



ระบบการจัดการฐานข้อมูล ภายใน บริษัทศิริวานิช (เอส แอนด์ ดับเบิลยู) จำกัด

DATABASE MANAGEMENT SYSTEM

WITHIN THE SIRIVANICH (S & W) CO.,LTD



นายกิตติศักดิ์	บำรุงเอี่ยมปัญญา	รหัส 44362531
นางสาวอมรา	สองสมุทร	รหัส 44362846
นางสาวโรจน์	ชุมมงคล	รหัส 44362895

ห้องสมุดคณะวิศวกรรมศาสตร์	
วันที่รับ.....	/ 7 / ๒๕๕3
เลขทะเบียน.....	14941615 e.e
เลขเรียกหนังสือ.....	น.ร.
มหาวิทยาลัยนเรศวร ๗๖๗๕ ๕	

ปฏิญานี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2547



ใบรับรองโครงการวิศวกรรม

หัวข้อโครงการ ระบบการจัดการฐานข้อมูลภายในบริษัทศิริวานิช (เอส แอนด์ ดับเบิลยู) จำกัด

ผู้ดำเนินโครงการ นาย กิตติศักดิ์ บำรุงเยี่ยมปัญญา รหัส 44362531
นางสาว อมรา สองสมุทร รหัส 44362846
นางสาว โรจน์ ชูมมงคล รหัส 44362895

อาจารย์ที่ปรึกษา ดร.สุชาติ เข้มมนั

สาขาวิชา วิศวกรรมคอมพิวเตอร์

ภาควิชา วิศวกรรมไฟฟ้าและคอมพิวเตอร์

ปีการศึกษา 2547

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้โครงการฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะกรรมการสอบโครงการวิศวกรรม

.....ประธานกรรมการ
(ดร.สุชาติ เข้มมนั)

.....กรรมการ
(ดร.ไพศาล มณีสว่าง)

.....กรรมการ
(อาจารย์ศิริพร เฉชะสีลาภิรักษ์)

หัวข้อโครงการ	ระบบการจัดการฐานข้อมูลภายในบริษัทศิริวานิช (เอส แอนด์ ดับเบิ้ลยู) จำกัด		
ผู้ดำเนินโครงการ	นาย กิตติศักดิ์ บำรุงเอี่ยมปัญญา	รหัส	44362531
	นางสาว อมรา สองสมุทร	รหัส	44362846
	นางสาวโรจน์ ชุมมงคล	รหัส	44362895
อาจารย์ที่ปรึกษา	ดร.สุชาติ เข้มมน		
สาขาวิชา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2547		

บทคัดย่อ

จากการศึกษาปัญหาที่เกิดขึ้นภายในบริษัทศิริวานิช (เอส แอนด์ ดับเบิ้ลยู) จำกัด พบว่า ปัญหาที่เกิดขึ้นหลักๆ มี 2 ประการ กล่าวคือ ปัญหาด้านการจัดการด้านบุคคล และอีกประการหนึ่ง คือปัญหาการจัดการด้านสินค้าคงคลัง โดยทางบริษัท มีความต้องการโปรแกรม ที่สามารถจัดการเกี่ยวกับปัญหาที่เกิดขึ้นได้ ดังนั้นคณะผู้จัดทำจึงได้ดำเนินการจัดทำโปรแกรมที่แก้ปัญหาดังกล่าว โดยใช้ Visual Basic 6.0 เป็นภาษาที่ใช้ในการพัฒนาโปรแกรม ร่วมกับการใช้ MySQL ในการเก็บบันทึกข้อมูล การทำงานหลักๆของตัวโปรแกรมสามารถทำงานด้าน การเพิ่ม การค้นหา การแก้ไข การคำนวณเงินเดือน และการพิมพ์รายงานสรุปผลต่างๆ ที่ตรงตามวัตถุประสงค์ของผู้ประกอบการ อันจะส่งผลให้การดำเนินกิจการเป็น ไปอย่างมีประสิทธิภาพ

Project Title Database Management System within the Sirivanich (S & W) CO., LTD
Name Mr. Kittisak Bamrungeiampanya ID. 44362531
Miss Ammara Songsamut ID. 44362846
Miss Rojanee Khummongkol ID. 44362895
Project Advisor Dr. Suchart Yammen
Major Computer Engineering
Department Electrical and Computer Engineering
Academic Year 2004

ABSTRACT

From learning about the problems at the Sirivanich (S & W) CO.,LTD, we found two main problems; that are, personal management and treasury management problems. In order to solve these problems, we develop a software application using Visual Basic 6.0 as a programing language and MySQL as a database management system. The software is able to add, query, edit, calculate salary as well as report and summary all information following the company's objective.

กิตติกรรมประกาศ

โครงการวิศวกรรม ระบบการจัดการฐานข้อมูลภายใน บริษัท สิริวานิช (เอส แอนด์ ดับเบิ้ลยู) จำกัด จะไม่สามารถสำเร็จได้ถ้าไม่ได้รับความช่วยเหลือจาก บุคคลจากหลายๆฝ่าย ซึ่งประกอบไปด้วยอาจารย์ สุชาติ แย้มเม่น อาจารย์ที่ปรึกษา ซึ่งคอยให้คำปรึกษาที่ดีเพื่อการพัฒนา โปรแกรมให้มีประสิทธิภาพที่ดีขึ้น คุณ สิริ วานสุวานิช ผู้จัดการและเจ้าของโรงงานสิริวานิช ที่อุปถัมภ์และอนุญาตให้คณะผู้จัดทำโครงการได้ใช้มีโอกาสร่างซอฟต์แวร์สำหรับโรงงาน รวมถึงการรวบรวมข้อมูลต่างๆ โดยให้ความเป็นกันเองและให้ความร่วมมือเป็นอย่างดีซึ่งรวมไปถึงพนักงานทุกท่านที่ให้ความเป็นมิตร ในการพาเยี่ยมชมและให้ข้อมูลที่เป็นประโยชน์ต่อการดำเนินงาน นอกจากนี้ยังมีกำลังใจและคำปรึกษาที่ได้รับจาก ครอบครัว และเพื่อนวิศวกรรมคอมพิวเตอร์ ซึ่งสิ่งต่างๆนี้คอยเป็นแรงผลักดันให้คณะผู้จัดทำมีกำลังใจในการทำให้โครงการผ่านไปได้ด้วยดี

สุดท้ายนี้ต้องขอขอบคุณต่อผู้ที่มีส่วนช่วยเหลือคณะผู้จัดทำในทุกๆด้าน ซึ่งรวมไปถึงบุคคลที่ไม่ได้กล่าวถึงมา ณ ที่นี้ด้วย

นาย กิตติศักดิ์ บำรุงเอี่ยมปัญญา
นางสาว อมรา สองสมุทร
นางสาว โรจน์ ชุมมงคล

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญตาราง	ช
สารบัญรูป	ซ
บทที่ 1 บทนำ	
1.1 ความเป็นมาและความสำคัญของโครงการ	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ขอบเขตของโครงการ	1
1.4 ขั้นตอนการดำเนินงาน	2
1.5 แผนการดำเนินงาน	3
1.6 ผลที่คาดว่าจะได้รับ	4
1.7งบประมาณที่ใช้ในการดำเนินงาน	4
บทที่ 2 หลักการและทฤษฎี	
2.1 ระบบฐานข้อมูล	5
2.1.1 ระบบจัดการฐานข้อมูล	5
2.1.2 หน้าที่ของ DBMS	7
2.1.3 Data Dictionary และ File Manager	7
2.1.4 Relation	7
2.1.5 Domain	8
2.1.6 คุณสมบัติของ Relation	9
2.1.7 ประเภทของ Relation	9
2.1.8 Catalog ของ Relational Model	10
2.2 VISUAL BASIC.....	12
2.2.1 ความรู้ทั่วไปเกี่ยวกับ Relation Database	12

สารบัญ(ต่อ)

2.2.2 ประเภทของความสัมพันธ์ระหว่าง Table	13
2.2.3 Data Control	13
2.2.4 Bound Control	15
2.2.5 Method ที่ใช้ในการจัดการข้อมูล	15
2.2.6 DBCombo และ DBList	20
2.2.7 DBGrid	20
2.3 ภาษาSQL	21
2.3.1 การเลือกข้อมูล	21
2.3.2 การสร้าง Table	23
2.3.3 การกำหนด key	24
2.3.4 การแก้ไข Table	26
2.3.5 การป้อนข้อมูล	27
2.3.6 การปรับปรุงข้อมูล	28
2.3.7 การลบข้อมูล	29
2.3.8 การลบ Table	30
บทที่ 3 วิธีการดำเนินงานและผลการดำเนินการ	
3.1 Data Flow Diagram	31
3.1.1 Context Diagram	31
3.1.2 Data Flow Diagram Level 0	32
3.1.3 Data Flow Diagram Level 1.....	32
3.2 Data Dictionary	35
3.2.1 Data Dictionary ของ Entity	35
บทที่ 4 การพัฒนาโปรแกรม	
4.1 แผนภาพการพัฒนาโปรแกรม	42
4.1.1 การลงทะเบียนเข้าใช้โปรแกรม	42
4.1.2 โปรแกรมในส่วนของผู้บริหารกิจการ	43

สารบัญ(ต่อ)

4.1.3 โปรแกรมส่วนพนักงาน	43
4.1.4 โปรแกรมในส่วนของการจัดการวัตถุดิบ	44
4.1.5 โปรแกรมในส่วนการเงิน	45
4.1.6 โปรแกรมในส่วนข้อมูลพนักงาน	45
4.1.7 โปรแกรมด้านคลังสินค้า	46
4.2 รูปแบบโปรแกรม	46
4.2.1 ระบบบริหารฝ่ายบุคคล	46
4.2.2 โปรแกรมระบบฝ่ายการเงินเดือน	54
4.2.3 ระบบสินค้าคงคลัง	59
บทที่ 5 บทสรุป	
5.1 บทสรุป	63
5.2 ข้อเสนอแนะ	63
เอกสารอ้างอิง	64
ประวัติผู้เขียน โครงการงาน	65

สารบัญตาราง

ตารางที่	หน้า
1.1 ตารางการดำเนินงานของโครงการ	3
2.1 TABLES	11
2.2 COLUMNS	11
2.3 การกำหนด Property ของ CommandButton	15
2.4 การกำหนดคุณสมบัติของ Object	17
2.5 การกำหนด Property ให้กับObject โดยเพิ่ม Method Delete	18
2.6 ฐานข้อมูล ตัวอย่างของEmpinfo	22



สารบัญรูป

รูปที่	หน้า
2.1 การกำหนด Property ของ CommandButton	16
2.2 โปรแกรมเมื่อใส่ Update Button ลงไป	17
2.3 โปรแกรมหลังจากใส่ button Delete เข้าไป	19
3.1 Context Diagram	31
3.2 Data Flow Diagram ของระบบLevel 0	32
3.3 Data Flow Diagram ของระบบ ใน Level1 Process 1 การรับวัตถุดิบ	33
3.4 Level 1 Process 2 การคิดเงินเดือนพนักงาน	33
3.5 Level 1 Process 3 การผลิต	34
3.6 Level 1 Process 5 คลังสินค้า.....	34
4.1 แผนภาพโปรแกรมลงทะเบียนผู้ใช้โปรแกรม	42
4.2 แผนภาพโปรแกรมในส่วนของผู้บริหาร	43
4.3 แผนภาพโปรแกรมในส่วนพนักงาน	44
4.4 แผนภาพโปรแกรมการจัดการวัตถุดิบ	44
4.5 แผนภาพโปรแกรมการจัดการด้านการเงิน	45
4.6 แผนภาพโปรแกรมข้อมูลพนักงาน	45
4.7 แผนภาพโปรแกรมคงคลังสินค้า	46
4.8 โปรแกรมลงทะเบียนใช้งาน.....	47
4.9 หน้าหลักของโปรแกรม	48
4.10 โปรแกรมบันทึกเวลาเข้าทำงานของพนักงาน	49
4.11 โปรแกรมบันทึกเวลาออกทำงานของพนักงาน	49
4.12 การเพิ่มข้อมูลพนักงาน.....	50
4.13 การแก้ไขข้อมูลของพนักงาน	50
4.14 โปรแกรมการเพิ่มผู้ใช้ในระบบ	51
4.15 โปรแกรมการเพิ่มข้อมูลแผนก.....	51
4.16 โปรแกรมการแก้ไขข้อมูลของแผนก.....	52
4.17 โปรแกรมการเพิ่มตำแหน่งของพนักงาน	52

สารบัญรูป(ต่อ)

4.18	โปรแกรมการแก้ไขข้อมูลของตำแหน่ง	53
4.19	การรายงานประวัติพนักงาน	53
4.20	หน้าต่างหลักของโปรแกรมระบบฝ่ายการคิดเงินเดือน	54
4.21	แสดงการคำนวณเงินค่าแรงสำหรับพนักงานรายวัน	54
4.22	แสดงการคำนวณเงินค่าแรงสำหรับพนักงานรายเดือน	55
4.23	แสดงการคำนวณเงินค่าแรงสุทธิประจำเดือน	56
4.24	หน้าต่างแสดงโปรแกรมพิมพ์รายชื่อพนักงานที่มาทำงานสาย	57
4.25	หน้าต่างแสดงโปรแกรมพิมพ์รายชื่อพนักงานที่ขาดงาน	57
4.26	หน้าต่างแสดงโปรแกรมพิมพ์รายได้ของพนักงานแต่ละแผนก	58
4.27	หน้าต่างการกำหนดอัตราเงินสำหรับหักค่ามาสาย	58
4.28	แสดงหน้าต่างโปรแกรมหลักของโปรแกรมในส่วนคงคลังสินค้า	59
4.29	แสดงส่วนของการเบิกสินค้าไปขาย	60
4.30	แสดงส่วนของการเบิกวัตถุดิบ	60
4.31	แสดงโปรแกรมการซื้อวัตถุดิบ	61
4.32	แสดงโปรแกรมการส่งสินค้า	62
4.33	รายงานการเบิกสินค้าไปขาย	62

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของโครงการ

ประมาณต้นเดือนมกราคม ปี พ.ศ. 2546 อาจารย์ที่ปรึกษาพร้อมกลุ่มนิสิตจำนวน 15 ท่าน ได้เดินทางไปยังโรงงานของบริษัท สิริวานิช (เอส แอนด์ คับเบิ้ลยู) จำกัด ซึ่งตั้งอยู่ที่ อำเภอเมือง จังหวัดพิษณุโลก โรงงานขนาดกลางนี้ผลิตข้าวโพดอบกรอบหลายชนิดหลายยี่ห้อซึ่งผลิตภัณฑ์ของโรงงานมีรสชาติที่อร่อยและถูกหลักอนามัยผ่านกรรมวิธีการผลิตที่มีเครื่องจักร และ การบรรจุหีบห่อที่ดีไปเสนอขายตามร้านค้าขายปลีกต่างๆทั่วภาคเหนือ

ในปัจจุบัน ผู้ประกอบการ ประสบปัญหาทางด้านการจัดการภายในโรงงานอันเนื่องมาจาก วัตถุดิบสูญหายไปโดยไม่ทราบสาเหตุ และ พนักงานทำงานแบ่งเป็นหลายช่วงเวลาทำให้มีการ บันทึกการทำงานจริงๆ ได้ไม่ตรงกับข้อมูลจริงซึ่งส่งผลไปถึงการคิดเงินเดือนของพนักงาน และยัง มีปัญหาเรื่องสินค้าที่อยู่ในคลังสินค้าที่การจดบันทึกและจำนวนสินค้าที่มีอยู่จริงนั้น ไม่ตรงกัน การ สูญหายของสินค้าส่วนใหญ่มาจากความผิดพลาดของผู้จดบันทึกลงในกระดาษงานเท่านั้นซึ่งมีผล ทำให้ต้นทุนการผลิตเพิ่มขึ้นด้วย ทางบริษัทจึงต้องการให้กลุ่มนิสิตจัดทำซอฟต์แวร์ที่จัดการเรื่อง ระบบฐานข้อมูลต่างๆ ภายในโรงงานเพื่อจัดเก็บลงในคอมพิวเตอร์ โดยที่นำซอฟต์แวร์นี้มาใช้ จัดการบริหารทั้งเรื่องวัตถุดิบ และค่าจ้างของพนักงานตลอดจนสินค้าหรือวัตถุดิบที่อยู่ในคลังสินค้า ได้อย่างมีประสิทธิภาพและประสิทธิผล

1.2 วัตถุประสงค์ของโครงการวิศวกรรม

- 1.2.1 เพื่อจัดทำระบบการจัดการสารสนเทศภายในโรงงาน
- 1.2.2 เพื่อพัฒนาระบบสารสนเทศภายในโรงงาน
- 1.2.3 เพื่อเพิ่มประสิทธิภาพการจัดการภายในโรงงานให้มีประสิทธิภาพมากยิ่งขึ้น

1.3 ขอบเขตของโครงการวิศวกรรม

จัดทำและพัฒนาระบบการจัดการสารสนเทศภายในโรงงานอันประกอบไปด้วยการควบคุม สินค้าคงคลัง การคิดเงินเดือนให้กับพนักงาน และ การควบคุมวัตถุดิบเพิ่มประสิทธิภาพการจัดการ ภายในโรงงานให้มีประสิทธิภาพมากยิ่งขึ้น

1.4 ขั้นตอนการดำเนินงานโครงการวิศวกรรม

ในการจัดทำโครงการนี้ได้มีขั้นตอนในการดำเนินการดังต่อไปนี้

- 1.4.1 เก็บรวบรวม ข้อมูลลูกจ้างบัญชี รายรับ – รายจ่าย ข้อมูลลูกค้า ข้อมูลการรับส่งผลิตภัณฑ์ และข้อมูลวัตถุดิบ
- 1.4.2 ศึกษากระบวนการควบคุมสินค้าคงคลังของโรงงาน
- 1.4.3 ศึกษาระบบสารสนเทศภายในโรงงาน
- 1.4.4 ออกแบบระบบฐานข้อมูลให้สอดคล้องกับระบบฐานข้อมูลที่ใช้
- 1.4.5 พัฒนาโปรแกรมระบบฐานข้อมูล ระบบควบคุมสินค้าคงคลัง และระบบสารสนเทศ
- 1.4.6 ทดสอบการใช้งานและตรวจสอบแก้ไขข้อผิดพลาดของโปรแกรม



1.5 แผนการดำเนินงาน

ตารางที่ 1.1 ตารางการดำเนินงานของโครงการ

กิจกรรม	ระยะเวลาดำเนินงาน (เดือน)												ผู้รับผิดชอบ/ปฏิบัติ	หมายเหตุ	
	1	2	3	4	5	6	7	8	9	10	11	12			
1. เก็บรวบรวมข้อมูลผู้จ้างบัญชี รายรับ-รายจ่าย ข้อมูลลูกค้า ข้อมูลการรับส่ง ผลิตภัณฑ์ และข้อมูลวัตถุดิบ	■													นายกิตติศักดิ์ นารุงเยี่ยมปัญญา นางสาวอมรา สองสมุทร นางสาวโรจน์ ขุนมงคล	2
2. ศึกษากระบวนการควบคุมสินค้าคงคลังของโรงงาน		■													1
3. ศึกษากระบวนการสารสนเทศภายในโรงงาน						■									2
4. ออกแบบระบบฐานข้อมูลให้สอดคล้องกับระบบฐานข้อมูลที่ใช้							■								1
5. พัฒนาโปรแกรมระบบฐานข้อมูล ระบบควบคุมสินค้าคงคลัง และระบบสารสนเทศ								■							2
6. ทดสอบการใช้งานและตรวจสอบแก้ไขข้อผิดพลาดของโปรแกรม											■				1

หมายเหตุ 1.ศึกษาที่ภาควิชา 2.ศึกษาที่โรงงาน

1.6 ผลที่คาดว่าจะได้รับ

1. ได้รับ โปรแกรมที่สามารถจัดการระบบการจัดการฐานข้อมูล ภายใน บริษัท ศิริวานิช (เอส แอนด์ ดับเบิ้ลยู) จำกัด
2. คณะผู้จัดทำได้รับประสบการณ์การเขียนโปรแกรมและออกแบบระบบฐานข้อมูล

1.7 งบประมาณที่ใช้

ค่าหนังสือใช้ประกอบการทำโครงการ	2,000 บาท
ค่าถ่ายเอกสาร	1,000 บาท
รวมเป็นเงิน	3,000 บาท

หมายเหตุ ทุกรายการสามารถดัดแปลงค่าใช้จ่ายกันได้



บทที่ 2

หลักการและทฤษฎีที่เกี่ยวข้อง

การจัดระบบสารสนเทศสำหรับโครงการ อาศัยหลักการ และ ทฤษฎีที่นำมาทำการจัดการระบบฐานข้อมูล โดยหลักการเขียนโปรแกรมการจัดการฐานข้อมูลของโครงการนั้น จะต้องมีพื้นฐานการใช้ภาษา SQL ซึ่งเป็นภาษาที่ใช้ในการจัดการฐานข้อมูล และรวมทั้ง ส่วนของโปรแกรม Visual Basic 6.0 ที่ใช้สำหรับติดต่อกับฐานข้อมูล นั้น มีหลักการและทฤษฎีที่เกี่ยวข้องดังต่อไปนี้

2.1 ระบบฐานข้อมูล(Database System)

