

โปรแกรมบันทึกตารางเวลาและเตือนความจำโดยส่งผ่านทาง  
หน้าจอคอมพิวเตอร์,เพจเจอร์,และมือถือ

ORGANIZER & REMINDING TOOL VIA  
PC MONITOR, PAGER & MOBILE PHONE



โดย

นายณพงศ์ ไหมทองคำ

นายพรเทพ ศรีพนมวัลย์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรอุตสาหกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2544

เลขหม.....  
เลขทะเบียน..... 46514  
วัน, เดือน, ปี..... 4 เม.ย. 2546

b.....
i.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อโครงการภาษาไทย

โปรแกรมบันทึกตารางเวลาและเตือนความจำ  
ส่งผ่านทางหน้าจอคอมพิวเตอร์, เพจเจอร์, และมือถือ

TITLE

ORGANIZER & REMINDING TOOL VIA  
PC MONITOR, PAGER & MOBILE PHONE

โดย

นายณพงศ์ ไหมทองคำ

นายพรเทพ ศรีพนมวัลย์

อาจารย์ผู้ควบคุมปริญญาโท

อ. มยุรี เลิศเวชกุล

ภาควิชา

วิศวกรรมสารสนเทศ

ปีการศึกษา

2544

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
อนุมัติให้รับปริญญาโทฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร  
อุตสาหกรรมศาสตรบัณฑิต

  
.....  
( มยุรี เลิศเวชกุล )  
อาจารย์ผู้ควบคุมปริญญาโท

ลิขสิทธิ์ของคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อโครงการภาษาไทย	โปรแกรมบันทึกตารางเวลาและเตือนความจำโดย
นักศึกษา	ส่งผ่านทางหน้าจอคอมฯ, เพจเจอร์, และมือถือ
อาจารย์ผู้ควบคุมปริญญาโท	นายณพงศ์ ไหมทองคำ รหัสประจำตัว43015723
ระดับการศึกษา	นายพรเทพ ศรีพนมวัลย์ รหัสประจำตัว43015735
ภาควิชา	อ. มยุรี เลิศเวชกุล
ปีการศึกษา	ปริญญาอุตสาหกรรมศาสตรบัณฑิต วิศวกรรมสารสนเทศ 2544

### บทคัดย่อ

ปริญญาโทฉบับนี้ได้เสนอ โปรแกรมบันทึกตารางเวลาและเตือนความจำโดยส่งผ่านทางหน้าจอคอมฯ, เพจเจอร์, และมือถือ โปรแกรมนี้สามารถที่จะส่งข้อความสำคัญที่เราต้องการเตือนความจำได้หลายวิธี ซึ่งบนเซิร์ฟเวอร์ผู้ใช้จะต้องบันทึกข้อความที่ใช้เตือนความจำเอาไว้ล่วงหน้าบนเซิร์ฟเวอร์ และ โปรแกรมจะทำการส่งข้อความเหล่านั้นมาเตือนเมื่อถึงเวลาที่กำหนดโดยอัตโนมัติ ระบบเตือนความจำนี้สามารถส่งข้อความเพื่อเตือนความจำได้ 3 ทาง ซึ่งประกอบด้วย การเตือนความจำทางหน้าจอคอมพิวเตอร์, เพจเจอร์ และ โทรศัพท์มือถือ(SMS) การบันทึกข้อความและตารางเวลาสามารถทำได้บันทึกจากเครื่องคอมพิวเตอร์หรือ โดยการใช้เทคโนโลยี WAP บันทึกข้อความผ่านทางมือถือ ได้ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<b>PROJECT TITLE</b>	<b>ORGANIZER &amp; REMINDING TOOL VIR PC MONITOR, PAGER &amp; MOBILE PHONE</b>
<b>STUDENT</b>	Mr. Napong Maitongkum No 43015723 Mr. Porntap Sripanomwan No 43015735
<b>ADVISER</b>	Assoc. Mayuree Lertwatchakul
<b>COURSE</b>	Bachelor of Industrial Technology
<b>DEPARTMENT</b>	Information Engineering
<b>YEAR</b>	2001

### Abstract

This thesis is to design and develop an Organizer and Reminding Tool through PC Monitor, Pager and Mobile Phone. The System can send stored warning messages to many interfaces concurrency. The System checks scheduling table periodically to find which message should be sent at that time. User may input a message and scheduling time via PC interface or mobile phone through WAP technology

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

จากความสำเร็จในการเขียน โปรแกรมบันทึกตารางเวลาและเตือนความจำโดย  
ส่งผ่านทางหน้าจอคอมพิวเตอร์, เพจเจอร์, และมือถือ คณะผู้จัดทำขอขอบพระคุณ ท่านอาจารย์ มยุรี  
เลิศเวชกุล (อาจารย์ที่ปรึกษา) ที่ได้ให้คำชี้แนะ และให้ความสนับสนุนในทุกๆด้าน และ ขอ  
ขอบคุณเพื่อนๆ ที่ให้ความช่วยเหลือในด้านต่างๆ ตลอดมา และขอบพระคุณ คุณพ่อ คุณแม่ เพื่อนๆ  
ที่ให้ความสนับสนุนเป็นอย่างดี จนกระทั่ง โครงการนี้สามารถสำเร็จลุล่วงไปได้ด้วยดี ขอขอบคุณครับ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูปภาพ	VII

## บทที่ 1 บทนำ

1.1 ความเป็นมาของโครงการ	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ขอบเขตของโครงการ	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ	2
1.5 วิธีการดำเนินงาน	2

## บทที่ 2 ทฤษฎีและหลักการ

2.1 ความรู้เบื้องต้นเกี่ยวกับโปรโตคอล	3
2.2 โปรโตคอล TCP/IP	3
2.3 เซิร์ฟเวอร์และไคลเอนต์	5
2.4 โปรโตคอล TCP	8
2.5 ไอพีแอดเดรส (IP Address)	8
2.6 ภาษา Visual Basic 6.0	11
2.7 วินซ็อกคอนโทรล (Winsock Control)	13
2.7.1 โพรซีเจอร์เหตุการณ์ในวินซ็อก	15
2.7.2 คุณสมบัติและวิธีของวินซ็อก	15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

เรื่อง	หน้า
2.8 โครงสร้างโปรโตคอล HTTP	16
2.8.1 ข้อความร้องขอ (request)	18
2.8.2 การร้องขอด้วยเมธอด POST	18
2.8.3 การเข้ารหัสก่อนส่งข้อความ	20
2.9 สร้างเซคเตอร์ให้สามารถติดต่อกับ CGI ได้	20
2.10 โปรแกรม Ethereal	23
2.11 Wireless Application Protocol (WAP)	25
2.11.1 จุดกำเนิด WAP	25
2.11.2 มาตรฐานสำหรับ WAP	26
2.11.3 WAP กับมาตรฐานอินเทอร์เน็ต	27
2.11.4 ศัพท์อื่นๆที่เกี่ยวข้อง	28
2.12 การดู WAP Site	30
2.13 เครื่องมือหลักๆ ที่มีใน Nokia WAP Toolkit	31
<b>บทที่ 3 การออกแบบและการเขียนโปรแกรม</b>	<b>35</b>
3.1 สถาปัตยกรรม	35
3.2 การส่งข้อมูลจากฟอร์ม ไปให้ CGI ในเซิร์ฟเวอร์	38
3.3 หลักการในการสื่อสารกับเว็บเซิร์ฟเวอร์ของเครือข่าย GSM	38
3.4 กระบวนการส่งข้อความเข้าเพจเจอร์ ผ่านทางเว็บไซด์	40
<b>บทที่ 4 การทดลองใช้งานโปรแกรม</b>	<b>41</b>
4.1 หน้าจออินเตอร์เฟซของ PC Monitor	41
4.2 หน้าจอ WAP Server บนเครื่อง PC	42
4.3 หน้าจออินเตอร์เฟซของ Mobile Phone	42

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

เรื่อง	หน้า
บทที่ 5 บทสรุปและวิจารณ์	45
5.1 บทสรุป	45
5.2 ปัญหาในการทำงาน	45
5.3 ข้อเสนอแนะ	45

### เอกสารอ้างอิง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูปภาพ

เรื่อง	หน้า
รูปที่ 2.3.2 ภาพแสดงการรับส่งข้อมูลผ่าน โพรโทคอล TCP/IP	7
รูปที่ 2.7 การเพิ่ม Winsock Control	13
รูปที่ 2.7.1 Control Microsoft Winsock Control 6.0 และ Properties Winsock Dialog	14
รูปที่ 2.8 ลักษณะข้อมูลที่รับส่งระหว่าง ไคลเอ็นต์กับเซิร์ฟเวอร์	17
รูปที่ 2.8.1 โครงสร้างของข้อมูลที่ส่งผ่าน โพรโทคอล HTTP	17
รูปที่ 2.8.2 เมธอด POST ที่เว็บเบราว์เซอร์สร้างขึ้นเพื่อส่งข้อมูลจากฟอร์มในเว็บเพจให้แก่ CGI	19
รูปที่ 2.10 capture –Ethereal	23
รูปที่ 2.12 Nokia WAP Toolkit 3.0	30
รูปที่ 2.13.2 รูปขณะโหลด โปรแกรม Nokia WAP Toolkit 3.0	32
รูปที่ 2.13.3 รูป Toolkit Window ของ Nokia Toolkit	33
รูปที่ 2.13.4 รูป Device Window ของ Nokia Toolkit	34
รูปที่ 3.1 กระบวนการที่เซิร์ฟเวอร์ (PC) ส่งข้อความเรียกเพจเจอร์ผ่านเน็ต	35
รูปที่ 3.1.1 Flowchart แสดงการทำงานของโปรแกรมบน Server PC	36
รูปที่ 3.1.2 Flowchart แสดงการรับค่าจากมือถือทาง WAP เข้าไปที่เซิร์ฟเวอร์ และส่งข้อความเตือนทาง SMS	37
รูปที่ 3.3 เว็บเพจ GSM Web Messaging	38
รูปที่ 3.3.1 เว็บเพจ Easycall Pager	39
รูปที่ 3.4 กระบวนการส่งข้อความเรียกเพจเจอร์ผ่านเว็บ	40
รูปที่ 4.1 หน้าจออินเทอร์เน็ตเฟสของ PC Monitor	41
รูปที่ 4.2 หน้าจอ WAP Sever บนเครื่อง PC	42
รูปที่ 4.3 หน้าจออินเทอร์เน็ตเฟสของ Mobile Phone	42
รูปที่ 4.3.1 แสดงการ Save ค่าข้อมูล ผ่านทางหน้าจออินเทอร์เน็ตเฟสของ Mobile Phone	43
รูปที่ 4.3.2 แสดงการทำงานร่วมกันของ Interface MobilePhone กับหน้าจอของ WAP Serverบน PC	44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1

## บทนำ

### 1.1 ความสำคัญและที่มาของโครงการ

ในปัจจุบันอินเทอร์เน็ตได้เข้ามามีบทบาทอย่างมากในชีวิตประจำวัน ไม่ว่าจะเป็นด้านธุรกิจ หรือด้านการศึกษา เช่น อินเทอร์เน็ตคาเฟ่ อีคอมเมิร์ซ ไอซีทีวี เป็นต้น

ต่อมาเริ่มมีเว็บไซต์ให้บริการส่งข้อความเข้าเครื่องเพจเจอร์และมือถือทางอินเทอร์เน็ตเพิ่มมากขึ้น โดยเราต้องทำการพิมพ์ข้อความที่ต้องการเดือนฝากไว้ที่เว็บไซต์ จึงเกิดแนวคิดที่จะทำการเขียนโปรแกรมบันทึกข้อความและเตือนความจำโดยบันทึกข้อความไว้ล่วงหน้าได้โดยเก็บข้อความนั้นไว้ที่คอมพิวเตอร์ของเรา เพื่อป้องกันการลืกลบข้อความที่เป็นความลับของเราได้

สำหรับโปรแกรมประยุกต์นี้ได้เกิดความคิดมาจากการที่เราสามารถสร้างโปรแกรมบันทึกข้อความและเตือนความจำในสิ่งที่สำคัญที่เรากลัวว่าจะลืมและเตือนเราได้หลายทาง โดยเป็นโปรแกรมที่เป็นส่วนตัวของเราเองที่จะเก็บข้อความที่ต้องการเตือนไว้ในคอมพิวเตอร์ของเราเอง ที่สามารถป้องกันข้อความที่เป็นความลับ โดยที่เราไม่จำเป็นต้องส่งข้อความที่เราต้องการเตือนล่วงหน้าไปที่เว็บไซต์ที่ให้บริการส่งข้อความเข้าเพจเจอร์/มือถือ แต่เราสามารถเก็บเอาไว้ในหน่วยความจำฮาร์ดดิสก์ของเราและเมื่อถึงเวลาที่เรากำหนดส่งข้อความแล้ว ก็จะมีกระบวนการส่งข้อความออกไปเตือนให้เราได้ในทันทีและในการเตือนความจำของโปรแกรมได้3ทางคือ ทางหน้าจอคอมพิวเตอร์ ทางเพจเจอร์ และทางโทรศัพท์มือถือ(SMS) และเราสามารถบันทึกข้อมูลได้อีกทางนอกจากทางการพิมพ์ข้อความทางคีย์บอร์ดก็คือเราสามารถพิมพ์จากทางโทรศัพท์มือถือผ่านทางเทคโนโลยีเว็บ(WAP)

### 1.2 วัตถุประสงค์

- เพื่อศึกษาการเขียนโปรแกรม Visual Basic
- เพื่อศึกษาระบบโปรโตคอลและเน็ตเวิร์ก
- เพื่อศึกษาการเขียนส่งข้อความผ่านทางเว็บไซต์ไปยังเพจเจอร์/มือถือ
- เพื่อศึกษาเทคโนโลยีเว็บ(WAP)
- สามารถนำความรู้ที่ได้ศึกษามา ก่อให้ประโยชน์สูงสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 1.3 ขอบเขตของโครงการงาน

- ใช้ วินซ็อก คอontrol (Winsock Control) ในการส่งข้อความผ่านเว็บ
- ใช้โปรโตคอล TCP/IP ในการส่งข้อความทางเว็บ
- ใช้โปรแกรม Ethereal ดักจับข้อมูลที่ส่ง ไปยังเว็บต่างๆ
- เขียนเว็บไซค์

### 1.4 วิธีการดำเนินงาน

Visual Basic 6.0 เป็นภาษาคอมพิวเตอร์ที่ได้รับความนิยมมาใช้ในการพัฒนาโปรแกรมบนวินโดวส์เนื่องจากง่ายต่อการศึกษา และความสามารถของ Visual Basic ก็ไม่แพ้ภาษาอื่นด้วยเช่นกัน เช่น การเขียนเว็บไซค์ โปรแกรมฐานข้อมูลต่างๆ

ในการเขียนโปรแกรมประยุกต์นี้ได้ใช้ วินซ็อก คอนโทรลเพื่อใช้สำหรับการติดต่อระบบเครือข่าย โดยใช้โปรโตคอล TCP/IP ในการส่งข้อความไปยังเว็บไซค์ที่เราต้องการ ส่วนการจับแฮดเดอร์ของเว็บไซค์ที่เราต้องการติดต่อนั้นใช้โปรแกรม Ethereal ในการ Capture ซึ่งจะกล่าวถึงโดยละเอียดต่อไป

### 1.5 ประโยชน์ที่คาดว่าจะได้รับ

- เพื่อใช้ช่วยในการเตือนความจำในเรื่องงานที่สำคัญได้หลายๆทาง
- ผู้ศึกษาได้รับความรู้จากการทำโครงการงานนี้

## บทที่ 2

### ทฤษฎีและหลักการ

#### 2.1 ความรู้เบื้องต้นเกี่ยวกับโปรโตคอล

การพัฒนาโปรแกรมประยุกต์บนระบบเครือข่ายจำเป็นต้องมีความรู้พื้นฐานเกี่ยวกับโปรโตคอลเครือข่ายคอมพิวเตอร์ มาตรฐานและการทำงานของโปรโตคอล ซึ่งในที่นี้จะกล่าวถึงเฉพาะเนื้อหาที่ควรรู้เท่านั้น

#### 2.2 โปรโตคอล ทีซีพี / ไอพี (TCP/IP)

ในปัจจุบันการสื่อสารผ่านระบบเครือข่าย(Network) รวมทั้งระบบ Internet เป็นที่แพร่หลายมากในแง่ของการใช้งาน, ใช้บริการ, อำนาจ, ความ สะดวก ในชีวิตประจำวันต่างๆเนื่องจากระบบสื่อสารที่รวดเร็วและสามารถเชื่อมโยง เข้ากับหลายๆระบบได้ ทำให้สามารถนำมาประยุกต์ใช้งานผ่านระบบเครือข่ายได้หลากหลายรูปแบบ ซึ่งในบทความนี้จะพยายามเน้นไปประยุกต์ใช้ ในการควบคุมอุปกรณ์หรือ โปรแกรมต่างๆ เรื่องที่ต้องรู้ในการที่จะเขียนโปรแกรมผ่านระบบเครือข่ายมีดังนี้

##### TCP/IP

ระบบเครือข่ายเอาเป็นว่าใช้คำว่า Internet ก็แล้วกันที่ใช้โปรโตคอลมาตรฐานชื่อ TCP/IP ในการสื่อสารผ่านระบบเพื่อติดต่อกับเครื่องคอมพิวเตอร์อื่นๆโปรโตคอล TCP/IP นั้นประกอบด้วยส่วนที่สำคัญ2ส่วนก็คือ

##### TCP(TransmissionControlProtocol)

##### IP(InternetProtocol)

ในการติดต่อสื่อสารกันจริงๆแล้วเราคงจะไม่สามารถเห็นขั้นตอนการทำงานของระบบ ได้เพราะเป็นการทำงานของ Software&Hardware แต่เราจะอธิบายเพื่อความเข้าใจของ โปรโตคอล TCP/IP ให้ดูกันดังมีส่วนประกอบดังนี้

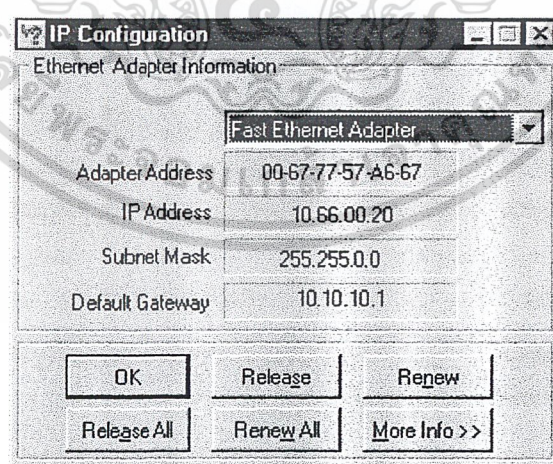
- **IP Address** : สำหรับการรับส่งข้อมูลในระบบ Internet จะถูกกำหนดและอ้างอิงด้วยหมายเลขประจำเครื่องนั้นก็เครื่อง IP Address ซึ่งในระบบ Internet จะมีเครื่องคอมพิวเตอร์เป็นจำนวนมากที่อยู่ในระบบ ในการที่จะใช้ IP Address อาจจะไม่สะดวก จึงได้มีการเปลี่ยนมาใช้เป็น ชื่อ ในความเข้ากันก็คือ Domain name โดยทั้งหมดนี้อยู่ในระบบ Name Services ซึ่งเป็นการอ้างอิงชื่อแทนหมายเลขนั่นเอง

