

ระบบจดหมายอิเล็กทรอนิกส์ผ่าน WAP  
A WAP-Based E-Mail System

โดย

นายทศพร เลิศบุรโณทัย

รหัส 44067464



\*H002106\*

อาจารย์ที่ปรึกษา

ผศ.ดร. นพพร โชติกกำจร

วัน เดือน ปี.....	02/10/06
เลขทะเบียน.....	
เลขเรียกหนังสือ.....	อพ. ๗๒๓๗ ร. ๒๕๔๖
"ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล."	

รายงานนี้เป็นส่วนหนึ่งของวิชา โครงการพัฒนาระบบงาน  
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ  
ภาคเรียนที่ 2 ปีการศึกษา 2546  
คณะเทคโนโลยีสารสนเทศ  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ชื่อหัวข้อ	ระบบจดหมายอิเล็กทรอนิกส์ผ่าน WAP
นักศึกษา	นายทศพร เลิศบุร โณทัย
อาจารย์ที่ปรึกษา	ผศ.ดร. นพพร โชติกกำธร
ระดับการศึกษา	วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2546

### บทคัดย่อ

การใช้งานจดหมายอิเล็กทรอนิกส์บนโทรศัพท์เคลื่อนที่ในปัจจุบันมีข้อจำกัดคือ Domain Name ที่ใช้งานเป็นผู้ให้บริการ ทำให้ผู้ใช้บริการต้องมี E-Mail Address เพิ่มขึ้น จึงเกิดแนวคิดว่าจะมีระบบที่ใช้ Domain Name ที่มีอยู่แล้วเพื่อลดความสับสนในการติดต่อสื่อสาร และสามารถแสดงผลให้เหมาะสมกับโทรศัพท์เคลื่อนที่ การพัฒนาระบบจดหมายอิเล็กทรอนิกส์ผ่าน WAP เป็นการนำ Mail Server และ Domain Name ที่ใช้อยู่ในองค์กรมาให้บริการ ช่วยลดความสับสนในการติดต่อสื่อสาร พัฒนาโดยติดตั้ง WAP Server เพื่อติดต่อกับ Mail Server ด้วย PHP Script และนำมาแสดงผลบนโทรศัพท์เคลื่อนที่ โดยแสดงผลเพียงบางส่วนของจดหมายก่อนเพื่อลดค่าใช้จ่ายการใช้งาน WAP สามารถอ่าน, เขียน, ตอบ และส่งต่อจดหมายอิเล็กทรอนิกส์จากโทรศัพท์เคลื่อนที่ที่สามารถแสดงผล XHTML ได้ แต่ยังไม่สามารถแนบเพิ่มข้อมูลหรือเปิดเพิ่มข้อมูลที่แนบมากับจดหมายได้เนื่องจากข้อจำกัดของอุปกรณ์

<b>Title</b>	A WAP-Based E-Mail System
<b>Student</b>	Mr. Tosaporn Lerdburanotai
<b>Advisor</b>	Asst. Prof. Dr. Nopporn Chotikakamthorn
<b>Level of study</b>	Master of Science in Information Technology
<b>Major</b>	Information Science
<b>Academic Year</b>	2003

### Abstract

Currently the usage of e-mail on mobile phones has a limitation because the domain name is registered under the service provider's name. As a result, users must have an additional e-mail address. There is a concept to use existing domain name to reduce complication in communications and display results suitable for mobile phones. A WAP-Based E-Mail System is thus developed so that existing company's mail server and domain name can be used. The WAP server is connected to a mail server through PHP script, before displaying the results on the screen of a mobile phone. To save cost, only a part of the e-mail will be displayed on the first page. This application supports reading, writing, replying, and forwarding e-mail from the mobile phones' XHTML browser. However, this application does not support attached files or attachment opening functions due to limitation of equipment functionality.

## กิตติกรรมประกาศ

การพัฒนาระบบจดหมายอิเล็กทรอนิกส์ผ่าน WAP ที่ได้พัฒนาขึ้นมาได้รับการสนับสนุนจากเป็นอย่างดี ทั้งให้คำปรึกษา คำแนะนำและให้ความรู้เกี่ยวกับระบบงาน ผู้จัดทำจึงใคร่ขอขอบคุณบุคคล ดังนี้

- อ. โอฟาร วงศ์วิรัตน์ อาจารย์ที่ปรึกษา ภาคเรียนที่ 1 ปีการศึกษา 2546 ที่กรุณาให้คำปรึกษาและคำแนะนำในการพัฒนาระบบงานนี้



นายทศพร เลิศบุรโณทัย

ผู้จัดทำ

# สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ .....	IV
สารบัญตาราง.....	VI
สารบัญรูป.....	VII
บทที่	
1 บทนำ.....	1
1.1 วัตถุประสงค์.....	1
1.2 ขั้นตอนการพัฒนาระบบ.....	2
1.3 ขอบเขตการดำเนินการ.....	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	3
2 WAP และ PHP.....	4
2.1 Wireless Application Protocol (WAP).....	4
2.2 WAP Protocol Stack.....	5
2.3 WAP Stack กับ Internet Stack.....	11
2.4 ขั้นตอนการทำงานบนเครือข่าย WAP.....	12
2.5 WAP Gateway.....	13
2.6 WAP 2.0 Specification.....	14
2.7 เทคโนโลยีและมาตรฐานใหม่ใน WAP 2.0.....	14
2.8 XHTML.....	15
2.9 XHTML Mobile Profile.....	15
2.10 ความแตกต่างระหว่าง XHTML กับ WML และ cHTML.....	16

## สารบัญ (ต่อ)

หน้า

2.11 PHP Script.....	17
2.12 ฟังก์ชันที่เกี่ยวกับ IMAP ใน PHP.....	18
3 การวิเคราะห์ ออกแบบและพัฒนาระบบ.....	20
3.1 หลักการพัฒนา WAP Application .....	20
3.2 ความต้องการของระบบ .....	20
3.3 สถาปัตยกรรมของระบบ .....	20
3.4 Logical Component Diagram.....	22
3.5 การเตรียม Server.....	22
3.6 การเตรียม Client .....	24
3.7 Data Flow Diagram.....	25
3.8 ฐานข้อมูลที่ใช้ในระบบ.....	29
3.9 ภาษาที่ใช้ในการพัฒนาระบบและฟังก์ชันต่างๆ.....	29
3.10 การแสดงผลบน Nokia Mobile Browser Simulator .....	31
4 สรุปการพัฒนา .....	42
4.1 สรุปผลการดำเนินงาน.....	42
4.2 ข้อเสนอแนะ.....	42
บรรณานุกรม .....	44
ภาคผนวก	
ภาคผนวก ก การติดตั้ง Web Server และ Mail Server บน Linux Redhat 7.3 .....	45
ประวัติผู้เขียน.....	49

# สารบัญตาราง

หน้า

ตารางที่

3.1 ตารางเก็บข้อมูลการแสดงผลของผู้ใช้งาน ..... 29



# สารบัญรูป

หน้า

รูปที่

2.1 WAP Protocol Stack.....	6
2.2 ส่วนประกอบของ WAE .....	6
2.3 ประเภทของ WSP ตามลักษณะการเชื่อมต่อ .....	8
2.4 WAP Stack กับ Internet Stack .....	11
2.5 การทำงานบนเครือข่าย WAP.....	12
3.1 การทำงานทั่วไปของ WAP Mail .....	21
3.2 สถาปัตยกรรมของการพัฒนาระบบ .....	21
3.3 Logical Component Diagram .....	22
3.4 Data Flow Diagram Level 0 .....	25
3.5 Data Flow Diagram Level 1 ของ Process 1 .....	26
3.6 Data Flow Diagram Level 1 ของ Process 2 .....	27
3.7 Data Flow Diagram Level 1 ของ Process 5 .....	28
3.8 ฟังก์ชันต่างๆ ในระบบ .....	30
3.9 การเข้าระบบ .....	32
3.10 หน้าจอ Login.....	32
3.11 รายงานข้อผิดพลาด .....	32
3.12 จุดหมายที่อยู่ในกล่องจดหมาย.....	33
3.13 การปรับแต่งระบบ .....	34
3.14 ข้อความในจดหมายที่มีแฟ้มแนบมาด้วย.....	35
3.15 ข้อความในจดหมายแบบย่อและแบบเต็ม .....	36
3.16 การสร้างจดหมายใหม่และทดสอบการรับด้วย Outlook Express .....	37
3.17 จดหมายที่สร้างจาก Outlook Express และเปิดอ่านจาก Browser.....	38
3.18 การตอบจดหมายและเปิดอ่านจาก Outlook Express.....	39
3.19 การส่งต่อจดหมายพร้อมเพิ่มข้อมูลที่แนบมา.....	40

## สารบัญญรูป (ต่อ)

หน้า

รูปที่

3.20 การเปิดอ่านจดหมายที่แนบเพิ่มข้อมูลมาด้วย .....	41
---	----



# บทที่ 1

## บทนำ

ในปัจจุบันมีผู้ใช้จดหมายอิเล็กทรอนิกส์ในการติดต่อสื่อสารเป็นจำนวนมาก ทั้งเรื่องส่วนตัวและทางธุรกิจ แต่ในการใช้งานมีข้อจำกัดคือต้องมีเครื่องคอมพิวเตอร์ที่เชื่อมต่อกับอินเทอร์เน็ต และติดตั้งซอฟต์แวร์สำหรับรับ-ส่งจดหมายอิเล็กทรอนิกส์ ดังนั้นหากมีความจำเป็นที่จะต้องใช้งานจดหมายอิเล็กทรอนิกส์ในการติดต่อทางด้านธุรกิจอย่างรีบด่วนภายนอกสำนักงานจึงไม่ได้รับความสะดวกรวดเร็ว ประกอบกับความแพร่หลายและความสามารถของโทรศัพท์เคลื่อนที่ที่มีเพิ่มขึ้นทั้งความสามารถในการเชื่อมต่อกับอินเทอร์เน็ตที่มีความเร็วสูงขึ้น การแสดงผลที่สวยงามขึ้น มีความละเอียดมากขึ้น จึงสามารถนำมาใช้เพิ่มช่องทางและความคล่องตัวในการใช้งานรับ-ส่งจดหมายอิเล็กทรอนิกส์ได้ในระดับหนึ่ง เพื่อช่วยอำนวยความสะดวกสำหรับการติดต่อทางธุรกิจที่มีความสำคัญได้อย่างรวดเร็ว

จากการศึกษาการใช้งานจดหมายอิเล็กทรอนิกส์บนโทรศัพท์เคลื่อนที่ในปัจจุบันพบว่า ผู้ให้บริการส่วนใหญ่จะเป็นรายเดียวกับผู้ให้บริการโทรศัพท์เคลื่อนที่ ทำให้ผู้ใช้บริการต้องมี E-Mail Address เพิ่มขึ้นและจะต้องใช้ Domain เดียวกับผู้ให้บริการ ซึ่งไม่เหมาะสมในการติดต่อธุรกิจ และการจัดการจดหมายอิเล็กทรอนิกส์ทำได้ยากเนื่องจากมี E-Mail Address มากกว่าหนึ่งแห่ง อาจทำให้ผู้ที่มาติดต่อเกิดความสับสน การใช้งานจดหมายอิเล็กทรอนิกส์บนโทรศัพท์เคลื่อนที่อาจไม่ได้ประโยชน์เท่าที่ควรหากจดหมายที่สำคัญถูกส่งไปยัง E-Mail Address ที่ใช้ในองค์กร ดังนั้นการพัฒนาให้การใช้งานจดหมายอิเล็กทรอนิกส์บนโทรศัพท์เคลื่อนที่ที่สามารถใช้ร่วมกับ E-Mail Address ที่มีอยู่แล้วนั้นน่าจะช่วยแก้ปัญหาต่างๆ ที่เกิดขึ้นได้

### 1.1 วัตถุประสงค์

วัตถุประสงค์ของโครงการนี้เพื่อพัฒนา Application ที่ให้บริการรับ-ส่งจดหมายอิเล็กทรอนิกส์บนโทรศัพท์เคลื่อนที่ด้วยภาษา XHTML และ PHP script ในการพัฒนา โดยการติดต่อกับ Mail Server ที่สนับสนุนโปรโตคอล IMAP หรือ POP3 แล้วนำมาแสดงผลต่างๆ บน XHTML Browser ที่มีอยู่ในโทรศัพท์เคลื่อนที่เพื่อเพิ่มความสะดวกและช่องทางในการใช้งาน และมีข้อแตกต่างจาก Web-based E-Mail ทั่วไปเนื่องจากข้อจำกัดของโทรศัพท์เคลื่อนที่คือจะทำการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แสดงผลบางส่วนของจดหมายก่อนเพื่อให้ผู้ใช้งานสามารถตัดสินใจที่จะอ่านจดหมายต่อหรือไม่ เพื่อเป็นการประหยัดจำนวนการรับส่งข้อมูลบนเครือข่ายไร้สายซึ่งมีค่าใช้จ่ายค่อนข้างสูง

## 1.2 ขั้นตอนการพัฒนาระบบ

การพัฒนาระบบจดหมายอิเล็กทรอนิกส์ผ่าน WAP เป็นการพัฒนา Application โดยการจำลองการทำงานของอุปกรณ์โทรศัพท์เคลื่อนที่ด้วย Nokia Mobile Internet Toolkit ประกอบด้วย Nokia Mobile Browser Simulator และ Nokia WAP Gateway Simulator ติดต่อกับ Web Server ที่สนับสนุนการใช้งาน WAP และ Mail Server โดยใช้ PHP Script มีขั้นตอนในการพัฒนาดังนี้

- ศึกษาเทคโนโลยี WAP ขั้นตอนการทำงาน ข้อจำกัดของอุปกรณ์ ภาษาที่ใช้ในการพัฒนา
- ศึกษาการพัฒนา Application บน โทรศัพท์เคลื่อนที่ด้วย XHTML
- ศึกษาการติดตั้ง Linux Server เพื่อให้บริการเป็น Web Server และ Mail Server
- ศึกษาการใช้ PHP Script เพื่อติดต่อกับ Mail Server
- วิเคราะห์ขั้นตอนและบริการต่างๆ ที่เพียงพอต่อการใช้งานจดหมายอิเล็กทรอนิกส์
- พัฒนาระบบเพื่อให้บริการจดหมายอิเล็กทรอนิกส์

## 1.3 ขอบเขตการดำเนินการ

ในการพัฒนาระบบจดหมายอิเล็กทรอนิกส์ผ่าน WAP มีขอบเขตการดำเนินการดังนี้

- ติดตั้ง Web Server ที่สนับสนุนการใช้งาน XHTML และ PHP
- ติดตั้ง Mail Server
- พัฒนาส่วนติดต่อกับผู้ใช้ทาง XHTML Browser บนโทรศัพท์เคลื่อนที่ด้วยภาษา XHTML
- ใช้ PHP script ในการติดต่อกับ Mail Server เพื่อตรวจสอบจดหมาย, อ่าน, สร้างจดหมายใหม่, ตอบ และส่งต่อจดหมาย แล้วแสดงผลบนโทรศัพท์เคลื่อนที่อย่างถูกต้อง

#### 1.4 ประโยชน์ที่คาดว่าจะได้รับ

ประโยชน์ที่คาดว่าจะได้รับจากโครงการพัฒนาระบบงานมีดังนี้

- ผู้ใช้งานจดหมายอิเล็กทรอนิกส์จะสามารถมีช่องทางเพิ่มขึ้นนอกเหนือจากการใช้งานผ่านเครื่องคอมพิวเตอร์
- รับ-ส่งจดหมายอิเล็กทรอนิกส์ที่มีความสำคัญได้ถึงแม้ไม่ได้อยู่ในสำนักงาน
- สามารถใช้โทรศัพท์เคลื่อนที่รับ-ส่งจดหมายอิเล็กทรอนิกส์ที่มี E-Mail Address เป็นขององค์กรได้ ลดข้อจำกัดที่การใช้งานจดหมายอิเล็กทรอนิกส์บนโทรศัพท์เคลื่อนที่จะต้องขึ้นกับผู้ให้บริการระบบ ไม่ยึดติดกับผู้ให้บริการ
- เพิ่มความคล่องตัวรวดเร็วในการตอบสนองจดหมายอิเล็กทรอนิกส์ที่มีความสำคัญทางธุรกิจ
- ใช้ทรัพยากรขององค์กรได้แก่ Mail Server ได้อย่างคุ้มค่า
- ลดค่าใช้จ่ายในการใช้งาน WAP เนื่องจากสามารถดูบางส่วนของจดหมายได้ก่อนการ Download จดหมายทั้งหมดมาอ่าน

## บทที่ 2

### WAP และ PHP

#### 2.1 Wireless Application Protocol (WAP)

Wireless Application Protocol หรือ WAP เป็นข้อกำหนดในการสื่อสารในเครือข่ายแบบไร้สายที่เป็นมาตรฐานเปิด ซึ่งทำให้อุปกรณ์ทุกชนิดที่ผลิตขึ้นมาตามมาตรฐานของ WAP สามารถสื่อสารกับอุปกรณ์หรือเครือข่ายที่ใช้โปรโตคอลต่างกันได้ เช่น TCP/IP, UDP/IP นอกจากนี้ WAP ยังสามารถทำงานได้บนระบบเครือข่าย GSM, CDMA รวมถึงระบบเครือข่ายอื่นๆ อีกด้วย

ก่อนหน้าที่จะมีการกำหนดมาตรฐานเปิดนี้ บริษัทชั้นนำทางด้านเครือข่ายไร้สายหลายบริษัท ได้พยายามสร้างมาตรฐานบริการในเครือข่ายโทรศัพท์เคลื่อนที่ที่เป็นของตนเอง และเนื่องจากการพัฒนาที่เป็นอิสระจากกันนี้เองทำให้เกิดบริการและผลิตภัณฑ์ต่างๆ มากมายโดยไม่เป็นมาตรฐานและไม่สามารถใช้งานร่วมกันได้ บริษัทต่างๆ จึงตัดสินใจร่วมมือพัฒนามาตรฐานบริการในเครือข่ายโทรศัพท์เคลื่อนที่ขึ้น โดยบริษัท Phone.com (เปลี่ยนชื่อมาจาก Unwired Planet), Ericsson, Motorola และ Nokia ได้ร่วมมือพัฒนามาตรฐานบริการในเครือข่ายไร้สายขึ้นซึ่งเรียกกลุ่มว่า WAP Forum และได้ออกข้อกำหนดของ WAP เวอร์ชัน 1.0 (WAP 1.0 Specifications) ออกมาเป็นมาตรฐานแรกในเดือนเมษายน ค.ศ.1998 และเปิดกว้างให้บุคคลหรือบริษัทภายนอกที่ ต้องการพัฒนา WAP เข้าร่วมเป็นสมาชิกของ WAP Forum ได้ และ WAP Forum ได้พัฒนา WAP ต่อเนื่องโดยก่อนการออก WAP เวอร์ชันใหม่จะต้องยึดปัจจัยสำคัญต่อไปนี้เป็นหลัก

