

ระบบการติดตามงาน
Request Tracking System



โดย
จิตรลดา วิจิตรโกสม
รหัส 44067482

อาจารย์ที่ปรึกษา
ผศ.ดร. จันทร์บุรณ์ สถิตวิวิวงค์

วัน เดือน ปี.....	0 6 2 1 5 6	2551
เลขทะเบียน.....		
เลขเรียกหนังสือ.....	วท. ๔๕๑๖	๒๕๕๖
"ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล."		

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
ภาคเรียนที่ 2 ปีการศึกษา 2546
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง



H002156

ชื่อหัวข้อ	ระบบการติดตามงาน
นักศึกษา	นางสาว จิตรลดา วิจิตรโกสุม
อาจารย์ที่ปรึกษา	ผศ.ดร. จันทร์นุรณธ์ สถิตวีริยวงศ์
ระดับการศึกษา	วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2546

บทคัดย่อ

การทำงานในลักษณะทีม เป็นเรื่องที่ผู้บริหารโครงการต้องมีการบริหารที่ดี เนื่องจากการทำงานย่อมมีการทำงานร่วมกัน ซึ่งถ้าบริหารเวลาและคนในทีมไม่ได้ย่อมส่งผลกระทบต่องานและชื่อเสียงของบริษัท ซึ่งสิ่งเหล่านี้เป็นเรื่องสำคัญในการบริหารงานที่ดีและทำให้งานประสบความสำเร็จตามเป้าหมายที่ได้วางไว้ อีกทั้งผู้บริหารที่ดีควรมีการติดตามว่างานที่ได้มอบหมายไป มีคนในทีมทำงานในส่วนใดบ้างและเป็นระยะเวลาเท่าใด เพื่อช่วยในการวัดประสิทธิภาพการทำงานและประเมินผลงานของคนในทีม เพื่อนำข้อมูลเหล่านี้มาปรับปรุงแก้ไขและพัฒนาการทำงานให้มีประสิทธิภาพยิ่งขึ้น

ในสัมมนานี้จะนำเสนอเครื่องมือในการติดตามงานอันประกอบไปด้วย การตรวจสอบงานที่ยังไม่มีผู้รับผิดชอบ และงานที่มีผู้รับผิดชอบแล้วมีความคืบหน้าอย่างไร และงานที่น่าจะมีปัญหาเสร็จไม่ทันกำหนดเวลา ซึ่งจะช่วยให้ผู้บริหารสามารถแก้ไขปัญหาได้ทันเวลา และควบคุมงานและการประมาณค่าใช้จ่ายได้ อีกทั้งการวัดและประเมินผลการทำงานของคนในทีมเพื่อนำไปพัฒนาการทำงานให้มีประสิทธิภาพต่อไป

Title	Request Tracking System
Student	Ms. Jitlada Vijitkosum
Advisor	Asst.Prof. Chanboon Sathitwiriyawong
Level of Study	Master of Science in Information Technology
Major	Information Science
Academic Year	2003

ABSTRACT

Teamwork is common and important in any working environment so that project manager should be proficient in management skill. Deficiency in time and team management would affect work and company reputation. Competent manager should also concern following up assigned work to track timeframe, assist in efficiency and effectiveness evaluation, and improve performance.

This seminar presents a tool helping in tracking work which is not set to be responsible for, progress of assigned work, and work which may not be finished in time. It aids manager in in-time troubleshooting, budget estimating and controlling, and also efficiency and effectiveness evaluation for better performance.

กิตติกรรมประกาศ

โครงการพัฒนาระบบเรื่องการติดตามงาน เพื่อเป็นเครื่องมือในการควบคุมดูแลงาน และ ช่วยวัดประสิทธิภาพของคนในทีม จะไม่สามารถสำเร็จลงได้หากขาดความร่วมมือจากทุกฝ่าย

ขอขอบพระคุณ ดร.จันทร์บุรณัฐ สถิตวิริยวงศ์ อาจารย์ที่ปรึกษาโครงการนี้ที่ได้กรุณาให้ คำปรึกษา แนะนำแนวทางการพัฒนาโครงการเป็นอย่างดียิ่งจน โครงการนี้สำเร็จลุล่วงไปด้วยดี

ขอขอบพระคุณคณาจารย์ทุกท่านที่ได้ให้วิชาการความรู้ให้สามารถนำมาประยุกต์ใช้ใน โครงการและการทำงาน

ขอขอบพระคุณเจ้าหน้าที่ทุกท่านในคณะเทคโนโลยีสารสนเทศที่ช่วยอำนวยความสะดวก ในการติดต่อกับทางคณะ

ขอบคุณกำลังใจจากเพื่อน ๆ ที่ช่วยให้การทำงานโครงการนี้สำเร็จลุล่วงลงได้

ขอบคุณคุณวรภณ เหมินทคุณ ที่คอยให้ความช่วยเหลือและคำปรึกษาในเรื่องเทคนิคการ เขียน โปรแกรม

จิตรลดา วิจิตร โกสุม

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	H
กิตติกรรมประกาศ.....	III
สารบัญ	IV
สารบัญตาราง.....	VI
สารบัญภาพ.....	VII
บทที่	
1. บทนำ.....	1
1.1 ปัญหาที่พบในปัจจุบัน.....	1
1.2 วัตถุประสงค์ของการพัฒนาระบบ	2
1.3 ขอบเขตของการพัฒนาระบบ.....	2
1.4 ผลที่คาดว่าจะได้รับ.....	2
1.5 ขั้นตอนการศึกษาระบบงาน.....	3
1.6 แนวทางการพัฒนาระบบ	3
1.7 Gantt Chart.....	5
2. ทฤษฎีที่เกี่ยวข้อง.....	6
2.1 ทฤษฎีที่ใช้ในการวิเคราะห์ระบบ.....	6
2.2 ฐานข้อมูล.....	15
2.3 ระบบการจัดการฐานข้อมูล.....	15
2.4 เทคโนโลยีฐานข้อมูลแบบ Client/Server.....	16
2.5 มาตรฐานของระบบ Client/Server ทางด้าน SQL.....	16
3. ขอบเขตการพัฒนาระบบ.....	18
4. การวิเคราะห์ระบบ.....	21
5. การพัฒนาระบบ.....	39

สารบัญ (ต่อ)

หน้า

6. บทสรุปและข้อเสนอแนะ.....	54
บรรณานุกรม.....	55
ภาคผนวก.....	56

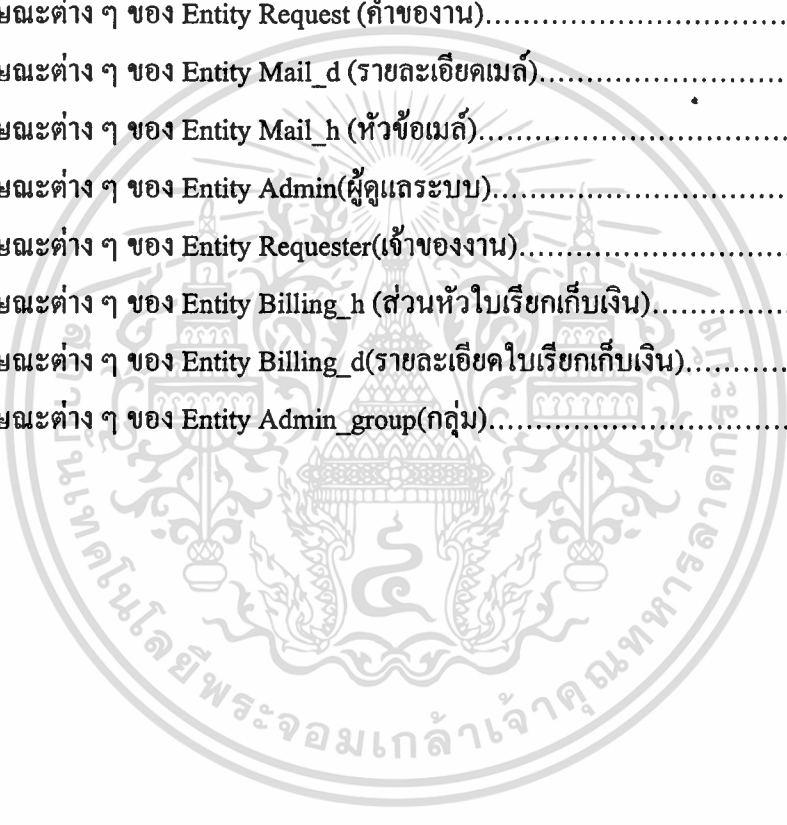


สารบัญตาราง

หน้า

ตารางที่

4.1 คุณลักษณะต่าง ๆ ของ Entity Request_working_detail (รายละเอียดการทำงาน).....	31
4.2 คุณลักษณะต่าง ๆ ของ Entity Request (คำขอรงาน).....	31
4.3 คุณลักษณะต่าง ๆ ของ Entity Mail_d (รายละเอียดเมลล์).....	32
4.4 คุณลักษณะต่าง ๆ ของ Entity Mail_h (หัวข้อเมลล์).....	32
4.5 คุณลักษณะต่าง ๆ ของ Entity Admin(ผู้ดูแลระบบ).....	33
4.6 คุณลักษณะต่าง ๆ ของ Entity Requester(เจ้าของงาน).....	33
4.7 คุณลักษณะต่าง ๆ ของ Entity Billing_h (ส่วนหัวใบเรียกเก็บเงิน).....	34
4.8 คุณลักษณะต่าง ๆ ของ Entity Billing_d(รายละเอียดใบเรียกเก็บเงิน).....	35
4.9 คุณลักษณะต่าง ๆ ของ Entity Admin_group(กลุ่ม).....	35



สารบัญภาพ

หน้า

รูปที่

1.1	แผนภาพแสดงระยะเวลาการดำเนินงาน.....	5
4.1	Context Diagram.....	22
4.2	Dataflow Diagram Level 1.....	23
4.3	Dataflow Diagram Level 2 ของ Process 1 (รับ Request).....	24
4.4	Dataflow Diagram Level 2 ของ Process 2 (ลงเวลาการทำงาน).....	25
4.5	Dataflow Diagram Level 2 ของ Process 3 (ออกบิลเพื่อเก็บเงิน).....	26
4.6	Dataflow Diagram Level 2 ของ Process 4 (ออกรายงาน).....	27
4.7	แผนภาพแสดงความสัมพันธ์ของข้อมูล.....	29
4.8	แผนภาพแสดงประเภทและความสัมพันธ์ของข้อมูล.....	30
5.1	แผนภาพแสดงหน้าจอการ Login.....	40
5.2	แผนภาพแสดงหน้าจอการรับ Request.....	41
5.3	แผนภาพแสดงหน้าจองานทั้งหมด.....	42
5.4	แผนภาพแสดงหน้าจอรายชื่องานที่ตนรับผิดชอบ.....	43
5.5	แผนภาพแสดงหน้าจอรายละเอียดงานที่ตนรับผิดชอบ.....	44
5.6	แผนภาพแสดงหน้าจอการกำหนดงานให้รับผิดชอบ.....	45
5.7	แผนภาพแสดงหน้าจอแสดงรายชื่อ Admin.....	46
5.8	แผนภาพแสดงหน้าจอเพิ่ม Admin.....	47
5.9	แผนภาพแสดงหน้าจอแสดงรายชื่อ Company.....	48
5.10	แผนภาพแสดงหน้าจอเพิ่ม Company.....	49
5.11	แผนภาพแสดงหน้าจอแสดงรายชื่อ Group.....	50

สารบัญภาพ (ต่อ)

หน้า

รูปที่

5.12	แผนภาพแสดงหน้าจอเพิ่ม Group.....	51
5.13	แผนภาพแสดงหน้าจอการ Login.....	52
5.14	แผนภาพแสดงหน้าจอสำหรับลงเวลาการทำงาน.....	53



บทที่ 1

บทนำ

1. ความเป็นมาของปัญหา

การทำงานเป็นสิ่งที่เราทุกคนต้องพบเจอทุกวัน ไม่ว่าจะเป็นการทำงานที่ได้รับมอบหมาย คนเดียวหรือทำงานในลักษณะทีม ซึ่งการควบคุมดูแลงาน ต่าง ๆ ให้สำเร็จลุล่วงเป็นเรื่องที่หัวหน้าโครงการต้องดูแลและควบคุมในทุก ๆ ด้านเป็นอย่างดี งานจึงจะสำเร็จลุล่วงทันเวลาและตรงตามเป้าหมายที่วางไว้ ซึ่งสิ่งที่หัวหน้าโครงการต้องดูแล เช่น เงินทุน ระยะเวลา คน เทคโนโลยี ลูกค้า และบริษัทที่มีส่วนร่วมในการทำงาน เป็นต้น ซึ่งถ้าหัวหน้าโครงการไม่ได้ติดตามหรือดูแล โอกาสที่โครงการนั้นจะไม่ประสบความสำเร็จ ก็เป็นไปได้สูง ซึ่งผลที่ตามมานอกเหนือจากในเรื่องของค่าใช้จ่าย ๆ เรื่องของการเสียเวลา และกำลังคนแล้ว ยังเป็นการเสียชื่อเสียงในการทำงานอีกด้วย รวมทั้งถ้าเป็นงานที่ติดต่อกับลูกค้าภายนอก ก็จะทำให้บริษัทเสียชื่อเสียงไปด้วย

ในสัมมนานี้จะนำเสนอเครื่องมือที่จะช่วยให้หัวหน้าโครงการหรือผู้บริหารสามารถทราบภาพรวมของโครงการทั้งหมดที่ได้รับพิจารณาว่ามีโครงการอะไรบ้างและใครเป็นผู้ดูแล และแต่ละโครงการมีความคืบหน้าอย่างไร ซึ่งช่วยให้โครงการ สามารถสำเร็จลุล่วงได้เป็นอย่างดี อีกทั้งผู้บริหารสามารถนำข้อมูลเหล่านี้ไปวางแผนในการทำงานในระยะยาวได้ต่อไป

1.1 ปัญหาที่พบในปัจจุบัน

- ❖ มีงานบางงานทำ ไม่ทันกำหนดเวลาเนื่องจากไม่ได้มีการเช็คและติดตามงาน
- ❖ ไม่สามารถควบคุมค่าใช้จ่ายได้ เนื่องจากงานไม่ทันตามกำหนดเวลาและต้องมีการเพิ่มคนและทำงานนอกเวลา
- ❖ ไม่สามารถพิจารณาหรือทราบว่าจะงานที่ทำไปใครมีส่วนทำงานใน project นี้บ้างและทำในส่วนตัว
- ❖ การประเมินราคางานต่าง ๆ ทำได้ค่อนข้างช้า และไม่มีหลักฐานการทำงานต่าง ๆ ที่มีหลักเกณฑ์
- ❖ ไม่มีรายละเอียดของงานและผู้ที่ต้องติดต่อดูแลเป็นใคร และถ้าพนักงานคนที่ทราบไม่อยู่หรือลาออกไปทำให้การทำงานเป็นไปไม่ได้ค่อนข้างช้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 วัตถุประสงค์ของการพัฒนาระบบ

- ❖ เพื่อช่วยให้สามารถติดตามงานทั้งหมดได้ และไม่มีงานในส่วนใดที่ไม่ได้ทำ
- ❖ เพื่อช่วยให้ผู้บริหารทราบถึงการทำงานของพนักงานแต่ละคนว่าทำงานมีประสิทธิภาพแค่ไหน
- ❖ เพื่อช่วยในการประเมินราคาของแต่ละโครงการให้เหมาะสมโดยอ้างอิงจากข้อมูลจริง และทำให้การพิจารณาในครั้ง ต่อ ๆ ไปทำได้อย่างรวดเร็ว และถูกต้อง
- ❖ เพื่อช่วยให้หัวหน้าโครงการสามารถควบคุมค่าใช้จ่ายที่จะใช้ในโครงการให้อยู่ในงบประมาณที่ได้รับจัดสรร
- ❖ เพื่อช่วยให้การทำงานเป็นทีมง่ายขึ้น และมีประสิทธิภาพมากขึ้น
- ❖ เพื่อช่วยให้ผู้บริหารสามารถวางแผน โครงการและการทำงานต่าง ๆ ได้ดียิ่งขึ้น เนื่องจากมองเห็นภาพรวมของแต่ละโครงการ

1.3 ขอบเขตของการพัฒนาระบบ

- ❖ ระบบสามารถบันทึกงานใหม่รวมถึงแก้ไขรายละเอียดของงานเดิมได้
- ❖ สามารถทราบสถานะของงานแต่ละงานได้ว่ามีความคืบหน้าไปอย่างไร
- ❖ ระบบสามารถบันทึกเวลาเริ่มต้นและสิ้นสุดการทำงานของพนักงานแต่ละคนได้
- ❖ ทำการคำนวณและประเมินราคาโครงการจาก ระยะเวลาทำงานของพนักงานแต่ละคน
- ❖ ทำการสรุปเพื่อให้เห็นภาพรวมของ โครงการ โดยสามารถดูได้ว่าโครงการนี้มีพนักงานทำกี่คน ใช้เวลาเท่าไร และพนักงานแต่ละคนทำงานในส่วนใดบ้าง

1.4 ผลที่คาดว่าจะได้รับ

- ❖ ช่วยให้หัวหน้างานสามารถติดตามงานได้
- ❖ ช่วยให้สามารถประเมินราคา หรือระยะเวลาในการทำงานได้อย่างถูกต้องและแม่นยำ
- ❖ ช่วยในการประเมินประสิทธิภาพในการทำงานของลูกน้องว่าใน 1 วันได้ทำงานอะไรไปบ้าง
- ❖ ช่วยให้คนในทีมงานที่ทำงานร่วมกันสามารถ ทราบรายละเอียดว่าใครได้ทำอะไรในส่วนใดไปแล้วบ้างเพื่อให้สามารถทำงานร่วมกันได้อย่างมีประสิทธิภาพ
- ❖ ช่วยให้ผู้บริหารเห็นภาพรวมของโครงการทั้งหมดว่ามีความคืบหน้าไปอย่างไร และถ้าโครงการใดมีแนวโน้มที่จะไม่สำเร็จทันตามกำหนดเวลา จะได้มีการวางแผนเพื่อแก้ปัญหาได้ทันเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ❖ ช่วยให้ผู้บริหารหรือหัวหน้าโครงการได้นำข้อมูลเหล่านี้ไปวางแผนในการทำงานและทิศทางต่าง ได้ต่อไป