- **Routing Configuration** : ข้อดีของโปรโตคอล TCP/IP ก็คือในการกำหนดเส้นทางสำหรับการรับส่ง ที่สามารถเลือกเส้นทางในการรับส่งข้อมูลได้อย่างอัตโนมัติหากถ้าเกิดเส้นทาง บ้างเส้นทางเสียหาย ระบบกลไกในการกำหนดเส้นทางสำหรับการรับส่งข้อมูลของโปรโตคอล TCP/IP ก็จะเลือกเส้นทางที่เหมาะสมถูกต้องให้สามารถรับส่งข้อมูลได้

- **Protocol, Ports, Sockets** : เป็นช่องทางสำหรับกำหนดทิศทางของการรับส่งข้อมูลนอกเหนือจากที่จะต้องกำหนดหลังจาก IP Address

### การตรวจหาหมายเลข IP Address ของเครื่องคอมพิวเตอร์

อาจมีข้อสงสัยว่าที่กล่าวทั้งนี้ถ้าเราอยากจะดูและรู้ IP Address ได้อย่างไร ไม่ยากครับใน Windows มีโปรแกรมที่ชื่อว่า IP Configuration ไฟล์ EXE จะอยู่ที่ Windows/Winipcfg.exe ครับ โดยมีหน้าต่างดังรูปด้านล่างนี้



รูปที่ 2.2 หน้าตา Windows/Winipcfg.exe

ถ้าต้องการดูรายละเอียดอื่นๆนอกเหนือจากนี้ก็กดปุ่ม More Info

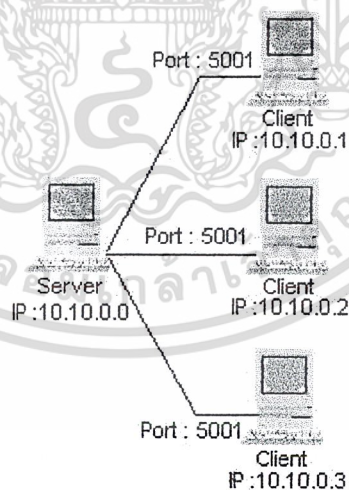
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.3 Server & Client

สิ่งที่เราจะพูดถึงในการเขียนโปรแกรมเพื่อใช้ในระบบเครือข่าย(Network)จุดหลักๆของระบบ จะแบ่งฝ่ายที่ต้องติดต่อรับส่งข้อมูลระหว่างกันออกเป็น 2 ส่วน คือ แม่ข่าย(Server)และลูกข่าย(Client) ซึ่งในการใช้งานจริงอาจมีส่วนประกอบอื่นๆอีก แต่เราจะขอไม่กล่าวถึงเนื่องจากต้องการให้เห็นภาพและเข้าใจง่ายขึ้นจึงยก แค่ 2 ส่วนนี้มากล่าว

Server จะเป็นส่วนทำหน้าที่เสมือนกองอำนาจการ,ประชาสัมพันธ์,เมสเสจเจอร์ รวมถึงผู้จัด ให้กับระบบ,ลูกข่าย ที่จะเป็นส่วนร้องขอข้อมูลจาก Server โดย Server ในที่นี้จะเป็นส่วนที่เก็บข้อมูล,จัดการ,บริหารข้อมูลหรือทรัพยากรระบบ เพื่อให้ฝ่ายลูกข่ายสามารถใช้บริการ

ServerและClient ต่างก็จะต้องมีตำแหน่งที่อยู่(IP Address),ช่องทางการติดต่อ(Port) โดยทั้งสองฝ่ายจะสามารถติดต่อถึงกันได้จะต้องอยู่ในช่องทางเดียวกัน ซึ่งเราสามารถกำหนดหมายเลขของ Port ได้ ทั้งนี้โปรแกรมที่ติดต่อนั้นจะต้องอ้างอิงหมายเลขของ Port ทุกครั้งเนื่องจากในระบบมีโปรแกรมมากมายที่กำลังติดต่อกันอยู่



รูปที่ 2.3 ตำแหน่งที่อยู่(IP Address),ช่องทางการติดต่อ(Port)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรโตคอล TCP/IP มีการจัดกลไกการทำงานเป็นชั้นหรือเลเยอร์เรียงต่อกันไป เหมือนกับ OSI - Reference Model โดยในแต่ละเลเยอร์นั้นจะมีการทำงานเทียบได้กับ OSI - Reference Model แต่บางเลเยอร์ของโปรโตคอล TCP/IP จะทำงานเทียบกับ OSI-Reference Model หลายเลเยอร์ปนกัน ซึ่งในแต่ละเลเยอร์ของ โปรโตคอล TCP/IP จะประกอบไปด้วย

- โพรเซส เลเยอร์ (Process Layer) ได้แก่ Telnet, FTP, DNS
- โฮสต์-ทู-โฮสต์ (Host-To-Host Layer) ได้แก่ ทีซีพี, ยูซีพี
- อินเทอร์เน็ตเลเยอร์ (Internetwork Layer) ได้แก่ ไอพีแอดเดรส, ไอซีเอ็มพี, เออาร์พี, อาร์เออาร์พี
- เน็ตเวิร์ค อินเตอร์เฟซ เลเยอร์ (Network Interface Layer) ได้แก่ อีเทอร์เน็ต, โทเค็นริง

โดยเมื่อได้เทียบลำดับชั้นเลเยอร์กับมาตรฐานของ OSI - Reference Model แล้ว จะเป็นดังรูปที่ 2.1 ซึ่งเราจะเห็นว่า บางเลเยอร์ของ TCP/IP นั้นจะเทียบได้กับ มาตรฐาน ISO Model ได้ 2 ชั้น อย่างเช่นเลเยอร์ของ โพรเซส เลเยอร์ของโปรโตคอล TCP/IP จะเทียบได้กับ 2 เลเยอร์ คือ แอปพลิเคชันเลเยอร์ กับชั้นฟิสิคัลเลเยอร์ของ OSI - Reference Model รวมกัน เป็นต้น

OSI - Reference Model	โปรโตคอล ทีซีพี/ไอพี
Application Layer	Process Layer
Preentation Layer	
Session Layer	
Transport Layer	Host- to-Host Layer
Network Layer	Internetwork Layer
Data Link Layer	Network Interface Layer
Physical Layer	

รูปที่ 2.3.1 TCP/IP เปรียบเทียบกับ OSI Model

ซึ่งในแต่ละเลเยอร์ได้มีการกำหนดหน้าที่การทำงานไว้ดังต่อไปนี้

1. เลเยอร์ชั้นโพรเซสเลเยอร์ จะเป็นแอปพลิเคชันโปรโตคอลเชื่อมต่อกับผู้ใช้และให้บริการต่าง ๆ โปรโตคอลหลัก ๆ ที่ทำงานและให้บริการในชั้นโพรเซสเลเยอร์ นี้ก็ตัวอย่างเช่น เอฟทีพี(FTP), เทลเน็ต(Telnet), เอชทีทีพี(HTTP),เอสเอ็มทีพี( SMTP) เป็นต้น จากรูปที่ 2.3 แสดงลำดับชั้นการทำงานของโปรโตคอล TCP/IP เทียบกับมาตรฐานของ OSI - Reference Model นั้น ในชั้นบนสุดที่เรียกว่าโพรเซสเลเยอร์ทำงาน 2 หน้าที่เทียบได้กับ แอปพลิเคชันเลเยอร์ และชั้นฟิสิคัลเลเยอร์

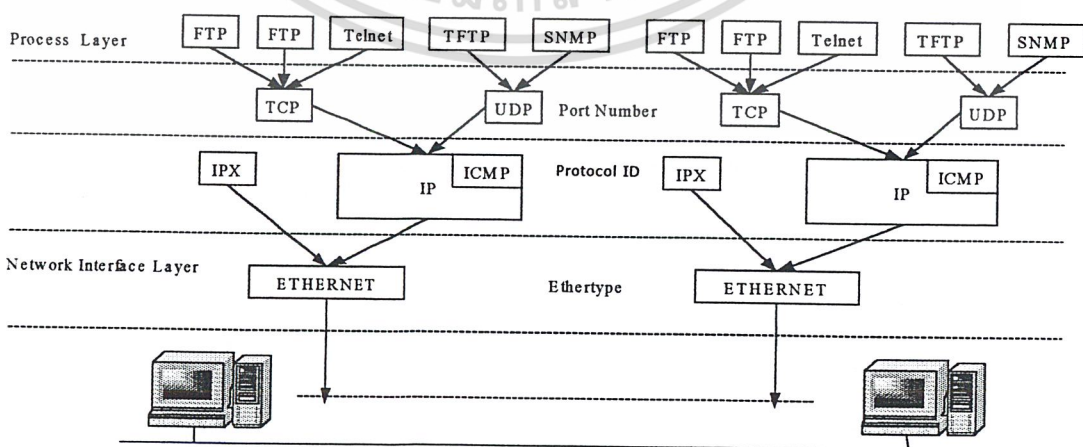
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. เลเยอร์ชั้นโฮสต์-ทูล-โฮสต์ จะเป็นทีซีพีหรือยูดีพีที่ทำหน้าที่คล้ายกับชั้นของเซสชันเลเยอร์และชั้น ทรานสปอร์ตเลเยอร์ของ OSI - Model คือควบคุมการรับส่งข้อมูล จากปลายด้านส่งถึงปลายด้านรับข้อมูล และตัดข้อมูลออกเป็นส่วนย่อยให้เหมาะสม กับเครือข่ายที่ใช้รับส่งข้อมูล รวมทั้งประกอบข้อมูลส่วนย่อยๆ นี้เข้าด้วยกันเมื่อถึงปลายทาง

3. เลเยอร์ชั้นอินเทอร์เน็ตเวิร์คเลเยอร์ ได้แก่ส่วนของโปรโตคอลไอพีซึ่งทำหน้าที่คล้ายกับชั้นของเน็ตเวิร์ค เลเยอร์ของ OSI - Model คือเชื่อมต่อคอมพิวเตอร์เข้ากับระบบเครือข่ายที่อยู่ชั้นล่างลงไป และทำหน้าที่เลือกเส้นทางการรับส่งข้อมูล ผ่านอุปกรณ์เครือข่ายต่าง ๆ จนไปถึงผู้รับข้อมูล ในขั้นนี้จะจัดการกับกลุ่มข้อมูลในลักษณะที่เรียกว่าเฟรม ในรูปแบบของ TCP/IP ที่เรารู้จักกันนั่นเอง

4. เลเยอร์ชั้นเน็ตเวิร์ค อินเทอร์เน็ต เลเยอร์ เป็นชั้นที่ควบคุมอาร์คแวร์ การรับส่งข้อมูลผ่านระบบเครือข่าย ซึ่งเทียบได้กับชั้นดาต้าลิงก์เลเยอร์ กับฟิสิคัลเลเยอร์ ของ โอเอสไอ-โมเดล ในขั้นนี้จะทำหน้าที่เชื่อมต่อกับฮาร์ดแวร์และควบคุมการรับส่งข้อมูลในระบบฮาร์ดแวร์ของเครือข่าย ซึ่งที่ใช้กันอยู่จะเป็นตามมาตรฐานของไออีอีไอเช่น ไออีอีไอ 802.3 จะเป็นการเชื่อมต่อผ่านแลน แบบอีเทอร์เน็ต แลนหรือ ไออีอีไอ 802.5 จะเป็นการเชื่อมต่อผ่านแลน แบบโทเค็นริง เป็นต้น

เราจะเห็นได้ว่าที่จริงแล้วโปรโตคอล TCP/IP นั้น แบ่งออกเป็น 2 โปรโตคอลซ้อนกันอยู่คือ ทีซีพีจะอยู่ในชั้นบน และไอพี จะอยู่ในชั้นถัดลงมา นั่นคือ TCP/IP ไม่ได้เป็นโปรโตคอลชนิดเดียวกันทั้งหมด และไม่ได้เชื่อมติดเป็นชั้นเดียวกันทั้งหมด ทีซีพี ก็มีมาตรฐานของเฟรมที่รับส่งข้อมูลของมันเอง และมีหน้าที่ในการรับส่งข้อมูลแตกต่างไปจากไอพี ซึ่งในการรับส่งข้อมูลนั้น เฟรมของทีซีพีที่อยู่ในชั้นบนทั้งหมดจะถูกผนึกอยู่ในส่วนที่เป็นข้อมูลของไอพี เหมือนกับที่แต่ละชั้นของ OSI - Reference Model ผนึกข้อมูลในชั้นถัดไปนั่นเอง



รูปที่ 2.3.2 ภาพแสดงการรับส่งข้อมูลผ่านโปรโตคอล TCP/IP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.4 โพรโทคอล TCP

เป็นโพรโทคอลที่มีการรับส่งข้อมูลแบบสตรีม (Stream Oriented Protocol) หมายความว่า การรับส่งข้อมูลจะไม่คำนึงถึงปริมาณข้อมูลที่จะส่งไป แต่จะแบ่งข้อมูลเป็นส่วนย่อยๆ ก่อน แล้วจึงจะส่งไปยังปลายทางอย่างต่อเนื่องเป็นลำดับข้อมูล ในกรณีที่ข้อมูลส่วนใดส่วนหนึ่งสูญหายไป ก็ จะส่งข้อมูลส่วนนั้นใหม่อีกครั้ง สำหรับปลายทางก็จะทำหน้าที่จัดเรียงส่วนของข้อมูลค้ำค่าแกรม (Datagram) ใหม่ให้ต่อเนื่องกันและประกอบกลับเป็นข้อมูลทั้งหมดได้ ซึ่งจะแยกข้อมูลที่ไม่ถูก ต้อง ออก ดังนั้นแอปพลิเคชันหรือโปรเซสใดที่อาศัยการส่งผ่านข้อมูลด้วยโพรโทคอลTCP จะต้อง ใช้หน่วยความจำและขนาดของช่องสัญญาณ (bandwidth) มากกว่ายูดีพี

การติดต่อระหว่างกันจะต้องเป็นแบบ connection-oriented คือต้องมีการสร้างการติดต่อกัน เป็น session ทั้ง 2 ด้านเสียก่อน แล้วจึงจะรับส่งข้อมูล ไปได้พร้อมกัน (full duplex) เหมือนกับการ ใช้โทรศัพท์ติดต่อกัน เมื่อผู้ติดต่อต้นทางเรียกให้ฝ่ายตรงข้ามรับสายแล้ว จึงเริ่มการสนทนา เช่น พูด คำว่า "สวัสดี" หรือ "ฮัลโหล" กันก่อนเพื่อให้แน่ใจว่าฝ่ายตรงข้ามพร้อมจะติดต่อด้วย จากนั้นจึงเริ่ม ต้นติดต่อกัน และเมื่อต้องการจะเลิกการติดต่อก็จะมีการพูดคำว่า "สวัสดี" ให้ฝ่ายตรงข้ามทราบว่า จะเลิกการติดต่อและวางสายไป ซึ่งในระหว่างการติดต่อกันนั้น แม้ว่าฝ่ายใดฝ่ายหนึ่งหรือทั้งสอง ฝ่ายจะเงียบ ไป คือไม่พูดอะไรเป็นเวลานาน ๆ แต่การเชื่อมโยงระหว่างทั้งสองด้านยังคงมีอยู่ไม่ขาด ไปจนกว่าฝ่ายใดฝ่ายหนึ่งจะวางสาย เช่นเดียวกับการติดต่อกันด้วยกลไกโพรโทคอลทีซีพี เมื่อแอป พลิเคชันต้องการส่งผ่านข้อมูลจะใช้โพรโทคอลที่เหมาะสมในชั้น โปรเซสเลเยอร์ ติดต่อกันและมี การสร้างช่องส่งข้อมูลผ่านพอร์ตที่กำหนดเพื่อส่งผ่านข้อมูลไปยังโพรโทคอลทีซีพี

ในระหว่างการรับส่งข้อมูลนี้ โพรโทคอลทีซีพีจะเพิ่มขบวนการสอบทานข้อมูลเพื่อให้ข้อมูลมีความถูกต้องไม่ผิดพลาดไปจากเดิม โดยการส่งสัญญาณสอบทานข้อมูล (Acknowledgement) และส่งข้อมูลให้ใหม่อีกครั้ง ถ้าปลายทางไม่ได้รับหรือเกิดความผิดพลาดขึ้น

ความน่าเชื่อถือของการส่งผ่านข้อมูลโดยโพรโทคอลทีซีพีจะมีมากกว่า แต่ก็ต้องอาศัย ทรัพยากรของระบบมากกว่าในการทำงานเช่นกัน

## 2.5 ไอพี แอดเดรส (IP Address)

หมายเลขไอพีแอดเดรส นั้นจะประกอบไปด้วยตัวเลขฐาน 2 จำนวน 4 ไบต์หรือ 32 บิต โดยจะแยกออกเป็น 2 ส่วน คือ หมายเลขเน็ตเวิร์คและ หมายเลขเครื่อง แต่เนื่องจากว่าตัวเลขไบนารี 32 หลัก เป็นตัวเลขที่จดจำได้ยากแต่เครื่องคอมพิวเตอร์ก็ใช้เลขเหล่านี้ได้อย่างถูกต้อง แต่เมื่อ กำหนดเลข 4 ไบต์ แต่ละไบต์ มีขนาด 0-255 เมื่อดูแล้วจะทำให้จำได้ง่ายขึ้น เนื่องจากขนาดของ

เครือข่ายมีขนาดแตกต่างกันมากดังนั้นจึงมีการกำหนดการแบ่งคลาสของเครือข่ายออกเป็น 3 คลาส คือ คลาส เอ คลาส บี คลาส ซี

