

ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล.

การพัฒนาระบบเว็บเบสอีเมลล์

The Development of Web-Based E-mail System



วัน เดือน ปี.....	๖ ก.พ. 255๐
เลขทะเบียน.....	๐๒๑๕๒
เลขเรียกหนังสือ.....	๑๗ ๗๑๕ ก ๒๕๔๖
"ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล."	

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
ภาคเรียนที่ 2 ปีการศึกษา 2546
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ	การพัฒนาระบบเว็บเบสอีเมล
นักศึกษา	นางสาว ศุภรัตน์ รัตนจิตบรรจง
อาจารย์ที่ปรึกษา	ผศ. ดร. จันทร์บุรณีย์ สถิตวิริยวงศ์
ระดับการศึกษา	วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2546

บทคัดย่อ

ในปัจจุบันเว็บเมลเป็นที่นิยมใช้กันอย่างแพร่หลาย เนื่องจากความสะดวกสบายในการใช้งานนั่นเอง โดยที่เว็บเมลเป็นระบบเครือข่ายอินเทอร์เน็ตที่ใช้สำหรับส่ง, รับและจัดการจดหมายอิเล็กทรอนิกส์หรือที่เรียกกันว่า อีเมล ซึ่งมีข้อดีใหญ่ๆ 2 ประการคือ ความง่ายในการใช้งานและความสามารถที่จะใช้งานได้ทุกที่ทุกเวลา โดยผ่านทางเว็บเบราว์เซอร์ที่ติดต่อกับอินเทอร์เน็ต ดังนั้น ในหลายๆองค์กรและธุรกิจต่างๆ จึงได้นำระบบเว็บเมลนี้ไปใช้ในการดำเนินงาน เป็นผลให้การทำงานเกิดประสิทธิภาพมากขึ้น

โครงการพัฒนาระบบงานนี้เป็นการพัฒนาระบบเว็บเบสอีเมลขึ้นมา โดยมีวัตถุประสงค์เพื่อที่จะนำระบบที่พัฒนานี้ไปประยุกต์ใช้ในการรับ-ส่งอีเมล บนระบบอินทราเน็ตภายในองค์กร ซึ่งจะช่วยให้การทำงานร่วมกันและการบริหารทรัพยากรต่างๆภายในองค์กรมีประสิทธิภาพยิ่งขึ้น รวมทั้งระบบเว็บเมลนี้ ยังสามารถใช้ในการรับ-ส่งอีเมลบนอินเทอร์เน็ตได้อีกด้วย

Title	The Development of Web-Based E-mail System
Student	Miss Suparutt Rattanajitbunjong
Advisor	Asst. Prof. Dr. Chanboon Sathitwiriwong
Level of Study	Master of Science in Information Technology
Major	Information Science
Academic Year	2003

ABSTRACT

At present Webmail is widely used for communication because of the friendly user of this application. Webmail is an Internet network system used for sending, receiving and managing electronic mail, or e-mail. It has two key advantages that are easiness and convenience for use at any place and time. So many corporates and business units, have used this tool to better their work performance.

This project is to develop Web-Based E-mail System and bring the system for sending and receiving e-mail on the Intranet system in the organization. This will bring together all works of business and improve the company's management. In addition, the Web-Based E-mail System is used on the Internet network.

กิตติกรรมประกาศ

โครงการพัฒนาระบบงานนี้ สำเร็จได้ด้วยความช่วยเหลือและการสนับสนุนจากหลายฝ่าย ดังต่อไปนี้

1. ผศ. ดร. จันทร์บุรณม์ สติตวิริยวงศ์ ซึ่งเป็นอาจารย์ที่ปรึกษาโครงการและอาจารย์ที่ปรึกษาสัมมนา ที่กรุณาให้คำปรึกษาในข้อปัญหาต่างๆที่เกิดขึ้นระหว่างพัฒนาระบบงาน
2. คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เอื้อเฟื้อหนังสือในการค้นคว้า เพื่อพัฒนาระบบงาน
3. คุณกรกฎ ที่กรุณาให้ข้อมูลเกี่ยวกับระบบเว็บเบสอีเมลล์และคำแนะนำในการพัฒนาระบบงาน
4. คุณธงชัย อำนวยจิตรธรรม และเพื่อนๆทุกท่านที่คอยให้คำแนะนำ ให้ความช่วยเหลือ และเป็นกำลังใจมาโดยตลอด

จึงใคร่ขอขอบคุณบุคคลดังกล่าวข้างต้นมา ณ โอกาสนี้

นางสาวสุภรัตน์ รัตนจิตบรรจง

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญรูป.....	VII
บทที่	
1. บทนำ.....	1
1.1 ความเป็นมาและความสำคัญ.....	1
1.2 วัตถุประสงค์ของการพัฒนาระบบ.....	2
1.3 ขอบเขตของการพัฒนาระบบ.....	2
1.4 ขั้นตอนการดำเนินงาน.....	2
1.5 ผลที่คาดว่าจะได้รับ.....	3
1.6 เครื่องมือที่ใช้ในการพัฒนาระบบ.....	3
2. ทฤษฎีที่เกี่ยวข้อง.....	5
2.1 วงจรการพัฒนาระบบ.....	5
2.2 HTML.....	6
2.3 PHP.....	7
2.4 MySQL.....	9
2.5 E-mail และ Webmail.....	10
2.6 Web Server.....	13
2.7 POP3 และ IMAP4.....	14
2.8 SMTP.....	17
2.9 MIME.....	18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
3. ระบบเว็บเบสอีเมลในปัจจุบัน.....	20
3.1 องค์ประกอบของการทำงานของเว็บเบสอีเมล.....	20
3.2 องค์ประกอบภายในอีเมล.....	21
3.3 ขั้นตอนการทำงานของระบบเว็บเบสอีเมล.....	22
3.4 ปัญหาที่พบในระบบปัจจุบัน.....	23
4. การวิเคราะห์และออกแบบระบบ.....	25
4.1 แนวความคิดในการออกแบบระบบ.....	25
4.2 ขั้นตอนการทำงานของระบบ.....	26
4.3 รายละเอียดกระบวนการทำงาน.....	29
4.4 การออกแบบฐานข้อมูล.....	35
5. การพัฒนาระบบ.....	41
5.1 การเข้าใช้งานระบบ.....	41
5.2 หน้าแม่ล้น้อกซ์.....	43
5.3 การจัดการจดหมายแต่ละฉบับ.....	53
6. บทสรุป.....	56
6.1 สรุปผลโครงการศึกษากรณีพิเศษ.....	56
6.2 ปัญหาและข้อจำกัดที่พบ.....	56
6.3 ประโยชน์ที่ได้จากการศึกษาและพัฒนาระบบ.....	57
6.4 ข้อเสนอแนะ.....	57
บรรณานุกรม.....	58
ประวัติผู้เขียน.....	60

สารบัญตาราง

หน้า

ตารางที่

2.1	คำสั่งเบื้องต้นสำหรับ MySQL.....	9
2.2	โปรแกรมเว็บเซิร์ฟเวอร์ที่ทำงานบนระบบจัดการต่างๆ.....	14
3.1	รายการหลักภายในอีเมล.....	21
4.1	รายละเอียดตาราง TBLUSER.....	37
4.2	รายละเอียดตาราง TBLMESSAGE.....	37
4.3	รายละเอียดตาราง TBLATTACHMENT.....	38
4.4	รายละเอียดตาราง TBLADDRESS.....	39
4.5	รายละเอียดตาราง TBLBLOCKEMAIL.....	39

สารบัญรูป

หน้า

รูปที่

2.1 หลักการทำงานของ PHP.....	8
2.2 หลักการทำงานของอีเมล์.....	11
2.3 ขั้นตอนและสถานะการทำงานของโปรโตคอล IMAP4.....	17
2.4 ไตอะแกรมการทำงานรับส่งเมล์.....	19
3.1 วงจรปกติของการส่งเมล์โดยต้องออนไลน์.....	20
3.2 ส่งเมล์และรับเมล์โดยไม่ต้องออนไลน์.....	21
3.3 การทำงานโดยรวมของเว็บเบสอีเมล์.....	22
3.4 การล็อกอินเพื่อใช้บริการ.....	23
3.5 การจัดการเกี่ยวกับเมล์บ็อกซ์.....	23
3.6 การจัดการกับตัวข้อความ.....	23
4.1 ขั้นตอนการทำงานของระบบ.....	28
4.2 Context Diagram ระบบเว็บเบสอีเมล์.....	30
4.3 Level 1 Data Flow Diagram ระบบเว็บเบสอีเมล์.....	31
4.4 Level 2 Data Flow Diagram ใน Process ที่ 2.....	32
4.5 Level 3 Data Flow Diagram ใน Process ที่ 2.....	33
4.6 Level 2 Data Flow Diagram ใน Process ที่ 3.....	35
4.7 Entity Relationship Diagram.....	36
5.1 หน้าจอในการเข้าใช้ระบบ.....	40
5.2 หน้าจอแสดงข้อผิดพลาดในการเข้าใช้ระบบ.....	41
5.3 หน้าจอแสดงหน้าหลักของระบบ.....	42
5.4 หน้าจอแสดงข้อความในจดหมาย.....	43
5.5 หน้าจอแสดงการสร้างจดหมาย.....	43

สารบัญรูป (ต่อ)

รูปที่	หน้า
5.6 หน้าจอแสดงผลการส่งจดหมาย.....	44
5.7 หน้าจอแสดงการลบจดหมายในหน้าเมลบ็อกซ์.....	45
5.8 หน้าจอแสดงผลการลบจดหมายในหน้าเมลบ็อกซ์.....	45
5.9 หน้าจอแสดงจดหมายในตู้ Sent Box.....	46
5.10 หน้าจอแสดงจดหมายในถังขยะ.....	46
5.11 หน้าจอแสดงผลการลบจดหมายทั้งหมดในถังขยะ.....	47
5.12 หน้าจอการสร้างและแก้ไขข้อความลงท้ายจดหมาย.....	47
5.13 หน้าจอแสดงรายการอีเมลหรือโดเมนที่ถูกบล็อก.....	48
5.14 หน้าจอการเพิ่มรายการอีเมลหรือโดเมนที่ต้องการบล็อก.....	49
5.15 หน้าจอแสดงสมุดบันทึกอีเมลแอดเดรส.....	49
5.16 หน้าจอการเพิ่มรายการในสมุดบันทึกอีเมลแอดเดรส.....	50
5.17 หน้าจอการค้นหาคำ.....	50
5.18 หน้าจอแสดงผลการค้นหา.....	51
5.19 หน้าจอการเปลี่ยนรหัสผ่าน.....	51
5.20 หน้าจอรายงานการออกจากระบบ.....	52
5.21 หน้าจอแสดงการตอบกลับจดหมาย.....	53
5.22 หน้าจอแสดงการส่งต่อจดหมาย.....	53
5.23 หน้าจอแสดงผลการลบจดหมาย.....	54

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญ

1.1.1 ความเป็นมา

ปัจจุบันเครือข่ายของอินเทอร์เน็ต (Internet) เป็นเครือข่ายที่เชื่อมโยงคอมพิวเตอร์และเครือข่ายคอมพิวเตอร์ย่อยทั่วทุกมุมโลกเข้าด้วยกัน เพื่อประโยชน์ในการติดต่อสื่อสารหรือแลกเปลี่ยนข้อมูลกันระหว่างคอมพิวเตอร์ ซึ่งคนทั่วไปเมื่อได้ยินคำว่า “อินเทอร์เน็ต” ก็มักจะนึกถึงเว็บและอีเมลล์กันเป็นส่วนมาก โดยอีเมลล์ (E-Mail : Electronic Mail) หรือที่เรียกว่าจดหมายอิเล็กทรอนิกส์นั้น เป็นรูปแบบการให้บริการที่เปรียบเสมือนการใช้อินเทอร์เน็ตแทนไปรษณีย์ในการใช้รับ-ส่งข้อมูลหรือเมสเสจ (Message) ที่เป็นข้อความไปยังผู้อื่น รวมทั้งการส่งไฟล์อื่นๆ ไปพร้อมกับเมสเสจนี้อีกด้วย ซึ่งการใช้บริการอีเมลล์ในทุกวันนี้มักจะนิยมใช้เว็บเมล (Webmail) ซึ่งเป็นโปรแกรมรับส่งเมลล์บนอินเทอร์เน็ตที่ไม่ว่าจะอยู่ที่ใดขอเพียงมีคอมพิวเตอร์ที่ต่อใช้งานบนอินเทอร์เน็ตก็สามารถใช้งานได้ทันที ในการรับส่งอีเมลล์ผ่านทางเว็บเบราว์เซอร์ (Web browser) กันอย่างแพร่หลาย นอกจากการใช้บริการอีเมลล์ผ่านทางเครือข่ายอินเทอร์เน็ตแล้ว ในปัจจุบันหลายๆองค์กรและธุรกิจต่างๆ ก็นิยมนำบริการรับ-ส่งอีเมลล์นี้มาประยุกต์ใช้ในเครือข่ายภายในองค์กรหรือที่เรียกว่าอินทราเน็ต (Intranet) อีกด้วย

โดยปกติการติดต่อสื่อสารกันระหว่างพนักงานในองค์กรนั้น จะเป็นการพูดคุยกันระหว่างบุคคลหรือกลุ่ม ซึ่งจะทำให้เสียเวลาในการทำงาน และอาจก่อให้เกิดปัญหาอื่นๆขึ้นมาได้ ดังนั้น จึงได้มีแนวคิดที่จะนำระบบคอมพิวเตอร์เข้ามาใช้ให้เกิดประโยชน์ โดยการใช้อีเมลล์ในการติดต่อสื่อสารกัน ซึ่งจะเป็นการเสริมการทำงานให้ประหยัดเวลาได้มากขึ้น

1.1.2 ความสำคัญ

- สามารถส่งจดหมายอิเล็กทรอนิกส์เมื่อไรก็ได้ตามที่ต้องการ ไม่ว่าจะเป็กลางคืนหรือกลางวัน
- จดหมายจะถึงมือผู้รับภายในเวลาอันรวดเร็ว ไม่ว่าจะผู้รับจดหมายนั้นจะอยู่ใกล้หรือไกล
- ผู้รับจดหมายก็สามารถรับและเปิดอ่านจดหมายได้เมื่อไรก็ได้ตามที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สามารถส่งจดหมายไปยังผู้รับคนเดียว หลายคน หรือจำนวนมากได้ ซอฟต์แวร์ของอีเมลล์ส่วนใหญ่จะมีวิธีช่วยให้เก็บรายชื่อพร้อมทั้งอีเมลล์แอดเดรสของผู้ที่เราต้องการส่งจดหมายไปหา และช่วยจัดเป็นกลุ่มด้วย

- สามารถเก็บจดหมายที่ได้รับบางฉบับไว้ได้ เช่น ไว้เตือนความจำว่ามีงานอะไรต้องทำ หรือ ได้ตกลงเรื่องอะไรไว้กับใครบ้าง

โครงการพัฒนาระบบงานนี้ จะได้ทำการศึกษาระบบเว็บเบสอีเมลล์ สำหรับใช้เป็นแนวทางในการพัฒนา เพื่อที่จะนำระบบที่ได้นี้ไปประยุกต์ใช้ในการรับ-ส่งอีเมลล์บนระบบอินทราเน็ตภายในองค์กร รวมทั้งการใช้งานผ่านอินเทอร์เน็ตอีกด้วย

1.2 วัตถุประสงค์ของการพัฒนาระบบ

1. เพื่อที่จะนำรูปแบบการทำงานของระบบเว็บเมลล์ (Webmail) มาเป็นเครื่องมือช่วยในการบริหารจัดการงานภายในองค์กรให้มีความสะดวกและตรงกับความต้องการขององค์กรนั้นๆ
2. เพื่อลดขั้นตอนและระยะเวลาในการทำงาน รวมทั้งการบริหารทรัพยากรต่างๆให้เป็นไปอย่างมีประสิทธิภาพ
3. เพื่อช่วยให้การทำงานร่วมกันภายในองค์กรเกิดประสิทธิภาพยิ่งขึ้น
4. เพื่อช่วยให้องค์กรหรือธุรกิจต่างๆ สามารถติดต่อกับบุคคลหรือองค์กรภายนอกได้

1.3 ขอบเขตของการพัฒนาระบบ

- ออกแบบโครงสร้างของระบบเว็บเบสอีเมลล์
- ระบบสามารถใช้ในการรับและส่งอีเมลล์ภายในองค์กรได้
- สามารถกำหนดรายละเอียด รวมถึงฟังก์ชันการทำงานในระบบได้ตามความต้องการขององค์กร
- ระบบสามารถจัดการอีเมลล์ในเมลล์บ็อกซ์ได้อย่างมีประสิทธิภาพ
- ระบบสามารถใช้ในการรับและส่งอีเมลล์ภายนอกองค์กรได้

1.4 ขั้นตอนการดำเนินงาน

1. ศึกษาระบบการทำงานของเว็บเบสอีเมลล์ เพื่อใช้เป็นความรู้พื้นฐานในการพัฒนาระบบ
2. วิเคราะห์ถึงข้อดีและข้อเสีย รวมทั้งปัญหาต่างๆที่พบในระบบเว็บเบสอีเมลล์ที่มีอยู่ในปัจจุบัน เพื่อเป็นแนวทางในการพัฒนาระบบ

3. นำข้อมูลที่ได้ทั้งหมด มาใช้ในการวิเคราะห์และออกแบบระบบ รวมทั้งการออกแบบฐานข้อมูล

4. ศึกษาเทคโนโลยีต่างๆที่เกี่ยวข้องในการพัฒนาระบบที่ทำงานบนเครือข่ายอินเทอร์เน็ตเพื่อที่จะเลือกเทคโนโลยีที่เหมาะสมในการพัฒนาระบบ

5. ทำการเขียนโปรแกรม (Coding) ให้ทำงานตามที่ได้วิเคราะห์และออกแบบไว้

6. ทดสอบระบบทั้งหมด และทำการแก้ไข ปรับปรุงข้อผิดพลาดของระบบ เพื่อให้ระบบทำงานได้อย่างถูกต้องและมีประสิทธิภาพ

7. สรุปผลการพัฒนาระบบ รวมทั้งระบุถึงปัญหาและข้อจำกัดของระบบที่ได้ทำการพัฒนา ตลอดจนข้อเสนอแนะต่างๆ

1.5 ผลที่คาดว่าจะได้รับ

- สามารถนำระบบที่พัฒนามาประยุกต์ใช้ในองค์กรได้อย่างมีประสิทธิภาพ
- ทำให้การทำงานภายในองค์กรมีประสิทธิภาพมากขึ้น ไม่ว่าจะเป็นการลดขั้นตอนการทำงาน การลดระยะเวลา รวมทั้งการบริหารทรัพยากรต่างๆ
- ทำให้องค์กรสามารถดำเนินงานติดต่อกับองค์กร หรือธุรกิจภายนอกได้อย่างมีประสิทธิภาพ ซึ่งจะช่วยส่งเสริมให้องค์กรมีความก้าวหน้ามากขึ้น
- การพัฒนาระบบงานคอมพิวเตอร์ด้วยเว็บเทคโนโลยีครั้งนี้ ทำให้ได้รับความรู้ความเข้าใจ เพื่อเป็นประโยชน์ต่อการพัฒนาระบบงานอื่นๆในอนาคต

1.6 เครื่องมือที่ใช้ในการพัฒนาระบบ

1. ฮาร์ดแวร์

- Computer Notebook มีลักษณะ ดังนี้
 - CPU Intel(R) Pentium(R) M processor 1.5 GHz
 - RAM 512 MB
 - Toshiba DVD-ROM SD-R 2312
 - HARDDISK 60 GB
- Printer มีลักษณะ ดังนี้
 - CANON BJC 5000

2. ซอฟต์แวร์

- Microsoft Windows XP Professional Version 2002 Service Pack 1
- Microsoft Word 2002
- Microsoft Internet Explorer Version : 6.0.2800.1106
- MySQL Database Server
- EditPlus2

3. เทคโนโลยีเว็บแอปพลิเคชัน

- HTML (Hyper Text Markup Language)
- PHP (PHP Hypertext Preprocessor)
- JavaScript
- Web Server
- Agro Mail Server



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

การพัฒนากระบบเว็บเบสอีเมล ได้นำทฤษฎีต่างๆ มาเป็นแนวทางในการพัฒนาระบบหลายทฤษฎี ดังต่อไปนี้

2.1 วงจรการพัฒนากระบบ

วงจรการพัฒนากระบบ (SDLC : System Development Life Cycle) เป็นวงจรที่แสดงกิจกรรมต่างๆ ในแต่ละขั้นตอน ตั้งแต่เริ่มจนกระทั่งสำเร็จ วงจรการพัฒนากระบบนี้ จะทำให้เข้าใจถึงกิจกรรมพื้นฐานและรายละเอียดต่างๆในการพัฒนาระบบ โดยมีอยู่ 7 ขั้นตอนดังนี้ (โอภาส เอี่ยมสิริวงศ์. 2545)

1. กำหนดปัญหา (Problem Definition) เป็นขั้นตอนของการกำหนดขอบเขตของปัญหา สาเหตุของการจากการดำเนินงานในปัจจุบัน ความเป็นไปได้กับการสร้างระบบใหม่ การกำหนดความต้องการ (Requirements) ระหว่างนักวิเคราะห์ระบบกับผู้ใช้งาน โดยข้อมูลเหล่านี้ได้จากการสัมภาษณ์ การรวบรวมข้อมูลจากการดำเนินงานต่างๆ เพื่อทำการสรุปข้อกำหนด (Requirement Specification) ที่ชัดเจน ในขั้นตอนนี้หากเป็นโครงการที่มีขนาดใหญ่ อาจเรียกขั้นตอนนี้ว่า ขั้นตอนการศึกษาความเป็นไปได้ (Feasibility Study)

2. วิเคราะห์ (Analysis) เป็นขั้นตอนของการวิเคราะห์การดำเนินงานของระบบปัจจุบัน โดยการนำ Requirement Specification ที่ได้มาจากขั้นตอนแรกมาวิเคราะห์ในรายละเอียด เพื่อทำการพัฒนาเป็นแบบจำลองเชิงตรรกะ (Logical Model) ซึ่งประกอบด้วย แผนภาพกระแสข้อมูล (Data Flow Diagram), คำอธิบายการประมวลผลข้อมูล (Process Description) และแบบจำลองข้อมูล (Data Model) ในรูปแบบของ ER-Diagram ทำให้ทราบถึงรายละเอียดขั้นตอนการดำเนินงานในระบบว่า ประกอบด้วยอะไรบ้าง, มีความเกี่ยวข้องหรือมีความสัมพันธ์กับสิ่งใด

3. ออกแบบ (Design) เป็นขั้นตอนของการนำผลลัพธ์ที่ได้จากการวิเคราะห์เชิงตรรกะ มาพัฒนาเป็นแบบจำลองเชิงกายภาพให้สอดคล้องกัน โดยการออกแบบจะเริ่มจากส่วนของอุปกรณ์และเทคโนโลยีต่างๆ และโปรแกรมคอมพิวเตอร์ที่นำมาพัฒนา การออกแบบจำลองข้อมูล, การออกแบบรายงานและการออกแบบจอภาพในการติดต่อกับผู้ใช้งาน, การจัดทำพจนานุกรมข้อมูล (Data Dictionary) ขั้นตอนของการวิเคราะห์และออกแบบจะมุ่งเน้นถึงสิ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต่อไปนี้ ซึ่งก็คือ การวิเคราะห์มุ่งเน้นการแก้ปัญหาอะไร (What) ส่วนการออกแบบจะมุ่งเน้นการแก้ปัญหาอย่างไร (How)

4. พัฒนา (Development) เป็นขั้นตอนของการพัฒนาโปรแกรม ด้วยการสร้างชุดคำสั่งหรือเขียนโปรแกรม เพื่อการสร้างระบบงาน โดยโปรแกรมที่ใช้ในการพัฒนาจะต้องพิจารณาถึงความเหมาะสมกับเทคโนโลยีที่ใช้งานอยู่ ซึ่งในปัจจุบันภาษาระดับสูงได้มีการพัฒนาในรูปแบบของ 4GL ซึ่งช่วยอำนวยความสะดวกต่อการพัฒนา รวมทั้งการมี CASE (Computer Aided Software Engineering) ต่างๆมากมายให้เลือกใช้ตามความเหมาะสม

5. ทดสอบ (Testing) เป็นขั้นตอนของการทดสอบระบบก่อนที่จะนำไปปฏิบัติใช้งานจริง ทีมงานจะทำการทดสอบข้อมูลเบื้องต้นก่อน ด้วยการสร้างข้อมูลจำลองเพื่อตรวจสอบการทำงานของระบบ หากมีข้อผิดพลาดเกิดขึ้น ก็จะย้อนกลับไปในขั้นตอนของการพัฒนาโปรแกรมใหม่ โดยการทดสอบระบบนี้จะมีการตรวจสอบอยู่ 2 ส่วนด้วยกันคือ การตรวจสอบไวยากรณ์ (Syntax) และการตรวจสอบวัตถุประสงค์งานว่าตรงกับความต้องการหรือไม่

6. ติดตั้ง (Implementation) เป็นขั้นตอนต่อมา หลังจากที่ได้ทำการทดสอบ จนมีความมั่นใจแล้วว่าระบบสามารถทำงานได้จริง และตรงกับความต้องการของผู้ใช้ระบบ จากนั้นจึงดำเนินการติดตั้งระบบ เพื่อใช้งานจริงต่อไป

