

การสร้างร้านหนังสือด้วยเว็บเซอร์วิส
Building Book Store with Web Service

โดย

นายอรรถพร หวังพูนทรัพย์

รหัส 43067042



H001935

อาจารย์ที่ปรึกษา

ดร.จันทร์บูรณ์ สถิตวิริยวงศ์

วัน เดือน ปี.....	19 ต.ค. 2550
เลขทะเบียน.....	01935
เลขเรียกหนังสือ.....	วท ๑356ก 2545
"ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล."	

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
ภาคเรียนที่ 2 ปีการศึกษา 2545
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ชื่อหัวข้อ	การสร้างร้านหนังสือด้วยเว็บเซอร์วิส
นักศึกษา	นายอรรถพร หวังพูนทรัพย์
อาจารย์ที่ปรึกษา	ดร.จันทร์บุรณ์ สถิตวิริยวงศ์
ระดับการศึกษา	วิทยาศาสตร์มหาบัณฑิต สาขาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2545

บทคัดย่อ

เทคโนโลยีเว็บเซอร์วิสกำลังมีบทบาทเพิ่มขึ้นมากในปัจจุบัน เนื่องจากว่าเว็บเซอร์วิสสามารถทำงานและเรียกใช้ข้ามแพลตฟอร์มได้ และในการใช้งานจะเรียกใช้เฉพาะส่วนที่ใช้งานจริงเท่านั้น สำหรับในโครงการนี้เป็นตัวอย่างแอปพลิเคชันที่ใช้ฐานข้อมูลประยุกต์เข้ากับการใช้เว็บเซอร์วิสในการซื้อหนังสือบนอินเทอร์เน็ต โดยใช้ ASP.NET เป็นภาษาที่ใช้ในการพัฒนา ประกอบด้วย 2 ส่วนหลักคือ ส่วนที่เป็นหน้าร้าน(Store Front) และหลังร้าน (Back Office) ซึ่งผลที่ได้จากการทำโครงการนี้สามารถนำไปใช้เป็นต้นแบบในการพัฒนาแอปพลิเคชันร้านค้าประเภทอื่นๆบนอินเทอร์เน็ตได้ต่อไป

Title	Building Book store with web service
Student	Mr.Attaporn Wangpoonsarp
Advisor	Dr.Junboon Sathitwiriya Wong
Level of Study	Master of Science in Information Technology
Major	Information Science
Academic Year	2002

ABSTRACT

Web Service Technology is popular in currently because it is called across any platform and return only real functional service. This project is application example using Database with web service for buying any books on internet and developing tool is ASP.NET. It include 2 part: Store Front and Back Office. For Result from developmet can apply with any store about e-commerce on internet.

สารบัญ

หน้า

บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VI
สารบัญภาพ	VII
บทที่	
1. บทนำ	1
1.1 ความเป็นมา	1
1.2 วัตถุประสงค์ในการพัฒนาระบบ	1
1.3 ขอบเขตของการพัฒนาระบบ	2
1.4 ขั้นตอนการดำเนินงาน	2
1.5 ผลที่คาดว่าจะได้รับ	3
1.6 เครื่องมือที่ใช้ในการพัฒนาระบบ	3
2. ทฤษฎีความรู้ที่เกี่ยวข้อง	4
2.1 HTTP	4
2.2 SOAP	6
2.3 XML	8
2.4 Web Service	10
2.5 .Net Framework	13
2.6 ASP.NET	15
2.7 VB.NET	17
2.8 ADO.NET	18

สารบัญ (ต่อ)

หน้า

3. การวิเคราะห์และออกแบบระบบงาน	21
3.1 โครงสร้างระบบ	21
3.2 การวิเคราะห์ระบบ	22
3.3 การออกแบบระบบฐานข้อมูล	32
4. การพัฒนาระบบ	36
4.1 ส่วนประกอบระบบ	36
4.2 หน้าจอของระบบ	37
5. บทสรุปและข้อเสนอแนะ	46
5.1 บทสรุป	46
5.2 ข้อเสนอแนะ	46
บรรณานุกรม	47
ประวัติผู้เขียน	48

สารบัญตาราง

ตารางที่	หน้า
1.1 ขั้นตอนการดำเนินงานโครงการ	2
3.1 ตารางในฐานข้อมูลทั้งหมดของระบบ	32
3.2 รายละเอียดของตาราง Customers	33
3.3 รายละเอียดของตาราง Categories	33
3.4 รายละเอียดของตาราง Books	34
3.5 รายละเอียดตาราง Orders	34
3.6 รายละเอียดตาราง OrderDetails	35
3.7 รายละเอียดตาราง Reviews	35



สารบัญภาพ

ภาพที่	หน้า
2.1 ลักษณะการทำงานของ HTTP request-response	5
2.2 SOAP กับ โพรโทคอลเครือข่าย	6
2.3 ตัวอย่างการแลกเปลี่ยนข้อมูลในรูป XML	10
2.4 ขั้นตอนการเรียกใช้เว็บเซอร์วิส	12
2.5 การทำงานของ ASP.NET	16
3.1 Context Diagram ของระบบร้านค้าหนังสือบนอินเทอร์เน็ต	22
3.2 Data Flow Diagram Level 1 ของระบบร้านค้าหนังสือบนอินเทอร์เน็ต	23
3.3 Data Flow Diagram Level 2 ของ Process 1 (Register)	24
3.4 Data Flow Diagram Level 2 ของ Process 2 (Login)	25
3.5 Data Flow Diagram Level 2 ของ Process 3 (Change Customer Information)	26
3.6 Data Flow Diagram Level 2 ของ Process 4 (Search & Browse)	27
3.7 Data Flow Diagram Level 2 ของ Process 5 (Order)	28
3.8 Data Flow Diagram Level 2 ของ Process 6 (Tracking)	29
3.9 Data Flow Diagram Level 2 ของ Process 8 (Review Book)	30
3.10 Data Flow Diagram Level 2 ของ Process 7 (Manage Information)	31
3.11 Entity Relationship Data Model ของระบบร้านค้าหนังสือ	32
4.1 หน้าจอแรกของระบบ Store Front.....	37
4.2 หน้าจอการเข้าสู่ระบบ	38
4.3 หน้าจอ Register กับระบบ	38
4.4 การเรียกดูหนังสือตามที่ต้องการได้จากหมวดหนังสือ	39
4.5 สามารถค้นหาหนังสือ	39
4.6 ผลที่ได้จากการค้นหา	39
4.7 หน้าจอรายละเอียดของหนังสือ	39
4.8 หน้าจอแสดงความคิดเห็นเกี่ยวกับหนังสือ	40
4.9 แสดงผลความคิดเห็นเมื่อคลิกปุ่ม Submit	40

สารบัญภาพ (ต่อ)

ภาพที่	หน้า
4.10 แสดงหน้าจอ Shopping cart	41
4.11 แสดงรายการสั่งซื้ออีกครั้ง	41
4.12 แสดงผลรายการที่มีการสั่งซื้อไปแล้ว	42
4.13 แสดงหน้าจอ Account History	42
4.14 แสดงหน้าจอ Order Detail	42
4.15 แสดงหน้าจอ Edit Account	43
4.16 แสดงหน้าจอแรกของระบบ Back Office	43
4.17 แสดงหน้าจอการ login เข้าสู่ระบบ	43
4.18 แสดงเมนูย่อยในการจัดการกับตารางBook	44
4.19 แสดงหน้าจอ Delete/Update Book	44
4.20 แสดงหน้าจอ New Book	44
4.21 แสดงเมนูย่อยในการสรุปยอดขาย	45
4.22 แสดงรายละเอียดในการสรุปยอดขาย	45

บทที่ 1

บทนำ

1.1 ความเป็นมา

ในปัจจุบันเทคโนโลยีบนอินเทอร์เน็ตได้ก้าวขึ้นไปอีกขั้นหนึ่ง เนื่องมาจากการเกิดขึ้นของเทคโนโลยีที่เรียกว่า “Web Service” สำหรับในประเทศไทยแล้วยังไม่ได้รับความแพร่หลายมากนัก แต่ที่ต่างประเทศบางหน่วยงานเริ่มนำเทคโนโลยีนี้มาใช้กันแล้ว สำหรับนิยามของเว็บเซอร์วิส ยังไม่ชัดเจนนัก แต่โดยทั่วไปแล้วจะนำลักษณะการทำงานของตัวเว็บเซอร์วิสมาอธิบายเป็นนิยาม ดังนั้นเว็บเซอร์วิสจะหมายถึง แอปพลิเคชันหรือโปรแกรมซึ่งทำงานอย่างใดอย่างหนึ่งในลักษณะให้บริการ โดยจะถูกเรียกใช้งานจากแอปพลิเคชันหรือโปรแกรมอื่นๆผ่านเว็บ การให้บริการของเว็บเซอร์วิสจะมีเอกสารที่อธิบายคุณสมบัติของบริการกำกับไว้ (WSDL) และมีการนำเสนอให้สาธารณชนรับทราบ ผู้ใช้จึงสามารถค้นหาเว็บเซอร์วิส (UDDI) ได้โดยไม่จำเป็นต้องรู้ที่อยู่จริงของแอปพลิเคชันหรือโปรแกรมนั้น

สำหรับในโครงการนี้ก็ได้นำเทคโนโลยีของเว็บเซอร์วิสเข้ามาใช้ในการพัฒนา เหตุผลที่ได้นำเว็บเซอร์วิสเข้ามาใช้เนื่องจากว่าเว็บเซอร์วิสสามารถทำงานและเรียกใช้ข้ามแพลตฟอร์มได้ (ข้อดีของโปรโตคอล SOAP) และยังใช้ทรัพยากรในการทำงานที่มีประสิทธิภาพ เนื่องจากการใช้งานจะเรียกใช้เฉพาะส่วนที่ใช้งานจริงเท่านั้น โดยใช้ ASP.NET เป็นภาษาที่ใช้ในการพัฒนา ซึ่งมีการจัดรูปแบบในการเขียนโปรแกรมที่เป็นระเบียบและง่ายต่อการใ้มากขึ้น ซึ่งโครงการนี้เป็นตัวอย่างแอปพลิเคชันที่มีการใช้ฐานข้อมูลมาประยุกต์เข้ากับการใช้งานเว็บเซอร์วิสในการทำ E-Commerce

1.2 วัตถุประสงค์ในการพัฒนาระบบ

- เพื่อให้ได้ระบบร้านหนังสือที่มีประสิทธิภาพในกาทำงานที่ดีขึ้น
- เพื่อให้เข้าใจในการทำงานของเว็บเซอร์วิส (Web Service) ซึ่งเป็นเทคโนโลยีรูปแบบใหม่ในการพัฒนาระบบบนเว็บ (Web Application)
- เพื่อให้เข้าใจหลักการเขียนโปรแกรมแบบใหม่ ที่สามารถใช้งานได้สะดวกขึ้น

1.3 ขอบเขตของการพัฒนาระบบ

ทำการพัฒนาระบบร้านหนังสือเป็น 2 ส่วนหลักด้วยกันคือในส่วนหน้าร้าน (Store Front) และหลังร้าน (Back Office) มีรายละเอียดดังต่อไปนี้

- ในส่วนหน้าร้าน จะประกอบไปด้วย ส่วนของสมาชิกที่สามารถ login เข้าไปแก้ไขข้อมูลส่วนตัว, ส่วนที่สามารถเรียกดูรายการหนังสือตามที่ต้องการ, ส่วนที่ใช้ในการซื้อหนังสือที่ต้องการ, ส่วนที่ใช้ในการดูรายการซื้อหนังสือของสมาชิก
- สำหรับในส่วนของหลังร้าน จะเกี่ยวกับการเปลี่ยนแปลง, ปรับปรุง, แก้ไขข้อมูลสินค้า รวมถึงการสรุปยอดขายหนังสือในช่วงเวลาหนึ่งๆ เพื่อที่จะใช้เป็นข้อมูลประกอบการตัดสินใจสั่งซื้อหนังสือมาให้บริการกับสมาชิก

1.4 ขั้นตอนการดำเนินงาน

ตารางที่ 1.1 ขั้นตอนการดำเนินงาน โครงการ

เวลา / ขั้นตอน	ต้น	ปลาย	ต้น	ปลาย	ต้น	ปลาย	ต้น	ปลาย	ต้น	ปลาย
	ค.ค.	ค.ค.	พ.ย.	พ.ย.	ธ.ค.	ธ.ค.	ม.ค.	ม.ค.	ก.พ.	ก.พ.
ศึกษาระบบร้านหนังสือที่มีอยู่ อินเทอร์เน็ตและร้านค้าจริง										
ศึกษาเครื่องมือที่ใช้ในการพัฒนาระบบ										
วิเคราะห์และออกแบบระบบ										
ทำการพัฒนาระบบในส่วนหน้าร้าน										
ทำการพัฒนาระบบในส่วนหลังร้าน										
ทดสอบการใช้งานและทำการแก้ไขปรับปรุงระบบ										

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

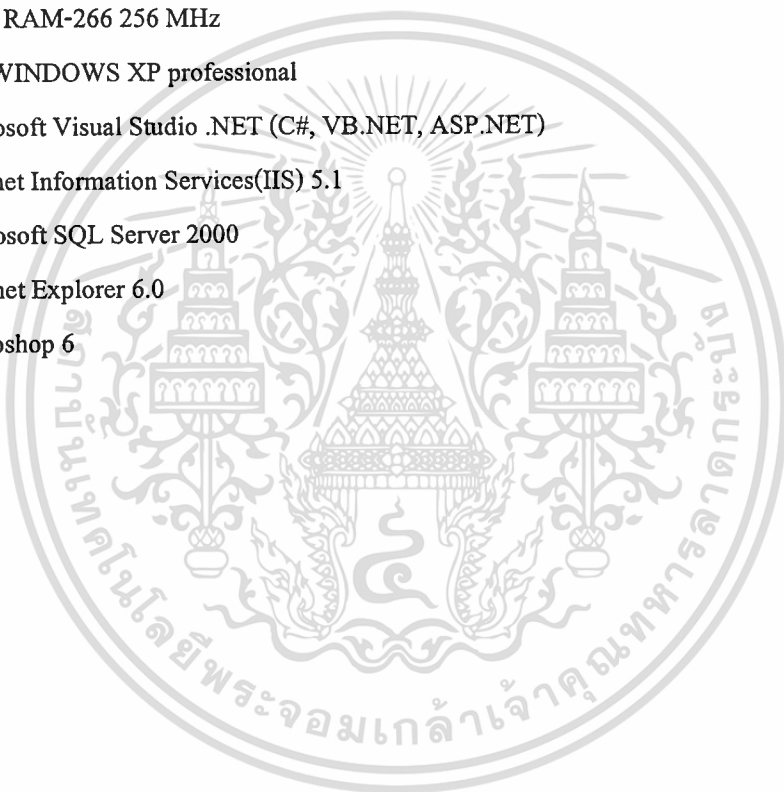
1.5 ผลที่คาดว่าจะได้รับ

- ได้ระบบร้านหนังสือที่สามารถนำไปใช้งานจริงได้
- ได้เข้าใจถึงวิธีการทำงานของเว็บเซอว์ริส
- ได้เรียนรู้ภาษาใหม่เพิ่มเติมในการพัฒนาระบบบนเว็บ

1.6 เครื่องมือที่ใช้ในการพัฒนาระบบ

ประกอบด้วย Hardware และ Software ดังต่อไปนี้

- CPU Athlon XP 1.7 GHz
- DDR RAM-266 256 MHz
- OS: WINDOWS XP professional
- Microsoft Visual Studio .NET (C#, VB.NET, ASP.NET)
- Internet Information Services(IIS) 5.1
- Microsoft SQL Server 2000
- Internet Explorer 6.0
- Photoshop 6



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 • ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและความรู้ที่เกี่ยวข้อง

2.1 HTTP

HTTP เป็นโพรโทคอลใน Application-Layer ซึ่งเป็นหัวใจสำคัญของเว็บ และถูกแบ่งการทำงานเป็น 2 ส่วน คือ Client program และ Server program ซึ่งจะทำงานบนระบบที่ต่างกัน มีการคุยกัน แลกเปลี่ยน HTTP message กัน HTTP จะกำหนดรูปแบบของข้อมูล(message) และวิธีการแลกเปลี่ยนข้อมูลระหว่าง client และ server