**- คลาส A**

กำหนดตัวเลขไอพีเน็ตเวิร์คเพียงไบต์แรกไบต์เดียว ที่เหลืออีกสามไบต์จึงเป็นรหัสประจำเครื่องที่อยู่ในเครือข่าย หมายเลข ไอพีแอดเดรส ตำแหน่งบิตแรก ของไอพีเน็ตเวิร์คจะต้องเป็น 0 บิตที่เหลืออีก 7 บิตจึงเป็นหมายเลขไอพีเน็ตเวิร์คทำให้คลาส A มีจำนวนเน็ตเวิร์คได้ทั้งหมด 127 เน็ตเวิร์ค ส่วน 3 ไบต์ที่เหลือ มีจำนวน 24บิต ทำให้มี โหนด(Node) ได้ 16,777,216 โหนด โดยมีช่วงไอพีแอดเดรส ที่ใช้ได้อยู่ที่ 1.0.0.0 - 127.255.255.255

**- คลาส B**

กำหนดตัวเลขไอพีเน็ตเวิร์คที่ 2 ไบต์ จึงเป็นรหัสประจำเครื่องที่อยู่ในเครือข่ายหมายเลขไอพีแอดเดรสตำแหน่ง 2 บิต แรกของไอพีเน็ตเวิร์คจะต้องเป็น 10 บิต ที่เหลืออีก 6 บิต จึงเป็นหมายเลข ไอพีเน็ตเวิร์ค ทำให้ คลาส B มีจำนวนเน็ตเวิร์คได้ทั้งหมด 16,384 เน็ตเวิร์ค ส่วน 2 ไบต์ ที่เหลือ มีจำนวน 16 บิต ทำให้มี โหนด ได้ 65,534 ได้ 65,534 Node โดยมีช่วงไอพีแอดเดรสที่ใช้ได้อยู่ที่ 128.0.0.0 - 191.255.255.255

**- คลาส C**

กำหนดตัวเลขไอพีเน็ตเวิร์คที่ 3 ไบต์แรก ที่เหลืออีก 1 ไบต์ จึงเป็นรหัสประจำเครื่องที่อยู่ในเครือข่าย หมายเลขไอพีแอดเดรสตำแหน่ง 3 บิตแรก ของไอพีเน็ตเวิร์คจะต้องเป็น 110 บิตที่เหลืออีก 5 บิต จึงเป็นหมายเลขไอพีเน็ตเวิร์คทำให้คลาส C มีจำนวนเน็ตเวิร์ค ได้ทั้งหมด 2,097,152 เน็ตเวิร์ค ส่วน 1 ไบต์ ที่เหลือมีจำนวน 8 บิต ทำให้มี Node ได้ 254 Node โดยมีช่วงไอพีแอดเดรสที่ใช้ได้อยู่ที่ 192.0.0.0 - 223.255.255.255

**- คลาส D**

ใช้สำหรับกระจายข้อมูลข่าวสารแบบหลายจุด ตำแหน่ง 4 บิตแรก ของไอพีเน็ตเวิร์คจะต้องเป็น 1110 มีช่วงไอพีแอดเดรสที่ใช้ได้อยู่ที่ 224.0.0.0 ถึง 239.255.255.255

**- คลาส E**

มีไว้ใช้สำรองในอนาคต ตำแหน่ง 5 บิตแรก ของไอพีเน็ตเวิร์คจะต้องเป็น 11110 มีช่วงไอพีแอดเดรสที่ใช้ได้อยู่ที่ 240.0.0.0 ถึง 247.255.255.255

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## พอร์ต (Port)

เนื่องจากในเวลาใดๆ สามารถมีโปรเซสของผู้ใช้สามารถใช้ยูติลิตี้หรือที่ซีพีได้พร้อมกัน ดังนั้นจึงต้องมีวิธีแยกแยะว่าข้อมูลเป็นของผู้ใช้คนใด ซึ่งวิธีที่ที่ซีพีและยูติลิตี้ใช้คือการใช้หมายเลขพอร์ต (Port Number) เมื่อโปรเซสไคลเอ็นต์ (Client Process) ต้องการที่จะติดต่อกับเซิร์ฟเวอร์ไคลเอ็นต์จะต้องเจาะจง เซิร์ฟเวอร์ที่ต้องการติดต่อ แต่ถ้าฝั่งผู้แอดเดรสอินเตอร์เน็ต 32 บิต เพียงอย่างเดียวนั้นไม่พอ เพราะว่าสามารถติดต่อกับโฮสต์ได้เพียงอย่างเดียวแต่ไม่สามารถเจาะจงโปรเซสที่จะทำการติดต่อได้ ดังนั้นเพื่อแก้ปัญหานี้ทั้งที่ซีพีและยูติลิตี้ได้มีการกำหนด หมายเลขพอร์ตมาตรฐาน (Well-know Port) ซึ่งเป็นที่รู้จักกัน เช่น ทุกๆระบบ TCP/IP ที่มีเซิร์ฟเวอร์ FTP (File Transfer Protocol) จะมีหมายเลขพอร์ตเป็น 21 เป็นต้น เมื่อที่ซีพีหรือยูติลิตี้ กำหนดหมายเลขพอร์ตที่ไม่ซ้ำกันให้โปรเซสของผู้ใช้ เราเรียกหมายเลขพอร์ตนี้ว่าหมายเลขพอร์ตชั่วคราว (Ephemeral Port Numbers) เมื่อไคลเอ็นต์เล็กใช้หมายเลขพอร์ตนี้แล้ว สามารถกำหนดหมายเลขพอร์ตนี้ให้ไคลเอ็นต์อื่นได้ โปรเซสที่ได้รับหมายเลขพอร์ตชั่วคราวนี้จะไม่สนใจว่ามีค่าเท่าใดแต่เป็นหน้าที่ของอีกโปรเซสหนึ่งที่ต่อกันที่ต้องสนใจ เพราะต้องส่งข้อมูลกลับมาที่พอร์ตนี้ในที่ซีพีและยูติลิตี้ นั้นหมายเลขพอร์ตตั้งแต่ 1-1023 เป็นพอร์ตที่สงวนไว้สำหรับหมายเลขพอร์ตมาตรฐาน

## Winsock ( Window Socket Application Programming Interface )

Winsock เป็นอินเทอร์เฟซของซ็อกเก็ต (Socket) แบบมาตรฐานที่ใช้สำหรับโปรแกรมประยุกต์ของเน็ตเวิร์คที่มีการใช้โปรโตคอล TCP/IP ซึ่งมีรากฐานเดียวกันกับซ็อกเก็ตของยูนิกซ์ตระกูล BSD (Berkeley Software Distribution) จากมหาวิทยาลัยแคลิฟอร์เนีย วิทยาเขต Berkeley สำหรับโปรโตคอลที่ใช้ใน Winsock จะมี 2 โปรโตคอลด้วยกันคือ Winsock TCP (Transmission Control Protocol) และ Winsock UDP (User Datagram Protocol) โดยการทำงานแบบ Winsock TCP จะคล้ายกับการติดต่อทางโทรศัพท์คือจะต้องมีการเชื่อมต่อกันก่อนจึงจะติดต่อกันได้ และการส่งข้อมูลจะมีการตรวจสอบด้วยว่าผู้รับได้รับข้อมูลครบถูกต้องหรือไม่ ถ้าไม่ถูกต้องหรือข้อมูลที่ส่งไปขาดหาย ผู้ส่งก็จะทำการส่งข้อมูลนั้นไปใหม่จนกว่าผู้รับจะได้รับข้อมูลที่ถูกต้อง ส่วนการทำงานแบบ Winsock UDP จะเป็นแบบ Connectionless คือจะไม่มีการเชื่อมต่อกันและไม่มีการตรวจสอบข้อมูลด้วย ว่าผู้รับได้รับข้อมูลที่ส่งไปไหม ซึ่งก็เหมือนกับสถานีส่งคลื่นวิทยุหรือการส่งจดหมายธรรมดา นั่นเอง สำหรับรายละเอียดการใช้งาน Winsock จะได้อีกกล่าวถึงต่อไป

## 2.6 ภาษา Visual Basic 6.0

เป็นภาษาหนึ่งที่มีความใกล้ชิดกับระบบปฏิบัติการวินโดวส์มากที่สุด เพราะมีความสามารถต่างๆ และคำสั่งต่างๆ ส่วนใหญ่จะใช้ร่วมกับความสามารถของวินโดวส์ได้โดยตรง ซึ่งในทางปฏิบัติก็หมายความว่า Visual Basic 6.0 ได้ดึงเอาความสามารถของวินโดวส์ มาเป็นส่วนหนึ่งของความสามารถของตัวแปลภาษา เพื่อช่วยให้การสร้างแอปพลิเคชันที่สามารถใช้ประโยชน์จากความสามารถของวินโดวส์ได้อย่างเต็มที่

วินโดวส์ 32 บิต เป็นระบบปฏิบัติการที่มีการนำเสนอความสามารถทั้งด้านมัลติทาสกิง (multitasking) และมัลติเธรดดิ้ง (multithreading) ให้กับวงคอมพิวเตอร์ส่วนบุคคล ซึ่งแอปพลิเคชันที่สร้างด้วย Visual Basic 6.0 ก็สามารถใช้ความสามารถนี้ได้เช่นกัน โดยวินโดวส์จะทำหน้าที่สลับการควบคุมการทำงานระหว่างแอปพลิเคชันให้โดยอัตโนมัติ และนอกจากนี้แอปพลิเคชันตัวเดียวกันยังสามารถทำงานในช่วงเวลาเดียวกันได้หลายๆ ก๊อปปี้ ซึ่งเราเรียกก๊อปปี้ของแอปพลิเคชันตัวหนึ่งๆ ว่า instance ซึ่งแตกต่างจากความสามารถด้านมัลติทาสกิงของวินโดวส์ 16 บิต (Win 3.x) ที่เราเรียกอย่างหนึ่งว่า nonpreemptive multitasking หรือ cooperative multitasking ทั้งนี้เนื่องจากการที่ระบบปฏิบัติการจะกำหนด CPU ให้กับแอปพลิเคชันหนึ่งๆ ในช่วงเวลาหนึ่งๆ นั้น แอปพลิเคชันนั้นจะสามารถควบคุมการใช้งาน CPU ได้อย่างสมบูรณ์ จนกว่าแอปพลิเคชันนั้นจะเสร็จสิ้นการทำงานในช่วงที่กำหนด และพร้อมที่จะโอนการควบคุมต่อให้กับแอปพลิเคชันอื่นต่อไป แอปพลิเคชันอื่นๆ จึงจะสามารถใช้ CPU ได้ ดังนั้นถ้าหากในระหว่างที่แอปพลิเคชันหนึ่งๆ ควบคุมการใช้งาน CPU เอาไว้และเกิดข้อผิดพลาดในแอปพลิเคชันนั้นๆ ก็จะทำให้ CPU ถูกล็อกโดยวินโดวส์ และแอปพลิเคชันอื่นก็จะไม่สามารถใช้ CPU ได้อีกต่อไปซึ่งในกรณีที่เลวร้ายที่สุดก็จะทำให้เครื่องหยุดการทำงานได้ทันที

เมื่อไมโครซอฟท์ได้พัฒนาวินโดวส์ 32 บิต ก็ได้มีการเพิ่มความสามารถด้านมัลติทาสกิง โดยการแบ่งแอปพลิเคชันที่กำลังทำงานออกเป็นขบวนการ (Process) หรืองาน (task) ซึ่งเราเรียกความสามารถนี้ว่ามัลติเธรดดิ้ง (multithreading) ซึ่งในการทำงาน วินโดวส์จะกระจายเวลาสำหรับการควบคุมให้กับแต่ละงานของแอปพลิเคชันต่างๆ ตลอดเวลา ดังนั้นในช่วงเวลาหนึ่งจะไม่มีแอปพลิเคชันใดเลยที่จะครอบครอง CPU เอาไว้ได้โดยลำพัง และเราเรียกขบวนการนี้ว่า preemptive multitasking สำหรับแอปพลิเคชันที่สร้างด้วย Visual Basic 6.0 นั้นจะสามารถใช้ความสามารถในด้านการมัลติเธรดดิ้งของวินโดวส์ 32 บิต

แอปพลิเคชันที่สร้างด้วย Visual Basic 6.0 ส่วนใหญ่จะประกอบด้วยองค์ประกอบหลัก 3 ส่วนด้วยกัน ดังนี้ ฟอรัม (Form), คอนโทรล (ActiveX Custom Control) และ ออบเจกต์ระบบ (System Object)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ฟอร์มและคอนโทรล

จะเป็นออบเจกต์ที่ถูกออกแบบเฉพาะ และถูกจัดการโดยตรงจาก Visual Basic 6.0 ซึ่งออบเจกต์จะมีความสามารถด้านการปรับแต่งและตอบสนองต่อเหตุการณ์ต่างๆ ได้อย่างมากมาย ซึ่งเราสามารถแบ่งความสามารถออกได้เป็น 3 ประเภท ดังนี้

<b>คุณสมบัติ(Property)</b>	หมายถึง แอตทริบิวต์ของออบเจกต์ ที่อนุญาตให้ผู้อ่านแก้ไขหรือปรับแต่งได้ตามที่ต้องการ เช่น คุณสมบัติ Visible หรือ Enabled ของฟอร์มเป็นต้น ซึ่งคุณสมบัติบางรายการจะส่งผลกระทบต่ออ้างถึงออบเจกต์นั้นๆ เป็นการเฉพาะก็มี เช่น Name เป็นต้น ซึ่งในกรณีหลังจะไม่มีผลต่อการแสดงผลใดๆ ทั้งสิ้น และบางคุณสมบัติจะไม่สามารถแก้ไขได้ เพราะถูกกำหนดให้เป็นคุณสมบัติที่สามารถอ่านค่าได้อย่างเดียว(read only) เท่านั้น
<b>โพรซีเยอร์เหตุการณ์(Event)</b>	หมายถึง เหตุการณ์ที่ถูกสร้างขึ้นตามความสามารถของวินโดวส์ เพื่อตอบสนองต่อการกระทำหรือการเปลี่ยนแปลงคุณสมบัติของออบเจกต์ โดยปกติผู้อ่านจะทำการเขียนโค้ดลงใน โพรซีเยอร์เหตุการณ์เพื่อตอบสนองต่อเหตุการณ์ที่เกิดขึ้น
<b>วิธี(Method)</b>	หมายถึง การกระทำบนออบเจกต์ เช่น วิธี Clear ของคอนโทรล ListBox เป็นต้นซึ่งในความเป็นจริงวิธีก็เป็นเพียงฟังก์ชันหนึ่งๆ ของออบเจกต์ ที่มีไว้เพื่อสนับสนุนการจัดการกับออบเจกต์เท่านั้น

## ออบเจกต์ระบบ

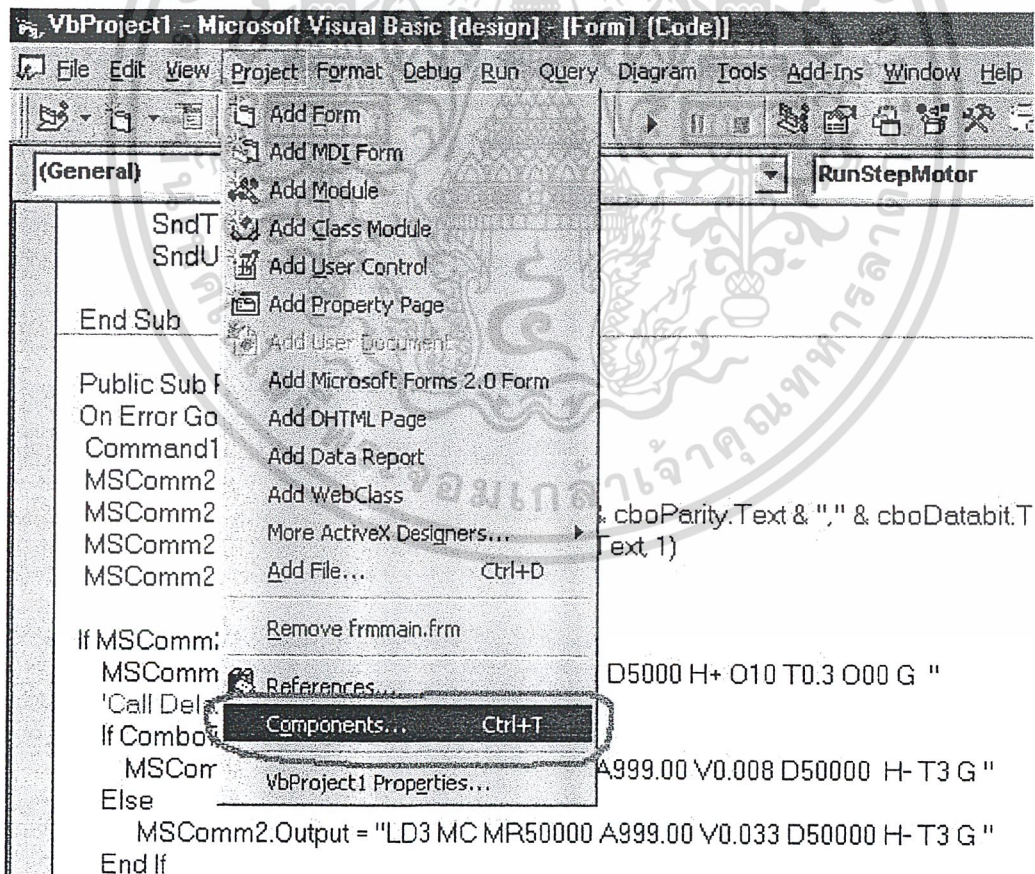
ออบเจกต์ระบบ หมายถึง ออบเจกต์ที่สนับสนุนโดยระบบปฏิบัติการวินโดวส์ ซึ่ง Visual Basic 6.0 ได้สนับสนุนการเรียกใช้ออบเจกต์ระบบได้โดยตรงอยู่แล้ว เช่น Printer Screen Menu หรือ Clipboard เป็นต้น โดยที่ผู้อ่านสามารถที่จะแก้ไขคุณสมบัติของออบเจกต์ระบบเหล่านั้นได้โดยตรงจาก Visual Basic 6.0

## 2.7 Winsock control

ในการใช้งาน Winsock Control ในที่นี้จะกล่าวถึงเฉพาะในส่วนที่ใช้ในการเขียนโปรแกรมเพียงเล็กน้อยเท่านั้น สำหรับผู้สนใจสามารถศึกษาได้จากหนังสืออ้างอิงท้ายเล่มได้

จากตอนที่แรกได้กล่าวเนื้อหาเบื้องต้นและโดยรวมเกี่ยวกับการอินเทอร์เน็ต Hardware ผ่านระบบเครือข่ายโดยมีโปรโตคอลTCP/IP เป็นตัวช่วยในการสื่อสารระหว่างคอมพิวเตอร์ด้วยกันได้ ทั้งนี้ในทางการเขียนโปรแกรมเพื่อติดต่อบนระบบ Windows นั้นก็ได้อำนวยความสะดวกให้เราสามารถที่จะเขียนโปรแกรมติดต่อกับระบบเครือข่ายได้ก็คือ MS Winsock Control ที่มีอยู่ในชุด Visual Studio ในเนื้อหานี้เราจะใช้เขียนโปรแกรมด้วยVisual Basicเพราะจะเข้าใจง่าย,สะดวก,และรวดเร็วต่อการทำความเข้าใจ

