมีฟิลด์ใดเป็นคีย์หลักถึงแม้ว่าในจุดนี้ เรายังไม่ได้ตัดสินใจเลยว่าในแต่ละรีเลชันควรมี Attribute อะไรบ้าง แต่เราก็สามารถพิจารณาถึงคีย์หลักได้ก่อนแล้วตัวอย่างเช่น คีย์หลักสำหรับรีเลชันพนักงานก็ควรจะได้แก่ รหัสพนักงาน ในขณะที่คีย์หลักสำหรับลูกค้าควรจะเป็นรหัสลูกค้า เป็นต้น
3. พิจารณาสมบัติของ Entity แต่ละตัว ในจุดที่เราจะศึกษาดูจากความต้องการ ของผู้ใช้ คุณสมบัติของ Entity แต่ละตัวควรจะประกอบด้วยฟิลด์อะไรบ้างนั่นเอง ดังเช่นตัวอย่างพนักงานและลูกค้า ในแง่ของพนักงานควรจะใช้เก็บข้อมูลเกี่ยวกับ ชื่อ ที่อยู่ เงินเดือน เป็นต้น ส่วนลูกค้าก็ควรมีข้อมูลเกี่ยวกับ ชื่อ ที่อยู่ เงินเดือน เป็นต้น ส่วนลูกค้าก็ควรมีข้อมูลเกี่ยวกับ ชื่อ ที่อยู่ ยอดเงินค้างชำระ และวงเงินเครดิต เป็นต้น ข้อมูลที่วิเคราะห์ได้จากข้อ 2 จะทำให้เราได้ Attribute ของแต่ละ Entity
4. พิจารณาความสัมพันธ์ระหว่างแต่ละ Entity ในส่วนนี้ผู้ออกแบบจำเป็นต้องพิจารณาว่า แต่ละ Entity มีความสัมพันธ์กันในแบบใด ซึ่งเราสามารถแบ่งชนิดของความสัมพันธ์ระหว่างแต่ละ Entity ได้ออกเป็นสามชนิดคือ แบบหนึ่งต่อหนึ่ง แบบหนึ่งต่อกลุ่ม และแบบกลุ่มต่อกลุ่ม

ขั้นตอนที่ 2 นอร์มัลไลซ์รีเลชัน

ในตอนนี้อาจจะนอร์มัลไลซ์รีเลชันแต่ละตัว โดยมีเป้าหมายให้เป็น 3NF (อันที่จริงแล้วเราต้องการ 4NF ซึ่งถ้าผู้ออกแบบตั้งใจนอร์มัลไลซ์ดีๆ ก็ย่อมจะอยู่ใน 4NF โดยปริยาย)

ขั้นตอนที่ 3 กำหนดฟิลด์ที่จะเป็นคีย์ต่าง* และคุณสมบัติของคีย์แต่ละตัว

การกำหนดคีย์ทั้งหมดอันได้แก่ คีย์หลัก คีย์คู่แข่ง คีย์รองบ และคีย์นอก ซึ่งการกำหนดว่าจะให้ข้อมูลในฟิลด์ใดเป็นคีย์ต่างๆ นั้นก็จำเป็นที่ผู้ออกแบบจะต้องเข้าใจการใช้งานของข้อมูลนั้นๆ นั้นก็จำเป็นที่

ผู้ออกแบบจะต้องเข้าใจการใช้งานของข้อมูลนั้นๆ รวมถึงความสัมพันธ์ระหว่าง Entity แต่ละตัว โดยเฉพาะอย่างยิ่งคีย์นอก อันเป็นสิ่งที่เราจะกล่าวถึงแต่ละตัว โดยเฉพาะอย่างยิ่งคีย์นอก อันเป็นสิ่งที่เราจะกล่าวถึงเป็นพิเศษในที่นี้ เพราะผู้ออกแบบจะต้องใช้วิจารณ์แผน และเหตุผลในการตัดสินใจถึงการออกแบบจุดต่างๆ ซึ่งจะเป็นตัวตัดสินถึงความสัมพันธ์ที่จะเกิดขึ้นระหว่าง Entity ในฐานข้อมูล การออกแบบจะต้องคำนึงถึงต่างๆ ดังต่อไปนี้ด้วย

1. จะยอมให้ใช้ค่าว่างได้หรือไม่ ความหมายของคำถามก็คือเราจะยอมให้ระบบเก็บข้อมูลของพนักงานที่ยังไม่ได้สังกัดแผนกใดเลยได้หรือไม่ โดยค่าของรหัสแผนกของพนักงานคนนั้นก็จะมีค่าว่าง สำหรับในการออกแบบในตัวอย่างลักษณะนี้มักจะไม่ใช่คีย์นอกเป็นค่าว่าง เพราะเวลาพนักงานเข้าทำงานใหม่ๆ ก็มักจะต้องมีการกำหนดอยู่แล้วว่าจะให้สังกัดแผนกใด
2. กฎเกณฑ์ของการเปลี่ยนค่ารหัสแผนกในรีเลชันเป็นอย่างไร ในกรณีนี้ถ้าหากว่าแผนกนั้นยังไม่มีพนักงานคนใดสังกัดอยู่เลย ก็ย่อมจะไม่เกิดปัญหาใดๆ แต่ถ้าหากว่าเป็นแผนกที่มีพนักงานสังกัดอยู่แล้ว ก็จำที่ผู้ออกแบบจะต้องตัดสินใจว่าจะให้การเปลี่ยนแปลงนี้เข้าอยู่ในข่ายของการกระทำเป็นทอดๆ ยอมแบบมีเงื่อนไขหรือการเปลี่ยนค่าว่าง
3. กฎเกณฑ์ของการลบแผนก เช่นเดียวกับการตัดสินใจในข้อ 2 กล่าวคือ ปัญหาของการลบแผนกใดแผนกหนึ่งออกไปจะเกิดขึ้นเมื่อมีพนักงานสังกัดอยู่ในแผนกนั้น ซึ่งผู้ออกแบบก็ควรจะตัดสินใจว่าจะออกแบบใช้การลบเป็นไปในรูปแบบของการเป็นขั้นตอน ยอมมีเงื่อนไขหรือการเปลี่ยนเป็นค่าว่าง

ขั้นตอนที่ 4 พิจารณาข้อจำกัดและกฎเกณฑ์อื่นๆ

ในขั้นตอนนี้ ผู้ออกแบบจำเป็นต้องรวบรวมความต้องการจากผู้ใช้งานที่มีข้อจำกัดอะไรบ้าง เช่น อนุญาตให้ลูกค้าคนใดมียอดเงินค้างชำระสูงกว่าวงเงินเครดิต เป็นต้น ในลักษณะเช่นนี้ DBMS บางตัวจะอนุญาตให้เราระบุข้อจำกัดนี้ในส่วนของ การสร้าง Schema ได้เลย แต่สำหรับ DBMS ที่ไม่มีขีดความสามารถเช่น ก็จำเป็นที่ผู้ออกแบบจะต้องรายงานข้อกำหนดเหล่านี้แก่โปรแกรมเมอร์เพื่อที่จะได้สอดใส่ข้อกำหนดนี้ไว้ในโปรแกรมที่เกี่ยวข้องกับการประมวลข้อมูลนั้นๆ

ขั้นตอนที่ 5 นำผลที่ได้จากการออกแบบใน 4 ขั้นตอนแรกมาผนวกกัน

หลังจากที่เราได้ผลของการออกแบบทั้ง 4 ขั้นตอนสำหรับผู้ใช้งานหนึ่งแล้วเราจะนำการออกแบบนี้ไม่ผนวกกับการออกแบบที่สร้างขึ้นสำหรับผู้ใช้งานอื่นๆ ผนวกก็ได้แก่การเพิ่มรีเลชันที่เราออกแบบสำหรับผู้ใช้งานนี้เข้าไปในระบบนั่นเองโดยยึดหลักที่ว่า ถ้ามีรีเลชันที่มีอยู่ก่อนแล้ว กล่าวคือ มีคีย์หลักตัวเดียวกันเราก็จะเพียงแต่เติม Attribute ที่ยังไม่มีรีเลชันเดิมเข้าไปเท่านั้น และเมื่อทำเสร็จเราก็ต้องตรวจสอบอีกครั้งให้การออกแบบยังคงเป็น 3NF ส่วนกฎเกณฑ์อื่นๆ ที่ใส่ไว้ใน Schema ไม่ได้จะถูกรวบรวมไว้เช่นเดียวกัน เพื่อส่งต่อไปโปรแกรมเมอร์รับผิดชอบ การออกแบบในระดับสารสนเทศจะเสร็จสมบูรณ์ก็ต่อเมื่อผู้ออกแบบได้กระทำขั้นตอนทั้ง 5 นี้ กับความต้องการของผู้ใช้แต่ละคนจนครบทุกคน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8 การเขียน Visual Basic ติดต่อ Microsoft Access

2.8.1 Open Database Connectivity (ODBC)

ODBC หรือ Open Database Connectivity เป็นมาตรฐานหลักที่ใช้ในการเชื่อมต่อกับฐานข้อมูลชนิดต่างๆ ซึ่งเป็นประโยชน์สำหรับนักพัฒนาโปรแกรมที่ต้องการสร้างโปรแกรมรองรับฐานข้อมูลชนิดต่างๆ โดยการเขียนโปรแกรมเพียงครั้งเดียว

สำหรับหลักการทำงานกับ ODBC นั้น เราต้องเริ่มโดยการเขียนโปรแกรมเพื่อติดต่อกับ ODBC ให้ได้เสียก่อน จากนั้นส่วนของการจัดการฐานข้อมูลจะเป็นหน้าที่ของ ODBC

จากการทำงานในลักษณะนี้เราจะเห็นได้ว่าการเปลี่ยนชนิดของฐานข้อมูลที่ใช้ จะไม่มีผลต่อการทำงานของโปรแกรมเพราะว่าโปรแกรมที่เราเขียนนั้นติดต่อกับ ODBC ไม่ได้ติดต่อกับฐานข้อมูลโดยตรง ดังนั้นเมื่อไรที่มีการเปลี่ยนแปลงฐานข้อมูล เราก็มักจำเป็นต้องแก้ไขโปรแกรมที่เราเขียน เพียงแต่แก้ไขที่ตัว ODBC เท่านั้น

ในส่วนของโปรแกรมและ Data Source จะเก็บในเครื่องไคลเอ็นท์ ส่วนฐานข้อมูลจะอยู่ที่เครื่องเซิร์ฟเวอร์ ถ้าเราต้องการเปลี่ยนเซิร์ฟเวอร์ฐานข้อมูลเราก็ต้องแก้ไขที่ Data Source ที่เครื่องไคลเอ็นท์ทุกเครื่อง แต่ไม่ต้องแก้ไขในส่วนของโปรแกรม VB นี้คือหลักการทำงานเบื้องต้นของการใช้ Data Source ติดต่อฐานข้อมูลใดๆ

2.8.2 การติดตั้ง ODBC Data Source

ในการสร้าง Data Source ติดต่อกับฐานข้อมูล Microsoft Access เราต้องมีไดรเวอร์ Microsoft Access Driver ด้วย ซึ่งไดรเวอร์ตัวนี้จะมีมาเมื่อตอนติดตั้ง VB แต่ถ้าไม่มีก็แสดงว่าตอนติดตั้งเราไม่เลือกติดตั้งไดรเวอร์ตัวนี้ให้ทำการติดตั้งไดรเวอร์นี้เพิ่มก่อนดำเนินขั้นตอนต่อไป

1. ให้เราเปิดไอคอน 32-bit ODBC ขึ้นมาจากใน Control Panel จากนั้นให้เราเลือกแท็บ System DSN เพื่อสร้าง Data Source ขึ้นมาใหม่ โดยให้เราคลิกเมาส์ปุ่ม Add
2. จะปรากฏไดอะล็อกซ์ Create a New Data Source ให้เราคลิก Microsoft Access Driver เสร็จแล้วให้เราคลิกเมาส์ปุ่ม Finish
3. จะปรากฏไดอะล็อกซ์ Create a New Data Source to Microsoft Access ให้เราเลือกชื่อ Data Source เซิร์ฟเวอร์ที่จะติดต่อกับฐานข้อมูลด้วย เสร็จแล้วให้เราคลิกเมาส์ปุ่ม NEXT>
4. จะปรากฏไดอะล็อกซ์ให้เรากำหนดชื่อ และรหัสผ่านที่จะเข้าใช้ฐานข้อมูลใน Microsoft Access เสร็จแล้วให้เราคลิกเมาส์ปุ่ม NEXT>
5. จะปรากฏไดอะล็อกซ์ให้เรากำหนดชื่อฐานข้อมูลที่จะติดต่อกับ Orders ซึ่งเราสร้างไว้แล้วเสร็จแล้วให้เราคลิกปุ่ม NEXT>
6. จะปรากฏไดอะล็อกซ์ให้เรากำหนดรายละเอียดเกี่ยวกับ Log File ที่ใช้ เสร็จแล้วให้เราคลิกเมาส์ปุ่ม Finish

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. จะปรากฏไดอะล็อกซ์แสดงรายละเอียดต่างๆ ที่เราได้กำหนดกับ Data Source เสร็จแล้วให้เราคลิกปุ่ม OK

2.9 หลักการใช้งานเครื่องสแกนลายนิ้วมือ

เครื่องสแกนลายนิ้วมือรุ่น DigitalPersona U.are.U 4000 จะมี library ไว้สำหรับประมวลผลลายนิ้วมือจะประกอบด้วย 2 ส่วนคือ ส่วนการประมวลผลลายนิ้วมือ (TMWIDW.DLL) และส่วนช่วยจัดเก็บข้อมูลลายนิ้วมือ (FPENROL.DLL) จะทำการรับลายนิ้วมือแล้วทำการประมวลผลและบอกค่าให้โดยอัตโนมัติผ่านทาง callback function สามารถใช้ได้กับภาษาที่สนับสนุนการเรียก DLL เช่น C, Delphi, VB ฯลฯ

2.9.1 การเตรียมขั้นตอนการใช้งานเครื่องสแกน

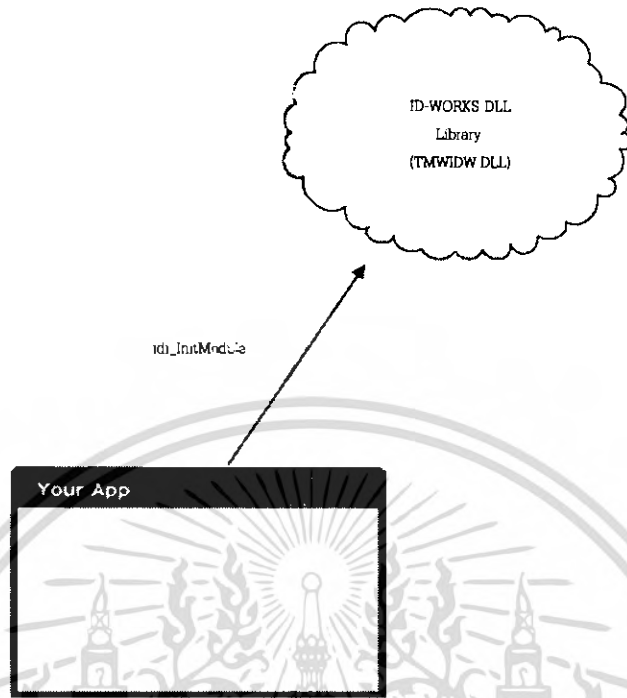
1. ติดตั้ง DigitalPersona Driver ลงในเครื่องก่อน
2. เมื่อติดตั้งแล้วให้เสียบเครื่องอ่านลายนิ้วมือเข้ากับเครื่อง
3. ไฟล์ DLL ที่จะใช้งานต้องอยู่ใน folder เดียวกับไฟล์ .exe ของโปรแกรมที่พัฒนาหรืออยู่ใน system folder ของ windows (เช่น c:\windows\system32)
4. ศึกษาเรื่องการใช้งาน DLL ในภาษาที่จะพัฒนา
5. ถ้าหากในชุดพัฒนาของเราไม่มี Headerfile ที่สนับสนุนในภาษาที่ต้องการจะต้องมีการแปลง Header file ไปยังภาษานั้นๆ
6. การแปลง Header file ด้วยตนเองอาจเกิดข้อผิดพลาดได้ง่ายและหาเจอได้ยากในบางครั้ง ดังนั้นในขั้นตอนการแปลงจึงควรระมัดระวังเป็นพิเศษ

2.9.2 ส่วนการประมวลผลลายนิ้วมือ - เริ่มการใช้งาน

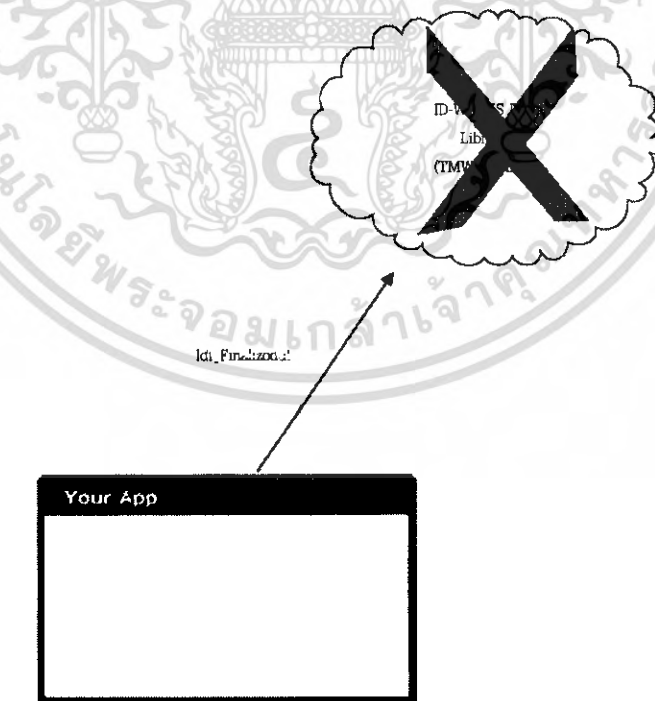
เพื่อให้ Library พร้อมใช้งานก่อนการใช้งานทั้งหมดจะต้องเรียก `idi_InitModule` เป็นขั้นแรกดังรูปที่ 2.13

2.9.3 ส่วนการประมวลผลลายนิ้วมือ - จบการใช้งาน

เมื่อหลังจากการเลิกใช้งานแล้ว (เช่น ก่อนปิดโปรแกรมของคุณ) จะต้องเรียก `idi_FinalizeModule` หลังเรียกฟังก์ชันดังกล่าวแล้ว จะไม่สามารถเรียกใช้งานฟังก์ชันในโมดูลได้อีกดังรูปที่ 2.14



รูปที่ 2.13 การเรียกฟังก์ชัน `idi_InitModule` เป็นการเริ่มใช้งาน



รูปที่ 2.14 การเรียกฟังก์ชัน `idi_FinalizeModule` เป็นการจบการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9.4 ส่วนการประมวลผลหลายนิ้วมือ - Context

ส่วนของ Context เป็นเหมือนโมดูลย่อยที่เราสร้างขึ้นมาเพื่อใช้งาน ID-WORKS Integrator ในโปรแกรม โดยการใช้งาน function ต่างๆ จะต้องอ้างถึง context เสมอส่วนในการสร้าง Context จะใช้ function `idi_CreateContext` ดังรูปที่ 2.15



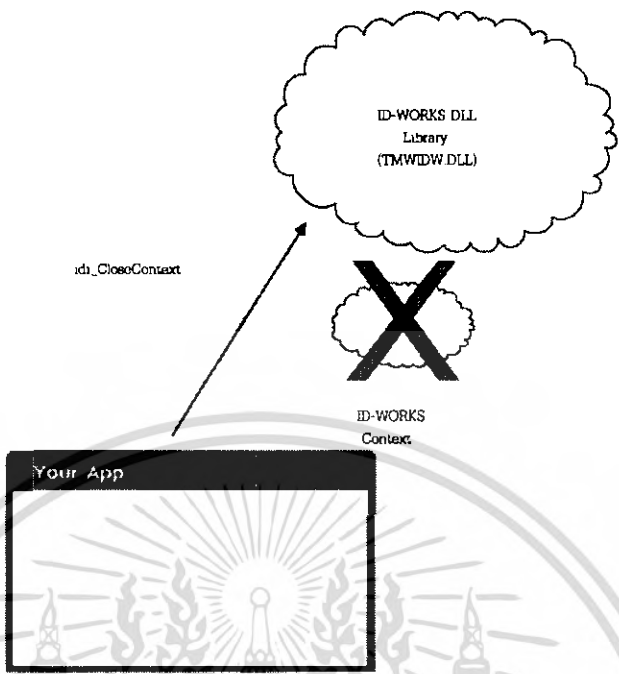
รูปที่ 2.15 การเรียกใช้งานของ Context

และถ้าไม่ต้องการใช้งาน Context แล้วให้ปิด Context ดังกล่าวโดยการเรียก `idi_CloseContext` ดังรูปที่ 2.16

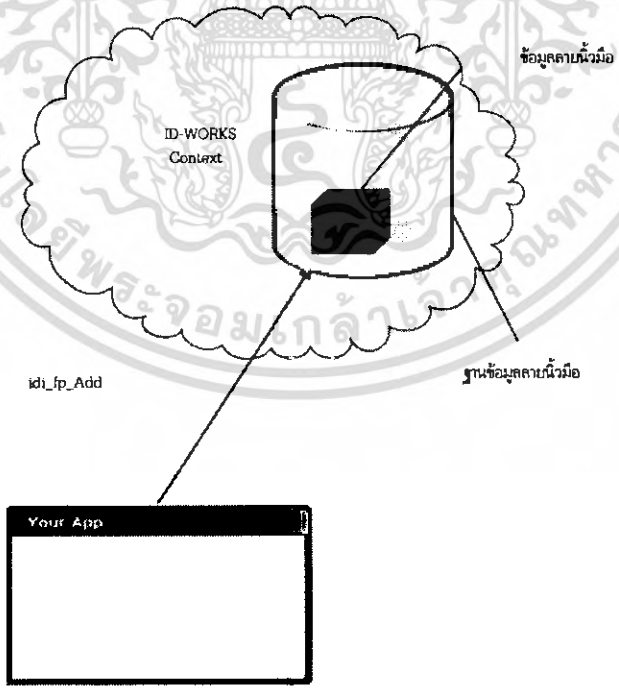
2.9.5 ส่วนการประมวลผลหลายนิ้วมือ - ข้อมูลหลายนิ้วมือ

ในแต่ละ Context ของ ID-WORKS จะมีฐานข้อมูลหลายนิ้วมือชั่วคราวเป็นของตัวเอง (จะหายไปเมื่อปิด Context) และฐานข้อมูลจะถูกใช้เมื่อมีการแตะนิ้วมือ ID-WORKS จะทำการประมวลผลและค้นหาหลายนิ้วมือจากฐานข้อมูลชั่วคราวนี้

ดังนั้นผู้พัฒนาจึงต้องเพิ่มข้อมูลเข้าไปในฐานข้อมูลชั่วคราว เพื่อให้ ID-WORKS สามารถทำงานได้ `idi_fp_Add` ใช้สำหรับเพิ่มข้อมูลหลายนิ้วมือที่จัดเก็บไว้แล้ว เข้าไปในฐานข้อมูลดังรูปที่ 2.17



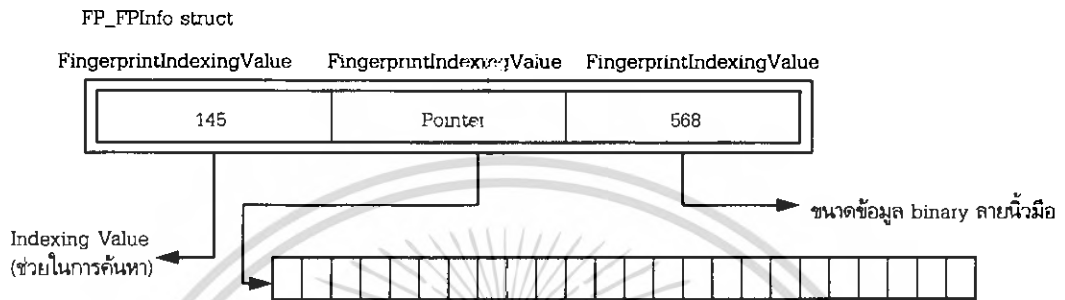
รูปที่ 2.16 การปิด Context



รูปที่ 2.17 การเพิ่มข้อมูลเข้าไปในฐานข้อมูลชั่วคราว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลลายนิ้วมือ สามารถเก็บได้โดยใช้ส่วนช่วยการจัดเก็บลายนิ้วมือ ส่วนในข้อมูลลายนิ้วมือจะประกอบด้วย 2 ส่วนหลักคือ ข้อมูลทั่วไปและข้อมูล binary ที่เป็นรายละเอียดของลายนิ้วมือซึ่งสามารถนิยมได้โดยใช้ FP_FPInfo ดังรูปที่ 2.18

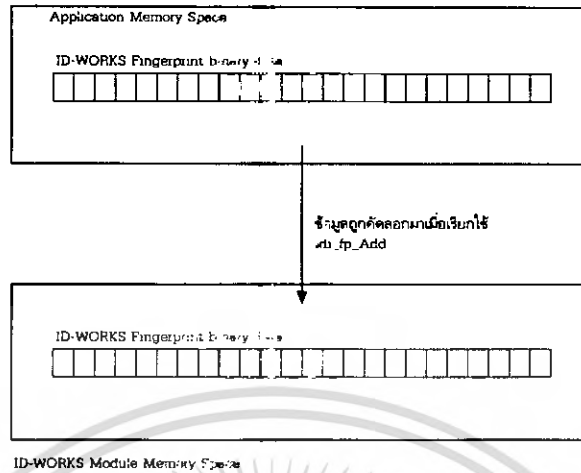


รูปที่ 2.18 ข้อมูลประเภท FP_FPINFO

ข้อมูลลายนิ้วมือ จะสามารถอ้างถึงได้ผ่านทางข้อมูลประเภท FP_FPINFO ได้ดังนี้

1. FP_FPINFO.FingerprintRawDataPtr จะต้องชี้ไปยังข้อมูล binary ของลายนิ้วมือ
2. FP_FPINFO.FingerprintRawDataSize จะต้องเป็นขนาดของข้อมูล binary ที่เป็นข้อมูลของลายนิ้วมือ

หากต้องการล้างฐานข้อมูลชั่วคราว ให้ใช้ฟังก์ชัน `idi_fp_ClearAll` เมื่อเราทำการเรียกใช้ฟังก์ชัน `idi_fp_Add` โมดูลจะทำการคัดลอกข้อมูลลายนิ้วมือ ไปเก็บไว้ในฐานข้อมูลชั่วคราว ซึ่งเมื่อคัดลอกแล้ว ก็จะไม่เกี่ยวกับ FP_FPInfo ที่ส่งเข้าไปกับฟังก์ชันอีก ดังนั้น memory ในส่วนที่เก็บข้อมูล binary ของลายนิ้วมือ จึงสามารถ deallocate ได้ดังรูปที่ 2.19



รูปที่ 2.19 การใช้ฟังก์ชันคัดลอกข้อมูลลายนิ้วมือ

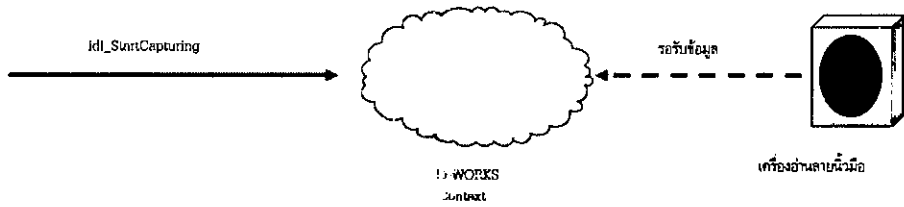
2.9.6 ส่วนการประมวลผลลายนิ้วมือ - เครื่องอ่านลายนิ้วมือ

ก่อนการใช้งานเครื่องอ่านลายนิ้วมือทุกครั้ง จะต้องมีการเรียกฟังก์ชัน `idi_ActivateSensor` ฟังก์ชันนี้จะต้องถูกเรียกในโปรแกรมในช่วงที่ เริ่มมีการใช้เครื่องอ่าน การเรียกฟังก์ชันนี้จะต้องมี Serial และ Key ของ ID-WORKS Integrator ซึ่งจะเป็นรหัสเฉพาะของเครื่องอ่านลายนิ้วมือแต่ละเครื่องซึ่งจะไม่ซ้ำกัน

ข้อมูลที่จำเป็นสำหรับการเรียก `idi_ActivateSensor`

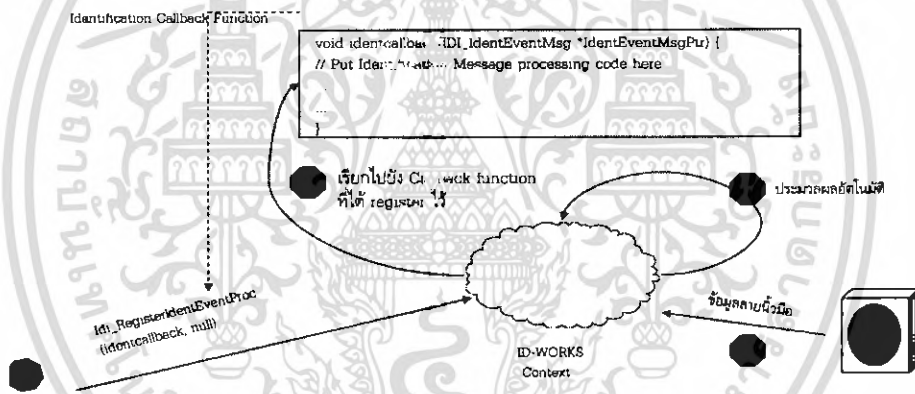
1. Serial ของเครื่องอ่าน สำหรับ U.are.U 4000 จะอยู่ในรูปของ GUID เช่น {2F58A5B6-F18B-3545-A764-82898C11A8B3}
2. ID-WORKS Integrator Serial จะเป็นรหัส 16 หลัก เช่น 29F6-C582-832E-8B56
3. ID-WORKS Integrator Key จะเป็นรหัส 32 หลัก เช่น 6325-C9B3-282C-2763-D4EF-363F-1F60-7F19
4. ในการใช้งานจริง ควรจัดทำหน้าตาสำหรับผู้ใช้เพื่อให้สามารถกรอก Serial และ Key ได้เองและเก็บใน Registry หรือ INI file

เมื่อส่งคำสั่ง `idi_Activate` แล้ว สามารถเริ่มการทำงานของเครื่องอ่านลายนิ้วมือได้ โดยใช้ฟังก์ชัน `idi_StartCapturing` หลังจากเรียกแล้ว โมดูล ID-WORKS จะรับข้อมูลจากเครื่องอ่านลายนิ้วมือ ดังรูปที่ 2.20



รูปที่ 2.20 การรับข้อมูลจากเครื่องอ่านลายนิ้วมือ

เมื่อสั่งคำสั่งเริ่มการทำงานของเครื่องอ่านลายนิ้วมือแล้ว โมดูล ID-WORKS จะส่ง Message กลับมาหา Application ผ่านทาง callback function ที่ register ไว้สามารถ register callback function ได้โดยใช้ฟังก์ชัน `Idi_RegisterIdentEventProc` ดังรูปที่ 2.21



