

เว็บเซอร์วิสสำหรับขายสินค้าออนไลน์

Web Service for e-commercial

นายจักรพรรดิ โปษาคู รหัส 46380010
นายวีรยุทธ หลวงสิทธิ รหัส 46380040

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ..... 7 / โส.ย. 2553 /

เลขทะเบียน..... 14942826

เลขเรียกหนังสือ..... ปร.

มหาวิทยาลัยนเรศวร ๑๒๔ ๙

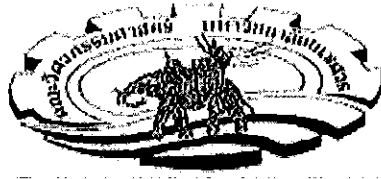
2550

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2550



ใบรับรองโครงการวิศวกรรม

หัวข้อโครงการ	เว็บไซต์สำหรับขายสินค้าออนไลน์
ผู้ดำเนินโครงการ	นายจักรพรรดิ โปธาค รหัส 46380010 นายวีรยุทธ หลวงสิทธิ์ รหัส 46380040
อาจารย์ที่ปรึกษา	อาจารย์จิราพร พุกสุข
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2550

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนครสวรรค์ อนุมัติให้โครงการฉบับนี้เป็นส่วนหนึ่งของการ
ศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์
คณะกรรมการสอบโครงการวิศวกรรม

.....ประธานกรรมการ
(อาจารย์จิราพร พุกสุข)

.....กรรมการ
(ดร.พนมขวัญ รัชเมงกล)

.....กรรมการ
(อาจารย์แสงชัย มังกรทอง)

หัวข้อโครงการ	เว็บเซอร์วิสสำหรับร้านค้าออนไลน์
ผู้ดำเนินโครงการ	นายจักรพรรดิ โปษาคู รหัส 46380010
	นายวีรยุทธ หลวงสิทธิ์ รหัส 46380040
อาจารย์ที่ปรึกษา	อาจารย์จิราพร พุกสุข
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2550

บทคัดย่อ

ปัจจุบันการใช้อินเทอร์เน็ตมีแนวโน้มที่สูงขึ้นและแพร่หลาย เทคโนโลยีเว็บเซอร์วิส (Web services) ถือเป็นอีกหนึ่งเทคโนโลยีบนโลกอินเทอร์เน็ตที่ได้รับความนิยมเป็นอย่างมาก เพราะไม่ว่าจะเป็นบริษัทเอกชนหรือหน่วยงานของรัฐบาลต่างให้การสนับสนุนเทคโนโลยีนี้เป็นอย่างมาก จึงทำให้เว็บเซอร์วิสนั้นได้ก้าวเข้ามามีบทบาทอย่างมากในสมัยนี้

เว็บเซอร์วิส เป็นพื้นฐานสำคัญของการให้บริการใหม่ๆ เช่นการพาณิชย์อิเล็กทรอนิกส์ระหว่างธุรกิจกับธุรกิจ

โดยปริญญาณิพนธ์นี้ได้อธิบายถึงรายละเอียดของเว็บเซอร์วิส พร้อมทั้งได้จัดทำระบบขึ้นมาระบบหนึ่ง ซึ่งเป็นเว็บเซอร์วิสที่ให้บริการ (Services) เกี่ยวกับการซื้อขายสินค้าคอมพิวเตอร์ประเภทโน้ตบุ๊ก (Note Book) ซึ่งจะช่วยให้ลูกค้าที่เข้าไปใช้บริการได้รับความสะดวกสบายมากยิ่งขึ้น

Project Title	Web Service for e-commercial		
Name	Mr. Jakkapat Pothard	ID. 46380010	
	Mr. Weerayut Luangsit	ID. 46380040	
Project Advisor	Miss. Jirapon poksook		
Major	Computer Engineering		
Department	Electrical and Computer Engineering		
Academic	2550		

ABSTRACT

At present, there is an extensively-increasing tendency of using internet and the Web services technology. The Web services, which are receiving a lot of interest, are being regarded as one of technology on the internet world because both of the private and the government sector give abundantly support to this technology then the Web services have many roles now.

The Web services are important for new services such as e-commerce between businesses.

This thesis explains details of the Web services including to producing a system that the Web services give services about computer Note Book dealing. These services are helpful for customers to consume goods comfortably.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้ สำเร็จลุล่วงได้เป็นอย่างดี ด้วยคำแนะนำ คำปรึกษาและความร่วมมือจากหลายๆ ฝ่ายด้วยกัน โดยเฉพาะอาจารย์ที่ปรึกษาที่ให้ โอกาสกับข้าพเจ้าได้ทำปริญญานิพนธ์ฉบับนี้ คอยให้ความเอาใจใส่แนะนำ และความช่วยเหลือเสมอมาคือ อาจารย์จิราพร พุกสุข ซึ่งต้องขอขอบพระคุณเป็นอย่างยิ่ง

ขอขอบคุณกรรมการโครงการคือ ดร.พนมขวัญ ธิษะมงคล , อาจารย์แสงชัย มังกรทอง ที่ได้ให้คำปรึกษาและข้อเสนอแนะต่างๆ และภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ที่ได้จัดเตรียมสิ่งอำนวยความสะดวก เพื่อให้การวิจัยและการพัฒนา โปรแกรมเป็นไปได้ด้วยความสะดวกและรวดเร็ว ซึ่งท้ายที่สุดแล้วก็ประกอบกันเป็นส่วนหนึ่งของโครงการนี้

สุดท้ายนี้ต้องขอขอบพระคุณบุคคลสำคัญที่สุดในชีวิตที่ทำให้ข้าพเจ้ามีวันนี้ ก็คือ บิดา มารดา และบุคคลในครอบครัวอันเป็นที่เคารพรัก ซึ่งได้เลี้ยงดู คอยส่งสอนข้าพเจ้ามาเป็นอย่างดี พร้อมให้โอกาสในการศึกษาอย่างเต็มที่และยังให้กำลังใจ ความรักเสมอมา ข้าพเจ้าขอกราบขอบพระคุณมา ณ ที่นี้ด้วย

นายจักรพรรดิ โปธาตุ 46380010

นายวีรยุทธ หลวงสิทธิ์ 46380040



สารบัญ

หน้า

บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญรูป	ฉ
บทที่ 1 บทนำ	
1.1 หลักการและเหตุผล	1
1.2 วัตถุประสงค์	1
1.3 ขอบข่ายของโครงการ	2
1.4 ขั้นตอนการดำเนินโครงการ	2
1.5 ผลที่คาดว่าจะได้รับ	3
1.6 งบประมาณที่ใช้	3
บทที่ 2 หลักการและทฤษฎี	
2.1 เว็บเซอร์วิสเทคโนโลยี	4
2.2 การพัฒนาของเอ็กซ์เอ็มแอล (XML) โปรโตคอล	6
2.3 โครงสร้างของโซฟ (SOAP)	7
2.4 ตัวกลางในการส่งเอกสาร โซฟ (SOAP Intermediaries)	8
2.5 Intermediaries ใน SOAP	10
2.6 ภาษาที่ใช้ในการพัฒนาเว็บเซอร์วิส	12
2.7 การเปรียบเทียบระหว่างข้อดีและข้อเสียของ .NET และ J2EE	12
2.8 ดอทเน็ตเฟรมเวิร์ค (.NET Framework)	14

สารบัญ(ต่อ)

หน้า

บทที่ 3 หลักการทำงานและโครงสร้างของโปรแกรม

3.1 ส่วนเว็บกลาง.....	20
3.2 เว็บร้านค้าย่อย (Shop)	21
3.3 ขั้นตอนการออกแบบ Data Flow Diagram.....	23

บทที่ 4 การทดลองและผลการทดลอง

4.1 ส่วนของลูกค้า.....	29
4.2 ส่วนต่างๆที่เกี่ยวข้องกับการซื้อขายสินค้า.....	38

บทที่ 5 สรุปผลการทดลอง

5.1 สรุปผลการทดลอง	39
5.2 ข้อเสนอแนะและการแก้ไข้ปัญหา.....	39
5.3 แนวทางในการพัฒนาต่อไปในอนาคต	39

เอกสารอ้างอิง.....	40
--------------------	----

ภาคผนวก ก.	41
-----------------	----

ประวัติผู้เขียนโครงงาน	61
------------------------------	----

สารบัญรูป

รูปที่	หน้า
2.1 ส่วนประกอบหลักของเทคนิคของเว็บเซอร์วิส (เว็บเซอร์วิส = SOAP+WSDL+UDDI)	5
2.2 รูปโครงสร้างการให้บริการ	6
2.3 รูปโครงสร้างการให้บริการ	7
2.4 การทำงานที่มีการส่งเอกสาร โฆษแบบจุดต่อจุดและแบบส่งตามเส้น	9
2.5 โครงสร้างของเว็บเซอร์วิส	11
2.6 การเปรียบเทียบคอตเน็ตเฟรมเวิร์คและจาวาแพลตฟอร์ม	14
2.7 โครงสร้างของคอตเน็ตเฟรมเวิร์ค	15
2.8 ซอร์สโค้ดตัวอย่าง	18
2.9 ผลลัพธ์เมื่อรันโปรแกรมตัวอย่าง	19
2.10 ผลลัพธ์ของเมสเสจที่เป็นภาษา XML	19
3.1 แสดงแผนภาพบริษัท (Context Diagram)	24
3.2 แสดง Data Flow Diagram Level 0	25
3.3 แสดง Data Flow Diagram Level 1 ของ Process 1	26
3.4 แสดง Data Flow Diagram Level 1 ของ Process 2	27
3.5 แสดง Data Flow Diagram Level 1 ของ Process 3	28
4.1 การสมัครเป็นสมาชิกเว็บกลาง	30
4.2 กรณีที่ยังไม่ได้สมัครเป็นสมาชิกเว็บกลาง	30
4.3 เป็นการแสดงถึงการสมัครเป็นสมาชิกเว็บกลางได้สมบูรณ์	31
4.4 เป็นการแสดงถึงสมาชิกไม่ได้ทำการล็อกอินเพื่อเข้าสู่ระบบ	31
4.5 การเลือกซื้อสินค้าตามรายการสินค้า	32
4.6 เมื่อเลือก Link Product	33
4.7 เมื่อเลือกยี่ห้อ Acer	33
4.8 เมื่อต้องการเลือกยี่ห้อ Compaq	34
4.9 รายการของการซื้อสินค้าร้านค้าที่ 1	34
4.10 รายการเลือกซื้อสินค้าทั้งสองร้านค้า	35
4.11 รายการเพิ่มจำนวนสินค้าที่เลือกซื้อ	36
4.12 เกิดความผิดพลาดในการกรอกแบบฟอร์มการสั่งซื้อสินค้า	36

4.13 เมื่อกรอกรายละเอียดได้สมบูรณ์ข้อมูลก็จะถูกจัดเก็บ 37

4.14 รายการเลือกซื้อสินค้าทั้งสองร้านค้าแต่ไม่ได้กรอกแบบฟอร์มให้ครบทั้งสอง 37

4.15 ฐานข้อมูลของการยืนยันการสั่งซื้อสินค้า..... 38



บทที่ 1

บทนำ

1.1 หลักการและเหตุผล

ปัจจุบันการพัฒนาเว็บได้ก้าวหน้าไปอย่างรวดเร็วเพื่อมารองรับการใช้งานที่มีความต้องการสูงขึ้น โดยเฉพาะการดำเนินการทางธุรกิจการค้าต่างๆ ที่เพิ่มมากขึ้นบนเครือข่าย แต่เนื่องด้วยการอัปเดตข้อมูลหรือเนื้อหาบนโฮมเพจโดยมนุษย์ อาจเกิดความผิดพลาด (Human Error) และมีค่าใช้จ่ายที่สูง ข้อมูลก็อาจไม่ทันต่อความต้องการ เพราะต้องรอผู้ที่ดูแลเว็บไซต์มาดำเนินการ ฉะนั้นจึงเกิดแนวคิดของกรที่สามารถดึงข้อมูลจากเซิร์ฟเวอร์เครื่องอื่นได้เองอัตโนมัติ เว็บเซอร์วิสจึงถือกำเนิดขึ้นจากแนวความคิดนี้

ในส่วนของโครงการนี้ ได้ทำการสร้างตัวกลางเว็บเซอร์วิสขึ้นมา โดยที่จะสามารถให้บริการระหว่างร้านค้า (Shop) ซึ่งส่วนของร้านค้า (Shop) ได้จัดทำโดยการจำลองขึ้นมา และลูกค้าของเว็บกลาง (End-User) ทางเว็บเซอร์วิสจะคอยดูแลระบบไม่ให้เกิดความผิดพลาดระหว่างการติดต่อกันของร้านค้า (Shop) และลูกค้า (End-User) เพื่ออำนวยความสะดวกให้กับผู้สนใจซื้อขายสินค้าและในการอัปเดต (Update) ข่าวสารต่างๆ ให้เป็นไปอย่างสะดวกและง่ายดาย ไม่จำเป็นต้องใช้คนมาคอยดูแลมากมาย ทางเว็บเซอร์วิสจะเป็นตัวช่วยดูแลในส่วนนี้

1.2 วัตถุประสงค์ของโครงการ

1. เพื่อศึกษาหลักการของเว็บเซอร์วิส และนำหลักการการทำงานของเว็บเซอร์วิสมาใช้ประยุกต์ให้เป็นประโยชน์ในชีวิตประจำวัน โดยได้ทำการจำลองร้านค้าการซื้อขายสินค้าประเภทโน้ตบุ๊ก (Notebook)
2. เพื่อพัฒนาและจะให้เห็นภาพแนวทางในการนำเว็บเซอร์วิสมาปรับใช้ให้เกิดประโยชน์อย่างเป็นรูปธรรม
3. เพื่อที่จะศึกษาทำความเข้าใจใน .Net Technology
4. เพื่อศึกษาเทคโนโลยีที่เกี่ยวข้องได้แก่ SOAP, XML, WSDL, UDDI เป็นต้น
5. เพื่อศึกษาการสร้างเว็บเซอร์วิส โดยการใช้ .Net Technology
6. เพื่อศึกษาการจัดเก็บข้อมูลในรูปแบบของภาษา SQL

1.3 ขอบข่ายของโครงการงาน

ปริญญาโทฉบับนี้ได้นำเสนอแนวทางการซื้อขายสินค้าออนไลน์(Online) โดยมีร้านค้ากลาง (Shop) หรือเว็บกลางเป็นตัวให้บริการกับผู้ซื้อ โดยในที่นี้เว็บกลางจะไปเรียกบริการรายละเอียดของสินค้ามาจากร้านค้าที่ได้จำลองขึ้นมา เมื่อลูกค้าตัดสินใจที่จะซื้อสินค้า เว็บกลางจะส่งรายละเอียดการซื้อสินค้าไปยังร้านค้าที่ลูกค้าได้ทำการซื้อสินค้าไว้ โดยที่เมื่อข้อมูลไปถึงยังร้านค้าที่ได้ทำการซื้อขายนั้น ร้านค้าจะส่งรายละเอียดยืนยันว่าได้รับข้อมูลการสั่งซื้อของลูกค้าแล้ว โดยผ่านทางเว็บกลาง

ส่วนของฐานข้อมูล (Database) นั้นจะเป็นส่วนที่เก็บรวบรวมข้อมูลของสินค้าที่มีการซื้อขายสินค้า ในที่นี้จะเก็บในรูปแบบของ SQL โดยที่จะแสดงรายละเอียดของสินค้าอย่างเช่น ชื่อสินค้า, ราคาสินค้า, รายละเอียดภายในสินค้า, รูปของสินค้า เป็นต้น

1.4 ขั้นตอนการดำเนินโครงการงาน

1. เลือกหัวข้อโครงการและปรึกษาอาจารย์เรื่องหัวข้อโครงการ
2. ศึกษาหลักการของเว็บเซอร์วิส (Web service)
3. ศึกษาเกี่ยวกับแพลตฟอร์มและภาษาที่ใช้ในการพัฒนาเว็บเซอร์วิส
4. ออกแบบโครงสร้างระบบซื้อขายสินค้าออนไลน์
5. กำหนดหลักการในการซื้อขายสินค้าผ่านเว็บเซอร์วิส
6. สร้างแอปพลิเคชันของระบบร้านค้ากลางและระบบต่างๆที่มีส่วนเกี่ยวข้องในการซื้อขายสินค้า
7. สร้างแอปพลิเคชันของระบบที่รวบรวมการให้บริการด้านต่างๆจากระบบการซื้อขายสินค้า
8. ทดสอบการทำงานของระบบการซื้อขายสินค้าและแก้ไขปัญหาที่เกิดขึ้นกับระบบ
9. สรุปและรวบรวมข้อมูลแล้วจัดทำรายงาน

1.5 ผลที่คาดว่าจะได้รับ

1. เข้าใจถึงการทำธุรกิจการค้าและบริการเป็นไปโดยอัตโนมัติในระดับแอปพลิเคชันกับแอปพลิเคชัน (A2A) โดยการแลกเปลี่ยนข้อมูลผ่านทางเว็บเซอร์วิส
2. เข้าใจการทำงานของเว็บเซอร์วิส (Web Service) อย่างง่ายได้
3. เข้าใจการทำงานเกี่ยวกับฐานข้อมูล (Data Base)
4. ความรู้ทางด้าน การเขียนโปรแกรมคอมพิวเตอร์
5. ความรู้ทางด้านภาษาต่างๆ ในการเขียนโปรแกรม
6. รู้จักพัฒนาปรับปรุงและนำโปรแกรมต่างๆ มาประยุกต์ใช้
7. ลดค่าใช้จ่ายในการจ้างการควบคุมการทำงานของระบบ
8. ข้อมูลมีความเป็นปัจจุบันมากกว่า
9. ทำให้เกิดการรวมตัวทางธุรกิจมากขึ้น
10. ทำให้ผู้บริโภคมีความสะดวกสบายในการซื้อสินค้ามากขึ้น

1.6 งบประมาณที่ใช้

- | | | | |
|------------------------|----------|-----|-----|
| 1. ค่าวัสดุสำนักงาน | เป็นเงิน | 500 | บาท |
| 2. ค่าวัสดุคอมพิวเตอร์ | เป็นเงิน | 300 | บาท |
| 3. ค่าถ่ายเอกสาร | เป็นเงิน | 500 | บาท |
| 4. ค่าวัสดุอื่นๆ | เป็นเงิน | 700 | บาท |

รวมเป็นเงินทั้งสิ้น 2,000 บาท (สองพันบาทถ้วน)

บทที่ 2

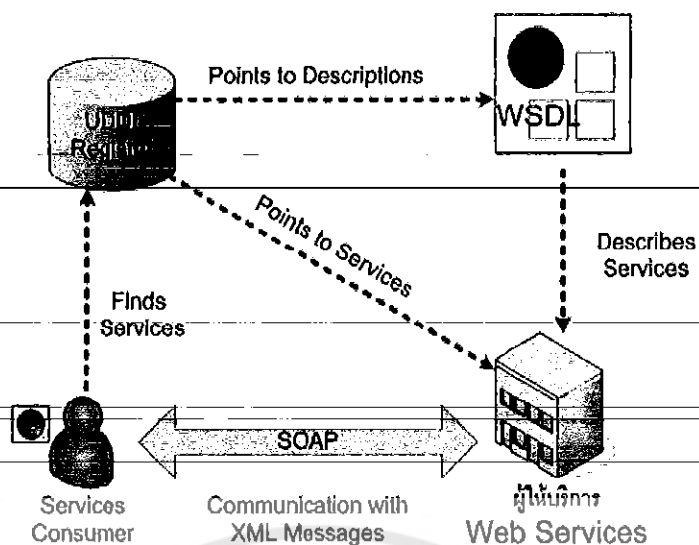
หลักการและทฤษฎี

ในบทนี้จะกล่าวถึงหลักการและทฤษฎีพื้นฐานต่างๆ ที่เกี่ยวข้องในการวิจัย และพื้นฐานของระบบเว็บเซอร์วิส ซึ่งเนื้อหาในบทนี้จะใช้ภาษากลางในการติดต่อสื่อสารระหว่างแอปพลิเคชันที่มีความแตกต่างกันหรือที่เราอยู่ในรูปของภาษา “XML” และเทคโนโลยีที่เกี่ยวข้อง ซึ่งเนื้อหาทั้งหมดจำเป็นอย่างยิ่งสำหรับการศึกษา และใช้พัฒนาระบบเว็บเซอร์วิส

2.1 เว็บเซอร์วิสเทคโนโลยี

ปัจจุบันอินเทอร์เน็ตได้รับการพัฒนาไปอย่างรวดเร็วมาก ซึ่งให้การเปิดบริการรูปแบบต่างๆ บนอินเทอร์เน็ต ที่เรียกว่า “เว็บเซอร์วิส” และมีการเชื่อมโยงในเว็บเซอร์วิสต่างๆ เข้าด้วยกันเพื่อที่จะให้บริการผู้ใช้บริการได้อย่างรวดเร็ว สะดวกสบายและมีประสิทธิภาพมากขึ้น แต่เนื่องจากมีปัญหาสำคัญอย่างหนึ่งคือ ภาษา HTML นั้นถูกพัฒนาขึ้นมาเพื่อใช้แสดงผลข้อมูลของเว็บไซต์นั้น ให้ผู้ใช้คอมพิวเตอร์สามารถดูผ่านเบราว์เซอร์ (Browser) เป็นสำคัญ ซึ่งจะไม่สะดวกในการที่จะนำมาใช้แลกเปลี่ยนหรือประมวลผลข้อมูลระหว่างโปรแกรมต่างๆ บนอุปกรณ์คอมพิวเตอร์ด้วยกันได้ ฉะนั้นจึงได้เกิดการร่วมมือขององค์กรต่างๆ ที่เกี่ยวข้องกันนั้น ได้ร่วมกันจัดทำภาษาที่เป็นภาษามาตรฐานของภาษาสากลขึ้นมาใหม่ซึ่งเรียกว่า XML (Extensible Markup Language) เพื่อให้สามารถที่จะใช้แลกเปลี่ยนข้อมูลระหว่างผู้ใช้และโปรแกรม หรือระหว่างโปรแกรมกับโปรแกรมด้วยกัน (Application to Application Interoperability , A2A) โดยภาษา XML นั้นเป็นมาตรฐานเปิด (เป็นมาตรฐานที่ได้รับการเผยแพร่โดยปราศจากข้อจำกัด สามารถนำมาปรับใช้ได้ฟรี ด้วยการกำหนดนิยามไว้อย่างดี ครอบคลุมขบวนการของวิวัฒนาการของมาตรฐาน และการนำมาตรฐานมาปฏิบัติต้องสามารถใช้อ็องค์ประกอบที่มีในท้องตลาดทั่วไปได้) โดยที่สามารถจะใช้งานได้ฟรี และไม่ขึ้นกับแพลตฟอร์มใดๆ ทั้งสิ้น โดยสามารถที่ใช้กับระบบปฏิบัติการ (Operating System) ต่างๆ ไม่ว่าจะเป็นวินโดวส์ (Windows) หรือลินุกซ์ (Linux)

