

ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล.

การพัฒนา Database Application บน Pocket PC
ด้วยเทคโนโลยี MobiLink Synchronization
Database Application on Pocket PC by MobiLink
Synchronization Technology



รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
ภาคเรียนที่ 2 ปีการศึกษา 2547
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

วัน เดือน ปี.....	21 ก.พ. 2550
เลขทะเบียน.....	02273
เลขเรียกหนังสือ.....	0๗. ๓ ๒๓4ก 2547
"ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล."	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในห้องสมุดเท่านั้น ไม่ให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ	การพัฒนา Database Application บน Pocket PC ด้วยเทคโนโลยี MobiLink Synchronization
นักศึกษา	นายธีระวัฒน์ ทองอนันต์
อาจารย์ที่ปรึกษา	รศ.ดร.อาริต ธรรมโน
ระดับการศึกษา	วิทยาศาสตร์มหาบัณฑิต สาขาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2547

บทคัดย่อ

การนำเทคโนโลยีเข้ามาทดแทนบางสิ่ง เป็นเรื่องที่ CEO และ CIO ต้องตระหนักตามเพราะทุกสิ่งที่ได้ เกี่ยวข้องกับการเปลี่ยนแปลงและการลงทุน รวมถึงมีผลต่อวัฒนธรรมองค์กร

เทคโนโลยี Unwired Enterprise อาจไม่ใช่คำตอบ แต่มั่นใจได้ว่าประสิทธิภาพและความถูกต้องจะมีมากขึ้นแน่นอน โดยเฉพาะความรวดเร็วในการให้บริการลูกค้าและช่วยเร่งการตัดสินใจในกรณีที่ต้องการข้อมูลทันทีเพื่อการบริหารที่มีความจำเป็นต้องทันเหตุการณ์

MobiLink คือ เทคโนโลยีการ synchronize ข้อมูล โดยเป็นการทำงานแบบสองทิศทาง คือ จะทำการ synchronize ข้อมูลระหว่าง ฐานข้อมูลหลัก ซึ่งเรียกว่า "Consolidate database" และ ฐานข้อมูลลูกข่าย ซึ่งเรียกว่า "Remote database" consolidate database สามารถเป็นฐานข้อมูลค่ายใดก็ได้แต่มีเงื่อนไขคือ ต้องติดต่อกับ MobiLink โดยผ่าน ODBC ที่สนับสนุนฐานข้อมูลนั้นๆ ส่วน Remote databases ต้องเป็นฐานข้อมูลของค่าย Sybase ที่ชื่อว่า Adaptive Server Anywhere หรือ UltraLite จึงจะสามารถทำงานร่วมกับ MobiLink ได้

Title	Database Application on Pocket PC by MobiLink Synchronization Technology
Student	Mr. Teerawat Tonganun
Advisor	Assoc. Prof. Dr. Arit Tummano
Level of Study	Master of Science in Information Technology
Academic Year	2004

ABSTRACT

The technology is instead something so CEO and CIO have to concerned because the result of technology affect to the investment including the culture of organization.

Unwired Enterprise technology maybe not absolutely answer but make convince the efficiency and accuracy to much more especially the rapid for severing customer and make decision incase desire the information immediately for management of current situation.

MobiLink is a session-based synchronization system that allows two-way synchronization between a main database, called the consolidated database, and many remote databases. The consolidated database, which can be one of several ODBC-compliant databases, holds the master copy of all the data. Remote databases can be either Adaptive Server Anywhere or UltraLite databases.

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
สารบัญ	III
สารบัญตาราง	VI
สารบัญรูป.....	VII
บทที่	
1. บทนำ.....	1
1.1 ความเป็นมา.....	1
1.2 วัตถุประสงค์ของการศึกษา.....	2
1.3 ขอบเขตของการศึกษา.....	3
1.4 ขั้นตอนและวิธีการดำเนินงาน.....	4
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	4
2. แนวคิดและทฤษฎีที่เกี่ยวข้อง.....	5
2.1 ทำความรู้จักกับพ็อกเก็ตพีซี.....	5
2.2 พีเจอร์มาตรฐานของพ็อกเก็ตพีซี.....	7
2.3 ประวัติศาสตร์ Pocket PCs & Win CE.....	13
2.4 การเริ่มต้นใช้งาน Pocket PC.....	16
2.5 การใช้งานสไตลัส.....	17
2.6 โปรแกรม ActiveSync.....	17
2.7 การสร้าง Partnership ของ ActiveSync.....	19
2.8 การยกเลิกการติดตั้งโปรแกรมใน Pocket PC.....	20
2.9 เกี่ยวกับ Pocket PowerBuilder.....	21
2.10 เกี่ยวกับ SQL Anywhere Studio.....	22
2.11 แนะนำการใช้งาน Adaptive Server Anywhere.....	23
2.12 การสร้างฐานข้อมูลจาก Adaptive Server Anywhere.....	24

สารบัญ (ต่อ)

	หน้า
2.13 แนะนำ MobiLink Synchronization.....	28
2.14 เกี่ยวกับ MobiLink synchronization.....	30
2.15 การ Set ค่าต่างๆ ในฐานข้อมูลลูกข่าย (Remote Database) เพื่อใช้ในการ synchronization.....	33
3. การออกแบบระบบ.....	40
3.1 ขอบเขตของระบบ.....	40
3.2 ขั้นตอนการทำงานของโปรแกรม.....	40
3.3 การออกแบบหน้าจอการทำงาน (User Interface Design).....	43
3.4 การออกแบบฐานข้อมูล.....	51
3.5 Physical Data Model.....	57
3.6 Entity Relationship Diagram.....	58
4. การพัฒนาโครงการ.....	59
4.1 ฟังก์ชันต่างๆเพื่อเรียกใช้งาน.....	59
4.2 หน้าจอลงทะเบียน.....	67
4.3 พัฒนาหน้าจอแสดงรายการลูกค้าที่เคลม.....	69
4.4 หน้าจอสำหรับการ synchronization.....	73
4.5 หน้าจอบันทึกและแสดงรายการ ชิ้นส่วนที่ลูกค้าต้องการเคลม.....	74
4.6 หน้าจอสำหรับทำงานเกี่ยวกับรูปภาพที่ถ่ายจากที่เกิดเหตุ.....	78
5. การทดลองและผลที่ได้.....	80
5.1 อุปกรณ์และเครื่องมือที่ใช้.....	80
5.2 การทดลอง.....	80
5.3 ผลการทดลอง.....	84
5.4 สรุปผลการทดลอง.....	84
5.5 ปัญหาและอุปสรรคที่พบในการพัฒนาโครงการ.....	85
5.6 สรุป.....	85

สารบัญ (ต่อ)

	หน้า
บรรณานุกรม.....	86
ประวัติผู้เขียน.....	87



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่

2.1 เปรียบเทียบ Pocket PCs ที่ใช้ระบบปฏิบัติการ Windows Mobile 2003 (Part 1).....	8
2.2 เปรียบเทียบ Pocket PCs ที่ใช้ระบบปฏิบัติการ Windows Mobile 2003 (Part 2).....	11



สารบัญรูป

รูปที่	หน้า
2.1 เครื่อง Pocket PC.....	5
2.2 Windows CE1.0.....	13
2.3 เครื่อง HP 300LX.....	14
2.4 PDA ที่ใช้โอเอส PocketPC 2000 (Windows CE 3.0).....	15
2.5 HP iPAQ h2210 ที่ใช้โอเอส PocketPC 2003.....	16
2.6 Styluses.....	17
2.7 โปรแกรม Microsoft ActiveSync บน PC.....	18
2.8 โปรแกรม Microsoft ActiveSync บน Pocket PC.....	18
2.9 การสร้าง Partnership ของ ActiveSync.....	19
2.10 หน้าจอเพื่อยกเลิกการติดตั้ง.....	20
2.11 หน้าจอเพื่อเลือกโปรแกรมที่ต้องการยกเลิกการติดตั้ง.....	20
2.12 หน้าจอเพื่อสร้างฐานข้อมูล.....	25
2.13 หน้าจอสุดท้ายของการสร้างฐานข้อมูล.....	26
2.14 หน้าจอการสร้างตารางในฐานข้อมูล.....	26
2.15 หน้าจอสุดท้ายของการสร้างตาราง.....	27
2.16 หน้าจอการตั้งค่าคอลัมน์ของตาราง.....	27
2.17 โครงสร้างของ MobiLink.....	31
2.18 หน้าจอการติดต่อเข้าสู่ฐานข้อมูลลูกข่าย.....	33
2.19 หน้าจอการสร้าง Publication.....	34
2.20 หน้าจอการตั้งชื่อ Publication.....	35
2.21 หน้าจอการเลือก Table สำหรับ Publication.....	36
2.22 หน้าจอการเลือก Column สำหรับ Publication.....	37
2.23 หน้าจอการสร้าง MobiLink User.....	38
2.24 หน้าจอการตั้งชื่อ MobiLink User.....	39

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.1 Flowchart แสดงขั้นตอนการทำงานของโปรแกรม ส่วนที่1.....	41
3.2 Flowchart แสดงขั้นตอนการทำงานของโปรแกรม ส่วนที่2.....	42
3.3 หน้าจอลงทะเบียนการใช้งานโปรแกรม.....	43
3.4 หน้าจอแสดงรายการลูกค้าที่เคลม.....	44
3.5 หน้าจอสำหรับการ synchronization.....	45
3.6 หน้าจอแสดง credit ของโปรแกรม.....	46
3.7 หน้าจอบันทึกและแสดงรายการ ชั้นส่วนที่ลูกค้าต้องการเคลม.....	46
3.8 หน้าจอบันทึกรายละเอียดเพิ่มเติมเกี่ยวกับการเคลมประกัน.....	47
3.9 หน้าจอสำหรับทำงานเกี่ยวกับรูปภาพที่ถ่ายจากที่เกิดเหตุ.....	48
3.10 หน้าจอสำหรับแก้ไขรูปภาพ.....	48
3.11 หน้าจอสำหรับเพิ่มรูปภาพ.....	49
3.12 หน้าจอสำหรับดูรายละเอียดของผู้เคลมประกัน.....	50
3.13 หน้าจอสำหรับดูรายละเอียดของตัวประกันที่ได้ทำไว้.....	50
3.14 Table : claim.....	51
3.15 Table : claim_images_NEW.....	52
3.16 Table : claim_items.....	52
3.17 Table : claim_status.....	53
3.18 Table : doodles_NEW.....	53
3.19 Table : employee.....	53
3.20 Table : points_NEW.....	54
3.21 Table : policy.....	54
3.22 Table : policy_holder.....	55
3.23 Table : precanned_claim_items.....	56
3.24 Physical Diagram.....	57

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.25 Entity Relationship Diagram.....	58
4.1 พัฒนาหน้าจอลงทะเบียน.....	67
4.2 พัฒนาหน้าจอแสดงรายการลูกค้าที่เคลม.....	69
4.3 พัฒนาหน้าจอสำหรับการ synchronization.....	73
4.4 พัฒนาหน้าจอบันทึกและแสดงรายการ ชิ้นส่วนที่ลูกค้าต้องการเคลม.....	74
4.5 พัฒนาหน้าจอสำหรับทำงานเกี่ยวกับรูปภาพที่ถ่ายจากที่เกิดเหตุ.....	78
5.1 หน้าต่างแสดงการทำงานของ MobiLink server.....	81
5.2 การลงทะเบียนเพื่อใช้งาน โปรแกรม.....	82
5.3 แสดงข้อมูลที่ได้มาจากการ synchronize.....	82
5.4 ทดสอบการเพิ่มข้อมูล.....	83
5.5 ทดสอบการเลือกรูปและขีดลายเส้น.....	83
5.5 หน้าจอการ Synchronize ข้อมูลจาก Pocket PC ไปยังฐานข้อมูลหลัก.....	84

บทที่ 1

บทนำ

1.1 ความเป็นมา

การนำเทคโนโลยีเข้ามาทดแทนบางสิ่ง เป็นเรื่องที่ CEO และ CIO ต้องตระหนักตาม เพราะทุกสิ่งได้เกี่ยวข้องกับการเปลี่ยนแปลงและการลงทุน รวมถึงมีผลต่อวัฒนธรรมองค์กร

จากทุกวันที่เรานั่งทำงานตั้งแต่เช้าจรดเย็นอยู่ที่สำนักงานมีเพื่อนร่วมงานหรือเจ้านายไปทานอาหารร่วมกัน เกิดบรรยากาศของความใกล้ชิดสนิทสนม เมื่อมีเทคโนโลยีเข้ามา ทำให้ไม่มีความจำเป็นต้องเข้ามาทำงาน เพราะใช้เวลาส่วนใหญ่อยู่ข้างนอกได้ ทำให้ความใกล้ชิดระหว่างบุคคลมีน้อยตามลงไป ผลคือผลเสียย่อมมาคู่กันเสมอ ดังนั้นผู้บริหารต้องคิดคำนวณว่าจะมีประโยชน์มากกว่ากัน

ความคิดรวบยอดของ Unwired Enterprise อาจไม่ใช่คำตอบ แต่มั่นใจได้ว่าประสิทธิภาพและความถูกต้องจะมีมากขึ้นแน่นอน โดยเฉพาะความรวดเร็วในการให้บริการลูกค้า และช่วยเร่งการตัดสินใจในกรณีที่ต้องการข้อมูลทันที เพื่อการบริหารที่มีความจำเป็นต่อทันเหตุการณ์

สำหรับตัวอย่างการใช้งานของระบบไร้สายเพื่อการรับส่งข้อมูลที่รวดเร็วและเห็นประโยชน์ได้ชัดเจนนั้น มีตัวอย่างเช่น ในการแข่งขันกีฬากอล์ฟรายการใหญ่ ๆ เช่น European Tour ซึ่งมีกาถ่ายทอดไปทั่วโลกนั้น การรายงานผลทางห้องข่าวย่อมมีความสำคัญมากที่จะต้องได้รับข้อมูลจากการแข่งขันทุกหลุมในสนามชนิดที่รวดเร็ว เพื่อปรับปรุง Leader Board ตลอดเวลารวมถึงข้อมูลส่วนบุคคลของนักกอล์ฟที่ทำให้รายงานสามารถสร้างสีสันแก่ผู้ชมการแข่งขันได้ เช่น การรายงานระยะจากจุดที่ดี การเลือกใช้ไม้ เป็นต้น

จากตัวอย่างที่กล่าวถึงนี้ ได้ใช้เทคโนโลยีที่สามารถรับส่งข้อมูลได้รวดเร็ว มีระบบฐานข้อมูลอยู่บนอุปกรณ์มือถือ ใช้การเขียนหรือกดข้อมูลแล้วส่งผ่านระบบไร้สายไปยังห้องรายงานการแข่งขันได้ตลอดเวลาทุกครั้งที่มีการตีลูก

ในที่สุดก็มีคำตอบคือ Mobile Devices ที่เป็นเครื่อง PDA บรรจุข้อมูลการแข่งขันไว้อย่างครบถ้วน เจ้าหน้าที่สามารถเดินตามนักกอล์ฟที่แข่งขัน และรายงานผลได้อย่างมีประสิทธิภาพ โดยไม่รบกวนนักกีฬาที่ทำการแข่งขัน และยังมี Back Up ข้อมูลไว้ทั้งสอง ทางเพื่อตรวจสอบอีกด้วย

อีกตัวอย่างหนึ่งที่มีความน่าสนใจและมีการนำเทคโนโลยีไร้สายไปใช้ได้ผลมาก คือ ระบบบริหารจัดการข้อมูลของระบบประกันภัยรถยนต์ ที่มีความรวดเร็วและทันสมัยเหมาะสมกับสถานการณ์ที่ต้องเผชิญกับรถติดและเสียเวลาอย่างมาก

เมื่อเกิดอุบัติเหตุบนท้องถนน ผู้ประสบเหตุจะต้องเสียเวลาในการรอเจ้าหน้าที่ประกัน เป็นเวลานาน ต้องรอคอยการเจรจาและการอนุมัติจากตัวแทนเจ้าหน้าที่กับสำนักงาน แล้วต้องนำรถไปที่อยู่เพื่อตีราคาว่าจะได้ข้อสรุปอาจใช้เวลานานมาก

บริษัทประกันที่ใส่ใจในการบริการที่รวดเร็วจะอยู่รอดได้ในการแข่งขันทางธุรกิจ ซึ่งไม่จำเป็นต้องมีเบี้ยประกันที่ราคาถูกเป็นตัวกำหนด เบี้ยประกันจะเป็นตัวบ่งบอกคุณภาพและความสะดวกรวดเร็วมากกว่าความมั่นคงของบริษัท

การลงทุนเทคโนโลยีที่กล่าวถึงไม่ได้มีมูลค่าสูงแต่อย่างใด และยังเป็นการลงทุนที่ง่ายของ บริษัทอีกด้วย การรับส่งรายงานข้อมูลอุบัติเหตุและกรรมธรรมที่รวดเร็วสามารถทำได้ด้วยเทคโนโลยีไร้สาย ไปยังตัวแทนของบริษัทที่อยู่ใกล้ที่เกิดเหตุมากที่สุดเพื่อ ไปถึงที่เกิดเหตุได้รวดเร็ว

เมื่อถึงที่เกิดเหตุตัวแทนสามารถตรวจสอบความถูกต้องของผู้ขับขี่และคู่กรณีได้ทันที มีการรายงานความเสียหายและกำหนดบริเวณที่ต้องมีการซ่อมเพื่อขออนุมัติ พร้อมรูปถ่ายผ่านระบบไร้สายไปยังสำนักงานใหญ่ได้ทันที พร้อมการประเมินราคาเบื้องต้น โดยสำนักงานใหญ่ สามารถรับข้อมูลทุกขั้นตอนผ่านทางหน้าจอคอมพิวเตอร์และสามารถส่งข้อความอนุมัติ หรือขอข้อมูลเพิ่มเติมได้ทันที จากนั้นพนักงานตัวแทนสามารถให้ผู้เสียหายสามารถเลือกซ่อมได้จากฐานข้อมูลที่มีอยู่ได้ทันที

ทั้งหมดที่กล่าวมานี้คือ การทำงานร่วมกันระหว่างอุปกรณ์ PDA ที่มีฐานข้อมูลที่มีประสิทธิภาพ และระบบจัดการรับส่งข้อมูลผ่านระบบไร้สายที่รวดเร็ว ด้วยผู้ให้บริการเครือข่ายและระบบประมวลผลกลางที่สำนักงานใหญ่ ซึ่งสามารถทำได้ง่ายและกระทำได้ทันที

ค่าใช้จ่ายเล็กน้อย บางครั้งก็สมควรแก่การลงทุนเพื่อเพิ่มประสิทธิภาพในการบริการลูกค้า และช่วยประเทศชาติประหยัดพลังงานอีกด้วย

1.2 วัตถุประสงค์ของการศึกษา

1. เพื่อศึกษาและทำความเข้าใจเกี่ยวกับระบบการทำงานเบื้องต้นบางส่วนของระบบประกันภัย
2. เพื่อศึกษาและทำความเข้าใจวิธีการใช้งานฐานข้อมูล Adaptive Server Anywhere เพื่อใช้เป็นฐานข้อมูลใน Pocket PC

3. เพื่อศึกษาและทำความเข้าใจวิธีการใช้งาน Pocket Builder เพื่อเป็น tool ในการเขียนโปรแกรมใน Pocket PC
4. เพื่อศึกษาและทำความเข้าใจวิธีการใช้งาน Mobilink Synchronization Technology เพื่อใช้เป็นกลไกในการ synchronized ข้อมูล ระหว่าง ฐานข้อมูลหลัก และ ฐานข้อมูลใน Pocket PC
5. เพื่อนำโปรแกรมประยุกต์ที่ได้พัฒนาขึ้นนี้ ไปเป็นโปรแกรมตัวอย่าง ในการนำเสนอขาย software (Pocket Builder) ให้แก่ลูกค้า
6. เพื่อให้โปรแกรมประยุกต์ที่ได้พัฒนาขึ้นนี้ เป็นต้นแบบในการปรับปรุงเป็น version ที่สมบูรณ์ยิ่งขึ้น เพื่อนำไปขายแก่ลูกค้าในวงการธุรกิจต่างๆ
7. เพื่อทำการศึกษาคูณลักษณะและข้อจำกัดของ Pocket PC เพื่อให้การพัฒนาโปรแกรมประยุกต์เป็นไปในแนวทางที่ถูกต้อง สามารถทำงานได้บนเครื่อง Pocket PC

1.3 ขอบเขตของการศึกษา

1. ทั้งฐานข้อมูลหลัก และ ฐานข้อมูลบนเครื่อง Pocket PC เป็น ฐานข้อมูล Adaptive Server Anywhere ของบริษัท Sybase เนื่องจากเป็นฐานข้อมูลที่สามารถใช้งานได้ง่าย ไม่ต้องทำการ Tuning มากนัก ที่สำคัญคือ มีประสิทธิภาพมากในการใช้งานบน mobile devices และรองรับการ synchronized ข้อมูล ด้วยเทคโนโลยี Mobilink Synchronization
2. ใช้ Pocket Builder เป็น Tool ในการเขียนโปรแกรม เนื่องจากมี features ต่างๆ ที่สนับสนุนการพัฒนาโปรแกรมบนเครื่อง Pocket Builder อย่างครบครัน และสนับสนุนการทำงานร่วมกันกับเทคโนโลยี Mobilink Synchronization
3. โปรแกรมประยุกต์ที่พัฒนาขึ้นสามารถทำงานหลักๆดังนี้
 - a. สามารถ Synchronized ข้อมูลไป-กลับ ระหว่าง Pocket PC กับ Server ไม่ว่าจะ เป็นข้อมูลชนิด text หรือ รูปภาพ
 - b. สามารถ บันทึกหรือแก้ไขข้อมูลต่างๆลงสู่ฐานข้อมูลได้
 - c. สามารถขีดเขียนลายเส้นบนรูป ที่ถ่ายจากที่เกิดเหตุได้
 - d. สามารถบันทึกการตั้งค่าต่างๆของระบบลงสู่ register ของเครื่อง Pocket PC ได้ เช่น หมายเลข IP Address ของเครื่องปลายทางที่ต้องการ synchronized ข้อมูล

1.4 ขั้นตอนและวิธีการดำเนินงาน

1. ศึกษาและทำความเข้าใจเกี่ยวกับระบบการทำงานเบื้องต้นบางส่วนของระบบประกันภัย
2. ศึกษาในส่วนของเครื่อง Pocket PC ได้แก่ วิธีการใช้เครื่อง, การติดตั้งโปรแกรม และการใช้โปรแกรม ActiveSync กับเครื่อง Pocket PC
3. ศึกษาและทำความเข้าใจวิธีการใช้งานฐานข้อมูล Adaptive Server Anywhere
4. ศึกษาและทำความเข้าใจวิธีการใช้งาน Pocket Builder
5. ศึกษาและทำความเข้าใจวิธีการใช้งาน Mobilink Synchronization Technology
6. ออกแบบและพัฒนาระบบงาน
7. ทำการทดสอบระบบงาน
8. สรุปผลการศึกษา
9. จัดทำเอกสารประกอบการจัดทำโครงการงาน

1.5 ประโยชน์ที่คาดว่าจะได้รับ

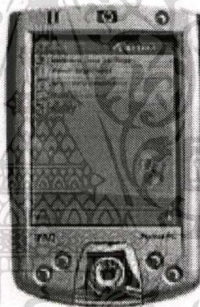
1. ได้โปรแกรมประยุกต์ทางด้านฐานข้อมูลบนเครื่อง Pocket PC ที่สามารถ Synchronized ข้อมูลในฐานข้อมูลไปยังฐานข้อมูลหลักที่เครื่องปลายทางได้
2. เข้าใจระบบการทำงานบางส่วนของระบบประกันภัย ที่เกี่ยวข้องกับขอบเขตของการพัฒนาโครงการงาน
3. ทักษะการใช้งานฐานข้อมูล Adaptive Server Anywhere, การเขียนโปรแกรมโปรแกรมบนเครื่อง Pocket PC โดยใช้ Pocket Builder และการใช้งานเครื่อง Pocket PC
4. ความรู้และเทคนิคในการ Synchronized ข้อมูลในฐานข้อมูล โดยวิธี Mobilink Synchronization
5. ได้โปรแกรมต้นแบบในการประยุกต์และพัฒนา เพื่อให้ครอบคลุมปัจจัยต่างๆและมีความสมบูรณ์เพิ่มขึ้นมากกว่านี้ เพื่อให้สามารถนำไปเสนอขายได้
6. สามารถนำมาเป็นแนวทางในการพัฒนาระบบงานลักษณะนี้สำหรับผู้ที่สนใจต่อไป
7. ปัญหาและอุปสรรคเพื่อนำมาทำการศึกษาและหาวิธีแก้ไขต่อไป