1.5 ขั้นตอนการศึกษาระบบงาน

- ❖ กำหนดขอบเขตงานที่ต้องการศึกษา โดยพิจารณาจากวัตถุประสงค์ของโครงการประกอบกับแนวคิดเบื้องต้นที่จะจัดทำระบบ
- ❖ ศึกษาถึงลักษณะของงาน เพื่อที่จะสามารถ จำแนกกลุ่มหรือประเภทของงานได้อย่างมีประสิทธิภาพ
- ❖ ศึกษาถึงเทคโนโลยีในปัจจุบันที่มีการทำงานในลักษณะนี้ว่ามีข้อดี และข้อเสียที่จุดใด เพื่อได้นำมาปรับปรุงเพิ่มเติมให้ได้โปรแกรมที่ตรงกับความต้องการ และดีที่สุดในที่สุด
- ❖ ศึกษาถึงวิธีการในการติดตามงาน รวมถึงการออกแบบรายงานที่ดีเพื่อให้ผู้บริหารสามารถได้รับประโยชน์สูงสุด
- ❖ วิเคราะห์ปัจจัยความเสี่ยงที่อาจจะเกิดขึ้นในการพัฒนาระบบงาน
- ❖ วิเคราะห์และออกแบบระบบงานตามที่ได้ศึกษาไว้
- ❖ ศึกษาถึงเทคโนโลยีในปัจจุบัน และเครื่องมือที่จะนำมาช่วยในการพัฒนาระบบงานเพื่อให้บรรลุตามวัตถุประสงค์ที่กำหนดไว้

1.6 แนวทางการพัฒนาระบบ

1.6.1 ทฤษฎีที่ใช้ในการพัฒนาระบบงาน

- ❖ ส่วนของการวิเคราะห์ระบบใช้หลักการ SDLC (System Development Life Cycle)
- ❖ ส่วนในการออกแบบใช้เทคนิค Dataflow Diagram ในการอธิบายการไหลของข้อมูลและการทำงานเพื่อนำไปใช้ในการเขียน โปรแกรม
- ❖ ส่วนของระบบฐานข้อมูล (Database) ใช้โครงสร้างของฐานข้อมูลแบบ Relational และใช้เทคนิคนอร์มัลไลเซชัน (Normalization) และใช้ ER-Diagram ในการอธิบายความสัมพันธ์ของฐานข้อมูล และอธิบาย Field ต่าง ๆ ที่ใช้ด้วย Datadictionary
- ❖ สถาปัตยกรรมที่ใช้ ทำงานในลักษณะไคลเอ็นท์/เซิร์ฟเวอร์ (Client/Server) โดยฐานข้อมูลและไฟล์ที่ใช้ในการทำงานจะอยู่ที่เครื่อง เซิร์ฟเวอร์ (server)

1.6.2 เครื่องมือที่ใช้ในการพัฒนาระบบงานประกอบด้วย

Hardware

- ❖ เครื่อง server รุ่น Pentium III 800 MHz, Harddisk 40 GB, RAM 256 MB
- ❖ เครื่อง client รุ่น : Pentium II 400 MHz, Harddisk 10 GB, RAM 64 MB

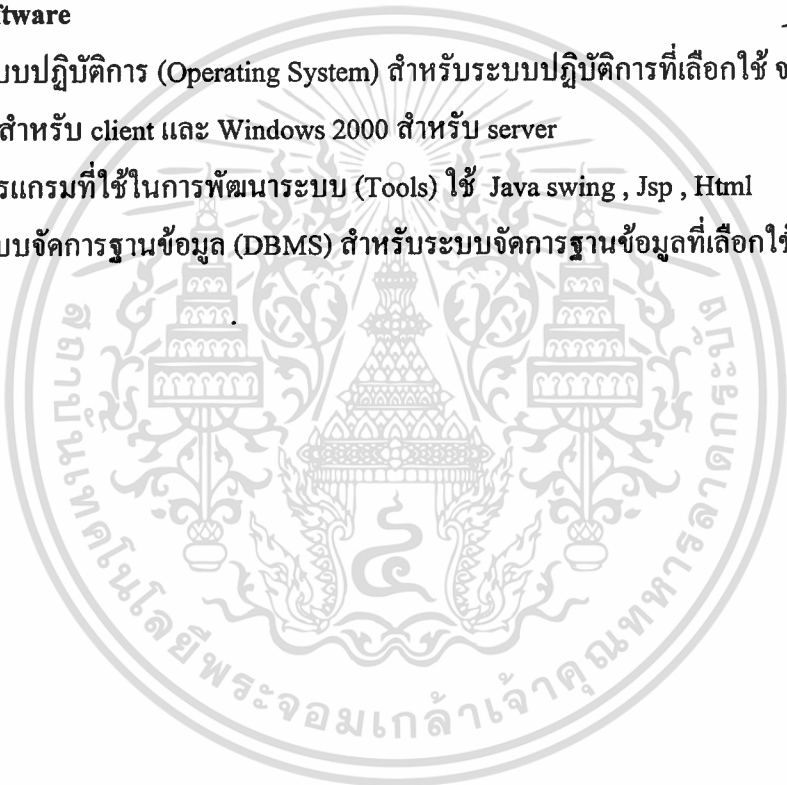
เครื่องพิมพ์

- ❖ สำหรับ client : Printer Dot Matrix Epson รุ่น LQ1170i
- ❖ สำหรับ server : Printer HP LaserJet 1100

Software

- ❖ ระบบปฏิบัติการ (Operating System) สำหรับระบบปฏิบัติการที่เลือกใช้ จะเป็น Windows 9x สำหรับ client และ Windows 2000 สำหรับ server
- ❖ โปรแกรมที่ใช้ในการพัฒนาระบบ (Tools) ใช้ Java swing , Jsp , Html
- ❖ ระบบจัดการฐานข้อมูล (DBMS) สำหรับระบบจัดการฐานข้อมูลที่เลือกใช้ Oracle version

8.0



บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

2.1 ทฤษฎีที่ใช้ในการวิเคราะห์ระบบ

2.1.1 การเข้าใจปัญหา (Problem Recognition)

การทำความเข้าใจกับปัญหาที่เกิดขึ้นในระบบ ว่าเกิดจากสาเหตุใด และส่งผลกระทบต่อสิ่งใดบ้าง โดยการที่จะเข้าใจปัญหาได้ต้องลงไปศึกษาและคลุกคลีกับปัญหานั้น ประกอบกับการหาความรู้เพิ่มเติมในสิ่งที่เราไม่ทราบ เพื่อให้สามารถวิเคราะห์ปัญหาได้ถูกต้องและมีประสิทธิภาพซึ่งขั้นตอนนี้เป็นสิ่งจุดเริ่มต้นสำหรับขั้นตอนต่างๆ ต่อไป

2.1.2 ศึกษาความเป็นไปได้ (Feasibility Study)

เป็นขั้นตอนที่เกี่ยวข้องกับการประเมินต้นทุนของทางเลือกต่าง ๆ ของการพัฒนา เพื่อใช้พิจารณาเลือกทางเลือกที่ดีที่สุด มีความเป็นไปได้และคุ้มค่าที่สุด การศึกษาความเป็นไปได้ของระบบ ถูกนำมาพิจารณาร่วม ก่อนการตัดสินใจสร้างระบบ เนื่องจากเหตุผลต่างๆ ดังนี้คือ

- ❖ เพื่อลดความเสี่ยง ต่อการสูญเสียทั้งเวลา ทรัพยากรของระบบในการพัฒนาระบบใหม่ และค่าใช้จ่ายที่เกิดขึ้นจากการสร้างระบบใหม่ ในกรณีที่ไม่สามารถนำระบบใหม่มาใช้ได้จริง
- ❖ เป็นการศึกษาผลกระทบ ที่เกิดขึ้น จากการสร้างระบบใหม่ขึ้นในด้านต่างๆ คือ
 - ด้านงบประมาณ
 - ด้านการยอมรับจากผู้ใช้งาน
 - ด้านความเป็นไปได้ทางเทคนิค
 - ด้านความเป็นไปได้ ในการนำไปใช้งานจริง
- ❖ ผลของการศึกษาความเป็นไปได้ จะทำให้สามารถนำมาช่วยประกอบการตัดสินใจว่า ระบบใหม่ที่จะถูกพัฒนาขึ้น เป็นระบบที่มีความสามารถตรงกับความต้องการของผู้ใช้งานจริงหรือไม่

2.1.3 วิธีการที่ใช้ในการวิเคราะห์และออกแบบระบบ

รวบรวมข้อมูล จากแหล่งข้อมูลต่างๆ คือ

- เอกสาร (.Document) ของระบบเดิม ที่ใช้งานอยู่ เพื่อให้ทราบ และ เข้าใจหลักการการทำงานของระบบเดิม
- ศึกษาระบบ โดยผู้ออกแบบและพัฒนาระบบ มาสังเกตการใช้งานจริง ของระบบปัจจุบันที่ใช้อยู่
- สอบถามผู้ใช้งานเพิ่มเติม ถึงความต้องการที่จะให้ระบบมีให้ โดย ผู้ใช้งานในที่นี้ คือผู้บริหารระดับสูง

❖ นำข้อมูลที่ได้จากการรวบรวม มาทำการวิเคราะห์ความเป็นไปได้ของระบบ ในด้านต่างๆ ดังนี้

- Operational Feasibility

ในการวิเคราะห์ ความเป็นไปได้ ในการปฏิบัติการจริง จะคำนึงถึง ปัจจัยที่มีผู้ใช้งาน ในระดับต่างๆ คือ

ระดับผู้ปฏิบัติงานหน้างาน

ผู้ใช้งานระบบในที่นี้ คือพนักงานที่ทำงานภายในทีม ผู้ใช้งาน ในระดับผู้บริหารระดับสูง

ที่เรียกดูรายงานต่างๆ ผู้ใช้งานระดับนี้ เป็นผู้ริเริ่มโครงการนี้ จึง ไม่น่าจะมีผลกระทบในทางลบกับระบบนี้ และระบบที่จะถูก พัฒนาขึ้นใหม่ น่าจะได้รับการสนับสนุน จากผู้ใช้งานในระดับนี้ ด้วย

ผู้ใช้งาน

ที่มีหน้าที่สนับสนุนระบบ เช่น ผู้ที่มีหน้าที่คือข้อมูลเพิ่มเติม ซึ่ง เป็นข้อมูลที่ไม่ได้รับจากการทำงานจริง , ผู้ดูแลระบบ ซึ่งมีหน้าที่ ดูแลระบบการติดตามงาน ให้มีสภาพที่พร้อมสำหรับการใช้งาน

- Technical Feasibility

การพิจารณาถึงความเป็นไปได้ในเชิงเทคนิค ซึ่งสามารถพิจารณาจาก ส่วนประกอบต่างๆของระบบใหม่ดังนี้คือ

ทางด้าน *Hardware*

พิจารณาจาก Hardware ที่ใช้อยู่ในปัจจุบันว่าสามารถรองรับกับระบบใหม่ที่กำลังจะพัฒนาหรือไม่ พิจารณาว่าต้องมีการเพิ่มเติมหรือเปลี่ยนแปลงหรือไม่

ทางด้าน Software

พิจารณาถึง Software ที่มีอยู่ในขณะนี้ว่าตรงตามความต้องการกับระบบหรือไม่

- Economic Feasibility

ความเป็นไปได้ทางการเงิน พิจารณาถึงงบประมาณในการเริ่มโครงการเปรียบเทียบกับระยะเวลาและจำนวนคนที่ทำระบบ เพื่อเป็นประโยชน์และข้อมูลในการวางแผนโครงการต่อไป

❖ ตัดสินใจถึงความเป็นไปได้ของระบบ
เมื่อได้ศึกษาถึงความเป็นไปได้ของระบบแล้วนำข้อมูลต่าง ๆ ที่ได้ศึกษามาพิจารณาหรือประชุมเพื่อตัดสินใจถึงความเป็นไปได้ ว่าระบบ สามารถใช้งานจริงได้หรือไม่ มีความเป็นไปได้มากน้อยแค่ไหน และสรุปผลการตัดสินใจในการพัฒนาระบบ

❖ การวิเคราะห์ (Analysis)
เริ่มต้นทำการศึกษาดังการทำงานของระบบเก่าว่ามีการทำงานอย่างไร เก็บรวบรวมสิ่งที่เป็นปัญหาในระบบเก่าว่าเกิดจากสาเหตุใด เพราะอะไร และนำมาวิเคราะห์ถึงการพัฒนาระบบใหม่ว่าจะทำการแก้ปัญหาที่เกิดขึ้นได้อย่างไร กำหนดขอบเขตของงาน โดยเมื่อได้ผลจากการวิเคราะห์ปัญหาและความต้องการของระบบใหม่เรียบร้อยแล้วก็นำเสนอให้ผู้บริหาร หรือผู้ใช้งานพิจารณาเพื่อดูแนวทางในการพัฒนาต่อไป

❖ การออกแบบ (Design)
หลังจากได้มีการกำหนดขอบเขตของงานในระบบใหม่ วิเคราะห์การทำงานในส่วนต่าง ๆ แล้ว ก็จะเริ่มขั้นตอนการออกแบบระบบ โดยทำการออกแบบให้สอดคล้องและตรงตามความต้องการของผู้ใช้งาน

❖ การพัฒนาระบบงาน (Construction)

เป็นขั้นตอนในการสร้างหรือพัฒนาระบบให้เกิดขึ้นจริงตามที่ได้วิเคราะห์และออกแบบไว้ ซึ่งอาจจะมีการทำงานเป็นทีมสำหรับระบบที่ใหญ่ก็อาจจะต้องมีการแบ่งงานให้ในทีมที่รับผิดชอบและหลังจากนั้นก็ทำการทดสอบระบบว่าสามารถทำงานได้ตามความต้องการหรือไม่และผลลัพธ์ที่ได้ถูกต้องหรือเปล่า รวมทั้งประสิทธิภาพการทำงานว่าเป็นอย่างไร

❖ การทดสอบ (Testing)

โดยแบ่งการทดสอบระบบได้ดังนี้

- ทดสอบระบบย่อย (Unit Testing) เป็นการทดสอบระบบของแต่ละคนที่รับผิดชอบ ทดสอบฟังก์ชันที่เขียนขึ้นมา
- ทดสอบระบบรวม (Integration Testing) นำเอาแต่ละส่วนมารวมกันก่อนแล้วจึงทดสอบ สร้างเหตุการณ์จำลอง รวมถึงข้อมูลต่าง ๆ หลังจากนั้นก็นำมาทดสอบทั้งหมดว่าสามารถทำงานได้ดีและเข้ากันได้หรือไม่ รวมทั้งความถูกต้องของผลลัพธ์ที่ได้
- ทดสอบประสิทธิภาพการทำงานของระบบ (System Testing) ทดสอบการทำงานของระบบว่ามีประสิทธิภาพแค่ไหน เช่นทำงานช้าหรือเร็วอย่างไรซึ่งเป็นการทดสอบขั้นสุดท้ายถึงการทำงานทั้งหมดของโปรแกรมที่สร้างขึ้น
- รวมถึงจัดทำเอกสารหรือคู่มือต่าง ๆ (Documentation) และการสอนการใช้งาน (Tranning)

❖ การปรับเปลี่ยน (Conversion)

เป็นการนำเอาระบบที่ได้พัฒนาขึ้นมาใช้ในการทำงาน หรือเป็นการเปลี่ยนจากระบบเก่ามาเป็นระบบใหม่ที่ได้พัฒนาขึ้น ซึ่งการปรับเปลี่ยนมีวิธีการและเทคนิคดังนี้

- เลิกใช้ระบบเก่าทันที (Direct Cutover)

เป็นการยกเลิกระบบเก่าที่ใช้งานทันทีหลังจากเริ่มใช้งานระบบใหม่ วิธีมีความเสี่ยงมากกว่าวิธีอื่น ๆ เนื่องจากถ้าระบบใหม่ไม่สามารถใช้งานได้ ผู้ใช้จะไม่สามารถทำงานได้เลย และส่งผลกระทบต่ออย่างมาก ต่องานที่ต้องทำอยู่ทุกวัน แต่วิธีนี้จะประหยัดค่าใช้จ่ายมาก วิธีนี้จึงเหมาะกับระบบที่มีความพร้อมสูง และระบบที่ได้ทดสอบมาเป็น

อย่างดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **ทำงานหรือควบคู่กัน (Parallel Operation)**
เป็นการใช้ระบบเก่าทำงานไปควบคู่กับระบบใหม่สักระยะหนึ่งเมื่อเห็นว่าระบบใหม่หนึ่งแล้วก็ยกเลิกระบบเก่าทันที วิธีนี้มีความเสี่ยงน้อยที่สุด แต่มีค่าใช้จ่ายมากที่สุด
 - **ใช้งานในสาขาใดสาขาหนึ่งก่อน (Pilot Operation)**
เป็นการนำเอาระบบใหม่ทั้งหมดที่ได้พัฒนาไปใช้ที่สาขาใดสาขาหนึ่งก่อนว่าคิดปัญหาอะไรหรือไม่ ซึ่งถ้าระบบใช้งานได้ดีก็นำไปใช้งานในที่อื่น ๆ ได้ เรียกสาขาที่ใช้งานก่อนว่า Pilot site วิธีนี้จะมีความเสี่ยงและค่าใช้จ่ายอยู่ในระดับปานกลางเมื่อเทียบกับเทคนิคอื่น ๆ ที่ได้กล่าวมา
 - **ใช้งานส่วนใดส่วนหนึ่งก่อน (Phased Operation)**
ในการทำงานของแต่ละระบบสามารถแบ่งการทำงานออกเป็นส่วน ๆ ได้ อาจจะแบ่งตามแผนก หรือการทำงาน ดังนั้นวิธีนี้จะเป็นการนำเอาระบบไปใช้ในส่วนหนึ่งก่อนถ้าสามารถใช้งานได้ดี ก็ใช้ในส่วนต่อ ๆ ไป จนครบทุกส่วน วิธีนี้มีความเสี่ยงและต้นทุนปานกลาง
- ❖ **การบำรุงรักษา (Maintenance)**
- การแก้ไขเปลี่ยนแปลงหรือเพิ่มเติม โปรแกรมหลังจากที่ได้มีการใช้งานไปแล้ว ระยะเวลาหนึ่ง โดยการดูแลรักษาระบบมี 4 ลักษณะดังนี้
- **แก้ไขข้อผิดพลาดที่ผิดไปจากความต้องการของผู้ใช้ (Collection Maintenance)** เช่น ใส่งูตรที่ใช้ในการคำนวณผิด การดูแลรักษาในลักษณะเช่นนี้ค่าใช้จ่ายจะลดไปเรื่อย ๆ เนื่องจากข้อผิดพลาดจะถูกแก้ไขไปตามลำดับ
 - **การปรับปรุงระบบตามสภาพแวดล้อมทางเทคนิคที่เปลี่ยนแปลงไป (Adaptive Maintenance)** เช่นมีการเปลี่ยนจากระบบปฏิบัติการ Dos เป็น Windows ซึ่งระบบต้องเปลี่ยนแปลงตาม การดูแลรักษาในลักษณะนี้ค่าใช้จ่ายมีแนวโน้มเพิ่มสูงขึ้นเรื่อย ๆ เมื่อเวลาผ่านไป เพราะเทคโนโลยีย่อมมีการเปลี่ยนแปลงอยู่เสมอ
 - **การปรับปรุงระบบตามความต้องการที่เกิดขึ้นใหม่ของผู้ใช้ (Perfective Maintenance)** เป็นความต้องการที่ผู้ใช้ไม่ได้กำหนดไว้ตั้งแต่แรก ซึ่งอาจเพราะคิดไม่ถึง หรือ เพราะเพิ่งเกิดความต้องการนี้