จากปัญหาต่าง ๆ ที่เกิดขึ้นในระบบแฟ้มข้อมูล ได้ก่อให้เกิดการจัดเก็บข้อมูลในรูปแบบใหม่ขึ้น ที่เรียกว่า ฐานข้อมูล (Database) การจัดเก็บข้อมูลในฐานข้อมูลนี้จะแตกต่างจากการเก็บข้อมูลแบบแฟ้มข้อมูล เนื่องจากฐานข้อมูลเป็นการนำเอาข้อมูลต่างๆ ที่มีความสัมพันธ์กัน ซึ่งแต่เดิมจัดเก็บอยู่ในแต่ละแฟ้มข้อมูลมาจัดเก็บไว้ในที่เดียวกัน เช่น ข้อมูลพนักงานสินค้าคงคลัง พนักงานขาย และลูกค้า ซึ่งแต่เดิมถูกจัดเก็บอยู่ในรูปของแฟ้มข้อมูลของฝ่ายต่างๆ ได้ถูกนำมาจัดเก็บรวมกันไว้ภายในฐานข้อมูลเดียว ซึ่งเป็นฐานข้อมูลรวมของบริษัท ส่งผลให้แต่ละฝ่ายสามารถใช้ข้อมูลร่วมกัน และสามารถแก้ไขปัญหาต่างๆ ที่เกิดขึ้นในระบบแฟ้มข้อมูลได้

ข้อมูลต่าง ๆ ที่ถูกจัดเก็บเป็นฐานข้อมูล นอกจากจะต้องเป็นข้อมูลที่มีความสัมพันธ์กันแล้ว ยังจะต้องเป็นข้อมูลที่ใช้สนับสนุนการดำเนินงานอย่างน้อยอย่างใดอย่างหนึ่งขององค์กร ดังนั้นจึงอาจกล่าวได้ว่าแต่ละฐานข้อมูลจะเทียบเท่ากับระบบแฟ้มข้อมูล 1 ระบบ และจะเรียกฐานข้อมูลที่จัดทำขึ้นเพื่อสนับสนุนการดำเนินงานอย่างใดอย่างหนึ่งนั้นว่า “ระบบฐานข้อมูล” (Database System) เช่น ระบบฐานข้อมูลเงินเดือน ซึ่งเป็นฐานข้อมูลที่จัดเก็บข้อมูลต่างๆ ที่สนับสนุนการคำนวณเงินเดือน หรือระบบฐานข้อมูลประชากร ซึ่งเป็นฐานข้อมูลที่จัดเก็บข้อมูลต่าง ๆ ที่สนับสนุนการจัดทำสำมะโนประชากร เป็นต้น

2.1.1 ระบบจัดการฐานข้อมูล (Database Management System :DBMS)

เป็นโปรแกรมที่ทำหน้าที่เป็นตัวกลางในการติดต่อระหว่างผู้ใช้กับฐานข้อมูลเพื่อจัดการและควบคุมความถูกต้องความซ้ำซ้อน และความสัมพันธ์ระหว่างข้อมูลต่าง ๆ ภายในฐานข้อมูล ซึ่งต่างจากระบบแฟ้มข้อมูลที่หน้าที่เหล่านี้จะเป็นหน้าที่ของโปรแกรมเมอร์ ในการติดต่อกับข้อมูลในฐานข้อมูลไม่ว่าจะด้วยการใช้คำสั่งในกลุ่มคำสั่ง DML หรือ DDL หรือจะด้วยโปรแกรมต่าง ๆ ทุกคำสั่งที่ใช้กระทำกับข้อมูลจะถูกโปรแกรม DBMS นำมาแปล (Compile) เป็นการกระทำ

(Operation) ต่างๆ ภายใต้คำสั่งนั้นๆ เพื่อนำไปกระทำกับตัวข้อมูลในฐานข้อมูลต่อไปสำหรับส่วนการทำงานต่างๆ ภายในโปรแกรม DBMS ที่ทำหน้าที่ในการแปลคำสั่งไปเป็นการกระทำต่างๆที่จะกระทำกับตัวข้อมูลนั้น ประกอบด้วยส่วนการกระทำต่างๆ ดังนี้

- Database Manager เป็นส่วนที่ทำหน้าที่กำหนดการกระทำต่างๆ ให้กับส่วน File Manager เพื่อไปกระทำกับข้อมูลที่เก็บอยู่ในฐานข้อมูล (File Manager เป็นส่วนที่ทำหน้าที่บริหาร และจัดการข้อมูลที่เก็บอยู่ในฐานข้อมูลในระดับกายภาพ)
- Query Processor เป็นส่วนที่ทำหน้าที่แปลประโยคคำสั่งของ Query Language ให้อยู่ในรูปแบบของคำสั่งที่ Database Manager เข้าใจ
- Data Manipulation Language Precompiler เป็นส่วนที่ทำหน้าที่แปล (Compile) ประโยคคำสั่งของกลุ่มคำสั่ง DML ให้อยู่ในรูปแบบที่ส่วน Application Programs Object Code จะนำไปเข้ารหัสเพื่อส่งต่อไปยังส่วน Database Manager ในการแปลประโยคคำสั่งของกลุ่มคำสั่ง DML ของส่วน Data Manipulation Language Precompiler นี้จะต้องทำงานร่วมกับส่วน Query Processor
- Data Definition language Precompiler เป็นส่วนที่ทำหน้าที่แปล (Compile) ประโยคคำสั่งของกลุ่มคำสั่ง DDL ให้อยู่ในรูปแบบของ MetaData ที่เก็บอยู่ในส่วน Data Dictionary ของฐานข้อมูล (MetaData ได้แก่รายละเอียดที่บอกถึงโครงสร้างต่าง ๆ ของข้อมูล)
- Application Programs Object Code เป็นส่วนที่ทำหน้าที่แปลงคำสั่งต่าง ๆ ของโปรแกรม รวมทั้งคำสั่งในกลุ่มของคำสั่ง DML ที่ส่งต่อมาจากส่วน Data Definition language Precompiler ให้อยู่ในรูปแบบของ Object Code ที่จะส่งต่อไปให้ Database Manager เพื่อกระทำกับข้อมูลกับฐานข้อมูล

โปรแกรม DBMS นี้ได้ถูกพัฒนาขึ้นมาเพื่อแก้ปัญหาทางด้าน Data Independence ที่ไม่มีในระบบแฟ้มข้อมูลดังนั้นจึงมีความอิสระจากทั้งตัว Hardware และตัวข้อมูลภายในฐานข้อมูล กล่าวคือ โปรแกรม DBMS จะมีการทำงานที่ไม่ขึ้นกับรูปแบบ (Platform) ของตัว Hardware ที่นำมาใช้กับระบบฐานข้อมูล รวมทั้งมีรูปแบบในการอ้างถึงข้อมูลที่ไม่ขึ้นอยู่กับโครงสร้างทางกายภาพของข้อมูล ด้วยการ ใช้ Query Language ในการติดต่อกับข้อมูลในฐานข้อมูลแทนคำสั่งของภาษาคอมพิวเตอร์ในยุคที่ 3 ส่งผลให้ผู้ใช้สามารถเรียกใช้ข้อมูลจากฐานข้อมูลโดยไม่จำเป็นต้องทราบประเภทของฐานข้อมูลนั้น หรือสามารถกำหนดลำดับที่ของ Filed ในการแสดงผลได้โดยไม่ต้องคำนึงถึงลำดับที่จริงของ Filed นั้น

2.1.2 หน้าที่ของ DBMS

สำหรับหน้าที่ของโปรแกรม DBMS มีดังนี้

- แปลงคำสั่ง ที่ใช้ในการจัดการกับข้อมูลภายในฐานข้อมูล ให้อยู่ใน รูปแบบที่ฐานข้อมูลเข้าใจ
- ในการนำคำสั่งต่างๆ ซึ่งได้รับการแปลแล้ว ไปส่งให้ฐานข้อมูลทำงาน เช่น การเรียกใช้ข้อมูล (Retrieve) การจัดเก็บข้อมูล (Update) การลบข้อมูล (Delete) การเพิ่มข้อมูล (Add) เป็นต้น
- ป้องกันความเสียหาย ที่จะเกิดขึ้นกับข้อมูลภายในฐานข้อมูล โดยจะคอยตรวจสอบว่าคำสั่งใดที่สามารถทำงานได้และคำสั่งใดที่ไม่สามารถทำงานได้
- รักษาความสัมพันธ์ของข้อมูลภายในฐานข้อมูลให้มีความถูกต้องอยู่เสมอ
- เก็บรายละเอียดต่างๆ ที่เกี่ยวข้องกับข้อมูลภายในฐานข้อมูล ไว้ใน Data Dictionary ซึ่งรายละเอียดเหล่านี้มักจะถูกเรียกว่า "ข้อมูลของฐานข้อมูล" (Metadata)
- ควบคุมให้ฐานข้อมูลทำงานได้อย่างถูกต้องและมีประสิทธิภาพ

2.1.3 Data Dictionary และ File Manager

ทุกฐานข้อมูลจะต้องมีส่วนที่ใช้เก็บข้อมูลในลักษณะ Metadata ซึ่งเป็นข้อมูลที่บอกถึงรายละเอียดของตัวข้อมูลที่เกี่ยวข้องกับข้อมูลเช่น โครงสร้างของข้อมูล โครงสร้างของ Table โครงสร้างของ Index กฎที่ใช้ควบคุมความถูกต้องของข้อมูล (Integrity Rule) กฎที่ใช้ในการรักษาความปลอดภัยให้กับข้อมูล (Security Rule) เป็นต้น ข้อมูลเหล่านี้จัดเป็นข้อมูลที่มีความจำเป็นต่อโปรแกรม DBMS ในการตัดสินใจที่จะดำเนินการใด ๆ กับฐานข้อมูลเช่น ข้อมูลที่เกี่ยวข้องกับกฎที่ใช้ในการรักษาความปลอดภัยให้กับข้อมูล จะถูกนำมาใช้ในการพิจารณาแก่ผู้ใช้ในการใช้งานฐานข้อมูล เป็นต้น สำหรับส่วนที่ใช้ในการจัดเก็บข้อมูลในลักษณะของ Metadata นี้ได้แก่ Data Dictionary หรือ Catalog

สำหรับ File Manager เป็นส่วนที่ทำหน้าที่บริหารและจัดการกับข้อมูลที่เก็บอยู่ในฐานข้อมูลในระดับกายภาพ

2.1.4 Relation

Relation มักจะถูกเรียกว่า "Table" เนื่องจาก Relation เป็นหน่วยที่ใช้จัดเก็บข้อมูลที่อยู่ในรูปแบบของตารางขนาด 2 มิติ ที่ประกอบด้วยแถว (Row) และสดมภ์ (Column) แถวของ Relation ได้แก่ ข้อมูล 1 รายการ ซึ่งเทียบเท่ากับ Record ในระบบแฟ้มข้อมูล ส่วนแต่ละสดมภ์ของ Relation ได้แก่ คุณลักษณะต่างๆ ของข้อมูลในแต่ละแถวซึ่งเทียบเท่ากับ Field ของ Record ในระบบแฟ้มข้อมูล สำหรับชื่อของแต่ละแถวของ Relation จะถูกเรียกว่า "Tuple" ส่วน

ชื่อของแต่ละสคคภของ Relation จะถูกเรียกว่า "Attribute" สำหรับข้อแตกต่างระหว่าง Relation กับเพิ่มข้อมูลในระบบเพิ่มข้อมูล มีดังนี้

- Relation จะเป็นส่วนที่จัดเก็บข้อมูลในระดับแนวความคิด ดังนั้น ผู้ใช้ในทุกระดับ ไม่ว่าจะ เป็นนักวิเคราะห์ระบบ โปรแกรมเมอร์ หรือผู้ใช้ทั่วไป จะมองข้อมูลที่จัดเก็บอยู่ใน Relation นี้ใน รูปของตารางเช่นเดียวกัน ทำให้เวลานำข้อมูลจาก Relation ไปใช้งาน หรือจัดเก็บข้อมูลลงใน Relation ผู้ที่นำข้อมูลไปใช้ หรือจัดเก็บไม่จำเป็นต้องทราบถึงโครงสร้างทางกายภาพของ Relation นั้น ซึ่งต่างจากเพิ่มข้อมูลในระบบเพิ่มข้อมูล ที่ผู้ใช้ในแต่ละระดับจะมีมุมมองต่อ ข้อมูลเดียวกันต่างกัน ดังที่กล่าวมาแล้วในบทที่ผ่านมา ดังนั้น Relation ใน Relational Model จึงมีความเป็นอิสระจากทั้งส่วนของ Software และ Hardware
- การจัดเก็บ Relation ในหน่วยความจำสำรองจะอยู่ในที่เดียวกัน ซึ่งต่างจากเพิ่มข้อมูลในระบบ เพิ่มข้อมูลแต่ละเพิ่มข้อมูลจะจัดเก็บอยู่อย่างกระจัดกระจาย สำหรับ Relation ที่ข้อมูลมีความสัมพันธ์กัน และนำมาจัดเก็บรวมกันนี้ จะเป็นฐานข้อมูล 1 ฐานข้อมูล

2.1.5 Domain

เป็นการนิยามขอบเขตของค่าที่เป็นไปได้กับข้อมูลในแต่ละ Attribute เพื่อป้องกันไม่ให้เกิดการป้อนข้อมูลที่เกินขอบเขตที่กำหนด เช่น

- ค่าที่นิยมให้กับข้อมูลจะต้องมีค่าเป็น Scalar กล่าวคือ จะต้องเป็นค่าที่มีความหมายในหน่วยที่ เล็กที่สุด ที่ไม่ปรากฏโครงสร้างที่สามารถแยกย่อยออกเป็นโครงสร้างย่อยๆ ได้อีก สำหรับข้อมูล ที่มีค่าเป็น Scalar นี้ จะเรียกข้อมูลนั้นว่ามีคุณลักษณะของ Atomicity
- ข้อมูลที่สามารถนำมากำหนด Domain ได้ จะต้องเป็นข้อมูลที่เป็นอิสระจากข้อมูลอื่น
- ข้อมูลที่สามารถนำมากำหนด Domain ได้ จะต้องเป็นข้อมูลประเภทเดียวกัน
- Domain ที่กำหนดให้กับ Attribute ที่จะต้องถูก Attribute อื่นอ้างอิง สามารถถ่ายทอด Domain ของตนให้กับ Attribute ในอีก Relation หนึ่งที่ยังถึงไปด้วย
- ค่าของ Domain ที่กำหนดให้กับข้อมูลไม่จำเป็นที่จะต้องปรากฏอยู่ในข้อมูลนั้นๆ

ประโยชน์ของการกำหนด Domain ให้กับข้อมูล นอกเหนือจากจะเป็นการกำหนดค่าที่เป็นไปได้ ที่ผู้ใช้สามารถกำหนดให้กับข้อมูลในส่วนนั้นๆ เพื่อป้องกันไม่ให้ผู้ใช้ป้อนข้อมูลเกิน ขอบเขตที่กำหนดไว้แล้ว ยังสามารถสร้างความเชื่อมั่นในการนำข้อมูลที่สัมพันธ์กันมาเปรียบเทียบ กันได้อีกด้วย

2.1.6 คุณสมบัติของ Relation

คุณสมบัติ Tuple และ Attribute ของแต่ละ Relation จะประกอบด้วย

- เนื่องจาก Relation ใน Relational Model อยู่ในรูปแบบของเซตทางคณิตศาสตร์ ที่ภายในเซต จะต้องประกอบด้วยสมาชิกที่มีค่าไม่ซ้ำกัน ดังนั้น ภายใน Relation ใดๆ จึงต้องมี Attribute ใด Attribute หนึ่ง ที่ทำให้แต่ละ Tuple ใน Relation มีข้อมูลที่ไม่ซ้ำกัน
- คิวเหตุผลเช่นเดียวกับข้อที่ 1 ลำดับที่ของสมาชิกภายในเซตใดๆ จะไม่มีผลต่อเซตนั้น ดังนั้น ภายใน Relation จึงไม่มีการกำหนดลำดับที่ให้กับแต่ละ Tuple ใน Relation กล่าวคือ จะไม่มีการกล่าวถึงคำว่า Tuple แรก หรือ Tuple สุดท้าย หรือ Tuple ลำดับที่ 5 หรือ Tuple ถัดไป หรือ Tuple ที่ผ่านมาใน Relation
- ภายใน Relation จะไม่มีกำหนดลำดับที่ให้กับแต่ละ Attribute เนื่องจากในการอ้างถึง Attribute ใน Relation จะใช้ชื่อของ Attribute นั้นในการอ้างถึง ดังนั้น จึงไม่มีการกล่าวถึงคำว่า Attribute แรก หรือ Attribute สุดท้ายหรือ Attribute ลำดับที่ 5 หรือ Attribute ที่ผ่านมา หรือ Attribute ถัดไปใน Relation เช่นเดียวกับ Tuple
- ค่าในทุก Attribute ของ Relation จะต้องมีความสมบัติ Atomicity ซึ่งเป็นคุณสมบัติที่กำหนดให้ค่าข้อมูลในแต่ละ Attribute ของ Relation จะต้องมีความหมายใดความหมายหนึ่งเพียงความหมายเดียว ไม่ใช้กลุ่มของสิ่งใด สิ่งหนึ่ง หรือกล่าวอีกนัยหนึ่ง ข้อมูลในแต่ละ Attribute ของ Relation จะต้องไม่ใช่ข้อมูลในลักษณะ Repeating Group เช่น กรณีที่พนักงานสามารถสังกัดฝ่ายได้มากกว่า 1 ฝ่าย ข้อมูลใน Attribute "DeptID" ของแต่ละ Tuple ของ Relation "EMPLOYEE" ซึ่งใช้เก็บรหัสของฝ่ายที่พนักงานแต่ละคนสังกัด จะไม่สามารถจัดเก็บทุกรหัสของฝ่ายที่พนักงานคนนั้นสังกัดภายใน Tuple
- ชื่อของแต่ละ Attribute ใน Relation เดียวกัน จะต้องไม่ซ้ำกัน
- ค่าที่ปรากฏในแต่ละ Attribute ใน Relation เดียวกัน จะต้องใช้แทนข้อมูลที่มีความหมายเดียวกัน

2.1.7 ประเภทของ Relation

Relation สามารถแบ่งออกเป็นประเภทต่างๆ ได้ดังนี้

- Named Relation เป็น Relation ที่สร้างขึ้นด้วยคำสั่ง SQL ซึ่งอาจเป็น Relation จริงในฐานข้อมูล หรือเป็นเพียง Relation ที่สร้างขึ้นด้วยคำสั่งของ Query Language
- Base Relation เป็น Named Relation ในส่วนที่เป็น Relation จริงในฐานข้อมูล ซึ่งใช้เก็บข้อมูลในหน่วยความจำสำรอง ดังนั้นจึงเป็น Relation จริงที่เกิดขึ้นจากการออกแบบฐานข้อมูล

- Derived Relation เป็น Named Relation ในส่วนของ Relation ซึ่งได้มาจากการใช้เงื่อนไข ประกอบกับคำสั่งของ Query Language กับ Base Relation
- Expressible Relation เป็น Relation ที่ได้มาจากการกระทำกับ Named Relation ด้วยเงื่อนไข ทางด้านความสัมพันธ์ของข้อมูลระหว่าง Named Relation ที่ต้องการ ดังนั้นเมื่อนำเอาทุก Expressible Relation มาประกอบกัน ผลลัพธ์ที่ได้ จึงได้แก่ ทุก ๆ Base Relation และ Derived Relation ที่ Expressible Relation นั้นใช้สร้างขึ้น
- View เป็น Derived Relation ประเภทหนึ่ง แต่จะเป็น Relation เสมือน (Virtual Relation) ที่ถูกสร้างขึ้นไว้ในฐานข้อมูล
- Snapshot เป็น Derived Relation ประเภทหนึ่งเช่นเดียวกับ View แต่จะต่างกันว่า Snapshot เป็น Relation ในฐานข้อมูล ที่สามารถอ่านข้อมูลได้เพียงอย่างเดียว และสามารถกำหนดเวลาในการปรับปรุงค่าของข้อมูลใน Snapshot ได้ เช่น ทุกวัน ทุกสัปดาห์ เป็นต้น
- Query Result เป็น Relation ชั่วคราว ซึ่งเกิดจากการใช้ประโยคคำสั่งของ Query Language กับ Relation ในฐานข้อมูลข้อมูลใน Relation ประเภทนี้ จะเกิดขึ้นก็ต่อเมื่อมีการเรียกใช้ เท่านั้น และหายไปเมื่อเลิกใช้งาน
- Immediate Result เป็น Relation ชั่วคราวที่เกิดขึ้นในขณะที่ทำการประมวลผลประโยคคำสั่งของ Query Language ที่มีความซับซ้อน เช่น คำสั่งต่อไปนี้ ในการประมวลผล จะเริ่มจาก คำสั่งในวงเล็บก่อน ซึ่งได้แก่ คำสั่ง "S JOIN SP" จากคำสั่งนี้ จะก่อให้เกิด Relation ชั่วคราวที่ใช้เก็บข้อมูลที่ได้จากการรวมกันของ "S" และ "SP" ก่อนที่จะนำไปใช้ในขั้นตอนต่อไป
- Stored Relation เป็น Expressible Relation ที่สามารถจัดเก็บค่าของข้อมูลได้ ดังนั้น ในบางครั้งจึงอาจกล่าวว่า Relation ประเภทนี้เป็น Base Relation Degree ได้แก่ จำนวนของ Attribute หรือกล่าวอีกนัยหนึ่ง คือ จำนวน Domain ในแต่ละ Relation Cardinality ได้แก่ จำนวนของ Tuple ใน Relation หนึ่งที่มีความสัมพันธ์กับ Tuple ในอีกอันหนึ่ง ซึ่งจะนำไปใช้กำหนดประเภทของความสัมพันธ์ระหว่างข้อมูลใน Relation ทั้งสอง

2.1.8 Catalog ของ Relational Model

Catalog ของฐานข้อมูลที่มีโครงสร้างข้อมูลแบบ Relational จะมีลักษณะเป็น Relation เช่นเดียวกับ Relation โดยทั่วไป แต่จะต่างจาก Relation ทั่วไป ในแง่ที่ว่า ข้อมูลที่จัดเก็บอยู่ใน Catalog จะไม่ใช่ข้อมูลโดยทั่วไป แต่จะเป็น Metadata แทน เช่น สมมุติให้ Catalog ประกอบด้วย 2 Relation ที่มีชื่อว่า "TABLES" และ "COLUMNS" เพื่อใช้จัดเก็บ Metadata

ของฐานข้อมูลที่ประกอบด้วย Relation "DEPARTMENT" และ "EMPLOYEE" ข้อมูลที่เก็บอยู่ใน Relation "TABLES" และ "COLUMNS" จะมีลักษณะดังรูป

ตารางที่ 2.1 TABLES

TABNAME	COLCOUNT	ROWCOUNT
DEPARTMENT	3	3
EMPLOYEE	4	4
.....