บทที่ 2

แนวคิดและทฤษฎีที่เกี่ยวข้อง

2.1 ทำความรู้จักกับพ็อกเก็ตพีซี

พ็อกเก็ตพีซี (Pocket PC) คือชื่อของระบบปฏิบัติการ และพ็อกเก็ตพีซี 2002 (Pocket PC 2002) คือระบบปฏิบัติการ Pocket PC ที่ได้รับการพัฒนา และปรับปรุงแก้ไขให้ทำงานดียิ่งขึ้น มีความสามารถและประสิทธิภาพที่สูงขึ้น นอกจากนี้คำว่าพ็อกเก็ตพีซียังถูกใช้เป็นชื่อเรียกของเครื่องพีดีเอ (PDA : Personal Digital Assistant) หรือเครื่องคอมพิวเตอร์มือถือแบบพกพาจากผู้ผลิตชั้นนำทั่วโลก อาทิ Compaq, HP, Casio, Fujitsu, Toshiba, Audiovox, NEC, Mitac, Packard Bell, Acer ฯลฯ



รูปที่ 2.1 เครื่อง Pocket PC

ความสามารถพื้นฐานของพ็อกเก็ตพีซี คือ การสร้างและจัดการกับการนัดหมาย รายการที่อยู่ติดต่อ รายการงานที่ต้องทำ การบันทึกข้อมูลด้วยข้อความหรือเสียง การอ่านหรือฟังอีบุ๊ก การทำงานด้านเวิร์ดโปรเซสเซอร์ ตารางข้อมูลการจัดการด้านการเงิน การเชื่อมต่อและท่องอินเทอร์เน็ต การรับและส่งอีเมลการฟังเพลงและชมไฟล์ภาพยนตร์ การเล่นเกม รวมทั้งการติดตั้งโปรแกรมใหม่ๆ เพิ่มเติมเพื่อจุดประสงค์ต่างๆ และการเพิ่มประสิทธิภาพและความยืดหยุ่นในการใช้งาน นอกจากนี้ยังสามารถแลกเปลี่ยนข้อมูลกับคอมพิวเตอร์พีซี เพื่อแลกเปลี่ยนและปรับปรุงข้อมูลให้สอดคล้องกันได้อย่างมีประสิทธิภาพ

Pocket PC OS นั้นต่างกับ Windows CE ที่ Pocket PC คือ Windows CE เวอร์ชันที่ได้รับการปรับปรุงโดยไมโครซอฟท์ ประกอบด้วยโปรแกรมต่างๆ (รวมทั้ง APIs สำหรับการพัฒนา

โปรแกรม) ที่ไม่มีอยู่ใน Windows CE 3.0 ในทางกลับกัน Windows CE ก็มีส่วประกอบที่ไม่มีอยู่ใน Pocket PC OS

สำหรับความแตกต่างระหว่าง Windows CE 3.0 OS และ Pocket PC OS คือ Pocket PC OS ประกอบด้วยเคอร์เนลที่ได้รับการปรับปรุงเพื่อเพิ่มประสิทธิภาพในการทำงาน รวมทั้งสนับสนุนการใช้งานร่วมกับ ActiveSync 3.1 ขึ้นไป ในทางกลับกัน Windows CE 3.0 ประกอบด้วย Distributed Component Object Model (DCOM) ซึ่งไม่มีอยู่ใน Pocket PC OS และสนับสนุนการใช้งานร่วมกับ ActiveSync 3.0 หรือก่อนหน้านี

นอกจากนี้ แม้ว่าเครื่องพีคอกเก็ตพีซีจากบริษัทต่างๆ จะหันมาใช้ซีพียู Intel StrongARM206 MHz เหมือนๆกัน แต่เครื่องจากผู้ผลิตแต่ละแห่งก็ยังมีรายละเอียดปลีกย่อยในส่วนของ OS ที่แตกต่างกันออกไป เช่น การเพิ่มความสามารถหรือโปรแกรมบางส่วนเพิ่มเติมเข้าไปในเครื่องของตน ดังนั้นเครื่องแต่ละยี่ห้อจึงอาจจะมีโปรแกรมและไดรเวอร์บางส่วนแตกต่างกัน

สำหรับผู้ที่ใช้เครื่องพีคอกเก็ตพีซีบางรุ่นและบางยี่ห้อ สามารถอัปเดตระบบปฏิบัติการ Pocket PC ไปใช้ Pocket PC 2002 ผ่านการแฟลชหน่วยความจำด้วยโปรแกรมชุดอัปเดตจากผู้ผลิต เช่น เครื่อง iPAQ ในตระกูล 3600 สามารถแฟลชได้ แต่เครื่อง HP Jornada 540 หรือ Casio Cassiopeia E-125 หรือ E-500 จะไม่สามารถทำได้

ปัจจุบันจะมีกลุ่มผู้ใช้พีคอกเก็ตพีซีหลัก ๆ ที่ได้รับความนิยมอยู่ 2 ระบบ คือ ระบบปฏิบัติการ (OS) เป็น Pocket PC 2002 กับระบบ Windows Mobile 2003 ที่เป็นระบบในปัจจุบัน ซึ่งอาจเรียกระบบปฏิบัติการทุกระบบของพีคอกเก็ตพีซีว่า Windows CE เครื่องพีคอกเก็ตพีซี ดูคล้ายกับ Windows บน PC แต่ก็ไม่ใช่ทั้งหมด เพราะมันคือ Windows บน PDA ส่วนที่คล้ายกันก็เช่น Start Menu, taskbar, desktop, Control Panels (หรือ Settings) ซึ่งส่วนทั้งหมดนี้สามารถเรียกใช้ได้ทันทีที่เปิดเครื่อง การเรียกใช้งาน สามารถทำได้โดยใช้ Stylus พีคอกเก็ตพีซีส่วนใหญ่จะมีหน้าจอขนาด 240 x 320 พิกเซล จำนวนสี 16 บิต 65,000 สี และทั้งหมดจะมีช่องใส่ Compact Flash หรือ SD หรือทั้งสองอย่าง เพื่อใช้เป็นที่เก็บข้อมูลและเป็นหน่วยความจำภายนอก ทำให้ไม่มีข้อจำกัดด้านปริมาณของไฟล์หรือข้อมูล เช่นเดียวกับ hard-disk บนเครื่อง PC และช่องสำหรับใส่การ์ดความจำเหล่านี้ สำหรับพีคอกเก็ตพีซีรุ่นใหม่ ๆ ยังอนุญาตให้การ์ดที่เป็นโมเด็มและเน็ตเวิร์กการ์ดได้ เป็นลักษณะของ IO Slots พีคอกเก็ตพีซีเข้ากันได้ดีในการเชื่อมต่อกับเครื่องพีซีที่ใช้ระบบ Windows แต่ก็มีปัญหา กับระบบของ Mac

2.2 พีเออร์มาตรฐานของพ็อคเก็ตพีซี

พ็อคเก็ตพีซีทุกเครื่องจะประกอบด้วย stereo headphone jack และ Windows Media Player ที่สามารถเล่นไฟล์ MP3 และไฟล์วิดีโอแบบ WMV ของมีเดียเพลเยอร์เองได้ทันทีโดยไม่ต้องตัดแปลง นอกจากนี้โปรแกรมประเภท Office ได้แก่ Pocket Word, Pocket Excel และ MS Money ก็มีมาให้อย่างครบครันโดยไม่ต้องหามาเพิ่มเติม บนพ็อคเก็ตพีซีทุกเครื่องจะมาพร้อมกับ MS ActiveSync ซึ่งเป็น desktop syncing แอปพลิเคชัน เพื่อใช้เชื่อมต่อระหว่างพ็อคเก็ตพีซีกับเครื่องพีซี การเชื่อมต่อแบบ ActiveSync นี้ จะเป็นการเชื่อมต่อข้อมูลใน Outlook เพื่อแลกเปลี่ยนข้อมูลระหว่างพ็อคเก็ตพีซีกับพีซี เช่น mail, calendar, contact, task เป็นต้น ActiveSync จะสนับสนุนการเชื่อมต่อข้อมูลแบบ "partnerships" ซึ่งเป็นการกำหนดชื่อผู้ใช้ กับเครื่องพีซี 2 เครื่องได้ เช่น เครื่องพีซีที่บ้าน และกับที่ทำงาน เป็นต้น ขณะเดียวกันยังใช้พ็อคเก็ตพีซีเพื่อรับส่งไฟล์หรือแอปพลิเคชันกับเครื่องพีซีอื่น ๆ ได้ในโหมดของ guest พ็อคเก็ตพีซีที่เป็นระบบ Pocket PC 2002 มาจนถึง Windows Mobile 2003 มักจะมาพร้อมกับแท่นเชื่อมต่อ (cradle connection) ที่เป็น USB

พ็อคเก็ตพีซีส่วนใหญ่จะมีหน่วยความจำภายในคือ RAM 32 หรือ 64 MB แต่ก็อาจมี 128 หรือ 256 MB และยังมีหน่วยความจำภายในแบบถาวรคือ ROM ขนาด 16, 32 หรือ 48 MB ในส่วนของ ROM นี้จะใช้เป็นที่เก็บระบบปฏิบัติการและแอปพลิเคชันบางตัวแบบถาวรไว้ใน ROM ส่วนของ CPU ปัจจุบันจะเป็น StrongARM ที่ความเร็ว 206 MHz หรือ CPU ตัวใหม่คือ XScale ที่มีความเร็ว 300 และ 400 MHz เครื่องพ็อคเก็ตพีซีส่วนใหญ่จะมี SD card slot หรือ CompactFlash มาให้หนึ่งช่อง แต่ก็ยังมีบางรุ่นที่มีช่องเสียบมาสองช่องโดยสามารถใช้ได้ทั้ง SD และ CompactFlash เมโมรี่การ์ดเหล่านี้ต่างก็นิยมใช้กันอยู่แล้วกับกล่องคิวิตอลส่วนใหญ่ และช่องเหล่านี้มักจะใช้ได้กับส่วนขยายที่เป็น IO เช่น CF modems, Ethernet cards และ WIFI (802.11b wireless Ethernet networking) card และพ็อคเก็ตพีซีทุกเครื่องจะมี IrDA 1.2 infrared, stereo headphone jack, stylus, built-in speaker และ microphone, syncing cradle และ ปุ่มคำสั่งต่าง ๆ บนเครื่อง จะเป็นส่วนที่มีมาแบบมาตรฐานของพ็อคเก็ตพีซีทุกเครื่อง

ตารางที่ 2.1 เปรียบเทียบ Pocket PCs ที่ใช้ระบบปฏิบัติการ Windows Mobile 2003 (Part 1)

รุ่น	หน่วยประมวลผล	หน่วยความจำ	ระบบปฏิบัติการ	หน้าจอและการอินพุท	การเพิ่มขยาย
H4155	Intel PXA255 @400MHz	รวม 32MB / แรม 64MB	WM2003	Touch screen 3.5" diagonal, 240x320 pixel,16bit	SD slot (SDIO, SD,MMC)
H4355	Intel PXA255 @400MHz	รวม 32MB / แรม 64MB	WM2003	Touch screen 3.5" diagonal, 240x320 pixel,16bit Integrated QWERTY	SD slot (SDIO, SD,MMC)
H1930	Samsung S3C2410 @203MHz	รวม 16MB / แรม 64MB	WM2003 Professional	3.5" diagonal, 240x320 pixel,16bit	SD slot (SDIO, SD, MMC)
H1940	Samsung S3C2410 @266MHz	รวม 32MB / แรม 64MB	WM2003 Premium	3.5" diagonal, 240x320 pixel,16bit	SD slot (SDIO, SD, MMC)
H2210	Intel PXA255 @400MHz	รวม 32MB / แรม 64MB	WM2003 Premium	3.5" diagonal, 240x320 pixel,16bit	SD slot (SDIO, SD,MMC), CF slot (Type 1 & 2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รุ่น	หน่วยประมวลผล	หน่วยความจำ	ระบบปฏิบัติการ	หน้าจอและการอินพุต	การเพิ่มขยาย
H5550	Intel PXA255 @400MHz	รวม 48MB / แรม 128MB	WM2003 Premium	3.8" diagonal, 240x320 pixel, 16bit	SD slot (SDIO, SD, MMC)
Dell Axim X31	Intel XScale 400MHz	รวม 64MB (Strata Flash Rom - เป็น File Store 35MB) แรม 64MB	WM2003 Premium	3.5" diagonal, 240x320 pixel, 16bit	SD slot (SDIO, SD, MMC)
MyPal A620/A620BT	Intel PXA255 @400MHz	รวม 32MB / แรม 64MB	WM2003	3.5" diagonal, 240x320 pixel, 16bit	Compact Flash type II slot
Viewsonic A36/37	Intel PXA255 @400MHz	รวม 32MB / แรม 64MB	WM2003	3.5" diagonal, 240x320 pixel, 16bit 300K Camera	SD slot (SDIO, SD, MMC)
Toshiba e355	Intel XScale PXA255 @300MHz	รวม 16MB + แรม 64MB (ใช้งานได้จริง 50.5MB)	WM2003 Professional	3.5" diagonal, 240x320 pixel, 16bit	SD slot (SDIO, SD, MMC)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รุ่น	หน่วยประมวลผล	หน่วยความจำ	ระบบปฏิบัติการ	หน้าจอและการอินพุท	การเพิ่มขยาย
Toshiba e405	Intel XScale PXA261 @300MHz	รวม 16MB + 32MB Nand Flash Rom (เป็น iPAQ File Store 31.22MB) แรม 64MB (ใช้งานได้จริง 51.5MB)	WM2003	3.5" diagonal, 240x320 pixel,16bit	SD slot (SDIO, SD, MMC)
Toshiba e750/755	Intel PXA255 @400MHz	รวม 32MB / แรม 64MB	WM2003	3.8" diagonal, 240x320 pixel,16bit	Secure Digital (SD), CompactFlash type II slot
JVC iO Pocket PC MP- PV131, MP- PV331	Intel PXA255 @400MHz	รวม 32MB / แรม 128MB	WM2003	3.5" diagonal, 240x320 pixel,16bit	Secure Digital (SD), CompactFlash

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 เปรียบเทียบ Pocket PCs ที่ใช้ระบบปฏิบัติการ Windows Mobile 2003 (Part 2)

รุ่น	การเชื่อมต่อ	แบตเตอรี่	ขนาด	น้ำหนัก	อุปกรณ์มาตรฐาน
H4155	อินฟราเรด, บลูทูธ, WiFi (802.11)	Removable, rechargeable Lithium-Ion (1000 mAh)	2.78"Wx4.47"Hx0.5"D	4.67 ออนซ์	Cradle Sync & Charge
H4355	อินฟราเรด, บลูทูธ, WiFi (802.11)	Lithium Ion 1560mAh (removable)	138.7 x 74.6 x 14.5 mm	165 กรัม	Cradle Sync & Charge
H1930	IrDA	Li-Ion Removable 8 hr.	2.75"Wx4.46"Hx0.50"D	4.3 ออนซ์	Travel Charge & Sync
H1940	IrDA, Bluetooth	Li-Ion Removable 8 hr.	2.75"Wx4.46"Hx0.50"D	4.3 ออนซ์	Travel Charge & Sync
H2210	IrDA, Bluetooth	Li-Ion Removable 12 hr.	3"Wx4.54"Hx0.61"D	5.1 ออนซ์	Cradle Sync & Charge
H5550	IrDA, Bluetooth, WiFi (802.11, iPAQ Expansion Jacket)	Li-Ion Removable 1250 mAmp	3.31"Wx5.43"Hx0.63"D	7.29 ออนซ์	Cradle Sync & Charge

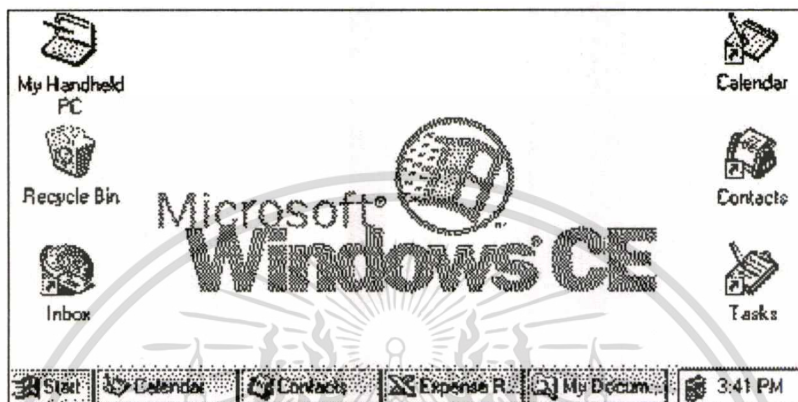
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รุ่น	การเชื่อมต่อ	แบตเตอรี่	ขนาด	น้ำหนัก	อุปกรณ์มาตรฐาน
Dell Axim X3i	IrDA, WiFi 802.11b	950 mAh Lithium	3.0"W x 4.6"H x 0.6"D	5 ออนซ์	Cradle Sync & Charge แถม Belt Clip Case
MyPal A620/A620BT	IrDA, Bluetooth	1300 mAh Lithium	125, 76.8, 13.3 mm	141 กรัม	AC Adapter, USB Cradle
Viewsonic A36/37	IrDA			4.8 ออนซ์	AC Adapter, USB Cradle
Toshiba e355	IrDA	900mAh Lithium Ion ถอดเปลี่ยน ไม่ได้	3.1"W x 4.9"H x 0.4"D	5.3 ออนซ์	AC Adapter, USB Cradle
Toshiba e405	IrDA	980mAh Lithium Ion ถอดเปลี่ยน ไม่ได้	3.0"W x 4.9"H x 0.4"D	4.7 ออนซ์	สายซิงค์ และชาร์จ อย่างละ 1 เส้น
Toshiba e750/755	802.11b	Lithium Ion, 3.7 volt 1000 mA, optional 300 mAh Lithium Ion, replaceable	4.9 x 3.10 x .6 inches	6.1 ออนซ์	AC Adapter, AC Power Cable, USB Cradle, soft case

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ประวัติศาสตร์ Pocket PCs & Win CE

Windows CE (Compact Edition) Version 1 ออกสู่ตลาดในปี 1996 ลักษณะระบบปฏิบัติการ มีลักษณะเหมือน Windows 95 ที่ลดความสามารถลงมาให้เหมาะกับอุปกรณ์ขนาดเล็ก การแสดงผลเป็นจอขาว/ดำ

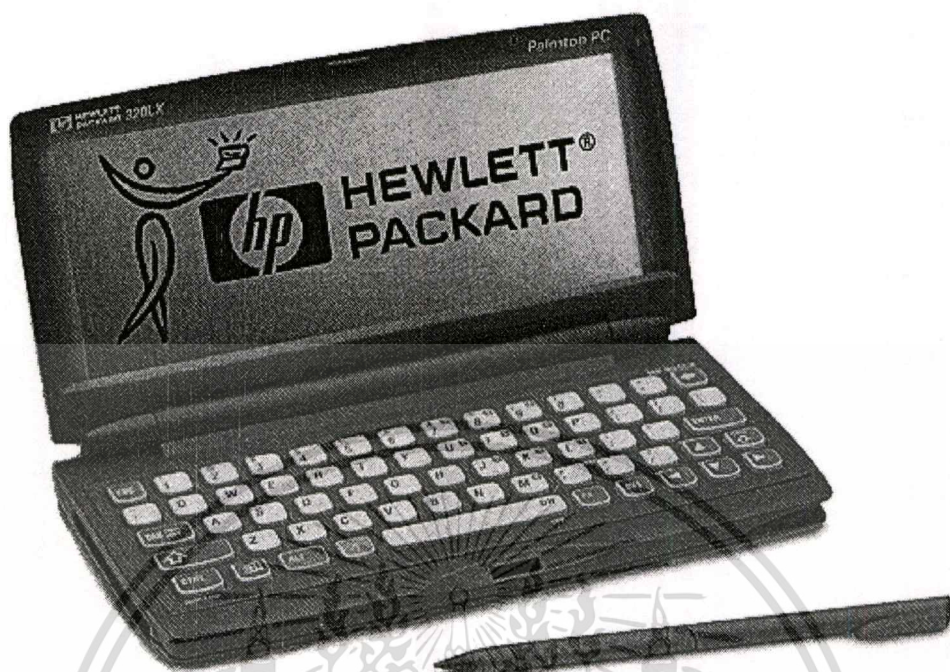


รูปที่ 2.2 Windows CE1.0

เครื่อง HP 300LX ที่ใช้ระบบปฏิบัติการ WindowsCE 1.0

- CPU SH3 @44MHz
- RAM 2-4MB
- Resolution 640x240 pixel

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 เครื่อง HP 300LX

Windows CE2.x วางตลาดช่วงปี 1997-1999 โดยเริ่มจากระบบปฏิบัติการ WindowsCE 2.0 ก่อน หลังจากนั้นในปี 1998 จึงมีการผลิต Palm- Size PC ที่ใช้ระบบปฏิบัติการ Windows CE2.01 และในปี 1999 ก็ผลิต Palm-Size PC ที่ใช้ระบบปฏิบัติการ Windows CE 2.11 ซึ่งต่อมาพัฒนาไปสู่ Pocket PC ในปัจจุบัน

คุณสมบัติเด่นๆ ของ Windows CE 2.0 ก็คือ

- รองรับ TrueType Font
- รองรับการแสดงผลเป็นสีได้ถึง 24bit
- สามารถใช้งาน PCMCIA ได้ (HPC)
- สามารถ Synchronize ผ่านระบบเน็ตเวิร์กได้

จุดเปลี่ยนที่สำคัญในยุคนี้อีกคือ Windows CE2.1 ซึ่งมีสิ่งที่เพิ่มเติมขึ้นมา คือ

- รองรับโพรโตคอล TCP/IP
- รองรับระบบไฟล์แบบ FAT32
- รองรับการใช้งานพอร์ต IrDA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

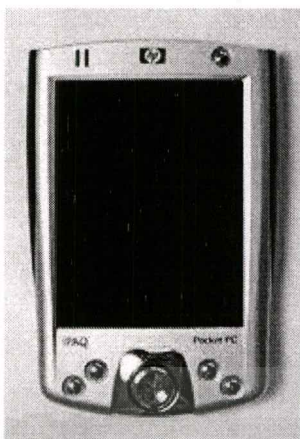
- รองรับการใช้งานร่วมกับพอร์ต USB
- ทำงานกับไฟล์ขนาดใหญ่ได้ดียิ่งขึ้น
- สามารถจัดเก็บไฟล์ข้อมูลเอาไว้บน Storage ภายนอกได้

Windows CE 3.x แบ่งการพัฒนาออกเป็นหลายๆ ส่วนคือ Handheld PC (HPC), Pocket PC, Smart Phone ส่วนที่พัฒนามาจนกลายเป็น PocketPC ก็คือ PocketPC 2000 ซึ่งเปิดตัวในปี 2000 และ PocketPC 2002 ในปี 2001 ลักษณะพิเศษคือตัดส่วนที่เป็น UI แบบ Windows ออกไปและยกเลิกปุ่ม Start และ Taskbar เปลี่ยนมาใช้รูปแบบอย่างในปัจจุบันแทน



รูปที่ 2.4 PDA ที่ใช้โอเอส PocketPC 2000 (Windows CE 3.0)

Windows CE 4.x รุ่นล่าสุดของสายการพัฒนาคือ Windows CE 4.x ซึ่งในส่วนของ PocketPC ก็คือ Windows Mobile 2003 หรืออาจเรียกว่า PocketPC 2003



รูปที่ 2.5 HP iPAQ h2210 ที่ใช้โอเอส PocketPC 2003

2.4 การเริ่มต้นใช้งาน Pocket PC

เมื่อเปิดเครื่องในครั้งแรก หรือในกรณีที่ทำการฮาร์ดรีเซ็ต จะพบกับขั้นตอนต่างๆ ในการตั้งค่าเริ่มต้นการทำงานของระบบ ดังนี้