ในการเปิดให้บริการทางเว็บเซอร์วิสนั้น นอกจากจะต้องใช้ภาษาสากลที่เป็นมาตรฐานคือ XML แล้วนั้น ยังต้องอาศัยเทคโนโลยีอีก 3 ประการคือ



รูปที่ 2.1 ส่วนประกอบหลักของเทคนิคของเว็บเซอร์วิส

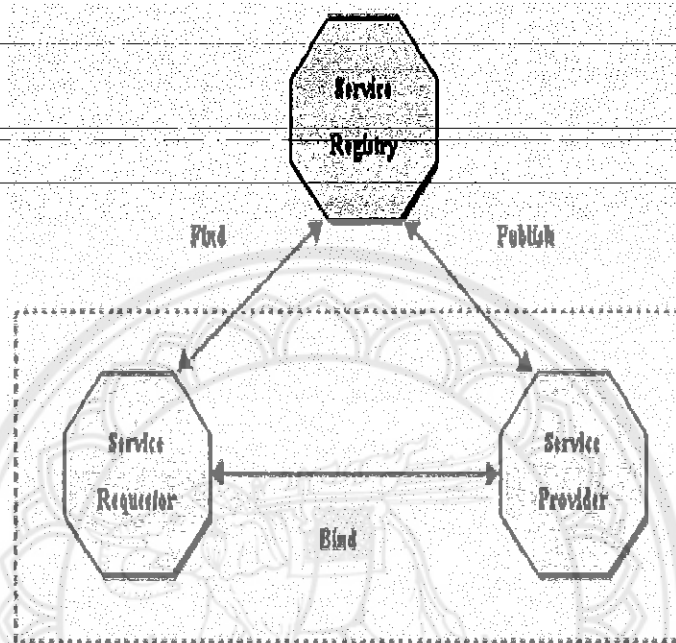
เว็บเซอร์วิส = SOAP+WSDL+UDDI

(ที่มา : <http://www.cmsthailand.com/ws/index.php>, 2004)

1. Simple Object Access Protocol (SOAP) เป็น โปรโตคอล (Protocol) ที่ใช้ในส่วนของการรับส่งข้อมูลซึ่งใช้งานบนภาษา XML ซึ่งมีขนาดเล็กและง่าย
2. Web Service Description Language (WSDL) เป็นข้อมูลที่อยู่บนภาษา XML ที่ ผู้ให้บริการ Web Service นั้นได้จัดทำขึ้นเพื่อเป็นข้อมูลสำหรับที่จะเรียกใช้งานจากเว็บเซอร์วิสต่างๆ ได้
3. Universal Description, Discovery and Integration (UDDI) เป็นเหมือนเสิร์ชเอนจิน (Search Engine) ตัวหนึ่งของผู้ให้บริการเว็บเซอร์วิสที่สามารถที่จะไปขึ้นทะเบียนการให้บริการของผู้ให้บริการและผู้เรียกใช้สามารถไปสืบค้นและ/หรือเชื่อมโยงเข้ากับเว็บเซอร์วิสของผู้ให้บริการได้

เนื่องจากจุดประสงค์หลักของการใช้งานเว็บเซอร์วิสนั้น ต้องการที่จะให้แอปพลิเคชัน มีการทำงานกับแอปพลิเคชันที่ทำงานอยู่ในเครื่องอื่น โดยที่ผ่านทางเครือข่าย โดยในปัจจุบันนี้ได้เทคโนโลยีการสื่อสารระหว่างอ็อบเจ็ค (Object) ในระยะไกล (Remote Procedure Calls: RPC) เครือข่ายที่ปัจจุบันใช้กันแพร่หลาย และสะดวกมากที่สุดได้แก่เครือข่ายอินเทอร์เน็ต (Internet) ซึ่งให้การสื่อสารผ่านโปรโตคอลเอชทีทีพี (HTTP) เป็นส่วนใหญ่ เทคโนโลยีระยะไกล (RPC) ที่กล่าวข้างต้นนั้นต่างก็มีปัญหาในด้านการนำมาใช้งานในแง่ของความเข้ากันได้ของการเรียกใช้งานข้ามเทคโนโลยี เนื่องจากเป็นเทคโนโลยีเฉพาะของแต่ละค่าย ซึ่งผู้พัฒนาระบบจะต้องพัฒนา โปรแกรมที่มีความซับซ้อน และยังมีความมีปัญหาในส่วนของการไฟร์วอลล์ (Firewalls) และพร็อกซีเซิร์ฟเวอร์ (Proxy Server) ด้วยเนื่องด้วยปกติแล้วเซิร์ฟเวอร์จะปิดการสื่อสารที่ไม่ใช่โปรโตคอล HTTP ออกไป เพื่อความปลอดภัยของระบบที่มีการติดต่อสื่อสารกับภายนอก

ดังนั้นทางเลือกของการสื่อสารที่จะนำมาใช้ในการทำเว็บเซอร์วิสนั้น ก็คือให้ทำงานอยู่บน โพรโทคอลเอชทีทีพี โดยที่โซฟนอกจากจะทำงานบน โพรโทคอลเอชทีทีพีแล้วยังเป็นมาตรฐานเปิดซึ่ง จะทำให้สามารถติดต่อสื่อสารกับเครื่องคอมพิวเตอร์ที่มีความแตกต่างกันทั้งระบบปฏิบัติการ ,เทคโนโลยี รวมไปถึงภาษาที่ใช้ในการพัฒนาด้วยก็ได้



รูปที่ 2.2 รูปโครงสร้างการให้บริการ

(ที่มา : <http://www.ku.ac.th/e-magazine/march45/it/web3.html>, สำนักบริการคอมพิวเตอร์, 2545)

การติดต่อสื่อสารกันระหว่างผู้ขอใช้บริการ (Service Requestor) และผู้ให้บริการ (Service Provider) จะเป็นการคุยกันระหว่างแอปพลิเคชันกับแอปพลิเคชัน (Application-to-Application) ซึ่งวิธีการพูดคุยหรือการเชื่อมต่อนั้นจะใช้ SOAP เป็นตัวกลาง

2.2 การพัฒนาของเอ็กซ์เอ็มแอล (XML) โพรโทคอล

SOAP มีความสัมพันธ์กับ XML โพรโทคอลเป็นอย่างดีเพราะ โซฟมีโครงสร้างเป็น XML ซึ่งเทคโนโลยีของ XML เองนั้นก็ได้มีการพัฒนาเพิ่มขึ้นเป็นอย่างมาก โดยในยุคแรก XML จะถูกใช้งานในลักษณะของข้อมูลตามมาตรฐาน XML 1.0 เช่น WDDX (Web Distributed Data Exchange), XML-RPC

ต่อมาในยุคที่สองได้มีการเพิ่มความสามารถของ XML ในเรื่องของ XML Namespaces และ Schema ซึ่งโซฟจัดว่าอยู่ในยุคที่สองของ XML โพรโทคอล ในยุคที่สองของ XML นั้นได้กำจัดข้อบกพร่องของยุคแรกๆในสองข้อหลักคือ

2.2.1 ในยุคแรกจะมีข้อบกพร่องในด้านการเปลี่ยนแปลงโครงสร้างเอกสาร เช่น ส่วน XML-RPC หรือ WDDX ถ้าต้องการเพิ่มให้โปรโตคอลรองรับข้อมูลใหม่ เช่น ข้อมูลไบนารี ทั้ง XML-RPC และ WDDX จะต้องทำการอัปเดตมาตรฐานของคนใหม่ ทำให้ผู้ที่เขียน โปรแกรมในภาษาต่างๆ จะต้องมีการเปลี่ยนแปลงตามไปด้วยซึ่งทำให้การเปลี่ยนเวอร์ชัน (Version) ของมาตรฐานไม่คล่องตัว แต่ในยุคที่สองของ XML โปรโตคอล จะใช้เนมสเปซ (Namespaces) เข้ามาช่วยในการแก้ปัญหานี้

2.2.2 เรื่องการทำงานกับข้อมูลที่มีหลายชนิด (Data type) เนื่องจากในยุคแรกจะใช้ดีทีดี Document Type Definition (DTD) อธิบายโครงสร้างของเอกสาร XML ซึ่งดีทีดี จะไม่มีส่วนของการอธิบายความหมายของชนิดข้อมูล แต่ในยุคที่สองจะใช้ XML Schema แทนซึ่งจะมีการอธิบายทั้งรูปแบบโครงสร้างของลำดับ (syntax) และข้อมูลของชนิดของข้อมูลด้วย

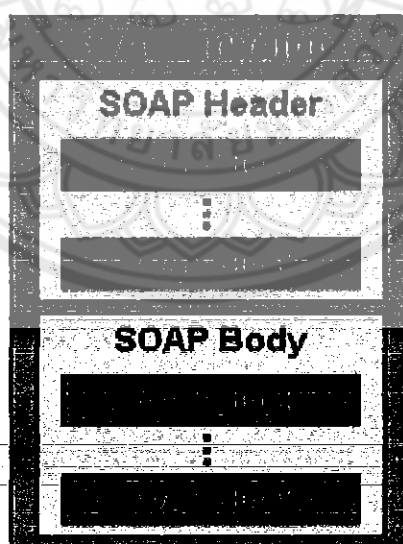
2.3 โครงสร้างของโซฟ (SOAP)

เอกสารโซฟนั้นมีโครงสร้างในรูปแบบ XML ซึ่งสามารถแบ่งเป็นส่วนของเอกสารได้เป็น 3 ส่วนหลักดังนี้คือ

2.3.1 SOAP envelop เนื้อหาสาระ (Content) ของเอกสารทั้งหมด

2.3.2 SOAP header ส่วนเพิ่มเติมเอกสาร โซฟซึ่งจะมีก็ได้ หรือไม่มีก็ได้

2.3.3 SOAP body ส่วนที่ใช้ในการเรียกใช้งานเซอร์วิส และผลลัพธ์ที่ได้จากเซอร์วิส



รูปที่ 2.3 รูปโครงสร้างการให้บริการ

(ที่มา : <http://msdn2.microsoft.com/en-us/library/ms951268.aspx>, 2007)

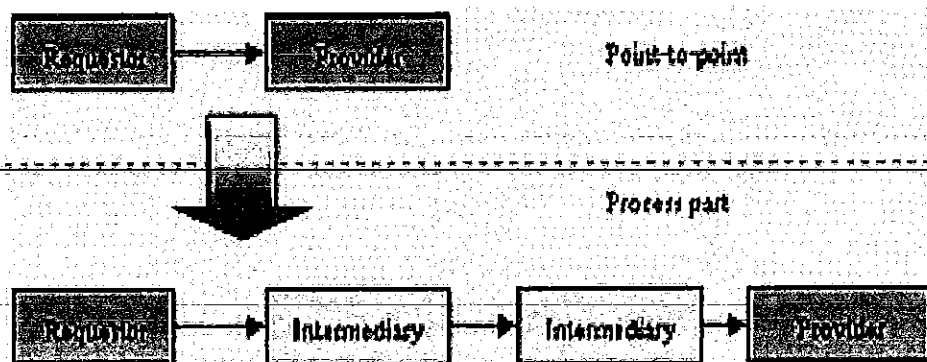
ตัวอย่างเอกสาร SOAP อย่างง่ายของการสอบถามราคาโน้ตบุ๊ก (Notebook) ยี่ห้อ Acer รุ่น Aspire 5051ANWXMิ ดังตัวอย่างข้างล่างนี้

```
<soap:Envelope>
  <soap:Body>
    <GetPrice>
      <Item>Aspire 5051ANWXMิ</Item>
    </GetPrice>
  </soap:Body>
</soap:Envelope>
```

จะเห็นว่า SOAP Envelope ก็เป็นเอกสารที่อยู่ในรูปแบบ XML ทั่วไป แต่สิ่งที่ทำให้ SOAP มีความสามารถมากขึ้นก็คือ SOAP ได้มีการกำหนดโครงสร้างของเอกสารเป็นส่วนๆ โดยอธิบายว่าส่วนใดมีหน้าที่อะไร มีข้อมูลอะไรอยู่ในส่วนนั้น และใครคือผู้ที่จะต้องสนใจในส่วนนั้นๆ และกล่าวถึงว่าส่วนใดจำเป็นในการใช้งาน ทำให้การพัฒนาโปรแกรมมีทิศทางที่ค่อนข้างชัดเจนขึ้น แต่สามารถเพิ่มขยายได้ในอนาคตอีกด้วย

2.4 ตัวกลางในการส่งเอกสาร SOAP (SOAP Intermediaries)

จากตัวอย่างเอกสาร SOAP นั้นจะเห็นว่าในส่วนของ <Header> Element จะเป็นส่วนที่จะขยายสารสนเทศ (information) ใหม่ๆ เข้าไปใน SOAP ได้ซึ่งตัวอย่างมีการอธิบายให้ผู้รับเอกสารตัวอย่างนี้ให้ทราบว่าในการสอบถามราคานี้ให้ใช้เงินในสกุลเงินบาท เป็นต้น ซึ่งจะเรียกการขยายในส่วนนี้ว่าเป็นการขยายในแนวตั้ง (Vertical extensibility) แต่ถ้าจะมากดูในด้านการขยายในแนวราบ (Horizontal extensibility) ของ SOAP นั่นก็คือทำอะไรที่จะให้มีการส่งเอกสาร SOAP ไปยังปลายทางโดยเส้นทางของการเดินทางของเอกสารอาจจะไม่เป็นลักษณะแบบจุดต่อจุด (Point-to-Point) แต่อาจจะเป็นการส่งเอกสารตามเส้นทางโดยเอกสารยังคงเป็นเอกสารตัวเดิมดังรูปด้านล่าง



รูปที่ 2.4 การทำงานที่มีการส่งเอกสาร โฆษแบบจุดต่อจุดและแบบส่งตามเส้นทาง

(ที่มา : <http://www.cmsthailand.com/ws/index.php>, 2004)

ตัวกลางในการส่งเอกสาร โฆษจากต้นทาง ในที่นี้คือ “Requestor” ไปยัง “Provider” เพื่อสร้างกระบวนการบางอย่างกับเอกสารซึ่งในการสร้างเส้นทางในการส่งแบบนี้อาจมีจุดประสงค์ดังต่อไปนี้คือ

2.4.1 ทำหน้าที่เป็น Crossing trust domains เนื่องจากการทำงานของ Web Services เป็นการใช้เรียกงานแบบระยะไกล หรือมีการทำงานแบบระบบกระจาย (Distributed System) ดังนั้นวิธีการในด้านความปลอดภัยจึงเป็นส่วนหนึ่งที่ต้องคำนึงถึง ซึ่งวิธีการหนึ่งก็คือให้มีการส่งเอกสารมายังจุดหนึ่งที่มีความน่าเชื่อถือได้ก่อน แล้วส่งต่อไปยังปลายทางอีกที่หนึ่ง ซึ่งหลักการใช้ Trust Domains อาจจะดูตัวอย่างง่ายๆ เช่น ในองค์กรอาจจะมีเครือข่ายอินเทอร์เน็ต (Intranet) อยู่ในองค์กร เครื่องคอมพิวเตอร์แต่ละเครื่องอาจจะสามารถติดต่อไปยังเซิร์ฟเวอร์ (Server) ได้โดยตรง แต่ถ้าต้องการออกไปยังอินเทอร์เน็ต หรือ เอ็กซ์ทราเน็ต (Extranet) ที่อยู่นอกองค์กรจะต้องผ่าน Trust Domains ก่อนในที่นี้อาจจะเป็นไฟลต์วอลล์ หรือ Virtual Private Network (VPN) gateways เพื่อความปลอดภัย

2.4.2 ทำหน้าที่เป็นตัวขยายการทำงานแบบระบบกระจาย (Distributed System) ถ้ามองดูเฉพาะแง่ของการขอใช้บริการจาก Requestor ไปยังผู้ให้บริการ Provider โดยผู้ให้บริการจะมีกระบวนการทำงานบางอย่างภายในและตอบกลับ ไปยังผู้ขอใช้ ซึ่งการทำงานแบบนี้ก็อยู่บนพื้นฐานของ Request and Respond นั้นเอง แต่ถ้าระบบมีการขยายใหญ่ขึ้นสิ่งที่จะต้องดูเพิ่มขึ้นก็คือความสามารถในการรองรับงานของผู้ให้บริการด้วย ดังนั้นก่อนที่จะมีการส่งเอกสาร โฆษไปยังเซิร์ฟเวอร์ ของผู้ให้บริการนั้น อาจจะมีการผ่าน Intermediary Server ก่อนเพื่อทำหน้าที่เป็นบัฟเฟอร์ (Buffer) รับเอกสาร โฆษมาพักไว้ และก็ส่งเอกสารให้ผู้ให้บริการ (Provider) อีกทอดหนึ่ง การทำงานเช่นนี้จะช่วยให้เซิร์ฟเวอร์ของผู้ให้บริการไม่รับภาระงานมากเกินไปในช่วงเวลาหนึ่งจนอาจทำให้เกิดความเสียหายได้และ Intermediary อาจจะมีการรวบรวมเอกสารแล้วส่งมาให้ผู้ให้บริการเป็นลักษณะ Batch เพื่อเพิ่มประสิทธิภาพการใช้งาน Intermediary ลักษณะนี้ผู้ขอใช้บริการอาจจะไม่เห็น Intermediary แต่จะมองเห็นว่าติดต่ออยู่กับผู้ให้บริการโดยตรง

2.4.3 การใช้งาน Intermediary ในลักษณะ Value-added ให้กับระบบ เช่น ใช้ Intermediary ในการเข้ารหัสข้อมูล (Encrypts) และเซ็นอิเล็กทรอนิกส์ (Digital signs) ในการส่งและรับ หรืออีกตัวอย่างหนึ่ง เช่นต้องการเก็บข้อมูลเพิ่มนำไปวิเคราะห์การทำงานของระบบ โดยการรับ-ส่งเอกสารจะผ่าน Intermediary แต่ไม่ได้ทำการประมวลผลเอกสารแต่จะเก็บของข้อมูลบางอย่างไว้เท่านั้นเช่นนำไปวิเคราะห์ค่า QoS (Quality of Service) หรือ Bottleneck ของระบบ เป็นต้น

2.5 Intermediaries ใน SOAP

เอกสาร SOAP จะถูกส่งจากผู้ขอใช้บริการผ่าน Intermediary's นั้นเป็นไปจนถึงปลายทางคือผู้ให้บริการ ดังนั้นเอกสาร SOAP ก็จะถูกเปิดเผยทั้งหมดกับ Intermediaries ด้วย แต่ในส่วนที่ Intermediaries สนใจจะอยู่เฉพาะส่วนที่เป็นหัวข้อ (header) เท่านั้น โดยในส่วนของหัวข้อจะมีส่วนของแอทริบิวต์ (Attribute) หนึ่ง เพื่อใช้ในการระบุว่าใครที่ควรจะได้รับทำงานกับส่วนหัวนี้ (ผู้ประมวลผลเอกสาร) ซึ่งแอทริบิวต์นั้นก็คือ actor โดยค่าที่ Actor จะต้องใส่ก็คือ URI (Universal Resource Indicator) ของ Intermediary นั้นเอง ดังตัวอย่าง

```
<soap:Header>
<i:Authentication
  xmlns:i="http://www.i3t.or.th/ws/security"
  soap:actor="urn:I3T:SecurityGateWay"
  soap:mustUnderstand="1">
  <i:username>Acompany</i:username>
  <i:password>SOAPpath</i:password>
</i:Authentication>
</soap:Header>
```

ดังนั้นเมื่อ Intermediary รับเอกสาร SOAP โดยถ้ามีชื่อ attribute actor ตรงกับของตัวเอง Intermediary จะเปลี่ยนเนื้อหาในส่วนหัวที่ได้เพื่อส่งต่อไปยัง Intermediary อื่นหรือไปยังผู้ให้บริการ ซึ่งจุดประสงค์หลักที่มีการออกแบบการทำงานแบบนี้ขึ้นมา เพื่อที่จะลดความซับซ้อนของเอกสารลง นอกจากนี้แล้วยังมียูอาร์แอล (URL) ของแอทริบิวต์แอคเตอร์ (attribute actor) ชนิดพิเศษอีกหนึ่งค่า นั่นคือ <http://schemas.xmlsoap.org/soap/actor/next> เพื่อบอกว่าผู้รับเอกสาร SOAP ในจุดต่อไปจะเป็นผู้ที่จะประมวลผลของเอกสาร ซึ่งกระบวนการนี้จะมีประโยชน์อย่างมากในการส่งเอกสาร SOAP เป็นทอดๆ (hop-by-hop)



รูปที่ 2.5 โครงสร้างของเว็บเซอร์วิส

(ที่มา : <http://www.cmsthailand.com/ws/index.php>, 2004)



2.6 ภาษาที่ใช้ในการพัฒนาเว็บเซอร์วิส

สำหรับภาษาที่ใช้ในการพัฒนานั้น สามารถแบ่งได้เป็น 2 ประเภทใหญ่ๆ ได้แก่คอตเน็ต (.NET) และเจทูอีอี (J2EE)

2.6.1 เอเอสพีคอตเน็ต (ASP.NET) ได้พัฒนาด้วยภาษาในกลุ่มของคอตเน็ตเทคโนโลยี (.NET Technology) เช่น ซีชาร์ป (C#), วีบี (VB) เป็นต้น

2.6.2 เจทูอีอี (J2EE) ได้แบ่งเป็น EJB, JSP, Java Servlet, โดยทั้งหมดใช้จาวา (Java)