ตารางที่ 2.2 COLUMNS

TABNAME	COLUMNNAME
DEPARTMENT	DEPT#
DEPARTMENT	DNAME
DEPARTMENT	BUDGET
EMPLOYEE	EMP#
EMPLOYEE	ENAME
EMPLOYEE	DEPT#
EMPLOYEE	SALARY
.....

จากรูป เมื่อผู้ใช้งานต้องการทราบถึงโครงสร้างของ Relation "DEPARTMENT" โปรแกรม DBMS จะเข้ามาค้นหารายชื่อ Attribute ต่าง ๆ ของ Relation "DEPARTMENT" ใน Relation "TABLES" เพื่อแสดงจำนวนของ Tuple (Attribute "ROWCOUNT") และจำนวนของ Attribute (Attribute "COLCOUNT") ส่วนชื่อของ Attribute ต่าง ๆ ของ Relation "DEPARTMENT" โปรแกรม DBMS จะเข้าไปค้นหาใน Relation "COLUMNS" แทน ซึ่งจะได้ผลออกมาเป็น Attribute DEPT# DNAME และ BUDGET

Catalog นอกจากจะเป็น Relation ที่แสดงถึงโครงสร้างของ Relation ต่าง ๆ ในฐานข้อมูลแล้ว ยังจะต้องสามารถแสดงโครงสร้างของตัวเองได้ด้วย เพื่อที่จะได้สามารถทราบ

ว่า Relation ใดเป็น Catalog และ Relation ใดใช้สำหรับจัดเก็บข้อมูล และแต่ละ Relation มีโครงสร้างเป็นเช่นใด

2.2 VISUAL BASIC

Visual basic เป็นภาษาคอมพิวเตอร์ที่ใช้ในการพัฒนาโปรแกรมในลักษณะ Front-End ซึ่งผู้พัฒนาโปรแกรมสามารถที่จะกำหนดรูปแบบของจอภาพให้ง่ายแก่การใช้งาน เพื่อลดความซ้ำซ้อนในการใช้งานของ Database Management System (DBMS) ซึ่งเป็นโปรแกรมที่ใช้ในการจัดการข้อมูลในฐานข้อมูลไม่ว่าจะเป็นการจัดเก็บ การค้นคืน การเพิ่ม การลบ หรือการแก้ไขข้อมูล

ในการแก้ไขข้อมูล Visual basic จะกระทำผ่านตัว Database Engine ซึ่งมีความสามารถในการเข้าถึงฐานข้อมูลได้หลายรูปแบบ เช่น Microsoft Access, dBase, Foxpro, Paradox หรือ Btrieve โดยแต่ละฐานข้อมูลจะมีโครงสร้างที่แตกต่างกันไป แต่ที่นิยมใช้กับ Visual basic ได้แก่ MySQL ซึ่งมีโครงสร้างแบบ Relation Database ดังนั้นในโครงการนี้เราจะใช้ MySQL เป็นฐานข้อมูลในการทำงานของโปรแกรม

2.2.1 ความรู้ทั่วไปเกี่ยวกับ Relation Database

Relation Database คือกลุ่มของข้อมูลที่สัมพันธ์กัน ซึ่งจัดเก็บอยู่ในรูปของตาราง แต่ละตารางจะประกอบไปด้วยกลุ่มของ Record โดยแต่ละ Record จะประกอบไปด้วยกลุ่มของ Field ดังนั้นจึงอาจกล่าวได้ว่าแต่ละตารางก็เปรียบเสมือนกับ Spread Sheet ที่มี Record แสดงอยู่ในรูปของแถว (row) และ Field แสดงอยู่ในรูปของสดมภ์ (column)

ในการเข้าถึงข้อมูลในแต่ละ Table อาจจะทำตามขอบเขตในการเข้าถึงข้อมูลได้ เช่น ต้องการให้ Table “ข้อมูลพนักงาน” แสดงข้อมูลของพนักงานโดยเฉพาะผู้ที่มียอดเงินเดือนมากกว่า 10,000 บาท ในขณะที่เดียวกันถ้าต้องการให้สะดวกต่อการดู ก็อาจสั่งให้ Table นั้นเรียงลำดับ (Sort) ข้อมูลเช่น เรียงลำดับข้อมูลตามเงินเดือน และถ้าต้องการให้เข้าถึงข้อมูลได้ด้วยความเร็ว ก็อาจจะใช้ดัชนี (index) มาช่วยในการเข้าถึงข้อมูลได้ นอกจากนี้ความสามารถที่น่าสนใจอย่างหนึ่งของ Relation Database ได้แก่ การนำข้อมูลจากหลายๆ Table ที่มีความสัมพันธ์กันมาใช้งานร่วมกัน (Join) ได้

ในการทำ Index แต่ละ Table จะต้องมี Field ที่ใช้เป็น Key ซึ่งแบ่งออกเป็น Primary Key และ Foreign Key โดย Field ที่ใช้ในการอ้างถึงข้อมูลใน Table ที่สัมพันธ์กันจะเรียกว่า “Foreign Key”

2.2.2 ประเภทของความสัมพันธ์ระหว่าง Table แบ่งออกเป็น 3 ประเภท คือ

ประเภทที่ 1 ความสัมพันธ์แบบ One-to-Many

หมายถึง ข้อมูล 1 Record ใน Table หนึ่งจะมีความสัมพันธ์กับข้อมูลในอีก Table หนึ่งมากกว่า 1 Record เช่น เจ้าของบัญชีเงินฝากธนาคาร 1 คน สามารถมีบัญชีเงินฝากได้มากกว่า 1 บัญชี

ประเภทที่ 2 ความสัมพันธ์แบบ One-to-One

หมายถึง ข้อมูลแต่ละ Record ใน Table หนึ่งจะมีความสัมพันธ์กับข้อมูลในอีก Table หนึ่งเพียง Record เดียว เช่น กรณีมีการกำหนดให้ บัญชีเงินฝากธนาคารแต่ละบัญชีจะมีเจ้าของบัญชีได้เพียงคนเดียว

ประเภทที่ 3 ความสัมพันธ์แบบ Many-to-Many

หมายถึง หลาย Record ใน Table หนึ่งจะมีความสัมพันธ์กับอีกหลาย Record ในอีก Table หนึ่งเช่น เจ้าของบัญชีเงินฝาก 1 คน สามารถเป็นเจ้าของบัญชีเงินฝากได้มากกว่า 1 บัญชี และแต่ละบัญชีเงินฝากสามารถมีเจ้าของบัญชีได้มากกว่า 1 คน

2.2.3 Data Control

Visual Basic อาศัย Control ชื่อ "Data" (มักจะเรียกว่า "Data Control") ในการทำงานร่วมกับ Database โดยที่ 1 Data Control จะใช้อ้างถึง 1 Table หรือมากกว่าในฐานข้อมูล(กรณีที่อ้างถึงมากกว่า 1 Table จะหมายถึงการนำเอา Table ต่างๆที่สัมพันธ์กันมา Join กัน)ข้อมูลที่ถูกอ่านจาก Table มาเก็บไว้ใน Data Control จะเรียกว่า "Recordset" ดังนั้นจึงอาจกล่าวได้ว่า Recordset ได้แก่ กลุ่มของ Record ใน Table ซึ่งอาจมาจาก 1 หรือมากกว่า 1 Table ที่ถูกอ่านเข้ามาเก็บไว้ในตัว Data Control ตามเงื่อนไขที่กำหนด ในการอ้างอิงถึง Recordset ใน Data Control ให้เขียนอยู่ในรูปแบบ ดังนี้

datacontrol.Recordset.methodproperty

โดยที่ *datacontrol* หมายถึง ชื่อของ Data Control *methodproperty* หมายถึง Method หรือ Property ของ Data Control

ในการทำงานกับ Data Control จะต้องอาศัย Property ต่างๆ ดังนี้

- Connect ใช้สำหรับกำหนดประเภทของฐานข้อมูล สำหรับ Visual Basic 6.0 จะสามารถเข้าถึงข้อมูลที่เป็น dbase (dbase III,dbase IV และ dbase 5.0) , Exel (Version 3.0, 4.0, 5.0 และ 8.0),

Foxpro (Version 2.0,2.5,2.6 และ3.0), Lotus (WK1,WK3 และ WK4), Paradox (Version 3.x,4.x และ5.x) และ Text File ทั่วไป

- DatabaseName ใช้สำหรับกำหนด Path และชื่อของ Database (สำหรับ Microsoft Access จะเป็น file.MDB)

- RecordsetType ใช้สำหรับกำหนดประเภทของ Recordset ซึ่งประกอบด้วย

- Table เป็น Recordset ที่กระทำกับ Table เพียง Table เดียว
- Dynaset เป็น Recordset ที่กระทำกับ Table ตั้งแต่ 1 Table ขึ้นไป Recordset ประเภทนี้จะสามารถแก้ไขข้อมูลได้ โดยข้อมูลที่ได้รับการแก้ไข จะถูกส่งผ่านไปยัง Table ที่อ้างถึงโดยอัตโนมัติ
- Snapshot เป็น Recordset ที่กระทำกับ Table ตั้งแต่ 1 Table ขึ้นไป แต่ Recordset ประเภทนี้จะไม่สามารถแก้ไขข้อมูลได้ ส่วนใหญ่จะใช้ในการแสดงผลเพียงอย่างเดียว
- RecordSource ใช้สำหรับ กำหนดชื่อของ Table เมื่อคลิกใน property นี้จะปรากฏ เป็น ListBox ที่ประกอบไปด้วย ชื่อของ Table ต่างๆใน Database ที่ กำหนดไว้ใน Property "DatabaseName"
- BOFAction ใช้กำหนดการทำงานให้กับ Data Control เมื่อเกิดสถานะ "Begin of File" ขึ้น ซึ่ง สามารถกำหนดได้ 2 ลักษณะดังนี้
 - "0-MoveFirst" (ค่า Default) เพื่อเลื่อน Pointer ไปยัง Record แรก
 - "1-BOF" เพื่อกำหนดค่า Property "BOF" เป็น True ซึ่งจะส่งผลให้ Data Control ใช้ Method "MoveLast" ไม่ได้
- EOFAction ใช้กำหนดการทำงานให้กับ Data Control เมื่อเกิดสถานะ "End of File" ขึ้นซึ่ง สามารถกำหนดได้ 3 ลักษณะดังนี้
 - "0-MoveLast" (ค่า Default) เพื่อเลื่อน Pointer ไปยัง Record สุดท้าย
 - "1-EOF" เพื่อกำหนดค่าของ Property "EOF" เป็น True ซึ่งจะส่งผล DataControl ใช้ Method "MoveNext" ไม่ได้
 - "2-AddNew" เพื่อเพิ่ม Record ให้กับ Recordset โดยอัตโนมัติ
- ReadOnly เป็น Property ที่มีลักษณะข้อมูลแบบตรรกะใช้สำหรับกำหนดให้ Data Control อ่านได้อย่างเดียวเมื่อกำหนดให้มีค่าเป็น True และจะสามารถแก้ไขได้ เมื่อกำหนด ให้มีค่า เป็น False

2.2.4 Bound Control

Bound Control ได้แก่ Control ต่างๆ ที่สามารถใช้งานร่วมกับ Recordset ใน Data Control เพื่อใช้ในการแสดงผลและรับข้อมูลจากทางจอภาพ Bound Control มาตรฐานใน Visual Basic ได้แก่ TextBox , ListBox ,CheckBox, Image Label และ Picture

2.2.5 Method ที่ใช้ในการจัดการข้อมูล

ในการทำงานกับ Database โดยทั่วไปจะเกี่ยวข้องกับการเพิ่ม แก้ไข ลบและค้นหาข้อมูล ซึ่งแต่ละการทำงานจะอาศัย Method ที่แตกต่างกันไป ดังนั้นในส่วนนี้ เราจะมาเรียนรู้การใช้ Method ต่างๆ ในการจัดการกับข้อมูลใน Database

• การเพิ่มข้อมูลเข้าไปยัง Table ต่างๆ ใน Database จะอาศัย Method “AddNew” เพื่อสร้าง Record ใหม่เพิ่มเข้าไปใน Recordset

Recordset.AddNew

โดยที่ Recordset หมายถึง ชื่อ Object ที่เป็นเจ้าของ Recordset ในกรณีที่ใช้ Data Control ให้กำหนดในรูป Datacontrol.Recordset โดย datacontrol ได้แก่ ชื่อของ Datacontrol นั้น

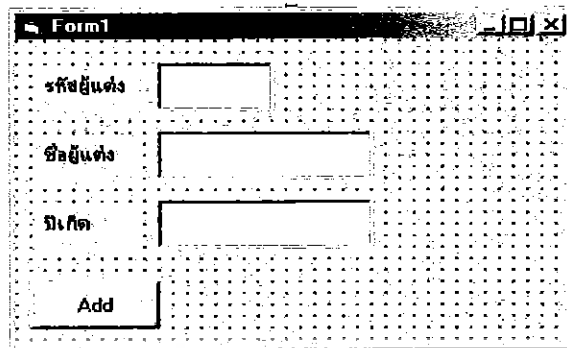
ตัวอย่าง การใช้งาน Method “AddNew”

- วาด Control “CommandButton” ต่างๆ ลงบน Form
- กำหนด Property ของ CommandButton ดังตาราง

ตารางที่ 2.3 การกำหนด Property ของ CommandButton

Object	Property	ค่าที่กำหนด
CommandButton	Name	AddCmnd
	Caption	&Add

เมื่อเสร็จแล้วจะได้จอภาพดังรูป



รูปที่ 2.1 รูปที่ได้จากการกำหนด Property ของ CommandButton

- พิมพ์คำสั่งใน Event "click" ของปุ่ม Add ดังนี้

```
Private Sub AddCmnd_Click()
    Author.Recordset.AddNew
End Sub
```

- Run จากนั้นให้คลิกที่ปุ่ม Add จะสังเกตเห็นว่า ข้อมูลใน TextBox ต่างๆจะหายไป

• การแก้ไขข้อมูล Method ที่ใช้ในการบันทึกข้อมูลที่มีการแก้ไข ได้แก่ Method "Update" โดยจะบันทึกข้อมูลที่ปรากฏอยู่ใน Object ที่ Bound Control กลับลงไปยัง Recordset รูปแบบของคำสั่งเป็นดังนี้

```
Recordset.Update
```

โดยที่ Recordset หมายถึง ชื่อ Object ที่เป็นเจ้าของ Recordset ในกรณีที่ใช้ DataControl ให้กำหนด ในรูป Datacontrol.Recordset โดย Datacontrol ได้แก่ ชื่อของ Data Control นั้น

ตัวอย่าง การใช้ Method "Update"

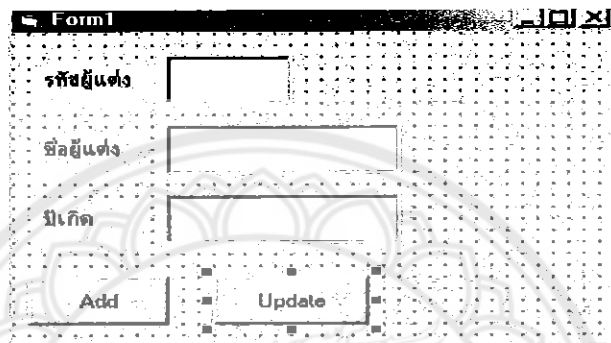
- จากตัวอย่างที่แล้วว่า Control "CommandButton" เพิ่มลงไปบน Form อีก 1 Object

- กำหนดคุณสมบัติ ของ Object ดังตาราง

ตารางที่ 2.4 การกำหนดคุณสมบัติของ Object

Object	Property	ค่าที่กำหนด
CommandButton	Name	UpdateCmnd
	Caption	&Update

เมื่อเสร็จแล้วจะได้จอภาพดังรูป



รูปที่ 2.2 หน้าตาของโปรแกรมเมื่อใส่ Update Button ลงไป

- พิมพ์คำสั่งลงใน Event "Click" ของปุ่ม Update ดังนี้

```
Private Sub UpdateCmnd_Click()
    Author.Recordset.Update
End Sub
```

- Run จากนั้นคลิกที่ปุ่ม Add ข้อมูลในแต่ละ Text Box จะถูก Clear ไป ให้ทดลองป้อนข้อมูลในแต่ละ TextBox

- การลบข้อมูล

Method ที่ใช้สำหรับลบข้อมูล Record ปัจจุบันออกจาก Recordset ได้แก่ Method "Delete"

```
recordset.Delete
```

โดยที่ recordset หมายถึง ชื่อ Object ที่เป็นเจ้าของ Recordset ในกรณีที่ใช้ Datacontrol ให้กำหนดในรูป datacontrol.Recordset โดย datacontrol ได้แก่ ชื่อของ Datacontrol นั้น

ในการลบ Record ทุกครั้งหลังจากลบควรถือจะมีการขยับ Pointer เนื่องจาก Record ปัจจุบันได้ถูกลบไปเพื่อป้องกันการสับสนว่า Pointer ชี้ไปที่ Record ไດ สำหรับ Method ที่ใช้ในการเลื่อน Pointer ได้แก่ Method "MoveFirst", "MoveNext", "MoveLast", และ "MovePrevious"

Recordset. {MoveFirst|MoveLast|MoveNext|MovePrevious}

โดยที่ Recordset หมายถึง ชื่อ Object ที่เป็นเจ้าของ recordset ในกรณีที่ใช้ DataControl ให้กำหนดในรูป datacontrol.Recordset โดย datacontrol ได้แก่ ชื่อของ datacontrol นั้น

ทั้ง 4 Method นี้ จะมีรูปแบบของคำสั่งเช่นเดียวกันสำหรับหน้าที่ของแต่ละ Method มีดังนี้

- MoveFirst ใช้สั่งให้ DataControl เลื่อน Pointer ไปยัง record แรกใน recordset
- MoveLast ใช้สั่งให้ DataControl เลื่อน Pointer ไปยัง record สุดท้ายใน recordset
- MoveNext ใช้สั่งให้ DataControl เลื่อน Pointer ไปยัง record ถัดไปใน recordset
- MovePrevious ใช้สั่งให้ DataControl เลื่อน Pointer ไปยัง record ก่อนหน้าใน recordset

ในกรณีที่ Pointer ของ Record ปัจจุบันชี้อยู่ที่ record แรกของ Recordset เมื่อใช้ Method "MovePrevious" ค่าของ Property "BOF" ของ Datacontrol จะถูกกำหนดให้มีค่าเป็น True และก็เช่นเดียวกัน กรณีที่ Pointer ของ Record ปัจจุบันชี้อยู่ที่ record สุดท้ายของ recordset เมื่อใช้ Method "MoveNext" ค่าของ Property "EOF" จะถูกกำหนดให้มีค่าเป็น True เช่นกัน และถ้าไม่ใช่ทั้ง 2 กรณีแล้วค่าของ Property "BOF" และ "EOF" จะถูกกำหนดให้มีค่าเป็น False เสมอ

ตัวอย่าง การใช้ Method "Delete"

- จากตัวอย่างที่แล้ววาด CommandButton เพิ่มเติมลงบน Form จากนั้นกำหนด Property ให้กับ Object ดังกล่าวดังตาราง

ตารางที่ 2.5 การกำหนด Property ให้กับ Object โดยเพิ่ม Method Delete

Object	Property	ค่าที่กำหนด
Command Button	Name	DeleteCmnd
	Caption	&Delete

เมื่อเสร็จแล้วจะได้จอภาพดังรูป

รูปที่ 2.3 แสดงหน้าต่างของโปรแกรมหลังจากใส่ button Delete เข้าไป

- พิมพ์คำสั่งให้กับปุ่ม Delete ดังนี้

```
Private Sub DeleteCmdnd_Click()
    Author.Recordset.Delete
    Author.Recordset.MoveLast
End Sub
```

- Run แล้วคลิกที่ปุ่มหมายเลข 4 ใน DataControl เพื่อขยับไปยัง Record สุดท้ายใน Recordset
- คลิกที่ปุ่ม Delete ข้อมูลบนจอภาพจะเปลี่ยนเป็นข้อมูล Record ก่อนหน้า เนื่องจาก Record สุดท้ายได้ถูกลบออกไปแล้ว
- การค้นหาข้อมูล

Method ที่ใช้ในการค้นหาข้อมูลใน Recordset จะประกอบไปด้วย

- FindFirst ใช้สำหรับค้นหา Record โดยทิศทางในการหาจะเริ่มจาก Record แรก ไปยัง Record สุดท้ายใน Recordset จนกระทั่งพบ record ซึ่งตรงกับเงื่อนไขการค้นหา
- FindLast ใช้สำหรับค้นหา Record โดยทิศทางในการหาจะเริ่มจาก Record สุดท้ายไปยัง Record แรกใน Recordset จนกระทั่งพบ record ซึ่งตรงกับเงื่อนไขการค้นหา