รูปที่ 2.21 การทำงานของเครื่องอ่านลายนิ้วมือ

2.9.7 ส่วนการประมวลผลลายนิ้วมือ - การจัดการ Message

เมื่อมีผู้ใช้งานแต่ละลายนิ้วมือบนเครื่องอ่านลายนิ้วมือ ก็จะสามารถจับภาพลายนิ้วมือได้ ฟังก์ชัน Callback ที่ register ไว้จะถูกเรียก ไม่ว่าจะรู้จักลายนิ้วมือนั้นหรือไม่ และส่วนคำสั่ง `IDI_IdentEvent-Msg.MsgID` บ่งบอกประเภทของ Message เช่นรู้จักลายนิ้วมือหรือไม่รู้จัก

การใช้งานค่า `MsgParam` ใน `IDI_IdentEventMsg` จะขึ้นกับค่า `MsgID` กรณี `IDI_IdentEvent-Msg.MsgID` เป็น `IDI_ID_FINGERPRINTMATCHED` ส่วนในค่า `MsgParam` จะเป็น Pointer ที่ชี้ไปยัง `IDI_IdentEventMatchedStruct` ในค่าของ `IDI_IdentEventMatchedStruct` จะมี Pointer ที่ชี้ไปยัง `FP_FPInfo` อีกทีหนึ่ง ในการใช้ฟังก์ชัน `Idi_RegisterIdentEventProc` นั้นสามารถที่จะส่งค่า `RefParam`

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งเป็น Pointer เข้าไปด้วยได้ ซึ่งจะถูกใส่กลับไปใน IDI_IdentEventMsg.RefParam เมื่อมีการเรียกกลับมายัง callback function เนื่องจาก idi_RegisterIdentEventProc จะรับ callback function เป็น function ธรรมดา ดังนั้นผู้พัฒนาบางท่านอาจมีปัญหากในการเรียกกลับไปยัง method ของ object ซึ่งสามารถแก้ปัญหาได้โดยใช้ประโยชน์ RefParam ในการอ้างถึง object ที่ทำการ Register Event ในกรณีเขียนโปรแกรมในรูปแบบของ object-oriented

2.10 สรุปฟังก์ชันการใช้งานของเครื่องสแกน U.are.U 4000

2.10.1 สรุปฟังก์ชันทั้งหมดใน IDWORKS Integrator-ส่วนการประมวลผลลายนิ้วมือ

2.10.1.1 การเริ่มและเลิกใช้งานไลบรารี

ดังตารางที่ 2.1

ตารางที่ 2.1 ฟังก์ชันการเริ่มและเลิกใช้งานไลบรารีของส่วนการประมวลผลลายนิ้วมือ

ชื่อฟังก์ชัน	คำอธิบาย
idi_InitModule	เริ่มการทำงานของไลบรารี
idi_FinalizeModule	จบการทำงานของไลบรารี

2.10.1.2 การสร้างและปิด context และการ Activate

ดังตารางที่ 2.2

ตารางที่ 2.2 ฟังก์ชันการสร้างและปิด context และการ Activate ของส่วนการประมวลผลลายนิ้วมือ

ชื่อฟังก์ชัน	คำอธิบาย
idi_CreateContext	สร้าง IDWORKS processor context
idi_CloseContext	ปิด IDWORKS processor context
idi_ActivateSensor	ลงทะเบียนเครื่องอ่านลายนิ้วมือ ก่อนเริ่มการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10.1.3 การควบคุมเครื่องอ่านลายนิ้วมือ

ดังตารางที่ 2.3

ตารางที่ 2.3 ฟังก์ชันการควบคุมเครื่องอ่านลายนิ้วมือของส่วนการประมวลผลลายนิ้วมือ

ชื่อฟังก์ชัน	คำอธิบาย
idi_StartCapturing	เริ่มการรอรับการแตะนิ้วมือ
idi_StopCapturing	หยุดรอรับการแตะนิ้วมือ
idi_RegisterIdentEventProc	Register callback function สำหรับเรียกกลับ เพื่อแจ้งผลการค้นหาลายนิ้วมือกับ application
idi_UnregisterIdentEventProc	Unregister callback function ที่ register ไว้โดยฟังก์ชัน idi_RegisterIdentEventProc

2.10.1.4 การเพิ่มและลบลายนิ้วมือ

ดังตารางที่ 2.4

ตารางที่ 2.4 ฟังก์ชันการเพิ่มและลบลายนิ้วมือของส่วนการประมวลผลลายนิ้วมือ

ชื่อฟังก์ชัน	คำอธิบาย
idi_fp_Add	เพิ่มลายนิ้วมือเข้าไปในฐานข้อมูลชั่วคราวใน context
idi_fp_ClearAll	ลบลายนิ้วมือทั้งหมดในฐานข้อมูลชั่วคราวใน context

2.10.2 สรุปฟังก์ชันทั้งหมดใน IDWORKS Integrator-ส่วนการลงทะเบียนลายนิ้วมือ

2.10.2.1 การเริ่มและเลิกใช้งานไลบรารี

ดังตารางที่ 2.5

ตารางที่ 2.5 ฟังก์ชันการเริ่มและเลิกใช้งานไลบรารีของส่วนการลงทะเบียนลายนิ้วมือ

ชื่อฟังก์ชัน	คำอธิบาย
fpe_InitModule	เริ่มการทำงานของไลบรารี
fpe_FinalizeModule	จบการทำงานของไลบรารี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10.2.2 การสร้างและปิด context และการ Activate

ดังตารางที่ 2.6

ตารางที่ 2.6 ฟังก์ชันการสร้างและปิด context และการ Activate ของส่วนการลงทะเบียนนิ้วมือ

ชื่อฟังก์ชัน	คำอธิบาย
fpe_CreateContext	สร้าง context สำหรับการลงทะเบียนลายนิ้วมือ
fpe_CloseContext	ปิด context สำหรับการลงทะเบียนลายนิ้วมือ
fpe_ActivateSensor	ลงทะเบียนเครื่องอ่านลายนิ้วมือ ก่อนเริ่มการใช้งาน

2.10.2.3 การลงทะเบียนลายนิ้วมือ

ดังตารางที่ 2.7

ตารางที่ 2.7 ฟังก์ชันการลงทะเบียนลายนิ้วมือของส่วนการลงทะเบียนลายนิ้วมือ

ชื่อฟังก์ชัน	คำอธิบาย
fpe_StartEnrolment	เริ่มการลงทะเบียนลายนิ้วมือ

2.10.3. สรุปฟังก์ชันทั้งหมดใน IDWORKS Integrator-ส่วนข้อมูลเครื่องอ่านลายนิ้วมือ

2.10.3.1 การเริ่มและเลิกใช้งานไลบรารี

ดังตารางที่ 2.8

ตารางที่ 2.8 ฟังก์ชันการเริ่มและเลิกใช้งานไลบรารีของส่วนข้อมูลเครื่องอ่านลายนิ้วมือ

ชื่อฟังก์ชัน	คำอธิบาย
isl_InitModule	เริ่มการทำงานของไลบรารี
isl_FinalizeModule	จบการทำงานของไลบรารี
isl_IsModuleInited	ตรวจสอบว่าได้ทำการ Initialize แล้วหรือไม่

2.10.3.2 การดึงข้อมูลเครื่องอ่านลายนิ้วมือ

ดังตารางที่ 2.9

ตารางที่ 2.9 ฟังก์ชันการดึงข้อมูลเครื่องอ่านลายนิ้วมือของส่วนข้อมูลเครื่องอ่านลายนิ้วมือ

ชื่อฟังก์ชัน	คำอธิบาย
isl_GetDeviceCount	ให้นับจำนวนเครื่องอ่านลายนิ้วมือในระบบ
isl_GetDeviceList	ดึงข้อมูลเครื่องอ่านลายนิ้วมือทั้งหมดในระบบ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบ การสร้าง และการทำงาน

3.1 กล่าวนำ

ในการออกแบบระบบบันทึกเวลาโดยใช้ลายนิ้วมือ จะประกอบด้วยส่วนประกอบต่างๆ คือ ส่วนโปรแกรมของระบบบันทึกเวลาโดยใช้ลายนิ้วมือ ส่วนคอมพิวเตอร์ที่ใช้ทำฐานข้อมูล และทำหน้าที่ติดต่อกับเครื่องสแกนลายนิ้วมือ ต่อมาส่วนของเครื่องสแกนลายนิ้วมือทำหน้าที่สแกนลายนิ้วมือ ดังรูปที่ 3.1



รูปที่ 3.1 ผังการทำงานของระบบบันทึกเวลาโดยใช้ลายนิ้วมือ

ส่วนคอมพิวเตอร์จะทำหน้าที่ติดต่อกับเครื่องสแกนลายนิ้วมือโดยจะติดต่อผ่าน USB Port ซึ่งข้อมูลลายนิ้วมือ จะส่งผ่านโปรแกรมตรวจสอบลายนิ้วมือและจะทำการตรวจสอบลายนิ้วมือของพนักงานลงไปเก็บไว้ในฐานข้อมูล

ซึ่งในบทนี้จะกล่าวถึงวิธีการออกแบบการสร้างและการทำงานของระบบบันทึกเวลาโดยใช้ลายนิ้วมือ

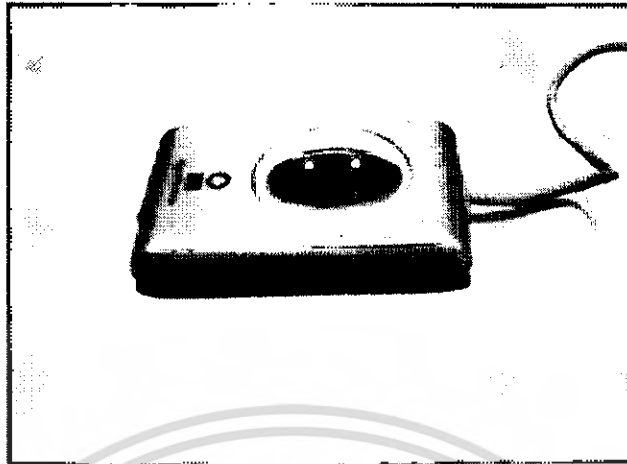
3.2 การออกแบบ

ในการออกแบบระบบบันทึกเวลาโดยใช้ลายนิ้วมือ ซึ่งแบ่งการออกแบบออกเป็น 4 ส่วน คือ

1. ส่วนของเครื่องสแกนลายนิ้วมือ
2. ส่วนของการเตรียมเครื่องคอมพิวเตอร์ที่ใช้
3. ส่วนของโปรแกรมระบบบันทึกเวลาโดยใช้ลายนิ้วมือ
4. ส่วนของฐานข้อมูล

3.2.1 ส่วนของเครื่องสแกนลายนิ้วมือ

เนื่องจากได้ใช้เครื่องสแกนลายนิ้วมือรุ่น DigitalPersona U.are.U 4000 ที่สำเร็จรูปมาใช้ โดยการส่งข้อมูลผ่านทาง USB Port โดยใช้การเขียนโปรแกรมติดต่อกับระบบบันทึกเวลาโดยใช้ลายนิ้วมือ ดังรูปที่ 3.1



รูปที่ 3.2 เครื่องสแกนลายนิ้วมือรุ่น DigitalPersona U.are.U 4000

3.2.2 ส่วนของการเตรียมเครื่องคอมพิวเตอร์ที่ใช้

ระบบปฏิบัติการที่ใช้งาน วินโดวส์ เอ็กซ์พี (Windows XP) ไมโครซอฟต์ วิวิลเบสิค เวอร์ชัน 2005 (Microsoft Visual Basis 2005) และไมโครซอฟต์แอ็กซีส 2003 (Microsoft Access 2003) เพื่อที่จะสามารถใช้โปรแกรมระบบบันทึกเวลาโดยใช้ลายนิ้วมือ ได้อย่างสมบูรณ์

3.2.3 ส่วนของโปรแกรมที่ใช้

ในโครงการนี้ได้ใช้โปรแกรมไมโครซอฟต์ วิวิลเบสิค เวอร์ชัน 2005 (Microsoft Visual Basis 2005) สำหรับการออกแบบโปรแกรมระบบบันทึกเวลาโดยใช้ลายนิ้วมือ

3.2.3.1.การออกแบบ

ฟอร์ม Time Attention By Fingerprint จะประกอบด้วยฟอร์มต่างๆดังนี้

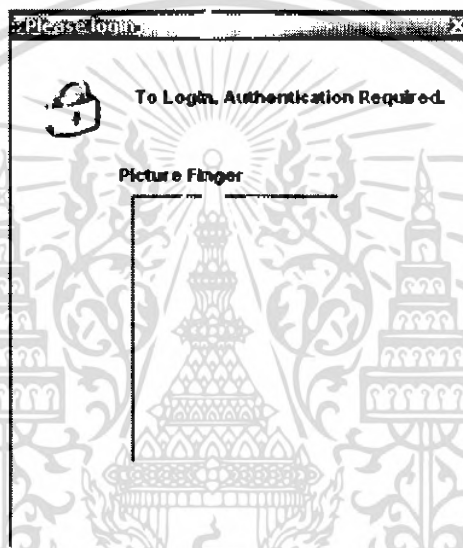
1. ฟอร์ม การ Login เข้าสู่ระบบ
2. ฟอร์ม หลักของระบบบันทึกเวลาโดยใช้ลายนิ้วมือ
3. ฟอร์ม บันทึกเวลาเข้า-ออกของพนักงาน
4. ฟอร์ม บันทึกข้อมูลและลายนิ้วมือของพนักงาน
5. ฟอร์ม การตรวจสอบเวลาเข้า-ออกของพนักงาน
6. ฟอร์ม การตรวจสอบการขาดงานหรือมกสายของพนักงาน
7. ฟอร์ม กำหนดเวลาการทำงานของพนักงาน
8. ฟอร์ม กำหนดวันหยุดของพนักงาน
9. ฟอร์ม ตารางตรวจสอบการเข้าทำงานและออกของพนักงานเฉพาะบุคคล
10. ฟอร์ม ตารางตรวจสอบการเข้าทำงานและออกของพนักงานทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. ฟอรั่ม ตารางตรวจสอบการขาดงานของพนักงานเฉพาะบุคคล
12. ฟอรั่ม ตารางตรวจสอบการขาดงานของพนักงานทั้งหมด
13. ฟอรั่ม ตารางตรวจสอบการหยุดงานงานของพนักงานเฉพาะบุคคล
14. ฟอรั่ม ตารางตรวจสอบการหยุดงานของพนักงานทั้งหมด

3.2.3.2. การสร้าง

1. ฟอรั่ม การ Login เข้าสู่ระบบ ซึ่งเป็นหน้าต่าง Login โดยการสแกนลายนิ้วมือเพื่อที่เข้าไปใช้งานในระบบของเจ้าหน้าที่จะมีฟอรั่มลักษณะดังรูปที่ 3.3



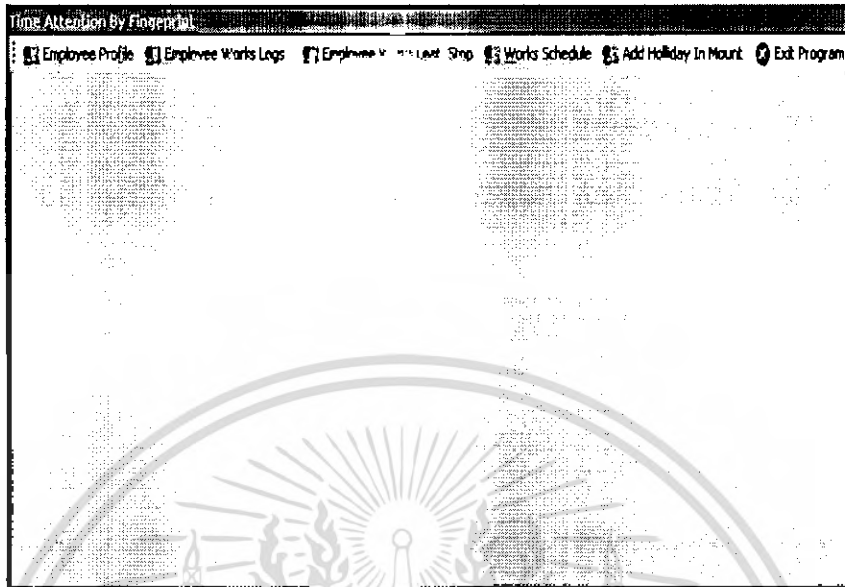
รูปที่ 3.3 ฟอรั่ม การ Login เข้าสู่ระบบ

2. ฟอรั่ม หลักของระบบบันทึกเวลาโดยใช้ลายนิ้วมือ ซึ่งเป็นหน้าต่างหลักของโปรแกรมระบบลงเวลาโดยใช้ลายนิ้วมือ ซึ่งจะประกอบไปด้วยปุ่ม Employee Profile, Employee WorksLog, Employee Works LateStop, Works Schedule, AddHolliday in Month, Exit Program จะมีฟอรั่มลักษณะดังรูปที่ 3.4

3. ฟอรั่ม บันทึกเวลาเข้า-ออกของพนักงาน ซึ่งเป็นหน้าต่างการบันทึกเวลาโดยใช้ลายนิ้วมือ ในหน้าต่างนี้เป็นการใช้งานของพนักงาน สำหรับให้พนักงานลงเวลาเข้า-ออก พนักงานจะต้องทำการสแกนลายนิ้วมือเพื่อบันทึกเวลาเข้าออก โดยการวางนิ้วบนเครื่องสแกน จะมีฟอรั่มลักษณะดังรูปที่ 3.5

4. ฟอรั่ม บันทึกข้อมูลและลายนิ้วมือของพนักงาน ซึ่งเป็นหน้าต่างข้อมูลของพนักงานใช้ในการจัดเก็บข้อมูลของพนักงานและเจ้าหน้าที่และยังสามารถ ค้นหาข้อมูล เดิมแก้ไขข้อมูลเดิม ค้นหารูป กำหนดสถานะและสแกนลายนิ้วมือของพนักงานและเจ้าหน้าที่ จะมีฟอรั่มลักษณะดังรูปที่ 3.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 ฟอรัม หลักของระบบบันทึกเวลาโดยใช้ลายนิ้วมือ

รูปที่ 3.5 ฟอรัม บันทึกเวลาเข้า-ออกของพนักงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.6 ฟอรม บันทึกข้อมูลและลายนิ้วมือของพนักงาน

5. ฟอรม การตรวจสอบเวลาเข้า-ออกของพนักงาน ซึ่งเป็นหน้าตางของข้อมูลพนักงาน ใ้ใช้ในการตรวจสอบการลงเวลาเข้าออก ซึ่งเป็นการใช้งานของเจ้าหน้าที่ สำหรับให้เจ้าหน้าที่ตรวจสอบการเข้ามาทำงานและออกงานในแต่ละวันได้ โดยทำการเลือกวันที่จะดู และยังสามารถเลือกรหัสประจำตัวของพนักงานหรือเลือกดูทั้งหมดได้ แล้วทำการค้นหาโดยการกดปุ่ม Find จะมีฟอรมลักษณะดังรูปที่ 3.7

รูปที่ 3.7 ฟอรม การตรวจสอบเวลาเข้า-ออกของพนักงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คนใดคนหนึ่งหรือทั้งหมด จากนั้นทำการเลือกดูวันมาสายหรือขาดงาน โดยเลือกที่ ReportLate หรือ Report Stop และยังสามารถเลือกดูเป็นรายวันหรือเดือนก็ได้ โดยกดปุ่ม All Month และ All Day จะมีฟอร์มลักษณะดังรูปที่ 3.8

7. ฟอร์ม กำหนดเวลาการทำงานของพนักงาน ซึ่งเป็นหน้าต่างของการกำหนดเวลาทำงานของพนักงานในแต่ละช่วงเวลาและยังสามารถกำหนดได้ในแต่ละแผนกหรือตำแหน่งต่างๆภายในบริษัท โดยการกำหนดแผนกและตำแหน่งแล้วกดปุ่ม Search ก็จะมีปรากฏรายชื่อของพนักงานที่มีอยู่ในแต่ละแผนกและตำแหน่งที่มีอยู่ แล้วทำการเลือกพนักงาน กำหนดเวลาให้แต่ละบุคคลและก็ทำการบันทึก จะมีฟอร์มลักษณะดังรูปที่ 3.9

8. ฟอร์ม กำหนดวันหยุดของพนักงาน ซึ่งเป็นหน้าต่างของการกำหนดวันหยุดของบริษัท โดยการเลือกวันที่ แล้วทำการกำหนดว่าเป็นวันหยุดอะไร กดปุ่ม save เพื่อทำการบันทึก จะมีฟอร์มลักษณะดังรูปที่ 3.10

9. ฟอร์ม ตารางตรวจสอบการเข้าทำงานและออกของพนักงานเฉพาะบุคคล ซึ่งเป็นฟอร์มที่ออกแบบการสร้างรูปแบบรายงานของข้อมูลพนักงานที่ทำการบันทึกเวลาเข้าออกแล้วแบบรายบุคคล คือการนำข้อมูลของพนักงานที่ทำการบันทึกเวลาเข้าออกแล้วมาสร้างเป็นรายงานเฉพาะบุคคล สามารถพิมพ์ข้อมูลของพนักงานออกมาเป็นรายงานได้ จะมีลักษณะฟอร์มดังรูปที่ 3.11

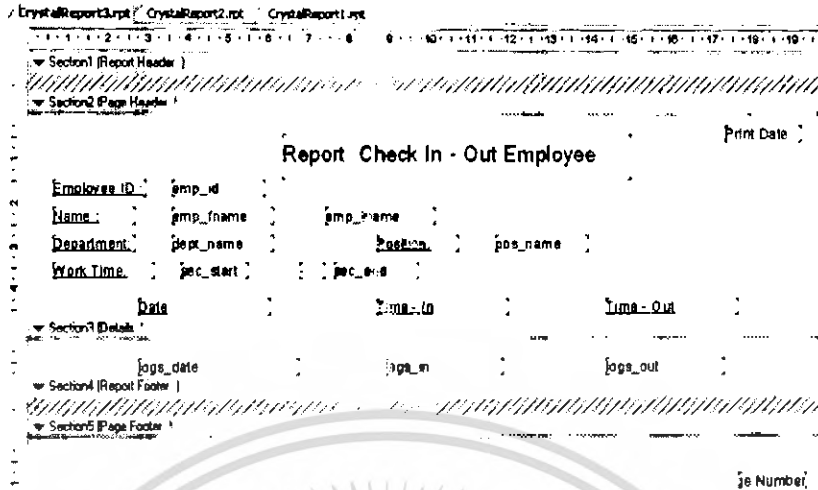
รูปที่ 3.8 ฟอร์ม การตรวจสอบการขาดงานหรือมาสายของพนักงาน

รูปที่ 3.9 ฟอรัม กำหนดเวลาการทำงานของพนักงาน

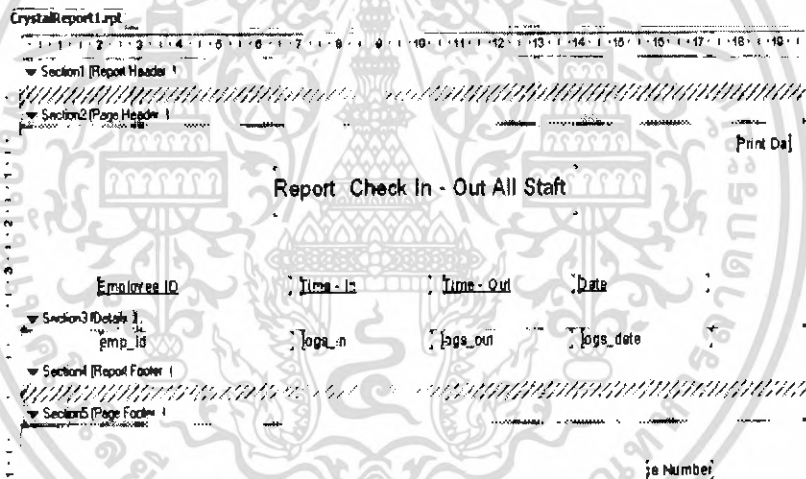
รูปที่ 3.10 ฟอรัม กำหนดวันหยุดของพนักงาน

10. ฟอรัม ตารางตรวจสอบการเข้าทำงานและออกของพนักงานทั้งหมด ซึ่งเป็นฟอร์มที่ออกแบบการสร้างรูปแบบรายงานของข้อมูลพนักงานที่ทำการบันทึกเวลาเข้าออกแบบเพื่อดูรายงานของพนักงานทุกคน คือการนำข้อมูลของพนักงานที่ทำการบันทึกเวลาเข้าออกแล้วมาสร้างเป็นรายงานของพนักงานทั้งหมด สามารถพิมพ์ข้อมูลของพนักงานออกมาเป็นรายงานได้ จะมีลักษณะฟอร์มดังรูปที่ 3.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 φόรึม ตารางตรวจสอบการเข้าทำงานและออกของพนักงานเฉพาะบุคคล



รูปที่ 3.12 φόรึม ตารางตรวจสอบการเข้าทำงานและออกของพนักงานทั้งหมด

11. φόรึม ตารางตรวจสอบการขาดงานของพนักงานเฉพาะบุคคล ซึ่งเป็นฟอร์มที่ออกแบบเพื่อรายงานของพนักงานทุกคน คือการนำข้อมูลของพนักงานที่มีการขาดงานมาสร้างเป็นรายงานของพนักงานทุกคนเพื่อทำการตรวจสอบ สามารถพิมพ์ข้อมูลของพนักงานที่ขาดงานออกมาเป็นรายงานได้ จะมีลักษณะฟอร์มดังรูปที่ 3.13

12. φόรึม ตารางตรวจสอบการขาดงานของพนักงานทั้งหมด ซึ่งเป็นฟอร์มที่ออกแบบ เพื่อดูรายงานของพนักงานทุกคนที่ขาดงาน คือการนำข้อมูลของพนักงานทุกคนที่มีการขาดงานมาสร้างเป็นรายงานเพื่อทำการตรวจสอบ สามารถพิมพ์ข้อมูลของพนักงานที่ขาดงานออกมาเป็นรายงานได้ จะมีลักษณะฟอร์มดังรูปที่ 3.14

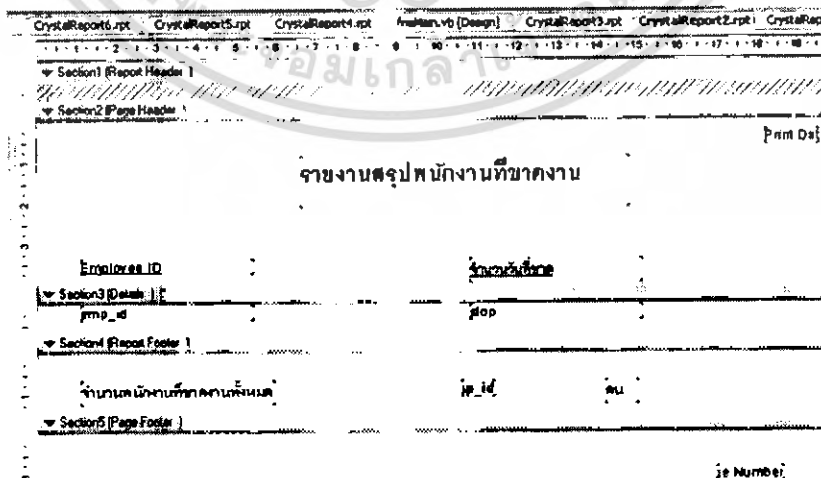
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

13. ฟอรั่ม ตารางตรวจสอบการหยุดงานของพนักงานเฉพาะบุคคล ซึ่งเป็นฟอรั่มที่ออกแบบเพื่อดูรายงานของพนักงานแต่ละบุคคลที่มาทำงานสาย คือการนำข้อมูลของพนักงานที่มาทำงานสายมาสร้างเป็นรายงานของพนักงานเฉพาะบุคคลมาสร้างเป็นรายงานเพื่อทำการตรวจสอบ สามารถพิมพ์ข้อมูลของพนักงานที่มาสายออกมาเป็นรายงานได้ จะมีลักษณะฟอรั่มดังรูปที่ 3.15

14. ฟอรั่ม ตารางตรวจสอบการหยุดงานของพนักงานทั้งหมด ซึ่งเป็นฟอรั่มที่ออกแบบเพื่อดูรายงานของพนักงานทุกคนที่มาทำงานสาย คือการนำข้อมูลของพนักงานที่มาสายมาสร้างเป็นรายงานของพนักงานทุกคนมาสร้างเป็นรายงานเพื่อทำการตรวจสอบ สามารถพิมพ์ข้อมูลของพนักงานที่มาสายออกมาเป็นรายงานได้ จะมีลักษณะฟอรั่มดังรูปที่ 3.16



รูปที่ 3.13 ฟอรั่ม ตารางตรวจสอบการขาดงานของพนักงานเฉพาะบุคคล



รูปที่ 3.14 ฟอรั่ม ตารางตรวจสอบการขาดงานของพนักงานทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CrystalReport5.rpt CrystalReport5.rpt CrystalReport5.rpt frmMain.vb (Design) CrystalReport3.rpt CrystalReport2.rpt CrystalRep

Section1 (Report Header 1)

Section2 (Page Header 1)

Print Data

รายงานสรุปมาสาย

Employee ID	Emp ID	Name	Emp Name	Department	Dept Name	Work Time	Sec Start	Sec End	Date
[ogs_data]									
Sum Late									
[ogs_data]									

Page Number

รูปที่ 3.15 ฟอรัม ตารางตรวจสอบการหยุดงานของพนักงานเฉพาะบุคคล

CrystalReport5.rpt CrystalReport5.rpt CrystalReport5.rpt frmMain.vb (Design) CrystalReport3.rpt CrystalReport2.rpt CrystalRep

Section1 (Report Header 1)

Section2 (Page Header 1)

Print Data

รายงานสรุปพนักงานที่มาสาย

Employee ID	Date	จำนวนชั่วโมง
[ogs_data]		
Sum Late		
[ogs_data]		

Page Number

รูปที่ 3.16 ฟอรัม ตารางตรวจสอบการหยุดงานของพนักงานทั้งหมด

3.2.4 ส่วนของการตรวจสอบลายนิ้วมือ

3.2.4.1 การออกแบบ

1. ฟอรัมลงทะเบียนลายนิ้วมือ

3.2.4.2. การสร้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. φόρμลงทะเบียนลายนิ้วมือจะเป็นหน้าต่างการลงทะเบียนลายนิ้วมือเป็นการสแกนลายนิ้วมือเพื่อบันทึกข้อมูลลายนิ้วมือของพนักงานลงไปในฐานะข้อมูล การสแกนลายนิ้วมือหนึ่งนิ้วจะทำการสแกนลายนิ้วมือเก็บข้อมูลทั้งหมด 4 ครั้ง จะมีลักษณะฟอร์ม ดังรูปที่ 3.17