Web page ประกอบด้วยออบเจกต์ (Object) คือ ไฟล์ประเภทต่าง ๆ เช่น HTML, JPEG, GIF, Java applet, audio clip เป็นต้น ซึ่งไฟล์เหล่านี้จะมี URL เป็นตัวบอกที่อยู่ของไฟล์และจะเก็บอยู่ในรูปแบบของไฟล์ HTML สมมติว่า Web page ประกอบด้วยไฟล์รูป JPEG 5 ไฟล์และเท็กซ์ไฟล์ HTML 1 ไฟล์ ดังนั้น Web page นี้จะประกอบด้วย 5 ออบเจกต์ ดังนั้น Web page นี้จะอ้างถึงออบเจกต์ต่าง ๆ ด้วย URL ของแต่ละออบเจกต์

ในแต่ละ URL จะประกอบด้วย 2 ส่วน คือ Host name ของ server และ path ของออบเจกต์นั้น ๆ ตัวอย่างเช่น

www.someSchool.edu/someDepartment/picture.gif

www.someSchool.edu คือ host name

[/someDepartment/picture.gif](http://www.someSchool.edu/someDepartment/picture.gif) คือ path name

Browser คือโปรแกรมที่ทำหน้าที่ติดต่อระหว่างผู้ใช้ (user) และ web server ที่เป็นที่ยอมรับได้แก่ Netscape Communication และ Microsoft Explorer นอกจากนี้ Browser ยังทำหน้าที่ implement HTTP ของ Client อีกด้วย

Web server คือที่เก็บของออบเจกต์ต่าง ๆ ซึ่งในแต่ละออบเจกต์จะระบุที่อยู่โดย URL เว็บ Server ที่เป็นที่ยอมรับได้แก่ Apache, Microsoft Internet Information Server และ Netscape Enterprise Server และ Web Server ทำหน้าที่ implement HTTP ของ Server ด้วย ดังภาพที่ 2.1 เมื่อผู้ใช้งานต้องการข้อมูลของ web page (เช่นคลิก Hyperlink) browser จะส่ง HTTP request เพื่อขอรายละเอียดของ web page คือออบเจกต์ต่าง ๆ ไปยัง server เมื่อ server ได้รับ request จะส่ง HTTP response message ซึ่งมีออบเจกต์ ต่าง ๆ ไปด้วย ปัจจุบันนี้ใช้ HTTP เวอร์ชัน 1.1

HTTP มี message 2 ประเภท

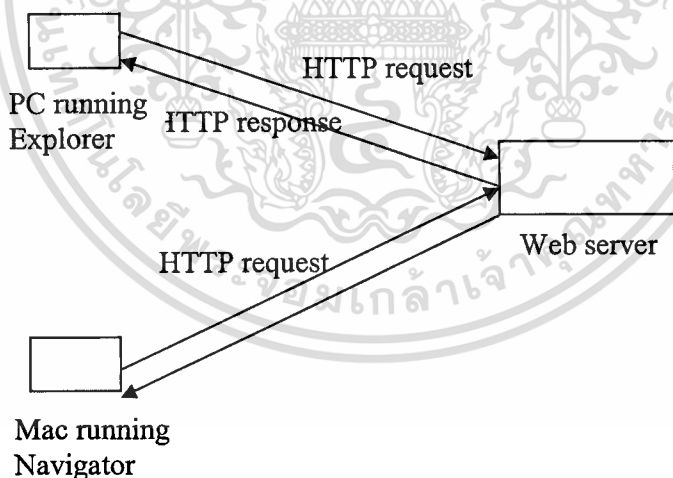
- request message
- response message

HTTP Request message จะประกอบด้วย 2 ส่วนคือ 1. Request line และ 2. header line ต่อจากนั้นจะเป็นส่วน entity body คือส่วนของข้อมูลที่ร้องขอไป

HTTP Response message จะประกอบด้วย 2 ส่วนคือ 1. Status line และ 2. header line ต่อจากนั้นจะเป็นส่วน entity body คือส่วนของข้อมูลที่ถูกร้องขอไป

ตัวอย่างของ Status code เช่น

- 200 OK : พบออบเจกต์ที่ต้องการและถูกส่งกลับมาเรียบร้อยแล้ว
- 301 Moved Permanently: ออบเจกต์ที่ต้องการถูกย้ายไปยัง URL ใหม่ ซึ่งจะระบุใน Location : header
- 400 Bad Request: sever ไม่เข้าใจความหมายการร้องขอดังกล่าว
- 404 Not found: ออบเจกต์ที่ต้องการไม่อยู่บน server ดังกล่าว
- 505 HTTP version not supported: HTTP version ของการร้องขอไม่สนับสนุนโดย server ดังกล่าว



ภาพที่ 2.1 ลักษณะการทำงานของ HTTP request-response

2.2 Simple Object Access Protocol (SOAP)

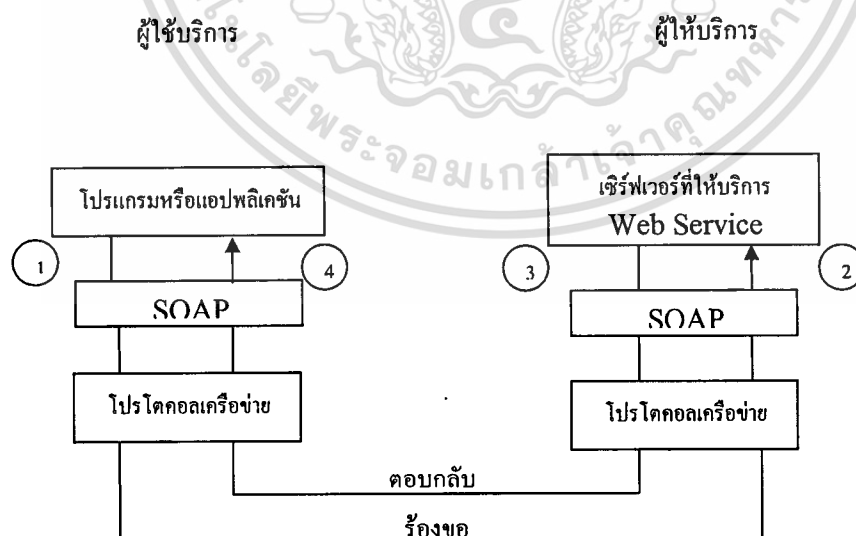
SOAP เป็นเทคโนโลยีที่พัฒนาโดย Microsoft และ Developmentor สำหรับกระจายแอปพลิเคชันที่เป็น XML ไปบนอินเทอร์เน็ต SOAP ได้จัดเตรียมเฟรมเวิร์คสำหรับจัดการกับแอปพลิเคชันทั้งด้านการเปลี่ยนแปลง, การเข้ารหัสข้อมูลและเก็บไว้ในโมดูล ซึ่ง SOAP มีองค์ประกอบหลักที่ทำงานประสานกัน 3 องค์ประกอบดังนี้

1. SOAP Envelope มีขึ้นเพื่อเป็นการอธิบายเนื้อหา และผู้รับ SOAP message
2. SOAP Encoding rule เป็นส่วนของการกำหนดคดกไกที่ใช้ในการแลกเปลี่ยนชนิดของข้อมูลที่ใช้ในแอปพลิเคชันที่เรียกใช้ (ส่วนที่เป็น XML)
3. SOAP Remote Procedure Call (RPC) เป็นส่วนที่ใช้สำหรับการทำรีโมตไปยังเครื่องคอมพิวเตอร์อื่นๆ

ซึ่งทุก SOAP Message จะประกอบด้วย

- SOAP envelope เป็นส่วนแรกของเอกสาร XML ที่แสดงว่าเป็น SOAP Message
- SOAP header เป็นส่วนที่บอกคุณสมบัติต่างๆของ SOAP Message นั้นๆว่าประกอบด้วยอะไรบ้าง
- SOAP body เป็นส่วนของข้อมูลที่ใช้ในการส่งข้อความ SOAP ตั้งแต่แรกจนถึงสิ้นสุด

SOAP เป็นโปรโตคอลที่สามารถทำงานได้กับโปรโตคอลเครือข่ายหลายโปรโตคอลได้ เช่น HTTP, SMTP ฯลฯ แต่ในการใช้งานจริงมักจะพบเฉพาะ โปรโตคอล HTTP



ภาพที่ 2.2 SOAP กับ โปรโตคอลเครือข่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในภาพที่ 2.2 ประกอบด้วย 4 ขั้นตอนด้วยกัน แต่ละขั้นตอนอธิบายการทำงานได้ดังนี้

- ผู้ขอใช้บริการสร้าง SOAP message เพื่อเรียกใช้บริการ (ในที่นี้เป็น web service)
 - ผู้ให้บริการได้รับ SOAP message จากผู้ขอใช้บริการซึ่งอยู่ในรูปแบบ XML จึงแปลข้อความนั้นกลับมายังในรูปแบบที่เว็บเซิร์ฟเวอร์เข้าใจ แล้วตรวจสอบว่าผู้ขอใช้บริการต้องการเรียกใช้ web service ชื่ออะไร เมธอดอะไร และมีการส่งพารามิเตอร์อะไรมาด้วยกับ SOAP message ที่ request มา แล้วจากนั้นทำการส่งไปให้คอมพิวเตอร์ที่ให้บริการ web service นั้นๆดำเนินการประมวลผล
 - หลังจากคอมพิวเตอร์ที่ให้บริการ web service ส่งผลลัพธ์มาแล้ว ผู้ให้บริการก็จะสร้าง SOAP message ที่มีผลลัพธ์นั้นออกมาด้วย แล้วจึงส่งผ่านทางโพรโทคอลเครือข่ายไปยังผู้ขอใช้บริการ
 - ผู้ขอใช้บริการได้รับ SOAP message ที่อยู่ในรูปแบบ XML จึงแปลข้อความนั้นกลับมายังในรูปแบบที่โปรแกรมของผู้ขอใช้บริการเข้าใจแล้วนำผลลัพธ์ที่ได้ไปใช้งานเช่น เช่น แสดงผล, คำนวณค่าต่างๆ เป็นต้น
- ซึ่ง SOAP message ที่รับ-ส่งไปมานั้นอยู่ในรูปแบบ XML และต้องมีการแปลกลับมายังในรูปแบบที่โปรแกรมหรือเว็บเซิร์ฟเวอร์เข้าใจ โดยใช้ XML Parser ซึ่ง XML Parser นี้เป็นโปรแกรมที่ทำหน้าที่แปลความหมายของเอกสาร XML โดยเฉพาะ
- ข้อดีของโพรโทคอล SOAP มีดังนี้**
- SOAP มีความสามารถเรียกใช้คอมพิวเตอร์หรือ web service ข้ามเครื่อง ข้ามแพลตฟอร์ม ข้ามภาษาได้ โดยอาศัยโพรโทคอลที่มีอยู่บนอินเทอร์เน็ตอย่าง HTTP สำหรับรูปแบบข้อความที่สื่อสารกันเป็นภาษา XML ลักษณะเป็นข้อความธรรมดาปิดด้วยแท็ก ทำให้เข้าใจได้ในทุกแพลตฟอร์ม แต่ต้องมี XML Parser มาแปลข้อความนั้นจึงจะเข้าใจได้
 - SOAP มีความสามารถในการผ่านระบบที่มีไฟร์วอลล์ (firewall) เพื่อป้องกันการบุกรุกได้ เนื่องจาก SOAP ทำงานอยู่กับโพรโทคอล HTTP ซึ่งไฟร์วอลล์โดยทั่วไปจะเปิดให้การสื่อสารด้วยโพรโทคอล HTTP ผ่านได้สะดวก ดังนั้น SOAP message จึงผ่านไปได้ ในขณะที่โพรโทคอลแบบเดิมเช่น DCOM, RMI, IIOP ไฟร์วอลล์มักไม่ยอมให้ผ่านข้อเสียของโพรโทคอล SOAP ดังนี้
 - SOAP message เป็นเอกสาร XML ทำให้เสียเวลาในการแปลกลับไปเป็นรูปแบบที่โปรแกรมเข้าใจ

- SOAP มีการทำงานกับโพรโตคอล HTTP ซึ่งโพรโตคอลนี้มีอัตราในการรับ-ส่งข้อมูลต่ำ โดยเฉพาะเมื่อนำไปเปรียบเทียบกับ โพรโตคอลอื่นๆ เช่น IIOP

2.3 Extensible Markup Language (XML)

XML ถูกพัฒนาขึ้นมาในปี ค.ศ. 1996 (ปี พ.ศ.2539) โดย World Wide Web Consortium's (W3C's) XML Working Group ซึ่งกลุ่มทำงานนี้ออกแบบและพัฒนา XML ขึ้นมาเนื่องจากข้อจำกัดหลายด้านของ HTML (HyperText Markup Language) ซึ่งเป็นภาษาพื้นฐานในการพัฒนาเว็บเพจ ข้อจำกัดหลักของ HTML ก็คือการขาดความสามารถในการขยายไม่ว่าจะเป็นการเพิ่มหรือเปลี่ยนแปลงพีเจอรในการพัฒนา XML จะยึดหลักมาตรฐานตามภาษาที่ชื่อ Standard Generalized Markup Language (SGML) โดย XML จะใช้พีเจอรที่ทรงประสิทธิภาพของ SGML และรวมมาตรฐานเบื้องต้นของ XML สำหรับ style sheet และ advanced hyperlink ด้วยภาษา Extensible Stylesheet Language (XSL) ที่รวม Cascading Style Sheet (CSS) สำหรับการจัดฟอร์แมตเอกสาร HTML และ Document Style and Semantics Specification Language (DSSSL) สำหรับจัดฟอร์แมตเอกสาร SGML ไว้ด้วยกัน

มาตรฐานหลักของ XML คือการรับ-ส่งข้อมูลระหว่างแอปพลิเคชันผ่านทางอินเทอร์เน็ต สำหรับเอกสาร XML ผู้สร้างเอกสารสามารถอธิบายประเภทของข้อมูล รวมทั้งสูตรทางคณิตศาสตร์, คำแนะนำในการปรับคอนฟิกูเรชันของซอฟต์แวร์, ดนตรี หรือแม้แต่รายงานด้านการเงินการคลัง คนและเครื่องจักรสามารถอ่านและเข้าใจเอกสาร XML ได้ทันที ใน .Net Framework ใช้ XML อย่างกว้างขวาง โดย Framework Class Library (FCL) ได้จัดเตรียมคลาสสำหรับการใช้งาน XML ได้ทันทีทำให้ผู้สร้างไม่ต้องเสียเวลามากเกินไป

การกำหนดโครงสร้างของเอกสาร XML (หรือเรียกว่า schema) เป็นการกำหนดรูปแบบโครงสร้างของเอกสาร XML (หรือที่เรียกว่า Data Model) ว่าประกอบด้วยแท็กอะไรบ้าง, มีแอตทริบิวต์อะไรบ้าง, ค่าของแอตทริบิวต์เป็นอะไรได้บ้าง, ลักษณะการซ้อนแท็กเป็นอย่างไรได้บ้าง เป็นต้น การทำเช่นนี้เพื่อเป็นรูปแบบในการติดต่อระหว่างองค์กร ที่จะแลกเปลี่ยนข้อมูลกันในรูปของ XML เพื่อให้เอกสารที่สร้างขึ้นมามีรูปแบบที่ตรงตามที่ตกลงกันไว้ และจะได้สามารถนำเอกสาร XML นั้นไปใช้งานอย่างถูกต้อง

การกำหนดโครงสร้างของเอกสาร XML นี้สามารถทำได้ 2 วิธีหลัก คือ

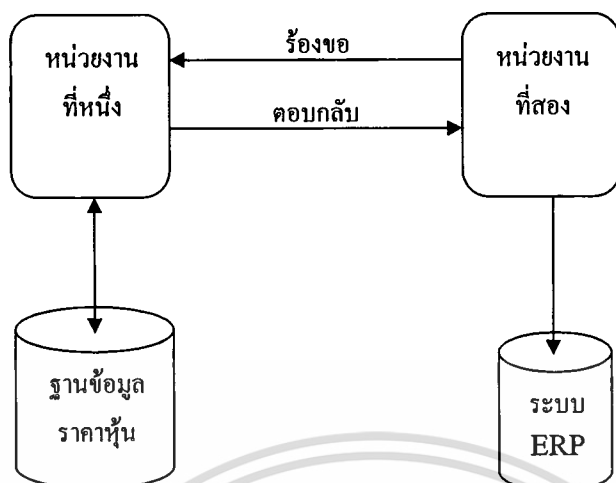
- DTD (Document Type Definition) ซึ่งรับมาจากภาษา SGML ทำให้ DTD กลายเป็นส่วนหนึ่งของมาตรฐาน XML ของ W3C

- XML Schema เป็นมาตรฐานของทางหน่วยงาน W3C ซึ่งไมโครซอฟต์ได้นำไปพัฒนาต่อแล้วตั้งชื่อใหม่เป็น XDR (XML Data Reduced) ซึ่งได้นำไปใช้กับผลิตภัณฑ์ของไมโครซอฟต์อย่างเช่น IE และ SQL Server 2000 เป็นต้น