1. หน้าจอแรกแสดงการต้อนรับจะปรากฏขึ้น ให้ทำการแตะพื้นที่ใดๆ บนหน้าจอเพื่อไปยังหน้าจอต่อไป
2. คำแนะนำในการใช้งานสไตลัส (Stylus) จะปรากฏขึ้น เพื่อให้ทราบว่า สามารถใช้สไตลัสในการแตะ และแตะค้างไว้เพื่อเปิดเมนูคำสั่งต่างๆ ให้ทำการแตะปุ่ม Next เพื่อไปยังหน้าจอต่อไป
3. ในหน้าจอที่ปรากฏขึ้น จะได้เรียนรู้การเลือกเปิดเมนูคำสั่งด้วยการแตะสไตลัสค้างไว้ ให้ทำการแตะสไตลัสบนรายการนัดหมายค้างไว้ครู่หนึ่ง เมนูจะปรากฏขึ้น ให้เลือกคำสั่ง Cut เพื่อย้ายการนัดหมายไปยังเวลาอื่นๆ
4. หน้าจอใหม่จะปรากฏขึ้น เพื่อให้ทำการวางรายการนัดหมายที่ย้ายในช่วงเวลาใหม่ ให้แตะในช่วงเวลาตามที่ถูกสไตลัสค้างไว้ครู่หนึ่ง เมนูจะปรากฏขึ้น ให้เลือกคำสั่ง Paste
5. จะเห็นว่าการนัดหมายถูกย้ายไปยังช่วงเวลาใหม่อย่างง่ายดาย จากนั้นให้แตะปุ่ม Next เพื่อไปยังหน้าจอต่อไป
6. ให้ตั้งเวลาให้ตรงกับพื้นที่ที่อาศัยอยู่ ในที่นี้คือ Bangkok สำหรับเวลาในประเทศไทย แล้วแตะปุ่ม Next
7. เมื่อการตั้งค่าต่างๆเสร็จเรียบร้อยแล้ว ให้แตะพื้นที่ใดๆ บนหน้าจอเพื่อเริ่มต้นการใช้งาน
8. หน้าจอ Today จะปรากฏขึ้นเพื่อให้เริ่มต้น ใช้งานพ็อกเก็ตพีซีได้ทันที

2.5 การใช้งานสไตลัส

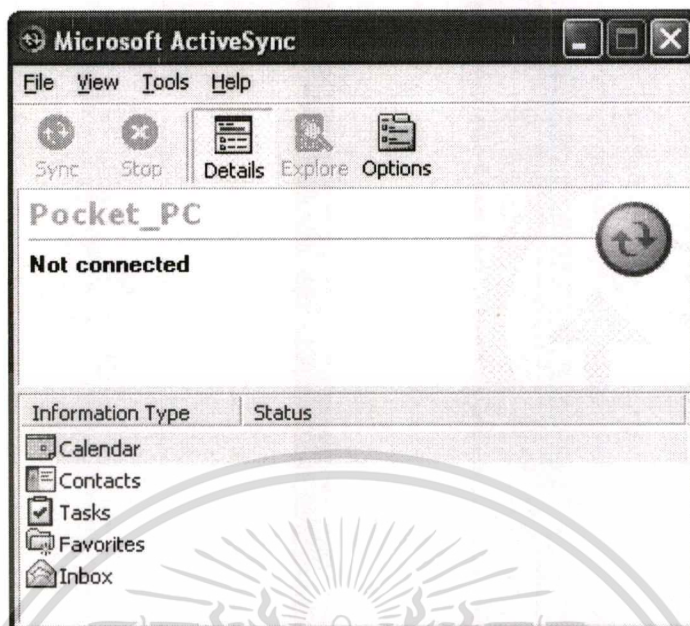
สไตลัส (Stylus) หรืออุปกรณ์ที่มีลักษณะคล้ายปากกา ซึ่งจะมาพร้อมกับเครื่องพีคกเกิดพีซีทุกๆเครื่อง สำหรับใช้ในการแตะหน้าจอของพีคกเกิดพีซีเพื่อใช้งาน โดยในการใช้งานสามารถทำได้ 3 ลักษณะ คือ

1. การแตะ คือ การนำสไตลัสแตะลงบนหน้าจอของพีคกเกิดพีซีเพื่อสั่งงานเช่น การเปิดเมนู, การเปิดโปรแกรม และ การเลือก เป็นต้น
2. การลาก คือ การแตะสไตลัสค้างไว้บนหน้าจอโดยไม่ยกสไตลัสขึ้น จากนั้นลากผ่านหน้าจอเพื่อทำการเลือกข้อความ ภาพ หรือการบังคับทิศทางในการเล่นเกมส์ เป็นต้น
3. การแตะค้างไว้ คือ การแตะสไตลัสค้างไว้บนหน้าจอที่ตำแหน่งเดิมโดยไม่ยกขึ้นและไม่เลื่อนไปยังตำแหน่งอื่นๆ เพื่อเปิดเมนูสำหรับการทำงานต่างๆ เช่น การแตะบนข้อความ เพื่อเปิดเมนูสำหรับการคัดลอก โดยจะมีการทำงานคล้ายๆกับการคลิกปุ่มขวาของเมาส์บน ระบบปฏิบัติการ Windows

รูปที่ 2.6 Styluses

2.6 โปรแกรม ActiveSync

Microsoft ActiveSync คือ โปรแกรมที่ทำงานบนพีคกเกิดพีซีและคอมพิวเตอร์พีซี โดยทำให้สามารถสร้างพาร์ตเนอร์ชิป (Partnership) เพื่อซิงโครไนซ์ข้อมูลผ่านสายเคเบิล เครเดิล อินฟราเรด โมเด็ม หรือการ์ดเครือข่าย ซึ่งจะทำให้ข้อมูลที่มีการสร้าง แก้ไข หรือเปลี่ยนแปลงระหว่างอุปกรณ์ทั้งสองได้รับการอัปเดตให้ตรงกันโดยอัตโนมัติ



รูปที่ 2.7 โปรแกรม Microsoft ActiveSync บน PC



รูปที่ 2.8 โปรแกรม Microsoft ActiveSync บน Pocket PC

ดังนั้นด้วย ActiveSync จึงทำให้สามารถ

- ซิงโครไนซ์ข้อมูลแบบอัตโนมัติหรือแบบกำหนดเองระหว่างโปรแกรมบนพ็อกเก็ตพีซีและพีซี

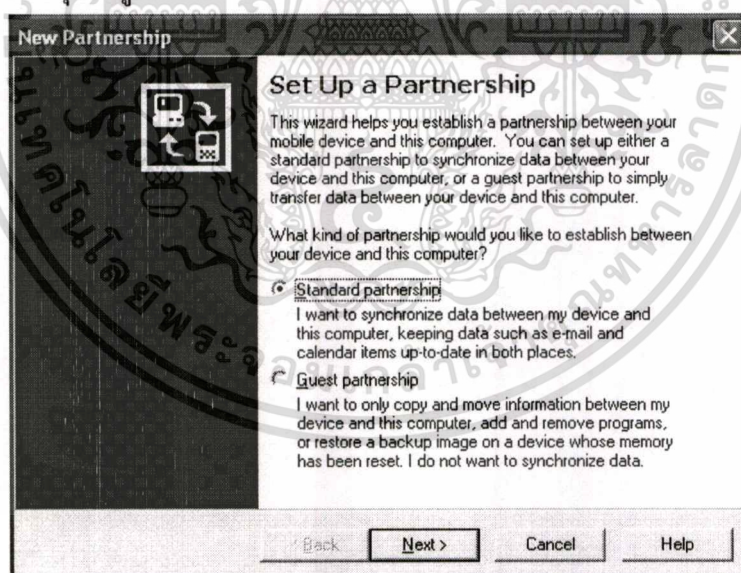
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ซิงโครไนซ์กับเซิร์ฟเวอร์
- การก๊อปปี้และย้ายไฟล์
- การติดตั้งโปรแกรม
- การจัดการและบริหารไฟล์ในพ็อกเก็ตพีซีจากเครื่องพีซี
- การแบ็กอัปและเรียกคืนข้อมูล

2.7 การสร้าง Partnership ของ ActiveSync

เมื่อเชื่อมต่อระหว่างพ็อกเก็ตพีซีและพีซีสำเร็จแล้ว ขั้นตอนต่อไปคือ การสร้าง Partnership สำหรับกำหนดรูปแบบการแลกเปลี่ยนข้อมูล โดยไดอะล็อกซ์บ็อกซ์สำหรับการตั้งค่าจะปรากฏขึ้น โดยมีตัวเลือก 2 ลักษณะคือ

- Standard partnership : เลือกออกแบบนี้ในกรณีที่ต้องการซิงโครไนซ์หรือแลกเปลี่ยนและปรับปรุงข้อมูลระหว่างพ็อกเก็ตพีซีกับเครื่องพีซี
- Guest partnership : เลือกออกแบบนี้ในกรณีที่ต้องการก๊อปปี้และย้ายข้อมูล รวมทั้งการติดตั้งและยกเลิกการติดตั้งโปรแกรม แต่ไม่ต้องการซิงโครไนซ์เพื่อแลกเปลี่ยนและปรับปรุงข้อมูล

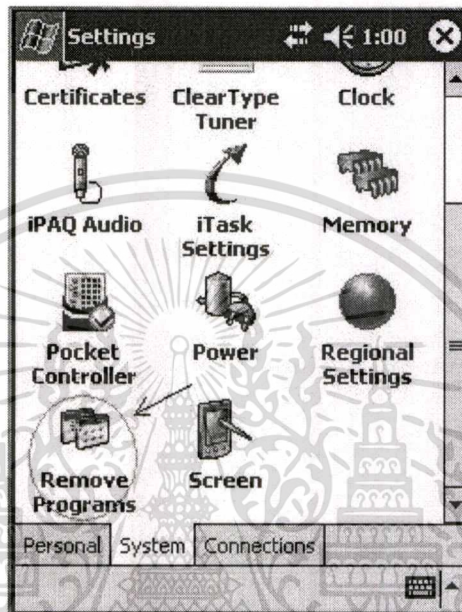


รูปที่ 2.9 การสร้าง Partnership ของ ActiveSync

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

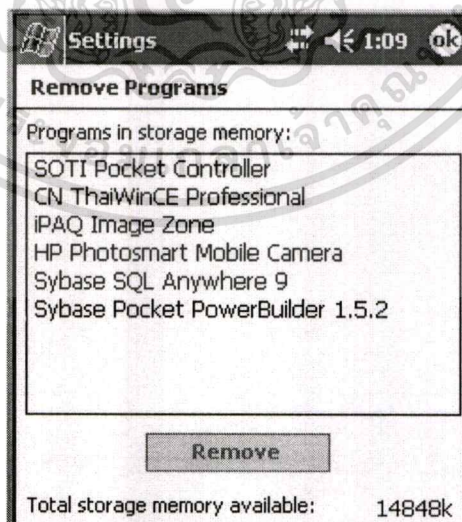
2.8 การยกเลิกการติดตั้งโปรแกรมใน Pocket PC

เมื่อติดตั้งโปรแกรมต่างๆลงในพ็อกเก็ตพีซี เพื่อใช้งานหรือทดลองใช้งาน หลังจากที่ไม่มีการใช้งานอีกต่อไป สามารถยกเลิกการติดตั้งโปรแกรมเหล่านั้น เพื่อให้ได้หน่วยความจำกลับคืนมา เปิดเมนู Start เลือกคำสั่ง Setting เลือกแท็บ System จากนั้น แตะไอคอน Remove Programs



รูปที่ 2.10 หน้าจอเพื่อยกเลิกการติดตั้ง

จากนั้นเลือกโปรแกรมที่ต้องการยกเลิกการติดตั้ง



รูปที่ 2.11 หน้าจอเพื่อเลือกโปรแกรมที่ต้องการยกเลิกการติดตั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9 เกี่ยวกับ Pocket PowerBuilder

Pocket PowerBuilder คือ เครื่องมือในการพัฒนาโปรแกรมประยุกต์สำหรับอุปกรณ์ที่ใช้ระบบปฏิบัติการ Windows CE ซึ่งได้จัดเตรียมคุณลักษณะต่างๆที่มีประสิทธิภาพไว้ให้แก่ผู้พัฒนา มากมาย เพื่อให้สามารถพัฒนาได้อย่างสะดวกรวดเร็ว และยังเป็น tool ที่ใช้งานง่ายสำหรับผู้พัฒนา Pocket PowerBuilder สามารถให้ผู้พัฒนาสร้างโปรแกรมประยุกต์แบบเชิงวัตถุ (object) และ กราฟฟิก จากนั้นเมื่อพัฒนาโปรแกรมประยุกต์เสร็จแล้ว ก็จะมีเครื่องมือในการกระจายโปรแกรมนี้ ไปสู่เครื่องฟ็อกเก็ตพีซี

ลักษณะเด่นของ Pocket PowerBuilder

- **4GL Power** มีสภาพแวดล้อมในการพัฒนาเป็นแบบวัตถุ (object) สนับสนุนการทำงานในเชิงวัตถุอย่างสมบูรณ์ ทั้งการ inheritance, polymorphism และ encapsulation ด้วยคุณสมบัติดังกล่าวทำให้สามารถสร้างโปรแกรมประยุกต์ได้อย่างมีประสิทธิภาพรวดเร็วทันการณ์สำหรับการตอบสนองงานทางธุรกิจ

Pocket PowerBuilder ได้จัดเตรียมfunction และ component ต่างๆจำนวนมาก ที่พร้อมให้เรียกใช้งานได้ทันที มีสภาพแวดล้อมการทำงานเป็นแบบ GUI จึงทำให้การพัฒนาโปรแกรมประยุกต์ที่มีความซับซ้อนเป็นไปอย่างรวดเร็วและง่ายดาย ไม่ว่าจะ inherit windows, user object, menu หรือ class ที่สร้างขึ้นมาก็สามารถทำได้อย่างรวดเร็ว โดยทำผ่าน GUI
- **Patented DataWindow Technology** เทคโนโลยี DataWindow ของ Sybase มีประสิทธิภาพมากในการเข้าถึงข้อมูลได้อย่างรวดเร็ว, การจัดการข้อมูล และการนำเสนอข้อมูลในรูปแบบที่ซับซ้อนบนอุปกรณ์ mobile โดยไม่ต้องทำการ code โปรแกรม ด้วยเทคโนโลยีของ DataWindow ทำให้สามารถสร้างคำสั่ง SQL, กำหนด rules, filter, sort, ควบคุมข้อมูลด้วยวิธีที่ง่ายผ่าน GUI และแสดงรูปแบบข้อมูลที่ซับซ้อน รวมถึงการนำเสนอข้อมูลแบบ free form, กราฟ, ตาราง หรือการจัดกลุ่มข้อมูล เป็นต้น

ถ้าหากผู้พัฒนาเคยสร้าง Datawindow ไว้ก่อนแล้วใน PowerBuilder ก็สามารถนำ DataWindow นั้นมาใช้ได้เลยใน Pocket PowerBuilder เพื่อเป็นการประหยัดเวลา
- **Workspace and Target** Workspace และ Target ได้จัดเตรียม ส่วนของการติดต่อกับผู้พัฒนาซึ่งเป็นส่วนสำคัญในการสร้างโปรแกรมประยุกต์ด้วย Pocket PowerBuilder ในส่วนที่เป็น workspace จะเป็นพื้นที่การทำงานแบบกราฟฟิกที่ใช้สร้างโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประยุกต์ ซึ่งอนุญาตให้สร้างโปรแกรมประยุกต์ได้หลายๆโปรแกรมพร้อมกัน สำหรับ target จะเป็นการรวม object ต่างๆที่เป็นส่วนประกอบของโปรแกรมประยุกต์ที่เราสร้างขึ้นเอาไว้ ซึ่งจะแสดงให้เห็นเป็นลักษณะกราฟฟิกบนหน้าจอการทำงาน

- **Wizard-driven development** Pocket PowerBuilder ได้จัดเตรียมเครื่องมือช่วยสร้างอัตโนมัติ (wizard) ไว้ให้หลายชนิด เช่น ใช้ในการสร้าง target ใหม่, สร้าง objects และ projects ใหม่, สร้างคำสั่ง SQL ที่ซับซ้อน ซึ่งทั้งหมดนี้สามารถทำได้อย่างง่ายดายผ่านทาง wizard
 - **Small-footprint, full-features, SQL database** Pocket PowerBuilder ทำงานเข้ากับฐานข้อมูล Sybase Adaptive Server Anywhere (ASA) ได้เป็นอย่างดี ซึ่ง ASA มี function รองรับการทำงานระดับองค์กร รวมถึงระบบการประมวลผลทรานแซกชันที่เปี่ยมประสิทธิภาพ, store procedure, triggers, row-level locking และการกู้คืนข้อมูลแบบอัตโนมัติ เป็นต้น
 - **Advanced data synchronization** Pocket PowerBuilder ได้รวมเทคโนโลยีในการ sync แบบ 2 ทิศทางของข้อมูลระหว่างระบบฐานข้อมูลหลักขององค์กร และฐานข้อมูลลูกข่ายที่อยู่บนอุปกรณ์ที่ใช้ระบบปฏิบัติการ Windows CE wizard เครื่องมือช่วยสร้างอัตโนมัติ (wizard)ที่ใช้สร้างการ sync ข้อมูล จะมีการใช้งานที่ง่ายและสะดวก โดยจะบอกขั้นตอนให้ปฏิบัติตาม
- เทคโนโลยีการ sync จะมีประสิทธิภาพสูงสุด ไม่ว่าจะการ sync นั้นจะเกิดขึ้นเป็นครั้งคราว หรือ เกิดขึ้นบ่อยชนิดเกือบเป็น real-time เทคโนโลยีเหล่านี้สนับสนุนทางด้านความปลอดภัยของการติดต่อสื่อสารข้อมูล

2.10 เกี่ยวกับ SQL Anywhere Studio

หลังจากติดตั้ง Pocket PowerBuilder แล้ว ก็จะได้ Sybase SQL Anywhere Studio ติดตั้งตามไปด้วย ซึ่งประกอบด้วย

- ระบบฐานข้อมูลเชิงสัมพันธ์ Adaptive Server Anywhere (ASA)
- Mobilink และ เทคโนโลยีการ sync ข้อมูล
- เครื่องมือในการดูแลและจัดการฐานข้อมูลที่เรียกว่า Sybase Central

2.11 แนะนำการใช้งาน Adaptive Server Anywhere

ระบบการจัดการฐานข้อมูลเชิงสัมพันธ์ ASA คือ แกนหลักของ SQL Anywhere Studio ฐานข้อมูล ASA ได้จัดเตรียมเครื่องมือต่าง ๆ สำหรับการเก็บและจัดการกับข้อมูล ซึ่งเครื่องมือเหล่านี้สามารถใช้ในการนำข้อมูลเข้าสู่ฐานข้อมูล, เปลี่ยนแปลงโครงสร้างฐานข้อมูล และทำการดูหรือแก้ไขข้อมูล

ฐานข้อมูล ASA ถูกออกแบบมาสำหรับงานที่ต้องการประสิทธิภาพของการใช้ SQL อย่างเต็มรูปแบบ และให้ทำงานได้ในหลากหลายสภาพแวดล้อม ฐานข้อมูล ASA ทำงานได้อย่างมีประสิทธิภาพไม่ว่าจะในสภาพแวดล้อมที่มีทรัพยากรของระบบอย่างเต็มที่ หรือในสภาพแวดล้อมที่มีข้อจำกัดเช่น ข้อจำกัดทางด้านกายภาพ เป็นต้น

เป้าหมายในการใช้ฐานข้อมูล ASA

ฐานข้อมูล ASA เหมาะสมที่จะใช้กับลักษณะงานดังต่อไปนี้

- **A workgroup database server** รองรับขนาดของกลุ่มการใช้งานตั้งแต่ผู้ใช้งานจำนวนน้อยจนถึงผู้ใช้งานจำนวนมาก สามารถใช้ฐานข้อมูล ASA เพื่อรองรับการทำงานแบบ multi-user ฐานข้อมูล ASA รองรับการทำงานแบบ multiple CPUs และรองรับหน่วยความจำได้สูงสุดถึง 64 GB
- **An embedded database** โปรแกรมประยุกต์จำนวนมากต้องการฐานข้อมูลที่ทำงานแบบเบื้องหลัง คือไม่ต้องมีการปรับแต่ง การตั้งค่า บำรุงรักษาระบบมากนัก, รวมถึงมีการจัดการข้อมูลส่วนบุคคล, ระบบจัดการเอกสาร ซึ่งต้องการเพียงแค่ให้จัดเก็บข้อมูลได้ สิ่งที่ดีกว่ามานี้ฐานข้อมูล ASA สามารถรองรับได้เป็นอย่างดี
- **Mobile computing** ปัจจุบันเครื่องคอมพิวเตอร์ notebook มีการใช้งานกันอย่างกว้างขวางในสถานที่ต่างๆ ซึ่งฐานข้อมูล ASA มีความเหมาะสมกับเครื่องคอมพิวเตอร์เหล่านี้ ด้วยเทคโนโลยี MobiLink และ SQL Remote replication จะสามารถทำให้เครื่องคอมพิวเตอร์ notebook ที่ทำงานเกี่ยวกับ transaction-based สามารถ sync ข้อมูลเหล่านั้นกลับมายังฐานข้อมูลหลักขององค์กรได้อย่างมีประสิทธิภาพ

มาตรฐานของฐานข้อมูล ASA

- **Full SQL relational database management system ASA** คือ ระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (RDBMS) ซึ่งมีความสามารถในการกู้คืนข้อมูลอย่างมีประสิทธิภาพ, การ

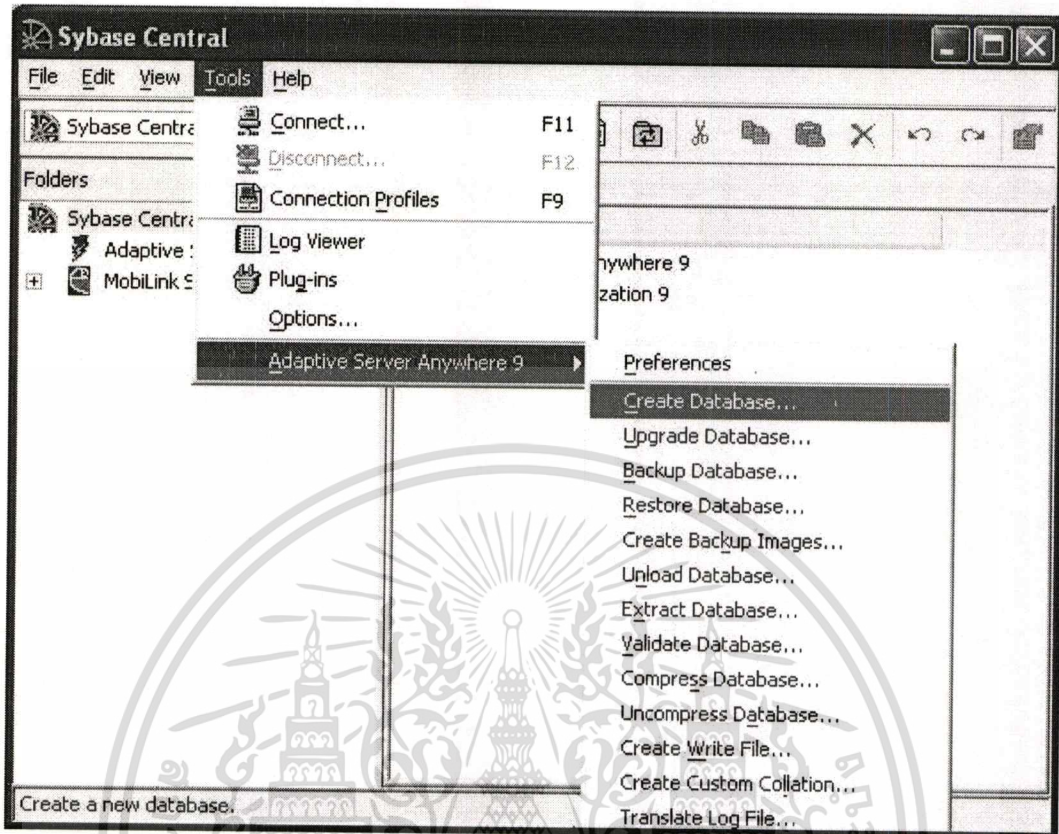
สำรองข้อมูลแบบ online, การสร้าง stored procedures, triggers, การควบคุมการทำงาน
พร้อมกันแบบ row-level, การประมวลผลคำสั่ง SQL อย่างมีประสิทธิภาพ

- **Economical hardware requirements** ASA ต้องการหน่วยความจำและพื้นที่ดิสก์เพียงเล็กน้อยเมื่อเทียบกับระบบจัดการฐานข้อมูลอื่นๆ
- **Easy to use** ASA สามารถปรับแต่งค่าระบบที่เหมาะสม (tuning) ได้อย่างอัตโนมัติ และมีความง่ายในการบำรุงรักษา
- **Standalone and network user** ASA สามารถใช้งานได้ทั้งแบบส่วนบุคคล(standalone) และแบบผู้ให้บริการในเครือข่าย (network server) ซึ่งต้องรองรับผู้ใช้งานจำนวนมาก หรือแม้แต่จะเป็นสถาปัตยกรรมแบบ three-tire ก็ตาม
- **High performance** ถึงแม้ว่า ASA จะถูกออกแบบมาให้สามารถใช้งานได้ง่ายและใช้ทรัพยากรของระบบน้อย แต่ก็ยังเป็นระบบจัดการฐานข้อมูลที่มีประสิทธิภาพสูง สามารถทำงานได้บนหลาย CPU และมี Query optimizer ที่มีประสิทธิภาพ
- **A cross-platform solution** ASA สามารถทำงานได้บนหลายระบบปฏิบัติการเช่น Windows, Novell NetWare, Sun Solaris และ Linux

