

การพัฒนาเว็บเซอร์วิสเพื่อการคำนวณคุณค่าทางโภชนาการของอาหาร  
Web Services Development for Nutritive Values Calculation

โดย

นายเลอชาติ ธรรมธีรเสถียร

รหัส 44067037



\*H001942\*

อาจารย์ที่ปรึกษา

รศ. บรรจง ปิยธำรง

วัน เดือน ปี..... 23 ต.ค. 2550.....  
เลขทะเบียน..... 01942.....  
เลขเรียกหนังสือ วท. ค 816 ก 2545.....  
"ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล."

รายงานนี้เป็นส่วนหนึ่งของวิชา โครงการพัฒนาระบบงาน  
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ  
ภาคเรียนที่ 2 ปีการศึกษา 2545  
คณะเทคโนโลยีสารสนเทศ  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ชื่อหัวข้อ            การพัฒนาเว็บเซอร์วิสเพื่อการคำนวณคุณค่าทางโภชนาการของอาหาร  
นักศึกษา                นายเลอชาติ ธรรมธีรเสถียร  
อาจารย์ที่ปรึกษา     รศ. บรรจง ปิยะขำรง  
ระดับการศึกษา        วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ  
แขนงวิชา              วิทยาการสารสนเทศ  
ปีการศึกษา             .2545

### บทคัดย่อ

การพัฒนาเว็บเซอร์วิสเพื่อการคำนวณคุณค่าทางโภชนาการของอาหารนี้ นอกจากจะเพื่อให้ได้มาซึ่งแอปพลิเคชันที่ใช้คำนวณคุณค่าทางโภชนาการของอาหาร โดยการคำนวณจากวัตถุดิบที่ใช้ปรุงอาหารนั้นๆ แล้ว ยังมีวัตถุประสงค์หลักอีกประการหนึ่งเพื่อศึกษาการทำงานของแอปพลิเคชันแบบกระจาย ในลักษณะเว็บเซอร์วิสโดยใช้ SOAP (Simple Object Access Protocol) เป็น โพรโตคอลหลัก ในการสื่อสารข้อมูลระหว่างแอปพลิเคชัน ซึ่งวิธีการในการพัฒนานี้จะเป็นการสร้างเว็บเซอร์วิสคอมโพเนนต์ต่างๆ ที่ใช้ในการคำนวณคุณค่าทางโภชนาการของอาหาร เพื่อรอการเรียกใช้บริการ โดยที่คอมโพเนนต์เหล่านั้น ถูกพัฒนาขึ้นจากรูปแบบที่แตกต่างกัน เช่น .NET web service, EJB และ ActiveX จากนั้นสร้างแอปพลิเคชันในการเรียกใช้บริการ โดยอาศัยข้อตกลงที่เป็นมาตรฐานเดียวกันในการสื่อสารข้อมูลระหว่างกัน ซึ่งผลที่คาดว่าจะได้รับการศึกษาพัฒนานี้คือ ทำให้เข้าใจการทำงานระหว่างแอปพลิเคชันหรือคอมโพเนนต์ของเว็บเซอร์วิส อีกทั้งเป็นแนวทางในการพัฒนาแอปพลิเคชันระดับธุรกิจในลักษณะเว็บเซอร์วิสต่อไป

**Title** Web Services Development for Nutritive Values Calculation  
**Student** Mr. Lerchart Thamtheerasathian  
**Advisor** Assoc. Prof. Banjong Piyatamrong  
**Level of Study** Master of Science in Information Technology  
**Major** Information Science  
**Academic Year** 2002

## ABSTRACT

Web services development for nutritive values calculation does not only provide an application that calculating the nutritive values from ingredients but also considers how distributed application applying SOAP (Simple Object Access Protocol) communicate with web service components. The development methodology is to create the nutritive values calculations components in forms of .NET web service, EJB and ActiveX and then waiting for service call from an application which regards the communication standard. The purpose of the development is to understand the interoperability among web service components. Moreover, it would be further development for enterprise web services.

## กิตติกรรมประกาศ

รายงานฉบับนี้สำเร็จลุล่วงลงได้ด้วยกำลังใจที่ดีจากหลายๆ ส่วน ขอขอบคุณครอบครัว พ่อแม่ และน้องสาว ที่ให้ความเข้าใจ ให้การสนับสนุนข้าพเจ้าเต็มทีในการศึกษา ขอขอบคุณเพื่อนๆ ที่คอยช่วยแสดงความคิดเห็น ทำให้เห็นมุมมองใหม่ๆ นอกเหนือจากความเห็นของข้าพเจ้าเอง ขอขอบคุณอาจารย์สถาบันวิจัยโภชนาการ มหาวิทยาลัยมหิดลที่ให้ความรู้เรื่องการคำนวณคุณค่าทางโภชนาการ และขอขอบคุณ รศ.บรรจง ปิยะธำรง ที่ให้คำปรึกษาที่ดีกับข้าพเจ้าเสมอมา

