

การพัฒนาโปรแกรมสั่งซื้อสินค้าผ่านอินเทอร์เน็ต
ในลักษณะเว็บเซอร์วิสด้วยภาษาจาวา

Development of Purchase Order Web Application Using Java Web Services

โดย

นางสาวรัชนิชา แซ่เตีย

รหัส 44067078

วัน เดือน ปี.....	๒๖ มี.ค. 2550
เลขทะเบียน.....	02023
เลขเรียกหนังสือ.....	วท. ร. 334 ก 2546
"ห้องสมุดคณะเทคโนโลยีสารสนเทศ ศจธ."	

อาจารย์ที่ปรึกษา

รศ.ดร.วิเชียร เปรมชัยสวัสดิ์

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
ภาคเรียนที่ 1 ปีการศึกษา 2546.
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง



H002023

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ	การพัฒนาโปรแกรมสั่งซื้อสินค้าผ่านอินเทอร์เน็ตในลักษณะเว็บเซอร์วิสด้วยภาษาจาวา
นักศึกษา	นางสาวรัชนิชา แซ่เตีย
อาจารย์ที่ปรึกษา	รศ.ดร.วิเชียร เปรมชัยสวัสดิ์
ระดับการศึกษา	วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2546

บทคัดย่อ

พัฒนาโปรแกรมสั่งซื้อสินค้าผ่านอินเทอร์เน็ตในลักษณะของเว็บแอปพลิเคชันและเว็บเซอร์วิส ด้วยรูปแบบที่โรงงานผลิต(Plant web services) ให้บริการข้อมูลของปริมาณสินค้าที่สามารถขายได้ในขณะนั้นให้กับตัวแทนจำหน่าย (Distributor server) เมื่อมีลูกค้า (Dealer) สั่งซื้อสินค้าในแต่ละครั้ง โดยใช้ภาษาจาวา, XML และ Java APIs สำหรับ XML เป็นเครื่องมือในการพัฒนา ซึ่งรูปแบบนี้ทำให้ตัวแทนจำหน่ายไม่ต้องบริหารจัดการสต็อกสินค้าเองและโรงงานผลิตสามารถวางแผนการผลิตได้มีประสิทธิภาพมากขึ้น

Title Development of Purchase Order Web Application Using
Java Web Services

Student Ms. Ruchaneechol Tia

Advisor Assoc. Prof. Dr. Wichian Premchaiswadi

Level of Study Master of Science in Information Technology

Major Information Science

Academic Year 2003



ABSTRACT

Development of purchase order web application uses Java Programming Language, XML and Java APIs for XML. The system process is Dealers (Client) submit order request to Web server which request to Plant sever for stock quantity and request to Distributor server for prices and customers detail. In this system, Distributor will reduce cost for maintaining store, and Plant will increase more efficiency in production plan.

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
สารบัญ.....	III
บทที่	
1 บทนำ.....	1
1.1 ความเป็นมา.....	1
1.2 วัตถุประสงค์ของการพัฒนาระบบ.....	1
1.3 ขอบเขตการพัฒนาระบบ.....	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	3
2 ทฤษฎีและเทคโนโลยีที่เกี่ยวข้อง.....	4
2.1 Extensible Markup Language (XML).....	4
2.2 เว็บเซอร์วิส.....	7
2.3 ภาษาจาวา.....	7
2.4 Java API for XML.....	9
2.5 Java Web Services Developer Pack.....	13
2.6 AS/400 Toolbox for Java.....	18
2.7 Java Database Connectivity Application Programming Interface.....	20
3 ศึกษาและวิเคราะห์ระบบปัจจุบัน.....	24
3.1 ขั้นตอนการทำงาน.....	24
3.2 ปัญหาของระบบปัจจุบัน.....	38
4 การวิเคราะห์และออกแบบเพื่อแก้ไขปัญหาระบบเดิม.....	40
4.1 ศึกษาความเป็นไปได้และความเหมาะสมในการพัฒนาระบบ.....	40
4.2 การวิเคราะห์และออกแบบระบบใหม่.....	40
4.3 ER Diagram.....	48

4.4	การออกแบบฐานข้อมูล.....	49
5	การพัฒนาระบบ.....	52
5.1	เครื่องมือที่ใช้ในการพัฒนาระบบ.....	52
5.2	รูปแบบการทำงานของระบบ.....	52
5.3	การรักษาความปลอดภัย.....	58
5.4	การบำรุงรักษาระบบ.....	58
6	ผลจากการดำเนินงาน.....	59
6.1	ผลจากการดำเนินงาน.....	59
6.2	ปัญหาและแนวทางแก้ไข.....	59
	บรรณานุกรม.....	60
	ประวัติผู้แต่ง.....	61



บทที่ 1

บทนำ

1.1 ความเป็นมา

ในปัจจุบันการติดต่อสื่อสารรับส่งข้อมูลผ่านอินเทอร์เน็ตมีการใช้กันอย่างแพร่หลาย มีการนำไปใช้ในระบบธุรกิจหลายรูปแบบ ทั้งในลักษณะของธุรกิจกับธุรกิจ (Business to Business) และธุรกิจกับลูกค้า (Business to Customer) สืบเนื่องจากความสะดวกในการใช้งานที่สามารถใช้ผ่านเว็บเบราว์เซอร์โดยผู้ใช้ไม่ต้องรับรู้ถึงโปรแกรมต่างๆที่ทำงานอยู่เบื้องหลัง รวมถึงความสามารถใช้งานโปรแกรมได้จากระยะไกล ดังนั้นองค์กรต่างๆส่วนใหญ่ในปัจจุบันจึงได้นำเอาข้อดีของอินเทอร์เน็ตเข้ามาประยุกต์ใช้กับธุรกิจทั้งการติดต่อกับองค์กรอื่นที่เป็นคู่ค้าทางธุรกิจและลูกค้า บางองค์กรได้ติดตั้งระบบใหม่ทดแทนระบบเก่าที่ใช้อยู่ทั้งหมด บางองค์กรอาศัยการปรับปรุงระบบเดิมที่มีอยู่ให้มีความสามารถเพิ่มมากขึ้นโดยไม่ต้องลงทุนเพิ่มมากนัก ซึ่งจากวิธีการในแบบที่สองนั้นเหมาะสำหรับองค์กรที่มีขนาดเล็กและมีความต้องการในการปรับตัวให้เข้ากับเทคโนโลยีในปัจจุบันเพื่อรองรับการขยายตัวของธุรกิจในอนาคต

จากข้อดีของการใช้อินเทอร์เน็ตดังที่กล่าวมาแล้ว ในบทความนี้จะได้นำเสนอการนำอินเทอร์เน็ตเข้ามาประยุกต์ใช้โดยปรับปรุงเพิ่มจากระบบเดิมที่ยังคงใช้งานอยู่ โดยสิ่งที่พัฒนาเพิ่มเติมขึ้นมานั้นจะช่วยเพิ่มประสิทธิภาพของการทำงาน และในระยะยาวจะช่วยประหยัดค่าใช้จ่ายในการดำเนินธุรกิจได้ สำหรับรูปแบบของธุรกิจที่น่าเสนอเป็นธุรกิจระหว่างองค์กรกับองค์กร และองค์กรกับลูกค้า โดยมีรูปแบบในลักษณะขององค์กรที่เป็นตัวแทนจำหน่าย (Distributor) ติดต่อซื้อสินค้ากับโรงงานผลิต (Plant) และติดต่อขายสินค้ากับลูกค้า (Dealer) โดยอาศัยวิธีการของเว็บเซอร์วิส (Web Services) ที่พัฒนาด้วยภาษาจาวาเป็นหลัก และเนื่องจากระบบเดิมของแต่ละฝ่ายมีรูปแบบที่ไม่เหมือนกัน ดังนั้นจึงต้องอาศัยภาษา XML (Extensible Markup Language) ที่มีคุณสมบัติสามารถนำไปเปลี่ยนเป็นรูปแบบที่ต้องการได้โดยอาศัยเครื่องมือในการเปลี่ยนรูปแบบมาเป็นตัวกลางเพื่อให้ระบบที่ต่างกันสามารถติดต่อรับส่งข้อมูลกันได้

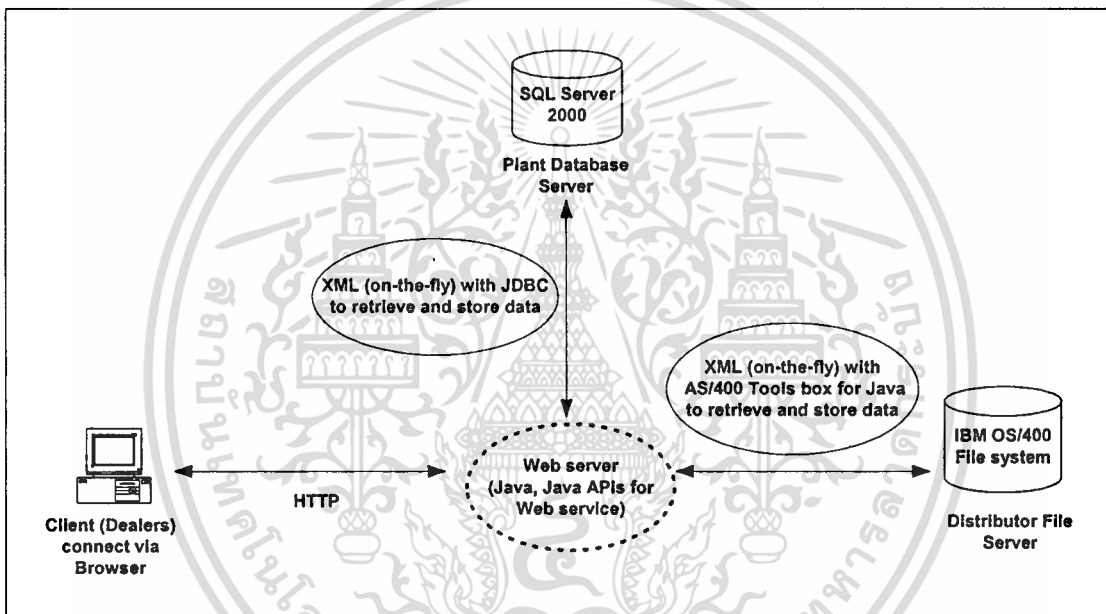
1.2 วัตถุประสงค์การพัฒนาระบบ

การพัฒนาระบบส่งซื้อสินค้าด้วยเว็บเซอร์วิส มีวัตถุประสงค์คือต้องการให้ระบบคอมพิวเตอร์ของตัวแทนจำหน่ายและระบบคอมพิวเตอร์ของโรงงานผลิต สามารถติดต่อรับส่งข้อมูลกันได้ แม้จะเป็นระบบคอมพิวเตอร์ที่แตกต่างกันทั้งทางด้านฮาร์ดแวร์และซอฟต์แวร์ โดยที่ทั้ง 2 ฝ่ายไม่เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่ขึ้นด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จำเป็นต้องมีการเพิ่มเติมประสิทธิภาพของฮาร์ดแวร์และลงทุนซื้อซอฟต์แวร์ใหม่ อีกทั้งระบบอื่นๆ ที่มีความสัมพันธ์กับระบบสั่งซื้อในระบบเดิม (Legacy system) นี้ ยังสามารถใช้งานได้ตามปกติ และเพื่อต้องการให้ลูกค้าได้รับความสะดวกมากขึ้นด้วยวิธีการสั่งซื้อสินค้าผ่านอินเทอร์เน็ต

1.3 ขอบเขตการพัฒนาระบบ

เป็นการนำเอาระบบสั่งซื้อสินค้าและระบบขายสินค้าของ ตัวแทนจำหน่าย เข้ามารวมไว้ด้วยกัน มีรายละเอียดดังนี้



รูปที่ 1.1 ภาพรวมของระบบ

จากรูปอธิบายขั้นตอนได้ดังนี้คือ

1. ลูกค้าติดต่อกับเว็บเซอร์เวอร์โดยใช้โปรโตคอล HTTP (Hyper Text Transfer Protocol) ผ่านเว็บเบราว์เซอร์ เริ่มต้นด้วยการตรวจสอบความต้องการของผู้ใช้งานระบบ เมื่อได้รับรหัสลูกค้ามาแล้วจะส่งไปที่เซอร์เวอร์ของตัวแทนจำหน่ายเพื่อขอข้อมูลที่เกี่ยวข้องกับลูกค้า เช่น ชื่อ, ที่อยู่, เครดิตเทอม, ส่วนลดทางการค้า พร้อมทั้งรหัสสินค้าและราคา เมื่อเซอร์เวอร์ของตัวแทนจำหน่ายส่งข้อมูลกลับมา เว็บเซอร์เวอร์จะนำไปแสดงบนหน้าจอให้ลูกค้าเลือกสั่งซื้อสินค้าตามความต้องการ กระบวนการรับส่งข้อมูลระหว่างเว็บเซอร์เวอร์กับเซอร์เวอร์ของตัวแทนจำหน่ายนั้น จะใช้เครื่องมือในการพัฒนาที่มีชื่อว่า AS/400 Toolbox for Java ซึ่งเป็นชุดของคลาสสร้างด้วยภาษาจาวาที่บริษัท ไอบีเอ็ม จำกัด ได้จัดเตรียมไว้ให้นักพัฒนาสามารถนำไปใช้ในการเข้าถึงข้อมูลของ AS/400 ที่มีลักษณะเป็น File system

2. เมื่อลูกค้าได้เลือกสินค้ากับจำนวนที่ต้องการพร้อมกับ Submit คำสั่งซื้อส่งไปให้เว็บเซิร์ฟเวอร์ แล้วตัวเว็บเซิร์ฟเวอร์จะส่งคำสั่งซื่อนั้นไปที่เซิร์ฟเวอร์ของโรงงานผลิต เพื่อเป็นการแจ้งให้ทางเซิร์ฟเวอร์ของโรงงานผลิตทราบถึงปริมาณสินค้าที่ลูกค้าสั่งซื้อ จากนั้นเซิร์ฟเวอร์ของ โรงงานผลิต จะส่งข้อมูลเกี่ยวกับวันที่ส่งสินค้า โดยจะบันทึกข้อมูลรายการสั่งซื้อในฐานะข้อมูล กระบวนการรับส่งข้อมูลระหว่างเว็บเซิร์ฟเวอร์และเซิร์ฟเวอร์ของ โรงงานผลิตจะใช้เครื่องมือที่เป็น API ทำหน้าที่เป็นตัวกลางระหว่างฐานข้อมูลและแอปพลิเคชัน ในที่นี้ใช้ JDBC API มาเชื่อมต่อระหว่างฐานข้อมูล SQL Server 2000 กับโปรแกรมที่ใช้ภาษาจาวาในการพัฒนา
3. เว็บเซิร์ฟเวอร์จะส่งข้อมูลเกี่ยวกับวันที่ส่งสินค้าพร้อมกับสรุปรายการสั่งซื้อสินค้าและจำนวนเงินให้ลูกค้าทราบ ในขณะที่เดียวกันจะส่งข้อมูลทั้งหมดกลับไปให้เซิร์ฟเวอร์ของตัวแทนจำหน่ายและเซิร์ฟเวอร์ของ โรงงานผลิตเพื่อเพิ่มรายการสั่งซื้อพร้อมรายละเอียดทั้งหมดในฐานข้อมูลของระบบและนำไปใช้ในระบบอื่นต่อไป

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. ในระบบเดิมนั้นตัวแทนจำหน่ายมีระบบคลังสินค้าที่ควบคุมการรับจ่ายสินค้าทั้งหมด โดยที่ตัวแทนจำหน่ายมีค่าใช้จ่ายในการดูแลคลังสินค้าในแต่ละเดือนค่อนข้างสูง จากระบบใหม่นี้จะช่วยให้ตัวแทนจำหน่ายไม่จำเป็นต้องมีคลังสินค้าอีกต่อไปเพราะตัวแทนจำหน่ายจะทำหน้าที่เสมือนกับเป็นตัวกลางในการรับคำสั่งซื้อจากลูกค้าไปที่โรงงานผลิตเท่านั้น
2. ในแต่ละวันทั้งตัวแทนจำหน่ายและ โรงงานผลิตมีค่าใช้จ่ายเกี่ยวกับค่าขนส่งเป็นจำนวนมาก โรงงานผลิตนำส่งสินค้าให้กับคลังสินค้าของตัวแทนจำหน่าย และตัวแทนจำหน่ายนำส่งสินค้าให้ลูกค้าอีกครั้งหนึ่ง ในระบบใหม่มีการขนส่งจากโรงงานผลิตไปที่ลูกค้าโดยตรง จะเห็นได้ว่าเป็นการลดค่าใช้จ่ายที่เกิดจากการขนส่งและสินค้าถึงลูกค้าได้เร็วขึ้น โดยที่ทั้งตัวแทนจำหน่ายและ โรงงานผลิตมีข้อตกลงรับผิดชอบค่าใช้จ่ายที่เกิดขึ้นร่วมกัน

บทที่ 2

ทฤษฎีและเทคโนโลยีที่เกี่ยวข้อง

2.1 XML (Extensible Markup Language)

เอกสาร XML นั้นจะหมายถึงฐานข้อมูลเฉพาะในความหมายที่ว่าเป็นการรวมกลุ่มของข้อมูลเท่านั้น ข้อดีของการทำให้เอกสาร XML เป็นฐานข้อมูลมีอยู่หลายประการ เช่น มันสามารถอธิบายความหมายของข้อมูลได้ด้วยตัวของมันเอง มีลักษณะของ Unicode คือเป็นรูปแบบที่ใช้ได้ทั่วไปหรือสามารถอธิบายข้อมูลในลักษณะที่เป็นโครงสร้างแบบแผนภูมิต้นไม้หรือกราฟ แต่ข้อเสียของมันคือการเข้าถึงข้อมูลทำได้ช้าเนื่องจากต้องตรวจสอบความถูกต้องของข้อมูล XML มีสิ่งต่างๆที่สามารถพบได้ในฐานข้อมูลเช่น ที่เก็บเอกสาร XML, รูปแบบของ DTD(Document Type Definition) หรือนิยามของเอกสาร XML, ภาษา Query เช่น XQuery, XPath, XQL และแอปพลิเคชันโปรแกรมมิ่งอินเตอร์เฟซ (Application Programming Interface, API) เช่น SAX(Simple API for XML), DOM(Document Object Model) เป็นต้น เอกสาร XML จะเหมาะสำหรับข้อมูลที่เป็นชุดๆ เช่น ข้อมูลของบุคคลที่ประกอบไปด้วยชื่อ เบอร์โทรศัพท์ ที่อยู่, ไฟล์ข้อมูลประเภทมัลติมีเดีย

2.1.1 รูปแบบของเอกสาร XML

1. เอกสาร Data-Centric คือเอกสารที่ใช้ XML เป็นตัวนำส่งข้อมูล ตัวอย่างของเอกสาร Data-centric คือ เอกสารใบสั่งซื้อ, ตารางการบิน, ข้อมูลทางวิทยาศาสตร์ และข้อมูลตลาดหุ้น เอกสาร Data-centric ถูกกำหนดลักษณะพิเศษโดยโครงสร้างที่มีรูปแบบแน่นอน เอกสารใบสั่งซื้อข้างล่างนี้เป็นตัวอย่างของเอกสาร Data-centric

```
<SalesOrder SONumber="12345">
  <Customer CustNumber="543">
    <CustName>Siam GS</CustName>
    <City>Bangkok</City>
    <PostCode>10260</PostCode>
  </Customer>
  <OrderDate>011215</OrderDate>
  <Item ItemNumber="1">
    <Part PartNumber="N50">
      <Description>
        <p><b>Nissan BigM:</b><br /></p>
```

เอกสารนี้เป็นเอกสารลิขสิทธิ์สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

</Description>
<Price>1200</Price>
</Part>
<Quantity>10</Quantity>
</Item>
<Item ItemNumber="2">

```

```

  <Part PartNumber="N70">
    <Description>
      <p><b>Toyota Hilux:<br /></b></p>
    </Description>
    <Price>950</Price>
  </Part>
  <Quantity>5</Quantity>
</Item>
</SalesOrder>

```

2. เอกสาร Document-centric โดยทั่วไปแล้วคือเอกสารที่ถูกออกแบบมาเพื่อรองรับการสร้างจากผู้ใช้ ตัวอย่างเช่น หนังสือ, E-mail, โฆษณา และเอกสาร XHTML (Extensible Hyper Text Markup Language) โครงสร้างของมันจะมีลักษณะที่ยืดหยุ่นสูงไม่มีกฎเกณฑ์ตายตัว ตัวอย่างของการใช้เอกสาร Document-centric อธิบายสินค้า

```

<Product>
  <Name>Turkey Wrench</Name>
  <Developer>Full Fabrication Labs, Inc. </Developer>
  <Summary> Like a monkey wrench, but not as big. </Summary>
  <Description>
    <Para>The turkey wrench, which comes in <i>both right- and left-handed versions (skyhook optional)</i>, is made of the <b>finest stainless steel</b>.</Para>
    <Para>You can:</Para>
    <List>
      <Item><Link URL="Order.html">Order your own turkey wrench</Link></Item>
    </List>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

</Description>

</Product>

2.1.2 การจัดเก็บและการเรียกข้อมูลกลับมาใช้งาน

1. การนำส่งข้อมูล (Transferring data) จากเอกสาร XML ไปที่ฐานข้อมูลจะมีการละทิ้งข้อมูลบางส่วนของเอกสารเช่น ชื่อของเอกสารและ DTD บางครั้งยังอาจจะละทิ้งข้อมูลของโครงสร้างทางกายภาพของเอกสารได้ด้วย เช่นข้อกำหนดและการใช้งานของเอนทิตี หรือส่วนของ CDATA (Character Data) การกำหนดข้อมูลที่ไม่ต้องสนใจโครงสร้าง หรือการเข้ารหัสของข้อมูล และในทำนองเดียวกัน เมื่อมีการนำส่งข้อมูลจากฐานข้อมูลไปที่เอกสาร XML ผลลัพธ์ก็คือจะได้เอกสาร XML ที่เหมือนกับไม่ได้บรรจุด้วย CDATA หรือการใช้เอนทิตีที่ไม่ได้กำหนดไว้ก่อน ส่วนลำดับของอีลิเมนต์และแอตทริบิวต์ จะปรากฏเหมือนกับลำดับการจัดเรียงที่ย้อนกลับมาจากฐานข้อมูล ตัวอย่างเช่นในกรณีที่ XML ที่ถูกนำส่งข้อมูลเอกสารใบสั่งซื้อจากฐานข้อมูลหนึ่งไปอีกฐานข้อมูลหนึ่งนั้น ไม่ว่าเลขที่ของใบสั่งซื้อจะถูกเก็บก่อนหรือหลังของวันที่เอกสารใบสั่งซื้อก็ไม่มีผลใดๆทั้งสิ้น

2. การจับคู่เอกสารไปเป็นฐานข้อมูล แอปพลิเคชันที่นำส่งข้อมูลระหว่างเอกสาร XML และฐานข้อมูลอาศัยการจับคู่ระหว่างเอกสารและฐานข้อมูล ตัวอย่างเช่น การจับคู่ระหว่างอีลิเมนต์ของเอกสารใบสั่งซื้อกับตารางที่เก็บข้อมูลของเอกสารใบสั่งซื้อทั้งหมด หรือการจับคู่ระหว่างอีลิเมนต์ที่ชื่อ OrderDate กับ คอลัมน์ที่ชื่อ SalesOrder Date เราจะแบ่งการจับคู่ออกเป็น 2 ประเภทคือ Template-driven กับ Model-driven

- การจับคู่แบบ Template-Driven ในการจับคู่แบบนี้ไม่มีการกำหนดกฎเกณฑ์ของการจับคู่ระหว่างเอกสารกับข้อมูลไว้ก่อน แต่จะฟังคำสั่งเฉพาะที่มีในแอปพลิเคชันนั้นๆไว้ในแบบตัวอย่างที่ถูกทำงานด้วย แอปพลิเคชันนำส่งข้อมูลแทน เช่น ฟังคำสั่ง SELECT statement ไว้ ในรูปแบบของอีลิเมนต์ <SelectStmt> การจับคู่แบบ Template-driven มีความยืดหยุ่นสูง และมีลักษณะสำคัญคือสามารถที่จะวางค่าของผลลัพธ์ในที่ใดๆก็ได้ในเอกสารที่เป็นผลลัพธ์ออกมา รวมถึงค่าของพารามิเตอร์ที่อยู่ใน SELECT statement คำสั่งถัดไป สามารถใช้คำสั่ง For loops และ If statements กำหนดค่าตัวแปรและฟังค์ชันต่างๆได้ และส่งผ่านค่าพารามิเตอร์ของ SELECT statement ไปที่ค่าพารามิเตอร์ของ HTTP ได้