7. บำรุงรักษา (Maintenance) เป็นขั้นตอนของการปรับปรุงแก้ไขระบบ หลังจากที่ได้มีการติดตั้งและใช้งานแล้ว ในขั้นตอนนี้ อาจเกิดจากจุดบกพร่องของโปรแกรม (Bug) ซึ่งโปรแกรมเมอร์จะต้องรีบแก้ไขให้ถูกต้อง หรือเกิดจากความต้องการของผู้ใช้งานที่ต้องการเพิ่มโมดูลในการทำงานอื่นๆ ซึ่งทั้งนี้ก็จะเกี่ยวข้องกับ Requirement Specification ที่เคยตกลงกันก่อนหน้านั้นด้วย ดังนั้นในส่วนงานนี้จะคิดค่าใช้จ่ายเพิ่มหรืออย่างไร เป็นเรื่องของรายละเอียดของผู้พัฒนาหรือนักวิเคราะห์ระบบจะต้องดำเนินการกับผู้ว่าจ้างต่อไป

2.2 HTML

HTML มาจากคำว่า HyperText Markup Language (ไพศาล โมลิสสกุลมงคล. 2544 : 27) ซึ่งเป็นรูปแบบของภาษาที่ใช้ในการเขียนโปรแกรมในเว็บเพจ เพื่อแสดงผลบนเว็บเบราว์เซอร์ ลักษณะของเอกสาร HTML จะเป็นแท็กซ์ไฟล์ธรรมดาที่ต้องอาศัยการแปลความจากเว็บเบราว์เซอร์ ในสมัยก่อนจุดประสงค์การใช้ HTML ก็เพื่อแสดงผลที่เป็นข้อความเป็นส่วนใหญ่ แต่ปัจจุบัน HTML ได้พัฒนามาจนถึงเวอร์ชัน 4 แล้ว ในเวอร์ชันนี้มีความสามารถเพิ่มเติมมากมาย ที่รวมทั้งความสามารถในด้านมัลติมีเดีย (Multimedia)

คำสั่งของภาษา HTML เรียกว่า “แท็ก” (Tag) ซึ่งแท็กนี้โดยทั่วไปจะอยู่ในรูปแบบ `<...>.....</...>` ซึ่งเว็บเบราว์เซอร์จะแปลงแท็กนี้แล้วแสดงผลให้เห็น โดยทั่วไปการสร้างเว็บเพจด้วยภาษา HTML โดยใช้เอดิเตอร์ต่างๆ เช่น NotePad ของ Windows เป็นเรื่องที่ดีและผู้ใช้เขียนโปรแกรมเข้าใจโครงสร้างโดยรวมทั้งหมด แต่โปรแกรมเมอร์จะต้องเข้าใจรูปแบบคำสั่งหรือแท็กของ HTML ทั้งหมด ซึ่งเป็นการยากและเสียเวลามาก ในปัจจุบันจึงได้มีเครื่องมือที่ช่วยสนับสนุนการเขียนโปรแกรมบนเว็บเพจในลักษณะ WYSIWYG (What You See Is What You Get) เช่น Microsoft Word หรือ Microsoft Frontpage เป็นต้น โปรแกรมสำเร็จรูปในลักษณะนี้ จะทำให้ประหยัดเวลาในการสร้างเว็บเพจ เพราะสามารถใช้งานได้เหมือนกับการสร้างเอกสารทั่วไป คือ คีย์ข้อความ แล้วเปลี่ยนรูปแบบอื่นๆตามต้องการ หลังจากนั้นเมื่อมีการจัดเก็บเอกสารแค่เพียงเลือกรูปแบบเอกสารที่ต้องการจัดเก็บให้เป็นแบบ HTML ก็จะสามารถนำไปแสดงผลบนเว็บเบราว์เซอร์ได้ทันที โดยโปรแกรมสำเร็จรูปนี้จะสร้างโค้ด HTML ให้โดยอัตโนมัติ

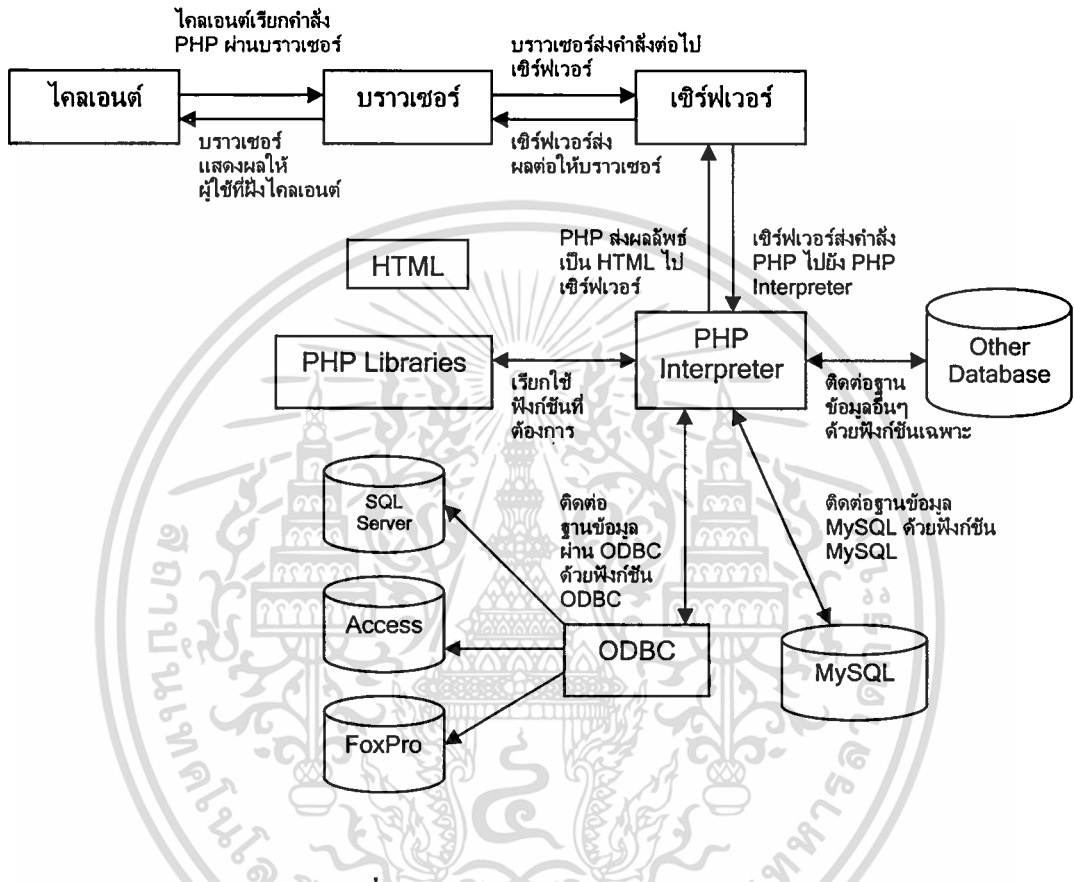
2.3 PHP

แต่เดิม PHP คือ Professional Home Page (ไพศาล โมติสสกุลมงคล. 2544 : 137) แต่ในปัจจุบัน PHP หมายถึง PHP Hypertext Preprocessor ซึ่งเป็นภาษาสคริปต์แบบหนึ่งที่เรียกว่า Server Side Script ที่ประมวลผลฝั่งเซิร์ฟเวอร์ แล้วส่งผลลัพธ์ไปฝั่งไคลเอนต์ผ่านเว็บเบราว์เซอร์ เช่นเดียวกับ ASP (Active Server Pages) ปัจจุบันได้รับความนิยมเป็นอย่างมาก ในการนำมาช่วยพัฒนาระบบงานบนเว็บที่เรียกว่า Web Development หรือ Web Programming เนื่องจากมีจุดเด่นหลายประการ รูปแบบของภาษา PHP มีเค้าโครงมาจากภาษา C และ Perl ที่นำมาปรับปรุงทำให้มีประสิทธิภาพสูง และทำงานได้เร็วขึ้น

2.3.1 หลักการทำงานของ PHP

จะเริ่มต้นที่ผู้ใช้ส่งความต้องการผ่านเว็บเบราว์เซอร์ทาง HTTP (HTTP Request) ซึ่งอาจจะเป็นการกรอกแบบฟอร์ม หรือใส่ข้อมูลที่ต้องการ ข้อมูลเหล่านั้นจะเป็นเอกสาร PHP (เอกสารนี้จะมีส่วนขยายเป็น php หรือ php3 แล้วแต่ผู้ใช้กำหนด เช่น search.php เป็นต้น) เมื่อเอกสาร PHP เข้ามาถึงเว็บเซิร์ฟเวอร์ก็จะถูกส่งไปให้ PHP เพื่อทำหน้าที่แปลคำสั่งแล้วเอ็กคิวต์ (execute) คำสั่งนั้น หลังจากนั้น PHP จะสร้างผลลัพธ์ในรูปแบบเอกสาร HTML ส่งกลับไปให้เว็บเซิร์ฟเวอร์ เพื่อส่งต่อไปให้เบราว์เซอร์แสดงผลทางฝั่งผู้ใช้ต่อไป (HTTP Response) ซึ่งลักษณะการทำงานแบบนี้จะคล้ายกับการทำงานของ CGI (Common Gateway Interface) หรืออาจกล่าวได้

ว่า PHP ก็คือ โปรแกรม CGI ประเภทหนึ่งก็ได้ ซึ่งจะทำงานคล้ายกับ ASP นั่นเอง ลักษณะการทำงานจะเป็นดังรูปที่ 2.1



รูปที่ 2.1 หลักการทำงานของ PHP

2.3.2 รูปแบบการเขียนโค้ด PHP

รูปแบบการเขียนโค้ด PHP สามารถแทรกลงในส่วนใดของแท็ก HTML ก็ได้ โดยเขียนได้ 5 แบบ ดังนี้

แบบที่ 1 การเขียนโค้ด PHP ในลักษณะทั่วไปแบบภาษา SGML

```
<?
คำสั่งในภาษา PHP ;
?>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบที่ 2 การเขียนโค้ด PHP ในลักษณะภาษา XML

```
<?php
คำสั่งภาษา PHP ;
?>
```

แบบที่ 3 การเขียนโค้ด PHP ในลักษณะ JavaScript

```
<Script Language="php">
คำสั่งภาษา PHP ;
</Script>
```

แบบที่ 4 การเขียนโค้ด PHP ในลักษณะ ASP