- **FindNext** ใช้สำหรับค้นหา Record โดยทิศทางการหาจะเริ่มจาก Record ปัจจุบันไปยัง Record สุดท้ายใน Recordset จนกระทั่งพบ record ซึ่งตรงกับเงื่อนไขการค้นหา
- **FindPrevious** ใช้สำหรับค้นหา Record โดยทิศทางการหาจะเริ่มจาก Record ปัจจุบันไปยัง Record แรกใน Recordset จนกระทั่งพบ record ซึ่งตรงกับเงื่อนไขการค้นหา

Recordset. {FindFirst|FindLast|FindNext|FindPrevious} criteria

โดยที่ recordset หมายถึง ชื่อ Object ที่เป็นเจ้าของ Recordset ในกรณีที่ใช้ Datacontrol ให้กำหนดในรูป datacontrol.Recordset โดย Datacontrol ได้แก่ ชื่อของ Datacontrol นั้น criteria หมายถึง ประโยคเงื่อนไขที่ใช้ในการค้นหา

2.2.6 DBCombo และ DBList

DBCombo เป็น control ที่มีลักษณะการทำงานคล้ายกับ comboBox ส่วน DBList เป็น control ที่มีลักษณะการทำงานคล้ายกับ ListBox แต่ต่างกันที่ 2 control นี้ จะสามารถเข้าถึงข้อมูลใน Recordset ได้โดยตรง แต่เนื่องจากทั้ง 2 control นี้ ไม่ได้เป็น BoundControl มาตรฐาน ดังนั้นจึงต้องเพิ่มเข้ามาไว้ใน Toolbox ก่อน

2.2.7 DBGrid

เป็น control ที่ใช้การนำเสนอข้อมูลในลักษณะของ Spreadsheet และเป็นอีก Control หนึ่งที่ไม่ใช่ Bound Control มาตรฐาน ดังนั้น จึงต้องเพิ่มเข้ามาไว้ใน Toolbox โดยเลือก "Microsoft Data Bound Grid Control (SP3)" ในจอภาพ Components สำหรับ Icon ที่ใช้แทน DBGrid ใน Toolbox

2.3 ภาษาSQL

SQL เป็นภาษามาตรฐานที่ใช้ติดต่อกับระบบฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management System) หรือ RDBMS ซึ่ง ANSI ได้ประกาศออกมาอย่างเป็นทางการ ดังนั้น ผู้ที่ทำงานกับฐานข้อมูลในปัจจุบันจำเป็นต้องรู้ เนื่องจากระบบฐานข้อมูลที่มีอยู่ในปัจจุบันเกือบทั้งหมดเป็นระบบฐานข้อมูลแบบ RDBMS SQL สามารถแบ่งคำสั่งออกเป็น 4 กลุ่ม คือ

- Data Manipulate(DML) เป็นคำสั่งจัดการข้อมูล ได้แก่ INSERT, UPDATE, DELETE, ROLLBACK, COMMIT
- Data Definition (DDL) เป็นคำสั่งจัดการกับไฟล์ในฐานข้อมูล ได้แก่ CREATE, ALTER, DROP
- Query เป็นคำสั่งการเรียกดูข้อมูล คือ SELECT
- Data Control เป็นคำสั่งจัดการความปลอดภัย

2.3.1 การเลือกข้อมูล

SELECTION statement ใช้ในการสร้างคิวรี หรือการสอบถามกับฐานข้อมูล และเลือกดึงข้อมูลที่ตรงกับเกณฑ์ที่กำหนด ไวยากรณ์

SELECT column1[,column2...] FROMtablename [WHEREcondition]; [] – optional

- column เป็นคอลัมน์ที่ต้องการให้แสดงผล ถ้าต้องการเลือกทุกคอลัมน์ใน Table ให้ใช้ *SELECT * FROM weather;
- Tablename เป็นการระบุ Table ที่ต้องการดึงข้อมูล
- WHERE WHERE clause เป็นตัวเลือกซึ่งจะมีหรือไม่ก็ได้ใช้กำหนด หรือเงื่อนไขในการเลือกเฉพาะข้อมูลที่ต้องการ

Condition ที่ใช้ใน WHERE clause คือ

=	เท่ากับ
>	มากกว่า
<	น้อยกว่า
>=	มากกว่าหรือเท่ากับ
<=	น้อยกว่าหรือเท่ากับ
<>	ไม่เท่ากับ
LIKE	ตัวกระทำ(Operator)*

ตัวอย่าง Table ชื่อ 'empinfo'

ตาราง 2.6 ฐานข้อมูลตัวอย่าง empinfo

first	last	id	age	city	state
John	Jones	99980	45	Payson	Arizona
Mary	Jones	99982	25	Payson	Arizona
Eric	Edwards	88232	32	San Diego	California
Mary Ann	Edwards	88233	32	Phoenix	Arizona
Ginger	Howell	98002	42	Cottonwood	Arizona
Sebastian	Smith	92001	23	Gila Bend	Arizona
Gus	Gray	22322	35	Bagdad	Arizona

*LIKE เป็นแบบแผนของ operator การจับคู่ สามารถใช้ใน condition ของ WHERE clause ซึ่ง LIKE เป็น operator ที่มีความสามารถสูงในการเลือกข้อมูล เครื่องหมายเปอร์เซ็นต์ "%" สามารถเป็น wild card ในการเลือก โดยการจับคู่ตัวอักษรไม่จำกัดที่อยู่หน้าหรือหลังตัวอักษรที่กำหนด เช่น R% ให้มีอักษร R อยู่ซ้ายสุด หรือหน้า %R ให้มีอักษร R อยู่ขวาสุดหรือท้าย %R% ให้มีอักษร R อยู่ตรงกลาง

2.3.2 การสร้าง Table

CREATE TABLE statement ใช้ในการสร้าง Table ขึ้นมาใหม่ไว้อากรณ์ รูปแบบการสร้าง Table อย่างง่าย

```
CREATE TABLE tablename
(column1 datatype,column2 datatype,column3 datatype, ...);
```

อธิบายความหมายของรูปแบบ CREATE statement

- tablename ระบุชื่อ Table
- column ระบุคอลัมน์ที่ต้องการ
- datatype กำหนดประเภทข้อมูลของคอลัมน์

การสร้าง Table ใหม่ให้พิมพ์คำสั่ง CREATE TABLE ตามด้วยชื่อ Table แล้วพิมพ์เครื่องหมายวงเล็บเปิด ตามด้วยชื่อคอลัมน์ แล้วกำหนด ประเภทข้อมูล ถ้าต้องการกำหนด constraint แล้วพิมพ์เครื่องหมายจุลภาค (,) จากนั้นให้พิมพ์คอลัมน์อื่นต่อไป โดยแบ่งแต่ละคอลัมน์ด้วยจุลภาค จนถึงคอลัมน์สุดท้าย ให้พิมพ์ตามเครื่องหมายวงเล็บปิด ตามด้วยเครื่องหมาย semicolon (;)

- การกำหนดคอลัมน์

ชื่อคอลัมน์ สามารถกำหนดเป็นตัวอักษร หมายเลข และเส้นใต้ แต่ไม่เกิน 30 ตัวอักษร และไม่เป็น reserve word ของ SQL (เช่น SELECT, CREATE เป็นต้น) ประเภทข้อมูล ต้องกำหนดให้เหมาะสมกับข้อมูลที่จะเก็บในคอลัมน์ เช่น first เป็นการเก็บชื่อพนักงาน ควรกำหนดเป็น CHAR หรือ VARCHAR (variable-length character), age เก็บอายุ ควรกำหนดเป็น NUMBER เป็นต้น

ประเภทข้อมูลที่ใช้บ่อย

- CHAR(size) ตัวอักษรชนิดความยาวคงที่ size ระบุจำนวนตัวอักษรได้ไม่เกิน 255 ไบต์
- VARCHAR(size) ตัวอักษรชนิดความยาวแปรผันในการเก็บ size ระบุจำนวนตัวอักษร
- NUMBER(size) ตัวเลข size กำหนดจำนวนหลักที่ต้องการ
- DATE วันที่
- NUMBER(size,d) ตัวเลขชนิดมีทศนิยม size กำหนดจำนวนหลักทั้งหมดรวมถึงทศนิยม d กำหนดตำแหน่งทศนิยม

2.3.3 การกำหนด Key

Key เป็นข้อกำหนดคุณสมบัติของคอลัมน์ใน Table ซึ่งทำหน้าที่เป็น index เพื่อเพิ่มความเร็วในการค้นหาข้อมูล และใช้กำหนดความสัมพันธ์ระหว่าง Table

Primary key สร้างจากคอลัมน์ (หรือประกอบด้วยหลายคอลัมน์) ที่มีค่าของคอลัมน์ในทุกแถวข้อมูลมีค่าไม่ซ้ำกัน เป็นการควบคุมค่าที่เก็บในคอลัมน์ และใช้ในการเชื่อมโยงกับ Table อื่นในแต่ละ Table กำหนด primary key ได้ 1 key

Foreign key เป็นคอลัมน์ใน Table ที่ primary key อยู่ที่ Table อื่น หมายถึงว่า ค่าของคอลัมน์ทั้งหมดใน Table ต้องมีค่าสอดคล้องกับค่าของคอลัมน์ที่เป็น primary key ของ Table ที่สัมพันธ์กัน

Index สามารถกำหนดขึ้นเพื่อใช้ในการเพิ่มความเร็วในการค้นหาข้อมูล การเรียงลำดับข้อมูล แต่ควรสร้างตามความจำเป็น เนื่องจากถ้ามีคอลัมน์ที่เป็น Index มากจะทำให้การทำงานช้าลง

ตัวอย่างการกำหนด primary key

```
CREATE TABLE employee
```

```
(employee_id (10),
```

```
first varchar(15),
```

```
last varchar(20),
```

1494161๑ ๔2

พ.ศ.

๒๕๖๕

๒๕๔๗

age number(3),
 address varchar(30),
 city varchar(20),
 state varchar(20),
 CONSTRAINT employee_primary_key PRIMARY KEY (employee_id));

ตัวอย่างการกำหนด foreign key

CREATE TABLE employee
 (employee_id (10),
 first varchar(15),
 last varchar(20),
 age number(3),
 address varchar(30),
 city varchar(20),
 state varchar(20),
 CONSTRAINT employee_foreign_key PRIMARY KEY (state));

ตัวอย่างการกำหนด index

ไวยากรณ์

CREATE INDEX indexname
 ON tablename (column);

- indexname กำหนดชื่อ index
- tablename ที่ Table ที่ต้องการกำหนดสร้าง index
- column ชื่อคอลัมน์ที่กำหนดเป็น index

ตัวอย่าง

CREATE INDEX Index_city
 ON employee (city);

2.3.4 การแก้ไข Table

ALTER statement ใช้ในการปรับปรุงคอลัมน์ใน Table ที่สร้างเสร็จแล้ว

ไวยากรณ์ 1 การปรับปรุงคอลัมน์

```
ALTER TABLE tablename
ADD (column1 datatype [constraint][,column2 datatype [constraint],
column3 datatype [constraint],...]);[] – optional
```

ไวยากรณ์ 2 การปรับปรุง constraint ของ Table

```
ALTER TABLE tablename
ADD (tableconstraint);
```

- tablename	ระบุชื่อ Table
- column	ระบุคอลัมน์ที่ต้องการ
- datatype	กำหนดประเภทข้อมูลของคอลัมน์
- constraint	ข้อกำหนด หรือคุณสมบัติของคอลัมน์
- tableconstraint	กำหนด constraint ของ Table เช่น กำหนด primary key หรือ foreign key

ตัวอย่าง การปรับปรุง column

```
ALTER TABLE employee
ADD(employee_id(10),
first varchar(20),
edu varchar(20),
```

ตัวอย่าง การปรับปรุง primary key

```
ALTER TABLE employee
ADD (CONSTRAINT employee_primary_key1 PRIMARY KEY (employee_id);
```

การปรับปรุง Table ใหม่ให้พิมพ์คำสั่ง CREATE TABLE ตามด้วยชื่อ Table ตามด้วย ADD แล้วพิมพ์เครื่องหมายวงเล็บเปิด ตามด้วยชื่อคอลัมน์ แล้วกำหนด ประเภทข้อมูล ถ้าต้องการ กำหนด constraint แล้วพิมพ์เครื่องหมายจุลภาค (,) จากนั้นให้พิมพ์คอลัมน์อื่นต่อไป โดยแบ่งแต่ละ คอลัมน์ด้วยจุลภาค จนถึงคอลัมน์สุดท้าย ให้พิมพ์ตามเครื่องหมายวงเล็บปิด ตามด้วยเครื่องหมาย semicolon (;) ถ้าเป็นการปรับปรุง constraint ของ Table ให้ constraint ของ Table ในวงเล็บ และ แก้ไขให้พิมพ์ชื่อคอลัมน์เดิมและการปรับปรุง ข้อกำหนดจะเข้าไปแทนที่ข้อกำหนดเดิม

2.3.5 การป้อนข้อมูล

INSERT statement ใช้ในการป้อนหรือเพิ่มแถวข้อมูลเข้า Table

ไวยากรณ์

```
INSERT INTO tablename
(column1,column2,...)
VALUES(value1,value2,...);
```

- tablename ระบุชื่อ Table
- column ระบุคอลัมน์ที่ต้องการ เพิ่มค่า
- value ค่าที่ต้องการป้อน

หมายเหตุ แต่ละค่าของ value กับ column จะต้องตรงกัน

- การป้อนข้อมูลจาก table อื่น

ตัวอย่าง ตามตัวอย่างเป็นการป้อนข้อมูลเข้าสู่ table "employee" ด้วยข้อมูลจาก table "temp_emp"

```
INSERT INTO employee
(empid, first, last, age, dept)
SELECT empno, firstname, lastname, age, department
FROM temp_emp
WHERE departmant IN ('Sales','Account','IT');
```

- การป้อนข้อมูลที่เป็น Date

ตัวอย่าง การป้อนค่าคอลัมน์ hiredate ด้วย '25-AUG-99'

```
INSERT INTO empid, firsrname, lastname, hiredate, dept)
VALUE (1150, 'Luke', 'Duke', '25-AUG-99', 'Sales')
```

ตัวอย่าง ในฐานะข้อมูล Oracle สามารถป้อนค่าด้วยฟังก์ชัน TO_DATE

```
INSERT INTO empid, firsrname, lastname, hiredate, dept)
VALUE (1150, 'Luke', 'Duke', TO_CHAR('25/10/1999','dd/mm/yyyy'), 'Sales');
```

2.3.6 การปรับปรุงข้อมูล

UPDATE statement ใช้ในการปรับค่าหรือเปลี่ยนข้อมูลตามเกณฑ์ที่กำหนด

ไวยากรณ์

```
UPDATE tablename
SET columnname1 = newvalue1 [, "columnname2" = "newvalue2"...]
WHERE column1 OPERATOR value [AND|OR column2 OPERATOR value];
[] = optional
```

- tablename ระบุชื่อ Table
- columnname ระบุคอลัมน์ที่ต้องการ แก้ไขค่า
- newvalue ค่าใหม่ที่ต้องการแก้ไข
- columnname ชื่อคอลัมน์ที่กำหนดเงื่อนไข
- value ค่าของเงื่อนไข

ตัวอย่าง

```
UPDATE phone_book
SET area_code = 623
WHERE prefix = 979;
UPDATE phone_book
```

```

SET last_name = 'Smith', prefix=555, suffix=9292
WHERE last_name = 'Jones';
UPDATE employee
SET age = age+1
WHERE first_name='Mary' AND last_name='Williams';

```

การปรับปรุงข้อมูล ให้พิมพ์คำสั่ง UPDATE ตามด้วยชื่อ Table แล้วตามด้วยคำสั่ง SET ชื่อคอลัมน์ที่ต้องการเปลี่ยนค่า ตามเครื่องหมายเท่ากับ ค่าใหม่ที่นำมาแทนที่ ถ้าปรับปรุงหลายคอลัมน์ ให้ชื่อคอลัมน์ เท่ากับ ค่าใหม่ แล้วแยกด้วยเครื่องหมายจุลภาค (ยกเว้นค่าสุดท้าย) แล้วตามด้วย WHERE แล้วตามด้วยเงื่อนไข ถ้ามีหลายเงื่อนไขให้เชื่อมด้วย AND หรือ OR แล้วปิดท้ายคำสั่งด้วยเครื่องหมาย semicolon (;)

2.3.7 การลบข้อมูล

DELETE statement เป็นคำสั่งที่ใช้การลบข้อมูล

ไวยากรณ์

```

DELETE FROM tablename
WHERE column OPERATOR value [AND|OR column OPERATOR value];
[ ] = optional

```

- tablename ระบุชื่อ Table
- columnname ชื่อคอลัมน์ที่กำหนดเงื่อนไข
- value ค่าของเงื่อนไข

ตัวอย่าง

```
DELETE FROM employee;
```

NOTE: ถ้าใช้คำสั่ง DELETE statement โดยไม่มี WHERE clause จะเป็นการลบข้อมูลทั้งหมดใน Table


```
DELETE FROM employee
WHERE lastname = 'May';
DELETE FROM employee
WHERE firstname = 'Mike' or firstname = 'Eric';
```

การลบเป็นการลบแถวข้อมูลออกจาก Table ให้ป้อนคำสั่ง DELETE FROM ตามด้วย ชื่อ Table ตามด้วย WHERE และเงื่อนไขในการลบไว้ ถ้ามีหลายเงื่อนไขให้เชื่อมด้วย AND หรือ OR แล้วปิดท้ายคำสั่งด้วยเครื่องหมาย semicolon (;)

2.3.8 การลบ Table

DROP statement ที่ใช้ในการลบ Table หรือยกเลิก

ไวยากรณ์

```
DROP TABLE tablename
```

ตัวอย่าง

```
DROP TABLE employee;
```

การลบ Table ให้พิมพ์คำสั่ง DROP TABLE ตามชื่อ Table ที่ต้องการลบแล้วปิดท้ายคำสั่งด้วยเครื่องหมาย semicolon (;)

บทที่ 3

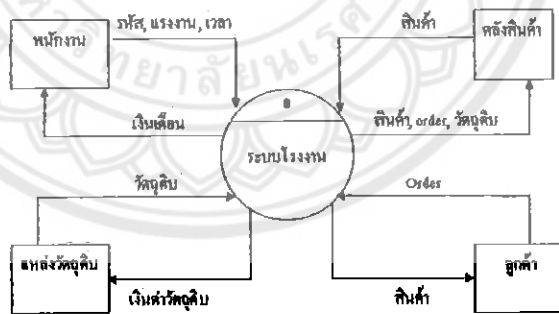
วิธีการดำเนินงานและผลดำเนินการโครงการ

การจัดการระบบสารสนเทศส่วนของการดำเนินการ ต้องศึกษาส่วนข้อมูลของบริษัท คือ ระบบกระบวนการ การทำงานส่วนต่างๆของบริษัท โดยทำการเก็บข้อมูล เพื่อนำข้อมูลที่ได้มานั้น มาวิเคราะห์ ปรับปรุง ออกแบบและสร้างระบบฐานข้อมูล ซึ่งจำเป็นต้องใช้ในการจัดทำโครงการงน โดยกระบวนการ การทำงานของโรงงานนั้นจะเขียนอยู่ในรูปของ Data Flow Diagram เพื่อให้เห็น ภาพรวมกระบวนการ การทำงานของบริษัท ดังต่อไปนี้

3.1 Data Flow Diagram

ในการเขียนโปรแกรมจัดการระบบฐานข้อมูลนั้น สิ่งที่ต้องทำก่อนการเขียนโปรแกรมคือ การวิเคราะห์ระบบนั้นก่อนว่าทำงานอย่างไร โดยการเขียนเป็นแผนภาพการไหลของข้อมูล (Data Flow Diagram) โดยแผนภาพการไหลของข้อมูลนี้แบ่งเป็น แผนภาพ Context Diagram แผนภาพ Level 0 แผนภาพ Level 1 ซึ่งในระบบอื่นอาจมีการวิเคราะห์ระบบที่สูงกว่า Level 1 แต่ในโครงการงนนี้เราจะวิเคราะห์ถึง Level 1 เท่านั้น

3.1.1 Context Diagram เป็นการวิเคราะห์ระบบเพียงแค่มียุคคลใดที่เกี่ยวข้องกับบริษัท และมีข้อมูลหรือสิ่งใดเข้าหรือออกจากระบบ ทางคณะผู้จัดทำได้วิเคราะห์ระบบของบริษัทดังรูป

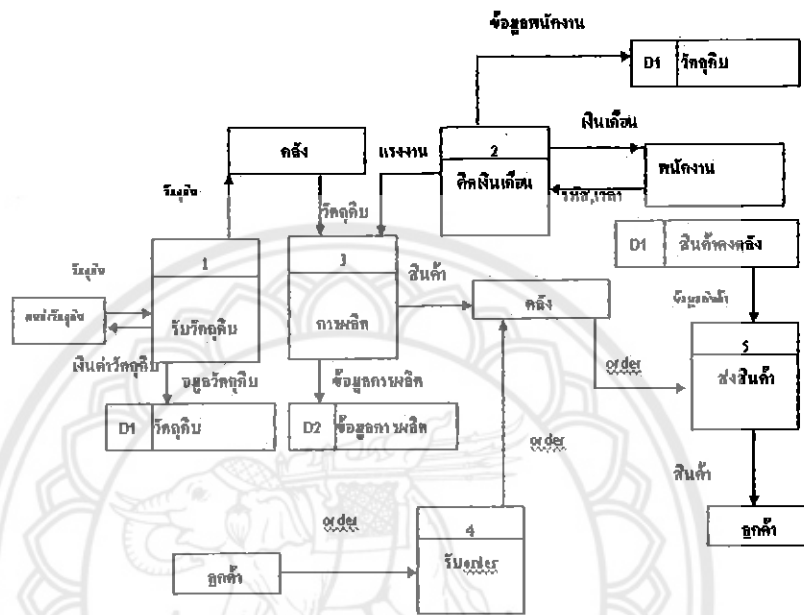


รูปที่ 3.1 Context Diagram

จากรูป Context Diagram แสดงระบบของบริษัท ฯ ซึ่งประกอบไปด้วย Entity ของ พนักงาน คลังสินค้า แหล่งวัตถุดิบและลูกค้า โดยแต่ละ Entity จะมีลูกศรชี้เข้าระบบซึ่งลูกศรนี้ หมายถึง มีการไหลของข้อมูลเข้าสู่ระบบ ส่วนลูกศรที่ชี้ออกหมายถึงผลที่ได้จากการดำเนินงาน