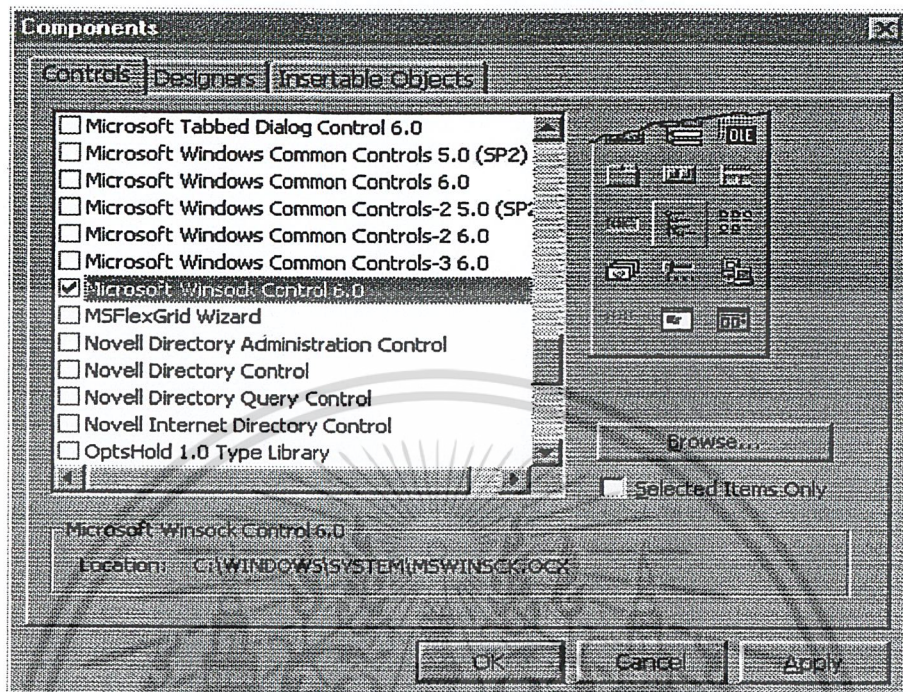
ก่อนอื่นก็เปิดโปรแกรม VB ขึ้นมาแล้วสร้าง Project Standard EXE ขึ้นใหม่ จากนั้นเราจะต้องเพิ่ม Winsock Control ก่อน โดยคลิกที่เมนู



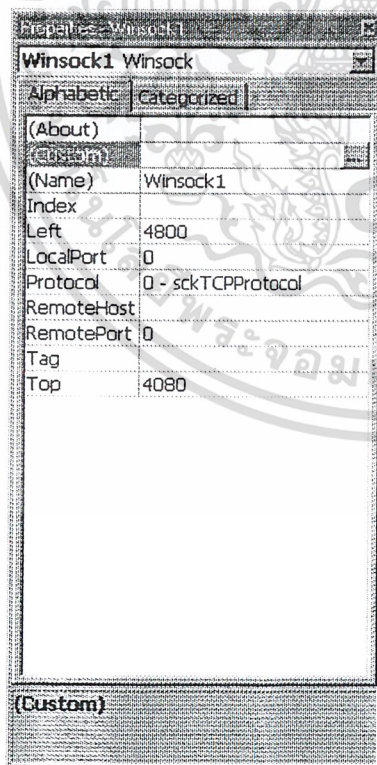
รูปที่ 2.7 การเพิ่ม Winsock Control

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้เลือก Control ชื่อ Microsoft Winsock Control 6.0 (สำหรับ VB6)



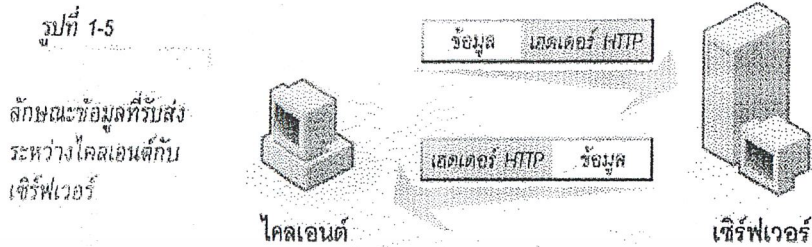
Properties Winsock Dialog



Properties	Detail
Name	ชื่อของคอนโทรลที่เราสามารถตั้งได้
LocalPort	Portsที่เราจะกำหนดในการติดต่อ
Protocol	เลือกชนิดของโปรโตคอล ประเภทที่เราใช้ 0- TCPProtocol
RemoteHost	IP Address ของเครื่องที่ใช้ ติดต่อ
RemotePort	Port Address ของเครื่องที่ ใช้ติดต่อ

รูปที่ 2.7.1 Control Microsoft Winsock Control 6.0 และ Properties Winsock Dialog

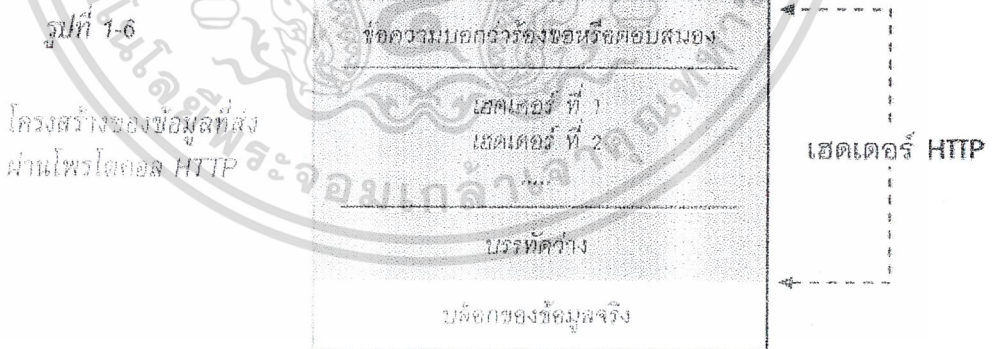
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 ลักษณะข้อมูลที่รับส่งระหว่างไคลเอนต์กับเซิร์ฟเวอร์

เนื่องจากข้อมูลในเซตเคอร์ HTTP เป็นตัวควบคุมหรือบอกให้ฝ่ายรับควรทำอะไรกับข้อมูลที่ส่งมาให้ ในบางครั้งจึงมีคนเรียกข้อมูลส่วนนี้ว่า Meta Informatio

ถึงแม้ว่าข้อมูลในเซตเคอร์ HTTP จะสำคัญต่อการนำไปตีความหมายของข้อมูลก็ตาม แต่ข้อมูลในเซตเคอร์ HTTP จะเป็นเพียงข้อความธรรมดา มิได้มีการเข้ารหัสหรือมีลักษณะพิเศษแตกต่างจากข้อมูลปกติแต่อย่างใด โดยโครงสร้างจะเป็นดังรูป



รูปที่ 2.8.1 โครงสร้างของข้อมูลที่ส่งผ่านโปรโตคอล HTTP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.8.1 ข้อความร้องขอ (request)

เมื่อไคลเอนต์เชื่อมต่อกับเซิร์ฟเวอร์สำเร็จแล้ว ไคลเอนต์ต้องเป็นฝ่ายเริ่มเปิดการพูดคุยด้วยการส่งข้อมูลไปยังเซิร์ฟเวอร์เพื่อบอกการร้องขอข้อมูล การร้องขอไปยังเซิร์ฟเวอร์สามารถทำได้หลายแบบ เช่น ร้องขอให้เซิร์ฟเวอร์ส่งไฟล์มาให้แบบนี้จะเรียกว่าการร้องขอแบบ GET หรือร้องขอเพื่อถามเซิร์ฟเวอร์ว่ามีไฟล์ที่ต้องการอยู่ในเซิร์ฟเวอร์หรือไม่แบบนี้เรียกว่าร้องขอแบบ HEAD หากเป็นการร้องขอให้เซิร์ฟเวอร์รับข้อมูลจากไคลเอนต์เรียกว่า POST ซึ่งการร้องขอแบบนี้หมายความว่าไคลเอนต์ต้องการส่งข้อมูลไปให้เซิร์ฟเวอร์นั่นเอง แต่ในความจริงเราสามารถส่งข้อมูลไปยังเซิร์ฟเวอร์ด้วยการร้องขอแบบ GET ก็ได้

วิธีการร้องขอมีให้ใช้หลายวิธี แต่ขึ้นอยู่กับเวอร์ชันของโปรโตคอล HTTP ที่ใช้ หากเป็นเวอร์ชัน 1.0 จะมีวิธีการร้องขอมาตรฐาน 3 วิธี คือ GET ,HEAD และ POST แต่ถ้าใช้โปรโตคอล HTTP เวอร์ชัน 1.1 จะมีวิธีการร้องขอเพิ่มจากเวอร์ชัน 1.0 อีกหลายวิธี เช่น OPTIONS ,PUT ,DELETE หรือTRACE เป็นต้นแต่ในโครงการนี้จะใช้การร้องขอแบบ POST จึงขออธิบายเฉพาะเจาะจงที่แบบ POST อย่างเดียว

### 2.8.2 การร้องขอด้วยเมธอด POST

สำหรับการร้องขอด้วยเมธอด POST จะใช้ในกรณีที่ต้องการส่งข้อมูลจากไคลเอนต์ไปยังเซิร์ฟเวอร์ คือไม่ได้ขอไฟล์จากเซิร์ฟเวอร์ แต่จะส่งข้อมูลไปให้เซิร์ฟเวอร์รับไปทำงาน เมื่อเซิร์ฟเวอร์รับข้อความร้องขอแบบ POST แล้ว ก็จะทราบว่าไคลเอนต์ต้องการส่งข้อมูลมาให้ รูปแบบข้อความร้องขอในบรรทัดแรกจะเขียนคล้ายกับเมธอด GET ดังนี้

---

```
POST /path/file http/1.0
```

---

การร้องขอด้วยเมธอด POST จะมีเงื่อนไขดังต่อไปนี้

- ข้อมูลที่จะส่งไปให้เซิร์ฟเวอร์จะอยู่ภายในบล็อกรหัสข้อมูลดังนั้นจึงต้องมีเฮดเดอร์เพื่อบอกรายละเอียดของข้อมูลในบล็อกข้อมูลแนบไปด้วย เช่น ข้อมูลเป็นแบบไหน (Content-Type) และข้อมูลมีกี่ไบต์ (Content-Length:) เสมอ

- /path/file คือชื่อ โปรแกรม CGI ในเซิร์ฟเวอร์ที่จะทำหน้าที่รับข้อมูลไปประมวลผล

-ข้อความตอบสนองที่เซิร์ฟเวอร์จะส่งกลับให้ไคลเอนต์ จะได้มาจากการทำงานของโปรแกรมหรือ CGI ในเซิร์ฟเวอร์ ดังนั้น CGI ที่รับข้อมูลไปจึงต้องทำหน้าที่ส่งข้อความตอบกลับไปให้ไคลเอนต์

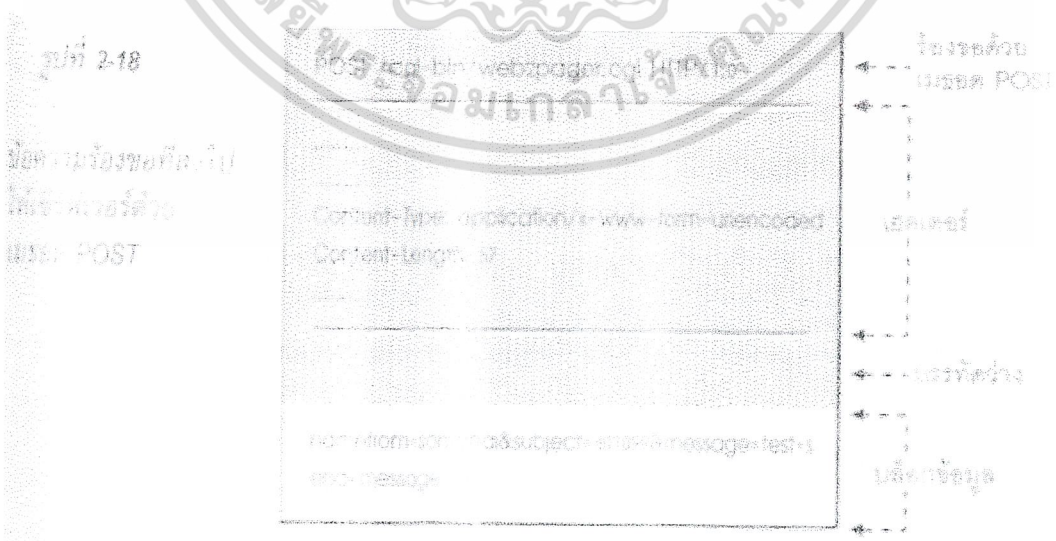
โดยส่วนใหญ่การส่งข้อมูลจากฟอร์มในเว็บเพจไปประมวลผลด้วย CGI ในเซิร์ฟเวอร์ จะใช้เมธอดนี้มากที่สุด ความจริงแล้วการส่งข้อมูลไปยังเซิร์ฟเวอร์สามารถใช้เมธอด GET ได้ แต่มีข้อจำกัดเรื่องความยาวของข้อมูลที่จะส่งไปให้กับเซิร์ฟเวอร์ ถ้าใช้เมธอด POST เพื่อร้องขอส่งข้อมูลไปยังเซิร์ฟเวอร์ เฮดเดอร์ที่ชื่อ Content-Type จะถูกกำหนดให้เป็น application/x-www-form-urlencoded เพื่อบอกแก่เซิร์ฟเวอร์ว่าข้อมูลที่ส่งมาให้มีการเข้ารหัสและเฮดเดอร์ Content-Length จะใช้สำหรับบอกความยาวของข้อมูลที่เข้ารหัสแล้ว เพราะข้อมูลที่กรอกผ่านอินพุตในฟอร์มจะถูกเว็บเบราว์เซอร์เข้ารหัสก่อนส่งเสมอ การเข้ารหัสนี้เรียกว่า URL-encode ซึ่งมีรายละเอียดดังนี้

-แปรงตัวอักษรหรือเครื่องหมายบางตัวให้อยู่ในรูป %XX โดยที่ XX จะเป็นค่ารหัสแอสกีของอักษรตัวนั้นตัวอักษรที่ต้องมีการแปลง เช่น =, &, % และ + เพราะเป็นเครื่องหมายที่ใช้เป็นตัวแบ่งแยกข้อมูลที่จะส่งไปให้เซิร์ฟเวอร์

-เปลี่ยนช่องว่าง (space) ทุกตัวเป็นเครื่องหมายบวก (+)

-รวมชื่อตัวแปรและค่าตัวแปรเข้าด้วยกัน โดยขึ้นกลางด้วยเครื่องหมาย = และขึ้นระหว่างตัวแปรด้วยเครื่องหมาย & เช่น number=152-401099&from=zorkia&message=Just+a+test

ตัวอย่างข้อความร้องขอด้วยเมธอด POST ที่เว็บเบราว์เซอร์สร้างขึ้นเพื่อส่งข้อมูลจากฟอร์มในเว็บเพจให้แก่ CGI ที่ชื่อ web2pager.cgi ในไดเรกทอรี /cgi-bin ของเซิร์ฟเวอร์ <http://203.146.1.214> รับไปทำงาน เป็นดังรูป



รูปที่ 2.8.2 เมธอด POST ที่เว็บเบราว์เซอร์สร้างขึ้นเพื่อส่งข้อมูลจากฟอร์มในเว็บเพจให้แก่ CGI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.8.3 ทำไมต้องเข้ารหัสข้อมูลก่อนส่ง

จากรูป ในช่องอินพุตที่ชื่อ “ข้อความ” ได้มีการป้อนข้อความเป็น “test send message” แต่ข้อความในบล็อกข้อมูลที่เว็บเบราว์เซอร์จะส่งไปให้เซิร์ฟเวอร์กลายเป็น “test+send+message” นั่นคือ ช่องว่างในข้อความจะถูกเปลี่ยนเป็นเครื่องหมาย + หากทุกตัว แล้วถ้าในข้อความมีเครื่องหมาย + ปนอยู่ด้วยจะเกิดอะไรขึ้น

ดังนั้นจึงเป็นกฎเกณฑ์ว่า ต้องมีการเข้ารหัสข้อมูลในฟอร์มก่อนที่จะส่งทุกครั้ง เพื่อกำจัดช่องว่างและอักขระพิเศษต่างๆที่ผู้ใช้อาจป้อนเข้ามา โดยจะแปลงให้อยู่ในรูปของ %XX เมื่อ XX คือเลขฐานสิบหกที่ใช้แทนอักขระนั้นการเข้ารหัสนี้เรียกว่า URL-encoded ตามตารางต่อไปนี้

อักขระ	รหัส %XX
%	%25
&	%26
'	%27
+	%2B
=	%3D
?	%3F

ตัวอย่างเช่น ถ้ามีการป้อนข้อความว่า “love + care = ?” เว็บเบราว์เซอร์จะต้องเข้ารหัสข้อความให้กลายเป็น “love+%2B+care+%3D+%3F”เสียก่อนถึงจะส่งออกไปได้ และไม่ว่าจะเป็นการส่งข้อมูลด้วยเมธอด GET หรือ POST ก็ต้องมีการเข้ารหัสข้อความเสมอ ความยาวของข้อมูลหลังจากการเข้ารหัสแล้วอาจจะมากกว่าหรือเท่ากับข้อมูลต้นฉบับในฟอร์มก็ได้ และเมื่อทาง CGI รับข้อมูลไปแล้วก็ต้องถอดรหัสข้อความกลับเสียก่อนที่จะนำข้อความนั้นไปใช้งานอื่นต่อไป

### 2.9 สร้างเซคเตอร์ให้สามารถติดต่อกับCGI ได้

วิธีการ สั่งให้CGI พุดคุยโต้ตอบด้วย ซึ่ง CGI ก็จะไม่คุยกับเราง่ายๆ ถ้าเราไม่รู้จักรรรมนิยมปฏิบัติที่ถูกต้อง

สิ่งสำคัญที่สุดสำหรับการสั่ง CGI ก็คือ “เซคเตอร์” ครบ อย่างน้อยที่สุดคือเราต้องรู้จักสร้างเซคเตอร์ตามรูปแบบมาตรฐานของ TCP/IP ที่ถูกต้องไม่อย่างนั้น CGIไม่เล่นด้วยครับ อยู่ดีๆ จะไปสั่งให้ webrmd.cgi ทำงานคงไม่มีทางได้แน่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เว็บเบราว์เซอร์ก็เป็นตัวอย่างหนึ่งคือการชำนาญเรื่องการสร้างเซคเตอร์และถอดเซคเตอร์ออกจากข้อความตอบรับมากๆ จะสมมติสถานการณ์อย่างเช่น การเปิดดูเว็บไซต์ของ witty.net ด้วยเบราว์เซอร์ Internet Explorer มาอธิบายเป็นขั้นๆ

1. เปิดโปรแกรมเว็บเบราว์เซอร์ IE (Internet Explorer) ขึ้นมา แล้วเรียกดูที่เว็บไซต์ <http://www.witty.net>
2. IE จะส่งคำร้องขอ ซึ่งประกอบไปด้วยเซคเตอร์ชนิด GET เพื่อร้องขอข้อมูลจากเซิร์ฟเวอร์ witty.net โดยที่ตัวอย่างข้อมูลภายในเซคเตอร์ของคำร้องขอเป็นดังนี้
3. เมื่อเว็บเซิร์ฟเวอร์ได้รับข้อมูลจากการร้องขอจาก IE ก็จะส่งข้อมูลตอบกลับออกมา ซึ่งเป็นข้อมูลที่มีทั้งเซคเตอร์ และไฟล์ HTML ซึ่งเป็น โฮมเพจของหน้าแรก ของ witty.net เอง โดยเซคเตอร์จะเป็นดังนี้
4. IE จะรับข้อมูลมา แล้วตัดเซคเตอร์ทิ้งไป เหลือแต่เฉพาะข้อมูลที่เป็นเว็บเพจของ witty.net เท่านั้น เพื่อจะได้แปลแท็กคำสั่ง HTML และแสดงผ่านทางเว็บเบราว์เซอร์ต่อไป

เพราะฉะนั้น ถ้าเราอยากส่งข้อมูลไปยังไฟล์ webcmd.cgi ให้ได้เราต้องเขียนแบบเซคเตอร์ให้เหมือนกับเซคเตอร์ของข้อมูลส่งมาจากไฟล์ webcmd.cgi

### คำอธิบายโปรแกรม

การทำงานของไฟล์ easycall.cls จะเริ่มต้นจาก...