การดูแลรักษาในลักษณะนี้อาจทำได้ยาก ถ้าระบบไม่ได้มีการ
ออกแบบไว้ตั้งแต่แรก

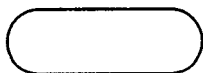
- การดูแลรักษาตามวงรอบ (Preventive Maintenance) การ ปรนนิบัติ
ดูแลรักษาตามวงรอบเพื่อยืดอายุการใช้งานระบบ และหลีกเลี่ยง
ปัญหาที่ก่อนที่จะปัญหาจะเกิดขึ้น

2.1.4 เครื่องมือที่ใช้ในการวิเคราะห์และออกแบบระบบ

เครื่องมือต่าง ๆ ที่ใช้ช่วยในการออกแบบ จะทำให้ผู้ที่เกี่ยวข้องในระบบนั้น ๆ มีความ
เข้าใจ ในระบบงานได้ง่ายขึ้น มองเห็นภาพที่ตรงกัน เนื่องจากเครื่องมือที่ช่วยในการ
ออกแบบระบบมักจะเป็นรูปภาพ ซึ่งเครื่องมือที่ใช้ในการออกแบบระบบมีมากมาย
ขึ้นอยู่กับจุดประสงค์ของการใช้งาน ทั้งนี้เราจำเป็นต้องทำการศึกษาการทำงานของ
เครื่องมือต่าง ๆ เพื่อให้สามารถนำเครื่องมือเหล่านั้นมาใช้งานได้ถูกขั้นตอน และเป็น
ประโยชน์ต่อผู้พัฒนาระบบ ต่อไป

Flowchart

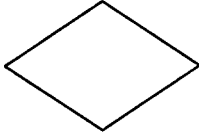
เป็นเครื่องมือที่ช่วยแสดงขั้นตอนและลำดับการทำงานต่าง ๆ ในโปรแกรมช่วยให้
โปรแกรมเมอร์และนักวิเคราะห์ระบบ รวมทั้งบุคคลอื่น เข้าใจระบบการทำงาน ซึ่งการ
เขียน Flowchart จะเขียนเรียงลำดับจากบนลงล่าง ซ้ายไปขวา ซึ่งจะใช้คำอธิบายสั้น ๆ
และมีการเรียงลำดับของงานแต่ละงานที่เกิดขึ้น สัญลักษณ์ที่ใช้ใน Flowchart มีดังนี้



ใช้แสดงจุดเริ่มต้นและจุดสิ้นสุดของ Flowchart



ใช้แสดงการทำงานต่าง ๆ



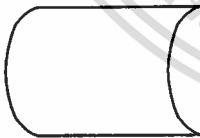
ใช้แสดงเงื่อนไขสำหรับตัดสินใจ



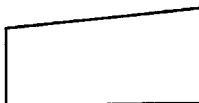
ใช้แสดงผลข้อมูลออกทางสื่อที่เป็นเอกสารหรือสิ่งพิมพ์
เช่น กระดาษ หรือ เครื่องพิมพ์



ใช้แสดงถึงหน่วยรับและแสดงผลข้อมูล



ใช้แสดงการเก็บข้อมูลต่างๆ ที่เป็นแบบมีการเข้าถึง
ข้อมูล โดยตรง



ใช้แสดงการนำข้อมูลเข้าผ่านทางตี๋บอร์ด



ใช้แสดงการไหลหรือเชื่อมขั้นตอนต่าง ๆ ในระบบงาน



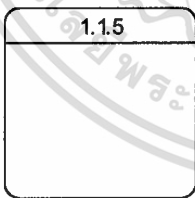
ใช้แสดงผลข้อมูลทางจอภาพ

Data flow diagram

เป็นเครื่องมือที่ช่วยแสดงการไหลของข้อมูล แสดงว่าระบบทำอะไร ซึ่งประกอบด้วยส่วนต่าง ๆ ดังนี้

- Process Symbol

ตัวประมวลผล ทำหน้าที่รับข้อมูลเข้า (Input) ประมวลผลเพื่อให้ได้ผลลัพธ์ (Output) ออกมาสัญลักษณ์ที่ใช้ได้แก่



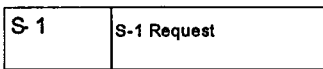
- Data flow Symbol

ตัวข้อมูลที่กำลังไหล เป็นตัวเชื่อมระหว่างต้นทางกับปลายทาง



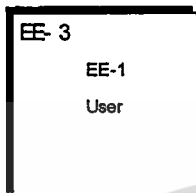
- Data store Symbol

เป็นตัวเก็บข้อมูลที่ระบบเป็นผู้ดูแล



- External entity Symbol

สิ่งที่อยู่ภายนอกระบบ บางครั้งเรียก terminator เป็นส่วนที่ระบบต้องติดต่อกับ



Context Diagram

เป็นชื่อเรียกของ level บนสุดในการเขียน diagram จะบอกถึงขอบเขตของระบบ ทำให้สามารถมองเห็นภาพรวมว่าระบบทำอะไร ติดต่อกับภายนอกอะไรบ้าง

Entity Relationship Diagram

Entity Relationship Diagram (E-R Diagram) หรือ แผนภาพความสัมพันธ์ของสิ่งต่างๆ ซึ่งคำว่าความสัมพันธ์ (Relationship) จะหมายถึงความสัมพันธ์ทางตรรกศาสตร์ของสิ่งต่างๆ (Entity) ภายในระบบนั้น ดังนั้นความสัมพันธ์ทางตรรกศาสตร์นั้นจะเป็นธรรมชาติของสิ่งต่างๆที่เกี่ยวข้องกันนั่นเอง ซึ่งเราสามารถสื่อออกมาในลักษณะของรูปภาพเพื่อใช้ในการนำเสนอซึ่งจะเรียกว่าแผนภาพความสัมพันธ์ของสิ่งต่างๆหรือเรียกย่อว่า แผนภาพ E-R ซึ่งในการพัฒนาโครงการครั้งนี้ได้ใช้ Embarcadeo ERStudio เป็นTool ในการออกแบบ

- Entity



- Flow



2.2 ฐานข้อมูล

ฐานข้อมูล (Database) คือที่อยู่ของข้อมูลที่มีความสัมพันธ์กัน หรืออาจจะเปรียบเทียบเป็นคลังข้อมูลก็ได้ ข้อมูลเหล่านี้จะถูกจัดเก็บร่วมกันอย่างมีระบบและมีรูปแบบ ทำให้ง่ายต่อการประมวลผล และการจัดการ โดยปกติการใช้งานจะต้องมีโปรแกรมเพื่อจัดการฐานข้อมูลที่มีอยู่ เรียกว่า DBMS (Database Management System) สำหรับฐานข้อมูลที่ได้รับคามนิยมมากที่สุดในปัจจุบันจะเป็นแบบ Relation Database ซึ่งจะจัดเก็บข้อมูลอยู่ในรูปของตาราง (Table) โดยที่ข้อมูลในแต่ละตารางมีความสัมพันธ์กัน ในการพัฒนาโครงการนี้ได้เลือกใช้ Oracle version 8.0 ในการสร้างระบบฐานข้อมูล

2.3 ระบบการจัดการฐานข้อมูล (Database Management System : DBMS)

เป็นซอฟต์แวร์ที่ถูกออกแบบเพื่อป้องกันและจัดการกับฐานข้อมูล วัตถุประสงค์ของ DBMS ก็เพื่อที่จะให้ความสะดวกและประสิทธิภาพของการเรียก และการเก็บข้อมูล DBMS จะมีหน้าที่ในการควบคุมดูแลการสร้างและเรียกใช้ฐานข้อมูล DBMS นี้จะเป็น Software ที่เปรียบเสมือนสื่อกลางระหว่างผู้ใช้และโปรแกรมต่างๆที่เกี่ยวข้องกับการใช้ฐานข้อมูล

ระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management System : RDBMS) โดยในปัจจุบันภาษาที่นิยมใช้ในการทำหน้าที่จัดการค้นหา เพิ่มเติม หรือแก้ไขเปลี่ยนแปลง คือ ภาษาเอสคิวแอล (Structure Query Language : SQL) ซึ่งเป็นภาษาที่สามารถจะเข้ามาจัดการกับข้อมูลได้ทั้ง 3 ลักษณะคือ

1. ภาษาที่เกี่ยวข้องกับการจัดการโครงสร้างข้อมูล (Data Definition Language : DDL) เป็นการจัดการที่มีไว้สำหรับผู้ดูแลระบบ (Database Administrator) หรือผู้พัฒนาระบบ (Application Developer) คำสั่งในการจัดการดังกล่าวคือ การสร้าง และทำลายโครงสร้างข้อมูล โดยคำสั่งของ SQL คือ CREATE เมื่อต้องการสร้างตารางข้อมูลหรือ DROP เมื่อต้องการยกเลิกหรือทำลายตารางข้อมูล

2. ภาษาที่เกี่ยวกับการควบคุมข้อมูล (Data Control Language : DCL) คำสั่งในภาษาดังกล่าวสนับสนุนให้สร้างการควบคุมความปลอดภัย เช่น การให้สิทธิในการใช้ข้อมูลทำได้โดยคำสั่ง GRANT หรือยกเลิกสิทธิทำได้โดยใช้คำสั่ง REVOKE และสนับสนุนการใช้คำสั่ง COMMIT สำหรับข้อมูลที่ทำงานอย่างเสร็จสิ้นสมบูรณ์ หรือคำสั่ง ROLLBACK เมื่อทำงานกับข้อมูลแล้วเกิดความขัดแย้งหรือไม่สมบูรณ์
3. ภาษาที่เกี่ยวกับการจัดการข้อมูล (Data Manipulation Language : DML) เป็นคำสั่งสำหรับพัฒนาและใช้งานทั่วไปในการเข้าถึงและจัดการข้อมูล ได้แก่ การสอบถามหรือเรียกค้นข้อมูล (Query) เพิ่มเติมข้อมูลใหม่ (Insert) การลบข้อมูล (Delete) หรือการแก้ไขเปลี่ยนแปลงข้อมูล (Update) ที่อยู่ในฐานข้อมูล

2.4 เทคโนโลยีฐานข้อมูลแบบไคลเอนต์ / เซิร์ฟเวอร์

ฐานข้อมูลแบบไคลเอนต์/เซิร์ฟเวอร์ (Client/Server) แบ่งการจัดการฐานข้อมูลเป็นสองระบบ คือ ไคลเอนต์พีซีซึ่งจะรันแอปพลิเคชันฐานข้อมูล และดาต้าเบสเซิร์ฟเวอร์ (Database Server) ซึ่งรัน DBMS ทั้งหมดหรือบางส่วน ไฟล์เซิร์ฟเวอร์ บนแลนยังเตรียมรีซอร์สที่ใช้ร่วมกัน เช่น เนื้อที่ดิสก์สำหรับแอปพลิเคชันและเครื่องพิมพ์ ดาต้าเบสเซิร์ฟเวอร์ หรือ บนเครื่องคอมพิวเตอร์ของตนเองแอปพลิเคชันบนไคลเอนต์พีซี ถูกเรียกว่า ระบบฟรอนต์เอนด์ (front-end system) ทำหน้าที่จัดการกับหน้าจอและการจัดการอินพุต/เอาต์พุตของผู้ใช้ ระบบส่วนแบ็กเอนด์ (back-end) บนดาต้าเบสเซิร์ฟเวอร์ ทำหน้าที่เก็บข้อมูลและการเข้าถึงดิสก์ ตัวอย่างเช่น ผู้ใช้ที่อยู่บนฟรอนต์เอนด์สร้างคำร้องขอ (คิวรี) เพื่อขอข้อมูลจากดาต้าเบสเซิร์ฟเวอร์ และแอปพลิเคชันฟรอนต์เอนด์ส่งคำร้องขอนี้ผ่านทางเน็ตเวิร์กไปยังเซิร์ฟเวอร์ ดาต้าเบสเซิร์ฟเวอร์ จะดำเนินการค้นหาอย่างแท้จริงและส่งเฉพาะข้อมูลที่เป็นคำตอบของคิวรีของผู้ใช้กลับไปเท่านั้น

2.5 มาตรฐานของระบบไคลเอนต์ / เซิร์ฟเวอร์ ทางด้าน SQL

มาตรฐานของ SQL ส่วนใหญ่ที่นิยมใช้กับระบบคอมพิวเตอร์ในแบบไคลเอนต์/เซิร์ฟเวอร์ ได้แก่ SQL/92 ซึ่งประกอบด้วยคำสั่ง Connect ที่ใช้ในการสร้างการติดต่อระหว่าง Client และ Server และคำสั่ง Disconnect ที่ใช้ในการเลิกการติดต่อระหว่าง ไคลเอนต์และเซิร์ฟเวอร์เนื่องจาก

โปรแกรมที่ใช้คำสั่ง SQL ที่ทำงานอยู่บนระบบคอมพิวเตอร์ในแบบไคลเอนต์/เซิร์ฟเวอร์จะต้องมี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้คำสั่ง Connect เพื่อสร้างการติดต่อระหว่าง ไคลเอนต์และเซิร์ฟเวอร์ ก่อน จึงจะสามารถใช้คำสั่ง SQL อื่นๆกับเซิร์ฟเวอร์ เพื่อเรียกใช้ข้อมูลในฐานข้อมูลได้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

ขอบเขตการพัฒนาระบบ

เป็นการกำหนดขอบเขตการทำงานเพื่อให้สามารถเข้าใจกระบวนการพัฒนาได้ชัดเจนและถูกต้องมากยิ่งขึ้น โดยขอบเขตการทำงานจะเป็นตัวกำหนดรายละเอียดและเนื้อหาที่ต้องทำโดยการวิเคราะห์ออกแบบและพัฒนา จำเป็นจะต้องมีการกำหนดขอบเขตในการพัฒนาก่อนเพื่อให้สามารถออกแบบระบบได้ถูกต้อง รวมถึงการประเมินระยะเวลาในการทำงานจะสามารถทำได้ รวมถึงใช้เป็นข้อตกลงในการทำงานได้อีกด้วย โดยจะทำการแบ่งระบบออกเป็นส่วน ๆ เพื่อให้สามารถเข้าใจระบบได้ง่ายยิ่งขึ้นซึ่งจะเป็นประโยชน์ในการวิเคราะห์ระบบเพื่อนำไปพัฒนาต่อไป

ขอบเขตการทำงานของระบบสามารถแบ่งออกเป็น 5 ส่วนหลัก ๆ ดังนี้

1. ส่วนในการรับ Request

- ซึ่งจะเริ่มจาก Requester ส่ง Request มาให้เพื่อให้ช่วยทำงานหรือแก้ปัญหา ระบบจะทำการออก เลข ID ให้ซึ่งเป็น Running no และมีการขอข้อมูลรายละเอียดของ Requester เพื่อใช้ในการติดต่อ ซึ่ง Requester ต้องใส่ Due Date หรือระยะเวลาในการทำงานไว้ให้เพื่อกำหนดการและเป็นข้อมูลให้ผู้ทำงานทราบ ซึ่งความสามารถของระบบควรมีดังต่อไปนี้
- ระบบต้องสามารถบันทึกข้อมูล Request ใหม่ ๆ ที่เข้ามาได้ อีกทั้งยังสามารถแก้ไขรายละเอียดใน Request เดิม ได้
 - สำหรับงานที่เป็นการ Support ถูกค่า อาจให้ลูกค้ากรอกข้อมูลเองในหน้า Website แต่ถ้าเป็น Project อาจให้ผู้รับผิดชอบ Project เป็นผู้กรอกข้อมูล Request
 - ผู้จ้างหรือบริษัทสามารถทราบสถานะงานที่ได้ Request มาว่าอยู่ในขั้นตอนใด

1. ลงเวลาการทำงาน

ผู้ทำงานต้องทำการ Login เข้าระบบและเลือกงานที่ต้องการทำ จากนั้นกดเริ่มเวลางาน เมื่อทำงานเสร็จเรียบร้อยแล้วต้องใส่รายละเอียดของงานที่ได้ทำ พร้อมทั้งลงเวลาสิ้นสุด ซึ่งระบบการลงเวลาควรสามารถทำงานได้ดังนี้

- สามารถเลือก Project ที่ต้องการทำงานแล้วเริ่มต้นลงเวลาการทำงานได้ และถ้าระบบเกิดมีข้อผิดพลาด หรือเครื่องคอมพิวเตอร์ที่ทำงานอยู่ไม่สามารถใช้งานได้ เวลาเริ่มทำงาน ที่ได้เริ่มต้นจับเวลาต้องยังคงอยู่ และสามารถ กดสิ้นสุดเวลาการทำงานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ถ้าการทำงานนั้นๆ ต้องออกไปทำนอกบริษัท โดยไม่สามารถติดต่อฐานข้อมูลได้ ผู้ทำงานต้องสามารถลงเวลาการทำงานได้ โดยต้องสามารถ Sync ข้อมูลมาที่เครื่องตัวเองและเมื่อกลับมาที่บริษัทต้องสามารถ Sync ข้อมูลลงสู่ Database ได้
- ถ้าในการทำงานไม่ได้บันทึกลงเวลาขณะทำงานได้ ต้องสามารถมาป้อนข้อมูลย้อนหลังได้
- ในการลงเวลาทุกครั้งผู้ทำงานต้องใส่ข้อมูลรายละเอียดของงานส่วนที่ทำได้แล้ว เพื่อใช้เป็นข้อมูลในการเรียกเก็บเงินลูกค้า และเป็นประโยชน์กับคนในทีมด้วย เพื่อจะได้ทราบว่างานในส่วนใดที่ทำได้บ้าง

2. กำหนดคนทำงาน

การกำหนดคนทำงานเป็นหน้าที่ของหัวหน้าทีม ซึ่งถ้าเป็นผู้ทำงานระดับปกติต้องไม่สามารถเข้าใช้งานในส่วนนี้ได้ ซึ่งหัวหน้าทีมที่เข้ามากำหนดคนทำงานต้องสามารถทราบปริมาณงานที่ผู้ทำงานแต่ละคนว่า ณ. ปัจจุบันมีงานอะไรบ้างที่ทำอยู่เพื่อจะได้สามารถกำหนดคนทำงานได้ถูก ซึ่งเมื่อกำหนดคนทำงานแล้ว งานที่ได้กำหนดต้องไปอยู่ใน My Tasks ของคนที่ได้กำหนดให้โดยอัตโนมัติ และสิ่งทีระบบต้องสามารถทำได้มีดังนี้