- การจับคู่แบบ Model-driven ในการจับคู่แบบนี้ข้อมูลในเอกสาร XML จะถูกสร้างขึ้นตามแบบที่ได้กำหนดไว้ก่อนล่วงหน้า ระบบจะตั้งอยู่บนแบบจำลองที่เห็นได้ชัดเจน ทำให้ใช้งานได้ง่ายเนื่องจากข้อมูลต้องถูกนำส่งตามแบบที่ได้ถูกกำหนดไว้ก่อน บางครั้งเอกสารอาจจะไม่ทำตามรูปแบบของแบบที่สร้างไว้ จึงมีการนำ XSLT(Extensible Style Sheet Transformations) มารวมเข้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไว้ในมิดเดิลแวร์แบบ Model-driven เพื่อให้แอปพลิเคชันเปลี่ยนรูปแบบเอกสารไปเป็นแบบที่สร้างไว้ก่อนนำส่งข้อมูลไปพื้นฐานข้อมูล โดยทั่วไปรูปแบบการจับคู่ที่ใช้กันสำหรับข้อมูลในเอกสาร XML อยู่มี 2 ประเภท คือ “Table-model” ซึ่งเป็นพื้นฐานของการจับคู่แบบตาราง ส่วนใหญ่จะนำไปใช้ในการนำส่งข้อมูลระหว่างเอกสาร XML และฐานข้อมูลเชิงสัมพันธ์ มันจำลองเอกสาร XML ให้เหมือนกับตาราง 1 ตารางหรือกลุ่มของตาราง และ “Object model” ซึ่งเป็นพื้นฐานสำหรับการจับคู่ของความสัมพันธ์เชิงวัตถุ การจับคู่แบบนี้จะทำใน 2 ขั้นตอน ขั้นแรกจะต้องจับคู่จาก DTD ไปเป็น Object schema ก่อน ขั้นที่สองนำ Object schema มาจับคู่เป็น Database schema การจับคู่แบบนี้จะใช้โดยฐานข้อมูลเชิงสัมพันธ์ที่เป็น XML-enabled กับ Middleware บางตัวที่นำส่งข้อมูลระหว่างเอกสาร XML และฐานข้อมูลเชิงสัมพันธ์

2.2 เว็บเซอร์วิส

แนวคิดของเว็บเซอร์วิสนั้น หมายถึงการทำงานของโปรแกรมใดๆในลักษณะให้บริการ โดยจะถูกเรียกใช้งานจากโปรแกรมอื่นๆผ่านเว็บ โดยที่การให้บริการของโปรแกรมเหล่านี้จะมีเอกสาร WSDL (Web Service Description Language) ที่อธิบายคุณสมบัติการให้บริการ เช่น ชื่อของโปรแกรม, ชื่อของเมธอดที่สามารถเรียกใช้ได้, พารามิเตอร์ที่ส่งไปยังเมธอด และชนิดข้อมูลของพารามิเตอร์กำกับไว้ และมีการนำเสนอให้สาธารณชนรับทราบโดยนำไปลงทะเบียนไว้ที่ฐานข้อมูลแห่งหนึ่งที่เรียกว่า UDDI Registry (Universal Description, Discovery and Integration) เพื่อที่ผู้ใช้บริการสามารถค้นหาโปรแกรมเหล่านั้นได้โดยไม่ต้องรู้ที่อยู่จริงของโปรแกรม และการเรียกใช้เมธอดของโปรแกรมเหล่านั้นผ่านเว็บจากเครื่องคอมพิวเตอร์ที่มีแพลตฟอร์มใด ๆ ก็ได้ ด้วยการใช้โพรโตคอลที่ชื่อว่า SOAP (Simple Object Access Protocol) ที่อาศัยไวยากรณ์ของ XML นั่นคือเป็นข้อความธรรมดาๆที่ล้อมรอบด้วยแท็กนั่นเอง

โดยทั่วไปเว็บเซอร์วิสคือแอปพลิเคชันที่ทำงานอยู่บน HTTP โพรโตคอล ที่ประกอบด้วย API มากกว่าที่จะเป็น GUI (Graphic User Interface) และด้วยเว็บเซอร์วิสนี้เองทำให้บริษัทลูกค้าสามารถรวมเอาการทำงานทางด้านธุรกิจและข้อมูลเข้าด้วยกันได้โดยไม่ต้องคำนึงความแตกต่างทางด้านฮาร์ดแวร์และระบบปฏิบัติการ หรือแม้แต่ภาษาที่ใช้ในการพัฒนาแอปพลิเคชัน

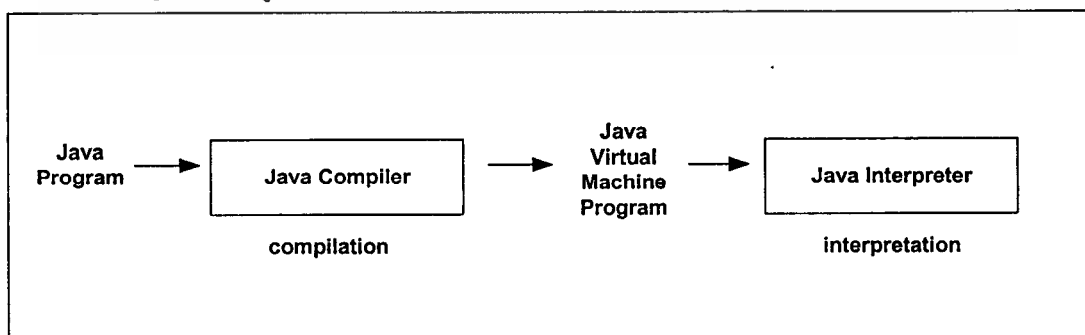
2.3 ภาษาจาวา

ภาษาจาวาเป็นภาษาเชิงวัตถุ (Object Oriented) ที่มีคุณลักษณะของการไม่ขึ้นอยู่กับระบบปฏิบัติการและฮาร์ดแวร์ (Independent platform) โดยโปรแกรมที่ถูกคอมไพล์แล้วสามารถนำไปใช้งานบนเครื่องคอมพิวเตอร์ใดๆก็ได้ และยังสามารถนำโปรแกรมนั้นมาใช้งานใหม่ (reuse) เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยอาจจะเปลี่ยนแปลงหรือเพิ่มเติมโปรแกรมบางส่วนได้ เพื่อให้เหมาะสมกับงานที่ต้องการ คุณสมบัติที่สำคัญอีกประการหนึ่งของภาษาจาวาคือในกรณีที่โปรแกรมถูกส่งผ่านระบบเครือข่ายไปทำงานบนเครื่องคอมพิวเตอร์อื่นๆ จะมีข้อกำหนดในภาษาหลายอย่างเป็นหลักประกันว่าโปรแกรมนั้นๆ จะไม่สามารถทำอันตรายต่อระบบ ข้อกำหนดดังกล่าวยกตัวอย่างเช่น ภาษาจาวาไม่ยอมให้อ้างถึงค่าในหน่วยความจำผ่านทาง Pointer และจะตรวจสอบว่าการอ้างถึงสมาชิกใน array อยู่ในขอบเขตหรือไม่ การเปลี่ยนแปลงโปรแกรมหรือค่าในหน่วยความจำซึ่งอาจเป็นอันตรายต่อโปรแกรมอื่นด้วยวิธีนี้จึงไม่สามารถเกิดขึ้นได้

จากคุณสมบัติของการเป็น Independent platform นั้น ภาษาจาวาต้องใช้วิธีการแปลภาษาทั้งแบบ Compilation และ Interpretation รวมทั้งการกำหนด Java Virtual Machine ดังรายละเอียดดังนี้

1. Compilation ทำหน้าที่วิเคราะห์ source code ซึ่งกระบวนการนี้จะได้ผลลัพธ์เป็น Executable code เรียกกระบวนการนี้ว่าเป็นการคอมไพล์โปรแกรม ข้อดีของวิธีนี้คือ Executable code ทำงานได้เร็วมากเพราะขั้นตอนในการแปลภาษาถูกแยกออกไปทำก่อนหน้าแล้ว และยังสามารถปรับให้โปรแกรมมีขนาดเล็ก มีประสิทธิภาพในการทำงาน
2. Interpretation ทำหน้าที่อ่าน source code ทีละบรรทัดแล้วแปลโปรแกรมบรรทัดนั้นเป็น Executable code และทำงาน จากนั้นอ่าน source code บรรทัดต่อไป ทำเช่นนี้เรื่อยไปจนจบโปรแกรม วิธีนี้มีทั้งการแปลภาษาและทำงานโปรแกรมเกิดขึ้นสลับกันไปจึงทำให้โปรแกรมทำงานได้ช้ากว่าแบบแรก แต่มีข้อดีคือการสร้างทำได้ง่ายกว่าและโปรแกรมมีขนาดเล็กกว่าด้วย
3. Java Virtual Machine ภาษาจาวาได้นำวิธีการคอมไพล์โปรแกรมทั้ง 2 แบบมารวมเข้าด้วยกัน เริ่มจากการคอมไพล์โปรแกรมด้วย Java Compiler ให้เป็นโปรแกรมของ Java Virtual Machine แล้วนำเอาโปรแกรมนี้ไปทำงานบนเครื่องคอมพิวเตอร์ใดๆก็ได้ที่มี Java Interpreter ดังรูปที่ 2.1



รูปที่ 2.1 ขั้นตอนการคอมไพล์โปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภายใน JVM จะมีหน่วยประมวลผลสมมติ (Virtual processor) ทำหน้าที่ประมวลผลคำสั่งของ JVM โดยปรกติหน่วยประมวลผลสมมติของ JVM ที่จำลองขึ้นบนคอมพิวเตอร์เครื่องหนึ่งจะแปลคำสั่งของ JVM (Byte code) เป็นคำสั่งของหน่วยประมวลผลในคอมพิวเตอร์เครื่องนั้นเรียกว่า Native code แล้วให้หน่วยประมวลผลทำงานคำสั่งนั้น ชุดคำสั่งของ JVM ถูกออกแบบมาเพื่อสนับสนุนการทำงานของโปรแกรมเชิงวัตถุ จึงมีคำสั่งเกี่ยวกับการสร้าง instances และการอ้างถึงสมาชิกใน instances ซึ่งไม่มีในหน่วยประมวลผลทั่วไป นอกจากนี้ JVM ยังถูกออกแบบให้สามารถจำลองได้บนหน่วยประมวลผลทั่วไปโดยคำนวณทั้งหมดบน Stacks กล่าวได้ว่า JVM เป็น Stack machine

ในปัจจุบันมีหลายบริษัทผลิตชุดโปรแกรมสำหรับพัฒนาโปรแกรมภาษาจาวา เช่น

- Java Developer Kit (JDK) ของบริษัท JavaSoft
- Visual Café ของบริษัท Symantec
- J Builder ของบริษัท Borland
- JDeveloper ของบริษัท Oracle
- Visual Age for Java ของบริษัท IBM
- Visual J++ ของบริษัท Microsoft

2.4 Java APIs for XML

XML ทำให้ข้อมูลเป็นข้อมูลที่สามารถนำไปใช้ที่ไหนก็ได้ ส่วน Java ทำให้โปรแกรมที่สร้างขึ้นสามารถนำไปใช้บนเครื่องคอมพิวเตอร์ใดๆ ก็ได้ และ Java APIs for XML ทำให้ใช้งาน XML ได้ง่ายขึ้น เมื่อนำ 3 อย่างนี้มารวมกันก็เป็นการรวมตัวกันที่ค่อนข้างจะลงตัว องค์กรส่วนใหญ่จะพบว่าการใช้ XML มีประโยชน์ในการรวมข้อมูล เช่น การนำข้อมูลของแผนกต่างๆ มาใช้ร่วมกันภายในองค์กรเอง หรือการใช้ข้อมูลร่วมกันกับองค์กรภายนอก ซึ่งเว็บเซอร์วิสเป็นแนวคิดของการสร้างการติดต่อสื่อสารกันระหว่างเครื่องคอมพิวเตอร์หลายๆเครื่อง ที่แตกต่างกันทั้งทางด้านฮาร์ดแวร์และซอฟต์แวร์ภายในองค์กรเดียวกันหรือต่างองค์กรให้เป็นระบบที่เป็นหนึ่งเดียวโดยมีรูปแบบที่เป็นกลางเอาไว้อ้างอิงถึงในการรับส่งข้อมูลถึงกัน การพัฒนาเว็บเซอร์วิสมีสิ่งที่ต้องคำนึงถึงอยู่หลายประการ เช่น การโปรแกรมโครงสร้าง การดูแลความปลอดภัย ความสามารถในการส่งข้อมูล การจัดการเกี่ยวกับทรานแซกชันและการดูแลจัดการการเชื่อมต่อสื่อสารกัน สิ่งสำคัญอีกประการหนึ่งก็คือต้องสามารถรองรับการใช้งานจากผู้ใช้จำนวนมากได้ แอปพลิเคชันจำเป็นต้องมีความยืดหยุ่นสูง ดังนั้นการพัฒนาเว็บเซอร์วิสด้วย Java APIs for XML ทำให้นักพัฒนาทำงานได้ง่ายและสะดวกมากยิ่งขึ้น Java APIs for XML โดยทั่วไปจะแบ่งได้เป็น 2 กลุ่มใหญ่ๆ คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1 Document-oriented

เป็น API ที่ทำงานกับตัวเอกสาร XML โดยตรง ประกอบด้วย

1. Java API for XML Processing (JAXP) เป็น API ที่มาช่วยให้การพัฒนาโปรแกรมที่ใช้ภาษาจาวา ทำงานกับ XML ทำได้ง่ายขึ้น โดยสามารถทำงานร่วมกับ SAX (Simple API for XML Parsing) หรือ DOM (Document Object Model) ดังนั้นขึ้นอยู่กับผู้ใช้งานจะเลือกการทำให้คอมพิวเตอร์เข้าใจถึงโครงสร้างและองค์ประกอบของเอกสาร XML ในลักษณะใด ในปัจจุบัน JAXP รองรับการใช้ XSLT (Extensible Stylesheet Language Transformations) เพื่อควบคุมการแสดงผลของข้อมูล และสามารถเปลี่ยนข้อมูลไปเป็นเอกสาร XML ในรูปแบบอื่นได้เช่น HTML นอกจากนี้ยังรองรับการใช้ namespace หรือการตั้งชื่อของอีลิเมนต์ และแอททริบิวต์ต่างๆ ในลักษณะที่ไม่ซ้ำซ้อนกัน โดยอนุญาตให้ผู้ใช้ทำงานกับ DTD (Document Type Definition) ที่อาจจะมีการตั้งชื่อที่ซ้ำกันได้ และยังสามารถทำการพาร์สจากภายในแอปพลิเคชันได้เรียกว่า “Plug ability Layer” ซึ่งเป็นการนำเอา SAX และ DOM เข้าไปรวมไว้ในแอปพลิเคชันนั่นเอง โดยที่ SAX เป็น API ที่มีลักษณะการทำงานที่ต่อเมื่อมีเหตุการณ์ใดๆ เกิดขึ้น เช่นการร้องขอข้อมูล หรือเราอาจจะเรียกโมเดลการทำงานแบบนี้ว่า event-based model นั้นหมายความว่า จะไม่มีการสร้างภาพรวมของเอกสารไว้ก่อนว่ามีโครงสร้างเป็นอย่างไร เมื่อมีแอปพลิเคชันมาร้องขอจึงจะทำงานตามเหตุการณ์ของการพาร์สซึ่งไปยังแอปพลิเคชันดังกล่าว SAX จะมองเพียงจุดเริ่มต้นและจุดสิ้นสุดของเอกสารและอีลิเมนต์เท่านั้น ในแต่ละครั้งที่ยืนยันทำการตรวจสอบโครงสร้างของ syntax มันจะส่งสัญญาณให้แอปพลิเคชันได้รู้ว่ามันกำลังทำงานถึงจุดไหน ส่วน DOM จะเป็นการกำหนดว่าเอกสาร XML จะมีการเข้าถึงและการโยกย้ายรวมไปถึงการจัดการต่างๆ ได้อย่างไรบ้าง DOM ถูกกำหนดขึ้นโดย W3C DOM Working Group ว่าเป็นกลุ่มของอินเตอร์เฟซสำหรับสร้างอ็อบเจกต์ที่เป็นตัวแทนของเอกสารให้เป็นโครงสร้างแบบต้นไม้หรือทรี การทำงานของ DOM จะเริ่มต้นขึ้นเมื่อโปรแกรมไปเรียก parser เพื่อให้ไปดึงเอกสาร XML เข้าสู่หน่วยความจำของคอมพิวเตอร์ จากนั้น DOM ก็จะทำให้โปรแกรมสามารถดึงข้อมูลมาใช้งานหรือดำเนินการต่างๆ กับข้อมูลได้ ข้อดีของ DOM คือเมื่อมีการสร้างทรีขึ้นมาแล้ว ผู้ใช้สามารถเข้าไปในโครงสร้างระดับต่างๆ ได้รวมทั้งเพิ่ม, ลบ และแก้ไขอีลิเมนต์ต่างๆ ของเอกสารได้ด้วยเมธอดของ DOM เช่น insert และ remove ซึ่งเป็นสิ่งที่แตกต่างจาก SAX เนื่องจาก SAX สามารถอ่านเอกสารได้อย่างเดียว รวมทั้ง DOM รองรับการใช้ข้อมูลแบบสุ่มได้ในขณะที่ SAX เข้าถึงข้อมูลแบบเรียงลำดับ

2. Java Architecture for XML Binding (JAXB) เป็นการจับคู่อีลิเมนต์ของเอกสาร XML ไปเป็นคลาสในภาษาจาวา ทำหน้าที่จัดเตรียมวิธีการที่สะดวกและรวดเร็วในการสร้างการจับคู่แบบ 2 ทางระหว่างเอกสาร XML และ อ็อบเจกต์ของภาษาจาวาโดยที่คอมไพล์เลอร์ของ JAXB จะสร้างคลาสเอกสารนี้เป็นเอกสารที่สแกนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของภาษาจาวาจาก DTD ผู้ใช้สามารถสร้างทริกของอ็อบเจกต์ได้ และสามารถสร้างเอกสาร XML ขึ้นมาใหม่จากทริกนี้ ในการใช้งาน JAXB สิ่งสำคัญคือต้องมี schema หรือ DTD ส่วนขั้นตอนของการทำ schema ให้เป็นคลาสมีดังนี้

- สร้าง binding schema ที่ประกอบด้วยขั้นตอนของการทำให้ schema ไปเป็นคลาส
- Schema compiler จะนำเอา DTD และ binding schema มาใช้เพื่อสร้างคลาสโดยที่ในแต่ละคลาสจะมี เมธอดชื่อ “get” และ “set” เอาไว้เข้าถึงข้อมูลของคลาสต่างๆ เราเรียกเมธอดเหล่านี้ว่า property

จะมีเมธอดที่สำคัญอยู่ 3 ตัวคือเมธอดสำหรับกระบวนการสร้างอ็อบเจกต์ที่เป็นตัวแทนของข้อมูล XML (Unmarshalling) เมธอดสำหรับกระบวนการตรวจสอบว่าอ็อบเจกต์นั้นๆเป็นไปตามข้อกำหนดของ DTD หรือไม่ (Validation) และเมธอดสำหรับกระบวนการสร้างข้อมูล XML จากอ็อบเจกต์ที่เป็นตัวแทนของข้อมูลนั่นเอง (Marshalling)

2.4.2 Procedure-oriented

เป็น API ที่ทำงานกับกระบวนการการทำงาน ประกอบด้วย

1. Java API for XML Messaging (JAXM) เป็นมาตรฐานในการส่งเอกสาร XML ผ่านเครือข่ายอินเทอร์เน็ตบนแอปพลิเคชันที่พัฒนาด้วยภาษาจาวาโดยใช้ SOAP (Simple Object Access Protocol) เป็นโปรโตคอลในการสื่อสาร และสามารถขยายไปทำงานกับโปรโตคอลอื่นในการส่งข้อความในระดับที่สูงขึ้นไปเช่น ebXML (Electronic Business XML) ใน JAXM จะอาศัยการทำงานของ Messaging provider เพื่อส่งข้อความ เริ่มต้นที่ข้อความจะถูกส่งผ่าน Messaging provider ของผู้ส่งไปที่ Messaging provider ของผู้รับก่อนที่จะถึงตัวผู้รับในที่สุด ทั้งนี้ข้อความอาจจะผ่านคนกลางมาก่อนหน้านี้ได้ ความสามารถของ Messaging provider ที่สำคัญก็คือเก็บรายละเอียดของข้อความได้ไม่ว่าจะเป็นเลขประจำตัวของข้อความ พร้อมทั้งจัดเก็บข้อความนั้นไว้และตรวจสอบดูว่าข้อความนั้นเดินทางผ่านที่ไหนมาบ้าง นอกจากนี้ข้อความใดที่ไปไม่ถึงผู้รับในครั้งแรกก็จะถูกส่งใหม่จนกว่าจะไปถึงผู้รับ ข้อดีของการใช้ Messaging provider คือลูกค้าที่ใช้เทคโนโลยี JAXM ผู้ใช้ไม่ต้องรับรู้ว่ามีขั้นตอนการทำงานอะไรบ้าง แค่เรียกใช้เมธอดของ JAXM เท่านั้น ที่เหลือ Messaging provider จะจัดการให้ทั้งหมด ในบางกรณีที่ JAXM ไม่ใช้บริการของ Messaging provider ผู้ใช้หรือที่เรียกว่า Standalone client มีข้อจำกัดในการส่งข้อความแบบ point-to-point เป็นลักษณะส่งตรงไปที่เว็บเซอร์เวอร์ที่ให้บริการแบบที่ส่งข้อความกลับมาให้เมื่อมีผู้ร้องขอ ซึ่งการร้องขอและส่งกลับของข้อความนี้จะทำงานด้วยกลไกแบบ synchronous หมายถึงเมื่อมีการส่งข้อความร้องขอไปแล้วจะรอจนกว่าได้รับข้อความตอบกลับมาจึงจะมีการส่งข้อความใหม่ตามไป ในกรณีนี้ Standalone client จะมีบทบาทเป็นแค่ผู้ส่งและผู้รับเท่านั้น ในทางตรงกันข้ามถ้าเป็น JAXM client ที่เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใช้บริการ Messaging provider สามารถจะเป็นได้ทั้ง client และ server คือร้องขอข้อมูลและรับคำร้องขอข้อมูลมาประมวลผลแล้วส่งผลลัพธ์คืนกลับให้ผู้ขอได้ เว็บเซอร์วิสที่ใช้ Messaging provider จะมีความสามารถในการทำ One-way messaging คือสามารถรับคำร้องขอเข้ามาแล้วเก็บไว้ก่อนไม่ตอบกลับในทันที นั่นก็คือเป็นลักษณะของการทำงานด้วยกลไกแบบ Asynchronous นั่นเอง โครงสร้างของข้อความใน JAXM ประกอบด้วย 2 ส่วน คือ ส่วนที่ใช้โดย SOAP เพื่อเป็นที่ใส่ข้อความที่เป็นข้อมูลของเอกสาร XML และส่วนสำหรับข้อมูลอื่นๆที่จะมีหรือไม่มีก็ได้ ในส่วนของ SOAP นั้นยังประกอบไปด้วยอ็อบเจกต์ที่ชื่อ SOAPEnvelop ซึ่งในจะแบ่งเป็น SOAPHeader และ SOAPBody ตัวข้อมูลของเอกสาร XML จะถูกเก็บไว้ใน SOAPBody ถ้าต้องการส่งข้อมูลอื่นที่ไม่ใช่เอกสาร XML เช่น รูปภาพ จำเป็นต้องมีส่วนสำหรับข้อมูลอื่นๆเพิ่มเข้าไปในส่วนประกอบของ SOAP

2. Java API for XML Registries (JAXR) ทำหน้าที่จัดหาวิธีในการเข้าถึง registries ผ่านทางอินเทอร์เน็ตซึ่ง registries นี้จะหาได้จากแหล่งข้อมูลขนาดใหญ่ที่เก็บรายชื่อของธุรกิจและบอกว่า เป็นธุรกิจที่เกี่ยวข้องกับสินค้าหรือบริการใดๆ นักพัฒนาสามารถเขียนโปรแกรมด้วยภาษาจาวาและใช้ JAXR ในการเข้าถึง registries นี้ที่อยู่บนระบบเปิดเช่น ebXML หรือ UDDI เพื่อเข้าถึง โปรแกรมต่างๆที่ธุรกิจที่เป็นสมาชิกอนุญาตให้นำไปใช้งานได้ หรือนำโปรแกรมที่อนุญาตให้ธุรกิจอื่นนำไปใช้มาไว้ที่นี้ได้เช่นกัน มีการพัฒนา DTD สำหรับเอกสาร XML แต่ละประเภท เช่น DTD มาตรฐานของใบสั่งซื้อที่ธุรกิจต่างๆสามารถนำไปใช้ได้โดยอาศัย JAXR ในการเข้าถึง ในปัจจุบัน Registries เป็นส่วนประกอบที่สำคัญของ Web service เนื่องจากทำให้ธุรกิจต่างๆสามารถรวมตัวกัน ได้แบบหลวมๆ และปรับเปลี่ยนได้ตลอดเวลา ธุรกิจที่มีการใช้ อิเล็กทรอนิกส์เข้ามาเป็นเครื่องมือในการทำธุรกิจโดยใช้ภาษาจาวา สามารถใช้ JAXR เพื่อลงทะเบียนใน registries โดยต้องมีข้อมูลคือชื่อของธุรกิจและสินค้าหรือบริการที่ธุรกิจนั้นทำ รวมทั้งประเภทของธุรกิจเพื่ออำนวยความสะดวกในการค้นหา เช่น การค้นหาธุรกิจอื่นสามารถใช้ JAXR เป็นเครื่องมือในการค้นหาได้ โดยจะต้องมีการกำหนดเงื่อนไขในการค้นหา ก่อน ซึ่ง JAXR รองรับการใช้เงื่อนไขของ SQL ด้วย และเพื่อให้มั่นใจว่าการติดต่อสื่อสารระหว่าง JAXR client กับ registries เป็นไปด้วยดี การส่งข้อความจะทำผ่าน JAXM ซึ่งทำงานอยู่หลังจากนั่นเอง