```
<%
คำสั่งภาษา PHP ;
%>
```

แบบที่ 5 การเขียนโค้ด PHP ในลักษณะพิเศษ

```
<%= $ตัวแปร ;
คำสั่งภาษา PHP ;
%>
```

2.4 MySQL

MySQL เป็นโปรแกรมฐานข้อมูลในลักษณะ Database Server ซึ่งทำงานได้ทั้งระบบ Telnet บน Linux Redhat หรือ Unix System และบน Win32 (Windows 95/98/ME) เพื่อใช้กับ อินเทอร์เน็ตและอินทราเน็ต หมายความว่า เราสามารถเรียกใช้ MySQL ได้ทั่วโลกกรณีเป็น อินเทอร์เน็ต และทั่วบริเวณที่เป็นอินทราเน็ต และยังสามารถเรียกใช้บนเว็บเบราว์เซอร์ได้ในกรณี ที่ใช้ภาษาอินเทอร์เน็ตเฟส (Interface) เข้ามาใช้งานฐานข้อมูล เช่น PHP, Perl, C, C++ ฯลฯ

2.4.1 คำสั่งเบื้องต้นสำหรับ MySQL

คำสั่งเบื้องต้นสำหรับ MySQL มีดังนี้

ตารางที่ 2.1 คำสั่งเบื้องต้นสำหรับ MySQL

คำสั่ง	การใช้งาน
help	ดูระบบช่วยเหลือ
status	แสดงสถานะของ MySQL เช่น เวอร์ชัน, ฐานข้อมูลที่ใช้อยู่, ผู้ใช้ปัจจุบัน เป็นต้น
exit	ออกจาก MySQL
quit	ออกจาก MySQL
use	ใช้ฐานข้อมูล
create database	สร้างฐานข้อมูลใหม่
create table	สร้างตารางใหม่
show databases	แสดงฐานข้อมูลที่มีอยู่ใน MySQL
show tables	แสดงตารางที่มีอยู่ในฐานข้อมูลปัจจุบันที่ใช้อยู่
select	เลือกฟิลด์ (field) ที่จะแสดงผลข้อมูล
insert into	เพิ่มข้อมูลเข้าสู่ตารางที่กำหนด
delete from	ลบข้อมูลออกจากตารางตามเงื่อนไข
load data local infile	โหลดข้อมูลจากเท็กซ์ไฟล์ (text file) เข้าสู่ตาราง

2.5 E-mail และ Webmail

2.5.1 E-mail

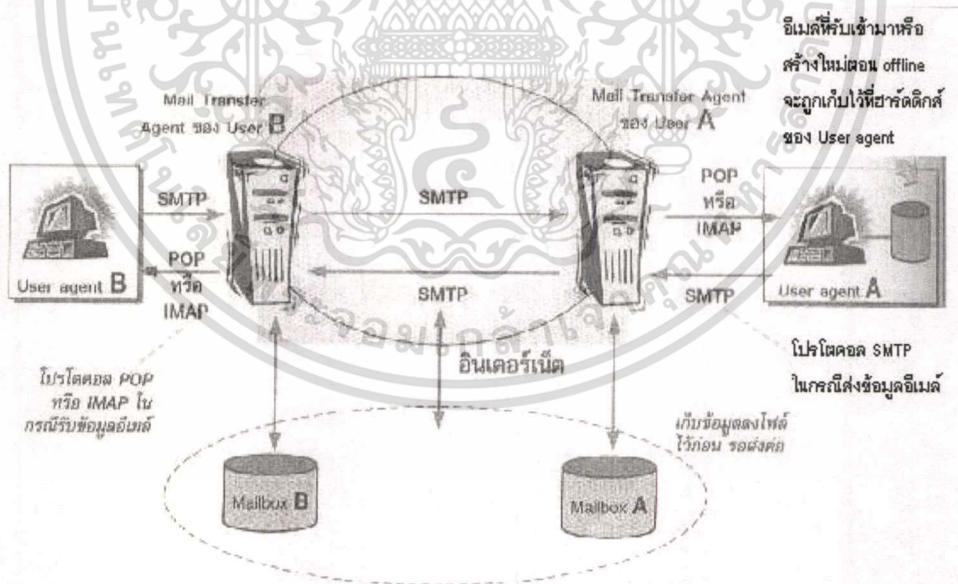
E-mail หรือจดหมายอิเล็กทรอนิกส์ได้เริ่มใช้งานกันมานานแล้ว ตั้งแต่ยุคของเครื่องเมนเฟรม (Mainframe) หรือมินิคอมพิวเตอร์ ซึ่งไอบีเอ็มได้พัฒนาระบบอีเมลที่เรียกว่า PROFS (Professional Office System) ออกมาใช้งาน นอกจากนี้ก็มีระบบ UNIX ต่อมาหลาย ๆ คำยก็ได้พัฒนาระบบอีเมลของตนขึ้นมา โดยส่วนใหญ่จะเป็นองค์ประกอบในแอปพลิเคชันที่ทำงานบนระบบเน็ตเวิร์ค เช่น Microsoft Mail ของไมโครซอฟต์ และ cc:Mail ของโลตัส เป็นต้น ซึ่งต่างก็ใช้เทคโนโลยีของตนเองและเป็นระบบปิด ดังนั้นการส่ง E-mail ไปยังผู้ใช้ที่ใช้ระบบเมลคนละค่ายกันจึงเป็นเรื่องที่ยุ่งยาก

ในยุคต่อมาที่ระบบเน็ตเวิร์คทั้ง LAN และ WAN ต่างมีมาตรฐาน และเป็นระบบเปิด (Open System) มากขึ้น ก็ได้ปรับเปลี่ยนการทำงานของอีเมลมาเป็นแบบไคลเอนต์เซิร์ฟเวอร์ที่เป็นพื้นฐานแบบที่ใช้กันในระบบ UNIX และมีการพัฒนาอีเมลเซิร์ฟเวอร์ขึ้นมาโดยเฉพาะ เช่น เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Exchange Server ของไมโครซอฟต์ หรือ Note Server ของโลตัส เป็นต้น ซึ่งผู้ใช้ติดต่อเข้าสู่ อีเมลเซิร์ฟเวอร์ได้ทั้งผ่านทางระบบ LAN หรือใช้โมเด็มเข้ามาจาก WAN ทำให้ผู้ใช้จะไม่เห็น ไฟล์ในฮาร์ดดิสก์บนเซิร์ฟเวอร์เลย ดังนั้นความปลอดภัยของระบบจึงมีมากขึ้น จนในปัจจุบัน ได้พัฒนาขึ้นมาเป็นระบบ Workflow ที่ใช้อีเมลเป็นพื้นฐาน การทำงานของอีเมลไคลเอนต์และ อีเมลเซิร์ฟเวอร์มีส่วนประกอบดังนี้ (สุวัฒน์ ปุณณชัยยะ และคณะ. 2543 : 125)

1. User Agent เป็นโปรแกรมคอมพิวเตอร์ทางด้านผู้ใช้งาน แบ่งเป็น 2 ส่วนคือ ส่วนของ ผู้ส่ง และส่วนของผู้รับ โดย User agent นี้จะติดต่อเข้าสู่เซิร์ฟเวอร์ของตนโดยผ่านระบบ LAN หรือ Dial-up ซึ่งในส่วนของผู้ส่ง User agent นี้จะเป็นส่วนที่ผู้ใช้ติดตั้งโปรแกรมไคลเอนต์ของอีเมล เพื่อเรียกใช้บริการอีเมล เช่น Outlook Express หรือ Eudora เป็นต้น

2. MTA (Mail Transfer Agent) เป็นโปรแกรมคอมพิวเตอร์ที่จะส่งอีเมลจากต้นทางไปยัง ผู้รับปลายทาง ซึ่งจะต้องส่งผ่านเครื่องจำนวนมากที่เชื่อมต่อกันในเครือข่ายโดยโปรแกรมเหล่านี้ จะช่วยกันส่งต่ออีเมลเป็นทอด ๆ จนไปถึงเครื่องที่มี Account หรือเมลบ็อกซ์ของผู้รับ และหากไม่ สามารถส่งเมลถึงผู้รับได้ (เพราะใส่ชื่อผิด) ยังทำหน้าที่ส่ง error mail กลับมาส่งผู้ส่งได้อีก ซึ่ง เครื่องที่มี MTA ทำงานอยู่มักจะมีเมลบ็อกซ์ของผู้ใช้ด้วย ซึ่งเป็น primary mailbox และเรียก เครื่องนี้ว่า Mail Server



รูปที่ 2.2 หลักการทำงานของอีเมล

2.5.2 Webmail

Webmail เป็นโปรแกรมรับ-ส่งเมลล์บนอินเทอร์เน็ตไม่ว่าจะอยู่ที่ใด ขอเพียงมีคอมพิวเตอร์ที่ต่อใช้งานบนอินเทอร์เน็ตก็สามารถใช้งานได้ทันที ที่รู้จักกันแพร่หลาย เช่น Hotmail, Yahoo Mail, Chaiyo Mail ซึ่งต่างจากโปรแกรมเมลล์ไคลเอนต์อื่นๆ เช่น Outlook Express หรือ Microsoft Outlook โปรแกรมเหล่านี้ต้องมีเครื่องเฉพาะใช้งาน ถ้าจะไปใช้เครื่องอื่นต้องทำการเซ็ทอัพ (setup) ใหม่

2.5.3 โพรโทคอลและประเภทการใช้งาน

การทำงานทั่ว ๆ ไปของอีเมลล์โดยสรุปมีเพียง 2 ประเภทคือ การส่งอีเมลล์และการรับอีเมลล์ โดยโพรโทคอล SMTP หรือ Simple Mail Transfer Protocol จะใช้ขณะที่ User Agent ส่งอีเมลล์มาที่ MTA (เฉพาะแบบ Offline) และใช้ขณะรับและส่งอีเมลล์ระหว่าง MTA ด้วยกัน สำหรับการใช้อีเมลล์แบบออฟไลน์ คือเครื่องที่ผู้ใช้อ่านเมลล์ไม่ได้ต่อกับเครื่องที่มีเมลล์บ็อกซ์ตลอดเวลา อาจเลือกดาวน์โหลดเมลล์มาเก็บไว้ที่เครื่องของตัวเองนั้น จะมีโพรโทคอลสำหรับรับอีเมลล์ที่เกี่ยวข้องอีก ที่ใช้งานกันแพร่หลายมีอยู่ 2 แบบคือ โพรโทคอล POP หรือ Post Office Protocol และ IMAP หรือ Internet Message Access Protocol ซึ่งจะทำหน้าที่ดาวน์โหลดหรืออัปโหลดอีเมลล์จากเครื่องของผู้ใช้ไปยังเครื่องที่มี MTA อยู่

รูปแบบของข้อมูลที่ใช้ในโพรโทคอลต่าง ๆ ของอีเมลล์นี้ถูกกำหนดไว้ใน RFC 822 ซึ่งแบ่งส่วนประกอบภายในอีเมลล์เป็น 2 ส่วนคือ

- ส่วนที่เป็นจ่าหน้าอีเมลล์

ในส่วนของจ่าหน้าอีเมลล์นี้มีไว้เป็นข้อมูลเพื่อให้ส่งไปถึงผู้รับ รูปแบบของข้อมูลจะเป็นข้อความหรือเท็กซ์ นำหน้าด้วยคำสำคัญ (Keyword) เช่น From หมายถึงชื่อผู้ส่ง ส่วน To หมายถึงผู้รับ เป็นต้น ซึ่งจะคล้ายกับการที่ต้องกำหนดเมื่อบันทึกอีเมลล์ ถัดจากคำสำคัญก็จะเป็นค่าของข้อมูลในชุดนั้น ๆ เช่น From ก็จะต่อด้วยชื่อของผู้ส่ง และ Reply To ก็จะต่อด้วยชื่อของผู้รับ เป็นต้น โดยแต่ละบรรทัดจะปิดท้ายด้วย Carriage Return และ/หรือ Line Feed (ขึ้นอยู่กับระบบปฏิบัติการที่ใช้ เช่น Windows จะปิดท้ายด้วย Carriage Return และ Line Feed ส่วนระบบปฏิบัติการอื่น เช่น Unix ก็อาจจะใช้เพียง Carriage Return เท่านั้น) เป็นเครื่องหมายของการสิ้นสุดบรรทัด จะเห็นได้ว่าในส่วนของการจ่าหน้าอีเมลล์นี้มีข้อความที่จำเป็นคือ รายละเอียดของผู้ส่งและผู้รับ ส่วนรายละเอียดอื่น ๆ เช่น รายชื่อผู้รับสำเนา (Cc) จะมีหรือไม่มีก็ได้

- ส่วนข้อมูลของอีเมล

ส่วนที่เป็นข้อมูลของอีเมล ซึ่งจะแบ่งย่อยออกได้เป็น 2 ส่วนคือ ส่วนหัว (Header) และ ส่วนเนื้อความของอีเมล ส่วนหัวนี้จะถูกสร้างขึ้นอย่างอัตโนมัติโดย User agent ของผู้ส่ง เพื่อให้ MTA ต่าง ๆ ระหว่างทางที่ส่งผ่านอีเมลฉบับนั้นได้อ่านไปใช้งาน ซึ่งประกอบด้วยข้อมูลต่าง ๆ หลายประเภท ตัวอย่างของข้อมูลในส่วนหัวของอีเมล ได้แก่ เลขทะเบียนของอีเมล (Message Header), วันที่และเวลาที่ส่ง เป็นต้น ส่วนที่เป็นเนื้อความของอีเมลนั้น จะเป็นบรรทัดที่แยกจากส่วนหัว โดยถูกคั่นด้วยบรรทัดว่าง ๆ (Null Line) และในแต่ละบรรทัดของเนื้อความจะสิ้นสุดบรรทัดด้วย Carriage Return และ/หรือ Line Feed

จากองค์ประกอบของโพรโทคอล และวิธีการรับ-ส่งอีเมลที่กล่าวมา ทำให้การใช้อีเมลในปัจจุบัน ซึ่งทำงานแบบไคลเอนต์เซิร์ฟเวอร์ (client-server) สามารถทำงาน ได้ 3 แบบ คือ

1. แบบออฟไลน์ (Offline) หรือเรียกว่า Download and Delete ซึ่งเป็นรูปแบบมาตรฐานทั่วไปในการใช้งานกับอีเมลของอินเทอร์เน็ต ซึ่งใช้โพรโทคอล POP หรือ IMAP โดย User Agent ของผู้รับจะดาวน์โหลดอีเมลทั้งหมดมาจากเมลเซิร์ฟเวอร์ และลบอีเมลเหล่านั้นออกไป (ในโปรแกรม ไคลเอนต์ของอีเมลบางโปรแกรมสามารถให้เลือกได้ว่าต้องการลบอีเมลที่ดาวน์โหลดมาแล้วทางฝั่งเซิร์ฟเวอร์นั้นทิ้งหรือไม่) ทำให้ผู้ใช้สามารถอ่านอีเมลนั้นได้ตลอดเวลา โดยไม่จำเป็นต้องติดต่อเมลเซิร์ฟเวอร์อีก แต่ User Agent จะไม่รู้ว่ามีอีเมลเข้ามาใหม่จนกว่าจะติดต่อเข้าไปยังเมลเซิร์ฟเวอร์ และดาวน์โหลดอีเมลเข้ามาใหม่

2. แบบออนไลน์ (Online) เป็นแบบที่อีเมลด้าน User Agent ของผู้รับจะต้องติดต่อกับเมลเซิร์ฟเวอร์ของผู้รับเองตลอดเวลาที่ใช้อีเมล ซึ่งระบบที่ให้บริการอีเมลแบบนี้ จะสามารถเปิด share mailbox ที่เซิร์ฟเวอร์ได้ตลอดเวลา เช่น NFS (Network File System) หรือ CIFS (Common Internet File System) เป็นต้น นอกจากนี้โพรโทคอลแบบ IMAP ยังสามารถใช้งานในแบบออนไลน์นี้ได้อีกด้วย

3. แบบ Disconnected เป็นแบบผสมผสานระหว่างแบบออฟไลน์ และแบบออนไลน์ โดยอาศัยเมลเซิร์ฟเวอร์ของผู้รับเป็นหลักในการจัดเก็บข้อมูลของอีเมล และในส่วนเนื้อที่ของ User Agent นี้จะเป็นที่เก็บอีเมลสำรอง โดยเมื่อมีการดาวน์โหลดอีเมลมากก็จะทำงานในแบบของออฟไลน์ เพื่อลดภาระที่ต้องติดต่อกับเมลเซิร์ฟเวอร์ตลอดเวลา แต่ข้อมูลอีเมลจะไม่ถูกลบออกจากเซิร์ฟเวอร์ ผู้ใช้สามารถโหลดอีเมลที่แก้ไขแล้วกลับไปยังเมลเซิร์ฟเวอร์ในภายหลังได้ เช่น การแก้ไขหรือตอบกลับอีเมล (Reply to) ที่ส่งมา เป็นต้น ซึ่งโพรโทคอลที่สามารถตอบสนองการใช้งานในแบบนี้ได้ก็คือ IMAP

2.6 Web Server

เว็บเซิร์ฟเวอร์ (Web Server) (ไพศาล โมลิสสกุลมงคล. 2544 : 22) คือแอปพลิเคชันที่ทำหน้าที่รับ และประมวลผลข้อมูลที่ร้องขอจากผู้ให้บริการอินเทอร์เน็ตโดยผ่านทางเว็บเบราว์เซอร์ หลังจากเว็บเบราว์เซอร์รับคำสั่งและประมวลผลแล้ว ผลลัพธ์จะถูกส่งกลับไปยังผู้ใช้ โดยแสดงผลในเว็บเบราว์เซอร์นั่นเอง นอกจากนี้เว็บเบราว์เซอร์จะให้บริการในอินเทอร์เน็ตแล้ว ยังสามารถนำมาประยุกต์ใช้ในเครือข่ายภายในองค์กร หรืออินทราเน็ตได้อีกด้วย

เว็บเซิร์ฟเวอร์สามารถติดตั้งบนคอมพิวเตอร์ที่มีระบบปฏิบัติการต่างๆ เช่น คอมพิวเตอร์ระบบยูนิกซ์ (Unix), คอมพิวเตอร์ภายใต้ Windows 95/98/ME, คอมพิวเตอร์ภายใต้ Windows NT หรือแม้แต่ Windows 2000 เป็นต้น สำหรับโปรแกรมเว็บเซิร์ฟเวอร์ที่ทำงานบนระบบจัดการต่างๆ ได้แสดงไว้ในตารางที่ 2.2

ตารางที่ 2.2 โปรแกรมเว็บเซิร์ฟเวอร์ที่ทำงานบนระบบจัดการต่างๆ

โปรแกรมเว็บเซิร์ฟเวอร์	ระบบปฏิบัติการ
Apache	Unix, Linux
Netscape	Unix
Apache	Windows 95/98/ME
Netscape Enterprise Server 2.0	Unix และ Windows NT
Netscape FastTrack Server 2.0	Unix และ Windows NT
Internet Information Server 3.0 และ 4.0	Windows NT