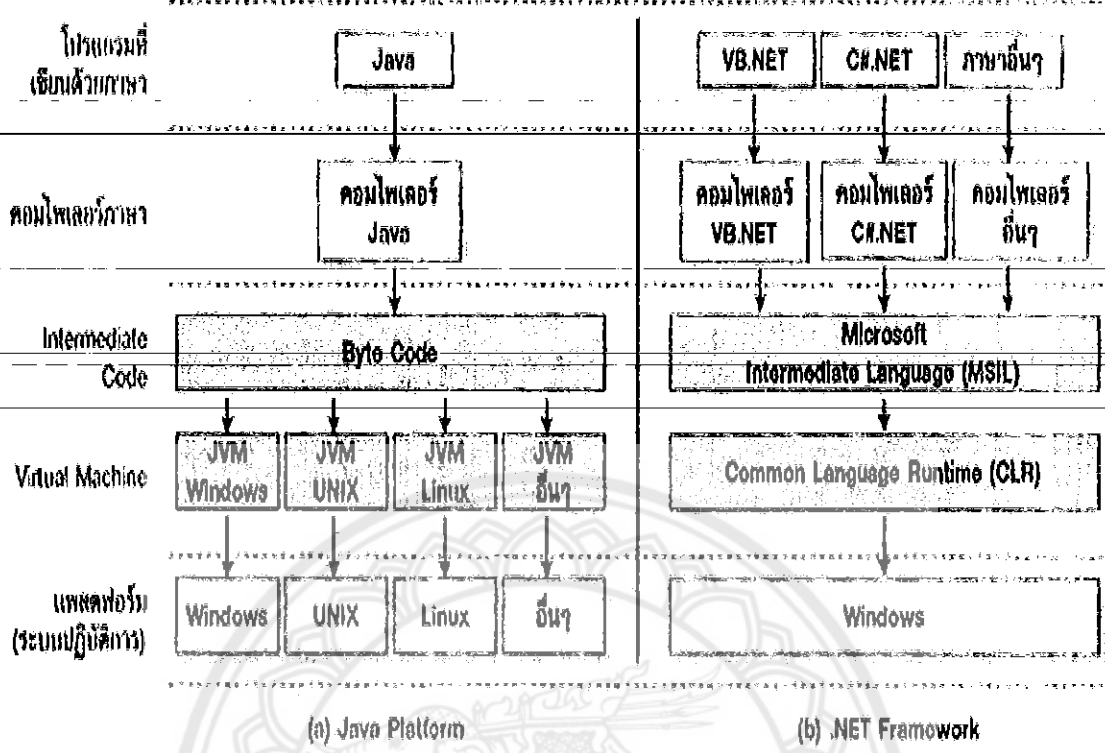
2.7 การเปรียบเทียบระหว่างข้อดีและข้อเสียของ .NET และ J2EE

	คอตเน็ต	เจทูอีอี
ข้อดี	<ol style="list-style-type: none"> 1.) คอตเน็ตเฟรมเวิร์ค(.NET Framework) ฟรี 2.) ไม่จำกัดภาษา สามารถเลือกตามความถนัดในการพัฒนา (Language neutral) แต่มีข้อจำกัดคือปฏิบัติงานได้บนระบบปฏิบัติการวินโดวส์เท่านั้น (Windows Centric) หรือที่เรียกว่า "Write any language , run on Windows" 3.) มีคุณสมบัติของโอโอพีอยู่เพราะภาษาในคอตเน็ต (.NET) นั้นยังคงจะยึดคุณสมบัติของโอโอพี 4.) ซ่อนซอร์สโค้ด (Code Behind) 5.) ง่ายต่อการพัฒนาเนื่องจากมีวิซวลสตูดิโอคอตเน็ต (Visual Studio .NET) ซึ่งน่าจะเป็นไอดีอี (IDE) ที่ดีที่สุด 6.) มีเอ็มเอสดีเอ็น (MSDN) ซึ่งง่ายต่อการทำความเข้าใจหากสงสัยเรื่องใดๆ 7.) ง่ายต่อการศึกษาและทำความเข้าใจ เขียนง่าย และเซอร์เวอร์ก็สร้าง (Maintain) ง่ายกว่า 	<ol style="list-style-type: none"> 1.) เจทูอีอี(J2EE) แจกฟรี 2.) มีเทคโนโลยีให้เลือกใช้มากมายตามความเหมาะสมที่จะนำมาใช้ในการพัฒนา 3.) มีคุณสมบัติของโอโอพี (OOP) ตามแบบฉบับจาวา (Java)

	คอตเน็ต	เจทูอีอี
ข้อเสีย	1.) รันได้เฉพาะบนไมโครซอฟท์แพลตฟอร์ม (Microsoft Platform)	1.) ถึงจาวาจะเป็นโอเพนซอร์ส (Open Source) และแจกฟรี แต่ระบบที่จะรันต้องมี
	2.) Visual Studio.NET มีราคาแพงมาก (แต่ก็มี Sharp developer และ Web Matrix ให้ใช้)	ต้นทุนสูงกว่า ไมโครซอฟท์แพลตฟอร์ม รวมถึงบุคลากรต้องมีความรู้พอสมควร จึงถือเป็นการเพิ่มต้นทุนแก่องค์กรในระยะยาว
		2.) ไลต์อีอีที่ใช้ในการพัฒนายังไม่ดีเท่าที่ควร
		3.) ยากต่อการศึกษาและทำความเข้าใจ คือเขียนยากกว่าและสร้างเซิร์ฟเวอร์ (Server Maintain) ยากกว่า ทั้งเจเอสพี (JSP) และโค้ดของคอตเน็ตจะถูก Compile เหมือน Application ทั่วไปแต่ในโค้ดของเจเอสพีจะถูกคอมไพล์เป็น HTTP Servlet Class ในตอนที่เว็บถูกเรียกใช้ครั้งแรก (ทำให้นานเฉพาะครั้งแรก) ซึ่งคอตเน็ตจะถูกคอมไพล์เป็น แอสเซมบลี (DLL แบบใหม่ของคอตเน็ต) เก็บไว้เลย ทำให้จาวาช้ากว่า คอตเน็ต

ตารางที่ 2.1 การเปรียบเทียบระหว่างข้อดีและข้อเสียของคอตเน็ต (.NET) และเจทูอีอี (J2EE)

จากข้อมูลการเปรียบเทียบระหว่างข้อดีและข้อเสียของคอตเน็ต (.NET) และเจทูอีอี (J2EE) นั้นทางผู้จัดทำโครงการ ได้ข้อสรุปว่าโครงการนี้เลือกใช้คอตเน็ต (.NET) มาใช้ในการพัฒนาโปรแกรม เนื่องจากข้อดีของคอตเน็ต (.NET) ที่มีมากกว่า เจทูอีอี (J2EE) แล้วคอตเน็ตก็ง่ายต่อการศึกษาและความเข้าใจด้วย



รูปที่ 2.6 การเปรียบเทียบคอกเนตเฟรมเวิร์ค (.NET Framework) และจาวาแพลตฟอร์ม (Java Platform) (ที่มา : <http://www.cmsthailand.com/ws/index.php>, 2004)

2.8 คอกเนตเฟรมเวิร์ค (.NET Framework)

ไมโครซอฟท์ต้องการที่จะสร้างสิ่งที่เป็นมาตรฐานขึ้นมา เพื่อให้ทุกสิ่งทุกอย่างสามารถติดต่อสื่อสารกันได้หมด โดยได้ทำการคิดค้นระบบให้เป็นระบบมาตรฐาน ระบบนี้ก็คือคอกเนตเฟรมเวิร์ค ซึ่งระบบนี้ไม่ใช่ระบบปฏิบัติการ (OS) แต่เปรียบเสมือนโปรแกรมหนึ่งที่จะสามารถสร้างสถานะแวดล้อมหนึ่ง ซึ่งสามารถทำงานในระบบคอกเนตนี้ได้

คอกเนตเฟรมเวิร์ค คือ โครงสร้างการพัฒนาโปรแกรมคอมพิวเตอร์ ที่ถูกออกแบบมาเพื่อให้อำนวยความสะดวกในการพัฒนาโปรแกรมสมัยใหม่ ที่ใช้งานในระบบเครือข่าย (Internet, Intranet, Mobile Devices, ฯลฯ) Bill Gates และ Steve Ballmer ได้บรรยายสรุปวิสัยทัศน์ ที่เกี่ยวกับคอกเนตเอาไว้ 3 ข้อหลัก ๆ ได้แก่

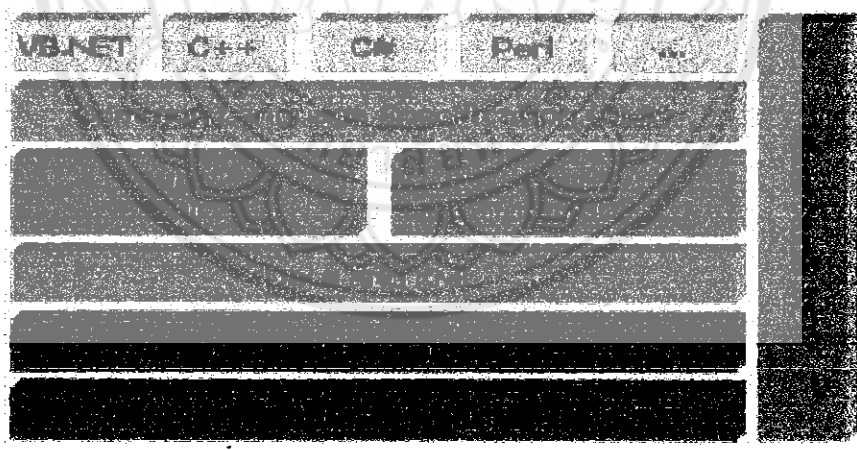
1. การพัฒนาโปรแกรมในรูปแบบของเว็บเซอร์วิส จะเป็นหัวใจหลักในการขับเคลื่อนโปรแกรมต่างๆ ที่ใช้งานบนเว็บเซอร์วิสอินเทอร์เน็ตจะช่วยให้การติดต่อสื่อสารระหว่างแอปพลิเคชันบนอินเทอร์เน็ตนั้นง่ายขึ้น และเป็นระบบมากยิ่งขึ้น

2. เว็บเซอร์วิสขั้นพื้นฐานเช่น การตรวจสอบผู้ใช้ที่ล็อกอิน (log in) เข้าสู่ระบบ จะถูกพัฒนาให้เป็นมาตรฐาน และสามารถนำไปใช้ได้ทั่วไปบนอินเทอร์เน็ต

3. พีซี (PC = desktop, notebook) และเครื่องมือแบบเคลื่อนที่ (Mobile Device) ที่ต่อเชื่อมกับระบบอินเทอร์เน็ตได้ เช่น PDA และ โทรศัพท์มือถือ จะมีบทบาทและประโยชน์มากขึ้นไปอีก เมื่อสามารถติดต่อใช้งาน โปรแกรมต่างๆ บนอินเทอร์เน็ตได้

ไมโครซอฟท์จึงได้พัฒนารูปแบบการพัฒนาเว็บคอมพิวเตอร์ขึ้นมา ซึ่งเรียกว่า คอทเน็ตเฟรมเวิร์ก นั่นเอง (อันที่จริงแล้วไมโครซอฟท์ไม่ได้เป็นผู้คิดค้นเรื่องพวกนี้ขึ้นมาแต่เพียงผู้เดียว อย่าเข้าใจผิด สิ่งต่างๆ เหล่านี้มีผู้คิดค้นพัฒนาจากหลายบริษัท หลายหน่วยงาน ยกตัวอย่างเช่น ซัน ไมโครซิสเต็ม (Sun Microsystems), ไอบีเอ็ม (IBM), ฯลฯ หากแต่ไมโครซอฟท์นำมาแนวคิดเหล่านั้นมาออกแบบให้อยู่ในรูปแบบที่สามารถต่อเชื่อมกันได้ง่ายขึ้น เป็นระบบมากขึ้น) เพราะฉะนั้น ถ้าจะให้นิยามคำว่าคอทเน็ต คงไม่สามารถชี้เฉพาะได้ว่าคอทเน็ตคืออะไร เพราะจริงๆ แล้วคอทเน็ตประกอบไปด้วยส่วนประกอบต่างๆ หลายส่วนด้วยกัน ส่วนประกอบเหล่านี้ถูกออกแบบมาเพื่อให้ทำงานได้ดียิ่งขึ้น

ในอนาคตไมโครซอฟท์ก็หวังที่จะนำเอาระบบนี้ไปติดตั้งลงบนอุปกรณ์ทุกชนิด เพื่อให้อุปกรณ์ทุกอย่างมีระบบๆ หนึ่งที่เหมือนกันหมด โดยคอทเน็ตเฟรมเวิร์ก นั้นมี ส่วนประกอบภายในแบ่งออกเป็นชั้นดังนี้คือ



รูปที่ 2.7 โครงสร้างของคอทเน็ตเฟรมเวิร์ก
(ที่มา : <http://www.cmsthailand.com/ws/index.php>, 2004)

2.8.1 Programming Language

เป็นภาษาที่ใช้สร้างโปรแกรมซึ่งสามารถทำงานได้ภายใต้สภาวะของคอตเน็ต โดยไมโครซอฟท์ได้เปิดตัวภาษาหลักๆที่จะใช้พัฒนาบนคอตเน็ตนี้มีจำนวน 3 ภาษาด้วยกันคือ

- 1.) ซีชาร์ปเป็นภาษาใหม่ที่ไม่โครซอฟท์พัฒนามาจากภาษาซีพลัสพลัส (C++) กับภาษาจาวา (Java) เป็นหลัก
- 2.) วิบีคอตเน็ต (VB.NET = Visual Basic .NET หรือ Visual Basic เวอร์ชัน 7.0) เป็นภาษาที่พัฒนามาจาก Visual Basic ในเวอร์ชัน 6.0
- 3.) เจสคลิปต์คอตเน็ต (Jscript.NET) เป็นภาษาที่ได้พัฒนามาจากเจสคลิปต์ (Jscript) ซึ่งเป็นจาวาสคลิปต์ (JavaScript) ในเวอร์ชันของไมโครซอฟท์

2.8.2 Base Classes Library

ไลบรารี (Library) นั้นเปรียบเทียบกับเสมือนชุดคำสั่งสำเร็จรูปย่อยๆที่เพิ่มเข้ามา ซึ่งส่วนใหญ่จะเป็นชุดคำสั่งที่ต้องใช้งานอยู่เป็นประจำ ดังนั้นจึงมีผู้คิดค้นเพื่ออำนวยความสะดวกในการเขียน โปรแกรม ซึ่งไลบรารีในภาษาต่างๆส่วนใหญ่จะอยู่ในรูปแบบไฟล์ incould แต่ถ้าเป็น ASP สิ่งที่เป็น ไลบรารีก็คือคอมโพเนนท์ต่างๆนั่นเอง ซึ่งภายในระบบคอตเน็ตจะสร้างสิ่งที่เรียกว่าเป็นไลบรารี พื้นฐานขึ้น ทำให้ไม่ว่าจะใช้ภาษาใดในการพัฒนาโปรแกรมก็สามารถที่จะเรียกใช้ไลบรารีที่เป็นตัวเดียวกันได้หมด

2.8.3 Common Language Runtime (CLR)

ถือว่าเป็นสิ่งที่สำคัญที่สุดของระบบคอตเน็ตนี้ก็ได้ว่า เพราะ CLR ที่ว่านี้มีหน้าที่ทำให้โปรแกรมที่เขียนขึ้นมาด้วยภาษาต่างๆกัน กลายเป็นภาษารูปแบบมาตรฐานเดียวกัน ทั้งหมดเราเรียกภาษาที่ว่านี้ว่า Intermediate Language (IL) ซึ่งเมื่อต้องการที่จะรันโปรแกรมใด CLR ที่ว่านี้ก็จะทำการตรวจสอบเครื่องที่รันว่ามีสภาวะแวดล้อมการทำงานเช่นใด หลังจากนั้นก็จะคอมไพล์เป็น โปรแกรมที่เหมาะสมต่อการทำงานของเครื่องนั้น ทำให้สามารถใช้งาน โปรแกรมต่างๆ ได้อย่างมีประสิทธิภาพสูงสุดในแต่ละเครื่องที่ได้ใช้งาน

2.8.4 ข้อดีของคอตเน็ตเฟรมเวิร์ค

ประโยชน์และข้อดีของคอตเน็ตเฟรมเวิร์คนี้พอที่จะสรุปออกมาได้เป็นข้อดังนี้

1. เป็นระบบที่มีไลบรารีที่เป็นมาตรฐานเดียวกัน แล้วก็เนื่องจากมีไลบรารีที่เป็นมาตรฐานเดียวกันทั้งหมดทำให้ไม่ต้องกังวล ว่าภาษาที่ใช้เขียนนั้นจะมีไลบรารีตัวนั้นตัวนี้หรือไม่ รวมทั้งไม่ต้องคอยวิตกกังวลว่าถ้าใช้ไลบรารีของภาษาหนึ่งแล้วอีกภาษาหนึ่งจะ ไม่มีไลบรารีตัวนั้น
2. ไม่ขึ้นอยู่กับระบบปฏิบัติการ (OS) เนื่องด้วยระบบปฏิบัติการที่แต่ละบุคคลหรือองค์กรใช้นั้น ย่อมไม่เหมือนกัน แต่ภายในคอตเน็ตเฟรมเวิร์คจะไม่มีปัญหานี้ขอเพียงแค่มียคอตเน็ตเฟรมเวิร์คก็จะทำให้สามารถใช้งานกับ โปรแกรมต่างๆ ได้ ซึ่งเป็นข้อดีตรงที่สามารถใช้โปรแกรมต่างๆ ได้ทุกๆของระบบปฏิบัติการ

3. สามารถใช้ในการพัฒนาได้ทุกภาษา เนื่องจากทำให้ไม่ต้องคอยมาศึกษาภาษาใหม่ๆเมื่อต้องการที่จะสร้างโปรแกรมในแต่ละครั้ง นอกจากนั้นเรายังสามารถเลือก ใช้ภาษาที่เราถนัดที่สุดในการพัฒนาโปรแกรมต่างๆได้ด้วย

4. มีการควบคุมถึงแวดล้อมในการทำงานเป็นอย่างดี เนื่องจากเป็นระบบที่เป็นมาตรฐานที่ทำให้การควบคุมจัดสรรระบบต่างๆ ทำได้ง่ายขึ้น ไม่ว่าจะเป็นการจัดสรรหน่วยความจำ ในส่วนของด้านการใช้งานเครื่องมือก็มีความรวดเร็วมากขึ้น แล้วลด โอกาสที่เครื่องจะแคงก็ได้อย่างมีประสิทธิภาพ

5. ความปลอดภัยที่มีมากขึ้นนั้น คอทเน็ดเฟรมเวิร์คสามารถที่จะกำหนดสิทธิของการใช้งานหรือการอนุญาตให้ใช้งาน (Permission) ของผู้ใช้งานได้มากขึ้นทำให้สามารถกำหนดว่าจะใช้โปรแกรมใน ส่วนใดในการใช้งานได้หรือไม่ได้ แล้วแต่เฉพาะบุคคล



```

using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Diagnostics;
using System.Web;
using System.Web.Services;

namespace hello
{
    /// <summary>
    /// Summary description for Service1.
    /// </summary>
    public class Service1 : System.Web.Services.WebService
    {
        public Service1()
        {
            //CODEGEN: This call is required by the ASP.NET Web Services Designer
            InitializeComponent();
        }

        #region Component Designer generated code

        //Required by the Web Services Designer
        private IContainer components = null;

        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        private void InitializeComponent()
        {
        }

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        protected override void Dispose( bool disposing )
        {
            if(disposing && components != null)
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #endregion

        // WEB SERVICE EXAMPLE
        // The HelloWorld() example service returns the string Hello World
        // To build, uncomment the following lines then save and build the project
        // To test this web service, press F5

        [WebMethod]
        public string HelloWorld()
        {
            return "Hello World";
        }
    }
}

```

Service1

The following operations are supported. For a formal definition, please review the [Service Description](#).

- [HelloWorld](#)

This web service is using <http://tempuri.org/> as its default namespace.

Recommendation: Change the default namespace before the XML Web service is made public.

Each XML Web service needs a unique namespace in order for client applications to distinguish it from other services on the Web. <http://tempuri.org/> Web services that are under development, but published XML Web services should use a more permanent namespace.

Your XML Web service should be identified by a namespace that you control. For example, you can use your company's Internet domain name as a namespace. Although many XML Web service namespaces look like URLs, they need not point to actual resources on the Web. (XML Web service namespaces

For XML Web services created using ASP.NET, the default namespace can be changed using the `WebService` attribute's `Namespace` property. The `Namespace` attribute is applied to the class that contains the XML Web service methods. Below is a code example that sets the namespace to "<http://microsoft.com/webservices/>".

C#

```
[WebService(Namespace="http://microsoft.com/webservices/")]
public class MyWebService {
    // implementation
}
```

Visual Basic.NET

```
<WebService(Namespace="http://microsoft.com/webservices/")> Public Class MyWebService
    ' implementation
End Class
```

For more details on XML namespaces, see the W3C recommendation on [Namespaces in XML](#).

For more details on WSDL, see the [WSDL Specification](#).

For more details on URIs, see [RFC 2396](#).