และสุดท้าย ขอขอบคุณทุกคนบนโลกที่มีความคิดเหมือนว่า การแบ่งปันความรู้ให้กับผู้อื่นนั้น ไม่ได้ทำให้ความรู้ของตัวเรานั้นลดน้อยลง

เลอชาติ ธรรมธรเสถียร

# สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
บทที่	
1. บทนำ.....	1
2. ทฤษฎีที่เกี่ยวข้อง.....	3
2.1 การคำนวณคุณค่าทางโภชนาการของอาหาร.....	3
2.2 เว็บเซอร์วิส.....	9
2.2.1 คำจำกัดความของเว็บเซอร์วิส.....	9
2.2.2 องค์ประกอบของเว็บเซอร์วิส.....	10
2.2.3 การทำงานของเว็บเซอร์วิสด้วย SOAP.....	14
3. การออกแบบเว็บเซอร์วิสเพื่อการคำนวณคุณค่าทางโภชนาการของอาหาร.....	16
3.1 การออกแบบเชิง Client/Server.....	16
3.1.1 ฟังก์ชันให้บริการ (Server).....	16
3.1.2 ฟังก์ชันเรียกใช้บริการ (Client).....	17
3.2 การออกแบบเชิงวัตถุโดยใช้ UML.....	18
3.2.1 Use Case Diagram.....	18
3.2.2 Class Diagram.....	19
3.2.3 Sequence Diagram.....	20
3.3 วิเคราะห์ Input/Process/Output.....	21
3.3.1 Interface Application.....	21
3.3.2 Web Services Components.....	22

4. การพัฒนาเว็บเซอร์วิสเพื่อการคำนวณคุณค่าทางโภชนาการ.....	26
4.1 Nutritive Value Calculation.....	28
4.1.1 การพัฒนาฐานข้อมูลคุณค่าทางโภชนาการของอาหาร.....	29
4.1.2 รายละเอียดการพัฒนาเว็บเซอร์วิส Nutritive Value Calculation.....	32
4.1.2.1 MainServices.....	32
4.1.2.2 NutrientDataAdmin.....	38
4.2 Measurement Conversion.....	53
4.3 Percentage of Requirement.....	59
5. การพัฒนาแอปพลิเคชันเพื่อเรียกใช้เว็บเซอร์วิสเพื่อการคำนวณ คุณค่าทางโภชนาการ.....	62
5.1 Client Interface Application.....	62
5.2 Data Admin Application.....	71
6. บทสรุปและข้อคิดเห็น.....	78
บรรณานุกรม.....	80
ประวัติผู้เขียน.....	81

# บทที่ 1

## บทนำ

แม้ว่าโลกในปัจจุบัน จะมีความเจริญก้าวหน้าทางวิทยาศาสตร์และเทคโนโลยีเป็นอย่างมาก แต่นั่นก็ไม่ได้เป็นเครื่องชี้วัดว่าประชากรที่อยู่ในสังคมที่เจริญก้าวหน้าเหล่านั้นจะมีสุขภาพดีตามไปด้วย ซึ่งประชากรที่มีสุขภาพดีเท่านั้นที่จะสามารถสร้างประโยชน์ให้กับสังคมได้อย่างเต็มที่

การที่จะมีสุขภาพดีได้นั้น อาจมาจากหลายๆ ปัจจัยเช่นการมีสุขภาพจิตที่ดี การอยู่ในสิ่งแวดล้อมที่ดี การออกกำลังกายอย่างสม่ำเสมอ แต่ปัจจัยที่ส่งผลต่อการมีสุขภาพดีมากที่สุดและเกี่ยวข้องกับชีวิตประจำวันของมนุษย์มากที่สุด นั่นคือ “การกิน” หรือการบริโภคอาหารเข้าไปในร่างกาย แต่ถ้าบริโภคเข้าไปโดยที่ไม่รู้เท่าทันอาหาร ก็อาจไม่เกิดผลดีต่อสุขภาพเท่าใดนัก ฉะนั้นการมีสุขภาพที่ดีนั้นส่วนสำคัญอย่างหนึ่งมาจากการบริโภคอาหารในสัดส่วนที่มีประโยชน์ต่อร่างกายเข้าไป ซึ่งนั่นก็คือการมีภาวะทางโภชนาการที่ดีนั่นเอง

คนเราทุกวันนี้หันมาให้ความสนใจกับสุขภาพ อาหารการกินมากยิ่งขึ้น แต่การคำนวณหาสัดส่วนคุณค่าทางโภชนาการที่จะได้รับจากอาหารก็ยังเป็นสิ่งที่ยากและเสียเวลาในความเข้าใจของคนโดยทั่วไป แต่ถ้าไม่มีการคำนวณดังกล่าวก็จะไม่สามารถทราบได้ว่า สิ่งที่เราบริโภคนั้นเหมาะสม สอดคล้องกับความต้องการของร่างกายหรือไม่ มากน้อยเพียงไร ซึ่งการคำนวณนี้เราสามารถทำให้มันง่ายและสะดวกต่อการใช้งานขึ้นด้วยเทคโนโลยีที่มีอยู่ในปัจจุบัน