Internet Information Server 5.0	Windows 2000
Microsoft FrontPage Server 97	Windows NT , Windows 95/98/ME และ Macintosh
O'Reilly Website Server 1.1	Windows NT , Windows 95/98/ME
Personal Web Server 1.0 และ 4.0	Windows 95/98/ME
Purveyor WebServer	Windows NT , Windows 95/98/ME
OmniHTTPd Professional	Windows NT , Windows 95/98/ME

2.7 POP3 และ IMAP4

2.7.1 POP3

POP หรือ Post Office Protocol เป็นโพรโทคอลที่ทำหน้าที่โหลดอีเมลมาจาก MTA ไปยัง User agent ซึ่งในปัจจุบันได้พัฒนาจนถึงเวอร์ชัน 3 แล้ว หรือที่เรียกย่อๆว่า POP3 โพรโทคอลนี้เป็นตัวแรกที่ถูกออกแบบมาเพื่อใช้รับอีเมล และเพื่อให้สนับสนุนการทำงานแบบออฟไลน์ จึงมีการกำหนดรายละเอียดไว้ใน RFC 1939 ด้วย ซึ่งกลไกของ POP3 นี้จะทำงานในแบบออฟไลน์ โดยติดต่อเข้าไปยังเมลเซิร์ฟเวอร์และดาวน์โหลดอีเมลทั้งหมดมาไว้ที่ User agent จากนั้นจะลบอีเมลที่เซิร์ฟเวอร์นั้นทิ้งไป เพื่อป้องกันการดาวน์โหลดซ้ำ แต่ผู้ใช้งานจะทำงานแบบออนไลน์ กับเซิร์ฟเวอร์ไม่ได้ เนื่องจากการอ่านอีเมลจะดึงอีเมลที่เก็บไว้ใน User agent ขึ้นมาให้อ่าน หลังจากที่ดาวน์โหลดมาเก็บไว้ ซึ่งในขณะนั้นอาจจะไม่ได้ออนไลน์ อยู่กับเน็ตเวิร์คก็ได้

โพรโทคอลของ POP3 นี้ จะทำงานในแบบไคลเอนต์เซิร์ฟเวอร์ คือมีโปรแกรม POP Server ในเมลเซิร์ฟเวอร์ และ POP Client ในเครื่องผู้รับ ซึ่งปกติจะฝังอยู่ในโปรแกรมที่เป็น User agent เลย โปรแกรมทั้งสองจะติดต่อกันโดยใช้ชุดคำสั่งที่เป็นรหัส ASCII คือเมื่อด้านที่รับทำคำสั่ง ก็จะทำงานตามคำสั่งนั้น แล้วตอบกลับมาโดยมีค่าเป็น +OK หมายถึงทำงานได้เรียบร้อย หรือ -ERR หมายถึงเกิดปัญหาขึ้น ทำงานไม่ได้ ซึ่งในคำสั่งที่ต้องมีการตอบกลับและส่งข้อมูลกลับมา โดยประกอบด้วยข้อมูลหลายๆบรรทัดนั้น POP3 จะให้บรรทัดสุดท้ายเป็นเครื่องหมายจุด (.) ตามด้วย Carriage Return และ Line Feed หมายถึงการสิ้นสุดชุดข้อมูล แต่ในกรณีที่ข้อมูลบรรทัดสุดท้ายมีข้อมูลที่เป็นจุดด้วย จะใช้เทคนิคที่เรียกว่า Character Stuffing เพื่อแก้ปัญหา โดยจะเติมจุดลงไปอีกหนึ่งตัว เพื่อเป็นตัวบ่งชี้ว่าข้อมูลนั้นเป็นจุด ซึ่งจะแตกต่างจากสัญลักษณ์แสดงการสิ้นสุดของข้อมูล

การทำงานของ POP3 จะทำร่วมกับโพรโทคอล TCP โดยทั่วไปจะใช้พอร์ต (Port) 110 ในการติดต่อ ขั้นตอนการทำงานของ POP3 ประกอบด้วย 3 สถานะคือ สถานะขออนุมัติ, สถานะรับส่งรายการ และสถานะปรับปรุงข้อมูล ซึ่งในแต่ละสถานะจะรับรู้คำสั่งต่างๆของโพรโทคอลแตกต่างกัน โดยมีรายละเอียดต่างๆดังนี้ (สุวรรณ บุญชัยยะ และคณะ. 2543 : 129)

- สถานะขออนุมัติ (Authorization State)

เมื่อเริ่มต้นติดต่อกับเซิร์ฟเวอร์จะเป็นการเข้าสู่สถานะการขออนุมัติ โดยไคลเอนต์จะต้องแจ้งชื่อผู้ใช้และรหัสผ่าน (Password) เพื่อขออนุมัติจากเซิร์ฟเวอร์ก่อน โดยไคลเอนต์จะใช้คำสั่ง USER เพื่อระบุชื่อผู้ใช้ หรือคำสั่ง PASS เพื่อกำหนด Password แต่ในกรณีที่ชื่อและ

Password ถูกเข้ารหัสไว้ และไม่ได้เป็นค่า ASCII ทั่วไป โคลเอนต์จะใช้คำสั่ง APOP ทำงานแทนคำสั่ง USER และ PASS

- สถานะรับส่งรายการ (Transaction State)

หลังจากที่ได้รับอนุมัติจากเซิร์ฟเวอร์แล้ว ก็จะเข้าสู่สถานะที่ใช้คำสั่งในการทำงานต่างๆ

- สถานะปรับปรุงข้อมูล (Update State)

เมื่อ User agent เลิกใช้งานด้วยคำสั่ง QUIT ของ POP3 เซิร์ฟเวอร์ก็จะเข้าสู่สถานะปรับปรุงข้อมูล เพื่อลบอีเมลที่ดาวน์โหลดเรียบร้อยแล้วออกไป จากนั้นก็จะเข้าสู่สถานะขออนุมัติใหม่โดยอัตโนมัติ เพื่อรอรับการทํางานครั้งต่อไป

2.7.2 IMAP4

IMAP4 หรือที่เรียกว่า Internet Message Access Protocol 4 เป็นอีกโพรโทคอลหนึ่งที่ใช้ในการรับอีเมล ซึ่งมีประสิทธิภาพและสามารถใช้งานได้หลากหลายแบบมากกว่า POP IMAP ได้รับการพัฒนามาหลายเวอร์ชัน จนถึงปัจจุบันเป็นเวอร์ชัน 4 จุดเด่นของโพรโทคอล IMAP ก็คือ ผู้ใช้สามารถเลือกดาวน์โหลดเฉพาะอีเมลที่ต้องการได้ โดยไม่จำเป็นต้องโหลดมาทั้งหมดเหมือนโพรโทคอล POP3 นอกจากนี้ IMAP ยังสามารถรองรับการทำงานได้ทั้งแบบออฟไลน์, ออนไลน์และ Disconnected อีกด้วย ดังนั้นประโยชน์ที่ได้รับจาก IMAP ก็คือ หากผู้ใช้อีเมลแอดเดรส (E-mail Address) เพียงชื่อเดียว แต่มีเครื่องที่ใช้งานอยู่หลายเครื่อง เช่น เครื่องที่บ้าน, ที่ทำงาน หรือแล็ปท็อป ก็จะสามารถเลือกดาวน์โหลดเฉพาะอีเมลที่ต้องการมาเก็บไว้ที่เครื่องใดก็ได้ แต่ถ้าเป็น POP3 การดาวน์โหลดจะต้องทำพร้อมกันหมดทุกอีเมล ดังนั้น IMAP จึงเป็นโพรโทคอลที่สามารถใช้งานกับสายสื่อสารที่มีความเร็วต่ำได้เป็นอย่างดี

การทำงานของ IMAP นี้จะเหมือนกับโพรโทคอลอื่นๆ โดยทำงานร่วมกับ TCP ใช้พอร์ตหมายเลข 143 และจะแบ่งสถานะต่างๆออกเป็น 4 สถานะ โดยในแต่ละสถานะจะมีวัตถุประสงค์และคำสั่งที่ใช้งานแตกต่างกันไป โดยมีรายละเอียดต่างๆดังนี้ (สุวรรณ ภูณชัชยะ และคณะ. 2543 : 133)

- สถานะก่อนอนุมัติ (Non-authenticated State)

เป็นสถานะที่กำลังรอให้โคลเอนต์ติดต่อเข้ามา เพื่อขออนุมัติใช้ ดังนั้นในด้านโคลเอนต์จะต้องแจ้งชื่อ Login ของเมลเซิร์ฟเวอร์นั้น และ Password ด้วยคำสั่ง LOGIN หรือ AUTHENTICATE ก่อนจึงจะเริ่มใช้งานได้ จากนั้นจึงเปลี่ยนไปเป็นสถานะได้รับการอนุมัติ

- สถานะได้รับการอนุมัติ (Authenticated State)

เป็นสถานะที่สามารถใช้คำสั่งต่างๆที่เกี่ยวกับการเลือกและใช้งานเมลล์บ็อกซ์ เช่น คำสั่ง SELECT เพื่อเลือกเมลล์บ็อกซ์ หรือคำสั่ง CREATE เพื่อสร้างเมลล์บ็อกซ์ เป็นต้น ในการเลือกคำสั่งเมลล์บ็อกซ์ด้วยคำสั่ง SELECT หรือ EXAMINE นี้จะเป็นการเปลี่ยนไปเป็นสถานะการเลือกเมลล์บ็อกซ์

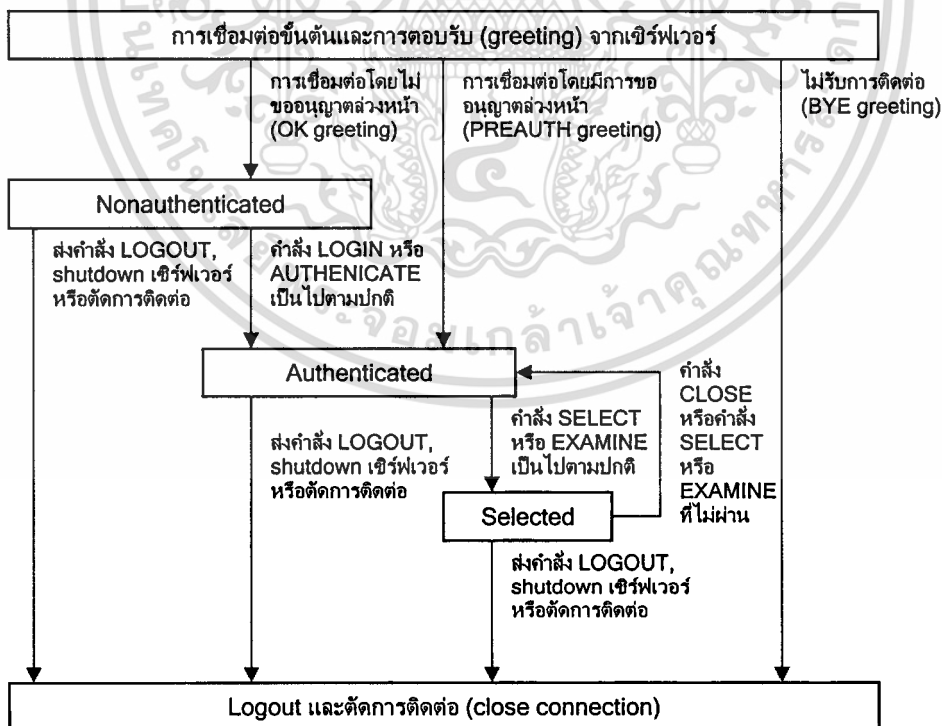
- สถานะเลือกเมลล์บ็อกซ์ (Select State)

เป็นสถานะที่จะเข้าใช้งานอีเมลในแต่ละเมลล์บ็อกซ์ หลังจากทีเลือกเมลล์บ็อกซ์ไว้แล้วในสถานะก่อนหน้านี้

- สถานะเลิกใช้งาน (Logout State)

เมื่อต้องการเลิกใช้งาน หรือสิ้นสุดการทำงานของ IMAP จะเข้าสู่สถานะการเลิกใช้งานโดยใช้คำสั่ง LOGOUT

จากสถานะทั้ง 4 นี้ไม่จำเป็นต้องทำงานเรียงต่อกันเสมอไป บางครั้งอาจจะมีการทำงานข้ามสถานะหนึ่งไปยังอีกสถานะหนึ่งได้ ซึ่งการทำงานในสถานะต่างๆของโปรโตคอล IMAP4 ตามมาตรฐาน RFC 2060 นี้จะแสดงเป็นขั้นตอนได้ดังรูปที่ 2.3



รูปที่ 2.3 ขั้นตอนและสถานะการทำงานของโปรโตคอล IMAP4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8 SMTP

โพรโทคอลที่คู่กับ POP3 ก็คือ SMTP (Simple Mail Transfer Protocol) (สุวรรณ ปุณณชัยยะ และคณะ. 2543 : 131) เพราะเป็นโพรโทคอลที่ใช้ส่งอีเมลจาก User agent ของผู้ส่งไปยัง MTA ของผู้ส่ง และส่งต่อไปยัง MTA เครื่องอื่นๆที่เป็นจุดผ่านในการเชื่อมต่อไปยังเครื่องของผู้รับ โพรโทคอล SMTP จะทำงานร่วมกับโพรโทคอล TCP โดยใช้พอร์ต 25 ซึ่งคำสั่งต่างๆของ SMTP จะเป็นลักษณะเดียวกับ POP3 คือเป็น ASCII ลงท้ายด้วย Carriage Return และ Line Feed ส่วนข้อความที่ตอบกลับมาจะนำหน้าด้วยเลข 3 หลัก เป็นสถานะแสดงสถานะการทำงานของคำสั่งที่ได้รับ

เมื่อเริ่มต้นการติดต่อ SMTP จะกำหนดให้ User agent ของผู้ส่งต้องส่งคำสั่ง HELLO พร้อมกับรายละเอียดด้านผู้ส่งออกไป จากนั้นจะส่งคำสั่ง MAIL เพื่อแจ้งให้เซิร์ฟเวอร์เตรียมรับอีเมล ในส่วนของเซิร์ฟเวอร์เมื่อพร้อมที่จะรับอีเมลก็จะตอบรับกลับมาด้วยคำสั่ง OK จากนั้นที่ด้านส่งก็จะเริ่มส่งโดยใช้คำสั่ง RCPT เพื่อกำหนดเมลแต่ละฉบับที่ส่งไป ซึ่งการส่งข้อมูลของอีเมลจะถูกระบุด้วยคำสั่ง DATA

การส่งอีเมลของโพรโทคอล SMTP ได้จัดเตรียมคำสั่งอื่นๆไว้เพื่ออำนวยความสะดวกและคล่องตัวในการทำงาน ซึ่งจะประกอบด้วยคำสั่ง VRFY เพื่อให้ด้านที่ส่งตรวจสอบรายชื่อว่าผู้ใช้รายนี้มีสิทธิ์ใช้งานอีเมลบ็อกซ์นั้นๆหรือไม่, คำสั่ง EXPN ใช้จัดการและตรวจสอบรายชื่อจากลิสต์รายชื่อ และคำสั่ง TURN ใช้สลับให้ไคลเอนต์ของผู้ส่งทำหน้าที่รับข้อมูลจากเซิร์ฟเวอร์แทน

เมื่อได้รับคำสั่งต่างๆของผู้ส่งแล้ว เซิร์ฟเวอร์จะมีหน้าที่ตรวจสอบความถูกต้องของคำสั่ง จากนั้นจึงทำงานตามคำสั่งและส่งผลตอบกลับมา ส่วนลักษณะของข้อมูลที่ตอบกลับ (reply message) นั้นจะเป็นข้อมูลที่อยู่ในรูปของเท็กซ์ที่เป็น ASCII โดยจะประกอบด้วยตัวเลขนำหน้าข้อความ 3 หลัก ทำหน้าที่แสดงสถานะการทำงานของเซิร์ฟเวอร์ และเปลี่ยนสถานะการทำงานของโพรโทคอล SMTP ด้วย ถัดจากตัวเลขจะคั่นด้วยช่องว่างแล้วตามด้วยข้อความ ซึ่งปิดท้ายด้วยเครื่องหมาย Carriage Return และ Line Feed ตัวอย่างเช่น 500 Syntax error, command unrecognized หมายถึง คำสั่งที่ส่งไปไม่ถูกต้อง หรือ 503 Bad sequence of commands หมายถึง ลำดับการส่งคำสั่งไม่ถูกต้อง เป็นต้น

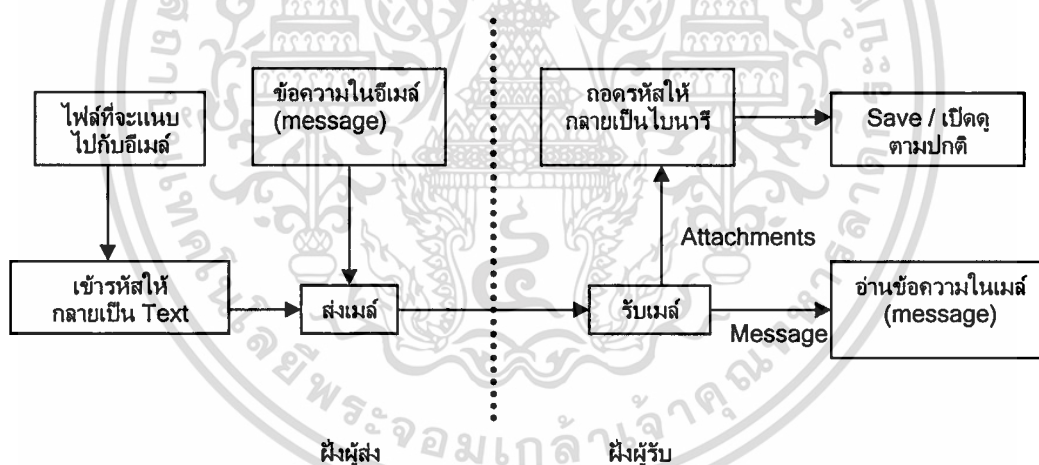
2.9 MIME

MIME ย่อมาจาก Multipurpose Internet Mail Extensions (กอบเกียรติ สระอุบล. 2545 : 180) ซึ่งจัดว่าเป็นโพรโทคอลตัวหนึ่งที่ใช้เป็นหลักมาตรฐาน เพื่อให้สามารถส่งไฟล์ไบนารี

(Binary) ไปกับเมลได้ ซึ่งไฟล์ไบนารีที่รู้จักกันทั่วไป ได้แก่ รูปภาพกราฟิก เอกสารเวิร์ด เวอร์คชีทเอ็กเซล ไฟล์ zip ไฟล์ movie และอื่นๆอีกมากมาย รวมไปถึงไวรัสตัวร้ายด้วย

ทั้งนี้เนื่องจากอีเมลที่เราส่งกันอยู่ปัจจุบัน ทำงานในลักษณะโหมดอักษร (text mode จัดอยู่ในกลุ่ม text based protocol) ทุกอย่างที่จะส่งไปจะเป็นอักษรหรือข้อความทั้งหมด ดังนั้นการที่จะส่งไฟล์ไบนารีรวมไปกับอักษรข้อความอีเมลตรงๆจะทำได้ ต้องผ่านการแปลงหรือเข้ารหัส (encoding) ให้ข้อมูลไบนารีนั้นกลายเป็นอักษรเสียก่อน จากนั้นก็นำข้อมูลไฟล์แนบ ซึ่งตอนนี้ถูกเข้ารหัสกลายเป็นอักษรข้อความไปแล้ว (แต่อ่านไม่รู้เรื่อง) ใส่รวมกลุ่มกับอีเมลที่จะส่ง แล้วส่งเมลออกไปพร้อมกันตามปกติ

เมื่ออีเมลถึงปลายทาง โปรแกรมอีเมลของผู้รับ เช่น เว็บเบสอีเมล (yahoo, hotmail) หรืออีเมลไคลเอนต์ (Outlook, Eudora) ก็จะถอดรหัส (decode) อักษรเฉพาะในส่วนไฟล์แนบ (Attachments) ให้กลายเป็นไบนารีอย่างเดิมเสียก่อน จากนั้นก็แล้วแต่ผู้เปิดอ่านเมลว่าจะทำอะไรต่อไป เช่น เก็บบันทึก (save) หรือเปิดดูไฟล์ ฯลฯ



รูปที่ 2.4 ไดอะแกรมการทำงานรับส่งเมล

บทที่ 3

ระบบเว็บเบสอีเมลในปัจจุบัน

3.1 องค์ประกอบของการทำงานของเว็บเบสอีเมล

การทำงานของระบบเว็บเบสอีเมลจะประกอบไปด้วยส่วนต่างๆ ดังต่อไปนี้

1. เมล์เซิร์ฟเวอร์ (Mail Server) เป็นเครื่องคอมพิวเตอร์ที่มีโปรแกรมทำหน้าที่ให้บริการรับส่งอีเมล เปรียบได้กับที่ทำการไปรษณีย์