รูปที่ 2.9 ผลลัพธ์เมื่อรัน โปรแกรมตัวอย่าง

```
<? xml version="1.0" encoding="utf-8" ?>
<string xmlns="http://tempuri.org/">Hello World</string>
```

รูปที่ 2.10 ผลลัพธ์ของเมสเสจที่เป็นภาษา XML

บทที่ 3

หลักการงานและโครงสร้างของโปรแกรม

3.1 ส่วนเว็บกลาง

ในส่วนของเว็บกลาง ผู้ที่เข้ามาใช้บริการสามารถเข้ามาหาข้อมูลของสินค้าหรือจะค้นหา (Search) รายการของสินค้าและข่าวสารหรือข้อมูลต่างๆ ได้

- **Link Product** จะแสดงส่วนของการค้นหาสินค้า (Product)

- **Link Acer** จะแสดงสินค้าโน้ตบุ๊กเฉพาะยี่ห้อ Acer โดยคอลลัมน์ซ้ายจะแสดง CPU ของ Intel ส่วน คอลลัมน์ขวาจะแสดง CPU ของ AMD

- **Link Compaq** จะแสดงสินค้าโน้ตบุ๊กเฉพาะยี่ห้อ Compaq โดยคอลลัมน์ซ้ายจะแสดง CPU ของ Intel ส่วน คอลลัมน์ขวาจะแสดง CPU ของ AMD

- **Link** จะแสดงส่วนของ Link ที่มีความเกี่ยวข้องกับข้อมูลหรือเว็บที่เว็บส่วนกลางต้องการแสดงให้แก่ผู้ใช้บริการ

- **Link Contact** จะเป็นการติดต่อผู้บริการ ในลักษณะต่างๆ เช่น ส่งเมลล์ หรือ เป็นการแนะนำบริษัท

เมื่อผู้ใช้บริการเข้ามายังหน้าหลักของเว็บ ไซต์ส่วนเว็บกลางจะสามารถเข้ารับบริการต่างๆจากเว็บกลางได้ดังต่อไปนี้

1. สามารถเลือกซื้อสินค้าต่างๆจากร้านที่เป็นสมาชิกกับเว็บกลาง โดยสามารถเลือกซื้อตามประเภทของสินค้าหรือค้นหาตามรายการของสินค้า เมื่อลูกค้าเลือกซื้อสินค้า ระบบจะมีการจัดเก็บสินค้าที่ลูกค้าได้เลือกซื้อไว้ เมื่อลูกค้าได้เลือกซื้อสินค้าครบตามที่ต้องการแล้ว ให้ลูกค้าทำการตรวจสอบรายละเอียดของรายการสินค้าที่ลูกค้าได้เลือกซื้อสินค้านก่อนกดยืนยันการซื้อสินค้า

2. ระบบจะทำการสรุปรายการสินค้าและราคาโดยรวมที่ผู้ซื้อ ได้เลือกซื้อสินค้า ก่อนกดยืนยันการซื้อสินค้า

3. ระบบจะแสดงแบบฟอร์มสำหรับกรอกข้อมูลของผู้ซื้อสินค้า โดยผู้ซื้อจะต้องทำการกรอกข้อมูลให้ถูกต้องเพื่อสิทธิและประโยชน์ของผู้ซื้อสินค้าโดยเฉพาะ แล้วให้ผู้ซื้อกดยืนยันการซื้อสินค้าในครั้งนั้น

4. เมื่อผู้ซื้อกดยืนยันการซื้อสินค้า ระบบจะจัดเก็บรายละเอียดข้อมูลของลูกค้าแล้วส่งข้อมูลไปยังร้านค้าย่อยที่เป็นสมาชิกของเว็บกลางที่ผู้ซื้อ ได้ซื้อสินค้าจากเว็บกลางโดยอัตโนมัติ

3.2 เว็บร้านค้าย่อย (Shop)

เว็บร้านค้าย่อย เป็นหน้าที่สำหรับขายสินค้าให้แก่ลูกค้า โดยสินค้าจะแบ่งออกตามประเภทที่กำหนดเพื่อสะดวกในการเลือกสินค้า และเว็บร้านค้าย่อยจะต้องเป็นสมาชิกของเว็บกลางด้วย ประโยชน์ที่เว็บร้านค้าย่อยจะได้รับคือ ทำให้เว็บร้านค้าย่อยเป็นที่รู้จักมากขึ้นและขายสินค้าได้มากขึ้น เพราะนอกจากจะขายสินค้าให้กับลูกค้าของร้านโดยตรงแล้วร้านค้ายังมีสิทธิที่จะเสนอขายสินค้าให้กับลูกค้าที่เป็นสมาชิกของเว็บกลางด้วย ซึ่งเว็บกลางจะไปดึงข้อมูลสินค้าที่มีอยู่ในฐานข้อมูลของเว็บร้านค้าย่อยมาแสดงบนหน้าเว็บ ไซต์ของเว็บกลาง เพียงแค่ร้านค้าย่อยสมัครสมาชิกกับทางเว็บกลางและทำการเปิดเซอวิสให้กับเว็บกลาง

เว็บร้านค้าย่อยที่ได้จำลองขึ้นใน โครงการนี้เป็นร้านค้าที่เปิดบริการขายสินค้าประเภทโน้ตบุ๊ก โดยเมื่อลูกค้าได้เข้าสู่เว็บไซต์ของร้านค้าย่อยลูกค้าก็จะเห็นสินค้าที่ได้จัดไว้เป็นหมวดหมู่ ตามประเภทสินค้าที่มีอยู่ ลูกค้าสามารถเข้าไปดูว่ามีสินค้าประเภทใดบ้าง ไม่เช่นนั้นเพื่อความสะดวกของลูกค้าทางเว็บกลางได้จัดให้ลูกค้าสามารถที่จะเข้าไปยัง Link Product ก็จะสามารถหาสินค้าได้ตามความต้องการ

3.2.1. ส่วนประกอบของเว็บไซต์ร้านค้า

3.2.1.1 หน้าแรกเว็บไซต์

ชื่อไฟล์ : default.aspx

รายละเอียด : เมื่อลูกค้าเข้ามายังเว็บไซต์ของเว็บร้านค้าย่อยแล้ว ลูกค้าจะพบกับหน้านี้เป็นหน้าแรก ซึ่งหน้านี้จะแสดงให้เห็นว่า ในเว็บร้านค้าย่อยมีสินค้าอะไรขายบ้าง และยังมีส่วนของ Link ที่อยู่ด้านบนทางขวาที่จะเข้าถึงข้อมูลต่างๆ ในหน้าต่อไปด้วย

3.2.1.2 หน้าของสินค้าในแต่ละประเภท

ชื่อไฟล์ : product.aspx

รายละเอียด : เมื่อลูกค้าเลือกมายังส่วนนี้ก็จะปรากฏหน้าเว็บไซต์ที่จะให้ลูกค้าเลือก ยี่ห้อที่ต้องการ และ หน่วยประมวลผล (CPU) หรืออาจจะเลือกให้เรียกแสดงสินค้าที่มีอยู่ทั้งหมดว่ามีสินค้าอะไรขายบ้าง โดยได้แสดงยี่ห้อ หน่วยประมวลผล (CPU) และราคาที่ต้องการ โดยการแสดงสินค้าแต่ละครั้งนั้นก็จะมีรายละเอียดของสินค้าแต่ละชนิดว่ามีคุณสมบัติอย่างไรบ้าง แสดงให้ลูกค้าได้ทราบรายละเอียด

3.2.1.3 หน้าแสดงรายการสินค้าที่ลูกค้าซื้อ

ชื่อไฟล์ : addToCart.aspx

รายละเอียด : เมื่อลูกค้าต้องการซื้อสินค้าให้ทำการ กดปุ่ม “หยิบใส่รถเข็น” สินค้าที่ลูกค้าเลือกซื้อแล้วจะถูกเพิ่มลงในรถเข็นสินค้าของลูกค้า ซึ่งจะแสดงรายละเอียดสินค้านี้

1. ชื่อสินค้า : แสดงชื่อสินค้าและยี่ห้อสินค้าที่ลูกค้าต้องการซื้อ
2. จำนวน : จำนวนสินค้าที่ลูกค้าต้องการสั่งซื้อ ซึ่งลูกค้าสามารถกรอกจำนวนที่ต้องการลงในช่องว่าง หลังจากนั้นให้ลูกค้ากดปุ่ม “Update Cart” เพื่อให้ระบบทำการคำนวณราคาใหม่อีกครั้ง
3. ราคา : เป็นราคาสินค้าต่อ 1 ชิ้น
4. ราคารวม : เป็นราคารวมของสินค้า โดยคิดตามจำนวนชิ้นที่ลูกค้าได้ทำการเลือกซื้อ เช่น ลูกค้าต้องการเลือกซื้อโน้ตบุ๊ก ยี่ห้อ Acer หน่วยประมวลผล (CPU) ของ AMD จำนวน 3 ชิ้น ราคาชิ้นละ 25,000 บาท ดังนั้นราคารวมจะเป็น 75,000 บาท
5. Delete Cart : เมื่อลูกค้าไม่ต้องการซื้อสินค้าชนิดใดแล้วก็สามารถที่จะลบออกจากรายการได้โดยการกดปุ่ม “Delete”
6. Update Cart : เมื่อมีการเปลี่ยนแปลงรายการสินค้า เช่น เพิ่มจำนวนสินค้าหรือลดจำนวนของสินค้าที่ต้องทำการกดปุ่ม “Update Cart” เพื่อที่จะคำนวณราคาสินค้าใหม่
7. Continue Shopping : เมื่อต้องการเลือกซื้อสินค้าเพิ่มอีก ลูกค้าก็สามารถกดปุ่ม “Continue Shopping” รายการสินค้าก็แสดงขึ้นมาให้ลูกค้าเลือกซื้ออีก
8. Check Out : เมื่อลูกค้าพอใจในสินค้าที่เลือกซื้อแล้ว โดยข้อมูลถูกต้องตามที่ลูกค้าเลือกซื้อ และลูกค้าได้กรอกรายละเอียดข้อมูลของลูกค้าแล้ว ให้กดปุ่ม “Check Out” เพื่อทำการยืนยันข้อมูลทั้งหมด

3.2.2. ขั้นตอนการสั่งซื้อสินค้า

1. ลูกค้าเลือกดูสินค้าที่ต้องการ เมื่อต้องการซื้อหรือพอใจในสินค้าแล้วให้กดปุ่ม “หยุดลงรถเงิน” หรือ ปุ่ม “Continue Shopping” เพื่อเพิ่มสินค้าที่ต้องการซื้ออีกพร้อมกับการระบุจำนวนสินค้าที่ต้องการซื้อให้เรียบร้อย
2. ถ้าต้องการเปลี่ยนแปลงรายการสินค้า เช่น ต้องการลบสินค้าที่ไม่ต้องการซื้อให้กดปุ่ม “Delete” หรือเปลี่ยนแปลงจำนวนสินค้า ก็ให้เปลี่ยนจำนวนในช่องว่างจำนวนเลย แล้วก็กดปุ่ม “Update Cart” เพื่อให้ระบบคำนวณยอดเงินใหม่อีกครั้ง
3. เมื่อพอใจในสินค้าแล้วให้ลูกค้าตรวจสอบสินค้าที่ต้องการซื้อว่าถูกต้องตามรายการที่ลูกค้าต้องการซื้อหรือไม่ เมื่อตรวจสอบจนแน่ใจแล้ว ก็ทำการกรอกรายละเอียดของลูกค้าให้ถูกต้องและครบถ้วนก่อนที่จะกดปุ่ม “Check Out” เพื่อยืนยันการซื้อสินค้าในครั้งนั้น รายละเอียดที่ลูกค้าได้กรอกไปนั้นส่วนของเว็บเซอร์วิสจะเป็นตัวจัดการส่งข้อมูลไปยังร้านค้าที่ลูกค้าได้ทำการซื้อสินค้า

3.3 ขั้นตอนการออกแบบ Data Flow Diagram

3.3.1 องค์ประกอบของเว็บแอปพลิเคชัน มีส่วนประกอบ 3 ส่วนใหญ่ คือ

1. ส่วนแสดงสินค้า
2. ส่วนค้นหาสินค้า
3. ส่วนการสั่งซื้อสินค้า

3.3.2 องค์ประกอบของเว็บเซอร์วิส มีส่วนประกอบ 3 ส่วนใหญ่ คือ

1. บริการแสดงข้อมูลสินค้า
2. บริการค้นหาสินค้า
3. บริการซื้อสินค้า

3.3.3 แบบจำลองการทำงานของระบบ

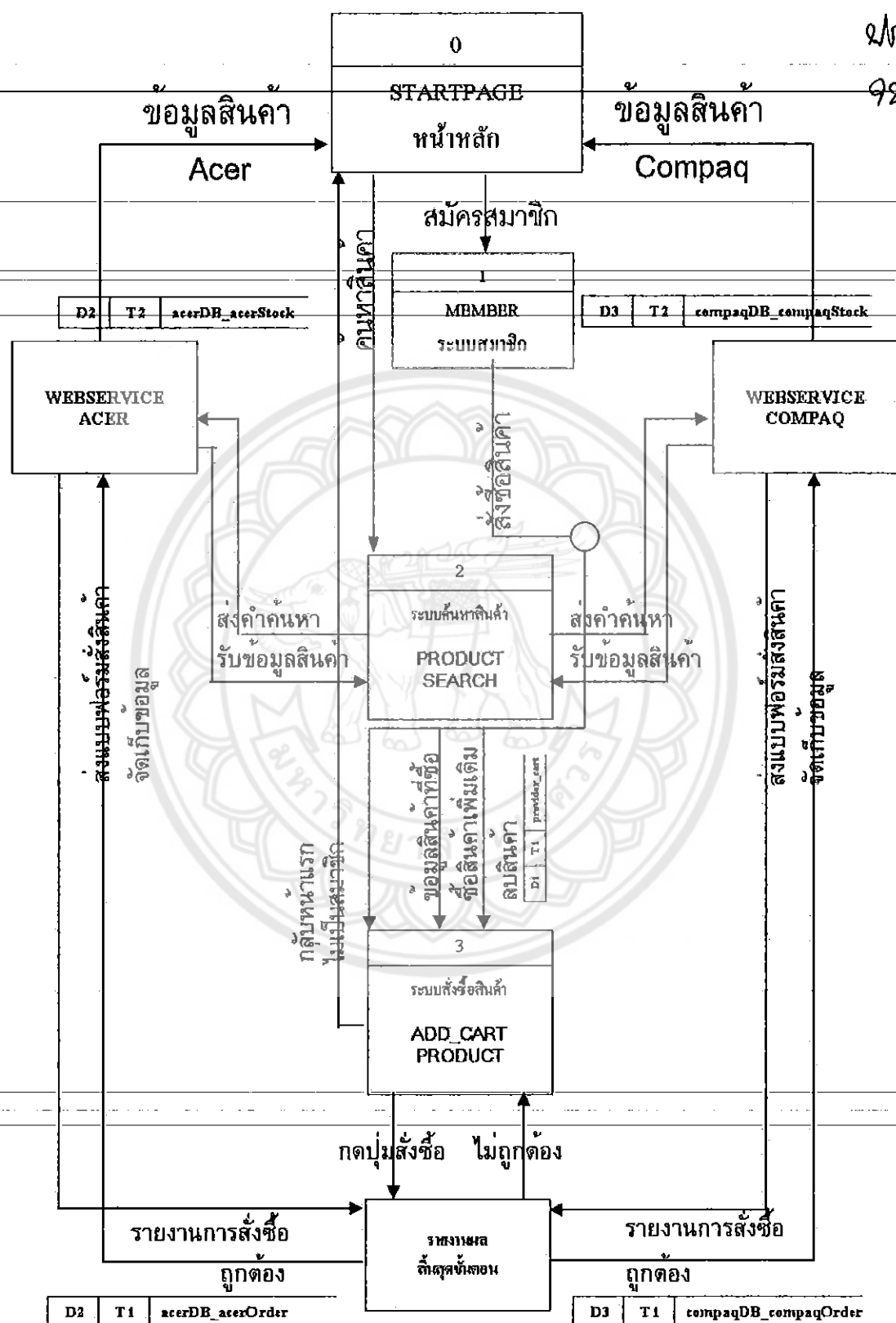
หลังจากวิเคราะห์องค์ประกอบของเว็บแอปพลิเคชันและเว็บเซอร์วิสแล้ว เราสามารถแสดงแบบจำลองขั้นตอนการทำงาน Process Modeling ด้วยการเลือกแผนภาพกระแสข้อมูล (Data Flow Diagram: DFD) ซึ่งแสดงให้เห็นภาพรวมการทำงานของระบบ ดังต่อไปนี้



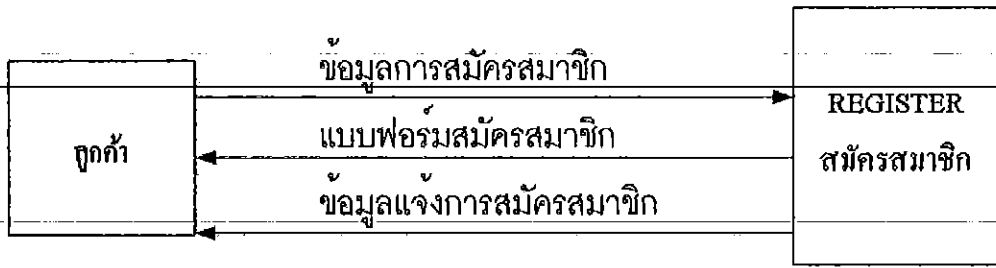


รูปที่ 3.1 แสดงแผนภาพบริบท (Context Diagram)

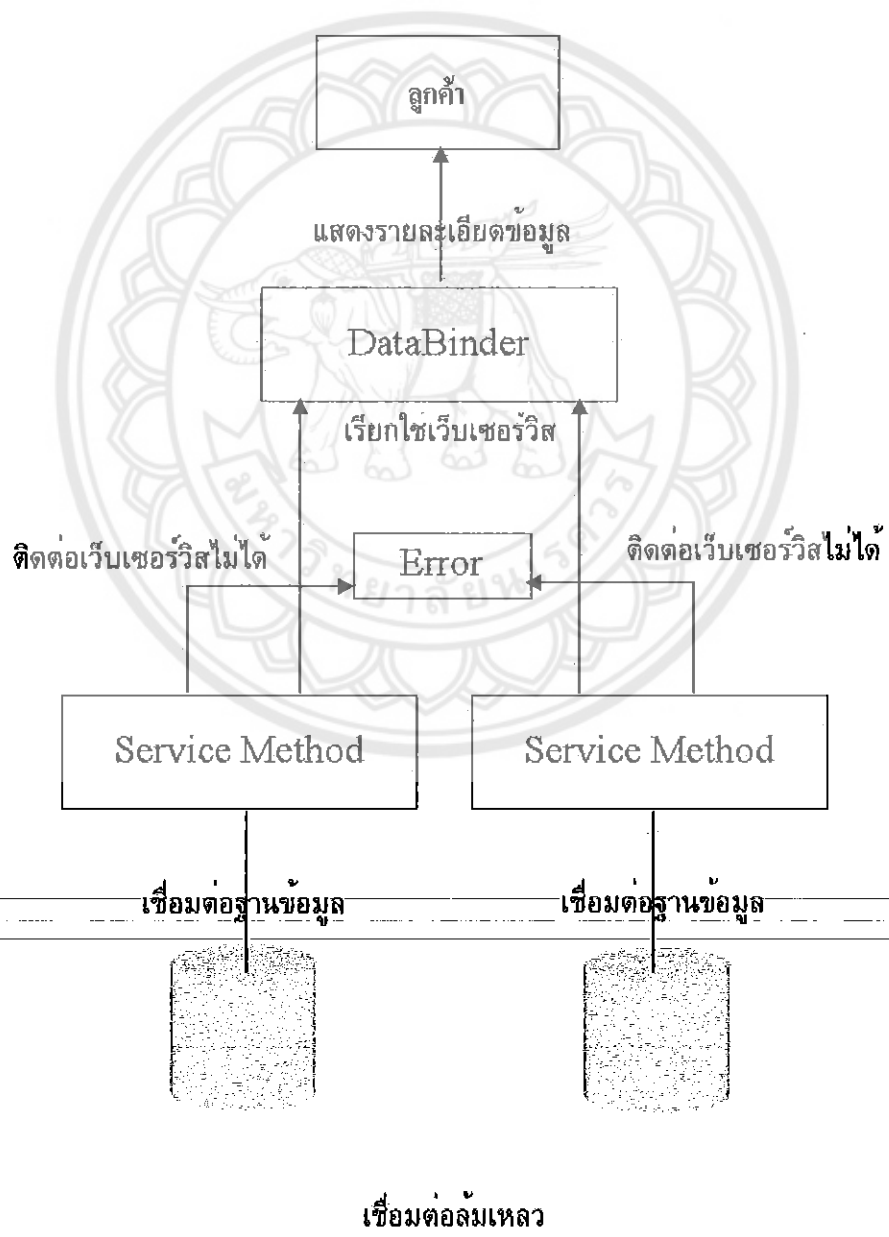
ร/ร.
92244
2550



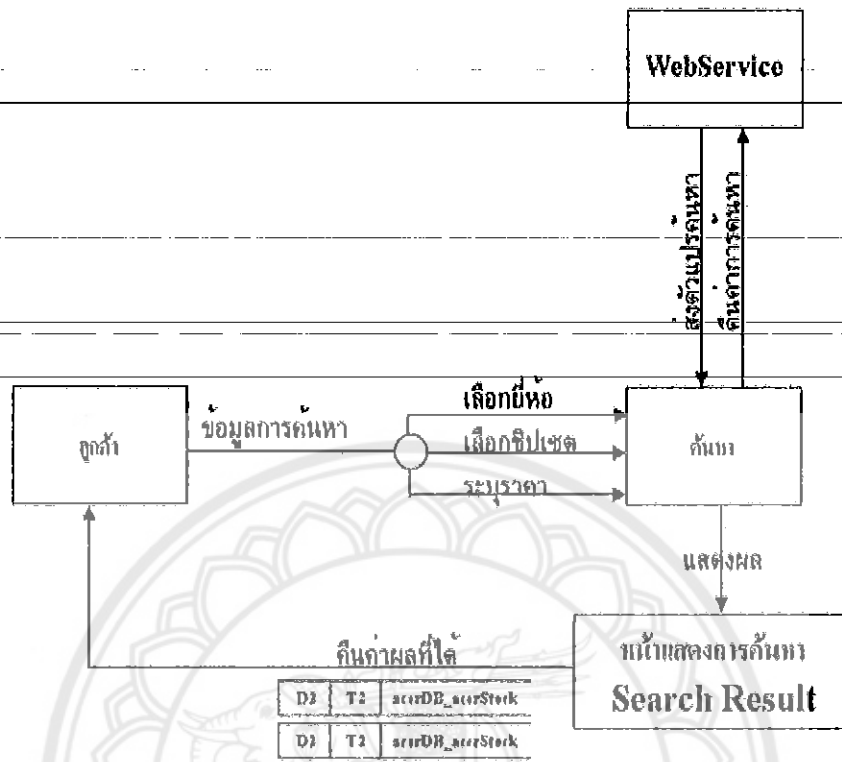
รูปที่ 3.2 แสดง Data Flow Diagram Level 0



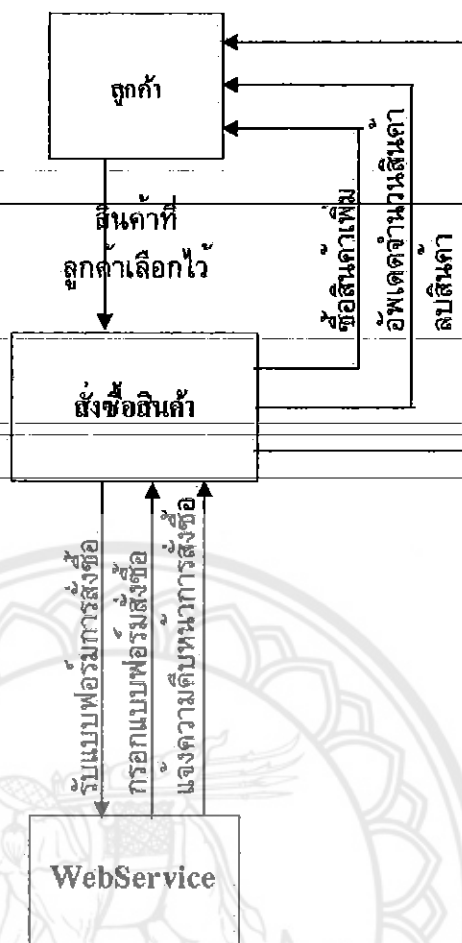
D1	T2	ProviderDB	Provider_user
----	----	------------	---------------