การกำหนดโครงสร้างของเอกสาร แบบ XML Schema มีข้อที่ได้เปรียบกว่า DTD คือ

- รองรับชนิดของข้อมูลไม่ต่ำกว่า 41 ชนิด
- เป็นชนิดข้อมูลที่รองรับในโปรแกรมฐานข้อมูลที่มีอยู่ในปัจจุบัน หมายความว่าสามารถนำข้อมูลที่อยู่ในรูป XML ไปเก็บไว้ในฐานข้อมูลได้ง่าย โดยพิจารณาจากชนิดข้อมูล
- กำหนดรูปแบบข้อมูล (pattern) เองได้ เช่น รูปแบบข้อมูลของหมายเลขโทรศัพท์เป็น “d-ddd-dddd” ให้ d แทนตัวเลข 1 ตัว
- สร้างชนิดข้อมูลใหม่โดยอิงจากชนิดของข้อมูลที่มีอยู่เดิม
- รองรับการใช้เนมสเปซ (namespace) ซึ่งมีไว้เพื่อป้องกันความสับสนในการระบุชื่อองค์ประกอบใดๆ ภายในเอกสาร XML คนละไฟล์ แต่ใช้ชื่อเดียวกันเพราะอาจเป็นไปได้ว่าเอกสาร XML ที่ใช้ในงานที่ต่างกัน มีชื่ออติเมนต์เหมือนกัน ชื่อชนิดข้อมูลเหมือนกัน แล้วอาจจะถูกเรียกใช้ภายในเอกสาร XML ไฟล์เดียวกัน สำหรับชื่อดีของ XML มีดังนี้
- ใช้สำหรับสร้างข้อมูลที่สามารถอธิบายความหมายของตัวเองได้ (self-describe data) จากความสามารถในการสร้างแท็กขึ้นมาอธิบายข้อมูลที่อยู่ภายในแท็ก ทำให้ข้อมูลนั้นมีความหมายอยู่ในตัวเอง เป็นข้อมูลที่สามารถเขียนโปรแกรมดึงข้อมูลไปใช้ได้โดยง่าย และแม้แต่มนุษย์ก็สามารถอ่านได้ด้วย ดังนั้นจึงสามารถกล่าวได้ว่า เอกสาร XML มีคุณสมบัติครบทั้งแบบ machine readable และ human readable
- ใช้สำหรับการแลกเปลี่ยนข้อมูล (data exchange) ด้วยความที่เป็นไฟล์ข้อความธรรมดา นี้เอง ทำให้ออกสาร XML เป็นภาษากลางที่ใช้ได้ในทุกแพลตฟอร์ม (platform) ไม่ว่าจะเป็นแพลตฟอร์มตระกูล WINDOWS หรือตระกูล UNIX หรืออื่นๆ ตัวอย่างเช่น ในภาพที่ 2.3 หน่วยงานที่ 2 ขอข้อมูลราคาหุ้นจากหน่วยงานที่ 1 ด้วยวิธีแลกเปลี่ยนข้อมูลในรูปแบบ XML โดยหน่วยงานที่ 1 จะดึงข้อมูลจากฐานข้อมูล แล้ว markup ด้วยแท็กที่กำหนดขึ้นมา เช่น <PriceQuote> 99.50 </PriceQuote> เป็นต้น จากนั้นเมื่อหน่วยงานที่ 2 ได้รับข้อมูลในรูปแบบของเอกสาร XML ก็จะสร้างโปรแกรมขึ้นมาดึงข้อมูลจริงๆ ที่อยู่ในแท็ก XML ไปใช้งานของตน ซึ่งในตัวอย่างนี้เป็นการนำไปใช้ในระบบ ERP (Enterprise Resource Planning)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.3 ตัวอย่างการแลกเปลี่ยนข้อมูลในรูปแบบ XML

- เป็นรูปแบบข้อความในการสื่อสาร (message format) ระหว่างแอปพลิเคชันหรือโปรแกรม XML ได้ถูกนำไปใช้กับเทคโนโลยีที่เรียกว่า Web Service แนวความคิดนี้คือการเตรียมซอฟต์แวร์สำหรับให้บริการทำงานบางอย่าง อยู่ในเซิร์ฟเวอร์เครื่องใดเครื่องหนึ่งภายในเครือข่ายอินเทอร์เน็ต โดยผู้ใช้ทั่วไปสามารถเรียกใช้บริการนั้นได้
- ประโยชน์ในเชิงเทคโนโลยีอินเทอร์เน็ตและการพัฒนาเว็บ
- เป็นรากฐานของภาษาใหม่ๆ ในการพัฒนาเว็บ ภาษาใหม่ๆ ในที่นี้ก็คือเช่น ภาษา XHTML, MathML, VML, WML และอีกมากมาย
- ใช้ในวงธุรกิจแบบ B2B (Business to Business) กรณีนี้ต้องใช้ภาษาเฉพาะ อย่างเช่น cXML (Commerce XML), xCBL (XML Common Business Language) เป็นต้น โดยมีแท็กที่ใช้สนับสนุนการจัดการเกี่ยวกับแคตตาล็อกสินค้า (catalog) และธุรกรรมเกี่ยวกับ e-Commerce ปัจจุบันในบางประเทศได้มีการนำ XML มาใช้แทนระบบ EDI (Electronic Data Interchange) บ้างแล้ว ซึ่งเป็นการแลกเปลี่ยนเอกสารอิเล็กทรอนิกส์ระหว่างองค์กร

2.4 Web Service

Web Service เป็นแอปพลิเคชันหรือโปรแกรมที่เก็บอยู่บนเครื่องคอมพิวเตอร์ที่สามารถติดต่อผ่านทางอินเทอร์เน็ตได้ แอปพลิเคชันหรือโปรแกรมนี้ทำหน้าที่ให้บริการอย่าง

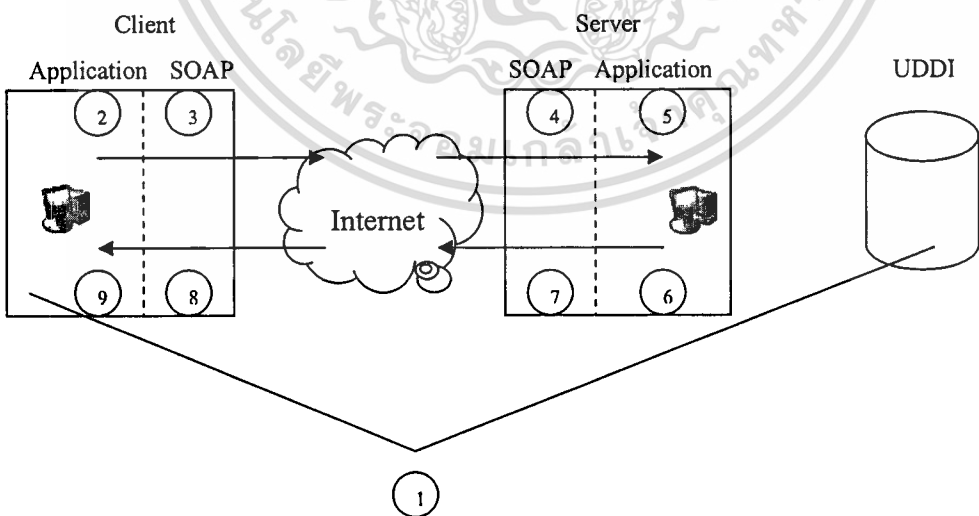
โดยอย่างหนึ่งตามที่ผู้ใช้ต้องการ โดยมีเอกสารอธิบายคุณสมบัติของบริการนั้นกำกับอยู่ และจะต้องมีข้อมูลแจ้งให้ผู้ใช้บริการทราบว่ามิบริการนี้อยู่ ผู้ใช้บริการสามารถค้นหาเว็บเซอร์วิสเหล่านั้นได้ถึงแม้จะไม่ทราบแหล่งที่เก็บเว็บเซอร์วิสนั้นแน่นอนก็ตาม

แนวคิดที่ใกล้เคียงกับเว็บเซอร์วิสนี้มีมานานแล้ว แต่อยู่ในเครื่องที่เป็น stand alone โดยทางไมโครซอฟต์ได้เสนอแนวคิด COM (Component Object Model) ไว้ก่อนแล้ว โดย COM เหล่านี้อยู่เบื้องหลังการทำงานของซอฟต์แวร์ หรือแอปพลิเคชันที่ต่างๆที่ติดตั้งลงในระบบปฏิบัติการ WINDOWS เนื่องจาก COM เป็นมาตรฐานในการสร้างซอฟต์แวร์หรือแอปพลิเคชันให้สามารถทำงานร่วมกันได้ ซึ่งเมื่อทำการติดตั้งซอฟต์แวร์จะมีไฟล์ที่มีนามสกุล dll อยู่ในระบบเป็นจำนวนมาก ไฟล์ dll เหล่านี้สร้างตามมาตรฐาน COM เพื่อให้ซอฟต์แวร์ที่ติดตั้งนี้ทำงานร่วมกับ WINDOWS ได้นั่นเอง การสร้างไฟล์ที่เป็น dll หรือ exe ให้เป็นไปตามมาตรฐาน COM นี้เอง อาจเรียกว่าเป็น COM component ซึ่งโปรแกรมเมอร์สามารถสร้างขึ้นมาได้เองโดยใช้ภาษาพวก Visual Basic, Visual C++, Visual J++ หรือ Delphi รวมทั้ง Visual Studio .NET ก็ได้ และ COM component เมื่อนำไปใช้งานบนอินเทอร์เน็ตจึงเรียกว่าเว็บเซอร์วิสนั่นเอง

ดังนั้นเว็บเซอร์วิสนี้ก็คือการนำเทคโนโลยี COM มาเพิ่มเติมความสามารถ โดยอาศัยอินเทอร์เน็ตเป็นตัวเสริม เว็บเซอร์วิสเป็นการเรียกใช้มรดกของคอมพิวเตอร์ที่ให้บริการผ่านทางเว็บโดยผู้ใช้บริการไม่ทราบเลยว่าคอมพิวเตอร์นั้นอยู่ที่เครื่องใด ประเทศไหน หรือใช้แพลตฟอร์มอะไร คอมพิวเตอร์ที่ว่านี้ก็คือแอปพลิเคชันที่ให้บริการบนอินเทอร์เน็ตนั่นเอง แต่เดิมการเรียกใช้คอมพิวเตอร์ข้ามแพลตฟอร์มจะมีข้อจำกัด อย่างเช่น DCOM (Distributed Component Object Model) ก็เรียกใช้งานได้เฉพาะแพลตฟอร์มของไมโครซอฟต์ หรือ RMI (Remote Method Invocation) ก็สามารถเรียกใช้งานคอมพิวเตอร์ที่พัฒนาบนจาวาเท่านั้น หรือแม้แต่ CORBA/IIOP (Common Object Request Broker Architecture/Internet Inter-ORB Protocol) ที่ไม่ขึ้นกับแพลตฟอร์มหรือภาษา แต่การใช้งานก็ค่อนข้างยุ่งยากเนื่องจากต้องมีการปรับแต่งคอมพิวเตอร์ทั้งฝั่งที่ให้บริการและฝั่งที่ให้บริการ และจะมีปัญหาเกี่ยวกับ firewall ที่มักไม่ให้ยอมผ่าน (เนื่องจาก firewall ยอมให้การสื่อสารผ่านโปรโตคอล HTTP ผ่านเข้าออกระบบทำได้สะดวก แต่โปรโตคอลอื่นถือเป็นสิ่งแปลกปลอม firewall จึงไม่ยอมให้ผ่าน) ดังนั้นจึงได้มีการเสนอโปรโตคอลตัวใหม่ขึ้นมาที่เรียกว่า SOAP (Simple Object Access Protocol) เพื่อให้สามารถทำงานข้ามแพลตฟอร์มของแอปพลิเคชันต่างๆ ได้

ขั้นตอนการเรียกใช้เว็บเซอร์วิส (ดังภาพที่ 2.4) มีดังนี้

- 1) ผู้ใช้บริการค้นหาเว็บเซอร์วิสที่ต้องการผ่านทาง UDDI (Universal Description Discovery and Integration) ซึ่ง UDDI นี้จะทำหน้าที่คล้ายกับเว็บไซต์จำพวก search engine ทำให้ผู้ใช้บริการทราบถึงชื่อ และรายละเอียดต่างๆของเว็บเซอร์วิส เช่นประเภทของเซอร์วิส, ที่ตั้งของเซอร์วิสในรูปแบบเอกสาร WSDL (Web Service Description Language) ถ้าผู้ใช้บริการทราบแล้วว่าเซอร์วิสนั้นเก็บอยู่ที่ใดก็ไม่จำเป็นต้องค้นหาผ่านทาง UDDI
- 2) ผู้ใช้บริการฝั่งไคลเอ็นต์เขียนคำสั่งเพื่อเรียกใช้เมธอดของเว็บเซอร์วิสนั้น โดยใช้ข้อมูลจากเอกสาร WSDL ที่ได้จากข้อ 1
- 3) SOAP ฝั่งไคลเอ็นต์นำคำสั่งที่เขียนในข้อ 2 มาแปลงให้อยู่ในรูป SOAP message แล้วส่งผ่านโปรโตคอล HTTP ไปยังเว็บเซิร์ฟเวอร์ที่เป็นเว็บเซอร์วิสที่ต้องการนั้น
- 4) SOAP ฝั่งเซิร์ฟเวอร์แปลง SOAP message ฝั่งไคลเอ็นต์ให้อยู่ในรูปแบบที่ระบบปฏิบัติการของเครื่องเซิร์ฟเวอร์เข้าใจ
- 5) หลังจากนั้นส่ง SOAP message ที่แปลงแล้วนั้นไปให้กับเว็บเซอร์วิสเพื่อประมวลผล
- 6) ส่งผลลัพธ์นั้นกลับไป SOAP ของฝั่งเซิร์ฟเวอร์
- 7) แปลงผลลัพธ์เป็น SOAP message แล้วส่งผ่าน โปรโตคอล HTTP กลับไปฝั่งไคลเอ็นต์
- 8) ไคลเอ็นต์แปลง SOAP message ที่รับมาจากฝั่งเซิร์ฟเวอร์ให้อยู่ในรูปแบบที่ระบบปฏิบัติการของเครื่องไคลเอ็นต์เข้าใจ หลังจากนั้นส่งต่อไปให้ผู้ให้บริการ
- 9) ผู้ใช้บริกรนำเว็บเซอร์วิสที่ได้ไปใช้งานตามต้องการ



ภาพที่ 2.4 ขั้นตอนการเรียกใช้เว็บเซอร์วิส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 .NET Framework

สิ่งที่สำคัญของสถาปัตยกรรม .NET ก็คือโค้ดในรูปแบบ VB และ C# จะถูกคอมไพล์เป็น Intermediate Language (IL) ไม่ใช่โค้ด native เช่นแอปพลิเคชันเดิมๆ แต่ขั้นตอนสุดท้ายของการแปลงเป็นโค้ดที่เอ็กซ์ิควิตได้ โดยปกติจะอยู่ในช่วงที่รัน โค้ดที่เป็น managed code C++ จะมีอปชันให้เลือกเป็น managed code ซึ่งสามารถถูกคอมไพล์เป็น IL ด้วย สิ่งนี้เองทำให้โค้ด C++ สามารถใช้งานร่วมกับ VB และ C# ด้วย โดยแนวคิดเรื่อง IL นี้มีลักษณะการในการทำงานคล้ายกับ Java ที่มีการคอมไพล์เป็น 2 ขั้นตอน อย่างไรก็ตาม Java เน้นที่ความเป็นอิสระข้ามแพลตฟอร์ม (cross-platform independence) ในขณะที่ .NET มุ่งเน้นไปที่ WINDOWS และมุ่งเน้นไปที่ความเป็นอิสระข้ามภาษา (cross-language)

Common Language Runtime (CLR)

ลักษณะของ CLR จะคล้ายกับ JVM ของ Java แต่ใช้รีซอร์สน้อยกว่าของ Java เนื่องจากการคอมไพล์บน JIT ทำให้มีการจัดการรีซอร์สที่ดีขึ้น โดยเฉพาะการจัดการหน่วยความจำที่ทำได้ดีทั้งระหว่างแอปพลิเคชัน และในระบบปฏิบัติการ สำหรับการทำให้เป็นแพลตฟอร์มที่เสถียรในเป้าหมายด้านความน่าเชื่อถือที่ต้องใช้ในการติดต่อแอปพลิเคชันของธุรกิจบนเว็บนั้น CLR จะมีงานที่สัมพันธ์กัน เช่นการเฝ้าดูในระหว่างการเอ็กซ์ิควิตโปรแกรม ซึ่งจะเป็น managed code สำหรับโปรแกรมที่รันภายใต้การเฝ้าติดตามของ CLR แต่สำหรับแอปพลิเคชันหรือคอมโพเนนต์จะเป็น unmanaged code ซึ่งรันใน native mode ภายนอก CLR