2. เมล์ไคลเอนต์ (Mail Client) เป็นเครื่องคอมพิวเตอร์ที่มีโปรแกรมทำหน้าที่ให้บริการเขียน-อ่านอีเมล เปรียบได้กับกระดาษและปากกา

3. SMTP (Simple Mail Transfer Protocol) เป็นโพรโทคอลที่ใช้ในการส่งอีเมลจากเมล์ไคลเอนต์ไปยังเมล์เซิร์ฟเวอร์และระหว่างเมล์เซิร์ฟเวอร์ด้วยกัน

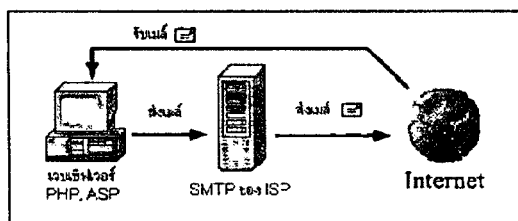
4. POP (Post Office Protocol) หรือ IMAP (Internet Message Access Protocol)

- POP ปัจจุบันเป็นเวอร์ชัน 3 เป็นโพรโทคอลที่ใช้สำหรับโหลด อีเมลจากเมล์เซิร์ฟเวอร์ไปยังไคลเอนต์ ซึ่งจะทำงานในแบบไคลเอนต์เซิร์ฟเวอร์

- IMAP ปัจจุบันเป็นเวอร์ชัน 4 เป็นอีกวิธีในการรับอีเมล ซึ่งได้รับการแก้ไขและปรับปรุงจาก POP จึงทำให้มีประสิทธิภาพและสามารถใช้งานได้หลากหลายมากกว่า IMAP ถูกออกแบบมาใช้งานในลักษณะ Remote

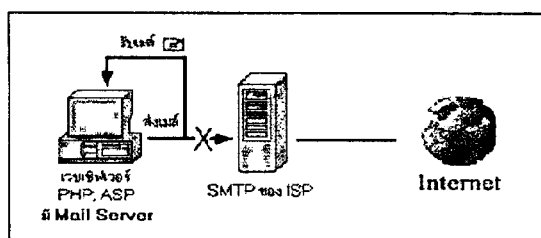
5. Web Server เป็นแอปพลิเคชันที่ทำหน้าที่รับและประมวลผลข้อมูลที่ร้องขอจากผู้ใช้บริการอินเทอร์เน็ต โดยผ่านเว็บเบราว์เซอร์

การทำงานของระบบเว็บเบสอีเมลที่ประกอบไปด้วยส่วนต่างๆข้างต้นนั้น สามารถแสดงได้ดังรูปที่ 3.1 และ 3.2



รูปที่ 3.1 วงจรปกติของการส่งเมลโดยต้องออนไลน์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 ส่งเมลล์และรับเมลล์โดยไม่ต้องออนไลน์

3.2 องค์ประกอบภายในอีเมลล์

เมลล์ไคลเอนต์ทุกชนิดจะมีรายการภายในอีเมลล์ตามแบบมาตรฐาน RFC 822 ที่ถูกเสนอ โดย David H. Crocker เมื่อปี 1982 โดยกำหนดชื่อหัวข้อว่า STANDARD FOR THE FORMAT OF ARPA INTERNET TEXT MESSAGES โดยรายการหลัก ๆ มีดังนี้

ตารางที่ 3.1 รายการหลักภายในอีเมลล์

รายการ	คำอธิบาย
To:	ใส่อีเมลล์แอดเดรสของผู้รับ เช่น sysadmin@kmitl.ac.th
From:	ใส่อีเมลล์แอดเดรสของผู้ส่ง เช่น knorasa@kmitl.ac.th ซึ่งมักจะถูกใส่อัตโนมัติโดยโปรแกรมที่ใช้ หรืออาจจะเป็นชื่อก็ได้ เช่น "Orasa Nuwan" <knorasa@kmitl.ac.th>
Subject:	ชื่อเรื่องจดหมาย ไม่จำเป็นต้องใส่ แต่แนะนำว่าควรใส่เพื่อให้สื่อให้รู้ว่าเนื้อหาของจดหมายจะเป็นเรื่องเกี่ยวกับอะไร เช่น "How are you?" เป็นต้น
CC:	ย่อมาจาก Carbon copy ใช้ในกรณีที่ต้องการสำเนาจดหมายให้บุคคลอื่น ๆ ทราบเรื่อง
Bcc:	ย่อมาจาก Blind carbon copy เป็นการสำเนาจดหมายเหมือน CC แต่ผู้รับในช่อง To จะไม่เห็นที่เราสำเนาจดให้ใครบ้าง
Attachments:	ใส่ชื่อไฟล์ที่ต้องการส่งแนบไปพร้อมข้อความ

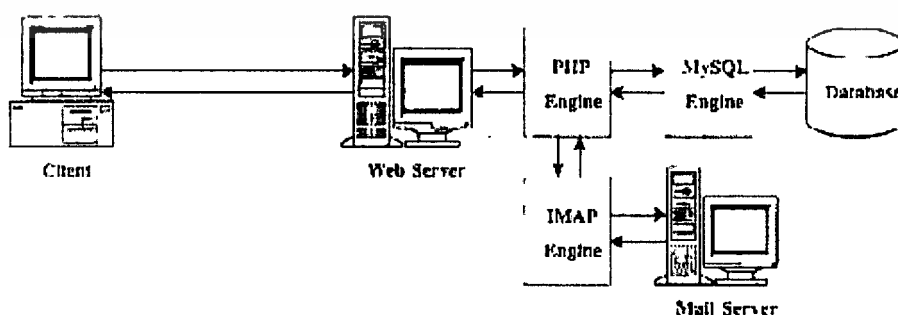
ตารางที่ 3.1 รายการหลักภายในอีเมล (ต่อ)

Body:	พิมพ์ข้อความจดหมาย
Signature:	ข้อความสั้น ๆ จำนวน 3-4 บรรทัด เพื่อระบุข้อมูลส่วนตัวของผู้ส่ง เช่น ชื่อ-สกุล ที่อยู่ ที่ทำงาน เบอร์โทร เป็นข้อความที่ผู้ส่งกำหนดขึ้นโดยพิมพ์ครั้งเดียว แล้วสามารถใช้ได้ตลอดไม่ต้องพิมพ์ซ้ำ

3.3 ขั้นตอนการทำงานของระบบเว็บเมล

การทำงานของระบบจะเริ่มจาก บราวเซอร์บนเครื่องไคลเอนต์ทำการร้องขอเว็บเพจไปยังเว็บเซิร์ฟเวอร์ เมื่อเว็บเซิร์ฟเวอร์ได้รับการร้องขอ จะส่งเว็บเพจหน้าแรกของเว็บเมลนั้นกลับไปยังเว็บบราวเซอร์ เพื่อรับข้อมูลของผู้ใช้และส่งกลับมายังเว็บเซิร์ฟเวอร์อีกที เมื่อเว็บเซิร์ฟเวอร์ได้รับข้อมูลจากเครื่องไคลเอนต์แล้ว จะส่งข้อมูลที่ไต่ไปประมวลผลบน PHP Engine โดย PHP Engine จะตรวจสอบว่า ข้อมูลภายในสคริปต์นั้น มีการเรียกใช้ฟังก์ชันการทำงานที่จะต้องเรียกใช้บริการของ database หรือเมลเซิร์ฟเวอร์หรือไม่

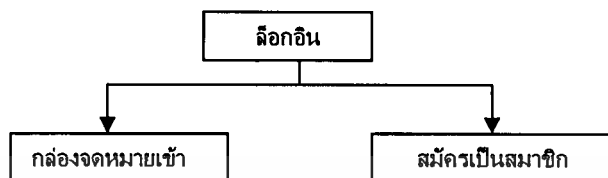
ในกรณีที่มีการเรียกใช้บริการบนฐานข้อมูล MySQL PHP Engine ก็จะทำการส่งข้อมูลดังกล่าวไปประมวลผลบน MySQL Engine อีกที ส่วนในกรณีที่เป็นการเรียกใช้บริการของเมลเซิร์ฟเวอร์ จะส่งข้อมูลที่ไต่ไปประมวลผลบน IMAP Engine แทน นอกจากนี้ผลลัพธ์ที่ได้จากการประมวลผลบน MySQL Engine สามารถส่งต่อไปยัง IMAP Engine เพื่อประมวลผลต่อได้ โดยผ่านทาง PHP Engine ผลลัพธ์ที่ได้จะถูกส่งกลับมายัง PHP Engine อีกที เพื่อทำการแปลงข้อมูลให้อยู่ในรูปแบบที่สามารถแสดงผลบนเว็บเพจได้ และส่งต่อไปยังเว็บเซิร์ฟเวอร์ เพื่อนำเว็บเพจนั้นไปแสดงผลบนบราวเซอร์ของเครื่องไคลเอนต์อีกที



รูปที่ 3.3 การทำงานโดยรวมของเว็บเมล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- โครงสร้างการทำงานของเว็บเมลสามารถแบ่งออกได้เป็นส่วนใหญ่ๆได้ 3 ส่วน ดังนี้ คือ
1. ส่วนที่ทำหน้าที่ในการล็อกอิน เพื่อใช้บริการ



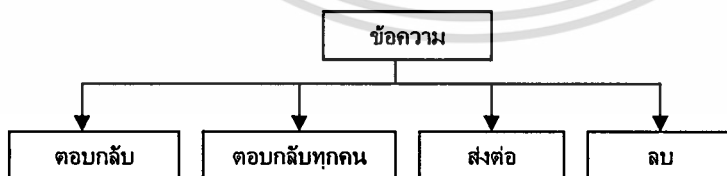
รูปที่ 3.4 การล็อกอินเพื่อใช้บริการ

2. ส่วนที่ทำหน้าที่ในการจัดการเกี่ยวกับเมลบ็อกซ์



รูปที่ 3.5 การจัดการเกี่ยวกับเมลบ็อกซ์

3. ส่วนที่ทำหน้าที่จัดการกับตัวข้อความ



รูปที่ 3.6 การจัดการกับตัวข้อความ

ในส่วนแรกจะจัดการเกี่ยวกับการล็อกอินของผู้ใช้ เพื่อติดต่อไปยังเมลเซิร์ฟเวอร์ ในกรณีที่ผู้ใช้ยังไม่ได้เป็นสมาชิกของเว็บเมลจะต้องทำการสมัครเป็นสมาชิกใหม่เสียก่อน ส่วนผู้ที่มีเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บัญชีรายชื่ออยู่แล้ว สามารถล็อกอินเพื่อเข้าใช้บริการได้ทันที ในส่วนที่สองจะจัดการเกี่ยวกับเมลล์บ็อกซ์ เมื่อผู้ใช้งานทำการล็อกอินเข้าสู่ระบบแล้ว สามารถจัดการข้อมูลเกี่ยวกับเมลล์บ็อกซ์ต่างๆ ได้ ดังนี้คือ แสดงรายการของข้อความทั้งหมด สร้างข้อความใหม่ แก้ไขรหัสผ่านของเมลล์บ็อกซ์นั้นๆ และออกจากระบบ ในส่วนสุดท้ายนี้จะเกี่ยวข้องกับการจัดการตัวข้อความโดยตรง ดังนั้นผู้ใช้งานสามารถตอบกลับ ส่งต่อ และลบข้อความนั้นๆ ได้

3.4 ปัญหาที่พบในระบบปัจจุบัน

- อินเทอร์เฟซไม่ตรงกับความต้องการของผู้ใช้ ซึ่งอาจจะเกิดจากการจัดวางรูปแบบที่ไม่เหมาะสม การใช้งานยาก ไม่สวยงาม ฯลฯ
- ขนาดพื้นที่ที่ใช้สำหรับเก็บจดหมายมีขนาดจำกัด
- การส่งไฟล์แนบ (Attachment) ส่งได้ขนาดไม่ใหญ่มากนัก
- ในกรณีที่มิไฟล์แนบส่งมาเป็นรูปภาพ เมื่อเปิดอ่านเมลล์แล้วเว็บเมลล์บางแห่งจะไม่สามารถแสดงรูปภาพ ได้ในทันที ต้องทำการดาวน์โหลดเครื่องคอมพิวเตอร์ก่อน
- จดหมายหาย ปัญหานี้อาจเกิดได้จากหลายสาเหตุ เช่น ความผิดพลาดของคน ความผิดพลาดของซอฟต์แวร์ ความล้มเหลวของฮาร์ดแวร์ เป็นต้น
- จดหมายส่งไปผิด คือ ไปยังผู้รับผิดคน ปัญหานี้อาจเกิดการที่ระบุอีเมลล์แอดเดรสของผู้รับผิด
- มีการปลอมจดหมาย

บทที่ 4

การวิเคราะห์และออกแบบระบบ

4.1 แนวความคิดในการออกแบบระบบ

การพัฒนาเว็บเบสอีเมลในโครงการพัฒนาระบบงานนี้ มีแนวความคิดในการออกแบบที่จะให้ระบบมีความสามารถดังต่อไปนี้

- กำหนดให้จำ User Name และรหัสผ่านไว้ในเครื่องได้ (Remember Login) ทราบต่อไปสามารถเปิดเข้าเมลบ็อกซ์ได้ โดยไม่ต้องป้อน User Name และรหัสผ่าน
- ที่หน้าหลัก (Inbox) มีการเน้นอีเมลที่ยังไม่ได้เปิดอ่านให้แตกต่างกับอีเมลที่เปิดอ่านแล้ว เพื่อให้เห็นเด่นชัด นอกจากนี้ยังบอกจำนวนอีเมลที่ยังไม่ได้เปิดอ่านอีกด้วย
- อีเมลใดที่มี Attachment จะมีสัญลักษณ์คลิปหนีบกระดาษ โข่วในหน้าหลักลิสต์ (list) รายการอีเมล
- ในกรณีที่ไฟล์ Attachment ส่งมาเป็นรูปภาพ เมื่อเปิดอ่านเมลแล้วระบบจะแสดงรูปภาพให้เลย
- ถอดรหัส MIME header อัตโนมัต เช่น เมลที่ส่งมาจาก Outlook
- ถอดรหัส body mail message ให้เมลที่ส่งมาแบบเข้ารหัส สามารถอ่านได้ เช่น เมลที่ส่งมาจาก Outlook
- คำนวณโหลด Attachment ทุกชนิดที่ส่งมากับอีเมลได้
- ข้อความที่เป็น URL, FTP, email จะเปลี่ยนเป็นลิงค์ (link) ให้อัตโนมัติ
- ลบอีเมลได้ที่ละฉบับหรือลบทีละหลายฉบับก็ได้
- มีตู้เก็บเมลที่ส่งออกไปแล้ว (Sent box) และสามารถย้ายไปไว้ในตู้ Inbox ได้ด้วย
- มีถังขยะเก็บอีเมลที่ถูกลบ (Trash can) และสามารถกู้กลับคืนมาไว้ในตู้ Inbox ได้อีกด้วย
- มีระบบอัตโนมัติเคลียร์อีเมลออกจากตู้เก็บเมลที่ส่งออกไปแล้ว (Sent box) และเคลียร์ถังขยะ (Trash can) เมื่อเลยเวลาตามที่กำหนดไว้ โดยสามารถตั้งได้ว่าลบอีเมลที่เก่ากว่ากี่วันทิ้งไปเพื่อป้องกันตู้ Sent box, ถังขยะเต็มไปด้วยอีเมลเก่าที่ไม่ได้ใช้

- มีระบบค้นหาคำในอีเมล (Search) เช่น ค้นหาอีเมลแอดเดรสผู้ส่ง, ค้นหาข้อความ
- มีระบบเก็บข้อความลงท้ายจดหมาย (Signature) และเรียกใช้ได้อัตโนมัติ
- มีระบบการบล็อก (Block) อีเมลและโดเมนที่ไม่ต้องการรับ
- มีระบบจัดเก็บอีเมลแอดเดรส (Address Book) เพื่อความสะดวกในการใช้งาน
- มีการแจ้งเตือน เมื่อขนาดเมลบ็อกซ์เต็มโควตา
- มีระบบบันทึกจำนวนครั้งที่พยายาม Login แต่เข้าไม่ได้ ซึ่งเกิดการห้สผ่านคิดรวมทั้งเก็บ IP และวันเวลาดำสุด เพื่อตรวจสอบว่ามีผู้พยายามลักลอบ Login เข้ามาหรือไม่
- ส่งเมลพร้อมกับแนบไฟล์ Attachment ได้มากกว่า 1 ไฟล์ โดยสามารถตั้งจำนวนไฟล์ที่จะส่งได้
 - มีการแสดงรายชื่อไฟล์และขนาดของไฟล์ที่จะแนบไปกับเมลให้ทราบ
 - เลือก Remove เพื่อยกเลิกไฟล์ Attachment ใดออกจากรายการก็ได้
 - มีการสร้าง MIME ไฟล์แนบและเข้ารหัสอัตโนมัติ
 - เมื่อผู้ใช้กรอกข้อมูลเข้าแบบฟอร์มส่งเมลแล้ว เช่น To:, เนื้อความหรืออื่นๆ จากนั้นคลิกปุ่มต่างๆ เช่น Remove, Add ข้อมูลในฟอร์มที่กรอกไว้ก่อนหน้านี้จะยังคงอยู่ ทำให้ไม่ต้องเสียเวลาพิมพ์ใหม่
 - ระบบจะทำการลบไฟล์ Attachment จากเซิร์ฟเวอร์ทิ้งโดยอัตโนมัติ หลังจากที่ผู้ใช้คลิกปุ่ม Remove หรือคลิกปุ่ม Send ส่งเมลแล้ว เพื่อไม่ให้เปลืองเนื้อที่บนเซิร์ฟเวอร์
 - มีฟังก์ชันตรวจสอบว่าไฟล์ Attachment อยู่บนเซิร์ฟเวอร์หรือไม่ ถ้าไม่พบก็จะส่งเฉพาะข้อความที่จะส่งไปเท่านั้น ป้องกันไม่ให้เกิดการส่งเกิด error
 - ในเมลมีการส่งข้อมูล 3 ส่วนด้วยกัน คือ แบบเว็บเพจ html, Text ธรรมดา และใน ส่วน Attachment ซึ่งถ้าโปรแกรมเมลผู้รับอ่านอีเมลแบบ html ไม่ได้ ก็จะเห็นเป็นข้อความ Text ธรรมดาแทน
 - กรอบรับข้อความ Textarea สามารถกำหนดขนาดอักษรให้ใหญ่ขึ้นได้
 - ระบบสามารถตรวจสอบเช็ครูปแบบอีเมลแอดเดรสได้ ว่าถูกต้องตามรูปแบบและครบถ้วนสมบูรณ์หรือไม่

4.2 ขั้นตอนการทำงานของระบบ

การทำงานของเว็บเมลจะเริ่มต้นจากผู้ทำการล็อกอินเข้าสู่ระบบเว็บเมล โดยจะแสดงหน้าแรกของเว็บเมลคือ index.html เมื่อเว็บเมลได้รับข้อมูลแล้วจะตรวจสอบว่ามีชื่อผู้ใช้และ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รหัสผ่านครบถ้วนหรือไม่ ถ้าครบถ้วนขั้นต่อไปจะเป็นกระบวนการตรวจสอบความถูกต้อง โดยจะนำชื่อผู้ใช้และรหัสผ่านที่ได้รับส่งไปยังฐานข้อมูล MySQL เพื่อทำการตรวจสอบว่า ผู้ใช้มีรายชื่ออยู่ในตาราง user หรือไม่ ถ้ามีจะทำการสร้างตัวแปร session เพื่อเก็บชื่อผู้ใช้ที่ผ่านการตรวจสอบ จากนั้นจะส่งชื่อผู้ใช้และรหัสผ่านไปยังเมลเซิร์ฟเวอร์อีกที เพื่อทำการดึงข้อมูลที่อยู่ในเมลบ็อกซ์นั้นมาแสดง การทำงานของส่วนนี้จะถูกรวบรวมไว้ในไฟล์ func.inc.php ถึงส่วนนี้แล้วจะได้เว็บเพจที่แสดงรายละเอียดของเฮดเดอร์ (header) ของทุกข้อความภายในเมลบ็อกซ์ที่อยู่ภายในรูปแบบของเอกสาร HTML ภายในเว็บเพจนี้ผู้ใช้จะสามารถจัดการกับข้อมูลต่างๆที่อยู่ในเมลบ็อกซ์ได้