1. ประกาศตัวแปรเพื่อจะได้กลายเป็นค่าพรอเพอร์ตี้ โดยจะมีการรับข้อมูลดังต่อไปนี้  
 พรอเพอร์ตี้ FromName สำหรับกำหนดชื่อผู้ส่ง  
 พรอเพอร์ตี้ Body สำหรับกำหนดข้อความ  
 พรอเพอร์ตี้ PageNumber สำหรับกำหนดค่าหมายเลขเพจเจอร์  
 ส่วนพรอเพอร์ตี้ Result จะเป็นผลลัพธ์ของการส่งข้อความ ถ้าส่งได้ก็เป็น True แต่ถ้าส่งไม่ได้จะเป็น False
2. ประกาศตัวแปรอ็อบเจ็คต์ objWinsock ใช้แทนคอนโทรลเลอร์ Winsock โดยกำหนดให้อ็อบเจ็คต์นี้ มีการใช้อีเวนต์ (event) ต่างๆ ของคอนโทรลเลอร์ได้ด้วย
3. ฟังก์ชัน SendMessage เป็นฟังก์ชันหลักของคอมโพเนนต์นี้ ภายในฟังก์ชันจะเรียกโปรแกรมย่อยมาทำงานทีละตัว ตั้งแต่ติดต่อเซิร์ฟเวอร์ (Connect) และส่งข้อมูล (SendRequest) ไปจนถึงปิดการทำงานของอ็อบเจ็คต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. โปรแกรมย่อย Connect สำหรับติดต่อกับเซิร์ฟเวอร์ ในกรณีนี้เราต้องการส่งข้อมูล ไปหา เซิร์ฟเวอร์ของ Easycall ก็กำหนดค่า objWinsock.RemoteHost = "easycall.lenso.co.th" และ objWinsock.RemotePort = 80
5. วงรอบสั่งให้ทำงานตลอด ตรวจจับที่ยังเชื่อมต่อกับเซิร์ฟเวอร์อยู่ เนื่องจากเราต้องมีการส่งข้อมูลไปยังเซิร์ฟเวอร์ แล้วก็ต้องรอการตอบกลับ โดยไม่สามารถให้โปรแกรมทำงานเสร็จได้ ภายในช่วงเวลาเดียว
6. objWinsock\_Connect เป็น โปรแกรมย่อยของอิวেন্টประจําคอนโทรล Winsock โปรแกรมย่อยนี้จะทำงานอยู่ตลอดตรวจจับที่มีการติดต่อกับเซิร์ฟเวอร์
7. SendRequest เป็น โปรแกรมย่อยสำหรับส่งข้อมูลไปยังเซิร์ฟเวอร์ของ Easycall ซึ่งประกอบไปด้วยส่วนที่เป็นเฮดเคอร์และส่วนที่เป็นข้อมูลจริงๆ คือ หมายเลขเพจเจอร์, ชื่อผู้ส่ง และข้อความ
8. สร้างเฮดเคอร์ชนิด Post สำหรับส่งข้อมูล โดยที่เรามีจุดประสงค์จะเลียนแบบให้ใกล้เคียงกับเฮดเคอร์ที่มาจากเว็บไซด์ Easycall องให้มากที่สุด ซึ่งน่าจะเป็นดังนี้

```

post http://203.170.228.139/web2sms.php HTTP/1.1
Referer:http://easycall.lenso.co.th/english/sendpage.html
User-Agent:Mozilla/4.0
Accept:*/*
Content-Type:application/x-www-form-urlencoded
Content-Length:" ขนาดความยาวของข้อมูล

```

โดยที่ซอร์ซโค้ดแต่ละบรรทัดมีความหมายดังนี้

บรรทัดที่ 1 คือการส่งข้อมูลไปยัง

บรรทัดที่ 2 คือตำแหน่งเว็บเพจอ้างอิง ซึ่งบอกแหล่งที่ว่าส่งข้อมูลมาจากที่ใด โดยในกรณีนี้มาจากไฟล์ของ Easycall เองคือ

บรรทัดที่ 6 คือความยาวของข้อมูล

สำหรับ vbCrLf คือเครื่องหมายเว้นบรรทัดซึ่งก็เหมือนกับคำสั่ง Chr(13)&Chr(10) ใน VBScript

9. ส่งข้อมูลด้วยคำสั่ง objWinsock.SendData

## 2.10 โปรแกรม Ethereal

The screenshot displays the Ethereal interface with a captured packet. The packet list shows two entries:

No.	Time	Source	Destination	Protocol	Info
7	3.306438	modem090.net.kmit1.ac	203.170.228.139	HTTP	POST /web2sms.php HTTP/1.1
9	4.106202	203.170.228.139	modem090.net.kmit1.ac	HTTP	HTTP/1.0 200 OK

The selected packet (No. 7) is expanded to show the following details:

- Transmission Control Protocol:** Src Port: 1212 (1212), Dst Port: 80 (80), Seq: 1199492525, Ack:
- Hypertext Transfer Protocol:**
  - POST /web2sms.php HTTP/1.1\r\n
  - Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, application/vnd.ms-powerpoint, .
  - Referer: http://www.gsm.ais900.com\r\n
  - Accept-Language: th\r\n
  - Content-Type: application/x-www-form-urlencoded\r\n
  - Accept-Encoding: gzip, deflate\r\n
  - User-Agent: Mozilla/4.0 (compatible; MSIE 5.5; windows 98; win 9x 4.90)\r\n
  - Host: www.gsm.ais900.com\r\n
  - Content-Length: 73\r\n
  - Connection: Keep-Alive\r\n
  - \r\n
  - data (73 bytes)

The packet bytes pane shows the raw data in hexadecimal and ASCII:

```

0190 40 33 49 43 20 33 2e 33 30 20 37 69 6e 64 61 77  MSIE 5.5 ; window
01a0 73 20 39 38 3b 20 57 69 6e 20 39 78 20 34 2e 39  s 98; win 9x 4.9
01b0 30 29 0d 0a 48 6f 73 74 3a 20 77 77 77 2e 67 73  0)..Host : www.gs
01c0 6d 2e 61 69 73 39 30 30 2e 63 6f 6d 0d 0a 43 6f  m.ais900 .com.Co
01d0 6e 74 65 6e 74 2d 4c 65 6e 67 74 68 3a 20 37 33  ntent-Le ngth: 73
01e0 0d 0a 43 6f 6e 6e 65 63 74 69 6f 6e 3a 20 4b 65  ..Connec tion: Ke
01f0 65 70 2d 41 6c 69 76 65 0d 0a 0d 0a 73 7a 47 73  ep-Alive ....szGS
0200 6d 4e 6f 3d 31 39 31 33 35 38 33 37 26 73 7a 47  mNo=1913 5837&szG
0210 73 6d 50 77 64 3d 26 73 7a 4d 65 73 67 3d 74 65  smPwD=&szMsg=te
0220 73 74 2b 73 65 6e 64 2b 6d 65 73 73 61 67 65 26  st+send+ message&
0230 73 7a 4c 65 6e 67 74 68 3d 31 36 26 33 65 6e 64  szLength =16&send
0240 3d 53 65 6e 64  =Send
  
```

The interface also shows a filter bar with "Data (data)" selected and a status bar at the bottom with "429" packets displayed.

รูปที่ 2.10 capture –Ethereal

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปคือโปรแกรม Ethereal ที่ทำการ capture เว็บของ GSM อธิบายได้ดังนี้

- ช่วงบนสุดของโปรแกรมเป็นการบอกให้รู้ข้อมูลจากฟอร์มถูกส่งไปให้ CGI ด้วยเมธอด POST
- ช่วงกลางของโปรแกรมเป็นการบอกให้รู้ว่าเซคเตอร์ของเว็บนี้เขียนอย่างไร มีข้อความที่ส่งไปทั้งหมด 73 Bytes
- ช่วงล่างสุดของโปรแกรมเป็นการบอกให้รู้ว่าข้อมูลที่ส่งไปให้เว็บมีอะไรบ้างเขียนอย่างไร