รูปที่ 3.3 แสดง Data Flow Diagram Level 1 ของ Process 1



รูปที่ 3.4 แสดง Data Flow Diagram Level 1 ของ Process 2



รูปที่ 3.5 แสดง Data Flow Diagram Level 1 ของ Process 3

บทที่ 4

การทดลองและผลการทดลอง

ในบทนี้จะกล่าวถึงการให้บริการซื้อขายสินค้าผ่านทางเว็บกลางโดยที่ลูกค้าจะต้องทำการเลือกซื้อสินค้าจากร้านค้าที่เป็นสมาชิกกับเว็บกลาง ในที่นี้ได้จำลองร้านค้าที่เป็นสมาชิกของเว็บกลางจำนวนสองร้านค้า โดยจะแบ่งขั้นตอนของกระบวนการซื้อขายสินค้าออกเป็น 2 ส่วนด้วยกัน คือ

1. ส่วนของลูกค้า

ในส่วนนี้ลูกค้าต้องทำการเลือกซื้อสินค้าและต้องกรอกรายละเอียดข้อมูลของลูกค้าเพื่อซื้อสินค้าและชำระเงินค่าสินค้า

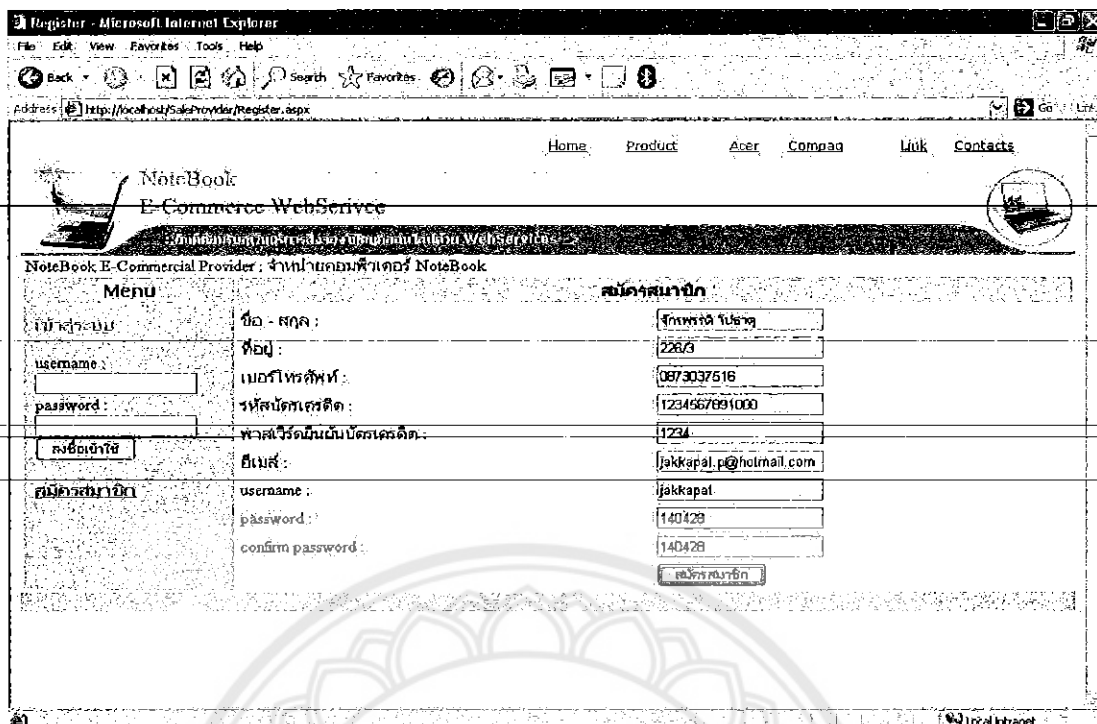
2. ส่วนต่างๆที่เกี่ยวข้องกับการซื้อขายสินค้า

ในส่วนนี้จะเป็นการประสานงานของระบบเว็บกลางกับร้านค้าโดยเป็นการแจ้งข้อมูลที่มีการซื้อขายสินค้าที่เกิดขึ้น เพื่อร้านค้าจะได้จัดเตรียมสินค้าเพื่อในการส่งสินค้าให้กับผู้ซื้อในเวลาต่อไป

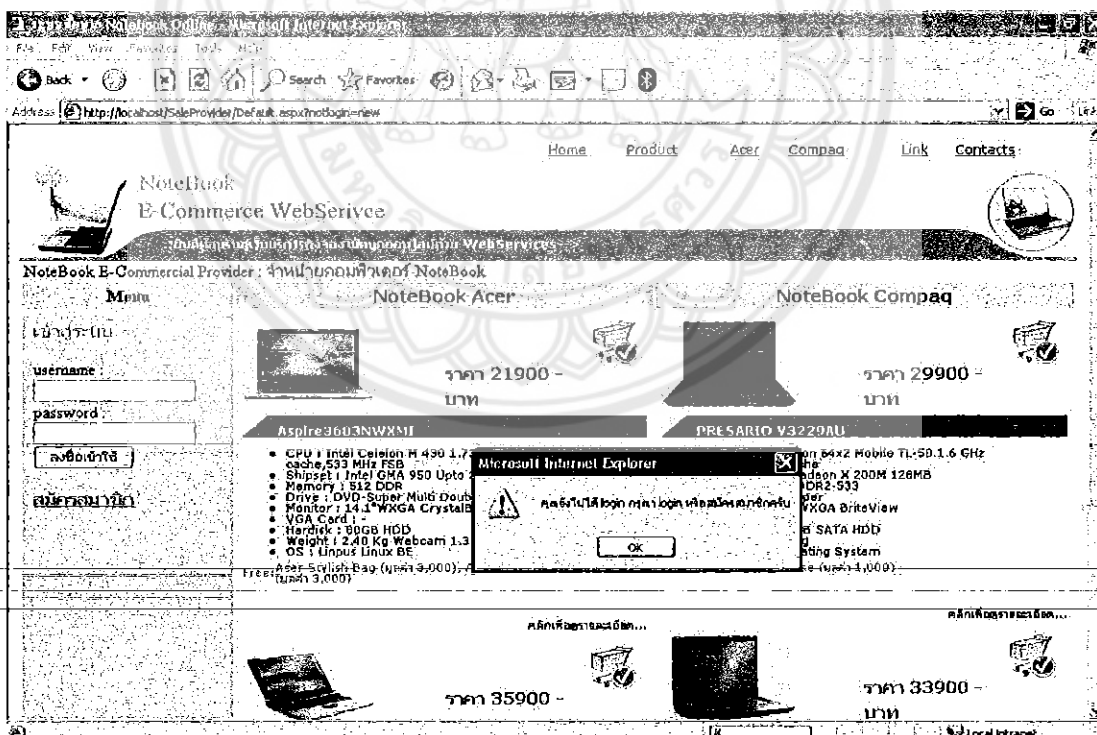
4.1 ส่วนของลูกค้า

เมื่อลูกค้าได้เข้ามายังเว็บกลางเพื่อเลือกซื้อสินค้า โดยลูกค้าสามารถที่จะเลือกสินค้าได้ตามประเภทสินค้าที่ได้มีการแสดงโชว์ไว้ หรือลูกค้าสามารถค้นหาตามรายการสินค้าที่ต้องการได้

จากรูปที่ 4.1 จะเห็นได้ว่าจะแสดงสินค้าทั้งสองร้านค้า โดยจะแบ่งแยกยี่ห้อสินค้าโดยเฉพาะซึ่งร้านค้าที่ 1 จะแสดงสินค้าโน้ตบุ๊กยี่ห้อ Acer แล้วร้านค้าที่ 2 จะแสดงรายการโน้ตบุ๊กยี่ห้อ Compaq ซึ่งทั้งสองร้านค้าจะแสดงรายละเอียดของโน้ตบุ๊กว่ามีส่วนประกอบอะไรบ้าง เพื่อให้เป็นทางเลือกแก่ลูกค้าในการตัดสินใจเลือกซื้อสินค้าด้วย

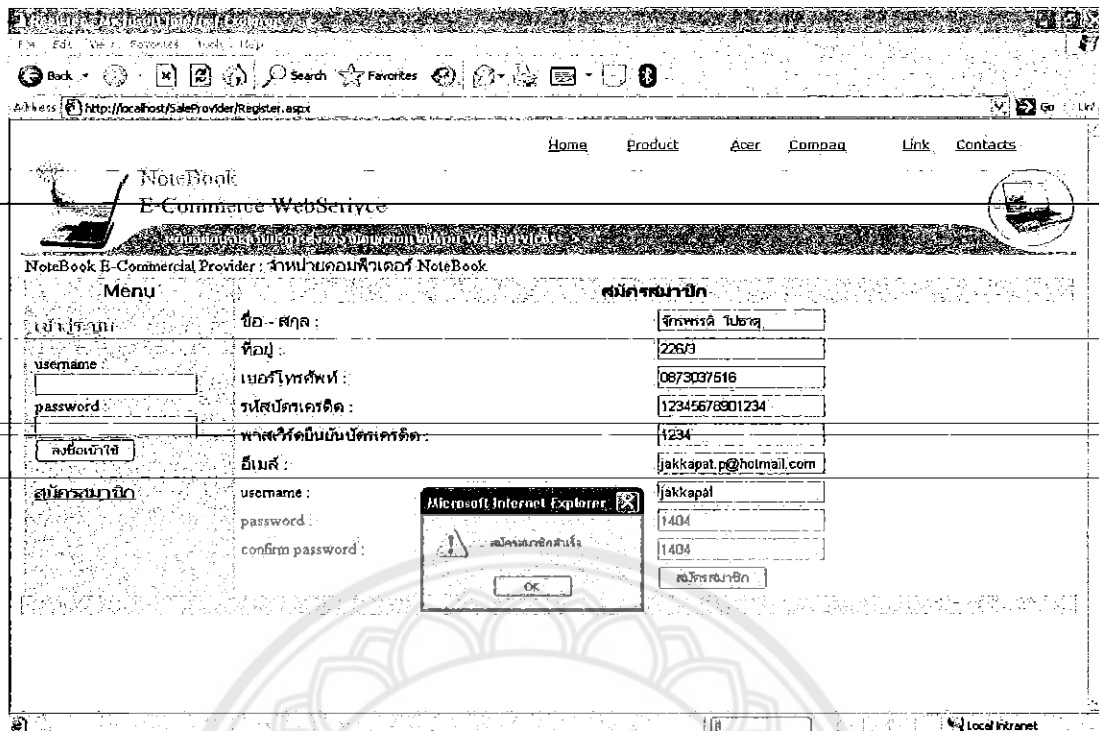


รูปที่ 4.1 การสมัครเป็นสมาชิกเว็บกลาง

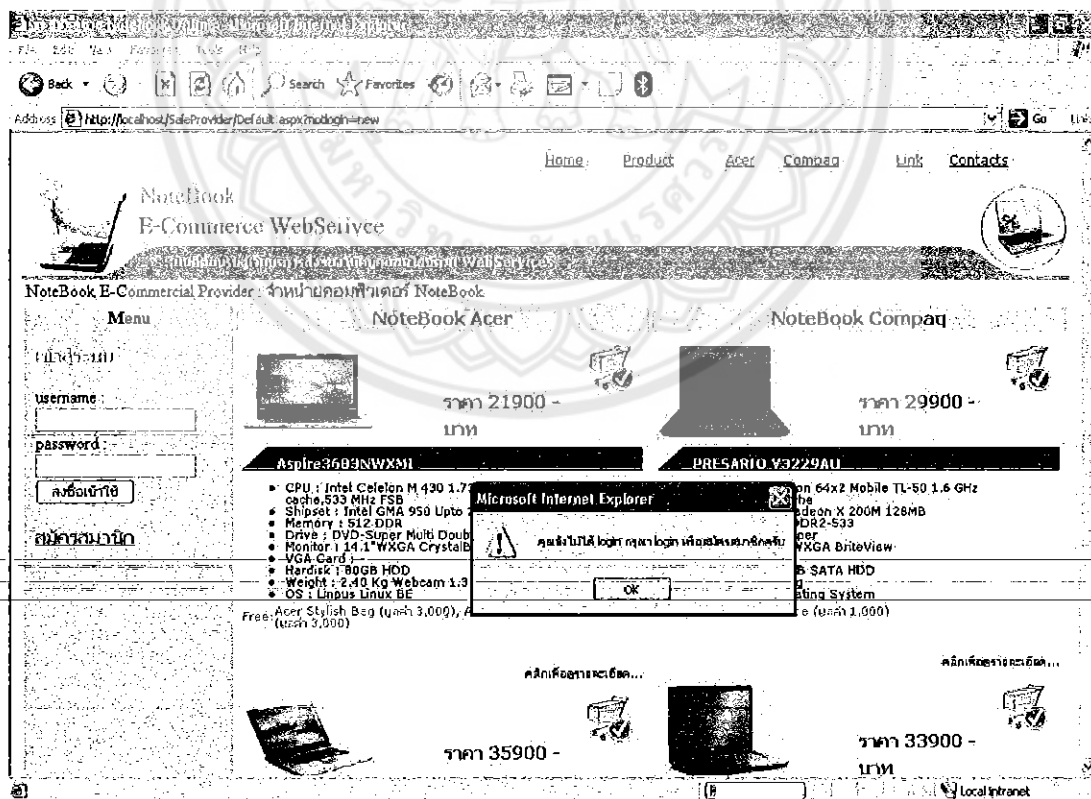


รูปที่ 4.2 กรณีที่ยังไม่ได้สมัครเป็นสมาชิกเว็บกลาง

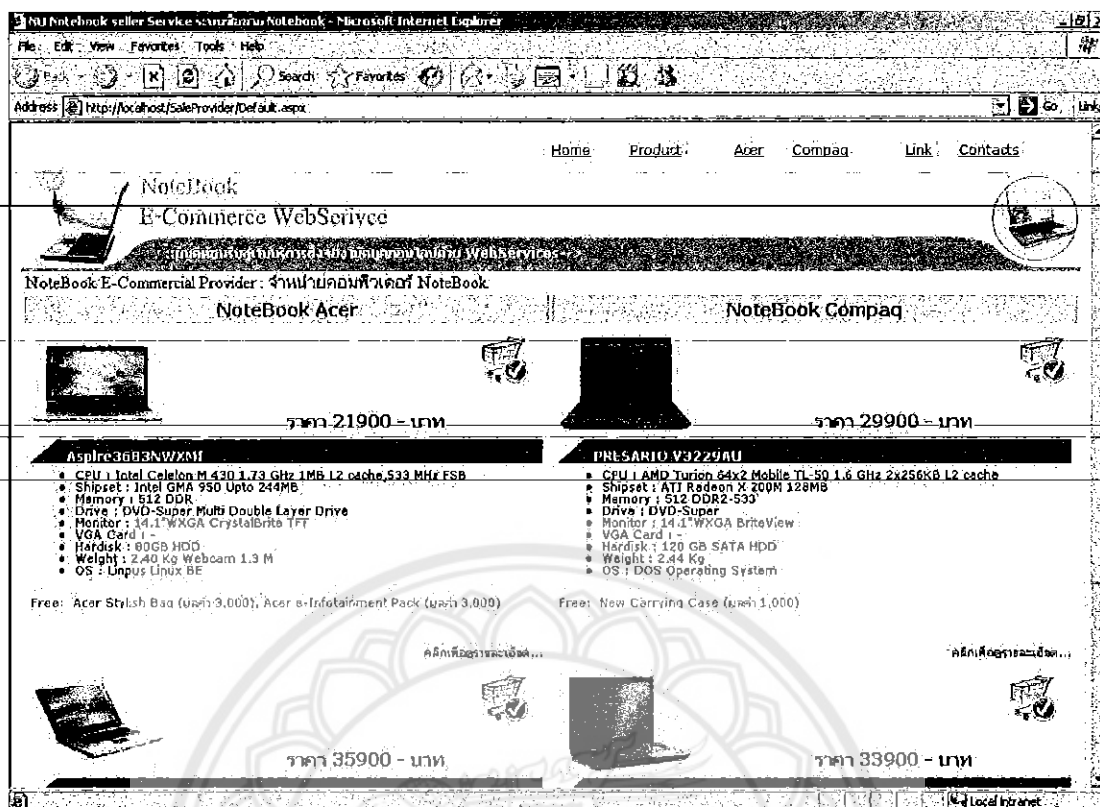
กรณีที่ผู้ใช้บริการเข้ามาใช้บริการแต่ยังไม่สมัครเป็นสมาชิกผู้ใช้บริการนั้นจะไม่สามารถที่จะเข้าไปเลือกซื้อสินค้าได้ แต่ผู้ใช้บริการสามารถที่จะเข้ามาดูได้เฉพาะรายละเอียดของสินค้าเท่านั้น



รูปที่ 4.3 เป็นการแสดงถึงการสมัครเป็นสมาชิกเว็บกลางได้สมบูรณ์



รูปที่ 4.4 เป็นการแสดงถึงสมาชิกไม่ได้ทำการล็อกอินเพื่อเข้าสู่ระบบ



รูปที่ 4.5 การเลือกซื้อสินค้าตามรายการสินค้า

จากรูปที่ 4.5 จะเห็นได้ว่าจะแสดงสินค้าทั้งสองร้านค้า โดยจะแบ่งแยกยี่ห้อสินค้าโดยเฉพาะซึ่งร้านค้าที่ 1 จะแสดงสินค้าโน้ตบุ๊กยี่ห้อ Acer แล้วร้านค้าที่ 2 จะแสดงรายการโน้ตบุ๊กยี่ห้อ Compaq ซึ่งทั้งสองร้านค้าจะแสดงรายละเอียดของโน้ตบุ๊กว่ามีส่วนประกอบอะไรบ้าง เพื่อให้เป็นทางเลือกแก่ลูกค้าในการตัดสินใจเลือกซื้อสินค้าด้วย

ในส่วนของเว็บกลาง ผู้ที่เข้ามาใช้บริการสามารถเข้ามาหาข้อมูลของสินค้าหรือจะค้นหา (Search) รายการของสินค้าและข่าวสารหรือข้อมูลต่างๆ ได้

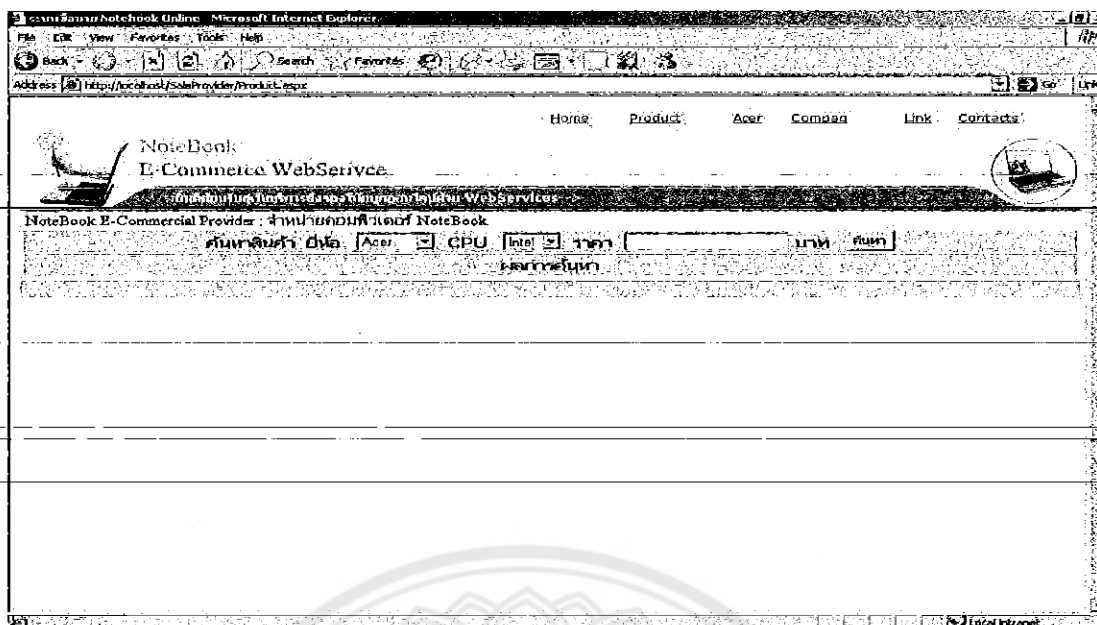
- **Link Product** จะแสดงส่วนของการค้นหาสินค้า (Product)

- **Link Acer** จะแสดงสินค้าโน้ตบุ๊กเฉพาะยี่ห้อ Acer โดยคอลลัมน์ซ้ายจะแสดง CPU ของ Intel ส่วน คอลลัมน์ขวาจะแสดง CPU ของ AMD

- **Link Compaq** จะแสดงสินค้าโน้ตบุ๊กเฉพาะยี่ห้อ Compaq โดยคอลลัมน์ซ้ายจะแสดง CPU ของ Intel ส่วน คอลลัมน์ขวาจะแสดง CPU ของ AMD