เมื่อผู้ใช้ระบบเลือกข้อความที่อยู่ในอินบ็อกซ์ (Inbox) ข้อความที่ถูกเลือกจะถูกนำมาแสดงรายละเอียดเกี่ยวกับเฮดเดอร์ของข้อความ รวมถึงเนื้อหาภายในข้อความนั้นด้วยไฟล์ viewmsg.php นอกจากนี้แล้ว ผู้ใช้ยังสามารถจัดการกับข้อความที่ถูกนำมาแสดง ซึ่งสามารถแบ่งการทำงานแยกย่อยออกมาได้อีก 4 ขั้นตอน ประกอบด้วย การตอบข้อความกลับไปยังผู้ส่ง การดาวน์โหลดไฟล์แนบ การส่งต่อข้อความ และลบข้อความออกจากเมลบ็อกซ์

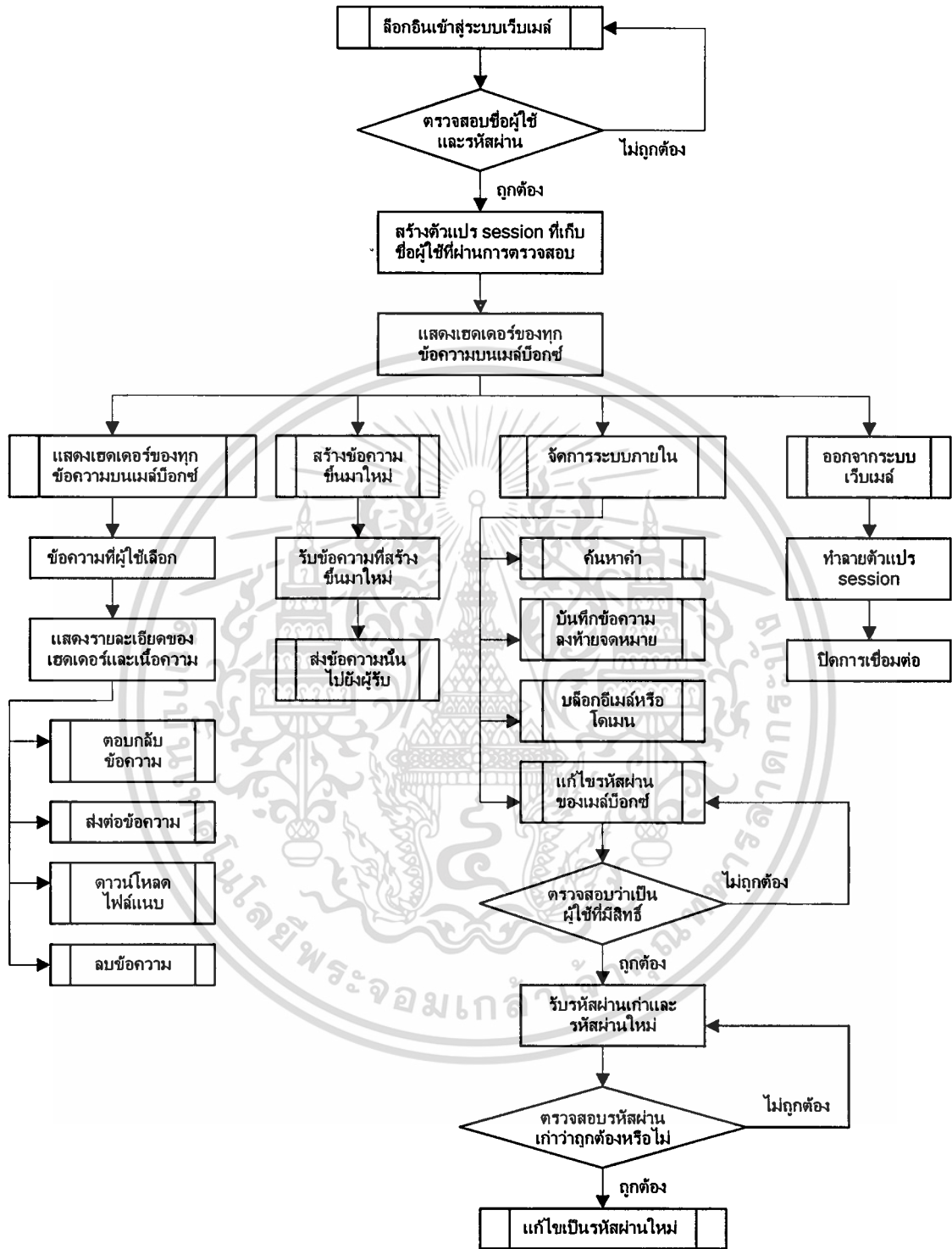
การสร้างข้อความขึ้นมาใหม่ ผู้ใช้จะต้องทำการป้อนอีเมลแอดเดรสของผู้รับ ตามด้วยหัวข้อหรือข้อความต่างๆ จากนั้นข้อมูลทั้งหมดที่ถูกสร้างขึ้น จะถูกส่งไปยังอีเมลแอดเดรสที่ระบุในส่วนของการแสดงข้อความทั้งหมดที่อยู่ในอินบ็อกซ์ ผู้ใช้จะทราบรายละเอียดเกี่ยวกับข้อมูลต่างๆที่อยู่ในแต่ละข้อความ เช่น ชื่อผู้ส่ง หัวเรื่อง วันที่ได้รับข้อความ ขนาดของข้อความ นอกจากนี้ ผู้ใช้ยังสามารถทราบได้อีกว่า ข้อความนั้นเป็นข้อความที่เคยอ่านมาก่อนหรือไม่

ในกรณีที่ผู้ใช้เลือกการจัดการระบบภายใน ก็จะมีการทำงานให้เลือก 4 อย่าง ประกอบด้วย

- การค้นหาค่า
- การบันทึกข้อความลงท้ายจดหมาย
- การบล็อกอีเมลหรือโดเมน
- การเปลี่ยนรหัสผ่าน จะต้องป้อนรหัสผ่านเก่าในส่วนของการเปลี่ยนรหัสผ่านของเมลบ็อกซ์ด้วย เพื่อเป็นการยืนยันว่าผู้ใช้มีสิทธิในเมลบ็อกซ์นั้นจริง หลังจากนั้นรหัสผ่านเมลบ็อกซ์จึงจะถูกเปลี่ยนเป็นรหัสผ่านใหม่

ในส่วนสุดท้ายจะเป็นการออกจากระบบ ภายในส่วนนี้ จะทำหน้าที่ในการทำลายตัวแปร session ที่ถูกสร้างไว้ก่อนหน้านี้ พร้อมทั้งปิดการเชื่อมต่อ

โดยขั้นตอนการทำงานจะแสดงดังรูปที่ 4.1



รูปที่ 4.1 ขั้นตอนการทำงานของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

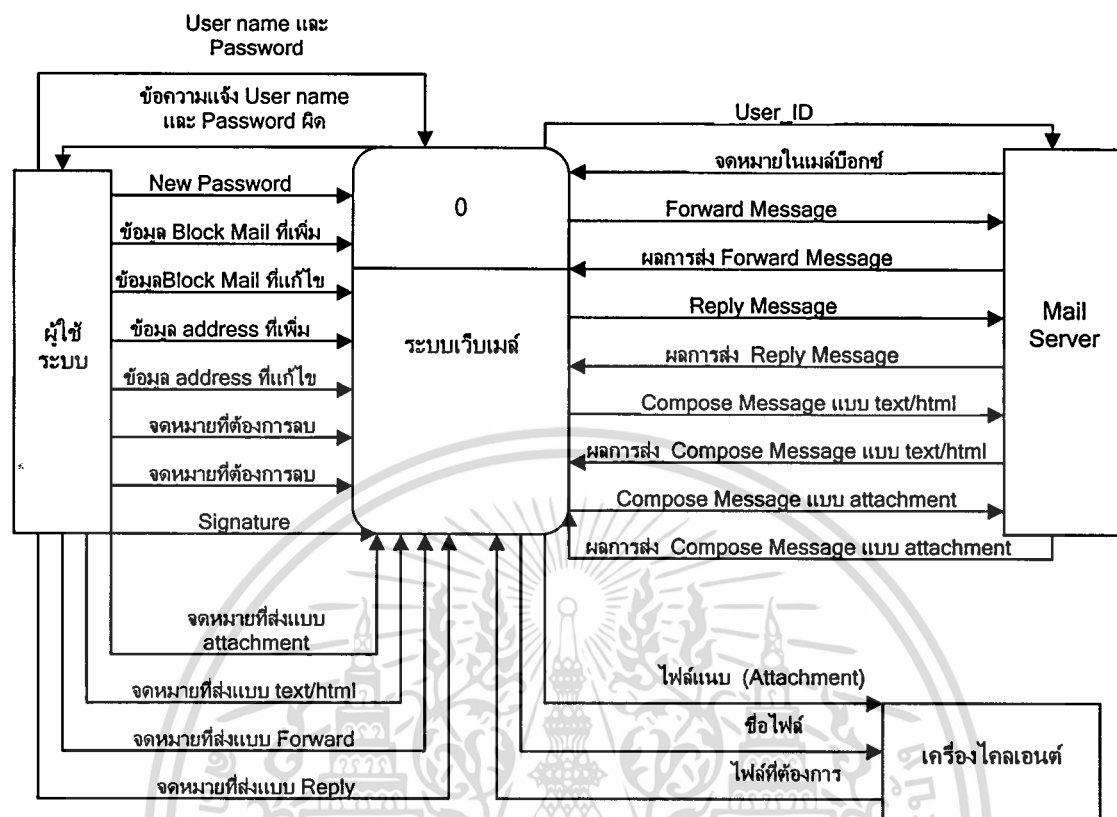
4.3 รายละเอียดกระบวนการทำงาน

การพัฒนาเว็บเบสอีเมลในโครงการพัฒนาระบบนี้ ได้ใช้หลักการ SDLC (System Development Life Cycle) ซึ่งจะทำการวิเคราะห์ปัญหาเป็นขั้นตอน และใช้เทคนิค DFD (Data Flow Diagram) เพื่อแสดงทิศทางการไหลของข้อมูล

จากการศึกษาและวิเคราะห์ระบบ สามารถนำมาเขียนเป็น Context Diagram ของระบบเว็บเมลล์ แสดงดังรูปที่ 4.2 ซึ่งสามารถอธิบายได้ดังนี้

ระบบเว็บเมลล์นี้มีข้อมูลที่ส่งผ่านระหว่างกัน คือ

- ระหว่างระบบกับผู้ใช้ระบบ
 - User Name และ Password
 - ข้อความแจ้ง User Name และ Password ผิด
- ระหว่างระบบกับ Mail Server
 - User ID
 - จดหมายในเมลล์บ็อกซ์
 - Forward Message
 - ผลการส่ง Forward Message
 - Reply Message
 - ผลการส่ง Reply Message
 - Compose Message แบบ text/html
 - ผลการส่ง Compose Message แบบ text/html
 - Compose Message แบบ attachment
 - ผลการส่ง Compose Message แบบ attachment
- ระหว่างระบบกับเครื่องไคลเอนต์
 - ชื่อไฟล์ที่ต้องการใช้แนบส่งไปกับอีเมล
 - ไฟล์ที่ต้องการแนบส่งไปกับอีเมล
 - ไฟล์ที่ดาวน์โหลดจากอีเมล

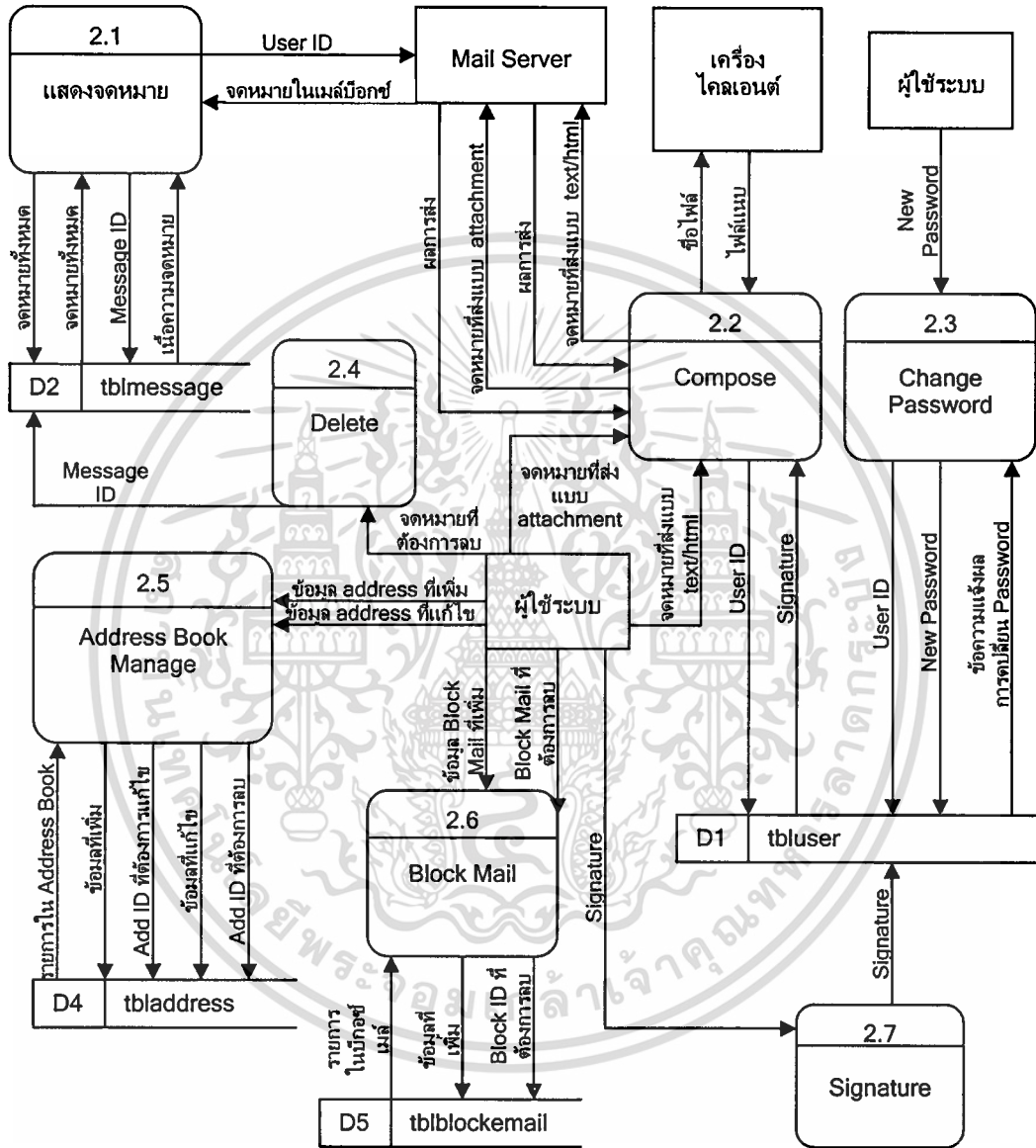


รูปที่ 4.2 Context Diagram ระบบเว็บเบสอีเมล

Level 1 Data Flow Diagram ระบบเว็บเบสอีเมล ประกอบไปด้วยขั้นตอนการทำงาน 3 ขั้นตอน ซึ่งแสดงดังรูปที่ 4.3 สามารถอธิบายได้ดังนี้

1. ผู้ใช้ระบบจะต้องทำการ Login เข้าสู่ระบบก่อน โดยการส่ง User Name และ Password ไปยังระบบ ระบบจะทำการตรวจสอบข้อมูลที่ได้รับว่า ผู้ใช้ระบบมีสิทธิในการเข้าใช้ระบบหรือไม่ ในกรณีที่ ในระบบไม่มี User Name และ Password ที่ผู้ใช้ระบบอยู่ใน Database ระบบก็จะแจ้งถึงความผิดพลาดของการ Login กลับไปยังผู้ใช้
2. จัดการแมล์บ็อกซ์ ระบบจะทำการแสดงจดหมายทั้งหมดในแมล์บ็อกซ์ โดยดึงจดหมายจากแมล์เซิร์ฟเวอร์มาไว้ในแฟ้ม tblmessage แล้วทำการลบจดหมายเหล่านั้นออกจากแมล์เซิร์ฟเวอร์ ในกรณีที่ผู้ใช้ระบบต้องการลบจดหมายหลายๆฉบับ ก็จะสามารถลบจดหมายได้จากขั้นตอนนี้ รวมถึงการส่ง Compose Message ทั้งแบบ text/html และแบบ attachment การค้นหา คำ การเพิ่มอีเมลแอดเดรส การเพิ่มรายการในบลิ๊อคแมล์ การเพิ่มคำลงท้ายจดหมาย และการเปลี่ยนรหัสผ่านในการเข้าระบบด้วย

Level 2 Data Flow Diagram ใน Process ที่ 2 เป็นการจัดการเกี่ยวกับเมลบ็อกซ์ ประกอบไปด้วยขั้นตอนการทำงาน ซึ่งแสดงดังรูป 4.4 สามารถอธิบายได้ดังนี้



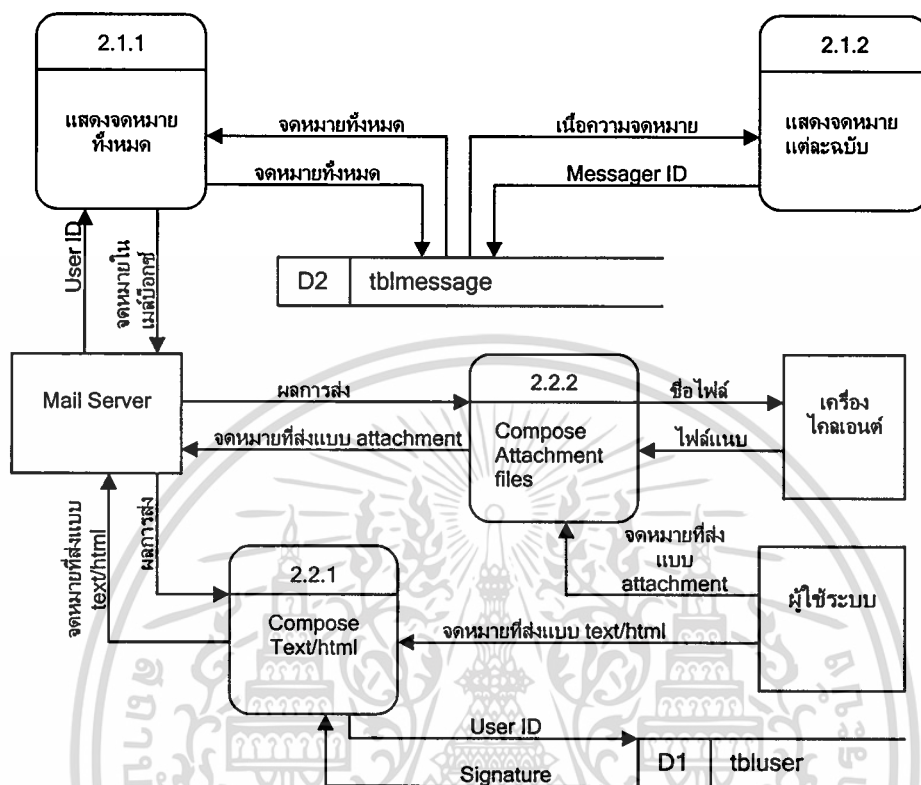
รูปที่ 4.4 Level 2 Data Flow Diagram ใน Process ที่ 2

- Process 2.1 เป็นการแสดงจดหมายในเมลบ็อกซ์ ซึ่งแบ่งออกเป็น 2 Processย่อย ได้แก่

- แสดงจดหมายทั้งหมดในเมลบ็อกซ์
- แสดงจดหมายแต่ละฉบับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งแสดงดังรูปที่ 4.5



รูปที่ 4.5 Level 3 Data Flow Diagram ใน Process ที่ 2

ในกรณีที่เป็นการแสดงจดหมายทั้งหมดในเมลบ็อกซ์ ระบบจะต้องส่ง User ID ไปยัง Mail Server แล้วทำการเก็บจดหมายทั้งหมดนั้นลงในแฟ้ม tblmessage ก่อน แล้วจึงจดหมายทั้งหมดจากแฟ้ม tblmessage ออกมาแสดง ถ้าต้องการดูในรายละเอียดของจดหมายแต่ละฉบับ ระบบก็จะต้องส่ง Message ID ไปยังแฟ้ม tblmessage เพิ่มอีกด้วย

- Process 2.2 เป็นการส่งจดหมายแบบ Compose Message คือ การส่งจดหมายฉบับใหม่ ซึ่งแบ่งออกเป็น 2 Processย่อย ซึ่งแสดงดังรูปที่ 4.5 ได้แก่

- Compose Message แบบ Text/html
- Compose Message แบบ Attachment files

ในกรณีที่เป็น Compose Message แบบ Text/html นั้น ถ้าต้องการคำลงท้ายจดหมาย (Signature) ก็สามารถทำได้โดยการส่ง User ID ไปยังแฟ้ม tbluser ส่วนการ Compose Message แบบ Attachment files จะต้องทำการส่งชื่อไฟล์ที่ต้องการไปยังเครื่องไคลเอนต์ จึงจะได้ไฟล์

แบบที่ต้องการส่ง นอกจากนี้จะมีการแสดงผลการส่ง เพื่อแจ้งให้ทราบว่าการส่งครั้งนั้นๆ เรียบร้อยและสมบูรณ์หรือไม่

- Process 2.3 เป็นการเปลี่ยนรหัสผ่านที่ใช้สำหรับเข้าระบบ โดยระบบจะต้องส่ง User ID และ New Password ไปยังแฟ้ม tbluser หลังจากนั้นจะมีการแจ้งผลการเปลี่ยนรหัสผ่านกลับมาให้ทราบ

- Process 2.4 เป็นการลบจดหมายที่หน้าอินบ็อกซ์ โดยผู้ใช้สามารถเลือกลบจดหมายได้มากกว่า 1 ฉบับ ซึ่งระบบจะทำการส่ง Message ID ที่ถูกเลือกไปยังแฟ้ม tblmessage เพื่อทำการเปลี่ยนสถานะของจดหมายนั้นๆ ให้เป็นสถานะถูกลบ

- Process 2.5 เป็นการจัดการ Address Book ไม่ว่าจะเป็นการเพิ่มอีเมลแอดเดรส, การแก้ไข หรือการลบ ซึ่งระบบจะทำการแสดงข้อมูลทั้งหมดที่มีอยู่ในแฟ้ม tbladdress ขึ้นมาแสดงบนหน้าจอ

- Process 2.6 เป็นการจัดการเกี่ยวกับการบล็อกอีเมล หรือโดเมนที่เราไม่ต้องการรับจดหมายหรือติดต่อด้วย โดยระบบจะทำการแสดงข้อมูลทั้งหมดที่มีอยู่ในแฟ้ม tblblockemail ขึ้นมาแสดงบนหน้าจอ ผู้ใช้สามารถที่จะเพิ่มหรือลบรายการที่มีอยู่ได้

- Process 2.7 เป็นการจัดการเกี่ยวกับค่าลงท้ายจดหมาย โดยเราสามารถที่จะเพิ่มหรือแก้ไขค่าลงท้ายจดหมายได้ข้อมูลที่ระบบดึงมาจากแฟ้ม tbluser ขึ้นมาแสดง

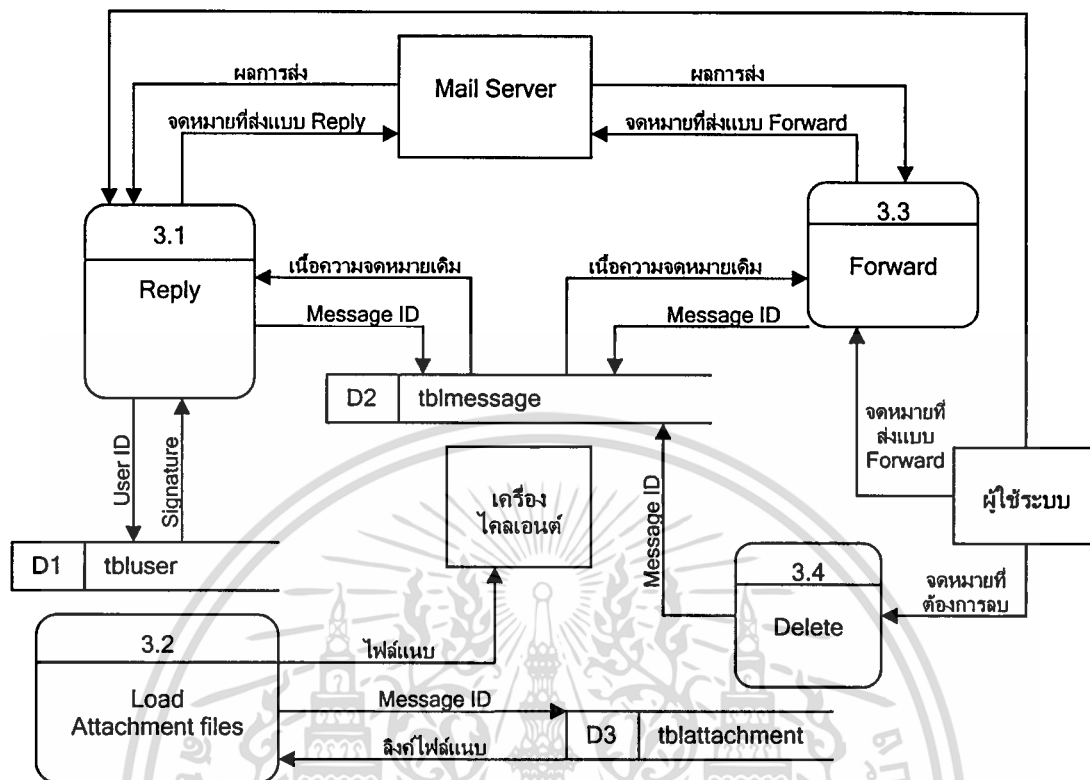
Level 2 Data Flow Diagram ใน Process ที่ 3 เป็นการจัดการเกี่ยวกับจดหมายในเมลบ็อกซ์แต่ละฉบับ ประกอบไปด้วยขั้นตอนการทำงาน ซึ่งแสดงดังรูป 4.6 สามารถอธิบายได้ดังนี้

- Process 3.1 เป็นการตอบกลับจดหมายกลับไปไปยังผู้ที่ส่งมา ซึ่งระบบจะต้องทำการส่ง Message ID ไปยังแฟ้ม tblmessage เพื่อนำเนื้อความเดิมในจดหมายออกมาแสดงด้วย ถ้าต้องการค่าลงท้ายจดหมาย ก็สามารถทำได้โดยการส่ง User ID ไปยังแฟ้ม tbluser

- Process 3.2 เป็นการดาวน์โหลดไฟล์แนบ (Attachment files) ที่มากับจดหมาย ลงเครื่องคอมพิวเตอร์ โดยระบบจะต้องส่ง Message ID ไปยังแฟ้ม tblattachment เพื่อจะนำไฟล์แนบที่ต้องการนั้นมาดาวน์โหลดเครื่อง

- Process 3.3 เป็นการส่งต่อจดหมายไปยังผู้อื่น ซึ่งระบบจะต้องทำการส่ง Message ID ไปยังแฟ้ม tblmessage เพื่อนำเนื้อความเดิมในจดหมายออกมาด้วย

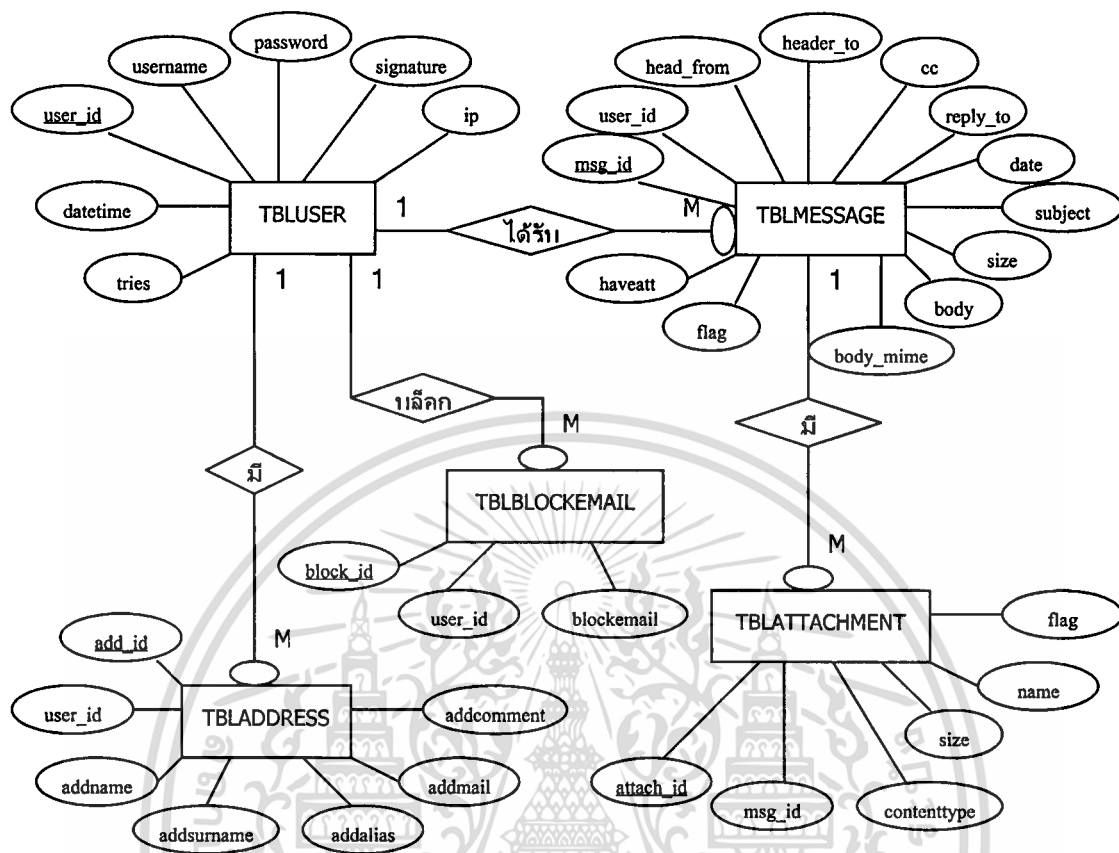
- Process 3.4 เป็นการลบจดหมายแต่ละฉบับที่กำลังเปิดอ่านอยู่ ซึ่งระบบจะทำการส่ง Message ID ที่ต้องการจะลบไปยังแฟ้ม tblmessage เพื่อทำการเปลี่ยนสถานะของจดหมายนั้นๆ ให้เป็นสถานะถูกลบ



รูปที่ 4.6 Level 2 Data Flow Diagram ใน Process ที่ 3

4.4 การออกแบบฐานข้อมูล

การออกแบบฐานข้อมูลขึ้นใช้งานในอาศัยแบบจำลองข้อมูล เพื่อนำเสนอรายละเอียดต่างๆ ในการออกแบบฐานข้อมูลของการพัฒนาระบบนี้ ได้ใช้แบบจำลอง ER Model (Entity Relationship Data Model) ดังแสดงรูปที่ 4.7



รูปที่ 4.7 Entity Relationship Diagram

ซึ่งประกอบด้วยตาราง 3 ตาราง ดังนี้

1. TBLUSER เก็บข้อมูลผู้ใช้
2. TBLMESSAGE เก็บเนื้อจดหมายและเฮดเดอร์ (header) อีเมลแต่ละฉบับ
3. TBLATTACHMENT เก็บข้อมูล Attachment เช่น ชื่อไฟล์ ชนิดไฟล์ แต่ไม่ได้เก็บข้อมูลไฟล์ Attachment เนื่องจากใช้วิธีเขียนไฟล์ลงในฮาร์ดดิสก์แทน เวลามีการดาวน์โหลด ก็ให้เปิดจากฮาร์ดดิสก์โดยตรง ไม่ผ่าน MySQL ทั้งนี้เพื่อลดภาระ MySQL Database Server นั้นเอง

ตารางที่ 4.1 รายละเอียดตาราง TBLUSER

Field	PK/ FK	Type	NULL	Description	Reference Table
<u>user_id</u>	PK	bigint(20)	NO	หมายเลขประจำของ user แต่ละคน	
username		varchar(45)	NO	ชื่อ username เป็นชื่อที่ใช้ Login	
password		varchar(30)	NO	รหัสผ่าน	
signature		text	YES	ข้อความลงท้ายจดหมาย	
tries		int(5)	YES	จำนวนครั้งที่พยายาม Login แต่ว่ารหัสผิด	
ip		varchar(20)	YES	ip ล่าสุดที่พยายาม Login เข้ามา แต่ว่ารหัสผิด	
datetime		varchar(28) ¹	YES	วันที่และเวลาที่ล่าสุดที่พยายาม Login เข้ามา แต่ว่า รหัสผิด	

ตารางที่ 4.2 รายละเอียดตาราง TBLMESSAGE

Field	PK/ FK	Type	NULL	Description	Reference Table
<u>msg_id</u>	PK	bigint(20)	NO	หมายเลข id สำหรับอีเมลล์ แต่ละฉบับ	
user_id	FK	bigint(20)	NO	หมายเลขประจำของ user แต่ละคน	TBLUSER
header_from		varchar(255)	YES	ข้อมูล header ว่าอีเมลล์ฉบับ นี้ใครส่งมา (From :)	