2.12 การสร้างฐานข้อมูลจาก Adaptive Server Anywhere

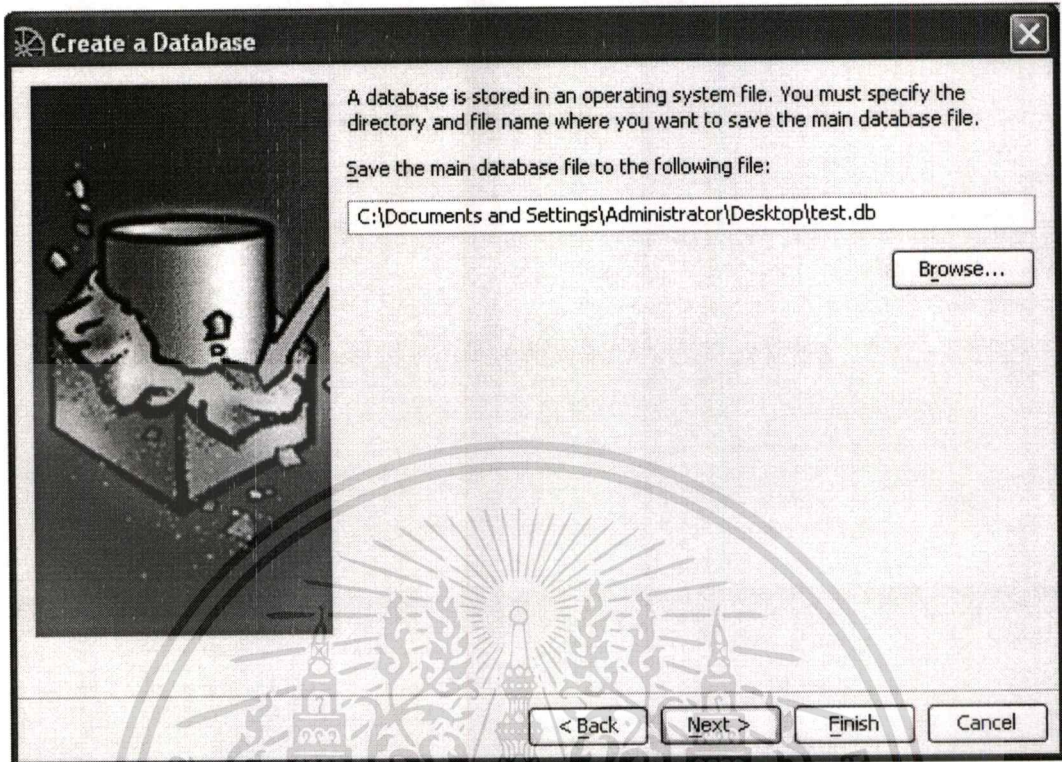
หลังจากติดตั้ง Adaptive Server Anywhere เสร็จแล้ว จะได้เครื่องมือที่ชื่อว่า Sybase Central เพื่อใช้ในการสร้างฐานข้อมูล

1. เปิดโปรแกรม Sybase Central
2. เลือกเมนู Tools > Adaptive Server Anywhere 9 เลือก Create Database



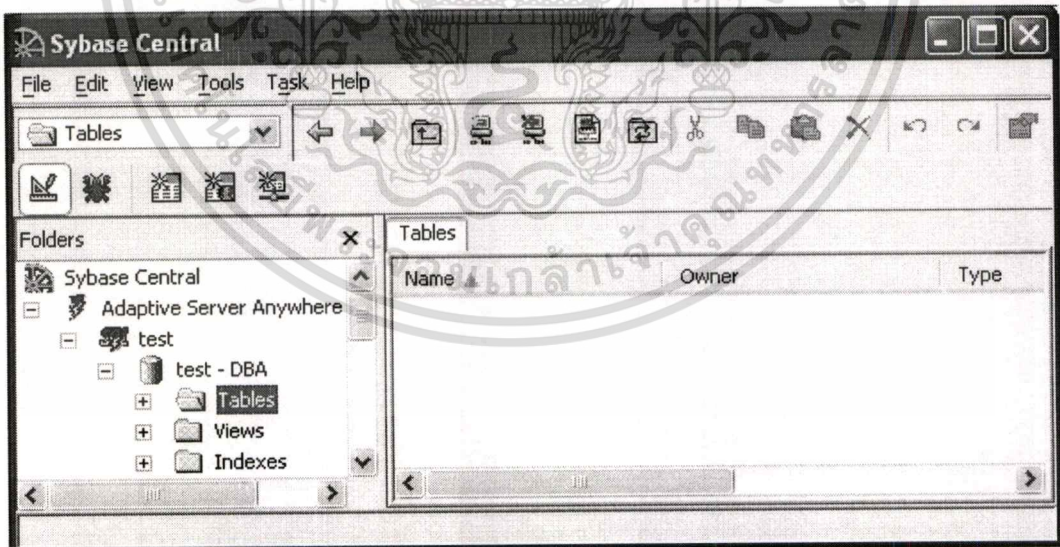
รูปที่ 2.12 หน้าจอเพื่อสร้างฐานข้อมูล

3. กดปุ่ม Next แล้วทำการเลือก directory และ ตั้งชื่อฐานข้อมูล
4. ทำตามขั้นตอนที่แจ้ง ไปเรื่อยๆ จนกระทั่ง ถึงหน้าจอสุดท้ายให้กดปุ่ม Finish



รูปที่ 2.13 หน้าจอสุดท้ายของการสร้างฐานข้อมูล

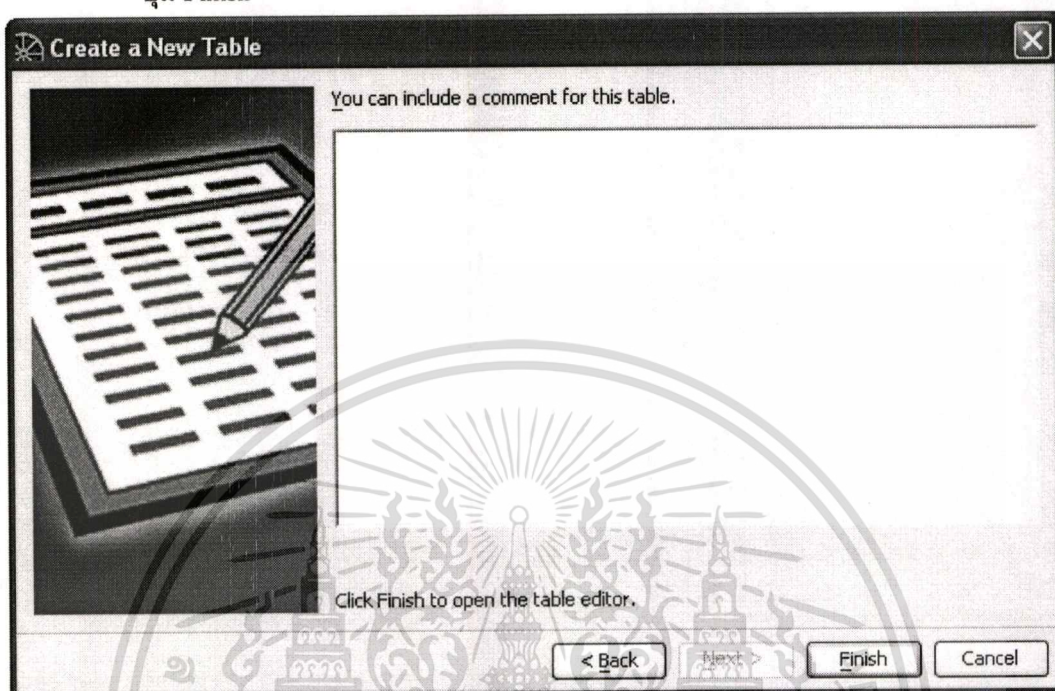
5. ทำการสร้างตารางโดย ไปเลือกที่ folder ที่ชื่อว่า Tables จากนั้นกดไอคอน 



รูปที่ 2.14 หน้าจอการสร้างตารางในฐานข้อมูล

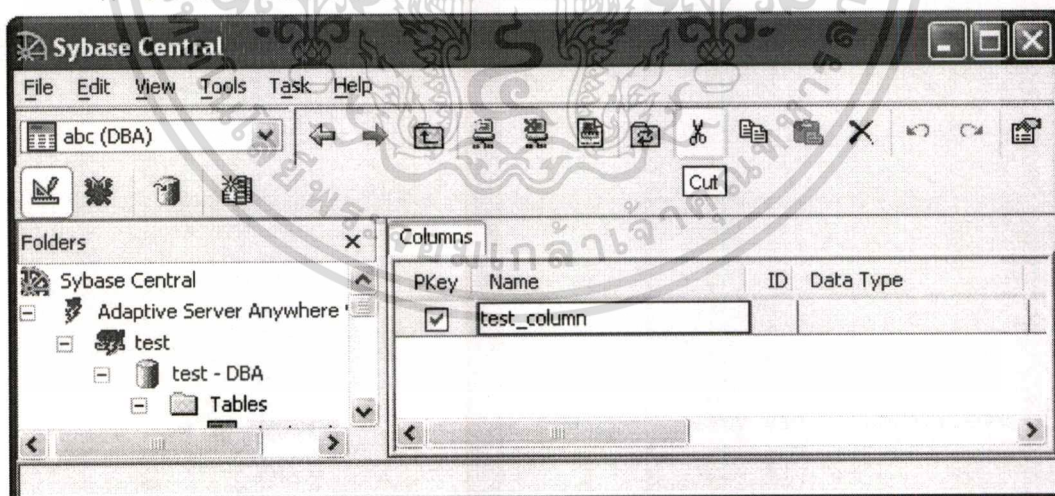
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. พิมพ์ชื่อตาราง แล้วทำตามขั้นตอนที่แจ้งไปเรื่อยๆ จนกระทั่ง ถึงหน้าจอสุดท้ายให้กดปุ่ม Finish




รูปที่ 2.15 หน้าจอสุดท้ายของการสร้างตาราง

7. จากนั้นให้ทำการเพิ่มคอลัมน์ที่ต้องการให้แก่ตาราง



รูปที่ 2.16 หน้าจอการสร้างคอลัมน์ของตาราง

8. เมื่อทำการเพิ่มคอลัมน์เสร็จเรียบร้อยแล้ว ให้คลิกไอคอน  เพื่อเป็นการบันทึกการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. ผลจากการสร้างฐานข้อมูลนี้ จะได้ไฟล์ของฐานข้อมูล ที่มีนามสกุลเป็น .log และ .db

2.13 แนะนำ MobiLink Synchronization

MobiLink คือ เทคโนโลยีการ synchronize ข้อมูล โดยเป็นการทำงานแบบสองทิศทาง คือ จะทำการ synchronize ข้อมูลระหว่าง ฐานข้อมูลหลัก ซึ่งเรียกว่า "Consolidate database" และ ฐานข้อมูลลูกข่าย ซึ่งเรียกว่า "Remote database"

Consolidate database สามารถเป็นฐานข้อมูลค่ายใดก็ได้แต่มีเงื่อนไขคือ ต้องติดต่อกับ MobiLink โดยผ่าน ODBC ที่สนับสนุนฐานข้อมูลนั้นๆ ส่วน Remote databases ต้องเป็นฐานข้อมูลของค่าย Sybase ที่ชื่อว่า Adaptive Server Anywhere หรือ UltraLite โดยการ synchronize ข้อมูลเริ่มจาก MobiLink remote site สร้างการติดต่อไปยัง MobiLink synchronization server ระหว่างที่กำลัง synchronize ข้อมูล MobiLink ลูกข่ายที่อยู่ ณ remote site จะ upload การเปลี่ยนแปลงของฐานข้อมูลลูกข่ายที่เกิดขึ้นหลังจากการ synchronize ครั้งล่าสุด ไปยัง MobiLink synchronization server แล้วไปทำการปรับปรุงข้อมูลที่ฐานข้อมูลหลัก จากนั้นจะส่งข้อมูลที่มีความสัมพันธ์กันกลับมายัง remote site

ลักษณะเด่นของ MobiLink

MobiLink synchronization server มีความยืดหยุ่นและความสามารถในการปรับตัว โดยสามารถใช้คำสั่งแบบ command-line เรียกใช้ dbmlsrv9 และ dbmlsync เพื่อปรับแต่งค่าต่างๆของระบบได้ดังต่อไปนี้

- **Data coordination** MobiLink อนุญาตให้สามารถ synchronize ข้อมูลเพียงบางส่วนได้ กระบวนการ synchronize ข้อมูล ถูกควบคุมโดยการเขียน script ในการเขียน script สามารถเขียนเป็น SQL, Java หรือ .NET ก็ได้ ทั้งนี้ขึ้นอยู่กับทักษะของผู้พัฒนาโปรแกรม
- **Automation** MobiLink มีความสามารถแบบอัตโนมัติหลายอย่างเช่น MobiLink synchronization server สามารถสร้าง script ที่เหมาะสมขึ้นเองเพื่อใช้ในการ synchronize เป็นต้น
- **Monitoring and reporting** MobiLink มีสองกลไกในการเฝ้าตรวจสอบการ synchronize คือ MobiLink monitor และ statistical script ซึ่งจะช่วยให้ทราบถึง monitor scripts, schema content, row-count values, script names, translated script contents และ row values

- **Performance tuning** MobiLink มีกลไกในการปรับแต่งค่า (tuning) เพื่อให้เกิดประสิทธิภาพที่ดีที่สุด เช่น ขนาดของ upload cache, จำนวน contention, จำนวนการติดต่อกับฐานข้อมูล, จำนวน worker threads, logging verbosity หรือขนาดของ BLOB cache

ลักษณะเด่นในการ synchronize ข้อมูลของ MobiLink

- **Choice of communication stream** การ synchronize สามารถทำอยู่บน protocol TCP/IP, HTTP หรือ HTTPS ในเครื่อง Palm สามารถ synchronize ผ่านโปรแกรม HotSync ถ้าเป็นเครื่องที่ใช้ระบบปฏิบัติการ Windows CE สามารถ synchronize ผ่านโปรแกรม ActiveSync
- **Two-way synchronization** การเปลี่ยนแปลงที่จะเกิดขึ้นกับฐานข้อมูลสามารถเกิดได้ทุกที่ สามารถที่จะเลือกให้การ synchronize ข้อมูลในแต่ละครั้งเป็นแบบ upload อย่างเดียว, download อย่างเดียว หรือจะทำทั้งสองอย่างพร้อมกันก็ได้
- **File-based synchronization** การ download ข้อมูล สามารถกระจายออกมาให้อยู่ในรูปแบบ files ได้ ซึ่งทำเพียงครั้งเดียวก็สามารถนำ files ที่ได้ไปกระจายสู่เครื่องลูกข่ายได้ทุกเครื่องที่ต้องการ
- **Server-initiated synchronization** สามารถเริ่มการ synchronize ข้อมูลได้จากที่ฐานข้อมูลหลัก (consolidate database) นั่นก็คือสามารถส่งข้อมูลที่มีการเปลี่ยนแปลงที่ฐานข้อมูลหลัก ไปยังฐานข้อมูลลูกข่ายได้
- **Remote-initiated** การ synchronize ข้อมูลระหว่าง ฐานข้อมูลลูกข่ายกับฐานข้อมูลหลัก สามารถเริ่มต้นขึ้นจากทางฝั่ง ฐานข้อมูลลูกข่าย
- **Session-based** ในการ synchronize ข้อมูล ไม่ว่าจะเป็นการ upload หรือ download สามารถทำได้ภายในทรานแซกชันเดียว เมื่อการ synchronize ข้อมูลในครั้งนั้นๆเสร็จสิ้นลงแล้ว ข้อมูลในฐานข้อมูลหลักและฐานข้อมูลลูกข่ายก็จะสอดคล้องกัน
- **Transactional integrity** ไม่ว่าในการ synchronize ข้อมูลครั้งนั้นๆจะทำสำเร็จทั้งหมดหรือไม่ ก็สามารถมั่นใจได้ว่า ฐานข้อมูลทั้งสองฝั่งจะยังคงความสมบูรณ์อยู่
- **Wide variety of hardware and software platforms** ฐานข้อมูลหลักที่จะใช้ติดต่อกับ MobiLink server สามารถใช้เป็นฐานข้อมูลได้หลายชนิด เช่น ASA, Adaptive Server Enterprise, Oracle, IBM DB2 หรือ Microsoft SQL Server ฐานข้อมูลลูกข่ายสามารถใช้เป็น ASA หรือ Ultralite

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MobiLink synchronization server สามารถทำงานอยู่บนระบบปฏิบัติการ Windows หรือ UNIX ฐานข้อมูล ASA สามารถทำงานอยู่บนระบบปฏิบัติการ Windows, Windows CE หรือ UNIX ส่วนฐานข้อมูล Ultralite สามารถทำงานอยู่บนเครื่อง Palm, Windows CE หรือ VxWorks เป็นต้น

- **Flexibility** MobiLink synchronization server ใช้ SQL, Java หรือ .Net เขียนเป็น script เพื่อใช้ในการควบคุมการ download และ upload ข้อมูล
- **Scalability and performance** การ synchronize ของ MobiLink มีประสิทธิภาพโดย server 1 ตัวสามารถรองรับการ synchronize ได้เป็นพันๆ process พร้อมกัน และสามารถใช้ MobiLink server หลายตัวทำงานพร้อมกัน เพื่อทำ load balancing ได้
- **Easy to get started** สามารถติดตั้งและปรับแต่งค่าของระบบในแบบพื้นฐานได้อย่างรวดเร็ว จากนั้นจึงค่อยปรับแต่งส่วนอื่นๆที่จำเป็นเพิ่มเติมเข้ามาเพื่อให้เหมาะสมกับการทำงาน

2.14 เกี่ยวกับ MobiLink synchronization

Data movement and synchronization

การเคลื่อนย้ายข้อมูลจะเกิดขึ้นเมื่อ ข้อมูลที่ต้องใช้ร่วมกันถูกกระจายไปยังฐานข้อมูลย่อยๆ ที่อยู่บนเครื่องปลายทางต่างๆ และการเปลี่ยนแปลงที่เกิดขึ้นกับฐานข้อมูลหนึ่งต้องเกิดการไปปรับปรุงที่ฐานข้อมูลอื่นๆด้วย เพื่อให้ข้อมูลมีความสอดคล้องกัน โดยสามารถเคลื่อนย้ายข้อมูลด้วยวิธี replication หรือ synchronization

การทำ replication จะเป็นการเคลื่อนย้ายทรานแซกชันทั้งหมดจากฐานข้อมูลหนึ่งไปยังอีกฐานข้อมูลหนึ่ง แต่การ synchronization จะเคลื่อนย้ายเพียงทรานแซกชันผลลัพธ์ ทั้งสองเทคนิคนี้เอาข้อมูลมาจากการตรวจสอบ ไฟล์ทรานแซกชัน log แต่การ synchronization จะใช้เพียงบางส่วนของไฟล์ log ไม่ได้ใช้ไฟล์ log ทั้งหมด จึงทำให้การเคลื่อนย้ายข้อมูลมีความรวดเร็วและมีประสิทธิภาพมากกว่า

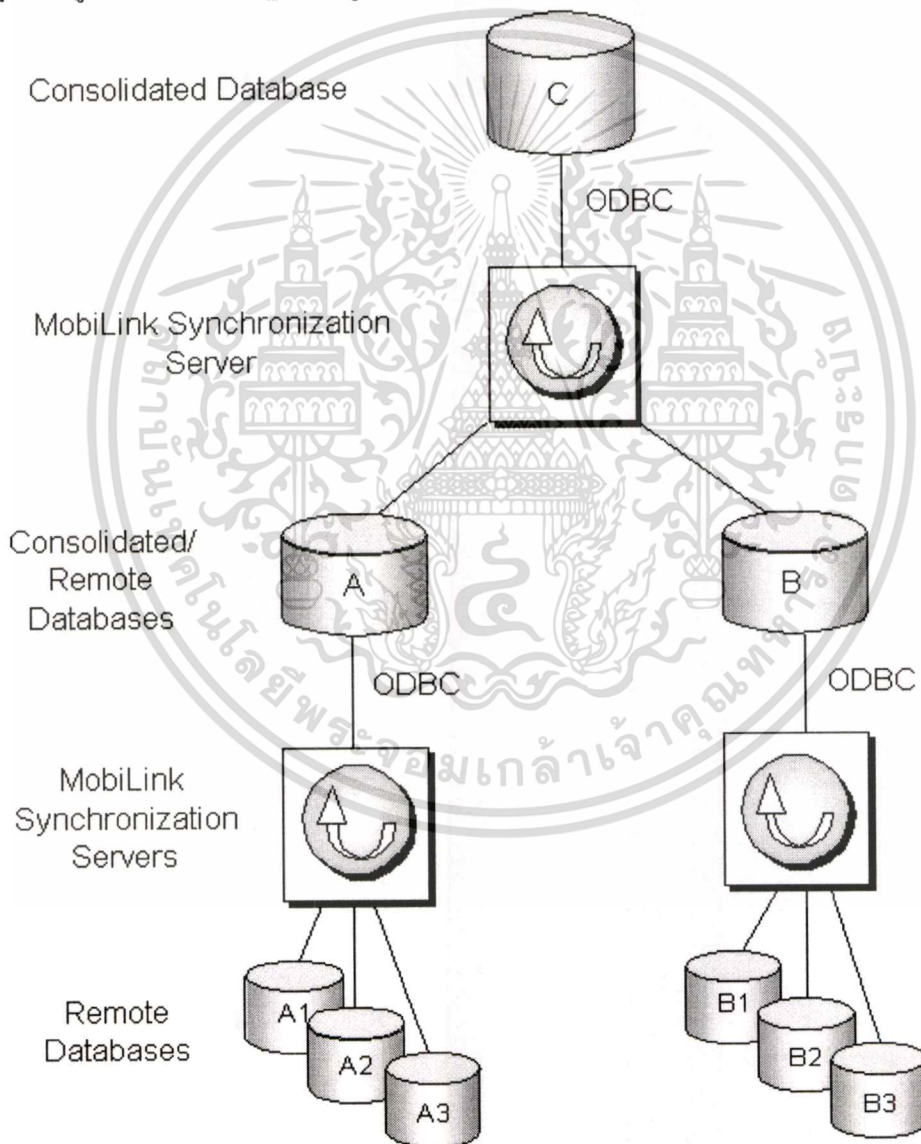
The MobiLink synchronization server

ใช้ dbmlsrv9 จัดการกระบวนการ synchronize และเป็นตัวกลางในการติดต่อระหว่างฐานข้อมูลหลักและฐานข้อมูลลูกข่าย การติดต่อสื่อสารทั้งหมดระหว่าง MobiLink synchronization server และ ฐานข้อมูลหลัก จะติดต่อผ่าน ODBC โดย synchronization server และ ฐานข้อมูลหลัก จะอยู่บนคอมพิวเตอร์เครื่องเดียวกันหรือไม่ก็ได้

MobiLink hierarchy

โครงสร้างของ MobiLink มีลักษณะเป็นระบบตามลำดับชั้น (hierarchy) โดยที่แต่ละ node อาจอยู่บน server, PC หรืออุปกรณ์ mobile ต่างๆ โครงสร้างพื้นฐานของ MobiLink ประกอบด้วย ฐานข้อมูลหลักและ ฐานข้อมูลย่อยที่อยู่บนอุปกรณ์ mobile สำหรับโครงสร้างที่ซับซ้อนขึ้นคือ ฐานข้อมูลที่อยู่บน node ใดๆ อาจเป็นทั้งฐานข้อมูลหลักและฐานข้อมูลย่อย

ตัวอย่างเช่น ฐานข้อมูลย่อย A1, A2 และ A3 รับ-ส่งข้อมูลกับฐานข้อมูล A และฐานข้อมูลย่อย B1, B2 และ B3 รับ-ส่งข้อมูลกับฐานข้อมูล B จากนั้นฐานข้อมูล A และ B ส่งต่อข้อมูลให้กับฐานข้อมูล C ซึ่งในกรณีนี้ฐานข้อมูล A และ B ต้องเป็น ASA เท่านั้น