อินเทอร์เน็ต เป็นเทคโนโลยีที่ได้รับความนิยมแพร่หลายเป็นอย่างมากในปัจจุบัน ซึ่งในอดีตเทคโนโลยีนี้มีจุดประสงค์เพื่อการแสดง หรือแลกเปลี่ยนข้อมูล เอกสารต่างๆ ระหว่างระบบเครือข่าย แต่ในปัจจุบัน ศักยภาพของอินเทอร์เน็ตได้พัฒนาไปถึงขั้นของการใช้แอปพลิเคชัน หรือเซอร์วิสต่างๆ ร่วมกันผ่านระบบเครือข่ายอินเทอร์เน็ต ซึ่งนั่นคือเทคโนโลยีที่เรียกว่าเว็บ เซอร์วิส (Web Services)

ในช่วงระยะเวลาหนึ่งถึงสองปีที่ผ่านมา เทคโนโลยีที่ได้รับการพูดถึงมากที่สุดเทคโนโลยีหนึ่ง คงหนีไม่พ้นเว็บเซอร์วิส เนื่องจากเป็นเทคโนโลยีที่ทำให้การสื่อสารแลกเปลี่ยนข้อมูลระหว่างแอปพลิเคชัน หรือคอมโพเนนต์ทำได้ง่ายขึ้น ซึ่งเทคโนโลยีของเว็บเซอร์วิสนี้มีโอกาสที่จะประสบความสำเร็จได้สูงมากในการเป็นแนวทางใหม่สำหรับการพัฒนาแอปพลิเคชันในอนาคต เนื่องจากบริษัทผู้ผลิตซอฟต์แวร์รายใหญ่ๆ ของโลก ไม่ว่าจะเป็นไมโครซอฟต์ ออเบียม ออราเคิล หรือซัน

เป็นต้น ต่างก็ให้ความสำคัญกับเทคโนโลยีนี้เป็นอย่างมาก จะเห็นได้จากมีผลิตภัณฑ์ที่สนับสนุนการพัฒนาเว็บเซอร์วิสอย่างจริงจังออกมาสู่ตลาดอย่างมากมาย จึงเป็นการดีที่จะได้มีการศึกษาเทคโนโลยีเว็บเซอร์วิสอย่างจริงจัง เพื่อการรู้เท่าทันเทคโนโลยี อีกทั้งจะทำให้เห็นจุดเด่นจุดด้อยของเทคโนโลยีนี้ในการปรับใช้ในสภาพของการทำงานจริงต่อไป

จากข้างต้นจะเห็นว่า การคำนวณหาปริมาณคุณค่าทางโภชนาการจากอาหาร เป็นสิ่งที่ เป็นประโยชน์ต่อการดูแลสุขภาพและภาวะทางโภชนาการ ฉะนั้นการพัฒนาแอปพลิเคชัน หรือเซอร์วิส เพื่อมารองรับการคำนวณนี้จะทำให้การคำนวณเป็นไปได้สะดวกและรวดเร็วมากยิ่งขึ้น และเพื่อให้ไม่มีข้อจำกัดในการนำไปใช้งาน จึงมีแนวคิดในการพัฒนาให้เป็นในรูปแบบของเว็บเซอร์วิส เนื่องจากลักษณะเฉพาะตัวของเว็บ เซอร์วิส คือยอมให้มีการเข้าใช้บริการได้ โดยไม่คำนึงว่าผู้ใช้จะอยู่บนระบบปฏิบัติการรูปแบบใดก็ตาม เพียงแค่ทำการร้องขอใช้บริการถูกต้องตามรูปแบบที่เว็บ เซอร์วิสประกาศไว้

ซึ่งเนื้อหาทั้งหมดของการพัฒนาเว็บเซอร์วิสเพื่อการคำนวณคุณค่าทางโภชนาการของอาหารนั้น จะแบ่งออกเป็น 3 ส่วน ส่วนแรกจะเป็นการศึกษาเนื้อหา และเทคโนโลยีที่เกี่ยวข้องกับการพัฒนา ส่วนที่สองจะเป็นการออกแบบเพื่อการพัฒนาในลักษณะต่างๆ และส่วนที่สามจะเป็นส่วนของการพัฒนาเว็บเซอร์วิสขึ้นตามที่ได้ออกแบบไว้ ซึ่งในส่วนของการพัฒนาเว็บเซอร์วิสเพื่อการคำนวณคุณค่าทางโภชนาการของอาหารนี้ มีวัตถุประสงค์หลักอีกประการหนึ่ง คือเพื่อต้องการศึกษาการพัฒนา การทำงาน และการเรียกใช้เว็บเซอร์วิส เพราะฉะนั้นจึงเลือกรูปแบบการพัฒนาให้มีความหลากหลาย ในสภาวะแวดล้อมของการพัฒนาแบบต่างๆ แล้วศึกษาการทำงานร่วมกันภายใต้สภาวะที่ต่างกันเหล่านั้น