เมื่อได้ซอร์ซโค้ด HTML ส่วนที่เป็นฟอร์มออกมาแล้ว ให้นำไปเขียนลงในโปรแกรมวิซวลเบสิก ในฟอร์มของเรา ดังนี้

```
Data = "szGsmNo=" & Text4.Text & "&szGsmPwd=" & "" & "&szMesg=" &
Text2.Text & "&szLength=" & szLen & "&Send=Send"
size = Len(Data)
Alldata = "post http://203.170.228.139/web2sms.php HTTP/1.1" & vbCrLf
Alldata = Alldata & "Referer:http://www.gsm.ais900.com" & vbCrLf
Alldata = Alldata & "User-Agent:Mozilla/4.0" & vbCrLf
Alldata = Alldata & "Accept:*/*" & vbCrLf
Alldata = Alldata & "content-Type:application/x-www-form-urlencoded" & vbCrLf
Alldata = Alldata & "Content-Length:" & size & vbCrLf & vbCrLf
Alldata = Alldata & Data & vbCrLf
WinsockClient.SendData Alldata
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.11 Wireless Application Protocol (WAP)

จากความเจริญเติบโตอย่างรวดเร็วของทั้ง Internet และ การรวมเอา ความสามารถเชิง Internet มาใช้กับเทคโนโลยี ของ โทรศัพท์มือถือทำให้ ณ ปัจจุบันนี้ โทรศัพท์มือถือที่เราใช้กันอยู่ทำอะไรๆ ได้มากกว่า แค่ใช้โทรออกหรือรับสาย เข้าเท่านั้นแล้ว ผู้คนเริ่มให้ความสนใจ และให้ความสำคัญ มากขึ้น กับสิ่งที่จะ ช่วยอำนวยความสะดวกสบาย ในการที่จะ ติดต่อค้นหาข้อมูล และบริการอื่นๆ บน Internet ซึ่งสามารถ ทำได้แม้แต่ในขณะที่กำลังเดินทางอยู่ โดยผ่านทางโทรศัพท์มือถือเพียงเครื่องเดียว แต่จะอย่างไรล่ะ ถึงจะนำเอา ความสามารถต่างๆ ที่อยู่บน Internet เหล่านั้นจับมาใส่ลง ในเครื่องโทรศัพท์มือถือได้ และ ยังต้องทำให้ใช้งานได้ง่ายด้วย เนื่องจากโทรศัพท์มือถือ นั้น มีข้อจำกัดในด้านการป้อนข้อมูล เข้า และการแสดงผล Wireless Application Protocol หรือ WAP จึงได้ถือกำเนิดขึ้น เพื่อช่วยให้ ผู้ใช้โทรศัพท์มือถือ สามารถใช้ความสามารถดังกล่าวข้างต้นเหล่านี้ได้

### 2.11.1 จุดกำเนิดแห่ง WAP

WAP ( Wireless Application Protocol ) เป็น Protocol ที่เป็นมาตรฐานสากล ที่เกิดจากความร่วมมือกันของ หลายๆบริษัท ผู้ผลิตโทรศัพท์มือถือ เพื่อนำเอาลูกเล่นหรือ ความสามารถต่างๆ ของ Wireless Application และ ของทางด้าน Internet ให้มาใช้ได้ บนเครื่องโทรศัพท์มือถือ WAP จะทำให้ผู้ใช้โทรศัพท์มือถือ ใช้งานทางด้าน Internet ทั่วๆไปได้ เหมือนๆกับใช้งานผ่านทางเครื่องคอมพิวเตอร์ โดยที่ WAP นั้นไม่ต้องการ CPU ที่มีประสิทธิภาพ สูงๆ ไม่ต้องการหน่วยความจำมากๆ ไม่ต้องการแหล่งพลังงานมาก แต่ก็มีข้อจำกัดอยู่หลายเรื่องด้วยกันที่อาจทำให้ผู้ใช้รู้สึกแปลกและ ตัดขาดบ้าง เช่น ข้อจำกัดด้าน Bandwidth, มี Display ขนาดเล็ก และ มีส่วนของการป้อนข้อมูลเข้า ( Input ) ที่แตกต่างจากการใช้งาน บนคอมพิวเตอร์อยู่พอสมควร แต่จุดเด่นของ WAP นั้นก็คือทำให้ ใช้งานได้ผ่านทางโทรศัพท์มือถือ ซึ่งสามารถพกพา หรือนำไปใช้งาน ณ ที่ไหนๆ ก็ได้ ซึ่งจุดนี้ก็น่าจะหักล้างกับข้อจำกัดต่างๆลงได้

WAP นั้นถูกออกแบบมาเพื่อให้ใช้กับอุปกรณ์ไร้สายต่างๆ ไม่จำกัดเพียงแค่โทรศัพท์มือถือเท่านั้น หากยังรวมถึง วิทยุติดตามตัว ( Pager ), วิทยุรับส่งที่เรียกว่า Two-Way Radio, Smartphone และรวมไปถึงอุปกรณ์สื่อสารต่างๆ ตั้งแต่ระดับ Low-End จนถึง High-End เลยทีเดียว ซึ่ง ระบบ Network ที่ใช้กับ WAP ได้นั้น ก็ใช้ได้หลากหลายรูปแบบ ทั้ง CDPD, CDMA, GSM, PDC, PHS, TDMA, FLEX, ReFLEX, iDEN, TETRA, DECT, DataTAC และ Mobitex. WAP เป็น Protocol เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับการสื่อสาร ซึ่งสามารถใช้งานได้กับระบบปฏิบัติการ ( OS : Operating ) ต่างๆ ได้หลากหลาย ทั้ง PalmOS, EPOS, Windows CE, FLEXOS, OS/9, JavaOS และอื่นๆ อีก WAP นั้นจะช่วยสนับสนุน Bearer หลักๆ ในการส่ง Message เช่น Short Message Service ( SMS ) , Circuit Switched Data, Unstructured Supplementary Services Data ( USSD ) และ ในอนาคตอันใกล้ ก็จะสนับสนุน General Packet Radio Services ( GPRS ) อีกด้วย และ นอกจากนี้ ก็ยังมีการนำเอาเทคโนโลยีใหม่ๆ ต่างๆ มาใช้กับโทรศัพท์มือถือ มากขึ้น โดยอาศัย WAP และ เทคโนโลยีที่เกี่ยวกับการส่ง Message เข้ามาช่วยในการทำงานอีกด้วย เช่น HSCSD, EDGE และ WCDMA<sup>๙</sup>

### 2.11.2 มาตรฐานสำหรับ WAP

ณ ปัจจุบันนี้ ทางผู้พัฒนา ก็ได้วางมาตรฐานของ WAP ไว้ เป็น Version 1.1 ซึ่งก็ได้วางไว้ตั้งแต่เมื่อราวๆ เดือนมิถุนายนปีที่แล้ว ( ค.ศ. 1999 ) และได้ใช้มาตรฐานนี้เรื่อยมา แต่ในขณะเดียวกัน ก็กำลังเตรียมการสำหรับ Version 1.2 ด้วย โดยได้พัฒนา ปรับปรุง แก้ไขข้อบกพร่องอีกหลายอย่าง จาก Version 1.1ซึ่งก็ได้คาดการณ์ว่าจะเสร็จเรียบร้อยและ ได้รับการยอมรับจากทางผู้พัฒนาในราว เดือนพฤศจิกายน ปีที่แล้วเช่นกัน หลังจากนั้นจึงจะได้ให้กับสมาชิก ทำการทดลองเพื่อช่วยวิจารณ์แล้ว จึงจะได้ใช้เป็นมาตรฐานแทนที่ Version 1.1 ต่อไป

WAP Version 1.1 นั้น ได้ทำการแก้ไขปรับปรุงจาก Version 1.0 โดยที่ยังคง Function ต่างๆ ไว้เช่นเดิม เพียงแต่ Version 1.0 นั้น ยังเป็นเพียงมาตรฐานที่ใช้ทดลองกันภายในกลุ่มบริษัท ผู้พัฒนาเพื่อทำการแก้ไขมาตรฐานบางอย่าง หลังจากนั้นจึงวางเป็น มาตรฐาน Version 1.1 เพื่อใช้เป็นสากลต่อไป โดยที่จุดที่ทำการ แก้ไขนั้น หลักๆ ก็คือทำการแก้ไขในส่วนของ WML ( Wireless Markup Language ) เพื่อให้ใช้ได้กับ มาตรฐาน XHTML ที่กำลังจะเกิดขึ้นต่อไปนั่นเอง

สำหรับ WAP Version 1.2 นั้น ก็ยังคงใช้ Function อะไรๆต่างๆ เหมือนกับ Version 1.1 แต่ก็ได้ใช้ข้อมูล ความเห็นต่างๆของ ผู้ทดสอบผู้พัฒนาในกลุ่มบริษัท มาปรับปรุงแก้ไข และได้เพิ่มเติม ในส่วนของ WAP Push Architecture ซึ่งเป็นภาคขยายของ Wireless Telephony Applications ( WTA ) นอกจากนี้ก็ยังได้ เพิ่มเติมให้รองรับกับ Technology ทางด้าน Network มากขึ้นอีกด้วย

สำหรับอุปกรณ์ Wireless ที่รองรับ หรือใช้งานกับมาตรฐาน WAP ใต้นั้น ก็เริ่มทยอยออกมาสู่ท้องตลาดบ้างแล้ว โดยอุปกรณ์ หลักๆ ก็คือ Mobile Phone ที่รองรับมาตรฐาน WAP หรือที่เรียกว่า WAP Phone ก็ได้เริ่มวางตลาดแล้วตั้งแต่ปลายๆ ปีที่แล้ว และในปีนี้ ( ค.ศ. 2000 ) ก็คาดว่า จะมีการจำหน่าย WAP Phone หลากหลาย รุ่นและยี่ห้อมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.11.3 WAP กับมาตรฐาน Internet

WAP นั้นจริงๆ แล้วก็ได้พัฒนาขึ้นมาจากมาตรฐาน Internet นี้เอง ซึ่งก็ใช้พื้นฐานเดียวกับ Internet คือ XML, UDP และ IP ซึ่ง WAP Wireless Protocol นี้ ก็ยังได้พัฒนาโดยใช้มาตรฐานของ Internet พวก HTTP และ TLS ด้วย เพียงแต่ ได้ทำการปรับปรุงแก้ไขบางส่วน เพื่อให้ใช้งานได้ด้วยอุปกรณ์ ไร้สาย ( Wireless Device ) ซึ่งมีข้อจำกัด ทั้งทางด้าน หน่วยแสดงผล ( Display ) ซึ่งมีขนาดเล็ก หรือในส่วนของ การป้อนข้อมูลเข้าที่ยากกว่า การใช้เครื่อง PC ด้วยเช่นกัน WML จึงเป็นผลที่เกิดขึ้นจาก การปรับปรุงแก้ไข มาตรฐานของการ เขียน Markup Language ซึ่งไม่สามารถใช้ HTML บนอุปกรณ์ไร้สาย จากเหตุผลข้างต้น

XML หรือ Extend Markup-Languaae นั้น เป็น Meta-Language ซึ่งกำหนดขึ้น โดย W3C ( World Wide Web Consortium ) ซึ่งหากจะมองว่าเป็นส่วนขยายของ SGML ( Standard Generalized Makriup Language ) ก็ได้ เพียงแต่ XML นี้ เป็นเพียงกฎ เงื่อนไข ต่างๆ ในการสร้าง มาตรฐาน ภาษาที่ใช้ในการ เขียน Application เท่านั้น คือยังไม่กำหนด Format ต่างๆ แน่ชัด ส่วน WML หรือ HTML นั้น มีการกำหนด Format ต่างๆ Tag ต่างๆ เป็นที่แน่ชัดแล้ว และมีการเจาะจง ให้ใช้ในด้านใดด้านหนึ่ง ดังนั้นจึงมองได้ว่า WML หรือ HTML นั้น เป็น XML Application

นอกจากนี้ WML ก็ยังมี WMLScript ซึ่งก็อาศัยหลักการ และโครงสร้าง ที่คล้ายกับ JavaScript หรือ ECMAScript เพียงแต่ทำการ ปรับปรุงให้ใช้งานได้อย่างเหมาะสมกับระบบ อุปกรณ์ไร้สายด้วย โดย WMLScript นี้ จะใช้หน่วยความจำ และ CPU ค่อนข้าง ได้ตัดเอา Function ต่างๆ ที่ไม่เกี่ยวข้อง หรือไม่มีความจำเป็นต้อง ใช้กับในระบบนี้ออก ไปอีกไม่น้อย

นอกจากนี้ WAP ก็ยังได้รวมเอาความสามารถเพื่อช่วย ในการรักษาความปลอดภัย ของข้อมูล ทั้งด้านการ Authentication และ ด้านการ Encryption ( เข้ารหัสข้อมูล ) ซึ่งเรียกว่า WTLS อีกด้วย

## 2.11.4 ศัพท์อื่นๆที่เกี่ยวข้อง

### Micro-Browser

เป็น Software ทางฝั่ง Client ( Client Software ) ที่ฝังอยู่ในตัว SIM ซึ่ง ออกแบบมาเพื่อให้เครื่องอุปกรณ์มือถือ สามารถใช้งานกับ Service ด้าน Internet ได้

### HSCSD

หรือ High Speed Circuit Switched Data จะช่วยลดเวลาในการส่งถ่ายข้อมูล ระหว่างโทรศัพท์มือถือกับ Internet และทำให้ช่วย การส่งถ่ายข้อมูลขนาดใหญ่ๆ นั้นทำได้แทบจะทันที พร้อมทั้งรับประกันความถูกต้องของข้อมูลที่มีการส่งถ่าย ออกไปอีกด้วย ด้วยความรวดเร็วในการส่งถ่ายข้อมูลระดับเดียวกับ ISDN บวกกับความสะดวกสบายต่างๆ ก็น่าจะช่วยดึงดูดให้มีผู้มาใช้งานด้านนี้ มากขึ้น และแน่นอนเมื่อมีผู้ใช้งานมากขึ้น ก็จะทำให้บริษัทมีรายได้เพิ่มมากขึ้นด้วย