ของระบบ เช่น จากรูป 3.1 Entity แหล่งวัตถุดิบจะมีวัตถุดิบที่จำเป็นต่อการผลิตไหลเข้าสู่ระบบของบริษัท ส่วนเงินค่าวัตถุดิบคือสิ่งที่โรงงานต้องให้กับแหล่งวัตถุดิบ เป็นต้น

3.1.2 Data Flow Diagram Level 0 เป็นการวิเคราะห์ระบบว่าในระบบนั้นควรมีกระบวนการหลักใดและมีข้อมูลใดไหลเข้าหรือออกจากกระบวนการใดบ้าง ในแต่ละกระบวนการ (Process) หลักนี้จะแยกย่อยลงไปอีกใน Data Flow Diagram Level 1

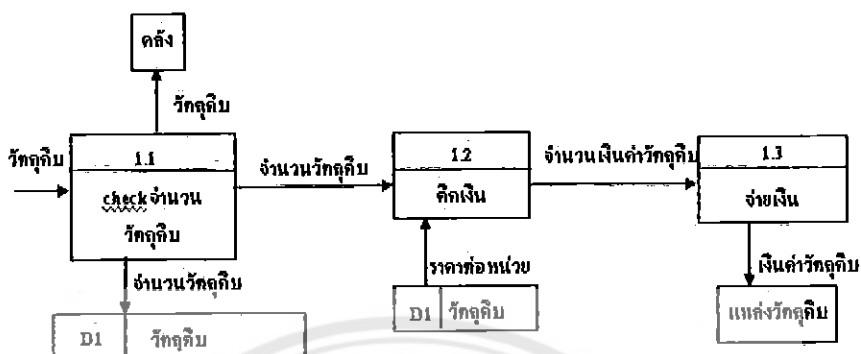


รูปที่ 3.2 Data Flow Diagram ของระบบLevel 0

จากรูป 3.2 เป็นการแสดงการทำงานโดยรวมที่มีอยู่ในระบบหลัก ๆ ของบริษัท ซึ่งประกอบไปด้วย Process ต่าง ๆ ได้แก่ รับวัตถุดิบ คิดเงินเดือน การผลิต รับorderและส่งสินค้า โดยในแต่ละProcess ก็จะมีการดำเนินการที่จำเป็นสำหรับProcess นั้น ๆ

3.1.3 Data Flow Diagram Level 1 ใน Level 1 นี้เป็นการวิเคราะห์กระบวนการทำงานในแต่ละกระบวนการให้ละเอียดกว่า Level 0 โดยในแต่ละกระบวนการใน Level 0 จะมีการแยกเป็นกระบวนการย่อยๆ อีกใน Level 1 ซึ่งเมื่อนำมาเขียนเป็นแผนภาพแล้วเราจะเขียนทีละกระบวนการหลัก ซึ่งมีการวิเคราะห์ในกระบวนการต่าง ๆ ดังนี้

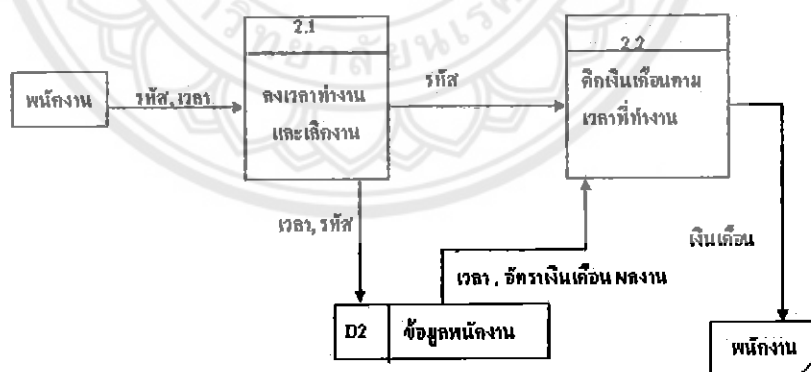
Level 1 Process1 การรับวัตถุดิบ



รูปที่ 3.3 Data Flow Diagram ของระบบ ใน Level1 Process 1 การรับวัตถุดิบ

จากรูป 3.3 เป็นการแสดง Data Flow Diagram ของระบบ ใน Level1 Process 1 การรับวัตถุดิบ โดยแยกมาจาก Level 0 ใน Process1 รับวัตถุดิบ โดยใน Level นี้จะกล่าวถึงการรับวัตถุดิบ โดยละเอียด ซึ่งในการรับวัตถุดิบแต่ละครั้งจะประกอบไปด้วย check จำนวนวัตถุดิบ คิดเงิน และ จ่ายเงิน โดยวัตถุดิบที่ได้มาจะถูกนำไปเก็บไว้ในคลังเพื่อรอการเรียกใช้ต่อไป

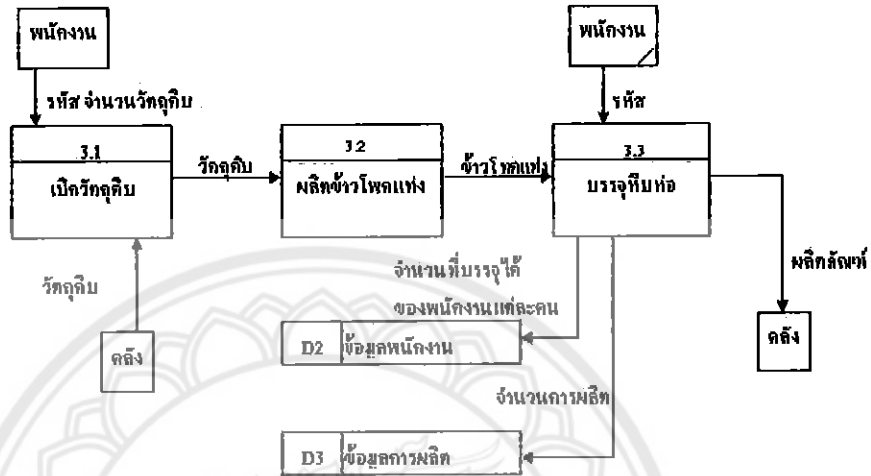
Level 1 Process 2 การคิดเงินเดือนพนักงาน



รูปที่ 3.4 Level 1 Process 2 การคิดเงินเดือนพนักงาน

จากรูปที่ 3.4 แสดงถึง Process ในการคิดเงินเดือนของพนักงาน เมื่อพนักงานมาทำงาน ต้องทำการลงเวลาที่มาทำงานและเลิกงาน โดยข้อมูลที่ได้จะถูกเก็บเอาไว้ในฐานข้อมูลเพื่อนำไปใช้ในการคำนวณเงินเดือน

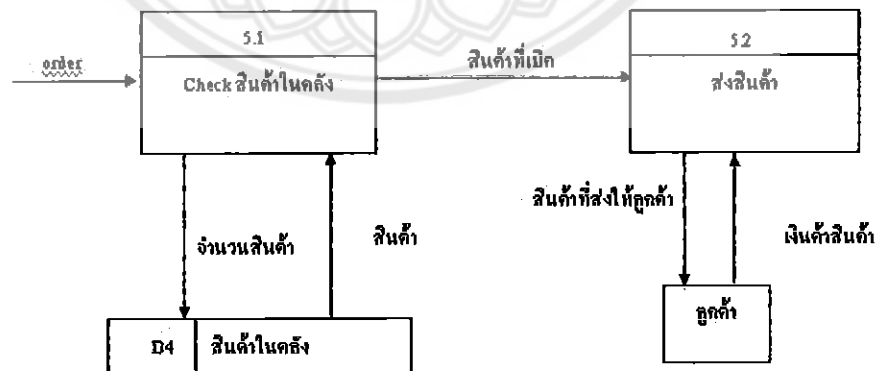
Level 1 Process 3 การผลิตภณช์



รูปที่ 3.5 Level 1 Process 3 การผลิตภณช์

จากรูป 3.5 แสดงถึง Process การผลิต โดยเมื่อต้องการใช้วัดดูดิบพนักงานต้องทำการเปิดวัดดูดิบเพื่อนำไปผลิตสินค้า เมื่อได้สินค้าออกมาแล้วจึงนำไปทำการบรรจุหีบห่อแล้วนำไปเก็บไว้ที่คลังเพื่อรอการจำหน่าย

Level 1 Process 5 คลังสินค้า



รูปที่ 3.6 Level 1 Process 5 คลังสินค้า

จากรูป 3.6 แสดง Process คลังสินค้า โดยเมื่อได้รับ order จากร้านค้าก็จะทำการตรวจสอบสินค้าในคลัง แล้วจึงทำการส่งสินค้าในคลังไปยังลูกค้า

3.2 Data Dictionary หากผู้อ่านไม่เข้าใจบางส่วนใน Data Flow Diagram ว่าแต่ละส่วนมีรายละเอียดอย่างไร ในหัวข้อนี้จะช่วยอธิบายส่วนต่าง ๆ ของ Data Flow Diagram ว่าในส่วนนั้นทำงานอย่างไร มีรายละเอียดอะไรบ้าง ใน Data Dictionary นี้ประกอบไปด้วย Data Dictionary ของ Entity และ Data Dictionary ของ กระบวนการ (Process)

3.2.1 Data Dictionary ของ Entity แสดงรายละเอียดต่างๆ ของ Entity ได้แก่ ชื่อที่เขียนในแผนภาพ(Label) ชื่อย่อ(Alias) รายละเอียดของ Entity (Description) หมายเหตุ (Note) และชื่อเต็มของ Entity (LongName)จากแผนภาพการไหลข้อมูลของระบบของบริษัทฯ มีดังต่อไปนี้

Entity พนักงาน

Label	พนักงาน
Alias	-
Description	พนักงานทุกคนที่ทำงานในบริษัท ศิริวานิช(เอส แอนด์ ดับบลิว)จำกัด ซึ่งพนักงานทุกคนเมื่อเข้ามาในระบบจะต้อง ใ้รหัส เวลาที่มาทำงาน และทำงาน โดยได้รับผลตอบแทนเป็นเงินเดือน
Notes	-
LongName	พนักงานบริษัท ศิริวานิช(เอส แอนด์ ดับบลิว)จำกัด

Entity คลังสินค้า

Label	คลังสินค้า
Alias	-
Description	เป็นที่เก็บสินค้า และวัตถุดิบของบริษัท ศิริวานิช จำกัด
Notes	-
LongName	คลังสินค้าของบริษัท ศิริวานิช จำกัด

Entity แหล่งวัตถุดิบ

Label	แหล่งวัตถุดิบ
Alias	-
Description	คือร้านค้าที่ขายวัตถุดิบให้กับทางบริษัท ฯ เพื่อนำไปผลิตสินค้าต่าง ๆ
Notes	-
LongName	=

Entity ลูกค้า

Label	ลูกค้า
Alias	-
Description	ร้านค้าปลีกที่รับสินค้าที่ผลิตจากบริษัท ไปขาย
Notes	-
LongName	ลูกค้าของบริษัท ศิริวานิช (เอส แอนด์ ดับเบิ้ลยู) จำกัด

3.2.2 Data Dictionary ของ Process แสดงรายละเอียดของแต่ละกระบวนการ ได้แก่ หมายเลขของกระบวนการ (No.) ชื่อกระบวนการ (Name) จุดมุ่งหมายของกระบวนการ (Purpose) ข้อมูลที่ไหลเข้ากระบวนการ (Input Data Flows) ข้อมูลที่ไหลออกจากกระบวนการ (Output Data Flows) และรายละเอียดของกระบวนการ (Process Description) ซึ่งจากแผนภาพการไหลของข้อมูลของบริษัทจะได้ Data Dictionary ของ Process ดังรูป

กระบวนการที่ 1 รับวัตถุดิบ

No.	1
Name	รับวัตถุดิบ
Purpose	เพื่อซื้อวัตถุดิบมาเก็บไว้ในคลัง
Input Data Flows	วัตถุดิบ
Output Data Flows	เงิน ข้อมูลวัตถุดิบ
Process Description	เป็นกระบวนการซื้อวัตถุดิบเพื่อไปผลิตสินค้าให้กับบริษัท ฯ ต่อไป

กระบวนการที่ 1.1 checkจำนวนวัตถุดิบ

No.	1.1
Name	checkจำนวนวัตถุดิบ
Purpose	เพื่อให้บริษัทได้รับวัตถุดิบเป็นจำนวนที่ถูกต้อง
Input Data Flows	วัตถุดิบ
Output Data Flows	วัตถุดิบ จำนวนวัตถุดิบ
Process Description	เป็นกระบวนการตรวจสอบจำนวนวัตถุดิบให้กับทางบริษัท ฯ

กระบวนการที่ 1.2 คิดเงินค่าวัตถุดิบ

No.	1.2
Name	คิดเงินค่าวัตถุดิบ
Purpose	เพื่อทราบจำนวนเงินที่ต้องจ่ายเงินค่าวัตถุดิบให้กับทางแหล่งวัตถุดิบ
Input Data Flows	จำนวนวัตถุดิบ ราคาต่อหน่วย
Output Data Flows	จำนวนเงินค่าวัตถุดิบ
Process Description	เป็นกระบวนการคำนวณเงินค่าวัตถุดิบที่ซื้อมา เพื่อจ่ายให้กับทางแหล่งวัตถุดิบ

กระบวนการที่ 1.3 จ่ายเงินค่าวัตถุดิบ

No.	1.3
Name	จ่ายเงินค่าวัตถุดิบ
Purpose	จ่ายเงินให้ถูกต้องกับจำนวนวัตถุดิบที่ซื้อมา ให้กับแหล่งวัตถุดิบ
Input Data Flows	จำนวนเงินค่าวัตถุดิบ
Output Data Flows	เงินค่าวัตถุดิบ
Process Description	เป็นกระบวนการจ่ายเงินค่าวัตถุดิบให้กับแหล่งวัตถุดิบ โดยได้รับข้อมูลจำนวนเงินมาจากProcess คิดเงิน

กระบวนการที่ 2 คิดเงินเดือน

No.	2
Name	คิดเงินเดือน
Purpose	เพื่อคิดค่าแรงให้กับพนักงาน ในบริษัท ฯ อย่างถูกต้อง
Input Data Flows	รหัส เวลา
Output Data Flows	เงินเดือน ข้อมูลพนักงาน แรงงาน
Process Description	เป็นกระบวนการคิดเงินเดือนให้พนักงาน โดยคิดตามชั่วโมงที่ทำงาน และผลงานที่ได้

กระบวนการที่ 2.1 ลงเวลาทำงานและเลิกงาน

No.	2.1
Name	ลงเวลาทำงานและเลิกงาน
Purpose	นำเวลาที่ได้ไปคิดเงินเดือน
Input Data Flows	รหัส เวลา
Output Data Flows	เวลา, รหัส
Process Description	เป็นกระบวนการที่เก็บข้อมูลว่าในแต่ละวันพนักงานมาทำงานและเลิกงานเมื่อไร แล้วนำไปเก็บในฐานข้อมูลเพื่อนำไปคิดเงินเดือน

กระบวนการที่ 2.2 คิดเงินเดือนตามเวลาที่ทำงาน

No.	2.2
Name	คิดเงินเดือนตามเวลาที่ทำงาน
Purpose	เพื่อจ่ายเงินเดือนให้พนักงานตามการทำงานที่แท้จริง
Input Data Flows	เวลา อัตราเงินเดือน ผลงาน รหัส
Output Data Flows	เงินเดือน
Process Description	เป็นกระบวนการคิดเงินเดือนพนักงาน ตามเวลาที่มาทำงาน ผลงานที่ทำไว้ อัตราเงินเดือนในแต่ละแผนก

กระบวนการที่ 3 การผลิต

No.	3
Name	การผลิต
Purpose	เพื่อผลิตสินค้าให้กับบริษัท ฯ
Input Data Flows	วัตถุดิบ แรงงาน
Output Data Flows	สินค้า ข้อมูลการผลิต
Process Description	เป็นกระบวนการผลิตสินค้าให้กับทางบริษัท ฯ

กระบวนการที่ 3.1 เบิกวัตถุดิบ

No.	3.1
Name	เบิกวัตถุดิบ
Purpose	เบิกจำนวนวัตถุดิบให้เป็นระบบ
Input Data Flows	รหัส จำนวนวัตถุดิบที่ต้องการ วัตถุดิบ
Output Data Flows	วัตถุดิบ
Process Description	เป็นกระบวนการเบิกจำนวนวัตถุดิบเพื่อนำไปใช้ในการผลิต

กระบวนการที่ 3.2 ผลิตข้าวโพดแห้ง

No.	3.2
Name	ผลิตข้าวโพดแห้ง
Purpose	ผลิตข้าวโพดแห้ง เพื่อนำไปปรุงรส และบรรจุหีบห่อต่อไป
Input Data Flows	วัตถุดิบ
Output Data Flows	ข้าวโพดแห้ง
Process Description	เป็นการผลิตข้าวโพดแห้ง โดยนำวัตถุดิบไปเข้าเครื่องจักรแล้วจะได้ข้าวโพดแห้งออกมา

กระบวนการที่ 3.3 บรรจุหีบห่อ

No.	3.3
Name	บรรจุหีบห่อ
Purpose	เพื่อให้ได้ผลิตภัณฑ์ที่สมบูรณ์ สะอาด มีคุณภาพ
Input Data Flows	ข้าวโพดแห้ง รหัสพนักงาน
Output Data Flows	จำนวนที่บรรจุได้ จำนวนการผลิต ผลิตภัณฑ์
Process Description	เป็นกระบวนการบรรจุหีบห่อ โดยนำข้าวโพดแห้งที่ได้มาบรรจุลงในถุง แล้วจะมีการบันทึกจำนวนการผลิตด้วย

กระบวนการที่ 4 รับorder

No.	4
Name	รับorder
Purpose	รับขอการสั่งซื้อสินค้าจากลูกค้า
Input Data Flows	order
Output Data Flows	order
Process Description	เป็นกระบวนการรับขอการสั่งซื้อจากลูกค้า เพื่อจัดส่งสินค้าให้ลูกค้าได้อย่างถูกต้อง

กระบวนการที่ 5 ส่งสินค้า

No.	5
Name	ส่งสินค้า
Purpose	ให้ลูกค้าได้รับสินค้าจากทางบริษัท
Input Data Flows	ข้อมูลสินค้า order
Output Data Flows	สินค้า
Process Description	เป็นกระบวนการส่งสินค้าให้กับลูกค้าตามขอการสั่งซื้อที่ลูกค้าต้องการ

กระบวนการที่ 5.1 checkสินค้าในคลัง

No.	5.1
Name	check สินค้าในคลัง
Purpose	เพื่อให้ทราบว่าสินค้าในคลังมีเพียงพอต่อการเบิกไปขายหรือไม่
Input Data Flows	order สินค้า
Output Data Flows	จำนวนสินค้า สินค้า
Process Description	เป็นกระบวนการตรวจสอบว่ามีสินค้าพอที่จะเบิกไปขายหรือไม่

กระบวนการที่ 5.2 ส่งสินค้า

No.	5.2
Name	ส่งสินค้า
Purpose	เพื่อให้สินค้าไปส่งไปถึงลูกค้า
Input Data Flows	สินค้าที่เบิก
Output Data Flows	สินค้าที่ส่ง เงินค่าสินค้า
Process Description	เป็นการส่งสินค้าตามที่ลูกค้าต้องการ

บทที่ 4

การพัฒนาโปรแกรม

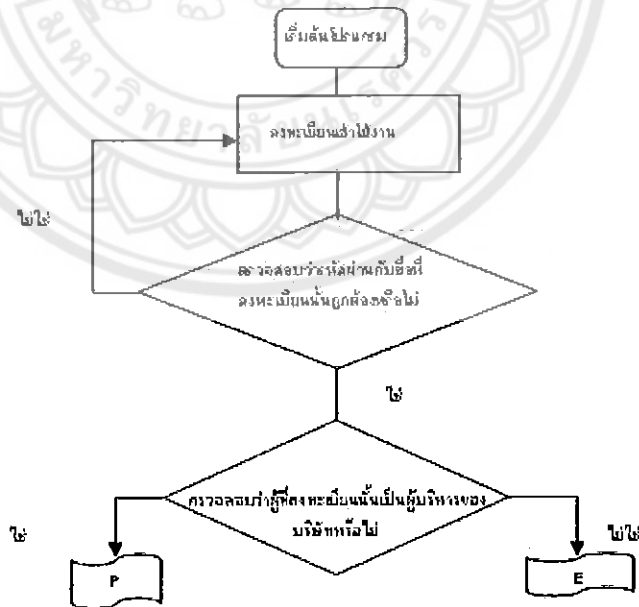
การจัดการระบบสารสนเทศของโครงการนั้น เมื่อได้ทำการวิเคราะห์และออกแบบสร้างระบบฐานข้อมูลแล้ว จะต้องมีการพัฒนาโปรแกรมเพื่อให้โปรแกรมมีประสิทธิภาพมากขึ้น โดยจะมีการเขียนอธิบายอยู่ในรูปของ แผนภาพการพัฒนาโปรแกรม (Flow Chart) ซึ่งจะช่วยในการอธิบายการทำงานของโปรแกรม และช่วยในการพัฒนาโปรแกรมให้ดียิ่งขึ้น โดยโครงการนี้ได้เขียน แผนภาพการพัฒนาโปรแกรม (Flow Chart) เพื่อให้เข้าใจหลักการการทำงานของโครงการให้เข้าใจมากขึ้น