## บทที่ 2

### ทฤษฎีที่เกี่ยวข้อง

ก่อนที่จะพัฒนาเว็บเซอร์วิสเพื่อการคำนวณคุณค่าทางโภชนาการของอาหารนั้น มีความจำเป็นที่จะต้องศึกษาถึงองค์ความรู้ต่างๆ ที่เกี่ยวข้องกับการพัฒนาเสียก่อน ซึ่งมีประเด็นที่ต้องศึกษาอยู่ด้วยกัน 2 ประเด็นหลักๆ ประเด็นแรกคือ ศึกษาถึงวิธีการในการคำนวณคุณค่าทางโภชนาการของอาหาร และประเด็นที่สองคือการศึกษาถึงลักษณะของเทคโนโลยีเว็บเซอร์วิสที่จะใช้ในการพัฒนา

#### 2.1 การคำนวณคุณค่าทางโภชนาการของอาหาร

ทุกวันนี้เราจะเห็นโดยทั่วไปว่า ผลิตภัณฑ์อาหารที่มีขายอยู่ในท้องตลาดนั้น ส่วนใหญ่จะมีตารางโภชนาการกำกับว่าผลิตภัณฑ์อาหารนั้นๆ มีสัดส่วนในแต่ละหมวดหมู่ทางโภชนาการเป็นเท่าใด ซึ่งผู้บริโภค ก็สามารถนำมาพิจารณาได้เลยว่า เหมาะสมกับที่ร่างกายต้องการหรือไม่ อย่างไร แต่ถ้าเป็นอาหารที่ปรุงเอง จะทราบได้อย่างไรว่าคุณค่าทางโภชนาการเป็นอย่างไร

การคำนวณหาคุณค่าทางโภชนาการจากอาหารอย่างแท้จริงและถูกต้องนั้น จะต้องกระทำขึ้นในห้องทดลองทางโภชนาการโดยผ่านกระบวนการ กรรมวิธีทางวิทยาศาสตร์ เช่นถ้าต้องการวิเคราะห์หาสารอาหารที่ได้จากใบกะเพรา ก็จะต้องนำใบกะเพราไปผ่านกระบวนการเตรียมวิเคราะห์ โดยแบ่งออกเป็นส่วนๆ เพื่อที่จะวิเคราะห์หาสารอาหารแต่ละประเภท ซึ่งบางประเภทใช้ตัวอย่างสด บางประเภทต้องนำไปบด หรือทำให้แห้งก่อน แล้วจึงนำไปวิเคราะห์

โดยทั่วไปนั้นจะทำการวิเคราะห์หาสารอาหารออกมาเป็นกลุ่มๆ เช่นกลุ่มพลังงาน กลุ่มวิตามิน หรือกลุ่มเกลือแร่ ซึ่งแต่ละกลุ่มก็จะมีวิธีการในการวิเคราะห์หาปริมาณในแต่ละแตกต่างกันไปซึ่งจะไม่ขอกกล่าวในเทคนิคที่ใช้ในการวิเคราะห์เหล่านั้น เนื่องจากยังมีวิธีการคำนวณหาคุณค่าทางโภชนาการของอาหารออกมาได้อีกวิธีหนึ่ง ซึ่งไม่ต้องอาศัยวิธีการที่ยุ่งยากในห้องทดลองแล้วให้ผลการวิเคราะห์ที่สามารถนำไปพิจารณาเรื่องการบริโภคได้โดยสังเขป นั่นคือการคำนวณวิเคราะห์จากเครื่องปรุง หรือส่วนผสมของอาหารนั้นเป็นหลัก ซึ่งวัตถุดิบต่างๆ ที่ใช้ในการปรุงอาหารส่วนใหญ่ก็ได้มีการคำนวณ และเผยแพร่เป็นตารางแสดงคุณค่าทางโภชนาการไว้แล้ว โดย

กองโภชนาการ กรมอนามัย กระทรวงสาธารณสุข และ USDA Nutrient Database for Standard Reference

ดังนั้นสิ่งที่ต้องใช้ในการคำนวณคุณค่าทางโภชนาการของอาหารมีดังนี้

- รายการส่วนผสมและเครื่องปรุงที่มีการระบุน้ำหนักไว้อย่างชัดเจน
- ตารางแสดงคุณค่าทางโภชนาการของอาหาร
- มาตรฐานเปรียบเทียบค่าหน่วยวัดในการชั่ง ตวง น้ำหนักของอาหาร
- ปริมาณแคลอรีและสารอาหาร ที่แนะนำให้บริโภคในแต่ละวัน

วิธีการคำนวณ เมื่อมีรายการส่วนผสมและเครื่องปรุงแล้ว ให้นำน้ำหนักของวัตถุดิบที่จะใช้ปรุงอาหารในแต่ละรายการมาแปลงหน่วยมาตรการชั่งตวง ให้มีหน่วยเป็นหน่วยเดียวกันซึ่งคือกรัม อีกทั้งเนื่องจากในตารางแสดงคุณค่าของอาหารนั้นจะแสดงสารอาหารต่างๆ เมื่อเทียบกับน้ำหนักของตัวอย่าง 100 กรัม โดยที่ มาตรฐานการแปลงหน่วยในการชั่ง ตวงอาหาร มีดังนี้