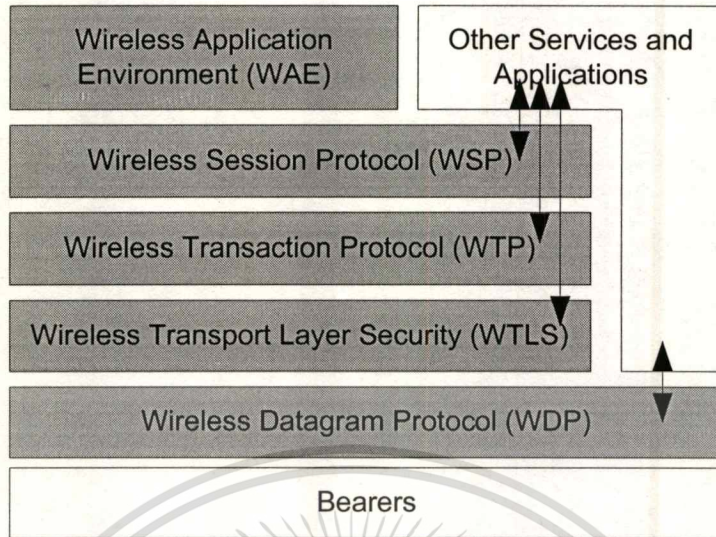
- ใช้มาตรฐานของอินเทอร์เน็ตให้มากที่สุดเท่าที่จะเป็นไปได้
- ใช้ Protocol Stack ของ WAP ซึ่งเป็นชนิดยัดเยียดได้ และสามารถแก้ไขได้ง่ายในอนาคต
- รองรับเครือข่ายไร้สายที่มีมาตรฐานต่างๆ ให้มากที่สุด โดยไม่ใช่เครือข่ายใดเครือข่ายหนึ่งเป็นมาตรฐานเท่านั้น เนื่องจาก WAP เป็นมาตรฐานเปิด
- ส่งข้อมูลเข้าไปในเครือข่ายให้น้อยที่สุดเท่าที่จะเป็นไปได้ เนื่องจากแบนด์วิธมีขนาดจำกัด
- ใช้ทรัพยากรที่มีอยู่บนอุปกรณ์อย่างมีประสิทธิภาพ

- ส่วนติดต่อกับผู้ใช้งานต้องรองรับอุปกรณ์ WAP ได้หลายชนิด ไม่เฉพาะยี่ห้อใดยี่ห้อหนึ่ง
- ต้องมีความปลอดภัยในการรับส่งข้อมูลสูง

## 2.2 WAP Protocol Stack

WAP Protocol Stack เป็นมาตรฐานสำหรับการสื่อสารในระบบ WAP ซึ่งใช้ OSI model (Open System Interconnection model) เป็นพื้นฐานเพื่อทำให้การพัฒนาโปรแกรมประยุกต์ของบริการในเครือข่ายไร้สายเป็นไปได้โดยมีความยืดหยุ่นและมีความแน่นอน โดยแบ่งออกเป็น 5 ระดับ (Layer) ดังนี้

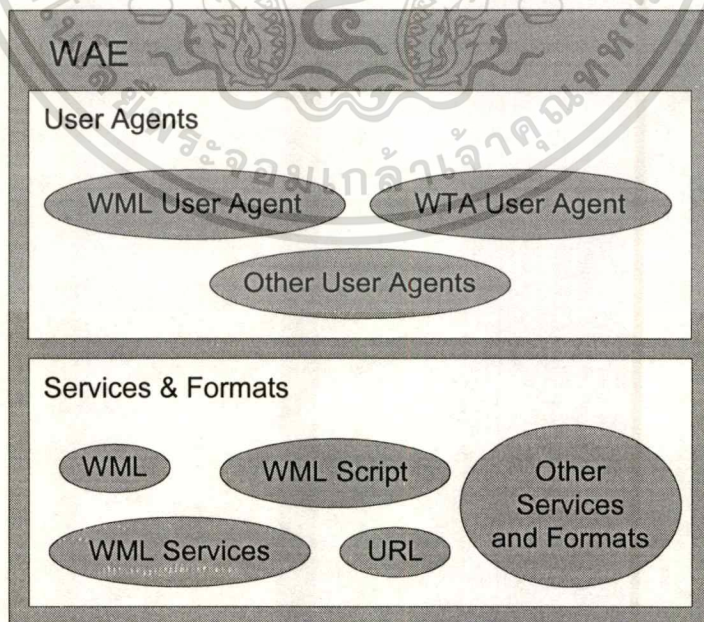
- ระดับ Application – Wireless Application Environment (WAE) เป็นตัวกำหนดสิ่งแวดล้อมของโปรแกรมประยุกต์ เช่น รูปแบบ (Format) ต่างๆ รวมถึงตัว WAP Browser ด้วย
  - ระดับ Session – Wireless Session Protocol (WSP) ทำหน้าที่ให้การทำให้ฟังก์ชันต่างๆ บนโปรแกรมประยุกต์สามารถทำงานได้อย่างถูกต้อง
  - ระดับ Transmission – Wireless Transaction Protocol (WTP) ทำหน้าที่ในการเพิ่มความน่าเชื่อถือ (Reliability) ของการรับส่งข้อมูล
  - ระดับ Security – Wireless Transport Layer Security (WTLS) ทำหน้าที่ในด้านความปลอดภัย (Security) ให้กับ WAP
  - ระดับ Transport – Wireless Datagram Protocol (WDP) ทำให้โปรแกรมที่พัฒนาสามารถทำงานบนเครือข่ายโทรศัพท์เคลื่อนที่ที่มีมาตรฐานต่างกันได้
- และเนื่องจาก WAP Stack มีความยืดหยุ่นสูงจึงทำให้ WSP, WTP, WTLS และ WDP สามารถติดต่อ (Interface) กับบริการอื่นๆ ที่ไม่ใช่ WAP ได้ดังรูปที่ 2.1



รูปที่ 2.1 WAP Protocol Stack

2.2.1. Wireless Application Environment (WAE)

WAE เป็นตัวกำหนดว่าโปรแกรมประยุกต์ใดจะสามารถทำงานบน WAP ได้ WAE มีจุดประสงค์คือทำให้โปรแกรมประยุกต์สามารถใช้งานกับอุปกรณ์หลายชนิดอย่างมีประสิทธิภาพ WAE จึงกำหนดรูปแบบของข้อมูลต่างๆ ซึ่งจากรูปที่ 2.2 จะเห็นว่า WAE แบ่งส่วนประกอบออกเป็น 2 ส่วนหลักๆ คือ ส่วนของ User Agents และส่วนของ Services และ Formats ต่างๆ



รูปที่ 2.2 ส่วนประกอบของ WAE

WML User Agent (WML UA) เป็นส่วนที่ใช้ในการทำส่วนติดต่อกับผู้ใช้งาน (User Interface) และแสดงข้อมูลต่างๆ บนจอของอุปกรณ์ WAP ซึ่งก็คือ WAP Browser ทำหน้าที่ในการแปล WML และ WMLScript ให้เป็นส่วนติดต่อกับผู้ใช้

Wireless Telephony Applications (WTA) ทำหน้าที่ในการเรียกบริการต่างๆ ของโทรศัพท์เคลื่อนที่ขึ้นมาด้วย WAP Browser WTA จะถูกใช้โดยผู้ให้บริการโทรศัพท์เคลื่อนที่เท่านั้น

Wireless Markup Language (WML) ถูกสร้างขึ้นโดยใช้ XML เป็นพื้นฐานเพื่อใช้ในอุปกรณ์ WAP มีคุณสมบัติดังนี้

- สามารถรองรับตัวอักษรรูปแบบต่างๆ ได้
- สามารถรองรับการใช้งานรูปภาพ (Image) ได้
- สามารถทำ User Input ได้
- สามารถเก็บ History ได้เหมือนอินเทอร์เน็ตเบราว์เซอร์
- สามารถรองรับ Navigation ต่างๆ เช่น Hyperlink, Soft-button ได้
- สามารถกำหนดตัวแปร (Variable) ได้
- สามารถ Encode ข้อมูลได้
- สามารถรองรับตัวอักษรแบบ Unicode ได้

Wireless Markup Language Script (WMLScript) ถูกสร้างขึ้นมาใช้ JavaScript เป็นพื้นฐาน สามารถทำให้ WML ทำงานได้ดีขึ้นเช่น

- Validate User Input หรือการตรวจสอบข้อมูลที่ผู้ใช้ใส่เข้าไปในระบบที่อุปกรณ์ WAP ได้โดยตรง
- ใช้ฟังก์ชันของอุปกรณ์ WAP ได้ เช่นการใช้ WMLScript เพื่อส่ง Short Message หรือเรียกโทรศัพท์เลขหมายอื่นๆ

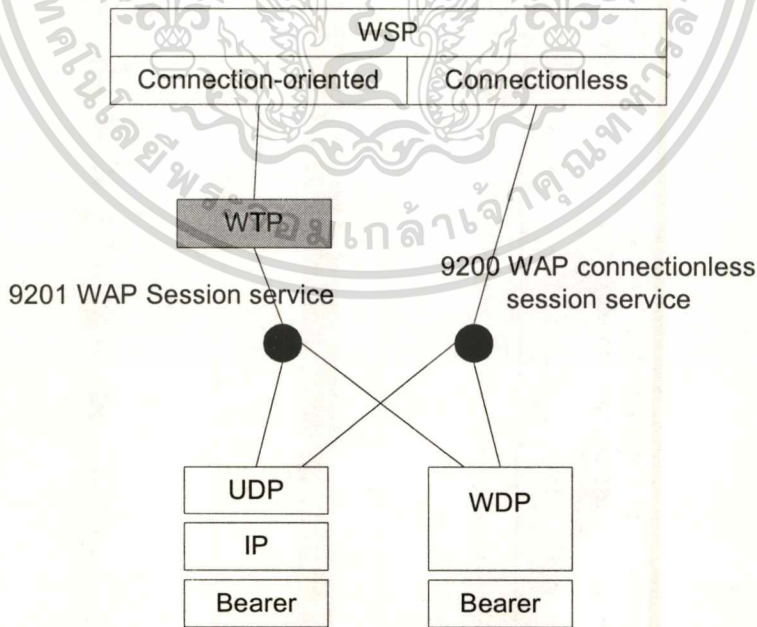
Universal Resource Locators (URL) ใช้ในการเข้าไปยัง Content Server เช่นเดียวกับอินเทอร์เน็ต แต่การเข้าไปยัง WTA Server จะใช้ URI (Universal Resource Identifiers) เนื่องจากการเข้าไปสู่ส่วนที่เป็นมาตรฐานเฉพาะของเครือข่ายโทรศัพท์เคลื่อนที่

### 2.2.2. Wireless Session Protocol (WSP)

WSP เป็นตัวกำหนดกฎเกณฑ์ต่างๆ ในการรับส่งข้อมูลระหว่างเซิร์ฟเวอร์กับไคลเอนต์ ซึ่งเป็นการแปลง HTTP 1.1 ให้เป็นไบนารีเพื่อให้ใช้กับเครือข่ายที่มีแบนด์วิดท์ต่ำ และเพิ่มฟังก์ชันบางอย่างนอกเหนือจากฟังก์ชันของ HTTP 1.1 ด้วยเช่น

- Header Encoding สามารถ Encode ชื่อและค่าของ Header ได้
- Capability Negotiation ใช้ในการพิจารณาว่าไคลเอนต์ที่ส่ง Request มาสามารถรองรับออปชันและฟังก์ชันของ WSP ได้หรือไม่ และใช้ในการแลกเปลี่ยน Header Code Page ด้วย
- Header Caching มีการเก็บ Header ไว้เมื่อมีการสร้าง Session ขึ้นแล้วเพื่อลดปริมาณข้อมูลที่ต้องส่งซ้ำซ้อน
- Long-lived Session ระยะเวลาในการใช้งาน Session นาน เนื่องจากการใช้งานอาจเกิดสัญญาณโทรศัพท์ขาด เมื่อมีสัญญาณใหม่ก็สามารถใช้ Session ต่อได้ทันที
- Push เซิร์ฟเวอร์สามารถส่งข้อมูล ไปยังไคลเอนต์โดยไม่ต้องมีการร้องขอ

WSP แบ่งออกตามลักษณะของการเชื่อมต่อออกเป็น 2 ประเภทคือ Connection-oriented และ Connectionless ดังรูปที่ 2.3



รูปที่ 2.3 ประเภทของ WSP ตามลักษณะการเชื่อมต่อ

Connection-oriented mode เกิดจากการใช้งานตามปกติ เช่น การเข้าไปใช้งานที่อยู่บนเซิร์ฟเวอร์ โดยมีการสร้าง Session ระหว่างไคลเอนต์กับเซิร์ฟเวอร์ ซึ่งใช้พอร์ต 9201 และมี WTP ช่วยให้การรับส่งข้อมูลเป็นไปอย่างถูกต้องและเชื่อถือ (Reliability) ได้

Connectionless ใช้พอร์ต 9200 ไม่มีการสร้าง Session ระหว่างไคลเอนต์กับเซิร์ฟเวอร์ เพราะข้อมูลถูกส่งได้โดยใช้เพียง Datagram เดียว เช่นการ Push การรับส่งข้อมูลแบบ Connectionless นี้ไม่สามารถรับรองได้ว่าข้อมูลถึงจุดหมายหรือไม่

### 2.2.3. Wireless Transaction Protocol (WTP)

WTP ทำหน้าที่ในการควบคุมการรับส่งข้อมูลบน WAP Stack โดย WTP เป็นตัวสร้างความน่าเชื่อถือในการสื่อสาร เนื่องจากจะมีการส่งข้อมูลใหม่หากเกิดการสูญหาย และป้องกันการส่งข้อมูลซ้ำ การรับส่งแต่ละครั้งจะเรียกว่า Transaction ซึ่งมีกลไกที่ WTP ใช้เพื่อยืนยันความน่าเชื่อถือของการรับส่งข้อมูลมี 3 ชนิดคือ

Class 0 : Unreliable one-way requests ไม่มีการส่งข้อมูลใหม่เมื่อมีการสูญหายเนื่องจากไม่มีการยืนยัน (Acknowledgement) ว่าผู้รับได้รับข้อมูลแล้ว

Class 1 : Reliable one-way requests ผู้รับจะมีการยืนยันว่าได้รับข้อมูล หากไม่มีการยืนยันผู้ส่งจะส่งข้อมูลใหม่อีกครั้ง

Class 2 : Reliable two-way request-reply transaction ใช้แก้ปัญหากรณีที่การยืนยันจากผู้รับเกิดการสูญหายทำให้การส่งข้อมูลซ้ำซ้อน เพื่อให้การรับส่งข้อมูลเป็นไปอย่างถูกต้องมากขึ้น ซึ่งวิธีการนี้ถูกนำมาใช้ใน Request/Reply Transaction บน WAP

### 2.2.4. Wireless Transport Layer Security (WTLS)

WTLS เป็นระดับที่เพิ่มความปลอดภัยให้กับ โปรแกรมประยุกต์โดยพัฒนามาจาก Transport Layer Security (TLS) หรือ Secure Sockets Layer (SSL) ที่ใช้ในอินเทอร์เน็ตในการทำ E-commerce โดย WTLS ถูกสร้างขึ้นเพื่อเครือข่ายโทรศัพท์เคลื่อนที่โดยเฉพาะเนื่องจากมีแบนด์วิดท์ต่ำ

โดยปกติเครือข่ายโทรศัพท์เคลื่อนที่ระบบดิจิทัลจะมีการเข้ารหัส (Encryption) ขณะส่งข้อมูลซึ่งทำให้มีความปลอดภัยสูงอยู่แล้ว WTLS จึงเป็นเพียงความสามารถที่นำมาใช้เพิ่มเติมบน WAP เพื่อเพิ่มความปลอดภัยให้สูงขึ้น เช่นในการทำ E-commerce แต่เนื่องจากโปรแกรมประยุกต์บางโปรแกรมไม่ต้องการความปลอดภัยสูง จึงสามารถเลือกที่จะไม่ใช้ระบบความปลอดภัยได้ ทำให้ข้อมูลที่ส่งมีขนาดไม่มากเนื่องจากไม่มี Overhead ดังนั้นโปรแกรมประยุกต์จึงสามารถใช้ได้ทั้งแบบใช้ความปลอดภัย (Secured Mode) และไม่ใช้ความปลอดภัย (Unsecured Mode)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บริการต่างๆที่ WTLS มีสามารถแยกใช้งานได้ขึ้นอยู่กับความต้องการของโปรแกรมประยุกต์และกฎหมายในแต่ละประเทศ

- Data Integrity – ใช้ Checksums เป็นตัวตรวจสอบความถูกต้องของข้อมูลว่าไม่มีการแก้ไขเกิดขึ้น
- Privacy – ใช้การเข้ารหัสเพื่อไม่ให้เกิดการลักลอบขโมยข้อมูลระหว่างทาง หรือเมื่อมีการลักลอบก็ไม่สามารถเข้าใจข้อมูลนั้นได้
- Authentication – ใช้เพื่อตรวจสอบผู้ที่จะเข้ามาใช้โปรแกรมประยุกต์บนเครื่องให้บริการ
- Denial-of-service – จะไม่ทำการประมวลผลข้อมูลที่ไม่ได้ถูกตรวจสอบอย่างถูกต้อง WTLS ให้ความปลอดภัยจากอุปกรณ์ที่ใช้งาน WAP ไปถึง WAP Gateway เท่านั้น ดังนั้นการใช้งานโปรแกรมประยุกต์ที่อยู่บน WWW Server จึงต้องใช้ SSL ควบคู่ไปด้วย