4.1 แผนภาพการพัฒนาโปรแกรม

แผนภาพ การพัฒนาโปรแกรม หรือ FlowChart เป็นตัวช่วยอธิบาย การทำงานของโปรแกรม ให้ง่ายขึ้น โดยใช้รูปเรขาคณิตและเส้นตรงมาอธิบาย โดยแบ่งโครงการเป็น ส่วนต่าง ๆ ดังรูป

4.1.1 การลงทะเบียนเข้าใช้โปรแกรม

เมื่อผู้ใช้เข้าใช้โปรแกรมจะต้องลงทะเบียนเข้าใช้ทุกครั้งเพื่อป้องกันบุคคลภายนอกลักลอบดูข้อมูลของบริษัท โดยฟังก์ชันของการลงทะเบียนเป็นดังรูป

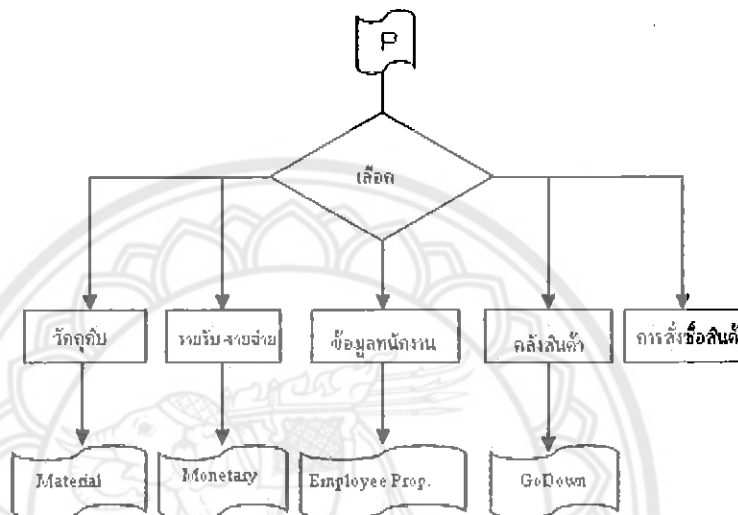


รูปที่ 4.1 แผนภาพโปรแกรมลงทะเบียนเข้าใช้โปรแกรม

จากรูป 4.1 แสดงการเข้าใช้ระบบ เมื่อเริ่มใช้โปรแกรมผู้ใช้ต้องทำการลงทะเบียนเข้าใช้ และระบบจะทำการตรวจสอบว่าผู้ใช้เป็นผู้ใช้ที่ถูกต้องหรือไม่ ถ้าถูกต้องระบบก็จะตรวจสอบว่าเป็นผู้บริหารของบริษัทหรือพนักงานทั่วไป เพื่อเรียกใช้งานในโปรแกรมที่ต่างกัน

4.1.2 โปรแกรมในส่วนของผู้บริหารกิจการ

เมื่อผู้ใช้เข้ามาใช้โปรแกรมแล้ว หากเป็นผู้บริหารก็จะสามารถเข้ามาใช้โปรแกรมในส่วนนี้ได้ ซึ่งผังการทำงานของโปรแกรมในส่วนนี้เป็นดังรูปที่ 4.2

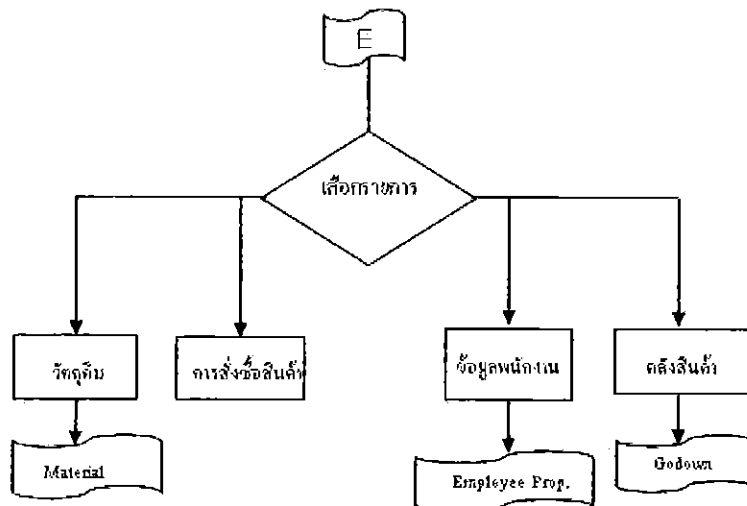


รูปที่ 4.2 แผนภาพโปรแกรมในส่วนของผู้บริหาร

จากรูปที่ 4.2 แสดงถึงการทำงานเมื่อทำการลงทะเบียนแล้วในส่วนของผู้บริหาร โดยจะมีโปรแกรมหลักให้เลือกใช้งานดังนี้ วัตถุดิบ รายรับ-รายจ่าย ข้อมูลพนักงาน คลังสินค้าและการสั่งซื้อสินค้า โดยแต่ละโปรแกรมก็จะมีการทำงานแยกย่อยลงไปอีก

4.1.3 โปรแกรมส่วนพนักงาน

จากรูปที่ 4.1 หากผู้ใช้ลงทะเบียนเข้ามาโดยที่ไม่ใช่ผู้บริหารของบริษัท ก็จะเข้ามาในโปรแกรมส่วนของพนักงาน ซึ่งในรายการ โปรแกรมของพนักงานนี้จะคล้าย ๆ กับโปรแกรมในส่วนของผู้บริหาร แตกต่างกันเพียงโปรแกรมในส่วนพนักงานจะไม่มีเรื่องของการเงิน ดังรูปที่ 4.3

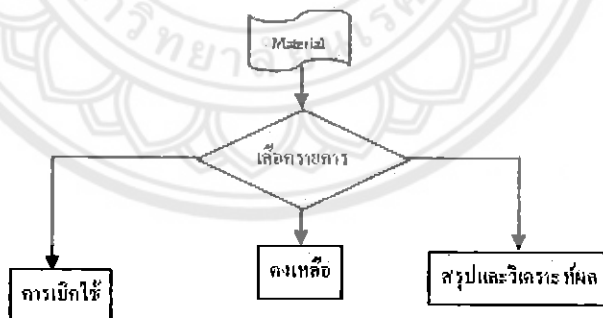


รูปที่ 4.3 แผนภาพโปรแกรมในส่วนพนักงาน

จากรูปที่ 4.3 แสดงถึงการทำงานเมื่อทำการลงทะเบียนแล้วในส่วนของพนักงาน โดยจะมีโปรแกรมหลักให้เลือกใช้งานดังนี้ วัตถุดิบ ข้อมูลพนักงาน คลังสินค้าและการสั่งซื้อสินค้า โดยแต่ละโปรแกรมก็จะมีการทำงานแยกย่อยลงไปเช่นเดียวกับในส่วนของผู้บริหาร จะสังเกตได้ว่าในส่วนของพนักงานไม่มีโปรแกรมในส่วนรายรับ-รายจ่ายเหมือนกับในส่วนของผู้บริหาร

4.1.4 โปรแกรมในส่วนของการจัดการวัตถุดิบ

จากรูป 4.2 หากผู้ใช้โปรแกรมเลือกการจัดการวัตถุดิบ(Material) ก็จะเข้ามาที่โปรแกรมในส่วนนี้ ซึ่งมีการทำงานดังรูปที่ 4.3

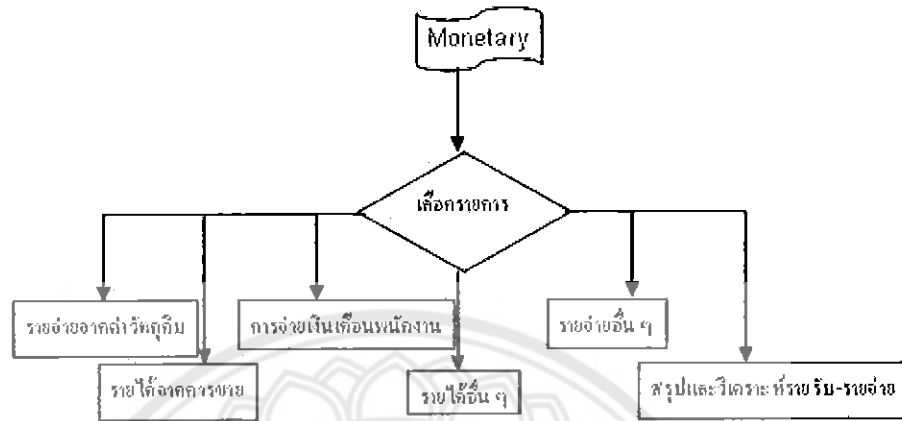


รูปที่ 4.4 แผนภาพโปรแกรมการจัดการวัตถุดิบ

จากรูปที่ 4.4 แสดงการจัดการด้านวัตถุดิบ ในส่วนนี้ของผู้บริหารและพนักงานทั่วไปจะมีการทำงานที่เหมือนกัน ซึ่งได้แก่ การเบิกใช้ แสดงวัตถุดิบที่คงเหลือ สรุปและวิเคราะห์ผล

4.1.5 โปรแกรมในด้านการเงิน

โปรแกรมในส่วนนี้มีเพียงผู้บริหารเท่านั้นที่จะเข้ามาใช้ได้ ซึ่งเกี่ยวกับรายรับรายจ่ายต่าง ๆ ของทางบริษัท ทำให้ผู้บริหารสามารถวางแผนการผลิตและจัดการภายในบริษัทได้ต่อไป

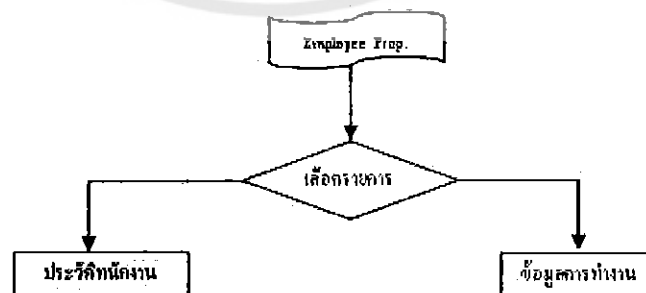


รูปที่ 4.5 แผนภาพโปรแกรมการจัดการด้านการเงิน

จากรูปที่ 4.5 แสดงโปรแกรมการจัดการด้านการเงิน ซึ่งผู้บริหารเท่านั้นที่จะสามารถเข้าใช้โปรแกรมในส่วนนี้ โดยสามารถเรียกดูรายจ่ายจากค่าวัสดุ รายได้จากการขาย การจ่ายเงินเดือนพนักงาน รายได้อื่น ๆ รายจ่ายอื่น ๆ และมีการสรุปลวิเคราะห์รายรับ-รายจ่ายด้วย

4.1.6 โปรแกรมในส่วนของข้อมูลพนักงาน

เป็นโปรแกรมที่ทำให้เราทราบประวัติพนักงานแต่ละคน การเข้ามาทำงานของพนักงานแต่ละคน โดยมีการทำงานดังแผนภาพการพัฒนาโปรแกรมดังรูป 4.6

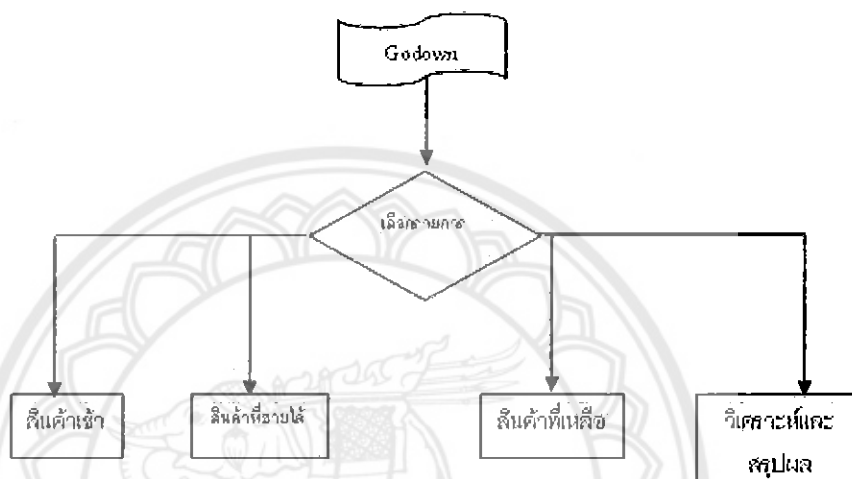


รูปที่ 4.6 แผนภาพโปรแกรมข้อมูลพนักงาน

จากรูปที่ 4.6 แสดงข้อมูลของพนักงาน ซึ่งในโปรแกรมนี้จะสามารถแสดงประวัติพนักงาน และข้อมูลการทำงานของพนักงานได้ โดยใช้รหัสพนักงาน

4.1.7 โปรแกรมด้านคลังสินค้า

เป็นโปรแกรมส่วนคลังสินค้าว่ามีสินค้าเหลือเท่าใด ผลิตสินค้าได้เท่าใด มีการวิเคราะห์และสรุปผล ดังรูปที่ 4.7



รูปที่ 4.7 แผนภาพโปรแกรมคลังสินค้า

จากรูปที่ 4.7 แสดงการทำงานของโปรแกรมคลังสินค้า ซึ่งประกอบไปด้วยการแสดงผลสินค้าเข้า สินค้าที่ขายได้ สินค้าที่เหลืออยู่ และสรุปวิเคราะห์ผลการขาย

4.2 รูปแบบโปรแกรม โดยจะมีรูปแบบโปรแกรมต่างๆ โดยแบ่งเป็น 3 ประเภท

4.2.1 ระบบบริหารฝ่ายบุคคล

ระบบบริหารฝ่ายบุคคลแบ่งออกเป็น 2 ส่วน คือ

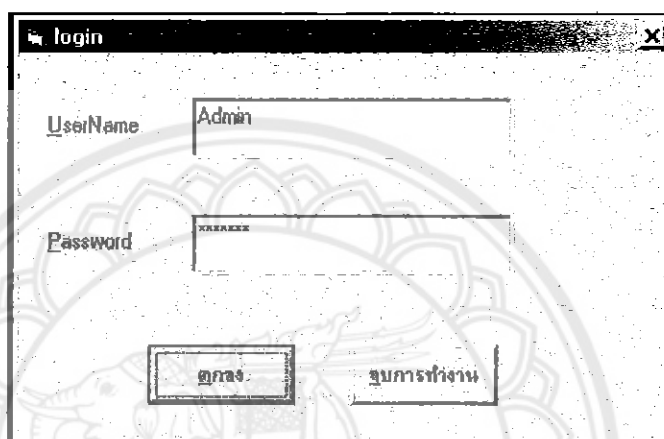
4.2.1.1 ระบบฝ่ายข้อมูลบุคคล เป็นระบบที่เก็บข้อมูลต่างของพนักงานทั้งชื่อนามสกุล ตำแหน่ง แผนก และ หน้าที่การ รวมทั้ง บันทึกการทำงานประจำวัน ขาด ลา มาสาย การทำงานของแต่ละแผนก และข้อมูลดังกล่าวก็จะส่งต่อไปทางด้าน ระบบฝ่ายการคิดเงินเดือน

4.2.1.2 ระบบฝ่ายการคิดเงินเดือน เป็นระบบคิดเงินเดือนที่รับข้อมูลจากระบบฝ่ายบุคคลมาคิดเป็นเงินเดือน โดยการคิดเงินเดือนของพนักงานนั้นแต่ละแผนกจะต่างออกไป โดยจะสรุปออกมาทุกๆเดือน

- โปรแกรมระบบฝ่ายบริหารบุคคล แบ่งเป็น 2 ส่วน คือ โปรแกรมระบบฝ่ายข้อมูลบุคคล และ โปรแกรมคิดเงินเดือน โดยมีรายละเอียดดังนี้

➤ โปรแกรมระบบฝ่ายข้อมูลบุคคล

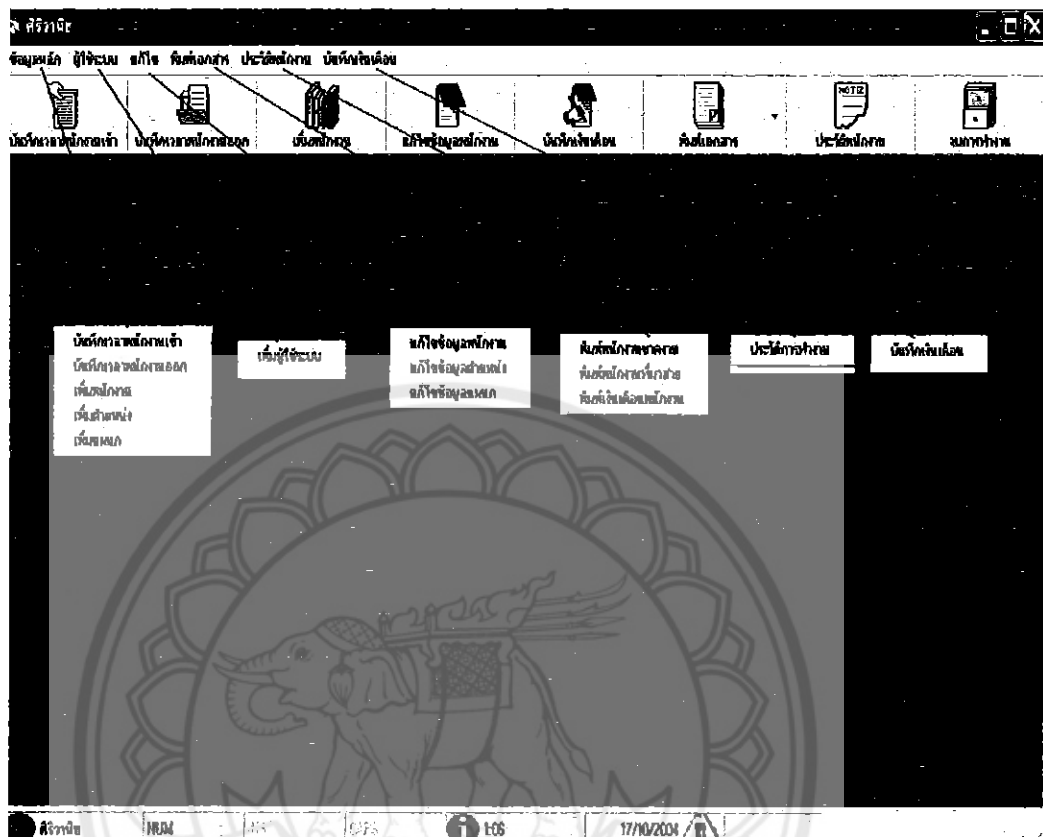
- การกรอกรหัสผ่าน โดยใส่ Username และ Password



รูปที่ 4.8 โปรแกรมลงทะเบียนเข้าใช้งาน

การใช้โปรแกรมทุกครั้งต้องมีการกรอกรหัสผ่านเพื่อที่จะใช้งานก่อนการใช้งาน เมื่อกรอกเรียบร้อยแล้ว เข้าสู่การใช้โปรแกรม โดยจะมีหน้าต่างหลักแสดงออกมา หากผู้ใช้กรอก Username และ Password ผิดก็จะมีการบันทึกว่ามีการล็อกอินใช้โปรแกรมเมื่อใดบ้าง

○ หน้าหลักของโปรแกรมระบบฝ่ายบริหารบุคคล



รูปที่ 4.9 หน้าหลักของโปรแกรมระบบฝ่ายบริหารบุคคล

โปรแกรมหลักเป็นศูนย์รวมของการทำงานต่างๆ ได้แก่ การบันทึกเวลาพนักงานเข้า-ออก การเพิ่มข้อมูลพนักงาน การแก้ไขข้อมูลพนักงาน การเพิ่มข้อมูลและแก้ไขตำแหน่ง และ แผนกของพนักงาน การเพิ่มผู้ใช้งาน การบันทึกเงินเดือนเป็นรายวัน-รายเดือน ประวัติพนักงาน การพิมพ์เอกสารรายงานเงินเดือน ข้อมูลการทำงานพนักงาน และรายงานเงินเดือน

○ การบันทึกเวลาเข้า-ออกพนักงาน โดยจะมีแบบฟอร์ม 2 แบบ บันทึก
ทั้งเวลาเข้า-ออก

รูปที่ 4.10 โปรแกรมบันทึกเวลาเข้าทำงานของพนักงาน

รูปที่ 4.11 โปรแกรมบันทึกเวลาออกทำงานของพนักงาน

การบันทึกเวลาเข้า-ออกของพนักงาน จะบันทึกวันเวลาที่พนักงานเข้า-ออกทำงานเพื่อทราบการขาด ลา มาสาย และ จำนวนชั่วโมงการทำงาน โดยจะกรอกรหัสพนักงานและทำการค้นหา ข้อมูลของพนักงานและทำการบันทึกวัน เวลาการทำงานเข้า-ออก

○ การเพิ่มข้อมูลพนักงาน

ข้อมูลทั่วไป		ตำแหน่ง		ชื่อ-นามสกุล		รูปพนักงาน
รหัสพนักงาน	คำนำหน้า	ชื่อ-นามสกุล				
9789748767956	นาย	วิฑูริยา วิฑูริยา				
แผนกงาน	ตำแหน่ง	เพศ				
บริหาร	ผู้ช่วย	ชาย <input checked="" type="radio"/> หญิง <input type="radio"/>				
เลขบัญชีธนาคาร	ประเภทงาน	เต็มเวลา <input checked="" type="radio"/> PartTime <input type="radio"/>				
123456789	ประจำ					
วันเริ่มต้นทำงาน	วันเกิด					
05/10/2002	12/05/1975					
ที่อยู่ บ้านเลขที่	หมู่ที่	ตำบล	อำเภอ			
655	10	คลองโพธิ์	คลองโพธิ์			
จังหวัด	รหัสไปรษณีย์					
สุโขทัย	64150					
เบอร์โทรศัพท์						
055629893						