1 ช้อนชา	=	5 มิลลิลิตร
1 ช้อนโต๊ะ	=	14.8 มิลลิลิตร
1 ถ้วยตวง	=	236.59 มิลลิลิตร
2 ถ้วยตวง	=	1 ไปน
2 ไปน	=	1 ควอท
4 ควอท	=	1 แกลลอน
2 แกลลอน	=	1 เป็ก
4 เป็ก	=	1 มุชเชล
1 ออนซ์	=	28.35 กรัม
8 ออนซ์	=	1 ถ้วยตวง
16 ออนซ์	=	1 ปอนด์
1 กิโลกรัม	=	2.2 ปอนด์
1 ปอนด์	=	453.59 กรัม

และมาตรฐานเทียบเท่าที่ควรรู้ได้แก่

ไข่ไก่ 1 ฟอง	=	¼ ถ้วย
ไข่แดง 1 ฟอง	=	1 ½ ช้อนโต๊ะ
ไข่ขาว 1 ฟอง	=	2 ช้อนโต๊ะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากแปลงหน่วยในการชั่ง ตวง วัดดูดิบให้เป็นกรัมเรียบร้อยแล้ว จากนั้นจะทำการ คั้นหาวัตถุดิบที่จะใช้ปรุงอาหารแต่ละรายการ ในตารางแสดงคุณค่าทางโภชนาการ ซึ่งจะแสดงถึง รายละเอียดสารอาหารที่มีอยู่ในวัตถุดิบแต่ละรายการนั้นเทียบต่อ 100 กรัม จากนั้นนำมาคำนวณ ตามสัดส่วนของสารอาหารตามน้ำหนักของแต่ละส่วนผสมวัตถุดิบนั้น แล้วจึงรวมค่าสารอาหาร และพลังงานต่างๆ ที่ได้รับจากอาหารทั้งหมด ซึ่งถ้าอาหารนั้นเตรียมเครื่องปรุงไว้สำหรับ หลายคน บริโภค ก็สามารถปรับลดอัตราส่วนตามจำนวนที่ได้บริโภคตามความต้องการ

เมื่อได้คำนวณสัดส่วนคุณค่าทางโภชนาการของอาหารนั้นๆ ออกมาแล้ว สิ่งที่ได้มานั้นก็ บอกได้แต่เพียงว่าสูตรหรือเครื่องปรุงอาหารที่ทำการคำนวณนั้น มีคุณค่าทางโภชนาการและ สารอาหารเป็นจำนวนเท่าใด แต่ก็ไม่สามารถบ่งบอกถึงว่า สารอาหารเหล่านั้นเป็นสัดส่วนที่ เหมาะสมตามที่ร่างกายต้องการหรือไม่ เนื่องจากการบริโภคอาหารไม่ว่าจะบริโภคมามากหรือน้อย เกินไปนั้นมักจะไม่เป็นผลดีต่อร่างกายเท่าใดนัก ดังนั้นสิ่งที่ต้องทำการคำนวณต่อมาคือ นำ สารอาหาร ที่คำนวณได้ทั้งหมด มาเทียบกับมาตรฐานหรือสัดส่วนปริมาณที่กองโภชนาการ กรมอนามัย กระทรวงสาธารณสุขแนะนำให้รับประทาน ซึ่งมีดังนี้

#### ปริมาณแคลอรีที่ควรบริโภคใน 1 วัน

ชาย อายุ	20-29	ปริมาณที่แนะนำ	2550
	30-39		2450
	40-49		2350
	50-59		2200
	60-69		2000
	70+		1750
หญิงอายุ	20-29	ปริมาณที่แนะนำ	1800
	30-39		1700
	40-49		1650
	50-59		1550
	60-69		1450
	70+		1250

หญิงมีครรภ์ ควรเพิ่มจากปริมาณที่ควรได้รับอีก 180

หญิงให้นมบุตร ควรเพิ่มจากปริมาณที่ควรได้รับอีก 900

เด็ก อายุ	1-3	ปริมาณที่แนะนำ	1200
	4-6		1550
	7-9		1900
	10-12		2300
เด็กชายอายุ	13-15	ปริมาณที่แนะนำ	2800
	16-19		3300
เด็กหญิงอายุ	13-15	ปริมาณที่แนะนำ	2350
	16-19		2200