3.2.5 ส่วนของฐานข้อมูล

เมื่อออกแบบระบบฐานข้อมูลของระบบเสร็จเรียบร้อยแล้ว ต้องจัดทำตารางเพื่อใช้จัดเก็บข้อมูลตามทีออกแบบ ซึ่งตารางในระบบโปรแกรมช่วยจัดการตารางเรียนตารางสอน มีรายละเอียดดังนี้คือ

3.2.5.1 ตารางกำหนดฐานข้อมูลมีทั้งหมด 8 ตาราง ดังตารางที่ 3.1

ตารางที่สร้างขึ้นมาทั้งหมดได้ผ่านการ Normalization เพื่อลดความซ้ำซ้อนของข้อมูลและมีรายละเอียดดังตารางที่ 3.1 ถึงตารางที่ 3.7 ในแต่ละตารางจะมีชื่อความในชองคีย์ ซึ่งมีความหมายดังนี้

PK หมายถึง คีย์หลักของตารางแอดทริบิวต์ จะมีข้อมูลไม่ซ้ำกันเลยในตารางนั้น

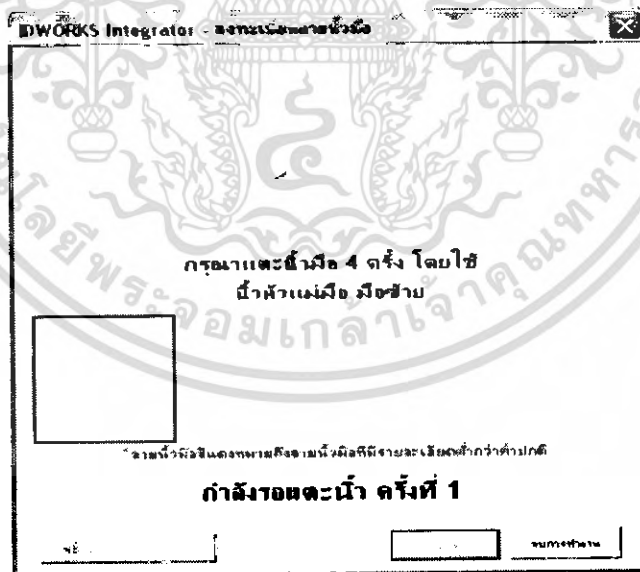
FK หมายถึง คีย์ที่อ้างอิงคีย์หลักของตารางอื่น

ในส่วนของชองอ้างอิงจะมีข้อความ 2 กรณี คือ

กรณีที่ 1 เป็นชื่อตาราง หมายถึง ข้อมูลของแอดทริบิวต์นั้นสอดคล้องกับค่าของคีย์หลักของตารางที่

อ้างอิง

กรณีที่ 2 เป็นข้อความ หมายถึง กระบวนการที่ทำให้ได้ข้อมูลของแอดทริบิวต์



รูปที่ 3.17 φόρμลงทะเบียนลายนิ้วมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.1 สรุปรายชื่อตารางทั้งหมดในโปรแกรมระบบบันทึกเวลาโดยใช้ลายนิ้วมือ

ลำดับ	ชื่อตาราง	ความหมาย
1	IdiFingerprint	กำหนดรายละเอียดของข้อมูลลายนิ้วมือ
2	tbl_Employee	กำหนดรายละเอียดของข้อมูลพนักงาน
3	tbl_Department	กำหนดรายละเอียดของแผนก
4	tbl_Position	กำหนดรายละเอียดของตำแหน่ง
5	tbl_Date-Holiday	กำหนดรายละเอียดของวันหยุด
6	tbl_Work-Late	กำหนดรายละเอียดของวันและเวลาที่พนักงานมาสาย
7	tbl_Work-Stop	กำหนดรายละเอียดของวันและเวลาที่พนักงานขาดงาน
8	tbl_WorkLog	กำหนดรายละเอียดของวันและเวลาที่พนักงานลงเวลา

3.2.5.2. การออกแบบฐานข้อมูล

1. ตารางรายละเอียดของ IdiFingerprint เป็นฐานข้อมูลในการเก็บข้อมูลลายนิ้วมือของพนักงานจะประกอบไปด้วย PK_EmpFingerPrintID, EmpID, FingerIndex, IndexValue, Features ดังตารางที่ 3.2 และรูปที่ 3.18

ตารางที่ 3.2 ตารางรายละเอียดของ IdiFingerprint

ลำดับ	ชื่อฟิลด์	คำอธิบาย	ชนิด	ความยาว	คีย์
1	PK_EmpFingerPrintID	ลำดับการบันทึก	Text	Long Integer	PK
2	EmpID	รหัสพนักงาน	Text	250	
3	FingerIndex	รหัสลายนิ้วมือ	Number	Long Integer	

PK_EmpFingerPrintID	EmpID	FingerIndex	IndexValue	Features
38	48035311	4	148	๘๙๓ Long binary
39	48035311	5	130	๘๙๓ Long binary
50	48035276	4	146	๘๙๓ Long binary
51	48035276	5	143	๘๙๓ Long binary
56	48035314	4	154	๘๙๓ Long binary
57	48035492	4	136	๘๙๓ Long binary
58	48035492	5	128	๘๙๓ Long binary
61	48035261	4	144	๘๙๓ Long binary
62	48035261	5	143	๘๙๓ Long binary
63	48035258	4	142	๘๙๓ Long binary
64	48035258	5	137	๘๙๓ Long binary

(AutoNumber)

รูปที่ 3.18 รายละเอียดข้อมูลในตาราง IdiFingerprint

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ตารางรายละเอียดของ tbl_employee เป็นฐานข้อมูลของพนักงานทั้งหมดประกอบไปด้วย emp_fname, emp_lname, emp_pid, emp_phone, emp_address, emp_birthdate, mp_salary, emp_id, dept_id, pos_id, sec_id, picture ดังตารางที่ 3.3 และรูปที่ 3.19

ตารางที่ 3.3 ตารางรายละเอียดของ tbl_employee

ลำดับ	ชื่อฟิลด์	คำอธิบาย	ชนิด	ความยาว	คีย์
1	Emp_id	รหัสพนักงาน	Text	20	PK
2	Emp_fname	ชื่อพนักงาน	Text	30	
3	Emp_lname	นามสกุลพนักงาน	Text	40	
4	Emp_pid	รหัสประจำตัวประชาชน	Text	13	
5	Emp_phone	เบอร์โทรศัพท์	Text	20	
6	Emp_address	ที่อยู่	Text	255	
7	Emp_salary	ฐานเงินเดือน	Number	Long Integer	
8	Emp_birthdate	วันเกิด	Date/Time	Short Date	
9	Dept_id	รหัสแผนก	Number	Long Integer	
10	Pos_id	รหัสตำแหน่ง	Number	Long Integer	
11	Sec_id	รหัสกะเวลา	Number	Long Integer	
12	Picture	รูปภาพ	Text	200	

	emp_id	emp_fname	emp_lname	emp_pid	emp_phone	emp_address	emp_salary
+ 1	1	1	1	1	1	1	1.00
+ 48035258	48035258	Warinnipar	Kajudman	1234567890	0814976659	479/2	18,000.00
+ 48035261	48035261	Kitisak	Yingyong	11200023116E	0866195498	802/765	35,000.00
+ 48035311	48035311	Ampawat	Phatcharaphatt	154010002236E	055610979	58/1	150,000.00
+ 48035314	48035314	Eakbadin	gingeson	1033261452	0815875231	78/1 m.2	15,000.00
+ rgg	rgg	rgg	rgg	rgg	4545	rgg	4,545.00
							0.00

emp_birthdate	dept_id	pos_id	sec_id	picture	status
10/5/2550	1	1	1	1	1
6/7/2526	1	1	1	1 D:\Project\NEMS 1	
22/3/2528	3	5	5	1 D:\Project\NEMS 1	
24/3/2528	1	1	1	1 D:\Project\NEMS 1	
7/4/2528	1	6	6	1 D:\Project\NEMS 0	1
10/5/2550	1	2	3	0	
	0	0	0	0	

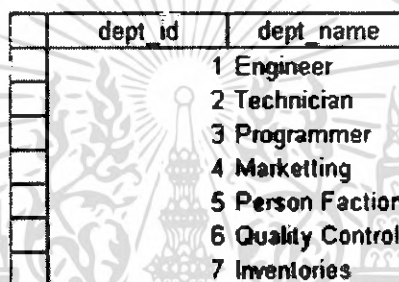
รูปที่ 3.19 รายละเอียดข้อมูลในตาราง tbl_employee

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ตารางรายละเอียดของ tbl_department เป็นฐานข้อมูลของแผนกต่างๆ ประกอบด้วย dept_id, dept_name ดังตารางที่ 3.4 และรูปที่ 3.20

ตารางที่ 3.4 ตารางรายละเอียดของ tbl_employee

ลำดับ	ชื่อฟิลด์	คำอธิบาย	ชนิด	ความยาว	คีย์
1	Dept_id	รหัสของแผนก	Number	Long Integer	PK
2	Dept_name	รายชื่อแผนก	Text	50	



	dept_id	dept_name
	1	Engineer
	2	Technician
	3	Programmer
	4	Marketing
	5	Person Faction
	6	Quality Control
	7	Inventories

รูปที่ 3.20 รายละเอียดข้อมูลในตาราง tbl_department

4. ตารางรายละเอียดของ tbl_position เป็นฐานข้อมูลของตำแหน่งต่างๆ ประกอบด้วย pos_id, pos_name ดังตารางที่ 3.5 และรูปที่ 3.21

ตารางที่ 3.5 ตารางรายละเอียดของ tbl_position

ลำดับ	ชื่อฟิลด์	คำอธิบาย	ชนิด	ความยาว	คีย์
1	Pos_id	รหัสตำแหน่ง	Number	Long Integer	
2	Pos_name	ชื่อตำแหน่ง	Text	50	

5. ตารางรายละเอียดของ tbl_section เป็นฐานข้อมูลของการกำหนดเวลา จะประกอบด้วย sec_id, sec_name, sec_start, sec_end ดังตารางที่ 3.6 และรูปที่ 3.22

pos_id	pos_name
1	Sale Engineer
2	Manager
3	Secretary
4	Director
5	Administrator
6	Project Engineer
7	Maintenance
8	Software Engineer
9	Electrical Techni

รูปที่ 3.21 รายละเอียดข้อมูลในตาราง tbl_position

ตารางที่ 3.6 ตารางรายละเอียดของ tbl_section

ลำดับ	ชื่อฟิลด์	คำอธิบาย	ชนิด	ความยาว	คีย์
1	Sec_id	รหัสกะเวลา	Number	Long Integer	PK
2	Sec_name	ชื่อกะเวลา	Text	30	
3	Sec_start	เวลาเริ่มต้น	Date/Time	Long Time	
4	Sec_end	เวลาหยุด	Date/Time	Long Time	

sec_id	sec_name	sec_start	sec_end
1	Morning	6:30:00	12:30:00
2	Afternoon	12:30:00	18:30:00
3	Night	18:30:00	23:30:00

รูปที่ 3.22 รายละเอียดข้อมูลในตาราง tbl_section

6. ตารางรายละเอียดของ tbl_holiday เป็นฐานข้อมูลของการกำหนดวันหยุดจะประกอบไปด้วย Date_Holiday, Date_Name ดังตารางที่ 3.7 และรูปที่ 3.23

ตารางที่ 3.7 ตารางรายละเอียดของ tbl_holiday

ลำดับ	ชื่อฟิลด์	คำอธิบาย	ชนิด	ความยาว	คีย์
1	Date_Holiday	วันหยุด	Date/Time	Long Time	
2	Date_Name	ชื่อวันหยุด	Text	50	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. ตารางรายละเอียดของ tbl_work_late เป็นฐานข้อมูลของการบันทึกการมาสายของพนักงาน จะประกอบไปด้วย emp_id, logs_date, logs_late ดังตารางที่ 3.7 และรูปที่ 3.24

Date Ho	day	Date Name
1/1/2550		วันขึ้นปีใหม่
1/2/2550		วันตรุษจีน
1/5/2550		วันแรงงานแห่งชาติ
10/12/2550		วันรัฐธรรมนูญ
12/8/2550		วันแม่แห่งชาติ
13/4/2550		วันสงกรานต์
13/7/2550		วันอาสาฬหบูชา
14/4/2550		วันสงกรานต์
14/7/2550		วันเข้าพรรษา
15/4/2550		วันสงกรานต์
15/5/2550		วันวิสาขบูชา
16/2/2550		วันมาฆบูชา
17/2/2550		ชดเชยวันมาฆบูชา
23/10/2550		วันปิยมหาราช
31/12/2550		วันสิ้นปี
5/12/2550		วันพ่อแห่งชาติ
5/5/2550		วันฉัตรมงคล
8/5/2550		วันวิษุวัต

รูปที่ 3.23 รายละเอียดข้อมูลในตาราง tbl_holiday

ตารางที่ 3.8 ตารางรายละเอียดของ tbl_work_late

ลำดับ	ชื่อฟิลด์	คำอธิบาย	ชนิด	ความยาว	คีย์
1	Emp_id	รหัสพนักงาน	Text	50	
2	Logs_date	วันที่สาย	Date/Time	Long Time	
3	Logs_late	ครั้งที่ขาดงาน	Text	50	

emp_id	logs_date	logs_late
48035258	10/5/2550	1
48035258	11/5/2550	1
48035261	10/5/2550	1
48035261	11/5/2550	1
48035311	11/5/2550	1
48035314	10/5/2550	1

รูปที่ 3.24 รายละเอียดข้อมูลในตาราง tbl_work_late

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. ตารางรายละเอียดของ tbl_work_stop เป็นฐานข้อมูลของการบันทึกการขาดของพนักงาน จะประกอบไปด้วย emp_id, month_stop, year_stop, stop ดังตารางที่ 3.9 และรูปที่ 3.25

ตารางที่ 3.9 ตารางรายละเอียดของ tbl_work_stop

ลำดับ	ชื่อฟิลด์	คำอธิบาย	ชนิด	ความยาว	คีย์
1	Emp_id	รหัสพนักงาน	Text	50	
2	Month_stop	เดือนที่มาสาย	Text	50	
3	Year_stop	ปีที่ทำงาน	Text	50	
4	Stop	จำนวนครั้งที่หยุด	Text	50	

emp_id	month_stop	year_stop	stop
48035258	5	2007	20
48035311	5	2007	20
48035261	5	2007	21
48035276	5	2007	21
48035223	5	2007	21
48035275	5	2007	20
48035235	5	2007	20
48035234	6	2007	21

รูปที่ 3.25 รายละเอียดข้อมูลในตาราง tbl_work_stop

9. ตารางรายละเอียดของ tbl_worlog เป็นฐานข้อมูลของการบันทึกเวลาเข้าออกของพนักงาน จะประกอบไปด้วย emp_id, logs_in, logs_out, logs_date ดังตารางที่ 3.10 และรูปที่ 3.26

ตารางที่ 3.10 ตารางรายละเอียดของ tbl_wokrlog

ลำดับ	ชื่อฟิลด์	คำอธิบาย	ชนิด	ความยาว	คีย์
1	Emp_id	รหัสพนักงาน	Text	50	
2	Logs_in	เวลาที่เข้างาน	Date/Time	Long Time	
3	Logs_out	เวลาที่ออกงาน	Date/Time	Long Time	
4	Logs_date	วันที่มาทำงาน	Date/Time	Long Time	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

4.1 กล่าวนำ

ในส่วนของบทนี้จะกล่าวถึงการทดลองและผลการทดลองของระบบบันทึกเวลาเวลาโดยใช้ลายนิ้วมือ โดยการนำโปรแกรม Visual Studio 2005 เข้ามาช่วยในการสร้างโปรแกรมและนำโปรแกรม MS Access มาเป็นตัวสร้างฐานข้อมูลโดยใช้การเชื่อมต่อฐานข้อมูลแบบ ODBC เพื่อดึงข้อมูลจากฐานข้อมูลและอัปเดต การเปลี่ยนแปลงต่างๆ ลงสู่ฐานข้อมูลรวมถึงการทำงานกับข้อมูลในหน่วยความจำที่ดึงมาจากฐานข้อมูล เช่น การกลั่นกรอง ค้นหา และเพิ่มข้อมูล เพื่อต้องการทราบว่าระบบลงทะเบียนเวลาโดยใช้ลายนิ้วมือที่ได้จัดทำขึ้นสามารถนำไปใช้ประโยชน์ในองค์กร

4.2 การทดลองการทำงานของระบบบันทึกเวลาโดยใช้ลายนิ้วมือ

หน้าที่การทำงานของระบบบันทึกเวลาเวลาโดยใช้ลายนิ้วมือคือการรับข้อมูลมาจาก MS Access โดยทำการติดต่อฐานข้อมูลแบบ ODBC และจะส่งข้อมูลไปแสดงผลในระบบบันทึกเวลาโดยใช้ลายนิ้วมือซึ่งตัวของระบบจะสามารถเพิ่ม และแก้ไขข้อมูลที่อยู่ใน MS Access ได้ ดังนั้นการทดลองใช้งานระบบลงทะเบียนเวลาโดยใช้ลายนิ้วมือจึงสามารถทดลองได้

4.3 การทดสอบเครื่องสแกนลายนิ้วมือ DigitalPersona U.are.U 4000

เนื่องจากโครงการนี้ได้ใช้เครื่องสแกนลายนิ้วมือแบบกึ่งสำเร็จรูป ดังนั้นจึงจำเป็นต้องตรวจสอบการทำงานของเครื่องสแกนลายนิ้วมือก่อนว่ามีปัญหาหรือชำรุดเสียหายหรือไม่ โดยในการทดสอบเครื่องได้ใช้โปรแกรมซึ่งทางบริษัทผู้จำหน่ายได้กำหนดมาให้

4.3.1 การทดลอง

เป็นการนำตัวเครื่องสแกนมาทำการทดสอบการทำงานของตัวเครื่องว่ามีประสิทธิภาพที่พร้อมใช้งานหรือไม่

ขั้นตอนที่ 1 นำตัวเครื่องสแกน รุ่น U.are U 4000 เชื่อมต่อกับคอมพิวเตอร์โดยผ่าน USB Port ทำการเรียกโปรแกรมที่ทางบริษัทให้มาเพื่อทำการทดสอบร่วมกับตัวเครื่องสแกน

ขั้นตอนที่ 2 ทำการลงทะเบียนระหว่างตัวโปรแกรมกับตัวเครื่องสแกน โดยต้องกรอกรายละเอียดของตัวเครื่องที่มีมาให้ ได้แก่

1. Sensor Serial Number : C2A57B3F-0234-3E4F-83B8-EE8F14CA70F0

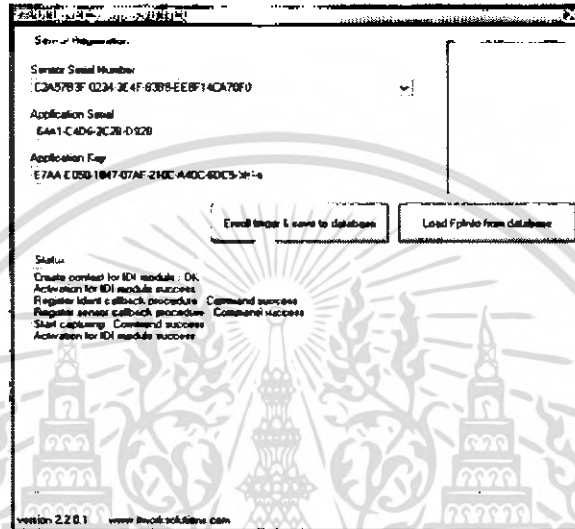
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. Application Serial : 64A1-C4D6-2C2B-D92B

3. Application Key : E7AA-E050-1847-07AF-210C-A40C-6DC5-3816

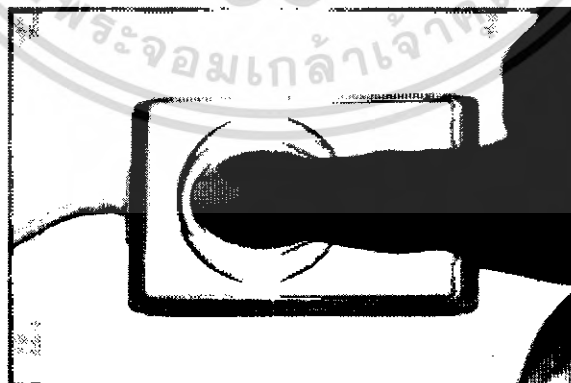
เมื่อทำการกรอกข้อมูลเรียบร้อยแล้วก็ทำการกดที่ปุ่ม Activate sensor & Start capturing

ดังรูปที่ 4.1



รูปที่ 4.1 หน้าต่างการลงทะเบียนตัวเครื่องสแกนกับโปรแกรม

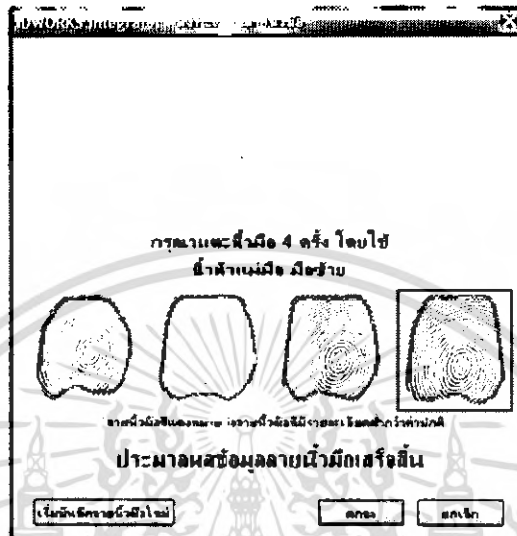
ขั้นตอนที่ 3 เริ่มการบันทึกลายนิ้วมือโดยการกดปุ่ม Enroll finger & save to database เพื่อทำการบันทึกลายนิ้วมือลงไปเก็บไว้ในฐานข้อมูลจำลอง ดังรูปที่ 4.2 และ รูปที่ 4.3



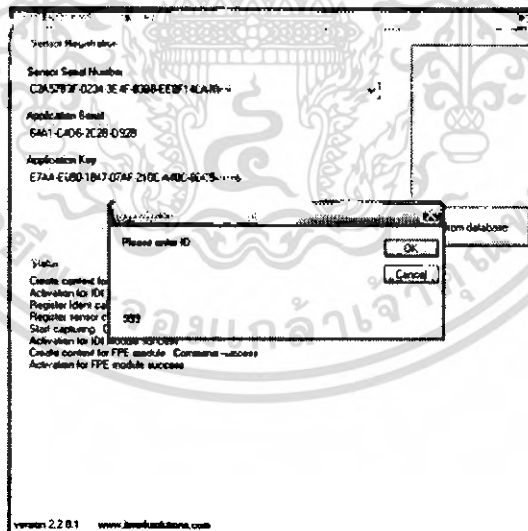
รูปที่ 4.2 การสแกนลายนิ้วมือเพื่อเก็บข้อมูลลายนิ้วมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 4 เมื่อทำการสแกนลายนิ้วมือเรียบร้อยแล้วทั้ง 4 ครั้ง จะมีคำสั่งให้ทำการบันทึกลายนิ้วมือลงในฐานข้อมูลจำลอง ดังรูปที่ 4.4



รูปที่ 4.3 การสแกนลายนิ้วมือเพื่อเก็บข้อมูลลายนิ้วมือ



รูปที่ 4.4 การบันทึกข้อมูลลายนิ้วมือลงในฐานข้อมูลจำลอง

ขั้นตอนที่ 5 ตรวจสอบข้อมูลที่ทำการบันทึกไว้ในฐานข้อมูลว่ามีข้อมูลที่ทำการบันทึกไว้แล้วมีความถูกต้องหรือไม่ ดังรูปที่ 4.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	PK_EmpFingerPrintID	Emp_ID	FingerIndex	IndexValue	Features
	8 999		4		145 byte Long binary
*	(AutoNumber)				

รูปที่ 4.5 ข้อมูลที่บันทึกในฐานข้อมูล

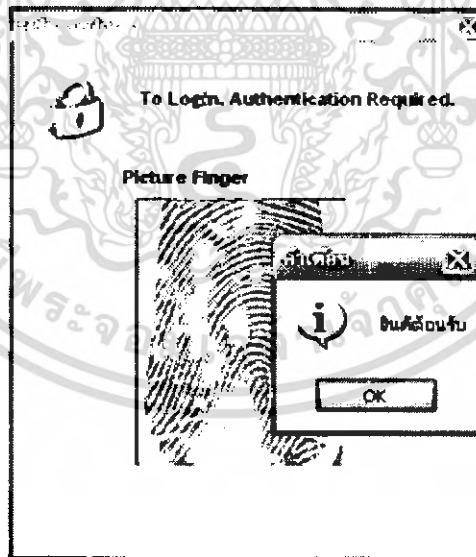
4.3.2 ผลการทดลอง

สามารถใช้งานเครื่องสแกนลายนิ้วมือได้อย่างสมบูรณ์ และ การสแกนลายนิ้วมือยังสามารถบันทึกข้อมูลของลายนิ้วมือและยืนยันข้อมูลได้ถูกต้องได้อย่างถูกต้อง

4.4 การเข้าสู่ระบบบันทึกเวลาโดยใช้ลายนิ้วมือของเจ้าหน้าที่

4.4.1 การทดลอง

เป็นการตรวจสอบระบบการสแกนลายนิ้วมือ เพื่อการเข้าไปใช้งานในระบบบันทึกเวลาโดยใช้ลายนิ้วมือ ในส่วนการทำงานนี้จะเป็นส่วนสำหรับการสแกนเพื่อเข้าไป Login ของเจ้าหน้าที่เท่านั้นดังรูปที่ 4.6



รูปที่ 4.6 การสแกนลายนิ้วมือ

4.4.2 ผลการทดลอง

ในส่วนของการทำงานนี้สามารถเข้าไป Login ด้วยการสแกนลายนิ้วมือของเจ้าหน้าที่เพื่อเข้าไปทำการแก้ไขข้อมูลในส่วนต่างๆของระบบบันทึกเวลาโดยใช้ลายนิ้วมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 การเข้าสู่ระบบบันทึกและแก้ไขข้อมูลชื่อของพนักงาน

4.5.1 การทดลอง

ขั้นตอนที่ 1 เมื่อเจ้าหน้าที่ทำการ Login เสร็จแล้วจะปรากฏหน้าต่าง Time Attention By Fingerprint และเลือกที่หน้าต่าง Employee Profile ดังรูปที่ 4.7



รูปที่ 4.7 หน้าต่าง Time Attention By Fingerprint

ขั้นตอนที่ 2 จากรูปที่ 4.8 เป็นรูปหน้าต่างของการบันทึกข้อมูล แก้ไขข้อมูลต่างๆของพนักงานโดยประกอบไปด้วยข้อมูล รหัสพนักงาน รหัสประชาชน วันเดือนปีเกิด ชื่อ นามสกุล ที่อยู่ หมายเลขโทรศัพท์ แผนก ตำแหน่ง ฐานเงินเดือน รูปพนักงาน ระบุว่าเป็นพนักงานหรือเจ้าหน้าที่ และทำการสแกนลายนิ้วมือของพนักงานหรือเจ้าหน้าที่เพื่อทำการเก็บข้อมูลลายนิ้วมือ ซึ่งเจ้าหน้าที่จะทำการบันทึกและแก้ไขข้อมูลของพนักงานและคลิกปุ่ม Update

Employee Project Form

Employee Information Management

Required Information:

Employee ID : 48936214
 Population ID : 1883261452
 Date of Birth : 7 กรกฎาคม 2528

Changeable Information:

First name : Eakbedin Last name: gingeeon
 Address : 78/1 no.2
 Phone number: 0815875231 Salary: 15000
 Department: Engineer Position: Software Engineer

Picture
 Brown

Status
 Admin
 User

Save Finger Print Update Find New

รูปที่ 4.8 การบันทึกและแก้ไขข้อมูลพนักงาน

4.5.2 ผลการทดลอง

เมื่อทำการอัปเดตข้อมูลใหม่ของพนักงานและเจ้าหน้าที่ ข้อมูลต่างๆที่ทำการแก้ไขหรือบันทึกใหม่ก็จะไปเก็บไว้ในฐานข้อมูลของระบบได้อย่างสมบูรณ์ถูกต้อง

4.6 การกำหนดกะการทำงานของพนักงาน

4.6.1 การทดลอง

เมื่อทำการบันทึกข้อมูลและลายมือของพนักงานเรียบร้อยแล้ว หลังจากนั้นเจ้าหน้าที่จะทำการกำหนดกะการทำงานของพนักงานเลือกว่าจะให้พนักงานแต่ละคนทำงานในกะการทำงานใด ถ้าจะเลือกการทำงานเป็นคนที่ให้ทำการเลือกที่ช่องสี่เหลี่ยมหน้าชื่อของพนักงานคนนั้น เมื่อทำการเลือกพนักงานแล้วก็เลือกกะการทำงานว่าจะทำช่วงไหนและทำการกด Save ดังรูปที่ 4.9

รูปที่ 4.9 หน้าต่าง Work Schedule

4.6.2 ผลการทดลอง

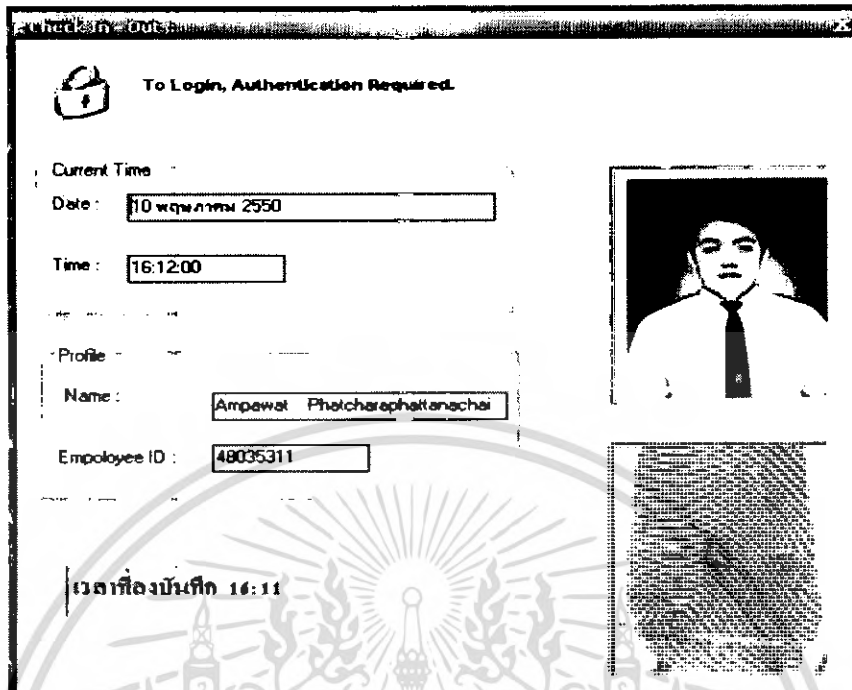
เมื่อทดลองทำการกำหนดกะการทำงานแล้วสามารถบันทึกข้อมูล แก้ไขข้อมูลกะการทำงานที่ฐานข้อมูลได้อย่างถูกต้องตามที่ตั้งไว้