CLR จะเฝ้าติดตามข้อผิดพลาดของโปรแกรมที่เป็นปัญหาใหญ่และเกิดขึ้นบ่อย เช่น การติดต่อสิ่งที่ยอยู่นอกขอบเขตของอาร์เรย์, ติดต่อหน่วยความจำในส่วน non-allocated memory หรือการเขียนหน่วยความจำอันเนื่องมาจากขนาดที่ใหญ่เกินไป เป็นต้น การพัฒนา .NET ที่ใช้ CLR เฝ้าติดตามแอปพลิเคชันนั้น จะทำให้มีประสิทธิภาพสูงขึ้นอย่างน้อย 10 %

สิ่งที่จะได้รับจาก .NET

- เขียนโปรแกรมน้อยลง บนแพลตฟอร์ม .NET เน้นการออกแบบโดยใช้คอมโพเนนต์อย่างมาก นักพัฒนาจึงให้ความสนใจกับการเขียนแอปพลิเคชันโดยนำคอมโพเนนต์ที่มีอยู่แล้วมาใช้งานให้เหมาะสม ตัวอย่างเช่น ASP.NET ได้รวบรวมคอมโพเนนต์ต่างๆ ไว้เป็นจำนวนมาก ซึ่งเป็นการทำงานที่โปรแกรมเมอร์ต้องเขียนบ่อยๆ เช่น ระบบเงินสินค้าสำหรับซื้อของทางอินเทอร์เน็ต (shopping cart) เป็นต้น
- รันแอปพลิเคชันที่น่าเชื่อถือได้มากขึ้น .NET ได้รวบรวมเทคโนโลยีเพื่อให้แอปพลิเคชันมีความเชื่อถือได้มากขึ้น ตัวอย่างเช่น หน่วยความจำ, เธรด และโปรเซส จะถูกจัดการโดย .NET Framework เพื่อให้แน่ใจได้ว่า จะไม่มีเหตุการณ์การรั่วไหลของ

หน่วยความจำเกิดขึ้น ASP.NET จะติดตามการทำงานของเว็บแอปพลิเคชันที่ทำงานอยู่ และสามารถเริ่มทำงานแอปพลิเคชันใหม่ได้โดยอัตโนมัติภายในเวลาที่ผู้บริหารระบบกำหนดไว้

- ประสิทธิภาพที่ดีขึ้น .NET จะช่วยปรับปรุงประสิทธิภาพของแอปพลิเคชันบนเว็บให้ดีขึ้น โดย ASP.NET ได้รวมคุณสมบัติในการคอมไพล์และการทำแคชระดับสูงไว้ ซึ่งช่วยปรับปรุงประสิทธิภาพขึ้น 2-3 เท่าเมื่อเทียบกับแอปพลิเคชัน Active Server Pages ที่มีอยู่
- ทำงานได้เร็วขึ้น การเขียนโปรแกรมที่ทำให้สามารถรันบนเครื่องใดก็ได้ไม่ใช่ของใหม่ ตัวอย่างเช่น Java ที่ภายหลังจากการคอมไพล์เป็น .class แล้ว สามารถนำไปรันบนเครื่องใดก็ได้ที่มีการติดตั้ง JVM (Java Virtual Machine) ซึ่ง JVM นี้ทำหน้าที่เป็น Interpreter ที่ละคำสั่งทำให้เมื่อรันโปรแกรมจะทำให้ช้าลงเป็นอย่างมาก แต่ .NET Framework จะคอมไพล์โปรแกรมเป็น Machine code อย่างแท้จริงที่เรียกว่า Intermediate Language (IL) แต่การคอมไพล์นั้นจะไม่ทำรวดเดียวตั้งแต่ต้นจนจบ เนื่องจากจะใช้เวลามาก โดยไมโครซอฟต์ได้ใช้เทคโนโลยีในการคอมไพล์ที่เรียกว่า Just In Time (JIT) ซึ่งจะคอมไพล์เฉพาะส่วนที่จะใช้งานก่อนแล้วจึงรัน หลังจากนั้นถ้าต้องการใช้งานส่วนใดก็จะคอมไพล์เพิ่มเติม โดยไม่คอมไพล์ซ้ำส่วนที่คอมไพล์ไปแล้วทำให้การทำงานเร็วขึ้นมาก
- เครื่องมือพร้อมในการพัฒนาแอปพลิเคชัน โดยใช้ Visual Studio.NET ไมโครซอฟต์ตั้งชื่อ .dll เป็น Assembly ซึ่งใน Assembly จะประกอบด้วยโค้ดและข้อมูลเกี่ยวกับ Assembly และที่เกี่ยวข้องกับอินเทอร์เฟซต่างๆ (ส่วนของข้อมูลนี้เรียกว่า MetaData) เนื่องจาก Assembly จะรวมทุกสิ่งทุกอย่างไว้ ทำให้เมื่อทำงานกับ .NET จะมีการแสดง MetaData ออกมาในลักษณะ Intellisense เช่น เมื่อคีย์คำสั่ง จะปรากฏพารามิเตอร์ทั้งหมดออกมาให้เลือกใช้ นอกจากนี้ใน Visual Studio .NET ยังได้จัดเตรียมเครื่องมือไว้มากมายให้เลือกใช้ ทำให้นักพัฒนาโปรแกรมทำได้ง่ายขึ้นมาก
- .NET สนับสนุนโปรแกรมที่หลากหลาย ภาษาต่างๆที่สามารถนำมาใช้งานได้คือ C++, Jscript, VB.NET, C# ที่เป็นภาษาใหม่มาพร้อมกับ .NET นอกจากนี้บริษัทคู่ค้าของไมโครซอฟต์ก็กำลังพัฒนาคอมไพเลอร์สำหรับโปรแกรมภาษาที่ใช้งานอย่างแพร่หลาย ไม่ว่าจะเป็น COBOL, Eiffel, Lisp, Python และ Smalltalk Rational รวมทั้งบริษัทที่มีชื่อเสียงด้าน UML ก็กำลังสนใจสร้างคอมไพเลอร์ Java บน .NET อีกด้วย รวมแล้วภาษาโปรแกรมที่สนับสนุนใน .NET รวม 27 ภาษา

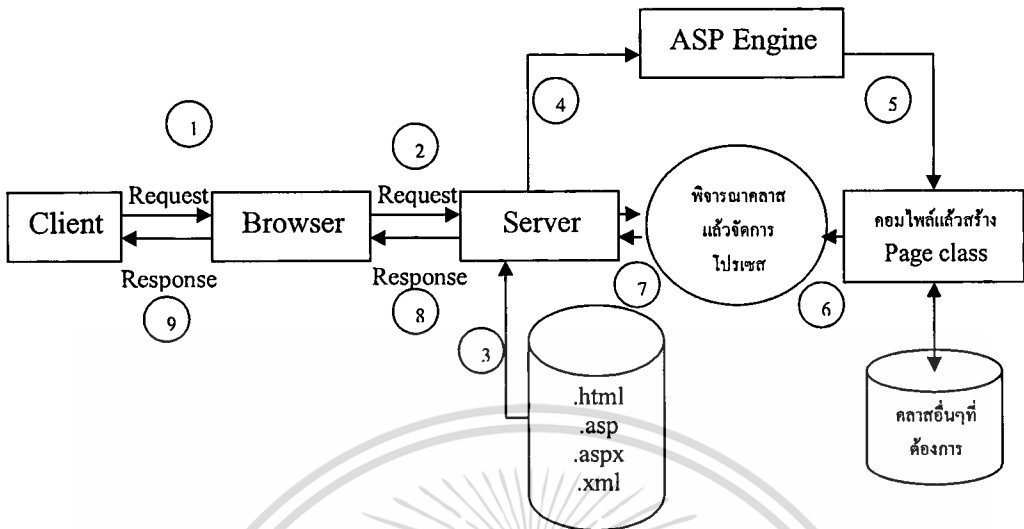
2.6 ASP.NET

เทคโนโลยีใหม่ในการสร้างเว็บเพจแบบไดนามิกมีการเปลี่ยนรูปแบบการเขียนใหม่ตามพื้นฐานเซอร์วิสของ CLR ภาษาใดก็ตามที่ได้รับการสนับสนุนจาก .NET ก็จะสามารถนำไปใช้ในเว็บเพจของ ASP.NET ได้ เว็บเพจตามเทคโนโลยี .NET จะมีสิ่งที่แตกต่างจาก ASP 3.0 โดยเพจใหม่ที่จะสร้างขึ้นจะใช้ส่วนขยายเป็น .aspx ในขณะที่ เว็บเซอร์วิสจะใช้ส่วนขยาย .asmx โดยที่แอปพลิเคชัน .NET ยอมให้มีเพจที่มีส่วนขยาย .asp และ aspx ร่วมกันได้ โดยที่เพจที่มีรูปแบบเดิม (asp) จะรันผ่านทาง dll ที่ชื่อ asp.dll แต่จะไม่ได้ใช้ประโยชน์จาก CLR ปัจจุบันนี้เพจที่มีส่วนขยายเป็น .aspx จะไม่มีการแปล (interpret) แต่จะคอมไพล์เป็นโค้ด MSIL หลังจากนั้นจะรันจาก intermediate code ที่สร้างขึ้นมา

หลักการการทำงานของ ASP.NET

เนื่องจาก ASP.NET จะเอ็กซิคิวต์ที่ฝั่งเซิร์ฟเวอร์ (Server Side) ส่วนการทำงานของบราวเซอร์เรียกว่าเป็นการทำงานฝั่งไคลเอ็นต์ (Client side) หลักการทำงานของ ASP.NET จะเป็นไปตามขั้นตอน (ดังภาพที่ 2.5) ดังนี้

1. ผู้ใช้ส่งคำร้องขอ (Request) จากไคลเอ็นต์ด้วยเพจ aspx ผ่านทางบราวเซอร์
2. บราวเซอร์ส่งคำร้องขอไปยังเซิร์ฟเวอร์ (Web Server)
3. เซิร์ฟเวอร์หาเพจที่ต้องการจากดิสก์ ถ้าเป็นเพจประเภทอื่นจะแสดงผล ถ้าเป็นเพจ aspx จะข้ามไปขั้นตอนที่ 4
4. ถ้าเป็นเพจที่เป็น aspx จะส่งไปยัง ASP Engine
5. ASP Engine ทำการคอมไพล์แล้วสร้าง Page Class ที่ต้องการ (ถ้า Page Class นั้นที่อยู่แล้วจะรันทันที) ถ้าในระหว่างการคอมไพล์ต้องการคลาสอื่นๆ เช่น code-behind class หรือคอมโพเนนต์ก็จะรวมกันในขั้นตอนนี้
6. ASP.NET พิจารณาคลาสต่างๆแล้วทำการ โปรเซส
7. ส่งผลลัพธ์ไปที่เซิร์ฟเวอร์
8. สร้างออปเจ็กต์ Response
9. เซิร์ฟเวอร์ส่งออปเจ็กต์ Response ไปยังไคลเอ็นต์



ภาพที่ 2.5 การทำงานของ ASP.NET

ส่วนที่เปลี่ยนจาก ASP ไปเป็น ASP.NET

- เปลี่ยนโครงสร้างภาษาหลักที่ใช้ทั้งหมด แต่เดิมภาษาที่ใช้ร่วมกับ ASP จะเน้นที่ VBScript เป็นหลัก แต่โปรแกรมเมอร์อาจจะใช้ Javascript หรือ Jscript เพื่อทำให้สคริปต์ ASP เหล่านั้นแทรกอยู่ใน HTML เพื่อทำงานตามความต้องการของโปรแกรมเมอร์ เมื่อพัฒนาเป็น ASP.NET ความสามารถของสคริปต์เหล่านี้ไม่เพียงพอที่จะเขียนแอปพลิเคชัน แต่ควรเป็นภาษาชั้นสูงที่เป็นภาษาเชิงวัตถุ (OOP) อย่างเช่น Visual Basic.NET หรือ C# โดยคอมไพเลอร์ทั้ง 2 ภาษานี้ติดมากับ Visual Studio .NET
- การคอมไพล์โค้ดอยู่ในรูปแบบ IL (Intermediate Language) แต่เดิมโค้ด ASP จะคอมไพล์ในลักษณะแปลภาษา (Interpret) ที่ละบรรทัดตั้งแต่ต้นจนจบ ตรงนี้เองสนับสนุนการทำงานของ .NET ตรงที่ว่าไม่ว่าจะเขียนโปรแกรมภาษาใดก็ตามสามารถคอมไพล์เป็น IL ได้ หรืออาจกล่าวได้ว่าสามารถคอมไพล์เพื่อให้ CLR (Common Language Runtime) ขอมรับได้นั่นเอง
- ตัวคอนโทรลทำให้การสร้างเว็บแอปพลิเคชันง่ายขึ้น นอกเหนือจากภาษา VB.NET และ C# ที่ใช้ในการเขียนโปรแกรมแล้ว ไมโครซอฟต์ยังได้พัฒนาสิ่งที่เรียกว่า “ตัวคอนโทรล” (Control) ซึ่งคอนโทรลเหล่านี้เป็นลักษณะ เซิร์ฟเวอร์คอนโทรล (Server Control) เช่น สมมติว่าต้องการเขียนสคริปต์เพื่อนำข้อมูลในฐานข้อมูลมาแสดงใน

รูปแบบตารางของ HTML โปรแกรมเมอร์จะต้องกำหนดส่วนอื่นๆของตารางเพื่อให้เกิดความสวยงาม และทำการวนรอบการดึงข้อมูลตำแหน่งใด เป็นต้น แต่สำหรับ ASP.NET เพียงใช้ตัวคอนโทรลที่ชื่อ “DataGrid” เท่านั้น

- ปรับปรุงประสิทธิภาพการเชื่อมต่อฐานข้อมูล การปรับปรุงส่วนนี้เกิดจากการพัฒนา ADO (ActiveX Data Object) เป็น ADO.NET ซึ่งทางไมโครซอฟต์ได้พัฒนาหลายส่วนเพื่อการเชื่อมต่อฐานข้อมูลทำได้ง่าย ซึ่งส่งผลทำให้การแสดงผลจากฐานข้อมูลเหล่านั้นทำได้โดยมีประสิทธิภาพและรวดเร็ว
- เปลี่ยนวิธีการปรับแต่งเว็บเซิร์ฟเวอร์และลงทะเบียนคอมโพเนนต์ แต่เดิมการปรับแต่งคอนฟิกูเรชันของเว็บเซิร์ฟเวอร์ เช่น script timeout หรือ session timeout จะถูกเก็บไว้ที่เว็บเซิร์ฟเวอร์และไม่สามารถเปลี่ยนแปลงได้ แต่สำหรับ ASP.NET ค่าคอนฟิกูเรชันเหล่านี้จะเก็บไว้ในไฟล์ config.web ในลักษณะ XML แทน ในการแก้ไขอาจจะเป็นใครที่ไหนก็ได้ที่มีสิทธิ์ก็สามารถแก้ไขได้ ส่วนคอมโพเนนต์ใหม่ที่ยังไม่ได้ลงทะเบียนก็สามารถทำได้โดยการอัปโหลดคอมโพเนนต์นั้นไปที่เซิร์ฟเวอร์ ในโฟลเดอร์ bin เท่านั้น หลังจากนั้นคอมโพเนนต์จะถูกลงทะเบียนและพร้อมใช้งานได้โดยอัตโนมัติ

2.7 VB.NET

VB.NET ย่อมาจาก Visual Basic.NET เป็นภาษาคอมพิวเตอร์ที่นำเอาภาษา Visual Basic 6.0 มาพัฒนาโดยการเพิ่มความสามารถที่รองรับเทคโนโลยี .NET Framework เข้าไป ทั้งนี้ก็เพื่อให้ VB.NET เป็นภาษาคอมพิวเตอร์ที่สามารถใช้พัฒนาโปรแกรมทั้งที่อยู่บน Web และบนโปรแกรม Windows ของเครื่อง Client โดยที่คำสั่งต่างๆของ VB.NET ที่นำมาประกอบกันขึ้นเป็นโปรแกรม Server-Side Script จะต้องกำหนดอยู่ภายใน Block 2 รูปแบบด้วยกันคือ แบบ Inline Code Block และ Declarative Code Block