3. Java API for XML-based RPC (JAX-RPC) เป็นเครื่องมือในการพัฒนาแอปพลิเคชัน ที่พัฒนาด้วยภาษาจาวากับเอกสาร XML ให้สามารถเรียกใช้ procedure จากระยะไกลได้ หรือเรียกว่า RPC (Remote Procedure call) ซึ่งในภาษาจาวาเองมี API 2 ตัวสำหรับกระบวนการทำ RPC คือ Java IDL และ Remote Method Invocation (RMI) ซึ่ง API ทั้ง 2 ตัวนี้สามารถทำ Marshall, Unmarshall และรับส่งการเรียกใช้ procedure สิ่งที่แตกต่างกันคือ JAX-RPC จะใช้กับ XML และเอกสารนี้เป็นเอกสารที่สแกนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เว็บเซอร์วิส และ Java IDL จะใช้กับ CORBA (Common Object Request Broker Architecture) ส่วน RMI จะใช้กับ RPC ที่ทั้ง เมธอด call และ เมธอด ที่ถูกร้องขอเป็นภาษาจาวาอย่างไรก็ตาม API เหล่านี้จะทำงานอยู่เบื้องหลังเหมือนกัน โดยปรกติ RPC บนเว็บเซอร์วิสจะเป็นกลุ่มของ procedures ที่จะถูกเรียกโดยผู้ใช้ในอินเทอร์เน็ต ดังนั้นเว็บเซอร์วิสจำเป็นต้องทำให้ตัวมันเองพร้อมที่จะถูกเรียกใช้งานได้ตลอดเวลา โดยจะต้องอธิบายตัวมันเองด้วย WSDL (Web Service Description Language) นั่นก็คือระบุคุณสมบัติของแต่ละบริการ ทำให้ผู้ใช้สามารถมองหาจากเอกสาร WSDL เพื่อเข้าถึงบริการได้ ถึงแม้ว่า JAX-RPC จะติดตั้ง RPC ในลักษณะส่งและรับข้อความที่ใช้ SOAP แต่ผู้ใช้จะไม่รับรู้ถึงรายละเอียดในระดับนี้ ดังนั้น JAX-RPC จึงเป็นเหมือนรูปแบบพิเศษของ SOAP Message เมื่อเปรียบเทียบกับ JAXM แล้ว จะพบว่า JAXM จะมีประโยชน์กับนักพัฒนามากกว่าในรูปแบบของ SOAP Message ในการใช้งานกับโปรโตคอลในระดับที่สูงกว่า เช่น ebXML ซึ่งจะทำงานอยู่เหนือ SOAP 1 เลเยอร์ และยังสามารถอีกหลายประการที่ JAX-RPC ทำไม่ได้ เช่น

- ใช้กลไกของการรับส่งข้อความแบบ Asynchronous
- หาเส้นทางและส่งข้อความไปให้กับผู้รับที่มีจำนวนมากกกว่าหนึ่ง
- มีการรับประกันว่าจะส่งข้อความไปให้ถึงผู้รับ

แต่ JAX-RPC เป็นทางเลือกที่ดีกว่าสำหรับแอปพลิเคชันที่ต้องการจะหลีกเลี่ยงความซับซ้อนของ SOAP Messaging อย่างไม่รู้ดีไม่ว่าจะเลือกใช้ JAXM หรือ JAX-RPC ผู้ใช้จะได้ความง่ายในการทำ XML messaging โดยใช้ภาษาจาวา

2.5 Java Web Service Developer Pack (JWSDP)

JWSDP เป็นเครื่องมือสำหรับนักพัฒนาที่ใช้ภาษาจาวาในการสร้างและทดสอบการทำงานของเว็บแอปพลิเคชันที่ใช้เทคโนโลยี XML และเว็บเซอร์วิส ใน JWSDP ประกอบด้วย

2.5.1 Java XML Pack

- Java API for XML Messaging (JAXM)
- Soap with Attachments API for Java (SAAJ)
- Java API for XML Processing (JAXP)
- Java API for XML Registries (JAXR)
- Java API for XML-based RPC (JAX-RPC)

2.5.2 Java Server Page Standard Tag Library (JSTL)

JSTL เป็นเทคโนโลยีที่ช่วยสร้าง JSP Pages ที่สร้างด้วยภาษาสคริปต์ที่อยู่ทางฝั่งของ server เพื่อรองรับข้อมูล ที่มีการเปลี่ยนแปลงตลอดเวลาในเว็บ JSP จะแยกส่วนของการนำเสนอออกจากเอกสารนี้เป็นเอกสารที่สแกนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยามให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งยังมีให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนของโปรแกรมมีง โดยการใช้อ็ลิตีเมนต์และ Tags ใน JSP ยกตัวอย่างเช่นถ้าต้องการติดต่อกับฐานข้อมูลจะต้องระบุถึงวิธีติดตั้ง Driver สำหรับฐานข้อมูล, วิธีสร้างการติดต่อสื่อสารที่ปลอดภัยและวิธีการทำงานกับข้อมูล สิ่งที่เกิดขึ้นก็คือจะต้องระบุสิ่งเหล่านี้ลงใน JSP ทุกๆหน้าที่มีการทำงานในส่วนนี้ JSP Tags จะช่วยลดงานในส่วนนี้ให้โดยการสร้าง Tags แล้วนำมาใช้ใน JSP ทดแทนการกำหนดวิธีต่างๆที่กล่าวมาแล้วในทุกๆหน้า อย่างไรก็ตาม ปัญหาที่เกิดจากการใช้ JSP Tags ที่มีรูปแบบไม่เหมือนกันทำให้ขาดคุณสมบัติของการนำไปทำงานกับเครื่องคอมพิวเตอร์ใดๆ ดังนั้นเพื่อแก้ไขปัญหานี้จึงมีการสร้างมาตรฐานโดย Java Community Process ได้พัฒนา JSTL ขึ้นมาให้เป็นมาตรฐานของ Tags และนำไปเป็นส่วนหนึ่งของ J2EE

2.5.2.1 การสร้าง Tags ใน JSTL

นักพัฒนาสามารถสร้าง Tags ขึ้นมาใช้งานเองได้นอกเหนือจาก Tags สำเร็จรูปที่มีมาให้ประโยชน์ของการสร้าง Tags คือช่วยลดการทำงานที่ซ้ำซ้อนและสามารถนำกลับมาใช้ใหม่ได้ โดยนำ Tags ที่สร้างเก็บไว้ในไฟล์ที่มีโครงสร้างเป็นเอกสาร XML ที่เรียกว่า Tag Library Descriptor (TLD) และสร้างคลาสสำหรับการทำงานกับ Tags ที่เรียกว่า tag handle ตัวอย่างการประกาศการใช้ Tags ใน JSP และการใช้งาน

ตัวอย่างด้านล่างเป็นการสร้างคลาสที่ชื่อว่า SimpleTag เป็นคลาสประเภท tag handle

```
import javax.servlet.jsp.*;
import javax.servlet.jsp.tagext.*;
import java.io.*;
public class SimpleTag extends TagSupport {
    public int doStartTag() {
        try {
            JspWriter out = pageContext.getOut();
            out.print("Hello Custom tag world...");
        } catch(IOException e) {
            System.out.println(e);
        }
        return(SKIP_BODY);
    }
}
```

เมื่อสร้างคลาสด้านบนแล้ว ต่อจากนั้นให้สร้างไฟล์ TLD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Simple.tld

```
<? Xml version="1.0" encoding = "ISO-8859-1" ?>
<! DOCTYPE taglib
PUBLIC "-//Sun Microsystems, Inc.//DTD JSP Tag Library 1.1//EN"
"http://java.sun.com/j2ee/dtds/web-jsptaglibrary_1_1.dtd">
<taglib>
<tlibversion>1.0</tlibversion>
<jspversion>1.1</jspversion>
<shortname>simple</shortname>
<urn></urn>
<info>Simple tag library</info>

<tag>
  <name>simple</name>
  <tagclass>SimpleTag</tagclass>
  <info>Simple example</info>
  <bodycontent>EMPTY</bodycontent>
</tag>
</taglib>
จากนั้นสามารถนำ Tags ที่สร้างไปใช้ใน JSP ได้
<HTML>
<HEAD>
<%@ taglib uri="Simple.tld" prefix="pre" %>
<TITLE>A simple example</TITLE>
</HEAD>
<BODY>
<H1><pre:simple/></H1>
</BODY>
</HTML>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างด้านบนแสดงถึงการที่ผู้ใช้สร้าง Tags ขึ้นใช้เอง ในขณะที่เดียวกัน JSTL ได้จัดเตรียม Tags สำเร็จรูปหลายๆประเภทที่สามารถนำไปใช้ได้ทันทีดังรายละเอียดในตารางที่ 2.1

ตารางที่ 2.1 JSTL Libraries and Prefixes

Area	Function	Tags	TLD	Prefix
Core	Expression Language Support Flow Control Import	<out>, <set>, <remove>, <catch> <forEach>, <forEachToken>, <if>, <choose>, <when>, <otherwise> <import>, <param>, <url>, <redirect>	/jstl-c	C
Area	Function	Tags	TLD	Prefix
fmt	Locale Message formatting Numbers and date formatting	<locale> <bundle>, <message>, <param>, <requestEncoding> <formatNumber>, <parseNumber>, <timeZone>, <formatDate>, <parseDate>	/jstl-fmt	Fmt
Area	Function	Tags	TLD	Prefix
Database	SQL	<driver>, <transaction>, <query>, <update>, <param>	/jstl-sql	Sql
Area	Function	Tags	TLD	Prefix
XML	Core Flow Control Transformation	<parse>, <out>, <set> <forEach>, <if>, <choose>, <when>, <otherwise> <transform>, <param>	/jstl-x	X

2.5.3 Tomcat Web server

ประกอบด้วย

- Java Servlet เป็นแอปพลิเคชันที่ทำงานอยู่บนฝั่งของเว็บเซิร์ฟเวอร์โดยบรรจุอยู่ใน Servlet engine ที่อาจจะประกอบด้วย Servlet หลายๆตัวรวมกัน ในการติดต่อสื่อสารกับ Client นั้น Servlet engine จะต้องทำงานร่วมกับเว็บเซิร์ฟเวอร์เมื่อใดที่มีคำร้องขอมาจาก Client ว่าต้องการจะติดต่อกับ Servlet ทางเว็บเซิร์ฟเวอร์จะส่งคำร้องขอนั้นมาให้ Servlet

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

engine เพื่อทำหน้าที่เรียก Servlet ที่ Client ต้องการขึ้นมาประมวลผล แล้วส่งผลลัพธ์กลับไปให้ Servlet engine เพื่อส่งต่อไปให้เว็บเซิร์ฟเวอร์และถึง Client ตามลำดับ

- Java Server Pages (JSP) เป็นภาษาสคริปต์ชนิดหนึ่งที่ใช้บนเว็บแอปพลิเคชัน โดยมีพื้นฐานมาจากภาษาจาวา และจะถูกเรียกใช้โดย JSP Container ที่เป็นส่วนประกอบอยู่ในเว็บเซิร์ฟเวอร์อยู่แล้ว โดยทั่วไป JSP Container เป็นตัวรับคำร้องขอจาก Client แล้วส่งผ่านไปยัง JSP และส่งผลลัพธ์ที่ได้จาก JSP กลับไปให้ Client

2.5.4 “Ant” Build Tools

เป็นเครื่องมือช่วยสำหรับการคอมไพล์โปรแกรม โดยอาศัยการทำงานร่วมกับไฟล์ชื่อ build.xml ที่บอกถึงค่าต่างๆที่กำหนดไว้ในลักษณะโครงสร้างแบบต้นไม้ การใช้งานนั้นเป็นการใส่คำสั่งแบบ command line เช่น

```
c:\>ant build
```

```
c:\>ant remove
```

```
c:\>ant install...
```

ผลลัพธ์ที่ได้จะขึ้นอยู่กับข้อกำหนดค่าต่างๆใน build.xml เช่น กำหนดให้คอมไพล์โปรแกรมจาวา และคัดลอกไปไว้ที่ไครเรททอรีย่อยอื่นๆ รวมทั้งสามารถกำหนดผู้ใช้งานที่มีสิทธิ์เข้าถึงเว็บเซิร์ฟเวอร์ได้ เช่น

```
<project name=“Battery Sales” default=“build” basedir=“.” >
```

```
<target name = “init”>
```

```
</target>
```

```
<property file = “${user.home}/build.properties” />
```

```
<property name = “context-name” value = “gssserver” />
```

```
<property name = “context-path” value = “server” />
```

```
<property name = “build-path” value = “${context-path}/build” />
```

```
<property name = “war-file” value = “${context-name}.war” />
```

```
<property name = “url” value = http://localhost:8080/manager />
```

```
<property name = “username” value = “rucha” />
```

```
<property name = “password” value = “java” />
```

```
<taskdef name = “deploy” classname = “org.apache.catalina.ant.DeployTask” />
```

```
<taskdef name = “undeploy” classname = “org.apache.catalina.ant.UndeployTask” />
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

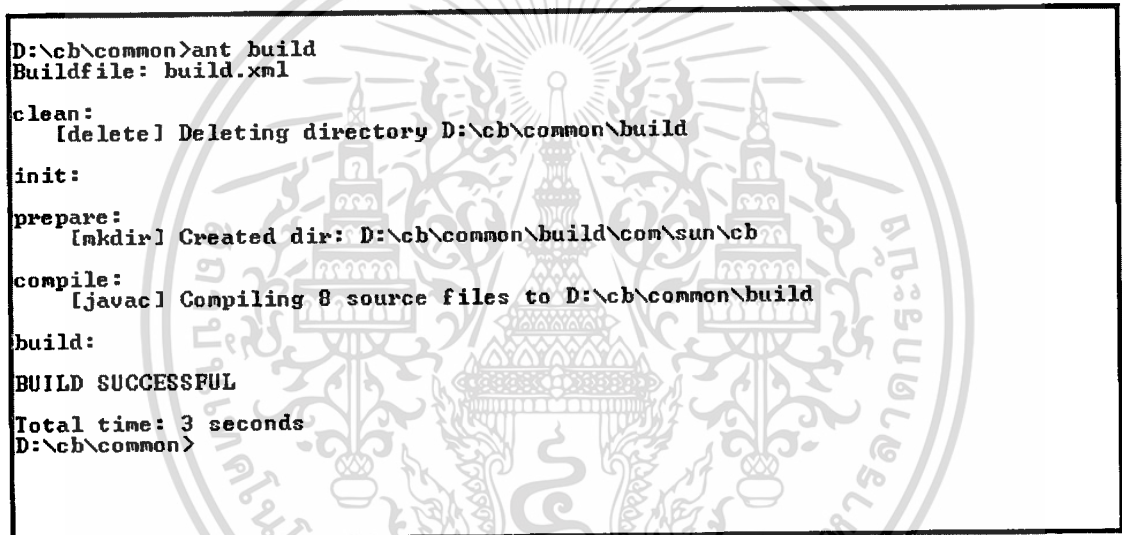
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<target name = "package" depends = "build" description = "Packages the WAR file" >
    <echo message = "Packaging the WAR..." />
    <delete file = "dist/${war-file}" />
    <jar jarfile = "dist/${war-file}" >
        <fileset dir = "${build-path}" />
    </jar>
</target>

```

ตัวอย่างผลลัพธ์จากการใช้คำสั่ง ant build



```

D:\cb\common>ant build
Buildfile: build.xml
clean:
  [delete] Deleting directory D:\cb\common\build
init:
prepare:
  [mkdir] Created dir: D:\cb\common\build\com\sun\cb
compile:
  [javac] Compiling 8 source files to D:\cb\common\build
build:
BUILD SUCCESSFUL
Total time: 3 seconds
D:\cb\common>

```

รูปที่ 2.2 ผลลัพธ์การใช้คำสั่ง “ant build”

2.5.5 “deploytool” Tools

เป็นเครื่องมือสำหรับการติดตั้งเว็บแอปพลิเคชันบน Tomcat server โดยติดตั้งจากไฟล์ที่มีนามสกุล .war เป็น package ที่ถูกสร้างขึ้นจากการทำงานของคำสั่ง ant

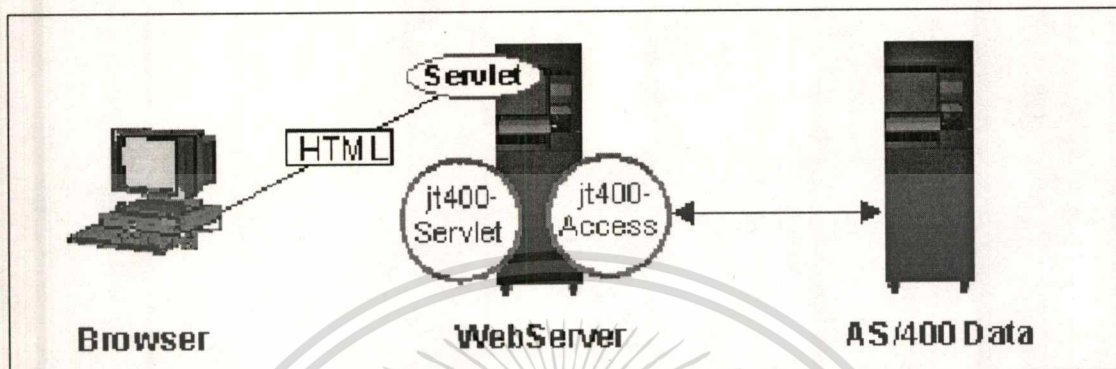
2.5.6 Registry Server

เป็น server ที่ Java WSDP เตรียมไว้ให้ใช้ทดสอบการลงทะเบียนของเว็บแอปพลิเคชันเพื่อประกาศถึงบริการของตัวเองใน UDDI

2.6 AS/400 Toolbox for Java

AS/400 Toolbox for Java เป็นชุดของคลาสที่สร้างโดยภาษาจาวา จัดเตรียมไว้สำหรับโปรแกรมที่สร้างด้วยภาษาจาวาและต้องการติดต่อกับฐานข้อมูลหรือระบบไฟล์ข้อมูลที่อยู่บนเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AS/400 ด้วยคลาสเหล่านี้ผู้พัฒนาสามารถสร้างแอปพลิเคชันแบบ Client/Server. Java Applet และ Java Servlet ที่ทำงานกับข้อมูลของ AS/400 โดยมีรูปแบบการทำงานดังรูปที่ 2.3



รูปที่ 2.3 รูปแบบการรับส่งข้อมูลระหว่าง Client กับ AS/400

คลาส Servlet และคลาส Access ที่อยู่บนเว็บเซิร์ฟเวอร์จะทำงานร่วมกันในการเข้าถึงข้อมูลบน AS/400 รูปที่ 2.3 แสดงถึงการทำงานของคลาส Servlet ระหว่างบราวเซอร์, เว็บเซิร์ฟเวอร์ และ AS/400 เริ่มจากบราวเซอร์สร้างการติดต่อมาที่เว็บเซิร์ฟเวอร์ที่มีไฟล์ชื่อ jt400Servlet.jar และ jt400Access.jar ติดตั้งอยู่โดยคลาส Servlet อาศัยการทำงานบางอย่างของคลาส Access ในการเข้าถึงข้อมูลบน AS/400 จากนั้นเว็บเซิร์ฟเวอร์จะติดต่อมาที่ AS/400 เพื่อขอข้อมูลตามที่ได้รับคำร้องขอจากบราวเซอร์ในกลุ่มคลาส Servlet ประกอบด้วยคลาส 3 คลาสดังนี้

2.6.1 คลาส RowData

ทำหน้าที่กำหนดตำแหน่งและเข้าถึงข้อมูลในตำแหน่งที่ต้องการ โดยมีคลาสลูก 3 คลาสคือ

- ListRowData คลาสนี้ทำหน้าที่เป็นตัวแทนของชุดข้อมูลต่างๆ ในระบบของ AS/400 เช่น ชุดข้อมูลของผู้ใช้ในระบบ, ชุดข้อมูลของชื่อเครื่องพิมพ์ และชุดข้อมูลของไฟล์
- RecordListRowData คลาสนี้ทำหน้าที่เป็นตัวแทนของชุดข้อมูลที่ได้มาจากรฐานข้อมูลบน AS/400 เช่น ชุดของพารามิเตอร์ในโปรแกรม หรือชุดของข้อมูลที่ได้มาจากไฟล์ข้อมูลลูกค้า โดยที่ข้อมูลที่ส่งไปมาระหว่าง AS/400 และ โปรแกรมจะต้องเปลี่ยนให้อยู่ในรูปแบบของ AS/400 และภาษาจาวา
- ResultSetRowData คลาสนี้ทำหน้าที่เป็นตัวแทนของชุดข้อมูลที่ได้มาจากการทำงานตามคำสั่งในประโยค SQL ที่ผ่านมาจาก JDBC API

2.6.2 คลาส RowMetaData

ทำหน้าที่กำหนดส่วนที่ทำหน้าที่ติดต่อหรืออินเตอร์เฟสกับอ็อบเจกต์ RowData เพื่อให้ได้มาซึ่งข้อมูลของคอลัมน์ต่างๆที่มีใน RowData นั้น เช่น ชื่อของคอลัมน์, ประเภทของข้อมูลในคอลัมน์ และขนาดของข้อมูล เป็นต้น ประกอบด้วยคลาสลูก 3 คลาสคือ

- ListMetaData คลาสนี้ทำหน้าที่ให้ได้มาซึ่งข้อมูลของคอลัมน์ใดๆ และเปลี่ยนคุณสมบัติของคอลัมน์ได้ด้วยคลาสนี้
- RecordFormatMetaData นักพัฒนาสามารถกำหนดรูปแบบของเรคคอร์ดที่นำมารองรับข้อมูลที่ได้รับมา
- ResultSetMetaData ทำหน้าที่ส่งข้อมูลของคอลัมน์ต่าง ๆ ของอ็อบเจกต์ ResultSetRowData กลับมาให้

2.6.3 คลาส Converter

ทำหน้าที่แปลงข้อมูล (row data) ให้อยู่ในรูปของชุดข้อมูล (array) ผลลัพธ์จะอยู่ในรูปแบบของ HTML สามารถนำไปแสดงได้ในไฟล์ HTML ประกอบด้วยคลาสลูก 3 คลาส คือ

- StringConverter ทำหน้าที่แปลงข้อมูล (row data) ให้อยู่ในรูปชุดของข้อมูล (array)
- HTMLFormConverter เป็นคลาสย่อยมาจากคลาส StringConverter ทำหน้าที่แปลงข้อมูล (row data) ให้อยู่ในรูปของตารางทีละ 1 ตารางในรูปแบบของ HTML โดยสามารถจัดการกับคุณสมบัติของตารางได้เช่น ความกว้าง, การจัดชิดซ้ายขวา เป็นต้น
- HTMLTableConverter เป็นคลาสย่อยมาจากคลาส StringConverter เช่นเดียวกัน แต่สามารถแปลงข้อมูล (row data) ให้อยู่ในรูปของตารางทีละหลายๆตาราง

2.7 JDBC API (Java Database Connectivity Application Programming Interface)

JDBC เป็น API ใช้ในการติดต่อกับฐานข้อมูลแบบเชิงสัมพันธ์โดยทำงานกับประโยค SQL ใน JDBC ประกอบด้วยชุดของคลาสและอินเตอร์เฟสที่สร้างด้วยภาษาจาวาเพื่อนำมาใช้ในโปรแกรมที่พัฒนาด้วยภาษาจาวาโดยเฉพาะ และโปรแกรมที่สร้างขึ้นสามารถนำไปใช้ในการติดต่อกับฐานข้อมูลแบบเชิงสัมพันธ์ใดๆก็ได้

การทำงานของ JDBC นั้นประกอบด้วย 3 ขั้นตอนดังนี้

1. สร้างการติดต่อไปที่ฐานข้อมูล
2. ส่งประโยค SQL
- 3. ประมวลผลจากผลลัพธ์ที่ได้

ตัวอย่างการทำงานทั้ง 3 ขั้นตอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// สร้างการติดต่อไปที่ฐานข้อมูล
Connection con = DriverManager.getConnection (
    "jdbc:odbc:wombat", "login", "password");

Statement stmt = con.createStatement();
// ส่งคำสั่งในรูปประโยค SQL ไปที่ฐานข้อมูล
ResultSet rs = stmt.executeQuery("SELECT a, b, c FROM Table1");
//ทำงานกับผลลัพธ์ที่ได้มาจากฐานข้อมูล
while (rs.next()) {
    int x = getInt("a");
    String s = getString("b");
    float f = getFloat("c");
}