4.7 การตรวจเช็คการลงเวลาเข้า-ออก ของพนักงาน

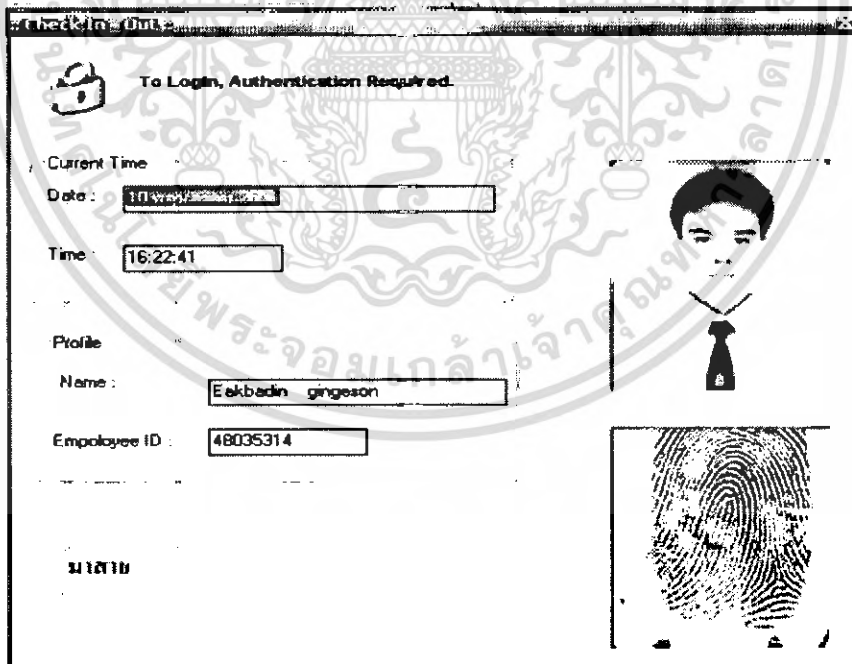
4.7.1 การทดลอง

ขั้นตอนที่ 1 เมื่อทำการบันทึกข้อมูลต่างๆของพนักงานและกะการทำงานแล้วลองทำการทดสอบโดยการเปิดหน้าต่าง Check In-Out ให้พนักงานวางนิ้วมือลงบนเครื่องสแกน หน้าต่างการใช้งานจะปรากฏ ชื่อพนักงาน รหัสพนักงาน รูปของพนักงาน รูปลายนิ้วมือ เวลาปัจจุบันและเวลาที่บันทึก จะปรากฏดังรูปที่ 4.10

ขั้นตอนที่ 2 กรณีที่พนักงานทำการสแกนลายนิ้วมือลงเวลาเกินกว่าเวลาที่กำหนดไว้ จะปรากฏผลที่หน้าต่างการทำงานว่า มาสาย ดังรูปที่ 4.11



รูปที่ 4.10 หน้าต่างการลงเวลาเข้า ของพนักงาน



รูปที่ 4.11 หน้าต่างการลงเวลาเข้า กรณีลงเวลาเกินกว่าเวลาที่กำหนดไว้

ขั้นตอนที่ 3 เมื่อพนักงานทำการสแกนลายนิ้วมือลงเวลาออกจากงาน จะปรากฏหน้าตงการใช้งานที่แสดงเวลาออกจากการทำงาน ดังรูปที่ 4.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.12 หน้าต่างการสแกนลายนิ้วมือลงเวลาออก

4.7.2 ผลการทดลอง

ตารางที่ 4.1 ผลการบันทึกวันที่และเวลาการเข้าทำงานของพนักงาน

พนักงานคนที่	วันที่และเวลาเข้าทำงาน	วันที่และเวลาที่บันทึกในฐานข้อมูล
1	2007 / 5 / 10 16 :11	2007 / 5 / 10 16 :11
2	2007 / 5 / 10 16 :13	2007 / 5 / 10 16 :13
3	2007 / 5 / 10 16 :15	2007 / 5 / 10 16 :15
4	2007 / 5 / 10 16 :18	2007 / 5 / 10 16 :18
5	2007 / 5 / 10 16 :20	2007 / 5 / 10 16 :20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 (ต่อ) ผลการบันทึกวันที่และเวลาการเข้าทำงานของพนักงาน

6	2007 / 5 / 10 16 :25	2007 / 5 / 10 16 :25
7	2007 / 5 / 10 16 :28	2007 / 5 / 10 16 :28
8	2007 / 5 / 10 16 :30	2007 / 5 / 10 16 :30
9	2007 / 5 / 10 16 :35	2007 / 5 / 10 16 :35
10	2007 / 5 / 10 16 :45	2007 / 5 / 10 16 :45

ตารางที่ 4.2 ผลการบันทึกวันที่และเวลาการออกจากทำงานของพนักงาน

พนักงานคนที่	วันที่และเวลาออกจากทำงาน	วันที่และเวลาที่บันทึกในฐานข้อมูล
1	2007 / 5 / 10 16 :12	2007 / 5 / 10 16 :12
2	2007 / 5 / 10 16 :14	2007 / 5 / 10 16 :41
3	2007 / 5 / 10 16 :16	2007 / 5 / 10 16 :16
4	2007 / 5 / 10 16 :19	2007 / 5 / 10 16 :19
5	2007 / 5 / 10 16 :21	2007 / 5 / 10 16 :21
6	2007 / 5 / 10 16 :26	2007 / 5 / 10 16 :26
7	2007 / 5 / 10 16 :29	2007 / 5 / 10 16 :29
8	2007 / 5 / 10 16 :31	2007 / 5 / 10 16 :31
9	2007 / 5 / 10 16 :36	2007 / 5 / 10 16 :36
10	2007 / 5 / 10 16 :48	2007 / 5 / 10 16 :48

4.8 การตรวจเช็คการมาสายและการขาดงานของพนักงาน

4.8.1 การทดลอง

ขั้นตอนที่ 1 หน้าต่างการตรวจเช็คการมาสายและการขาดงานของพนักงาน การเข้าหน้าต่งการตรวจเช็คการมาสายและการขาดงานของพนักงานทำได้โดย การเข้า Login โดยใช้การสแกนลายนิ้วมือของเจ้าหน้าที่ ตรวจสอบโดยการเลือกเดือนและปีที่ต้องการจะดูรายงาน และถ้าต้องการดูรหัสของพนักงานคนใดคนหนึ่งหรือทั้งหมด สามารถเลือกดูวันมาสายโดยการเลือก Report Late ถ้าต้องการดูทั้งเดือนให้กดที่ปุ่ม All Month ถ้าต้องการดูเป็นวันให้กดที่ปุ่ม All Day ดังรูปที่ 4.13

ขั้นตอนที่ 2 หน้าต่งการตรวจเช็คการขาดงานของพนักงาน พนักงานคนใดคนหนึ่งหรือทั้งหมดสามารถเลือกดูวันที่ขาดงาน โดยการเลือก Report Stop ขั้นตอนคล้ายกับขั้นตอนที่ 1 ที่ดังรูปที่ 4.14

Please select the date of logs you would like to find

Date Period Selection:

From: May Year: 2007 Emp: All

Report Stop Report Late All Month All Day

Main Report

11/5/259

รายงานสรุปพนักงานที่มาสาย

Employee ID	Date	จำนวนที่มาสาย
48035261	10-พ.ค.-07	1
1	10-พ.ค.-07	1
48035314	13-พ.ค.-07	1

Current Page No.: 1 Total Page No.: 1 Zoom Factor: 100%

รูปที่ 4.13 หน้าต่างรายงานสรุปพนักงานที่มาสาย

Please select the date of logs you would like to find

Date Period Selection:

From: May Year: 2007 Emp: All

Report Stop Report Late All Month All Day

Main Report

11/5/259

รายงานสรุปพนักงานที่ขาดงาน

Employee ID	จำนวนที่ขาดงาน
48035261	1
48035311	1
48035258	1

Current Page No.: 1 Total Page No.: 1 Zoom Factor: 100%

รูปที่ 4.14 หน้าต่างรายงานสรุปพนักงานที่ขาดงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

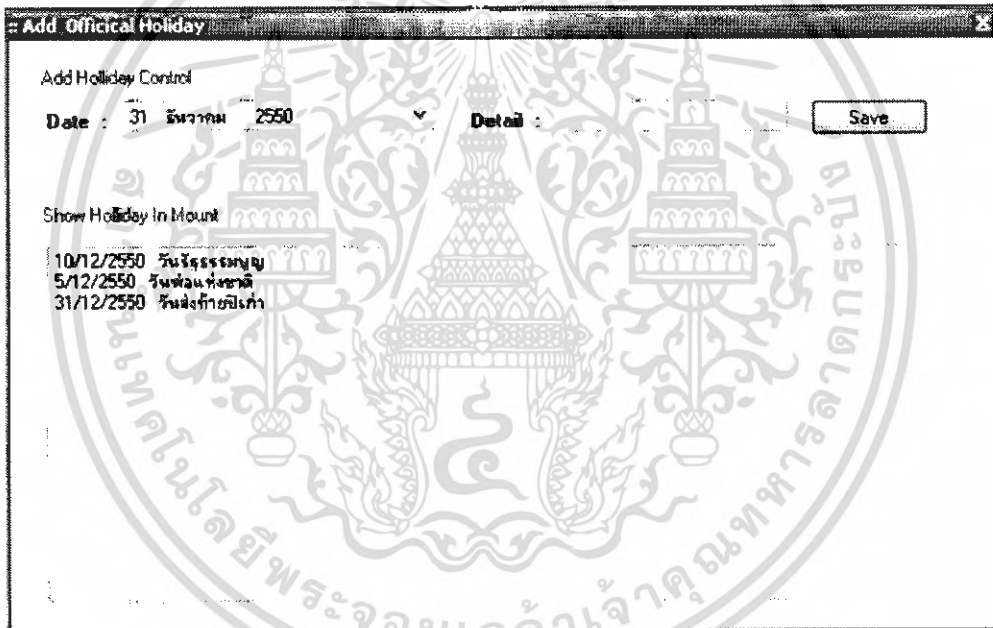
4.8.2 ผลการทดลอง

จากการตรวจสอบการลงเวลาของพนักงานที่มาสายและขาดงานสามารถแสดงรายงานสรุปพนักงานที่ขาดงานและมาสายได้อย่างถูกต้องตามที่ได้ทำการสแกนบันทึกเวลา

4.9 การกำหนดวันหยุดของบริษัท

4.9.1 การทดลอง

เลือกปุ่ม Add Official Holiday เพื่อเพิ่มและลบข้อมูลวันหยุดของบริษัท โดยกำหนดวันเดือนปีทีในช่อง Date และหมายเหตุที่ช่อง Detail และทำการกดปุ่ม Save ข้อมูลที่เราทำการบันทึกเพิ่มเข้าไปที่ช่อง Show Holiday In Mount และไปเก็บที่ฐานข้อมูล ดังรูปที่ 4.15



รูปที่ 4.15 หน้าต่างแสดงข้อมูลวันหยุดของบริษัท

4.9.2 ผลการทดลอง

เมื่อทำการทดสอบการเพิ่มข้อมูลและลบข้อมูลวันหยุดของบริษัท หน้าต่างการทำงานนี้สามารถแสดงรายงานข้อมูลวันหยุดของบริษัทได้อย่างถูกต้องตามที่ได้ทำการบันทึกไว้

บทที่ 5

บทสรุป

5.1 สรุป

การสร้างระบบบันทึกเวลาโดยใช้ลายนิ้วมือนั้น จะมีส่วนประกอบหลักที่สำคัญอยู่ 2 ส่วน ก็คือส่วนของเครื่องสแกนลายนิ้วมือ และอีกส่วนหนึ่งก็คือส่วนของโปรแกรมและฟังก์ชันการใช้งานทั้งหมด ของระบบบันทึกเวลาโดยใช้ลายนิ้วมือ

ในส่วนของการออกแบบระบบบันทึกเวลาโดยใช้ลายนิ้วมือ จะประกอบด้วย ฟอर्मลืออกอิน ฟอर्मซึ่งการออกแบบฟอर्मทั้งหมดนี้ก็สามารถนำไปเขียนโปรแกรมใช้งานได้ตามที่ออกแบบไว้

ส่วนของเครื่องสแกนลายนิ้วมือใช้รุ่น DigitalPersona U.are.U 4000 ที่สำเร็จรูปมาใช้ โดยการส่งข้อมูลผ่านทาง USB Port โดยใช้การเขียนโปรแกรมติดต่อกับระบบส่งงานโดยการยืนยันลายนิ้วมือ

ในการทดลองใช้โปรแกรมระบบบันทึกเวลาโดยใช้ลายนิ้วมือ ซึ่งผลการทดลองใช้งานกับนักศึกษาจำนวน 20 คนในการบันทึกเวลาไม่มีปัญหาข้อผิดพลาดเลย เครื่องสแกนลายนิ้วมือรุ่น DigitalPersona U.are.U 4000 สามารถอ่านข้อมูลลายนิ้วมือของพนักงานได้อย่างถูกต้องแม่นยำมีความผิดพลาดน้อย ในส่วนของโครงการระบบบันทึกเวลาโดยใช้ลายนิ้วมือนี้ สามารถนำไปประยุกต์แก้ไขใช้งานกับงานอื่นๆ ได้อีกตามที่ต้องการ

5.2 ปัญหาและวิธีการแก้ไข

จากการดำเนินการสร้างและทดสอบระบบบันทึกเวลาโดยใช้ลายนิ้วมือปรากฏว่าเกิดปัญหาหลายประการ ซึ่งสามารถสรุปได้ดังนี้

1. โปรแกรมระบบบันทึกเวลาโดยใช้ลายนิ้วมือเป็นโปรแกรมแบบ Standalone ไม่สามารถใช้โปรแกรมผ่านระบบเน็ตเวิร์คได้

วิธีการแก้ไข ทำการเปลี่ยนระบบฐานข้อมูลจาก Access เป็น SQL Sever

2. เครื่องสแกนลายนิ้วมือมีการประมวลผลช้า

วิธีการแก้ไข ทำการเปลี่ยนเครื่องสแกนลายนิ้วมือที่มีประสิทธิภาพการทำงานที่สูงกว่านี้

5.3 แนวทางการพัฒนา

โครงการนี้สามารถนำเครื่องสแกนลายนิ้วมือไปประยุกต์ใช้งาน ในการเขียนโปรแกรมการตรวจสอบการค้นหาข้อมูลตัวบุคคล และสามารถนำไปใช้กับระบบรักษาความปลอดภัยต่างๆ ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนในโครงการนี้จะใช้เครื่องสแกนลายนิ้วมือเป็นสื่อ ในการรับข้อมูลและยืนยันข้อมูลที่เข้ามา ซึ่งลายนิ้วมือนั้นอาจเกิดการปลอมแปลงได้

ดังนั้นเพื่อเป็นการป้องกันการปลอมแปลงทำได้โดยการออกแบบโปรแกรมอาจเพิ่มฟังก์ชันต่างๆ เข้าไปเพื่อความเหมาะสมแก่ การใช้งานหรืออีกวิธีหนึ่งคือการนำเครื่องมืออย่างอื่นมาเป็นสื่อในการรับและยืนยันข้อมูลได้เช่น เครื่องสแกนใบหน้า เครื่องสแกนม่านตา เป็นต้น การนำเครื่องสแกนลายนิ้วมือไปประยุกต์ใช้งานเกี่ยวกับงานระบบอื่นๆ ได้อีก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

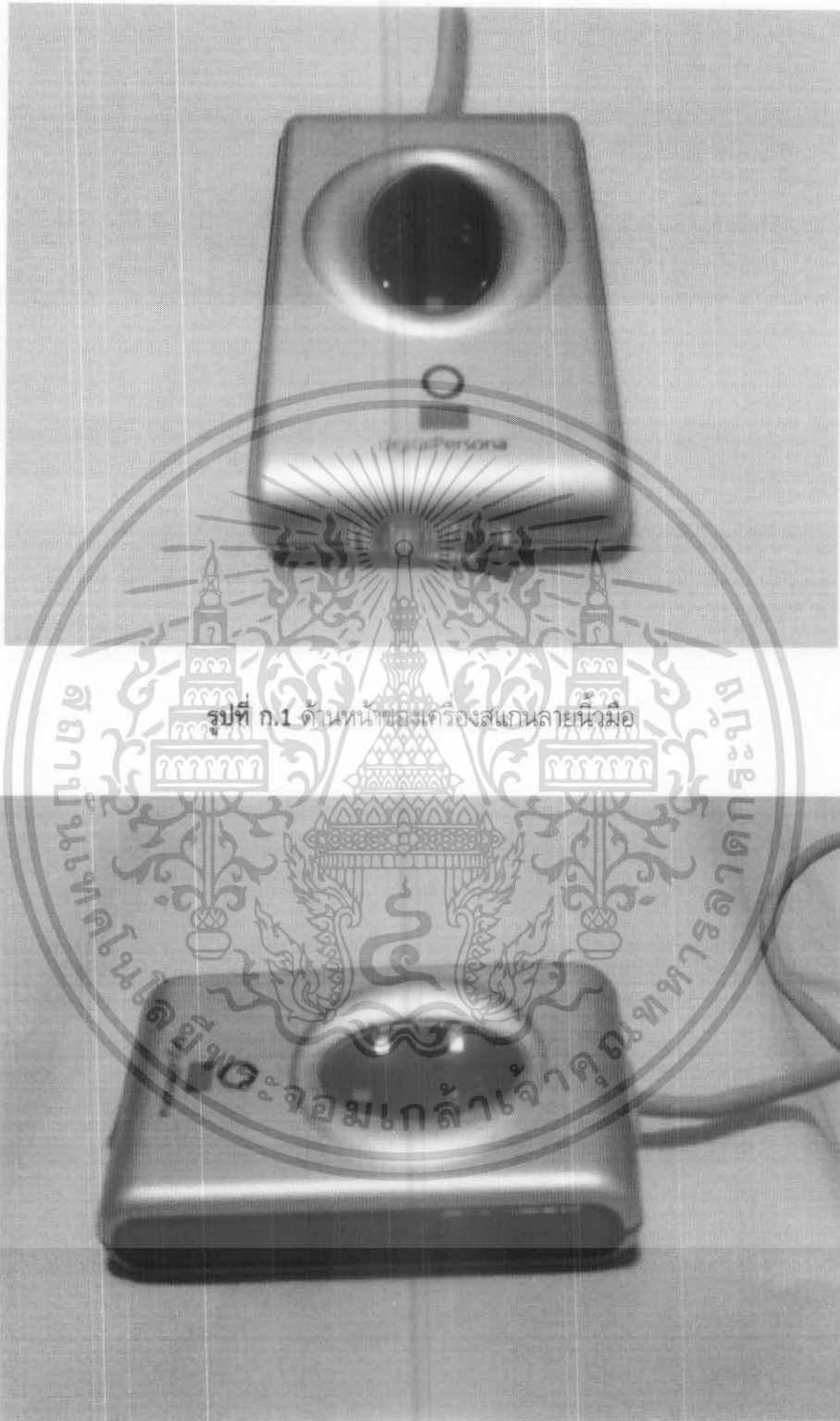
- นันทน์ แขวงโสภา. 2548. **อินไซด์ Access 2003**. กรุงเทพฯ: โปรวิชั่น.
- พร้อมเลิศ หล่อวิจิตร. 2549. **คู่มือเรียน Visual Basic 2005**. กรุงเทพฯ: โปรวิชั่น.
- วราภรณ์ โกวิทรวงูร. 2543. **ระบบฐานข้อมูลและการออกแบบ**. กรุงเทพฯ: พิกซ์อักษร
- วิเชียร เปรมชัยสวัสดิ์. 2548. **ระบบฐานข้อมูล**. พิมพ์ครั้งที่ 7. กรุงเทพฯ: ศ.ส.ท
- ศุภชัย สมพานิช. 2546. **สร้างระบบงานฐานข้อมูลด้วย Visual Basic. NET**. กรุงเทพฯ: ไอดีซี
- ศุภชัย สมพานิช. **เรียนรู้และฝึกฝนการเขียนโปรแกรมใช้งานฐานข้อมูลด้วย Visual Basic ฉบับมืออาชีพ**.
นนทบุรี: ไอดีซี



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.1 ด้านหน้าของเครื่องสแกนลายนิ้วมือ

รูปที่ ก.2 ด้านข้างของเครื่องสแกนลายนิ้วมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ข
รายการอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ข.1 รายการอุปกรณ์ของระบบบันทึกเวลาโดยใช้ลายนิ้วมือ

ชื่ออุปกรณ์	รายละเอียด	จำนวน
เครื่องสแกนลายนิ้วมือ	- SigialPerson U.are U 4000	1 ตัว
เครื่องคอมพิวเตอร์	- ใช้ระบบปฏิบัติการของตัว Windows XP - มีหน่วยประมวลผลกลาง (ซีพียู) ความเร็วอย่างต่ำ 1.6 กิกะไบต์ - มีพื้นที่ว่างฮาร์ดดิสก์ ๕00 เมกกะไบต์ - มีหน่วยความจำข้อมูลชั่วคราว 256 เมกกะไบต์ - การ์ดแสดงผลอย่างต่ำ 32 เมกกะไบต์ - มีช่อง USB Port วางอย่างน้อย 1 ช่อง - ใช้จอคอมพิวเตอร์อย่างต่ำ 15 นิ้ว - โปรแกรม (Microsoft Visual Basis 2005) และ (Microsoft Access 2003)	1 เครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ค
รายละเอียดและคุณสมบัติของอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

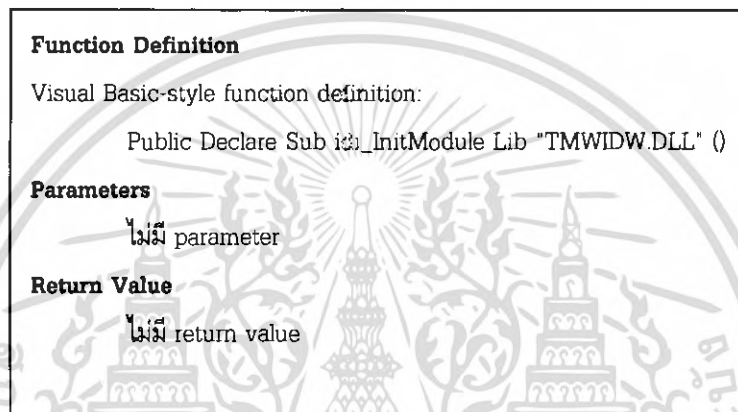
1. IDWORKS Integrator – ส่วนการประมวลผลลายนิ้วมือ

ทุกฟังก์ชันจะ return ค่าเป็น IDI_R_OK หากทำงานได้สำเร็จ ยกเว้นหากได้หมายเหตุไว้ว่าเป็น
อย่างอื่น

1.1 รายละเอียดฟังก์ชัน (Function Definitions)

1. idi_InitModule

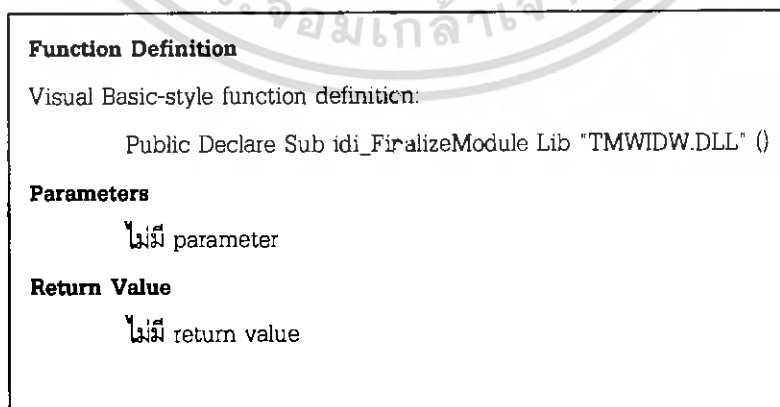
ก่อนการใช้งานโมดูล จะต้องเรียกฟังก์ชันนี้ก่อนที่จะมีการใช้งานฟังก์ชันอื่นๆ เสมอ ดังรูปที่ ค.1



รูปที่ ค.1 รายละเอียดฟังก์ชันของ idi_InitModule

2. idi_FinalizeModule

เมื่อเลิกใช้งานโมดูล ต้องเรียกฟังก์ชันนี้เพื่อเคลียร์ข้อมูลต่างๆ ที่อาจค้างอยู่ในโมดูล ดังรูปที่ ค.2



รูปที่ ค.2 รายละเอียดฟังก์ชันของ idi_FinalizeModule

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. idi_CreateContext

รายละเอียดฟังก์ชันของ idi_CreateContext ดังรูปที่ ค.3

<p>Function Definition</p> <p>Visual Basic-style function definition</p> <pre>Public Declare Function idi_CreateContext Lib "TMWIDW.DLL" _ (ByRef AProcessorContext As Long) _As Long</pre> <p>Parameters</p> <p>AProcessorContext</p> <p>ถ้าการสร้าง context สำเร็จ AProcessorContext จะ return ค่า Context ออกมา</p> <p>Return Value</p> <p>ค่าที่ส่งกลับมาจะเป็น IDI_RCODs โดยมีความหมายดังนี้</p> <ul style="list-style-type: none"> - IDI_R_OK : การทำงานเสร็จสมบูรณ์ - IDI_R_MODULENOTINITED : ยังไม่มีการ Initialize Module - IDI_R_INVALIDCONTEXT : ค่า Context ไม่ถูกต้อง

รูปที่ ค.3 รายละเอียดฟังก์ชันของ idi_CreateContext

4. idi_CloseContext

ฟังก์ชัน idi_CloseContext ใช้สำหรับปิดการทำงานของ context ดังรูปที่ ค.4

<p>Function Definition</p> <p>Visual Basic-style function definition:</p> <pre>Public Declare Function idi_CloseContext Lib "TMWIDW.DLL" _ (ByVal AProcessorContext As Long) _As Long</pre> <p>Parameters</p> <p>AProcessorContext</p> <p>Context ที่ต้องการเลิกการใช้งาน</p>

รูปที่ ค.4 รายละเอียดฟังก์ชันของ idi_CloseContext

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Return Value

- ค่าที่ส่งกลับมาจะเป็น IDI_RCODE โดยมีความหมายดังนี้
- IDI_R_OK : การทำงานเสร็จสมบูรณ์
 - IDI_R_MODULENOTINITED : ยังไม่มีการ Initialize Module
 - IDI_R_INVALIDCONTEXT : ค่า Context ไม่ถูกต้อง

รูปที่ ค.4 (ต่อ) รายละเอียดฟังก์ชันของ idi_CloseContext**5. idi_ActivateSensor**

ฟังก์ชันนี้ใช้สำหรับการใช้งาน Sensor โดยการส่งค่า parameter ของ sensor ที่ต้องการใช้งาน เข้าไปดังรูปที่ ค.5

Function Definition

Visual Basic-style function definition:

```
Public Declare Function idi_ActivateSensor Lib "TMWIDW.DLL" _
    (ByVal AProcessorContext As Long, _ByVal ASensorType As Integer, _
    ByVal ASensorSerialNo As String, _ByVal AAppSerial As String, _
    ByVal AAppKey As String ...) As Long
```

Parameters

AProcessorContext

IDWORKS Processor Context

ASensorType

ประเภทของ Sensor

(สามารถดูได้จากค่าคงที่ที่ขึ้นต้นด้วย IDI_ST_)

ASensorSerialNo

เลข Serial ของ Sensor (null-terminated)

AAppSerial

เลข IDWORKS Integrator Serial (null-terminated)

AAppKey

เลข IDWORKS Integrator Key (null-terminated)

รูปที่ ค.5 รายละเอียดฟังก์ชันของ idi_ActivateSensor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Return Value

- ค่าที่ส่งกลับมากจะเป็น IDI_RCODE โดยมีความหมายดังนี้
- IDI_R_OK : การทำงานเสร็จสมบูรณ์
 - IDI_R_MODULENOTINITED : ยังไม่มีการ Initialize Module
 - IDI_R_INVALIDCONTEXT : ค่า Context ไม่ถูกต้อง

รูปที่ ค.5 (ต่อ) รายละเอียดฟังก์ชันของ idi_ActivateSensor**6. idi_DeactivateSensor**

ใช้สำหรับการยกเลิกการใช้งานหมายเลขเครื่องอ่านลายนิ้วมือ ตามหมายเลขเครื่องอ่านลายนิ้วมือที่กำหนดดังรูปที่ ค.6

Function Definition

Visual Basic-style function definition.

```
Public Declare Function idi_DeactivateSensor Lib "TMWIDW.DLL" _
    (ByVal AProcessorContext As Long, _ByVal ASensorType As Long, _
    ByVal ASensorSerialNo As String _) As Long
```

Parameters

AProcessorContext

IDWORKS Processor Context

ASensorType

ประเภทของ Sensor (สามารถดูได้จากค่าคงที่ที่ขึ้นต้นด้วย IDI_ST_)

ASensorSerialNo

เลข Serial ของ Sensor (null-terminated)

Return Value

- ค่าที่ส่งกลับมากจะเป็น IDI_RCODE โดยมีความหมายดังนี้
- IDI_R_OK : การทำงานเสร็จสมบูรณ์
 - IDI_R_MODULENOTINITED : ยังไม่มีการ Initialize Module
 - IDI_R_INVALIDCONTEXT : ค่า Context ไม่ถูกต้อง

รูปที่ ค.6 รายละเอียดฟังก์ชันของ idi_DeactivateSensor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. idi_StartCapturing

เมื่อเรียกฟังก์ชันนี้ โปรแกรมจะเริ่มการทำงานของ sensor เพื่อรับลายนิ้วมือจากผู้ใช้ หากมีการแตะลายนิ้วมือ ลายนิ้วมือจะถูกประมวลผลโดยอัตโนมัติ ผลการประมวลผลของลายนิ้วมือ (เช่น "รู้จักลายนิ้วมือ" หรือ "ไม่รู้จัก") จะถูกส่งในรูปของ Message ผ่านไปทาง callback function ที่ได้ register ไว้โดยฟังก์ชัน `idi_RegisterIdentEventProc` ดังรูปที่ ค.7

Function Definition

Visual Basic-style function definition:

```
Public Declare Function idi_StartCapturing Lib "TMWIDW.DLL" _
    (ByVal AProcessorContext As Long, _ByVal ASensorSerialNo As String_
    ) As Long
```

Parameters

AProcessorContext
IDWORKS Processor Context

ASensorSerialNo
หมายเลข Sensor ที่ต้องการให้เริ่มการทำงาน
หมายเหตุ : Sensor ที่สามารถใช้ได้ จะต้องผ่านการ Activate โดยใช้ฟังก์ชัน `idi_ActivateSensor` ก่อน

Return Value

ค่าที่ส่งกลับมาจะเป็น `IDI_RCODE` โดยมีความหมายดังนี้

- `IDI_R_OK` : การทำงานเสร็จสมบูรณ์
- `IDI_R_MODULENOTINITED` : ยังไม่มีการ Initialize Module
- `IDI_R_INVALIDCONTEXT` : ค่า Context ไม่ถูกต้อง

รูปที่ ค.7 รายละเอียดฟังก์ชันของ `idi_StartCapturing`

8. idi_StopCapturing

ฟังก์ชันนี้จะหยุดการทำงานของ Sensor ตัวที่ระบุดังรูปที่ ค.8

Function Definition

Visual Basic-style function definition:

```
Public Declare Function idi_StopCapturing Lib "TMWIDW.DLL" _
    (ByVal AProcessorHandle As Long, _ByVal ASensorSerialNo As String _
    ) As Long
```