ปริมาณสารอาหารต่างๆ ที่แนะนำให้บริโภค

โปรตีนทั้งหมด	47-54	กรัม
ไขมัน	ไม่น้อยกว่า 1/5 ของแคลอรี	
คาร์โบไฮเดรต	ไม่น้อยกว่า 1/5 ของแคลอรี	
แคลเซียม	400-500	มิลลิกรัม
เหล็ก	6-16	มิลลิกรัม
วิตามินเอ	2500	หน่วยสากล
วิตามินอี	5-30	มิลลิกรัม
วิตามินซี	30	มิลลิกรัม
วิตามินบี1	0.05% ของแคลอรี หรือ 0.7-1 มิลลิกรัม	
วิตามินบี2	0.055% ของแคลอรี หรือ 1-1.4 มิลลิกรัม	
ไนอาซิน	0.66% ของแคลอรี หรือ 12-17 มิลลิกรัม	
วิตามินบี5	5-30	มิลลิกรัม
วิตามินบี6	5-10	มิลลิกรัม
โฟเลต	100	ไมโครกรัม
วิตามินบี12	6	ไมโครกรัม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ตัวอย่างการคำนวณ

ตัวอย่างที่จะใช้แสดงการคำนวณคุณค่าทางโภชนาการนี้ นั้น เป็น การทำเนื้อผัดน้ำมันหอย โดยมีเครื่องปรุงส่วนผสมอย่างง่าย ๆ ดังต่อไปนี้

เนื้อสันใน	300	กรัม
แป้งมัน	1	ช้อนโต๊ะ
น้ำตาล	1	ช้อนชา
น้ำมันหอย	¼	ถ้วยตวง
น้ำมันพืช	¾	ถ้วยตวง
น้ำสต็อกหมู	1	ถ้วยตวง

จากนั้น จะทำการเปลี่ยนหน่วยการวัดตวงให้อยู่ในรูปกรัม เพื่อนำไปคำนวณเปรียบเทียบ จากตารางแสดงคุณค่าทางโภชนาการที่แสดงไว้จากหน่วย 100 กรัม

เนื้อสันใน	300	กรัม
แป้งมัน	$1 \times 14.8$	$= 14.8$ กรัม
น้ำตาล	$1 \times 5$	$= 5$ กรัม
น้ำมันหอย	$\frac{1}{4} \times 236.59$	$= 59.15$ กรัม
น้ำมันพืช	$\frac{3}{4} \times 236.59$	$= 177.44$ กรัม
น้ำสต็อกหมู	$1 \times 236.59$	$= 236.59$ กรัม

แล้วจึงนำไปคำนวณเทียบกับคุณค่าทางโภชนาการที่แสดงในตารางแสดงคุณค่าทางโภชนาการ ซึ่งผลการคำนวณที่ได้เป็นดังตาราง 2.1 โดยที่ตัวเลขที่แสดงในตาราง ในแต่ละช่องนั้น ด้านบนจะเป็นค่าที่ได้จากการเปิดจากตาราง ซึ่งแสดงไว้เทียบกับปริมาณส่วนผสม 100 กรัม และตัวเลขด้านล่างจะแสดงค่าที่ได้จากการคำนวณเทียบปรับให้เป็นไปตามปริมาณส่วนผสมที่ ใส่ลงไปในการอาหาร และสุดท้าย ค่าที่แสดงในแถวล่างสุดของ ตารางก็คือปริมาณสัดส่วนร้อยละของเนื้อผัดน้ำมันหอย ต่อปริมาณความต้องการสารอาหารต่างๆของร่างกาย แต่เนื่องจากปริมาณส่วนผสมของอาหารนี้ นั้น อาจจะเป็นส่วนผสมเพื่อการบริโภคได้หลายหน่วย ดังนั้นหากต้องการทราบว่าในแต่ละหน่วยบริโภค หรือเฉพาะส่วนที่บริโภคเข้าไปนั้น มีปริมาณสารอาหารเป็นเท่าใด ก็สามารถปรับลดตัวเลขที่คำนวณออกมาได้เหล่านั้น ตามอัตราส่วนที่บริโภคได้

ตาราง 2.1 แสดงคุณค่าทางโภชนาการของเนื้อผัดน้ำมันหอย โดยกำหนดให้ความต้องการพลังงานต่อหนึ่งวันเป็น 2550 แคลอรี