#### 2.2.5. Wireless Datagram Protocol (WDP)

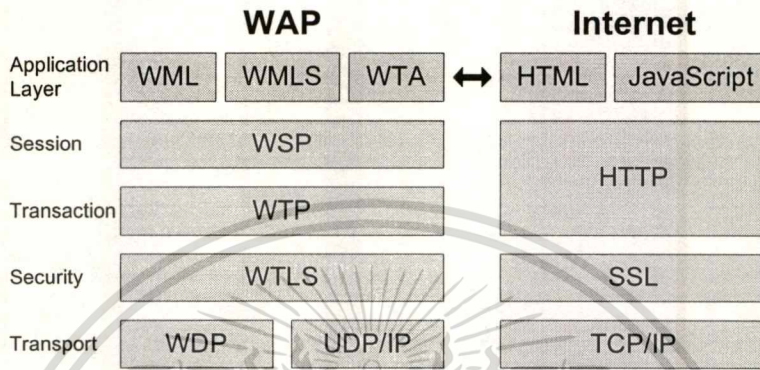
WDP เป็นระดับที่ทำให้โปรแกรมประยุกต์สามารถทำงานได้เป็นปกติโดยไม่ขึ้นกับ Bearer (เปรียบเทียบกับได้กับ Network Layer) ถ้า Bearer สามารถรองรับ TCP/IP ได้ ก็จะใช้ UDP (User Datagram Protocol) แต่ถ้า Bearer ไม่สามารถรองรับ TCP/IP ได้ ก็จะใช้ WDP ทำหน้าที่เช่น Fragmenting และ Reassembling แทน เช่นการใช้งาน WAP ผ่านระบบ SMS (Short Message Service) บนเครื่องโทรศัพท์รุ่นเก่าที่ไม่มี WAP Browser

ในระบบ GSM ใช้ IP เป็นมาตรฐาน ดังนั้นจึงใช้ UDP แทนการใช้ WDP โดยหากเกิดความผิดพลาดที่ทำให้ข้อมูลไม่สามารถส่งไปยังผู้รับได้ก็จะมีกรส่งข้อความแจ้งกลับมายังผู้ส่งซึ่งเป็นหน้าที่ของ WCMP (Wireless Control Message Protocol) ซึ่งทำหน้าที่คล้ายกับ ICMP (Internet Control Message Protocol) ใน IP

ปัจจุบันมีการใช้ Bearer ที่เป็นแพคเกจส์ (Packets) มาใช้งานบนเครือข่าย GSM นั่นคือ General Packet Radio Service (GPRS) โดย GPRS จะติดตั้งเข้ากับระบบโทรศัพท์เคลื่อนที่ในปัจจุบัน ช่วยให้ความเร็วในการทำงานของ WAP เพิ่มขึ้น การเชื่อมต่อจะเกิดขึ้นตลอดเวลาที่เครื่องโทรศัพท์เคลื่อนที่ทำงานอยู่บนเครือข่าย และจะได้รับ IP Address จากเครือข่ายในลักษณะใกล้เคียงกับการทำงานของคอมพิวเตอร์บนเครือข่าย IP

### 2.3 WAP Stack กับ Internet Stack

จากการศึกษาหน้าที่และการทำงานของ WAP Stack สามารถเปรียบเทียบกับ Internet Stack ดังแสดงในรูปที่ 2.4 ได้ดังต่อไปนี้



รูปที่ 2.4 WAP Stack กับ Internet Stack

ระดับ Transport – บนอินเทอร์เน็ตจะใช้ TCP/IP เป็นตัวทำหน้าที่ในการรับส่งข้อมูล แต่บน WAP จะใช้ WDP หรือ UDP เป็นตัวรับส่งข้อมูล โดยหากระดับ Network เป็นประเภทที่มีการรองรับ IP ก็จะใช้ UDP ถ้าไม่รองรับก็จะใช้ WDP เพื่อสามารถรองรับระบบเครือข่ายโทรศัพท์เคลื่อนที่ต่างๆ ได้

ระดับ Security – บนอินเทอร์เน็ตจะมีการใช้ SSL ซึ่งใช้กับ TCP/IP แต่บน WAP ใช้ WTLS จะต้องใช้กับ Datagram เท่านั้น

ระดับ Transaction และ Session – บน WAP มีการใช้ WTP และ WSP โดย WTP ทำหน้าที่คล้ายกับ TCP ที่เป็นตัวควบคุมการรับส่งข้อมูลบนอินเทอร์เน็ต แต่ WTP จะทำการลดขนาดและจำนวนของการส่งข้อมูลและยังสามารถเลือกส่งเฉพาะข้อมูลที่หายไปโดยไม่ต้องส่งใหม่ทั้งหมด ส่วน WSP ก็จะแบ่งออกเป็น 2 ประเภทคือ Connection Mode สำหรับข้อมูลที่ต้องการความแน่นอนและความน่าเชื่อถือโดยอาศัย WTP เข้ามาช่วย หากไม่ต้องการความแน่นอนก็จะใช้ประเภท Connectionless Mode เช่นการ Push เป็นต้น ซึ่งบนอินเทอร์เน็ตมี HTTP ทำหน้าที่ของ WSP แต่ HTTP ไม่สามารถ Push แบบไม่มี Session หรือ Connection ได้

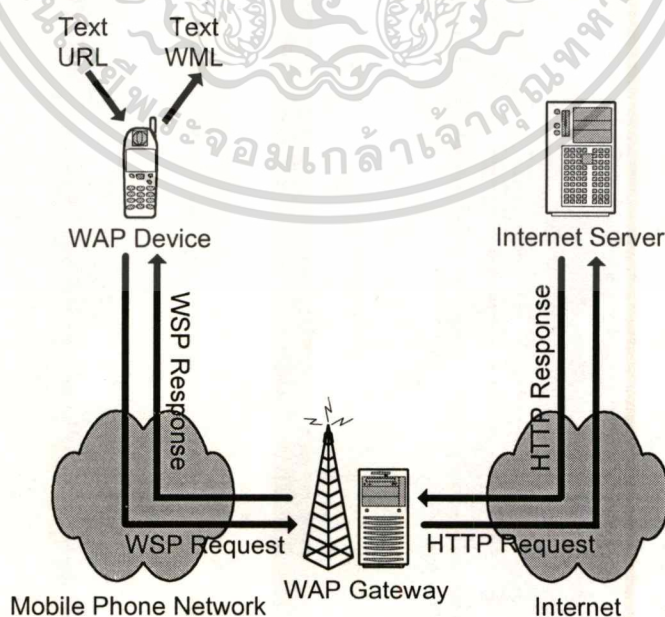
ระดับ Application – เป็นตัวกำหนดรูปแบบของข้อมูล HTML และ WML จะทำหน้าที่เหมือนกันคือเป็นรูปแบบของข้อมูล ทั้ง HTML เป็นพื้นฐานของ XML ขณะที่ WMLพัฒนามาจาก XML สำหรับ WMLScript จะถูกใช้ทดแทน JavaScript แต่สิ่งที่บนอินเทอร์เน็ตไม่มีคือ WTA ซึ่งทำหน้าที่ต่างๆ ของโทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเห็นได้ว่า WAP ถูกสร้างขึ้นโดยพยายามใช้มาตรฐานเดิมที่มีอยู่แล้วเป็นหลักแต่มีการปรับปรุงเพื่อให้สามารถทำงานบนข้อจำกัดของระบบเครือข่ายโทรศัพท์เคลื่อนที่ที่มีหลากหลายและมีความเร็วต่ำได้

## 2.4 ขั้นตอนการทำงานบนเครือข่าย WAP

เมื่อผู้ใช้งานกำหนด URL ของ WAP Site ที่ต้องการลงในอุปกรณ์ WAP อุปกรณ์ WAP นั้น จะทำการเปลี่ยน HTTP Request ซึ่งมีรูปแบบเป็นข้อความปกติ (Text) เป็น WSP Request ซึ่งมีรูปแบบเป็นไบนารี (Binary) เพื่อลดขนาดข้อมูลแล้วส่งไปยัง WAP Gateway โดยผ่านเครือข่ายโทรศัพท์เคลื่อนที่ หลังจากที่ WAP Gateway ได้รับข้อมูลก็จะทำการ Decode ข้อมูลกลับจาก WSP Request ที่ได้ให้เป็น HTTP Request เพื่อส่งต่อไปยังเครือข่ายอินเทอร์เน็ต เมื่อพบ Internet Server ที่ต้องการ Internet Server ก็สร้าง WML Page ส่งกลับมายังเครือข่ายอินเทอร์เน็ตเป็น HTTP Response เมื่อ WAP Gateway ได้รับก็จะ Encode ข้อมูล HTTP Response ที่ได้รับให้เป็น WSP Response แล้วส่งไปบนเครือข่ายโทรศัพท์เคลื่อนที่สู่อุปกรณ์ WAP หลังจากที่อยู่อุปกรณ์ WAP ได้รับ WSP Response ก็จะแปลง WML ที่อยู่ในรูปแบบไบนารีให้เป็น WML ในรูปแบบข้อความปกติ WAP Browser ที่อยู่บนอุปกรณ์ WAP ก็จะแสดงผลที่ได้ให้ผู้ใช้งาน ซึ่งขั้นตอนต่างๆ สามารถแสดงได้ดังรูปที่ 2.5



รูปที่ 2.5 การทำงานบนเครือข่าย WAP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.5 WAP Gateway

เป็นตัวกลางระหว่างเครือข่ายโทรศัพท์เคลื่อนที่กับเครือข่ายอินเทอร์เน็ต ทำให้อุปกรณ์ WAP สามารถใช้งาน WAP ได้ไม่ว่าอุปกรณ์นั้นจะเป็นอะไร และใช้อะไรเป็นตัวกลางในการเข้าระบบ หน้าที่ที่สำคัญของ WAP Gateway คือ

1. กำหนดว่าอุปกรณ์ WAP ใช้มาตรฐานอะไรในการติดต่อเข้ามาเพื่อเลือกที่จะใช้ WDP หรือ UDP ในการเชื่อมต่อ ทำให้การแสดงผลจากโปรแกรมประยุกต์มีการทำงานที่ถูกต้อง
2. เก็บข้อมูลเกี่ยวกับความสามารถและความต้องการพิเศษ (Capability and Preference Information หรือ CPI) ของอุปกรณ์ WAP เพื่อให้ทราบว่ามีฟังก์ชันใดบ้างที่สามารถทำงานได้บนอุปกรณ์ WAP นั้น
3. แปลงโปรโตคอลระหว่างเครือข่ายโทรศัพท์เคลื่อนที่กับเครือข่ายอินเทอร์เน็ต เช่นการแปลงระหว่าง WSP กับ HTTP
4. ส่งคำร้องขอที่ได้รับมาจากอุปกรณ์ WAP ไปยังผู้ให้บริการบนอินเทอร์เน็ต โดยการสร้าง HTTP Connection
5. จัดการเรื่องความปลอดภัยโดยใช้ WTLS ซึ่งความปลอดภัยประเภทต่างๆที่นำมาใช้บน WAP ได้ มีดังนี้
  - Mobile Subscriber Integrated Services Digital Network (MSISDN) – WAP Gateway สามารถรู้หมายเลขโทรศัพท์ของอุปกรณ์ WAP ที่ติดต่อมา
  - HTTP Basic Authentication – การใช้ชื่อผู้ใช้ (Username) และรหัสผ่าน (Password) ในการตรวจสอบผู้ใช้
  - HTTP Proxy Authentication – สามารถตรวจสอบชื่อผู้ใช้ที่เชื่อมต่อผ่าน Proxy Server ได้
  - Data Encryption – สามารถเข้ารหัสข้อมูลทั้งแบบ 56 และ 128 บิต
  - LDAP – ทำงานร่วมกับ LDAP เพื่อเพิ่มความปลอดภัย
6. บีบอัดข้อมูล (Encode) ที่มาจากเครือข่ายอินเทอร์เน็ตเพื่อส่งไปเครือข่ายโทรศัพท์เคลื่อนที่ และคลายข้อมูล (Decode) ที่มาจากเครือข่ายโทรศัพท์เคลื่อนที่เพื่อส่งไปเครือข่ายอินเทอร์เน็ต
7. แปลงโปรโตคอลอื่นๆ นอกเหนือจาก HTTP เช่น POP3, SMTP หรือ IMAP4 ไปเป็น WSP

## 2.6 WAP 2.0 Specification

หลังจากที่มีการกำหนดมาตรฐาน WAP เวอร์ชัน 1 มาตั้งแต่เมื่อปี ค.ศ.1998 ก็ยังมีการพัฒนาอย่างต่อเนื่องจนถึงปัจจุบันทำให้เกิด WAP เวอร์ชัน 2.0 ขึ้น WAP Forum จึงพยายามที่จะทำให้การใช้งาน WAP ใกล้เคียงกับการใช้งานอินเทอร์เน็ตโดยซึ่งมีคุณสมบัติที่ทำได้สำเร็จแล้ว ดังนี้

- สนับสนุนมาตรฐาน โพรโตคอลที่ใช้บนอินเทอร์เน็ตเช่น IP, TCP และ HTTP โดยการปรับปรุงให้เหมาะกับการสื่อสารแบบไร้สายและใช้ประโยชน์จากเครือข่ายอินเทอร์เน็ตที่มีอยู่มากที่สุด
- รองรับบริการและโปรแกรมประยุกต์ที่พัฒนาโดยใช้มาตรฐาน WAP เวอร์ชันก่อนหน้ารวมถึงระบบเครือข่ายโทรศัพท์เคลื่อนที่ทั้งแบบเก่าและแบบใหม่
- สามารถพัฒนาโปรแกรมประยุกต์ที่มีการโต้ตอบกับผู้ใช้ได้
- เพิ่มความสะดวกและคุ้นเคยให้กับผู้ใช้ในการใช้งานบนข้อจำกัดของอุปกรณ์ WAP เช่น การแสดงผลที่มีขนาดเล็ก หน่วยความจำที่มีขนาดจำกัด
- ใช้เครือข่ายอย่างคุ้มค่าที่สุดและใช้พลังงานของอุปกรณ์ WAP ให้น้อยที่สุด
- เพิ่มความยืดหยุ่นสำหรับบริษัทผู้ผลิตอุปกรณ์ WAP ในการออกแบบส่วนติดต่อกับผู้ใช้เพื่อให้ตรงความต้องการของตลาด

## 2.7 เทคโนโลยีและมาตรฐานใหม่ใน WAP 2.0

WAP 2.0 มีความสามารถและเทคโนโลยีใหม่ๆ เพิ่มขึ้นจากเวอร์ชันก่อนหน้าได้แก่

- สามารถรองรับระบบเครือข่ายโทรศัพท์เคลื่อนที่ใหม่ๆ เนื่องจากปัจจุบันผู้ให้บริการโทรศัพท์เคลื่อนที่ที่มีการพัฒนาเครือข่ายให้ดีขึ้น มีความเร็วที่เพิ่มขึ้นเช่น GPRS และ HSCSD (High-Speed Circuit-Switched Data) รวมทั้งการเข้าสู่ระบบเครือข่ายโทรศัพท์เคลื่อนที่ยุคที่ 3 (Third-Generation หรือ 3G) เช่น W-CDMA และ CDMA2000 ที่มีแบนด์วิดท์สูงขึ้น จึงพร้อมที่จะรองรับข้อมูลแบบใหม่ๆ เช่น Streaming และทำให้โทรศัพท์เคลื่อนที่ติดต่อกับระบบอยู่ตลอดเวลา (Always On)
- ใช้ TCP/IP เป็นมาตรฐาน เนื่องจากระบบเครือข่ายโทรศัพท์เคลื่อนที่ใหม่ๆ มีความสามารถรองรับแพ็คเกจ IP ได้
- รองรับความสามารถที่เพิ่มขึ้นของโทรศัพท์เคลื่อนที่ที่มีความเร็วในการประมวลผลที่สูงขึ้น มีการแสดงผลที่ละเอียดขึ้นรวมทั้งการแสดงผลเป็นสีได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ใช้เทคโนโลยีใหม่เพื่อเพิ่มความเป็นมิตรกับผู้ใช้ โดยพยายามทำให้ผู้ใช้งานอินเทอร์เน็ตผ่านโทรศัพท์เคลื่อนที่มีความรู้สึกเหมือนการใช้งานอินเทอร์เน็ตปกติโดยเพิ่มการสนับสนุน XHTML และ Cascading Style Sheets (CSS)

## 2.8 XHTML

XHTML (Extensible Hypertext Markup Language) เป็นภาษาที่ใช้ในการพัฒนาเว็บที่สนับสนุนรูปแบบของข้อมูลหลายประเภทเช่น รูปภาพ, เสียง, วิดีโอ โดยมีมาตรฐานที่ควบคุมโดย World Wide Web Consortium (W3C) ซึ่งเป็นการพัฒนาต่อจาก HTML 4 มีการกำหนดโครงสร้างและรูปแบบที่แน่นอน

W3C กำหนดให้ XHTML Basic มีความสามารถในการแสดงผลและการกำหนดรูปแบบข้อมูลที่ดีเพียงพอที่จะใช้งานบนอุปกรณ์หลายๆ ประเภท เช่น เครื่องคอมพิวเตอร์ทั่วไป, เครื่องรับโทรทัศน์, Personal Digital Assistant (PDA) และอุปกรณ์สื่อสารไร้สายต่างๆ

XHTML Basic ได้รับการดัดแปลงจาก XHTML 1.0 โดยตัดความสามารถบางอย่างออก เช่นการแสดงผลแบบเฟรมที่อาจมีปัญหาในการแสดงผลบนอุปกรณ์ที่มีขนาดเล็ก โดยตัว XHTML เองก็มีการพัฒนาต่อจาก HTML และทำตามข้อกำหนดของ XML จึงทำให้การสร้างเว็บโดย XHTML มีโครงสร้างที่ไม่ซับซ้อนมากนัก ซึ่งเป็นผลดีสำหรับการแสดงผลบนอุปกรณ์ขนาดเล็กและการใช้พลังงานที่มีจำกัด และจากข้อดีและมาตรฐานใหม่ต่างๆ ทำให้ W3C เสนอแนะให้การสร้างเว็บในอนาคตหันมาใช้ XHTML ในการพัฒนาทั้งบนเครื่องคอมพิวเตอร์ทั่วไปรวมถึงบนบราวเซอร์หลายชนิด ซึ่งแตกต่างจากการพัฒนาโดยใช้ WML (Wireless Markup Language) ซึ่งอาจมีการแสดงผลที่แตกต่างกันบนบราวเซอร์ชนิดต่างกัน

## 2.9 XHTML Mobile Profile

WAP Forum กำหนดให้ XHTML Mobile Profile มีคุณสมบัติมากกว่า XHTML Basic โดยนำคุณสมบัติบางอย่างจาก XHTML 1.1 มาใช้เพิ่มเติมดังนี้

- `<hr>`
- `<b>`
- `<i>`

- <big>
- <small>
- <style>
- style="..."