Parameters

AProcessorContext
IDWORKS Processor Context

ASensorSerialNo
หมายเลข Sensor ที่ต้องการหยุดการทำงาน

Return Value

ค่าที่ส่งกลับมาจะเป็น IDI_RCODE โดยมีความหมายดังนี้

- IDI_R_OK : การทำงานเสร็จสมบูรณ์
- IDI_R_MODULENOTINITED : ยังไม่มีการ Initialize Module
- IDI_R_INVALIDCONTEXT : *A Context ไม่ถูกต้อง

รูปที่ ค.8 รายละเอียดฟังก์ชันของ idi_StopCapturing

9. idi_RegisterSensorEventProc

การรีจิสเตอร์ callback function เพื่อรับ event ของเครื่องอ่านลายนิ้วมือดังรูปที่ ค.9

Function Definition

Visual Basic-style function definition:

```
Public Declare Function idi_RegisterSensorEventProc Lib "TMWIDW.DLL" _
    (ByVal AProcessorContext As Long, _ByVal AEventProc As Long, _
    ByVal ARefParam As Any _) As Long
```

รูปที่ ค.9 รายละเอียดฟังก์ชันของ idi_RegisterSensorEventProc

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Parameters

AProcessorContext

IDWORKS Processor Context

AEventProc

Pointer หรือ Address ของ Ca..back function ที่จะถูกเรียกเมื่อมี event เกี่ยวกับเครื่องอ่านลายนิ้วมือ

ARefParam

เป็นค่าที่จะถูกส่งกลับมากับ Message ใน IDI_SensorEventMsg.RefParam (มีประโยชน์เพื่อใช้อ้างอิง เช่น สามารถตั้งค่าเป็น Pointer ของ object ที่ต้องการได้)

Return Value

ค่าที่ส่งกลับมาจะเป็น IDI_RCODE โดยมีความหมายดังนี้

- IDI_R_OK : การทำงานเสร็จสมบูรณ์
- IDI_R_MODULENOTINITED ยังไม่มีการ Initialize Module
- IDI_R_INVALIDCONTEXT : ค่า Context ไม่ถูกต้อง

รูปที่ ๑.๑ (ต่อ) รายละเอียดฟังก์ชันของ idi_RegisterSensorEventProc

10. idi_UnregisterSensorEventProc

ยกเลิกการรีจิสเตอร์ allback function ที่ระบุดังรูปที่ ๑.10

Function Definition

Visual Basic-style function definition:

```
Public Declare Function idi_UnregisterSensorEventProc Lib "TMWIDW.DLL" _
    ByVal AProcessorContext As Long, _ByVal AEventProc As Long, _) As Long
```

Parameters

AProcessorContext

IDWORKS Processor Context

AEventProc

Callback function ที่ต้องการยกเลิกการรับ event

รูปที่ ๑.10 รายละเอียดฟังก์ชันของ idi_UnregisterSensorEventProc

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Return Value

ค่าที่ส่งกลับมาจะเป็น IDI_RCODE โดยมีความหมายดังนี้

- IDI_R_OK : การทำงานเสร็จสมบูรณ์
- IDI_R_MODULENOTINITED : ยังไม่มีการ Initialize Module
- IDI_R_INVALIDCONTEXT : ค่า Context ไม่ถูกต้อง

รูปที่ ค.10 (ต่อ) รายละเอียดฟังก์ชันของ idi_UnregisterSensorEventProc

11. idi_RegisterIdentEventProc

Register ฟังก์ชัน callback สำหรับรับ event การประมวลผลหลายนิ้วมืองดรูปที่ ค.11

Function Definition

Visual Basic-style function definition:

```
Public Declare Function idi_RegisterIdentEventProc Lib "TMWIDW.DLL" _
    (ByVal AProcessorContext As Long, _ByVal AEventProc As Long, _ ByRef
    ARefParam As Any _) As Long
```

Parameters

AProcessorContext

IDWORKS Processor Context

AEventProc

Callback function ที่จะถูกเรียกเมื่อมีการประมวลผลหลายนิ้วมือง

ARefParam

เป็นค่าที่จะถูกส่งกลับมากับ Message ใน IDI_IdentEventMsg.RefParam (มีประโยชน์เพื่อใช้อ้างอิง เช่น สามารถตั้งค่าเป็น Pointer ของ object ที่ต้องการได้)

Return Value

ค่าที่ส่งกลับมาจะเป็น IDI_RCODE โดยมีความหมายดังนี้

- IDI_R_OK : การทำงานเสร็จสมบูรณ์
- IDI_R_MODULENOTINITED : ยังไม่มีการ Initialize Module
- IDI_R_INVALIDCONTEXT : ค่า Context ไม่ถูกต้อง

รูปที่ ค.11 รายละเอียดฟังก์ชันของ idi_RegisterIdentEventProc

12. idi_UnregisterIdentEventProc

ยกเลิกการ Register ที่ได้ทำไปแล้วจากการเรียกฟังก์ชัน idi_RegisterIdentEventProc ดังรูปที่ ค.12

Function Definition

Visual Basic-style function definition:

```
Public Declare Function idi_UnregisterIdentEventProc Lib
"TMWIDW.DLL" (ByVal AProcessorContext As Long, _
ByVal AEventProc As Long _) As Long
```

Parameters

AProcessorContext
IDWORKS Processor Context

AEventProc
Callback function ที่ต้องการยกเลิกการ register

Return Value

ค่าที่ส่งกลับมาเป็น IDI_RCODE โดยมีความหมายดังนี้

- IDI_R_OK : การทำงานเสร็จสมบูรณ์
- IDI_R_MODULENOTINITED : ยังไม่มีการ initialize module
- IDI_R_INVALIDCONTEXT : ค่า Context ไม่ถูกต้อง

รูปที่ ค.12 รายละเอียดฟังก์ชันของ idi_UnregisterIdentEventProc

13. idi_fp_Add

เพิ่มลายนิ้วมือเข้าไปในฐานข้อมูลชั่วคราวภายในของโมดูลลายนิ้วมือ ที่เพิ่มเข้าไปจะทำให้โมดูลรู้จักลายนิ้วมือนั้นเมื่อมีการประมวลผลลายนิ้วมื่อดังรูปที่ ค.13

Function Definition

Visual Basic-style function definition:

```
Public Declare Function idi_fp_Add Lib "TMWIDW.DLL" _
    (ByVal AProcessorContext As Long, _ByRef AFPInfo As FP_FPInfo _) As Long
```

Parameters

AProcessorContext

IDWORKS Processor Context ;

AFPInfoPtr

Pointer ที่ชี้ไปยังข้อมูลหลายนิ้วมือ (FP_FPInfo) ที่ต้องการเพิ่มเข้าไปในฐานข้อมูลภายใน

Return Value

ค่าที่ส่งกลับมาจะเป็น IDI_RCODE โดยมีความหมายดังนี้

- IDI_R_OK : การทำงานเสร็จสมบูรณ์
- IDI_R_MODULENOTINITED : ยังไม่มีการ Initialize Module
- IDI_R_INVALIDCONTEXT : ค่า Context ไม่ถูกต้อง

รูปที่ ค.13 รายละเอียดฟังก์ชันของ idi_fp_Add

14. idi_fp_ClearAll

เป็นการล้างฐานข้อมูลภายในของโมดูลดังรูปที่ ค.14

Function Definition

Visual Basic-style function definition:

```
Public Declare Function idi_fp_ClearAll Lib "TMWIDW.DLL" _
    (ByVal AProcessorContext As Long _) As Long
```

Parameters

AProcessorContext

IDWORKS Processor Context

รูปที่ ค.14 รายละเอียดฟังก์ชันของ idi_fp_Add

Return Value
ค่าที่ส่งกลับมาจะเป็น IDI_RCODE โดยมีความหมายดังนี้
- IDI_R_OK : การทำงานเสร็จสมบูรณ์
- IDI_R_MODULENOTINITED : ยังไม่มีการ Initialize Module
- IDI_R_INVALIDCONTEXT : ค่า Context ไม่ถูกต้อง

รูปที่ ค.14 (ต่อ) รายละเอียดฟังก์ชันของ `idi_fp_Add`

1.2 ประเภทข้อมูล (Data Types)

1. IDI_CONTEXT

IDI_CONTEXT เป็นประเภทข้อมูล 32-bit สำหรับ Processor Context ที่เสมือนกับเป็นหน่วยในการประมวลผลข้อมูล ซึ่งสามารถสร้างได้หลาย context แยกกัน ในการใช้งานแต่ละครั้งดังรูปที่ ค.15

Data-type Definition
Visual Basic-style data type definition: ใช้ประเภท Long

รูปที่ ค.15 ประเภทข้อมูลของ IDI_CONTEXT

2. IDI_IDENTEVENTMSG

IDI_IDENTEVENTMSG เป็นประเภทข้อมูลแบบ structure ที่จะถูกส่งกลับมาให้กับ callback function ที่ได้ทำการregister ไว้โดยฟังก์ชัน `idi_RegisterIdentEventProc` ดังรูปที่ ค.16

Data-type Definition
Visual Basic-style data type definition:
Public Type IDI_IdentEventMsg
ProcessorContext As Long
MsgID As Long
MsgParam As Long
RefParam As Long
End Type

รูปที่ ค.16 ประเภทข้อมูลของ IDI_IDENTEVENTMSG

หมายเหตุ

ค่าที่ถูกส่งมาใน MsgParam จะขึ้นกับค่า MsgID ซึ่งจะมีความหมายดังตารางต่อไปนี้

ตารางที่ ค.1 ค่า MsgID กับค่า MsgParam ของฟังก์ชัน IDI_IDENTEVENTMSG

ค่า MsgID	ความหมายของค่าใน MsgParam
IDI_ID_FINGERPRINTMATCHED	Pointer ที่ชี้ไปยัง IDI_IDENTEVENTMATCHSTRUCT
IDI_ID_FINGERPRINTNOTMATCHED	0

3. IDI_IDEVENTTYPE

IDI_IDEVENTTYPE เป็นประเภทข้อมูลขนาด 32-bit สำหรับเก็บชนิดของ Identification Message ที่จะถูกส่งกลับมายังฟังก์ชันที่ได้ทำการ register ไว้โดยใช้ฟังก์ชัน RegisterIdentEventProc ซึ่งประเภทข้อมูลนี้จะใช้ใน IDI_IDEVENTMSG.MsgID ดังรูปที่ ค.17

Data-type Definition
Visual Basic-style data type definition:
ใช้ประเภท Long

รูปที่ ค.17 ประเภทข้อมูลของ IDI_IDEVENTTYPE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. IDI_IDENTEVENTPARAM

IDI_IDENTEVENTPARAM เป็นประเภทข้อมูลขนาด 32-bit ที่จะถูกส่งมากับ Identification Message IDI_IDENTEVENTMSG ซึ่งความหมายจะแตกต่างกันสำหรับแต่ละ message ขึ้นอยู่กับค่า IDI_IDENTEVENTMSG.MsgID ดังรูปที่ ค.18

Data-type Definition

Visual Basic-style data type definition:

ใช้ประเภท Long

รูปที่ ค.18 ประเภทข้อมูลของ IDI_IDENTEVENTPARAM

5. IDI_REFPARAM

IDI_REFPARAM เป็นข้อมูล Pointer ที่ไว้ใช้อ้างอิงกับการเรียกใช้ฟังก์ชัน `idi_RegisterIdentEventProc` ได้โดยค่าที่ส่งกลับมากจะเป็นค่าเดียวกันกับค่า `ARefParam` ที่ส่งเข้าไปเมื่อเรียกฟังก์ชัน `idi_RegisterIdentEventProc` ดังรูปที่ ค.19

Data-type Definition

Visual Basic-style data type definition:

ใช้ประเภท Long

รูปที่ ค.19 ประเภทข้อมูลของ IDI_REFPARAM

6. IDI_IDENTEVENTPROC

IDI_IDENTEVENTPROC เป็นการประกาศประเภทของฟังก์ชันที่ใช้สำหรับการ callback เมื่อมีหน้าต่างมีมือมาทำการแตะนิ้วมือ โดย pointer ของฟังก์ชันที่ส่งเข้าไปเป็นพารามิเตอร์ `AEventProc` ของ `idi_RegisterIdentEventProc` จะต้องมี arguments เหมือนกับที่ประกาศใน IDI_IDENTEVENTPROC ดังรูปที่ ค.20

<p>Data-type Definition</p> <p>Visual Basic-style data type definition</p> <p>Public Sub IdentProc(ByRef AIdentEventMsg As IDI_IdentEventMsg)</p> <p>Parameters</p> <p>AIdentEventMsgPtr Pointer ที่ชี้ไปยัง IDI_IDENTEVENTMSG ซึ่งบอกรายละเอียดของ Message</p>

รูปที่ ค.20 ประเภทข้อมูลของ IDI_IDENTEVENTPROC

7. IDI_IDENTEVENTMATCHEDSTRUCT

ในกรณีที่มีการแตะนิ้วมือและพบข้อมูล IDI_IDENTEVENTMATCHEDSTRUCT จะเป็นข้อมูลที่ถูส่งกลับมากับ IDI_IDENTEVENTMSG ซึ่ง IDI_IDENTEVENTMSG.MsgParam จะเป็น pointer ที่ชี้ไปยังข้อมูลดังกล่าวดังรูปที่ ค.21

<p>Data-type Definition</p> <p>Visual Basic-style data type definition</p> <p>Public Type IDI_IdentEventMatchedStruct</p> <p>FPInfoPtr As Long</p> <p>MatchingScore As Integer</p> <p>End Type</p> <p>Members</p> <p>FPInfoPtr เป็น Pointer ที่ชี้ไปยังข้อมูล FP_FPINFO ที่บอกรายละเอียดของลายนิ้วมือ</p> <p>MatchingScore เป็นค่าคะแนนความเหมือนกันของลายนิ้วมือ</p>

รูปที่ ค.21 ประเภทข้อมูลของ IDI_IDENTEVENTMATCHEDSTRUCT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. IDI_SENSOREVENTMSG

IDI_SENSOREVENTMSG เป็นประเภทข้อมูลแบบ structure ที่จะถูกส่งกลับมาให้กับ callback function ที่ได้ทำการ register ไว้โดยฟังก์ชัน idi_RegisterSensorEventProc ดังรูปที่ ค.22

Data-type Definition

Visual Basic-style data type definition:

```
Public Type IDI_SensorEventMsg
    ProcessorContext As Long
    MsgID As Long MsgParam As Long
    RefParam As Long
End Type
```

Members

ProcessorContext
Context ที่เป็นผู้ส่ง message มาให้

MsgID
ประเภทของ Message

MsgParam
ข้อมูลของ Message นั้นๆ

RefParam
เลขอ้างอิงที่ถูกส่งเข้ามากับการเรียก idi_RegisterSensorEventProc ก่อนหน้านี้

รูปที่ ค.22 ประเภทข้อมูลของ IDI_SENSOREVENTMSG

หมายเหตุ

ค่าที่ถูกส่งมาใน MsgParam จะขึ้นกับค่า MsgID ซึ่งจะมีความหมายดังตารางต่อไปนี้

ตารางที่ ค.2 ค่า MsgID กับค่า MsgParam ของฟังก์ชัน IDI_SENSOREVENTMSG

ค่า MsgID	ความหมายของค่าใน MsgParam
IDI_SE_IMAGEACQUIRED	Pointer ที่ชี้ไปยัง IDI_SENSORIMAGEACQUIREDSTRUCT
IDI_SE_IMAGEPROCESSED	Pointer ที่ชี้ไปยัง IDI_SENSORIMAGEACQUIREDSTRUCT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. IDI_SENSOREVENTPROC

IDI_SENSOREVENTPROC เป็นการประกาศประเภทของฟังก์ชัน ที่ใช้สำหรับการ callback เมื่อมี event เกี่ยวกับเครื่องอ่านลายนิ้วมือ เช่น ได้รับรูปลายนิ้วมือ ประเภทฟังก์ชันนี้จะใช้กับฟังก์ชัน IDI_RegisterSensorEventProc โดย pointer หรือ address ของฟังก์ชันที่ใช้ส่งเข้าไปเป็นพารามิเตอร์ AEventProc จะต้องมี arguments เหมือนกับที่ประกาศใน IDI_SENSOREVENTPROC ดังรูปที่ ค.23

Data-type Definition
Visual Basic-style data type definition: Public Sub SensorProc(ByRef ASensorEventMsg As IDI_SensorEventMsg)
Parameters ASensorEventMsgPtr Pointer ที่ชี้ไปยัง IDI_SENSOREVENTMSG ซึ่งบอกรายละเอียดของ Message

รูปที่ ค.23 ประเภทข้อมูลของ IDI_SENSOREVENTPROC

10. IDI_SENSOREVENTIMAGEACQUIREDSTRUCT

เมื่อมีการแตะลายนิ้วมือ และเครื่องอ่านลายนิ้วมืออ่านรูปลายนิ้วมือสำเร็จ โมดูลจะส่ง event ให้กับ function ที่ได้ทำการวีจิสเตอร์ไว้โดย idi_RegisterSensorEvent ซึ่งใน MsgParam ของ event ดังกล่าวจะเป็นค่า pointer ของ IDI_SENSOREVENTIMAGEACQUIREDSTRUCT ดังรูปที่ ค.24

Data-type Definition
Visual Basic-style data type definition: Public Type IDI_SENSOREVENTIMAGEACQUIREDSTRUCT ImageWidth As Long ImageHeight As Long ImageBitmapPtr As Long End Type

รูปที่ ค.24 ประเภทข้อมูลของ IDI_SENSOREVENTIMAGEACQUIREDSTRUCT

Members

ImageWidth

ความกว้างของรูปลายนิ้วมือ

ImageHeight

ความสูงของรูปลายนิ้วมือ

ImageBitmapPtr

Pointer ที่ชี้ไปยังข้อมูล gray-scale bitmap ของลายนิ้วมือ โดยมีขนาดเท่ากับ ImageWidth x ImageHeight และมีรูปแบบเป็น 8-bit grayscale

รูปที่ ค.24 (ต่อ) ประเภทข้อมูลของ IDI_SENSOREVENTIMAGEACQUIREDSTRUCT

11. FP_FPINFO

FP_FPINFO เป็นประเภทข้อมูลแบบ structure ที่ใช้สำหรับการส่งผ่านข้อมูลลายนิ้วมือเข้าไปยังฐานข้อมูลชั่วคราวใน Processor Context ดังรูปที่ ค.25

Data-type Definition

Visual Basic-style data type definition:

Public Type FP_FPInfo

FingerprintIndexingValue As Long

FingerIndex As Long

FingerprintRawDataPtr As Long

FingerprintRawDataSize As Long

Tag As Long

End Type

Members

FingerprintIndexingValue

เป็นค่า Indexing ของลายนิ้วมือ ที่ใช้สำหรับ optimize การค้นหาลายนิ้วมือ

FingerIndex

เป็นหมายเลขบอกนิ้ว ซึ่งจะไล่ตั้งแต่ 0 คือนิ้วก้อยมือซ้าย ถึง 9 คือนิ้วก้อยมือขวา

รูปที่ ค.25 ประเภทข้อมูลของ FP_FPINFO

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FingerprintRawDataPtr

เป็น Pointer ที่ชี้ไปยัง memory buffer ที่เก็บข้อมูลลายนิ้วมือ

FingerprintRawDataSize

เป็นขนาดของ Memory buffer ที่ชี้โดย FingerprintRawDataPtr

Tag

เป็นเลขกำกับลายนิ้วมือ ซึ่งผู้พัฒนาสามารถใช้อ้างอิงได้ เช่น สามารถเก็บเลข ID ของบุคคลในฐานข้อมูล

รูปที่ ค.25 (ต่อ) ประเภทข้อมูลของ FP_FPINFO

1.3 ค่าคงที่ (Constants)

1. ค่าคงที่สำหรับ IDI_RCODE

ตารางที่ ค.3 ค่าคงที่สำหรับ IDI_RCODE

ชื่อค่าคงที่	ค่า	ความหมาย
IDI_R_OK	0	การทำงานเสร็จสิ้น ไม่มีข้อผิดพลาด
IDI_R_GENERALFAILURE	-1	มีข้อผิดพลาดทั่วไป
IDI_R_INIT_FAILED	-2	ไม่สามารถเริ่มการใช้งานได้
IDI_R_MODULENOTINITED	-3	เรียกฟังก์ชันโดยไม่มี Initialize Module ก่อน
IDI_R_SENSORNOTFOUND	-4	ไม่พบเครื่องอ่านลายนิ้วมือตามที่กำหนด
IDI_R_SENSORALREADYACTIVATED	-5	เครื่องอ่านลายนิ้วมือได้ถูก Activate ไปก่อนหน้านั้นแล้ว
IDI_R_INVALIDSENSORTYPE	-6	ประเภทเครื่องอ่านลายนิ้วมือไม่ถูกต้อง
IDI_R_INVALIDREGISTRATION	-7	การ Activate ไม่ถูกต้อง
IDI_R_INVALIDCONTEXT	-16	Context ไม่ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ค่าคงที่สำหรับ IDI_IDENTEVENTTYPE

ตารางที่ ค.4 ค่าคงที่สำหรับ IDI_IDENTEVENTTYPE

ชื่อค่าคงที่	ค่า	ความหมาย
IDI_ID_FINGERPRINTMATCHED	0	พบลายนิ้วมือ
IDI_ID_FINGERPRINTNOTMATCHED	1	ไม่พบลายนิ้วมือ

3. ค่าคงที่สำหรับ IDI_SENSORTYPE

ตารางที่ ค.5 ค่าคงที่สำหรับ IDI_SENSORTYPE

ชื่อค่าคงที่	ค่า	ความหมาย
IDI_ST_UAREU	1	เครื่องอ่านลายนิ้วมือ DigitalPersona U.are.U 4000

4. ค่าคงที่สำหรับ IDI_SENSOREVENTTYPE

ตารางที่ ค.6 ค่าคงที่สำหรับ IDI_SENSOREVENTTYPE

ชื่อค่าคงที่	ค่า	ความหมาย
IDI_SE_IMAGEACQUIRED	1	Event รับรูปจากเครื่องอ่านลายนิ้วมือ
IDI_SE_IMAGEPROCESSED	2	Event ประมวลผลรูปลายนิ้วมือเสร็จสิ้น

2. ID-WORKS Integrator – ส่วนข้อมูลเครื่องอ่านลายนิ้วมือ

เป็นไลบรารีสำหรับดึงข้อมูลของเครื่องอ่านลายนิ้วมือที่ต่ออยู่ทั้งหมด

2.1 รายละเอียดฟังก์ชัน (Function Definitions)

1. isl_IsModuleInited

ตรวจสอบว่า module ได้ถูก initialize แล้วหรือไม่ดังรูปที่ ค.26

Function Definition

Visual Basic-style function definition:

```
Public Declare Function isl_IsModuleInited Lib "ITWSSL.DLL" () As Long
```

Parameters

ไม่มี parameters

Return Value

- ISL_R_OK : โมดูลได้ initialize แล้ว
- ISL_R_MODULENOTINITED : โมดูลยังไม่ได้ initialize

รูปที่ ค.26 รายละเอียดฟังก์ชันของ isl_IsModuleInited

2. isl_InitModule

เริ่มการทำงานของโมดูลดังรูปที่ ค.27

Function Definition

Visual Basic-style function definition:

```
Public Declare Function isl_InitModule Lib "ITWSSL.DLL" () As Long
```

Parameters

ไม่มี parameters

Return Value

- ISL_R_OK : การเริ่มต้น
- ISL_R_MODULENOTINITED : ไม่สามารถเริ่มต้นโมดูลได้

รูปที่ ค.27 รายละเอียดฟังก์ชันของ isl_IsModuleInited

3. isl_FinalizeModule

เลิกการทำงานของโมดูล เมื่อเลิกการใช้งานโมดูล (เช่น ในช่วงก่อนปิดโปรแกรม) จะต้องเรียกฟังก์ชันนี้เพื่อเคลียร์ข้อมูลต่างๆ ที่อาจค้างอยู่ในโมดูลดังรูปที่ ค.28

Function Definition

Visual Basic-style function definition:

```
Public Declare Function isl_FinalizeModule Lib "ITWSSL.DLL" () As Long
```

Parameters

ไม่มี parameters

Return Value

- ISL_R_OK : สำเร็จ
- ISL_R_MODULENOTINITED ไม่สามารถเริ่มต้นโมดูลได้

รูปที่ ค.28 รายละเอียดฟังก์ชันของ isl_FinalizeModule

4. isl_GetDeviceCount

ใช้สำหรับนับจำนวน sensor ทั้งหมดที่ต่ออยู่ดังรูปที่ ค.29

Function Definition

Visual Basic-style function definition

```
Public Declare Function isl_GetDeviceCount Lib "ITWSSL.DLL"  
(ByRef ASensorCount As Long _) As Long
```

Parameters

ASensorCount

เป็นจำนวน sensor ที่ส่งกลับมา

Return Value

- ISL_R_OK : สำเร็จ จำนวน sensor จะถูกใส่เข้ามาใน ASensorCount
- ISL_R_MODULENOTINITED : ฟังก์ชัน isl_InitModule ยังไม่ถูกเรียก

รูปที่ ค.29 รายละเอียดฟังก์ชันของ isl_GetDeviceCount

5. isl_GetDeviceList

ใช้สำหรับดึงข้อมูลของ sensor ทั้งหมดที่ต่ออยู่ดังรูปที่ ค.30

<p>Function Definition</p> <p>Visual Basic-style function definition:</p> <pre>Public Declare Function isl_GetDeviceList Lib "ITWSSL.DLL" _ (ByVal ASensorList As Long, _ByVal ASensorCount As Long _) As Long</pre> <p>Parameters</p> <p>ASensorList เป็น pointer ของ array (แบบ C/C++) ของ FP_FPInfo ซึ่งต้องมีความยาวอย่างน้อยเท่ากับ ASensorCount</p> <p>ASensorCount คือจำนวน sensor ที่ต้องการดึงข้อมูล สามารถหาได้จากฟังก์ชัน isl_GetDeviceCount</p> <p>Return Value</p> <ul style="list-style-type: none"> - ISL_R_OK : สำเร็จ จำนวน sensor จะถูกใส่ค่ามาใน ASensorCount - ISL_R_MODULENOTINITED : ฟังก์ชัน isl_InitModule ยังไม่ถูกเรียก

รูปที่ ค.30 รายละเอียดฟังก์ชันของ isl_GetDeviceList

2.2 ประเภทข้อมูล (Data Types)

1. ISL_RC CODE

ISL_RC CODE เป็นประเภทข้อมูล 32 บิต ที่เป็นค่าที่ส่งกลับมาเมื่อเรียกฟังก์ชัน เพื่อบอกถึงผลลัพธ์ของฟังก์ชันนั้นๆ ดังรูปที่ ค.31

<p>Data-type Definition</p> <p>Visual Basic-style data type definition:</p> <p>ใช้ระเภท Long</p>

รูปที่ ค.31 ประเภทข้อมูลของ ISL_RC CODE

2. ISL_DEVICETYPE

ISL_DEVICETYPE เป็นประเภทข้อมูล 32 บิตที่ใช้สำหรับระบุประเภทของเครื่องอ่านลายนิ้วมือดังรูปที่ ค.32

Data-type Definition

Visual Basic-style data type definition:

ใช้ประเภท Long

รูปที่ ค.32 ประเภทข้อมูลของ ISL_DEVICETYPE

3. ISL_DEVICEID

ISL_DEVICEID เป็นประเภทข้อมูล 64 บิต ที่ใช้เป็นเลขประจำตัวของเครื่องอ่านลายนิ้วมือดังรูปที่ ค.33

Data-type Definition

Visual Basic-style data type definition:

ใน Visual Basic จะไม่มี Integer ขนาด 64 bit แต่ใน header file ของ Visual Basic ที่มาพร้อมกับ SDK จะแก้ปัญหาใน structure ที่ใช้ประเภทข้อมูลแบบนี้แล้ว

รูปที่ ค.33 ประเภทข้อมูลของ ISL_DEVICEID

4. ISL_CHAR

ISL_CHAR เป็นประเภทข้อมูล 8 บิต ที่ใช้เก็บตัวอักษร จะใช้เก็บข้อความทั่วไปในโมดูลดังรูปที่ค.34

Data-type Definition

Visual Basic-style data type definition:

ใช้ประเภท Byte

รูปที่ ค.34 ประเภทข้อมูลของ ISL_CHAR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ISL_SENSORINFO

ISL_SENSORINFO เป็นประเภทข้อมูลแบบ structure ที่ใช้เก็บข้อมูลเกี่ยวกับรายละเอียดต่างๆ ของเครื่องอ่านลายนิ้วมือดังรูปที่ ค.35

Data-type Definition

Visual Basic-style data type definition:

```

Public Type ISL_SENSORINFO
    DeviceType As Long
    DeviceId As Long
    DeviceName(ISL_MAXDEVICENAMELEN) As Byte
    DeviceSerial(ISL_MAXDEVICESERIALLEN) As Byte
End Type

```

Members

DeviceType
ประเภทของเครื่องอ่านลายนิ้วมือ

DeviceID
หมายเลขประจำเครื่องอ่านลายนิ้วมือ

DeviceName
ชื่อเครื่องอ่านลายนิ้วมือ เช่น DigitalPersona U.are.U 4000

DeviceSerial
S/N ประจำเครื่องอ่านลายนิ้วมือ

รูปที่ ค.35 ประเภทข้อมูลของ ISL_SENSORINFO

2.3 ค่าคงที่ (Constants)

1. ค่าคงที่ทั่วไป

ตารางที่ ค.7 ค่าคงที่ทั่วไปของส่วนข้อมูลเครื่องอ่านลายนิ้วมือ

ชื่อค่าคงที่	ค่า	ความหมาย
ISL_MAXDEVICENAMELEN	64	เรียกฟังก์ชันโดยไม่มี Initialize Module ก่อน
ISL_MAXDEVICSERIALLEN	64	ไม่พบเครื่องอ่านลายนิ้วมือตามที่กำหนด

2. ค่าคงที่สำหรับ ISL_RCODE

ตารางที่ ค.8 ค่าคงที่สำหรับ ISL_RCODE ของส่วนข้อมูลเครื่องอ่านลายนิ้วมือ

ชื่อค่าคงที่	ค่า	ความหมาย
ISL_R_OK	0	การทำงานเสร็จสิ้น ไม่มีข้อผิดพลาด
ISL_R_MODULENOTINITED	-1	มีข้อผิดพลาดทั่วไป
ISL_R_ALREADYINITED	-2	ไม่สามารถเริ่มการใช้งานได้

3. ค่าคงที่สำหรับ ISL_DEVICETYPE

ตารางที่ ค.9 ค่าคงที่สำหรับ ISL_DEVICETYPE ของส่วนข้อมูลเครื่องอ่านลายนิ้วมือ

ชื่อค่าคงที่	ค่า	ความหมาย
ISL_DT_UAREU	0	การทำงานเสร็จสิ้น ไม่มีข้อผิดพลาด

3. ID-WORKS Integrator – ส่วนช่วยการบันทึกลายนิ้วมือ

ทุกฟังก์ชันจะ return ค่าเป็น FPE_R_OK หากทำงานได้สำเร็จ ยกเว้นหากได้หมายเหตุไว้ว่าเป็นอย่างอื่น