- สามารถแสดงรายชื่อของงานที่ยังไม่มีผู้รับผิดชอบเพื่อให้หัวหน้างานสามารถเข้าไปกำหนดคนทำงาน
- สามารถค้นหาข้อมูลได้
- แสดงจำนวนงานที่ผู้ทำงานแต่ละคนรับผิดชอบอยู่แล้วยังทำไม่เสร็จว่ามีจำนวนเท่าไร เพื่อให้หัวหน้างานประเมินว่าควรจะกำหนดงานที่เหลือให้ใครได้บ้าง เพื่อให้งานไม่ไป Load อยู่ที่คนใดคนหนึ่ง
- สามารถดูรายละเอียดของงานได้ว่างานได้ว่างานที่ยังค้างอยู่มีงานอะไรบ้างและจะต้องเสร็จภายในวันใด เพื่อสามารถประเมินผลได้ถูก
- สำหรับส่วนของการกำหนดคนทำงานต้องมีการเช็คสิทธิ์เฉพาะบางคนให้สามารถทำงานได้ โดยไม่ให้ผู้ทำงานทั่วไปสามารถมาทำงานในส่วนนี้ได้ ผู้ที่สามารถกำหนดคนทำงานได้ต้องเป็นหัวหน้าทีม หรือหัวหน้า Project

3. ออกบิลเพื่อเก็บเงิน

เมื่อทำงานเสร็จเรียบร้อยต้องมีการเปลี่ยนแปลงแก้ไขสถานะของงานว่าเสร็จเรียบร้อยแล้ว เพื่อให้สามารถดึงข้อมูลงานที่ได้ทำสำเร็จแล้วมาเป็นข้อมูลในการออกบิลเพื่อเรียกเก็บเงิน โดยจะทำการคำนวณระยะเวลาการทำงานและคูณกับอัตราที่ได้ตั้งไว้หรือออกเป็นในสรุป

จำนวนเงินพร้อมทั้งแจกแจงรายละเอียดงานที่ได้ทำตามระยะเวลาด้วย เพื่อให้ลูกค้าทราบถึงที่มาของจำนวนเงินที่เรียกเก็บ โดยกระบวนการทำงานนี้ต้องสามารถทำได้ดังนี้

- สามารถสรุปยอดรวมทั้งหมดที่ต้องชำระได้
- สามารถแจกแจงรายละเอียดให้ลูกค้าได้ว่าจำนวนเงินที่เรียกเก็บจากลูกค้า เกิดจากการทำงานใดบ้าง และเป็นเวลาเท่าไร
- พิมพ์ชื่อที่อยู่เบอร์โทรศัพท์ท่อกลูกค้าให้โดยอัตโนมัติที่ด้านบนของใบเรียกเก็บเงิน
- เลขเอกสารต้องเป็น Running no เพื่ออ้างอิงกับทางบัญชีได้
- มีชื่อที่อยู่ เบอร์โทรศัพท์ให้ติดต่อบริษัทกรณีที่มีปัญหา

4. ออกรายงาน

เป็นการสรุปรายงานในส่วน ๆ ต่าง ๆ ให้เห็นภาพรวมในแต่ละมุมมองเพื่อช่วยผู้บริหารในการนำไปวิเคราะห์และปรับปรุงให้ดียิ่งขึ้นต่อไป

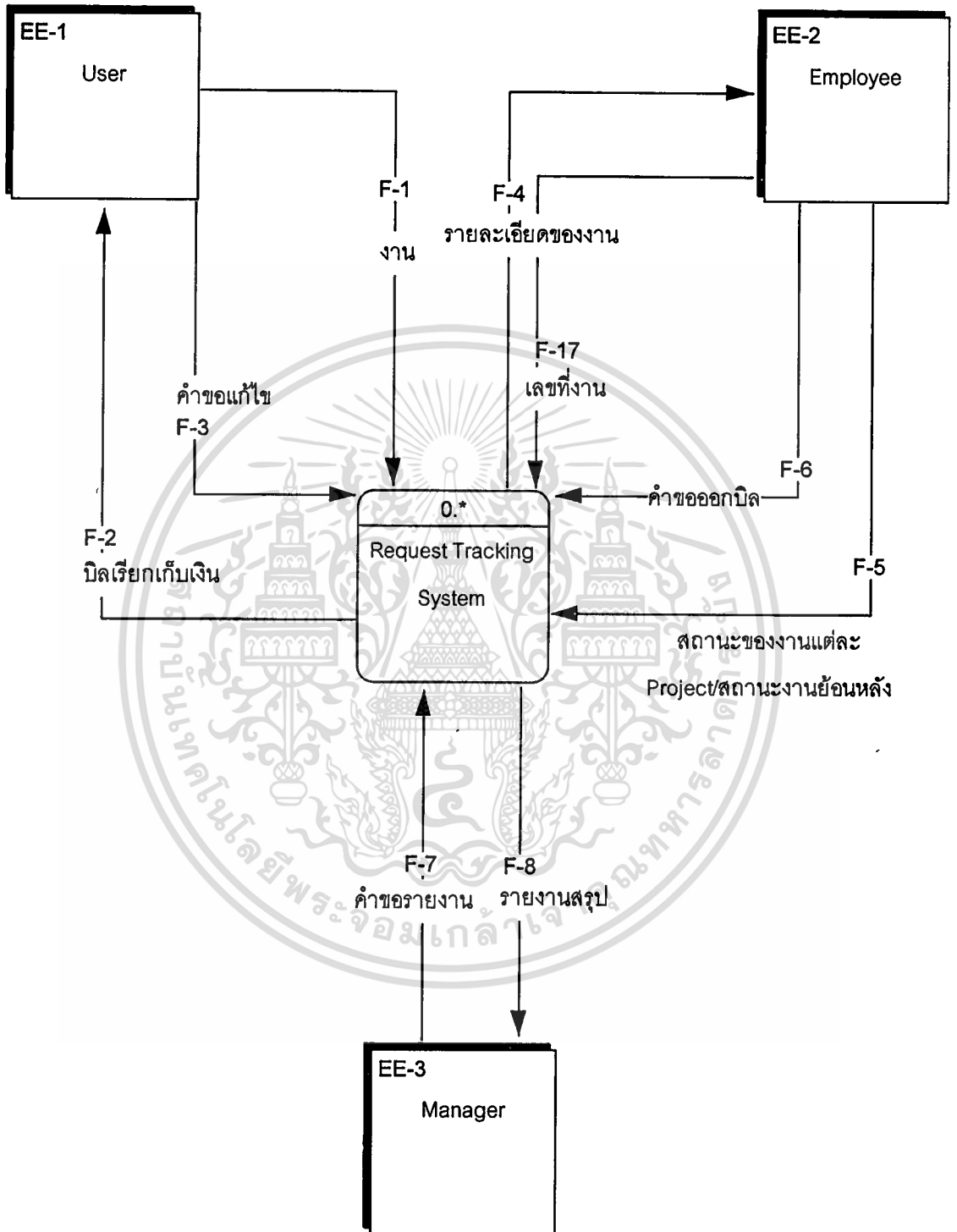
- สามารถดูรายละเอียดของงานได้ว่า งานนั้น ๆ มีคนที่ร่วมทำงานทั้งหมดกี่คนและแต่ละคนทำในส่วนใดบ้างและใช้เวลาเท่าไร
- สามารถดูรายละเอียดได้ว่าผู้ทำงานแต่ละคนทำงานอะไร ไปบ้าง
- สามารถดูรายละเอียดการทำงานต่าง ๆ ตามช่วงเวลาที่กำหนดได้
- สามารถดูได้ว่างานใดบ้างที่ไม่สามารถเสร็จทันตามกำหนดเวลา
- สามารถดูได้ว่างานใดบ้างเสร็จตามกำหนดเวลาหรือก่อนเวลา
- สามารถดูข้อมูลเพื่อสรุปผลได้ว่างานแต่ละ Project ใช้เวลาในการทำงานโดยเฉลี่ยเท่าไรเพื่อนำไปใช้ในการประเมินหรือตั้งราคางาน โดยได้ประเมินจากการทำงานจริง
- สามารถให้หัวหน้าหรือหัวหน้า Project วัดประสิทธิภาพของคนในทีมได้ว่ามีประสิทธิภาพการทำงานเป็นอย่างไร

บทที่ 4

การวิเคราะห์ระบบ

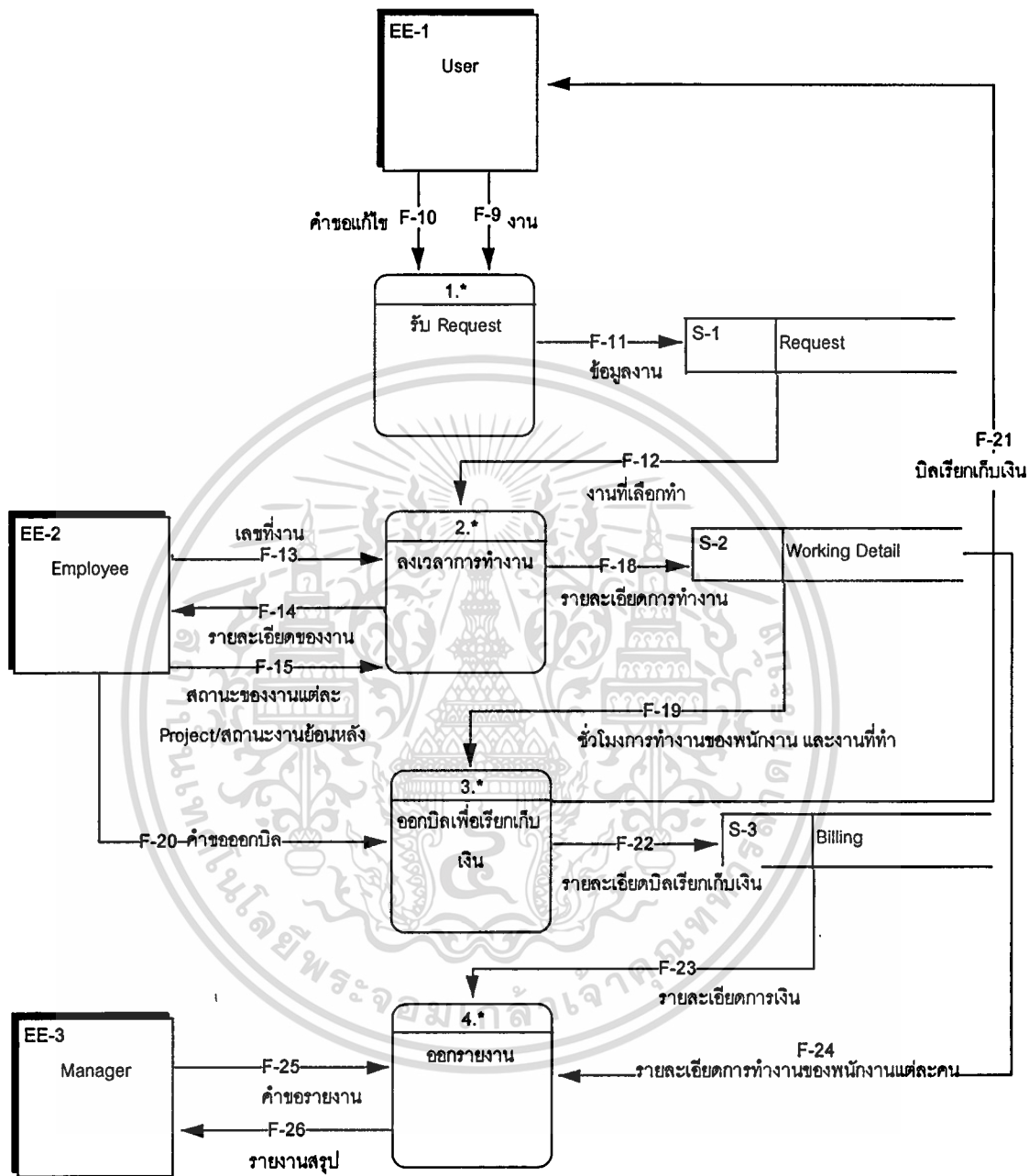
เพื่อแสดงแนวคิดในการออกแบบระบบ เพื่อให้ผู้วิเคราะห์ระบบ ผู้พัฒนา รวมถึงผู้ใช้งาน สามารถเข้าใจระบบงานได้ดียิ่งขึ้น และมองเห็นภาพรวมของโครงการในทิศทางเดียวกัน อีกทั้งยังเป็นประโยชน์ต่อผู้พัฒนารุ่นต่อ ๆ ไป สามารถนำเอกสารที่ได้ออกแบบระบบไว้มาศึกษาก็จะสามารถเข้าใจระบบได้อย่างรวดเร็ว โดยในเนื้อหาของบทนี้ได้แสดงการแปลงแนวคิดที่เป็นทฤษฎีมาสู่การปฏิบัติ โดยจะประกอบไปด้วย Context Diagram ดังแสดงในรูปที่ 4.1 DFD แสดงในรูปที่ 4.2 และ ER-Diagram ดังแสดงในรูปที่ 4.1





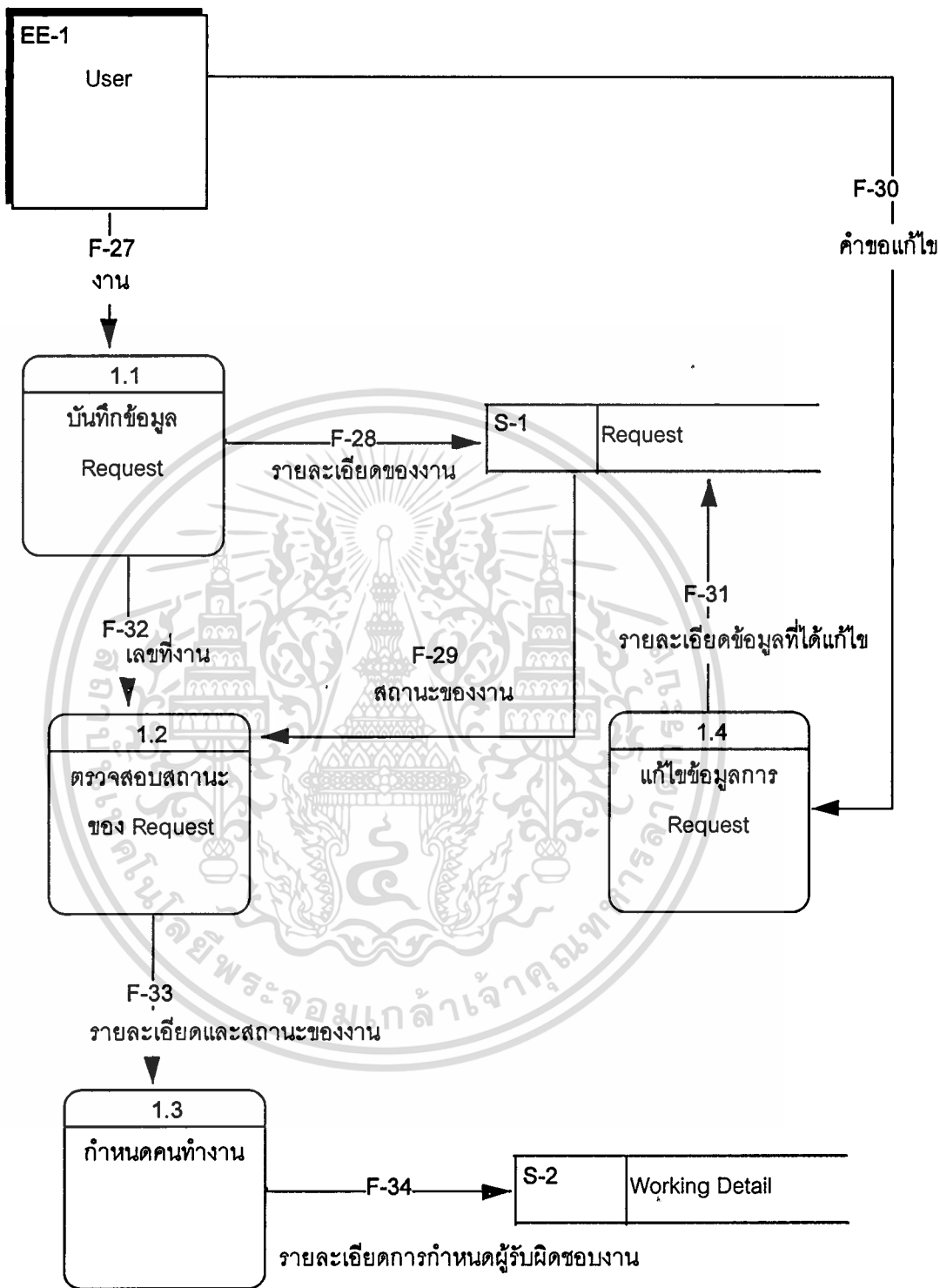
รูปที่ 4.1. Context Diagram ของระบบ Request Tracking System