เอกสารที่สร้างโดย XHTML MP จะสามารถแสดงผลได้บนบราวเซอร์หลายชนิดรวมถึงอุปกรณ์สื่อสารไร้สายที่มีอยู่มากมายโดยมีคุณสมบัติที่ต่างกัน และจากการที่ XHTML MP ถูกพัฒนาขึ้นโดยความร่วมมือของทั้ง WAP Forum กับ W3C ทำให้การรวมตัวของการใช้งาน WAP และการใช้งานอินเทอร์เน็ตเข้าใกล้กันมากขึ้น ทำให้จะเป็นที่ยอมรับในวงกว้างในระยะยาว

การสนับสนุน XHTML MP ทำให้เกิดข้อดีขึ้นสำหรับบราวเซอร์บนอุปกรณ์สื่อสารไร้สายหลายอย่างเนื่องจาก XHTML มีการกำหนดรูปแบบที่มีข้อบังคับมากกว่า HTML และ XHTML ก็พัฒนามาจาก HTML ดังนั้นแท็กต่างๆ ก็จะมีลักษณะเดียวกับแท็กของ HTML ที่มีอยู่ในปัจจุบัน ทำให้เครื่องมือที่ใช้ในการพัฒนาและออกแบบเว็บในปัจจุบันสามารถทำงานได้กับ XHTML นอกจากนี้ผู้พัฒนายังสามารถดูการแสดงผลจาก XHTML MP ได้จากบราวเซอร์มาตรฐานบนคอมพิวเตอร์ทั่วไปได้อย่างถูกต้อง

## 2.10 ความแตกต่างระหว่าง XHTML กับ WML และ cHTML

ถึงแม้ว่า WML จะมีพื้นฐานจาก XML เหมือน XHTML ก็ตาม แต่ WML กับ XHTML MP ก็มีความแตกต่างกันบางอย่างดังนี้

- ใน WML ไม่มีการกำหนดเนมสเปซ แต่ใน XHTML มีการใช้และกำหนดเป็น XHTML
- XHTML MP ใช้ <html> เป็นอิลิเมนต์หลัก ในขณะที่ WML ใช้ <wml> เป็นอิลิเมนต์หลัก
- XHTML MP ต้องมี <head> แต่ใน WML อาจไม่ต้องมีก็ได้
- ไม่มีการใช้งาน <template> จาก WML
- XHTML MP ใช้ <body> แทน <card> ที่ใช้ใน WML
- แอตทริบิวต์ ordered ใน WML ถูกยกเลิกออกจาก XHTML MP
- อีเวนต์ต่างๆ ใน WML จะใช้ไม่ได้
- แท็ก <do> จาก WML ก็ใช้ไม่ได้เช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หากกล่าวถึง cHTML ภาษาที่ใช้ในการพัฒนาโปรแกรมประยุกต์ในระบบ i-mode ของบริษัท DoCoMo จากประเทศญี่ปุ่น จะพบข้อแตกต่างจาก XHTML MP ที่เห็นได้อย่างชัดเจนคือ

- cHTML ไม่ใช่มาตรฐาน W3C
- cHTML ไม่สนับสนุนการใช้งาน Cascading Style Sheets

ซึ่งเห็นได้ว่า XHTML MP ซึ่งใช้มาตรฐานจาก W3C และ WAP Forum น่าจะได้รับการสนับสนุนและมีความน่าเชื่อถือจากผู้ผลิตและผู้พัฒนามากกว่า และการที่ cHTML ไม่สนับสนุนการใช้งาน CSS ทำให้ความสามารถในการจัดการการแสดงผลอาจทำได้ยากกว่า

จากการที่ XHTML MP มีการพัฒนาขึ้นโดยเป็นมาตรฐานของ W3C และ WAP Forum ทำให้ได้รับความน่าเชื่อถือ การพัฒนาโปรแกรมประยุกต์สามารถเรียนรู้ได้ง่ายเพราะพัฒนาจาก HTML และใช้มาตรฐาน XML เครื่องมือที่ใช้ในการพัฒนามีอยู่มากมาย นอกจากนี้ความสามารถในการใช้งาน Cascading Style Sheets ยังช่วยให้การแสดงผลมีความน่าสนใจมากขึ้น สามารถกำหนด CSS มาตรฐานให้ผู้พัฒนาใช้งานได้ตามชนิดของอุปกรณ์ การแก้ไขรูปแบบทำได้โดยแก้ไขที่ CSS เพียงทีเดียว และจากการที่ cHTML ไม่เป็นมาตรฐานของ W3C แต่ขึ้นกับบริษัทเพียงบริษัทเดียวจึงทำให้ผู้ใช้บริการมีทางเลือกน้อย และอุปกรณ์ส่วนใหญ่ที่ใช้กันนอกประเทศญี่ปุ่นยังไม่สนับสนุนเท่าที่ควร ประกอบกับ XHTML MP ก็มีความสามารถใกล้เคียงกัน ดังนั้น XHTML น่าจะเป็นทางเลือกที่น่าสนใจในการพัฒนาโปรแกรมประยุกต์บนอุปกรณ์สื่อสารไร้สาย

## 2.11 PHP Script

PHP หรือ Personal Homepage เป็นภาษาจาวาสคริปต์ scripting language คำสั่งต่างๆจะเก็บอยู่ในไฟล์ที่เรียกว่า สคริปต์ (script) เป็นโปรแกรมตีความภาษา (Interpreter) ที่ใช้ในการตีความเว็บเพจที่เขียนขึ้นมาโดยใช้ไวยากรณ์หรือ syntax ของภาษา C โดยที่ทีมงานผู้พัฒนา ได้อาศัยเค้าโครงการทำงานมาจาก Perl แล้วพัฒนาใหม่ด้วย C++ แรกเริ่ม PHP จะรู้จักในนามว่า Personal Homepage แต่ต่อมาด้วยศักยภาพของภาษาซี ทำให้เว็บเพจที่เขียนด้วย PHP ทำงานได้รวดเร็วและหลากหลายและได้เปลี่ยนชื่อเป็น PHP Hypertext Preprocessor ลักษณะของ PHP ที่แตกต่างจากภาษาสคริปต์แบบอื่นๆ คือ PHP ได้รับการพัฒนาและออกแบบมาเพื่อใช้งานในการสร้างเอกสารแบบ HTML โดยสามารถสอดแทรกหรือแก้ไขเนื้อหาได้โดยอัตโนมัติ ดังนั้นจึงกล่าวว่า PHP เป็นภาษาที่เรียกว่า server-side หรือ HTML-embedded scripting language เป็นเครื่องมือที่สำคัญชนิดหนึ่งที่จะช่วยให้สามารถสร้างเอกสารแบบ Dynamic HTML ได้อย่างมีประสิทธิภาพมากขึ้น

PHP ได้รับการเผยแพร่เป็นครั้งแรกในปี ค.ศ.1994 จากนั้นก็มีการพัฒนาต่อมาตามลำดับเป็นเวอร์ชัน 1 ในปี 1995 เวอร์ชัน 2 (ตอนนั้นใช้ชื่อว่า PHP/FI) ในช่วงระหว่าง 1995-1997 และเวอร์ชัน 3 ช่วง 1997 ถึง 1999 จนถึงเวอร์ชัน 4 ในปัจจุบัน

PHP เป็นผลงานที่เติบโตมาจากกลุ่มของนักพัฒนาในเชิงเปิดเผยรหัสต้นฉบับ หรือ Open Source สามารถนำมาใช้งานได้ฟรีโดยดาวน์โหลดได้จาก <http://www.php.net> ดังนั้น PHP จึงมีการพัฒนาไปอย่างรวดเร็ว และแพร่หลายโดยเฉพาะอย่างยิ่งเมื่อใช้ร่วมกับ Apache Web server ระบบปฏิบัติการอย่างเช่น Linux หรือ FreeBSD เป็นต้น ในปัจจุบัน PHP สามารถใช้ร่วมกับ Web Server หลายๆตัวบนระบบปฏิบัติการอย่างเช่น Windows 95/98/NT เป็นต้น

## 2.12 ฟังก์ชันที่เกี่ยวข้องกับ IMAP ใน PHP

PHP สนับสนุนการทำงานกับ Mail Server ทั้ง IMAP และ POP3 โดยผ่าน IMAP function library ซึ่งมีฟังก์ชันต่างๆ ให้ใช้งานเป็นจำนวนมาก โดยมีฟังก์ชันที่นำมาใช้งานดังนี้

1. `imap_open` ทำหน้าที่สร้างการเชื่อมต่อไปยัง Mail Server โดยมีรูปแบบ `imap_open ( string mailbox, string username, string password [, int options] )`
  - mailbox: ชื่อหรือ IP address ของ Mail Server
  - username: ชื่อผู้ใช้
  - password: รหัสผ่าน
2. `imap_close` ทำหน้าที่ปิดการเชื่อมต่อ `imap_close ( resource imap_stream [, int flag] )`
  - `imap_stream`: ค่า `imap stream` ที่ได้จากการเชื่อมต่อ
3. `imap_header` ใช้อ่าน header ของข้อความแล้วคืนค่ากลับมาเป็น object `imap_header ( resource imap_stream, int msg_number [, int fromlength [, int subjectlength [, string defaulthost]]] )`
  - `imap_stream`: ค่า `imap stream` ที่ได้จากการเชื่อมต่อ
  - `msg_number`: หมายเลขของจดหมาย
4. `imap_fetchstructure` ใช้ในการตรวจสอบ โครงสร้างของข้อความแล้วคืนค่ากลับมาเป็น object `imap_fetchstructure ( resource imap_stream, int msg_number [, int options] )`

5. `imap_delete` ใช้ในการลบข้อความที่กำหนดไว้ใน `msg_number` ออกจาก `imap_stream` ที่ได้จากการเชื่อมต่อ

`imap_delete ( int imap_stream, int msg_number [, int options] )`

6. `imap_body` ทำหน้าที่ในการอ่านข้อความ

`imap_body ( resource imap_stream, int msg_number [, int options] )`



## บทที่ 3

### การวิเคราะห์ ออกแบบและพัฒนาระบบ

#### 3.1 หลักการพัฒนา WAP Application

เนื่องจากการใช้งาน WAP บนโทรศัพท์เคลื่อนที่มีข้อจำกัดต่างๆ เช่นขนาดหน้าจอแสดงผลที่เล็กกว่าคอมพิวเตอร์ ความสามารถของ CPU ขนาดของหน่วยความจำ การป้อนข้อมูล ทำให้การพัฒนา WAP Application มีข้อควรคำนึงถึงดังนี้

- ควรใช้ข้อความที่สั้นกระชับและสื่อความหมาย
- ข้อมูลที่แสดงไม่ควรมีความยาวเกิน 4-5 หน้าจอ
- ใช้รูปหรือสัญลักษณ์ขนาดเล็กแทนข้อความ
- ไม่สามารถแนบเพิ่มข้อมูลไปกับจดหมายได้
- มีการใช้งานเครือข่ายน้อยเพื่อประหยัดค่าใช้จ่าย

#### 3.2 ความต้องการของระบบ

ระบบจดหมายอิเล็กทรอนิกส์ผ่าน WAP มีความต้องการของระบบดังนี้

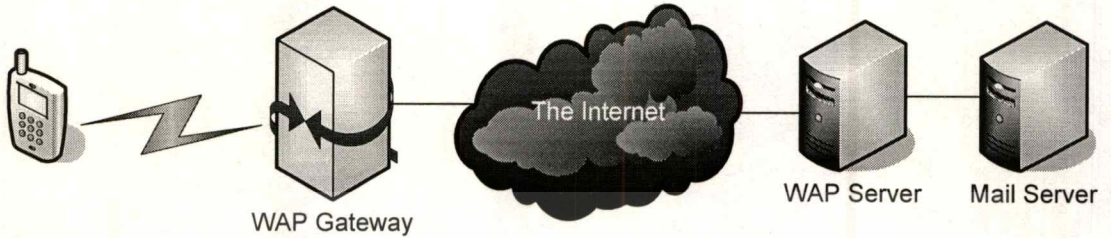
- ใช้งานร่วมกับ Mail Server ที่ใช้อยู่ในปัจจุบันได้
- ใช้งานร่วมกับ Web Server ในปัจจุบันได้
- แสดงผลครั้งละน้อยเพื่อประหยัดค่าใช้จ่ายในการใช้ WAP และให้ผู้ใช้งานปรับแต่งได้
- อ่าน, เขียน, ตอบกลับ และส่งต่อจดหมายอิเล็กทรอนิกส์ได้อย่างถูกต้อง

#### 3.3 สถาปัตยกรรมของระบบ

ในการใช้งาน WAP Mail โดยทั่วไปจะมีลักษณะการทำงานดังนี้ เริ่มจาก WAP Browser บนโทรศัพท์เคลื่อนที่ทำการเรียก URL ที่ต้องการ โทรศัพท์เคลื่อนที่ จะทำการติดต่อไปยัง WAP Gateway ที่ตั้งอยู่ที่ผู้ให้บริการโทรศัพท์เคลื่อนที่ โดย WAP Gateway นี้จะทำการเชื่อมต่อกับเครือข่าย Internet อยู่ WAP Gateway จะทำตัวเป็น Proxy ติดต่อกับ WAP Server ตาม URL ที่ร้อง

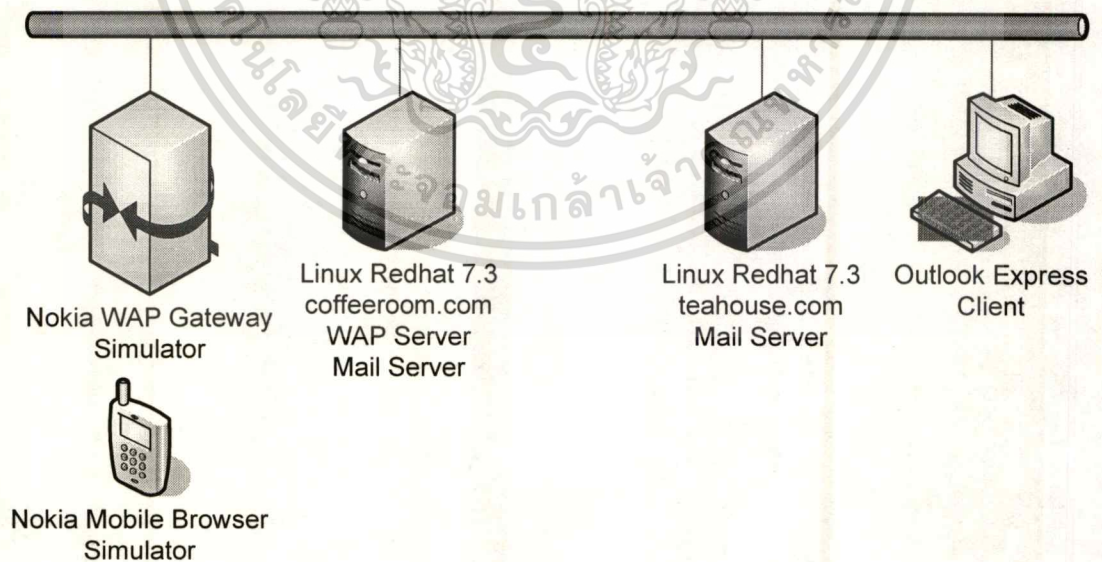
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขอมา WAP Server จะติดต่อกับ Mail Server เพื่อนำจดหมายใน Mail Server มาแสดงผลกลับไปยัง Browser บนโทรศัพท์เคลื่อนที่ ดังแสดงในรูปที่ 3.1



รูปที่ 3.1 การทำงานทั่วไปของ WAP Mail

ในการพัฒนาระบบจดหมายอิเล็กทรอนิกส์ผ่าน WAP นี้ทำการจำลองการทำงานของ WAP Server และ Mail Server โดยการติดตั้ง Linux Redhat 7.3 เพื่อทำหน้าที่เป็น WAP Server และ Mail Server 1 เครื่อง และอีกเครื่องทำหน้าที่เป็น Mail Server เท่านั้น เพื่อทดสอบการรับ-ส่งจดหมายอิเล็กทรอนิกส์ระหว่างกัน และจำลองการทำงานของโทรศัพท์เคลื่อนที่โดยการติดตั้ง Nokia Mobile Browser Simulator และ Nokia WAP Gateway Simulator เพื่อติดต่อกับ WAP Server และใช้งาน Microsoft Outlook Express เพื่อตรวจสอบจดหมายแบบปกติ ดังแสดงในรูปที่ 3.2

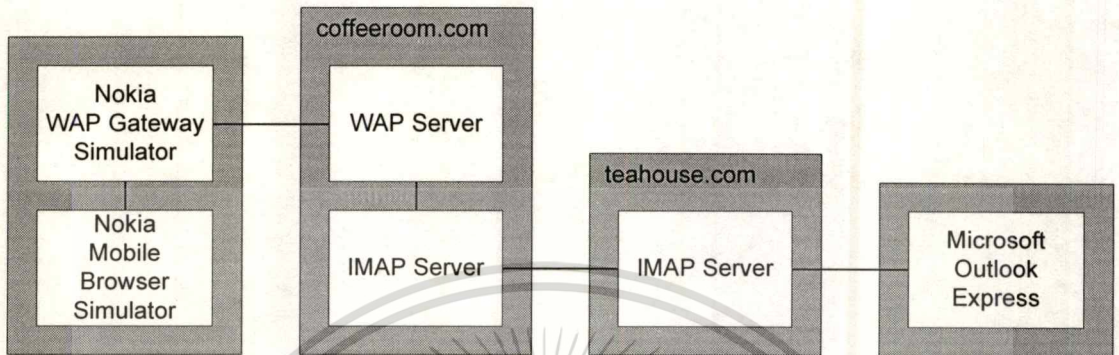


รูปที่ 3.2 สถาปัตยกรรมของการพัฒนาระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4 Logical Component Diagram