รูปที่ 2.17 โครงสร้างของ MobiLink

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Synchronization script

เมื่อ MobiLink ลูกข่าย เริ่มการรับ-ส่งข้อมูล จะเกิดเหตุการณ์ (event) ต่างๆบนMobiLink server MobiLink จะค้นหา script ของ event นั้นๆ เพราะฉะนั้นถ้าต้องการให้ MobiLink server ทำงานในลักษณะต่างๆ ก็ต้องเขียน script ขึ้นมาสำหรับ event ที่ต้องการ ดังนั้นจึงเรียกได้ว่าการรับส่งข้อมูลผ่าน MobiLink เป็นกระบวนการแบบ event-driven การเขียน script สามารถเขียนได้ตั้งแต่ระดับเริ่มการติดต่อ (connection-level) หรือระดับ event สำหรับแต่ละ table ในฐานข้อมูลย่อย โดย script เหล่านี้ถูกจัดเก็บไว้ในฐานข้อมูลหลัก

Publications, articles, users and subscriptions

Publication คือ database object ที่อยู่บนฐานข้อมูลลูกข่าย ซึ่งจะระบุ table และ column ในการรับ-ส่งข้อมูล โดยใน publication จะประกอบด้วย articles ซึ่งเป็น database object ใช้ในการเป็นตัวแทนของ table ทั้งหมด หรือ ส่วนย่อยของ columns และ rows ใน table

User คือ database object ที่อยู่บนฐานข้อมูลลูกข่าย ใช้ในการระบุผู้รับ-ส่งข้อมูล ซึ่งจะมี 1 user ต่อ 1 ฐานข้อมูลย่อย

Subscription ใช้ระบุ protocol ในการ synchronization (TCP/IP, HTTP หรือ HTTPS เป็นต้น), address (เช่น myserver.sybase.com) และทางเลือกอื่นๆเพิ่มเติมในการติดต่อ

The synchronization process

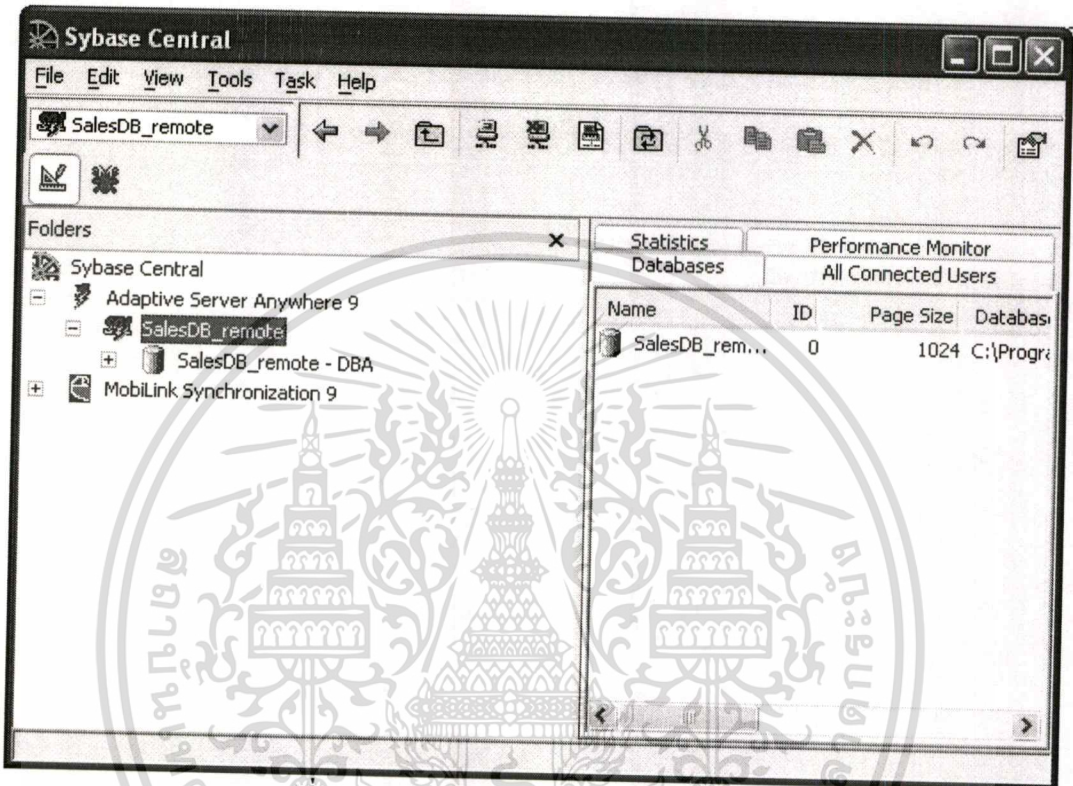
Dbmlsync ติดต่อไปยังฐานข้อมูลลูกข่ายโดยใช้ TCP/IP, HTTP หรือ HTTPS และเตรียม stream ของข้อมูล (upload stream) เพื่อใช้ upload ไปยังฐานข้อมูลหลัก Dbmlsync ใช้ข้อมูลที่บรรจุอยู่ใน ทราานแซกชัน log ของฐานข้อมูลลูกข่ายเพื่อสร้างเป็น upload stream ซึ่ง upload stream เก็บ MobiLink user name และ password, version ของ script ที่ใช้ในการ synchronize ข้อมูล, เวลาครั้งล่าสุดที่ทำการ synchronize, โครงร่างของ table และ column ที่อยู่ใน publication และผลลัพธ์ของ ทราานแซกชันนับตั้งแต่ครั้งล่าสุดที่ทำการ synchronize

หลังจากสร้าง upload stream แล้ว Dbmlsync จะใช้ข้อมูลที่เก็บอยู่ใน publication ที่ระบุไว้ และ subscription เพื่อติดต่อไปยัง MobiLink synchronization server และทำการแลกเปลี่ยนข้อมูล

เมื่อ MobiLink synchronization server ได้รับข้อมูล ก็จะไปปรับปรุงฐานข้อมูลหลัก จากนั้นจะสร้าง download stream ซึ่งจะบรรจุการเปลี่ยนแปลงข้อมูลที่มีความสอดคล้องกัน แล้วส่งกลับไปยังฐานข้อมูลลูกข่าย

2.15 การ Set ค่าต่างๆ ในฐานข้อมูลลูกข่าย (Remote Database) เพื่อใช้ในการ synchronization

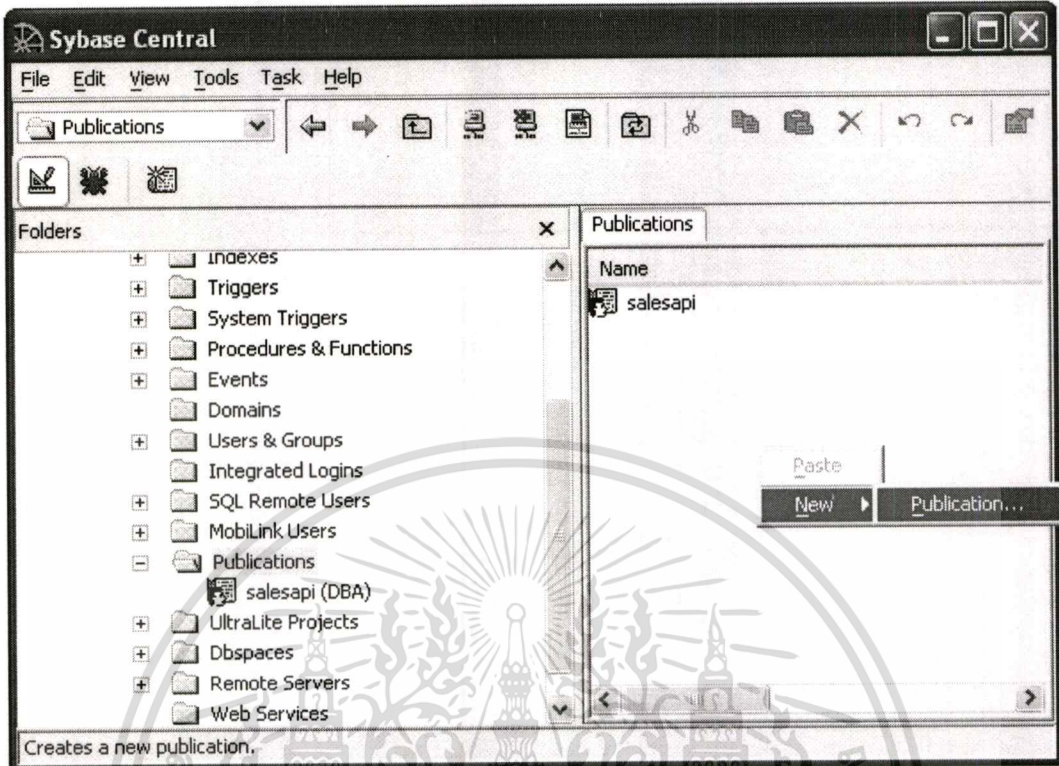
1. เปิดโปรแกรม Sybase Central
2. สร้างการติดต่อไปยังฐานข้อมูลลูกข่าย



รูปที่ 2.18 หน้าจอการติดต่อเข้าสู่ฐานข้อมูลลูกข่าย

3. ดับเบิลคลิกที่ Folder "Publications"
4. คลิกขวา → New → Publications

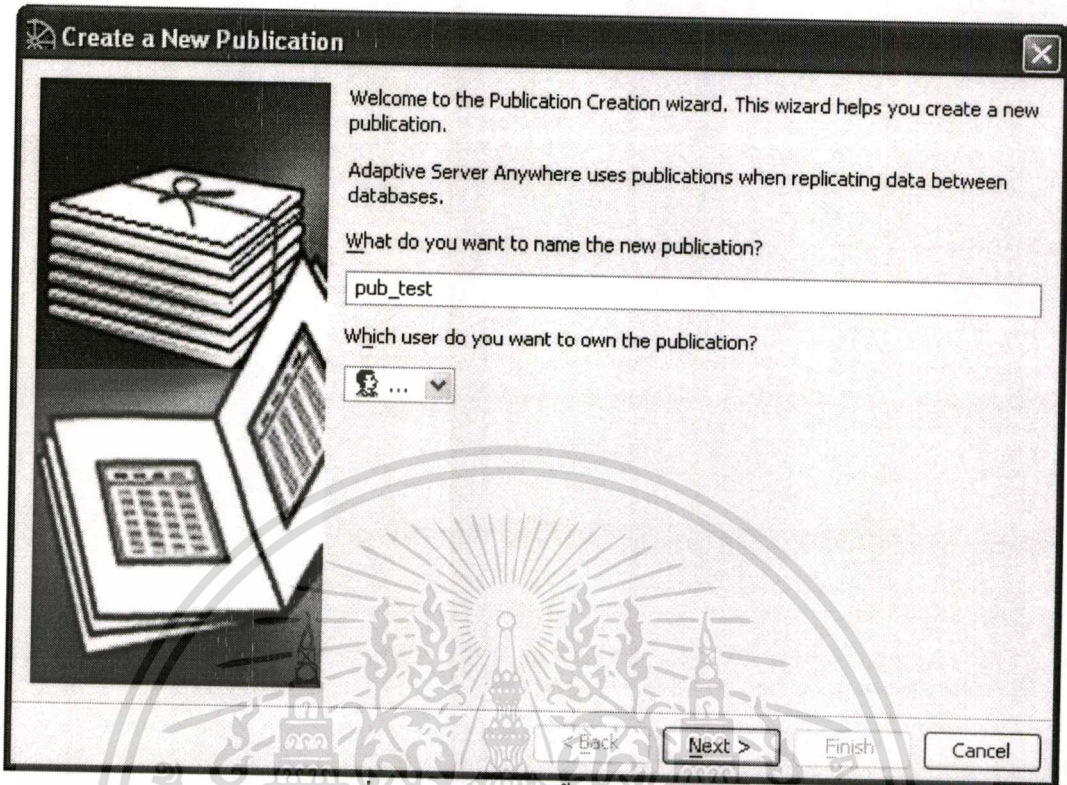
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.19 หน้าจอการสร้าง Publication

5. พิมพ์ชื่อ Publication จากนั้นกด Next

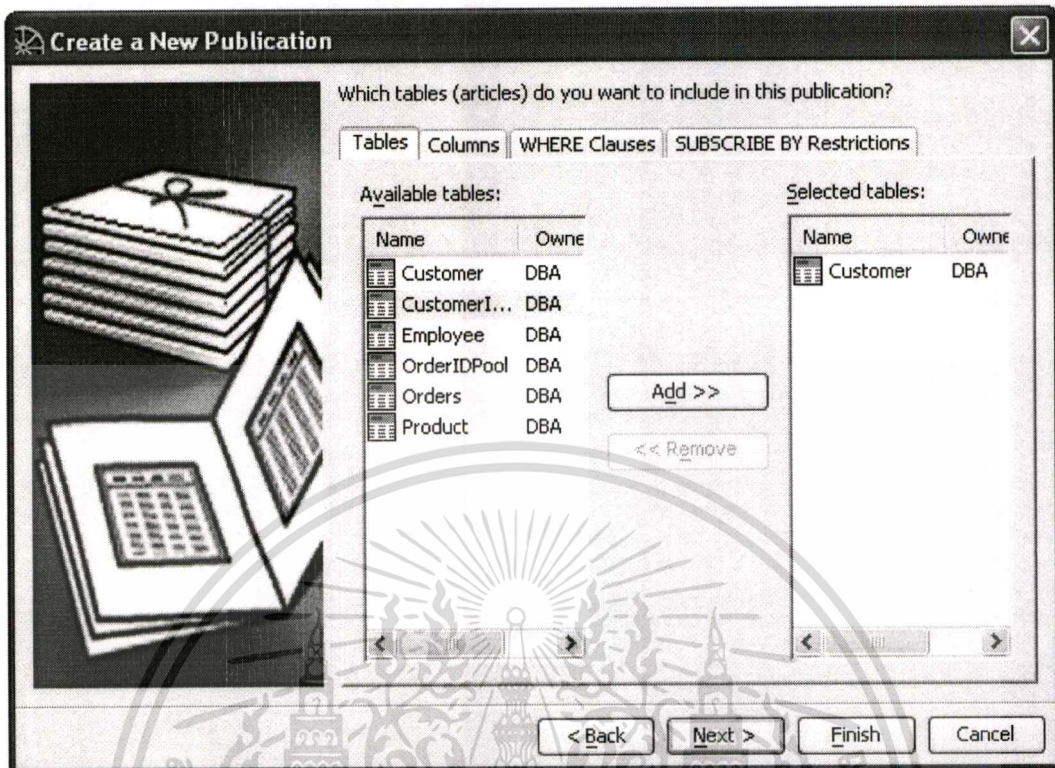
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.20 หน้าจอการตั้งชื่อ Publication

6. เลือก table ที่ต้องการใช้ในการ synchronization

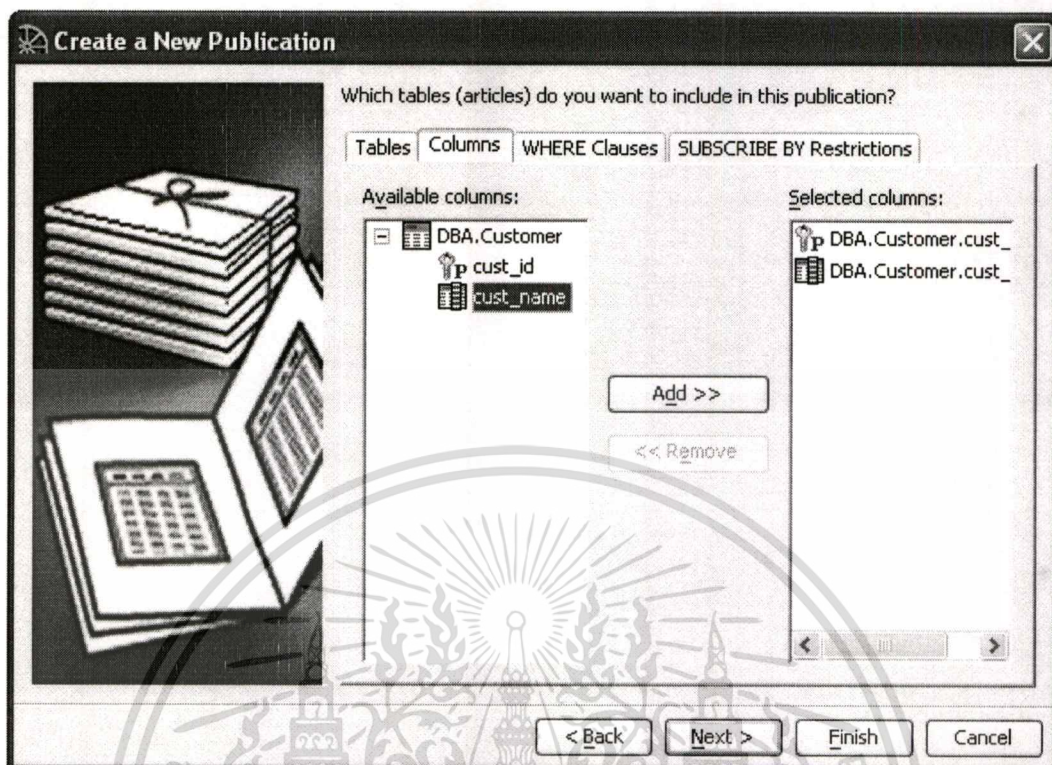
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.21 หน้าจอการเลือก Table สำหรับ Publication

- เลือก column ที่ต้องการใช้ในการ synchronization

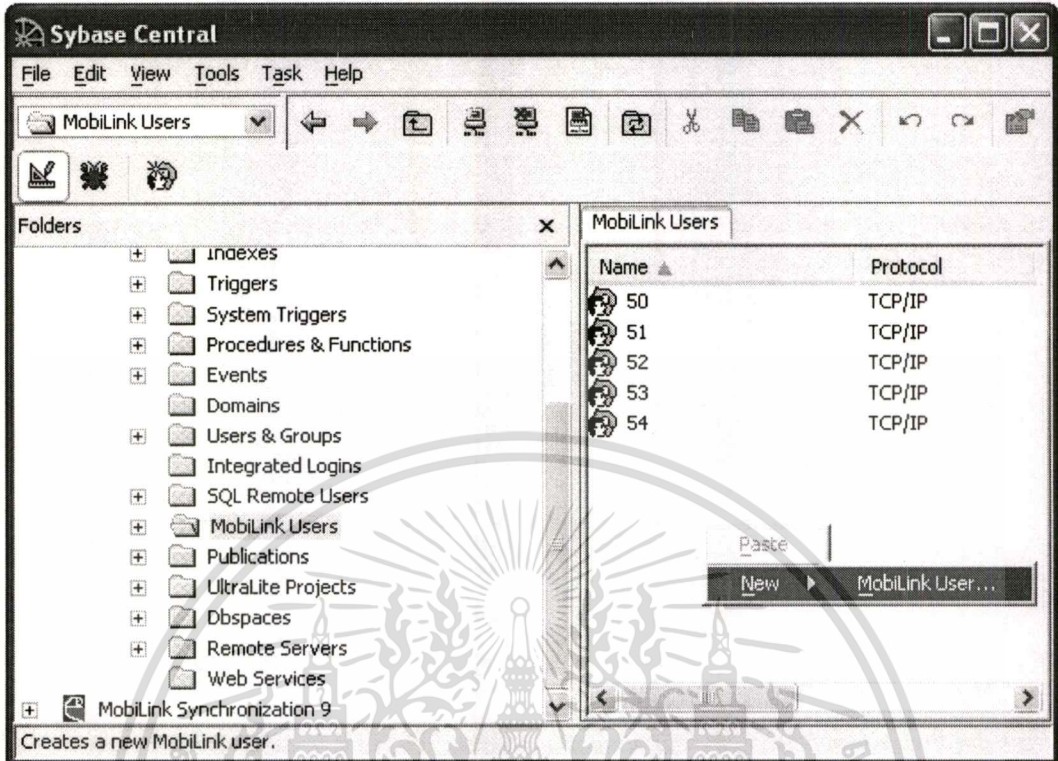
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.22 หน้าจอการเลือก Column สำหรับ Publication

8. กด Finish เพื่อเป็นการเสร็จสิ้นการสร้าง Publication
9. ดับเบิลคลิกที่ Folder “MobiLink Users”
10. คลิกขวา → New → MobiLink Users

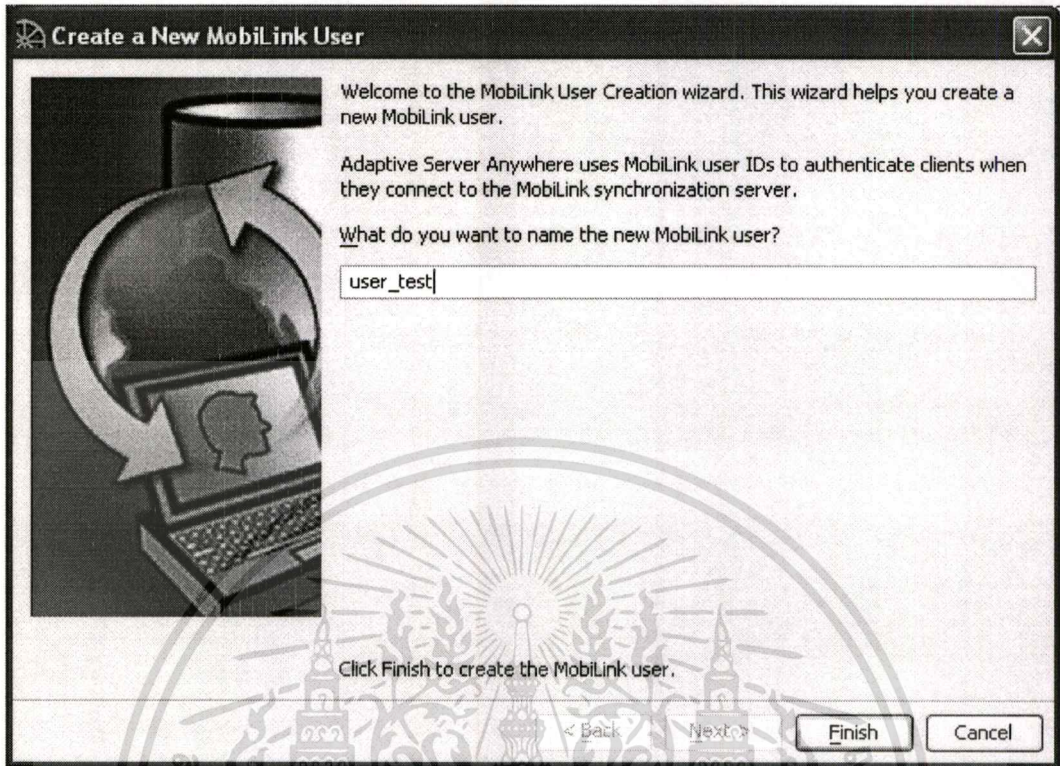
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.23 หน้าจอการสร้าง MobiLink User

11. ตั้งชื่อสำหรับใช้เป็น MobiLink User แล้วกด Finish.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.24 หน้าจอการตั้งชื่อ MobiLink User