รูปที่ 4.12 การเพิ่มข้อมูลพนักงาน

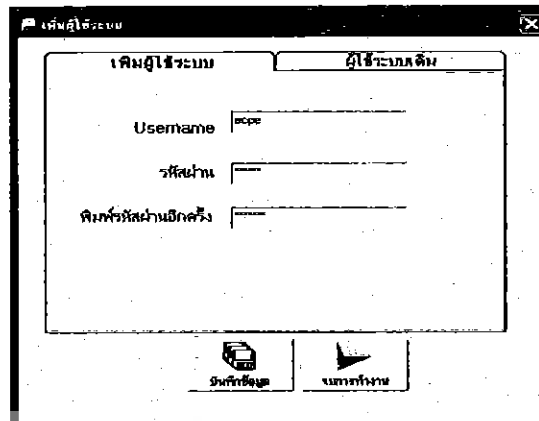
○ การแก้ไขข้อมูลพนักงาน

ข้อมูลทั่วไป		ตำแหน่ง		ชื่อ-นามสกุล		รูปพนักงาน
รหัสพนักงาน	คำนำหน้า	ชื่อ-นามสกุล				
3789747822346	นาย	วิฑูริยา วิฑูริยา				
แผนกงาน	ตำแหน่ง	เพศ				
บริหาร	ผู้ช่วย	ชาย <input checked="" type="radio"/> หญิง <input type="radio"/>				
เลขบัญชีธนาคาร	ประเภทงาน	เต็มเวลา <input checked="" type="radio"/> PartTime <input type="radio"/>				
034635406456	ประจำ					
วันเริ่มต้นทำงาน	วันเกิด					
02/10/2003	26/03/1960					
ที่อยู่ บ้านเลขที่	หมู่ที่	ตำบล	อำเภอ			
20/18	4	โพธิ์ทอง	โพธิ์ทอง			
จังหวัด	รหัสไปรษณีย์					
สุโขทัย	65000					
เบอร์โทรศัพท์						
055251174						

รูปที่ 4.13 การแก้ไขข้อมูลของพนักงาน

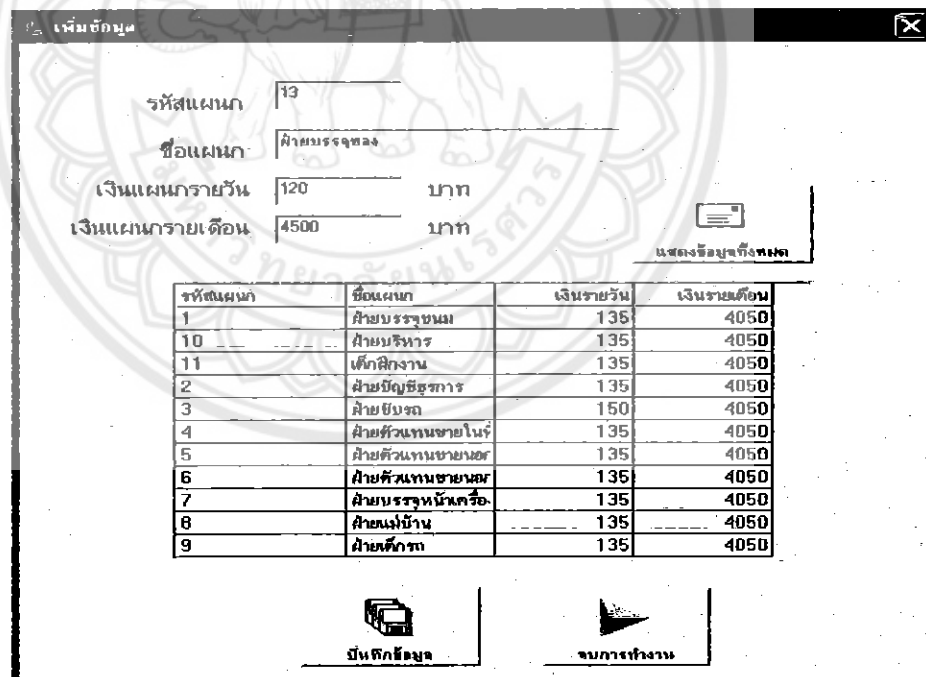
การเพิ่มข้อมูลพนักงาน โดยจะกรอกประวัติของพนักงาน ชื่อ-นามสกุล แผนก ตำแหน่ง หน้าที่การงาน ที่อยู่ เบอร์โทรศัพท์ วันเกิด วันเริ่มต้นทำงาน รูปภาพของพนักงาน และ เลขบัญชีธนาคาร รวมทั้ง การแก้ไขข้อมูลเพื่อความคิดพลาดหรือการเปลี่ยนแปลงข้อมูลพนักงาน โดยมีการค้นหา ทั้งรหัสพนักงานและ ชื่อพนักงาน ทำการแก้ไขและบันทึกข้อมูลลงฐานข้อมูล

- การเพิ่มผู้ใช้งาน เป็นการเพิ่มผู้ใช้โปรแกรมเพื่อทำการกรอกข้อมูล



รูปที่ 4.14 โปรแกรมการเพิ่มผู้ใช้ในระบบ

- การเพิ่มและแก้ไขข้อมูลแผนก เป็นหน้าต่างที่ต้องการที่จะเพิ่ม หรือแก้ไขข้อมูลของ แผนกต่างๆ



รหัสแผนก	ชื่อแผนก	เงินรายวัน	เงินรายเดือน
1	ฝ่ายบรรณกรรม	135	4050
10	ฝ่ายบริหาร	135	4050
11	ศึกษานิเทศก์	135	4050
2	ฝ่ายบัญชีและการเงิน	135	4050
3	ฝ่ายช่างเทคนิค	150	4050
4	ฝ่ายวิศวกรรมช่างไฟฟ้า	135	4050
5	ฝ่ายวิศวกรรมช่างโยธา	135	4050
6	ฝ่ายวิศวกรรมช่างเครื่องกล	135	4050
7	ฝ่ายบรรณวิทยากร	135	4050
8	ฝ่ายแม่บ้าน	135	4050
9	ฝ่ายศึกษานิเทศก์	135	4050

รูปที่ 4.15 โปรแกรมการเพิ่มข้อมูลแผนก

แก้ไขข้อมูลแผนก

รหัส: 1

ชื่อแผนก: ฝ่ายบรรณานุกรม

เงินเดือนรายวัน: 135 บาท

เงินเดือนรายเดือน: 4050 บาท

เดือนเพื่อหาข้อมูล: เรคอร์ดที่ 1 / 11

แก้ไขข้อมูล

แผนกทั้งหมด

รหัสแผนก	ชื่อแผนก	เงินรายวัน	เงินรายเดือน
1	ฝ่ายบรรณานุกรม	135	4050
10	ฝ่ายบริหาร	135	4050
11	ศึกษาดูงาน	135	4050
2	ฝ่ายวิทยุสื่อสาร	135	4050
3	ฝ่ายช่างรถ	150	4050
4	ฝ่ายควบคุมหอโถง	135	4050
5	ฝ่ายควบคุมหอสมุด	135	4050
6	ฝ่ายควบคุมหอสมุด	135	4050
7	ฝ่ายบรรณานุกรม	135	4050
8	ฝ่ายช่าง	135	4050

ออกจาการทำงาน

รูปที่ 4.16 โปรแกรมการแก้ไขข้อมูลของแผนก

○ การเพิ่มและแก้ไขตำแหน่งพนักงาน เป็นหน้าต่างที่ใช้การแก้ไขและเพิ่มข้อมูลเกี่ยวกับตำแหน่งงาน

เพิ่มตำแหน่งพนักงาน

เพิ่มตำแหน่ง: แผนก: ฝ่ายบริหาร

รหัสประจำตำแหน่ง: 101

ชื่อตำแหน่ง: ศึกษาดูงาน

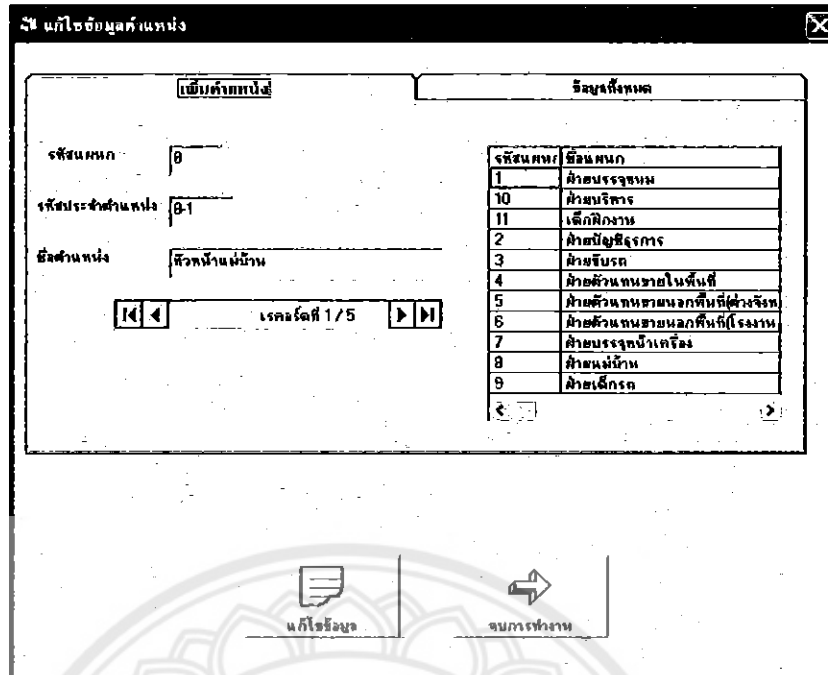
ข้อมูลทั้งหมด

รหัสแผนก	ชื่อแผนก	รหัส
1	ฝ่ายบรรณานุกรม	101
10	ฝ่ายบริหาร	101
11	ศึกษาดูงาน	101
2	ฝ่ายวิทยุสื่อสาร	101
3	ฝ่ายช่างรถ	101
4	ฝ่ายควบคุมหอโถง	101
5	ฝ่ายควบคุมหอสมุด	101
6	ฝ่ายควบคุมหอสมุด	101
7	ฝ่ายบรรณานุกรม	101
8	ฝ่ายช่าง	101

บันทึกข้อมูล

ยกเลิกตำแหน่ง

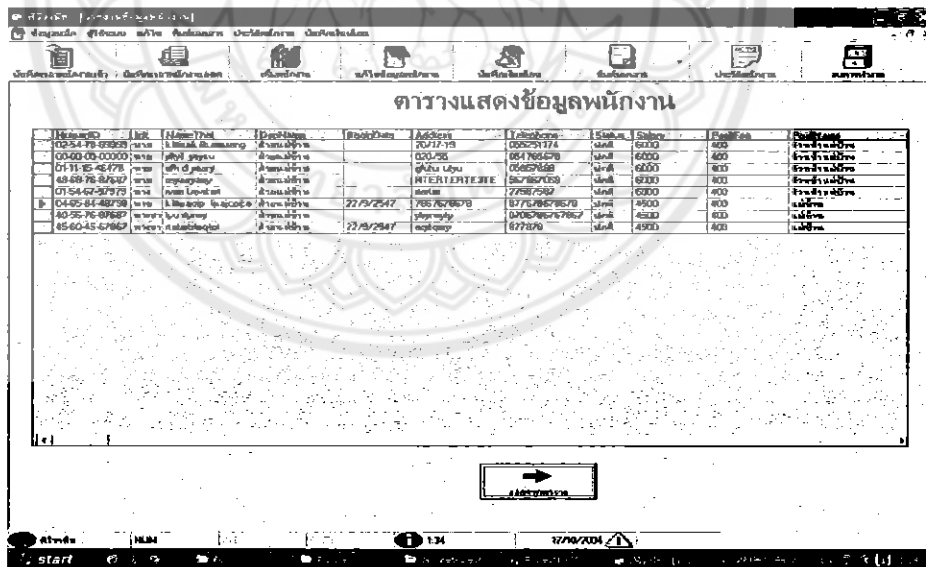
รูปที่ 4.17 โปรแกรมการเพิ่มตำแหน่งของพนักงาน



รูปที่ 4.18 โปรแกรมการแก้ไขข้อมูลของตำแหน่ง

○ การรายงานประวัติพนักงาน แสดงประวัติของพนักงาน ตำแหน่ง

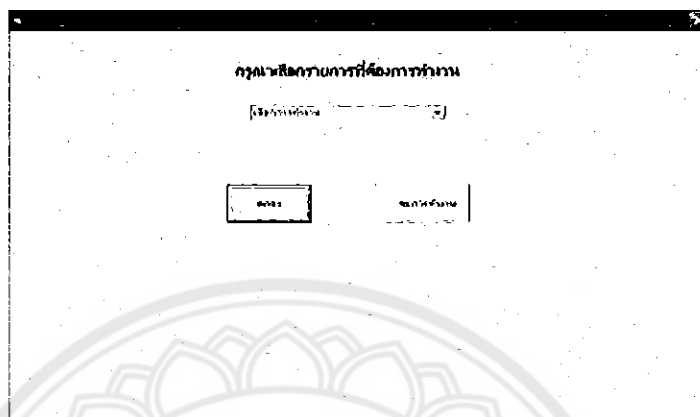
เป็นต้น



รูปที่ 4.19 การรายงานประวัติพนักงาน

4.2.2 โปรแกรมระบบฝ่ายการคิดเงินเดือน

โปรแกรมการคำนวณเงินเดือนพนักงานสามารถแบ่งออกได้ 3 ส่วนหลักๆ คือ การคำนวณเงินค่าแรงสำหรับพนักงานรายวัน, การคำนวณเงินค่าแรงสำหรับพนักงานรายเดือน และการคำนวณเงินค่าแรงรายเดือน(หารายได้สุทธิของเดือนนั้นๆ) สำหรับแต่ละส่วนจะอธิบายในลำดับต่อไป



รูปที่ 4.20 หน้าต่างหลักของโปรแกรมระบบฝ่ายการคิดเงินเดือน

จากรูปที่ 4.20 มีไว้เพื่อให้ผู้ใช้ได้ทำการเลือกการทำงานในส่วนการคิดเงินเดือน โดยมีการทำงานหลักๆดังนี้ การคำนวณรายได้รายวัน, การคำนวณรายได้รายเดือน, การคำนวณรายได้สุทธิประจำเดือน, การพิมพ์รายชื่อพนักงานที่มาสาย, การพิมพ์รายชื่อพนักงานที่ขาดงาน, การพิมพ์เอกสารแสดงรายได้ และการกำหนดอัตราเงินหักค่ามาสาย

○ โปรแกรมการคำนวณเงินค่าแรงของพนักงานรายวัน

รูปที่ 4.21 แสดงการคำนวณเงินค่าแรงสำหรับพนักงานรายวัน

จากรูปที่ 4.21 แสดงถึงหน้าต่างหลักของโปรแกรมสำหรับคิดเงินค่าแรงสำหรับพนักงานรายวัน โดยเมื่อผู้ใช้ต้องการใช้งาน ในขั้นแรกต้องทำการเลือกวันที่ที่ต้องการคำนวณ ต่อมาจึงทำการกรอกรหัสพนักงาน แล้วกดปุ่ม "ตกลง" เพื่อทำการตรวจสอบข้อมูลในฐานข้อมูลว่ามีพนักงานคนนี้อยู่ในฐานข้อมูลหรือไม่ จากนั้นระบบก็จะทำการแสดงระยะเวลาการทำงานของพนักงานในวันที่ทำการคำนวณ ซึ่งผู้ใช้จำเป็นต้องกรอกจำนวนชั่วโมงที่ทำโอที, จำนวนกล่องที่ทำได้ และเงินที่หักในกรณีที่พนักงานขาดงาน แล้วกดปุ่ม "คำนวณค่าแรง" เพื่อดูจำนวนเงินที่ได้ในวันนั้น แล้วจึงทำการบันทึกข้อมูลลงฐานข้อมูล

○ โปรแกรมการคำนวณเงินค่าแรงของพนักงานรายเดือน

ข้อมูลพนักงาน	จำนวนชั่วโมงที่ทำงาน (รวมเวลาพัก)	หักค่าจ้างขาด (บาท)
รหัสพนักงาน: 9769747822946	25	0
ชื่อ-นามสกุล: วิไลดา โสมศิริ	จำนวนชั่วโมง OT: 0	เงินค่า OT: 2005
ตำแหน่ง: นาง	จำนวนกล่องที่ทำได้: 5	เงินค่ากล่อง: 271
ตำแหน่งเงินเดือน: 4050	หักค่าขาดงาน(บาท): 0	รวมค่าจ้าง: 271

บันทึกฐานข้อมูล จบการคำนวณ

บริษัทวิโรจน์ (เอส.แอล.ดี.บี.เอส) จำกัด ระบบการคิดการฐานข้อมูล (โปรแกรมคำนวณเงินเดือนรายเดือน) NUM IHS 17/10/2004 18:01

รูปที่ 4.22 แสดงการคำนวณเงินค่าแรงสำหรับพนักงานรายเดือน

จากรูปที่ 4.22 แสดงถึงหน้าต่างหลักของโปรแกรมสำหรับคิดเงินค่าแรงสำหรับพนักงานรายเดือน โดยเมื่อผู้ใช้ต้องการใช้งาน ในขั้นแรกต้องทำการเลือกวันที่ที่ต้องการคำนวณ ต่อมาจึงทำการกรอกรหัสพนักงาน แล้วกดปุ่ม "ตกลง" เพื่อทำการตรวจสอบข้อมูลในฐานข้อมูลว่ามีพนักงานคนนี้อยู่ในฐานข้อมูลหรือไม่ จากนั้นระบบก็จะทำการแสดงระยะเวลาการทำงานของพนักงานในวันที่ทำการคำนวณ ซึ่งผู้ใช้จำเป็นต้องกรอกจำนวนชั่วโมงที่ทำโอที, จำนวนกล่องที่ทำได้ และเงินที่หักในกรณีที่พนักงานขาดงาน แล้วกดปุ่ม "แสดงผล" เพื่อดูจำนวนเงินที่คำนวณได้เพื่อต้องนำไปเก็บลงฐานข้อมูล แล้วจึงทำการบันทึกข้อมูลลงฐานข้อมูล

○ โปรแกรมการคำนวณเงินค่าแรงสุทธิประจำเดือน

โปรแกรมคำนวณรายได้สุทธิประจำเดือน

เดือนที่: 12 ปี: 2567

วันที่: 15 ตุลาคม 2567

ข้อมูลพนักงาน:

รหัสพนักงาน: 0000003

ชื่อ-นามสกุล: ศิริวันสุรางค์

ตำแหน่ง: พนักงานประจำ

ตำแหน่ง: พนักงาน

เงินเดือนสุทธิ:

เงินเดือน: 7500

เงินค่า OT: 250

เงินรวมก่อน: 0

ผลประโยชน์: 0

เงินเดือน: 1000

เงินประกันสังคม: 200

เงินได้มียก: 0

รายได้สุทธิ:

8025

ข้อมูลการหักเงิน:

ค่าจ้างประจำเดือน: 30

ค่าล่วงเวลา: 20

ค่าแรงอื่น: 500

เงินค่าประกันสังคม: 5

เงินประกันสุขภาพ: 0

ค่าสมาชิกประกันสังคม: 0

คำนวณเงินได้สุทธิ

วันที่: 17/10/2567 16:05

รูปที่ 4.23 แสดงการคำนวณเงินค่าแรงสุทธิประจำเดือน

จากรูปที่ 4.23 แสดงถึงหน้าต่างหลักของโปรแกรมสำหรับคิดเงินค่าแรงสุทธิประจำเดือน โดยเมื่อผู้ใช้ต้องการใช้งาน ในขั้นแรกต้องทำการเลือกวันที่ที่ต้องการคำนวณ ต่อมาจึงทำการกรอกรหัสพนักงาน แล้วกดปุ่ม "ตกลง" เพื่อทำการตรวจสอบข้อมูลในฐานข้อมูลว่ามีพนักงานคนนี้อยู่ในฐานข้อมูลหรือไม่ จากนั้นระบบก็จะทำการแสดงข้อมูลที่จำเป็นต่อการคำนวณ และผู้ใช้จำเป็นต้องกรอกข้อมูลที่ต้องการนำไปคำนวณเงินค่าแรงสุทธิลงไปในช่วงที่กำหนดให้ แล้วกดปุ่ม "คำนวณเงินเดือน" เพื่อดูจำนวนเงินที่คำนวณได้เพื่อต้องนำไปเก็บลงฐานข้อมูล แล้วจึงทำการบันทึกข้อมูลลงฐานข้อมูล

○ โปรแกรมพิมพ์รายชื่อพนักงานที่มาทำงานสาย

วันที่	เลขบัตรประชาชน	ชื่อพนักงาน	ชื่อ
2547-10-15	00000001	นาย	นายสมชาย ใจดี
2547-10-15	00000002	นาย	นายสมชาย ใจดี

รูปที่ 4.24 หน้าต่างแสดงโปรแกรมพิมพ์รายชื่อพนักงานที่มาทำงานสาย

จากรูปที่ 4.24 แสดงการทำงานในส่วนของการสั่งพิมพ์รายชื่อพนักงานที่มาทำงานสาย โดยจะทำการแยกตามวันที่และแผนก โดยสามารถเรียกดูข้อมูลได้โดยการเลือกปุ่ม "แสดงข้อมูล"

○ โปรแกรมพิมพ์รายชื่อพนักงานที่ขาดงาน

วันที่	เลขบัตรประชาชน	ชื่อพนักงาน	ชื่อ
2547-10-15	00000003	นาย	นายสมชาย ใจดี
2547-10-15	00000004	นาย	นายสมชาย ใจดี