```

2.7.1 ชนิดของ JDBC Driver

สามารถแบ่งได้เป็น 4 ประเภทดังนี้

1. JDBC-ODBC Driver ทำให้ JDBC สามารถติดต่อกับฐานข้อมูลโดยใช้ ODBC driver เป็นตัวกลาง ซึ่ง client แต่ละเครื่องที่ต้องการเชื่อมต่อจะต้องมี ODBC binary code อยู่ในเครื่อง driver ประเภทนี้เหมาะสำหรับสภาพแวดล้อมที่ไม่มีปัญหาในการติดตั้ง driver ในเครื่อง client หรือเหมาะสำหรับแอปพลิเคชันที่อยู่บนฝั่ง server ที่สร้างด้วยภาษาจาวาในลักษณะสถาปัตยกรรมแบบทรี-tier
2. Native-API partly-Java driver ทำหน้าที่เปลี่ยนคำสั่งที่ส่งมาจาก JDBC ให้อยู่ในรูปแบบที่ใช้กับฐานข้อมูล Oracles, Sybase, Informix, DB2 และอื่นๆ โดยที่เครื่องของ client จะต้อง มี driver ติดตั้งอยู่ด้วย
3. JDBC-Net pure Java driver ทำหน้าที่แปลงคำสั่งที่ส่งมาจาก JDBC ให้อยู่ในรูปแบบ โพรโตคอลที่เป็นกลาง สามารถนำไปใช้ได้ไม่เฉพาะกับฐานข้อมูลเท่านั้น
4. Native-protocol pure Java driver ทำหน้าที่เปลี่ยนคำสั่งที่ส่งมาจาก JDBC ให้อยู่ในรูปแบบ โพรโตคอลที่นำไปใช้กับฐานข้อมูลโดยตรง ซึ่งทำให้เครื่อง client สามารถส่งคำสั่งไปที่ DBMS server ได้โดยตรง ซึ่งโพรโตคอลเหล่านี้จะอยู่ในรูปแบบที่เป็นระบบปิดขึ้นอยู่กับแต่ละบริษัทที่ผลิตระบบฐานข้อมูลเชิงสัมพันธ์

2.7.2 JDBC2XML Class

คลาส JDBC2XML เป็นคลาสที่ทำหน้าที่ทำงานตามคำสั่งที่มาจากประโยค SQL แล้วส่งผลลัพธ์ (Result set) กลับมาเป็นเอกสาร XML เพื่อให้โปรแกรมนำไปใช้งานต่อไป เมธอดของคลาส JDBC2XML มีดังนี้

1. encodeXML() เป็นตัวกำหนดกฎในการสร้างเอกสาร XML โดยการออกแบบโครงสร้างของเอกสารที่รวมเอา metadata และ ข้อมูลของชุดของผลลัพธ์ไว้ด้วยกัน โดยมีอีลิเมนต์หลัก <resultset> ซึ่งประกอบด้วยอีลิเมนต์ลูกอีก 2 ตัวคือ <metadata> และ <record>
 - 1.1. อีลิเมนต์ <metadata> ประกอบด้วยอีลิเมนต์ <field> ที่มาจากแต่ละคอลัมน์ในฐานข้อมูลว่ามีชื่อแอททริบิวต์และชนิดของข้อมูลเป็นอะไร
 - 1.2. อีลิเมนต์ <record> ประกอบด้วยอีลิเมนต์ <record> ของแต่ละ row ในชุดของผลลัพธ์ และตัว <record> เองก็ประกอบด้วยอีลิเมนต์ <field> ซึ่งเปลี่ยนมาจากคอลัมน์ต่างๆในชุดของผลลัพธ์
2. writeXML() ทำหน้าที่สร้างเอกสาร XML จากชุดของผลลัพธ์ที่ได้ หลังจากออกแบบโครงสร้างของเอกสาร XML แล้ว จะเรียกใช้เมธอด writeXML() เพื่อให้ JDBC result set เป็นเอกสาร XML ด้วยโครงสร้างที่ได้กำหนดไว้ ขั้นตอนการทำงานจะเริ่มจากการสร้าง stringBuffer เพื่อเก็บ output ที่จะได้ออกมา จากนั้นจะสร้างอีลิเมนต์ <metadata> ก่อนแล้วสร้างทีละ record เมื่อสร้างจนครบจำนวนแล้วก็จะยกเลิกอีลิเมนต์ <resultset> ทิ้งไปและได้เป็นเอกสาร XML ออกมาแทน
3. execute() ทำหน้าที่ค้นหาและส่งผลลัพธ์กลับมาในรูปแบบเอกสาร XML การทำงานตามคำสั่งในประโยค SQL กับ JDBC data source จนได้ผลลัพธ์ออกมาเป็นเอกสาร XML นั้นจะต้องใช้พารามิเตอร์ 5 ตัว คือ JDBC driver, JDBC URL, USER ID, Password และประโยค SQL การทำงานเริ่มต้นด้วยการ register และเปิดการทำงานของ JDBC driver และเชื่อมต่อไปยังฐานข้อมูลและสร้างประโยค SQL ในรูปของอ็อบเจกต์ขึ้นมาเพื่อทำงานตามคำสั่งในประโยค SQL โดยใช้เมธอด executeQuery() จากนั้นเมื่อได้ผลลัพธ์กลับมาแล้วจะแปลงให้เป็นเอกสาร XML ด้วยเมธอด writeXML() จากนั้นปิดการเชื่อมต่อและ result set แล้วส่งเอกสาร XML ไปยังผู้ขอ ถ้ามีข้อผิดพลาดเกิดขึ้นจะใช้อีลิเมนต์ <error> เป็นตัวจัดการส่งข้อความไปยังผู้ขอ

2.7.3 XMLDataGateway Servlet

XML servlet ใช้คลาส JDBC2XML เพื่อจัดเตรียมช่องทางการติดต่อระหว่างเอกสาร XML บนโพรโตคอล HTTP กับ JDBC data source ในการทำงานนั้น XML servlet จะต้องสร้างคลาสซึ่งเป็นคลาสลูกของคลาส javax.servlet.http.HttpServlet

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.4 การใช้ XML สำหรับแอปพลิเคชันที่ใช้ JDBC แบบกระจาย

ด้วยการอาศัยคลาสที่ชื่อว่า WebRowSet สามารถสร้าง JDBC result set และแปลงให้เป็นเอกสาร XML ส่งผ่านเครือข่ายด้วยโพรโทคอล TCP/IP ไปที่ client เพื่อจัดการกับข้อมูล (manipulate) แล้วส่งข้อมูลกลับมาที่ data source เพื่อปรับปรุงให้ทันสมัย

2.7.5 Rowset Interface

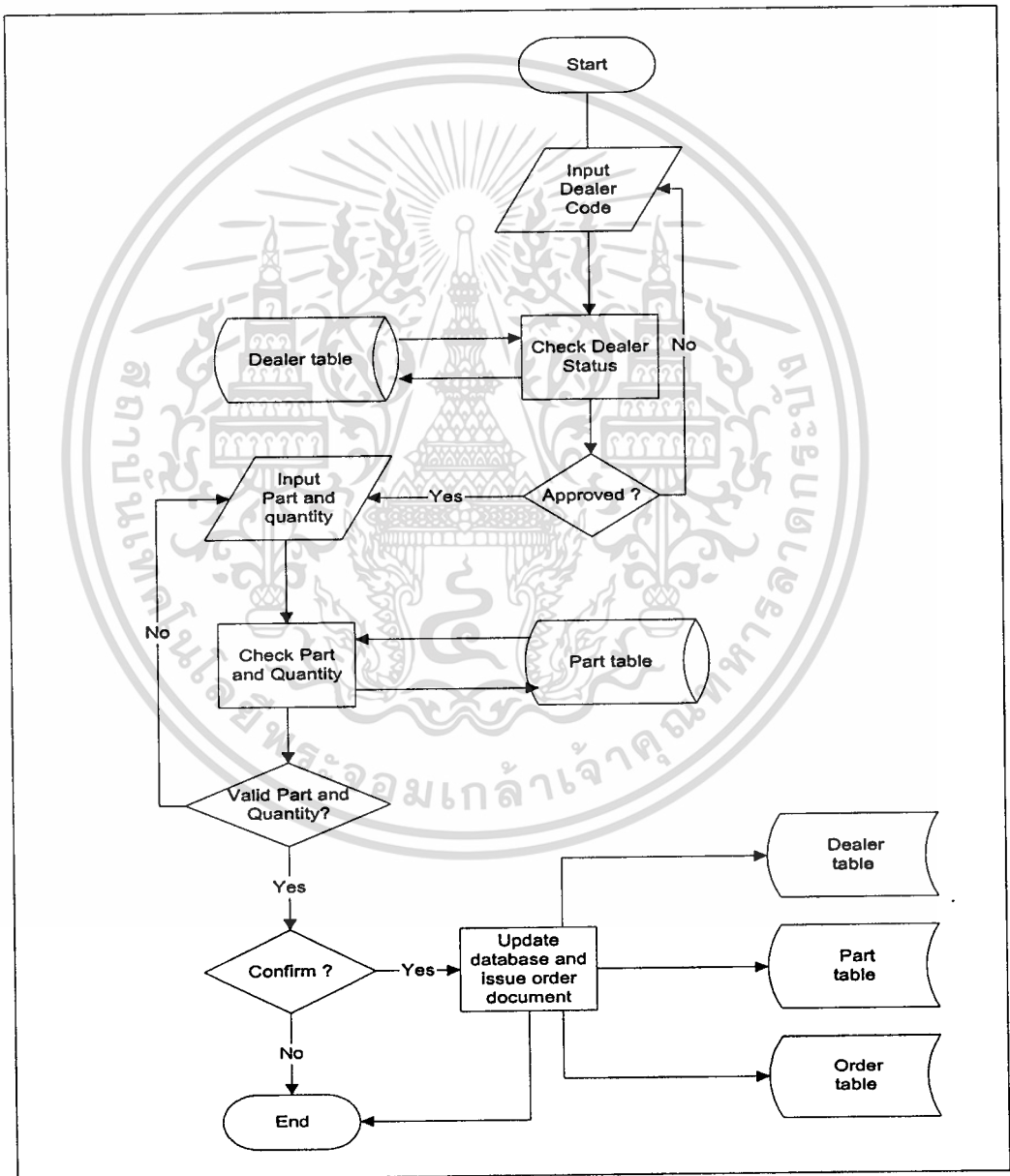
Rowset interface อยู่ใน JDBC 2.0 API สามารถใช้กรอบการทำงาน (Template) ซึ่งมีมาให้แล้วสร้างคลาสต่างๆที่เกี่ยวข้องกับการทำงานได้ สำหรับ data source นั้นเป็นได้ตั้งแต่ไฟล์ตัวอักษร ทั่วๆไปจนถึงเอกสาร XML คุณสมบัติของ JDBC result set ก็ยังสามารถปรับปรุงแก้ไขและส่งผ่านเครือข่ายได้ ที่สำคัญสามารถเลื่อนตำแหน่งไปมาภายใน result set ได้ การทำงานทั้งหมดนี้ทำได้ผ่านทางคลาสที่ชื่อว่า WebRowSet นอกจากนั้นยังสามารถใช้ข้อมูลในเอกสาร XML กลับมาสร้างการเชื่อมต่อเพื่อปรับปรุงข้อมูลใน data source ได้อีกครั้ง

บทที่ 3

ศึกษาและวิเคราะห์ระบบปัจจุบัน

3.1 ขั้นตอนการทำงาน

ในที่นี้เป็นการศึกษาและวิเคราะห์ระบบขายสินค้า, ระบบสั่งซื้อสินค้าและระบบสินค้าคงคลังของตัวแทนจำหน่าย กับระบบขายสินค้าของโรงงานผลิต



รูปที่ 3.1 Flowchart โปรแกรมบันทึกการขายของตัวแทนจำหน่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

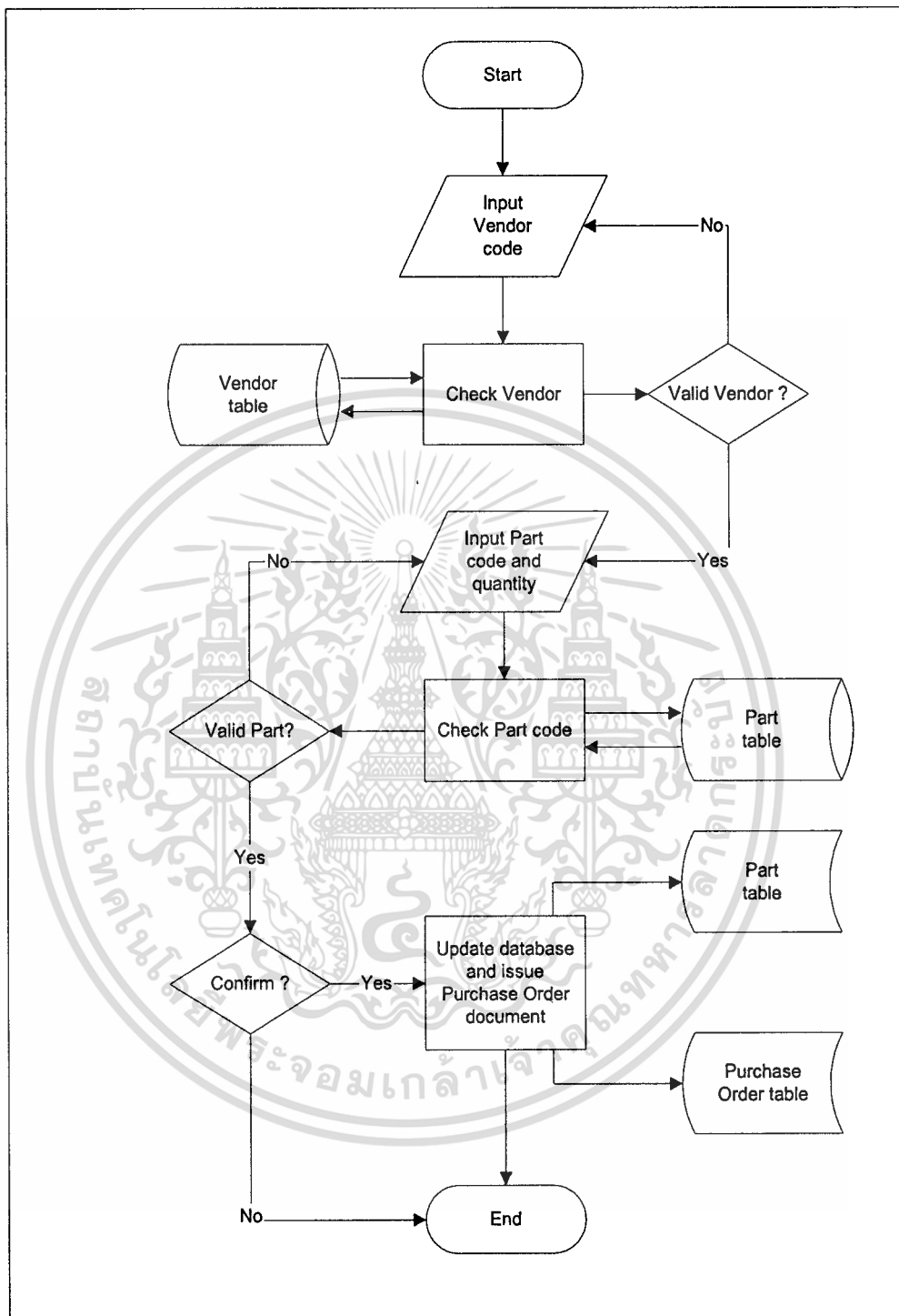
1. ระบบขายสินค้าในส่วนของบริษัทรายการสั่งซื้อโดยลูกค้าของตัวแทนจำหน่าย

ในปัจจุบันตัวแทนจำหน่ายมีพนักงานขายรับผิดชอบลูกค้าในแต่ละภาคของประเทศ แบ่งเป็น กรุงเทพฯ, ภาคกลาง, ภาคเหนือ, ภาคตะวันออกเฉียงเหนือ และภาคใต้ เมื่อพนักงานขายรับคำสั่งซื้อจากลูกค้าไม่ว่าจะเป็นทางโทรศัพท์, โทรสาร และจากการพบลูกค้าในแต่ละเดือนแล้ว จะนำเอาคำสั่งซื้อนั้นมาป้อนใส่ในโปรแกรมบันทึกการขายสั่งซื้อสินค้าในระบบขายสินค้าตามขั้นตอนในรูปที่ 3.1 โปรแกรมจะให้รหัสของลูกค้าเพื่อตรวจสอบว่ามียอดลูกหนี้ค้างชำระเกินวงเงินเครดิตที่ให้ไว้หรือไม่, จำนวนวันเครดิต พร้อมทั้งส่วนลดทางการค้าของ ลูกค้าแต่ละรายจากแฟ้มข้อมูลหลักของลูกค้าในเซิร์ฟเวอร์ของตัวแทนจำหน่าย และแสดงรายละเอียด เช่น ชื่อ, ที่อยู่ ที่หน้าจอแสดงผล ถ้าตรวจสอบแล้วพบว่าไม่มียอดลูกหนี้ค้างชำระเกินวงเงินเครดิตที่ให้ไว้หรือไม่พบรหัสลูกค้านี้ จะแสดงข้อความให้ผู้ป้อนข้อมูลทราบและกลับไปจุดเริ่มต้นของโปรแกรม แต่ถ้าตรวจสอบแล้วพบว่าผ่านเงื่อนไขที่ตั้งไว้ทั้งหมดจะแสดงหน้าจอให้ผู้ใช้ป้อนรหัสสินค้าและจำนวนที่ต้องการ จะตรวจสอบความปริมาณสินค้าในคลังสินค้าจากแฟ้มข้อมูลสินค้าว่ามีเพียงพอหรือไม่ ถ้ามีเพียงพอจะผ่านกระบวนการคำนวณจำนวนเงินรวมและเพิ่มรายการข้อมูลและเปลี่ยนแปลงรายการข้อมูลในแฟ้มข้อมูลที่เกี่ยวข้อง ก่อนจะพิมพ์เป็นแบบฟอร์มเอกสารทางเครื่องพิมพ์

2. ระบบสั่งซื้อสินค้าในส่วนของการบันทึกการขายสั่งซื้อสินค้าจากโรงงานผลิตของตัวแทน

จำหน่าย

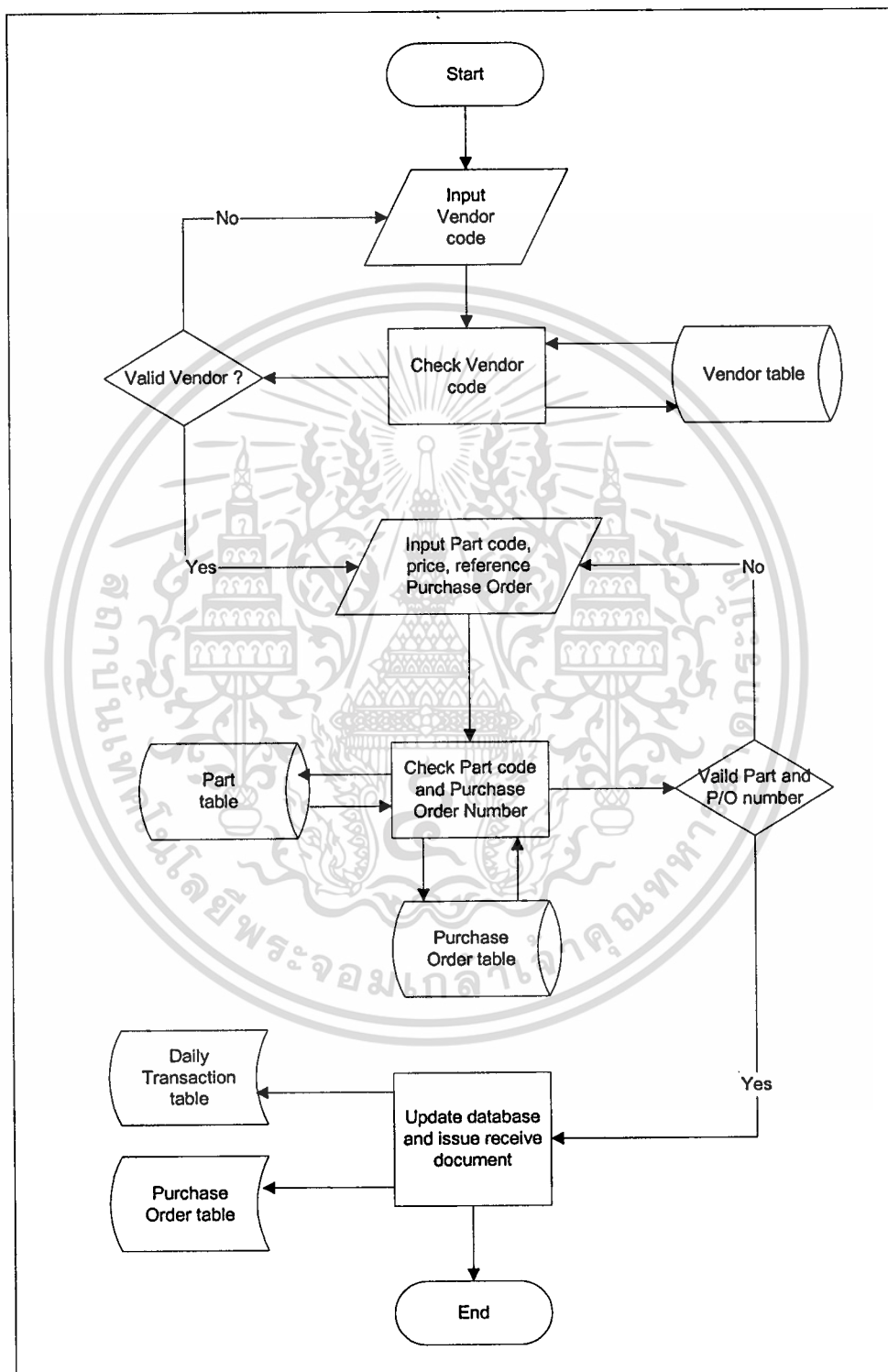
ในการสั่งซื้อสินค้าจากโรงงานผลิต ตัวแทนจำหน่ายจะประมาณการจำนวนที่จะขายได้ล่วงหน้า 1 เดือนและจัดทำเอกสารใบสั่งซื้อส่งให้กับโรงงาน โดยฝ่ายจัดซื้อใช้โปรแกรมบันทึกการขายสั่งซื้อในระบบสั่งซื้อสินค้าป้อนข้อมูลให้กับระบบ โดยต้องใส่รหัสผู้จำหน่าย (Vendor) เพื่อให้โปรแกรมตรวจเช็คความถูกต้องและดึงเอารายละเอียดของผู้จำหน่ายมาแสดงที่หน้าจอแสดงผล จากนั้นโปรแกรมจะให้ผู้ใช้ป้อนรหัสสินค้าและจำนวนสั่งซื้อ ขั้นตอนต่อไปโปรแกรมจะตรวจสอบความถูกต้องของรหัสสินค้า ถ้ารหัสสินค้าไม่ถูกต้องจะให้ผู้ใช้ใส่รหัสสินค้าที่ถูกต้องก่อน ถ้ารหัสสินค้าถูกต้องทั้งหมดจะให้ผู้ใช้ยืนยันรายการสั่งซื้อนี้ ถ้ายืนยันรายการนี้โปรแกรมจะเพิ่มรายการข้อมูลและปรับปรุงรายการข้อมูลที่เพิ่มข้อมูลต่างๆที่เกี่ยวข้อง และพิมพ์แบบฟอร์มเอกสารใบสั่งซื้อทางเครื่องพิมพ์ก่อนจบการทำงาน ถ้าผู้ใช้ยกเลิกรายการนี้โปรแกรมจะจบการทำงาน ดังแสดงในรูปที่ 3.2



รูปที่ 3.2 Flowchart โปรแกรมบันทึกการสั่งซื้อสินค้าจากโรงงานของ ตัวแทนจำหน่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ระบบสินค้าคงคลังในส่วนของโปรแกรมบันทึกการรับเข้าสินค้าของตัวแทนจำหน่าย



รูปที่ 3.3 Flowchart โปรแกรมบันทึกการรับเข้าสินค้าจากโรงงานของตัวแทนจำหน่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในแต่ละวัน โรงงานผลิตจะส่งสินค้าให้กับตัวแทนจำหน่าย เมื่อตัวแทนจำหน่ายได้รับสินค้าพร้อมเอกสารอินวอยซ์ ซึ่งในเอกสารอินวอยซ์นั้นประกอบด้วยรายละเอียดของรหัสสินค้า, จำนวนที่ส่ง และอ้างอิงถึงเลขที่เอกสารใบสั่งซื้อที่โรงงานผลิตได้รับมาจากตัวแทนจำหน่าย หลังจากตรวจเช็คจำนวนรับเข้าตรงกับเอกสารอินวอยซ์แล้ว จะนำข้อมูลในเอกสารอินวอยซ์มาป้อนใส่ในโปรแกรมรับเข้าสินค้าของระบบสินค้าคงคลัง โดยจะอ้างถึงรหัสผู้จำหน่ายตรวจสอบชื่อที่อยู่ว่าถูกต้องหรือไม่ ถ้าไม่ถูกต้องโปรแกรมจะแสดงข้อความให้ผู้ใช้ทราบพร้อมกับให้ใส่รหัสที่ถูกต้อง จากนั้นผู้ใช้จะป้อนรหัสสินค้า, จำนวนที่รับเข้า และเลขที่ใบสั่งซื้อที่อ้างอิงถึง โปรแกรมตรวจสอบความถูกต้องกับเพิ่มข้อมูลสินค้า, เพิ่มข้อมูลใบสั่งซื้อ ถ้าไม่ถูกต้องจะกลับไปขั้นตอนที่ให้ป้อนรหัสสินค้าเพื่อแก้ไข ถ้าถูกต้องจะผ่านกระบวนการเพิ่มรายการข้อมูลในเพิ่มข้อมูลทรานแซกชันประจำวัน ปรับปรุงจำนวนสินค้าในเพิ่มข้อมูลสินค้า และปรับปรุงจำนวนรับเข้าในเลขที่ใบสั่งซื้อที่อ้างอิงถึงในเพิ่มข้อมูลใบสั่งซื้อ พร้อมกับพิมพ์แบบฟอร์มเอกสารใบรับเข้าสินค้าทางเครื่องพิมพ์ แล้วจบการทำงาน ดังขั้นตอนที่แสดงให้เห็นในรูปที่ 3.3 สำหรับตารางจัดเก็บข้อมูลที่เกี่ยวข้องกับระบบมีดังนี้