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบระบบ

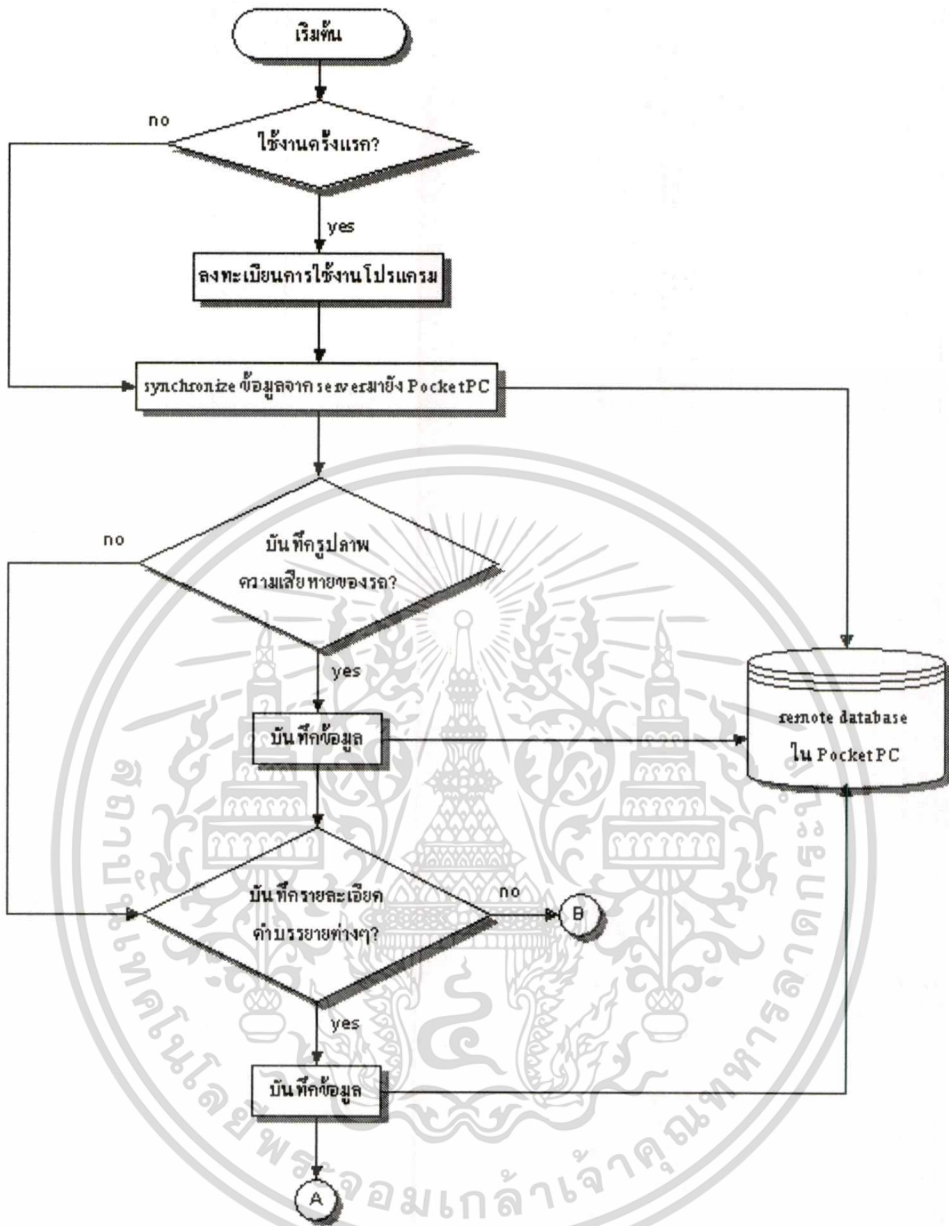
3.1 ขอบเขตของระบบ

โปรแกรมประยุกต์ที่พัฒนาขึ้นสามารถทำงานหลักๆดังนี้

1. สามารถ Synchronized ข้อมูลไป-กลับ ระหว่าง Pocket PC กับ Server ไม่ว่าจะ เป็น ข้อมูลชนิด text หรือ รูปภาพ
2. สามารถ บันทึกหรือแก้ไขข้อมูลต่างๆลงสู่ฐานข้อมูลได้
3. สามารถขีดเขียนลายเส้นบนรูป ที่ถ่ายจากที่เกิดเหตุได้
4. สามารถบันทึกการตั้งค่าต่างๆของระบบลงสู่ register ของเครื่อง Pocket PC ได้ เช่น หมายเลข IP Address ของเครื่องปลายทางที่ต้องการ synchronized ข้อมูล

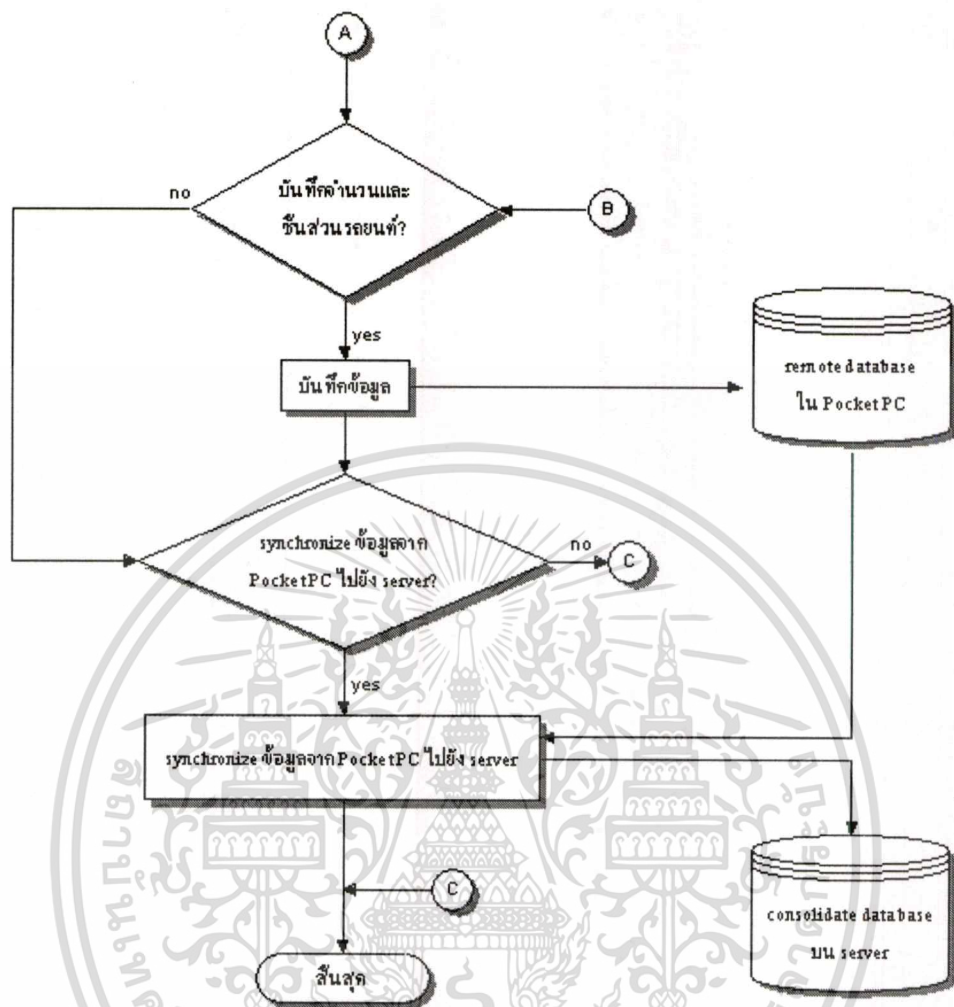
3.2 ขั้นตอนการทำงานของโปรแกรม

1. ผู้ใช้งาน โปรแกรม ลงทะเบียนการใช้งาน โปรแกรม
2. ผู้ใช้งาน synchronize ข้อมูลจาก server (consolidate database) มายังเครื่อง Pocket PC (remote database) ซึ่งเป็นข้อมูลการแจ้งอุบัติเหตุของลูกค้า
3. ผู้ใช้งาน โปรแกรมดำเนินการเก็บข้อมูลต่างๆลงสู่ฐานข้อมูลในเครื่อง Pocket PC (remote database) ดังนี้
 - รูปภาพความเสียหายของรถ
 - รายละเอียดคำบรรยายต่างๆ
 - จำนวนและชิ้นส่วนของอุปกรณ์รถยนต์ที่ประเมินว่าต้องใช้ในการซ่อมรถ
4. ผู้ใช้งาน synchronize ข้อมูลจาก เครื่อง Pocket PC (remote database) ไปยัง server (consolidate database)



รูปที่ 3.1 Flowchart แสดงขั้นตอนการทำงานของโปรแกรม ส่วนที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 Flowchart แสดงขั้นตอนการทำงานของโปรแกรม ส่วนที่2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การออกแบบหน้าจอการทำงาน (User Interface Design)

The image shows a screenshot of a software window titled "ลงทะเบียนการใช้งาน" (Registration for Use). The window contains the SYBASE logo and the text "Sybase Solution Corporation (Thailand)". Below the logo, there is a text input field labeled "user name:" and a button labeled "ลงทะเบียน" (Register).

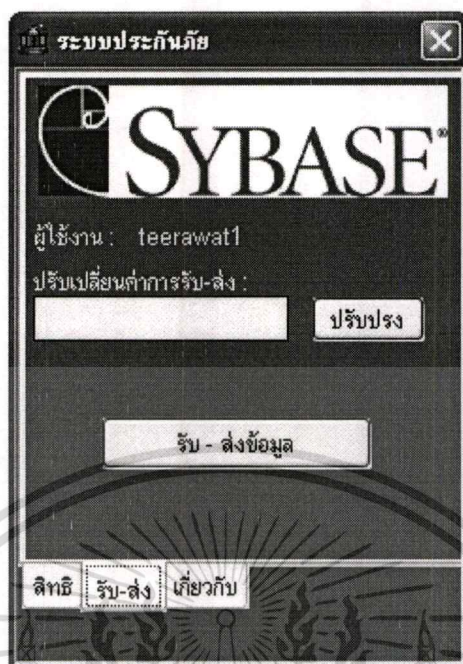
รูปที่ 3.3 หน้าจอลงทะเบียนการใช้งาน โปรแกรม

จากรูปที่ 3.3 เป็นหน้าจอไว้สำหรับใส่ user name ซึ่ง user name นี้จะเป็นอันเดียวกันกับ MobiLink user name หน้าจอนี้จะเปิดขึ้นมาในครั้งแรกของการใช้งานโปรแกรมเท่านั้น เมื่อมีการลงทะเบียนเรียบร้อยแล้ว ในการใช้งานโปรแกรมครั้งต่อไป จะไม่มีหน้าจอนี้ขึ้นมาอีกแล้ว [ลงทะเบียน] ทำการบันทึก user name ลงไปในฐานข้อมูล



รูปที่ 3.4 หน้าจอแสดงรายการลูกค้าที่เคลม

จากรูปที่ 3.4 เป็นหน้าจอแสดงรายการของผู้ที่เกิดอุบัติเหตุ ซึ่งเจ้าหน้าที่ต้องออกไปพบ ณ
 ที่เกิดเหตุ
 [แสดงรายละเอียด] แสดงรายละเอียดต่างๆของผู้ที่เกิดอุบัติเหตุ
 [ส่งมอบ] บิดงานของผู้ที่เกิดอุบัติเหตุรายนี้ ทำเมื่อได้ดำเนินงานทุกอย่างเรียบร้อยแล้ว

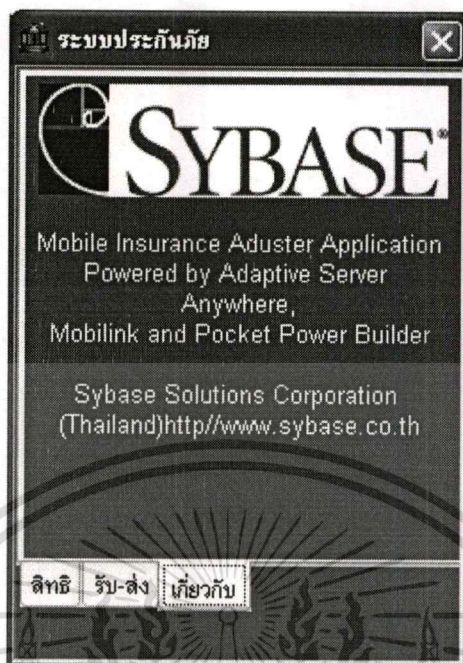


รูปที่ 3.5 หน้าจอสำหรับการ synchronization

จากรูปที่ 3.5 เป็นหน้าจอสำหรับการ synchronization โดยจะทำการรับ-ส่งข้อมูลระหว่างฐานข้อมูลหลักที่ server กับ ฐานข้อมูลลูกข่ายบนเครื่อง Pocket PC

[ปรับปรุง] ทำการบันทึกที่อยู่ของเครื่องปลายทาง เช่น 192.168.170.1 ลงสู่ registry ของเครื่อง Pocket PC

[รับ-ส่งข้อมูล] ทำการ synchronization ระหว่างฐานข้อมูลหลักกับฐานข้อมูลลูกข่าย



รูปที่ 3.6 หน้าจอแสดง credit ของ โปรแกรม

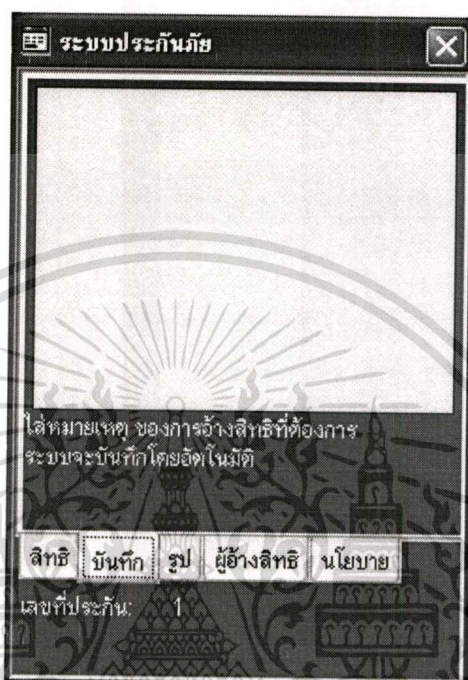
จากรูปที่ 3.6 เป็นหน้าจอแสดงรายละเอียดเกี่ยวกับตัว โปรแกรม



รูปที่ 3.7 หน้าจอบันทึกและแสดงรายการ ชิ้นส่วนที่ลูกค้าต้องการเคลม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.7 เป็นหน้าจอสำหรับบันทึกชิ้นส่วนต่างๆที่คาดว่าจะต้องใช้ในการซ่อมรถยนต์ เมื่อเลือกชิ้นส่วนและจำนวนเรียบร้อยแล้ว ก็จะแสดงรายละเอียดที่ได้เลือกไปพร้อมราคารวมออกมาที่หน้าจอ



รูปที่ 3.8 หน้าจอบันทึกรายละเอียดเพิ่มเติมเกี่ยวกับการเคลมประกัน

จากรูปที่ 3.8 เป็นหน้าจอสำหรับบันทึกรายละเอียดเพิ่มเติมเกี่ยวกับการเคลมประกัน โดยเมื่อออกจากหน้านี้ระบบจะบันทึกข้อมูลลงฐานข้อมูลให้อัตโนมัติ

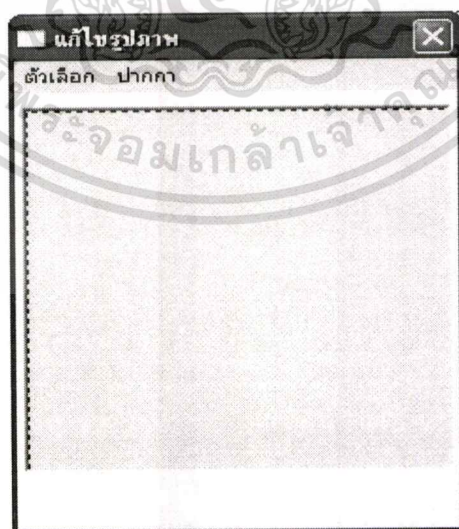


รูปที่ 3.9 หน้าจอสำหรับทำงานเกี่ยวกับรูปภาพที่ถ่ายจากที่เกิดเหตุ

จากรูปที่ 3.9 เป็นหน้าจอสำหรับทำงานเกี่ยวกับรูปภาพที่ถ่ายจากที่เกิดเหตุ โดยจะแสดงรายการของภาพที่ได้บันทึกเอาไว้บนหน้าจอ

[แก้ไขรูปภาพ] ไปสู่หน้าจอที่ใช้แก้ไขรูปภาพเดิมที่เคยบันทึกไว้แล้ว

[เพิ่มรูปภาพ] ไปสู่หน้าจอเพื่อเลือกรูปภาพที่ต้องการบันทึกลงฐานข้อมูล



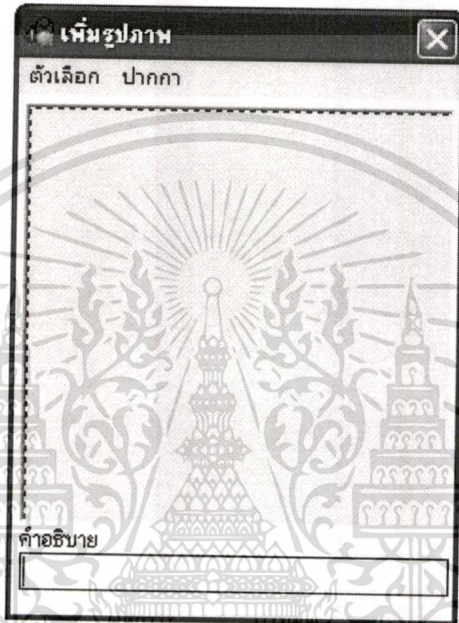
รูปที่ 3.10 หน้าจอสำหรับแก้ไขรูปภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.10 เป็นหน้าจอสำหรับแก้ไขรูปภาพเดิมที่เคยบันทึกไว้ก่อนหน้า อาจเพื่อเป็นการแก้ไขลายเส้นที่ได้เขียนไว้บนรูป

[ตัวเลือก] เพื่อเลือกการทำงานของหน้าจอนี้ เช่น บันทึกการแก้ไขรูป

[ปากกา] ใช้เพื่อเลือกสีของปากกาที่ต้องการใช้ขีดเขียนบนรูป



รูปที่ 3.11 หน้าจอสำหรับเพิ่มรูปภาพ

จากรูปที่ 3.11 เป็นหน้าจอสำหรับเพิ่มรูปภาพที่ถ่ายจากที่เกิดเหตุ เมื่อเลือกรูปแล้วต้องทำการเพิ่มคำอธิบายสำหรับรูปด้วยจึงจะสามารถบันทึกลงฐานข้อมูลได้

[ตัวเลือก] เพื่อเลือกการทำงานของหน้าจอนี้ เช่น บันทึกรูป หรือ ลบลายเส้นที่ได้ขีดเขียนลงบนรูป

[ปากกา] ใช้เพื่อเลือกสีของปากกาที่ต้องการใช้ขีดเขียนบนรูป

รูปที่ 3.12 หน้าจอสำหรับดูรายละเอียดของผู้เคลมประกัน

จากรูปที่ 3.12 เป็นหน้าจอสำหรับดูรายละเอียดของผู้เคลมประกัน ซึ่งข้อมูลไม่สามารถแก้ไขได้

รูปที่ 3.13 หน้าจอสำหรับดูรายละเอียดของตัวประกันที่ได้ทำไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.13 เป็นหน้าจอสำหรับดูรายละเอียดของตัวประกันที่ได้ทำไว้ ซึ่งข้อมูลไม่สามารถแก้ไขได้

3.4 การออกแบบฐานข้อมูล (Database Design)

ในการพัฒนาโปรแกรมได้ออกแบบฐานข้อมูลเพื่อใช้เก็บข้อมูลต่างๆ ซึ่งมีรายละเอียดดังต่อไปนี้

PKey	Name	ID	Data Type	Size
<input checked="" type="checkbox"/>	claim_id	1	binary	16
<input type="checkbox"/>	disp_claim_id	2	integer	
<input type="checkbox"/>	ref_policy_id	3	binary	16
<input type="checkbox"/>	claim_short_descr	4	varchar	128
<input type="checkbox"/>	claim_notes	5	long varchar	
<input type="checkbox"/>	claim_date	6	timestamp	
<input type="checkbox"/>	ref_claim_status_id	7	integer	

รูปที่ 3.14 Table : claim

ตาราง claim ใช้สำหรับเก็บรายละเอียดของงานที่ถูกค่าต้องการเคลมประกัน ซึ่งประกอบด้วย

- claim_id ใช้เป็น primary key ของตาราง
- disp_claim_id ใช้เป็นลำดับของเรคคอร์ดข้อมูล ซึ่งก็คือ running number นั้นเอง
- ref_policy_id ใช้เป็น foreign key เพื่ออ้างอิงไปยังตาราง policy
- claim_short_descr ใช้เก็บข้อมูลคำบรรยายลักษณะงานที่ถูกค่าต้องการเคลมแบบสั้นๆ
- claim_notes ใช้เก็บข้อมูลคำบรรยายอื่นๆเพิ่มเติม
- claim_date ใช้เก็บวันที่ที่เปิดงานขึ้นนี้ขึ้นมา
- ref_claim_status_id ใช้เป็น foreign key เพื่ออ้างอิงไปยังตาราง claim_status

PKey	Name	ID	Data Type	Size
<input checked="" type="checkbox"/>	claim_image_NEW_id	1	binary	16
<input type="checkbox"/>	ref_claim_id	2	binary	16
<input type="checkbox"/>	claim_image_NEW_data	3	long binary	
<input type="checkbox"/>	ref_doodle_NEW_id	4	binary	16
<input type="checkbox"/>	claim_image_NEW_descr	5	varchar	128

รูปที่ 3.15 Table : claim_images_NEW

ตาราง claim_images_NEW ใช้สำหรับเก็บรายละเอียดของรูปภาพที่เกี่ยวข้องกับข้อมูลการเคลมในตาราง claim ซึ่งประกอบด้วย

- claim_image_NEW_id ใช้เป็น primary key ของตาราง
- ref_claim_id ใช้เป็น foreign key เพื่ออ้างอิงไปยังตาราง claim
- claim_image_NEW_data ใช้เก็บรูปภาพของรถที่ได้รับความเสียหาย ณ ที่เกิดเหตุ
- ref_doodle_NEW_id ใช้เป็น foreign key เพื่ออ้างอิงไปยังตาราง doodle_NEW
- claim_image_NEW_descr ใช้เก็บคำอธิบายรูปภาพ

PKey	Name	ID	Data Type	Size
<input checked="" type="checkbox"/>	ref_claim_id	1	binary	16
<input checked="" type="checkbox"/>	claim_item_seq	2	integer	
<input type="checkbox"/>	claim_item_quant	3	double	
<input type="checkbox"/>	claim_item_value	4	double	
<input type="checkbox"/>	claim_item_descr	5	varchar	128
<input type="checkbox"/>	claim_item_date	6	timestamp	

รูปที่ 3.16 Table : claim_items

ตาราง claim_items ใช้สำหรับเก็บรายละเอียดของอุปกรณ์ที่คาดว่าจะใช้ในการซ่อมรถ ซึ่งประกอบด้วย

- ref_claim_id ใช้เป็น foreign key เพื่ออ้างอิงไปยังตาราง claim และใช้ร่วมกับฟิลด์ claim_item_seq เพื่อเป็น primary key
- claim_item_seq ใช้เป็นลำดับของเรคคอร์ดข้อมูล ซึ่งก็คือ running number นั้นเอง และใช้ร่วมกับฟิลด์ ref_claim_id เพื่อเป็น primary key
- claim_item_quant ใช้เก็บจำนวนของชิ้นส่วนรถที่จะใช้ซ่อม
- claim_item_value ใช้เก็บราคาชิ้นส่วนรถแต่ละชิ้น
- claim_item_descr ใช้เก็บคำอธิบายของชิ้นส่วน ซึ่งอาจจะเป็นชื่อของชิ้นส่วนนั้นๆ
- claim_item_date ใช้เก็บวันที่สร้างเรคคอร์ดนี้ขึ้นมา

PKey	Name	ID	Data Type	Size
<input checked="" type="checkbox"/>	claim_status_id	1	integer	
<input type="checkbox"/>	claim_status	2	vvarchar	25

รูปที่ 3.17 Table : claim_status

ตาราง claim_status ใช้สำหรับเก็บคำอธิบายสถานะของงานที่ลูกค้าต้องการเคลมประกัน ซึ่งประกอบด้วย

- claim_status_id ใช้เป็น primary key ของตาราง
- claim_status ใช้เก็บคำอธิบายสถานะของงานที่ลูกค้าต้องการเคลมประกัน

PKey	Name	ID	Data Type	Size
<input checked="" type="checkbox"/>	doodle_NEW_id	1	binary	16
<input type="checkbox"/>	doodle_NEW_running	2	integer	

รูปที่ 3.18 Table : doodles_NEW

ตาราง doodles_NEW ใช้สำหรับเป็นตารางตัวกลางที่ช่วยในการเชื่อมโยงความสัมพันธ์ระหว่างรูปภาพในตาราง claim_images_NEW และลายเส้นของรูปภาพนั้นๆที่อยู่ในตาราง points_NEW ซึ่งประกอบด้วย

- doodle_NEW_id ใช้เป็น primary key ของตาราง
- doodle_NEW_running ใช้เป็นลำดับของเรคคอร์ดข้อมูล ซึ่งก็คือ running number นั้นเอง

PKey	Name	ID	Data Type	Size
<input checked="" type="checkbox"/>	emp_id	1	binary	16
<input type="checkbox"/>	emp_lname	2	vvarchar	20
<input type="checkbox"/>	emp_fname	3	vvarchar	20
<input type="checkbox"/>	emp_login	4	vvarchar	20
<input type="checkbox"/>	emp_pwd	5	vvarchar	20

รูปที่ 3.19 Table : employee

ตาราง employee ใช้สำหรับเก็บข้อมูลต่างๆของพนักงาน ซึ่งประกอบด้วย

- emp_id ใช้เป็น primary key ของตาราง
- emp_lname ใช้เก็บนามสกุลพนักงาน
- emp_fname ใช้เก็บชื่อพนักงาน
- emp_login ใช้เก็บชื่อในการเข้าสู่ระบบโปรแกรม