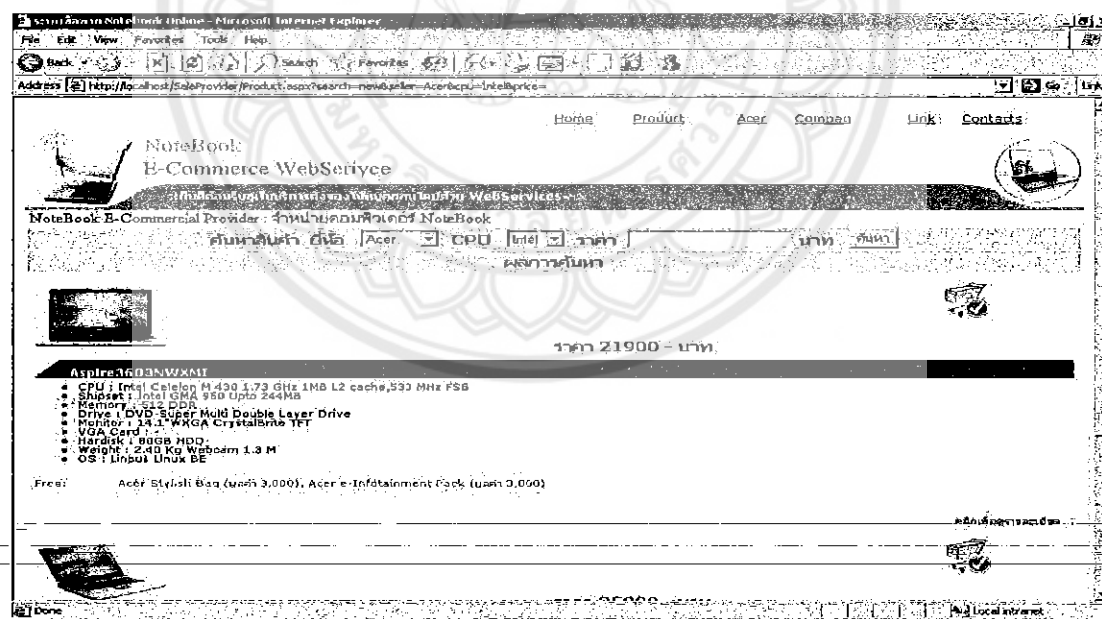
- **Link** จะแสดงส่วนของ Link ที่มีความเกี่ยวข้องกับข้อมูลหรือเว็บที่เว็บส่วนกลางต้องการแสดงให้แก่ผู้ใช้บริการ

- **Link Contact** จะเป็นการติดต่อผู้บริการ ในลักษณะต่างๆ เช่น ส่งเมลล์ หรือ เป็นการแนะนำบริษัท



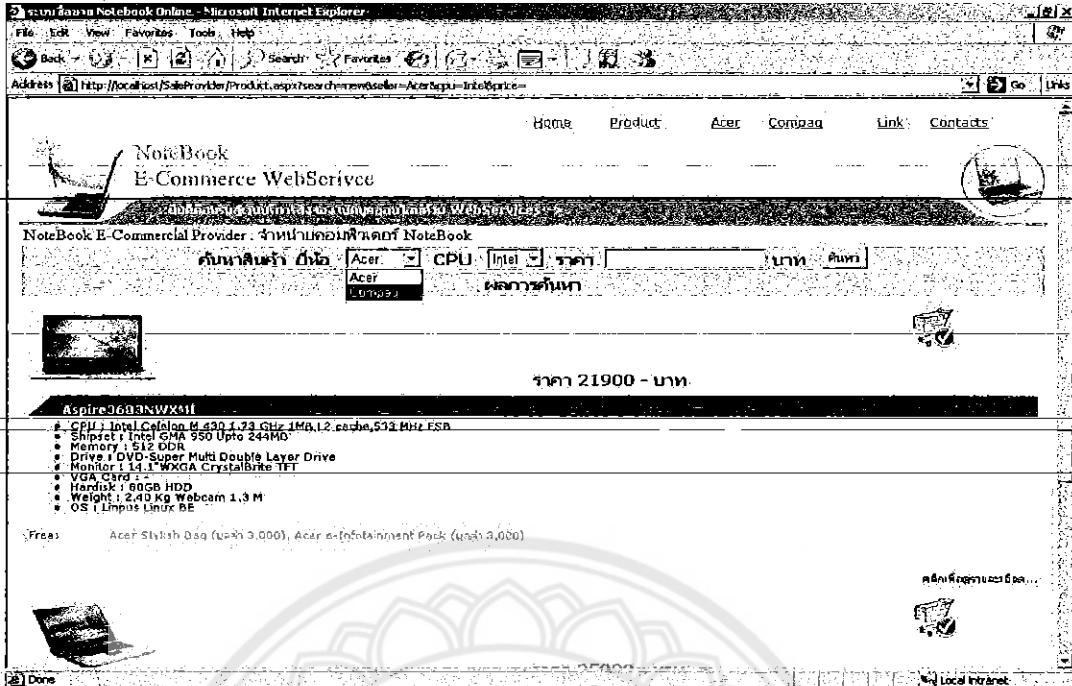
รูปที่ 4.6 เมื่อเลือก Link Product

จากรูปที่ 4.6 เมื่อลูกค้าเลือก Link Product ก็จะแสดงการค้นหาสินค้า โดยให้ลูกค้าเลือกตามรายการซึ่งประกอบด้วย ชื่่อ, หน่วยประมวลผล (CPU), ราคาที่ต้องการในวงเงินที่ลูกค้ามีอยู่

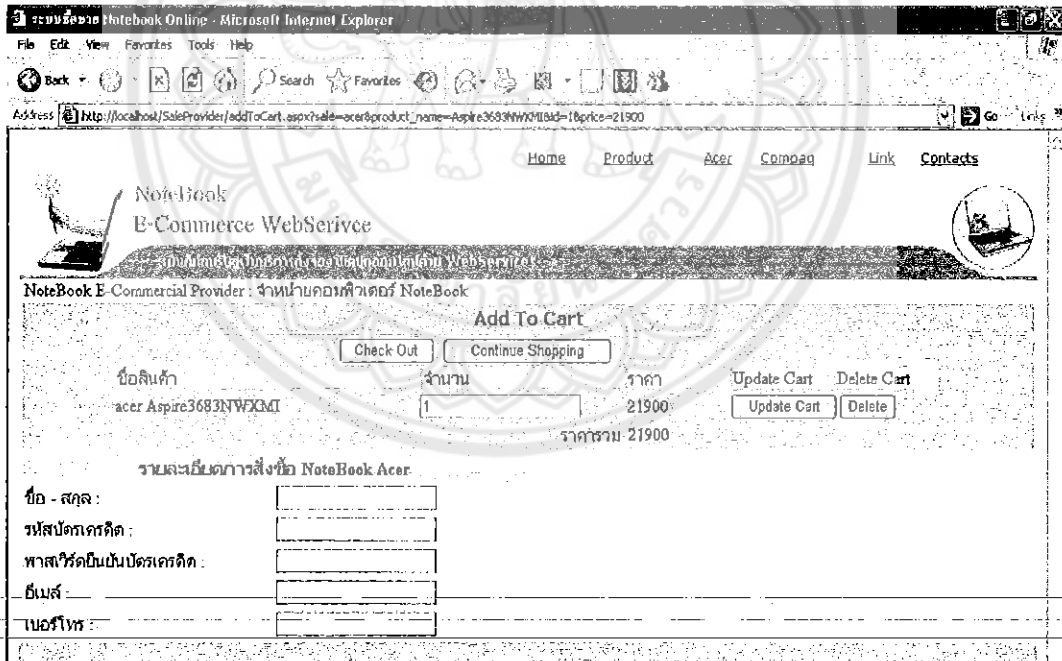


รูปที่ 4.7 เมื่อเลือกยี่ห้อ Acer

จากรูปที่ 4.7 เมื่อเลือกค้นหาสินค้า โดยเลือกยี่ห้อ Acer ผลการค้นหา ก็จะแสดงเฉพาะสินค้ายี่ห้อ Acer นั้น ส่วนหน่วยประมวลผล (CPU) และราคา ก็จะแสดงข้อมูลตามที่ลูกค้าต้องการเลือกซื้อสินค้า



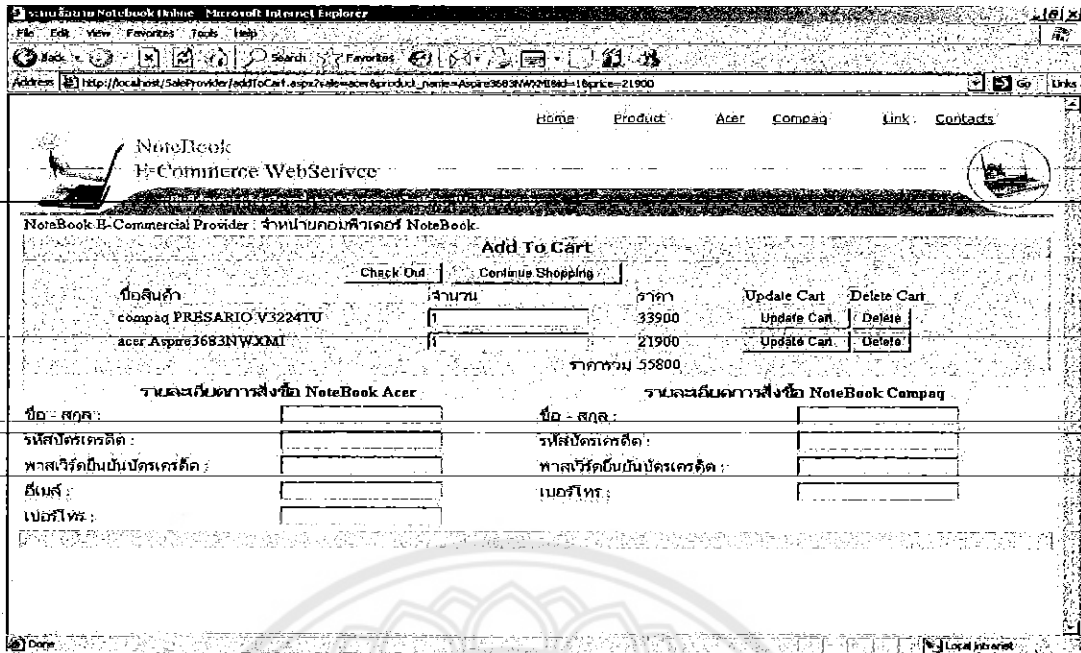
รูปที่ 4.8 เมื่อต้องการเลือกยี่ห้อ Compaq



รูปที่ 4.9 รายการของการซื้อสินค้าร้านค้าที่ 1

ลูกค้าทำการเลือกดูสินค้า ก่อนจะตัดสินใจซื้อสินค้า เมื่อลูกค้าตัดสินใจซื้อสินค้าชิ้นนี้แล้วให้ทำการกดปุ่ม “หยิบใส่รถเข็น” ในที่นี้ลูกค้าตัดสินใจเลือกซื้อสินค้าดังนี้

1. acer Aspire3683NWXMI จากร้านค้าที่ 1 จำนวน 1 เครื่อง ดังรูปที่ 4.9



รูปที่ 4.10 รายการเลือกซื้อสินค้าทั้งสองร้านค้า

รูปที่ 4.10 เป็นรายการเลือกซื้อสินค้าที่ลูกค้าต้องการเลือกซื้อสินค้าทั้งสองยี่ห้อจากทั้งสองร้านค้า

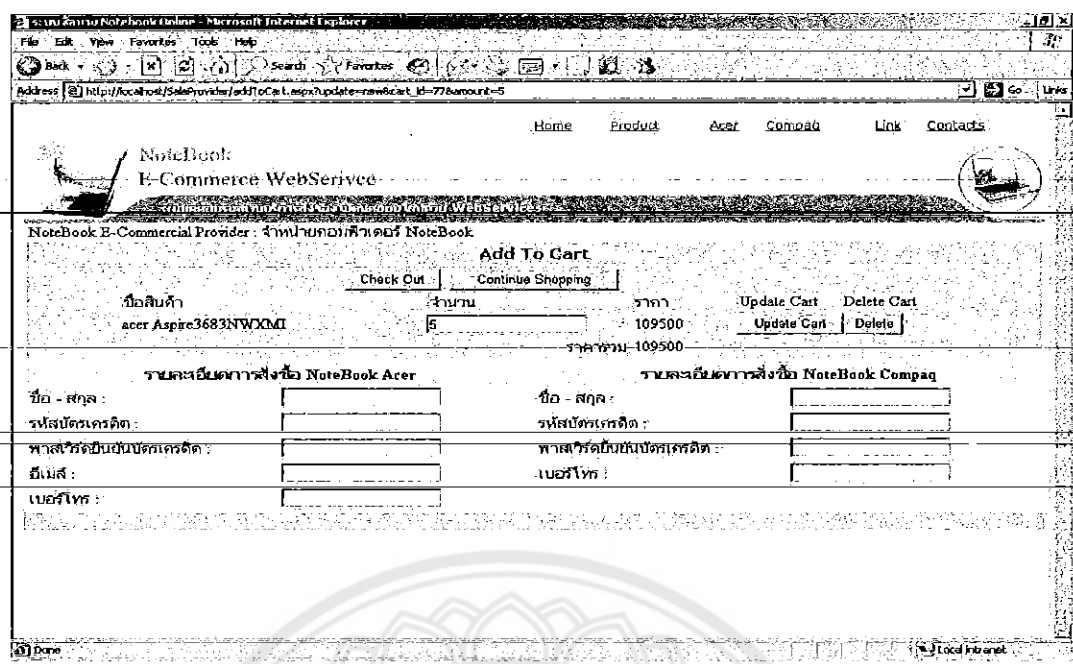
ลูกค้าสามารถที่จะเปลี่ยนแปลงรายการสินค้าได้โดยการกดปุ่มต่อไปนี้

กดปุ่ม "Continue Shopping" คือ สามารถที่จะเพิ่มรายการสินค้าที่ลูกค้าสนใจได้

กดปุ่ม "Update Cart" คือ อัปเดตราคาตามจำนวนชิ้นของสินค้าที่เลือกซื้อเพิ่มขึ้น เช่น สินค้าราคาชิ้นละ 25,000 บาท ซื้อ 2 ชิ้น ราคารวม 50,000 บาท ให้ทำการกดปุ่มนี้ทุกครั้งเมื่อทำการเพิ่มหรือลดจำนวนสินค้าที่เลือกซื้อ

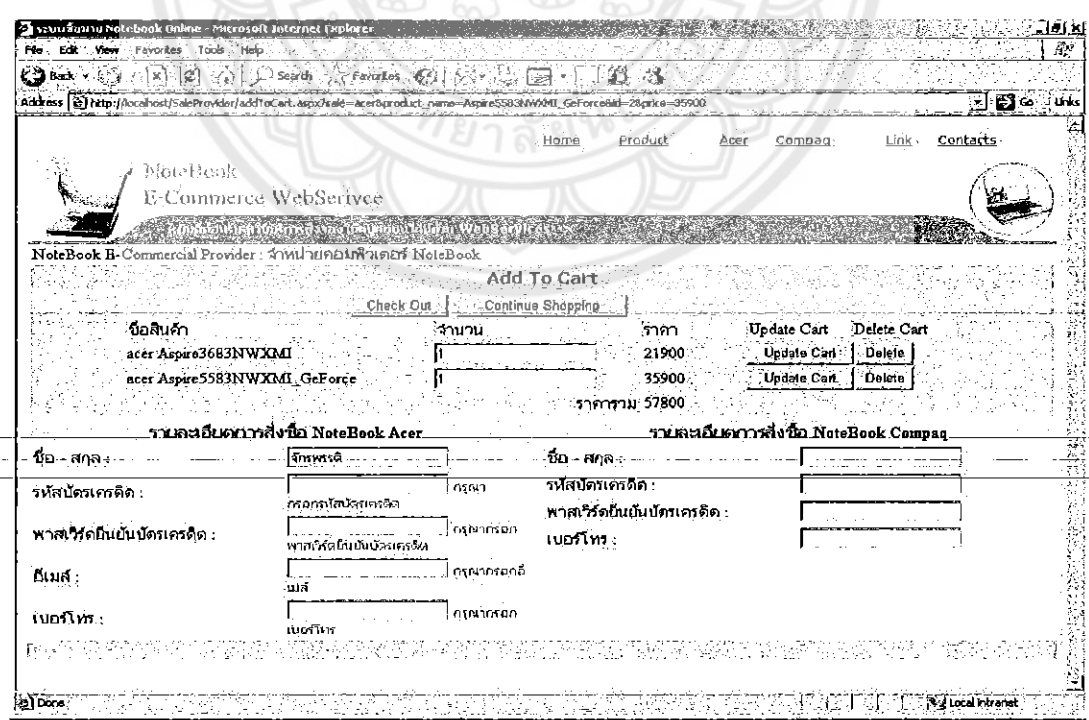
กดปุ่ม "Delete" คือ ลบสินค้าที่ลูกค้าไม่ต้องการซื้อแล้วให้สินค้าชนิดนี้ออกไปจากรายการเลือกซื้อสินค้า

กดปุ่ม "Check Out" คือ เป็นการยืนยันที่ลูกค้าจะเลือกซื้อสินค้าต่างๆ ภายในรายการที่ลูกค้าได้เลือกซื้อไว้ แต่ก่อนทำการกดปุ่มนี้ลูกค้าจะต้องกรอกรายละเอียดข้อมูลของลูกค้าลงในแบบฟอร์มการสั่งซื้อสินค้าเพื่อทางร้านค้าจะได้จัดส่งสินค้าที่ลูกค้าได้ทำการสั่งซื้อและลูกค้าจะต้องตรวจสอบรายการที่ลูกค้าเลือกซื้อให้ละเอียด เพื่อสิทธิและประโยชน์ของลูกค้า



รูปที่ 4.11 รายการเพิ่มจำนวนสินค้าที่เลือกซื้อ

เมื่อลูกค้าเลือกสินค้าตามที่ต้องการแล้ว ก็มาถึงส่วนการกรอกแบบฟอร์มการสั่งซื้อ ถ้าลูกค้ากรอกรายละเอียดข้อมูลส่วนตัวของลูกค้าไม่สมบูรณ์ แล้วกดปุ่ม “Check Out” เว็บกลางจะทำหน้าที่ฟ้องว่าเกิดความผิดพลาดของการกรอกข้อมูลดังรูป



รูปที่ 4.12 เกิดความผิดพลาดในการกรอกแบบฟอร์มการสั่งซื้อสินค้า

NoteBook E-Commerce WebService

NoteBook E-Commercial Provider: ร้านขายคอมพิวเตอร์ NoteBook

Add To Cart

Check Out | Continue Shopping

ชื่อสินค้า	จำนวน	ราคา	Update Cart	Delete Cart
		ราคา		
ราคารวม 0				

รายละเอียดการสั่งซื้อ NoteBook Acer

รายละเอียดการสั่งซื้อ NoteBook Compaq

ชื่อ - สกุล : [กรกฎาดิ นามราช] ชื่อ - สกุล : [กรกฎาดิ นามราช]

รหัสบัตรเครดิิต : [123456789] รหัสบัตรเครดิิต : [กรกฎาดิ นามราช]

พาสเวิร์ดยืนยันบัตรเครดิิต : [1234] พาสเวิร์ดยืนยันบัตรเครดิิต : [กรกฎาดิ นามราช]

อีเมล : [jkkkapsl.p@hotmail.com] เบอร์โทร : [กรกฎาดิ นามราช]

เบอร์โทร : [082-3097516] เบอร์โทร : [กรกฎาดิ นามราช]

Complete ระบบจัดการข้อมูลของท่านแล้ว

รูปที่ 4.13 เมื่อกรอกรายละเอียดได้สมบูรณ์ข้อมูลก็จะถูกจัดเก็บ

เมื่อลูกค้าซื้อสินค้าทั้งสองร้านค้าแต่ไม่ได้กรอกแบบฟอร์มการสั่งซื้อสินค้าของร้านค้าใดร้านหนึ่ง เว็บกลางก็จะฟ้องว่าลูกค้ายังไม่ได้กรอกแบบฟอร์มการสั่งซื้อสินค้าของอีกร้านหนึ่ง ดังรูป

NoteBook E-Commerce WebService

NoteBook E-Commercial Provider: ร้านขายคอมพิวเตอร์ NoteBook

Add To Cart

Check Out | Continue Shopping

ชื่อสินค้า	จำนวน	ราคา	Update Cart	Delete Cart
acer Aspire3683NWXMI	1	21900	Update Cart	Delete
acer Aspire5583NWXMI GeForce	1	35900	Update Cart	Delete
ราคารวม 57800				

รายละเอียดการสั่งซื้อ NoteBook Acer

รายละเอียดการสั่งซื้อ NoteBook Compaq

ชื่อ - สกุล : [กรกฎาดิ นามราช] ชื่อ - สกุล : [กรกฎาดิ นามราช]

รหัสบัตรเครดิิต : [กรกฎาดิ นามราช] รหัสบัตรเครดิิต : [กรกฎาดิ นามราช]

พาสเวิร์ดยืนยันบัตรเครดิิต : [กรกฎาดิ นามราช] พาสเวิร์ดยืนยันบัตรเครดิิต : [กรกฎาดิ นามราช]

อีเมล : [กรกฎาดิ นามราช] เบอร์โทร : [กรกฎาดิ นามราช]

เบอร์โทร : [กรกฎาดิ นามราช] เบอร์โทร : [กรกฎาดิ นามราช]

รูปที่ 4.14 รายการเลือกซื้อสินค้าทั้งสองร้านค้าแต่ไม่ได้กรอกแบบฟอร์มให้ครบทั้งสอง

4.2 ส่วนต่างๆที่เกี่ยวข้องกับการซื้อขายสินค้า

ในส่วนนี้จะเริ่มต้นจากที่ลูกค้าได้ทำการกดปุ่ม “Check Out” เพื่อยืนยันในการสั่งซื้อสินค้า ซึ่งจะเห็นได้ในหน้าที่ได้แสดงรายละเอียดแบบฟอร์มการสั่งซื้อสินค้า ซึ่งจะมีการอัปเดตฐานข้อมูลเกิดขึ้น

โดยข้อมูลดังกล่าวจะแสดงถึงว่าสินค้านี้ได้มีการสั่งซื้อแล้ว

order_id	order_amt	order_amount	order_custome...	order_custome...	order_custome...	order_custome...
2	1	5	วิเศษดี โปรม	123456789	1234	Wkapat.p@hot...
ALL	ALL	ALL	ALL	ALL	ALL	ALL

รูปที่ 4.15 ฐานข้อมูลของการยืนยันการสั่งซื้อสินค้า

บทที่ 5

สรุปผลการทดลอง

5.1 สรุปผลการทดลอง

ในการทำปฏิยานิพนธ์นี้ เรื่อง “เว็บเซอร์วิสสำหรับร้านค้าออนไลน์” ได้สร้าง โปรแกรมจำลอง การทำธุรกรรมการค้าบนอินเทอร์เน็ต โดยสร้างขึ้นมาเพื่ออำนวยความสะดวกให้กับผู้ใช้บริการที่ต้องการ ซื้อและขายสินค้า และยังช่วยเหลือร้านค้าต่างๆ ในด้านการทำธุรกรรมการค้า โดยที่ร้านค้าสามารถมีส่วนร่วม ในธุรกรรมการค้าบนอินเทอร์เน็ต เพื่อที่ร้านค้าจะได้ขายสินค้าได้เพิ่มมากขึ้นอีกช่องทางหนึ่งของการ ค้าขายในปัจจุบัน ซึ่งระบบที่สร้างขึ้นมาจะมีการอัปเดตข้อมูลเป็นระยะๆ