3.1 รายละเอียดฟังก์ชัน (Function Definitions)

1. fpe_InitModule

เริ่มการใช้งานไลบรารีดังรูปที่ ค.36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Function Definition

Visual Basic-style function definition:

```
Public Declare Function fpe_InitModule Lib "FPENROL.DLL" () As Long
```

Parameters

ไม่มี Parameter

Return Value

- FPE_R_OK : การทำงานสำเร็จ
- FPE_R_MODULENOTINITED : ไม่สามารถเริ่มการทำงานได้

รูปที่ ค.36 รายละเอียดฟังก์ชันของ fpe_InitModule

2. fpe_FinalizeModule

เรียกเมื่อเลิกใช้งานไลบรารีดังรูปที่ ค.37

Function Definition

Visual Basic-style function definition:

```
Public Declare Function fpe_FinalizeModule Lib "FPENROL.DLL" () As Long
```

Parameters

ไม่มี Parameter

Return Value

- FPE_R_OK : การทำงานสำเร็จ
- FPE_R_MODULENOTINITED : ไม่ได้มีการ Initialize Module ก่อนหน้านี้

รูปที่ ค.37 รายละเอียดฟังก์ชันของ fpe_FinalizeModule

3. fpe_CreateContext

สร้าง context สำหรับ fingerprint enrolment ดังรูปที่ ค.38

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Function Definition

Visual Basic-style function definition:

```
Public Declare Function fpe_CreateContext Lib "FPENROL.DLL" _
    (ByRef AEnrolmentContext As Long _) As Long
```

Parameters

AEnrolmentContext

ถ้าการสร้าง Context สำเร็จ ค่าดังกล่าวจะเป็น Context สำหรับ Enrolment

Return Value

- FPE_R_OK : การทำงานสำเร็จ
- FPE_R_MODULENOTINITED : ไม่ได้มีการ Initialize Module ก่อนหน้านี้

รูปที่ ค.38 รายละเอียดฟังก์ชันของ fpe_CreateContext

4. fpe_CloseContext

ปิด context ของ fingerprint enrolment ดังรูปที่ ค.39

Function Definition

Visual Basic-style function definition:

```
Public Declare Function fpe_CloseContext Lib "FPENROL.DLL" _
    (ByVal AEnrolmentContext As Long _) As Long
```

Parameters

AEnrolmentContext

หากการสร้าง context สำเร็จ AEnrolmentContext จะถูกตั้งค่าเป็นค่า context ที่ใช้ในการลงทะเบียนลายนิ้วมือ

Return Value

- FPE_R_OK : การทำงานสำเร็จ
- FPE_R_MODULENOTINITED : ไม่ได้มีการ Initialize Module ก่อนหน้านี้

รูปที่ ค.39 รายละเอียดฟังก์ชันของ fpe_CloseContext

5. fpe_ActivateSensor

เริ่มการใช้งาน sensor ที่ต้องการดังรูปที่ ค.40

Function Definition

Visual Basic-style function definition:

```
Public Declare Function fpe_ActivateSensor Lib "FPENROL.DLL" _
  (ByVal AEnrolmentContext As Long, _ByVal ASensorType As Long, _
  ByVal ASensorSerialNo As String, _ByVal AAppSerial As String, _
  ByVal AAppKey As String _) As Long
```

Parameters

AProcessorContext
Context ที่ใช้ในการลงทะเบียนลายนิ้วมือ

ASensorType
ประเภทของ Sensor (สามารถดูได้จากค่าคงที่ที่ขึ้นต้นด้วย FPE_ST_)

ASensorSerialNo
เลข Serial ของ Sensor (null-terminated)

AAppSerial
เลข IDWORKS Integrator Serial (null-terminated)

AAppKey
เลข IDWORKS Integrator Key (null-terminated)

Return Value

- FPE_R_OK : การทำงานสำเร็จ และ Activate เครื่องอ่านลายนิ้วมือเรียบร้อย
- FPE_R_MODULENOTINITFD : ไม่ได้มีการ Initialize Module ก่อนหน้านี้

รูปที่ ค.40 รายละเอียดฟังก์ชันของ fpe_ActivateSensor

6. fpe_StartEnrolment

เริ่มการ Enroll ลายนิ้วมือ ดังรูปที่ ค.41

Function Definition

Visual Basic-style function definition:

```
Public Declare Function fpe_StartEnrolment Lib "FPENROL.DLL" _
    (ByVal AEnrolmentContext As Long, _ByVal AFingerIndex As Integer, _
    ByRef ARawDataPtr As Any, _ByVal AMaxRawDataSize As Long, _
    ByRef AOutputRawDataSize As Long, _ByRef AIndexingValue As Long _
    ) As Long
```

Parameters

AEnrolmentContext

Fingerprint Enrolment Context

ARawDataPtr

Pointer ที่ชี้ไปที่ buffer สำหรับเก็บข้อมูล binary ของลายนิ้วมือ

AMaxRawDataSize

ขนาดสูงสุดที่สามารถจัดเก็บข้อมูล binary ลงใน ARawDataPtr ได้

AOutputRawDataSize

จะเป็นขนาดที่ฟังก์ชันส่งออกมาเพื่อบอกขนาดของ buffer ที่ได้ใช้ไป หากเป็น 0 หมายถึง ผู้ใช้
กดยกเลิก

AIndexingValue

ถ้าไม่ใช่ null ฟังก์ชันจะส่งค่า FingerprintIndexingValue ออกมาให้ เพื่อนำไปจัดเก็บ

Return Value

- FPE_R_OK : การทำงานสำเร็จ
- FPE_R_MODULENOTINTITED : ไม่ได้มีการ Initialize Module ก่อนหน้านี้

รูปที่ ๑.41 รายละเอียดฟังก์ชันของ fpe_StartEnrolment

หมายเหตุ

ถ้าผู้ใช้กดยกเลิก ค่าที่ return มา ก็จะเป็น FPE_R_OK แต่ AOutputRawDataSize จะมีค่าเป็น 0 ซึ่งหมายถึง การทำงานเสร็จสมบูรณ์ แต่ไม่มีข้อมูลลายนิ้วมือ เนื่องจากผู้ใช้กดยกเลิก

3.2 ประเภทข้อมูล (Data Types)**1. FPE_RCODE**

FPE_RCODE เป็นประเภทข้อมูล 32 บิต ที่เป็นค่าที่ส่งกลับมาเมื่อเรียกฟังก์ชัน เพื่อบอกถึงผลลัพธ์ของฟังก์ชันนั้นๆ ดังรูปที่ ๑.42

Data-type Definition

Visual Basic-style data type definition:

ใช้ประเภท Long

รูปที่ ค.42 ประเภทข้อมูลของ FPE_RCODE

2. FPE_CONTEXT

FPE_CONTEXT เป็นประเภทข้อมูล 32-bit สำหรับ Context ที่เสมือนกับเป็นหน่วยย่อยในการลงทะเบียนลายนิ้วมือดังรูปที่ ค.43

Data-type Definition

Visual Basic-style data type definition:

ใช้ประเภท Long

รูปที่ ค.43 ประเภทข้อมูลของ FPE_CONTEXT

3. FPE_SENSORTYPE

FPE_SENSORTYPE เป็นประเภทข้อมูล 32 บิตที่ใช้สำหรับระบุประเภทของเครื่องอ่านลายนิ้วมือ ดังรูปที่ ค.44

Data-type Definition

Visual Basic-style data type definition:

ใช้ประเภท Long

รูปที่ ค.44 ประเภทข้อมูลของ FPE_SENSORTYPE

3.3 ค่าคงที่ (Constants)

1. ค่าคงที่ทั่วไป

ตารางที่ ค.10 ค่าคงที่ทั่วไปของส่วนช่วยการบันทึกถลายนิ้วมือ

ชื่อค่าคงที่	ค่า	ความหมาย
FPC_MAXRAWDATASIZE	4096	ขนาด memory buffer สูงสุดที่ต้องการในการลงทะเบียนลายนิ้วมือ

2. ค่าคงที่สำหรับ ISL_RCODE

ตารางที่ ค.11 ค่าคงที่สำหรับ ISL_RCODE ของส่วนช่วยการบันทึกถลายนิ้วมือ

ชื่อค่าคงที่	ค่า	ความหมาย
FPE_R_OK	0	การทำงานเสร็จสิ้น ไม่มีข้อผิดพลาด
FPE_R_INVALIDCONTEXT	-1	Context ไม่ถูกต้อง
FPE_R_GENERALFAILURE	-2	มีข้อผิดพลาดทั่วไป
FPE_R_MODULENOTINITED	-3	เรียกฟังก์ชันโดยไม่มี การ Initialize Module
FPE_R_SENSORNOTFOUND	-4	ไม่พบเครื่องอ่านลายนิ้วมือที่ระบุ
FPE_R_INTERNALERROR	-4096	มีข้อผิดพลาดภายใน module

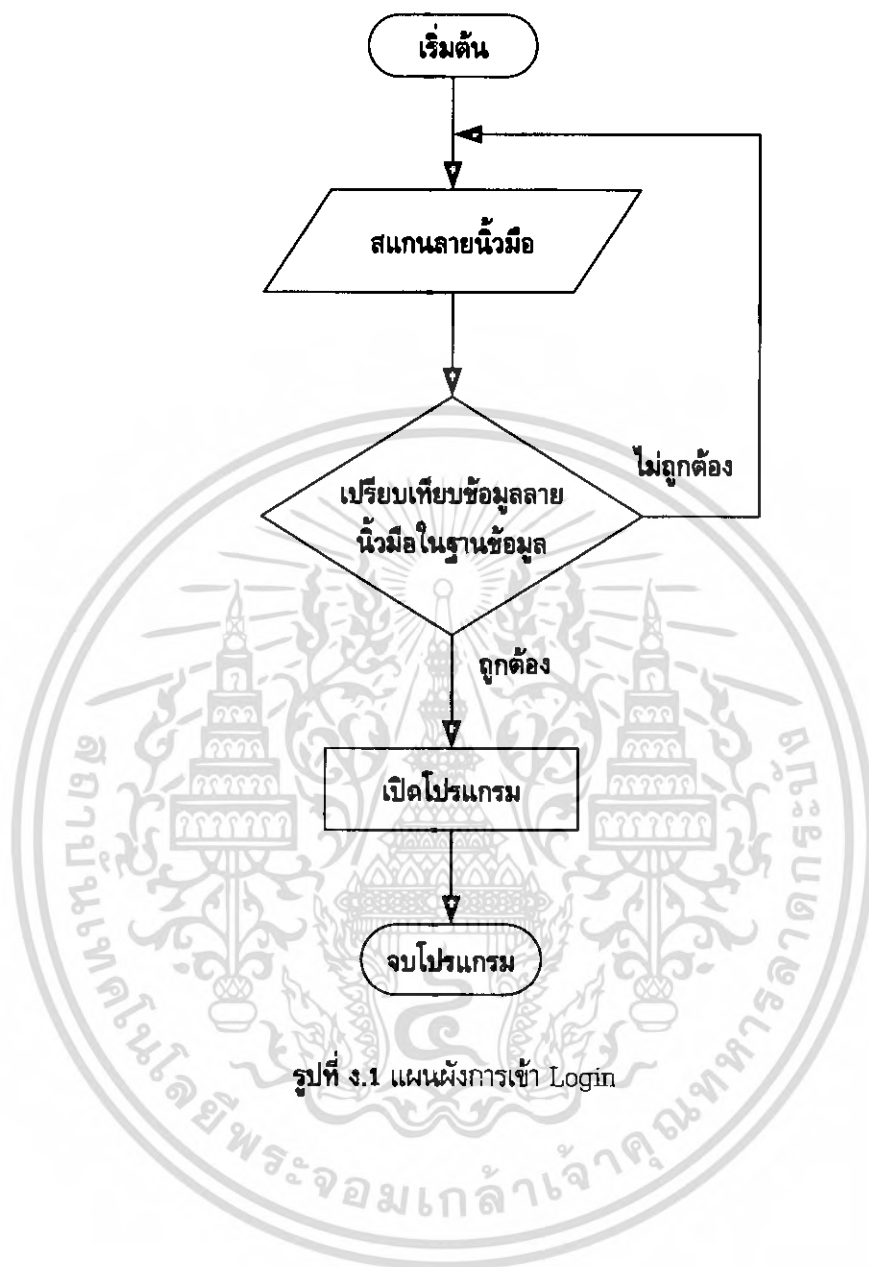
3. ค่าคงที่สำหรับ ISL_DEVICETYPE

ตารางที่ ค.12 ค่าคงที่สำหรับ ISL_DEVICETYPE ของส่วนช่วยการบันทึกถลายนิ้วมือ

ชื่อค่าคงที่	ค่า	ความหมาย
FPE_ST_UAREU	1	ประเภทของเครื่องอ่านลายนิ้วมือ

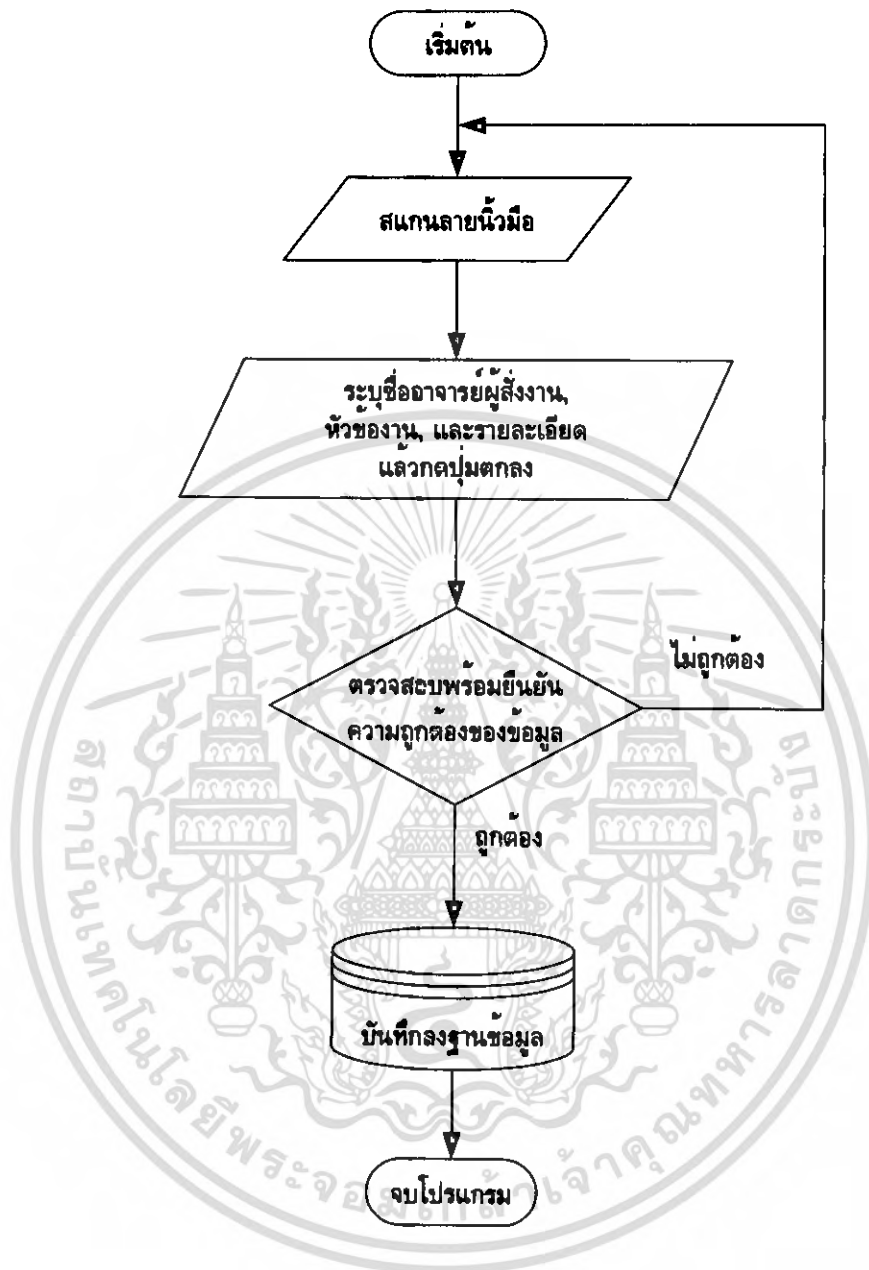


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



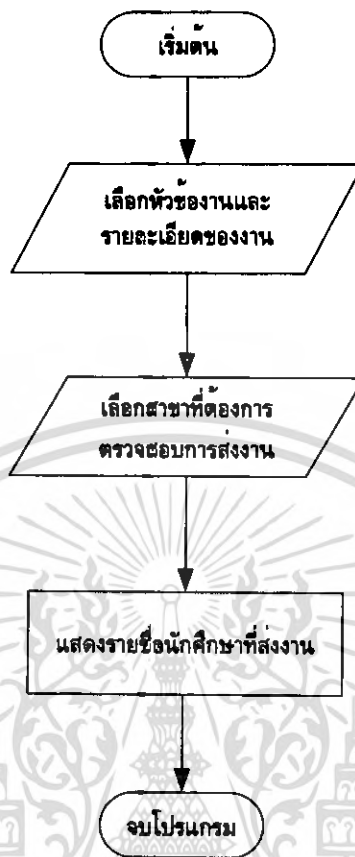
รูปที่ ง.1 แผนผังการเข้า Login

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



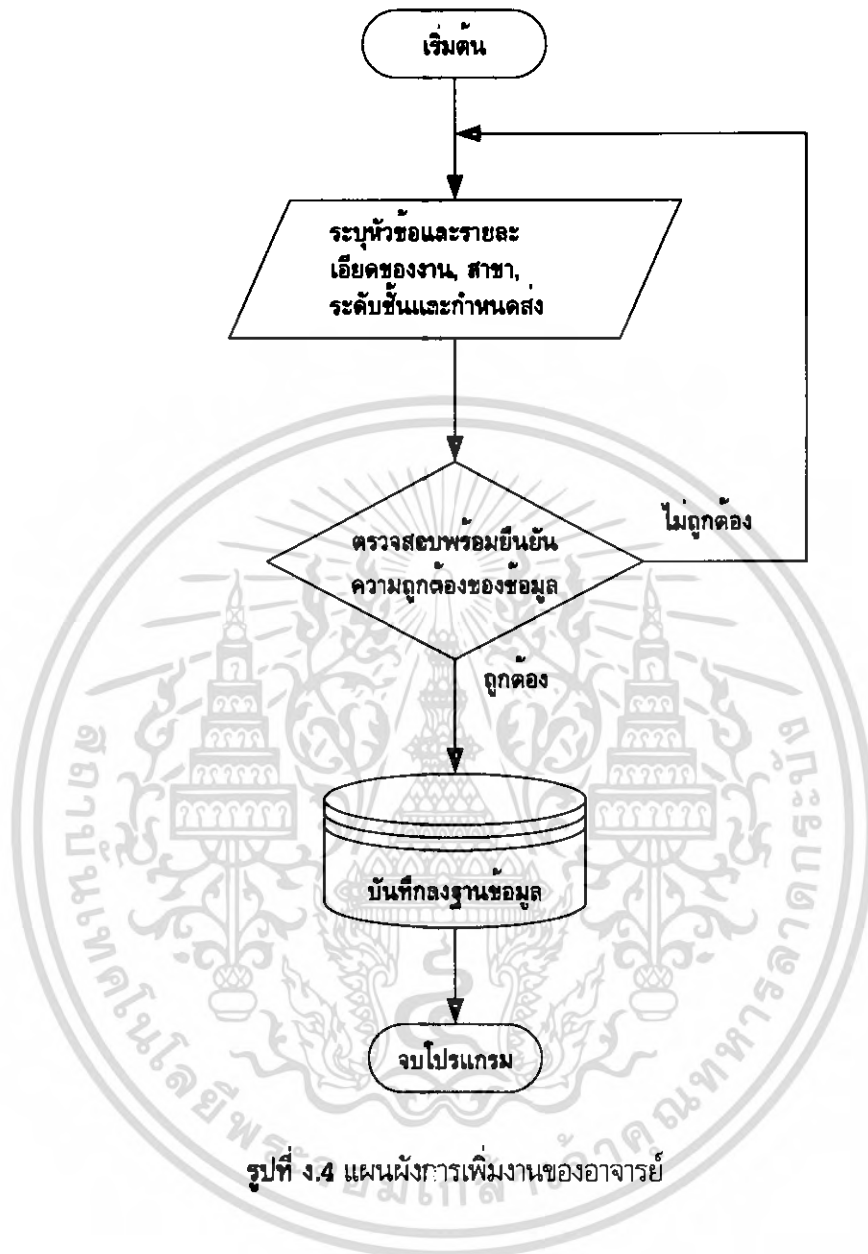
รูปที่ 3.2 แผนผังระบบสั่งงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

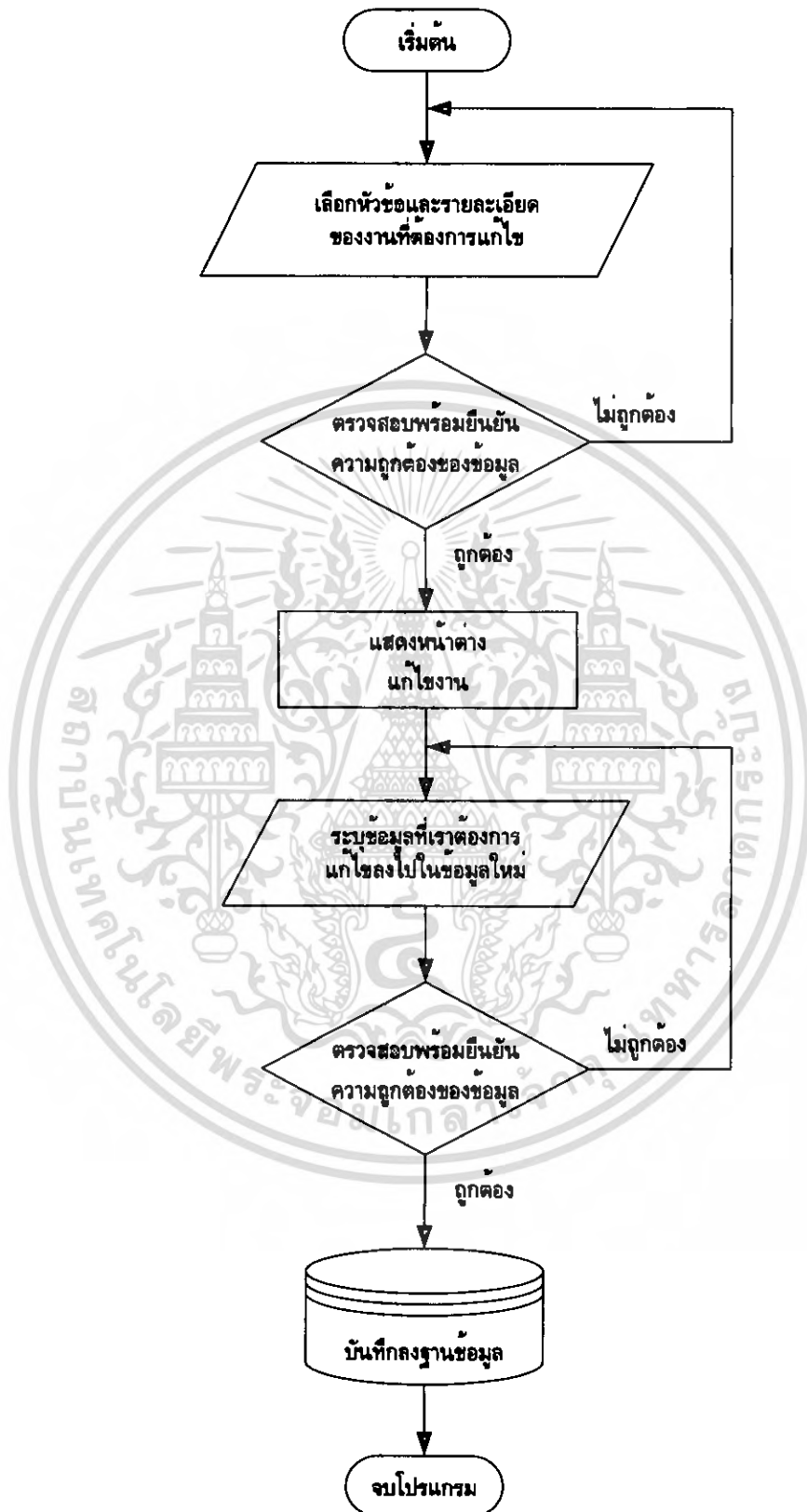


รูปที่ ๓.๓ แผนผังการตรวจสอบการส่งงานของนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

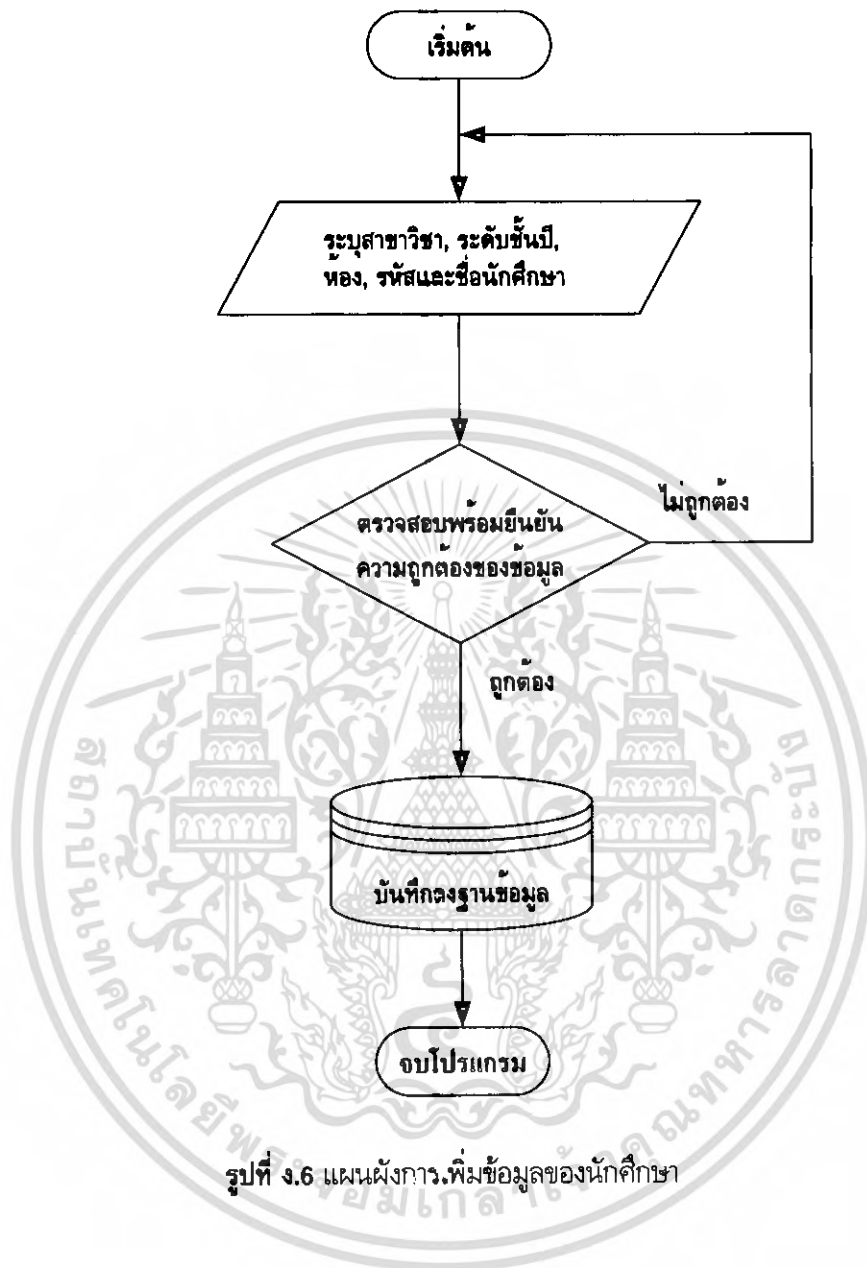


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

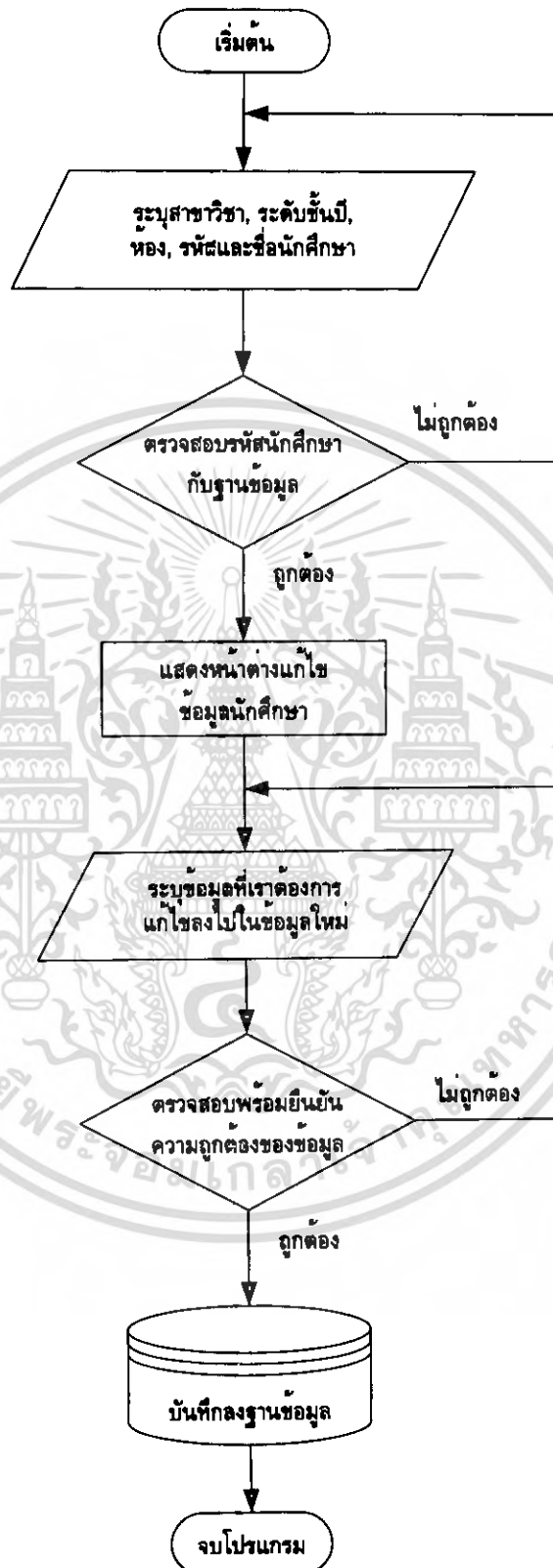


รูปที่ ๓.5 แผนผังการแก้ไขงานของอาจารย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