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



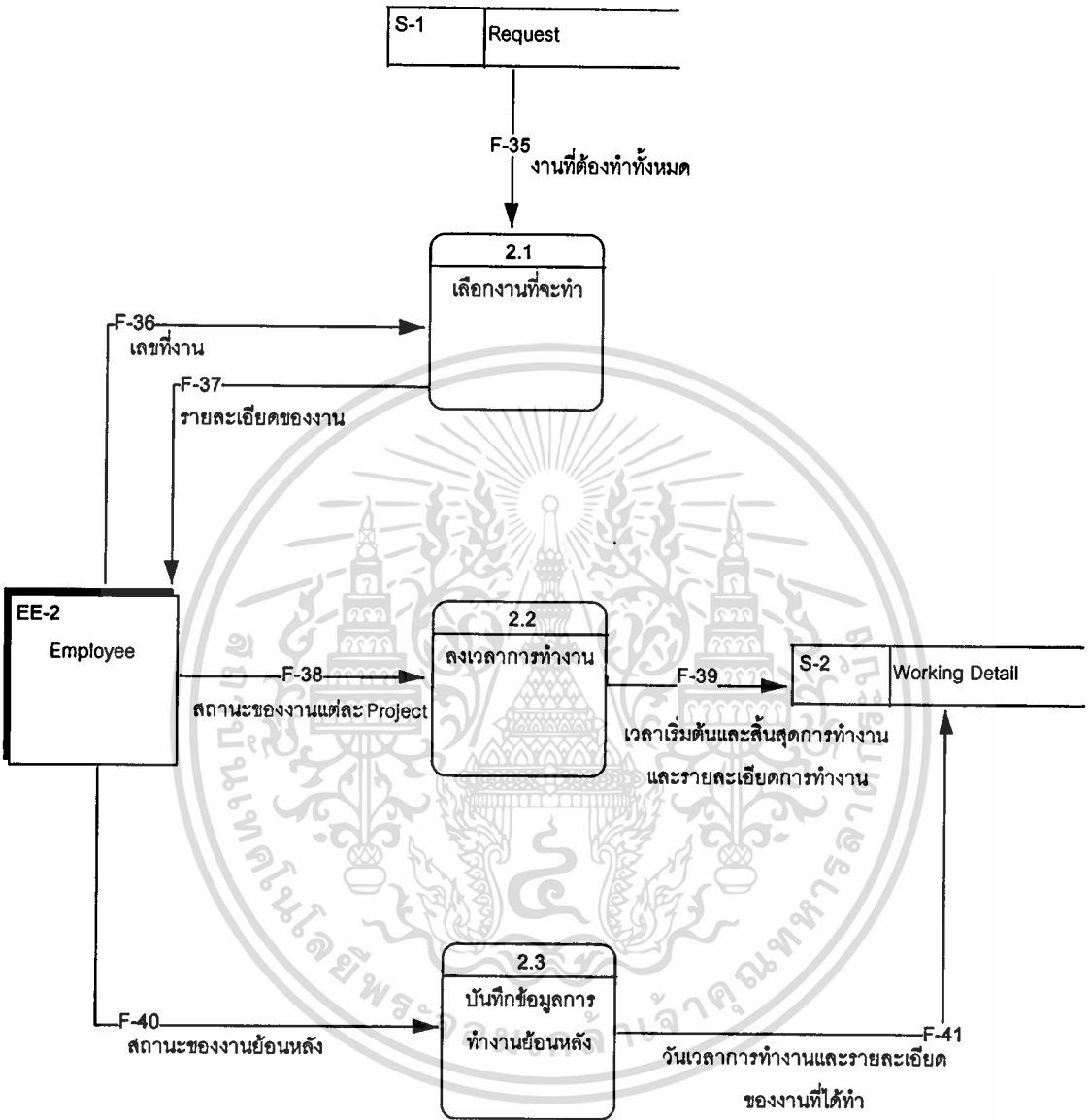
รูปที่ 4.2. Dataflow Diagram Level 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

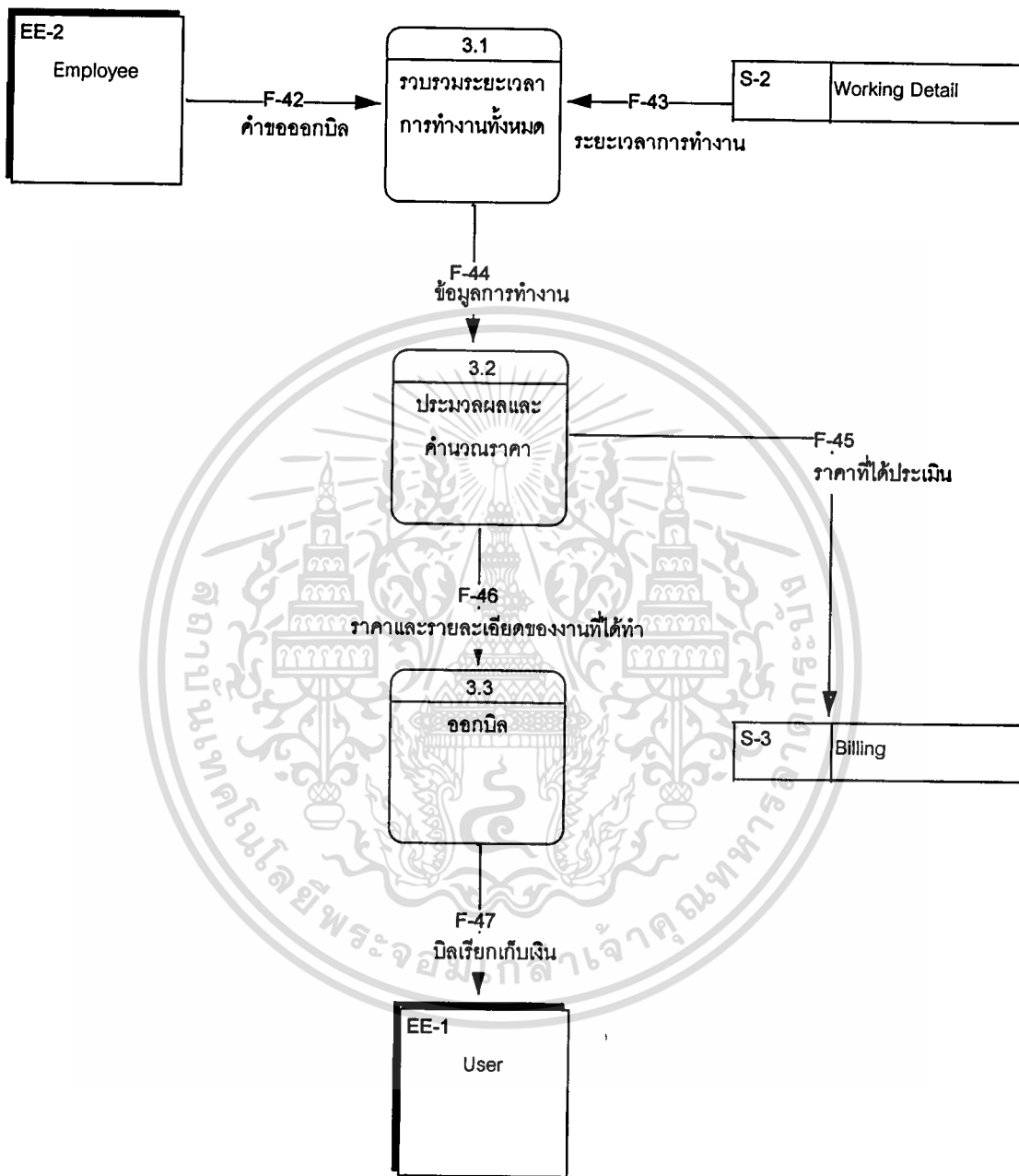


รูปที่ 4.3 Dataflow Diagram Level 2 ของ Process 1 (รับ Request)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

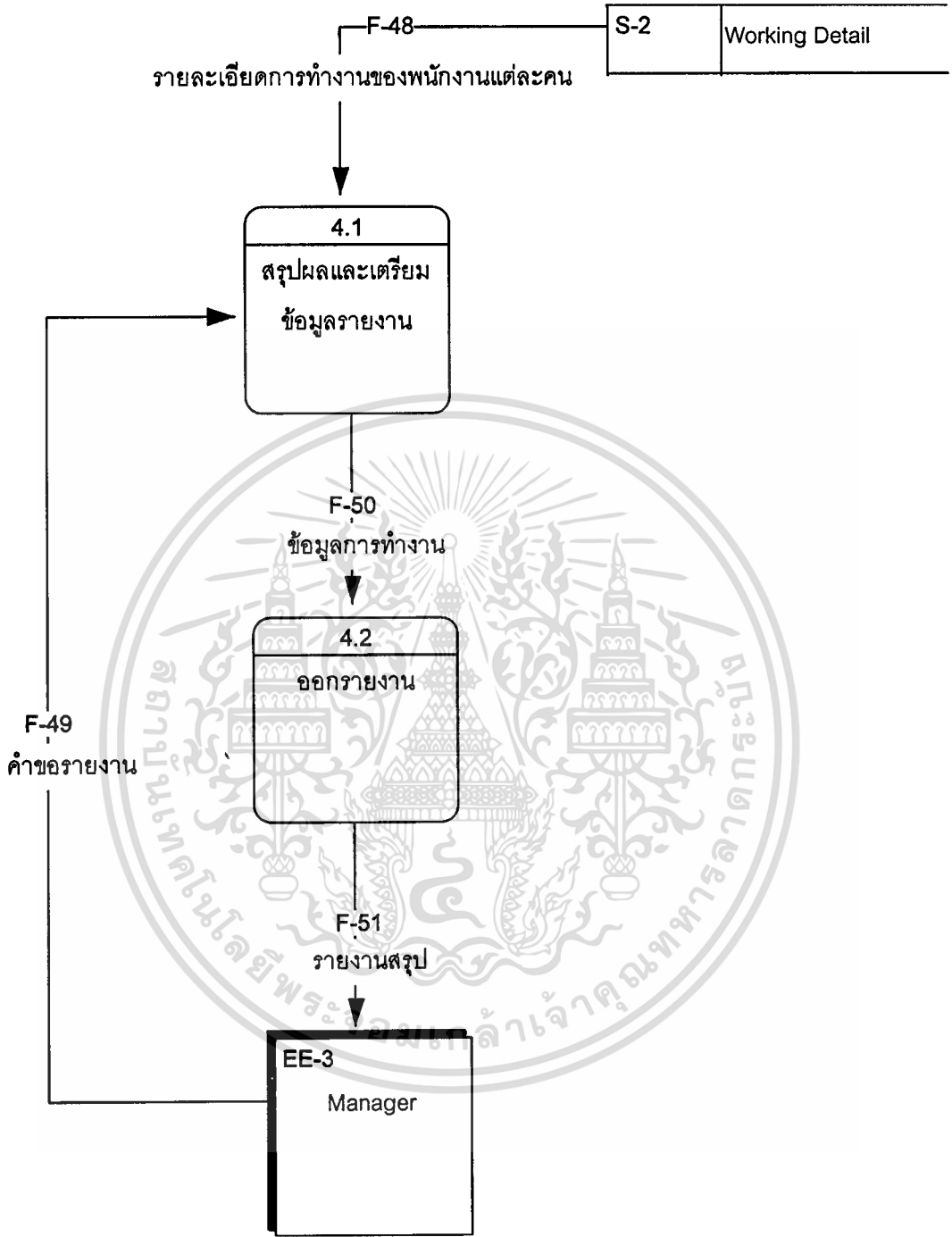


รูปที่ 4.4 Dataflow Diagram Level 2 ของ Process 2 (ลงเวลาการทำงาน)



รูปที่ 4.5 Dataflow Diagram Level 2 ของ Process 3 (ออกบิลเพื่อเก็บเงิน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 Dataflow Diagram Level 2 ของ Process 4 (ออกรายงาน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Element process Description

ProcessId:0

ProcessName:Tracking Request System

Description: ระบบสำหรับติดตามงาน โดยรับ Request ที่เข้ามาและสามารถติดตามงานได้ว่า ปัจจุบันสถานะของงานนั้นเป็นอย่างไร ใครเป็นผู้รับผิดชอบ และงานนั้นสำเร็จหรือไม่อย่างไร ใช้เวลาในการทำงานเท่าไร และสามารถสรุปเป็นรายงานเพื่อประเมินการทำงานของคนในทีมและประสิทธิภาพการทำงาน โดยรวมทั้งหมดของบริษัท

Element process Description

ProcessId:1

ProcessName: รับ Request

Description: สำหรับบันทึกข้อมูล Request ที่ได้รับมาจาก User และกำหนดคนเพื่อรับผิดชอบ Request ต่าง ๆ โดยสามารถเพิ่มและแก้ไข Request ได้ และสามารถแสดงรายชื่องานที่ยังไม่มีผู้รับผิดชอบเพื่อให้หัวหน้างานทราบเพื่อ กำหนดคนให้ทำงานต่อไป

Element process Description

ProcessId:2

ProcessName: ลงเวลาการทำงาน

Description: สำหรับเลือกงานที่จะทำและลงเวลาการเริ่มและสิ้นสุดการทำงาน เก็บบันทึกเอาไว้เพื่อใช้ในการประเมินประสิทธิภาพการทำงาน อีกทั้งสามารถบันทึกข้อมูลการทำงานย้อนหลังได้

Element process Description

ProcessId:3

ProcessName: ออกบิลเพื่อเก็บเงิน

Description: สำหรับรวบรวมระยะเวลาการทำงานทั้งหมด ประมวลผลและคำนวณราคา เพื่อออกบิลเก็บเงินกับลูกค้า

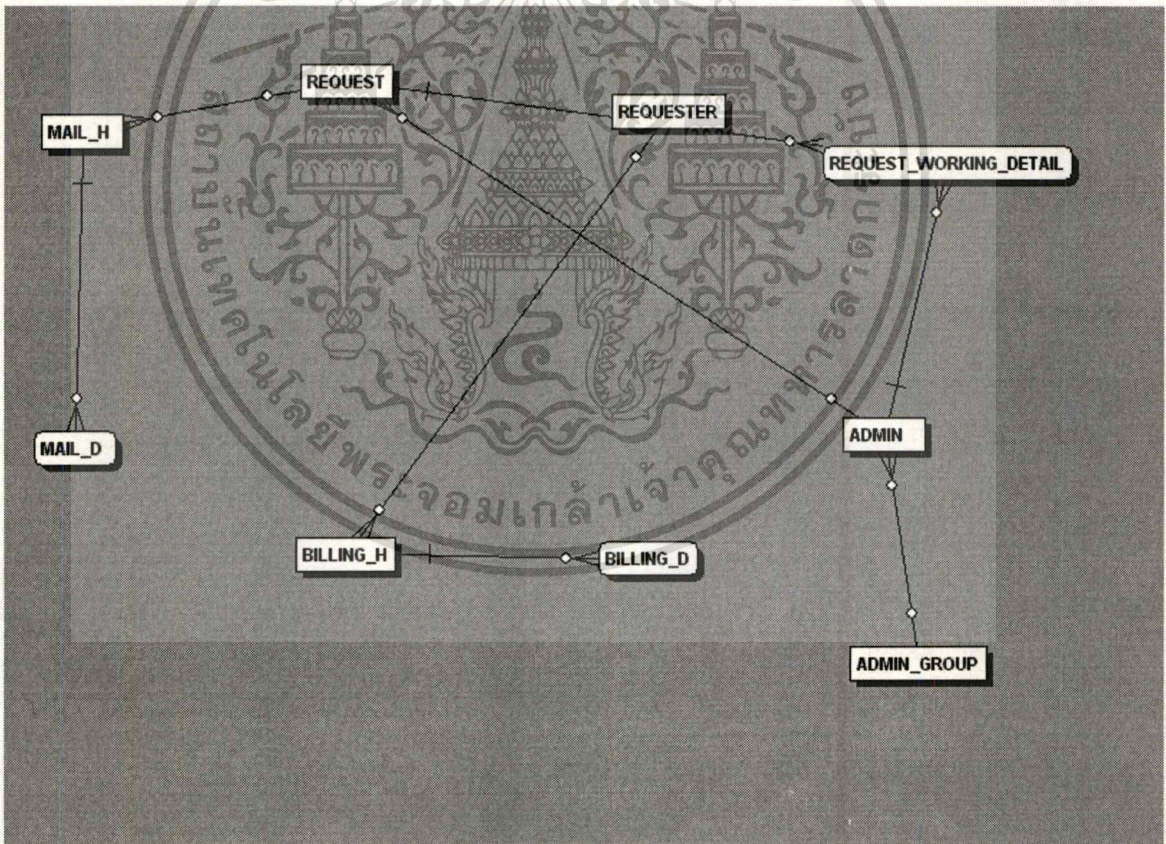
Element process Description

ProcessId:4

ProcessName: ออกรายงาน

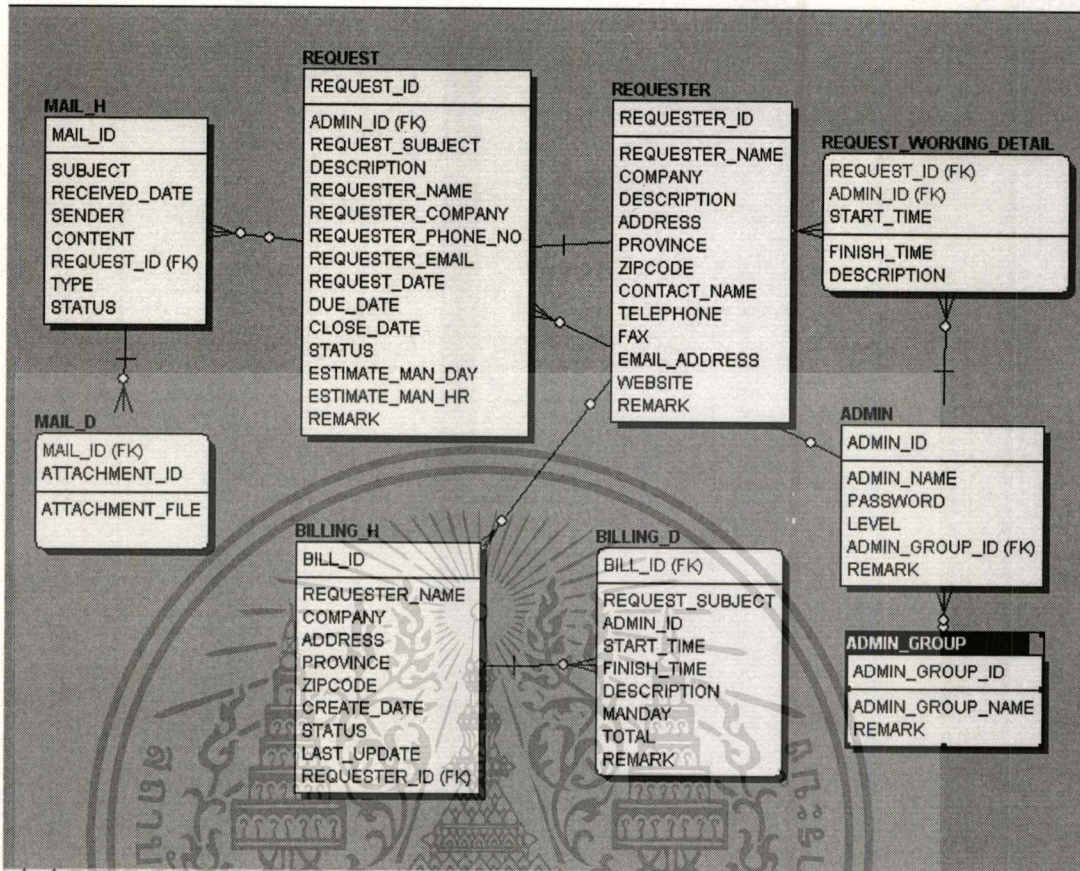
Description: สำหรับสรุปผลการทำงานทั้งหมดเพื่อออกเป็นรายงานให้ผู้บริหาร เพื่อใช้ในการประเมินผลงานของพนักงาน และประเมินประสิทธิภาพโดยรวม เพื่อเป็นข้อมูลในการวางแผนพัฒนาประสิทธิภาพในการทำงานต่อไป