### GPRS

หรือ General Packet Radio Service ซึ่งเป็นตัวสร้าง Direct Link ระหว่างโทรศัพท์มือถือ และ Internet เข้าด้วยกัน ซึ่ง GPRS นี้ จะทำให้สามารถ มีอัตราในการส่งถ่ายข้อมูล ได้มากที่สุดถึง 100 kbps เลยทีเดียว GPRS นั้น จะสร้าง IP ให้อย่างต่อเนื่อง และมี Bandwidth กว้าง ทำให้สามารถ นำเอา Application ต่างๆ มาใช้ได้อย่างหลากหลาย อาทิ Multimedia Messaging, Wireless Corporate Intranet, Remote Control และ อื่นอีกมากมาย โดยผู้ใช้ จะเสียค่าใช้จ่าย เพียงแค่ในส่วนของจำนวนข้อมูลที่มีการส่งถ่ายเท่านั้น ทั้ง HSCSD และ GPRS จะทำงานในลักษณะเกือหนุนกัน เพื่อให้ทำงานได้อย่าง กว้างขวางมากยิ่งขึ้น โดยแต่ละตัวจะสนับสนุน Application ที่แตกต่างกัน ซึ่งแต่ละ Application นั้นก็เป็นส่วนเติมเต็มให้กันและกัน เพื่อนำไปสู่ยุคใหม่ ( ยุคที่ 3 ) ของระบบ Cellular Radio ( 3G : 3rd Generation ) ที่กำลังจะมาถึงในอนาคตอันใกล้

### EDGE

หรือ Enhanced Data rates for Global Evolution ซึ่งคาดว่าจะมีใช้อย่าง แพร่หลายในปี 2001 จะช่วยให้โทรศัพท์มือถือสามารถใช้บริการด้าน Multi-Media ต่างๆ ได้มากยิ่งขึ้น เช่น VDO Conferent , การถ่ายโอนข้อมูลขนาดใหญ่ๆ , การต่อเข้ากับ Internet ผ่านทางเครื่อง โทรศัพท์มือถือเอง หรือผ่านทางเครื่อง Note Book หรือ เครื่อง PC Computer ด้วย EDGE นี้ ผู้ใช้บริการจะสามารถ

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น เมื่อผู้ใดเห็น ขัดแย้งหรือมีข้อสงสัยประการใด กรุณาแจ้งมาที่ฝ่ายวิชาการ  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จัดส่ง VDO-Postcard ให้แก่เพื่อนๆ หรือครอบครัวผ่านทาง Internet ด้วยเครื่องโทรศัพท์ มือถือเพียงเครื่องเดียว ได้อย่างไม่ยากเย็น เทคโนโลยีของ EDGE นี้ ใช้ได้ทั้งระบบ GSM และ TDMA ซึ่งก็เป็นระบบ Digital ด้วยกันทั้งคู่ แต่แตกต่างกันเรื่องระบบ Cellular และโครงสร้างภายใน โดย EDGE นั้นสามารถใช้ได้กับในระบบ Network และความถี่ของสัญญาณที่ใช้ในปัจจุบันนี้ได้ เพียงแต่อาจต้องมีการแก้ไข Hardware ด้าน Network และ แก้ไข Software บ้างเล็กน้อย

### WCDMA

หรือ Wideband Code Division Multiple Access โดยที่ Wideband นั้นจะมีความสามารถอยู่ระหว่าง Narrowband และ Broadband Narrowband คือความสามารถเชิงโทรคมนาคม ที่รองรับการส่งข้อมูลไม่กี่จำพวก เช่น ข้อมูลเสียง,รูปภาพที่ถอดแบบจากต้นฉบับ, Slow-Scan VDO และ ข้อมูลที่มีการ ส่งถ่ายด้วยความเร็วต่ำๆ ในระดับ Kilobit เท่านั้น โดยคุณภาพนั้นก็อยู่ใน ระดับที่ยอมรับได้ ต่างกับ Broadband ที่สามารถจัดส่งข้อมูลได้ในระดับ Megabit ซึ่งทำให้สามารถส่งข้อมูลพวก Full-Color Image, หรือ Full-Motion VDO ได้ โดยที่ความเร็วในการส่งถ่ายโดยรวมจะเร็วกว่าแบบ Narrowband ตั้งแต่ 100 ถึง 1,000 เท่าเลยทีเดียว โดยสรุป Wideband นั้นก็จะสามารถรองรับการส่งถ่าย ข้อมูลได้ถึงระดับ Megabit ซึ่งก็สามารถถ่ายโอนข้อมูลจำพวกภาพเคลื่อนไหวได้ หรือทำให้การ Download ข้อมูล ข่าวสารต่างๆ เป็นไปได้อย่างรวดเร็ว WCDMA นั้นกล่าวโดยรวมก็คือเป็น Mode ในการเข้าถึงข้อมูล ที่ใช้สำหรับ Radio Interface ของ 3G ( 3rd Generation ของ Cellular Radio ) ลักษณะพิเศษของมันคือ จะมีความจุของข้อมูลสูง , มี Cell Radius ขนาดเล็ก และ มีการส่งถ่ายของ Spectrum Radio แบบกระจายเป็นวงกว้าง

## 2.12 การดู WAP Site

หนทางเดียวที่จะไม่ใช้โทรศัพท์มือถือดู WAP Site ก็มีหลายแบบคือ ใช้โปรแกรมจำลอง WAP Browser หรือ WAP Emulator ในเครื่องพีซีเรียกดูผ่านทางอินเทอร์เน็ต ซึ่งมีรายละเอียดดังนี้

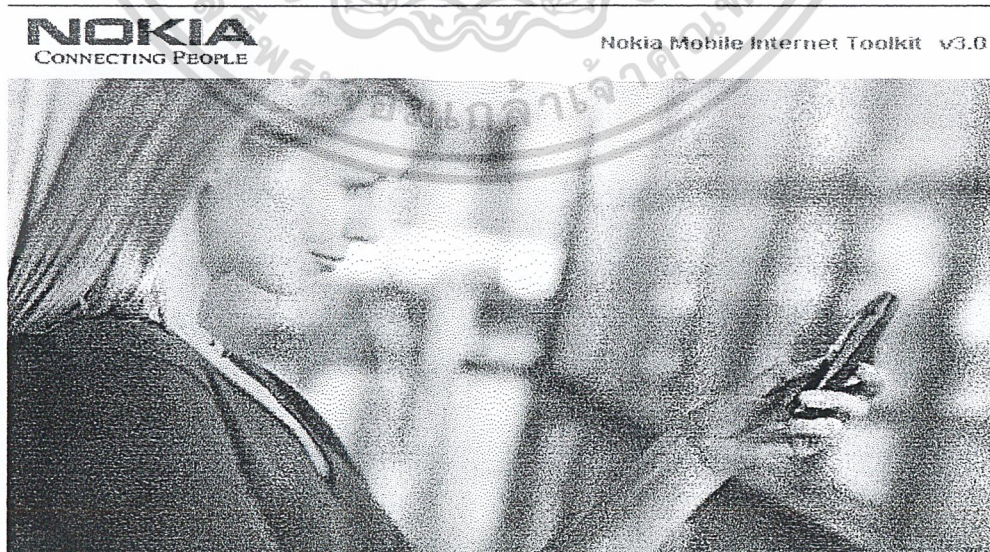
- **WAP Browser**

เป็นซอฟต์แวร์ที่ทำงานในเครื่องพีซี ลักษณะหน้าตาเหมือนกับเว็บเบราว์เซอร์อย่าง Netscape หรือ IE มาก ตัวอย่างเช่น WinWAP, Klondike WML Browser เป็นต้น

- **WAP Emulator**

เป็นซอฟต์แวร์ที่ทำงานในเครื่องพีซีเช่นกัน แต่โชว์หน้าตาซึ่งจำลองมาจากโทรศัพท์มือถือจริงๆ โปรแกรม WAP Emulator อาจแบ่งได้เป็น 2 กลุ่ม 2 แบบ คือ แบบที่ทำงานเดียวกับแบบที่ทำงานเป็นส่วนหนึ่งของโปรแกรมจำลอง Toolkit (Toolkit คือชุดของซอฟต์แวร์ที่ใช้ในการพัฒนาและทดสอบ WAP Application ที่เขียนขึ้นมา)

โปรแกรม WAP Emulator ที่ทำงานเดี่ยวๆ มีอย่างเช่น Ericsson R380 Emulator ฯลฯ ส่วน WAP Emulator ที่ทำงานอยู่ใน Toolkit มีอาทิเช่น WAP Emulator แบบ Blueprint และแบบ Nokia 7110 ที่ทำงานบน Nokia WAP Toolkit 3.0 ฯลฯ



Toolkit Application Loading, Please Wait ...

รูปที่ 2.12 Nokia WAP Toolkit 3.0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.13 เครื่องมือหลักๆที่มีใน Nokia WAP Toolkit

1. WAP Editor ซึ่งมีทั้ง WML Editor และ WMLScript Editor
2. WBMP Graphics Editor ใช้ในการสร้างรูปภาพขาว-ดำแบบ WBMP รวมทั้งเปลี่ยนรูปภาพที่มีนามสกุล GIF หรือ JPG มาเป็นแบบ WBMPได้ด้วยซึ่งเครื่องมือตัวนี้ทุกคิกของค่ายอื่นๆยังไม่มี
3. WAP Emulator หรือ Device Emulator ซึ่งจำลองรูปแบบของโทรศัพท์จริงๆในการแสดงผลเอกสาร WML ต่างๆ
4. เครื่องมือช่วยติดตามสถานะของการติดต่อร้องขอข้อมูล จากเว็บเซิร์ฟเวอร์ซึ่งส่วนนี้ใช้ทดแทนการขาดหายไปของ Information Window ที่มีใน UP.SDK, WapIDE หรือ MADK ได้บ้าง
5. Nokia WAP Server Simulator ซึ่งเป็น WAP Gateway ที่แถมมา เป็นฉบับย่อของผลิตภัณฑ์ Nokia WAP Server ซึ่งมีอยู่จริงของทาง โนเกีย
6. ตัวอย่างแอปพลิเคชันที่เขียนด้วย WML, WMLScript ซึ่งปกติแล้วจะอยู่ในไดเรกทอรี C:\Program File\Nokia\WAP\_Toolkit\_3.0\Samples

สำหรับคุณสมบัติพื้นฐานของเครื่องพีซีที่จะนำมาทดลองพัฒนา WAP Application ด้วย Nokia WAP Toolkit 3.0 จะต้อง...

- ใช้ Windows 95/98 หรือ Windows NT 4.0 Server Pack 3
- ความเร็วในการประมวลผลของพีซีอย่างต่ำ 266 MHz
- RAM อย่างน้อย 64 MB
- เนื้อที่ว่างในฮาร์ดดิสก์ อย่างน้อย 20 MB
- ความละเอียดของหน้าจอ 1024 x 768 และความละเอียดของสี 16 บิต
- ติดตั้ง Java Runtime Environment เวอร์ชัน 1.2.2 ขึ้นไป

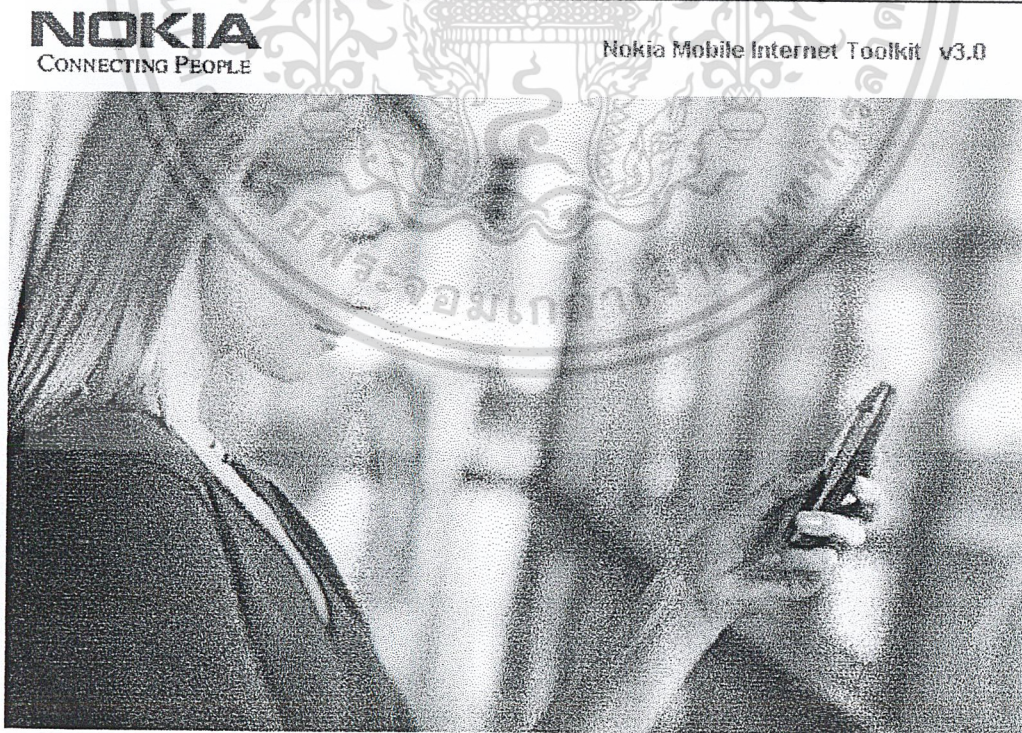
### 2.13.1 ไฟล์ที่ใช้ในการติดตั้ง Nokia WAP Toolkit 3.0 มี 2 ไฟล์

1. Java Runtime Environment เวอร์ชัน 1.2.2 ขึ้นไป (jre-1\_2\_2\_005win.exe) ซึ่งหาได้จาก <http://www.javasoft.com/products/jdk/1.2/jre/index.html>
2. Nokia Toolkit2\_0.zip ซึ่งหาได้จาก [www.nokia.com/corporate/wap/sdk.html](http://www.nokia.com/corporate/wap/sdk.html) โดยต้องสมัครสมาชิกกับทาง Developer zone ที่เว็บไซต์ของโนเกียเสียก่อนจึงจะดาวน์โหลดได้

เมื่อติดตั้ง Java Runtime Environment (jre) เสร็จเรียบร้อยแล้ว ก็ unzip ไฟล์ NokiaToolkit3\_0.zip ออกมา แล้วติดตั้งโดยดับเบิลคลิกที่ไฟล์ setup.exe หลังจากนั้นโปรแกรมจะให้เราเลือก Setup Type ซึ่งจะมี 3 แบบ คือ Typical, Compact และ Custom ขอแนะนำให้เลือกแบบ Typical เพื่อติดตั้งส่วนประกอบต่างๆ ทั้งหมดที่มีในชุดนี้ เมื่อติดตั้งเสร็จแล้ว จะมีการสร้าง shortcut ไว้ที่เมนู Start -> Program ของระบบ windows ชื่อว่า Nokia WAP Toolkit 3.0

### 2.13.2 หน้าต่าง 2 แบบของ Nokia WAP Toolkit

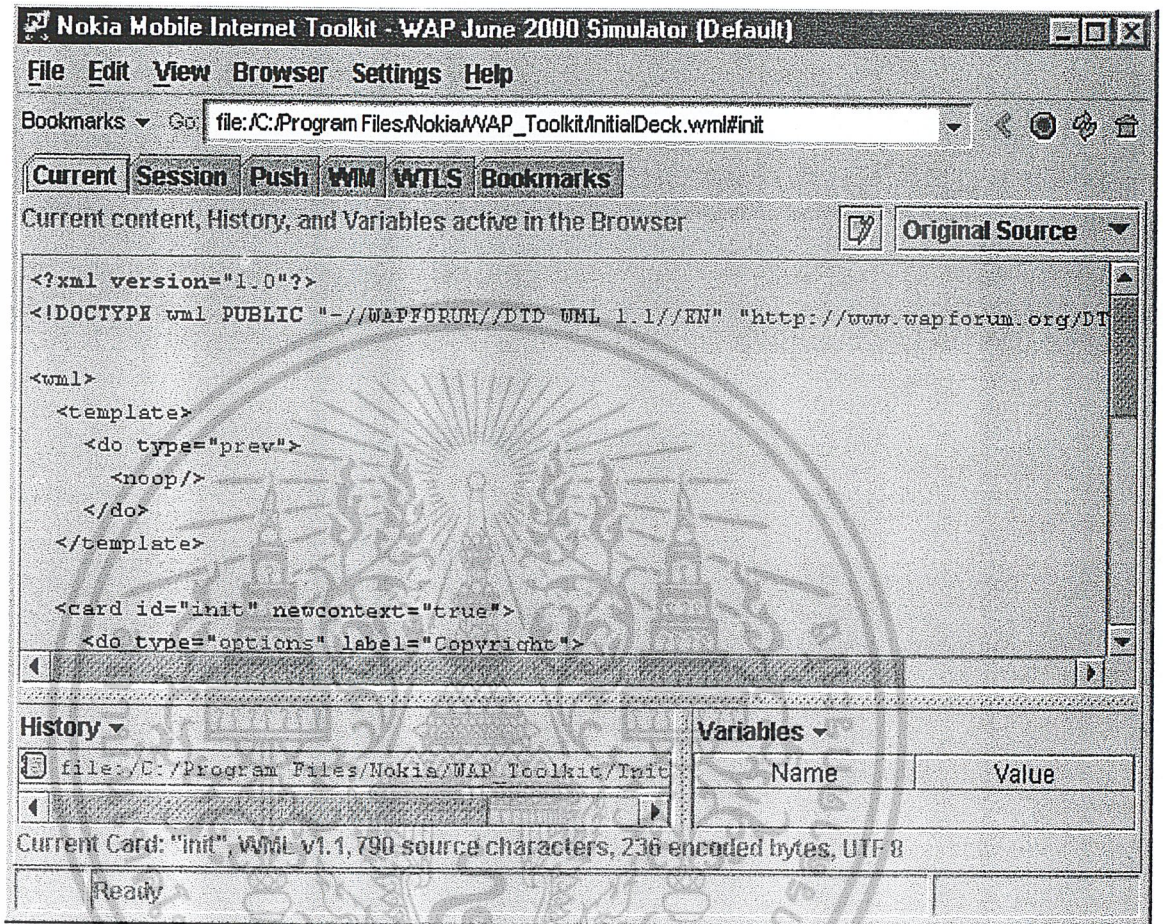
ให้ทดลองเรียกโปรแกรม WAP Toolkit จากเมนู Start -> Program -> Nokia WAP Toolkit 3.0 -> WAP Toolkit โปรแกรมก็จะเริ่มโหลดข้อมูล และขณะโหลดก็จะแสดงหน้าจอ ดังภาพ



รูปที่ 2.13.2 รูปขณะโหลดโปรแกรม Nokia WAP Toolkit 3.0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นจะพบหน้าต่าง 2 แบบ ดังภาพข้างล่างนี้ หน้าต่างนี้เรียกว่า “Toolkit Window”



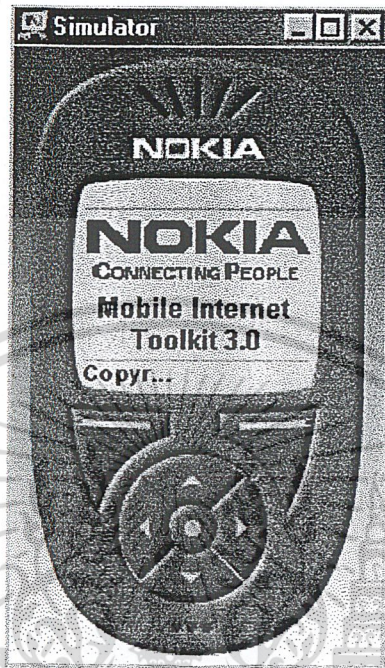
รูปที่ 2.13.3 รูป Toolkit Window ของ Nokia Toolkit

- **Toolkit Window**

คือ WAP Editor ซึ่งใช้เป็นเครื่องมือช่วยเขียนเอกสาร WML/WMLScript หรือเปิดเอกสาร WML/WMLScript (ทั้งจาก WAP Site ต่างๆ หรือเอกสาร WML ที่เราเขียนไว้แล้ว) เพื่อแสดงผลใน Device Window นอกจากนี้ Toolkit Window ยังสามารถคอมไพล์เอกสาร WML ที่เราเขียนให้เป็นรูปแบบ binary ได้อีกด้วย ส่วนหน้าที่อื่นๆ ยังมีอีกมากมาย สำหรับผู้ที่เขียน WML ด้วยโปรแกรมแท็กซ์เอดิเตอร์อื่นๆ เช่น NotePad ก็สามารถทำได้ จากนั้นค่อยนำเอกสาร WML ที่เขียนไว้ มาเปิดที่ Toolkit Window เพื่อแสดงผลใน Device Window

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนหน้าต่างนี้เรียกว่า “Device Window”



รูปที่ 2.13.4 รูป Device Window ของ Nokia Toolkit

- **Device Window**

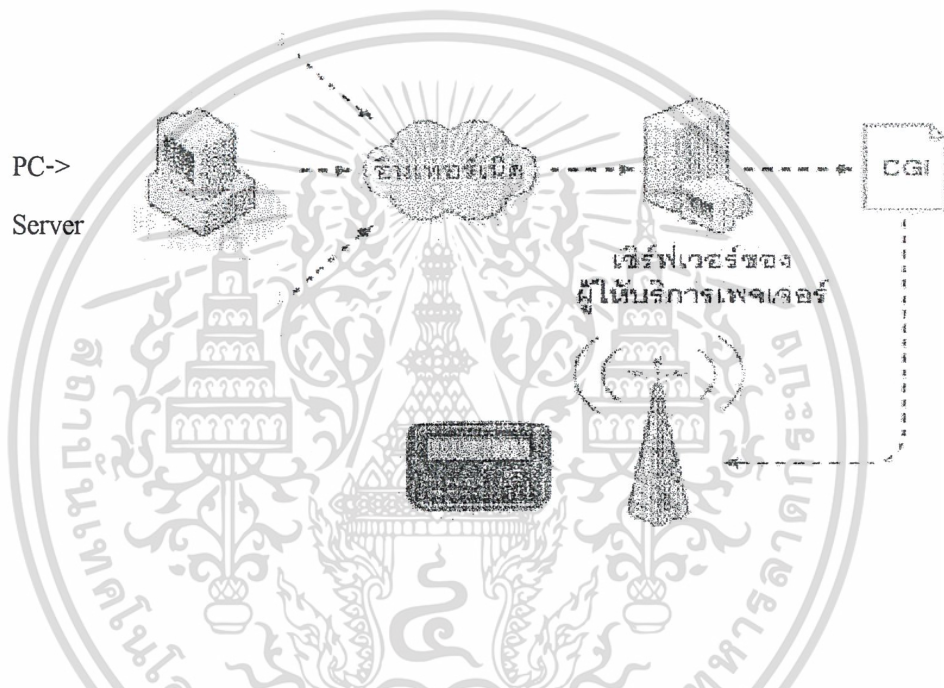
คือ WAP Emulator ซึ่งแสดงให้เห็นเป็นรูปจำลองของโทรศัพท์ เราสามารถกดปุ่มหรือเลือกเมนูด้วยซอฟต์แวร์ต่างๆได้โดยใช้เมาส์ รูปจำลองโทรศัพท์นี้เป็นคังรูปด้านบน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

### การออกแบบและการเขียนโปรแกรม

#### 3.1 สถาปัตยกรรม



รูปที่ 3.1 กระบวนการที่เซิร์ฟเวอร์ (PC) ส่งข้อความเรียกเพจเจอร์ผ่านเน็ต

จากรูป PC ที่เป็น Server ที่มีการเก็บบันทึกข้อความและตารางเวลาเอาไว้ เมื่อทำการตรวจสอบตรงตามเวลาที่บันทึกไว้ให้ส่งข้อความก็จะทำการส่งข้อมูลผ่านทางอินเทอร์เน็ตไปยังเซิร์ฟเวอร์ของผู้ให้บริการเพจเจอร์ Phonselink หรือเซิร์ฟเวอร์ของผู้ให้บริการมือถือ GSM จะมีวิธีการในการสื่อสารกับเว็บเซิร์ฟเวอร์ของผู้ให้บริการเหล่านี้ โดยจะอธิบายในลำดับต่อไป

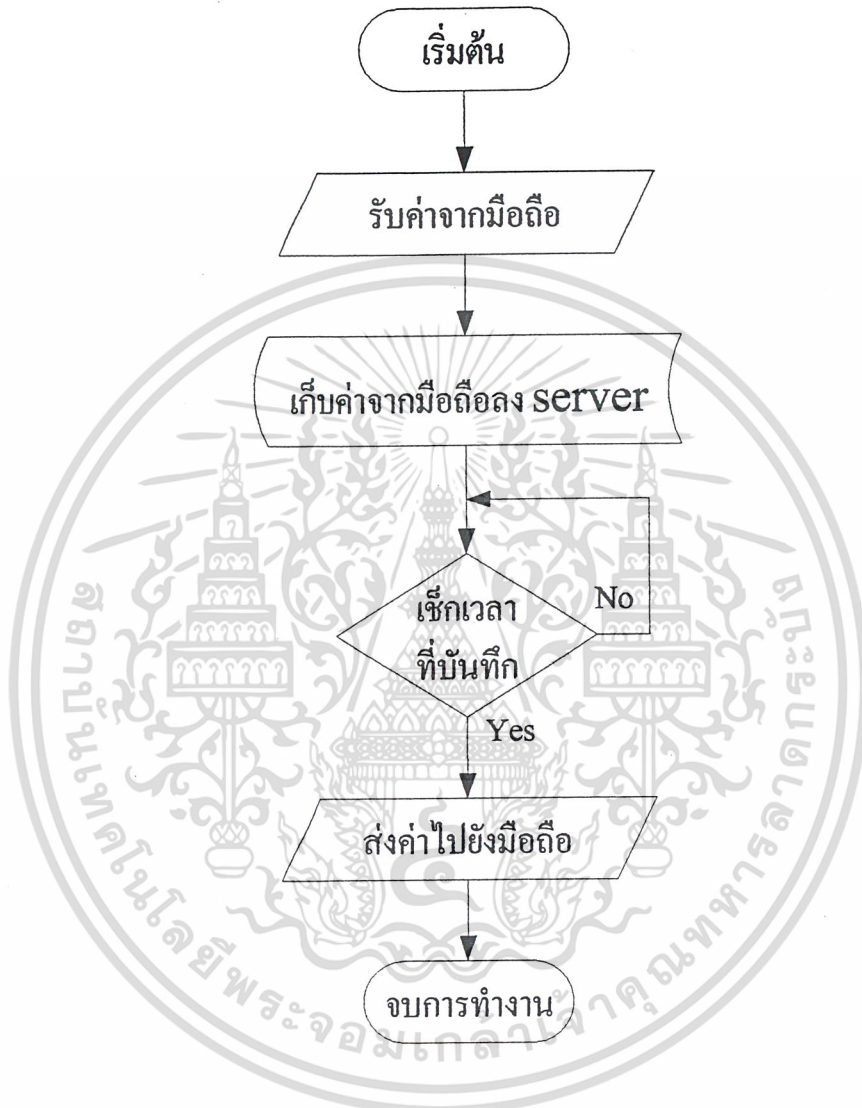