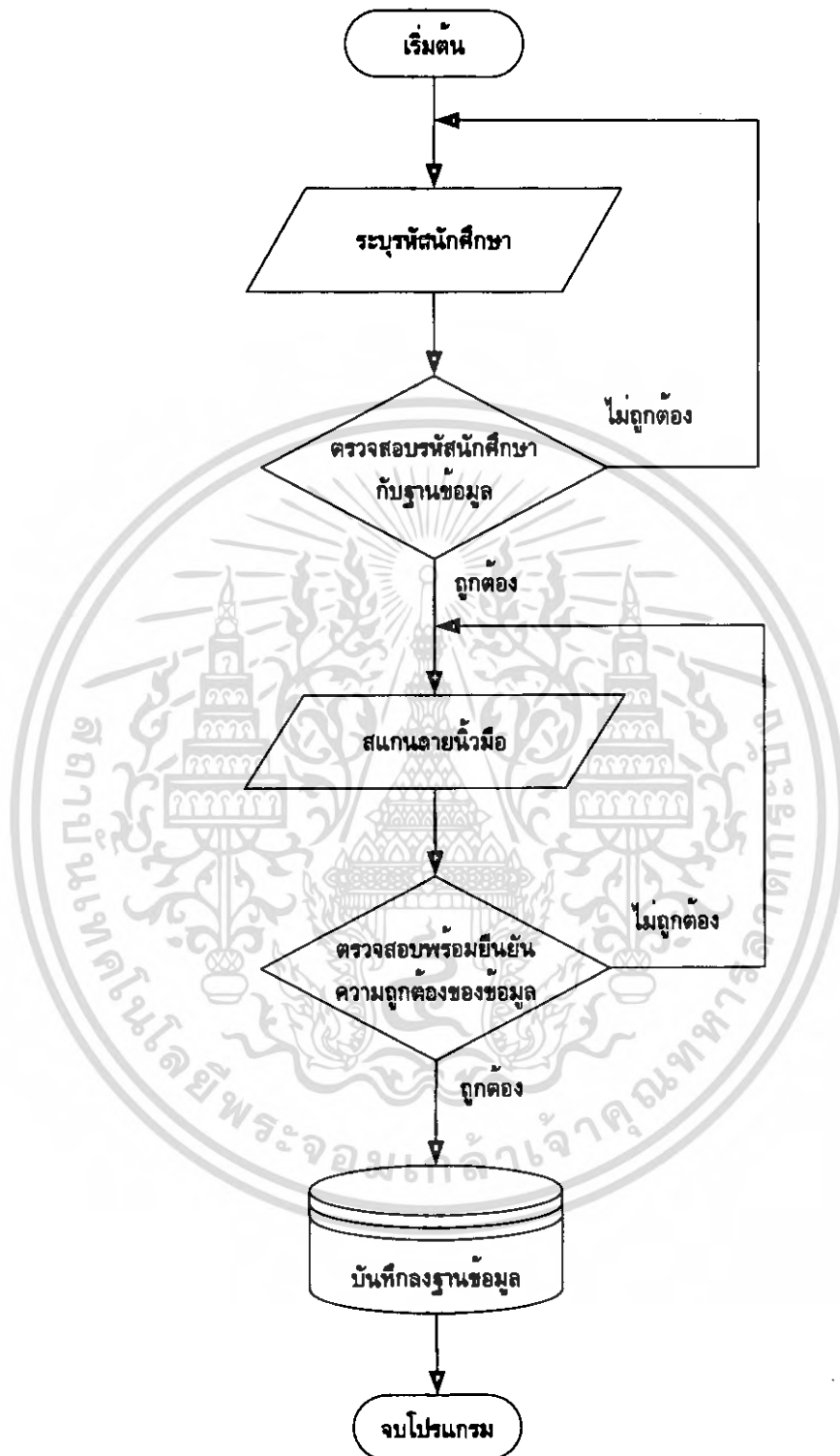


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



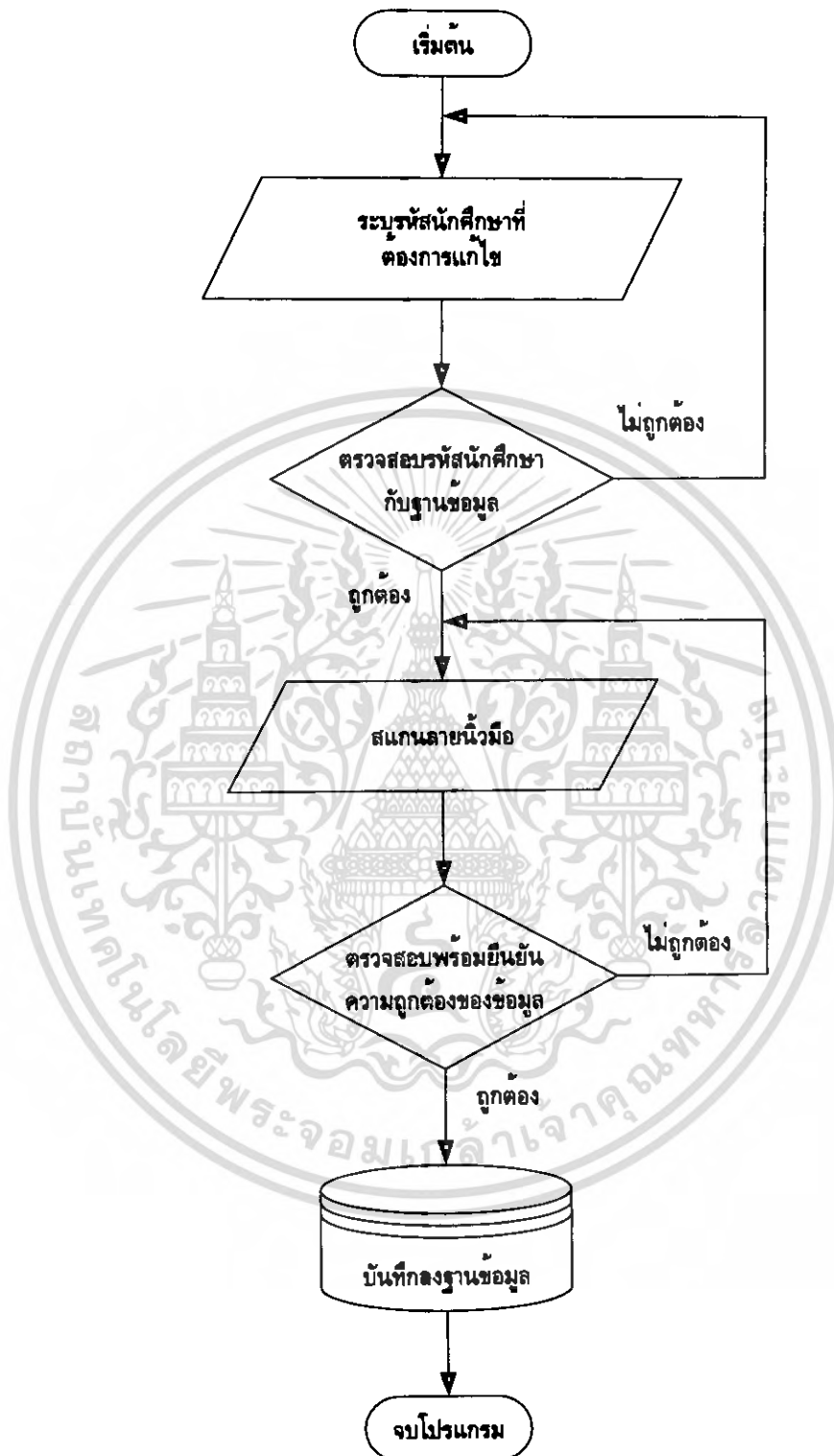
รูปที่ ง.7 แผนผังการแก้ไขข้อมูลนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ง.8 แผนผังการบันทึกข้อมูลการสแกนลายนิ้วมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

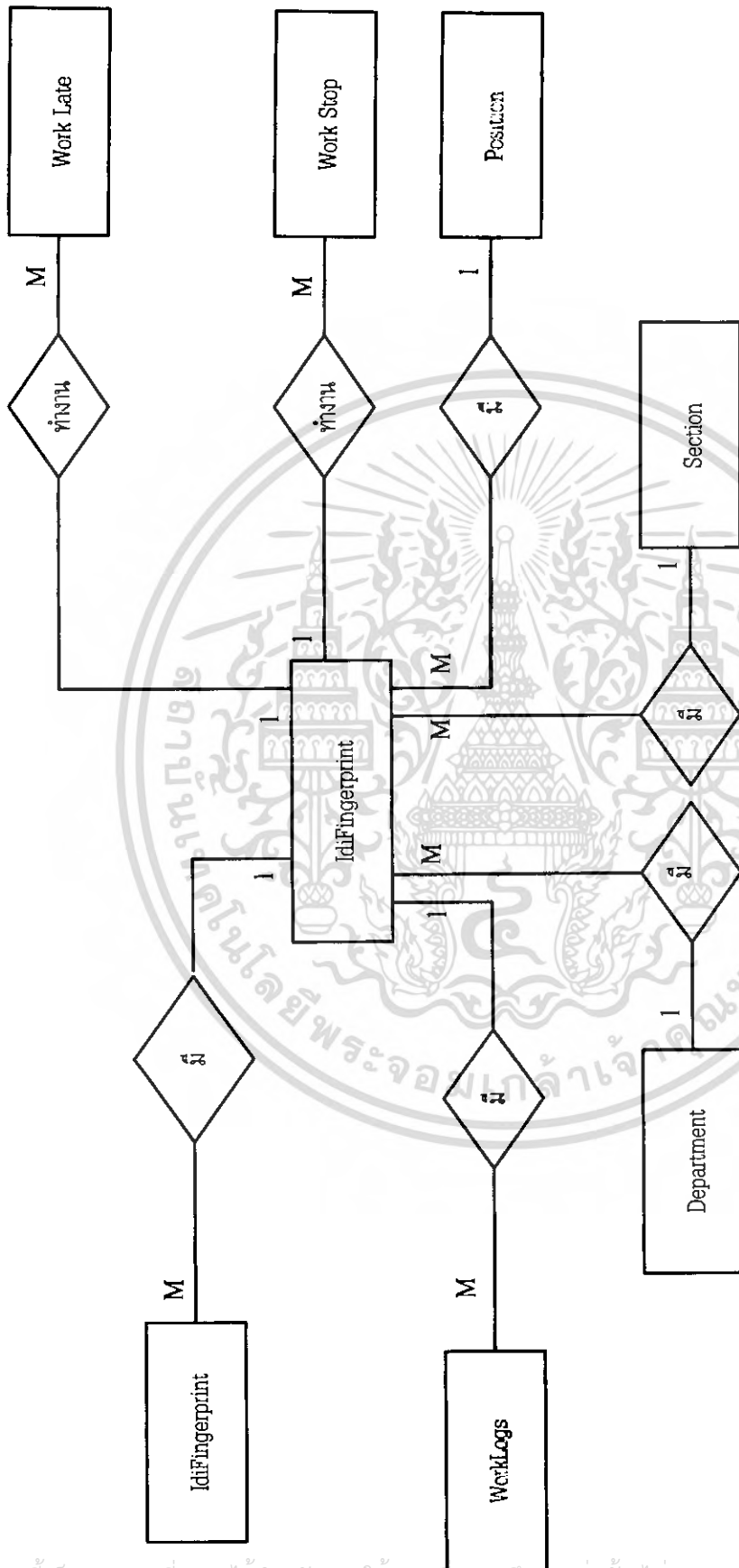


รูปที่ ๙.๙ แผนผังการแก้ไขข้อมูลลายนิ้วมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ จ.1 แผนผังความสัมพันธ์ของระบบบันทึกเวลาโดยใช้ลายนิ้วมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมระบบลงเวลาโดยใช้ลายนิ้วมือ

```

Option Explicit On
Public Class frmMain
    Private Sub ToolStripButton1_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles ToolStripButton1.Click
        Dim frmEmp As New frmEmp
        frmEmp.ShowDialog(Me)
    End Sub

    Private Sub ToolStripButton2_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles ToolStripButton2.Click
        Dim frmEmpLogs As New frmEmpLogs
        CloseAllChildForm()
        frmEmpLogs.MdiParent = Me
        frmEmpLogs.Show()
    End Sub

    Private Sub ToolStripButton3_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles ToolStripButton3.Click
        Dim frmEmplate As New frmEmplate
        CloseAllChildForm()

        frmEmplate.MdiParent = Me
        frmEmplate.Show()
    End Sub

    Private Sub ToolStripButton4_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles ToolStripButton4.Click
        Dim frmwork_time As New frmwork_time

        frmwork_time.ShowDialog(Me)

    End Sub

    Private Sub ToolStripButton5_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles ToolStripButton5.Click
        Me.Close()
    End Sub

    Private Sub CloseAllChildForm()
        Dim f As Form
        For Each f In Me.MdiChildren
            f.Close()
        Next
    End Sub

    Private Sub frmMain_Load(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles MyBase.Load

    End Sub

    Private Sub ToolStripButton6_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles ToolStripButton6.Click
        Dim frmAddOfffical_Holiday As New frmAddOfffical_Holiday
        frmAddOfffical_Holiday.ShowDialog(Me)
    End Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

End Class

โปรแกรมลงเวลาเข้าออกพนักงาน

```
Imports System
Imports System.Drawing
Imports System.Collections
Imports System.ComponentModel
Imports System.Windows.Forms
Imports System.Data
Imports System.Data.OleDb
Imports Microsoft.VisualBasic
Imports Itworks.Idworks
Public Class frmcheckin_out
    Inherits System.Windows.Forms.Form

#Region " Windows Form Designer generated code "

    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call

    End Sub

    'Form overrides dispose to clean up the component list.
    Protected Overrides Sub Dispose(ByVal disposing As Boolean)
        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
            End If
        End If
        MyBase.Dispose(disposing)
    End Sub

    'Required by the Windows Form Designer
    Private components As System.ComponentModel.IContainer

    'NOTE: The following procedure is required by the Windows Form
    Designer
    'It can be modified using the Windows Form Designer.
    'Do not modify it using the code editor.
    Friend WithEvents GroupBox1 As System.Windows.Forms.GroupBox
    Friend WithEvents Timer1 As System.Windows.Forms.Timer
    Friend WithEvents Label1 As System.Windows.Forms.Label
    Friend WithEvents txtdate As System.Windows.Forms.TextBox
    Friend WithEvents Label2 As System.Windows.Forms.Label
    Friend WithEvents txtname As System.Windows.Forms.TextBox
    Friend WithEvents txte_ID As System.Windows.Forms.TextBox
    Friend WithEvents Label3 As System.Windows.Forms.Label
    Friend WithEvents Label4 As System.Windows.Forms.Label
    Friend WithEvents PicEM As System.Windows.Forms.PictureBox
    Friend WithEvents txtTime As System.Windows.Forms.TextBox
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Private WithEvents FingerPrintPictureBox As
System.Windows.Forms.PictureBox
Friend WithEvents Label6 As System.Windows.Forms.Label
Friend WithEvents PictureBox1 As System.Windows.Forms.PictureBox
Friend WithEvents txt_eid As System.Windows.Forms.TextBox
Friend WithEvents GroupBox2 As System.Windows.Forms.GroupBox
Private WithEvents ActivateSensorButton As
System.Windows.Forms.Button
Private WithEvents EnrollmentButton As
System.Windows.Forms.Button
Private WithEvents LoadAllFingerPrintInfoButton As
System.Windows.Forms.Button
Private WithEvents StatusGroupBox As
System.Windows.Forms.GroupBox
Friend WithEvents StatusTextBox As System.Windows.Forms.TextBox
Private WithEvents SensorRegistrationGroupBox As
System.Windows.Forms.GroupBox
Private WithEvents ApplicationKeyTextBox As
System.Windows.Forms.TextBox
Private WithEvents ApplicationSerialTextBox As
System.Windows.Forms.TextBox
Private WithEvents ApplicationSerialLabel As
System.Windows.Forms.Label
Private WithEvents ApplicationKeyLabel As
System.Windows.Forms.Label
Private WithEvents SensorSerial ComboBox As
System.Windows.Forms.ComboBox
Private WithEvents SensorSerialNumberLabel As
System.Windows.Forms.Label
Friend WithEvents GroupBox3 As System.Windows.Forms.GroupBox
Friend WithEvents Label5 As System.Windows.Forms.Label
<System.Diagnostics.DebuggerStepThrough()> Private Sub
InitializeComponent()
Me.components = New System.ComponentModel.Container
Dim resources As
System.ComponentModel.ComponentResourceManager = New
System.ComponentModel.ComponentResourceManager(GetType(frmcheckin_out
))
Me.GroupBox1 = New System.Windows.Forms.GroupBox
Me.txtTime = New System.Windows.Forms.TextBox
Me.Label2 = New System.Windows.Forms.Label
Me.txtdate = New System.Windows.Forms.TextBox
Me.Label1 = New System.Windows.Forms.Label
Me.Timer1 = New System.Windows.Forms.Timer(Me.components)
Me.txtname = New System.Windows.Forms.TextBox
Me.txte_ID = New System.Windows.Forms.TextBox
Me.Label3 = New System.Windows.Forms.Label
Me.Label4 = New System.Windows.Forms.Label
Me.PicEM = New System.Windows.Forms.PictureBox
Me.FingerPrintPictureBox = New
System.Windows.Forms.PictureBox
Me.Label5 = New System.Windows.Forms.Label
Me.Label6 = New System.Windows.Forms.Label
Me.PictureBox1 = New System.Windows.Forms.PictureBox
Me.txt_eid = New System.Windows.Forms.TextBox
Me.GroupBox2 = New System.Windows.Forms.GroupBox
Me.ActivateSensorButton = New System.Windows.Forms.Button

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Me.EnrollmentButton = New System.Windows.Forms.Button
Me.LoadAllFingerPrintInfoButton = New
System.Windows.Forms.Button
Me.StatusGroupBox = New System.Windows.Forms.GroupBox
Me.StatusTextBox = New System.Windows.Forms.TextBox
Me.SensorRegistrationGroupBox = New
System.Windows.Forms.GroupBox
Me.ApplicationKeyTextBox = New System.Windows.Forms.TextBox
Me.ApplicationSerialTextBox = New
System.Windows.Forms.TextBox
Me.ApplicationSerialLabel = New System.Windows.Forms.Label
Me.ApplicationKeyLabel = New System.Windows.Forms.Label
Me.SensorSerialComboBox = New System.Windows.Forms.ComboBox
Me.SensorSerialNumberLabel = New System.Windows.Forms.Label
Me.GroupBox3 = New System.Windows.Forms.GroupBox
Me.GroupBox1.SuspendLayout()
CType(Me.PicEM,
System.ComponentModel.ISupportInitialize).BeginInit()
CType(Me.FingerPrintPictureBox,
System.ComponentModel.ISupportInitialize).BeginInit()
CType(Me.PictureBox1,
System.ComponentModel.ISupportInitialize).BeginInit()
Me.GroupBox2.SuspendLayout()
Me.StatusGroupBox.SuspendLayout()
Me.SensorRegistrationGroupBox.SuspendLayout()
Me.GroupBox3.SuspendLayout()
Me.SuspendLayout()
'
'GroupBox1
'
Me.GroupBox1.Controls.Add(Me.txtTime)
Me.GroupBox1.Controls.Add(Me.Label2)
Me.GroupBox1.Controls.Add(Me.txtdate)
Me.GroupBox1.Controls.Add(Me.Label1)
Me.GroupBox1.Location = New System.Drawing.Point(12, 87)
Me.GroupBox1.Name = "GroupBox1"
Me.GroupBox1.Size = New System.Drawing.Size(292, 112)
Me.GroupBox1.TabIndex = 0
Me.GroupBox1.TabStop = False
Me.GroupBox1.Text = "Current Time"
'
'txtTime
'
Me.txtTime.BackColor =
System.Drawing.SystemColors.InactiveCaptionText
Me.txtTime.BorderStyle =
System.Windows.Forms.BorderStyle.FixedSingle
Me.txtTime.Location = New System.Drawing.Point(56, 68)
Me.txtTime.Name = "txtTime"
Me.txtTime.ReadOnly = True
Me.txtTime.Size = New System.Drawing.Size(96, 20)
Me.txtTime.TabIndex = 3
'
'Label2
'
Me.Label2.Location = New System.Drawing.Point(8, 68)
Me.Label2.Name = "Label2"

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Me.Label2.Size = New System.Drawing.Size(40, 23)
Me.Label2.TabIndex = 2
Me.Label2.Text = "Time : "
'
'txtdate
'
Me.txtdate.BackColor =
System.Drawing.SystemColors.InactiveCaptionText
Me.txtdate.BorderStyle =
System.Windows.Forms.BorderStyle.FixedSingle
Me.txtdate.Location = New System.Drawing.Point(56, 24)
Me.txtdate.Name = "txtdate"
Me.txtdate.ReadOnly = True
Me.txtdate.Size = New System.Drawing.Size(224, 20)
Me.txtdate.TabIndex = 1
'
'Label1
'
Me.Label1.Location = New System.Drawing.Point(8, 24)
Me.Label1.Name = "Label1"
Me.Label1.Size = New System.Drawing.Size(40, 23)
Me.Label1.TabIndex = 0
Me.Label1.Text = "Date : "
'
'Timer1
'
'txtname
'
Me.txtname.BackColor =
System.Drawing.SystemColors.InactiveCaptionText
Me.txtname.BorderStyle =
System.Windows.Forms.BorderStyle.FixedSingle
Me.txtname.Location = New System.Drawing.Point(104, 34)
Me.txtname.Name = "txtname"
Me.txtname.ReadOnly = True
Me.txtname.Size = New System.Drawing.Size(180, 20)
Me.txtname.TabIndex = 2
'
'txte_ID
'
Me.txte_ID.BackColor =
System.Drawing.SystemColors.InactiveCaptionText
Me.txte_ID.BorderStyle =
System.Windows.Forms.BorderStyle.FixedSingle
Me.txte_ID.Location = New System.Drawing.Point(104, 71)
Me.txte_ID.Name = "txte_ID"
Me.txte_ID.ReadOnly = True
Me.txte_ID.Size = New System.Drawing.Size(96, 20)
Me.txte_ID.TabIndex = 4
'
'Label3
'
Me.Label3.Location = New System.Drawing.Point(12, 29)
Me.Label3.Name = "Label3"
Me.Label3.Size = New System.Drawing.Size(57, 28)
Me.Label3.TabIndex = 5

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Me.Label3.Text = "Name : "
'
'Label4
'
Me.Label4.Location = New System.Drawing.Point(8, 73)
Me.Label4.Name = "Label4"
Me.Label4.Size = New System.Drawing.Size(88, 28)
Me.Label4.TabIndex = 6
Me.Label4.Text = "Empoloyee ID : "
'
'PicEM
'
Me.PicEM.BorderStyle =
System.Windows.Forms.BorderStyle.Fixed3D
Me.PicEM.Location = New System.Drawing.Point(361, 92)
Me.PicEM.Name = "PicEM"
Me.PicEM.Size = New System.Drawing.Size(135, 173)
Me.PicEM.SizeMode =
System.Windows.Forms.PictureBoxSizeMode.StretchImage
Me.PicEM.TabIndex = 9
Me.PicEM.TabStop = False
'
'FingerPrintPictureBox
'
Me.FingerPrintPictureBox.BorderStyle =
System.Windows.Forms.BorderStyle.Fixed3D
Me.FingerPrintPictureBox.Location = New
System.Drawing.Point(361, 290)
Me.FingerPrintPictureBox.Name = "FingerPrintPictureBox"
Me.FingerPrintPictureBox.Size = New System.Drawing.Size(135,
169)
Me.FingerPrintPictureBox.SizeMode =
System.Windows.Forms.PictureBoxSizeMode.StretchImage
Me.FingerPrintPictureBox.TabIndex = 16
Me.FingerPrintPictureBox.TabStop = False
'
'Label5
'
Me.Label5.BorderStyle =
System.Windows.Forms.BorderStyle.Fixed3D
Me.Label5.Font = New System.Drawing.Font("AngsanaUPC",
15.75!, System.Drawing.FontStyle.Bold,
System.Drawing.GraphicsUnit.Point, CType(222, Byte))
Me.Label5.ForeColor = System.Drawing.Color.Red
Me.Label5.Location = New System.Drawing.Point(30, 379)
Me.Label5.Name = "Label5"
Me.Label5.Size = New System.Drawing.Size(274, 38)
Me.Label5.TabIndex = 18
'
'Label6
'
Me.Label6.AutoSize = True
Me.Label6.Font = New System.Drawing.Font("Arial", 8.25!,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point,
CType(222, Byte))
Me.Label6.ForeColor = System.Drawing.Color.Red
Me.Label6.Location = New System.Drawing.Point(76, 26)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Me.Label6.Name = "Label6"
Me.Label6.Size = New System.Drawing.Size(197, 14)
Me.Label6.TabIndex = 50
Me.Label6.Text = "To Login, Authentication Required."
'
'PictureBox1
'
Me.PictureBox1.Image =
CType(resources.GetObject("PictureBox1.Image"), System.Drawing.Image)
Me.PictureBox1.Location = New System.Drawing.Point(15, 14)
Me.PictureBox1.Name = "PictureBox1"
Me.PictureBox1.Size = New System.Drawing.Size(47, 48)
Me.PictureBox1.SizeMode =
System.Windows.Forms.PictureBoxSizeMode.StretchImage
Me.PictureBox1.TabIndex = 49
Me.PictureBox1.TabStop = False
'
'txt_eid
'
Me.txt_eid.Location = New System.Drawing.Point(15, 33)
Me.txt_eid.Name = "txt_eid"
Me.txt_eid.Size = New System.Drawing.Size(150, 20)
Me.txt_eid.TabIndex = 48
'
'GroupBox2
'
Me.GroupBox2.Controls.Add(Me.ActivateSensorButton)
Me.GroupBox2.Controls.Add(Me.EnrollmentButton)
Me.GroupBox2.Controls.Add(Me.LoadAllFingerPrintInfoButton)
Me.GroupBox2.Location = New System.Drawing.Point(90, 12)
Me.GroupBox2.Name = "GroupBox2"
Me.GroupBox2.Size = New System.Drawing.Size(56, 15)
Me.GroupBox2.TabIndex = 47
Me.GroupBox2.TabStop = False
Me.GroupBox2.Text = "GroupBox2"
'
'ActivateSensorButton
'
Me.ActivateSensorButton.FlatStyle =
System.Windows.Forms.FlatStyle.System
Me.ActivateSensorButton.Location = New
System.Drawing.Point(0, 17)
Me.ActivateSensorButton.Name = "ActivateSensorButton"
Me.ActivateSensorButton.Size = New System.Drawing.Size(176,
45)
Me.ActivateSensorButton.TabIndex = 7
Me.ActivateSensorButton.Text = "Activate sensor && Start
capturing"
'
'EnrollmentButton
'
Me.EnrollmentButton.FlatStyle =
System.Windows.Forms.FlatStyle.System
Me.EnrollmentButton.Location = New System.Drawing.Point(182,
17)
Me.EnrollmentButton.Name = "EnrollmentButton"
Me.EnrollmentButton.Size = New System.Drawing.Size(176, 45)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        Me.EnrollmentButton.TabIndex = 8
        Me.EnrollmentButton.Text = "Enroll finger && save to
database"
    ,
        'LoadAllFingerPrintInfoButton
    ,
        Me.LoadAllFingerPrintInfoButton.Enabled = False
        Me.LoadAllFingerPrintInfoButton.FlatStyle =
System.Windows.Forms.FlatStyle.System
        Me.LoadAllFingerPrintInfoButton.Location = New
System.Drawing.Point(362, 17)
        Me.LoadAllFingerPrintInfoButton.Name =
"LoadAllFingerPrintInfoButton"
        Me.LoadAllFingerPrintInfoButton.Size = New
System.Drawing.Size(176, 45)
        Me.LoadAllFingerPrintInfoButton.TabIndex = 9
        Me.LoadAllFingerPrintInfoButton.Text = "Load FpInfo from
database"
    ,
        'StatusGroupBox
    ,
        Me.StatusGroupBox.Controls.Add(Me.StatusTextBox)
        Me.StatusGroupBox.FlatStyle =
System.Windows.Forms.FlatStyle.System
        Me.StatusGroupBox.Location = New System.Drawing.Point(173,
13)
        Me.StatusGroupBox.Name = "StatusGroupBox"
        Me.StatusGroupBox.Size = New System.Drawing.Size(59, 18)
        Me.StatusGroupBox.TabIndex = 46
        Me.StatusGroupBox.TabStop = False
        Me.StatusGroupBox.Text = " Status "
    ,
        'StatusTextBox
    ,
        Me.StatusTextBox.Location = New System.Drawing.Point(8, 16)
        Me.StatusTextBox.Multiline = True
        Me.StatusTextBox.Name = "StatusTextBox"
        Me.StatusTextBox.RightToLeft =
System.Windows.Forms.RightToLeft.No
        Me.StatusTextBox.ScrollBars =
System.Windows.Forms.ScrollBars.Both
        Me.StatusTextBox.Size = New System.Drawing.Size(528, 248)
        Me.StatusTextBox.TabIndex = 16
    ,
        'SensorRegistrationGroupBox
    ,
        Me.SensorRegistrationGroupBox.Controls.Add(Me.ApplicationKeyTextBox)
        Me.SensorRegistrationGroupBox.Controls.Add(Me.ApplicationSerialTextBo
x)
        Me.SensorRegistrationGroupBox.Controls.Add(Me.ApplicationSerialLabel)
        Me.SensorRegistrationGroupBox.Controls.Add(Me.ApplicationKeyLabel)
        Me.SensorRegistrationGroupBox.Controls.Add(Me.SensorSerialComboBox)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Me.SensorRegistrationGroupBox.Controls.Add(Me.SensorSerialNumberLabel
)
    Me.SensorRegistrationGroupBox.FlatStyle =
System.Windows.Forms.FlatStyle.System
    Me.SensorRegistrationGroupBox.Location = New
System.Drawing.Point(15, 12)
    Me.SensorRegistrationGroupBox.Name =
"SensorRegistrationGroupBox"
    Me.SensorRegistrationGroupBox.Size = New
System.Drawing.Size(69, 19)
    Me.SensorRegistrationGroupBox.TabIndex = 45
    Me.SensorRegistrationGroupBox.TabStop = False
    Me.SensorRegistrationGroupBox.Text = " Sensor Registration "
    'ApplicationKeyTextBox
    '
    Me.ApplicationKeyTextBox.Location = New
System.Drawing.Point(8, 144)
    Me.ApplicationKeyTextBox.Name = "ApplicationKeyTextBox"
    Me.ApplicationKeyTextBox.Size = New System.Drawing.Size(376,
20)
    Me.ApplicationKeyTextBox.TabIndex = 5
    Me.ApplicationKeyTextBox.Text = "E7AA-E050-1847-07AF-210C-
A40C-6DC5-3816"
    'ApplicationSerialTextBox
    '
    Me.ApplicationSerialTextBox.Location = New
System.Drawing.Point(8, 96)
    Me.ApplicationSerialTextBox.Name = "ApplicationSerialTextBox"
    Me.ApplicationSerialTextBox.Size = New
System.Drawing.Size(376, 20)
    Me.ApplicationSerialTextBox.TabIndex = 4
    Me.ApplicationSerialTextBox.Text = "64A1-C4D6-2C2B-D92B"
    'ApplicationSerialLabel
    '
    Me.ApplicationSerialLabel.AutoSize = True
    Me.ApplicationSerialLabel.FlatStyle =
System.Windows.Forms.FlatStyle.System
    Me.ApplicationSerialLabel.Location = New
System.Drawing.Point(8, 80)
    Me.ApplicationSerialLabel.Name = "ApplicationSerialLabel"
    Me.ApplicationSerialLabel.Size = New System.Drawing.Size(88,
13)
    Me.ApplicationSerialLabel.TabIndex = 3
    Me.ApplicationSerialLabel.Text = "Application Serial"
    'ApplicationKeyLabel
    '
    Me.ApplicationKeyLabel.AutoSize = True
    Me.ApplicationKeyLabel.FlatStyle =
System.Windows.Forms.FlatStyle.System
    Me.ApplicationKeyLabel.Location = New System.Drawing.Point(8,
128)
    Me.ApplicationKeyLabel.Name = "ApplicationKeyLabel"