	Proximate Analysis				Minerals		Fat Soluble Vitamins		Water Soluble Vitamins							
	Protein	Total lipid (fat)	Carbohydrate	Energy	Calcium, Ca	Iron, Fe	Vitamin A, IU	Vitamin E	Vitamin C, ascorbic acid	Vitamin B1, Thiamin	Vitamin B2, Riboflavin	Vitamin B3, Niacin	Vitamin B5, Pantothenic acid	Vitamin B-6	Folate	Vitamin B-12
	g	g	g	kcal	mg	mg	IU	mg	Mg	mg	mg	mg	mg	mg	mcg	mcg
เนื้อสันใน	17.76	23.02	0	283	7	2.29	0	0	0	0.11	0.2	2.93	0.3	0.37	6	2.55
3	53.28	69.06	0	849	21	6.87	0	0	0	0.33	0.6	8.79	0.9	1.11	18	7.65
น้ำมันหอย	1.35	0.25	10.92	51	32	0.18	23	0.035	0.1	0.01	0.124	1.474	0.016	0.016	15	0.41
0.5915	0.799	0.148	6.459	30.167	18.928	0.106	13.605	0.021	0.059	0.006	0.073	0.872	0.009	0.009	8.873	0.243
น้ำมันพืช	0	100	0	763	0	0	0	5.17	0	0	0	0	0	0	0	0
1.7744	0	177.44	0	1353.87	0	0	0	9.17365	0	0	0	0	0	0	0	0
น้ำตาลดอกหมุ	1.97	1.58	4.65	40	9	0.35	1611	0	0.8	0.018	0.021	0.494	0.14	0.036	4	0
2.3659	4.661	3.738	11.001	94.636	21.293	0.828	3811.465	0.000	1.893	0.043	0.050	1.169	0.331	0.085	9.464	0.000
น้ำตาล	0	0	99.9	387	1	0.06	0	0	0	0	0.019	0	0	0	0	0
0.05	0	0	4.995	19.35	0.05	0.003	0	0	0	0	0.001	0	0	0	0	0
แป้งมัน	0.3	0.1	87.2	351	84	0.9	0	0	0	0.06	0.02	0.1	0	0	0	0
0.148	0.0444	0.0148	12.906	51.948	12.432	0.1332	0	0	0	0.0089	0.003	0.0148	0	0	0	0
รวม	58.784	250.401	35.361	2398.968	73.703	7.941	3825.069	9.194	1.952	0.387	0.727	10.845	1.241	1.205	36.336	7.893
มาตรฐาน	54	510	510	2550	500	16	2500	30	30	1.275	1.4025	16.83	10	2	100	6
%	108.86	49.10	6.93	94.08	14.74	49.63	153.00	30.65	6.51	30.38	51.83	64.44	12.41	60.23	36.34	131.54

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.2 เว็บ เซอร์วิส (Web Services)

เว็บเซอร์วิส ได้รับความสนใจเป็นอย่างมากในปัจจุบัน เนื่องจาก บริษัทผู้ผลิตและพัฒนาซอฟต์แวร์ แอปพลิเคชันต่างๆ ได้ให้ความสำคัญกับการสร้าง Distributed Application ยกตัวอย่างไม่ว่าจะเป็น Windows DNA ของ Microsoft หรือ J2EE ของ Java ซึ่งต่างก็มีความพยายามที่จะทำให้ แอปพลิเคชันของตนสามารถเรียกคอมโพเนนต์ที่อยู่ต่างสถานที่กัน ได้ แต่ถึงอย่างไรก็ตาม ยังเป็นการเรียกใช้คอมโพเนนต์ที่อยู่ภายใต้ระบบปฏิบัติการรูปแบบเดียวกัน เว็บเซอร์วิสจึงเข้ามาช่วยลบล้างข้อจำกัดในจุดนี้ ทำให้คอมโพเนนต์ที่สร้างขึ้น สามารถถูกเรียกใช้จากต่างสถานที่ และต่างรูปแบบระบบปฏิบัติการได้

### 2.2.1 คำจำกัดความของเว็บเซอร์วิส

มีการนิยามลักษณะของเว็บเซอร์วิสไว้ ในความหมายต่างๆ เช่น

- เว็บเซอร์วิสคือแอปพลิเคชันที่สามารถใช้งานร่วมกันได้ แม้ว่าผู้เรียกใช้จะอยู่ต่างระบบก็ตาม โดยผ่านช่องทางของเว็บ
- เว็บเซอร์วิสคือแอปพลิเคชันที่มีลักษณะเป็น โปรแกรมย่อยๆ ที่สามารถอธิบายลักษณะการให้บริการของตัวเองได้ มีการเผยแพร่ มีการบอกที่อยู่ และสามารถเรียกใช้งานผ่านระบบเครือข่ายได้
- เว็บเซอร์วิสเป็นแอปพลิเคชันที่ยอมให้เรียกใช้จากระบบอื่นข้ามเครือข่ายได้ มีข้อตกลงที่มีลักษณะง่ายต่อการใช้งาน อีกทั้งเป็นเทคโนโลยีการสื่อสารที่ไม่ยึดติดกับผลิตภัณฑ์อย่างใดอย่างหนึ่ง
- เว็บเซอร์วิสเป็นโปรแกรมที่สามารถถูกเรียกได้โดยโปรแกรมอื่นผ่านเว็บ ซึ่งสามารถใช้รูปแบบของการพัฒนา และระบบปฏิบัติการที่หลากหลาย

จะเห็นได้ว่าเว็บเซอร์วิส ถูกให้คำจำกัดความไว้อย่างกว้างๆ ว่าเป็น กลไกที่จะทำให้เกิดการใช้เซอร์วิสคอมโพเนนต์ หรือแอปพลิเคชัน ที่มีลักษณะเป็นการทำงานย่อยร่วมกัน ไม่มีการขึ้นอยู่กับสิ่งใดสิ่งหนึ่ง ไม่ว่าจะเป็นภาษาในการพัฒนา หรือระบบปฏิบัติการ มีการทำงานผ่านระบบเครือข่ายอินเทอร์เน็ต อินทราเน็ตที่มีอยู่เดิม โดยใช้เทคโนโลยี XML เป็นพื้นฐาน โดยที่เว็บเซอร์วิส นั้นสามารถใช้ประโยชน์จากสิ่งที่มีอยู่เดิมได้ เช่นมีคอมโพเนนต์อยู่แล้ว ก็สามารถนำมาประกาศขึ้นเป็นเว็บเซอร์วิสได้ ไม่จำเป็นต้อง Reengineering ระบบใหม่ทั้งหมด ซึ่งเรียกได้ว่าเว็บเซอร์วิสเป็นวิวัฒนาการของการพัฒนาแอปพลิเคชันให้มีประสิทธิภาพในการทำงานร่วมกันมากขึ้น ไม่ใช่การปฏิรูปเปลี่ยนแปลงการพัฒนาแอปพลิเคชัน