ตารางที่ 3.1 ตารางข้อมูลของ Customer

ลำดับที่	ชื่อ Field	คำอธิบาย	ชนิดของข้อมูล	Primary key	Not Null
1	CUSNO	รหัส Customer	char(7)	Yes	Yes
2	CUSNAM	ชื่อ	char(70)		Yes
3	CUSADR1	ที่อยู่ 1	char(70)		Yes
4	CUSADR2	ที่อยู่ 2	char(70)		Yes
5	CUSPVC	จังหวัด	char(15)		Yes
6	CUSZIP	รหัสไปรษณีย์	char(5)		Yes
7	CUSTEL	โทรศัพท์	char(20)		No
8	CUSTYP	เกรดลูกค้า (A, B, C, D)	char(1)		Yes
9	CUSLIM	วงเงินเครดิต	numeric(11,0)		Yes
10	CUSTRM	รหัสเทอมเครดิต (1 = 30 วัน, 2 = 60 วัน, 3 = 90 วัน)	numeric(1,0)		Yes
11	CUSDIS	ส่วนลดการขาย	numeric(5,2)		Yes

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับใช้ภายในระบบเท่านั้น ไม่สามารถนำออกเผยแพร่โดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับที่	ชื่อ Field	คำอธิบาย	ชนิดของข้อมูล	Primary key	Yes
12	CUSABL	ลูกหนึ่งคงเหลือ	numeric(11,2)		Yes
13	CUSPDC	จำนวนเงินเช็คล่วงหน้า	numeric(11,2)		Yes
14	CUSSLG	รหัสเขตการขาย (10 = ภาคเหนือ, 20 = ภาคอีสาน, 30 = ภาคกลาง, 40 = ภาคใต้, 50 = กรุงเทพฯ)	char(2)		Yes
15	CUSSLM	รหัสพนักงานขาย	char(3)		Yes
16	CUSCFG	รหัสสถานะของ Customer (1 = เปิด, 2 = ปิด)	char(1)		Yes
17	CUSFDT	วันที่ติดต่อกครั้งแรก	date		Yes
18	CUSCDT	วันที่ปิดการติดต่อ	date		No
19	CUSUID	รหัสผู้ใช้	char(8)		Yes
20	CUSUPD	วันที่เปลี่ยนแปลงหลังสุด	date		Yes

ตารางที่ 3.2 ตารางข้อมูลสินค้า

ลำดับที่	ชื่อ Field	คำอธิบาย	ชนิดของข้อมูล	Primary key	Not Null
1	PRDNO	รหัสสินค้า	char(20)	Yes	Yes
2	PRDCAT	รหัสหมวดสินค้า	char(2)		Yes
3	PRDSUB	รหัสหมวดย่อยสินค้า	char(2)		Yes
4	PRDDES	ชื่อสินค้า	char(25)		Yes
5	PRDTYP	ประเภทสินค้า (I = ต่างประเทศ, L = ในประเทศ)	char(2)		Yes
6	PRDSUP	รหัส Vendor	char(10)		Yes
7	PRDLPO	เลขที่เอกสาร P/O หลังสุด	char(10)		No
8	PRDLPD	วันที่เอกสาร P/O หลังสุด	date		No

เอกสารนี้เป็นเอกสารของบริษัทฯ สำหรับใช้ภายในเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ภายนอก

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับที่	ชื่อ Field	คำอธิบาย	ชนิดของข้อมูล	Primary key	Not Null
9	PRDUTS	ต้นทุนเฉลี่ยต่อหน่วย	numeric(11,3)		Yes
10	PRDSTDP	ราคาขายมาตรฐาน	numeric(11,3)		Yes
11	PRDQOH	ปริมาณสินค้าคงคลัง	numeric(9,0)		Yes
12	PRDQOR	ปริมาณสินค้าที่สั่งซื้อ	numeric(9,0)		Yes
13	PRDVOH	มูลค่ารวมทั้งหมด	numeric(11,3)		Yes
14	PRDMIQ	ปริมาณสินค้ารับเข้าระหว่างเดือน	numeric(9,0)		Yes
15	PRDMOQ	ปริมาณสินค้าจ่ายออกระหว่างเดือน	numeric(9,0)		Yes
16	PRDMIA	มูลค่าสินค้ารับเข้าระหว่างเดือน	numeric(11,3)		Yes
17	PRDMOA	มูลค่าสินค้าจ่ายออกระหว่างเดือน	numeric(11,3)		Yes
18	PRDLSO	วันที่ขายสินค้าครั้งสุดท้าย	date		No
19	PRDUID	รหัสผู้ใช้	char(8)		Yes
20	PRDUPD	วันที่เปลี่ยนแปลงครั้งสุดท้าย	date		Yes

ตารางที่ 3.3 ตารางข้อมูล Transaction ประจำวัน

ลำดับที่	ชื่อ Field	คำอธิบาย	ชนิดของข้อมูล	Primary key	Not Null
1	DITSEQ	ลำดับที่	numeric(2,0)	Yes	Yes
2	DITTYP	รหัสประเภทของเอกสาร (12 = ใบส่งขายสินค้า, 54 = ใบรับเข้าสินค้า)	char(2,0)	Yes	Yes
3	DITDNO	เลขที่เอกสาร	char(6)	Yes	Yes
4	DITDTE	วันที่เอกสาร	date		
5	DITPRO	รหัสสินค้า	char(20)		
6	DITCNO	เลขที่เอกสารลดหนี้	char(12)		

เอกสารนี้เป็นเอกสารตัวอย่างเท่านั้น ไม่สามารถนำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับที่	ชื่อ Field	คำอธิบาย	ชนิดของข้อมูล	Primary key	Not Null
7	DITREF	เลขที่เอกสารอ้างอิง	char(12)		
8	DITORD	เลขที่ใบสั่งขายอ้างอิง	char(12)		
9	DITPON	เลขที่เอกสาร P/O อ้างอิง	char(10)		
10	DITTRM	รหัสเทอมเครดิต (1 = 30 วัน, 2 = 60 วัน, 3 = 90 วัน)	numeric(2,0)		
11	DITDUE	วันที่ครบกำหนดชำระเงิน	date		
12	DITDOQ	จำนวนหน่วยในรายการ	numeric(9,0)		
13	DITDOA	มูลค่าสินค้าต่อรายการ	numeric(11,2)		
14	DITUPC	ราคาขายต่อหน่วย	numeric(11,2)		
15	DITCST	ต้นทุนเฉลี่ยต่อหน่วย	numeric(11,3)		
16	DITTOQ	จำนวนหน่วยรวมของเอกสาร	numeric(9,0)		
17	DITTOA	มูลค่าสินค้ารวมของเอกสาร หลังหักส่วนลด	numeric(11,2)		
18	DITDIS	ส่วนลดการขาย	numeric(5,2)		
19	DITNMT	จำนวนเงินรวมของเอกสาร	numeric(9,0)		
20	DITTXA	จำนวนภาษีมูลค่าเพิ่ม	numeric(9,0)		
21	DITTXR	อัตรภาษีมูลค่าเพิ่ม	numeric(5,2)		
22	DITSUP	รหัส Vendor/Customer	char(10)		
23	DITDAC	รหัสเขตการขาย (10 = ภาคเหนือ, 20 = ภาคอีสาน, 30 = ภาคกลาง, 40 = ภาคใต้, 50 = กรุงเทพฯ)	char(2)		
24	DITCAC	รหัสพนักงานขาย	char(2)		
25	DITCAT	รหัสหมวดสินค้า	char(2)		

เอกสารนี้เป็นเอกสารของบริษัทฯ ใช้สำหรับศึกษาเท่านั้น ไม่ควรนำข้อมูลไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับที่	ชื่อ Field	คำอธิบาย	ชนิดของข้อมูล	Primary key	Not Null
26	DITSUB	รหัสหมวดย่อยสินค้า	char(2)		
27	DITUID	รหัสผู้ใช้	char(8)		
28	DITUPD	วันที่เปลี่ยนแปลงล่าสุด	date		

ตารางที่ 3.4 ตารางข้อมูลประวัติการขาย (Header)

ลำดับที่	ชื่อ Field	คำอธิบาย	ชนิดของข้อมูล	Primary key	Not Null
1	HOHCUS	รหัส Customer	char(7)	Yes	Yes
2	HOHNO	เลขที่เอกสารใบสั่งขาย	char(10)	Yes	Yes
3	HONINV	เลขที่เอกสารอินวอยซ์	char(12)		No
4	HOHDTE	วันที่เอกสารใบสั่งขาย	date		Yes
5	HOHIND	วันที่เอกสารอินวอยซ์	date		No
6	HOHTYP	รหัสของเอกสาร	char(2)		Yes
7	HOHDCT	รหัสประเภทของการขาย (D = ขายมีส่วนลด, N = ขายราคา Net, T = ขายราคาตามจำนวน)	char(1)		Yes
8	HOHTRM	รหัสเทอมเครดิต (1 = 30 วัน, 2 = 60 วัน, 3 = 90 วัน)	numeric(2,0)		Yes
9	HOHDUE	วันที่ครบกำหนด	date		Yes
10	HOHFLG	รหัสประเภทสินค้า	char(1)		Yes
11	HOHDIS	ส่วนลดการขาย	numeric(5,2)		Yes
12	HOHNET	จำนวนเงินรวมของเอกสาร	numeric(11,2)		Yes
13	HOHTOT	มูลค่าสินค้านรวมของเอกสาร หลังหักส่วนลด	numeric(11,2)		Yes
14	HOHTXA	จำนวนภาษีมูลค่าเพิ่ม	numeric(11,2)		Yes

เอกสารนี้เป็นเอกสารสำหรับใช้ภายในเท่านั้น ไม่ให้ประโชยณที่เป็นการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับที่	ชื่อ Field	คำอธิบาย	ชนิดของข้อมูล	Primary key	Not Null
15	HOHTXR	อัตราภาษีมูลค่าเพิ่ม	numeric(5,2)		Yes
16	HOHSTS	รหัสสถานะของเอกสาร (0 = ใบสั่งขาย, 5 = อินวอยซ์, 3 = ยกเลิก)	char(1)		Yes
17	HOHSLG	รหัสเขตการขาย (10 = ภาคเหนือ, 20 = ภาคอีสาน, 30 = ภาคกลาง, 40 = ภาคใต้, 50 = กรุงเทพฯ)	char(2)		Yes
18	HOHSLM	รหัสพนักงานขาย	char(3)		Yes
19	HOHUID	รหัสผู้ใช้	char(8)		Yes
20	HOHUPD	วันที่เปลี่ยนแปลงหลังสุด	date		Yes

ตารางที่ 3.5 ตารางข้อมูลประวัติการขาย (Detail)

ลำดับที่	ชื่อ Field	คำอธิบาย	ชนิดของข้อมูล	Primary key	Not Null
1	HODSEQ	ลำดับที่ของรายการ	numeric(2,0)	Yes	Yes
2	HODNO	เลขที่เอกสารใบสั่งขาย	char(10)	Yes	Yes
3	HODINV	เลขที่เอกสารอินวอยซ์	char(12)		No
4	HODPRD	รหัสสินค้า	char(20)		Yes
5	HODQOR	จำนวนหน่วยที่สั่งซื้อ	numeric(9,0)		Yes
6	HODQSP	จำนวนหน่วยที่จัดส่ง	numeric(9,0)		Yes
7	HODUPC	ราคาขายต่อหน่วย	numeric(11,2)		Yes
8	HODCST	ต้นทุนเฉลี่ยต่อหน่วย	numeric(11,3)		Yes
9	HODSMT	มูลค่าสินค้าต่อรายการ	numeric(11,2)		Yes
10	HODCAT	รหัสหมวดสินค้า	char(2)		Yes

เอกสารนี้เป็นเอกสารต้นฉบับสำหรับใช้งานในระบบคอมพิวเตอร์เท่านั้น ไม่สามารถนำออกจากระบบได้

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับที่	ชื่อ Field	คำอธิบาย	ชนิดของข้อมูล	Primary key	Not Null
11	HODSUB	รหัสหมวดย่อยสินค้า	char(2)		Yes
12	HODUID	รหัสผู้ใช้	char(8)		Yes
13	HODUPD	วันที่เปลี่ยนแปลงหลังสุด	date		Yes

ตารางที่ 3.6 ตารางข้อมูลใบสั่งซื้อ (Header)

ลำดับที่	ชื่อ Field	คำอธิบาย	ชนิดของข้อมูล	Primary key	Not Null
1	NUMPO	เลขที่เอกสารใบสั่งซื้อ	char(12)	Yes	Yes
2	DTEPO	วันที่เอกสารใบสั่งซื้อ	date		Yes
3	CODSUP	รหัส Vendor	char(6)		Yes
4	DTESHP	วันที่ส่งสินค้า	date		Yes
5	RMKPO	หมายเหตุ	char(20)		No
6	DTERCV	วันที่รับสินค้า	date		Yes
7	AMTPO	จำนวนเงินรวมของเอกสาร	numeric(11,2)		Yes
8	QTYTOTPO	จำนวนหน่วยสินค้าที่สั่งซื้อ	numeric(9,0)		Yes
9	QTYTOTRC	จำนวนหน่วยสินค้าที่รับเข้า	numeric(9,0)		No
10	FLGPO	รหัสสถานะของเอกสาร (0 = ยังไม่รับเข้า, 1 = รับเข้าเป็นบางส่วน, 2 = รับเข้าครบจำนวน, 3 = ยกเลิก)	char(1)		Yes
11	USRID	รหัสผู้ใช้	char(8)		Yes
12	DTEUPD	วันที่เปลี่ยนแปลงหลังสุด	date		Yes

ตารางที่ 3.7 ตารางข้อมูลใบสั่งซื้อ (Detail)

ลำดับที่	ชื่อ Field	คำอธิบาย	ชนิดของข้อมูล	Primary key	Not Null
1	NUMPO	เลขที่เอกสารใบสั่งซื้อ	char(12)	Yes	Yes

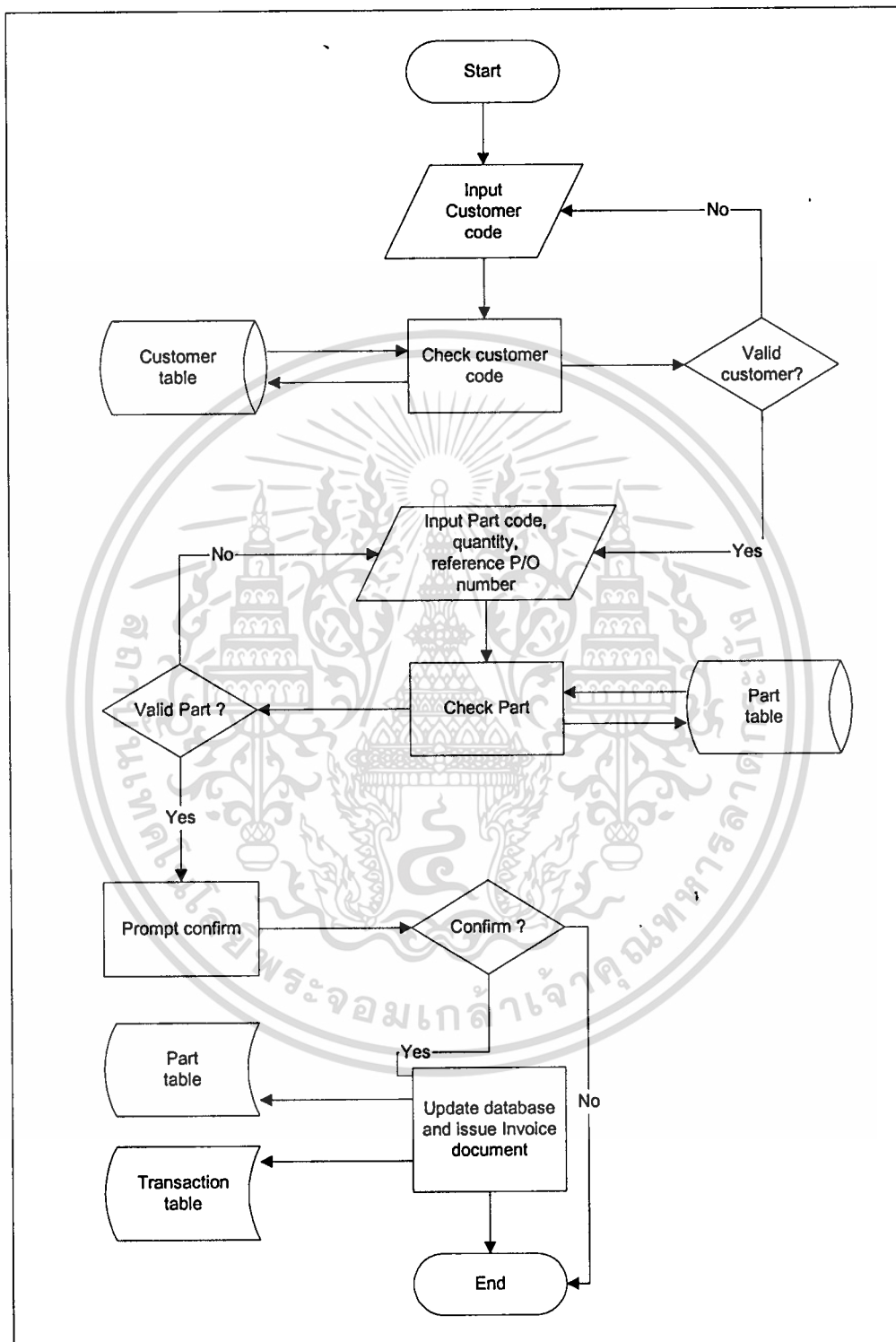
เอกสารนี้เป็นเอกสารตัวอย่างเท่านั้น ไม่ควรกรณิใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับที่	ชื่อ Field	คำอธิบาย	ชนิดของข้อมูล	Primary key	Not Null
2	PRDNO	รหัสสินค้า	char(20)		Yes
3	PRDCAT	รหัสหมวดสินค้า	char(2)		Yes
4	PRDSUB	รหัสหมวดย่อยสินค้า	char(2)		Yes
5	QTYPO	จำนวนหน่วยสั่งซื้อ	numeric(9,0)		Yes
6	PRDUTS	ราคาซื้อต่อหน่วย	numeric(11,3)		Yes
7	QTYRCV	จำนวนหน่วยรับเข้า	numeric(9,0)		No
8	DTERCV	วันที่รับเข้า	date		No
9	FLGPOD	สถานะของสินค้า (0 = ยังไม่รับเข้า, 1 = รับเข้าเป็นบางส่วน, 2 = รับเข้าครบจำนวน, 3 = ยกเลิก)	char(1)		Yes

3.1.4 ระบบขายสินค้าในส่วนของโปรแกรมบันทึกการขายการสั่งซื้อโดย ตัวแทนจำหน่ายของโรงงานผลิต

สำหรับระบบขายสินค้าของโรงงานผลิต หลังจากได้รับเอกสารใบสั่งซื้อจากตัวแทนจำหน่ายที่เป็นคำสั่งซื้อล่วงหน้า 1 เดือน จะนำเอาตัวเลขเหล่านั้นมาวางแผนปริมาณการผลิต และจัดส่งให้ครบภายใน 1 เดือนถัดไป ในการส่งสินค้าแต่ละครั้งฝ่ายขนส่งจะแนบใบอินวอยซ์ไปด้วยเพื่อให้ตัวแทนจำหน่าย ทราบว่าสินค้าที่ส่งนั้นอ้างอิงถึงเลขที่เอกสารใบสั่งซื้อใด เมื่อได้รับรายงานสินค้าคงคลังประจำวันที่มีรายละเอียดของสินค้าที่ผลิตได้ในแต่ละวันแล้ว ฝ่ายขายจะทำหน้าที่ในการป้อนข้อมูลในโปรแกรมบันทึกการขายสินค้าเพื่อพิมพ์เอกสารอินวอยซ์นี้ โปรแกรมจะตรวจสอบรหัสของ ตัวแทนจำหน่าย ก่อนเพื่อเช็คความถูกต้องและดึงชื่อ ที่อยู่มาแสดงบนหน้าจอ หลังจากนั้นจะให้ผู้ใช้ป้อนรหัสสินค้า, จำนวนขาย และเลขที่ใบสั่งซื้อที่อ้างอิงถึง โดยจะตรวจสอบกับแฟ้มข้อมูลสินค้าหลักในเรื่องความถูกต้องของรหัสสินค้าและจำนวน ถ้ามีข้อผิดพลาดโปรแกรมจะให้ผู้ใช้แก้ไขให้ถูกต้อง ถ้าถูกต้องแล้วจะให้ผู้ใช้ยืนยันการเพิ่มรายการข้อมูล ถ้าผู้ใช้ยกเลิกโปรแกรมจะจบการทำงาน ถ้าผู้ใช้ตอบตกลงโปรแกรมจะเพิ่มรายการข้อมูลในแฟ้มข้อมูลทรานแซคชัน และปรับปรุงจำนวนสินค้าคงเหลือในแฟ้มข้อมูลสินค้า พร้อมกับพิมพ์เอกสารอินวอยซ์ทางเครื่องพิมพ์ก่อนจบการทำงาน ดังขั้นตอนที่แสดงไว้ในรูปที่ 3.4

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ การเผยแพร่โดยไม่ได้รับความเห็นชอบจากมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 Flowchart โปรแกรมบันทึกการขายสินค้าของโรงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.8 ตารางข้อมูลสินค้าหลัก

ลำดับที่	ชื่อ Field	คำอธิบาย	ชนิดของข้อมูล	Primary key	Not Null
1	PRODUCTNO	รหัสสินค้า	char(20)	Yes	Yes
2	PRODUCTCAT	รหัสหมวดสินค้า	char(2)		Yes
3	PRODUCTSUB	รหัสหมวดย่อยสินค้า	char(2)		Yes
4	PRODUCTDES	ชื่อสินค้า	char(25)		Yes
5	PRODUCTPRICE	ราคาขายต่อหน่วย	numeric(9,0)		Yes
6	PRODUCTONHAND	ปริมาณสินค้าคงคลัง	numeric(9,0)		Yes
7	PRODUCTONORDER	ปริมาณสินค้าที่สั่งซื้อเข้ามา	numeric(9,0)		Yes
8	PRODUCTQBORDER	ปริมาณสินค้าค้างส่ง	numeric(9,0)		Yes
9	PRODUCTLSALES	วันที่ขายสินค้าหลังสุด	datetime		No
10	USERNAME	รหัสผู้ใช้	char(8)		Yes
11	LASTUPDATE	วันที่เปลี่ยนแปลงหลังสุด	datetime		Yes

ตารางที่ 3.9 ตารางข้อมูลการขายสินค้า (Header)

ลำดับที่	ชื่อ Field	คำอธิบาย	ชนิดของข้อมูล	Primary key	Not Null
1	ORDERNO	เลขที่ใบสั่งซื้อ	char(6)	Yes	Yes
2	ORDERDTE	วันที่ใบสั่งซื้อ	datetime		Yes
3	ORDERSTS	สถานะใบสั่งซื้อ U = ยังไม่ได้จัดส่งสินค้า P = จัดส่งสินค้าแล้ว	char(1)		Yes
4	CUSTNO	รหัสลูกค้า	char(7)		Yes
5	CUSTNAM	ชื่อลูกค้า	char(70)		Yes
6	CUSTADR1	ที่อยู่ลูกค้า	char(70)		Yes
7	CUSTADR2	ที่อยู่ลูกค้า	char(70)		Yes

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ห้ามเผยแพร่โดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับที่	ชื่อ Field	คำอธิบาย	ชนิดของข้อมูล	Primary key	Not Null
8	ORDERSHPD	วันที่จัดส่งสินค้า	datetime		Yes
9	USERNAME	รหัสผู้ใช้	char(8)		No
10	LASTUPDATE	วันที่ปรับปรุงครั้งล่าสุด	datetime		Yes

ตารางที่ 3.10 ตารางข้อมูลการขายสินค้า (Detail)

ลำดับที่	ชื่อ Field	คำอธิบาย	ชนิดของข้อมูล	Primary key	Not Null
1	ORDERNO	เลขที่ใบสั่งซื้อ	char(6)	Yes	Yes
2	ORDERITEM	ลำดับที่	numeric(2,0)	Yes	Yes
3	PRODUCTNO	รหัสสินค้า	char(20)		Yes
4	ORDERQTY	จำนวนสั่งซื้อ	numeric(5,0)		Yes
5	UNITPRICE	ราคาต่อหน่วย	numeric(7,2)		Yes