ระบบจดหมายอิเล็กทรอนิกส์ผ่าน WAP มีส่วนประกอบต่างๆ ดังแสดงในรูปที่ 3.3 ดังนี้



รูปที่ 3.3 Logical Component Diagram

ในเครื่อง Client ที่จำลองการแสดงผลบนโทรศัพท์เคลื่อนที่ Nokia Mobile Browser Simulator จะติดต่อกับ Nokia WAP Gateway Simulator เท่านั้น ไม่สามารถติดต่อ WAP Server ด้วยตัวเองได้ Nokia WAP Gateway Simulator จะร้องขอมายัง WAP Server ที่ Server coffeeroom.com เพื่อขอข้อมูลการแสดงผล WAP Server จะติดต่อส่วนของ IMAP Server ซึ่งเปิดให้บริการจดหมายอิเล็กทรอนิกส์อยู่และนำจดหมายที่ได้มาแปลงเป็นผลลัพธ์ที่สามารถแสดงได้บนโทรศัพท์เคลื่อนที่ ในการรับ-ส่งจดหมายอิเล็กทรอนิกส์ระหว่าง Server ทั้งสองจะทำการติดต่อผ่าน IMAP Server ของทั้งสองเครื่อง เครื่อง Client อีกเครื่องจะใช้ Microsoft Outlook Express ในการติดต่อกับ IMAP Server บน teahouse.com เพื่อใช้งานต่อไป

### 3.5 การเตรียม Server

การพัฒนาบบจดหมายอิเล็กทรอนิกส์ผ่าน WAP นี้ทำการติดตั้งโปรแกรม VMware เพื่อจำลองการทำงานของ Server 2 เครื่อง โดยใช้ระบบปฏิบัติการและ Application ต่างๆ ดังนี้

- ติดตั้งระบบปฏิบัติการ Linux Redhat 7.3 ทั้ง 2 เครื่อง เครื่องที่หนึ่งให้ชื่อ host เป็น linux และ domain เป็น coffeeroom.com เครื่องที่สองให้ชื่อ domain เป็น teahouse.com
- ติดตั้ง bind ในเครื่องที่หนึ่ง (coffeeroom.com) เพื่อให้บริการ Domain Name Service
- ติดตั้ง imap 2001a เพื่อให้บริการ Mail Service ทั้ง 2 เครื่อง

- เครื่องที่หนึ่งจะติดตั้ง Apache Web Server เพื่อให้บริการ Web และติดตั้ง PHP 4.1.2 เพิ่มเติมเพื่อใช้ในการพัฒนา รวมทั้งติดตั้ง IMAP Library สำหรับ PHP เพื่อใช้ในการติดต่อกับ Mail Server โดยใช้ PHP Script ด้วย
- ติดตั้ง MySQL Server ในเครื่องที่หนึ่งเพื่อใช้สร้างฐานข้อมูลสำหรับปรับแต่งการแสดงผลตามความต้องการของผู้ใช้งาน
- ติดตั้ง Webmin บนเครื่องทั้งสอง เพื่อช่วยในการแก้ไข จัดการ Server การเพิ่มเติม ลบแก้ไข user ในระบบ

### 3.5.1. bind

ทำการเพิ่มข้อมูลลงใน /etc/named.conf ดังนี้

```
zone "coffeeroom.com" IN {
    type master;
    file "coffeeroom.com.zone";
    allow-update { none; };
};
```

แล้วทำการสร้าง /var/named/coffeeroom.com.zone โดยมีข้อมูลดังนี้

```
$TTL 86400
@ IN SOA linux.coffeeroom.com. root.linux.coffeeroom.com. (
    1997022700 ; Serial
    28800 ; Refresh
    14400 ; Retry
    3600000 ; Expire
    86400 ) ; Minimum

    IN NS linux.coffeeroom.com.
    IN MX 10 mail.coffeeroom.com.
linux IN A 192.168.0.100
mail IN CNAME linux
www IN CNAME linux
```

```
wapmail      IN      CNAME      linux
;
```

### 3.5.2. Apache Web Server

ในการใช้งาน Apache Web Server เพื่อให้บริการ WAP บนเครื่องที่หนึ่งต้องมีการกำหนด MIME Type เพิ่มเติมใน /etc/mime.types ดังนี้

```
text/vnd.wap.wml          wml
application/vnd.wap.wmlc  wmlc
text/vnd.wap.wmls        wmls
application/vnd.wap.wmlsc wmlsc
image/vnd.wap.wbmp       wbmp
text/html                 xhtml, htm, html
application/xhtml+xml     xhtml, htm, html
application/vnd.wap.xhtml+xml xhtml, htm, html
```

นอกจากนี้มีการกำหนด Virtual host เพื่อบริการ WAP Mail และ WWW ทั่วไปบน Web Server เครื่องเดียวกัน โดยทำการแก้ไข /etc/httpd/conf/httpd.conf ดังนี้

```
<VirtualHost wapmail.coffeeroom.com>
ServerAdmin webmaster@wapmail.coffeeroom.com
DocumentRoot /var/www/html/wapmail
ServerName wapmail.coffeeroom.com
</VirtualHost>
```

### 3.6 การเตรียม Client

เครื่องมือที่ใช้ในส่วนของ Client มีดังนี้

- ติดตั้ง Nokia WAP Gateway Simulator version 4.0 เพื่อจำลองการทำงานของ WAP Gateway
- ติดตั้ง Nokia Mobile Browser Simulator version 4.0 เพื่อจำลองการแสดงผลที่จะเกิดขึ้น

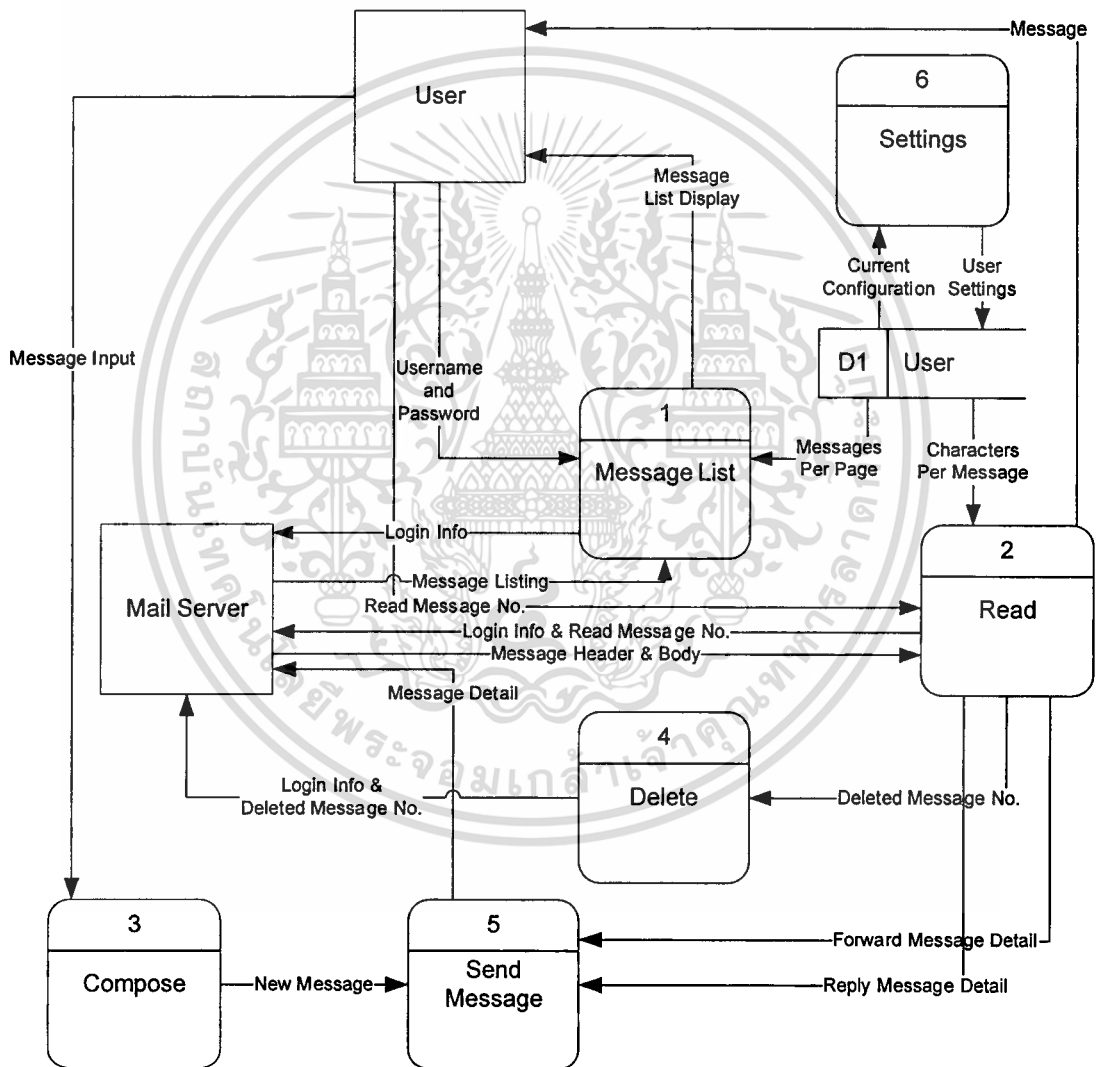
บนโทรศัพท์เคลื่อนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- EditPlus Text Editor ใช้สำหรับแก้ไข Source Code ของระบบ
- Microsoft Outlook Express ใช้ในการทดสอบการรับ-ส่งจดหมายอิเล็กทรอนิกส์กับ Nokia Mobile Browser

### 3.7 Data Flow Diagram

ระบบจดหมายอิเล็กทรอนิกส์ผ่าน WAP มี Data Flow Diagram Level 0 ดังรูปที่ 3.4



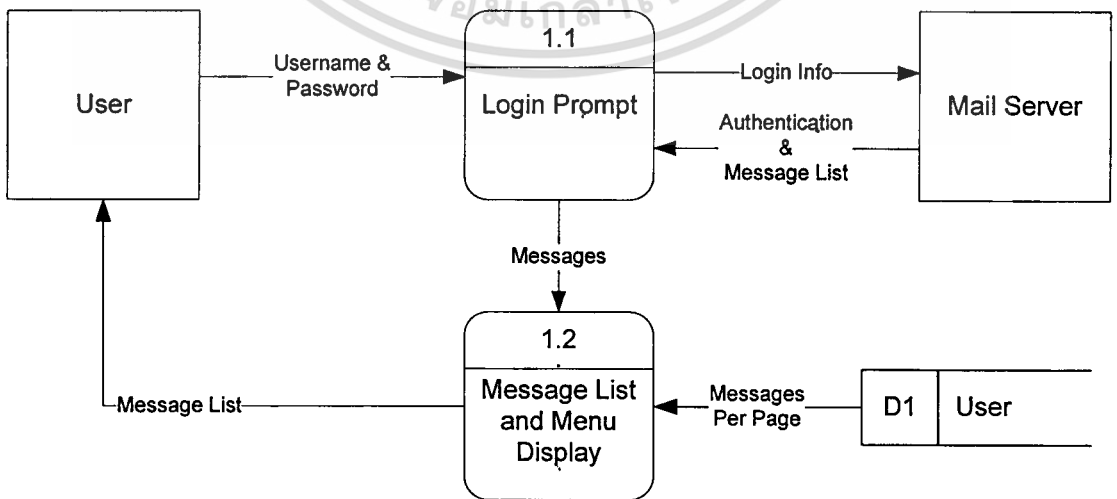
รูปที่ 3.4 Data Flow Diagram Level 0

จากรูปที่ 3.4 ระบบจดหมายอิเล็กทรอนิกส์ผ่าน WAP มีการทำงานต่างๆ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Process Message List จะรับค่า Username และ Password จากผู้ใช้งานมาทำการตรวจสอบและสร้างการเชื่อมต่อกับ Mail Server ด้วย Username และ Password ที่ได้รับ พร้อมกันนั้นจะติดต่อกับฐานข้อมูลเพื่อรับข้อมูลจำนวนจดหมายที่จะแสดงต่อหน้าเพื่อกำหนดจำนวนการแสดงผล และทำการรับรายละเอียดของกล่องจดหมายในส่วนของ Header ต่างๆ มาประมวลผลและแสดงผลให้ผู้ใช้งาน
- Process Read จะเป็นการแสดงรายละเอียดของจดหมายที่ระบุหมายเลขมาจากผู้ใช้งาน รวมถึงการติดต่อกับฐานข้อมูลเพื่อรับข้อมูลจำนวนตัวอักษรที่จะแสดงต่อจดหมายแต่ละฉบับ และทำการติดต่อกับ Mail Server เพื่อรับรายละเอียดของจดหมายเพื่อนำจดหมายฉบับนั้นมาแสดงผล ต่อไป
- Process Compose เป็นการสร้างจดหมายใหม่ที่ได้รับรายละเอียดต่างๆ มาจากผู้ใช้งาน
- Process Delete ทำหน้าที่ลบจดหมายออกจากกล่องจดหมายโดยจะทำการลบจดหมายออกจากกล่องจดหมายทันที
- Process Send Message ทำหน้าที่ส่งจดหมายที่เกิดจากการสร้างจาก Process Compose รวมถึงจดหมายที่ตอบกลับหรือส่งต่อจากการอ่านจดหมายใน Process Read Message
- Process Settings ทำหน้าที่สำหรับรับการตั้งค่าจากผู้ใช้งาน ประกอบด้วยจำนวนจดหมายที่จะแสดงต่อหน้าใน Process Message List และจำนวนตัวอักษรที่จะแสดงผลต่อจดหมายแต่ละฉบับ แล้วนำไปเก็บลงในฐานข้อมูลเพื่อเรียกใช้งานภายหลัง

Process 1 Message List สามารถแสดงรายละเอียดได้ดังนี้

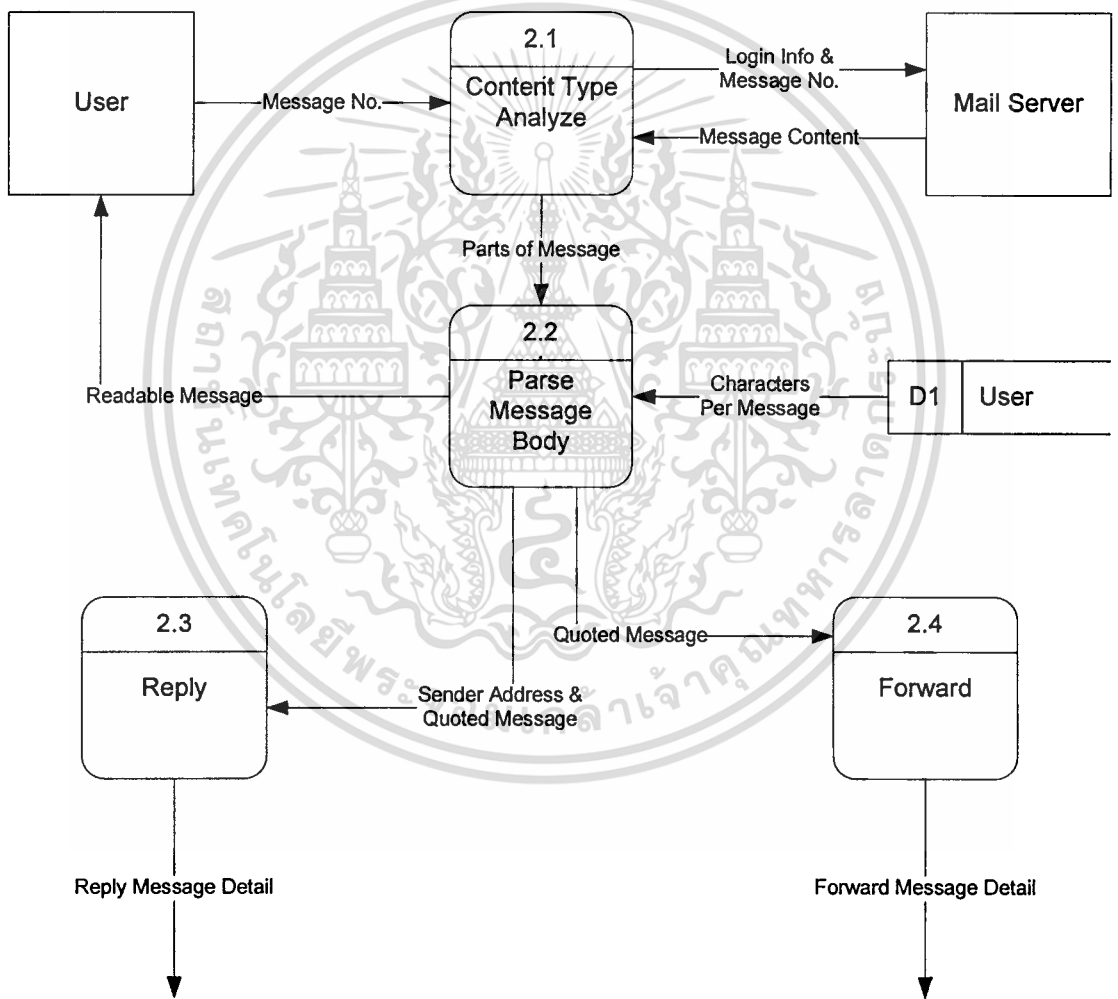


รูปที่ 3.5 Data Flow Diagram Level 1 ของ Process 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดในรูปที่ 3.5 ซึ่งมี Process ย่อยเพื่อทำการเชื่อมต่อกับ Mail Server และผู้ใช้งาน โดยจะรับรายละเอียด Username และ Password จากผู้เข้ามาติดต่อรับจดหมายและนำมาแสดงผล เฉพาะข้อมูลส่วน Header ของจดหมายในกล่องจดหมายเป็นจำนวนฉบับตามที่ได้รับค่ามาจากฐานข้อมูลเพื่อให้ผู้ใช้ได้เลือกอ่านอย่างละเอียดต่อไป