ตัวแปรของ VB.NET ก่อนที่จะนำมาใช้งาน จะต้องทำการประกาศ (Declare) พร้อมกับระบุประเภทของข้อมูลที่จะใช้จัดเก็บทุกครั้ง ประเภทของข้อมูลที่สามารถใช้งานใน VB.NET ได้แก่ Boolean, Byte, Char, Date, Decimal, Double, Integer, Long, Object, Short, Single และ String นอกจากตัวแปรโดยทั่วไปแล้ว ในกรณีที่ต้องการให้ตัวแปรจัดเก็บค่าใดค่าหนึ่งตลอดการใช้งานในโปรแกรมโดยไม่มีกรเปลี่ยนแปลงค่า ก็ยังสามารถกำหนดให้ตัวแปรนั้นเป็นตัวแปรประเภท Constant ในการจัดเก็บแทนตัวแปรทั่วไปได้เช่นเดียวกัน

ในภาษา VB.NET จะประกอบด้วยคำสั่งและฟังก์ชันพื้นฐานต่างๆดังนี้

- Operator ได้แก่ เครื่องหมายที่ใช้กระทำกับตัวแปรหรือค่าต่างๆ เพื่อให้ได้ผลลัพธ์ตามต้องการมีอยู่ด้วยกัน 5 กลุ่ม คือ Assignment Operator, Arithmetic Operators, Comparison Operators, Logical (Bitwise) Operators และ String Operator
- ส่วนที่ใช้ควบคุมลำดับการทำงานและทิศทางของโปรแกรมมีอยู่ 2 กลุ่มด้วยกันคือ กลุ่มคำสั่ง Iteration และกลุ่มคำสั่ง Conditional
- ฟังก์ชันต่างๆ ที่ช่วยให้การพัฒนาโปรแกรม Script กระทำได้ง่ายขึ้น ซึ่งสามารถแบ่งออกเป็น 4 กลุ่มคือ ฟังก์ชันที่เกี่ยวข้องกับข้อความ, ฟังก์ชันที่เกี่ยวข้องกับตัวเลข, ฟังก์ชันที่เกี่ยวข้องกับวันที่และเวลา และฟังก์ชันที่ใช้แปลงระหว่างตัวแปรต่างชนิดกัน สิ่งนี้ VB.NET ต่างจาก VB 6.0 ที่เห็นได้ชัดมีดังนี้
- มีการเพิ่มพีเจอรที่เป็น OOP เข้ามา ซึ่งใน VB 6.0 นั้นยังไม่มีพีเจอรลักษณะนี้ จึงทำให้ผู้ใช้ที่เคยใช้ VB 6.0 มาก่อน อาจจะต้องมาเรียนรู้ OOP เพิ่มเติม เพื่อให้สามารถใช้งาน VB.NET ได้อย่างมีประสิทธิภาพ
- ทำงานได้เร็วขึ้น เนื่องจาก .NET Framework จะคอมไพล์โปรแกรมเป็น Machine code ที่เรียกว่า Intermediate Language (IL) แต่การคอมไพล์นั้นจะไม่ทำรวดเดียวตั้งแต่ต้นจนจบ เนื่องจากจะใช้เวลามาก โดยไมโครซอฟต์ได้ใช้เทคโนโลยีในการคอมไพล์ที่เรียกว่า Just In Time (JIT) ซึ่งจะคอมไพล์เฉพาะส่วนที่จะใช้งานก่อนแล้วจึงรัน หลังจากนั้นถ้าต้องการใช้งานส่วนใดก็จะคอมไพล์เพิ่มเติม โดยไม่คอมไพล์ซ้ำส่วนที่คอมไพล์ไปแล้วทำให้การทำงานเร็วขึ้นมา

2.8 ADO.NET

ADO.NET เป็นเทคโนโลยีที่พัฒนาต่อเนื่องมาจาก ADO (ActiveX Data Object) เพื่อใช้งานทางด้านฐานข้อมูลใน .NET โดยทำหน้าที่เป็นตัวกลางในการติดต่อระหว่าง OLE DB Provider กับแอปพลิเคชันที่พัฒนาด้วยภาษาระดับสูงที่สนับสนุนใน .NET Framework เช่น Visual Basic.NET หรือ C# เป็นต้น

Namespace ทั้ง 6 ตัวของ ADO.NET มีดังนี้

- System.Data เป็น namespace หลักของ ADO.NET ที่ประกอบด้วยคลาสพื้นฐานที่ Provider ใช้งานสำหรับสถาปัตยกรรม ADO.NET โดยจะกำหนดคลาสที่แสดงตาราง, คอลัมน์, แถว และ recordset ตัวอย่างคลาสที่อยู่ใน namespace นี้คือ DataView, DataViewManager, DataSet, DataTable, DataRow, DataColumn และ DataRelation

- System.Data.Common เป็น namespace ที่กำหนดคลาสทั่วไปที่จำเป็นในการทำงานร่วมกับคลาสของ Provider คลาสในกลุ่มนี้ ได้แก่ DBConnection, DataAdapter, DbDataAdapter, DataColumnMapping และ DataTableMapping
- System.Data.OleDb เป็น namespace ที่กำหนดคลาสเพื่อใช้งานกับแหล่งข้อมูล OLE DB โดยใช้ .NET OleDb Data Provider ซึ่งการทำงานกับข้อมูลแบบ OLE DB นี้จะต้องมี OLE DB Provider สำหรับข้อมูลนั้น คลาสในกลุ่มนี้จะเริ่มต้นชื่อด้วย OleDb แล้วตามด้วยคอมโพเนนต์ เช่น OleDbConnection, OleDbDataAdapter, OleDbDataReader, OleDbCommandBuilder, OleDbError, OleDbParameter, OleDbPermission และ OleDbTransaction
- Microsoft.Data.Odbc เป็น namespace ที่คล้ายกับ System.Data.OleDb เพียงแต่ว่า namespace นี้จะกำหนดคลาสเพื่อทำงานกับ ODBC .NET Data Provider ในแหล่งข้อมูลที่เป็น ODBC ดังนั้นก่อนการใช้งานจะต้องติดตั้ง Driver สำหรับฐานข้อมูลแบบ ODBC ก่อน โดยปกติจะได้อาจมาจากกล่อง Microsoft Office คลาสในกลุ่มนี้จะเริ่มต้นด้วย Odbc แล้วตามด้วยคอมโพเนนต์ เช่น OdbcConnection, OdbcDataAdapter, OdbcDataReader, OdbcCommand, OdbcCommandBuilder, OdbcError, OdbcParameter, OdbcPermission และ OdbcTransaction
- System.Data.SqlClient เป็น namespace ที่กำหนดคลาสเพื่อทำงานกับ Sql .NET Data Provider ในฐานข้อมูลที่เป็น SQL Server 7.0 หรือใหม่กว่า ซึ่งจะคล้ายกับ ODBC และ OleDb คลาสในกลุ่มนี้จะเริ่มต้นด้วย Sql แล้วตามด้วยคอมโพเนนต์ เช่น SqlConnection, SqlDataAdapter, SqlDataReader, SqlCommand และ SqlTransaction
- System.Data.SqlTypes เป็น namespace ที่มีกลุ่มคลาสสำหรับ SQL Server โดยเฉพาะ คลาสในกลุ่มนี้ ได้แก่ SqlBinary, SqlMoney, SqlString, SqlDouble, SqlDateTime และ Sqlnumeric

Component ใน ADO.NET

ADO.NET เป็นชุดคอมโพเนนต์ชุดหนึ่ง que พัฒนาโดยบริษัท ไมโครซอฟต์ เพื่อการติดต่อกับฐานข้อมูล ใน ADO.NET แบ่งคอมโพเนนต์เป็น 2 กลุ่มคือ .NET Data Provider และ DataSet

1) .NET Data Provider

ในกลุ่ม .NET Data Provider เป็นกลุ่มคอมโพเนนต์ที่ใช้ในการจัดการกับข้อมูลในฐานข้อมูล ภายในจะมีออปเจกต์หลักๆ 4 ออปเจกต์คือ

- Connection เป็นออปเจกต์แรกที่ต้องกำหนดขึ้นมาก่อนที่จะใช้ออปเจกต์อื่น โดยทำหน้าที่ในการติดต่อกับฐานข้อมูล
- Command เป็นออปเจกต์ที่ใช้ในการจัดการกับข้อมูล เช่น อ่าน, เพิ่ม, แก้ไข หรือลบ ข้อมูลที่ติดต่อกับฐานข้อมูลแล้วจากออปเจกต์ Connection
- DataReader เป็นออปเจกต์ที่เป็นผลลัพธ์ของออปเจกต์ Command
- DataAdapter เป็นออปเจกต์ที่เป็นตัวกลางในการติดต่อระหว่าง .NET Data Provider และ DataSet เพื่อให้สามารถนำ DataSet มาใช้ภายใต้ออปเจกต์ Connection ได้

2) DataSet

ส่วนนี้เป็นคอมโพเนนต์ที่ใช้ติดต่อและจัดการกับข้อมูลทั่วไป และข้อมูลที่อยู่ในรูปของ XML ภายในกลุ่มจะมี 2 คอลเล็กชันคือ DataTableCollection ที่ทำหน้าที่แทนข้อมูลในแต่ละเรกคอร์ด, ฟیلด์ รวมทั้งโครงสร้างของข้อมูลเช่น คีย์หลัก, คีย์รอง, กฎ constraint ต่างๆ ส่วนอีกคอลเล็กชันคือ DataRelationCollection ซึ่งจะทำหน้าที่แทนความสัมพันธ์ระหว่างข้อมูล



บทที่ 3

การวิเคราะห์และออกแบบระบบงาน

3.1 โครงสร้างระบบ

โครงสร้างของระบบร้านหนังสือบนอินเทอร์เน็ต ประกอบด้วย 2 ส่วนหลักคือ

1. ส่วนของหน้าร้าน (Store Front) เป็นรูปแบบที่เหมือนกับการจัดวางสินค้าทั่วไป การออกแบบ ส่วนของหน้าร้านนั้นต้องให้สอดคล้องกับความต้องการของกลุ่มเป้าหมาย เนื่องจากส่วนของหน้าร้านนั้นเป็นสิ่งแรกที่จะนำเสนอต่อลูกค้า จำเป็นต้องมีความน่าสนใจและสามารถดึงดูดให้ผู้ที่พบเห็นเข้ามาซื้อสินค้า อีกทั้งยังเป็นส่วนให้บริการต่างๆแก่ลูกค้าที่เป็นสมาชิกด้วย โดยในส่วนของหน้าร้านจะประกอบด้วย

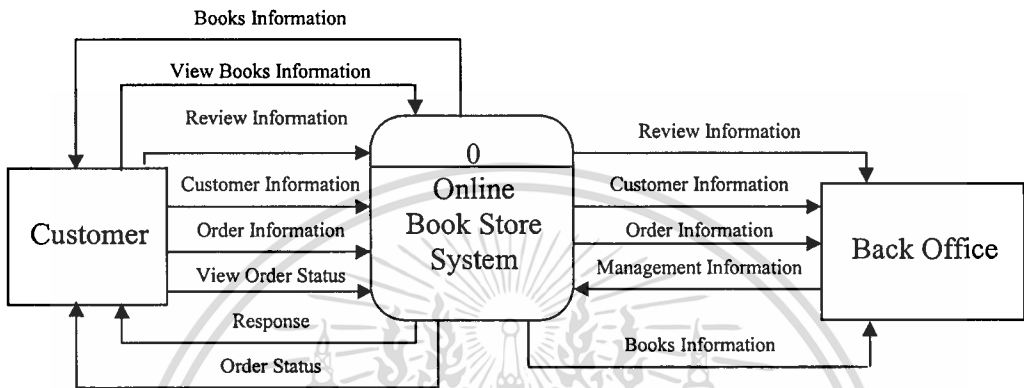
- การสร้างรายการสินค้า
- การสร้างระบบค้นหาหนังสือ
- การโฆษณาประชาสัมพันธ์หนังสือ รายการหนังสือที่ขายดีประจำสัปดาห์
- การสร้างระบบรถเข็นอิเล็กทรอนิกส์ (Shopping Cart)
- การสร้างระบบติดตามผลการสั่งซื้อ (Tracking System)
- การสร้างส่วนบริการลูกค้า (Customer Service)
- การสร้างระบบวิจารณ์หนังสือ

2. ส่วนของหลังร้าน (Back Office) เป็นส่วนที่ใช้ในการปรับปรุง แก้ไขและลบข้อมูล ต่างๆ ส่วนของหลังร้านจะประกอบไปด้วย

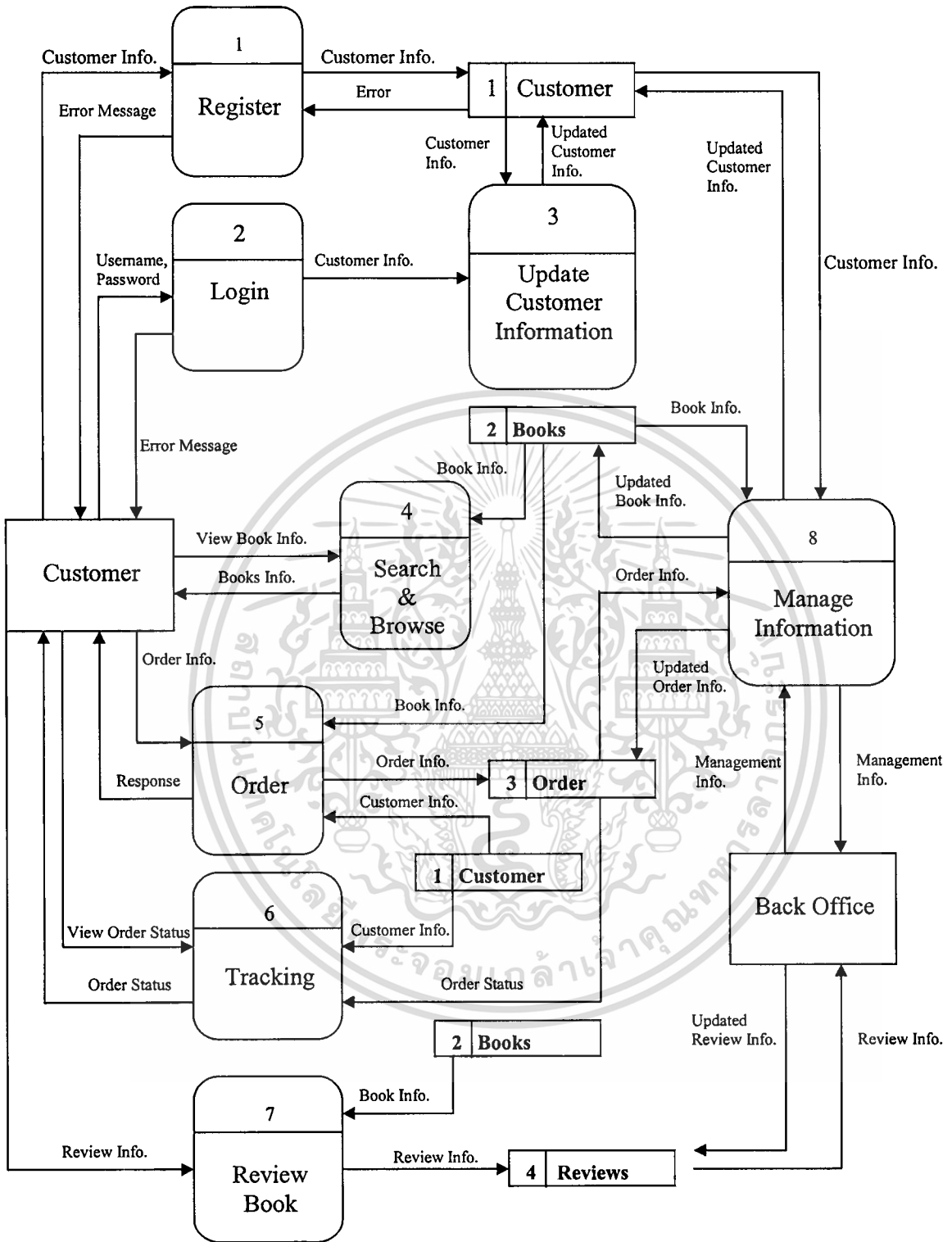
- การปรับปรุง แก้ไขและลบข้อมูล Customers
- การปรับปรุง แก้ไขและลบข้อมูล Books
- การปรับปรุง แก้ไขและลบข้อมูล Orders
- การปรับปรุง แก้ไขและลบข้อมูล Reviews