ER-Diagram ดังแสดงในรูปที่ 4.7



รูปที่ 4.7 แผนภาพแสดงความสัมพันธ์ของข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.8 แผนภาพแสดงประเภทและความสัมพันธ์ของข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 คุณลักษณะต่าง ๆ ของ Entity Request_working_detail (รายละเอียดการทำงาน)

Column	Definition	Type	Null	Keys	Remarks
REQUEST_ID	รหัสคำขอ	CHAR(10)	N	PK	
ADMIN_ID	รหัสผู้ทำรายการ	CHAR(10)	N	PK,FK	Admin
START_TIME	เวลาเริ่มงาน	DATE	N	PK	
FINISH_TIME	เวลาสิ้นสุดงาน	DATE	Y		
DESCRIPTION	คำอธิบาย	VARCHAR(100)	N		

ตารางที่ 4.2 คุณลักษณะต่าง ๆ ของ Entity Request (คำขอ)

Column	Definition	Type	Null	Keys	Remarks
REQUEST_ID	รหัสคำขอ	CHAR(10)	N	PK	
ADMIN_ID	รหัสผู้ทำรายการ	CHAR(10)	Y	FK	Admin
REQUEST_SUBJECT	ชื่องาน	CHAR(10)	N		
DESCRIPTION	คำอธิบายงาน	VARCHAR(255)	Y		
REQUESTER_ID	รหัสเจ้าของงาน	CHAR(10)	N	FK	Requester
REQUEST_DATE	วันที่รับงาน	DATE	N		
DUE_DATE	วันที่ส่งมอบงาน	DATE	Y		
CLOSE_DATE	วันที่ปิดงาน	DATE	Y		
STATUS	สถานะ	CHAR(1)	Y		
ESTIMATE_MAN_D AY	วันที่ทำโดยประมาณ	NUMERIC(3, 0)	Y		
ESTIMATE_MAN_HR	ชั่วโมงที่ทำ โดยประมาณ	NUMERIC(2, 0)	Y		
REMARK	หมายเหตุ	VARCHAR(300)	Y		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 คุณลักษณะต่าง ๆ ของ Entity Mail_d (รายละเอียดเมลล์)

Column	Definition	Type	Null	Keys	Remarks
MAIL_ID	รหัสเมลล์	CHAR(10)	N	PK	
ATTACHMENT_ID	รหัสไฟล์แนบ	CHAR(10)	N		
ATTACHMENT_FILE	ไฟล์แนบ	VARCHAR(255)	Y		

ตารางที่ 4.4 คุณลักษณะต่าง ๆ ของ Entity Mail_h (หัวข้อเมลล์)

Column	Definition	Type	Null	Keys	Remarks
MAIL_ID	รหัสเมลล์	CHAR(10)	N	PK,FK	Mail_d
SUBJECT	หัวข้อ	VARCHAR(18)	Y		
RECEIVED_DATE	วันที่รับเมลล์	DATE	Y		
SENDER	ผู้ส่ง	VARCHAR(255)	Y		
CONTENT	เนื้อหา	CHAR(10)	Y		
REQUEST_ID	รหัสคำขอ	CHAR(10)	Y	FK	Request
TYPE	ประเภท	CHAR(1)	Y		
STATUS	สถานะ	CHAR(1)	Y		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.5 คุณลักษณะต่าง ๆ ของ Entity Admin(ผู้ดูแลระบบ)

Column	Definition	Type	Null	Keys	Remarks
ADMIN_ID	รหัสผู้ทำรายการ	CHAR(10)	N	PK	
ADMIN_NAME	ชื่อผู้ทำรายการ	VARCHAR(100)	Y		
PASSWORD	รหัสผ่าน	CHAR(10)	N		
LEVEL	ระดับในการใช้งาน	CHAR(10)	N		
ADMIN_GROUP_ID	รหัสกลุ่ม	CHAR(10)	N	FK	Admin_group
REMARK	หมายเหตุ	VARCHAR(100)	Y		

ตารางที่ 4.6 คุณลักษณะต่าง ๆ ของ Entity Requester(เจ้าของงาน)

Column	Definition	Type	Null	Keys	Remarks
REQUESTER_ID	รหัสเจ้าของงาน	CHAR(10)	N	PK	
REQUESTER_NAME	ชื่อเจ้าของงาน	VARCHAR(100)	Y		
COMPANY	บริษัท	VARCHAR(50)	Y		
DESCRIPTION	ลักษณะงาน	VARCHAR(100)	Y		
ADDRESS	ที่อยู่	VARCHAR(200)	Y		
PROVINCE	จังหวัด	VARCHAR(50)	Y		
ZIPCODE	รหัสไปรษณีย์	NUMBER	Y		
CONTACT_NAME	ชื่อผู้ติดต่อ	VARCHAR(50)	Y		
TELEPHONE	เบอร์โทรศัพท์	NUMBER	N		
FAX	แฟกซ์	NUMBER	Y		
EMAIL_ADDRESS	อีเมลแอดเดรส	VARCHAR(50)	Y		
WEBSITE	เว็บไซต์	VARCHAR(50)	Y		
REMARK	หมายเหตุ	VARCHAR(100)	Y		

ตารางที่ 4.7 คุณลักษณะต่าง ๆ ของ Entity Billing_h (ส่วนหัวใบเรียกเก็บเงิน)

Column	Definition	Type	Null	Keys	Remarks
BILL_ID	รหัสใบเรียกเก็บเงิน	CHAR(10)	N	PK,FK	Billing_d
REQUESTER_ID	รหัสเจ้าของงาน	CHAR(10)	N	FK	Requester
REQUESTER_NAME	ชื่อเจ้าของงาน	VARCHAR(100)	N		
COMPANY	บริษัท	VARCHAR(50)	Y		
ADDRESS	ที่อยู่	VARCHAR(200)	Y		
PROVINCE	จังหวัด	VARCHAR(50)	Y		
ZIPCODE	รหัสไปรษณีย์	NUMBER	Y		
CREATE_DATE	วันที่ออกใบเรียกเก็บเงิน	DATE	N		
STATUS	สถานะ	CHAR(1)	N		
LAST_UPDATE	วันที่แก้ไขล่าสุด	DATE	N		

ตารางที่ 4.8 คุณลักษณะต่าง ๆ ของ Entity Billing_d(รายละเอียดใบเรียกเก็บเงิน)

Column	Definition	Type	Null	Keys	Remarks
BILL_ID	รหัสใบเรียกเก็บเงิน	CHAR(10)	N	PK	
REQUEST_ID	รหัสเข้าของงาน	CHAR(10)	N	FK	Request
REQUEST_SUBJECT	ชื่องาน	CHAR(10)	N		
ADMIN_ID	รหัสผู้ทำรายการ	CHAR(10)	N	FK	Admin
START_TIME	เวลาเริ่มงาน	DATE	N		
FINISH_TIME	เวลาสิ้นสุดงาน	DATE	Y		
DESCRIPTION	คำอธิบาย	VARCHAR(100)	N		
MANDAY	จำนวนชั่วโมงทำงาน	NUMBER	N		
TOTAL	จำนวนชั่วโมงรวม	NUMER	Y		
REMARK	หมายเหตุ	VARCHAR(200)	Y		

ตารางที่ 4.9 คุณลักษณะต่าง ๆ ของ Entity Admin_group(กลุ่ม)

Column	Definition	Type	Null	Keys	Remarks
ADMIN_GROUP_ID	รหัสกลุ่ม	CHAR(10) (PK)	N	PK	
ADMIN_GROUP_NAME	ชื่อกลุ่ม	VARCHAR(100)	Y		
REMARK	หมายเหตุ	VARCHAR(100)	Y		

Table Description**Table Id : ตารางที่ 4.1****Table Name: Request_working_detail**

Description: สำหรับเก็บข้อมูลการทำงานทั้งหมดที่ ผู้ทำงานแต่ละคนได้ทำ โดยจะมีเวลาเริ่มทำงานเวลาสิ้นสุดการทำงานและรายละเอียดของงานที่ได้ทำ เพื่อนำไปใช้เป็นข้อมูลในการออกใบเรียกเก็บเงินรวมทั้งเป็นข้อมูลในรายงานสำหรับผู้บริหาร

Table Description**Table Id : ตารางที่ 4.2****Table Name: Request**

Description: สำหรับเก็บข้อมูลงานที่ Requester หรือ Customer ได้ขอมา โดยจะเก็บรายละเอียดของงานที่ต้องทำ บริษัท และชื่อเจ้าของงาน พร้อมทั้งมีสถานะของงานว่างานนั้น ๆ อยู่ในขั้นตอนใดบ้าง

โดยสถานะของงานแบ่งได้ดังนี้ P = Pending คืองานนี้ยังไม่มีผู้รับผิดชอบ
 W = Working คืองานนี้กำลังดำเนินการอยู่
 C = Complete คืองานนี้ได้ทำเสร็จสิ้นสมบูรณ์แล้ว
 X = Cancel คืองานนี้ได้ทำการยกเลิกแล้ว

และเก็บข้อมูล Request date และ Due date เพื่อนำไปใช้เป็นข้อมูลในการบริหารและควบคุมเวลาการทำงาน

Table Description**Table Id : ตารางที่ 4.3****Table Name: Mail_d**

Description: ออกแบบไว้สำหรับเก็บข้อมูล Request ที่ส่งมาทางเมล โดยนำข้อมูลเกี่ยวกับการ Attach file มาใส่ที่ Table นี้เพื่อเก็บเป็นข้อมูลเพื่อนำไปใช้งานในภายหลัง

Table Description**Table Id : ตารางที่ 4.4****Table Name: Mail_h**

เอกสารนี้จัดทำขึ้นเพื่อการศึกษานี้เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Description: ออกแบบไว้สำหรับเก็บข้อมูล Request ที่ส่งมาทางเมล โดยจะนำข้อมูลผู้ส่ง ชื่อเมล และรายละเอียดภายในเมลมาเก็บไว้เพื่อนำไปใช้ในการนำไปสร้างเป็น Request โดยอัตโนมัติ

Table Description

Table Id : ตารางที่ 4.5

Table Name: Admin

Description: สำหรับเก็บข้อมูลผู้เข้าใช้งานระบบประกอบไปด้วยผู้ทำงาน ลูกค้า และหัวหน้า Project โดยแต่ละคนที่เข้าทำงานจะมีระดับในการเข้าใช้งานแตกต่างกันไป เช่น ลูกค้าก็สามารถเข้ามาคีย์ข้อมูล Request ได้อย่างเดียวไม่สามารถดูรายละเอียดในส่วนอื่นได้ เป็นต้น โดยผู้ทำงานแต่ละคนต้องมี การใส่กลุ่มด้วยว่าเป็นผู้ทำงานในส่วนใด เช่น ทำในส่วน Network หรือ Developer เพื่อนำไปใช้ในการสรุปผลงาน

Table Description

Table Id : ตารางที่ 4.6

Table Name: Requester

Description: สำหรับเก็บข้อมูลเจ้าของงานหรือบริษัทที่ว่าจ้างให้ทำงาน โดยจะเก็บรายละเอียดของบริษัท ที่อยู่ รวมถึงเบอร์ โทรศัพท์และบุคคลที่สามารถติดต่อได้ เพื่อใช้เป็นฐานข้อมูลลูกค้า และมี ประโยชน์ในการนำไปออกใบเรียกเก็บเงิน เมื่องานเสร็จสิ้น

Table Description

Table Id : ตารางที่ 4.7

Table Name: Billing_h

Description: สำหรับเก็บข้อมูลส่วนหัวของใบเรียกเก็บเงินเช่น รายละเอียดของเจ้าของงานหรือบริษัทที่ผู้ว่าจ้าง สถานะของใบเรียกเก็บเงินนั้น โดยข้อมูลในส่วนนี้จะนำไปใช้ในการออกใบเรียกเก็บเงินและรายงานต่อไป

Table Description

Table Id : ตารางที่ 4.8

Table Name: Billing_d

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Description: สำหรับเก็บข้อมูลรายละเอียดการออกไปเรียกเก็บเงิน โดยจะคำนวณราคาและคิ่งข้อมูล มาเพื่อออกไปเรียกเก็บเงินกับลูกค้า

Table Description

Table Id : ตารางที่ 4.9

Table Name: Admin_Group

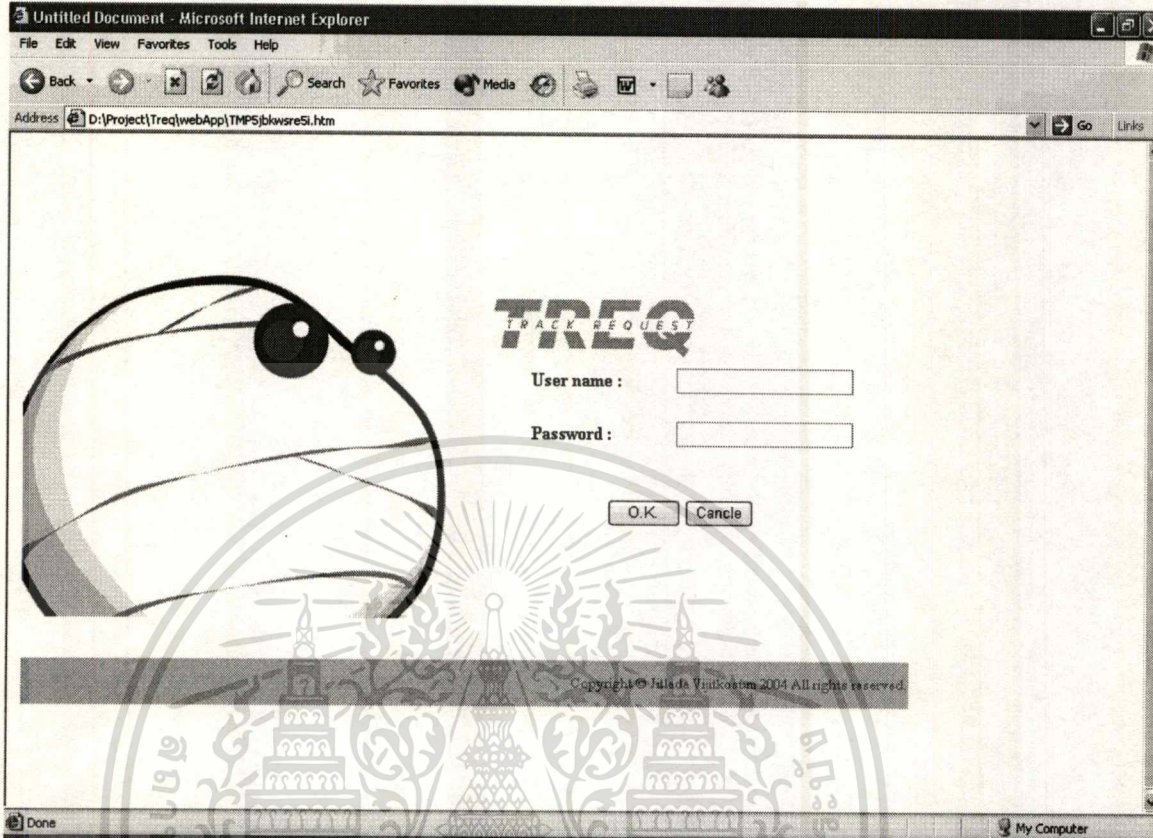
Description: สำหรับเก็บข้อมูลกลุ่มของพนักงานว่าประกอบไปด้วยกลุ่มใดบ้างเช่น กลุ่ม Network กลุ่ม Developer เป็นต้น เพื่อใช้ในการแบ่งกลุ่มของพนักงาน



บทที่ 5

การพัฒนาระบบ

เพื่อแสดงหน้าจอการใช้งานระบบ ตามที่ได้มีการวิเคราะห์และออกแบบระบบมา ซึ่งในการพัฒนาระบบนี้ได้แบ่งการทำงานออกเป็น 2 ส่วนคือ ส่วนที่เป็น Application และส่วนที่เป็น Web โดยส่วนที่เป็น Web เพื่อใช้กับงานที่บุคคลอื่นสามารถเข้าถึงได้ เพื่อเป็นการสะดวกในการใช้งานไม่ต้องติดตั้ง เช่น หน้าขอ Request หรือ Report ต่าง ๆ เพื่อให้หัวหน้างานสามารถเข้ามาดูข้อมูลได้ตลอดเวลาไม่ว่าจะอยู่ที่ใด โดยจะพัฒนาโดยใช้ ภาษา Html ,JSP และ Java ส่วนที่ 2 เป็นส่วนของ Application เพื่อใช้ในการบันทึกเวลา โดยตัวโปรแกรมใช้ Java Swing เขียน เพื่อให้ผู้ทำงานสามารถใช้งานได้ง่ายขึ้น



รูปที่ 5.1 แผนภาพแสดงหน้าจอการ Login

หน้าจอกการทำงาน: หน้าจอ Login

จุดประสงค์: ใช้สำหรับตรวจสอบสิทธิ์การเข้าใช้งานโปรแกรม

วิธีการใช้งาน: ใส่รหัสผู้ใช้งานและรหัสผ่าน จากนั้นกดปุ่ม ok เพื่อเป็นการตรวจสอบสิทธิ์ก่อน

เข้าใช้งาน

Track Request

Home Request All Tasks My Tasks Assign Tasks Report

Request Id: Status: Pending

Subject: Type: Project

Descriptions:

Requester: Company:

Phone: Email:

Request date: Due date:

Estimate man days: Estimate man hours:

Remarks:

Submit Cancel

Track Request 2004. Contact: webmaster@admin.net

รูปที่ 5.2 แผนภาพแสดงหน้าจอรับ Request

หน้าจอรองาน: หน้าจอ Request

จุดประสงค์: ใช้สำหรับบันทึก Request ที่เข้ามา

วิธีการใช้งาน: กรอกข้อมูลรายละเอียดของงาน และรายละเอียดเจ้าของงาน จากนั้นกดปุ่ม Submit เพื่อทำการบันทึกข้อมูล

Track Request

Home Request All Tasks My Tasks Assign Tasks Report

RequestId: Subject: Search

Requester:

RequestId	Subject	Requester	Request date	Due date
<input type="checkbox"/> 1001	Test1	Ms.Test	01/01/2004	16/01/2004
<input type="checkbox"/> 1002	Test2	Ms.Test2	05/01/2004	20/01/2004

Poke

Track Request 2004, Contact: webmaster@gadmip.net

Macromedia Dreamweaver MX - [:: Track Request :: (webApp/Alljob.jsp)]

รูปที่ 5.3 แผนภาพแสดงหน้าของงานทั้งหมด

หน้าการทำงาน: หน้าจอ All Tasks

จุดประสงค์: ใช้สำหรับแสดงงานทั้งหมดที่ยังไม่มีผู้รับผิดชอบ เพื่อให้ผู้ที่เกี่ยวข้องกับงานนั้นเข้ามาเลือกงานเพื่อจะทำ