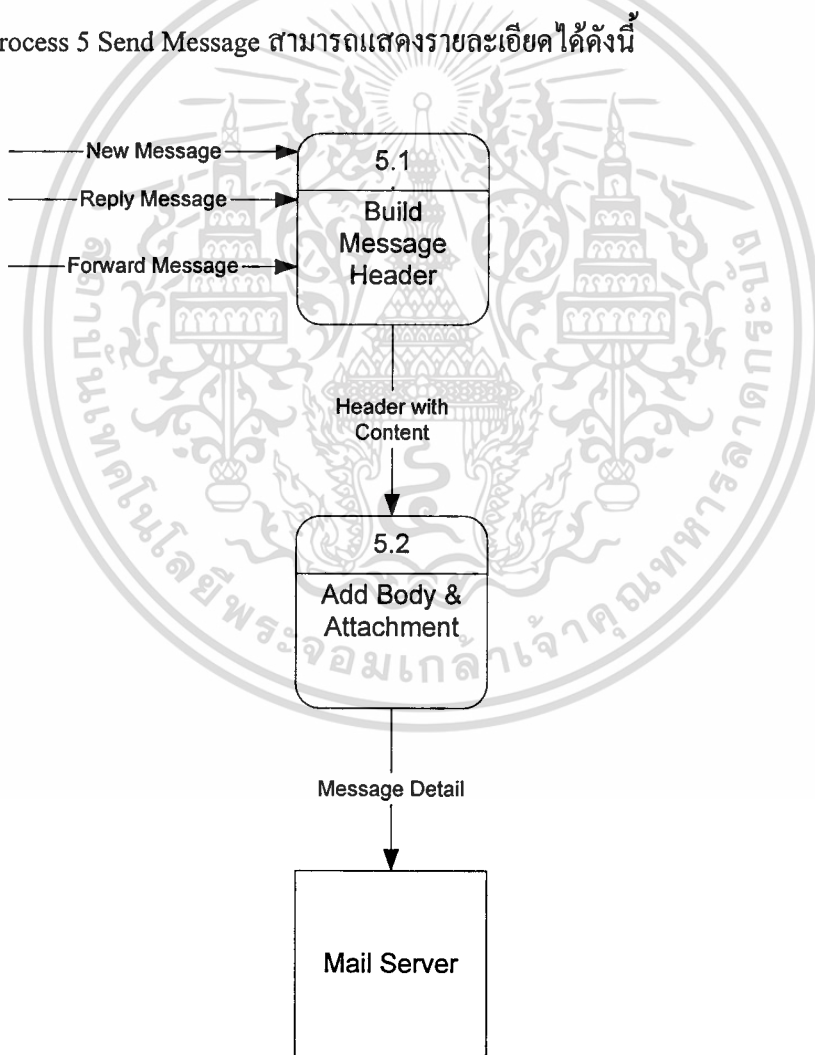
Process 2 Read Message สามารถแสดงรายละเอียดได้ดังนี้



รูปที่ 3.6 Data Flow Diagram Level 1 ของ Process 2

จากรูปที่ 3.6 แสดง Process ย่อยของการอ่านจดหมายคือเมื่อผู้ใช้งานส่งข้อมูลหมายเลขของจดหมายที่ต้องการอ่านมายัง Process Read Message นี้ Process จะนำหมายเลขจดหมายไปอ้างอิงกับ Mail Server และนำเนื้อหาของจดหมายทั้งหมดส่งกลับมายัง Process ย่อย Content Type Analyze เพื่อตรวจสอบว่าจดหมายฉบับนั้นมีความซับซ้อนมากน้อยเพียงใด และวิเคราะห์ Part ต่างๆ ของจดหมายว่ามีการแนบเพิ่มข้อมูลต่างๆ มาด้วยหรือไม่ แล้วจึงทำการแสดงผลให้กับผู้ใช้งานตามจำนวนตัวอักษรที่ได้รับจากฐานข้อมูล นอกจากนี้หากมีการตอบจดหมาย หรือส่งต่อจดหมายฉบับนี้จะมีการอ้างอิงถึงข้อความที่ได้รับมาและรายละเอียดของผู้รับต่อเพื่อส่งไปยัง Process Send Message เพื่อส่งจดหมายต่อไป

Process 5 Send Message สามารถแสดงรายละเอียดได้ดังนี้



รูปที่ 3.7 Data Flow Diagram Level 1 ของ Process 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.7 แสดงการทำงานของขั้นตอนการส่งจดหมายเริ่มจากการรับข้อความทั้งที่ได้จากการสร้างขึ้นใหม่ การตอบกลับ หรือการส่งต่อมาสร้าง Header และตัวจดหมายรวมไปถึงเพิ่มข้อมูลที่ส่งต่อมาด้วย หลังจากนั้นจะติดต่อกับ Mail Server เพื่อส่งจดหมายนี้ออกไป

### 3.8 ฐานข้อมูลที่ใช้ในระบบ

จากความต้องการของระบบที่ต้องการให้ผู้ใช้งานสามารถปรับแต่งจำนวนและความยาวของการแสดงผลได้นั้น การพัฒนาระบบจึงมีการใช้งานฐานข้อมูลเพื่อเก็บข้อมูลต่างๆ โดยมีโครงสร้างดังตารางที่ 3.1 โดยการออกแบบใช้ MySQL เป็นฐานข้อมูลดังนั้นชนิดและความยาวของข้อมูลในแต่ละ Field จะเป็นดังนี้

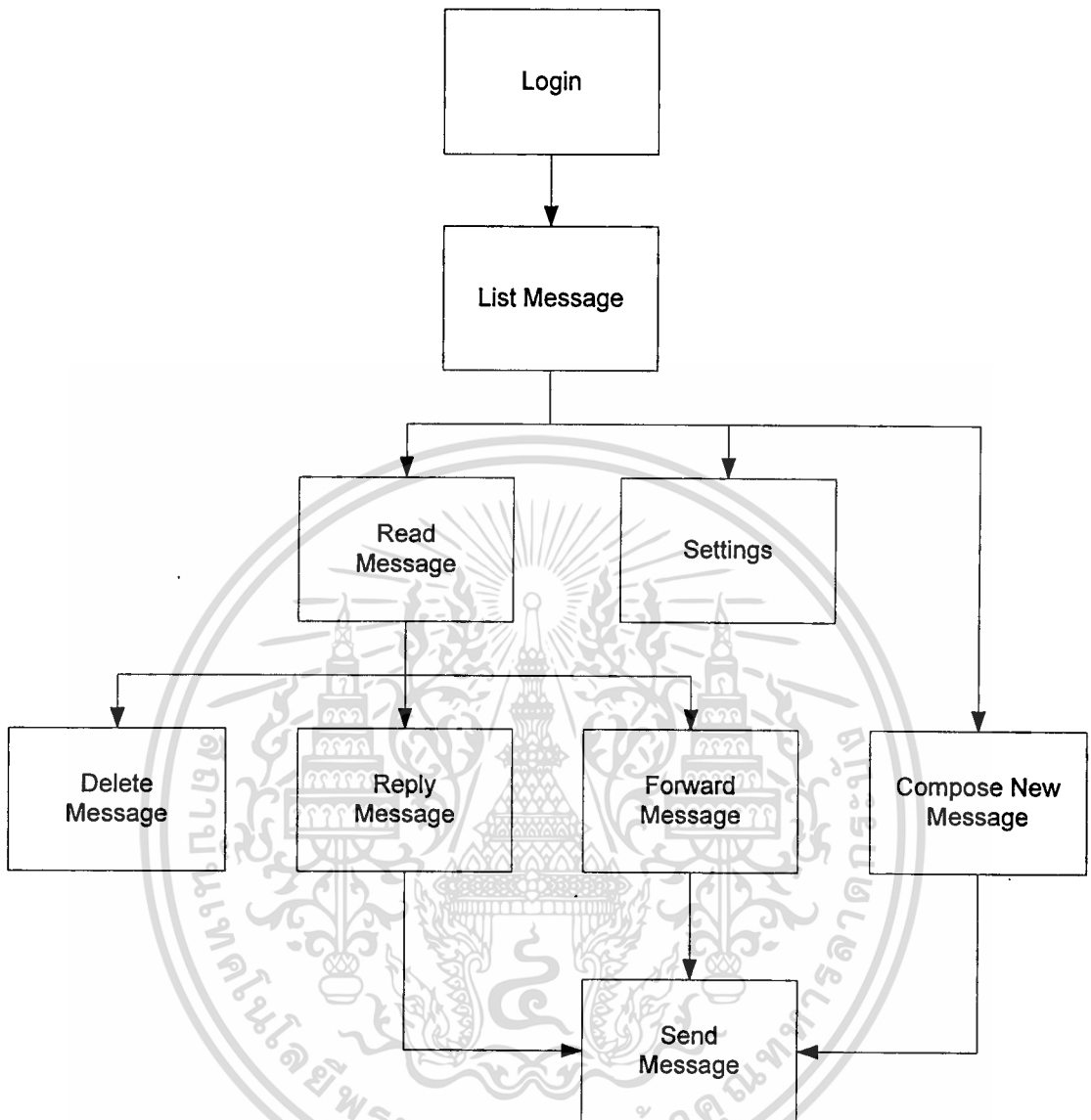
- uname : เก็บชื่อของผู้ใช้งานเป็นข้อความที่มีความยาวไม่เกิน 15 ตัวอักษร
- ppage : เก็บจำนวนจดหมายที่จะนำมาแสดงผลต่อหน้าในหน้าหลัก ชนิดของข้อมูลเป็น TINYINT สามารถเก็บข้อมูลตัวเลขแบบ Unsigned ได้ตั้งแต่ 0 ถึง 255
- pmsg : เก็บจำนวนตัวอักษรที่จะแสดงต่อจดหมายหนึ่งฉบับ ชนิดของข้อมูลเป็น SMALLINT สามารถเก็บข้อมูลตัวเลขแบบ Unsigned ได้ตั้งแต่ 0 ถึง 65535

ตารางที่ 3.1 ตารางเก็บข้อมูลการแสดงผลของผู้ใช้งาน (User)

Field Name	Type	Length	Key	Description
uname	TEXT	15	PK	ชื่อผู้ใช้งาน
ppage	TINYINT	3		จำนวนจดหมายต่อหน้า
pmsg	SMALLINT	5		จำนวนตัวอักษรต่อจดหมายหนึ่งฉบับ

### 3.9 ภาษาที่ใช้ในการพัฒนาระบบและฟังก์ชันต่างๆ

การพัฒนาระบบได้เลือกใช้ PHP เป็นภาษาที่ใช้พัฒนา โดยทำการแบ่งการทำงานออกเป็นฟังก์ชันต่างๆ ในระบบดังรูปที่ 3.8



รูปที่ 3.8 ฟังก์ชันต่างๆ ในระบบ

ลักษณะการทำงานของระบบเริ่มจากการรอรับข้อมูลชื่อผู้ใช้และรหัสผ่านจากผู้ใช้งาน โดยการกรอกฟอร์ม Login เมื่อได้รับข้อมูลจะทำการตรวจสอบว่าเป็นผู้มีสิทธิใช้งาน Mail Server หรือไม่หากไม่พบข้อมูลหรือไม่สามารถติดต่อ Mail Server ได้ระบบจะทำการแสดงผลความผิดพลาดที่เกิดขึ้น เมื่อตรวจสอบแล้วพบว่าชื่อและรหัสผ่านถูกต้องและทำการติดต่อ Mail Server ได้แล้วฟังก์ชัน List Message จะตรวจสอบกล่องจดหมายของผู้ใช้ และฐานข้อมูลเพื่อรับค่าจำนวนจดหมายที่จะแสดงผลต่อหน้า หลังจากนั้นจะนำค่า Header ที่ได้มาแสดงผลจดหมายที่มีบนหน้าจอ รวมถึงการแสดงผลเมนูต่างๆ ให้ผู้ใช้เลือกใช้งาน ในหน้าจอหลักนี้ผู้ใช้สามารถทำการสร้างจดหมาย เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นับญาติให้นำไปเผยแพร่ขึ้นด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใหม่โดยจะเรียกการทำงานของฟังก์ชัน Compose New Message ซึ่งจะแสดงแบบฟอร์มรับรายละเอียดของจดหมายที่ต้องการสร้างและส่งไปยังฟังก์ชัน Send Message ในหน้าจอหลักนั้นผู้ใช้สามารถเลือกอ่านจดหมายฉบับที่ต้องการได้ โดยหลังจากผู้ใช้เลือกจดหมายที่ต้องการอ่านจากหน้าจอหลักฟังก์ชัน Read Message จะติดต่อกับ Mail Server และฐานข้อมูลเพื่อนำจดหมายฉบับนั้นขึ้นมาแสดงผล โดยมีเงื่อนไขว่าหากจดหมายฉบับนั้นมีความยาวมากกว่าจำนวนตัวอักษรที่จะแสดงผลต่อจดหมายแต่ละฉบับที่ได้มาจากฐานข้อมูล ฟังก์ชัน Read Message จะแสดงผลเฉพาะส่วนต้นๆ ของจดหมายหากผู้ใช้มีความต้องการอ่านข้อความเต็มก็สามารถเลือกได้ในหน้านี้ แต่หากจดหมายมีความยาวไม่มากกว่าที่กำหนดไว้ก็จะแสดงผลข้อความทั้งหมดโดยอัตโนมัติรวมถึงแสดงรายชื่อแฟ้มข้อมูลที่ถูกแนบมาด้วยแต่ไม่สามารถแสดงผลบนหน้าจอ โทรศัพท์เคลื่อนที่ได้ในปัจจุบันนี้ เมื่อผู้ใช้อ่านจดหมายแล้ว ผู้ใช้สามารถเลือกที่จะตอบจดหมาย, ส่งต่อ หรือลบจดหมายทิ้ง โดยระบบจะไปเรียกใช้งานฟังก์ชัน Reply Message, Forward Message หรือ Delete Message ต่อไป เมื่อผู้ใช้เลือกลบจดหมายฟังก์ชัน Delete Message จะทำการลบจดหมายจากกล่องจดหมายใน Mail Server ในทันที หากผู้ใช้ต้องการตอบจดหมายฟังก์ชัน Reply Message จะสร้างฟอร์มในการสร้างจดหมายแต่จะระบุชื่อผู้รับจดหมายนี้เป็นชื่อของผู้ที่ส่งจดหมายนี้มาและเพิ่ม Re: หน้า Subject เดิมให้โดยอัตโนมัติ รวมทั้งนำเนื้อความในจดหมายมาแสดงผลเพื่อรอการเพิ่มเติมหรือแก้ไขเพื่อส่งไปให้ฟังก์ชัน Send Message ทำงานต่อ หรือหากผู้ใช้มีความต้องการส่งต่อจดหมายฟังก์ชัน Forward Message จะสร้างฟอร์มที่มีลักษณะคล้าย Reply Message ขึ้นมาแต่จะไม่มีชื่อผู้รับและเพิ่ม Fw: หน้า Subject เดิมรวมทั้งหากมีการแนบแฟ้มข้อมูลมาก็สามารถส่งต่อแฟ้มข้อมูลนั้นได้ด้วย หลังจากนั้นฟังก์ชัน Send Message จะถูกเรียกขึ้นมาใช้งาน

ฟังก์ชัน Send Message จะเป็นการนำข้อมูลที่ได้รับจากฟังก์ชัน Compose New Message, Reply Message และ Forward Message มาประมวลผลเพื่อเพิ่ม Header ของจดหมายและตรวจสอบตัวจดหมายแล้วทำการติดต่อกับ Mail Server เพื่อทำการส่งออก

ฟังก์ชัน Settings เป็นการปรับแต่งค่าในการแสดงผลจำนวนจดหมายต่อหน้า และจำนวนตัวอักษรต่อจดหมายแต่ละฉบับ และทำการเก็บลงในฐานข้อมูลเพื่อเรียกใช้งานภายหลัง

### 3.10 การแสดงผลบน Nokia Mobile Browser Simulator

การใช้งานระบบจดหมายอิเล็กทรอนิกส์ผ่าน WAP ทำได้โดยพิมพ์ URL ใน Nokia Mobile Browser Simulator เป็น <http://wapmail.coffeeroom.com> ดังรูปที่ 3.9



รูปที่ 3.9 การเข้าระบบ

เมื่อกดปุ่ม Load จะปรากฏหน้าจอ Login ดังรูปที่ 3.10 ใส่ชื่อผู้ใช้งานและรหัสผ่านแล้วกดปุ่ม OK



รูปที่ 3.10 หน้าจอ Login

เมื่อระบบตรวจสอบผู้ใช้งานเรียบร้อยแล้วหากเกิดข้อผิดพลาดจะพบข้อความดังรูปที่ 3.11 แต่หากไม่เกิดข้อผิดพลาดระบบจะแสดงจดหมายในกล่องจดหมายดังรูปที่ 3.12



รูปที่ 3.11 รายงานข้อผิดพลาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 จดหมายที่อยู่ในกล่องจดหมาย

รูปที่ 3.12 เป็นหน้าจอหลักที่มีรายละเอียดดังนี้

- แสดงชื่อผู้ใช้งานที่มุมบนขวา
- แสดงจดหมายเรียงตามวัน เวลาที่ได้รับ โดยจะแสดงจดหมายที่มาถึงหลังสุดก่อน
- แบ่งการแสดงผลจดหมายออกเป็นหน้า ขึ้นอยู่กับจำนวนจดหมายที่ปรับแต่งจากหน้า Settings หากมีจดหมายมากกว่าที่กำหนดสามารถคลิก Next หรือ Previous เพื่อแสดงหน้าอื่นๆ
- แสดงจำนวนจดหมายที่ยังไม่ได้เปิดอ่านเป็นจำนวนตัวเลขหลังข้อความ Inbox
- จดหมายที่ยังไม่ได้เปิดอ่านจะแสดงบนพื้นสีเข้มกว่าจดหมายที่เปิดอ่านแล้ว
- จดหมายที่มีแฟ้มแนบมาด้วยจะมีรูปตัวหนีบกระดาษแสดงอยู่หน้าขนาดของจดหมาย
- สามารถสร้างจดหมายใหม่ได้จากการเลือก Compose