3.2 การวิเคราะห์ระบบ

ภาพรวมของการทำงานของระบบสั่งซื้อหนังสือบนอินเทอร์เน็ต จะแสดงในรูปแบบของ Context Diagram ดังภาพที่ 3.1 ส่วนการแสดงกระแสนการไหลของข้อมูล จะแสดงโดย Data Flow Diagram Level 1 ซึ่งจะแสดงให้เห็นถึงข้อมูลต่างๆที่สัมพันธ์กันในแต่ละการทำงาน โดยจะแสดงในภาพที่ 3.2

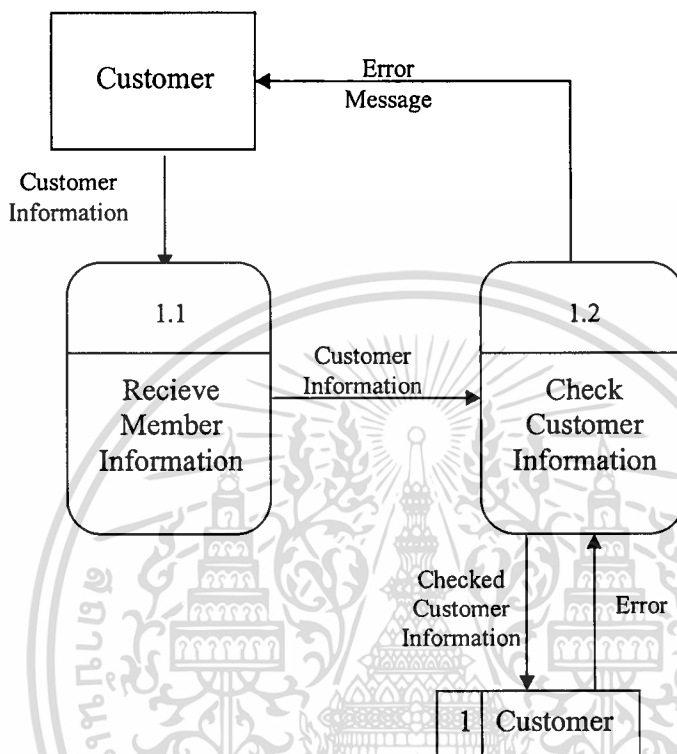


ภาพที่ 3.1 Context Diagram ของระบบร้านค้าหนังสือบนอินเทอร์เน็ต

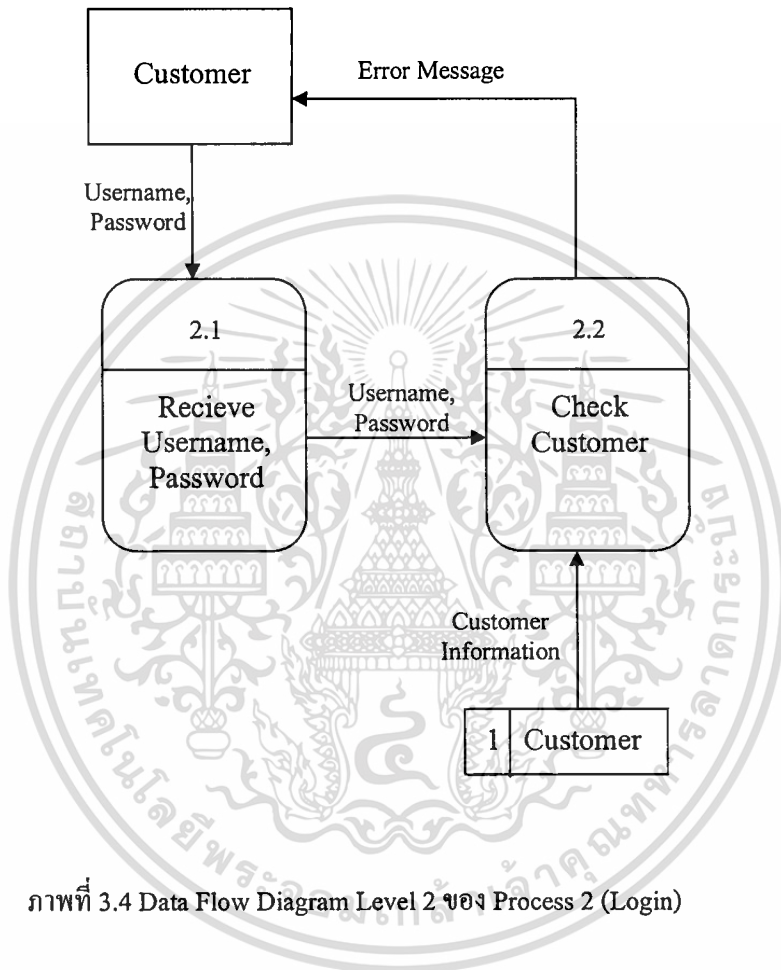


ภาพที่ 3.2 Data Flow Diagram Level 1 ของระบบร้านค้าหนังสือบนอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

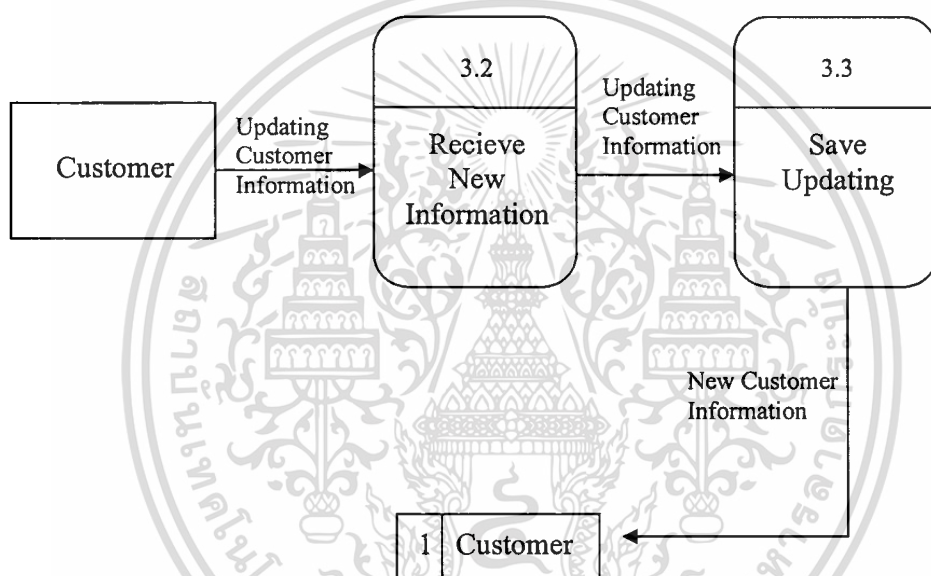


ภาพที่ 3.3 Data Flow Diagram Level 2 ของ Process 1 (Register)



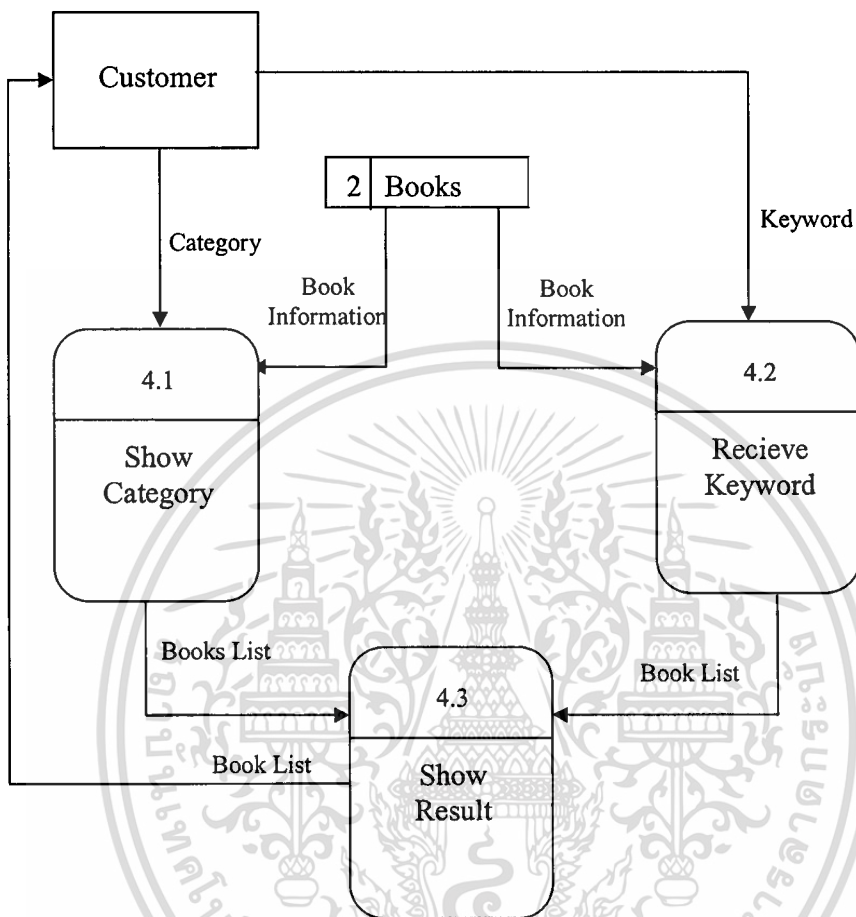
ภาพที่ 3.4 Data Flow Diagram Level 2 ของ Process 2 (Login)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



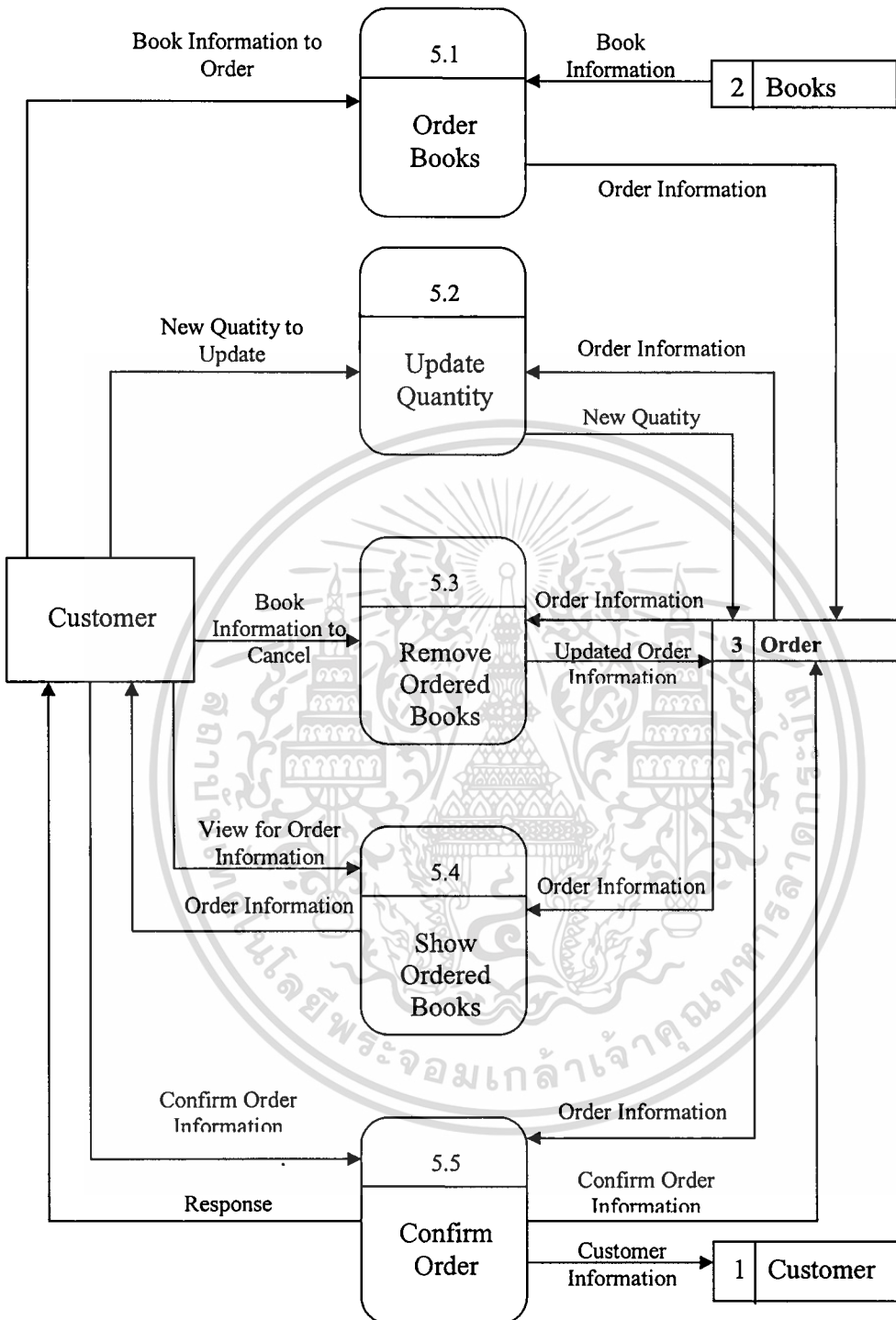
ภาพที่ 3.5 Data Flow Diagram Level 2 ของ Process 3 (Change Customer Information)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



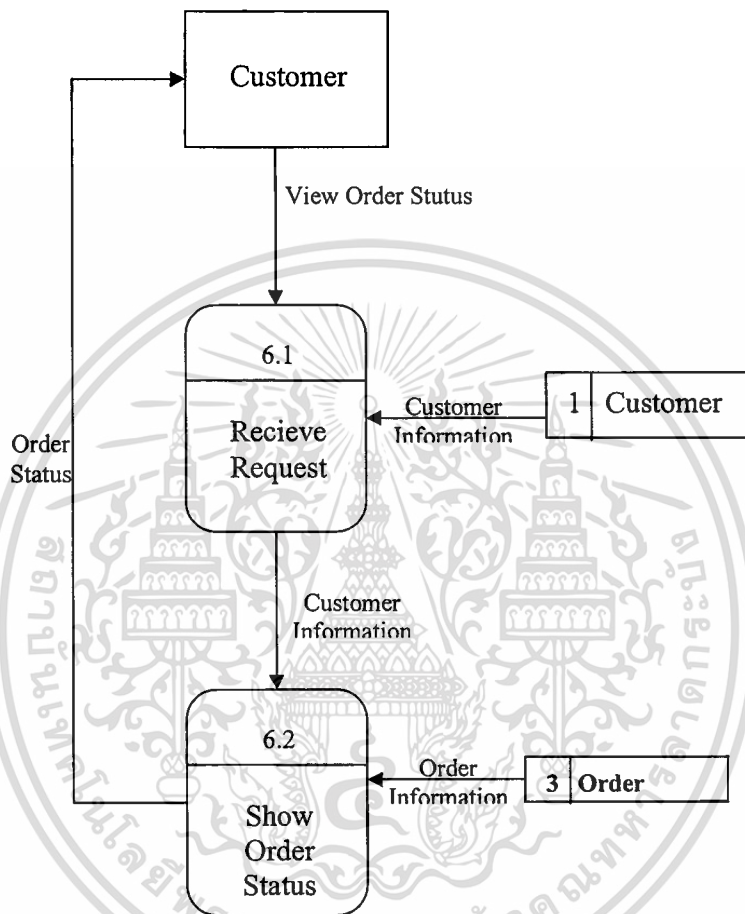
ภาพที่ 3.6 Data Flow Diagram Level 2 ของ Process 4 (Search & Browse)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



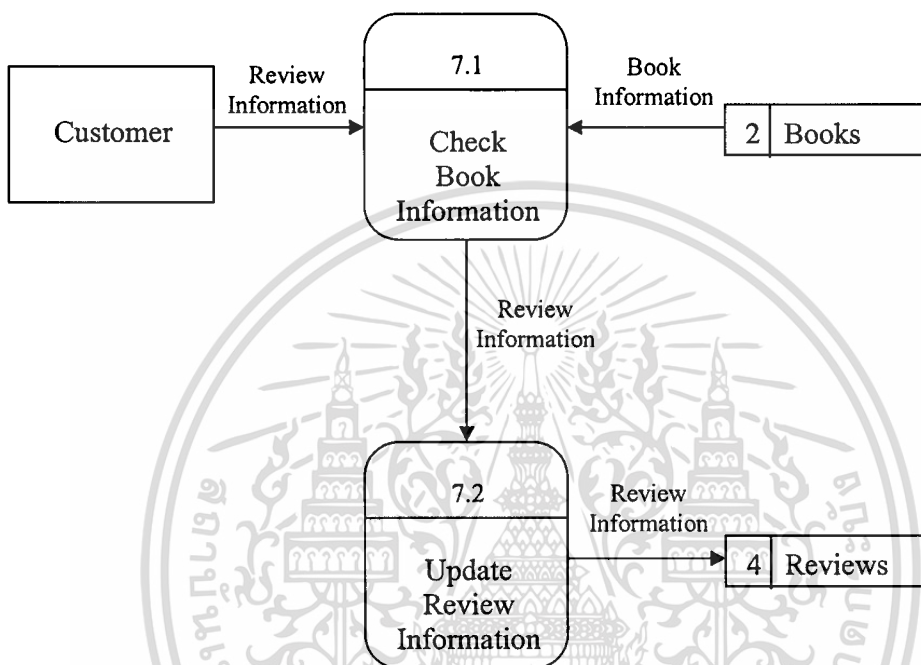
ภาพที่ 3.7 Data Flow Diagram Level 2 ของ Process 5 (Order)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