3.2 ปัญหาของระบบปัจจุบัน

จากขั้นตอนการทำงานในปัจจุบันทั้งทางตัวแทนจำหน่ายและ โรงงานผลิต พบว่ามีปัญหาอยู่หลายประการดังนี้

1. ในแต่ละเดือนจำนวนสั่งซื้อและจำนวนรับเข้าสินค้าตามเอกสารของตัวแทนจำหน่าย ส่วนใหญ่จะไม่เท่ากัน เนื่องจากความผิดพลาดจากการป้อนข้อมูลด้วยโปรแกรมของทางโรงงานผลิต ทำให้พนักงานของทั้ง 2 บริษัทต้องตรวจสอบเอกสารเพื่อหาจุดผิดพลาดเพื่อปรับปรุงให้ถูกต้อง
2. ประสิทธิภาพการสั่งซื้อที่จัดทำโดยตัวแทนจำหน่ายนั้น บางครั้งจะคลาดเคลื่อนจากตัวเลขจริงเกินกว่าระดับที่ตั้งไว้ ในกรณีที่ประสิทธิภาพน้อยเกินไปทำให้ตัวแทนจำหน่ายมีจำนวนสินค้าไม่พอจำหน่าย และทำให้โรงงานผลิตต้องเร่งการผลิตสินค้าในเดือนต่อไป จำนวนมากกว่าปกติ ส่วนในกรณีที่ประสิทธิภาพมากเกินไป ทำให้มีปริมาณสินค้าคงคลังในคลังสินค้าตัวแทนจำหน่ายมากเกินไป นั้นหมายถึงทำให้ต้นทุนที่จมอยู่นั้นเอง
3. การวางแผนการผลิตของโรงงานผลิตแต่ละวันนั้น สินค้าและจำนวนที่ผลิตบางครั้งจะไม่ตรงตามความต้องการของตัวแทนจำหน่าย เนื่องจากโรงงานจะไม่ทราบถึงช่วงเวลาในการสั่งซื้อสินค้าแต่ละรหัสสินค้าของลูกค้าว่าเป็นอย่างไร สินค้าบางกลุ่มลูกค้าต้องการสั่งซื้อ

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับใช้ภายในองค์กรเท่านั้น ไม่อนุญาตให้เผยแพร่หรือใช้ประโยชน์อื่นใด
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในช่วงต้นเดือนหรือกลางเดือน เป็นต้น ดังนั้นทำให้เกิดเหตุการณ์ที่ตัวแทนจำหน่ายไม่มีสินค้าจำหน่าย

4. ช่องทางการสั่งซื้อสินค้าโดยลูกค้าในปัจุบันจะผ่านทางพนักงานขายที่ดูแลลูกค้าต่างๆ ซึ่งตัวแทนจำหน่ายต้องจ่ายค่าเบี้ยเลี้ยงในการเดินทางเยี่ยมชมลูกค้าและค่าเงินรางวัลในการขายสินค้าในแต่ละเดือนเป็นจำนวนเงินที่สูงมาก' ซึ่งค่าใช้จ่ายเหล่านี้ทำให้สินค้ามีต้นทุนที่แพงมากขึ้น ทำให้เสียเปรียบคู่แข่งซึ่งปัจจุบันนำเทคโนโลยีทางอินเทอร์เน็ตเข้ามาประยุกต์ใช้และลดค่าใช้จ่ายที่กล่าวข้างต้นลง ทำให้ต้นทุนสินค้าของคู่แข่งถูกกว่า
5. ในปัจจุบันตัวแทนจำหน่ายต้องรับภาระเรื่องค่าเช่าและการดูแลคลังสินค้าในแต่ละเดือนค่อนข้างสูง รวมทั้งค่าขนส่งที่บริษัทต้องนำส่งสินค้าให้กับลูกค้าทั่วประเทศด้วย



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การวิเคราะห์และออกแบบเพื่อแก้ปัญหาของระบบเดิม

4.1 ศึกษาความเป็นไปได้และความเหมาะสมในการพัฒนาระบบ

จากปัญหาที่เกิดขึ้นจากระบบในปัจจุบัน จะพบว่าปัญหาหลักของระบบก็คือ การที่ระบบคอมพิวเตอร์ของตัวแทนจำหน่าย และระบบคอมพิวเตอร์ของโรงงานผลิต ไม่ได้เชื่อมโยงข้อมูลซึ่งกันและกันนั่นเอง ดังนั้นถ้าพิจารณาถึงปัจจัยต่างๆแล้วจะพบว่ามีความเหมาะสมที่จะพัฒนาระบบใหม่ขึ้นมาเพื่อปรับปรุงประสิทธิภาพการทำงานให้ดียิ่งขึ้น ดังนี้

4.1.1 เทคโนโลยีอินเทอร์เน็ต

เทคโนโลยีอินเทอร์เน็ตในปัจจุบันผู้ใช้งานสามารถใช้งานได้ทั่วประเทศไทยและมีราคาถูกลงวิธีการในการใช้งานทำได้ง่ายและสะดวกขึ้น ทำให้ประชาชนให้ความสนใจกับเทคโนโลยีนี้มากขึ้นเรื่อยๆ ดังนั้นการนำเอาอินเทอร์เน็ตเข้ามาประยุกต์ใช้กับระบบสั่งซื้อสินค้าของตัวแทนจำหน่าย น่าจะเป็นอีกทางเลือกหนึ่งที่เป็นการปรับตัวเข้ากับเทคโนโลยีในปัจจุบันและอนาคต

4.1.2 อุปกรณ์คอมพิวเตอร์

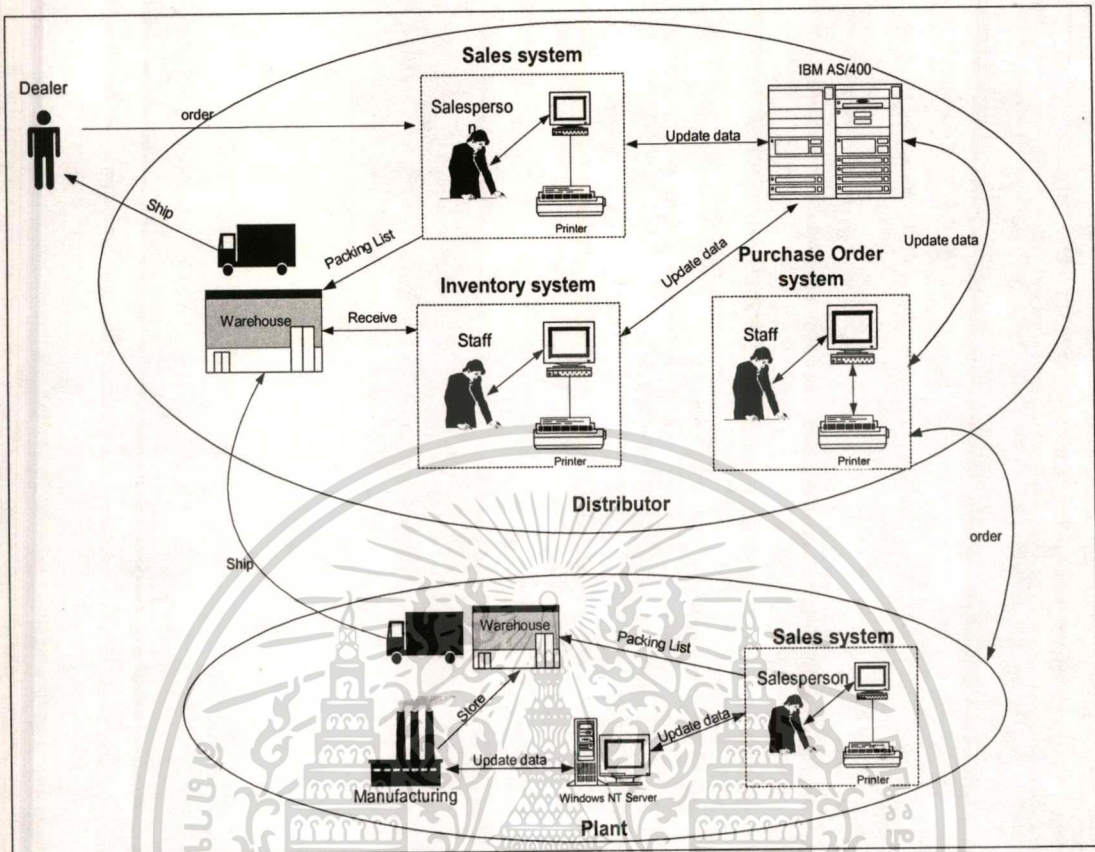
ในขณะนี้ราคาของเครื่องคอมพิวเตอร์พร้อมระบบปฏิบัติการที่สามารถใช้อินเทอร์เน็ตได้มีราคาค่อนข้างต่ำ ดังนั้นลูกค้าสามารถรองรับกับการลงทุนในส่วนนี้ได้

4.1.3 ต้นทุนในการพัฒนาระบบใหม่

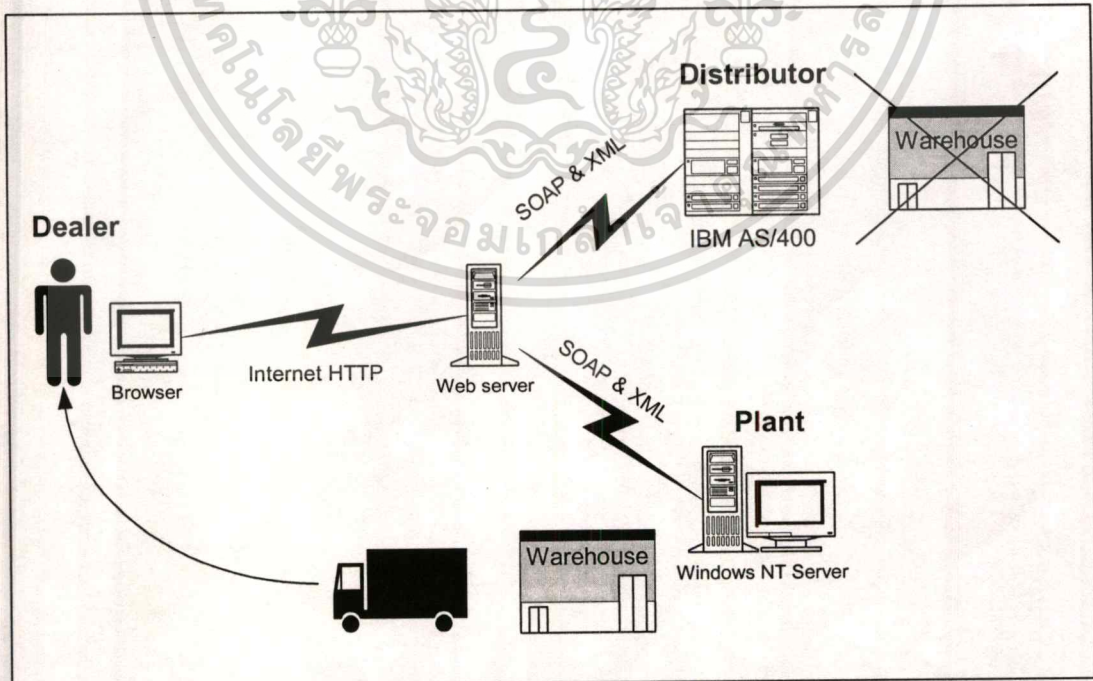
แนวคิดที่จะพัฒนาระบบใหม่เพื่อแก้ปัญหาที่เกิดขึ้นด้วยเว็บเซิร์ฟเวอร์ที่พัฒนาด้วยภาษาจาวานั้น มีการใช้ซอฟต์แวร์ที่สามารถนำมาใช้ได้โดยไม่ต้องเสียค่าใช้จ่ายและเป็นซอฟต์แวร์ที่สามารถนำไปใช้บนเครื่องคอมพิวเตอร์ที่มีระบบปฏิบัติการที่แตกต่างกันได้ ดังนั้นจึงเหมาะสมกับการนำมาใช้ในสภาพแวดล้อมที่ทั้งระบบคอมพิวเตอร์ของตัวแทนจำหน่ายและโรงงานผลิตแตกต่างกันทั้งในด้านของฮาร์ดแวร์และซอฟต์แวร์ โดยที่ไม่ต้องลงทุนเพิ่มเติมในส่วนนี้เพียงแต่พัฒนาโปรแกรมเพิ่มเติมขึ้นมาบางส่วนเชื่อมต่อกับระบบเดิมเท่านั้น

4.2 การวิเคราะห์และออกแบบระบบใหม่

จากปัญหาที่พบในระบบเดิม ทำให้มีแนวคิดในการเชื่อมต่อข้อมูลระหว่างระบบคอมพิวเตอร์ของตัวแทนจำหน่าย และระบบคอมพิวเตอร์ของโรงงานผลิตในส่วนที่เกี่ยวข้องกับระบบขายสินค้า, ระบบสั่งซื้อสินค้าและระบบคลังสินค้าเข้าด้วยกัน พิจารณาได้จากรูปที่ 4.1 และรูปที่ 4.2



รูปที่ 4.1 ความสัมพันธ์เดิมของระบบต่างๆระหว่างตัวแทนจำหน่าย และโรงงานผลิต

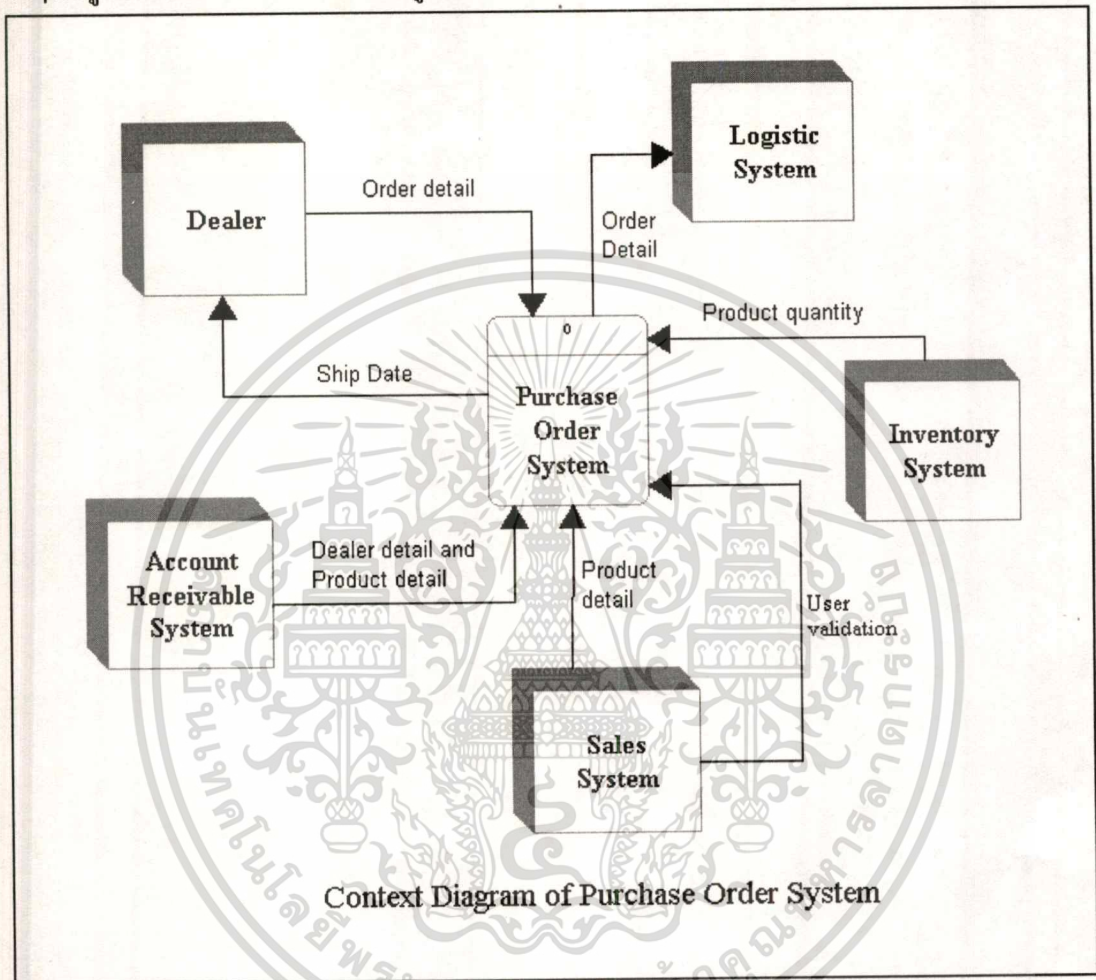


รูปที่ 4.2 ความสัมพันธ์ใหม่ของระบบต่างๆระหว่างตัวแทนจำหน่าย และโรงงานผลิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1 Context Diagram

จากรูปที่ 4.2 สามารถวิเคราะห์ห่อออกมาในรูปของไดอะแกรมที่แสดงความสัมพันธ์ระหว่างสิ่งต่างๆที่อยู่ในระบบและนอกระบบ ดังรูปที่ 4.3



รูปที่ 4.3 Context Diagram ของระบบใหม่

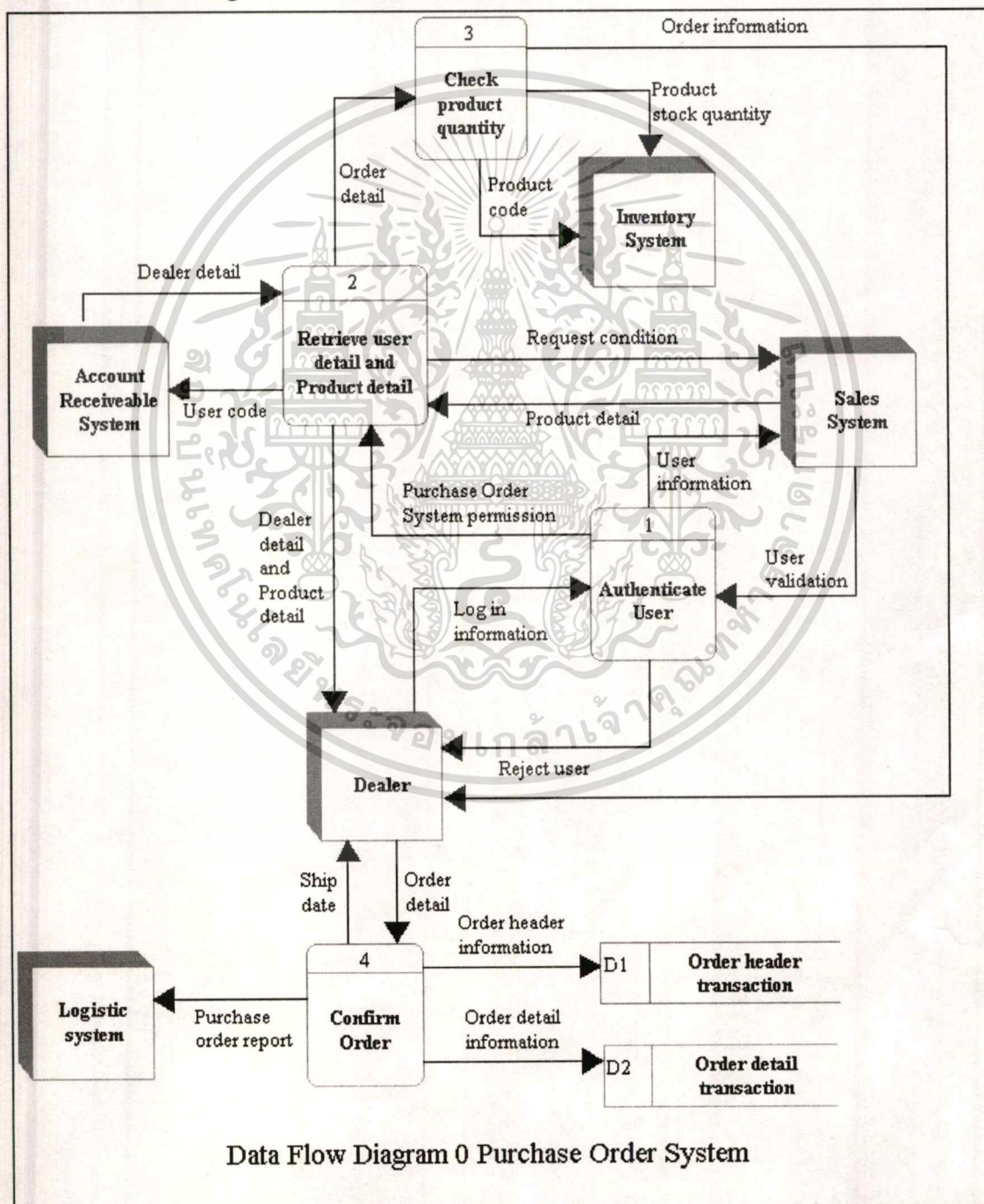
จากรูปแสดงให้เห็นถึงการติดต่อระหว่างระบบสั่งซื้อสินค้า (ต่อไปเรียกว่า “ระบบ”) กับเอนทิตีอื่นๆ ดังนี้

- ลูกค้า (Dealer) เป็นผู้ใช้งานในระบบนี้เพื่อสั่งซื้อสินค้า และระบบจะส่งวันที่ส่งสินค้ากลับไปให้
- ระบบการขาย (Sales system) เป็นระบบที่ส่งข้อมูลผู้ที่มีสิทธิ์ใช้ระบบรวมถึงรายละเอียดของสินค้าเช่น รหัสสินค้า, กลุ่มของสินค้า, ราคาของสินค้า เป็นต้น กลับไปให้ระบบ
- ระบบสินค้าคงคลัง (Inventory System) ส่งปริมาณสินค้าคงคลังของแต่ละสินค้าที่มีในขณะนั้นไปให้ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ระบบลูกหนี้ (Account Receivable System) ส่งรายละเอียดของ ลูกค้า กลับไปให้ระบบ เช่น ชื่อ-ที่อยู่, เครดิตเทอม และส่วนลดทางการค้า เป็นต้น
- ระบบขนส่งสินค้า (Logistic System) รับข้อมูลการสั่งซื้อ เช่น รหัสสินค้า, จำนวน, วันที่จัดส่ง และที่อยู่ในการจัดส่งสินค้า

4.2.2 Data Flow Diagram 0 Purchase order system

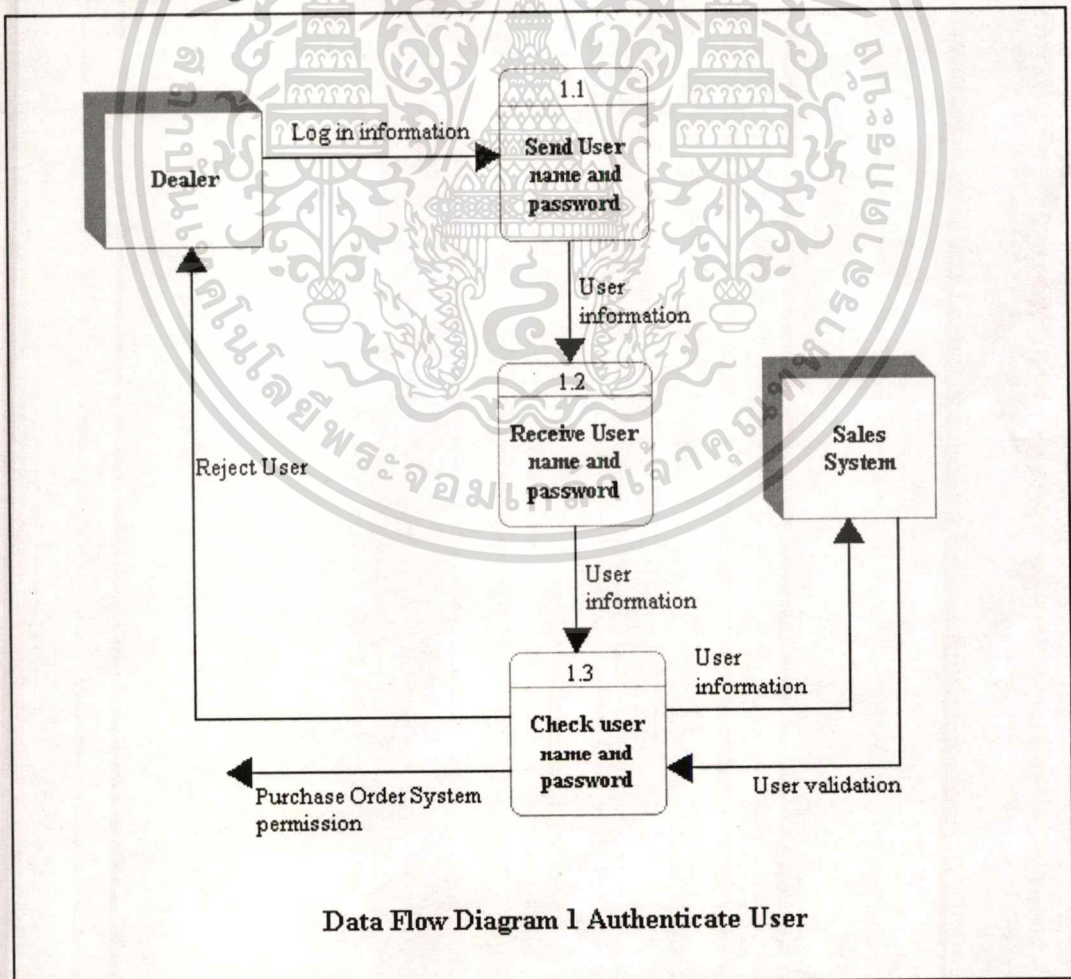


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 4.4 Data Flow Diagram 0 ของระบบใหม่ แต่ให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จาก Context Diagram สามารถกระจายออกมาเป็น Data Flow Diagram ระดับ 0 เพื่อแสดงให้เห็นการทำงานของระบบว่ามีกระบวนการหลักๆ ดังนี้