วิธีการใช้งาน: ที่หน้าจอจะแสดงงานทั้งหมดที่ยังไม่มีผู้รับผิดชอบ ผู้ทำงานต้องเข้ามาดูเพื่อเลือกงานที่เกี่ยวข้องกับตัวเอง ซึ่งถ้าต้องการค้นหาสามารถป้อนข้อมูลที่ตนเองทราบจากนั้นกดปุ่ม Search เพื่อให้ระบบค้นหาข้อมูลให้ เมื่อพบงานที่ต้องทำแล้วให้กดปุ่ม Poke เพื่อเป็นการบอกผู้ทำงานคนอื่น ๆ ว่าตนรับผิดชอบงานนั้น ๆ รายชื่อของงานนั้นจะเข้าไปอยู่ที่ My Tasks ของผู้ทำงาน

Track Request

Home Request All Tasks My Tasks Assign Tasks Report

Requester: Status: Pending Search

Request date: Due date:

RequestId	Subject	Requester	Request date	Due date	Status
1001	Test1	Ms. Test	01/01/2004	16/01/2004	Working
1002	Test2	Ms-Test2	05/01/2004	20/01/2004	Complete

OK Cancel Close Job

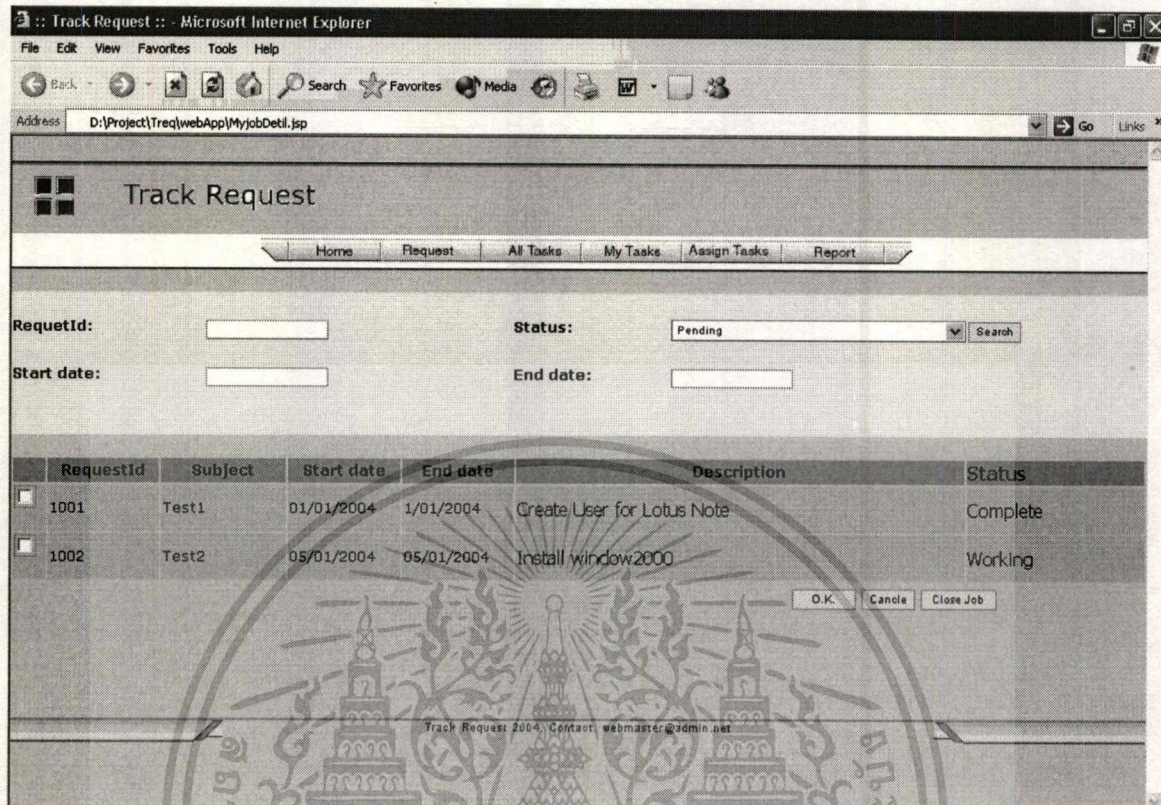
Track Request 2004. Contact: webmaster@admin.net

รูปที่ 5.4 แผนภาพแสดงหน้าจอรายชื่องานที่ตนรับผิดชอบ

หน้าจอรายงาน: หน้าจอ My Tasks

จุดประสงค์: ใช้สำหรับแสดงงานทั้งหมดที่ตนเป็นผู้รับผิดชอบ เพื่อจัดลำดับความสำคัญของงานทั้งหมดที่มีอยู่ และวางแผนการทำงาน

วิธีการใช้งาน: ที่หน้าจอจะแสดงงานทั้งหมดที่ตนรับผิดชอบ โดยถ้าต้องการเริ่มทำงานให้เลือกงานที่ต้องการทำ จากนั้นกดปุ่ม Start Time เพื่อเป็นการบันทึกเวลาการทำงาน



รูปที่ 5.5 แผนภาพแสดงหน้าจอรายละเอียดงานที่คนรับผิดชอบ

หน้าจการทำงาน: หน้าจอ My Tasks Detail

จุดประสงค์: ใช้สำหรับแสดงหรือบันทึกรายละเอียดการทำงาน

วิธีการใช้งาน: สำหรับบันทึกข้อมูลที่ได้ทำงานนั้น ๆ ย้อนหลัง ซึ่งประกอบด้วย เวลาเริ่มงาน เวลาสิ้นสุด และรายละเอียดการทำงานนั้น ๆ

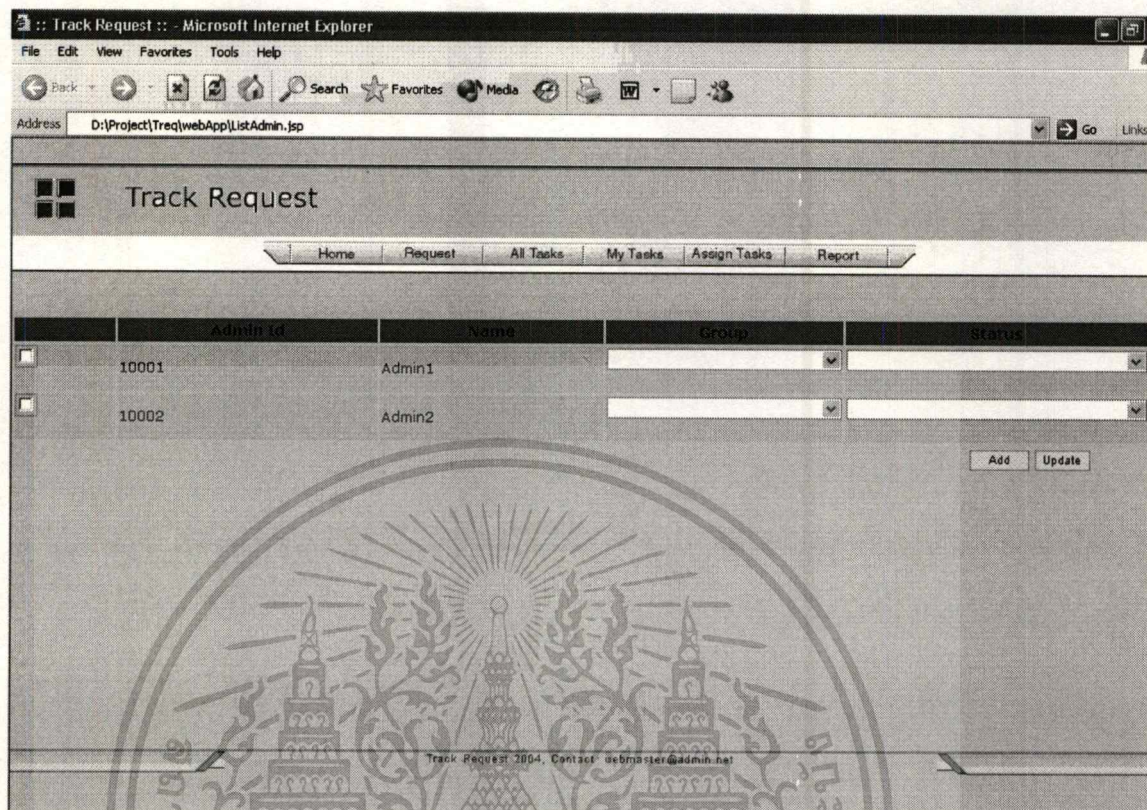
RequestId	Subject	Requester	Request date	Due date	Worker	Project	Job
<input type="checkbox"/> 1001	Test1	Ms.Test	01/01/2004	16/01/2004	<input type="checkbox"/> Ms.Test	1	5
<input type="checkbox"/> 1002	Test2	Ms.Test2	05/01/2004	20/01/2004	<input type="checkbox"/> Ms.Test2	1	2

รูปที่ 5.6 แผนภาพแสดงหน้าจอการกำหนดงานให้รับผิดชอบ

หน้าจการทำงาน: หน้าจอ Assign Tasks

จุดประสงค์: ใช้สำหรับกำหนดงานให้คนในทีม โดยจะแสดงงานทั้งหมดที่ยังไม่มีผู้รับผิดชอบ และจำนวนงานทั้งหมดที่ผู้ทำงานรับผิดชอบอยู่ โดยสรุปตามประเภทของงาน เพื่อประเมินว่าควรกำหนดงานนั้นให้ผู้ทำงานคนใด

วิธีการใช้งาน: ที่หน้าจอจะแสดงงานทั้งหมดที่ยังไม่มีผู้รับผิดชอบ และถ้าต้องการกำหนดงานให้ผู้ทำงานคนใดให้คลิกเลือกงาน ด้านซ้าย และเลือกคนที่ทำงาน จากนั้นกดปุ่ม ok



รูปที่ 5.7 แผนภาพแสดงหน้าจอแสดงรายชื่อ Admin

หน้าจอกำหนดงาน: หน้าจอ List Admin

จุดประสงค์: ใช้สำหรับแสดงรายชื่อพนักงานทั้งหมด

วิธีการใช้งาน: เมื่อต้องการเพิ่มพนักงานใหม่ให้กดปุ่ม Add หรือถ้าต้องการแก้ไขรายละเอียดของพนักงานนั้น กดปุ่ม Updte เพื่อทำการแก้ไข

Track Request

Home Request All Tasks My Tasks Assign Tasks Report

Name:

Admin Id:

Password:

Status: Pending

Remarks:

O.K. Cancel

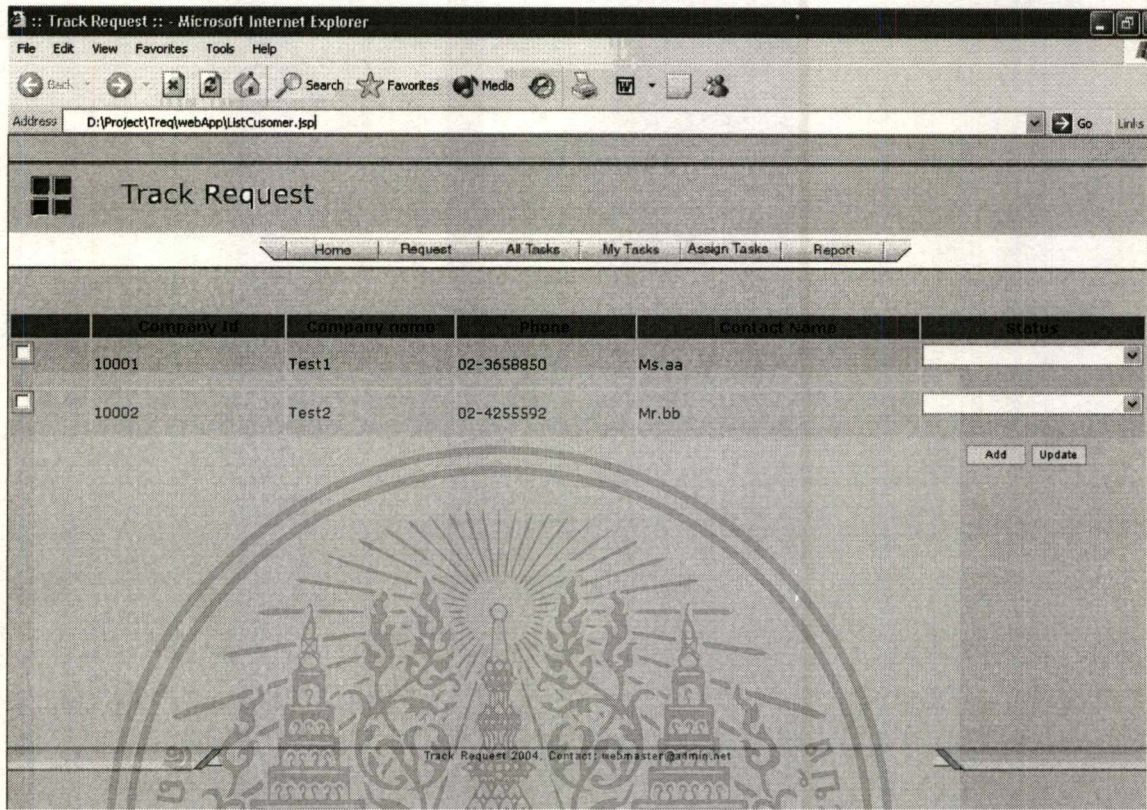
Track Request 2004. Contact: webmaster@admin.net

รูปที่ 5.8 แผนภาพแสดงหน้าจอเพิ่ม Admin

หน้าจการทำงาน:หน้าจอ Add Admin

จุดประสงค์:ใช้สำหรับเพิ่มรายชื่อพนักงาน หรือแก้ไขพนักงาน

วิธีการใช้งาน: เมื่อกดปุ่ม Add จากหน้าจอ List Admin จะเป็นการเพิ่ม Adminใหม่ให้กรอกข้อมูล ชื่อ รหัส พร้อมด้วย รหัสผ่าน และระบุกลุ่มของพนักงาน ระดับในการเข้าใช้งาน จากนั้นกดปุ่ม o.k.



รูปที่ 5.9 แผนภาพแสดงหน้าจอแสดงรายชื่อ Company

หน้าจอกำหนดงาน: หน้าจอ List Company

จุดประสงค์: ใช้สำหรับแสดงรายชื่อบริษัททั้งหมด

วิธีการใช้งาน: เมื่อต้องการเพิ่มบริษัทใหม่ให้กดปุ่ม Add หรือถ้าต้องการแก้ไขรายละเอียดของบริษัทนั้น กดปุ่ม Updte เพื่อทำการแก้ไข



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Track Request

Home Master Request All Tasks My Tasks Assign Tasks Report

Company Id: Company Name:

Description:

Address:

Province: Zipcode:

Phone: Fax:

E-Mail: Website:

Contact name: Remarks:

Submit Cancel

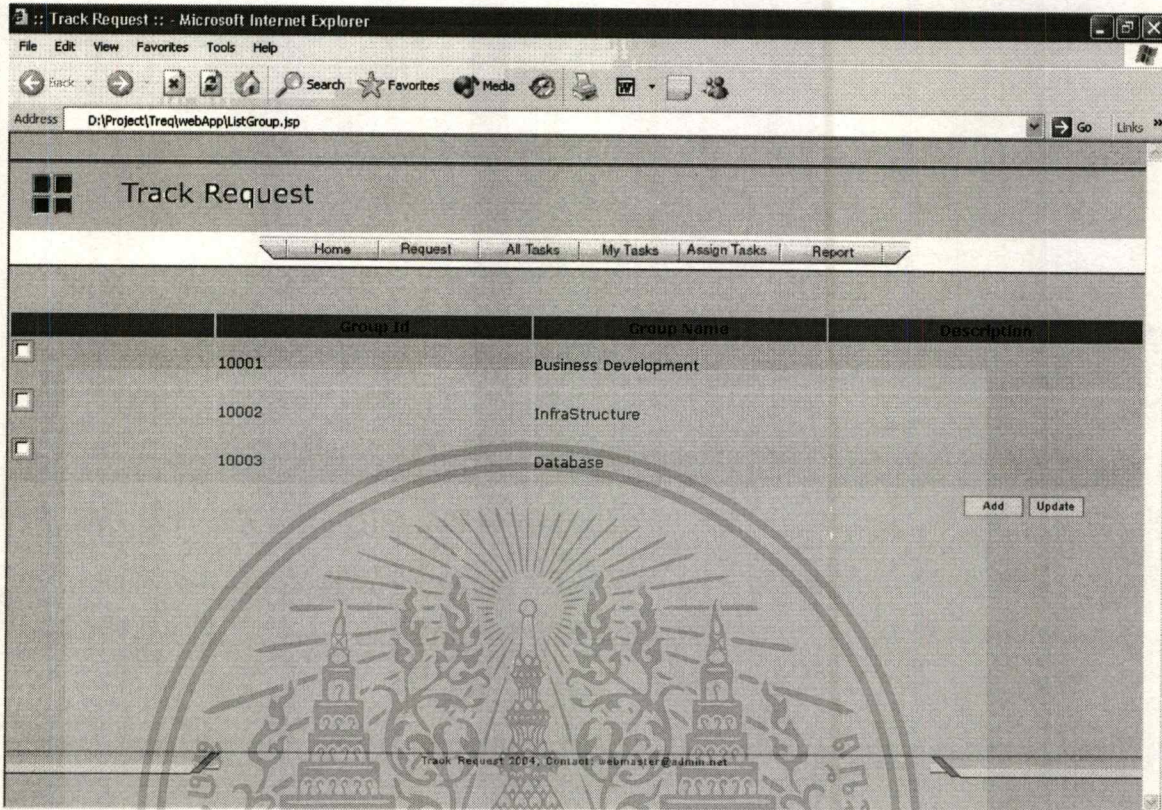
Track Request 2004, Contact: webmaster@admle.net

รูปที่ 5.10 แผนภาพแสดงหน้าจอเพิ่ม Company

หน้าจอกำหนดงาน: หน้าจอ Add Company

จุดประสงค์: ใช้สำหรับเพิ่มรายชื่อบริษัท หรือแก้ไขบริษัท

วิธีการใช้งาน: เมื่อกดปุ่ม Add จากหน้าจอ List Company จะเป็นการเพิ่ม Company ใหม่ให้กรอกข้อมูลชื่อบริษัท ที่อยู่ ผู้ติดต่อ เบอร์โทรศัพท์ จากนั้นกดปุ่ม o.k.