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การปรับแต่งระบบทำได้โดยเลือก Settings ผู้ใช้งานสามารถกำหนดจำนวนจดหมายต่อหน้าและความยาวของจดหมายที่จะอ่านได้ หากเป็นการใช้งานครั้งแรกระบบจะตั้งค่าเริ่มต้นเป็น 5 ฉบับต่อหน้า และความยาวในการอ่านจดหมายเป็น 300 ตัวอักษรต่อฉบับ ดังแสดงในรูปที่ 3.13



รูปที่ 3.13 การปรับแต่งระบบ

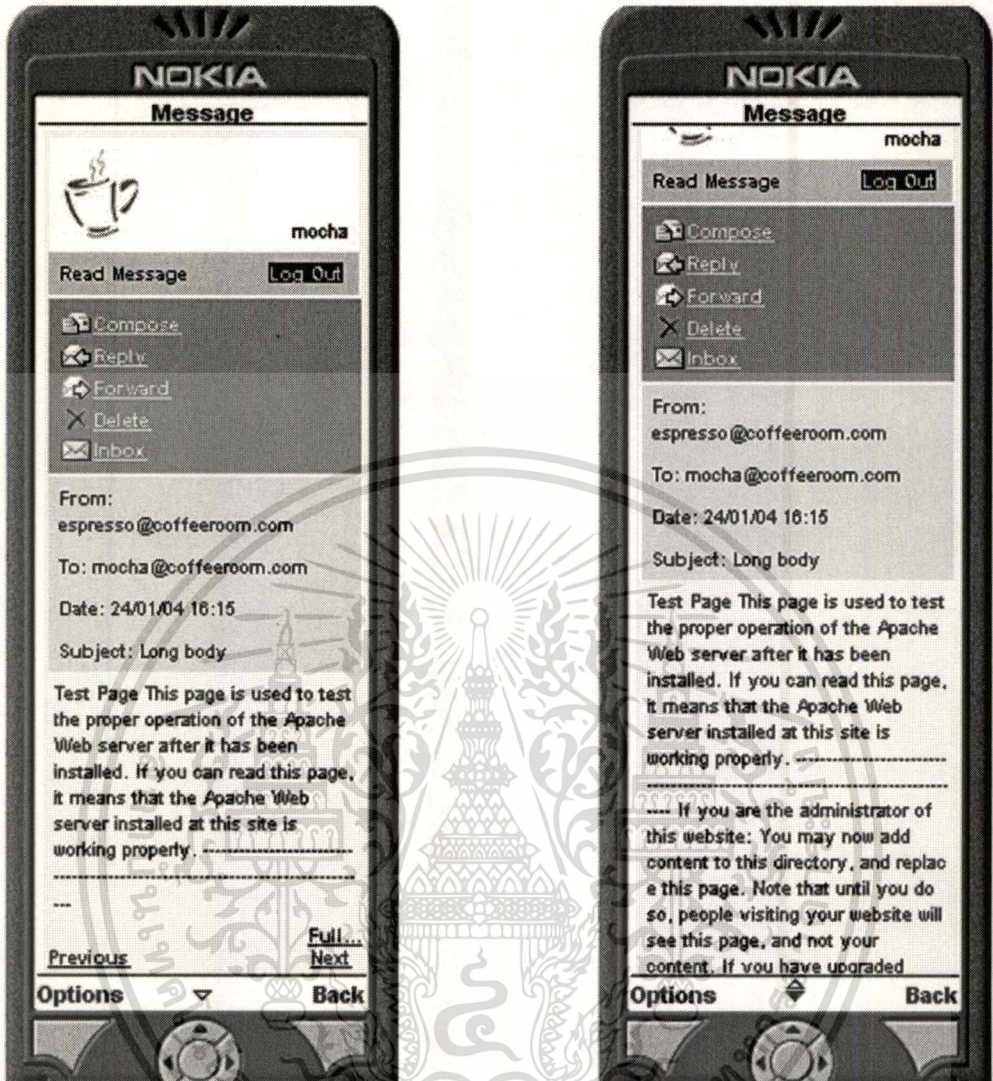
การเปิดอ่านจดหมายทำได้โดยเลือกหัวข้อจดหมาย ระบบจะแสดงข้อความในจดหมายดังรูปที่ 3.14 หากมีแฟ้มแนบมาด้วยจะแสดงให้เห็นขนาดและชื่อแฟ้มแต่ไม่สามารถเปิดได้เนื่องจากเป็นข้อจำกัดของโทรศัพท์เคลื่อนที่



รูปที่ 3.14 ข้อความในจดหมายที่มีแฟ้มแนบมาด้วย

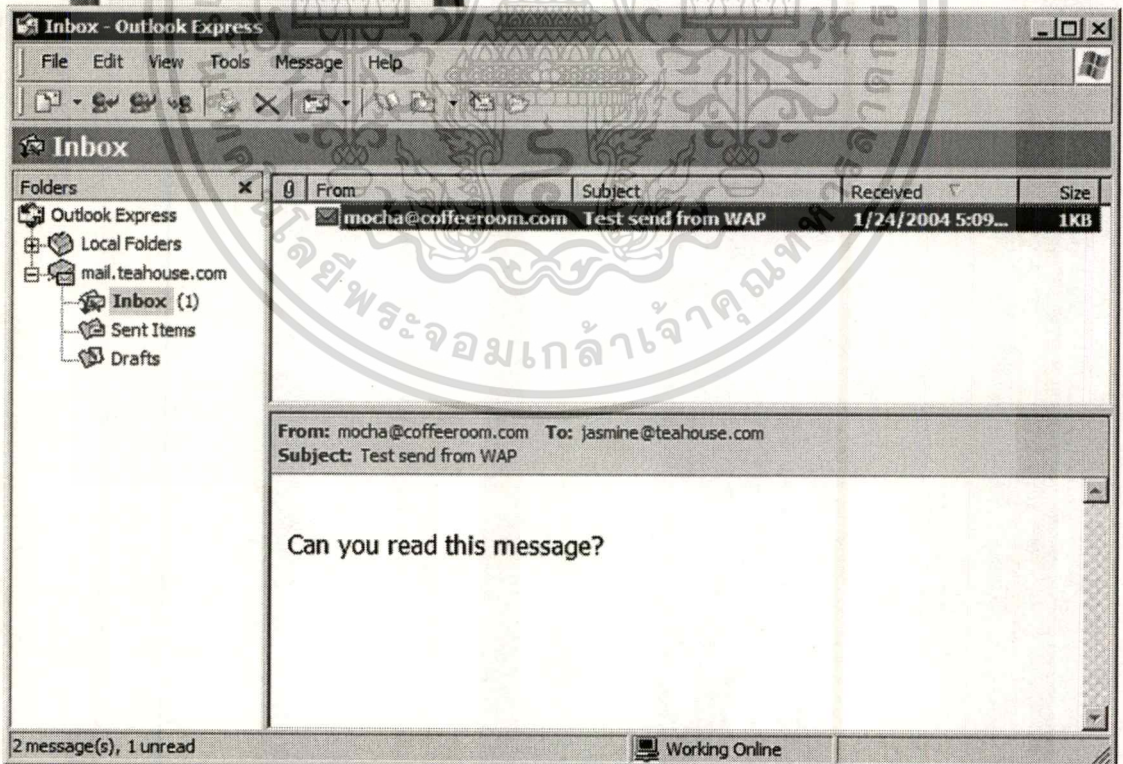
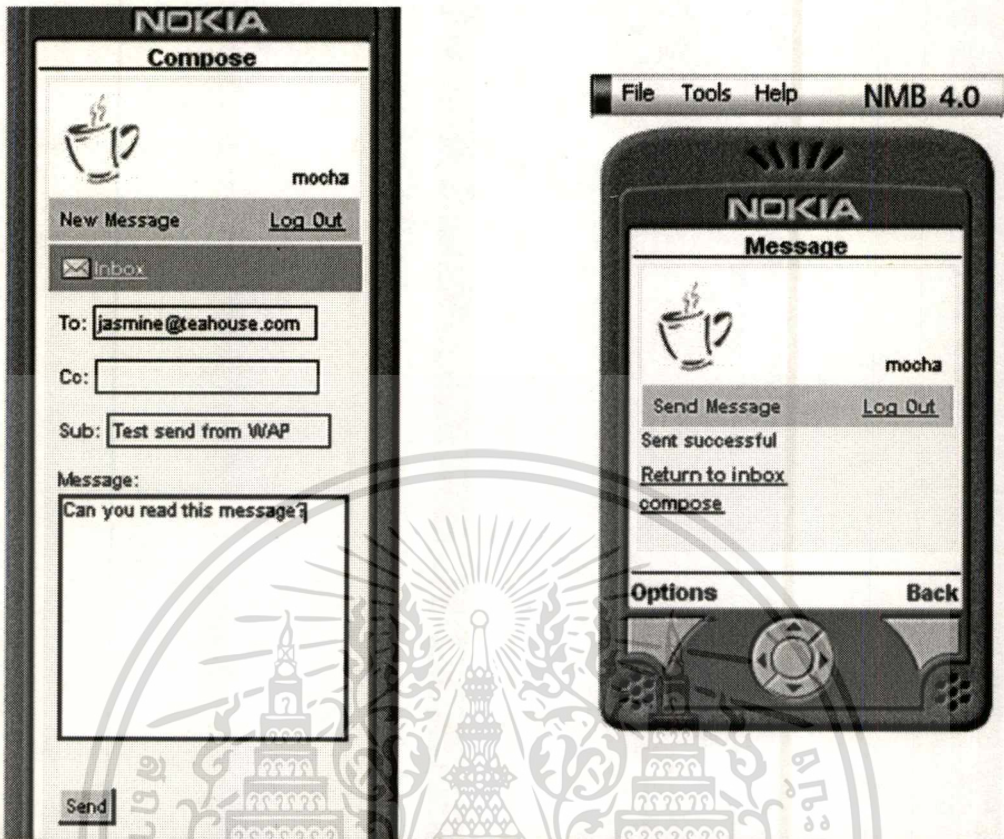
ผู้ใช้งานสามารถอ่านจดหมายฉบับอื่นๆ ได้โดยการเลือก Previous หรือ Next หรืออาจกลับไปหน้าหลักโดยการเลือก Inbox

หากจดหมายที่เปิดอ่านมีความยาวมากกว่าจำนวนตัวอักษรที่ปรับแต่งจากหน้า Settings ระบบจะทำการแสดงเฉพาะส่วนต้นของจดหมายโดยแสดงข้อความที่มีความยาวที่ปรับแต่งไว้ หากผู้ใช้งานมีความต้องการเปิดอ่านจดหมายฉบับเต็มจะทำได้โดยการคลิก Full... ดังแสดงในรูปที่ 3.15



รูปที่ 3.15 ข้อความในจดหมายแบบย่อและแบบเต็ม

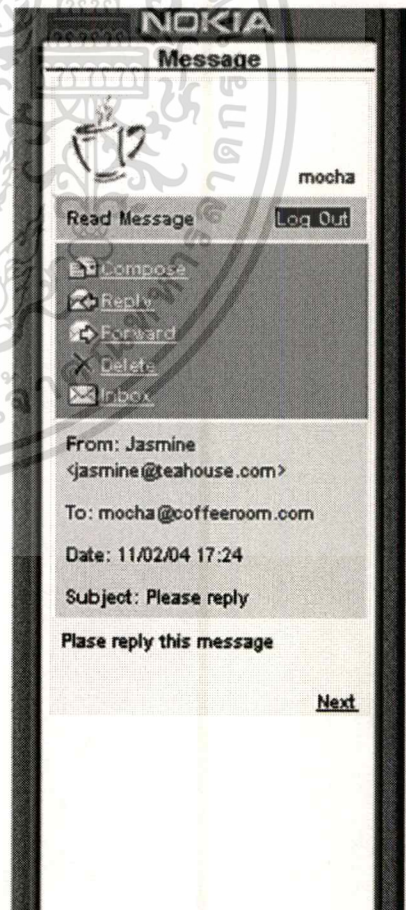
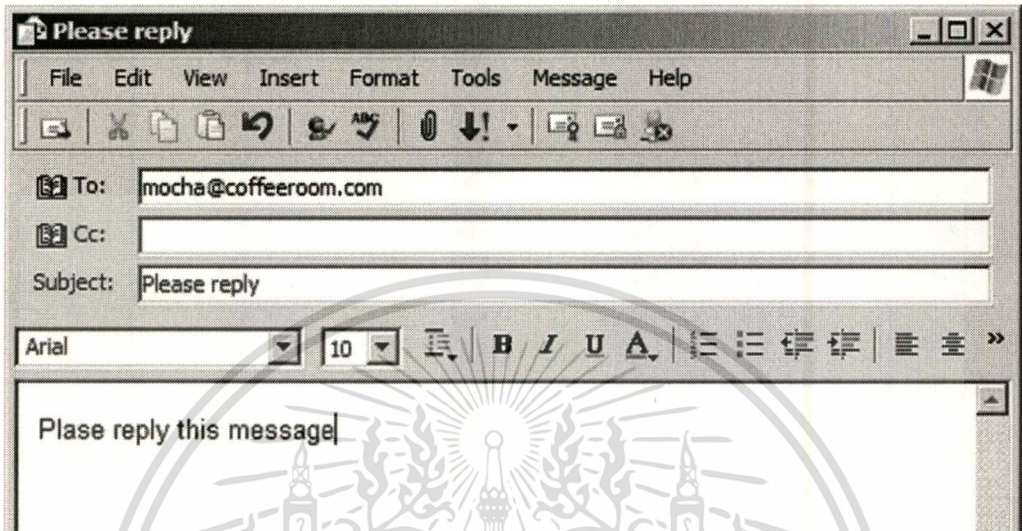
เมื่อต้องการสร้างจดหมายใหม่ทำได้โดยการเลือก Compose ที่หน้าต่างๆ ระบบจะสร้างฟอร์มรับข้อความ เมื่อกด Send ระบบจะทำการส่งจดหมายและแจ้งให้ทราบว่าจดหมายส่งออกสำเร็จ ผู้ใช้งานสามารถกลับไปหน้าจอหลักหรือสร้างจดหมายฉบับใหม่อีกได้ และเมื่อทดสอบการรับจดหมายที่ส่งออกจากระบบด้วยโปรแกรม Outlook Express พบว่าสามารถได้รับข้อความอย่างถูกต้องดังรูปที่ 3.16



รูปที่ 3.16 การสร้างจดหมายใหม่และทดสอบการรับด้วย Outlook Express

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

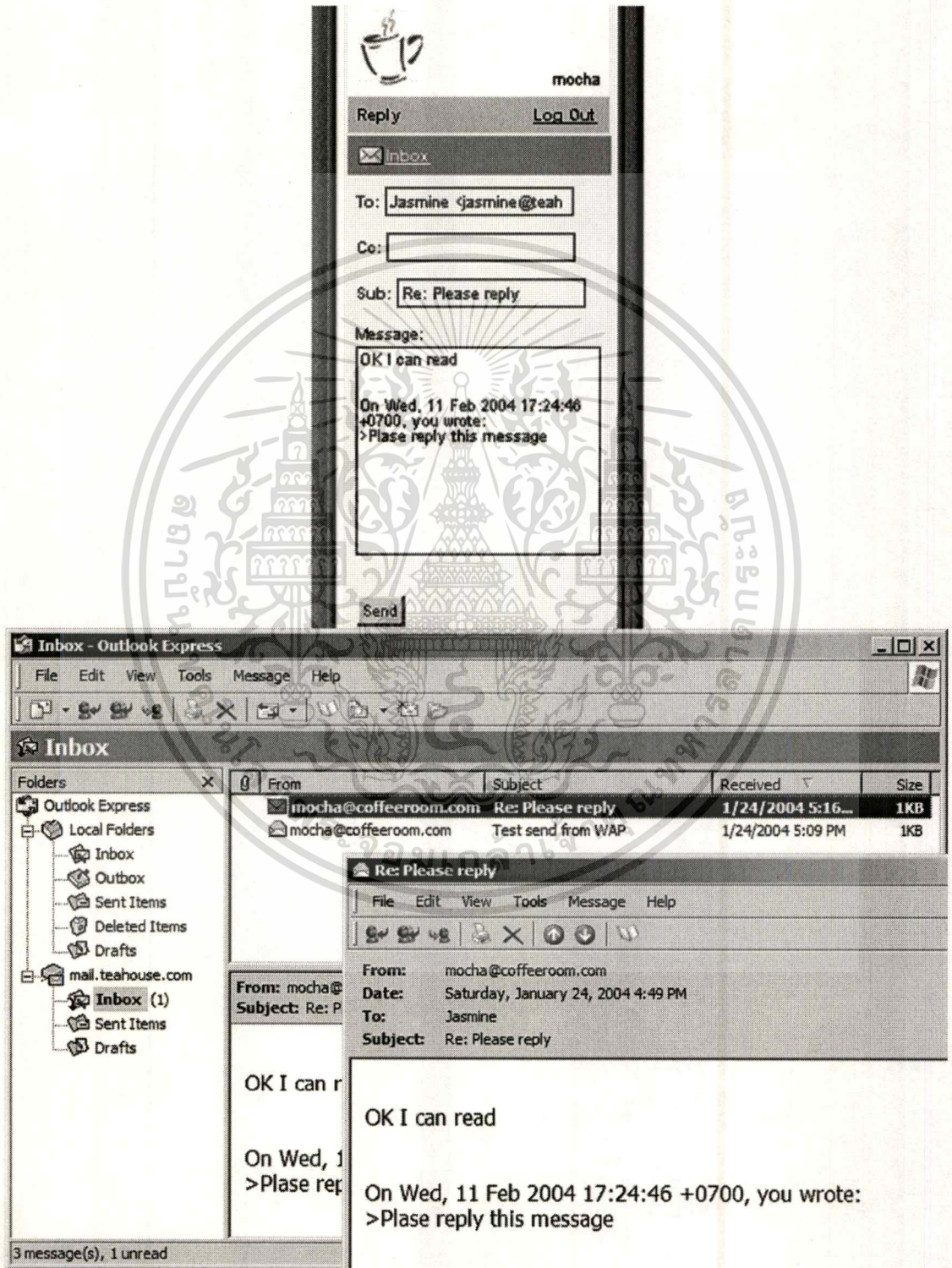
ทดสอบการรับจดหมายจากโปรแกรม Outlook Express แล้วทำการอ่านข้อความในจดหมายพบว่าสามารถอ่านได้ดังรูปที่ 3.17



รูปที่ 3.17 จดหมายที่สร้างจาก Outlook Express และเปิดอ่านจาก Browser

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การตอบจดหมายสามารถเลือก Reply ในหน้าจดหมายที่อ่านอยู่ ระบบจะทำการสร้างฟอร์มเพื่อตอบจดหมายดังกล่าวไปยังผู้ส่งดังรูปที่ 3.18 .