- emp_pwd ใช้เก็บรหัสผ่านในการเข้าสู่ระบบโปรแกรม

PKey	Name	ID	Data Type	Size
<input checked="" type="checkbox"/>	ref_doodle_NEW_id	1	binary	16
<input checked="" type="checkbox"/>	point_NEW_id	2	integer	
<input type="checkbox"/>	point_NEW_stream	3	long varchar	
<input type="checkbox"/>	point_NEW_color	4	unsigned int	

รูปที่ 3.20 Table : points_NEW

ตาราง points_NEW ใช้สำหรับเก็บข้อมูลหลายเส้นที่วาดลงบนรูป ซึ่งจะอ้างอิงไปยังรูปที่เก็บอยู่ในตาราง claim_images_NEW ซึ่งประกอบด้วย

- ref_doodle_NEW_id ใช้เป็น foreign key เพื่ออ้างอิงไปยังตาราง doodle_NEW และใช้ร่วมกับฟิลด์ point_NEW_id เพื่อเป็น primary key ของตาราง
- point_NEW_id ใช้เป็นลำดับของเรคคอร์ดข้อมูล ซึ่งก็คือ running number นั้นเอง และใช้ร่วมกับฟิลด์ ref_doodle_NEW_id เพื่อเป็น primary key
- point_NEW_stream ใช้เก็บข้อมูลหลายเส้นที่วาดลงบนรูป
- point_NEW_color ใช้เก็บค่าสีของหลายเส้นที่วาดลงบนรูป

PKey	Name	ID	Data Type	Size
<input checked="" type="checkbox"/>	policy_id	1	binary	16
<input type="checkbox"/>	disp_policy_id	2	integer	
<input type="checkbox"/>	ref_holder_id	3	binary	16
<input type="checkbox"/>	policy_name	4	varchar	30
<input type="checkbox"/>	deductible	5	integer	

รูปที่ 3.21 Table : policy

ตาราง policy ใช้สำหรับเก็บข้อมูลต่างๆที่เกี่ยวกับกรมธรรม์ที่ลูกค้าทำไว้ ซึ่งประกอบด้วย

- policy_id ใช้เป็น primary key ของตาราง
- disp_policy_id ใช้เป็นลำดับของเรคคอร์ดข้อมูล ซึ่งก็คือ running number นั้นเอง
- ref_holder_id ใช้เป็น foreign key เพื่ออ้างอิงไปยังตาราง policy_holder
- policy_name ใช้เก็บชื่อกรมธรรม์
- deductible ใช้เก็บจำนวนวงเงินที่ลูกค้าสามารถเคลมได้

PKey	Name	ID	Data Type	Size
<input checked="" type="checkbox"/>	holder_id	1	binary	16
<input type="checkbox"/>	holder_fname	2	varchar	30
<input type="checkbox"/>	holder_lname	3	varchar	30
<input type="checkbox"/>	holder_gender	4	char	1
<input type="checkbox"/>	holder_dob	5	date	
<input type="checkbox"/>	holder_addr	6	varchar	64
<input type="checkbox"/>	holder_city	7	varchar	64
<input type="checkbox"/>	holder_state	8	varchar	10
<input type="checkbox"/>	holder_zip	9	varchar	10
<input type="checkbox"/>	holder_areacode	10	varchar	10
<input type="checkbox"/>	holder_phone	11	varchar	10
<input type="checkbox"/>	ref_emp_id	12	binary	16

รูปที่ 3.22 Table : policy_holder

ตาราง policy_holder ใช้สำหรับเก็บข้อมูลต่างๆที่เกี่ยวข้องกับลูกค้าผู้ถือกรมธรรม์ ซึ่งประกอบด้วย

- holder_id ใช้เป็น primary key ของตาราง
- holder_fname ใช้เก็บชื่อลูกค้า
- holder_lname ใช้เก็บนามสกุลลูกค้า
- holder_gender ใช้เก็บเพศของลูกค้า
- holder_dob ใช้เก็บวันเดือนปีเกิดของลูกค้า
- holder_addr ใช้เก็บที่อยู่ของลูกค้า
- holder_city ใช้เก็บแขวงที่อยู่ของลูกค้า
- holder_state ใช้เก็บเขตที่อยู่ของลูกค้า
- holder_zip ใช้เก็บรหัสไปรษณีย์
- holder_areacode ใช้เก็บรหัสหน้าเบอร์โทรศัพท์
- holder_phone ใช้เก็บเบอร์โทรศัพท์
- ref_emp_id ใช้เป็น foreign key เพื่ออ้างอิงไปยังตาราง employee

PKey	Name	ID	Data Type	Size
<input checked="" type="checkbox"/>	item_id	1	integer	
<input type="checkbox"/>	item_name	2	varchar	32
<input type="checkbox"/>	item_cost	3	double	

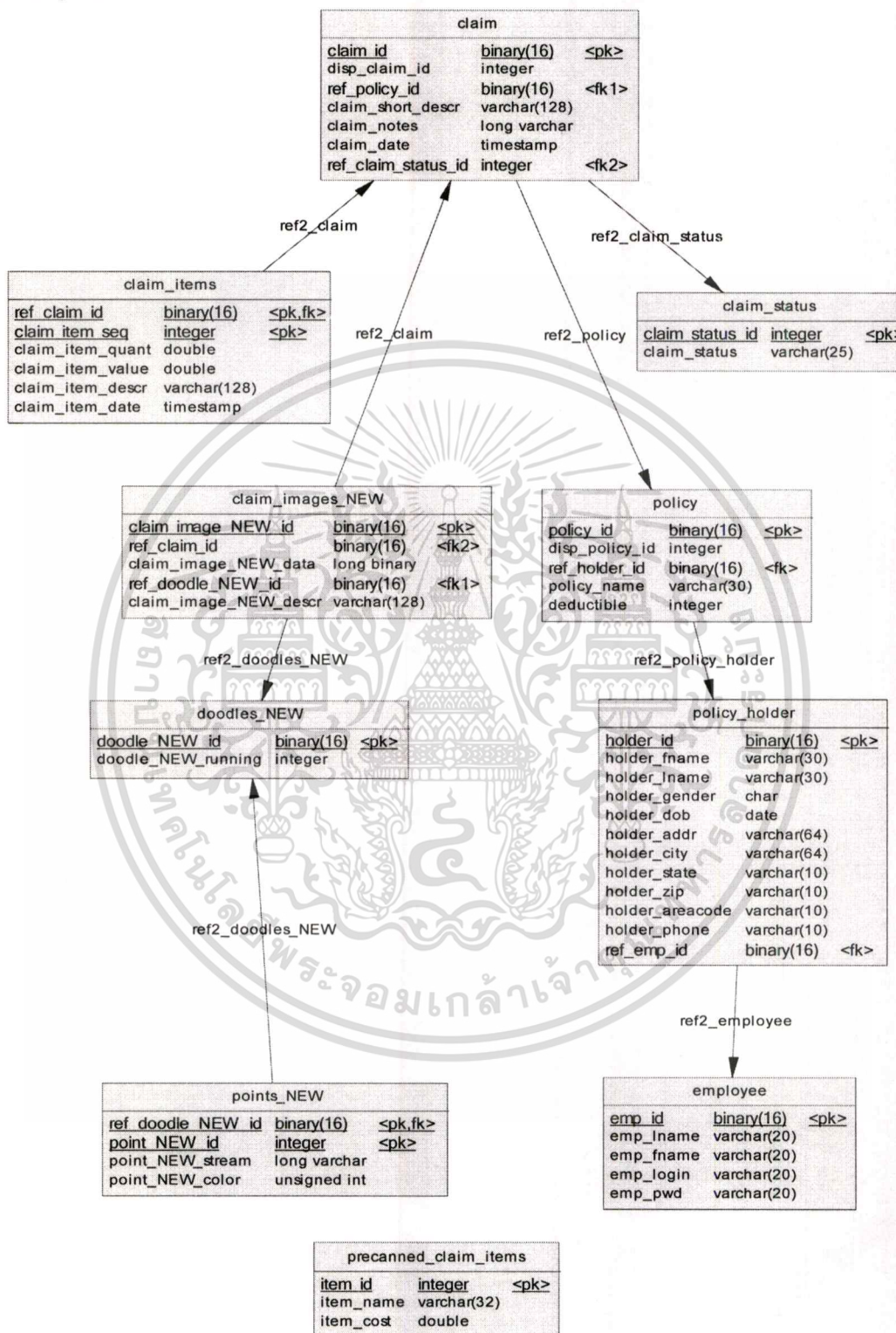
รูปที่ 3.23 Table : precanned_claim_items

ตาราง precanned_claim_items ใช้สำหรับเก็บข้อมูลของชิ้นส่วนอุปกรณ์ ที่จะใช้ในการเคลมประกัน ซึ่งประกอบด้วย

- item_id ใช้เป็น primary key ของตาราง
- item_name ใช้เก็บชื่อของชิ้นส่วนอุปกรณ์
- item_cost ใช้เก็บราคาของชิ้นส่วนอุปกรณ์



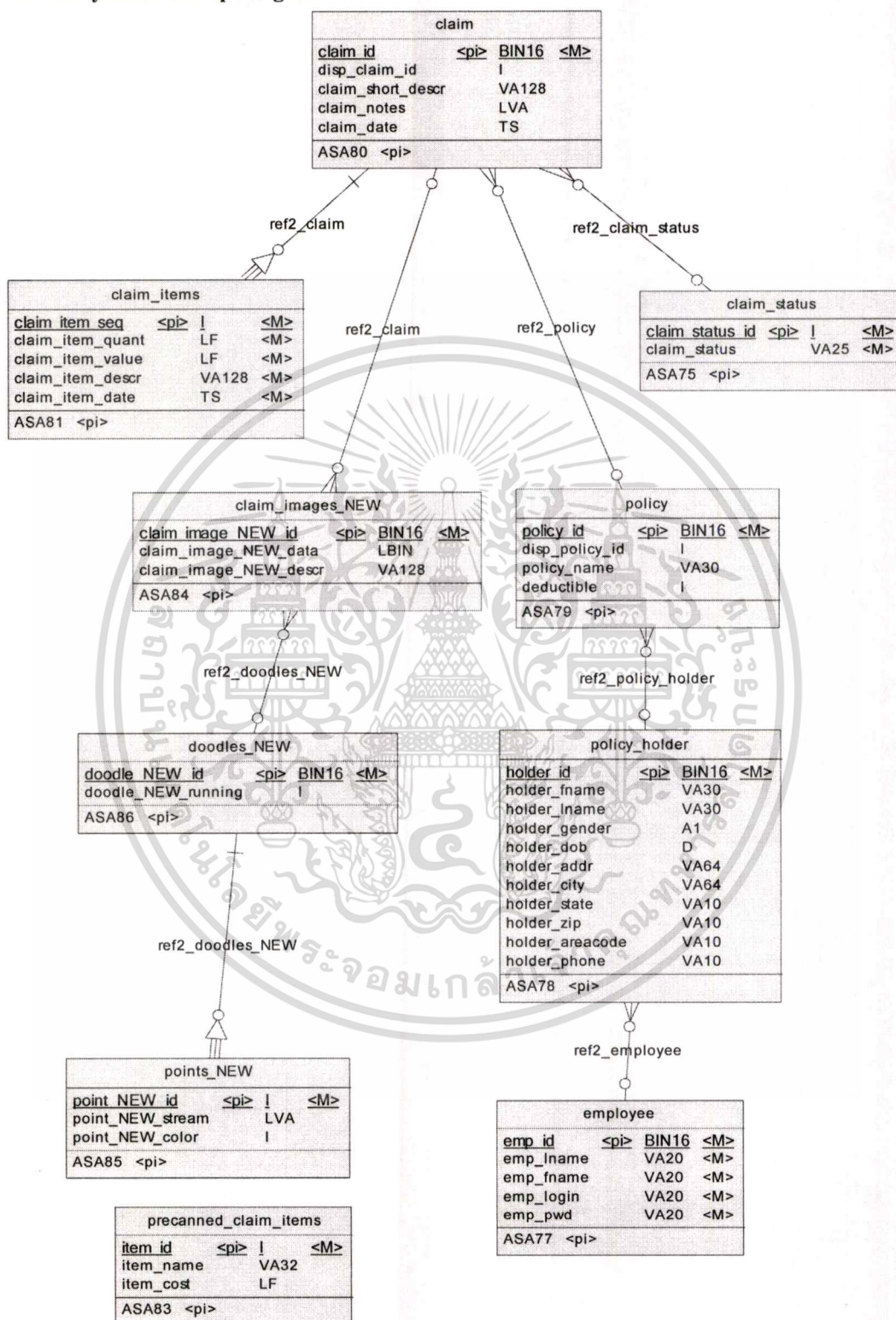
3.5 Physical Data Model



รูปที่ 3.24 Physical Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 Entity Relationship Diagram



รูปที่ 3.25 Entity Relationship Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การพัฒนาโครงการ

4.1 ฟังก์ชันต่างๆเพื่อเรียกใช้งาน

- `f_callsp_get_sync_user` ใช้สำหรับดึง MobiLink user มาจากฐานข้อมูล

ลักษณะการทำงาน คือ เรียกใช้ store procedure ที่ชื่อ `sp_get_sync_user_test` เพื่อทำการดึง MobiLink user มาจากฐานข้อมูล จากนั้นมีการตรวจสอบว่า store procedure ทำงานได้ถูกต้องหรือไม่ ถ้าถูกต้องจะคืนค่า MobiLink user ถ้าผิดก็จะแสดงข้อผิดพลาดออกมา

```
string ls_mluser
DECLARE Getmluser_sp PROCEDURE FOR sp_get_sync_user_test;
EXECUTE Getmluser_sp;

if SQLCA.sqlcode = -1 then
    MessageBox("Stored Procedure Error!",SQLCA.sqlerrtext)
    return "ERROR"
else
    FETCH Getmluser_sp INTO :ls_mluser;
    CLOSE Getmluser_sp;
    return ls_mluser
end if
```

- `f_openpic` ใช้สำหรับเลือกรูปที่ต้องการเพื่อเตรียมบันทึกลงสู่ฐานข้อมูล

ลักษณะการทำงาน คือ เปิดหน้าต่างสำหรับการเลือกรูปขึ้นมา เมื่อเลือกรูปเสร็จแล้ว จะทำการดึงข้อมูลรูป มาไว้ในตัวแปรชนิด Blob เพื่อนำไปใช้ต่อไป

```
String sFullName, sFileName
```

```
If GetFileOpenName ("เลือกรูป", sFullName, sFileName, "BMP", &
```

```

"GIF Files (*.GIF),*.GIF," + &
"JPG Files (*.JPG),*.JPG," + &
"All Files (*.*) ,*.*)" &
< 1 then return "FAIL"

```

Integer fp, i, iLoops

Long iLen, iLoad

Blob blobImage, blobTmp

```
iLen = FileLength(sFullName)
```

```
IF iLen > 32765 THEN
```

```
    IF Mod(iLen, 32765) = 0 THEN
```

```
        iLoops = iLen/32765
```

```
    ELSE
```

```
        iLoops = (iLen/32765) + 1
```

```
    END IF
```

```
ELSE
```

```
    iLoops = 1
```

```
END IF
```

```
fp = FileOpen(sFullName, StreamMode!)
```

```
If fp = -1 Then
```

```
    MessageBox("ERROR", 'Cannot open picture file')
```

```
    Return "FAIL"
```

```
End If
```

```
SetPointer(HourGlass!)
```

```
For i=1 To iLoops
```

```
    iLoad = FileRead(fp, blobTmp)
```

```
    blobImage = blobImage + blobTmp
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Next

```
gbl_blob_image = blobImage
```

```
SetPointer(Arrow!)
```

```
FileClose(fp)
```

```
return "OK"
```

- `f_savepic` ใช้สำหรับบันทึกข้อมูลรูปลงสู่ฐานข้อมูล

ลักษณะการทำงาน คือ ทำการเก็บค่าต่างๆเกี่ยวกับรูป ซึ่งประกอบด้วย ข้อมูลรูป, ลายเส้นที่ขีดเขียนบนรูป และสีของลายเส้น เก็บลงสู่ฐานข้อมูล

```
if w_openpic.sle_descr.text = " then
```

```
    messagebox('ข้อผิดพลาด','กรุณาเพิ่มคำอธิบาย')
```

```
    return
```

```
end if
```

```
insert into doodles_new (doodle_NEW_id) values (newid(*));
```

```
if sqlca.sqlcode <> 0 then
```

```
    messagebox('ข้อผิดพลาด',sqlca.sqlerrtext)
```

```
    rollback;
```

```
    return
```

```
end if
```

```
string ls_doodle_NEW_id
```

```
select uuidtostr(doodle_NEW_id) into :ls_doodle_NEW_id from doodles_NEW &
```

```
where doodle_NEW_running = (select max(doodle_NEW_running) from doodles_NEW);
```

```
string ls_claim_image_NEW_descr
```

```
ls_claim_image_NEW_descr = w_openpic.sle_descr.text
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

insert into claim_images_NEW
(ref_claim_id,ref_doodle_NEW_id,claim_image_NEW_descr)values &
(strtouid(:gs_claim_id),strtouid(:ls_doodle_NEW_id),:ls_claim_image_NEW_descr);

if sqlca.sqlcode <> 0 then
    messagebox('ข้อผิดพลาด',sqlca.sqlerrtext)
    rollback;
    return
end if

updateblob claim_images_NEW set claim_image_NEW_data = :gbl_blob_image &
where claim_image_NEW_data is null;

if sqlca.sqlcode <> 0 then
    messagebox('ข้อผิดพลาด',sqlca.sqlerrtext)
    rollback;
    return
end if

String sTmp
sTmp = w_openpic.uo_open.of_GetSignature()

integer li_long,li_countPipe,i
string ls_stream[]
li_long = len(stmp)

integer li_Curpos = 1
for i=1 to li_long
    if mid(sTmp,i,1) <> '|' then

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        ls_stream[li_Curpos] = ls_stream[li_Curpos] + mid(sTmp,i,1)
    else
        li_Curpos++
    end if
next
li_countPipe = li_Curpos - 1

integer li_realSubDoodle
li_realSubDoodle = 0
for i=1 to li_countPipe
    if ls_stream[i] <> "" then
        li_realSubDoodle++
    end if
next

if li_countPipe < 1 then
    COMMIT;
    messagebox('ผลการทำงาน','บันทึกข้อมูลเรียบร้อยแล้ว')
    close (w_openpic)
    return // Suddenly EXIT
end if

integer j=0
for i=1 to li_countPipe
    if ls_stream[i] <> "" then
        j++
        insert into points_NEW
(ref_doodle_NEW_id,point_NEW_stream,point_NEW_color) &
values (strtouuid(:ls_doodle_NEW_id),:ls_stream[i],:gul_color[j]);
    end if

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if sqlca.sqlcode <> 0 then
    messagebox('ข้อผิดพลาด',sqlca.sqlerrtext)
    rollback;
    return
end if
next

```

```

COMMIT;
messagebox('ผลการทำงาน','บันทึกข้อมูลเรียบร้อยแล้ว')
close (w_openpic)

```

- **f_savepic_editpic** ใช้สำหรับดึงข้อมูลของรูปที่เคยบันทึกลงฐานข้อมูลแล้วกลับขึ้นมาแก้ไข แล้วทำการบันทึกการเปลี่ยนแปลงกลับสู่ฐานข้อมูล ลักษณะการทำงาน คือ ดึงข้อมูลต่างๆที่เกี่ยวข้องกับรูปที่เลือก ขึ้นมาจากฐานข้อมูล ซึ่งประกอบด้วย ข้อมูลรูป,ลายเส้นที่ขีดเขียนบนรูป และสีของลายเส้น จากนั้นจัดเตรียมข้อมูลต่างๆที่ได้แก้ไขกับรูป กลับ ไปบันทึกยังฐานข้อมูล

```

string stmp2
stmp2 = w_editpic.wo_editpic.of_GetSignature()
string stmp3
stmp3 = mid(stmp2,gi_oldLength+1,len(stmp2))
integer li_long,li_countPipe,i
string ls_stream[]

```

```
li_long = len(stmp3)
```

```
integer li_Curpos = 1
```

```
for i=1 to li_long
```

```
    if mid(stmp3,i,1) <> "|" then
```

```

        ls_stream[li_Curpos] = ls_stream[li_Curpos] + mid(sTmp3,i,1)
    else
        li_Curpos++
    end if
next
li_countPipe = li_Curpos - 1

integer li_realSubDoodle
li_realSubDoodle = 0
for i=1 to li_countPipe
    if ls_stream[i] <> "" then
        li_realSubDoodle++
    end if
next

if li_countPipe < 1 then
    COMMIT;
    messagebox('ผลการทำงาน','บันทึกข้อมูลเรียบร้อยแล้ว')
    close (w_editpic)
    return
end if

integer j=0
for i=1 to li_countPipe
    if ls_stream[i] <> "" then //this check for pocket pc
        j++
        insert into points_NEW
        (ref_doodle_NEW_id,point_NEW_stream,point_NEW_color) &
        values (strtouid(:gs_doodle_NEW_id),:ls_stream[i],:gul_color[j]);
    end if

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if sqlca.sqlcode <> 0 then
    messagebox('ข้อผิดพลาด',sqlca.sqlerrtext)
    rollback;
    return
end if
next

```

```

COMMIT;
messagebox('ผลการทำงาน','บันทึกข้อมูลเรียบร้อยแล้ว')
close (w_editpic)

```

- **f_totalcost** ใช้สำหรับคำนวณราคาของชิ้นส่วนรถยนต์ที่เคลมประกัน ลักษณะการทำงาน คือ ดึงข้อมูลราคาและจำนวนชิ้นส่วนจากฐานข้อมูล มาคูณกัน แล้วนำทุกรายการมารวมกันเป็นราคารวม

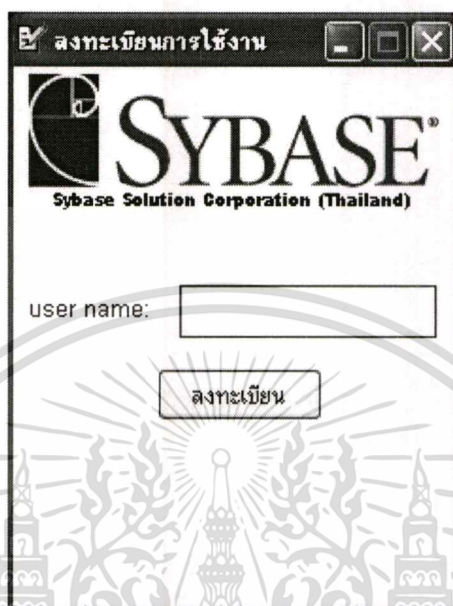
```

string ls_totalCost
select sum(claim_item_quant * claim_item_value) into :ls_totalCost
from claim_items where ref_claim_id = STRTOUUID(:arg_claim_id);

if sqlca.sqlcode <> 0 then
    messagebox('Error',sqlca.sqlerrtext)
    return "ERROR"
else
    return ls_totalCost
end if

```

4.2 หน้าจอลงทะเบียน



รูปที่ 4.1 พัฒนาหน้าจอลงทะเบียน

Code โปรแกรมของ (window) w_login ที่ event open()

```
this.p_1.picturename = "\program files\logo.bmp"
```

Code โปรแกรมของ (CommandButton) cb_1 ที่ event clicked()

```
// validate input
```

```
if parent.sle_1.text = "" then
```

```
    return
```

```
end if
```

```
// confirm user to registration
```

```
int li_confirm
```

```
li_confirm = MessageBox("ยืนยัน", "คุณต้องการลงทะเบียนด้วย :: "&
```

```
    +sle_1.text+ " :: ใช่หรือไม่", Exclamation!, YesNo!)
```

```
if li_confirm <> 1 then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

return
end if

//call store procedure (sp_create_temp_tables_employee)
DECLARE emp_sp PROCEDURE FOR sp_create_temp_tables_employee &
    in_user = :parent.sle_1.text;

EXECUTE emp_sp;

if SQLCA.sqlcode = -1 then
    MsgBox("Stored Procedure Error!",SQLCA.sqlerrtext)
return
else
    CLOSE emp_sp;
end if

//call store procedure (sp_new_user_test)
DECLARE NewUser_sp PROCEDURE FOR sp_new_user_test &
    in_username = :parent.sle_1.text;

EXECUTE NewUser_sp;

if SQLCA.sqlcode = -1 then
    MsgBox("Stored Procedure Error!",SQLCA.sqlerrtext)
return
else
    CLOSE NewUser_sp;
    open (w_insurance_main)
    close(parent)
end if

```

4.3 พัฒนาหน้าจอแสดงรายการลูกค้าที่เคลม



รูปที่ 4.2 พัฒนาหน้าจอแสดงรายการลูกค้าที่เคลม

Code โปรแกรมของ (window) w_insurance_main ที่ event activate()