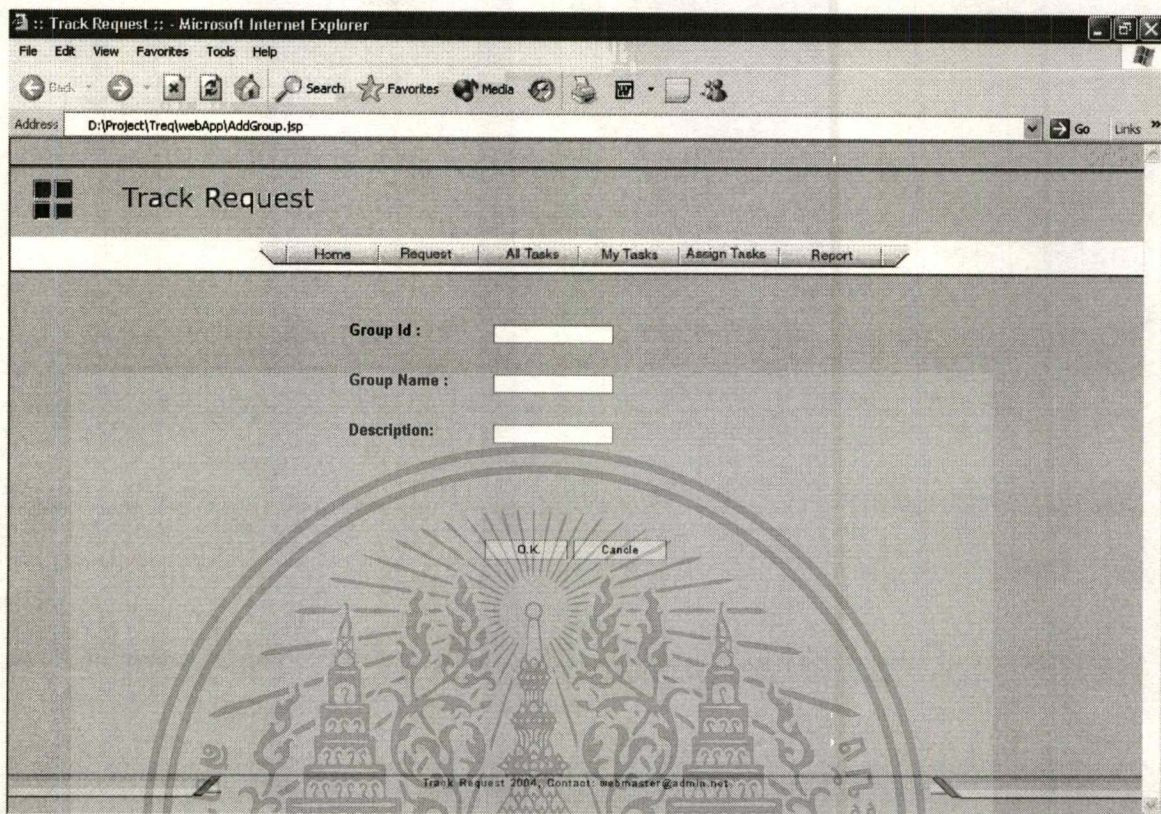


รูปที่ 5.11 แผนภาพแสดงหน้าจอแสดงรายชื่อ Group

หน้าจการทำงาน: หน้าจอ List Group

จุดประสงค์: ใช้สำหรับแสดงรายชื่อกลุ่มหรือแผนกที่แบ่งให้กับผู้ทำงาน

วิธีการใช้งาน: เมื่อต้องการเพิ่มกลุ่มใหม่ให้กดปุ่ม Add หรือถ้าต้องการแก้ไขรายละเอียดของกลุ่มนั้น กดปุ่ม Updte เพื่อทำการแก้ไข



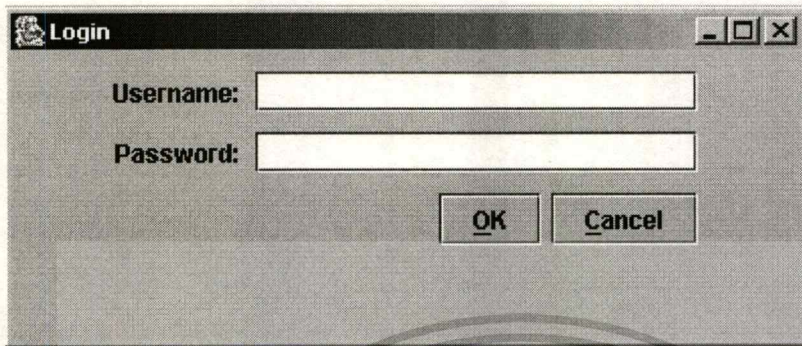
รูปที่ 5.12 แผนภาพแสดงหน้าจอเพิ่ม Group

หน้าจอกำหนดงาน: หน้าจอ Add Group

จุดประสงค์: ใช้สำหรับเพิ่มรายชื่อกลุ่ม หรือแก้ไขกลุ่ม

วิธีการใช้งาน: เมื่อกดปุ่ม Add จากหน้าจอ List Group จะเป็นการเพิ่ม Group ใหม่ให้กรอกข้อมูลชื่อกลุ่ม คำอธิบาย จากนั้นกดปุ่ม o.k.

ส่วนของ Application



รูปที่ 5.13 แผนภาพแสดงหน้าจอการ Login

หน้าจอการทำงาน: หน้าจอ Login

จุดประสงค์: ใช้สำหรับตรวจสอบสิทธิ์ของผู้เข้ามาใช้งาน

วิธีการใช้งาน: ป้อนข้อมูล ชื่อและรหัสผ่าน จากนั้นกดปุ่ม o.k. ระบบจะทำการตรวจสอบสิทธิ์ในการใช้งาน

Welcome JDV ...Enjoy your work

Ticket No: Subject:

Requester: Contact:

Job Description

Start End

Duedate Status

รูปที่ 5.14 แผนภาพแสดงหน้าจอสำหรับลงทะเบียนการทำงาน

หน้าจอกำหนดงาน: หน้าจอ บันทึกเวลาการทำงาน

จุดประสงค์: ใช้สำหรับบันทึกข้อมูลเวลาการทำงาน พร้อมทั้งให้บันทึกรายละเอียดงานที่ได้ทำไป

วิธีการใช้งาน: เลือกรหัสของงานที่ต้องการจะทำ ระบบจะทำการดึงข้อมูลรายละเอียดของงานนั้นขึ้นมา เมื่อต้องการเริ่มทำงานให้กดปุ่ม Start และถ้าสิ้นสุดการทำงานแล้วให้เขียนงานที่ได้ทำไปในช่อง Job Description จากนั้นกดปุ่ม Finish หรือถ้าต้องการยกเลิก ให้กดปุ่ม Cancel

ส่วนในการ Sync

ซึ่งสามารถใช้ได้กรณีที่ต้องออกไปข้างนอกและระบบไม่สามารถติดต่อ Database ได้ จะทำการดึงข้อมูลของงานลงมาไว้บนเครื่องของผู้ใช้ ผู้ใช้ก็ทำการ Login ระบบเข้ามาใหม่และเริ่มลงทะเบียนการทำงาน เหมือนที่ได้อธิบายไว้ตอนต้น จากนั้นเมื่อกลับมาที่สำนักงานก็ทำการ Sync ข้อมูลในเครื่องที่ได้ทำการลงทะเบียนไว้ ขึ้นบนฐานข้อมูล

บทที่ 6

บทสรุปและข้อเสนอแนะ

โปรแกรมนี้จัดทำขึ้นเพื่อใช้เป็นเครื่องมือในการควบคุมดูแลงาน ติดตามงานและช่วยในการวัดประสิทธิภาพของคนในทีม และช่วยผู้ทำงานในการจัดลำดับงานและความสำคัญของงานว่าควรทำงานใดก่อนหลัง ซึ่งจะช่วยให้สามารถทำงานได้เสร็จทันตามเป้าหมาย อีกทั้งคนอื่น ๆ ในทีมสามารถทราบว่าจะงานใดได้ทำสำเร็จไปแล้วบ้างเพื่อเป็นข้อมูลในการทำงานร่วมกันเป็นทีมในกรณีที่พนักงานคนหนึ่งลาหยุดคนในทีมคนอื่น ๆ ก็สามารถทราบสถานะของงานว่าได้ทำส่วนไหนไปบ้างอย่างไร และเป็นข้อมูลเพื่อนำไปเรียกเก็บเงินกับลูกค้า ช่วยให้สามารถคำนวณจำนวนเงินได้อย่างถูกต้องและน่าเชื่อถือ และผู้บริหารสามารถทราบประสิทธิภาพของคนในทีม เพื่อนำไปใช้ในการวางแผนและพัฒนาประสิทธิภาพให้ดียิ่งขึ้นไป

ข้อเสนอแนะในส่วน โปรแกรมยังมีส่วนที่ยังไม่สามารถตรวจสอบการทำงานได้อย่างแท้จริงว่าพนักงานแต่ละคนที่มีการบันทึกเวลาการทำงานนั้นได้มีการทำงานจริงหรือไม่ ในส่วนนี้โปรแกรมยังไม่สามารถตรวจสอบได้

บรรณานุกรม

ศิริลักษณ์ วัฒนกิจอำนวย ระบบฐานข้อมูล. กรุงเทพฯ: มหาวิทยาลัยธรรมศาสตร์
 สานิต ชัยวิวัฒน์ตระกูล. 2545. เก่ง JSP ให้ครบสูตร. กรุงเทพฯ: วิตตี้ กรุ๊ป.

Nijssen G.M and Halpin T.A., **Conceptual schema and Relational Database Design**,
 Prentice- Hall Pub. Co., 1989.

Rao R. Bindu, **C++ and the OOP Paradigm**, McGraw – Hill Inc., 1993.

Elmasri Ramez and Navathe B. Shamkant, **Fundamental of Database Systems**, Addison –
 Wesley Pub. Co.,1989.

On-line version of book from Addison-Wesley. [Online]. Available:

java.sun.com/docs/books/tutorial/

Hands-on, how-to features and columns by Java experts; [Online]. Available:

<http://www.javaworld.com/>

A detailed tutorial for Java Servlets version 2.1/2.2 and JavaServer Pages (JSP)

; [Online]. Available: www.apl.jhu.edu/~hall/java/Servlet-Tutorial/

The JavaMail API is implemented as a Java platform ; [Online]. Available:

java.sun.com/products/javamail/

The Apache developer community ; [Online]. Available: jakarta.apache.org/tomcat/

ภาคผนวก

Apache และ Tomcat ที่ได้ลองใช้แล้วพบว่าเหมาะสมในการนำมาใช้ด้วยกัน คือ Apache 2.0.43 และ Tomcat 4.1.18

วิธีการลง Apache

ไฟล์ที่ใช้ชื่อไฟล์คือ apache_2.0.43-win32-x86-no_ssl.exe โดยการ install นั้นควรจะลง ใน path ที่ไม่มีช่องว่าง เช่น C:\ApacheGroup\Apache2

ข้อควรระวัง ในการ install บน Windows 2000 Server คือ จะมี IIS มาให้แล้ว ทำให้ไม่สามารถ start Apache ได้ จึงต้องหยุด service ของ IIS ไว้ก่อน

วิธีการลง Tomcat

ไฟล์ที่ใช้ ชื่อไฟล์คือ tomcat-4.1.18.exe โดยการ install นั้นต้องลงใน path ที่ไม่มีช่องว่าง เช่น C:\ApacheGroup\Tomcat

ข้อควรระวัง Tomcat version นี้จะมีการ comment บางอย่างไว้ใน \$Tomcat/conf/web.xml (\$Tomcat คือ home folder ที่ install Tomcat ไว้ ในที่นี้คือ C:\ApacheGroup\Tomcat) ทำให้ไม่สามารถใช้ run servlet ได้ จะ run ได้แต่ JSP หากต้องการให้ run servlet ได้ทุกไฟล์ ต้องไปเอา comment ในส่วนนี้ออก:

```
<!-- The mapping for the invoker servlet -->
<servlet-mapping>
  <servlet-name>invoker</servlet-name>
  <url-pattern>/servlet/*</url-pattern>
</servlet-mapping>
```

วิธีการสร้าง Web Application ใน Tomcat

การสร้าง Web Application ใน Tomcat ทำได้โดยแก้ไขในไฟล์ \$Tomcat/conf/server.xml ในส่วน <Context> tag ซึ่งจะอยู่ภายใต้ <Host> tag โดยแต่ละ <Context> จะแสดงถึงแต่ละ Web Application ที่ Tomcat นั้นมีอยู่ วิธีการเขียน <Context> แบบง่ายๆ เป็นดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<!-- WinCall Context -->
```

```
<Context path="/WinCall" docBase="D:/Project/WinCall/webApp" debug="0"
  reloadable="true" crossContext="true"/>
```

โดย path จะแสดงถึง path ใน URL ในที่นี้ถ้า Tomcat ถูก host อยู่บน server <http://www.wtec.co.th> แล้ว path ที่เป็น URL ของ Web Application นี้ก็คือ <http://www.wtec.co.th/WinCall> docBase คือ folder นอกสุดของ Web Application และจะเป็นที่เก็บไฟล์ JSP หรือ HTML ในที่นี้คือ D:/Project/WinCall/webApp ดังนั้นหากมีไฟล์ Test1.jsp อยู่ใน folder D:/Project/WinCall/webApp จะ access ได้ผ่าน URL <http://www.wtec.co.th/WinCall/Test1.jsp> และหากมีไฟล์ Test2.jsp อยู่ใน folder D:/Project/WinCall/webApp/backOffice จะสามารถ access ได้ผ่าน URL <http://www.wtec.co.th/WinCall/backOffice/Test1.jsp>

reloadable = true คือ ให้ reload Servlet ใหม่ทุกครั้งที่มี Servlet ใหม่โดยไม่ต้อง restart Tomcat แต่หากมีการใช้ Oracle OCI คู่กับ JDBC ในการ connect DataBase จะต้อง restart Tomcat ทุกครั้งที่มีการแก้ไขหรือนำ .class ไฟล์ใหม่ไปวาง ดังเช่นใน WebStore

ในการสร้าง Web Application แต่ละครั้งจะต้อง restart Tomcat ด้วย และหากมี Servlet หรือ Class ของ Java จะต้องไว้ที่ \$webApp/WEB-INF/classes (\$webApp คือ folder ที่มี Web Application อยู่) ซึ่งในที่นี้คือ D:/Project/WinCall/webApp/WEB-INF/classes และหากมี library ที่ต้องการใช้ เช่น classes12.jar เพื่อใช้ connect กับ Oracle ก็ให้นำไปไว้ที่ \$webApp/WEB-INF/lib

วิธีการเชื่อมระหว่าง Apache และ Tomcat

นำไฟล์ mod_jk2-2.0.43.dll ที่อยู่ใน \\wsnt00\software\Application & Web Server ไปวางไว้ที่ \$Apache/modules (\$Apache คือ home folder ของ Apache โดยในที่นี้คือ C:\ApacheGroup\Apache2) แล้ว rename ให้เป็น mod_jk2.dll โดยไฟล์นี้จะใช้ได้กับ Apache 2.0.43 เท่านั้น ไม่สามารถใช้กับ version อื่นได้

จากนั้นเข้าไปแก้ไขไฟล์ httpd.conf ที่อยู่ใน \$Apache/conf โดยเพิ่มบรรทัดไว้ข้างใต้สุดของบรรทัด ที่มี LoadModule ตามนี้

```
LoadModule jk2_module modules/mod_jk2.dll
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากนั้นให้สร้างไฟล์ชื่อ workers2.properties ไว้ที่ \$Apache/conf โดยมีรายละเอียดในไฟล์ตามนี้

```
# Shared memory handling. Needs to be set.
```

```
[shm]
```

```
file=C:\ApacheGroup\Apache2\logs\shm.file
```

```
size=1048576
```

```
# Example socket channel, override port and host.
```

```
[channel.socket:localhost:8009]
```

```
port=8009
```

```
host=localhost
```

```
# Define the worker
```

```
[ajp13:localhost:8009]
```

```
channel=channel.socket:localhost:8009
```

```
# Uri mapping
```

```
[uri:/examples/*]
```

```
worker=ajp13:localhost:8009
```

```
# Uri mapping
```

```
[uri:/WinCall/*]
```

```
worker=ajp13:localhost:8009
```

โดย Uri mapping คือ ส่วนที่ map ระหว่าง Apache กับ Web Application ใน Tomcat หากมีการสร้าง Web Application เพิ่มใน Tomcat และต้องการให้ Apache forward request ไปที่ Tomcat ก็จะต้องเพิ่มในส่วน Uri mapping ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการทำ Virtual Host

ทำการแก้ไขในไฟล์ \$Apache/conf/httpd.conf โดยการเพิ่มบรรทัด

```
NameVirtualHost *:80
```

เพื่อบอกว่า Virtual Host ใช้ port 80 (ใน web บอกว่าไม่ต้องใส่ก็ได้ แต่ลองแล้วไม่ work เพราะฉะนั้นจึงต้องใส่ด้วย)

การเพิ่ม Virtual Host สามารถทำได้ดังตัวอย่าง

```
<VirtualHost *:80>
    ServerAdmin wtec@winstore.net
    DocumentRoot D:/WinTech
    ServerName www.wtec.co.th
    ServerAlias wtec.co.th
</VirtualHost>
```

ServerAdmin ไม่จำเป็นต้องมีก็ได้

DocumentRoot เป็นการบอกว่าไฟล์ถูกเก็บไว้ที่ไหน

ServerName คือชื่อของ server ใน URL

ServerAlias สามารถใช้ URL อื่นเพื่อ access web ในส่วนเดียวกัน

ดังนั้นในตัวอย่างจึงสามารถเข้า web ของ WTEC ได้จากทั้ง <http://www.wtec.co.th> และ <http://wtec.co.th> โดยเมื่อเข้า URL เหล่านี้แล้ว จะไปมองหาไฟล์ index.htm หรือ index.html ที่ D:/WinTech ซึ่งหาก WTEC ใช้ในส่วน Tomcat ด้วยโดยอยู่ภายใต้ /WinTech จะต้อง redirect ไปที่ <http://www.wtec.co.th/WinTech/index.jsp> เป็นต้น

Reference: 1. <http://jakarta.apache.org/tomcat/tomcat-4.1-doc/jk2/index.html>

2. <http://httpd.apache.org/docs-2.0/vhosts/name-based.html>

ประวัติผู้เขียน

ชื่อ: นางสาว จิตรลดา วิจิตร โกสุม
วันเกิด: 25 พฤศจิกายน 2521
สถานที่เกิด: กรุงเทพมหานครฯ
ประวัติการศึกษา: มหาวิทยาลัยหอการค้าไทย
สถานที่ทำงาน: บริษัท WinStore Technologies Co., Ltd.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้