header_to		varchar(255)	YES	ข้อมูล header ว่าอีเมลล์ฉบับ นี้ส่งถึงใครบ้าง (To :)	

ตารางที่ 4.2 รายละเอียดตาราง TBLMESSAGE (ต่อ)

cc		varchar(255)	YES	ข้อมูล header ว่าอีเมลฉบับนี้ส่งมา CC ถึงใคร	
reply_to		varchar(255)	YES	ข้อมูล header ว่าอีเมลฉบับนี้ตอบกลับให้ใคร	
date		varchar(35)	YES	ข้อมูลวันที่รับส่งอีเมล	
subject		varchar(255)	YES	หัวเรื่องจดหมาย	
size		int(10)	YES	ขนาดของอีเมล (KB)	
body		text	YES	ข้อความจดหมาย	
body_mime		varchar(12)	YES	ชนิดของ message body เช่น เป็นแบบข้อความธรรมดา text/plain, หรือแบบ text/html	
flag		tinyint(2)	YES	สถานะของอีเมลฉบับนั้นๆ 0 มาใหม่ยังไม่ได้เปิดอ่าน (อยู่ใน Inbox) 1 เปิดอ่านแล้ว (ยังอยู่ใน Inbox) 2 ถูกลบไปแล้วและอยู่ในถังขยะ (Trash can) 3 เมล์ส่งออกไปแล้ว และถูกเก็บไว้ในตู้ Sent Box	
haveatt		tinyint(4)	YES	สถานะว่าอีเมลนี้มี Attachment หรือไม่	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 รายละเอียดตาราง TBLATTACHMENT

Field	PK/ FK	Type	NULL	Description	Reference Table
attach_id	PK	bigint(20)	NO	หมายเลข id สำหรับไฟล์ แนบแต่ละไฟล์	
msg_id	FK	bigint(20)	NO	หมายเลข id สำหรับอีเมล แต่ละฉบับ	TBLMESS AGE
contenttype		varchar(35)	NO	ชนิดของไฟล์ Attachment เช่น image/gif x-application ฯลฯ	
size		float	NO	ขนาดของ Attachment	
name		varchar(40)	NO	ชื่อไฟล์ Attachment ใช้ ตอนที่ผู้เปิดอีเมลแล้วดาวน์โหลด ชื่อนี้จะปรากฏที่ Dialog Box : Save As Filename	
flag		tinyint(4)	NO	สถานะของ Attachment	

ตารางที่ 4.4 รายละเอียดตาราง TBLADDRESS

Field	PK/ FK	Type	NULL	Description	Reference Table
add_id	PK	bigint(20)	NO	หมายเลข id สำหรับอีเมล แอดเดรสแต่ละรายการ	
user_id	FK	bigint(20)	NO	หมายเลขประจำของ user แต่ละคน	TBLUSER
addname		varchar(50)	YES	ชื่อจริง	
addsurname		varchar(50)	YES	นามสกุล	

ตารางที่ 4.4 รายละเอียดตาราง TBLADDRESS (ต่อ)

addalias		varchar(50)	YES	ชื่อเรียก	
addmail		varchar(50)	NO	อีเมลแอดเดรส	
addcomment		text	YES	รายละเอียดหรือหมายเหตุ	

ตารางที่ 4.5 รายละเอียดตาราง TBLBLOCKEMAIL

Field	PK/ FK	Type	NULL	Description	Reference Table
block_id	PK	bigint(20)	NO	หมายเลข id สำหรับอีเมลแอดเดรสหรือ โดเมนที่ถูกบล็อกแต่ละรายการ	
user_id	FK	bigint(20)	NO	หมายเลขประจำของ user แต่ละคน	TBLUSER
blockemail		varchar(40)	NO	รายการอีเมลแอดเดรสหรือ โดเมนที่ถูกบล็อก	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

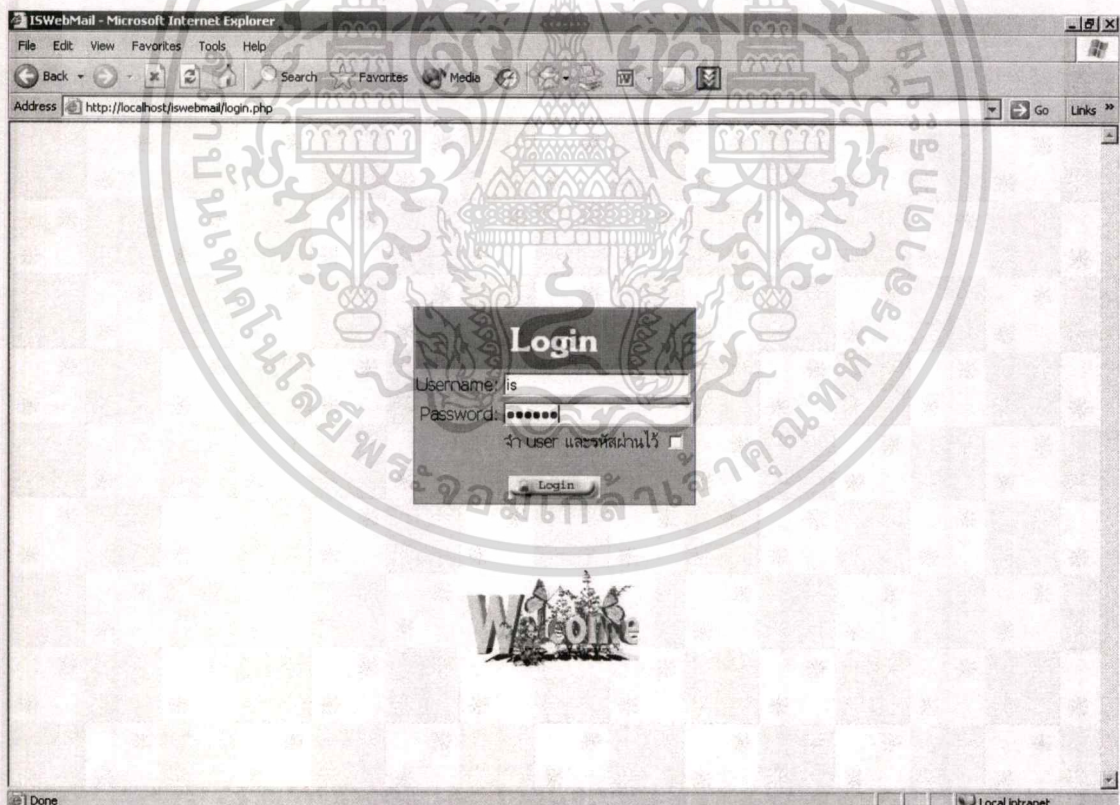
บทที่ 5

การพัฒนาระบบ

จากการพัฒนาระบบเว็บเบสอีเมลตามที่วิเคราะห์และออกแบบระบบนั้น สามารถแสดงการทำงานของระบบได้เป็นส่วนๆ ดังนี้

5.1 การเข้าใช้งานระบบ

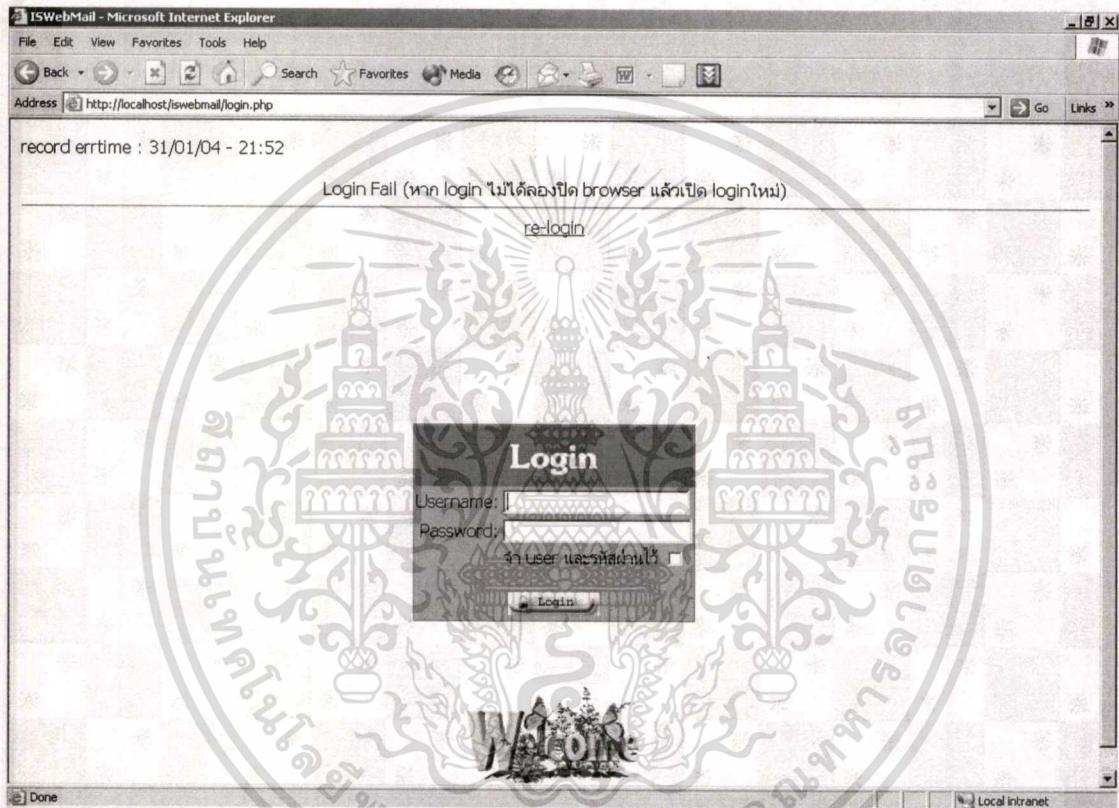
ก่อนที่จะเข้าใช้งานระบบเว็บเบสอีเมลได้นั้น ผู้ใช้ต้องทำการล็อกอินเข้าสู่ระบบก่อน โดยการใส่ Username และ Password ที่มีสิทธิในการใช้งานระบบ



รูปที่ 5.1 หน้าจอในการเข้าใช้ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าผู้ใช้ใส่ Username หรือ Password ผิด หรือผู้ใช้ไม่มีสิทธิ์ในการเข้าใช้งานระบบนั้น ระบบจะแสดงข้อผิดพลาดให้ผู้ใช้ทราบ รวมทั้งให้ผู้ใช้ทำการล็อกอินเพื่อเข้าระบบใหม่อีกครั้ง ซึ่งในการล็อกอินเพื่อเข้าระบบแต่ละครั้ง ระบบจะทำการบันทึกจำนวนครั้งที่พยายามล็อกอิน แต่เข้าไม่ได้เก็บไว้ เพื่อตรวจสอบว่ามีผู้พยายามลักลอบเข้ามาล็อกอินหรือไม่ โดยจะทำการเก็บ ip และวันที่ เวลาล่าสุดที่พยายามล็อกอินเข้ามา



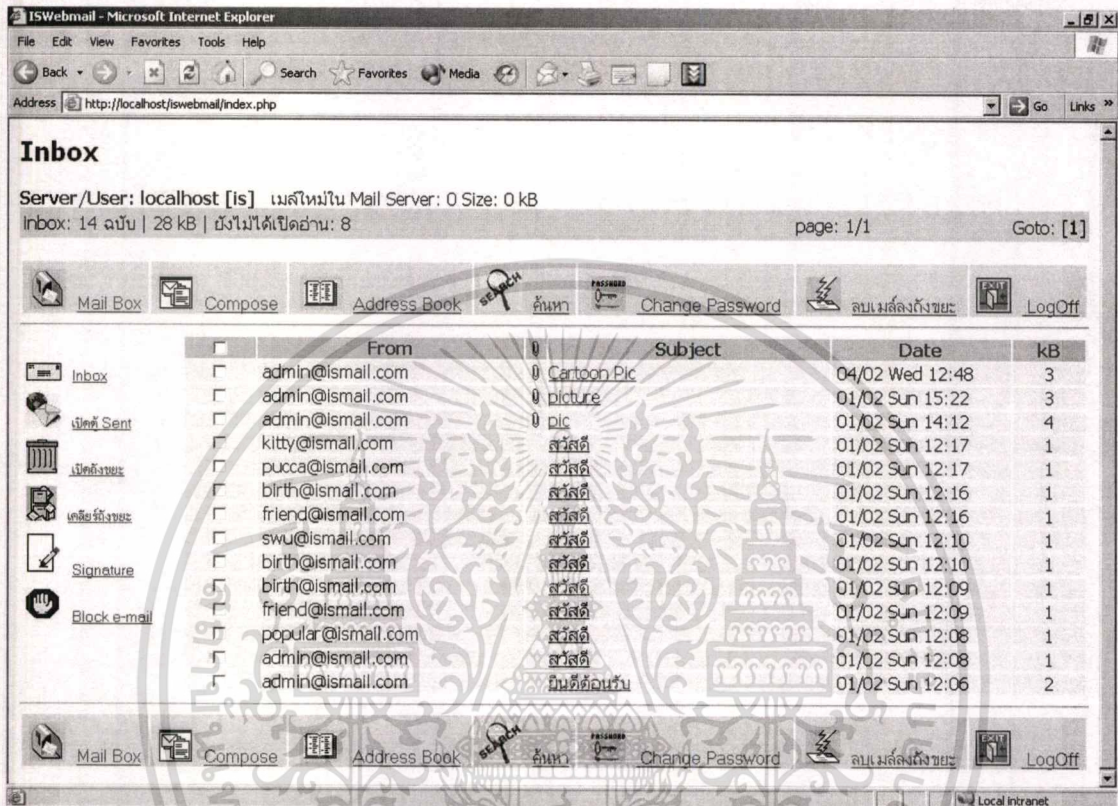
รูปที่ 5.2 หน้าจอแสดงข้อผิดพลาดในการเข้าใช้ระบบ

ถ้าผู้ใช้ล็อกอินผ่าน ระบบจะแสดงหน้าหลักของระบบ ซึ่งในหน้านี้อาจจะแสดงจดหมายทั้งหมดที่มีในเมลบ็อกซ์ และจะบอกถึงรายละเอียดต่างๆในเมลบ็อกซ์ของผู้ใช้ ไม่ว่าจะเป็นจำนวนจดหมายที่มีในเมลบ็อกซ์ ขนาดของเมลบ็อกซ์ จำนวนจดหมายที่ยังไม่ได้เปิดอ่าน โดยจะแสดงสีของจดหมายที่ยังไม่ได้เปิดอ่านให้แตกต่างจากจดหมายที่เปิดอ่านแล้ว

นอกจากนี้แล้ว ในหน้าหลักนี้จะประกอบไปด้วยเมนูต่างๆที่ใช้ในการใช้งานระบบ ไม่ว่าจะเป็นการส่งจดหมายถึงผู้อื่น การเพิ่มรายชื่อในสมุด Address Book การค้นหาคำหรือ keyword ที่ต้องการ การเปลี่ยน Password เป็นต้น โดยเมนูเหล่านี้จะมีทั้งด้านบนและด้านล่าง เพื่อให้ผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถใช้งานได้สะดวก ส่วนเมนูทางด้านซ้ายมือ จะเป็นเมนูย่อยของเมนูแม่บล็อกซ์ ซึ่งจะใช้ในการจัดการแม่บล็อกซ์ของผู้ใช้ระบบ




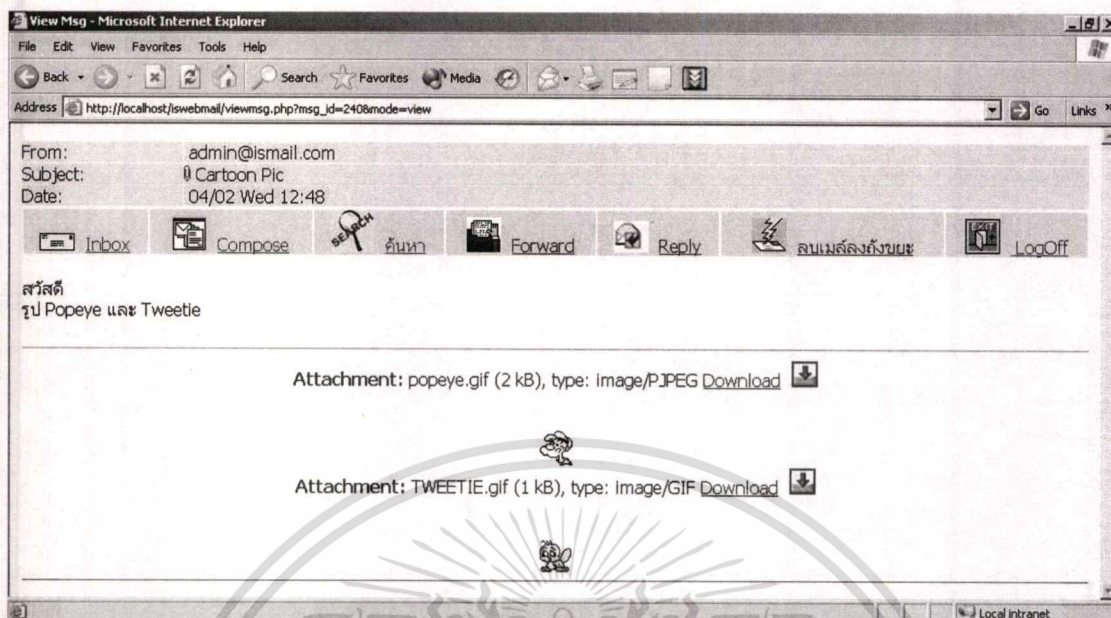
รูปที่ 5.3 หน้าจอแสดงหน้าหลักของระบบ

5.2 หน้าแม่บล็อกซ์

5.2.1 การอ่านจดหมาย

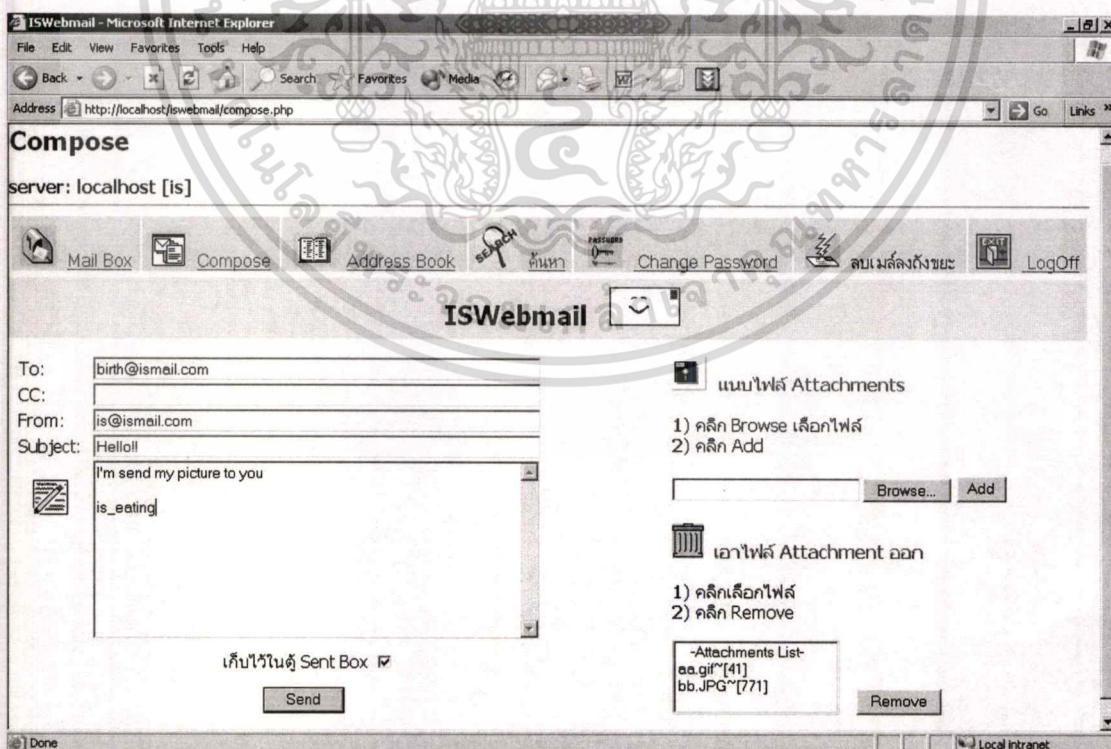
ผู้ใช้งานสามารถเลือกอ่านจดหมายที่ต้องการได้โดยการคลิก (click) ที่หัวข้อของจดหมาย (Subject) โดยระบบจะแสดงข้อความในจดหมาย รวมทั้งรูปภาพในกรณีที่มีการส่งรูปภาพแนบมาด้วย ซึ่งจดหมายฉบับที่มีรูปภาพแนบมาด้วย จะมีการแสดงสัญลักษณ์ 0 หน้าหัวข้อจดหมาย ดังรูป 5.3

เมื่อผู้ใช้งานคลิกเลือกจดหมายแล้ว ระบบจะแสดงหน้าจอดังรูป 5.4 ในกรณีที่มีรูปภาพแนบมาด้วย ผู้ใช้งานสามารถที่จะดาวน์โหลดรูปภาพที่ต้องการเก็บไว้ในเครื่องได้ทันที ที่ลิงค์ Download หรือคลิกที่รูปภาพ  ก็ได้



รูปที่ 5.4 หน้าจอแสดงข้อความในจดหมาย

5.2.2 การสร้างจดหมาย (Compose)



รูปที่ 5.5 หน้าจอแสดงการสร้างจดหมาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผู้ใช้สามารถสร้างจดหมายกลับเพื่อส่งไปยังผู้อื่นได้โดยการเลือกที่ Compose ซึ่งระบบจะแสดงหน้าจอ เพื่อใช้สำหรับเขียนข้อความต่างๆ รวมทั้งการส่งไฟล์แนบ โดยระบบจะทำการใส่เฮดเดอร์ (Header) ในช่อง From: ให้โดยอัตโนมัติ และในกรณีที่ผู้ใช้ระบบได้ทำการบันทึกข้อความลงท้ายจดหมายไว้ ระบบก็จะทำการใส่ข้อความลงท้ายจดหมายนั้นให้โดยอัตโนมัติในช่องที่ใช้สำหรับเขียนข้อความจดหมายด้วย

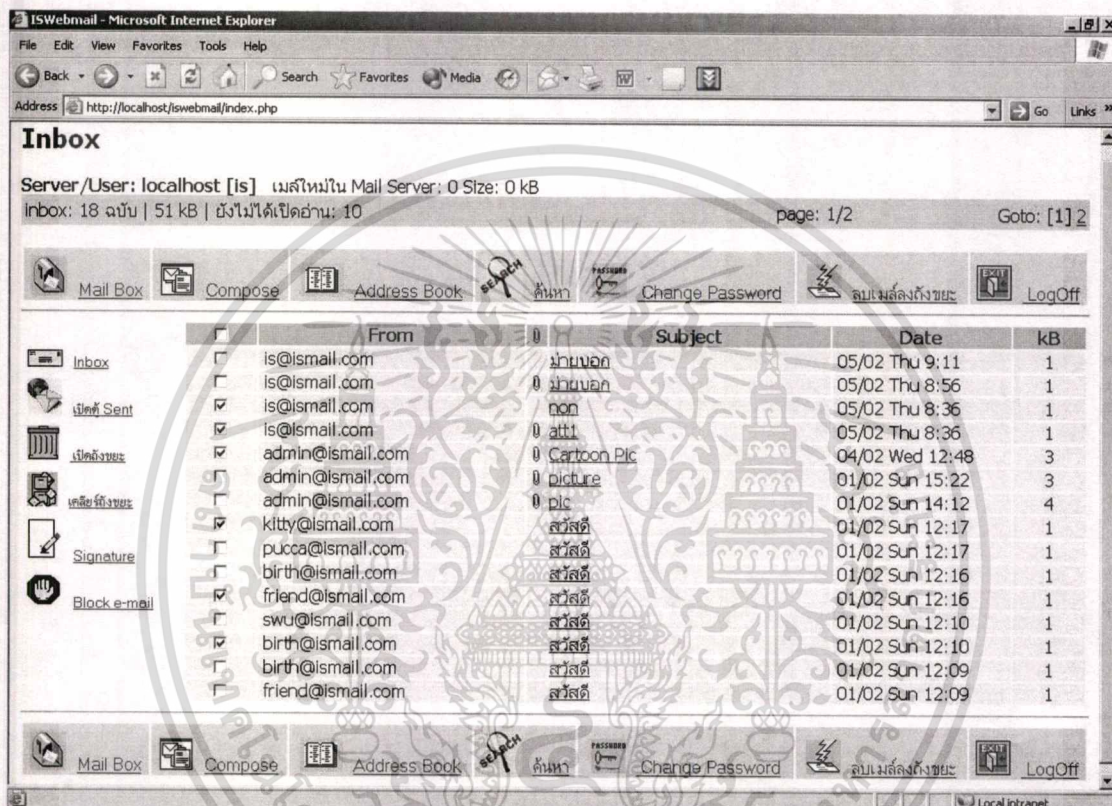
การแนบไฟล์ Attachment นั้น ผู้ใช้สามารถแนบไฟล์ได้สูงสุดถึง 5 ไฟล์ โดยในขั้นแรก ผู้ใช้ต้องการการคลิกที่ปุ่ม Browse เพื่อเลือกไฟล์ที่ต้องการ เมื่อเลือกแล้วก็ทำการคลิกที่ปุ่ม Add ทำเช่นนี้จนครบจำนวนไฟล์ที่ต้องการส่ง ระบบก็จะทำการแสดงชื่อไฟล์ ประเภทของไฟล์ที่ผู้ใช้ได้ทำการเพิ่มเข้าไป และแสดงขนาดของไฟล์นั้น ซึ่งเป็นขนาดไฟล์ที่เข้ารหัสแล้ว นอกจากนี้ ถ้าไฟล์ใดที่ผู้ใช้ได้ทำการเพิ่ม (Add) เข้าไปแล้ว แต่ต้องการลบเพื่อเลือกไฟล์ใหม่ ก็สามารถทำได้โดยการเลือกที่ชื่อไฟล์แนบที่ต้องการลบ ในช่อง Attachments List แล้วกดปุ่ม Remove ไฟล์ดังกล่าวนั้นก็จะถูกลบออกไป นอกจากนี้ ผู้ใช้ยังสามารถเลือกได้ว่าจะทำการเก็บบันทึกจดหมายที่ส่งนั้นไว้ในตู้ Sent Box หรือไม่ เมื่อผู้ใช้เขียนจดหมาย และทำการเพิ่มไฟล์ที่ต้องการเรียบร้อยแล้ว ก็ให้กดปุ่ม Send เพื่อทำการส่งจดหมายฉบับนั้น โดยระบบจะรายงานผลการส่งให้ผู้ใช้ทราบ ดังรูป 5.6



รูปที่ 5.6 หน้าจอแสดงผลการส่งจดหมาย

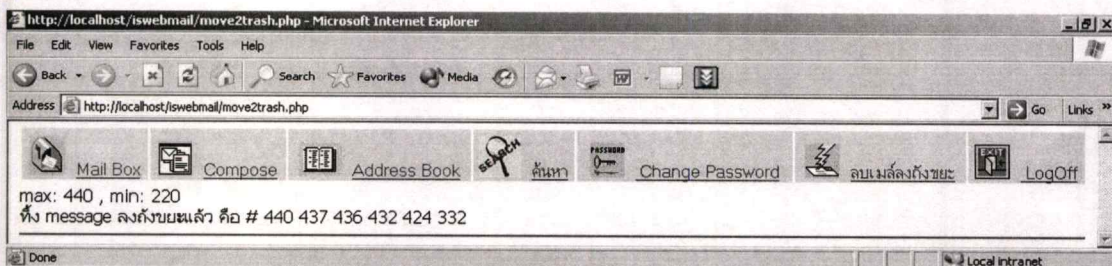
5.2.3 การลบจดหมาย

ในหน้าแม่ลี้บ็อกซ์นี้ ผู้ใช้สามารถลบจดหมายได้ที่ละหลายฉบับ โดยการเลือกจดหมายที่ต้องการลบที่ checkbox ในกรณีที่ต้องการลบจดหมายทั้งหมด ผู้ใช้ระบบสามารถทำได้โดยเลือกที่ checkbox ที่ส่วนบนสุดของจดหมาย หลังจากนั้นก็เลือกที่ “ลบแม่ลี้ลงถังขยะ”



รูปที่ 5.7 หน้าจอแสดงการลบจดหมายในหน้าแม่ลี้บ็อกซ์

โดยระบบจะรายงานผลการลบจดหมาย ดังรูป 5.8

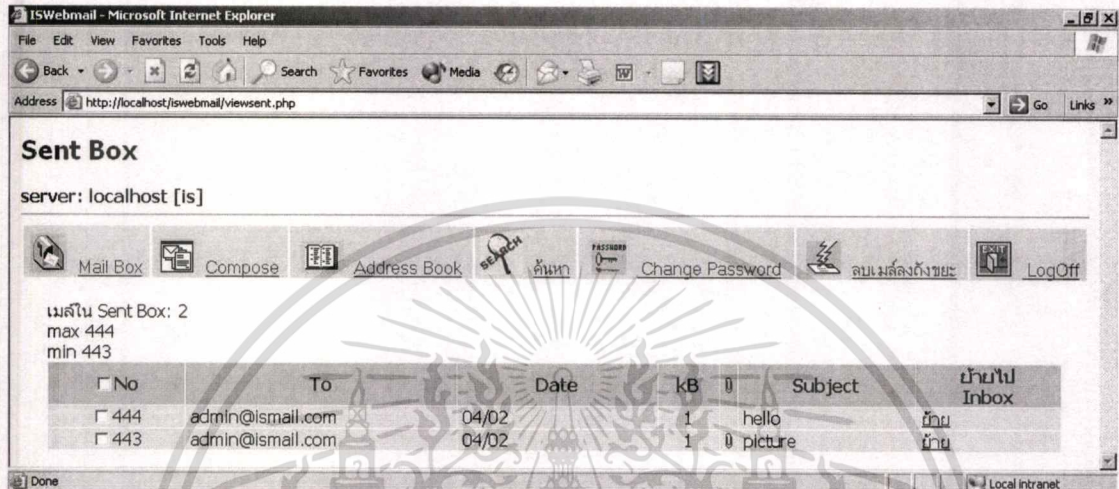


รูปที่ 5.8 หน้าจอแสดงผลการลบจดหมายในหน้าแม่ลี้บ็อกซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.4 ตู้ Sent Box

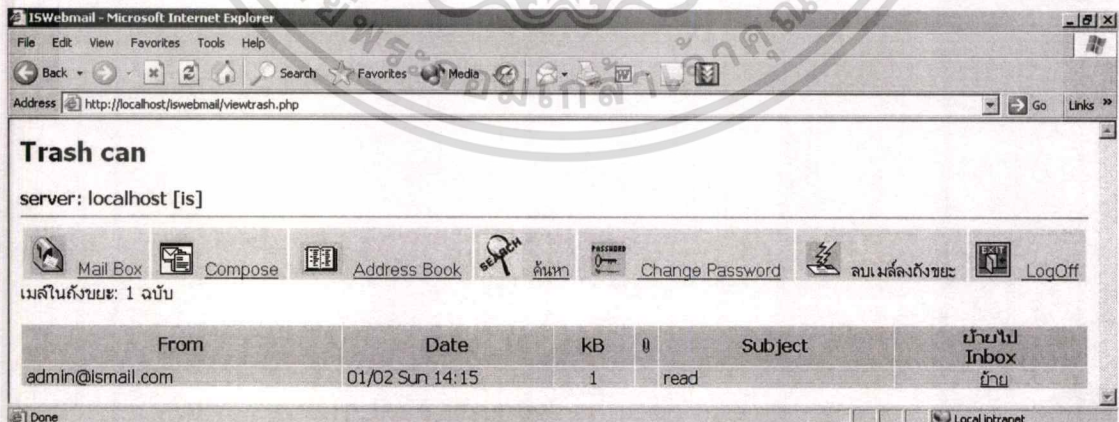
ผู้ใช้ระบบสามารถเลือกดูจดหมายที่ได้ทำการส่งไปแล้วได้ ว่าได้ทำการส่งจดหมายไปยังใครบ้าง และยังสามารถที่จะย้ายจดหมายในตู้ sent box นั้น ไปไว้ในตู้ inbox ได้อีกด้วย



รูปที่ 5.9 หน้าจอแสดงจดหมายในตู้ Sent Box

5.2.5 ถังขยะ

ผู้ใช้ระบบสามารถเลือกดูจดหมายที่ได้ทำการลบไปแล้วได้ ว่าได้ลบจดหมายฉบับใดบ้าง และยังสามารถที่จะย้ายจดหมายในถังขยะนั้น ไปไว้ในตู้ inbox ได้อีกด้วย

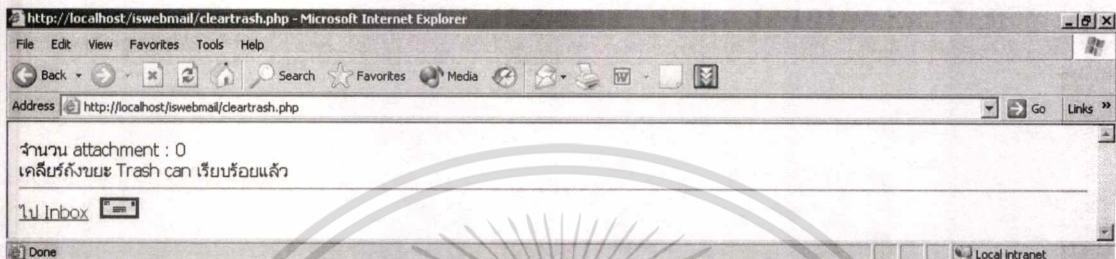


รูปที่ 5.10 หน้าจอแสดงจดหมายในถังขยะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

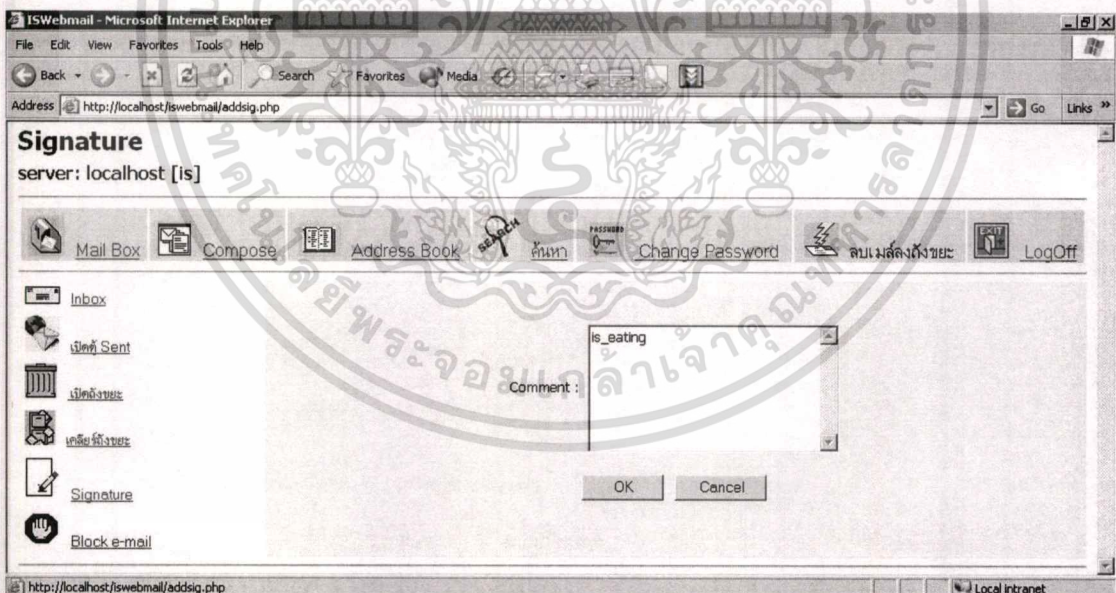
5.2.6 เคลียร์ถังขยะ

ผู้ใช้งานระบบสามารถทำการลบจดหมายทั้งหมดในขยะได้เลยโดยไม่ต้องรอนครบกำหนดระยะเวลาที่ระบบได้กำหนดไว้ เพื่อจะได้เพิ่มเนื้อที่ในระบบเมลล์บ็อกซ์ โดยทำการเลือกที่ “เคลียร์ถังขยะ” และระบบจะทำการรายงานผลการลบจดหมายในขยะ ดังรูป 5.11



รูปที่ 5.11 หน้าจอแสดงผลการลบจดหมายทั้งหมดในถังขยะ

5.2.7 การสร้างข้อความลงท้ายจดหมาย (Signature)



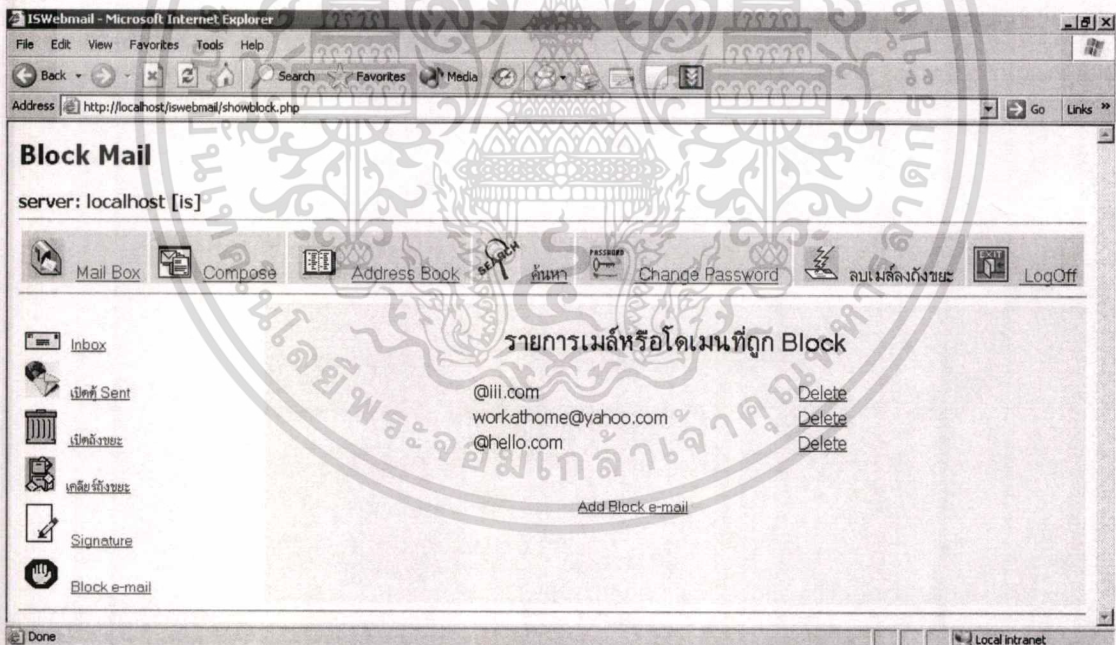
รูปที่ 5.12 หน้าจอการสร้างและแก้ไขข้อความลงท้ายจดหมาย

ผู้ใช้งานระบบสามารถสร้างข้อความลงท้ายจดหมายได้ตามที่ต้องการ และสามารถแก้ไขข้อความนั้นได้ทุกเวลาที่ต้องการอีกด้วย โดยการเลือกที่ “Signature” ถ้าผู้ใช้งานระบบได้เคยทำการสร้างข้อความจดหมายไว้แล้ว ระบบก็จะทำการแสดงข้อความลงท้ายจดหมายเดิมขึ้นมา ผู้ใช้เอกสารเป็นเอกสารที่ส่งวนเวียนสำหรับการเขียนเพื่อการรศกษาเท่านั้น ไม่อนุญาตเห็นไปใช้ประโยชน์ในการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบก็สามารถแก้ไขข้อความนั้นได้และทำการเลือกปุ่ม “OK” เพื่อทำการบันทึกข้อความลงท้ายจดหมายที่ได้แก้ไขนั้น ในกรณีที่ผู้ใช้ระบบยังไม่เคยทำการสร้างข้อความลงท้ายจดหมาย ระบบก็จะแสดง textarea ว่างๆ เพื่อให้ผู้ใช้ทำการใส่ข้อความลงท้ายจดหมายที่ต้องการ

5.2.8 การบล็อกเมล (Block Mail)

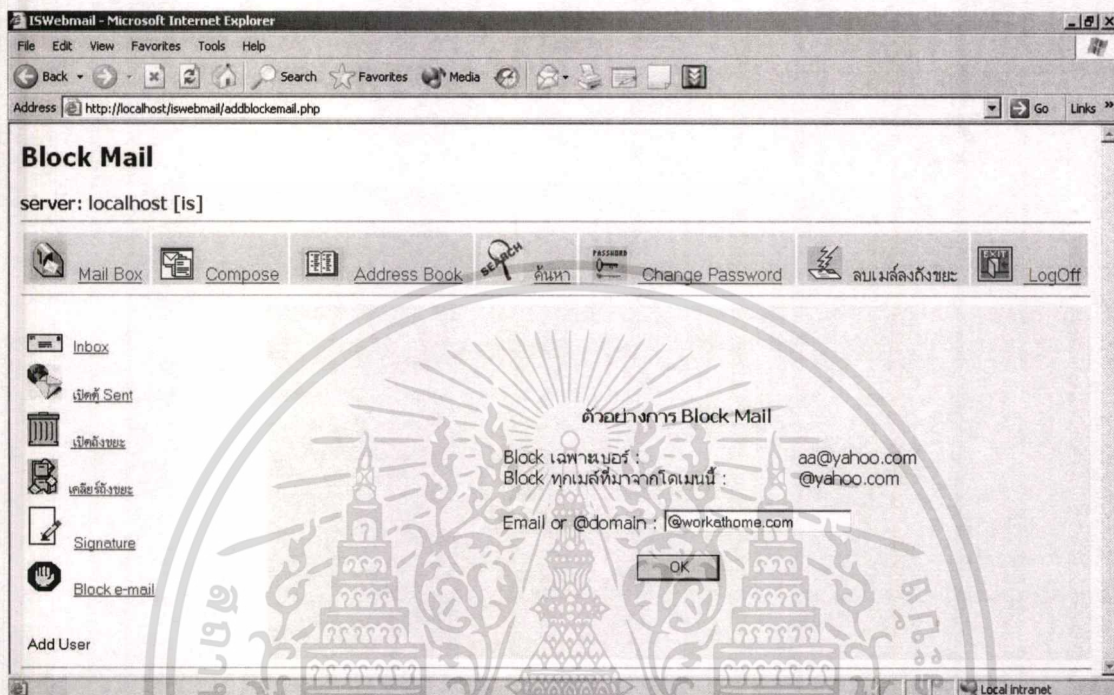
ผู้ใช้ระบบสามารถที่จะปฏิเสธจดหมายบางฉบับได้โดยการทำการบล็อกอีเมลหรือโดเมนที่ไม่ต้องการนั้น โดยเมื่อมีจดหมายเข้ามาในเมลบ็อกซ์ของผู้ใช้ ระบบก็จะทำการตรวจสอบก่อนว่าจดหมายฉบับนั้นตรงกับรายการที่ถูกบล็อกในบล็อกเมลหรือไม่ ซึ่งถ้าตรง ระบบก็จะทำการลบจดหมายฉบับนั้นออกจากฐานข้อมูลทันที โดยถ้าผู้ใช้ต้องการดูรายการที่ถูกบล็อกในบล็อกเมล ให้เลือกที่ “Block e-mail” ระบบจะแสดงรายการอีเมลหรือโดเมนที่ผู้ใช้ได้ทำการเพิ่มไว้แล้วขึ้นมา ดังรูป 5.13 ซึ่งในหน้านี้ผู้ใช้สามารถที่จะลบรายการที่ไม่ต้องการบล็อกแล้วได้ทันที โดยคลิกที่ “Delete”