```
// refresh data on DW
tab_1.tabpage_1.dw_listclaim.settransobject(sqlca)
tab_1.tabpage_1.dw_listclaim.retrieve( )
```

Code โปรแกรมของ (window) w_insurance_main ที่ event open()

```
// set picture on each tabpage
tab_1.tabpage_1.p_1.picturename = "\program files\logo.bmp"
tab_1.tabpage_3.p_2.picturename = "\program files\logo.bmp"
tab_1.tabpage_2.p_3.picturename = "\program files\logo.bmp"

// show message on tab_About
tab_1.tabpage_3.mle_about.text = 'Mobile Insurance Aduster Application'+~r~n'+ &
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
'Powered by Adaptive Server Anywhere,' + &
'~r~n' + 'Mobilink and Pocket Power Builder' + '~r~n' + &
'~r~n' + 'Sybase Solutions Corporation (Thailand)+' &
'http://www.sybase.co.th'
```

```
// Prepare data for sync Tab
```

```
string res
```

```
res = f_callsp_get_sync_user() // call store procedure
```

```
if res <> "ERROR" then
```

```
    tab_1.tabpage_2.st_user.text = res
```

```
    gs_mluser = res
```

```
else
```

```
    return
```

```
end if
```

```
//get data from registry
```

```
this.postevent("ue_postopen")
```

Code โปรแกรมของ (window) w_insurance_main ที่ event ue_postopen()

```
string ls_regkey, ls_work
```

```
integer rc
```

```
ls_regkey = "Host"
```

```
rc = RegistryGet(i_uosync.APP_REGPATH, ls_regkey, RegString!, ls_work)
```

```
if rc = 1 then
```

```
    tab_1.tabpage_2.sle_host.text = ls_work
```

```
end if
```

Code โปรแกรมของ (CommandButton) tab_1.tabpage_1.cb_1 ที่ event clicked()

```
// check row in datawindow
```

```
if tab_1.tabpage_1.dw_listclaim.getrow( ) = 0 then
```

```
    return
```

```
end if
```

```
// Assign value of claim_id to gs_claim_id
```

```
gs_claim_id = parent.dw_listclaim.getItemstring(parent.dw_listclaim.GETROW(),5 )
```

```
// Assign value of holder_id to gs_holder_id
```

```
SELECT uuidtostr(PH.holder_id) into :gs_holder_id &
```

```
from policy_holder PH, claim C, policy P &
```

```
WHERE (C.ref_policy_id = P.policy_id) and &
```

```
(P.ref_holder_id = PH.holder_id) and &
```

```
(C.claim_id = STRTOUUID(:gs_claim_id));
```

```
// Assign value of policy_id to gs_policy_id
```

```
SELECT uuidtostr(P.policy_id) into :gs_policy_id &
```

```
from policy P, claim C &
```

```
WHERE (C.ref_policy_id = P.policy_id) and &
```

```
(C.claim_id = STRTOUUID(:gs_claim_id));
```

```
// Assign value of disp_claim_id to gui_disp_claim_id
```

```
gui_disp_claim_id = parent.dw_listclaim.getItemnumber(parent.dw_listclaim.GETROW(),1 )
```

```
open (w_show_claim)
```

Code โปรแกรมของ (CommandButton) tab_1.tabpage_1.cb_2 ที่ event clicked()

```

// confirm user to send job
int li_confirm
li_confirm = MessageBox("ยืนยัน", "คุณต้องการส่งงานใช่หรือไม่?", Exclamation!, YesNo!)
if li_confirm <> 1 then
    return
end if

// check row in datawindow
if tab_1.tabpage_1.dw_listclaim.getrow() = 0 then
    return
end if

// Assign value of claim_id to gs_claim_id
string ls_claim_id
ls_claim_id = parent.dw_listclaim.getItemstring(parent.dw_listclaim.GETROW(),5)

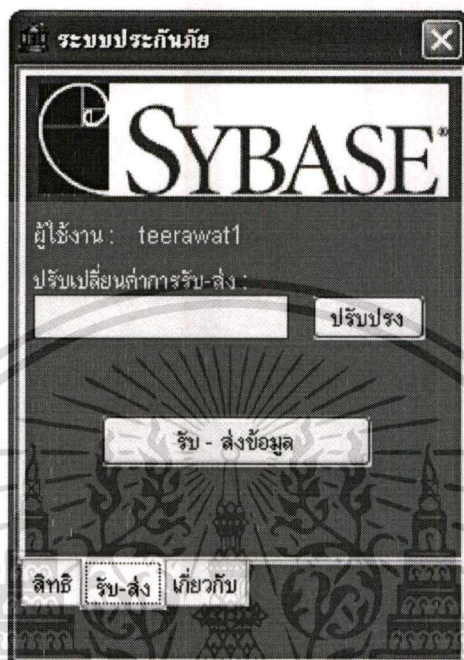
update claim set ref_claim_status_id=2 &
where claim_id = strtouuid(:ls_claim_id);

if sqlca.sqlcode <> 0 then
    messagebox("ข้อผิดพลาด",sqlca.sqlerrtext)
    return
else
    commit;
    messagebox("ผลการทำงาน","ดำเนินการเรียบร้อยแล้ว")
end if

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 หน้าจอสำหรับการ synchronization



รูปที่ 4.3 พัฒนาหน้าจอสำหรับการ synchronization

Code โปรแกรมของ (CommandButton) tab_1.tabpage_2.cb_update ที่ event clicked()

```
string ls_work, ls_regkey
```

```
integer rc
```

```
ls_work = w_insurance_main.tab_1.tabpage_2.sle_host.text
```

```
ls_regkey = "Host"
```

```
rc = RegistrySet(i_uosync.APP_REGPATH, ls_regkey, RegString!, ls_work)
```

```
if rc <> 1 then
```

```
    messagebox("ERROR","registry")
```

```
else
```

```
    messagebox("ผลการทำงาน","ดำเนินการเรียบร้อยแล้ว")
```

```
end if
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Code โปรแกรมของ (CommandButton) tab_1.tabpage_2.cb_sync ที่ event clicked()
gf_insurance_configure_sync()

4.5 หน้าจอบันทึกและแสดงรายการ ชั้นส่วนที่ถูกคัดลอกการเคลม



รูปที่ 4.4 พัฒนาหน้าจอบันทึกและแสดงรายการ ชั้นส่วนที่ถูกคัดลอกการเคลม

Code โปรแกรมของ (Window) w_show_claim ที่ event activate()

// show disp_claim_id on the bottom of page

```
this.st_disp_claim_id.text = string(gui_disp_claim_id)
```

// refresh data window on Tabpage1

```
tab_1.tabpage_1.dw_detailclaim.settransobject( sqlca)
```

```
tab_1.tabpage_1.dw_detailclaim.retrieve( gs_claim_id)
```

// refresh total cost on Tabpage1

```
tab_1.tabpage_1.em_totalCost.text = f_totalcost(gs_claim_id)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// refresh DW on Tabpage3 (image)
tab_1.tabpage_3.dw_picdescr.settransobject( sqlca)
tab_1.tabpage_3.dw_picdescr.retrieve(gs_claim_id)

// refresh DW on Tabpage4 (claimant)
tab_1.tabpage_4.dw_claimant.settransobject( sqlca)
tab_1.tabpage_4.dw_claimant.retrieve( gs_holder_id)
```

```
// refresh DW on Tabpage5 (claim_policy)
tab_1.tabpage_5.dw_cliam_policy.settransobject( sqlca)
tab_1.tabpage_5.dw_cliam_policy.retrieve( gs_policy_id)
```

Code โปรแกรมของ (Window) w_show_claim ที่ event close()

```
// update note on Tabpage2
UPDATE claim set claim_notes = :tab_1.tabpage_2.mle_note.text &
WHERE claim_id = STRTOUUID(:gs_claim_id);

if sqlca.sqlcode <> 0 then
    messagebox('ข้อผิดพลาด',sqlca.sqlerrtext)
    return
else
    commit;
end if
```

Code โปรแกรมของ (Window) w_show_claim ที่ event close()

```
//ddlb_precanned on Tabpage1
string ls_itemName,ls_itemCost
```

```
DECLARE Cur_precanned CURSOR FOR
```

```

SELECT item_Name,item_Cost from precanned_claim_items;
OPEN Cur_precanned;

FETCH NEXT Cur_precanned INTO :ls_itemName,:ls_itemCost;
do while SQLCA.sqlcode = 0
    tab_1.tabpage_1.ddlb_precanned.Additem(ls_itemName+space(1)+&
        '+'+space(1)+'B'+ls_itemCost)
    FETCH NEXT Cur_precanned INTO :ls_itemName,:ls_itemCost;
loop
CLOSE Cur_precanned;

// comment on Tabpage2
tab_1.tabpage_2.st_comment.text = &
"ใส่หมายเหตุ ของการอ้างสิทธิที่ต้องการ" + crlf + "ระบบจะบันทึกโดยอัตโนมัติ"

// retrieve note on Tabpage2
string ls_claim_note

SELECT claim_notes INTO :ls_claim_note FROM claim &
WHERE claim_id = STRTOUUID(:gs_claim_id);

tab_1.tabpage_2.mle_note.text = ls_claim_note

```

Code โปรแกรมของ (CommandButton) tab_1.tabpage_1.cb_1 ที่ event clicked()

```

// Validate input data
if parent.ddlb_precanned.text = " then
    messagebox('คำเตือน','กรุณาเลือกรายการ')
    return
elseif integer(parent.em_amount.text) < 1 then
    messagebox('คำเตือน','กรุณากรอกจำนวนให้ถูกต้อง')

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

return
end if

// Prepare value for add to DB
integer li_claim_item_quant
double ld_claim_item_value

// ref_claim_id ==> gs_claim_id
li_claim_item_quant = integer(parent.em_amount.text)
ld_claim_item_value = double(is_itemCost)
// claim_item_descr ==> is_itemName

// Confirm your decision
int li_confirm
li_confirm = MessageBox("ยืนยัน", "ต้องการบันทึกข้อมูล?", Exclamation!, YesNo!)

if li_confirm = 1 then
    // Insert data to claim_items TABLE
    INSERT INTO claim_items (ref_claim_id, claim_item_quant, &
    claim_item_value, claim_item_descr) VALUES(&
    STRTOUUID(:gs_claim_id), :li_claim_item_quant, &
    :ld_claim_item_value, :is_itemName);

    if sqlca.sqlcode <> 0 then
        messagebox('ข้อผิดพลาด', sqlca.sqlerrtext)
        return
    else
        commit;
        messagebox('แจ้งผลการทำงาน', 'ปรับปรุงฐานข้อมูลเรียบร้อยแล้ว')
    end if
end if

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

end if

4.6 หน้าจอสำหรับทำงานเกี่ยวกับรูปภาพที่ถ่ายจากที่เกิดเหตุ



รูปที่ 4.5 พัฒนาหน้าจอสำหรับทำงานเกี่ยวกับรูปภาพที่ถ่ายจากที่เกิดเหตุ

Code โปรแกรมของ (CommandButton) tab_1.tabpage_3.cb_editpic ที่ event clicked()

```
// check row in datawindow
```

```
if tab_1.tabpage_3.dw_picdescr.getrow() = 0 then
```

```
    return
```

```
end if
```

```
// Assign value of claim_image_NEW_id to gs_claim_image_NEW_id
```

```
gs_claim_image_NEW_id = parent.dw_picdescr.getItemstring
```

```
    (parent.dw_picdescr.GETROW(),2 )
```

```
// Assign value of ref_doodle_NEW_id to gs_doodle_NEW_id
```

```
gs_doodle_NEW_id = parent.dw_picdescr.getItemstring(parent.dw_picdescr.GETROW(),3 )
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
open (w_editpic)
```

Code โปรแกรมของ (CommandButton) tab_1.tabpage_3.cb_addpicture ที่ event clicked()

```
string err,ls_points
```

```
err = f_openpic()
```

```
if err <> "OK" then
```

```
    return
```

```
end if
```

```
open(w_openpic)
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การทดลองและผลที่ได้

5.1 อุปกรณ์และเครื่องมือที่ใช้

Hardware: Pocket PC

- Memory
 - System RAM Size: 64 MB
 - System ROM Size: 32 MB
- Version
 - OS Version: Windows CE 4.20
- Display
 - Display Horizontal Pixels: 240 pixels
 - Display Vertical Pixels: 320 pixels
- System
 - Manufacturer: Hewlett-Packard Company
 - Model ID: hp iPAQ h2200
 - Processor Type: Intel(R) PXA255

Software: เครื่องมือที่ใช้ในการพัฒนา

- Pocket PowerBuilder version 1.5.2
- Adaptive Server Anywhere version 9
- Microsoft ActiveSync version 3.7

5.2 การทดลอง

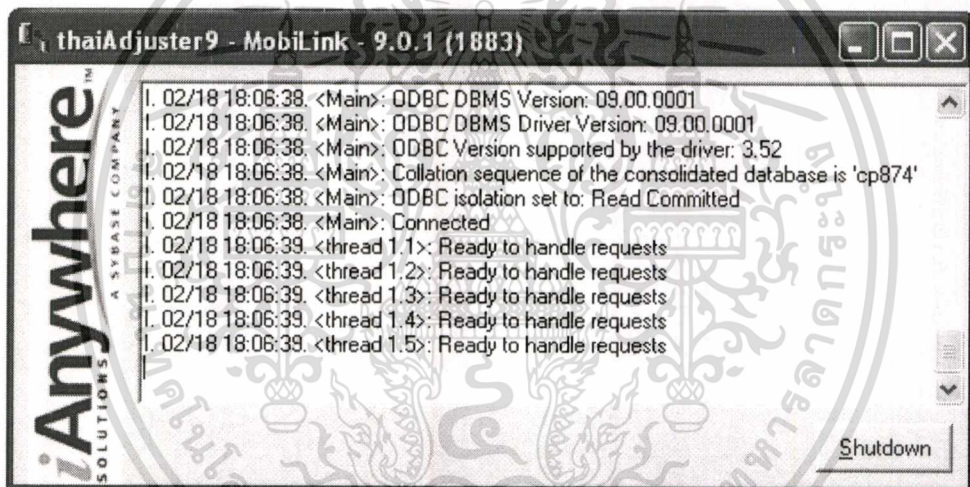
ติดตั้ง software บน Pocket PC

- ติดตั้ง Pocket PowerBuilder version 1.5.2 โดยใช้ไฟล์
PocketPowerBuilder.ARM.CAB
- ติดตั้ง Adaptive Server Anywhere version 9 โดยใช้ไฟล์ asa_ce.ARM.30.CAB

- ติดตั้งโปรแกรมประยุกต์ที่ได้พัฒนาขึ้น โดยใช้ไฟล์ insurance.exe
- ติดตั้ง datasource name โดยใช้ไฟล์ thaiCEAdjuster9.DSN
- ติดตั้งฐานข้อมูล ASA ที่ได้สร้างไว้ ชื่อไฟล์ thaiCEAdjuster9.db และ thaiCEAdjuster9.log
- ติดตั้งโปรแกรม ThaiWinCE2.00.15.4.exe เพื่อให้ Pocket PC สามารถใช้งานภาษาไทยได้

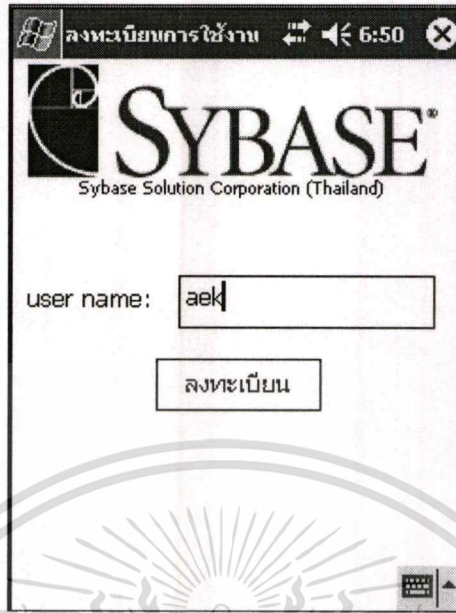
ขั้นตอนการทดลอง

1. Start MobiLink server ที่อยู่บน server เดียวกันกับฐานข้อมูลหลัก ในที่นี้ใช้เป็นเครื่องคอมพิวเตอร์ Notebook



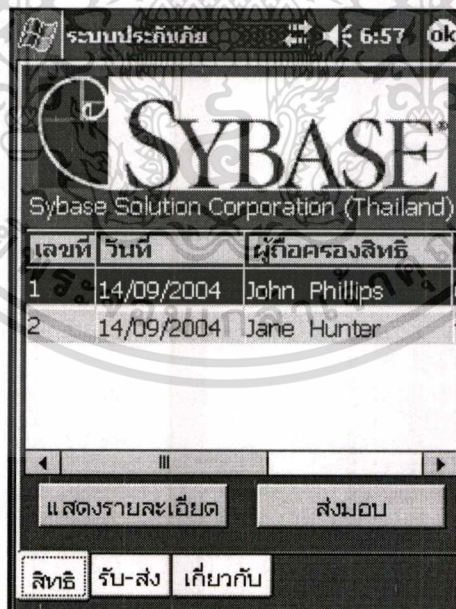
รูปที่ 5.1 หน้าต่างแสดงการทำงานของ MobiLink server

2. เชื่อมต่อ Pocket PC กับ Notebook ผ่าน cradle และ โปรแกรม ActiveSync
3. เปิดโปรแกรมประยุกต์ที่อยู่ใน Pocket PC
4. ลงทะเบียนการใช้งาน



รูปที่ 5.2 การลงทะเบียนเพื่อใช้งานโปรแกรม

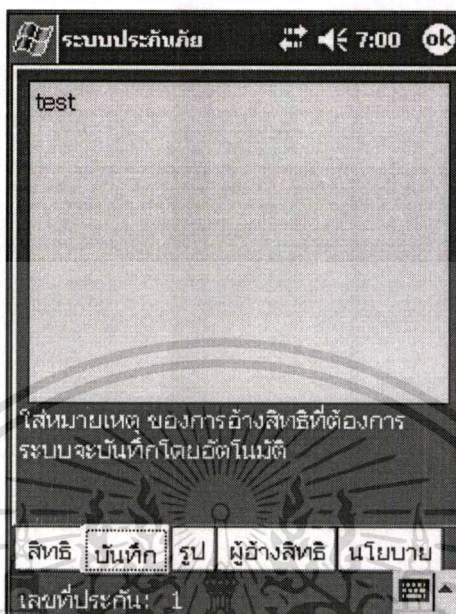
5. Synchronize ข้อมูลจากฐานข้อมูลหลักมาสู่ฐานข้อมูลใน Pocket PC ถ้าการ synchronize ข้อมูลทำสำเร็จจะได้รายการข้อมูลแสดงดังรูปที่ 5.3



รูปที่ 5.3 แสดงข้อมูลที่ได้มาจากการ synchronize

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. เพิ่มรายการต่างๆลงฐานข้อมูล



รูปที่ 5.4 ทดสอบการเพิ่มข้อมูล

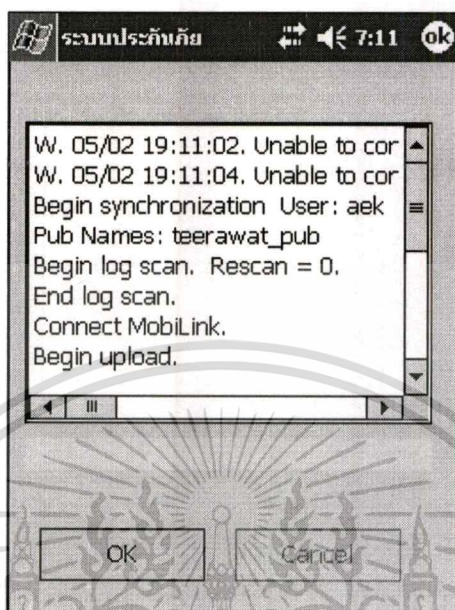
7. เลือกรูปและทำการขีดเขียนลายเส้นบนรูปด้วยปากกาสีต่างๆ



รูปที่ 5.5 ทดสอบการเลือกรูปและขีดลายเส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. Synchronize ข้อมูลจากฐานข้อมูลใน Pocket PC ไปยังฐานข้อมูลหลัก



รูปที่ 5.5 หน้าจอการ Synchronize ข้อมูลจาก Pocket PC ไปยังฐานข้อมูลหลัก

5.3 ผลการทดลอง

จากการทดลองตามขั้นตอนที่ได้ระบุไว้ก่อนหน้านี้ โปรแกรมสามารถทำงานได้ถูกต้อง

5.4 สรุปผลการทดลอง

1. สามารถ start ให้ MobiLink server ขึ้นมาทำงานได้
2. โปรแกรมประยุกต์ที่พัฒนาขึ้นสามารถทำงานได้บน Pocket PC
3. โปรแกรมสามารถทำงานในขั้นตอนลงทะเบียนได้ถูกต้อง
4. โปรแกรมสามารถ synchronize ข้อมูลจากฐานข้อมูลหลักมาสู่ฐานข้อมูลใน Pocket PC ได้
5. โปรแกรมสามารถบันทึกข้อมูลลงสู่ฐานข้อมูลในเครื่อง Pocket PC ได้
6. โปรแกรมสามารถทำงานในขั้นตอนการเลือกรูปและทำการขีดเขียนลายเส้นบนรูปด้วยปากกาสีต่างๆได้
7. โปรแกรมสามารถ synchronize ข้อมูลจากฐานข้อมูลใน Pocket PC ไปยังฐานข้อมูลหลักได้

5.5 ปัญหาและอุปสรรคที่พบในการพัฒนาโครงการงาน

- แหล่งความรู้ในการศึกษา เพื่อนำมาพัฒนาโปรแกรมค่อนข้างมีน้อย เนื่องจากเทคโนโลยีนี้ยังไม่ค่อยมีการใช้งานอย่างกว้างขวางในประเทศไทย จึงไม่สามารถหาข้อมูลที่เป็นภาษาไทยได้ จึงทำให้การศึกษาวិธีการใช้งานของเทคโนโลยี MobiLink ค่อนข้างช้า
- ข้อจำกัดทางด้าน hardware เนื่องจากอุปกรณ์มีราคาค่อนข้างสูง จึงทำให้ประสิทธิภาพของ Pocket PC ที่ใช้ทดสอบไม่สูงมาก

5.6 สรุป

การพัฒนา Database Application บน Pocket PC ด้วยเทคโนโลยี MobiLink Synchronization ถือได้ว่าประสบความสำเร็จในระดับหนึ่ง โดยสามารถบรรลุวัตถุประสงค์ที่ได้ตั้งใจไว้ จากการพัฒนานี้ทำให้ทราบถึงเทคโนโลยีรุ่นใหม่ที่มีประสิทธิภาพ ที่เหมาะสมกับสภาพความต้องการทางธุรกิจในปัจจุบัน ผู้พัฒนาได้ตระหนักว่าความรู้ในครั้งนี้อาจได้ศึกษาเพิ่มเติมจะเป็นประโยชน์ในการประกอบอาชีพมาก เนื่องจากเป็นเทคโนโลยี ที่น่าจะมีแนวโน้มเจริญเติบโตมากขึ้นเพื่อตอบสนองความต้องการทางธุรกิจในรูปแบบต่างๆ

บรรณานุกรม

สยาม สงวนรัมย์. 2547. “คัมภีร์ Pocket PC Super Guide ฉบับสมบูรณ์”. กรุงเทพฯ. ซีเอ็ดยูเคชั่น.

Terry Stepien.2004. “Adaptive Server Anywhere”. [online]. Available:

<http://sybooks.sybase.com/onlinebooks/>

Thomas Volk.2004. “Pocket PowerBuilder”. [online]. Available:

<http://www.isug.com/ISUG3/Index.html>



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

นายธีระวัฒน์ ทองอนันต์ เกิดเมื่อวันที่ 8 พฤศจิกายน พ.ศ. 2522 ที่จังหวัดกระบี่ จบการศึกษาระดับมัธยมศึกษาตอนต้นจากโรงเรียนอามาศย์พานิชกุล จังหวัดกระบี่ จบการศึกษาระดับมัธยมศึกษาตอนปลายจากโรงเรียนหอวัง กรุงเทพฯ และจบการศึกษาในระดับปริญญาตรีจากสาขาวิทยาการคอมพิวเตอร์และสารสนเทศ คณะวิทยาศาสตร์ประยุกต์ สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ ปัจจุบันทำงานในตำแหน่งโปรแกรมเมอร์ บริษัท Sybase Solution Corporations (Thailand)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้