ดังนั้นในปัจจุบันที่มีการแข่งขันด้านการค้าที่สูง และร้านค้าต้องการที่จะตอบสนองความต้องการ แก่ลูกค้าให้เร็วที่สุดเพื่อที่จะขายสินค้าให้ได้ ซึ่งจากการที่ได้ทำโครงการนี้ ทำให้ทราบถึงรายละเอียดว่า ระบบที่ได้ใช้กับโครงการนี้ได้สร้างความสะดวกให้แก่ลูกค้าที่มาซื้อสินค้าและยังช่วยเหลือร้านค้าต่างๆ ด้วย

5.2 ข้อเสนอแนะและการแก้ไขปัญหา

5.2.1 ในการออกแบบ ความปลอดภัยในระบบธุรกิจบนอินเทอร์เน็ตทั้งในด้านข้อมูลและการเงิน นั้นก็มีการป้องกัน ไม่ได้โดยสมบูรณ์ เราจึงต้องหาความรู้ตลอดเวลา เพื่อรับมือกับเหตุการณ์ที่ไม่อาจ คาดฝันได้ทันการณ์

5.2.2 ในการสร้างและออกแบบเว็บเซอร์วิส ซึ่งเป็นตัวกลางในการติดต่อของสองแอปพลิเคชันนั้น ซึ่งยังเป็นเรื่องใหม่สำหรับประเทศไทย ดังนั้นจึงต้องมีการศึกษาและค้นคว้าหาความรู้มาเพิ่มเติม เพื่อ นำมาปรับใช้งานและทำให้เกิดประโยชน์สูงสุดต่องานที่จะนำมาใช้ในภายภาคหน้า

5.2.3 ภาษาที่ใช้ในสร้างเว็บไซด์แต่ละระบบคือ ASP ซึ่งมีข้อบกพร่องในด้านความปลอดภัย ใน กรณีที่เกิดจาก SQL Poisoning & Injections ซึ่งอาจทำให้ผู้บุกรุกสามารถเพิ่มเติมและเปลี่ยนแปลง SQL statement เพื่อให้ทำงานอื่นได้ ดังนั้นต้องรับมือกับผู้ใช้ที่ไม่ประสงค์ดีที่จะมาทำลายระบบ

5.3 แนวทางในการพัฒนาต่อในอนาคต

5.3.1 สามารถเพิ่มและเปลี่ยนแปลง URL ของเว็บบริษัทขายสินค้าประเภท Notebook ได้

เอกสารอ้างอิง

- [1] นายศุภชัย สมพานิช. คู่มือ ASP.NET 2.0 ฉบับสมบูรณ์. นนทบุรี : สำนักพิมพ์ ไอดีซี. 2549.
- [2] นายสุรสิทธิ์ คิวประสพศักดิ์. อินเทอร์เน็ต Visual Basic.NET ฉบับสมบูรณ์. กรุงเทพมหานคร : สำนักพิมพ์ บริษัท โปรวิชั่น จำกัด. 2546.
- [3] นายทวีชัย หงส์สุมาลัย, นายสงวนชัย สุวรรณชีวะศิริ. อินเทอร์เน็ต ASP.NET ฉบับสมบูรณ์. กรุงเทพมหานคร : สำนักพิมพ์ บริษัท โปรวิชั่น จำกัด. 2546.
- [4] นายกฤษณ์ วรสุวรรณรักษ์, นายกฤษดา วัฒนาจารุพงศ์, นายชลธรรม ประภากรวรรกุล. “การพัฒนาแอปพลิเคชันด้วยเว็บเซอร์วิส” วิทยานิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์. สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง 2548.
- [5] นายณัฐพล เขียวประเสริฐ, นายณัฐพล เอื้อพูนวิริยะ. “เว็บเซอร์วิสเพื่อการบริหารจัดการบริษัททัวร์” วิทยานิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์. สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง 2548.
- [6] นางสาวฐปพร น้อยเมือง, นายณัฐ เบ็ญจพรกุลพงศ์, นางสาวนุจรศ แดงไทย. “ระบบขายสินค้าคอมพิวเตอร์บนเว็บเซอร์วิส” วิทยานิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์. สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง 2548.
- [7] ซีเอ็มเอสไทยแลนด์คอกคอม. “Web Services” [Online]. Available : <http://www.cmsthailand.com/ws/index.php>. 2003.
- [8] สำนักบริการคอมพิวเตอร์. “ธุรกิจผ่านสื่ออิเล็กทรอนิกส์” [Online]. Available : <http://www.ku.ac.th/e-magazine/march45/it/web3.html>. 2545.
- [9] Microsoft. “.NET Framework Developer Center” [Online]. Available : <http://msdn2.microsoft.com/en-us/netframework/default.aspx>. 2007.

ภาคผนวก ก.

เว็บเซอร์วิสที่เปิดให้บริการและการติดตั้งโปรแกรม

ในที่นี้จะกล่าวถึงเว็บเซอร์วิสที่ถูกเปิดบริการทั้งหมดในขั้นตอนของการซื้อขายสินค้าไม้ดื่ก
แบ่งเป็นส่วนหลักๆด้วยกัน คือ

1. ส่วนของร้านค้าย่อย

ในส่วนนี้ได้มีการจำลองร้านค้าย่อยขึ้นมา 2 ร้านด้วยกัน แต่ได้มีการเปิดให้บริการที่เหมือนกันจึงได้
อธิบายการเปิดบริการในร้านค้าย่อยที่ 1 เพียงร้านเดียว ซึ่งได้เปิดบริการต่างๆดังต่อไปนี้



รูปที่ 1.1 การให้บริการของเว็บเซอร์วิสในส่วน of ร้านค้าย่อยที่ 1

1.1 บริการ SearchAcerFromStock เป็นการเปิดบริการเพื่อให้เว็บกลางสามารถสืบค้นหาข้อมูลได้
ตามรายการสินค้าที่ต้องการ เพราะว่าตัวเว็บกลางจะไม่มีข้อมูลของสินค้า

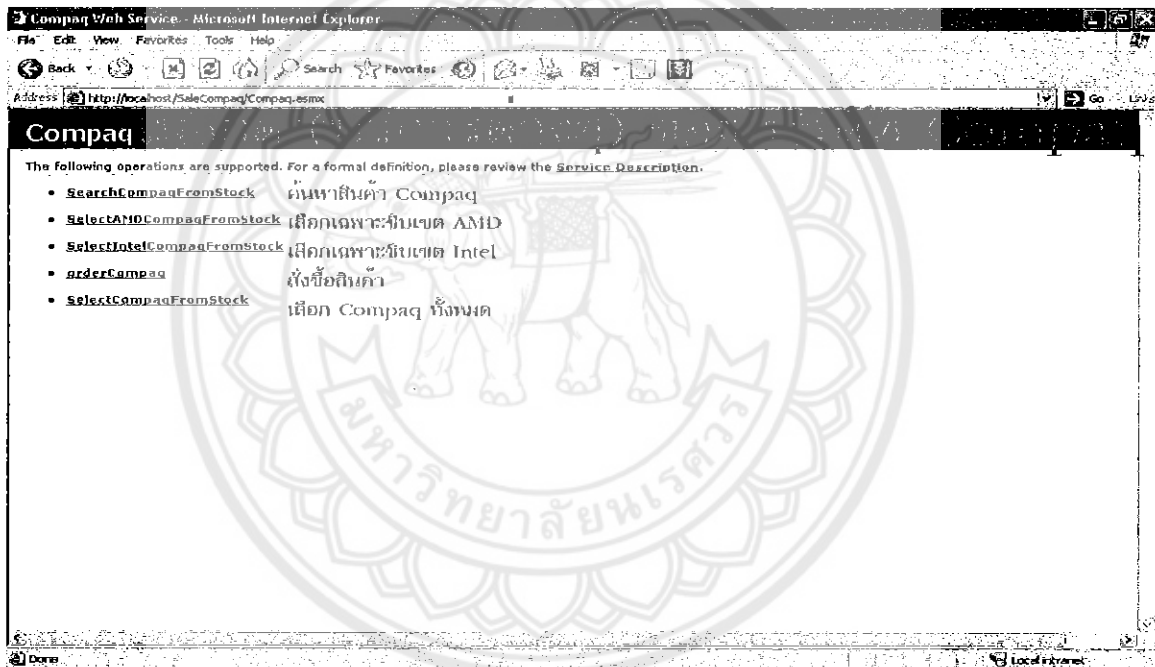
1.2 บริการ SelectAMDFromStock เป็นการเปิดบริการเพื่อให้เห็นข้อมูลรายการสินค้า
เฉพาะหน่วยประมวลผล (CPU) AMD

1.3 บริการ **SelectIntelAcerFromStock** เป็นการเปิดบริการเพื่อให้เห็นข้อมูลรายการสินค้าเฉพาะหน่วยประมวลผล (CPU) Intel

1.4 บริการ **SelectAcerFromStock** เป็นการเปิดบริการเพื่อให้เห็นข้อมูลรายการสินค้าโน้ตบุ๊กเฉพาะยี่ห้อ Acer โดยระบบจะทำการแสดงรายละเอียดของสินค้านั้น ประกอบด้วย

1. ชื่อและยี่ห้อสินค้า
2. ราคาของสินค้าต่อ 1 หน่วย
3. รายละเอียดของสินค้า
4. ชื่อไฟล์รูปภาพของสินค้า

1.5 บริการ **OrderAcer** เป็นการเปิดบริการเพื่อให้เห็นข้อมูลรายการการสั่งซื้อสินค้าสินค้า



รูปที่ 1.2 การให้บริการของเว็บเซอร์วิสในส่วนของร้านค้าย่อยที่ 2

SearchCompaqFromStock

Test นี้เป็นตัวทดสอบเว็บเซอร์วิสว่า ถ้าเรารับค่า Soap จากหน้าเว็บมา แล้วจะส่งค่าอะไรกลับไป
แสดงผล

To test the operation using the HTTP POST protocol, click the 'Invoke' button.

Parameter	Value
cpu:	<input type="text"/>
Price:	<input type="text"/>
<input type="button" value="Invoke"/>	

รูปที่ 1.3 การเรียกบริการ SearchCompaqFromStock โดยใส่ค่าตัวแปรต่างๆ

SOAP

The following is a sample SOAP request and response. The placeholders shown need to be replaced with actual values.

```
POST /SaleCompaq/Compaq.asmx HTTP/1.1
```

```
Host: localhost
```

```
Content-Type: text/xml; charset=utf-8
```

```
Content-Length: length
```

```
SOAPAction: "http://tempuri.org/SaleCompaq/Compaq/SearchCompaqFromStock"
```

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
```

```
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
```

```
<soap:Body>
```

```
<SearchCompaqFromStock xmlns="http://tempuri.org/SaleCompaq/Compaq">
```

```
<cpu>string</cpu>
```

```
<Price>string</Price>
```

```
</SearchCompaqFromStock>
```

```
</soap:Body>
```

```

</soap:Envelope>
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <SearchCompaqFromStockResponse xmlns="http://tempuri.org/SaleCompaq/Compaq">
      <SearchCompaqFromStockResult>
        <xsd:schema>schema</xsd:schema>xml</SearchCompaqFromStockResult>
      </SearchCompaqFromStockResponse>
    </soap:Body>
  </soap:Envelope>

```

HTTP POST

```

POST /SaleCompaq/Compaq.asmx/SearchCompaqFromStock HTTP/1.1
Host: localhost
Content-Type: application/x-www-form-urlencoded
Content-Length: length
cpu=string&Price=string
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8"?>
<DataSet xmlns="http://tempuri.org/SaleCompaq/Compaq">
  <schema xmlns="http://www.w3.org/2001/XMLSchema">schema</schema>xml</DataSet>

```

รูปที่ 1.4 เอกสาร SOAP ที่ได้จากกรเรียกบริการดังกล่าว

```

<?xml version="1.0" encoding="utf-8" ?>
- <definitions xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:s="http://www.w3.org/2001/XMLSchema"
  xmlns:s0="http://tempuri.org/SaleCompaq/Compaq"
  xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:tm="http://microsoft.com/wsdl/mime/textMatching/"
  xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
  targetNamespace="http://tempuri.org/SaleCompaq/Compaq"
  xmlns="http://schemas.xmlsoap.org/wsdl/">
- <types>
- <s:schema elementFormDefault="qualified"
  targetNamespace="http://tempuri.org/SaleCompaq/Compaq">
  <s:import namespace="http://www.w3.org/2001/XMLSchema" />
- <s:element name="SelectCompaqFromStock">
  <s:complexType />
  </s:element>
- <s:element name="SelectCompaqFromStockResponse">
- <s:complexType>
- <s:sequence>
- <s:element minOccurs="0" maxOccurs="1" name="SelectCompaqFromStockResult">
- <s:complexType>
- <s:sequence>
  <s:element ref="s:schema" />
- <s:any />
  </s:sequence>
  </s:complexType>
  </s:element>
  </s:sequence>
  </s:complexType>
  </s:element>

```

- <s:element name="SelectIntelCompaqFromStock">

<s:complexType />

</s:element>

- <s:element name="SelectIntelCompaqFromStockResponse">

<s:complexType>

<s:sequence>

- <s:element minOccurs="0" maxOccurs="1" name="SelectIntelCompaqFromStockResult">

<s:complexType>

<s:sequence>

<s:element ref="s:schema" />

<s:any />

</s:sequence>

</s:complexType>

</s:element>

</s:sequence>

</s:complexType>

</s:element>

- <s:element name="SelectAMDCompaqFromStock">

<s:complexType />

</s:element>

- <s:element name="SelectAMDCompaqFromStockResponse">

<s:complexType>

<s:sequence>

- <s:element minOccurs="0" maxOccurs="1" name="SelectAMDCompaqFromStockResult">

<s:complexType>

<s:sequence>

<s:element ref="s:schema" />

<s:any />

</s:sequence>

</s:complexType>

</s:element>

```
</s:sequence>
```

```
</s:complexType>
```

```
</s:element>
```

```
- <s:element name="SearchCompaqFromStock">
```

```
- <s:complexType>
```

```
- <s:sequence>
```

```
<s:element minOccurs="0" maxOccurs="1" name="cpu" type="s:string" />
```

```
<s:element minOccurs="0" maxOccurs="1" name="Price" type="s:string" />
```

```
</s:sequence>
```

```
</s:complexType>
```

```
</s:element>
```

```
- <s:element name="SearchCompaqFromStockResponse">
```

```
- <s:complexType>
```

```
- <s:sequence>
```

```
- <s:element minOccurs="0" maxOccurs="1" name="SearchCompaqFromStockResult">
```

```
- <s:complexType>
```

```
- <s:sequence>
```

```
<s:element ref="s:schema" />
```

```
<s:any />
```

```
</s:sequence>
```

```
</s:complexType>
```

```
</s:element>
```

```
</s:sequence>
```

```
</s:complexType>
```

```
</s:element>
```

```
- <s:element name="orderCompaq">
```

```
- <s:complexType>
```

```
- <s:sequence>
```

```
<s:element minOccurs="1" maxOccurs="1" name="compaq_id" type="s:int" />
```

```
<s:element minOccurs="1" maxOccurs="1" name="compaq_amount" type="s:int" />
```

```
<s:element minOccurs="0" maxOccurs="1" name="customer_name" type="s:string" />
```



```

<s:element minOccurs="0" maxOccurs="1" name="customer_credit_id" type="s:string" />
<s:element minOccurs="0" maxOccurs="1" name="customer_credit_password" type="s:string" />
<s:element minOccurs="0" maxOccurs="1" name="customer_tel" type="s:string" />

```

```
</s:sequence>
```

```
</s:complexType>
```

```
</s:element>
```

```
- <s:element name="orderCompaqResponse">
```

```
- <s:complexType>
```

```
- <s:sequence>
```

```
<s:element minOccurs="0" maxOccurs="1" name="orderCompaqResult" type="s:string" />
```

```
</s:sequence>
```

```
</s:complexType>
```

```
</s:element>
```

```
</s:schema>
```

```
</types>
```

```
- <message name="SelectCompaqFromStockSoapIn">
```

```
<part name="parameters" element="s0:SelectCompaqFromStock" />
```

```
</message>
```

```
- <message name="SelectCompaqFromStockSoapOut">
```

```
<part name="parameters" element="s0:SelectCompaqFromStockResponse" />
```

```
</message>
```

```
- <message name="SelectIntelCompaqFromStockSoapIn">
```

```
<part name="parameters" element="s0:SelectIntelCompaqFromStock" />
```

```
</message>
```

```
- <message name="SelectIntelCompaqFromStockSoapOut">
```

```
<part name="parameters" element="s0:SelectIntelCompaqFromStockResponse" />
```

```
</message>
```

```
- <message name="SelectAMDCmpaqFromStockSoapIn">
```

```
<part name="parameters" element="s0:SelectAMDCmpaqFromStock" />
```

```
</message>
```

```
- <message name="SelectAMDCmpaqFromStockSoapOut">
```

```
<part name="parameters" element="s0:SelectAMDCompaqFromStockResponse" />
</message>
```

```
- <message name="SearchCompaqFromStockSoapIn">
```

```
<part name="parameters" element="s0:SearchCompaqFromStock" />
</message>
```

```
- <message name="SearchCompaqFromStockSoapOut">
```

```
<part name="parameters" element="s0:SearchCompaqFromStockResponse" />
</message>
```

```
- <message name="orderCompaqSoapIn">
```

```
<part name="parameters" element="s0:orderCompaq" />
</message>
```

```
- <message name="orderCompaqSoapOut">
```

```
<part name="parameters" element="s0:orderCompaqResponse" />
</message>
```

```
- <portType name="CompaqSoap">
```

```
- <operation name="SelectCompaqFromStock">
```

```
<input message="s0:SelectCompaqFromStockSoapIn" />
```

```
<output message="s0:SelectCompaqFromStockSoapOut" />
```

```
</operation>
```

```
- <operation name="SelectIntelCompaqFromStock">
```

```
<input message="s0:SelectIntelCompaqFromStockSoapIn" />
```

```
<output message="s0:SelectIntelCompaqFromStockSoapOut" />
```

```
</operation>
```

```
- <operation name="SelectAMDCompaqFromStock">
```

```
<input message="s0:SelectAMDCompaqFromStockSoapIn" />
```

```
<output message="s0:SelectAMDCompaqFromStockSoapOut" />
```

```
</operation>
```

```
- <operation name="SearchCompaqFromStock">
```

```
<input message="s0:SearchCompaqFromStockSoapIn" />
```

```
<output message="s0:SearchCompaqFromStockSoapOut" />
```

```
</operation>
```

```

- <operation name="orderCompaq">
  <input message="s0:orderCompaqSoapIn" />
  <output message="s0:orderCompaqSoapOut" />
</operation>
</portType>
- <binding name="CompaqSoap" type="s0:CompaqSoap">
  <soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="document" />
- <operation name="SelectCompaqFromStock">
  <soap:operation soapAction="http://tempuri.org/SaleCompaq/Compaq/SelectCompaqFromStock"
    style="document" />
- <input>
  <soap:body use="literal" />
</input>
- <output>
  <soap:body use="literal" />
</output>
</operation>
- <operation name="SelectIntelCompaqFromStock">
  <soap:operation
    soapAction="http://tempuri.org/SaleCompaq/Compaq/SelectIntelCompaqFromStock"
    style="document" />
- <input>
  <soap:body use="literal" />
</input>
- <output>
  <soap:body use="literal" />
</output>
</operation>
- <operation name="SelectAMDCCompaqFromStock">

```

```

<soap:operation
  soapAction="http://tempuri.org/SaleCompaq/Compaq/SelectAMDCompaqFromStock"
  style="document" />

```

```

- <input>
  <soap:body use="literal" />
</input>

```

```

- <output>

```

```

<soap:body use="literal" />

```

```

</output>

```

```

</operation>

```

```

- <operation name="SearchCompaqFromStock">

```

```

  <soap:operation soapAction="http://tempuri.org/SaleCompaq/Compaq/SearchCompaqFromStock"
    style="document" />

```

```

- <input>

```

```

  <soap:body use="literal" />

```

```

</input>

```

```

- <output>

```

```

  <soap:body use="literal" />

```

```

</output>

```

```

</operation>

```

```

- <operation name="orderCompaq">

```

```

  <soap:operation soapAction="http://tempuri.org/SaleCompaq/Compaq/orderCompaq"
    style="document" />

```