รูปที่ 3.1.1 Flowchart แสดงการทำงานของโปรแกรมบน Server PC

Server PC ทำการบันทึกค่าและเช็คตารางเวลาและส่งข้อความไปยังที่เรากำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และเรายังสามารถบันทึกข้อความและตารางเวลาลงในเซิร์ฟเวอร์ได้อีกทางจากโทรศัพท์มือถือ



รูปที่ 3.1.2 Flowchart แสดงการรับค่าจากมือถือทาง WAP เข้าไปเก็บที่เซิร์ฟเวอร์และส่งข้อความ  
คืนทาง SMS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

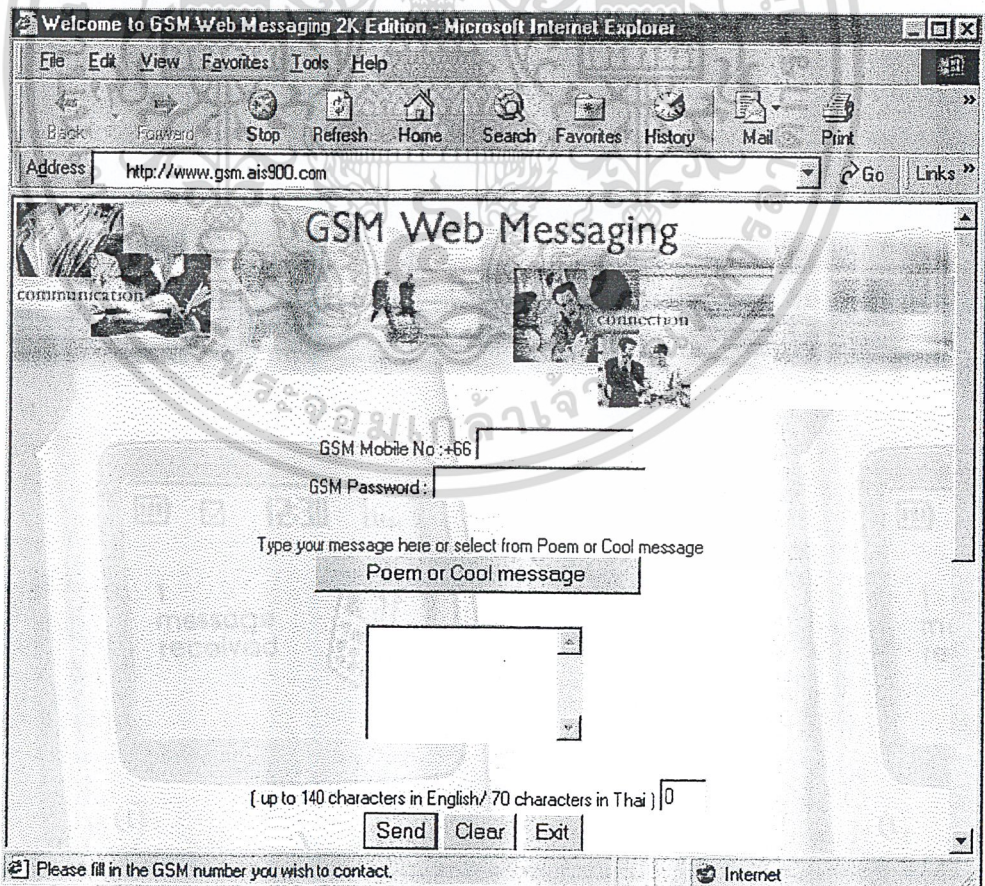
### 3.2 การส่งข้อมูลจากฟอร์มไปให้CGI ในเซิร์ฟเวอร์

แต่เชื่อว่าเราจะส่งข้อมูลไปให้ CGI ในเซิร์ฟเวอร์ได้ทุกแห่งเพราะบางแห่งอาจมีการหลอก ล่อและป้องกันไว้ วิธีการแก้ไขเพื่อแอบเรียก CGI ในเซิร์ฟเวอร์ของเว็บเพจคนอื่นให้ทำงาน

ในที่นี้เราจะเอาซอร์ซโค้ด HTML ของเว็บเพจที่มีการใช้ฟอร์มส่งข้อมูลเข้า CGI มาแกะ อินพุตของฟอร์ม เพื่อนำมาเขียนในโปรแกรมโครงงานนี้ในฟอร์มของเราเอง แต่ยังส่งข้อมูลเข้าไปยัง CGI ในเซิร์ฟเวอร์ได้เหมือนเดิมอินพุตในฟอร์มโปรแกรมของเรา จะเป็นข้อมูลพื้นฐานสำหรับการเขียนด้วยโปรแกรมวิซวลเบสิก เพื่อส่งข้อมูลไปให้ CGI รับไปทำงาน เพราะข้อมูลอินพุตในฟอร์มจะบอกให้เราทราบได้ว่า CGI ในเซิร์ฟเวอร์รองรับข้อมูลแบบไหน ต้องส่งด้วยเมธอด GET หรือ POST และส่งข้อมูลด้วยตัวแปลาอะไรบ้าง

### 3.3 หลักการในการสื่อสารกับเว็บเซิร์ฟเวอร์ของเครือข่าย GSM และ Easycall Pager

เมื่อเว็บเพจ GSM Web Messaging ซึ่งให้บริการส่งข้อความเข้าโทรศัพท์มือถือระบบ GSM อยู่ที่ <http://www.gsm.ais900.com> ดังรูป



รูปที่ 3.3 เว็บเพจ GSM Web Messaging

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเปิดเว็บเพจผู้ให้บริการแล้ว ให้คลิกขวาที่เว็บเพจ เลือกเมนู View Source เข้าไปดูซอร์ซโค้ด HTML ค้นตำแหน่งที่มีการเขียนแท็ก <FROM> คัดลอกซอร์ซโค้ด HTML เฉพาะส่วนที่เป็นฟอร์ม ตั้งแต่ <FORM> ถึง </FROM> ออกมา ตัดเอาแท็ก HTML และโค้ดอื่นๆเช่น JavaScript ที่แทรกอยู่ในแท็กอินพุตออกให้หมดจนเหลือเฉพาะแท็ก <INPUT> และข้อความอธิบายอินพุต

จากซอร์ซโค้ด HTML ที่แกะออกมา บอกข้อมูลให้เราทราบดังต่อไปนี้

- ข้อมูลจะถูกส่งไปให้ CGI ด้วยเมธอด POST
- CGI ในเซิร์ฟเวอร์ที่รองรับข้อมูลชื่อ web2gsm.cgi อยู่ในไดเรกทอรี /cgi-gsmweb ของเซิร์ฟเวอร์ www.ais900.com
- ต้องส่งข้อมูล 3 ตัวไปให้ CGI ในชื่อของ szGsmNo, szGsmPwd และ szMesg คือ เบอร์โทรศัพท์มือถือ, รหัสผ่าน และข้อความที่จะส่ง ตามลำดับ

จากซอร์ซโค้ดที่เราดู ยังไม่สามารถรู้ได้ว่าข้อมูลที่ส่งไปยัง CGI ของเว็บ GSM นั้นมีอะไรบ้าง และเซคเตอร์ของ web คืออะไร เพราะฉะนั้นจึงนำเอาโปรแกรม Ethereal มาช่วยในการจับหรือตรวจดูเซคเตอร์ของ GSM ว่ามีการเขียนอย่างไร

ส่วนทางด้านเพจเจอร์ก็ทำเช่นเดียวกันกับของโทรศัพท์มือถือ

The screenshot shows a web browser window titled 'Welcome to EasyCall Pager Web Site - Microsoft Internet Explorer'. The address bar shows 'http://easycall.lenso.co.th/'. The main content area features a 'Send Message' form with the following fields: 'From' (Art), 'EC. NO.' (264153), and 'Message' (Text SendMessage). Below the form, there is a 'Set time' section with dropdown menus for 'Day (Now)', 'Hour (Now)', and 'Min (Now)'. The page also includes a navigation menu on the left with links like 'Home', 'About Us', 'Send Page', 'Products', 'Services', 'Activities', 'Faq', and 'Contact Us'. A sidebar on the right contains 'EasyCall' branding and a 'High Speed Internet From Q-NET' advertisement. The status bar at the bottom shows 'Internet' and the time '13:53'.

รูปที่ 3.3.1 เว็บเพจ Easycall Pager

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างเว็บส่งข้อความทางเพจเจอร์

เว็บไซต์ [easycall./lenso.co.th](http://easycall./lenso.co.th)

ไฟล์นี้เป็นฟอร์มสำหรับรับข้อมูล เพื่อส่งไปยังสคริปต์ CGI (Common Gateway Interface) และก็ทำการ capture ดูเซคเตอร์แล้วนำไปเขียนลงใน from ใน โปรเจค

### 3.4 กระบวนการส่งข้อความเข้าเพจเจอร์ผ่านทางเว็บไซต์

ไม่ว่าจะส่งข้อความเข้าเพจเจอร์หรือโทรศัพท์มือถือ ข้อความนั้นต้องผ่าน CGI ใน เซิร์ฟเวอร์ผู้ให้บริการอยู่ติดตั้งกระบวนการทำงานในรูป



รูปที่ 3.4 กระบวนการส่งข้อความเรียกเพจเจอร์ผ่านเว็บ

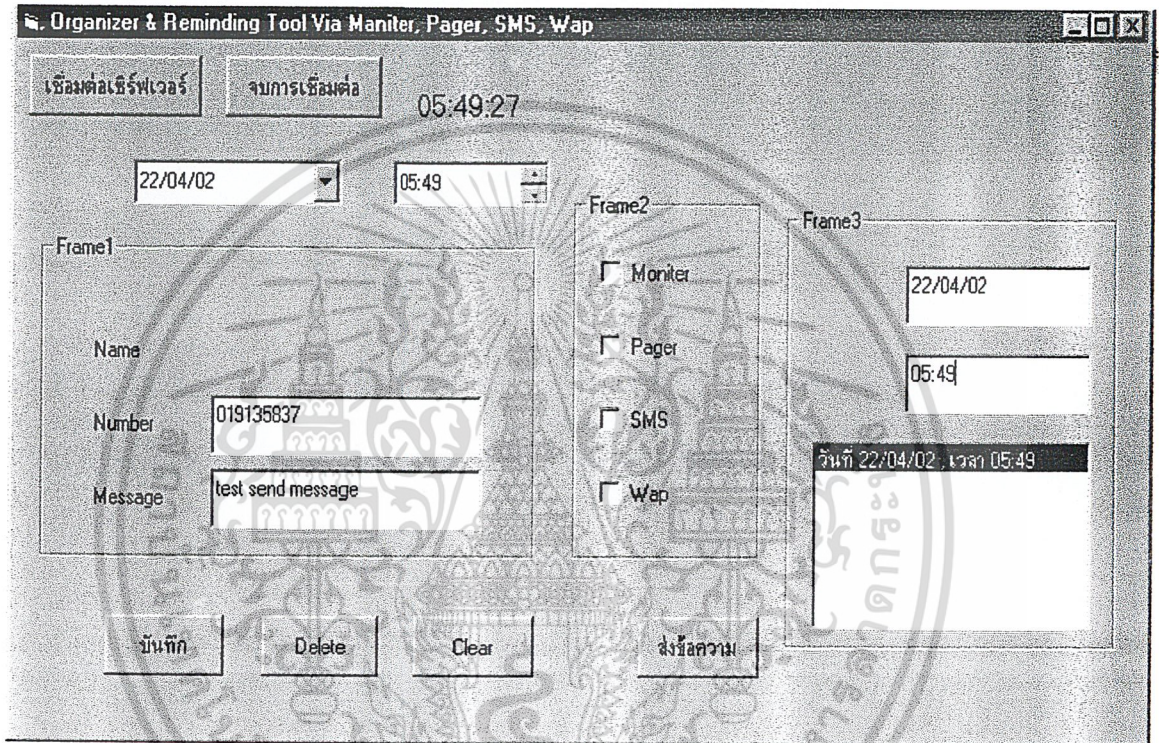
เป้าหมายของทุกโปรแกรม ต้องติดต่อกับเซิร์ฟเวอร์ผู้ให้บริการ เพื่อส่งข้อมูลไปให้ CGI ใน เซิร์ฟเวอร์รับไปทำงานให้ได้ สิ่งที่สำคัญคือ ในการส่งข้อมูลไปยัง CGI ในเซิร์ฟเวอร์ผู้ให้บริการ แต่ละรายนั้น ชื่อตัวแปรและค่าต่างๆที่ส่งไปให้จะแตกต่างกัน ฉะนั้นหากต้องการส่งข้อความเข้าเพจเจอร์หรือมือถือของผู้ให้บริการรายใด ก็ต้องไปแกะอินพุตจากฟอร์มในเว็บเพจของผู้ให้บริการรายนั้น เพื่อดูว่าจะต้องส่งข้อมูลอะไรไปให้บ้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4 การทดลองใช้งานโปรแกรม

### 4.1 หน้าจออินเตอร์เฟซของ PC Monitor

ลักษณะและการใช้งานของโปรแกรม



รูปที่ 4.1 หน้าจออินเตอร์เฟซของ PC Monitor

จากรูป อธิบายการใช้งาน

- จะมีคอม โบบล็อกสำหรับเลือกวันที่ และเวลาที่ต้องการบันทึก
- Frame 1 สำหรับเขียนเบอร์โทรศัพท์ และ message ข้อความที่ต้องการเตือนความจำ
- Frame 2 สำหรับเลือกว่าต้องการส่งข้อความเตือนทางด้านใดบ้าง
- Frame 3 เป็นตารางแสดงวันที่, เวลาและข้อความทั้งหมดที่บันทึกไว้

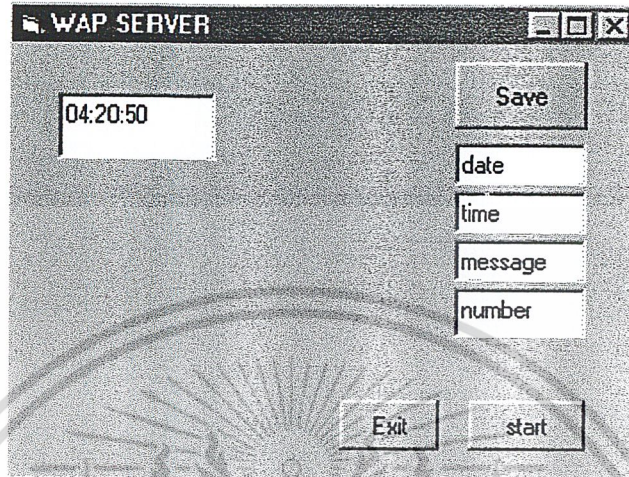
ปุ่มบันทึก เมื่อเราพิมพ์ข้อมูลทั้งหมดเสร็จแล้วก็ทำการกดปุ่มบันทึกเก็บไว้

ปุ่ม Delete ใช้สำหรับลบข้อความออกไปจากบันทึก

ปุ่ม Clear ใช้สำหรับเคลียข้อความที่พิมพ์ผิดในบล็อกทั้งหมดแต่ไม่ลบข้อมูลออกจากบันทึก

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่ขึ้นที่นอกรั้ว  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

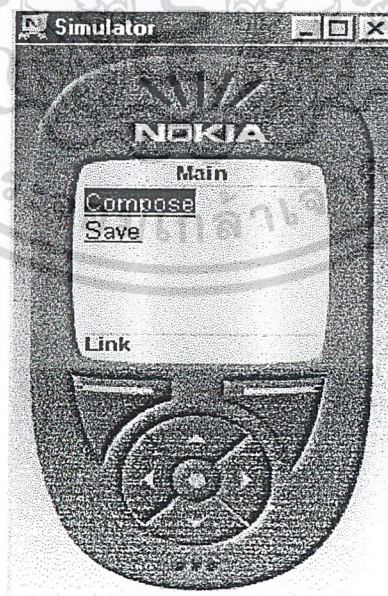
## 4.2 หน้าจอ WAP Server บนเครื่อง PC



รูปที่ 4.2 หน้าจอ WAP Sever บนเครื่อง PC

การทำงานของหน้าจอนี้เราสามารถบันทึกข้อมูลต่างๆลงที่หน้าจอนี้ได้ทั้งหมด แล้วทำการ Save เก็บลงในไฟล์ข้อมูลบนเครื่อง PC

## 4.3 หน้าจออินเทอร์เน็ตเฟสของ Mobile Phone



รูปที่ 4.3 หน้าจออินเทอร์เน็ตเฟสของ Mobile Phone

**Compost** คือการบันทึกข้อความทางหน้าจออินเทอร์เน็ตเฟสของ Mobile Phone แล้วทำการส่งข้อความ

ความมาที่มือถือทันทีโดยไม่มีกรบันทึก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3.1 แสดงการ Save ค่าข้อมูล ผ่านทางหน้าจออินเทอร์เน็ตเฟสของ Mobile Phone

จากรูปข้อมูลที่บันทึกมีดังนี้

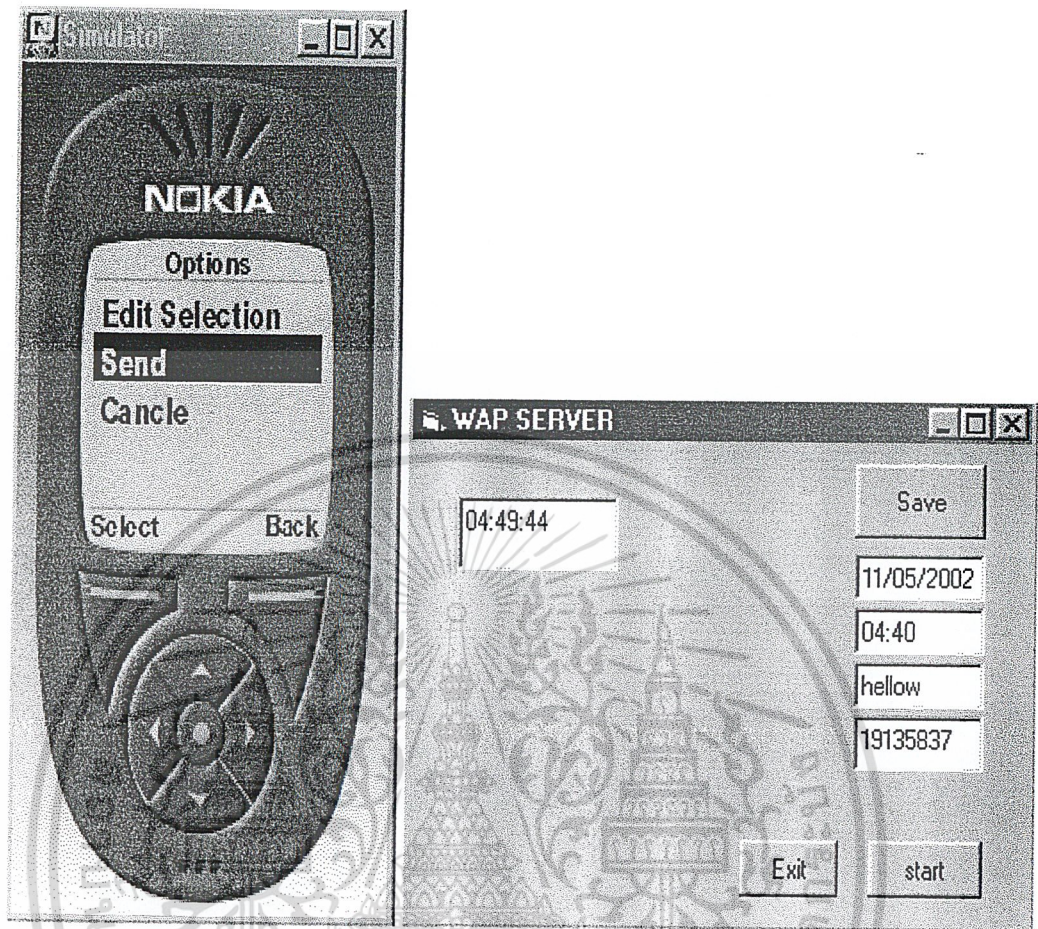
Message : hello

Number : 19135837

Time : 04:40

Date : 11/05/2002

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3.2 แสดงการทำงานร่วมกันของ Interface Mobile Phone กับหน้าจอของ WAP Server บน PC จากรูป ที่หน้าจอ Mobile Phone เมื่อเรากดปุ่ม Select ที่ปุ่มซอร์ฟคีย์ เพื่อทำการ Send ข้อมูลที่เราบันทึกไว้ ข้อมูลที่เราบันทึกไว้ทั้งหมดก็จะถูกส่งมาที่ Server และที่ Server ก็จะทำการเช็ค ข้อมูลตารางเวลาที่จะต้องส่งข้อความ SMS มาที่โทรศัพท์มือถือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### บทสรุปและวิจารณ์

#### 5.1 บทสรุป

โครงการเขียนโปรแกรมเดือนความจำนี้ สามารถทำงานได้ตามเป้าหมาย และขอบเขตที่กำหนดไว้ได้ โดยได้ทำการเขียนโปรแกรมด้วย Visual Basic โดยใช้ Winsock Control ในการส่งข้อความผ่านเว็บ และใช้โปรแกรม Ethereal เป็นตัวจับแพ็กเก็ตเพื่อดูเฮดเดอร์ของเว็บไซต์ที่เราจะใช้ CGI ของเว็บนั้นส่งข้อความเข้าเครื่องเพจเจอร์/มือถือให้กับเรา เป็นการศึกษาโปรโตคอลต่างๆที่ใช้ในการส่งข้อความทางเว็บไซต์

#### 5.2 ปัญหาในการทำงาน

- เนื่องจากผู้จัดทำไม่ค่อยมีประสบการณ์ในการเขียน โปรแกรมเกี่ยวกับเว็บจึงใช้เวลาในการศึกษาหาข้อมูลนาน
- ปัญหาเกี่ยวกับการตรวจจับเฮดเดอร์ของเว็บไซต์บางเว็บที่มีความซับซ้อนมากมีการป้องกันอย่างดี จึงมีบางเว็บไม่สามารถค้นหารูปแบบในการเรียกใช้ CGI ของเว็บนั้นได้
- เนื่องจากเทคโนโลยี WAP ยังไม่นิยมใช้กันแพร่หลายนัก และการให้บริการของศูนย์ก็คิดอัตราการใช้งาน WAP ในราคาสูง

#### 5.3 ข้อเสนอแนะและแนวทางการพัฒนาโปรแกรม

- สำหรับผู้ที่สนใจโครงการนี้สามารถทำการพัฒนารูปแบบหน้าจอแสดงภาพให้ดูดีขึ้นตามต้องการ
- สำหรับผู้ที่สนใจโครงการนี้สามารถทำการพัฒนารูปแบบการส่งข้อมูลที่เดือนความจำเพิ่มขึ้นเช่น เดือนความจำในรูปแบบเสียง รูปภาพ เป็นต้น
- การเขียนโปรแกรมของโครงการนี้สามารถเขียนได้ในหลายๆภาษา

## เอกสารอ้างอิง

- สุรัตน์ บัณฑิตลักษณะ “หนังสือเพิ่มพลังอินเทอร์เน็ตที่ฟลูม สาม ASP” บริษัท วิตตี้กรุ๊ป จำกัด  
ทรงเกียรติ ภาวดี “หนังสือแกะรอย CGI #2” บริษัท วิตตี้กรุ๊ป จำกัด  
ชนพล ฉันทวีชัย “หนังสือเริ่มเขียนโปรแกรม Visual Basic” บริษัท วิตตี้กรุ๊ป จำกัด  
ตราวุธ อ้อยศรีสกุล “หนังสือเปิดมิติ Mobile Internet ด้วย WAP” บริษัท วิตตี้กรุ๊ป จำกัด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้