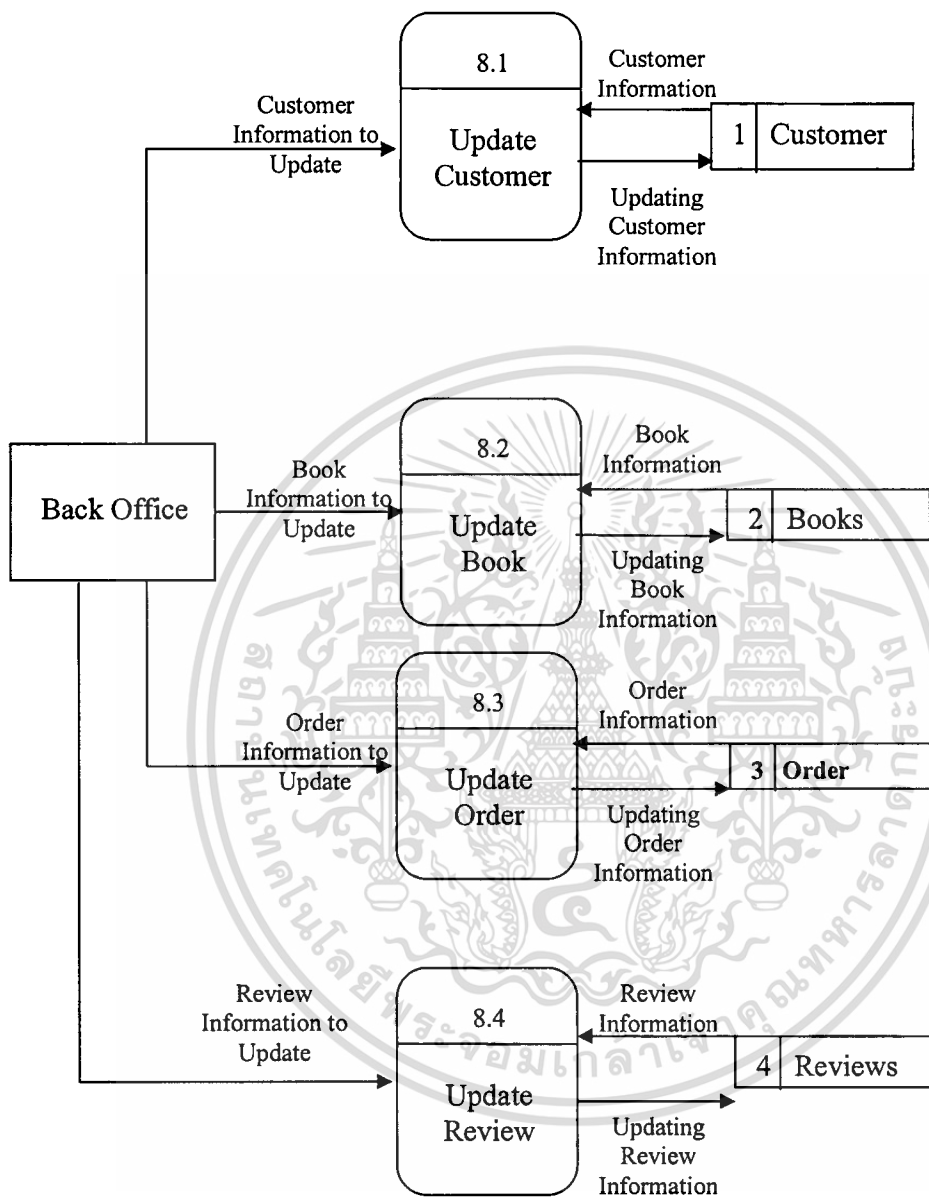
ภาพที่ 3.8 Data Flow Diagram Level 2 ของ Process 6 (Tracking)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.9 Data Flow Diagram Level 2 ของ Process 7 (Review Book)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



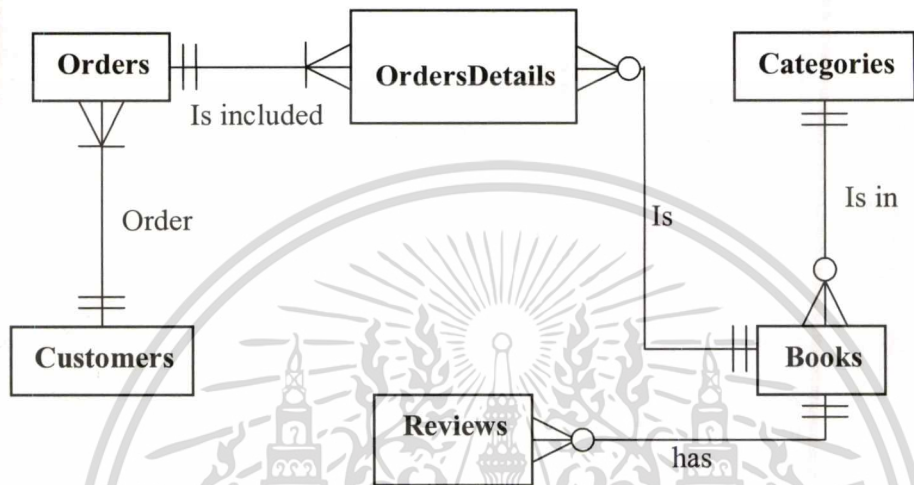
ภาพที่ 3.10 Data Flow Diagram Level 2 ของ Process 8 (Manage Information)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การออกแบบฐานข้อมูล

โดยการออกแบบฐานข้อมูลของระบบนี้ จะแสดงด้วย Entity Relationship Data Model

ดังภาพที่ 3.10



ภาพที่ 3.11 Entity Relationship Data Model ของระบบร้านค้านั่งโต๊ะ

ตารางที่ 3.1 ตารางในฐานข้อมูลทั้งหมดของระบบ

ชื่อตาราง	คำอธิบาย
Customers	ตารางข้อมูลลูกค้า
Categories	ตารางข้อมูลหมวดหนังสือ
Books	ตารางข้อมูลหนังสือ
Orders	ตารางข้อมูลการสั่งซื้อหนังสือ
OrderDetails	ตารางข้อมูลหนังสือที่ลูกค้าสั่งซื้อ
Reviews	ตารางข้อมูลแสดงความคิดเห็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.2 รายละเอียดของตาราง Customers

คีย์	ชื่อเขตข้อมูล	ชนิดข้อมูล	คำอธิบาย
P.K.	CustID	Integer	รหัสสมาชิก
	CustEmail	Text(50)	ใช้เป็น Account ในการเข้าสู่ระบบ
	CustPassword	Text(20)	รหัสผ่าน
	CustFirstName	Text(30)	ชื่อ
	CustLastName	Text(40)	นามสกุล
	CustAddress	Text(50)	ที่อยู่
	CustCity	Text(30)	อำเภอ/เขต
	CustProvince	Text(30)	จังหวัด
	CustCountry	Text(30)	ประเทศ
	CustZipcode	Text(10)	รหัสไปรษณีย์
	CustTel	Text(15)	เบอร์โทรศัพท์

ตารางที่ 3.3 รายละเอียดของตาราง Categories

คีย์	ชื่อเขตข้อมูล	ชนิดข้อมูล	คำอธิบาย
P.K.	CategoryID	Integer	รหัสหมวดหนังสือ
	CategoryName	Text(50)	ชื่อหมวดหนังสือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.4 รายละเอียดของตาราง Books

คีย์	ชื่อเขตข้อมูล	ชนิดข้อมูล	คำอธิบาย
P.K.	BookID	Integer	รหัสหนังสือ
	BookISBN	Text(10)	หมายเลขเรียกหนังสือ
	BookTitle	Text(75)	ชื่อหนังสือ
	BookAuthor	Text(120)	ผู้แต่ง
	BookPublisher	Text(50)	สำนักพิมพ์
	BookUnitCost	Integer	ราคาหนังสือ 1 เล่ม
	BookStock	Integer	จำนวนที่มีอยู่
	BookImage	Text(15)	ชื่อไฟล์รูปหนังสือ
F.K.	CategoryID	Integer	รหัสหมวดหนังสือ

ตารางที่ 3.5 รายละเอียดตาราง Orders

คีย์	ชื่อเขตข้อมูล	ชนิดข้อมูล	คำอธิบาย
P.K.	OrderID	Integer	หมายเลขสั่งซื้อ
F.K.	CustID	Integer	รหัสลูกค้า
	OrderDate	Date/Time	วันที่สั่งซื้อหนังสือ
	ShipDate	Date/Time	วันที่จัดส่งหนังสือ
	ShipFirstName	Text(30)	ชื่อผู้รับ
	ShipLastName	Text(40)	นามสกุลผู้รับ
	ShipAddress	Text(50)	ที่อยู่ผู้รับ
	ShipCity	Text(30)	อำเภอ/เขตผู้รับ
	ShipProvince	Text(30)	จังหวัดผู้รับ
	ShipCountry	Text(30)	ประเทศผู้รับ
	ShipZipcode	Text(10)	รหัสไปรษณีย์ผู้รับ
	ShipTel	Text(15)	เบอร์โทรศัพท์ผู้รับ
	CreditType	Integer	ประเภทบัตรเครดิต
	CreditNo	Text(15)	หมายเลขบัตรเครดิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.6 รายละเอียดตาราง OrderDetails

คีย์	ชื่อเขตข้อมูล	ชนิดข้อมูล	คำอธิบาย
P.K.	OrderID	Integer	หมายเลขสั่งซื้อ
P.K.	BookID	Integer	รหัสหนังสือ
	Quantity	Integer	ปริมาณการสั่งซื้อ
	BookUnitCost	Integer	ราคาหนังสือ 1 เล่ม

ตารางที่ 3.7 รายละเอียดตาราง Reviews

คีย์	ชื่อเขตข้อมูล	ชนิดข้อมูล	คำอธิบาย
P.K.	ReviewID	Integer	หมายเลขแสดงความคิดเห็น
	BookID	Integer	หมายเลขหนังสือ
	Name	Text(50)	ชื่อ
	Email	Text(50)	อีเมล
	Rating	Integer	ระดับคะแนน
	Comment	Text(500)	ความคิดเห็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การพัฒนาระบบ

4.1 ส่วนประกอบระบบ

ระบบร้านหนังสือในโครงการนี้ ประกอบด้วย 2 ส่วนหลักคือ ส่วนของหน้าร้านและส่วน
ของหลังร้าน โดยแต่ละส่วนมีรายละเอียด ดังนี้

4.1.1 ส่วนของหน้าร้าน (Store Front)

- ส่วนของการแสดงรายการหนังสือแนะนำ เป็นส่วนที่ใช้แนะนำหนังสือใหม่ หนังสือที่นำ
อ่าน โดยสามารถที่จะเข้าไปดูรายละเอียดหนังสือที่แนะนำเรื่องนั้นได้
- ส่วนของการค้นหาหนังสือ สามารถที่จะค้นหาหนังสือตามลูกค้าต้องการได้จาก keyword
เช่น ชื่อหนังสือ, รายละเอียดต่างๆของหนังสือ เป็นต้น
- ส่วนของการสมัครสมาชิก สามารถที่จะบันทึกข้อมูลของลูกค้าใหม่ ที่ต้องการเข้ามาซื้อ
หนังสือกับร้านค้า
- ส่วนของการเข้าใช้ระบบ เมื่อลูกค้าต้องการที่จะสั่งซื้อหนังสือกับทางร้านค้า ลูกค้าจำเป็นต้อง
จะต้อง Login ก่อนทุกครั้งจึงจะมีสิทธิ์ในการซื้อหนังสือได้
- ส่วนของการแก้ไขข้อมูลลูกค้า ลูกค้าสามารถที่จะทำการแก้ไขข้อมูลส่วนตัวกับทางร้านค้า
ได้ เช่น แก้ไขข้อมูลที่อยู่, เบอร์โทรศัพท์ เป็นต้น
- ส่วนของการแสดงรายละเอียดของหนังสือ ลูกค้าสามารถที่จะดูรายละเอียดหนังสือแต่ละ
เล่มได้ โดยการเลือกจาก link ของหนังสือที่ได้จากการค้นหาหรือได้จากการเลือกหนังสือ
เองจากเมนูหมวดหมู่ของสินค้า
- ส่วนของการเพิ่มรายการหนังสือลงไปใน shopping cart ลูกค้าสามารถที่จะเพิ่มรายการ
หนังสือที่ต้องการได้ โดยการเลือกจาก add to cart
- ส่วนของการแก้ไขรายการที่สั่งซื้อ ลูกค้าสามารถที่จะเพิ่มจำนวนหนังสือในแต่ละรายการ
ที่เลือกได้ ตามต้องการและถ้าหากไม่ต้องการหนังสือเล่มนั้นแล้วก็สามารถที่จะลบรายการ
นั้นออกจากรายการสั่งซื้อได้
- ส่วนของการสั่งซื้อ ลูกค้าต้องระบุข้อมูลบัตรเครดิตและข้อมูลที่จะให้ร้านค้าทำการส่ง
หนังสือจากรายการที่ได้เลือกไว้แล้ว
- ส่วนของการติดตามรายการที่สั่งซื้อ ลูกค้าสามารถที่จะดูรายละเอียดหนังสือที่สั่งซื้อไป
แล้วและติดตามการจัดส่งหนังสือของทางร้านได้ว่าได้ทำการจัดส่งแล้วหรือไม

- ส่วนของการแสดงรายการหนังสือที่มักนิยมซื้อด้วยกัน ส่วนนี้จัดทำขึ้นเพื่อช่วยให้ลูกค้าสามารถตัดสินใจเลือกซื้อหนังสือประเภทเดียวกันกับหนังสือที่กำลังเลือกอยู่ได้ง่ายขึ้น

4.1.2 ส่วนของหลังร้าน (Back Office)

- ส่วนของการแก้ไข,ลบ และเพิ่มข้อมูล ผู้ดูแลระบบสามารถที่จะดูแลข้อมูลเกี่ยวกับรายการหนังสือ, หนังสือ, ลูกค้าและบทวิจารณ์หนังสือผ่านทางอินเทอร์เน็ตได้

4.2 หน้าจอของระบบ

4.2.1 หน้าจอในส่วนของ Store Front



ภาพที่ 4.1 แสดงหน้าจอแรกของระบบ Store Front

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Sign Into Your Account

Email

Password

Remember My Sign-In Across Browser Restarts

If you are a new user and you don't have an account , then register for one now.