รูปที่ 4.25 หน้าต่างแสดงโปรแกรมพิมพ์รายชื่อพนักงานที่ขาดงาน

จากรูปที่ 4.25 แสดงการทำงานในส่วนของการสั่งพิมพ์รายชื่อพนักงานที่ขาดงาน โดยจะทำการแยกตามวันที่และแผนก โดยสามารถเรียกดูข้อมูลได้โดยการเลือกปุ่ม "แสดงข้อมูล"

○ การพิมพ์รายชื่อของพนักงานแต่ละแผนก

รหัสพนักงาน	ชื่อ	รวมได้	โอที	รวมค่าชดเชย	เบ็ดเสร็จ	ปีงบประมาณ	วันที่	จำนวนวัน	รวม	ขาด
0000001	อ.วิเศษ อิ่มเกษม	3210	0	1615	0	240		0	0	30
0000002	วิเศษ อิ่มเกษม	2622	0	561	0	6		0	0	12

รูปที่ 4.26 หน้าต่างแสดง โปรแกรมพิมพ์รายชื่อของพนักงานแต่ละแผนก

จากรูปที่ 4.26 แสดงการทำงานในส่วนของการสั่งพิมพ์รายชื่อของพนักงานแต่ละแผนก โดยจะทำการแยกตามวันที่และแผนก โดยสามารถเรียกดูข้อมูลได้โดยการเลือกปุ่ม "แสดงข้อมูล"

○ การกำหนดอัตราเงินสำหรับหักค่ามาสาย

ประเภท	จำนวนเงิน(จำนวนหลัก)
ประเภท 1 ชั่วโมง	15
ประเภท 1 ชั่วโมงครึ่ง	20
ประเภท 2 ชั่วโมง	25
ประเภท 3 ชั่วโมง	30
ประเภท 4 ชั่วโมง	50

รูปที่ 4.27 หน้าต่างการกำหนดอัตราเงินสำหรับหักค่ามาสาย

จากรูปที่ 4.27 สำหรับส่วนนี้ มีไว้เพื่อให้ผู้ใช้สามารถกำหนดอัตราเงินที่ใช้เมื่อพนักงานมาทำงานสาย โดยเมื่อทำการกดปุ่ม "เพิ่มรายการ" ก็จะทำการเพิ่มเงื่อนไขที่ทำการกำหนดลงไป หลังจากนั้นก็สามารถทำการกดปุ่ม "บันทึกข้อมูล" เพื่อทำการบันทึกลงฐานข้อมูล แต่ในกรณีที่ต้องการยกเลิกการทำงานก็สามารถคลิกปุ่ม "Restore" ได้ แล้วข้อมูลเดิมก่อนหน้าก็จะปรากฏขึ้นมา

4.2.3ระบบสินค้าคงคลัง



รูปที่ 4.28 แสดงหน้าต่างโปรแกรมหลักของโปรแกรมในส่วนคลังสินค้า

จากรูปที่ 4.28 เป็นหน้าต่างหลักของโปรแกรมคลังสินค้า ประกอบไปด้วยโปรแกรมการเบิกสินค้า การส่งสินค้า การเพิ่มข้อมูล การแก้ไขข้อมูล การลบข้อมูล ออกจากโปรแกรม ซึ่งรายการต่าง ๆ เหล่านี้มีทั้งในแถบที่เป็นปุ่มรูปต่าง ๆ และแถบแสดงรายการที่เป็นตัวอักษร

ส่วนรูปต่อไปเป็นรูปที่ 4.29 เป็นการแสดงโปรแกรมการเบิกสินค้า ซึ่งข้อมูลที่จะต้องกรอกก็คือ รหัสพนักงานขาย รหัสพนักงานบัญชี 2 ท่าน ซึ่งจะต้องกรอกเป็นตัวเลขเท่านั้น อาจจะกรอกโดยใช้เครื่องมืออ่านรหัสแท่ง(barcode) หรือใช้คีย์บอร์ดในการกรอกรหัสก็ได้ เมื่อกรอกแล้วต้องกดปุ่มEnterด้วย ถ้าไม่กดก็จะมีข้อความขึ้นมาเตือน เมื่อกดปุ่มEnterในแต่ละช่องแล้วทางด้านขวามือของช่องนั้นจะมีชื่อของพนักงานขึ้นมาหากกรอกรหัสถูกต้อง หากรหัสไม่ถูกต้องจะมีข้อความขึ้นมาเตือนเช่นกัน ส่วนต่อมาเป็นส่วนที่เลือกคือ หมายเลขต.ร.ม. สินค้า และส่วนสุดท้ายคือจำนวนสินค้าซึ่งในช่องนี้จะรับข้อมูลเป็นตัวเลขเท่านั้น เมื่อกรอกครบแล้วกดปุ่ม "เพิ่ม" หากต้องการเบิกสินค้าเพิ่มสามารถเพิ่มอีกได้ แต่หากไม่ต้องการกดตกลงเครื่องก็จะพิมพ์ใบเบิกสินค้าออกมา

โปรแกรมบัญชี (ภาคเรียนที่ ๑)

โปรแกรมหลัก แก้ไขข้อมูล เพิ่มข้อมูล ลบข้อมูล รายงาน ข้อมูล

แก้ไขค่า ยกเลิกการแก้ไข เพิ่ม แก้ไขข้อมูล ลบข้อมูล แสดงรายการ

วันที่ 17/10/2547

พนักงานขาย: ชื่อ:

พนักงานขายที่ 1: ชื่อ:

พนักงานขายที่ 2: ชื่อ:

หมวดหมู่: ประเภท: ประเภท:

สินค้า: ราคา: จำนวน:

จำนวนสินค้า: 0 ชิ้น

สินค้า	ราคาต่อหน่วย	จำนวน	เป็นจำนวนเงิน

รวมราคากิ่งหนวด 0 บาท

ตกลง ยกเลิกการแก้ไขค่า

start New Connection - Project1 - Microsoft... Internet - Microsoft... โปรแกรมบัญชี - pit2 - Part 19:35

รูปที่ 4.29 แสดงส่วนของการเบกสินค้าไปขาย

โปรแกรมบัญชี (ภาคเรียนที่ ๑)

โปรแกรมหลัก แก้ไขข้อมูล เพิ่มข้อมูล ลบข้อมูล รายงาน ข้อมูล

แก้ไขค่า ยกเลิกการแก้ไข เพิ่ม แก้ไขข้อมูล ลบข้อมูล แสดงรายการ

วันที่ 17 ตุลาคม 2547

พนักงานขาย: ชื่อ:

พนักงานขายที่ 1: ชื่อ:

พนักงานขายที่ 2: ชื่อ:

หมวดหมู่: ประเภท: ประเภท:

สินค้า: ราคา: จำนวน:

จำนวนสินค้า: 0 ชิ้น

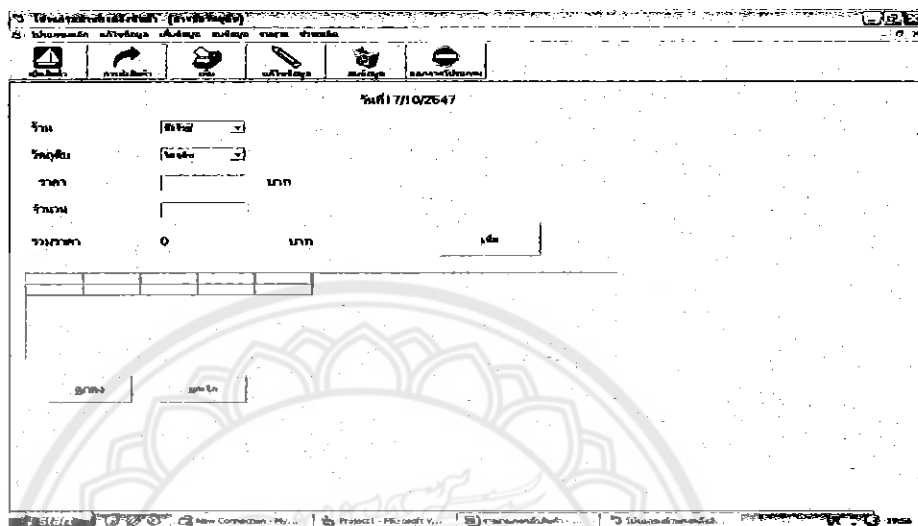
พนักงานขาย	ราคาต่อหน่วย	จำนวน(ชิ้น)

ตกลง ยกเลิก

start New Connection - My... Project1 - Microsoft Y... Internet - Microsoft... โปรแกรมบัญชี - pit2 - Part 19:35

รูปที่ 4.30 แสดงส่วนของการเบกวัตถุดิบ

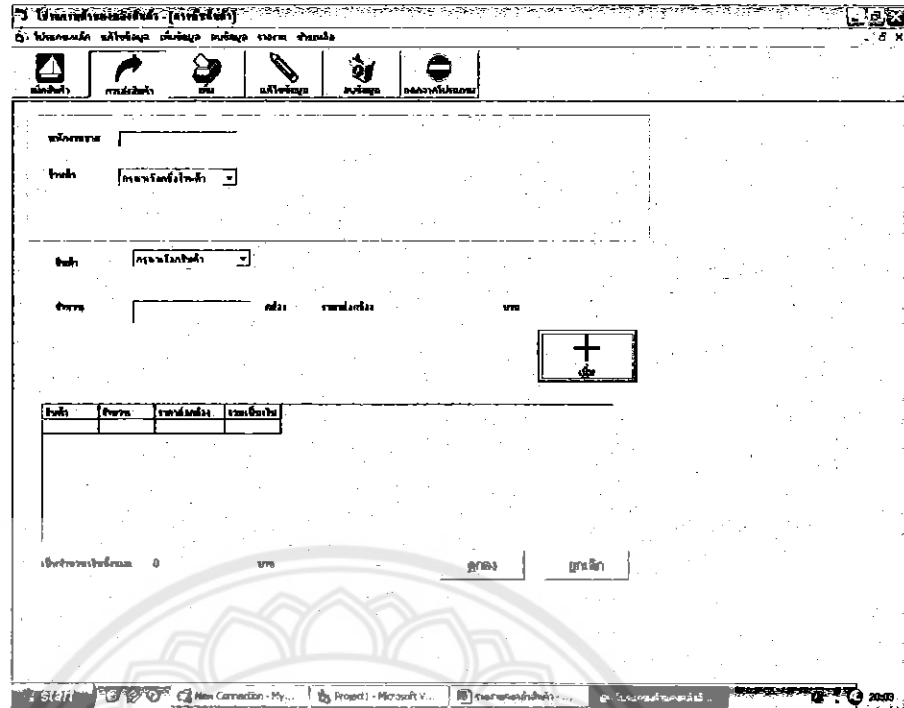
จากรูปที่ 4.30 แสดงโปรแกรมในส่วนของการเบิกวัตถุดิบไปใช้ โดยที่พนักงานต้องกรอกรหัสประจำตัวพนักงาน เลือกวัตถุดิบที่ต้องการจะเบิก จำนวนที่ต้องการเบิก แล้วคลิกเพิ่ม หากต้องการเบิกเพิ่มก็คลิกเลือกที่ชื่อวัตถุดิบใหม่ได้เลย แต่ไม่สามารถเบิกวัตถุดิบเดียวกันในวันเดียวกันและรหัสพนักงานเดียวกันได้



รูปที่ 4.31 แสดงโปรแกรมการซื้อวัตถุดิบ

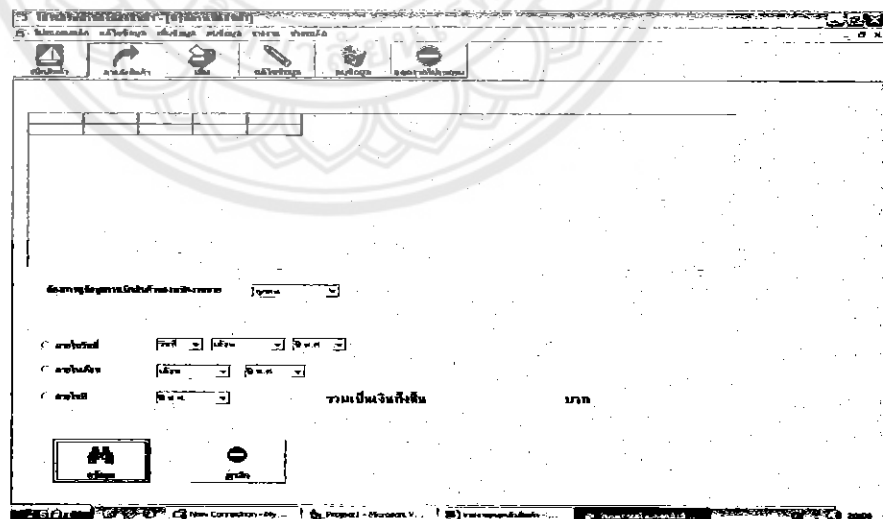
ในส่วนของการซื้อวัตถุดิบก็จะต้องเลือกในส่วน of ร้านค้า วัตถุดิบ เมื่อเลือกวัตถุดิบแล้วในส่วนที่อยู่ข้าง ๆ ช่องจำนวน จะกลายเป็นหน่วยของวัตถุดิบนั้น เช่น กระสอบ กิโลกรัม เป็นต้น ในส่วน of ราคาและจำนวน ต้องกรอกเป็นตัวเลขเท่านั้น เมื่อกดปุ่มเพิ่มก็จะเพิ่มข้อมูลลงในตารางและฐานข้อมูล และในส่วนรวมราคาจะเปลี่ยนไป เมื่อกดเพิ่มแล้ว หากกรอกข้อมูลครบและไม่ผิดพลาด ก็สามารถเพิ่มข้อมูลต่อไปได้เลย

ส่วนรูปที่ 4.32 เป็นรูปที่แสดงโปรแกรมการส่งสินค้า สิ่งที่ต้องกรอกคือ รหัสพนักงานขาย จำนวนที่ส่ง ส่วนที่ต้องเลือกคือ ร้านค้า และ สินค้า เมื่อเลือกและกรอกข้อมูลครบแล้ว กดเพิ่ม หากไม่มีอะไรผิดพลาดข้อมูลจะบันทึกลงในตารางและฐานข้อมูล แล้วจะเพิ่มสินค้าอีกก็ได้ เมื่อกดตกลง ก็จะพิมพ์ใบเสร็จสินค้าออกมา



รูปที่ 4.32 แสดง โปรแกรมการส่งสินค้า

ในรูป 4.33 เป็นโปรแกรมส่วนที่แสดงการสรุปผลการนำสินค้าไปขาย โดยดูจากพนักงานขายเป็นรายบุคคล หรือดูจากทั้งบริษัทก็ได้ แล้วเลือกช่วงเวลาที่ต้องการ คือ เป็นรายวัน รายเดือน รายปี แล้วเลือกปุ่มดูข้อมูล ก็จะมีข้อมูลปรากฏขึ้นมาที่ตารางด้านบน และมีปุ่มพิมพ์ขึ้นมาด้วย



รูปที่ 4.33 รายงานการเบิกสินค้าไปขาย

บทที่ 5

บทสรุป

5.1 บทสรุป

จากการจัดทำโปรแกรมเพื่อจัดการระบบฐานข้อมูลภายในบริษัทศิริวานิช(เอส แอนด์ คับเบิ้ลยู) จำกัด ได้ข้อสรุปจากโครงการซึ่งสามารถสรุปได้ดังนี้

บริษัทสามารถทำการจัดการระบบฐานข้อมูลด้านการบริหารฝ่ายบุคคล และการจัดการด้านสินค้าคงคลังได้อย่างมีประสิทธิภาพ โดยเป็นไปตามวัตถุประสงค์ของบริษัท และจากความสามารถของตัวโปรแกรมนั้นส่งผลให้ปัญหาที่บริษัทเคยประสบ เช่น การสูญหายของข้อมูล หรือการจัดเก็บข้อมูล นั้นลดลงเป็นอันมาก ซึ่งสร้างความพอใจต่อผู้บริหารบริษัทและพนักงานคอมพิวเตอร์เป็นอันมาก นอกจากนี้ปัญหาที่ลดลงแล้ว ผู้ใช้ยังได้รับความสะดวกสบายในการทำงาน ไม่ว่าจะเป็นในเรื่องของการบันทึก การแก้ไข และรวมไปถึงการพิมพ์สรุปผลต่างๆ และจากการดำเนินการในครั้งนี้นำให้คณะผู้จัดทำได้รับประสบการณ์หลายๆด้าน อาทิเช่น ประสบการณ์การเขียนและพัฒนาโปรแกรมระบบฐานข้อมูล ซึ่งทำให้คณะผู้จัดทำมีความเข้าใจในวิธีการเขียน โปรแกรมเพื่อติดฐานข้อมูลมากยิ่งขึ้น นอกเหนือจากการเรียนในห้องเรียน เพราะในการทำงานจริงนั้นได้ประสบกับปัญหาต่างๆ ที่เกิดขึ้น เช่น ปัญหาที่เกิดจากการเขียนภาษา SQL ซึ่งต้องมีความละเอียดรอบคอบในการเขียน เพราะถ้าผิดพลาดเพียงน้อยนิด เช่นการลืมใส่จุดภาคเพียงที่เดียว ก็สามารถส่งผลต่อการทำงานของโปรแกรมได้ เช่น โปรแกรมอาจจะไม่สามารถบันทึกข้อมูลลงฐานข้อมูลได้ นอกจากนี้ยังช่วยให้มีพัฒนาทักษะ และความคิด โดยช่วยให้สามารถคิดได้อย่างมีระบบ และท้ายที่สุดสิ่งที่ได้รับจากการทำโครงการในครั้งนีก็คือ การที่ช่วยให้รู้จักทำงานเป็นกลุ่ม ซึ่งรวมไปถึงการสานสัมพันธ์อันดีต่อบุคลากรในบริษัท อันส่งผลให้การจัดทำโครงการสามารถดำเนินการและผ่านไปได้ด้วยดี โดยประสบการณ์ที่ได้รับในครั้งนี ทางคณะผู้จัดทำสามารถนำไปประยุกต์ใช้ได้โดยตรงต่อการทำงานในชีวิตจริง

5.2 ข้อเสนอแนะ

ในข้อเสนอแนะนี้เป็นการกล่าวถึงสิ่งที่คณะผู้จัดทำมีความประสงค์ที่จะเสนอแนะต่อผู้ที่สนใจที่จะนำโปรแกรมการจัดการระบบฐานข้อมูลภายใน บริษัทศิริวานิช (เอส แอนด์ คับเบิ้ลยู) จำกัด ไปทำการพัฒนาต่อโดย ผู้ที่จะนำโปรแกรม ไปพัฒนา สามารถทำการพัฒนาเพิ่มเติมใน ด้านการพยากรณ์ (วางแผน) การผลิต ว่าบริษัทจะมีวิธีใดบ้างที่สามารถทำให้ผลกำไรเพิ่มขึ้น ในขณะที่เกิดการสูญเสียวัตถุดิบ (เช่นการสูญเสียวัตถุดิบโดยไม่ทราบสาเหตุขณะที่ทำการผลิต) ลดต่ำลง เป็นต้น

เอกสารอ้างอิง

- [1] กิตติ ภัคดีวัฒนะกุล และจำลอง ทรูอดสาหะ. กัมภีร์ระบบฐานข้อมูล. กรุงเทพฯ: ไทยเจริญการพิมพ์.(2544)
- [2] กิตติ ภัคดีวัฒนะกุล และจำลอง ทรูอดสาหะ. Visual Basic 6 ฉบับโปรแกรมเมอร์. กรุงเทพฯ:ไทยเจริญการพิมพ์.(2544)
- [3] สัจจะ จรัสรุ่งรวีร .คู่มือการเขียนโปรแกรมและใช้งาน Visual Basic 6.0 .กรุงเทพฯ: อินโฟเพรส.(2544)
- [4] อีชีวินสุขชัย สมพานิช. สร้างระบบฐานข้อมูลด้วย Visual Basic ฉบับโปรแกรมเมอร์ . กรุงเทพฯ: คำนสุทธการพิมพ์.(2545)



ประวัติผู้เขียนโครงการ



ชื่อ นาย กิตติศักดิ์ บำรุงเยี่ยมปัญญา
 ภูมิลำเนา 20/17-19 ถ.สิงห์วัฒน์ ต.ในเมือง จ.พิษณุโลก 65000
 ประวัติการศึกษา

- จบระดับมัธยมศึกษา โรงเรียนเฉลิมขวัญสตรี
- ปัจจุบันกำลังศึกษาอยู่ในระดับปริญญาตรีชั้นปีที่ 4 สาขาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

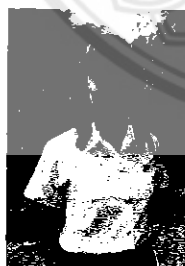
E-mail : kittipan_jo@hotmail.com



ชื่อ นางสาว อมรา สองสมุทร
 ภูมิลำเนา 130 หมู่ 3 ต.บางขุนไทร อ.บ้านแหลม จ.เพชรบุรี 76110
 ประวัติการศึกษา

- จบระดับมัธยมศึกษา โรงเรียนพรหมานุสรณ์จังหวัดเพชรบุรี
- ปัจจุบันกำลังศึกษาอยู่ในระดับปริญญาตรีชั้นปีที่ 4 สาขาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail : a_songsamut@yahoo.com



ชื่อ นางสาว โรจณี ขุมมงคล
 ภูมิลำเนา 6 หมู่ 3 ต.อุทัยเก่า อ.หนองฉาง จ.อุทัยธานี 61110
 ประวัติการศึกษา

- จบระดับมัธยมศึกษา โรงเรียนเตรียมอุดมศึกษา
- ปัจจุบันกำลังศึกษาอยู่ในระดับปริญญาตรีชั้นปีที่ 4 สาขาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail : rkhummongkol@hotmail.com