รูปที่ 3.18 การตอบจดหมายและเปิดอ่านจาก Outlook Express

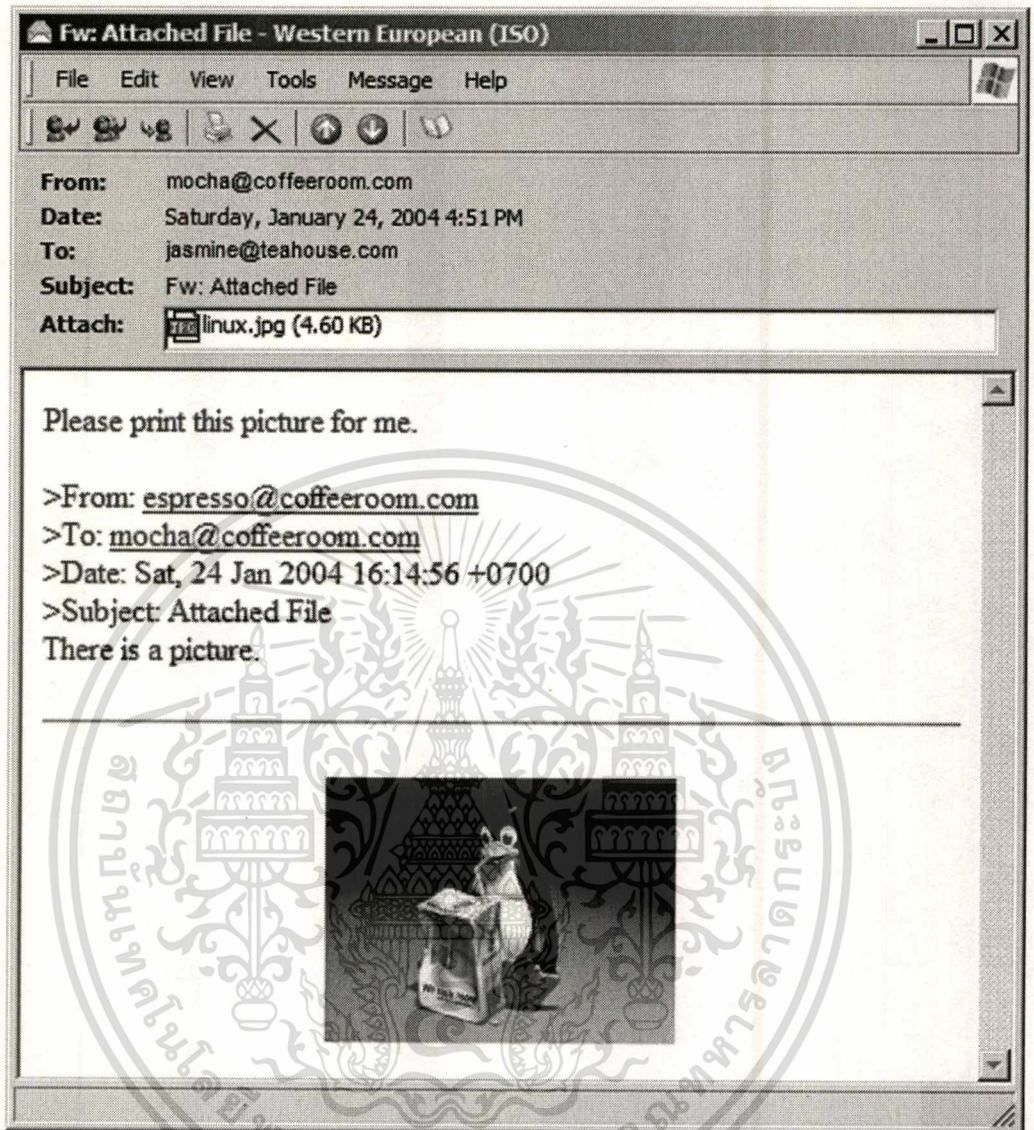
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การส่งต่อจดหมายทำได้โดยเลือก Forward ระบบจะสร้างฟอร์มสำหรับส่งต่อข้อความ รวมทั้งหากมีเพิ่มข้อมูลแนบมาด้วยก็สามารถส่งต่อไปได้เช่นกัน แต่ไม่สามารถแนบเพิ่มข้อมูลอื่นเพิ่มได้ (เนื่องจากเป็นข้อจำกัดของโทรศัพท์เคลื่อนที่) ดังแสดงในรูปที่ 3.19



รูปที่ 3.19 การส่งต่อจดหมายพร้อมเพิ่มข้อมูลที่แนบมา

เมื่อเปิดอ่านจาก Outlook Express สามารถรับเพิ่มข้อมูลที่แนบมาด้วยได้อย่างถูกต้องดังรูปที่ 3.20



รูปที่ 3.20 การเปิดอ่านจดหมายที่แนบเพิ่มข้อมูลมาด้วย

## บทที่ 4

### สรุปการพัฒนา

โครงการพัฒนาระบบงานนี้จัดทำขึ้นเพื่อเสนอให้เห็นประโยชน์การนำเทคโนโลยีที่มีอยู่มาประยุกต์ใช้งานเพื่อเพิ่มประสิทธิภาพในการทำงาน โดยใช้ทรัพยากรที่มีอยู่ให้เป็นประโยชน์มากที่สุด

#### 4.1 สรุปผลการดำเนินงาน

การพัฒนาระบบจดหมายอิเล็กทรอนิกส์ผ่าน WAP นี้เกิดขึ้นเพื่อช่วยเพิ่มช่องทางการใช้งานจดหมายอิเล็กทรอนิกส์ที่ใช้ในองค์กรให้มีความคล่องตัวมากขึ้น เนื่องจากโทรศัพท์เคลื่อนที่ในปัจจุบันมีความสะดวกเพิ่มขึ้น มีการใช้งานอย่างแพร่หลาย ทำให้สามารถใช้งานจดหมายอิเล็กทรอนิกส์นอกสำนักงานได้ ทั้งนี้ยังคำนึงถึงการประหยัดค่าใช้จ่ายในการใช้งาน WAP โดยทำการแสดงข้อมูลครั้งละจำนวนน้อย ป้องกันความสับสนในการจำ e-mail address เนื่องจากใช้เพียงที่เดียว และการตรวจสอบทำเพียงที่เดียว รวมถึงแสดงความเป็นมืออาชีพขององค์กร

จากการพัฒนาระบบพบว่าสามารถใช้งานได้ดีในระดับหนึ่งคือ สามารถรับ-ส่งจดหมายที่เป็นข้อความทั่วไปได้อย่างถูกต้อง การแสดงผลบนโทรศัพท์เคลื่อนที่มีความสวยงามขึ้น ถึงแม้ว่าจะต้องออกแบบให้แสดงเฉพาะส่วนที่จำเป็นโดยคำนึงถึงข้อจำกัดของโทรศัพท์เคลื่อนที่ สามารถติดต่อกับ Mail Server ที่ใช้งานโดยไม่จำเป็นต้องแก้ไขการทำงานอื่นๆ ไม่มีผลกระทบต่อการใช้งานในปัจจุบันเพราะยังสามารถใช้งานจดหมายอิเล็กทรอนิกส์แบบปกติได้จากการใช้งานโปรแกรมอ่านจดหมายที่ใช้อยู่ทั่วไปเช่น Outlook Express หรือ Eudora

#### 4.2 ข้อเสนอแนะ

ระบบจดหมายอิเล็กทรอนิกส์ผ่าน WAP ที่พัฒนาขึ้นนี้มีความสามารถที่จำกัดเนื่องจากในปัจจุบันยังไม่สามารถเปิดเอกสารที่แนบมากับจดหมายได้ และไม่สามารถแนบเอกสารไปกับจดหมายฉบับใหม่ได้ แต่การส่งต่อเอกสารที่แนบมากับจดหมายนั้นสามารถทำได้ การนำไปพัฒนาต่อควรคำนึงถึงข้อจำกัดต่างๆ คือหน่วยความจำค่อนข้างน้อย หน้าจอแสดงผลขนาดเล็ก การพิมพ์ข้อความที่มีความยาวทำได้ยาก ซึ่งในปัจจุบันผู้ผลิตและหน่วยงานต่างๆ ได้พัฒนาการใช้โทรศัพท์มือถือเพื่อเชื่อมต่อกับ Internet ให้ดียิ่งขึ้น โก้ถักการใช้งานจากคอมพิวเตอร์มากขึ้น การ

พัฒนาระบบนี้ต่อไปในอนาคตอาจทำให้การใช้งานจดหมายอิเล็กทรอนิกส์ง่ายขึ้น มีความสามารถ และประสิทธิภาพมากขึ้นได้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม

- กอบเกียรติ สระอุบล. 2545. **สร้างเว็บเพจด้วย PHP ฉบับประยุกต์ เล่ม 2**. กรุงเทพฯ: บี อี แอนด์ ซี. กิตติ ภัคดีวัฒน์กุล และคณะ. 2545. **PHP ฉบับโปรแกรมเมอร์**. พิมพ์ครั้งที่ 2. กรุงเทพฯ: เททีพี คอมพ์ แอนด์ คอนซัลท์.
- ไพโรจน์ ไววนิชกิจ. 2546. "I-MODE กับการซื้อขายของประวัติศาสตร์มือถือ." **Internet Magazine**. 12(7): 107-109.
- ภัทระ เกียรติเสวี. 2542. **สร้างอินเทอร์เน็ตเซิร์ฟเวอร์ด้วย Linux**. พิมพ์ครั้งที่ 2. กรุงเทพฯ: ซีเอ็ดยูเคชั่น.
- Massachusetts Institute of Technology. 2001. **IMAP at MIT: Beginner's Guide to IMAP**. [Online]. Available: <http://web.mit.edu/is/help/imap/beginner.html>.
- Nokia Corporation. 2002. **XHTML White Paper**. [Online]. Available: <http://www.forum.nokia.com>.
- Wireless Application Protocol Forum. 2002. **WAP 2.0 Technical White Paper**. [Online]. Available: <http://www.wapforum.org>.

## ภาคผนวก ก

### การติดตั้ง Web Server และ Mail Server บน Linux Redhat 7.3

ระบบจดหมายอิเล็กทรอนิกส์ผ่าน WAP ใช้ Linux Redhat 7.3 เป็นระบบปฏิบัติการ หลังจากติดตั้ง Linux Redhat 7.3 ตามปกติเรียบร้อยแล้ว จะต้องทำการติดตั้งส่วนประกอบต่างๆ เพิ่มเติมดังนี้

#### IMAP Mail Server

ติดตั้ง IMAP Mail Server เพื่อให้บริการ E-Mail มีขั้นตอนดังนี้

- ใส่ Linux Redhat 7.3 CD แผ่นที่ 1 แล้วใช้คำสั่ง
 

```
# mount /mnt/cdrom/
# cd /mnt/cdrom/RedHat/RPMS/
# rpm -Uvh m4-1.4.1-7.i386.rpm
```
- เปลี่ยน CD เป็นแผ่นที่ 2 แล้วใช้คำสั่ง
 

```
# mount /mnt/cdrom/
# cd /mnt/cdrom/RedHat/RPMS/
# rpm -Uvh sendmail-cf-8.11.6-15.i386.rpm
# rpm -Uvh imap-2001a-10.i386.rpm
```
- ปรับแต่งระบบโดยใช้คำสั่ง
 

```
# vi /etc/mail/sendmail.mc
```

 เพิ่มคำว่า `dnl` หน้าบรรทัด `DAEMON_OPTIONS`

```
dnl DAEMON_OPTIONS(`Port=smtp,Addr=127.0.0.1, Name=MTA')
```

 แล้วบันทึก
- สร้างแฟ้ม `sendmail.cf` โดยใช้คำสั่ง
 

```
# m4 /etc/mail/sendmail.mc > /etc/sendmail.cf
```
- เพิ่มบรรทัดในแฟ้ม `/etc/mail/access` 2 บรรทัดล่าง
 

```
# Check the /usr/share/doc/sendmail/README.cf file for a description
# of the format of this file. (search for access_db in that file)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
# The /usr/share/doc/sendmail/README.cf is part of the sendmail-doc
# package.
```

```
#
```

```
# by default we allow relaying from localhost...
```

```
localhost.localdomain          RELAY
```

```
localhost                      RELAY
```

```
127.0.0.1                     RELAY
```

```
192.168.0                      RELAY <==== เพิ่ม
```

```
coffeeroom.com                 RELAY <==== เพิ่ม
```

- จากนั้นใช้คำสั่ง

```
# makemap hash /etc/mail/access.db < /etc/mail/access
```

- แก้ไขเพิ่ม etc/mail/access โดยใช้คำสั่ง

```
# local-host-names - include all aliases for your machine here.
```

```
coffeeroom.com
```

- Restart IMAP ด้วยคำสั่ง

```
# /etc/init.d/xinetd restart
```

## MySQL

ติดตั้ง MySQL เพื่อใช้งาน Database มีขั้นตอนดังนี้

- ใส่ Linux Redhat 7.3 CD แผ่นที่ 1 แล้วใช้คำสั่ง

```
# mount /mnt/cdrom/
```

```
# cd /mnt/cdrom/RedHat/RPMS/
```

```
# rpm -Uvh perl-5.6.1-34.99.6.i386.rpm
```

- เปลี่ยน CD เป็นแผ่นที่ 3 แล้วใช้คำสั่ง

```
# mount /mnt/cdrom/
```

```
# cd /mnt/cdrom/RedHat/RPMS/
```

```
# rpm -Uvh mysql-3.23.49-3.i386.rpm
```

## Apache Web Server

ติดตั้ง Apache Web Server เพื่อให้บริการ WWW รวมทั้ง WAP มีขั้นตอนดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ใส่ Linux Redhat 7.3 CD แผ่นที่ 1 แล้วใช้คำสั่ง

```
# mount /mnt/cdrom/
# cd /mnt/cdrom/RedHat/RPMS/
# rpm -Uvh expat-1.95.2-2.i386.rpm
```

- เปลี่ยน CD เป็นแผ่นที่ 2 แล้วใช้คำสั่ง

```
# mount /mnt/cdrom/
# cd /mnt/cdrom/RedHat/RPMS/
# rpm -Uvh mm-1.1.3-4.i386.rpm
# rpm -Uvh apache-1.3.23-11.i386.rpm
```

ระบบจะสามารถใช้งาน WWW โดย Directory เริ่มต้นของ Server จะเป็น /var/www/html แต่ในที่นี้เพื่อให้ Server สามารถใช้งาน WAP และต้องการให้การเรียกใช้งาน WAP-Mail ได้ด้วยความสะดวกจะต้องมีการแก้ไขเพิ่มข้อมูลเพื่อให้ Apache รู้จัก Content/Type อื่นที่ใช้กับ WAP เพิ่มเติม และใช้งาน Virtual host เพื่อสะดวกต่อการเรียกใช้ดังนี้

- เพิ่มเติมใน /etc/mime.types

text/vnd.wap.wml	wml
application/vnd.wap.wmlc	wmlc
text/vnd.wap.wmls	wmls
application/vnd.wap.wmlsc	wmlsc
image/vnd.wap.wbmp	wbmp
text/html	xhtml, htm, html
application/xhtml+xml	xhtml, htm, html
application/vnd.wap.xhtml+xml	xhtml, htm, html

- เพิ่มเติมใน /etc/httpd/conf/httpd.conf

```
<VirtualHost wapmail.coffeeroom.com>
ServerAdmin webmaster@wapmail.coffeeroom.com
DocumentRoot /var/www/html/wapmail
ServerName wapmail.coffeeroom.com
</VirtualHost>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## PHP

ติดตั้ง PHP เพื่อใช้งาน PHP Script รวมถึง Module ที่ใช้สำหรับ IMAP และ MySQL

- ใส่ Linux Redhat 7.3 CD แผ่นที่ 1 แล้วใช้คำสั่ง

```
# mount /mnt/cdrom/
# cd /mnt/cdrom/RedHat/RPMS/
# rpm -Uvh freetype-2.0.9-2.i386.rpm
# rpm -Uvh gmp-4.0.1-3.i386.rpm
# rpm -Uvh libjpeg-6b-19.i386.rpm
# rpm -Uvh libtool-libs-1.4.2-7.i386.rpm
# rpm -Uvh libpng-1.0.12-2.i386.rpm
# rpm -Uvh pspell-0.12.2-8.i386.rpm
# rpm -Uvh libxml2-2.4.19-4.i386.rpm
```

- เปลี่ยน CD เป็นแผ่นที่ 2 แล้วใช้คำสั่ง

```
# mount /mnt/cdrom/
# cd /mnt/cdrom/RedHat/RPMS/
# rpm -Uvh curl-7.9.5-2.i386.rpm
# rpm -Uvh gd-1.8.4-4.i386.rpm
# rpm -Uvh mm-1.1.3-4.i386.rpm
# rpm -Uvh php-4.1.2-7.i386.rpm
# rpm -Uvh php-imap-4.1.2-7.i386.rpm
# rpm -Uvh php-mysql-4.1.2-7.i386.rpm
```

หลังจากนั้นใช้คำสั่ง `/etc/init.d/httpd restart` เพื่อ Restart Apache Web Server ให้สามารถ  
ใช้ Configuration ใหม่ รวมทั้งใช้งาน Module ต่างๆ ที่ติดตั้งเพิ่มเข้าไป

## ประวัติผู้เขียน

ชื่อ	นายทศพร เลิศบุร โนนทัย
วันเดือนปีเกิด	27 สิงหาคม 2516
สถานที่เกิด	กรุงเทพมหานคร
การศึกษา	วิทยาศาสตร์บัณฑิต สาขาวิทยาการคอมพิวเตอร์ มหาวิทยาลัยเกษตรศาสตร์ พ.ศ. 2540
อาชีพปัจจุบัน	Senior Analyst บริษัท ไอทีวัน จำกัด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้