รูปที่ 5.13 หน้าจอแสดงรายการอีเมลหรือโดเมนที่ถูกบล็อก

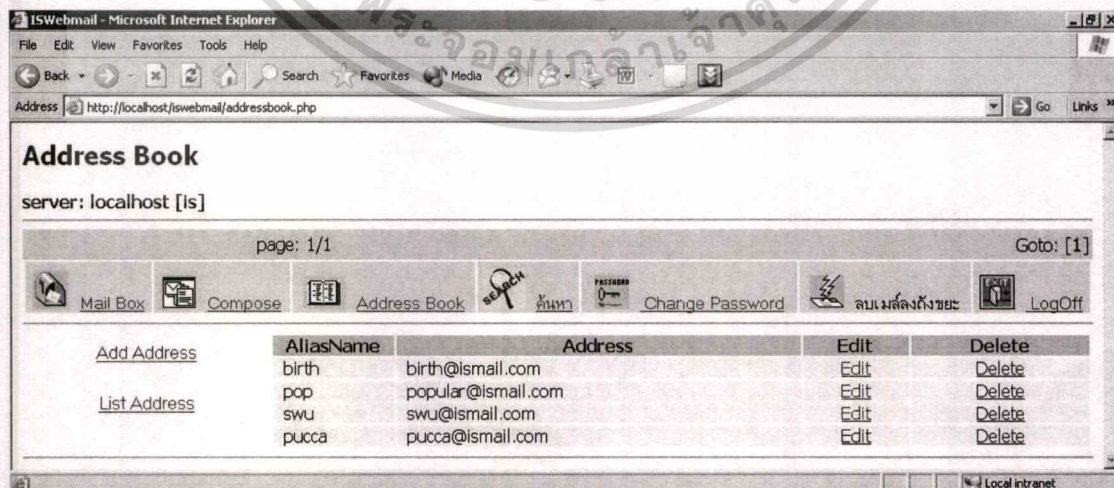
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนการเพิ่มรายการบล็อกเมลนั้นสามารถทำได้โดยเลือกที่ “Add Block e-mail” ระบบก็จะแสดงหน้าจอให้ผู้ใส่อีเมลหรือโดเมนที่ต้องการบล็อก ดังรูป 5.14



รูปที่ 5.14 หน้าจอการเพิ่มรายการอีเมลหรือโดเมนที่ต้องการบล็อก

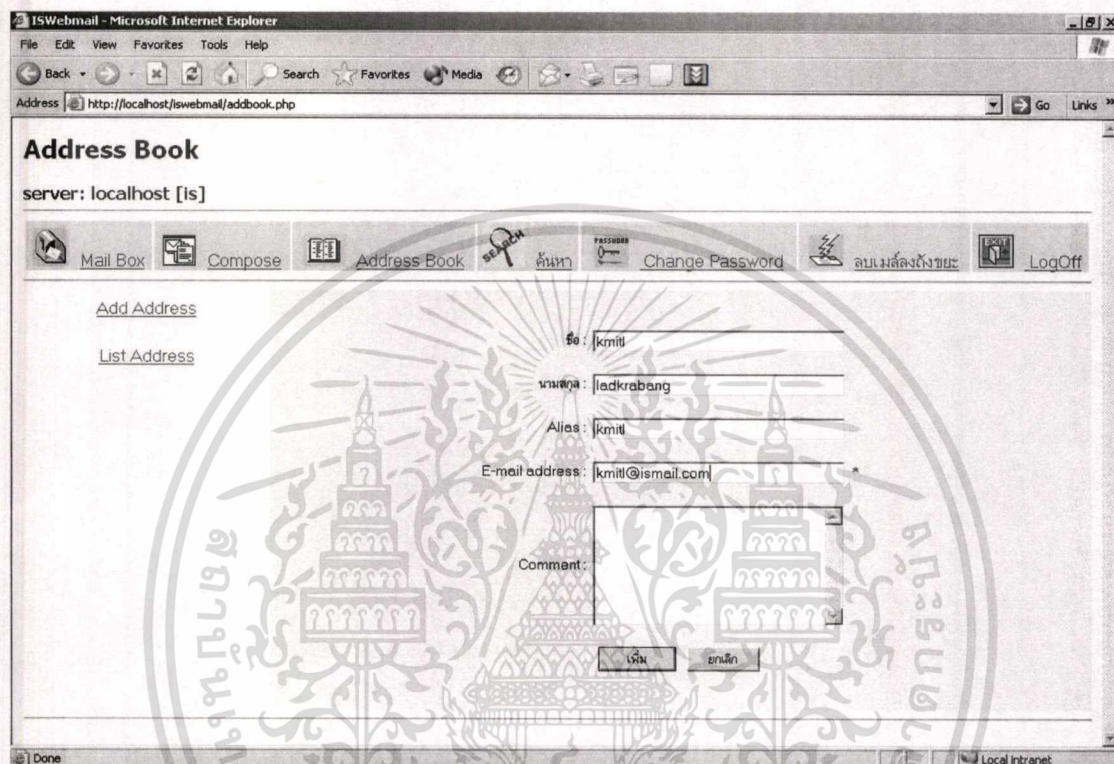
5.2.9 สมุดบันทึกอีเมลแอดเดรส (Address Book)



รูปที่ 5.15 หน้าจอแสดงสมุดบันทึกอีเมลแอดเดรส

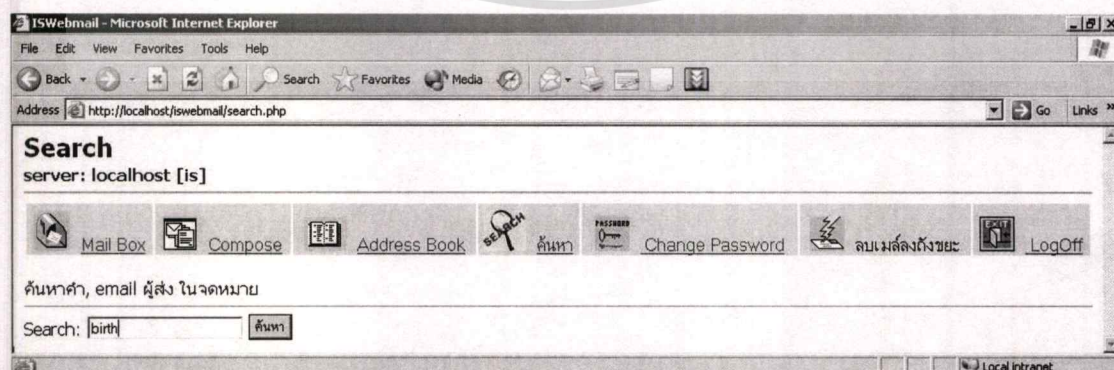
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อผู้ใช้เลือกที่ “Address Book” ระบบจะแสดงรายการอีเมลที่มีในสมุดขึ้นมาทั้งหมด ซึ่งผู้ใช้ระบบสามารถที่จะแก้ไขหรือลบข้อมูลในแต่ละรายการได้ ส่วนการเพิ่มรายการเข้าไปในสมุดบันทึกอีเมลแอดเดรสนี้ ทำได้โดยการเลือกที่ “Add Address”



รูปที่ 5.16 หน้าจอการเพิ่มรายการในสมุดบันทึกอีเมลแอดเดรส

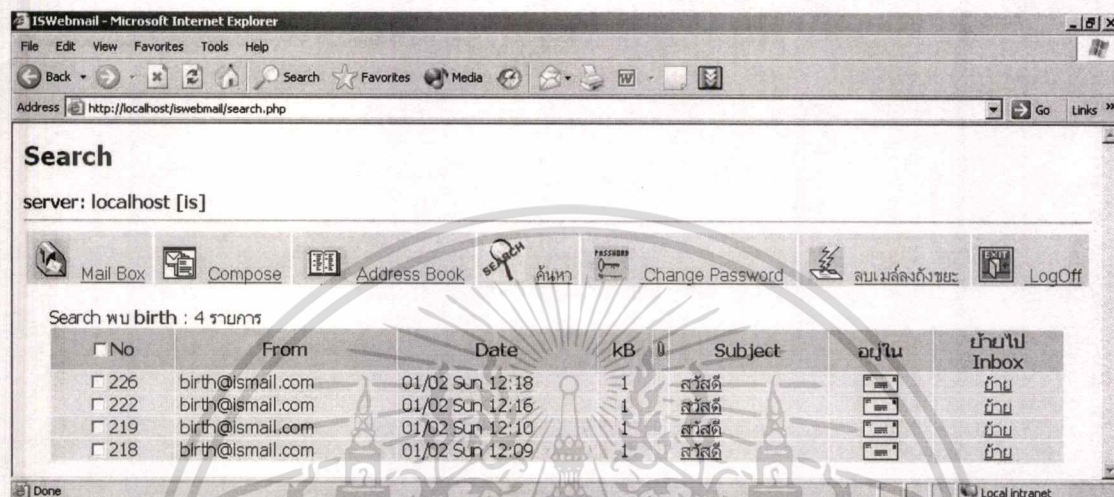
5.2.10 การค้นหาคำในอีเมล



รูปที่ 5.17 หน้าจอการค้นหาคำ

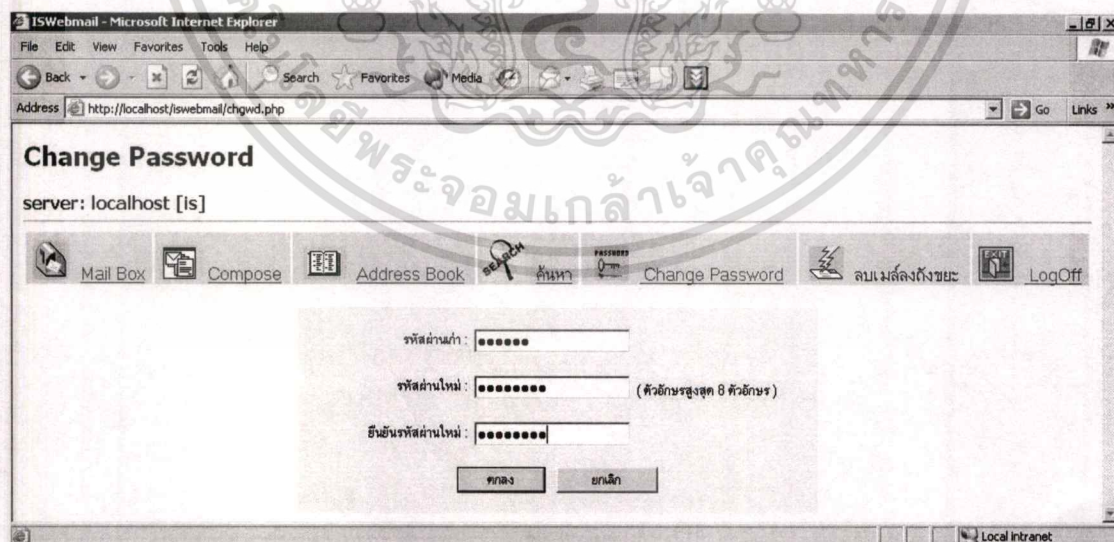
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผู้ใช้ระบบสามารถค้นหาคำต่างๆในอีเมลได้ เช่น การค้นหาอีเมลแอดเดรสผู้ส่ง การค้นหาข้อความต่างๆ เป็นต้น โดยเลือกที่เมนู “ค้นหา” ระบบจะแสดงหน้าจอเพื่อรับคำที่ใช้ต้องการค้นหา หลังจากนั้นก็จะแสดงผลการค้นหาที่ได้ดังรูป 5.18



รูปที่ 5.18 หน้าจอแสดงผลการค้นหา

5.2.11 การเปลี่ยนรหัสผ่าน (Change Password)



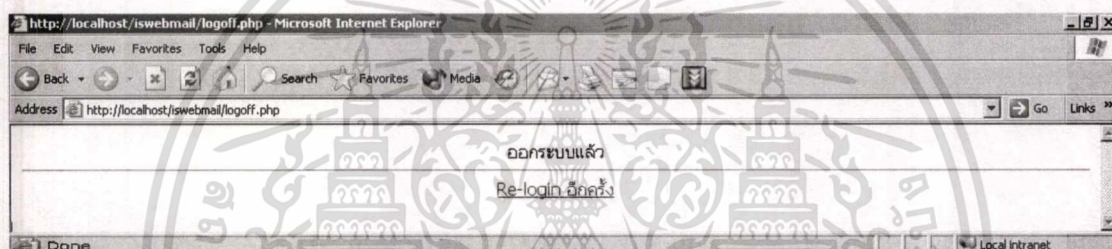
รูปที่ 5.19 หน้าจอการเปลี่ยนรหัสผ่าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการเปลี่ยนรหัสผ่านนั้น ผู้ใช้ต้องใส่รหัสผ่านเดิมก่อน แล้วจึงใส่รหัสผ่านใหม่ที่ต้องการ โดยถ้ารหัสผ่านเดิมที่ผู้ใช้ใส่ ไม่ตรงกับฐานข้อมูล ผู้ใช้ก็ไม่สามารถที่จะเปลี่ยนรหัสผ่านได้ และถ้ารหัสผ่านใหม่ที่ผู้ใช้ใส่ไม่ต้องกันในช่อง “รหัสผ่านใหม่” และ “ยืนยันรหัสผ่านใหม่” ระบบก็จะไม่ทำการเปลี่ยนรหัสผ่านให้ตามที่ผู้ใช้ต้องการ

5.2.12 การออกจากระบบ (LogOff)

เมื่อผู้ใช้ต้องการออกจากระบบ ก็ทำได้โดยการเลือกที่เมนู LogOff ระบบก็จะทำการยกเลิกการเก็บคุกกี้ผู้ใช้และยกเลิก session และแสดงหน้าจอแจ้งให้ผู้ใช้ทราบว่าได้ออกจากระบบแล้ว รวมทั้งมีลิงค์ที่ใช้สำหรับการล็อกอินใหม่ด้วย



รูปที่ 5.20 หน้าจอรายงานการออกจากระบบ

5.3 การจัดการจดหมายแต่ละฉบับ

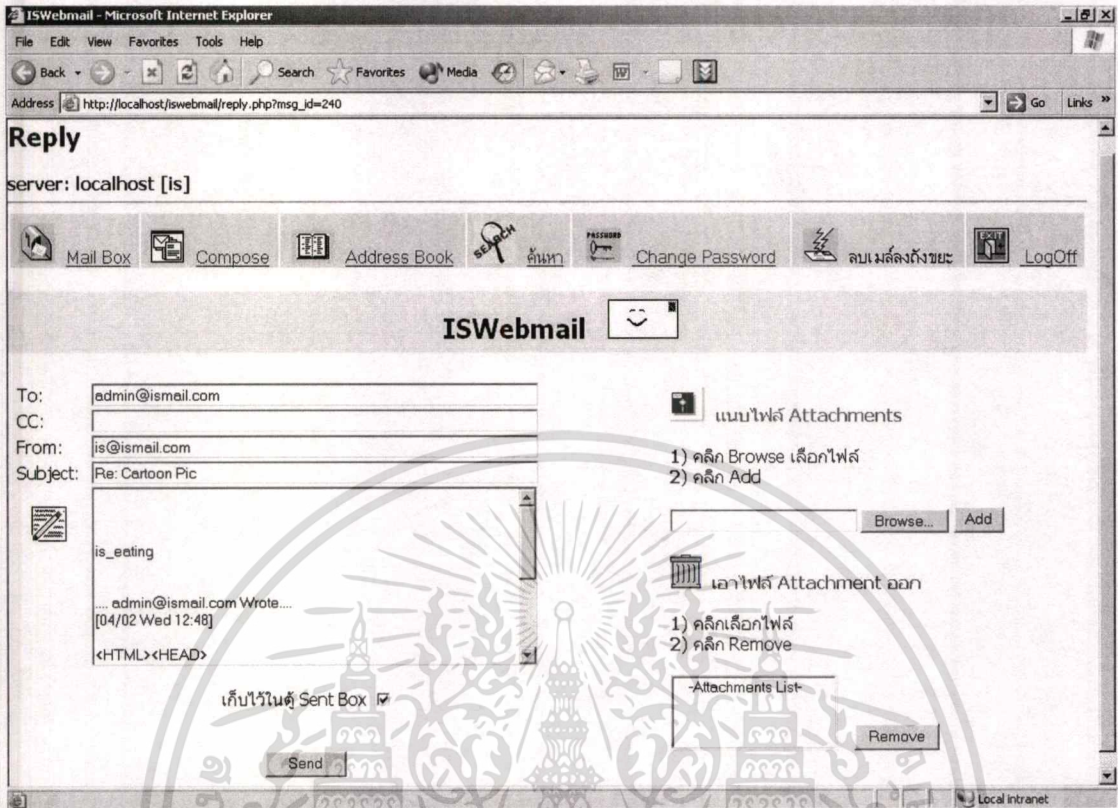
5.3.1 การตอบกลับจดหมาย (Reply)

เมื่อผู้ใช้ได้อ่านจดหมายแล้ว ผู้ใช้สามารถตอบจดหมายกลับไปยังผู้ส่งได้โดยการเลือกที่ Reply ซึ่งระบบจะทำการ

- ใส่เฮดเดอร์ (Header) ซึ่งประกอบไปด้วย To: From: Subject: ให้โดยอัตโนมัติ
- ใส่ข้อความเดิมที่ผู้ส่งได้ส่งมาในจดหมายในส่วนล่าง
- ในกรณีที่ผู้ใช้ระบบได้ทำการบันทึกข้อความลงท้ายจดหมายไว้ ระบบก็จะทำการใส่

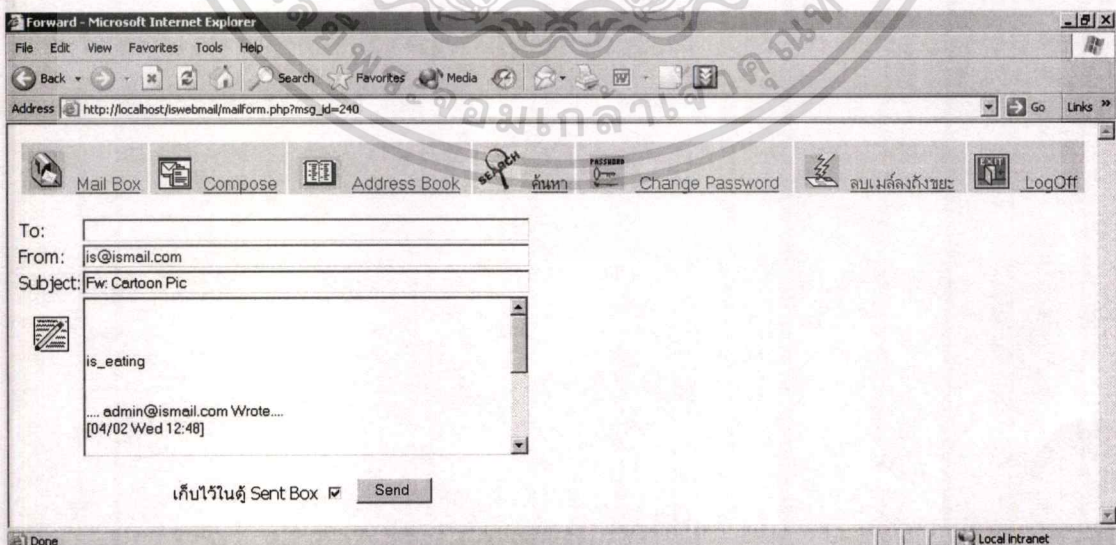
ข้อความลงท้ายจดหมายนั้นให้โดยอัตโนมัติ

นอกจากนี้ ผู้ใช้ยังสามารถเลือกได้ว่าจะทำการเก็บบันทึกจดหมายที่ส่งนั้นไว้ในตู้ Sent Box หรือไม่ และยังสามารถทำการแนบไฟล์ที่ต้องการส่งไปได้อีกด้วย



รูปที่ 5.21 หน้าจอแสดงการตอบกลับจดหมาย

5.3.2 การส่งต่อจดหมาย (Forward)



รูปที่ 5.22 หน้าจอแสดงการส่งต่อจดหมาย

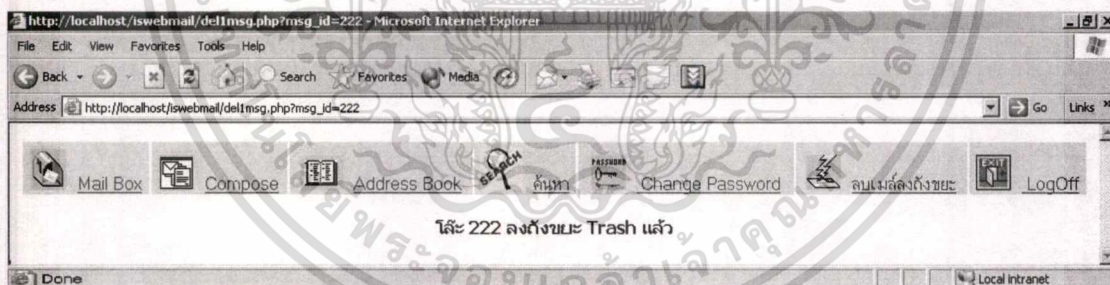
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อผู้ใช้ได้อ่านจดหมายแล้ว ผู้ใช้สามารถส่งต่อจดหมายนี้ไปยังผู้อื่นได้ โดยการเลือกที่ Forward ซึ่งระบบจะทำการ

- ใส่เฮดเดอร์ (Header) ซึ่งประกอบไปด้วย From: และ Subject: ให้โดยอัตโนมัติ
 - ใส่ข้อความเดิมที่ผู้ส่งได้ส่งมาในจดหมายในส่วนล่าง
 - ในกรณีที่ผู้ใช้ระบบได้ทำการบันทึกข้อความลงท้ายจดหมายไว้ ระบบก็จะทำการใส่ข้อความลงท้ายจดหมายนั้นให้โดยอัตโนมัติ
 - ในกรณีที่มีไฟล์แนบ ระบบก็จะทำการส่งไฟล์แนบให้ด้วย
- นอกจากนี้ ผู้ใช้ยังสามารถเลือกได้ว่าจะทำการเก็บบันทึกจดหมายที่ส่งนั้นไว้ในตู้ Sent Box หรือไม่

5.3.3 การลบจดหมายแต่ละฉบับที่อ่าน

ขณะที่ผู้ใช้อ่านจดหมายแต่ละฉบับ ผู้ใช้สามารถลบจดหมายที่อ่านนี้ได้ โดยการเลือกที่ “ลบเมลล์ลงถังขยะ” ซึ่งระบบจะทำการย้ายจดหมายฉบับนั้น ไปไว้ที่ถังขยะ ทำให้ผู้ใช้ระบบสามารถที่จะย้ายจดหมายฉบับนั้นกลับมาในอินบ็อกซ์ได้ และระบบก็จะแจ้งผลการลบจดหมายให้ผู้ใช้ทราบ ดังรูป



รูปที่ 5.23 หน้าจอแสดงผลการลบจดหมาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

บทสรุป

6.1 สรุปผลโครงการศึกษาระบบพิเศษ

จากการพัฒนาระบบงาน ผลที่ได้รับเป็นไปตามขอบเขตที่วางไว้ อันประกอบไปด้วยหน้าที่การทำงาน ดังต่อไปนี้

- ระบบสามารถใช้ในการรับและส่งอีเมลภายในองค์กรได้ ไม่ว่าจะเป็นการสร้างจดหมายขึ้นมาใหม่ การตอบกลับจดหมาย หรือการส่งต่อจดหมาย
- สามารถกำหนดรายละเอียด รวมถึงฟังก์ชันการทำงานในระบบได้ตามความต้องการขององค์กร เช่น การบล็อกอีเมลแอดเดรสหรือโดเมนที่ไม่ต้องการ การเปลี่ยนรหัสผ่านของผู้ใช้ระบบได้อย่างสะดวกและรวดเร็ว การเปลี่ยนแปลงหรือแก้ไขข้อความลงท้ายจดหมาย สมุดบันทึกอีเมลแอดเดรส การจัดเก็บจดหมายที่ได้ส่งไปแล้วไว้ในตู้ sent box
- ระบบสามารถจัดการอีเมลในเมลบ็อกซ์ได้อย่างมีประสิทธิภาพ เช่น ผู้ใช้สามารถที่จะลบจดหมายหลายฉบับหรือทีละฉบับที่กำลังอ่านได้ การย้ายจดหมายที่ได้ลบแล้วจากถังขยะไปไว้ใน Inbox ได้ การย้ายจดหมายจากตู้ sent box ไปไว้ใน Inbox ได้
- ระบบสามารถใช้ในการรับและส่งอีเมลภายนอกองค์กรได้

6.2 ปัญหาและข้อจำกัดที่พบ

- เนื่องจากเมลเซิร์ฟเวอร์ที่ใช้ในการพัฒนา ซึ่งก็คือ Agro Mail Server ผู้ใช้ไม่สามารถเช็คเมลได้ในทันที โดยผู้รับจะได้รับเมลเมื่อเวลาผ่านไปแล้ว 5 วินาที
- เนื่องจากเครื่องเซิร์ฟเวอร์ไม่สามารถใช้ฟังก์ชัน IMAP ได้ เพราะไม่ได้ทำการติดตั้งไว้ในกรณีที่เครื่องเซิร์ฟเวอร์ใช้ระบบปฏิบัติการ Windows จึงต้องทำการติดตั้ง Extension IMAP (php_imap.dll) เอง โดยจะทำงานได้ดีก็ต่อเมื่อใช้ร่วมกับ OmniHTTPD เท่านั้น ซึ่งถ้าทำการติดตั้งกับ Windows IIS หรือ PWS แล้วผลลัพธ์ที่ได้จะไม่สมบูรณ์

6.3 ประโยชน์ที่ได้จากการศึกษาและพัฒนาระบบ

- สามารถนำระบบที่พัฒนามาประยุกต์ใช้ในองค์กรได้อย่างมีประสิทธิภาพ
- ทำให้การทำงานภายในองค์กรมีประสิทธิภาพมากขึ้น ไม่ว่าจะเป็นการลดขั้นตอนการทำงาน การลดระยะเวลา รวมทั้งการบริหารทรัพยากรต่างๆ
- ทำให้องค์กรสามารถดำเนินงานติดต่อกับองค์กรหรือธุรกิจภายนอกได้อย่างมีประสิทธิภาพ ซึ่งจะช่วยส่งเสริมให้องค์กรมีความก้าวหน้ามากขึ้น
- ได้รับความรู้และความเข้าใจในเรื่องการทำงานของระบบเว็บเบสอีเมลล์
- สามารถนำความรู้ที่ได้ไปพัฒนาระบบงานอื่นๆ ในอนาคตได้อย่างมีประสิทธิภาพ

6.4 ข้อเสนอแนะ

- เนื่องจากระบบเว็บเบสอีเมลล์ที่พัฒนานี้ พัฒนามาจากการศึกษาการทำงานขององค์กรแห่งหนึ่ง ระบบนี้จึงอาจจะยังไม่ครอบคลุมการทำงานทั้งหมด หากมีการนำไปใช้งานกับองค์กรอื่นๆ อาจจะต้องมีการพัฒนาโปรแกรมเพิ่มเติม เพื่อให้สอดคล้องกับเงื่อนไขและความต้องการสำหรับองค์กรนั้นๆ
- เนื่องจากเครื่องเซิร์ฟเวอร์ที่รัน PHP ได้นั้น บางเครื่องอาจจะไม่สามารถใช้ฟังก์ชัน IMAP ได้ เพราะไม่ได้ทำการติดตั้งไว้ เพราะฉะนั้น ในการใช้งานระบบจะต้องทำการดาวน์โหลดโมดูล หรือ Extension php_imap.dll เข้ามาติดตั้งในเครื่องเซิร์ฟเวอร์ โดยเก็บไว้ในไดเรกทอรีเดียวกับไฟล์สคริป PHP ที่มีการเรียกใช้ฟังก์ชัน IMAP

บรรณานุกรม

กิตติ ภักดีวัฒน์กุล และคณะ. 2545. **PHP ฉบับโปรแกรมเมอร์**. กรุงเทพฯ: เคทีพี คอมพ์ แอนด์ คอนซัลท์.

กอบเกียรติ สระอุบล. 2545. **สร้างเว็บเพจด้วย PHP ฉบับประยุกต์**. กรุงเทพฯ: บี อี แอนด์ ซี.

ไพศาล โมลิตสฤตมมงคล. 2544. **พัฒนา Web Database ด้วย PHP**. กรุงเทพฯ: ไทยเจริญการพิมพ์.

ภาณุพันธุ์ จันทรา. 2546. **บทความคอมพิวเตอร์**. [Online]. Available:

<http://www.e-books2003.com/article-com-4.html>.

สมจิตร อัจฉินทร์ และงามนิช อัจฉินทร์. 2542. **ระบบฐานข้อมูล**. พิมพ์ครั้งที่ 4. กรุงเทพฯ: ศูนย์หนังสือมหาวิทยาลัยขอนแก่น.

สุวัฒน์ ปุณณชัยยะ และคณะ. 2543. **เปิดโลกของ TCP/IP และโปรโตคอลของอินเทอร์เน็ต**. พิมพ์ครั้งที่ 1. กรุงเทพฯ: ด้านสุทธการพิมพ์.

อำไพ พรประเสริฐสกุล. 2543. **การวิเคราะห์และออกแบบระบบ**. พิมพ์ครั้งที่ 4. กรุงเทพฯ: ซีเอ็ดดูเคชั่น.

Bielefeldt, Erik. 2002. **Webmail Design Document**. [Online]. Available:

<http://eveander.com/arsdigita/doc/design/webmail>.

Chau, K. 2001. **Implementation of Simple Mail Transfer Protocol (SMTP) for Mail System in OceanStore**. U.C. Berkely: Computer Science Division.

Dreamscape. 2001. **DREAMSCAPE WEBMAIL**. [Online]. Available:

<http://webmail.dreamscape.com/howto.html>.

Information Services of the Art Institute of Chicago. 2003. **Webmail Documentation**. [Online].

Available: <https://webmail.artic.edu/guide/>.

Information Systems Service. 2002. **Introductory Exercises Using the Webmail Program**

IMP. n.p. : University of Leeds.

OnlineCMS. 2003. **WebMail vs Email Clients**. [Online]. Available:

<http://www.w-media.com/index.cfm>.

Share360 Support. 2001. **WebMail Overview**. [Online]. Available:

<http://cybozu.com/supportcenter/s360v2/manuals/manualbyvolume.cbml>.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Tung, C. and Tsai, S. 2001. **Tuning Webmail Performance - The Design and Implementation of Open Webmail**. Taiwan: National Cheng Kung University.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

ชื่อผู้เขียน : นางสาวศุภรัตน์ รัตนจิตบรรจง

วันเดือนปีเกิด : 9 เมษายน พ.ศ. 2523

สถานที่เกิด : กรุงเทพมหานคร

วุฒิการศึกษาระดับปริญญาตรี : วิทยาศาสตร์บัณฑิต

สถานที่สำเร็จการศึกษา : มหาวิทยาลัยศรีนครินทรวิโรฒ ประสานมิตร

ปีที่สำเร็จการศึกษา : พ.ศ. 2545



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้