```

- <input>

```

```

  <soap:body use="literal" />

```

```

</input>

```

```

- <output>

```

```

  <soap:body use="literal" />

```

```

</output>

```

```

</operation>

```

```

</binding>

```

```
- <service name="Compaq">  
- <port name="CompaqSoap" binding="s0:CompaqSoap">  
  <soap:address location="http://localhost/SaleCompaq/Compaq.asmx" />  
</port>  
</service>  
</definitions>
```

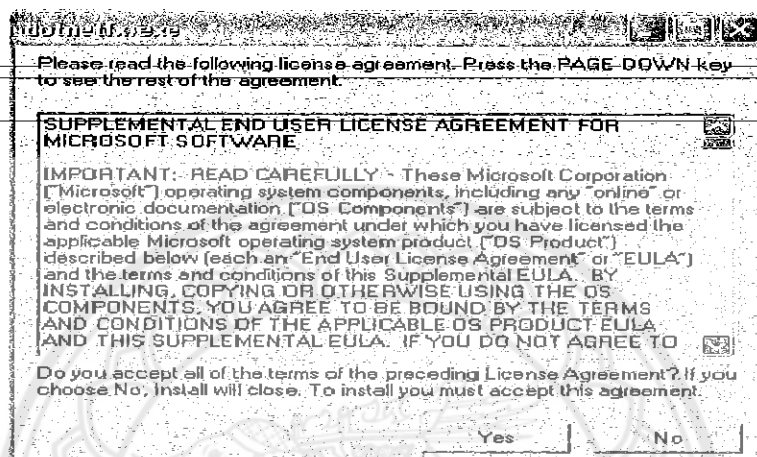
เอกสาร XML ที่เป็นผลลัพธ์จากการเรียกบริการดังกล่าว



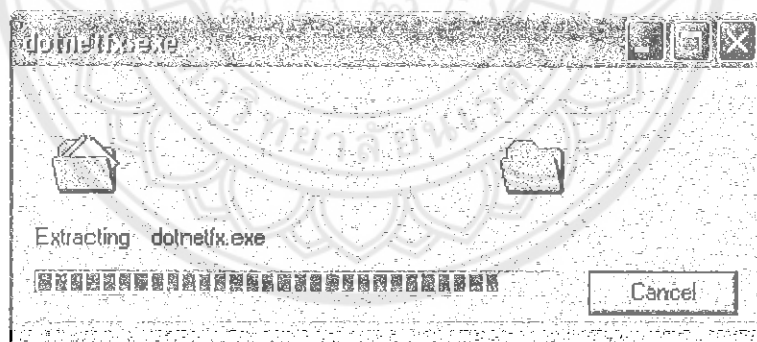
2. การติดตั้งโปรแกรม ASP.NET

สำหรับการติดตั้งโปรแกรม ASP.NET นั้น เราต้องสำรวจทรัพยากรของเครื่องคอมพิวเตอร์ก่อนว่า สามารถที่จะรองรับโปรแกรม ASP.NET ได้หรือไม่ ถ้าเครื่องคอมพิวเตอร์สามารถรองรับโปรแกรมได้ ก็เริ่มทำการ Install ASP.NET ได้

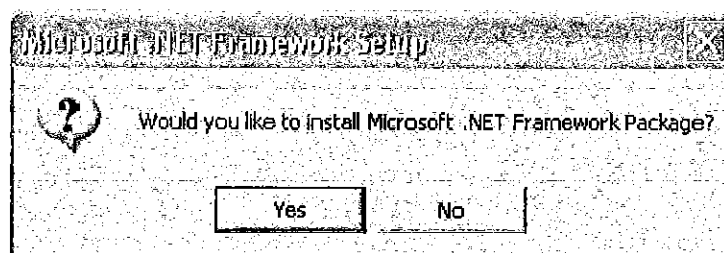
2.1 นำแผ่น Install ASP.NET นำไปใส่เครื่อง โดยที่จะต้องปฏิบัติตามขั้นตอนดังต่อไปนี้



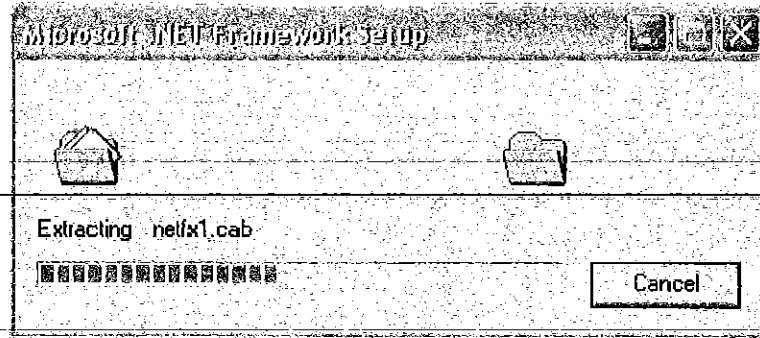
รูปที่ 1.5 เมื่อ Install ASP.NET ให้ทำการกด "Yes"



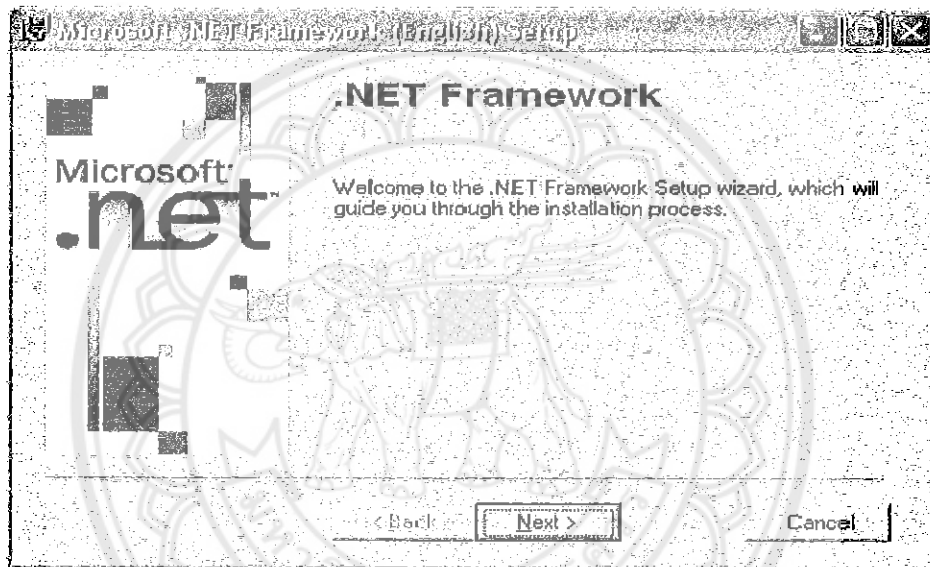
รูปที่ 1.6 โปรแกรมทำการ Install ASP.NET



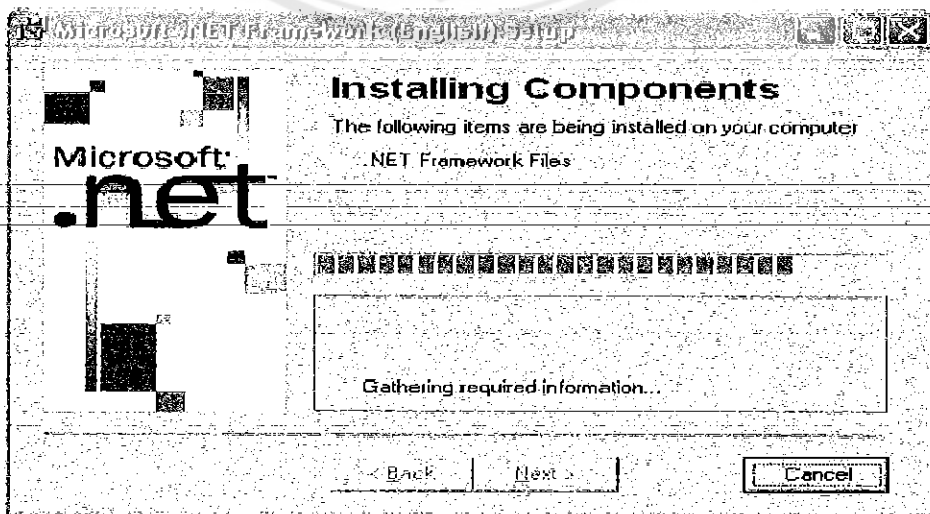
รูปที่ 1.7 ให้ทำการกด "Yes"



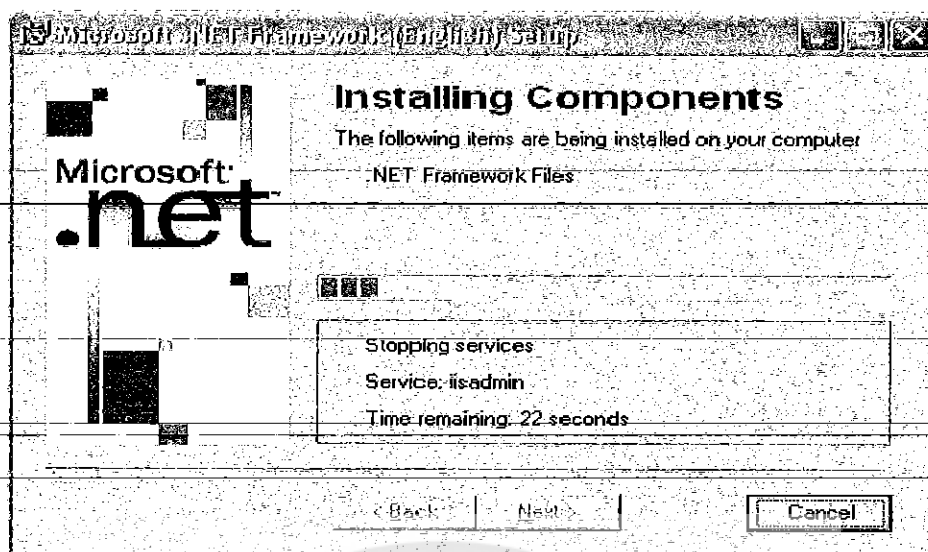
รูปที่ 1.8 โปรแกรมทำการ Install ASP.NET



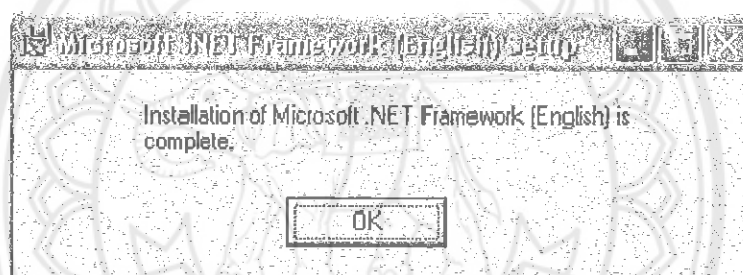
รูปที่ 1.9 ให้ทำการกด "Next" เพื่อเข้าสู่ระบบการ Install



รูปที่ 1.10 โปรแกรมอยู่ในขั้นตอนการติดตั้งโปรแกรม



รูปที่ 1.11 โปรแกรมดำเนินการติดตั้งโปรแกรม

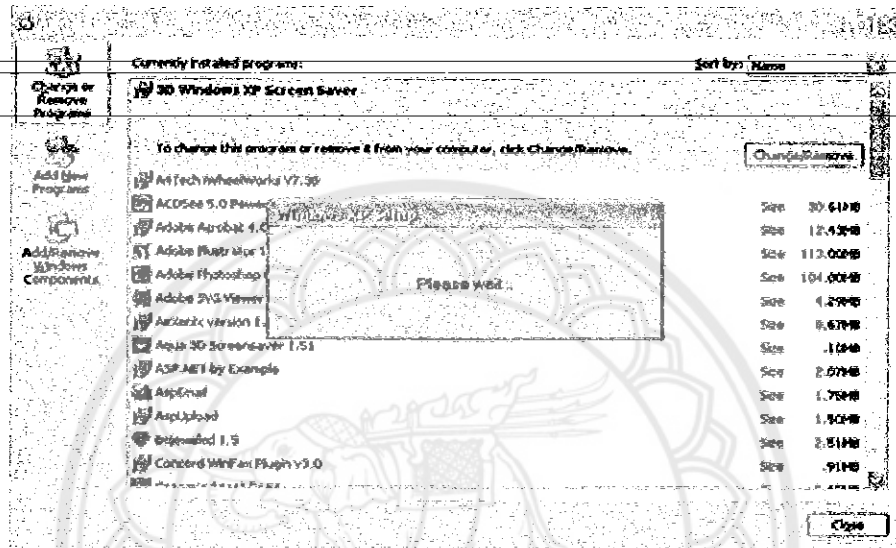


รูปที่ 1.12 เสร็จสิ้นการดำเนินการติดตั้งโปรแกรม กด "OK"

3. การ Install Internet Information Services (IIS)

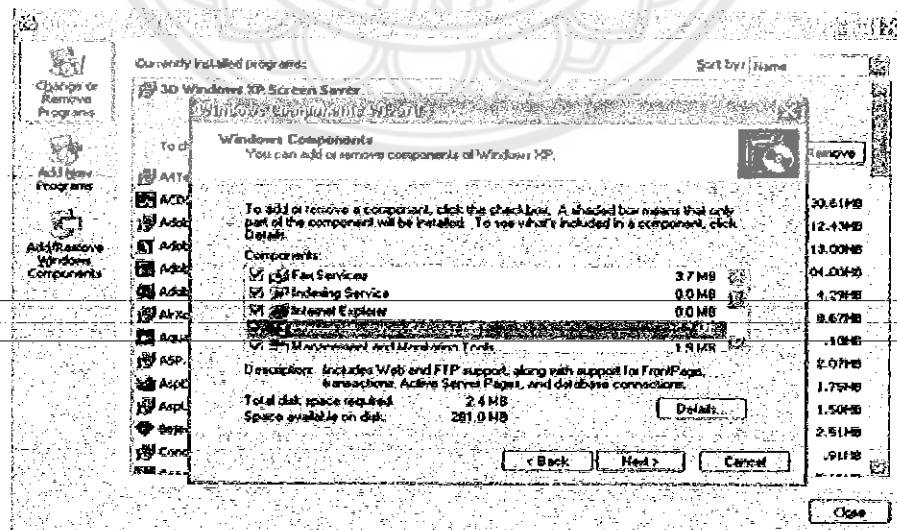
ในการลง IIS และการ Set up นั้นซึ่งจริงๆ แล้วไม่ยากอย่างที่คิด ในการลง IIS สามารถลงได้แบบอัตโนมัติจากแผ่น Windows XP/2000 หรือเข้าไปที่ Control Panel

แบบแรกเข้าไปที่ Control Panel แล้วดับเบิลคลิกที่ Add or Remove Program ซึ่งจะเกิดหน้าจอของ Add or Remove Program และให้คลิกที่ Add/Remove Windows Components



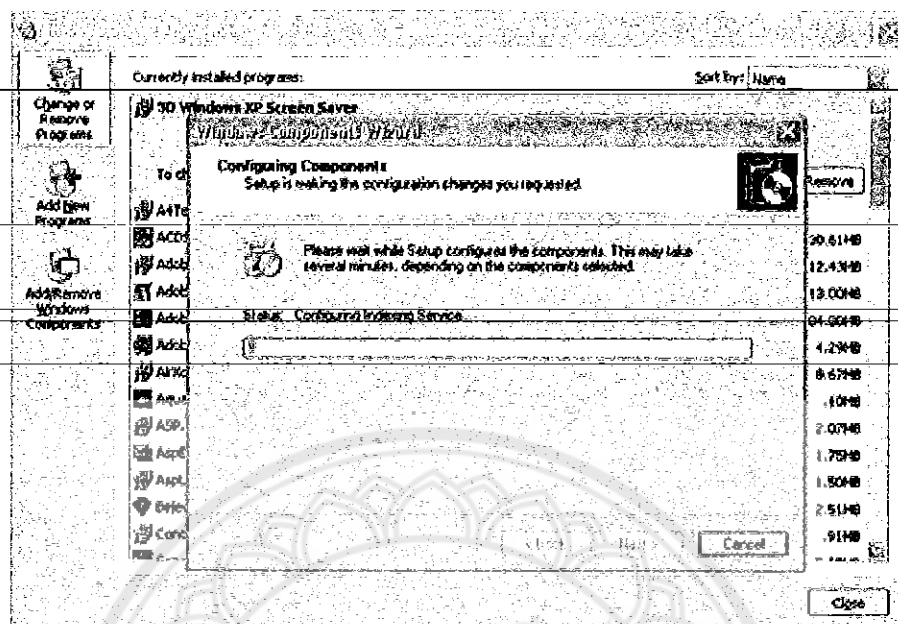
รูปที่ 1.13 การ Install Internet Information Services (IIS)

จากนั้นให้ใส่เครื่องหมายถูก หน้า Internet Information Services (IIS)

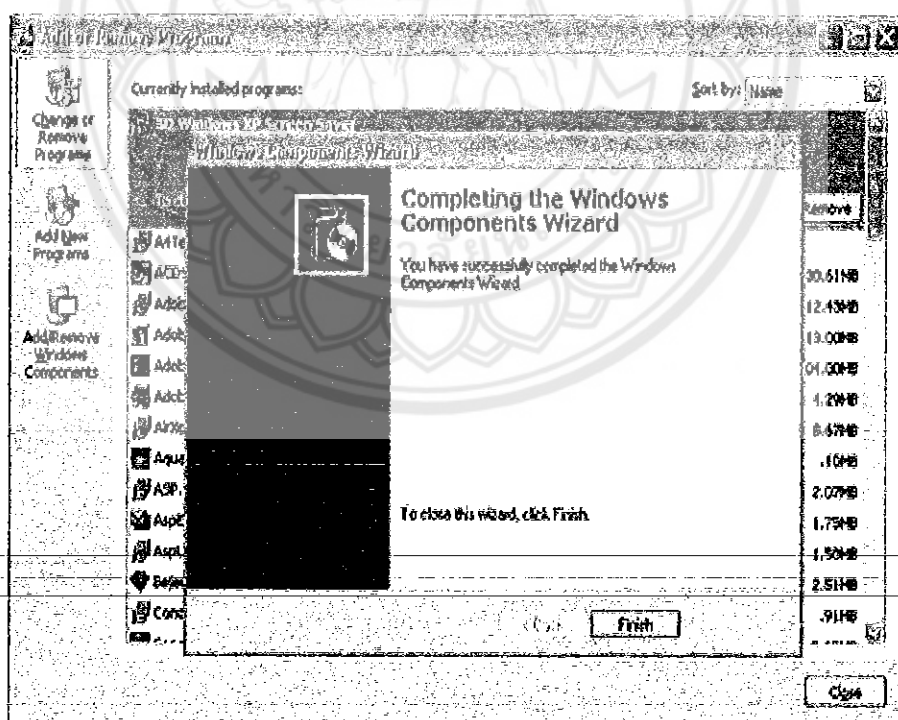


รูปที่ 1.14 การ Install Internet Information Services (IIS)

และตัว Windows จะทำการลง IIS ให้เองอัตโนมัติ จากนั้นก็กด Finish เป็นการเสร็จสิ้น



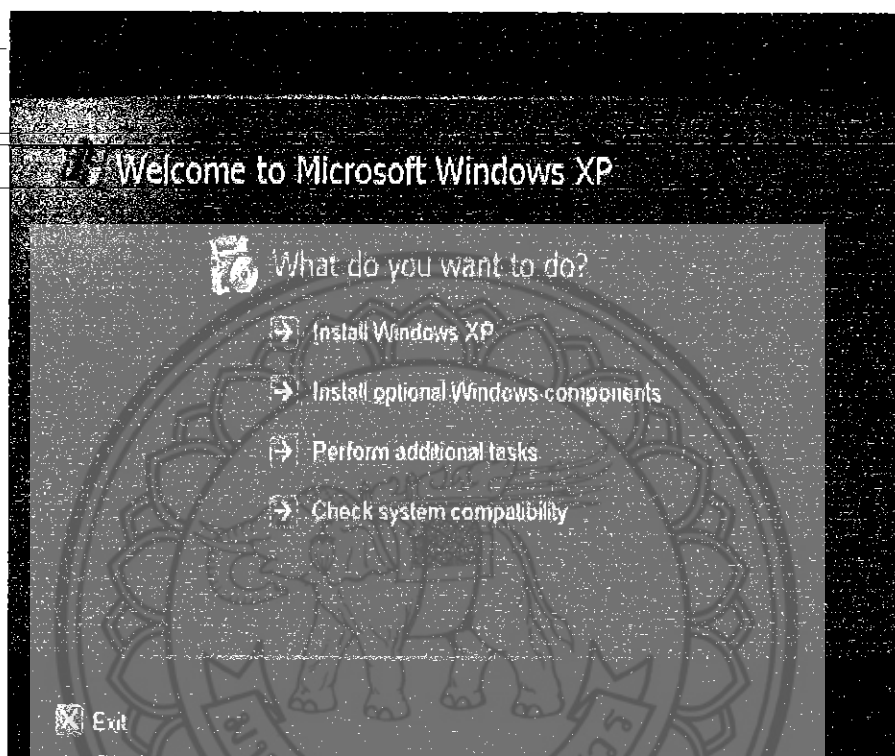
รูปที่ 1.15 การ Install Internet Information Services (IIS)



รูปที่ 1.16 กด Finish เป็นการเสร็จสิ้น

แบบที่สองโดยวิธีการลงจากแผ่น Windows XP/2000 ซึ่งจะเป็นการลงแบบ Auto run เมื่อใส่แผ่น Windows XP/2000

ให้คลิกที่ Install optional Windows component ซึ่งขั้นตอนทั้งหมดก็เหมือนกับการลงโดยผ่าน Control Panel Note ทักขั้นตอนต้องใส่แผ่น Windows XP/2000 ไว้ในไดรฟ์ CD Rom



รูปที่ 1.17 การ Install Internet Information Services (IIS) แบบสอง

4. การ Set up Internet Information Services (IIS)

เมื่อลง IIS แล้วก็มาถึงขั้นตอนการตั้งค่าการใช้งาน โดยให้เปิด Browser ขึ้นมาและพิมพ์
<http://localhost>



รูปที่ 1.18 การ Set up Internet Information Services (IIS)

ซึ่งปกติแล้ว Path ตั้งต้นของ IIS จะถูก Set ไว้ที่ C:\Inetpub\wwwroot เพราะฉะนั้นเมื่อสร้างไฟล์ asp ก็ให้เก็บไว้ที่ Path นี้ แต่สามารถที่จะ set ให้ IIS เรียกไปที่ Path อื่นได้ หรือตั้งค่า Path ตั้งต้นที่ไม่ใช่ C:\Inetpub\wwwroot ก็ได้

ให้เข้าไปที่ Control Panel ดับเบิลคลิกที่ Administrative Tools จะได้น้ำจอของ Administrative Tools และให้ดับเบิลคลิกที่ Internet Information Services จะได้น้ำจอของ Internet Information Services ตามรูป

ประวัติผู้เขียนโครงการ



ชื่อ นายจักรพรรดิ โปชาต
 ภูมิลำเนา 226/3 หมู่ 8 ต.แม่สาย อ.แม่สาย จ.เชียงราย 57130
 ประวัติการศึกษา - ปีการศึกษา 2539 จบชั้นประถมศึกษาปีที่ 6
 โรงเรียนพรพิบูลพิทยา จ.เชียงราย
 - ปีการศึกษา 2545 จบชั้นมัธยมศึกษาปีที่ 6
 โรงเรียนแม่สายประสิทธิ์ศาสตร์ จ.เชียงราย
 - ปัจจุบัน ปีการศึกษา 2549 กำลังศึกษาในระดับ
 ปริญญาตรี ชั้นปีที่ 4 สาขาวิศวกรรมคอมพิวเตอร์
 คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร จ.พิษณุโลก
 E-mail - jakkapat_p@yahoo.com



ชื่อ นายวีรยุทธ หลวงสิทธิ์
 ภูมิลำเนา 63 หมู่ 9 ต.แม่สาย อ.แม่สาย จ.เชียงราย 57130
 ประวัติการศึกษา - ปีการศึกษา 2539 จบชั้นประถมศึกษาปีที่ 6
 โรงเรียนพรพิบูลพิทยา จ.เชียงราย
 - ปีการศึกษา 2545 จบชั้นมัธยมศึกษาปีที่ 6
 โรงเรียนแม่สายประสิทธิ์ศาสตร์ จ.เชียงราย
 - ปัจจุบัน ปีการศึกษา 2549 กำลังศึกษาในระดับ
 ปริญญาตรีชั้นปีที่ 4 สาขาวิศวกรรมคอมพิวเตอร์
 คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร จ.พิษณุโลก
 E-mail - wonwon10126@hotmail.com