- Authenticate user เป็นกระบวนการตรวจสอบผู้ใช่ว่ามีสิทธิในการใช้ระบบหรือไม่
- Retrieve user detail and product detail หลังจากผ่านการตรวจสอบสิทธิการใช้ระบบแล้ว จะเข้าสู่กระบวนการแสดงผลบนหน้าจอถึงชื่อลูกค้า, รหัสสินค้าพร้อมด้วยราคาเพื่อรอรับการทำงานในลำดับต่อไป
- Check product quantity เป็นกระบวนการที่ตรวจสอบปริมาณสินค้าคงคลังในระบบ สินค้าคงคลังว่ามีเพียงพอต่อการสั่งซื้อหรือไม่
- Confirm Order เป็นกระบวนการยืนยันคำสั่งซื้อ เพื่อจบขั้นตอนการสั่งซื้อ

4.2.3 Data Flow Diagram 1 Authenticate user



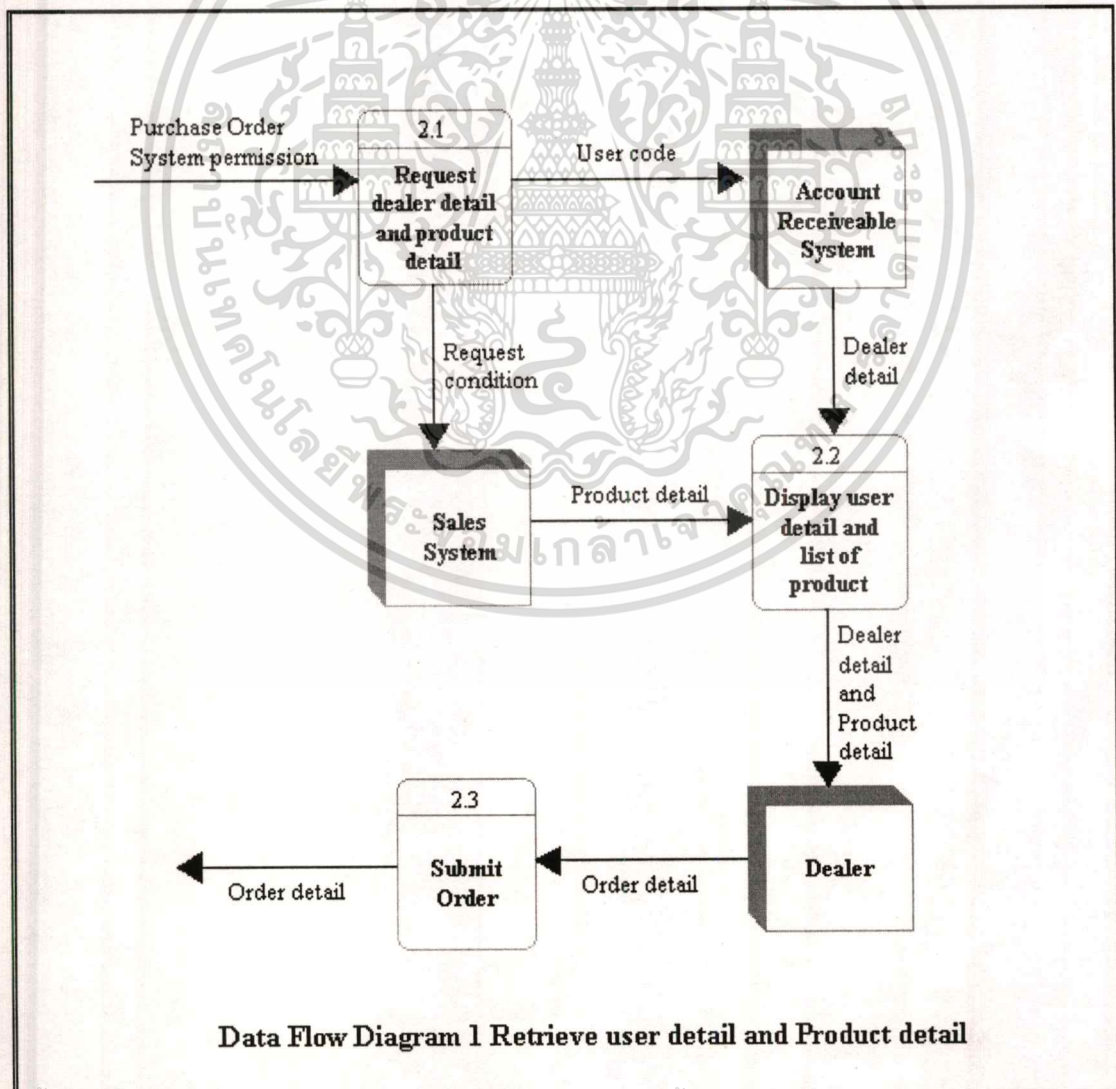
รูปที่ 4.5 Data Flow Diagram 1 Authenticate user ของระบบใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกระบวนการตรวจสอบสิทธิ์ผู้ใช้นั้นจะประกอบไปด้วยกระบวนการย่อยๆดังนี้

- Send username and password เป็นกระบวนการให้ลูกค้าใส่รหัสผู้ใช้และรหัสผ่านบนหน้าจอแสดงผลแล้วส่งเข้ามาในระบบ
- Receive username and password เป็นกระบวนการรับรหัสผู้ใช้และรหัสผ่านเข้ามาในระบบ
- Check username and password เป็นกระบวนการที่ตรวจสอบความถูกต้องของรหัสผู้ใช้และรหัสผ่าน โดยนำข้อมูลมาจากระบบขายสินค้าที่มีไฟล์ข้อมูลของรหัสผู้ใช้และรหัสผ่านอยู่ มาเปรียบเทียบกับข้อมูลที่ได้รับมาจากลูกค้าว่าถูกต้องหรือไม่ ถ้าถูกต้องจะทำงานในกระบวนการถัดไป แต่ถ้าไม่ถูกต้องจะแสดงผลให้ผู้ใช้งานทราบถึงการปฏิเสธการเข้าใช้งาน

4.2.4 Data Flow Diagram 1 Retrieve user detail and product detail

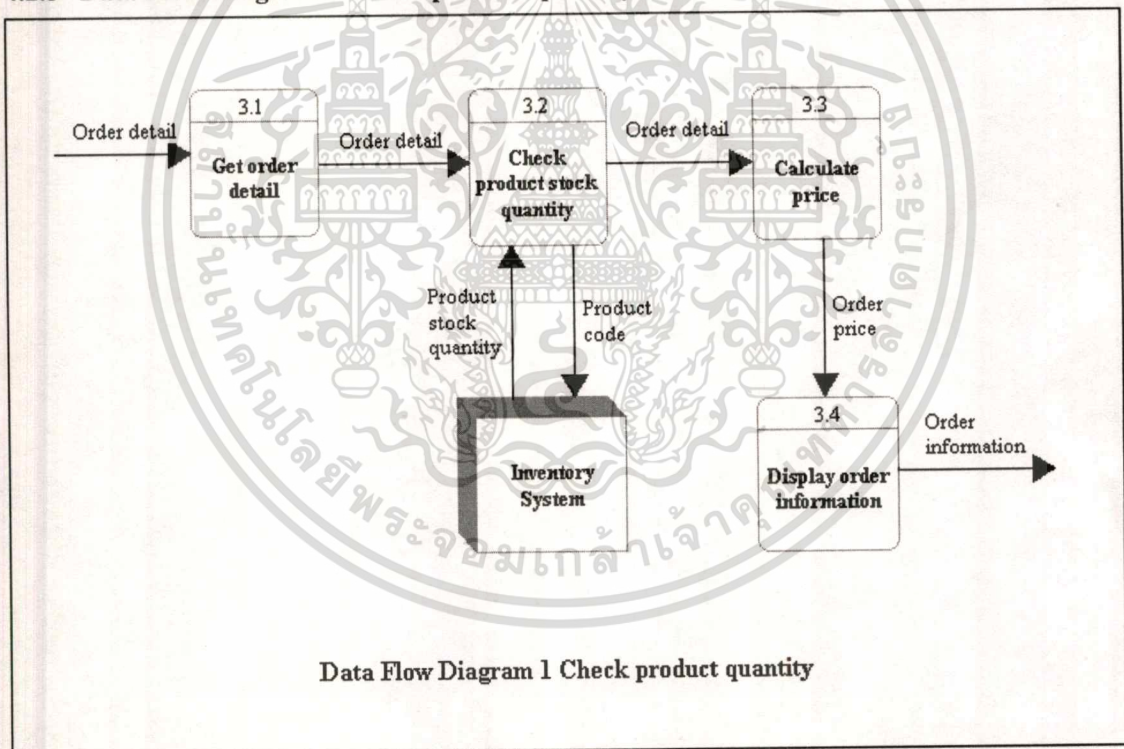


เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของสถาบันวิจัยและพัฒนาเทคโนโลยีสารสนเทศและการสื่อสาร
รูปที่ 4.6 Data Flow Diagram 1 Retrieve user detail and product detail ของระบบใหม่
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อผู้ใช้งานผ่านการตรวจสอบสิทธิ์การใช้งานแล้ว จะมาถึงกระบวนการนำเอาข้อมูลของลูกค้า นั้นๆ และรหัสสินค้าและราคา มาแสดงผลบนหน้าจอ ประกอบด้วยกระบวนการย่อยๆ ดังนี้

- Request dealer detail and product detail เป็นกระบวนการที่ขอข้อมูลของลูกค้ามาจาก ระบบลูกหนี้ และข้อมูลของสินค้ามาจากระบบขายสินค้า
- Display user detail and list of product เป็นกระบวนการที่นำเอาข้อมูลที่ได้มาจากระบบ ลูกหนี้และระบบขายสินค้ามาแสดงผลบนหน้าจอเพื่อให้ลูกค้าได้เลือกรหัสสินค้าและ จำนวนที่ต้องการสั่งซื้อ
- Submit order เป็นกระบวนการที่ให้ลูกค้ายืนยันการสั่งซื้อตามรหัสสินค้าและจำนวนที่ แสดงบนหน้าจอ

4.2.5 Data Flow Diagram 1 Check product quantity



รูปที่ 4.7 Data Flow Diagram 1 Check product quantity ของระบบใหม่

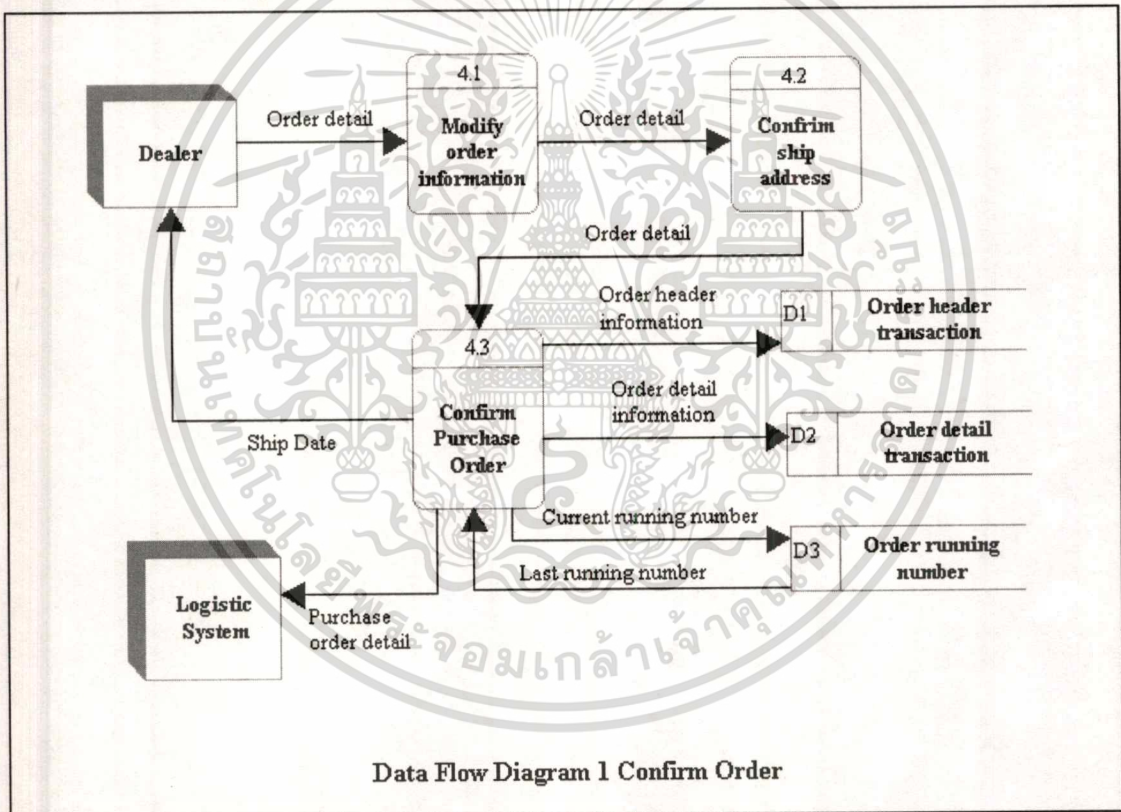
เมื่อลูกค้ายืนยันการสั่งซื้อตามรหัสสินค้าและจำนวนที่แสดงบนหน้าจอแล้ว ระบบจะนำเอา รายละเอียดของการสั่งซื้อเข้าสู่กระบวนการตรวจสอบปริมาณสินค้าคงคลังเพื่อจะได้นำเอาข้อมูล นั้นมาคำนวณวันที่จัดส่งสินค้าให้ลูกค้า โดยมีกระบวนการย่อยๆ ดังนี้

- Get order detail เป็นกระบวนการรับเอาข้อมูลที่แสดงอยู่บนหน้าจอที่ลูกค้าใส่ไว้เข้ามา

เอกสารนี้เป็นเอกสารในระบบไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Check product stock quantity เป็นกระบวนการตรวจสอบปริมาณสินค้าคงคลังในขณะนั้นจากระบบสินค้าคงคลังว่ามีเพียงพอต่อการขายหรือไม่ ถ้ามีเพียงพอจะจัดส่งให้ภายใน 1 วันทำการ แต่ถ้าไม่เพียงพอจะเพิ่มจำนวนวันเป็น 3 วันทำการ เพราะโรงงานต้องใช้เวลาในการผลิตตามคำสั่งซื้อนั้น
- Calculate price เป็นกระบวนการคำนวณมูลค่าสินค้าต่อรายการ, มูลค่าสินค้ารวม, จำนวนรวม, ภาษีมูลค่าเพิ่ม และจำนวนเงินรวมของการสั่งซื้อ
- Display order information เป็นกระบวนการที่นำเอาข้อมูลที่ได้จากการคำนวณมาแสดงผลบนหน้าจอให้ลูกค้าทราบ

4.2.6 Data Flow Diagram 1 Confirm Order



รูปที่ 4.8 Data Flow Diagram 1 Confirm order ของระบบใหม่

กระบวนการนี้เป็นการยืนยันคำสั่งซื้อครั้งสุดท้ายเพื่อระบบจะได้นำเอาข้อมูลการสั่งซื้อไปเพิ่มในไฟล์ข้อมูลการสั่งซื้อและแสดงผลให้ลูกค้าทราบถึงวันที่จัดส่งสินค้าพร้อมกับส่งข้อมูลการสั่งซื้อไปที่ระบบขนส่งสินค้า ดังกระบวนการย่อยๆดังนี้

- Modify order information เป็นกระบวนการตรวจสอบปริมาณสินค้าคงคลังของแต่ละรหัสสินค้าและคำนวณราคาต่างๆ เมื่อผู้ใช้มีการเปลี่ยนแปลงรายละเอียดของการสั่งซื้อ

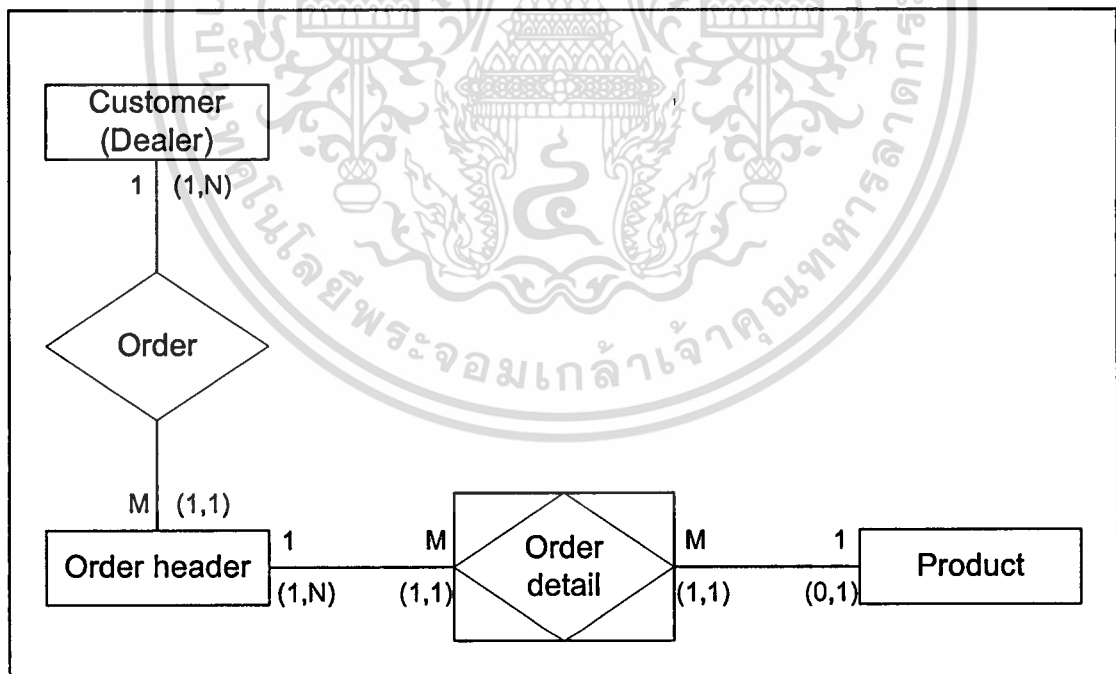
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Confirm ship address เป็นกระบวนการที่ให้ลูกค้ายืนยันสถานที่จัดส่งสินค้า เพราะลูกค้าบางรายต้องการให้ส่งสินค้าไปสถานที่อื่นเช่น ร้านค้าย่อยหรือคลังสินค้าของลูกค้าเอง
- Confirm purchase order เป็นกระบวนการรับการยืนยันครั้งสุดท้ายของการสั่งซื้อสินค้า จากลูกค้า และระบบจะนำเอาเลขที่ใบสั่งซื้อครั้งสุดท้ายจากไฟล์ running number มาคำนวณให้ได้เลขที่ใบสั่งซื้อปัจจุบัน เพื่อใช้เป็นเลขที่ใบสั่งซื้อของใบสั่งซื้อที่กำลังทำรายการอยู่ จากนั้นจะส่งเลขที่ใบสั่งซื้อนั้นกลับไปเพื่อปรับปรุงไฟล์ให้ถูกต้อง จากนั้นนำข้อมูลทั้งหมดของใบสั่งซื้อเพิ่มในไฟล์ Order header transaction และ Order detail transaction ก่อนจะแสดงข้อมูลให้ลูกค้าทราบถึงรายละเอียดในใบสั่งซื้อพร้อมวันที่ที่จัดส่งสินค้า

4.3 ER Diagram

จาก Data Flow Diagram ทำให้ทราบถึงการไหลของข้อมูลที่มีในระบบ ต่อไปจะสร้างโมเดลของความสัมพันธ์ระหว่างเอนทิตีต่างๆในระบบในลักษณะของ ER Diagram



รูปที่ 4.1 ER Diagram

จาก Diagram จะเห็นว่าความสัมพันธ์ต่างๆดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ความสัมพันธ์ระหว่าง Customer และ Order header เป็นแบบ Many to One โดยแต่ละ Customer สามารถมีรายการใบสั่งซื้อได้หลายๆรายการ หรือไม่มีก็ได้ และใบสั่งซื้อแต่ละใบเป็นของ Customer เพียงหนึ่งรายเท่านั้น
- ความสัมพันธ์ระหว่าง Order header และ Order detail เป็นแบบ Many to One โดยใบสั่งซื้อแต่ละรายการใน Order header จะมีการสั่งซื้ออย่างน้อยหนึ่งรายการ และแต่ละรายการใน Order detail จะเป็นของใบสั่งซื้อเพียงหนึ่งใบเท่านั้น
- ความสัมพันธ์ระหว่าง Order detail และ Product เป็นแบบ Many to One โดยแต่ละรายการของใบสั่งซื้อจะอ้างอิงรหัสสินค้าได้เพียงรหัสเดียวเท่านั้น และสินค้าแต่ละรหัสสามารถมีได้หลายรายการหรือไม่มีเลขก็ได้

4.4 การออกแบบฐานข้อมูล

เนื่องจากการพัฒนาระบบสั่งซื้อสินค้าใหม่เป็นเพียงระบบใหม่ที่เพิ่มเติมเข้ามาเพื่อเชื่อมต่อองค์กร 2 องค์กรในส่วนของคำสั่งซื้อจากลูกค้าถึงตัวแทนจำหน่าย และจากตัวแทนจำหน่ายถึงโรงงานผลิต โดยอาศัยฐานข้อมูลที่มีอยู่เดิมทั้งทางด้านตัวแทนจำหน่ายและโรงงานผลิต ดังนั้นในส่วนของการออกแบบฐานข้อมูลจึงถือเอาตามระบบเดิมที่มีอยู่แล้ว ซึ่งในระบบเองนั้นจะประกอบไปด้วยเพิ่มข้อมูล OrderHeader, OrderDetail และ Running_Number ดังรายละเอียดตามตารางข้างล่างนี้

ตารางที่ 4.1 ตารางข้อมูล OrderHeader

ลำดับที่	ชื่อ Field	คำอธิบาย	ชนิดของข้อมูล	Primary key	Not Null
1	ORHNO	เลขที่ใบสั่งซื้อ	Char(12)	Yes	Yes
2	ORHDTE	วันที่ใบสั่งซื้อ	Date		Yes
3	ORHCUS	รหัส ลูกค้า	Char(7)		Yes
4	ORHTRM	เครดิตเทอม	Numeric(2)		Yes
5	ORHDIS	ส่วนลดการค้า	Numeric(5,2)		Yes
6	ORHTOQ	จำนวนรวมสินค้า	Numeric(9)		Yes
7	ORHTOT	มูลค่าสินค้านรวม	Numeric(11,2)		Yes
8	ORHTXR	อัตราภาษีมูลค่าเพิ่ม	Numeric(5,2)		Yes
9	ORHTXA	จำนวนภาษีมูลค่าเพิ่ม	Numeric(11,2)		Yes
10	ORHUID	รหัสผู้ใช้	Char(8)		Yes

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อการศึกษานเท่านั้น ไม่ควรนำข้อมูลไปใช้ประโยชน์ทางธุรกิจ

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับที่	ชื่อ Field	คำอธิบาย	ชนิดของข้อมูล	Primary key	Not Null
11	ORHUPD	วันที่ปรับปรุงครั้งล่าสุด	Date		Yes
12	ORHSTS	สถานะใบสั่งซื้อ (U=ยังไม่ได้จัดส่ง, P=จัดส่งแล้ว)	Char(1)		Yes
13	ORHSHN	ชื่อสถานที่จัดส่งสินค้า	Char(70)		Yes
14	ORHSHA1	ที่อยู่จัดส่งสินค้า 1	Char(70)		Yes
15	ORHSHA2	ที่อยู่จัดส่งสินค้า 2	Char(70)		Yes