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Me.ApplicationKeyLabel.Size = New System.Drawing.Size(80, 13)
Me.ApplicationKeyLabel.TabIndex = 2
Me.ApplicationKeyLabel.Text = "Application Key"
'
'SensorSerialComboBox
'
Me.SensorSerialComboBox.Location = New
System.Drawing.Point(8, 48)
Me.SensorSerialComboBox.Name = "SensorSerialComboBox"
Me.SensorSerialComboBox.Size = New System.Drawing.Size(376,
21)
Me.SensorSerialComboBox.TabIndex = 1
'
'SensorSerialNumberLabel
'
Me.SensorSerialNumberLabel.AutoSize = True
Me.SensorSerialNumberLabel.FlatStyle =
System.Windows.Forms.FlatStyle.System
Me.SensorSerialNumberLabel.Location = New
System.Drawing.Point(8, 32)
Me.SensorSerialNumberLabel.Name = "SensorSerialNumberLabel"
Me.SensorSerialNumberLabel.Size = New
System.Drawing.Size(109, 13)
Me.SensorSerialNumberLabel.TabIndex = 0
Me.SensorSerialNumberLabel.Text = "Sensor Serial Number"
'
'GroupBox3
'
Me.GroupBox3.Controls.Add(Me.Label13)
Me.GroupBox3.Controls.Add(Me.txtname)
Me.GroupBox3.Controls.Add(Me.txt_e_ID)
Me.GroupBox3.Controls.Add(Me.Label14)
Me.GroupBox3.Location = New System.Drawing.Point(15, 220)
Me.GroupBox3.Name = "GroupBox3"
Me.GroupBox3.Size = New System.Drawing.Size(292, 112)
Me.GroupBox3.TabIndex = 51
Me.GroupBox3.TabStop = False
Me.GroupBox3.Text = "Profile"
'
'frmcheckin_out
'
Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
Me.BackColor = System.Drawing.SystemColors.ControlLight
Me.ClientSize = New System.Drawing.Size(512, 471)
Me.Controls.Add(Me.GroupBox3)
Me.Controls.Add(Me.Label6)
Me.Controls.Add(Me.PictureBox1)
Me.Controls.Add(Me.txt_eid)
Me.Controls.Add(Me.GroupBox2)
Me.Controls.Add(Me.StatusGroupBox)
Me.Controls.Add(Me.SensorRegistrationGroupBox)
Me.Controls.Add(Me.Label5)
Me.Controls.Add(Me.FingerPrintPictureBox)
Me.Controls.Add(Me.PicEM)
Me.Controls.Add(Me.GroupBox1)
Me.FormBorderStyle =
System.Windows.Forms.FormBorderStyle.FixedToolWindow

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    Me.MaximizeBox = False
    Me.Name = "frmcheckin_out"
    Me.StartPosition =
System.Windows.Forms.FormStartPosition.CenterScreen
    Me.Text = ":: Check In - Out ::"
    Me.GroupBox1.ResumeLayout(False)
    Me.GroupBox1.PerformLayout()
    CType(Me.PicEM,
System.ComponentModel.ISupportInitialize).EndInit()
    CType(Me.FingerPrintPictureBox,
System.ComponentModel.ISupportInitialize).EndInit()
    CType(Me.PictureBox1,
System.ComponentModel.ISupportInitialize).EndInit()
    Me.GroupBox2.ResumeLayout(False)
    Me.StatusGroupBox.ResumeLayout(False)
    Me.StatusGroupBox.PerformLayout()
    Me.SensorRegistrationGroupBox.ResumeLayout(False)
    Me.SensorRegistrationGroupBox.PerformLayout()
    Me.GroupBox3.ResumeLayout(False)
    Me.GroupBox3.PerformLayout()
    Me.ResumeLayout(False)
    Me.PerformLayout()

End Sub

#End Region
Dim mycon As OleDbConnection
Dim mycmd As OleDbCommand
Dim myreader As OleDbDataReader
Dim mysql As String
Private Const _FINGERPRINT_CONNECTION_STRING As String =
"Provider=Microsoft.Jet.OLEDB.4.0;Data
Source=D:\Project\EMS\EMS\App_Data\EMS.mdb;Persist Security
Info=False"
Private _FingerPrintConnection As OleDbConnection
Private _IdiContext As UInt32 = Convert.ToUInt32(0)
Private _FpeContext As UInt32 = Convert.ToUInt32(0)

Private Sub frmcheckin_out_FormClosed(ByVal sender As Object,
ByVal e As System.Windows.Forms.FormClosedEventArgs) Handles
Me.FormClosed
    FpeWrapper.CloseContext(Convert.ToUInt32(_FpeContext))
    FpeWrapper.FinalizeModule()
    IdiWrapper.CloseContext(Convert.ToUInt32(_IdiContext))
    IdiWrapper.FinalizeModule()
End Sub

Private Sub frmcheckin_out_FormClosing(ByVal sender As Object,
ByVal e As System.Windows.Forms.FormClosingEventArgs) Handles
Me.FormClosing
    mycon.Close()
End Sub

Private Sub frmcheckin_out_Load(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles MyBase.Load

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mycon = New OleDbConnection("provider =
microsoft.jet.oledb.4.0;" & "data source
=D:\Project\EMS\EMS\App_Data\EMS.mdb")

mycon.Open()
Timer1.Enabled = True
txt_eid.Hide()
SensorRegistrationGroupBox.Hide()
StatusGroupBox.Hide()
GroupBox2.Hide()
'TextBox2.Hide()
_FingerprintConnection = New
OleDbConnection(_FINGERPRINT_CONNECTION_STRING)
IslWrapper.InitModule()
Try
    If IslResultTypes.OK = IslWrapper.IsModuleInited Then
        Dim sensor_count As Integer = 0
        Dim result_code As IslResultTypes =
IslWrapper.GetDeviceCount(sensor_count)
        If sensor_count > 1 Then
            Dim sensor_list As IslWrapper.DeviceInfo()
            If IslResultTypes.OK =
IslWrapper.GetDeviceList(sensor_list, sensor_count) Then
                SensorSerialComboBox.Items.Clear()
                Dim i As Integer = 0
                While i < sensor_count
SensorSerialComboBox.Items.Add(sensor_list(i).SensorID.ToString)
System.Math.Min(System.Threading.Interlocked.Increment(i), i - 1)
                End While
                SensorSerialComboBox.SelectedIndex = 0
            End If
        End If
    End If
Finally
    IslWrapper.FinalizeModule()
End Try
IdiWrapper.InitModule()
FpeWrapper.InitModule()
loadSerial()
IdiWrapper.fp_ClearAll(_IdiContext)
Dim fingerprint_dataset As DataSet = New DataSet
Dim fingerprint_adapter As OleDbDataAdapter = New
OleDbDataAdapter
    Dim fingerprint_command_builder As OleDbCommandBuilder = New
OleDbCommandBuilder(fingerprint_adapter)
    fingerprint_adapter.SelectCommand = New OleDbCommand("SELECT
* FROM IDIFINGERPRINT ORDER BY IndexValue", _FingerprintConnection)
    fingerprint_adapter.Fill(fingerprint_dataset)
    For Each each_row As DataRow In
fingerprint_dataset.Tables(0).Rows
        Dim fp_info As FingerprintInfo = New FingerprintInfo
        LoadFingerprintInfo(each_row, fp_info)
        IdiWrapper.fp_Add(_IdiContext, fp_info)
    Next

```

```

Me.ShowStatus("Load all finger print info success")

End Sub

Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Timer1.Tick
    txtdate.Text = Date.Now.ToLongDateString
    txtTime.Text = Date.Now.ToLongTimeString
End Sub

Private Sub IdiWrapper_OnIdentEvent(ByVal e As
Itworks.Idworks.IdentEventArgs)
    Me.ShowStatus("-----");
    Me.ShowStatus("Sensor ID = " +
e.IdentMessage.SensorInfo.SensorID);

    Select Case e.IdentMessage.MsgID
        '0 indicate that IDWORKS cat match this finger with
        fingerprint data in memory
        '(add fingerprint data to memory via idi_fp_Add function)
        'and pass fingerprint info (include tag) via
        IdentEventMsg.
        Case 0
            Me.ShowStatus("O - Found Tag = " +
e.IdentMessage.Tag + ", Matching Score : " +
e.IdentMessage.QualityScore.ToString)

            '1 indicate that ICWORKS can not match this finger
            with any fingerprint data in memory.
            Case 1
                Me.ShowStatus("X - Not Found")

        End Select

    Me.ShowStatus("-----");

    txt_eid.Text = e.IdentMessage.Tag 'ส่วนโปรแกรมจะนำชื่อส่งค่าที่อ่านมาได้'
    If txt_eid.Text <> "" Then

        Dim showpicture As String = ""

        mycmd = New OleDbCommand("SELECT * FROM tbl_employee
WHERE emp_id = '" & txt_eid.Text & "'", mycon);

        myreader = mycmd.ExecuteReader()
        While myreader.Read()

            txtname.Text = myreader.Item("emp_fname") & " " &
myreader.Item("emp_lname")
            showpicture = myreader.Item("picture")
            txt_eID.Text = myreader.Item("emp_id")
        End While

        PicEM.ImageLocation = showpicture & ""

        ' ส่วน record
        Dim txtsql As String

```

```

Dim count As Integer = 0

mycmd = New OleDbCommand("SELECT *FROM tbl_workLogs WHERE
(((tbl_workLogs.emp_id)=' " & txt_eID.Text & "') AND
((tbl_workLogs.logs_date)=# " & Date.Now.Month & "/" & Date.Now.Day &
"/" & Date.Now.Year & "#))", mycon)
myreader = mycmd.ExecuteReader()
While myreader.Read()

    count += 1

End While
myreader.Close()
If count = 0 Then

    txtsql = "insert into tbl_workLogs
(emp_id,logs_in,logs_date) values '" & txt_eID.Text & "', '" &
txtsql &= Date.Now.ToShortTimeString & "', '" &
Date.Now.Date & "'"
    mycmd = New OleDbCommand(txtsql, mycon)
    Try
        mycmd.ExecuteNonQuery()
    Catch Errfrominsert As Exception
        MsgBox("Cannot Check In ")
    End Try
    ' MsgBox("1111")
    Label5.Text = "เวลาที่ลงบันทึก " &
Date.Now.ToShortTimeString
Else
    Label5.Text = "เวลาที่ลงบันทึก " &
Date.Now.ToShortTimeString
    txtsql = "UPDATE tbl_workLogs SET logs_out = '" &
Date.Now.ToShortTimeString & "' WHERE emp_id = '" & txt_eID.Text &
"'AND logs_date= #" & Date.Now.Month & "/" & Date.Now.Day & "/" &
Date.Now.Year & "#"

    mycmd = New OleDbCommand(txtsql, mycon)
    Try
        mycmd.ExecuteNonQuery()
    Catch Errfrominsert As Exception
        MsgBox("Cannot Check out ")
    End Try

End If

' เช็คว่าสาขาใหม่
mycmd = New OleDbCommand("SELECT sec_id FROM
tbl_employee WHERE emp_id = '" & txt_eid.Text & "'", mycon)
myreader = mycmd.ExecuteReader()

Dim sec_id As String = ""
Dim checkin_out As Date
While myreader.Read()

```

```

        sec_id = myreader.Item("sec_id")

    End While

    mycmd = New OleDbCommand("SELECT * FROM tbl_section
WHERE sec_id = " & sec_id, mycon)

    myreader = mycmd.ExecuteReader()

    While myreader.Read()

        checkin_out = myreader.Item("sec_start")

    End While

    'MsgBox("111111")

    If checkin_out.ToShortTimeString <
Date.Now.ToShortTimeString And count = 0 Then
        Label5.Text = "none"
        ' MsgBox("1")
        count = 0

        mycmd = New OleDbCommand("SELECT * FROM tbl_work_late
WHERE (((tbl_work_late.emp_id)=' " & txt_eID.Text & "') AND
((tbl_work_late.logs_date)=#" & Date.Now.Month & "/" & Date.Now.Day &
"/" & Date.Now.Year & "#))", mycon)
        myreader = mycmd.ExecuteReader()
        While myreader.Read()

            count += 1

        End While
        If count = 0 Then

            txtsql = "insert into tbl_work_late
(emp_id,logs_date,logs_late) values(' " & txt_eID.Text & "',' " &
Date.Now.Date & "','1') "
            mycmd = New OleDbCommand(txtsql, mycon)
            mycmd.ExecuteNonQuery()

        End If

    Else

        Label5.Text = "เวลาที่ลงบันทึก " &
Date.Now.ToShortTimeString
        ' MsgBox("2")

    End If

End If

txt_eid.Text = ""

End Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คู่มือการใช้งาน
ระบบบันทึกเวลาโดยใช้ลายนิ้วมือ



ภาควิชาครุศาสตร์วิศวกรรม
คณะครุศาสตร์อุตสาหกรรม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

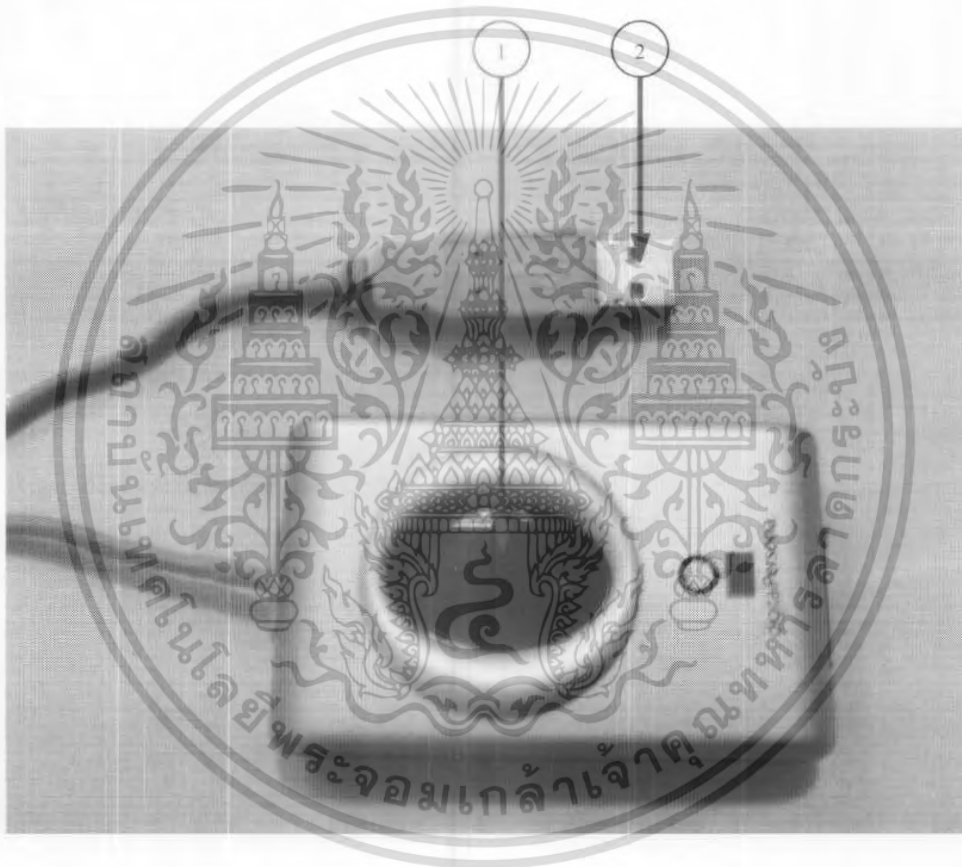
ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. คำแนะนำเบื้องต้น

ก่อนที่จะลงมือใช้งานระบบส่งงานโดยการยืนยันลายนิ้วมือ การทำการศึกษาการใช้งานจากคู่มือให้เข้าใจเพื่อการใช้งานที่ถูกต้อง และเป็นการป้องกันความเสียหายที่อาจเกิดกับเครื่องสแกนลายนิ้วมือและโปรแกรมระบบส่งงานโดยการยืนยันลายนิ้วมือ

2. ส่วนประกอบของเครื่องสแกนลายนิ้วมือ

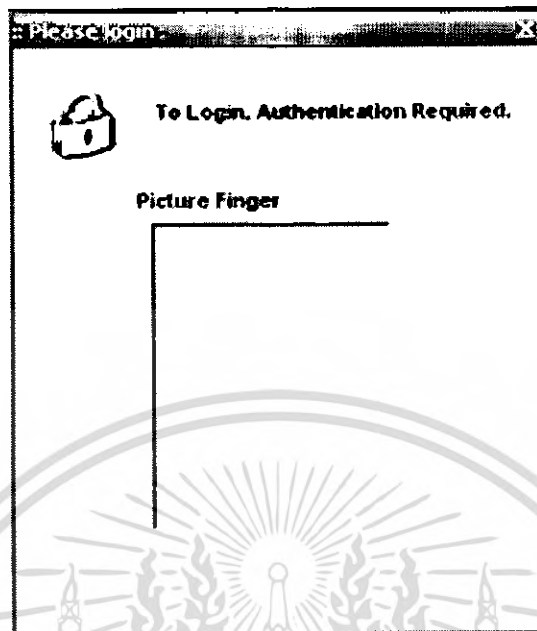


รูปที่ ข.1 ส่วนประกอบและปุ่มควบคุมของเครื่องสแกนลายนิ้วมือ

จากรูปที่ ข.1 มีรายละเอียดต่างๆ ดังนี้

1. หน้าสัมผัสฟิงเกอร์สแกน
2. หัวต่อ USB Port

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ๓.2 สำหรับ login เข้าสู่ระบบบันทึกเวลาโดยใช้ลายนิ้วมือ
ในส่วนของผู้ใช้งาน

Current Time

Date:

Time:

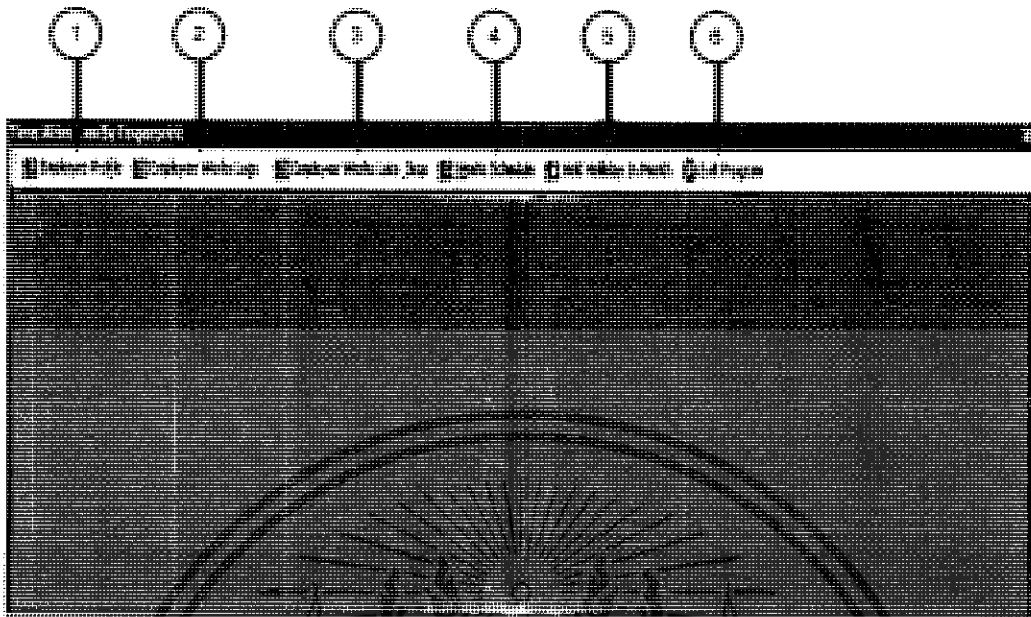
Profile

Name:

Employee ID:

รูปที่ ๓.3 หน้าต่างสำหรับบันทึกเวลาโดยใช้ลายนิ้วมือในส่วนของผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ๔.4 ส่วนประกอบและปุ่มการใช้งานของระบบบันทึกเวลาโดยใช้ลายนิ้วมือ
ในส่วนของพนักงาน

3. การติดตั้งและการใช้งาน

- 3.1) ต่อสาย USB Port เข้ากับ USB Port ของคอมพิวเตอร์
- 3.2) เปิดเครื่องคอมพิวเตอร์เตอร์
- 3.3) ให้ทำการก๊อปปี้ไฟล์เตอร์ที่ชื่อ "ระบบบันทึกเวลาโดยใช้ลายนิ้วมือ" ไปไว้ในเครื่องคอมพิวเตอร์
- 3.4) ทำการดับเบิลคลิกที่ไฟล์ที่ชื่อ "Admin" เพื่อทำการเปิดโปรแกรม
- 3.5) การทำงานของปุ่มต่างๆ แต่ละปุ่มมีดังนี้
 - รูปที่ ๔.2 จะเป็นการ login เข้าสู่ระบบบันทึกเวลาโดยใช้ลายนิ้วมือของเจ้าหน้าที่ ข้อมูลลายนิ้วมือของเจ้าหน้าที่อยู่ในฐานข้อมูลก่อน เมื่อทำการสแกนลายนิ้วมือของเจ้าหน้าที่ จะปรากฏหน้าต่างดังรูปที่ ๔.3
 - รูปที่ ๔.3 จะเป็นการ login บันทึกเวลาการเข้า-ออก ของพนักงาน
 - ปุ่มที่ 1 ของรูป ๔.4 จะเป็นปุ่มสำหรับบันทึกและแก้ไขข้อมูลส่วนตัวของพนักงาน
 - ปุ่มที่ 2 ของรูป ๔.4 จะเป็นปุ่มสำหรับดูบันทึกการเข้า-ออกเวลาของพนักงาน
 - ปุ่มที่ 3 ของรูป ๔.4 จะเป็นปุ่มสำหรับดูบันทึกการขาดและมาสายของพนักงาน
 - ปุ่มที่ 4 ของรูป ๔.4 จะเป็นปุ่มสำหรับกำหนดระยะเวลาการทำงานของพนักงาน
 - ปุ่มที่ 5 ของรูป ๔.4 จะเป็นปุ่มสำหรับกำหนดวันหยุดของบริษัท
 - ปุ่มที่ 6 ของรูป ๔.4 ปุ่มออกจากโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. การแก้ไขปัญหาเบื้องต้น

เมื่อท่านเกิดประสบปัญหาเกิดขึ้นในการใช้งานเครื่องสแกนลายนิ้วมือ สามารถตรวจสอบแนวทางการแก้ไขปัญหาจากตารางที่ ข.1

ตารางที่ ข.1 การแก้ไขปัญหาเบื้องต้น

อาการ	สาเหตุและ/หรือวิธีการแก้ไข
เครื่องสแกนลายนิ้วมือไม่ทำงาน	ตรวจสอบการต่อสาย USB Port ที่ต่อกับคอมพิวเตอร์
ถ้าโปรแกรมเกิด Error	ก็ควรปิดโปรแกรมแล้วเปิดโปรแกรมใหม่

5. การดูแลรักษาและข้อควรระมัดระวัง

5.1 การดูแลรักษา

1. ควรปิดฝาป้องกันฝุ่นอยู่เสมอเมื่อไม่ใช้งาน
2. ในการทำความสะอาดฝุ่นละอองที่อาจตกหล่นอยู่ให้ ใช้สก็อตเทปชิ้นเล็กๆ ด้านที่มีความเหนียวติดลงบนช่องอ่านลายนิ้วมือแล้วดึงออกถ้าต้องการเช็ดทำความสะอาด ต้องใช้ผ้านุ่มๆ ไม่มีขุยหลุดออกมาแต่น้ำหรือน้ำยาเช็ดกระจกที่เป็นส่วนผสมของแอมโมเนียเช็ดเบาๆ
3. เมื่อไม่ใช้งานควรถอดเครื่องสแกนลายนิ้วมือออกจาก USB Port

5.2 ข้อควรระวัง

1. อย่าให้เครื่องสแกนโดนน้ำหรือความชื้นโดยเด็ดขาด
2. อย่าตัดแปลงหรือแกะเครื่องโดยไม่มีความรู้และความชำนาญ
3. อย่าวางเครื่องให้โดนแสงแดดหรือในที่ที่มีอุณหภูมิสูงๆ
4. นิ้วมือที่นำมาสแกนควรสะอาดและแห้ง
5. อย่าใช้กระดาษเช็ดลงไปบนช่องอ่านลายนิ้วมือ อย่าจุ่มหรือเทน้ำลงบนเครื่องอ่านลายนิ้วมือ และแอลกอฮอล์จะทำลายแผ่นฟิล์มบางๆ บนเครื่องอ่านลายนิ้วมือได้
6. ไม่ควรให้เล็บโดนหน้าสัมผัสของตัวเครื่องสแกน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ข้อมูลจำเพาะ

ตารางที่ ข.2 ข้อมูลจำเพาะ

คุณสมบัติ	รายละเอียด
หลักการวัดการสแกนลายนิ้วมือ	ใช้เครื่องสแกนลายนิ้วมือรุ่น DigialPersona U.are.U 4000
การแสดงผล	แสดงผลโดยใช้คอมพิวเตอร์
การติดต่อ	ใช้ USB Port
ความเที่ยงตรง	99% ที่ทำการวัดจำนวน 10 ครั้ง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง



ชื่อ - สกุล

นางสาววรินทร์นิภา ขจัดมาน

วันเดือนปีเกิด

6 กรกฎาคม พ.ศ.2526

ภูมิลำเนา

479/2 ซอยวัดพระยาไย้ง แขวงถนนเพชรบุรี เขตราชเทวี
จังหวัดกรุงเทพมหานคร 10400

ประวัติการศึกษา

ประถมศึกษา

โรงเรียนกิ่งเพชร จังหวัดกรุงเทพมหานคร

มัธยมศึกษาตอนต้น

โรงเรียนสาธิตมัธยมศึกษาบ้านราชวิถีเพชรบุรี วิทยาลัยการณ

ในพระบรมราชูปถัมภ์ จังหวัดกรุงเทพมหานคร

ประกาศนียบัตรวิชาชีพ

โรงเรียนช่างฝีมือทหาร จังหวัดกรุงเทพมหานคร

ประกาศนียบัตรวิชาชีพชั้นสูง

โรงเรียนช่างฝีมือทหาร จังหวัดกรุงเทพมหานคร

ปริญญาตรี

สาขาวิชาวิศวกรรมโทรคมนาคม

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ความสนใจพิเศษ

เทคโนโลยีคอมพิวเตอร์และโทรคมนาคม

คติเตือนใจ

จะดีจะชั่วอยู่ที่ตัวเราทำ จะสูงจะต่ำอยู่ที่เราทำตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง



ชื่อ - สกุล

นายกิตติศักดิ์ ยิ่งยง

วันเดือนปีเกิด

22 มีนาคม พ.ศ. 2528

ภูมิลำเนา

802/765 หมู่ 12 ตำบลคูคต อำเภอลำลูกกา
จังหวัดปทุมธานี 12130

ประวัติการศึกษา

ประถมศึกษา

โรงเรียนบูรณวิทย์ จังหวัดกรุงเทพมหานคร

มัธยมศึกษาตอนต้น

โรงเรียนโพธิสารพิทยากร จังหวัดกรุงเทพมหานคร

ประกาศนียบัตรวิชาชีพ

วิทยาลัยเทคนิคดอนเมือง จังหวัดกรุงเทพมหานคร

ประกาศนียบัตรวิชาชีพชั้นสูง

วิทยาลัยเทคนิคดอนเมือง จังหวัดกรุงเทพมหานคร

ปริญญาตรี

สาขาวิศวกรรมโทรคมนาคม

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ความสนใจพิเศษ

เทคโนโลยีคอมพิวเตอร์

คติเตือนใจ

ฝันให้ไกลแล้วไปให้ถึง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง



ชื่อ - สกุล

นางสาวกชฉัตร สิงห์ดารา

วันเดือนปีเกิด

25 มิถุนายน พ.ศ. 2527

ภูมิลำเนา

62 หมู่ 11 ตำบลรอบเมือง อำเภอเมือง
จังหวัดร้อยเอ็ด 45000

ประวัติการศึกษา

ประถมศึกษา

โรงเรียนบ้านเปลือยนอก จังหวัดร้อยเอ็ด

มัธยมศึกษาตอนต้น

โรงเรียนสตรีศึกษา จังหวัดร้อยเอ็ด

ประกาศนียบัตรวิชาชีพ

วิทยาลัยเทคนิคร้อยเอ็ด จังหวัดร้อยเอ็ด

ประกาศนียบัตรวิชาชีพชั้นสูง

วิทยาลัยเทคนิคร้อยเอ็ด จังหวัดร้อยเอ็ด

ปริญญาตรี

สาขาวิชาวิศวกรรมโทรคมนาคม

ภาควิชาครุศาสตร์วิศวกรรม

คณะครุศาสตร์อุตสาหกรรม สจล.

ความสนใจพิเศษ

เทคโนโลยีคอมพิวเตอร์

คติเตือนใจ

อย่ากลัวในสิ่งที่ยังไม่ได้ลงมือทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง



ชื่อ - สกุล

นายอัมพวัฒน์ พันธ์พัฒนชัย

วันเดือนปีเกิด

24 มีนาคม พ.ศ. 2528

ภูมิลำเนา

เลขที่ 58/1 หมู่ 2 ตำบลยางซ้าย อำเภอเมือง
จังหวัดสุโขทัย 64000

ประวัติการศึกษา

ประถมศึกษา

โรงเรียนอนุบาลสุโขทัย จังหวัดสุโขทัย

มัธยมศึกษาตอนต้น

โรงเรียนสุโขทัยวิทยาคม จังหวัดสุโขทัย

ประกาศนียบัตรวิชาชีพ

วิทยาลัยเทคนิคสุโขทัย จังหวัดสุโขทัย

ประกาศนียบัตรวิชาชีพชั้นสูง

วิทยาลัยเทคนิคสุโขทัย จังหวัดสุโขทัย

ปริญญาตรี

สาขาวิชาวิศวกรรมโทรคมนาคม

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ อดุลยเดชวิกรม

ความสนใจพิเศษ

การสื่อสารโทรคมนาคมและเทคโนโลยีคอมพิวเตอร์

คติเตือนใจ

ทำดีได้ดี ทำชั่วได้ชั่ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้