ภาพที่ 4.2 หน้าจอการเข้าสู่ระบบ ต้องมีการ login ก่อนจึงจะเลือกซื้อหนังสือได้ หากยังไม่มี Account ให้ทำการ Register ก่อน จากนั้นจึงจะ login เข้าสู่ระบบได้

Create a New Account

Full Name

Email

Password

Confirm Password

Address

City

Province

ภาพที่ 4.3 หน้าจอ Register ซึ่งมีไว้เพื่อลงทะเบียนกับระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

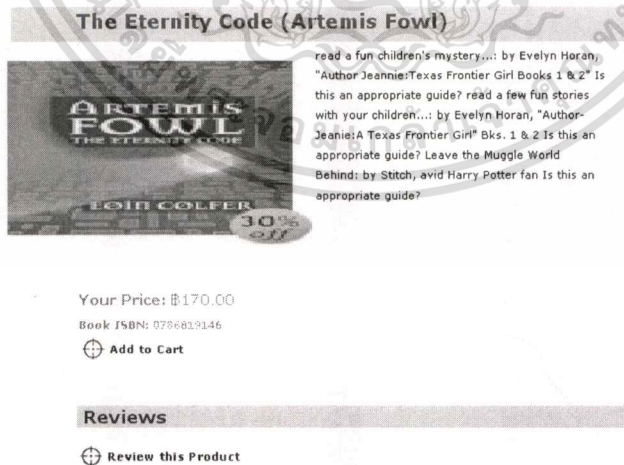


ภาพที่ 4.4 เมื่อเข้าสู่ระบบแล้วสามารถที่ทำการเรียกดูหนังสือตามที่ต้องการได้จากหมวดหนังสือ



ภาพที่ 4.5 สามารถค้นหาหนังสือได้จาก Search Engine จากภาพจะเป็นการค้นหาคำว่า "Harry Potter"

ภาพที่ 4.6 ผลที่ได้จากการค้นหา มี 2 รายการ



ภาพที่ 4.7 สามารถเข้าไปดูรายละเอียดของหนังสือได้โดยคลิกที่ชื่อหนังสือจากภาพที่ 4.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Add Review - The Eternity Code (Artemis Fowl)

Name

Musician

Email

test2@hotmail.com

Rating

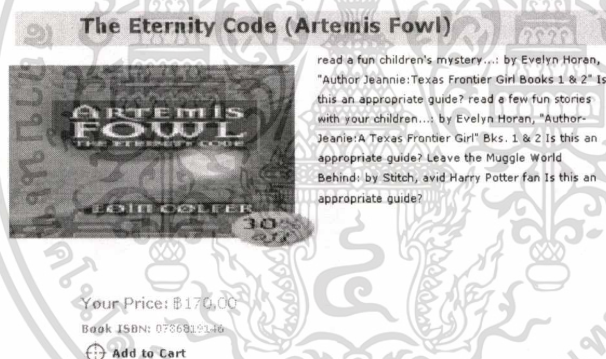
- ⭐⭐⭐⭐⭐
 ⭐⭐⭐⭐☆
 ⭐⭐⭐☆☆
 ⭐⭐☆☆☆
 ⭐☆☆☆☆

Comments

This is a fun book for children

Submit

ภาพที่ 4.8 สามารถแสดงความคิดเห็นเกี่ยวกับหนังสือในแต่ละเล่มได้



The Eternity Code (Artemis Fowl)

read a fun children's mystery... by Evelyn Horan, "Author Jeannie: Texas Frontier Girl Books 1 & 2" Is this an appropriate guide? read a few fun stories with your children... by Evelyn Horan, "Author- Jeannie: A Texas Frontier Girl" Bks. 1 & 2 Is this an appropriate guide? Leave the Muggle World Behind! by Stitch, avid Harry Potter fan Is this an appropriate guide?

Your Price: ฿170.00
Book ISBN: 0786819116

Add to Cart

Reviews

Review this Product

Musician says... ⭐⭐⭐☆☆

This is a fun book for children

← ผลการ Review จะปรากฏที่นี่

ภาพที่ 4.9 แสดงผลเมื่อกดปุ่ม Submit จากภาพที่ 4.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Shopping Cart

Book ID	Book Name	ISBN	Quantity	Price	Subtotal	Remove
376	Bear Wants More	068984509X	<input type="text" value="1"/>	\$ 170.00	\$170.00	<input type="checkbox"/>
377	Harry Potter and the Order of the Phoenix	043935806X	<input type="text" value="3"/>	\$ 300.00	\$900.00	<input type="checkbox"/>
374	The Eternity Code (Artemis Fowl)	0786819146	<input type="text" value="1"/>	\$ 170.00	\$170.00	<input type="checkbox"/>

Total: \$1,240.00

[Update Your Shopping Cart](#) [Final Check Out](#)

ภาพที่ 4.10 แสดงหน้าจอ Shopping cart

เมื่อต้องการซื้อหนังสือเล่มนี้ก็ให้คลิกปุ่ม Add to Cart จากภาพที่ 4.9 หนังสือเล่มนั้นก็จะเข้าไปสู่ shopping Cart ซึ่งที่นี้ลูกค้าสามารถแก้ไขข้อมูลการซื้อได้ เช่น ลบรายการซื้อออก, แก้ไขปริมาณการซื้อ เป็นต้น นอกจากนี้ยังมีการคำนวณรายการซื้อทั้งหมดให้ด้วย เมื่อพอใจหนังสือที่เลือกแล้วก็ให้คลิกปุ่ม Find Check Out

Review and Submit Your Order

Please check all the information below to be sure it's correct.

Product Name	Model Number	Quantity	Price	Subtotal
Bear Wants More	068984509X	1	\$ 170.00	\$170.00
Harry Potter and the Order of the Phoenix	043935806X	3	\$ 300.00	\$900.00
The Eternity Code (Artemis Fowl)	0786819146	1	\$ 170.00	\$170.00

Total: \$1,240.00

Please insert Data for Send to Receiver

Full Name

Address

Telephone

Email

Credit Type

VISA MASTER

Credit No.

[Submit](#)

ภาพที่ 4.11 แสดงผลที่ได้จากภาพที่ 4.10 แสดงรายการสั่งซื้ออีกครั้ง เพื่อให้ผู้ซื้อยืนยันรายการสั่งซื้อพร้อมระบุผู้รับและรูปแบบการจ่ายเงิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Check Out Complete!

Your Order Number Is: 126

Product Name	Model Number	Quantity	Price	Subtotal
Bear Wants More	068984509X	1	\$ 170.00	\$170.00
Harry Potter and the Order of the Phoenix	043935806X	3	\$ 300.00	\$900.00
The Eternity Code (Artemis Fowl)	0786819146	1	\$ 170.00	\$170.00

Total: \$1,240.00

ภาพที่ 4.12 แสดงผลที่ได้จากภาพที่ 4.11 เมื่อคลิกปุ่ม Submit แล้ว เป็นการแสดงรายการสั่งซื้อที่ลูกค้าได้มีการสั่งซื้อไปแล้ว

Account History

Order ID	Order Date	Order Total	Ship Date	Show Details
117	14/2/2546	\$1.99	14/2/2546	Show Details
118	14/2/2546	\$99.99	16/2/2546	Show Details
119	14/2/2546	\$3.99	16/2/2546	Show Details
120	14/2/2546	\$69.99	15/2/2546	Show Details
121	14/2/2546	\$29.99	15/2/2546	Show Details
122	14/2/2546	\$599.99	15/2/2546	Show Details
123	9/3/2546	\$15.00	9/3/2546	Show Details
124	10/3/2546	\$20,092.00	12/3/2546	Show Details
125	10/3/2546	\$2,090.00	10/3/2546	Show Details
126	19/3/2546	\$1,240.00	21/3/2546	Show Details

⊕ Edit Your Account

ภาพที่ 4.13 ลูกค้าสามารถดูรายการย้อนหลังที่มีการสั่งซื้อไปแล้วได้จากการคลิกที่ปุ่ม Account ซึ่ง เป็น Menu ส่วนบริการลูกค้าที่อยู่ด้านบนขวาของหน้าจอหลัก (ภาพที่ 4.1)

Order Details

Your Order Number Is: 125
Order Date: 10/3/2546
Ship Date: 10/3/2546

Book Name	Book ISBN	Quantity	Price	Subtotal
DNA: The Secret of Life	0375415467	2	\$ 400.00	\$800.00
Red Carpet Diaries	0345453786	5	\$ 240.00	\$1,200.00
Dressing the Man: The Art of Permanent Fashion	1929685653	3	\$30.00	\$90.00

Total: \$2,090.00

ภาพที่ 4.14 เมื่อคลิกที่ Show Details ในภาพที่ 4.13 ลูกค้าสามารถดูรายการสั่งซื้อที่ผ่านมาได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Edit Your Account

Customer can update your account from this page

Full Name

Email Address

Password

Address

City

Province

ภาพที่ 4.15 แสดงหน้าจอ Edit Account ลูกค้าสามารถทำการแก้ไขปรับปรุงข้อมูลส่วนตัวได้

4.2.1 หน้าจอในส่วนของ Back Office

Open Your Brain Open Your Knowledge
Open With Me ... Open **Book4U** Sign In

BOOK4U
Com

Update Book
Update Customer
Update Order
Update Review
Summary Sale

Welcome to Book4U.com

This is Back Office for Book Store website.
You can Update, Delete, Add Information.
Easy to use.
Interface friendly,
then you will love this site.

Administration Section

← เมนูส่วนการทำงาน BackOffice

↑ Login ที่นี่

ภาพที่ 4.16 แสดงหน้าจอหลักของ Back Office

Sign Into Your Account

User

Password

Remember My Sign-In Across Browser Restarts

ภาพที่ 4.17 แสดงหน้าจอการ login เข้าสู่ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Update Book

Update/delete Book

New Book

ภาพที่ 4.18 หลังจาก login เข้าสู่ระบบแล้วเมื่อเลือกที่ Update Book จากภาพที่ 4.16 จะแสดงเมนูย่อยในการจัดการกับตารางBook

Update/Delete Book

Book ID	Book Title	Book ISBN	Book Stock	Unit Cost	Book Author	Book Publisher	Remove
355	Xenosaga Or	0744000	2	140	Dan Birlaw	Brady Gam	<input type="checkbox"/>
356	Dark Star Sa	0618134	2	280	Paul Therou	Houghton N	<input type="checkbox"/>
357	Roadmap to	0735713	2	225	Joseph Low	Macromedi	<input type="checkbox"/>
358	Strength Trai	0736044	2	190	Frederic D	Human Kin	<input type="checkbox"/>
359	Using Samb	0596000	2	400	David Colli	O'Reilly & J	<input type="checkbox"/>
360	Whistler and	1929688	3	160	Eric Denke		<input type="checkbox"/>
362	Thomas Gair	1929688	3	160	Michael Ro		<input type="checkbox"/>
363	DNA: The Se	0375414	3	400	James D. V	Knopf	<input type="checkbox"/>
364	Golf for Enlig	0609603	3	210	Deepak Ch	Crown Pu	<input type="checkbox"/>
365	The Hardy Bl	0060522	4	250	Michael Kn	Regan Boo	<input type="checkbox"/>
367	Salt: A Work	0142000	4	150	Mark Kurlf	Penguin UE	<input type="checkbox"/>

Update Book

ภาพที่ 4.19 แสดงข้อมูลที่สามารถ update และ delete ได้ หลังจาก click update/delete book จากภาพที่ 4.18

Create a New Book

Insert Book with this page

Book Title

Category Id

Book ISBN

Book Author

Book Publisher

Unit Cost

Book Stock

Description

Submit

ภาพที่ 4.20 แสดงฟอร์มให้ใส่ข้อมูลเกี่ยวกับหนังสือใหม่ที่จะเพิ่มเข้าสู่ฐานข้อมูล หลังจาก click new book จากภาพที่ 4.18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Summary Book Sale

Summary Sale(Product)

ภาพที่ 4.21 แสดงเมนูย่อยในการสรุปผลการขายหนังสือเมื่อมีการเลือกเมนู
Summary Sale จากภาพที่ 4.16

Summary Sale (Product)

Book ID	Book Title	Quantity	Unit Cost	total Cost
404	The Artist's Way: A Spiritual Path	9	\$160.00	\$1,440.00
355	Xenosaga Official Strategy Guide	8	\$140.00	\$1,120.00
386	Red Carpet Diaries	6	\$240.00	\$1,440.00
397	Dressing the Man: The Art of Permanent Fashion	6	\$30.00	\$180.00
401	Mastering Digital Printing: The Photographer	5	\$240.00	\$1,200.00
362	Thomas Gainsborough	5	\$60.00	\$300.00
363	DNA: The Secret of Life	4	\$400.00	\$1,600.00
368	Adobe Photoshop 7.0 Classroom in a Book	4	\$445.00	\$1,780.00
374	The Eternity Code (Artemis Fowl)	4	\$170.00	\$680.00
378	The Get Fuzzy Experience	4	\$110.00	\$440.00
406	Triumph and Tragedy in Mudville	3	\$250.00	\$750.00
364	Golf for Enlightenment	3	\$210.00	\$630.00
377	Harry Potter and the Order of the Phoenix	3	\$300.00	\$900.00
356	Dark Star Safari	1	\$280.00	\$280.00
360	Whistler and His Circle in Venice	1	\$150.00	\$150.00
375	Mac OS X: The Missing Manual, Second Edition	1	\$300.00	\$300.00
376	Bear Wants More	1	\$170.00	\$170.00
372	Walt Disney World 2003	1	\$170.00	\$170.00
402	The Ultimate Kauai Guidebook	1	\$150.00	\$150.00
379	Races of Faerun	1	\$300.00	\$300.00
384	Perfect I'm Not: Boomer on Beer	1	\$260.00	\$260.00
400	Where Is Baby's Belly Button? A Lift-the-Flap Book	1	\$60.00	\$60.00

ภาพที่ 4.22 แสดงรายละเอียดการขายหนังสือแต่ละเล่มของทางร้าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทสรุปและข้อเสนอแนะ

5.1 บทสรุป

การนำเทคโนโลยีที่เรียกว่า Web Service เข้ามาใช้งานจะเริ่มมีบทบาทมากขึ้นในการใช้งานอินเทอร์เน็ตอันใกล้นี้เนื่องจากว่า Web Service มีข้อดีในการพัฒนาแอปพลิเคชันบนอินเทอร์เน็ตที่สามารถทำงานข้ามแพลตฟอร์มได้ สำหรับเครื่องมือที่ใช้ในการพัฒนา Web Service ปัจจุบันเริ่มมีมากขึ้น ในโครงการนี้ก็ได้อาศัยเครื่องมือในการพัฒนาจากไมโครซอฟต์ที่เรียกว่า .NET ซึ่งสามารถอำนวยความสะดวกให้กับการพัฒนาแอปพลิเคชัน ได้อย่างดีทำให้ได้ระบบร้านหนังสือบนอินเทอร์เน็ตที่มีประสิทธิภาพ นอกจากนี้ระบบนี้ยังสามารถนำไปเป็นตัวอย่างในการพัฒนาแอปพลิเคชันร้านค้าบนอินเทอร์เน็ตได้อีกด้วยเนื่องจากมีลักษณะการทำงานที่ใกล้เคียงกัน

5.2 ข้อเสนอแนะและปัญหาที่เกิดขึ้น

- เนื่องจาก Web Service มีการใช้เทคโนโลยีในการพัฒนาหลายตัวด้วยกัน ในช่วงแรกของการศึกษา อาจจะต้องเสียเวลาในการศึกษาอยู่พอสมควร จึงทำให้ช่วงแรกของการพัฒนาเป็นไปอย่างล่าช้า
- เพิ่มความสามารถให้แอปพลิเคชันส่งเมลไปแจ้งข่าวสารในกับลูกค้าเพื่อแจ้งข่าวเกี่ยวกับหนังสือของทางร้านเป็นระยะเมื่อมีหนังสือเข้าใหม่
- เพิ่มความสามารถให้แอปพลิเคชันโดยมีการเปรียบเทียบราคาหนังสือกับร้านค้านั้นๆ
- เพิ่มปริมาณหนังสือให้เยอะขึ้น เพื่อทดสอบการทำงานของแอปพลิเคชันในการประมวลผลและดึงข้อมูล

บรรณานุกรม

จำลอง คุรุอุตสาหกรรม. 2545. ASP.NET ฉบับโปรแกรมเมอร์. กรุงเทพฯ : เคทีพี.

ไพศาล โมลิสกุลมงคล. 2545. Microsoft Visual C# .NET. กรุงเทพฯ : ดวงกมลสมัย.

สราวุธ อ้อยศรีสกุล. 2544. ถอดรหัส.NET + Web Services. กรุงเทพฯ : วิตตี้ กรู๊ป.

สราวุธ อ้อยศรีสกุล. 2544. เริ่มคิด-เริ่มสร้าง-เริ่มใช้ XML. กรุงเทพฯ : วิตตี้ กรู๊ป.

123aspx Studio. 2002. Chart-ASP.NET, components. [Online]. Available:

<http://www.123aspx.com>.

GizMos. 2001. Web Services กับการส่งค่ากลับในรูปแบบต่าง ๆ. [Online]. Available:

<http://www.thaisharp.net>.

Salcentral group. 2001. Web Service Tutorials. [Online]. Available:

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

Yasser Shohoud. 2001. Web Service Overview. [Online]. Available: <http://www.xmlws.com>.



ประวัติผู้เขียน

ชื่อ-นามสกุล นายอรรถพร หวังพูนทรัพย์
 เกิดเมื่อวันที่ 5 กุมภาพันธ์ 2522
 สถานที่เกิด อำเภอหาดใหญ่ จังหวัดสงขลา

ประวัติการศึกษา

- จบระดับชั้นประถมศึกษาจากโรงเรียนกิตติวิทย์ (หาดใหญ่)
- จบระดับชั้นมัธยมศึกษาตอนต้น-ปลาย ที่โรงเรียนหาดใหญ่วิทยาลัย (หาดใหญ่)
- จบระดับปริญญาตรี (วิทยาศาสตร์บัณฑิต) จากคณะวิทยาศาสตร์ ภาควิชา วิทยาการคอมพิวเตอร์ มหาวิทยาลัยสงขลานครินทร์ วิทยาเขตหาดใหญ่