ตารางที่ 4.2 ตารางข้อมูล OrderDetail

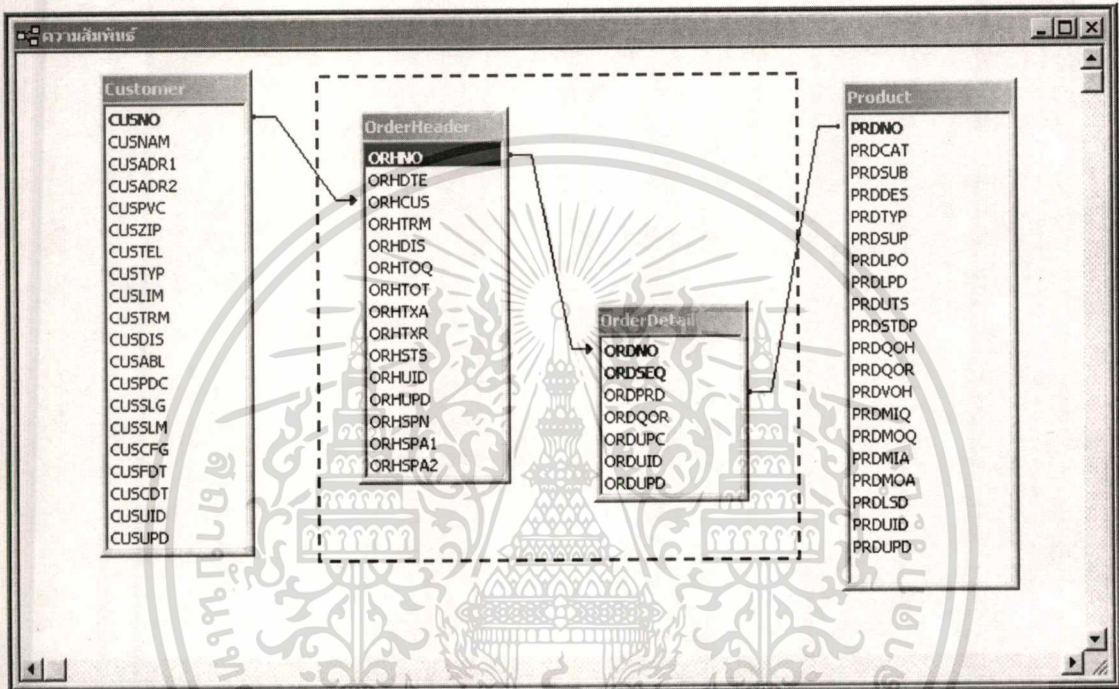
ลำดับที่	ชื่อ Field	คำอธิบาย	ชนิดของข้อมูล	Primary key	Not Null
1	<u>ORDNO</u>	เลขที่ใบสั่งซื้อ	Char(12)	Yes	Yes
2	<u>ORDSEQ</u>	ลำดับที่	Numeric(2)		Yes
3	ORDPRD	รหัสสินค้า	Char(20)		Yes
4	ORDQOR	จำนวนสินค้า	Numeric(2)		Yes
5	ORDUPC	ราคาต่อหน่วย	Numeric(11,2)		Yes
6	ORDUID	รหัสผู้ใช้งาน	Char(8)		Yes
7	ORDUPD	วันที่ปรับปรุงครั้งล่าสุด	Date		Yes

ตารางที่ 4.3 ตารางข้อมูล Running number

ลำดับที่	ชื่อ Field	คำอธิบาย	ชนิดของข้อมูล	Primary key	Not Null
1	<u>RUNTP</u>	เลขที่ใบสั่งซื้อ	Char(2)	Yes	Yes
2	<u>RUNNO</u>	ลำดับที่	Numeric(6)		Yes
3	<u>RUNUID</u>	รหัสผู้ใช้งาน	Char(8)		Yes
4	<u>RUNUPD</u>	วันที่ปรับปรุงครั้งล่าสุด	Date		Yes

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตารางด้านบนเป็นเพิ่มข้อมูลที่มีในระบบเอง แต่ในการทำงานนั้นระบบต้องอ้างอิงถึงข้อมูลที่เกี่ยวกับ ลูกค้า และรหัสสินค้าที่อยู่ในระบบอื่นๆ ในรูปที่ 4.1 เป็นการอธิบายถึงความสัมพันธ์ระหว่างเพิ่มข้อมูลประกอบด้วย เพิ่มข้อมูล Customer, OrderHeader, OrderDetail และ Product แสดงถึง foreign key ของตาราง OrderHeader และ OrderDetail ที่อ้างอิงถึง primary key ของตาราง Customer และ Product



รูปที่ 4.2 ความสัมพันธ์ระหว่างเพิ่มข้อมูล

บทที่ 5

การพัฒนาระบบ

5.1 เครื่องมือที่ใช้ในการพัฒนาระบบ

5.1.1 เครื่องคอมพิวเตอร์

หน่วยประมวลผลกลาง : Intel Pentium III 700 MHz

หน่วยความจำ : SDRAM 256 MB

ฮาร์ดดิส : 20 GB

5.1.2 ซอฟต์แวร์ต่างๆ

ระบบปฏิบัติการ : Windows 2000 Professional SP4

Browser : Internet Explorer 6.0

เครื่องมือพัฒนาภาษาจาวา : J2SDK 1.4.0

เครื่องมือพัฒนาเว็บแอปพลิเคชัน : Java Web Services Developer Pack (JWSDP)

ซอฟต์แวร์สร้างและแก้ไข Source code : EditPlus 2

เครื่องมือติดต่อข้อมูลบน AS/400 : AS/400 Toolbox for Java

: Client/Access Express for Windows V4.3

ฐานข้อมูลเชิงสัมพันธ์ของโรงงาน : SQL Server 2000

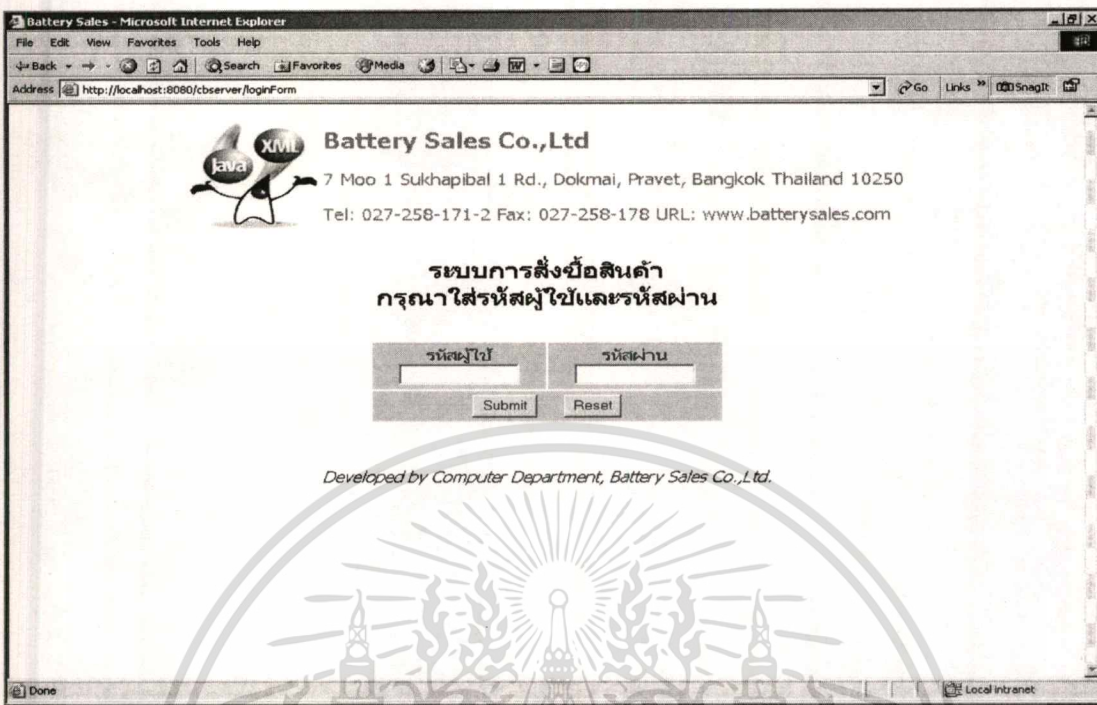
ระบบไฟล์ของ ตัวแทนจำหน่าย : AS/400 File System

Driver : SQL Server 2000 Driver for JDBC Service Pack 1

: JDBC Driver

5.2 รูปแบบการทำงานของระบบ

การทำงานของระบบสั่งซื้อสินค้าผ่านทางอินเทอร์เน็ตเริ่มจากการให้ลูกค้าเข้ามาที่เมนูการทำงานโดยผ่านทางบราวเซอร์ โดยจะต้องมีการตรวจสอบสิทธิ์ของการเข้าใช้งานในระบบก่อน ดังรูปที่ 5.1 ลูกค้า จะต้องใส่รหัสผู้ใช้และรหัสผ่าน เพื่อยืนยันการมีสิทธิ์เข้าใช้ระบบ



รูปที่ 5.1 หน้าจอตรวจสอบสิทธิ์การใช้งาน

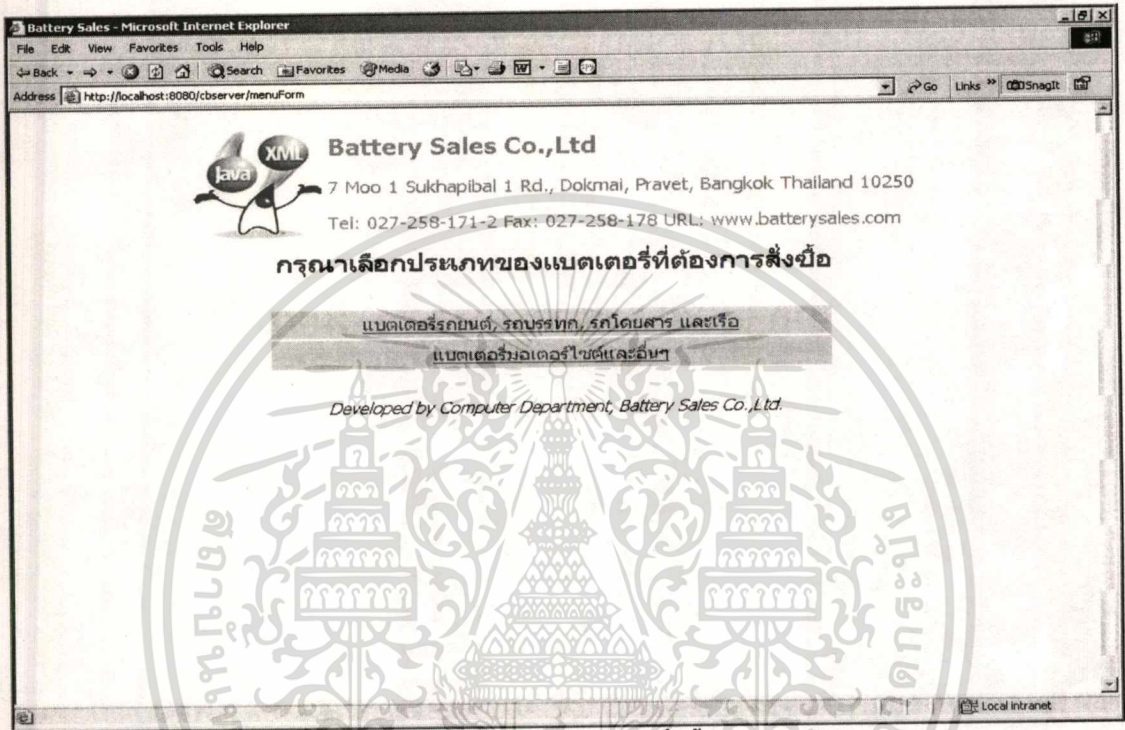
ถ้าหากลูกค้าใส่รหัสผ่านไม่ถูกต้อง ระบบจะแสดงข้อความดังรูปที่ 5.2



รูปที่ 5.2 หน้าจอแสดงข้อความเมื่อรหัสผ่านไม่ถูกต้อง

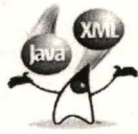
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากผ่านการตรวจสอบสิทธิ์การใช้งานแล้ว แอปพลิเคชันจะแสดงผลหน้าจอเมนูให้ลูกค้าสามารถเลือกหมวดของการตั้งสินค้าว่าเป็น แบตเตอรี่รถยนต์ หรือแบตเตอรี่รถมอเตอร์ไซด์ ดังรูปที่ 5.3 สาเหตุที่ให้ผู้ใช้งานเลือกหมวดกลุ่มสินค้า เนื่องจากต้องการให้ใบสั่งซื้อแต่ละใบอยู่ในหมวดกลุ่ม



รูปที่ 5.3 หน้าจอเมนูการสั่งซื้อ

สินค้าเดียวกันเพื่อใช้ประโยชน์ในระบบอื่นๆ เช่นในระบบลูกหนี้ จะมีการให้โบนัสส่วนลดปลายปีกับลูกค้าที่มียอดซื้อสินค้าตามหลักเกณฑ์ที่ตั้งไว้ตามหมวดกลุ่มสินค้า เมื่อผู้ใช้งานเลือกหมวดกลุ่มสินค้าแล้ว หน้าจอจะแสดงผลของแบบฟอร์มการทำรายการสั่งซื้อ ดังรูปที่ 5.4



Battery Sales Co.,Ltd

7 Moo 1 Sukhapibal 1 Rd., Dokmai, Pravet, Bangkok Thailand 10250

Tel: 027-258-171-2 Fax: 027-258-178 URL: www.batterysales.com

กรุณาเลือกรหัสสินค้าและใส่จำนวนที่ต้องการสั่งซื้อ จากนั้นคลิกปุ่ม “คำนวณราคา” เพื่อคำนวณราคาของสินค้า.

รหัสลูกค้า :

ชื่อลูกค้า :

เครดิต : วัน

Purchase Order Form				
Items	Part No.	Unit Price	Quantity	Amount
1	DD75	2150	0	0.00
2	D120	2200	0	0.00
3	D60	1400	0	0.00
4	D60L	1400	0	0.00
5	NS40	900	0	0.00
6	NS40L	900	0	0.00
7	NS60	1000	0	0.00
8	NS60L	1000	0	0.00
9	NS70	1300	0	0.00
10	NS70L	1300	0	0.00
11	N100	2050	0	0.00
12	N150	3150	0	0.00
13	N50	1120	0	0.00
14	SP100L	1135	0	0.00
15	SUPER DIN 75	1695	0	0.00
รวม			0	0.00
ส่วนลดทางการขาย %				0.00
มูลค่าสินค้า				0.00
ภาษีมูลค่าเพิ่ม 7 %				0.00
จำนวนเงินรวม				0.00
จำนวนบาท				

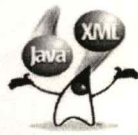
[จำนวนการต่อไป](#)

Developed by Computer Department, Battery Sales Co.,Ltd.

รูปที่ 5.4 หน้าจอแบบฟอร์มการสั่งซื้อ

ในหน้าจอนี้จะแสดงรหัสของสินค้าและราคาขาย และบอกขั้นตอนให้ ลูกค้า เลือกรหัสสินค้าและจำนวนที่ต้องการสั่งซื้อ จากนั้นให้คลิกปุ่ม “คำนวณราคา” จะได้ผลดังแสดงในรูปที่ 5.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Battery Sales Co.,Ltd

7 Moo 1 Sukhapibal 1 Rd., Dokmai, Pravet, Bangkok Thailand 10250

Tel: 027-258-171-2 Fax: 027-258-178 URL: www.batterysales.com

กรุณาเลือกรหัสสินค้าและใส่จำนวนที่ต้องการสั่งซื้อ จากนั้นคลิกปุ่ม **คำนวณราคา** เพื่อคำนวณราคาของสินค้า.

รหัสลูกค้า : 0005010

ชื่อลูกค้า : Ruchaneechol Co.,Ltd

เครดิต : 90 วัน

Purchase Order Form					
Items	Part No.	Unit Price	Quantity	Amount	
1	DD75	2150	5	10750.00	
2	D120	2200	10	22000.00	
3	D60	1400	10	14000.00	
4	D60L	1400	10	14000.00	
5	NS40	900	0	0.00	
6	NS40L	900	0	0.00	
7	NS60	1000	5	5000.00	
8	NS60L	1000	5	5000.00	
9	NS70	1300	10	13000.00	
10	NS70L	1300	10	13000.00	
11	N100	2050	0	0.00	
12	N150	3150	0	0.00	
13	N50	1120	8	8960.00	
14	SP100L	1135	8	9080.00	
15	SUPER DIN 75	1695	15	25425.00	
รวม			96	140215.00	
ส่วนลดทางการขาย 30 %				42064.50	
มูลค่าสินค้า				98150.50	
ภาษีมูลค่าเพิ่ม 7 %				6870.54	
จำนวนเงินรวม				105021.04	

คำนวณราคา

ทำการต่อไป

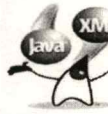
Developed by Computer Department, Battery Sales Co.,Ltd.

รูปที่ 5.5 หน้าจอแบบฟอร์มการสั่งซื้อเมื่อคลิกปุ่ม “คำนวณราคา”

ในหน้าจอนี้แสดงรหัสลูกค้า, ชื่อของลูกค้า พร้อมข้อมูลเรื่องเครดิตที่อยู่ในส่วนบนของฟอร์ม เพื่อยืนยันว่าใช่ลูกค้ารายนั้นจริงๆ และมีแสดงให้เห็นอัตราส่วนลดทางการขายและอัตราภาษีมูลค่าเพิ่มทางด้านล่างของฟอร์ม และถ้าผู้ใช้ต้องการแก้ไขรายการสั่งซื้อให้แก้ไขและคำนวณราคาใหม่ เมื่อผู้ใช้คลิกปุ่ม “ทำการต่อไป” จะปรากฏหน้าจอแสดงผลของการยืนยันสถานที่ส่ง

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์สำหรับใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปเผยแพร่ขึ้นต้นการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Battery Sales Co.,Ltd

7 Moo 1 Sukhapibal 1 Rd., Dokmai, Pravat, Bangkok Thailand 10250

Tel: 027-258-171-2 Fax: 027-258-178 URL: www.batterysales.com

*** รายการสั่งซื้อของท่านเสร็จสมบูรณ์แล้ว ***

รหัสลูกค้า : 0005010

ชื่อลูกค้า : Ruchaneechol Co.,Ltd

ที่อยู่ : 12 Sukhumvit 71 Rd. Prakanong
Wattana, Bangkok 10110

เลขที่ใบสั่งซื้อ : 600006

วันที่ใบสั่งซื้อ : Tue Oct 07

เครดิต : 90 วัน

วันที่จัดส่งสินค้า : Wed Oct 08

Purchase Order Detail		
Items	Part No.	Qty
1	DD75	5
2	D120	10
3	D60	10
4	D60L	10
5	NS60	5
6	NS60L	5
7	NS70	10
8	NS70L	10
9	NS50	8
10	SP100L	8
11	SUPER DIN 75	15

ทราบค่าใหม่

ออกจากรอบ

Developed by Computer Department, Battery Sales Co.,Ltd.

รูปที่ 5.7 หน้าจอสรุปผลการสั่งซื้อ

5.3 การรักษาความปลอดภัย

การรักษาความปลอดภัยในระบบเป็นลักษณะของการตรวจสอบสิทธิ์การเข้าใช้ระบบของลูกค้ ว่าตรงกับฐานข้อมูลของทางตัวแทนจำหน่ายหรือไม่ ถ้าไม่ตรงก็จะปฏิเสธการเข้าใช้ระบบ ซึ่งรหัสผู้ใช้และรหัสผ่านทางตัวแทนจำหน่ายจะเป็นผู้กำหนดให้กับลูกค้ และลูกค้ไม่สามารถเปลี่ยนแปลงรหัสผ่านเองได้ โดยทางตัวแทนจำหน่ายจะเปลี่ยนรหัสผ่านให้ลูกค้เป็นระยะทุก 6 เดือน เพื่อป้องกันการบุกรุกเข้ามาใช้ระบบโดยบุคคลอื่น

5.4 การบำรุงรักษาระบบ

สำหรับการบำรุงรักษาระบบนั้น เป็นการดูแลการทำงานของเว็บเซิร์ฟเวอร์ว่าสามารถให้บริการกับผู้ใช้ได้ดีหรือไม่ ตรวจสอบการทำงานของโปรแกรมในจุดที่มีความผิดพลาดเกิดขึ้น และแก้ไขให้ทำงานได้ดีขึ้น รวมทั้งตรวจสอบความถูกต้องของข้อมูลด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

ผลจากการดำเนินงาน

6.1 ผลจากการดำเนินงาน

สามารถออกแบบและพัฒนากระบวนการสั่งซื้อสินค้าผ่านทางอินเทอร์เน็ต ที่มีแนวคิดของการเชื่อมต่อข้อมูลระหว่างตัวแทนจำหน่ายและโรงงานผลิต โดยทางตัวแทนจำหน่ายสามารถยกเลิกการมีคลังสินค้าเก็บสินค้าที่สั่งมาจากโรงงานผลิตเพื่อรอรับคำสั่งซื้อจากลูกค้าที่สั่งซื้อผ่านทางตัวแทนขายหรือทางโทรศัพท์และทางแฟกซ์ ระบบใหม่นี้จะลดขั้นตอนการสั่งซื้อสินค้าโดยตัวแทนจำหน่ายจากโรงงานผลิต และการจัดส่งสินค้าที่มีค่าใช้จ่ายค่อนข้างสูง ให้อยู่ในขั้นตอนเดียวได้

ระบบสามารถเพิ่มข้อมูลของการสั่งซื้อแต่ละครั้งเข้าไปในแฟ้มข้อมูลที่เกี่ยวข้องของระบบได้ รวมทั้งยังสามารถให้ระบบอื่น ๆ นำไปใช้งานต่อได้ด้วย เช่นระบบขนส่งสินค้าสามารถนำเอารายละเอียดของการจัดส่งสินค้าไปใช้ หรือระบบขายลูกหนี้สามารถนำเอาข้อมูลไปใช้ในส่วนที่เกี่ยวกับการเก็บเงินได้เมื่อครบกำหนดชำระเงิน

6.2 ปัญหาและแนวทางในการแก้ไข

สามารถแยกเป็นประเด็นได้ดังนี้

- ปัญหาจากการรักษาความปลอดภัย เนื่องจากระบบที่ออกแบบไว้นั้นมีการรักษาความปลอดภัยเพียงตรวจสอบสิทธิ์ผู้เข้าใช้ระบบเท่านั้น ไม่ได้รวมถึงการป้องกันการขโมยข้อมูลที่มีการรับส่งระหว่างเซิร์ฟเวอร์ ดังนั้นจึงอาจทำให้ข้อมูลไม่ปลอดภัยได้ แนวทางแก้ไขควรมีวิธีการป้องกันการขโมยข้อมูลเหล่านี้โดยนำข้อมูลมาเข้ารหัสก่อนจะส่งออก
- ลูกค้าไม่สามารถติดตามใบสั่งซื้อของตนเองได้ว่าในขณะนี้มีสถานะเป็นเช่นไร เช่น มีการจัดส่งสินค้าแล้ว หรือยังไม่ได้จัดส่ง แนวทางแก้ไขระบบควรจะเชื่อมต่อข้อมูลเรื่องการจัดส่งสินค้ากับระบบขนส่งสินค้าเพื่อแสดงสถานะของใบสั่งซื้อให้ลูกค้าทราบ
- ลูกค้าไม่สามารถดูรายละเอียดใบสั่งซื้อย้อนหลังได้ แนวทางแก้ไขควรมีเพิ่มหน้าจอแสดงผลของใบสั่งซื้อแต่ละใบโดยสร้างเงื่อนไขในการค้นหาจากฐานข้อมูลให้กับลูกค้า เป็นการเพิ่มความสะดวกในการดูข้อมูลไม่ต้องพิมพ์ใบสั่งซื้อในลักษณะที่เป็นกระดาษเพื่อเก็บไว้อ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

วีระศักดิ์ ชิงฉาวร. 2543 **Java Programming Volume I**. กรุงเทพฯ: ซีเอ็ดดูเคชั่น

สุนทริน วงศ์ศิริกุล, 2543 สร้างเว็บให้ทันสมัย **XML Extensible Markup Language**. กรุงเทพฯ:

Successmedia

Kevin Williams, Michael Brundage, Patrick Denglet, Jeff Gabriel, Andy Hoskinson, Michael Kay, Thomas Maxwell, Marcelo Ochoa, Johnny Papa and Mohan Vanmane. 2001. **Professional XML Databases**. Birmingham: WROX Press.

Cynthia M. Murch , 1998 **AS/400 Toolbox for Java**. [Online]. Available:

[Http://www.ibm.com/servers/eserver/iserires/whpapr/toolbox_java.html](http://www.ibm.com/servers/eserver/iserires/whpapr/toolbox_java.html).

Rashin Mogha, V.V Preetham. 2002. **Java Web Services Programming**. New York: Wiley Publishing.

Ronald Bourret, 2001 **XML and Databases**. [Online]. Available:

[Http://www.rpbouret.com/xml/XMLAndDatabases.htm](http://www.rpbouret.com/xml/XMLAndDatabases.htm).

ประวัติผู้เขียน

ชื่อ-นามสกุล	นางสาวรัชนิชา แซ่เตีย
วัน/เดือน/ปี เกิด	1 เมษายน พ.ศ. 2510
สถานที่เกิด	จังหวัดตราด
ประวัติการศึกษา	
ประถมศึกษา	โรงเรียนเทศบาลวิมลวิทยา จังหวัดตราด
มัธยมศึกษาตอนต้น	โรงเรียนอัมพรไพศาล จังหวัดนนทบุรี
มัธยมศึกษาตอนปลาย	โรงเรียนศรีวิกรม์ จังหวัดกรุงเทพมหานคร
ปริญญาตรี	มหาวิทยาลัยรามคำแหง สาขารัฐศาสตร์การทูต
ประวัติการทำงาน	
พ.ศ. 2531- พ.ศ.2532	พนักงานแผนกคอมพิวเตอร์ บริษัท อิชเบลล่า(ไทยแลนด์) จำกัด
พ.ศ.2532 - พ.ศ.2543	โปรแกรมเมอร์, พนักงานดูแลระบบคอมพิวเตอร์ บริษัท สยามกลการอะไหล่ จำกัด
พ.ศ.2545 - ปัจจุบัน	หัวหน้าแผนกคอมพิวเตอร์ บริษัท สยามอีเอสเซลส์ จำกัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้