## 2.2.2 องค์ประกอบของเว็บเซอร์วิส

การที่จะทำให้เว็บเซอร์วิสมีความสามารถดังที่ได้ให้คำจำกัดความไว้นั้น มีองค์ประกอบหลักของการทำงานอยู่ 4 ประการด้วยกันคือ

- เซอร์วิส คอมโพเนนต์ หรือ แอปพลิเคชัน ที่จะให้บริการ โดยพัฒนาจากภาษาใดก็ได้ และอยู่บนแพลตฟอร์มระบบปฏิบัติการใดก็ได้ ซึ่งให้บริการได้โดยผ่านระบบเครือข่าย และรูปแบบเครือข่ายที่ใช้ก็คือ อินเทอร์เน็ตและอินทราเน็ตที่มีอยู่เดิม เนื่องจากอินเทอร์เน็ต มีความสามารถในการเข้าถึงได้ทุกที่ทุกเวลา หลากหลายรูปแบบ อีกทั้งมีค่าใช้จ่ายในการเข้าใช้ที่ต่ำ ถึงแม้ว่าจะมีความเร็วที่ไม่มากนัก แต่ก็อยู่ในสภาวะที่ยอมรับได้
- ระบบที่พัฒนาขึ้นเป็นเว็บเซอร์วิสนั้น ต้องมีความสามารถในการค้นหาเพื่อที่จะเข้าใช้ระบบอื่นได้ ซึ่งมีสิ่งที่เกี่ยวข้องอยู่ 2 เรื่องด้วยกัน คือต้องมีเครื่องมือช่วยที่ใช้ในการค้นหา และสิ่งที่อธิบายถึงการให้บริการและวิธีการเข้าใช้ ในส่วนนี้ได้มีการคิดเทคโนโลยีขึ้นมารองรับ เช่น WSDL (Web Services Description Language) ซึ่งเป็นการนิยามว่าเว็บเซอร์วิสนั้นๆ สามารถให้บริการอะไรได้บ้าง และ UDDI (Universal Description, Discovery and Integration) เป็นเทคโนโลยีที่ใช้ค้นหาเว็บเซอร์วิสที่มีอยู่อย่างมากมายในอินเทอร์เน็ต
- ภาษากลางเพื่อใช้อธิบายข้อมูล ซึ่งเทคโนโลยีที่นำมาใช้คือ XML เนื่องจากลักษณะเฉพาะตัวของ XML ซึ่งมีบทบาทเป็น ภาษาของข้อมูล โดยเฉพาะอย่างยิ่งสำหรับอินเทอร์เน็ต อีกทั้งด้วยการที่ XML มีลักษณะเป็นข้อความธรรมดาทำให้มีความสามารถทำงานร่วมกันได้กับทุกภาษา ทุกระบบปฏิบัติการ และยังง่ายต่อการหาเครื่องมือในการตีความ (Parser) ทำให้เทคโนโลยีที่เกี่ยวข้องกับเว็บเซอร์วิสทั้งหมดมีพื้นฐานมาจาก XML ทั้งสิ้น
- ข้อตกลง หรือ โปรโตคอลที่จะใช้ร่วมกัน เพื่อให้การแลกเปลี่ยนข้อมูลระหว่างกันเป็นไปในรูปแบบที่สามารถเข้าใจร่วมกันได้ จะต้องมีโปรโตคอลที่ตกลงกันไว้ร่วมกัน และโปรโตคอลที่นำมาใช้นี้ จะต้องมีความสามารถในการทำงานบนทุกระบบปฏิบัติการ และสามารถทำงานบนอินเทอร์เน็ตโปรโตคอลที่มีอยู่เดิมได้เป็นอย่างดี ซึ่งโปรโตคอลที่ได้รับนิยามม ใช้ในงานของเว็บเซอร์วิส คือ SOAP (Simple Object Access Protocol)

การพัฒนาเซอร์วิสคอมโพเนนต์ให้มีลักษณะเป็นเว็บเซอร์วิสในปัจจุบันนั้น ส่วนใหญ่จะเป็นไปในแนวทางเดียวกันคือใช้ SOAP และ WSDL เป็นแกนหลักในการพัฒนา แต่ถ้าพิจารณาจากนิยามของเว็บเซอร์วิสแล้ว นั้น อาจไม่จำเป็นต้องใช้ข้อกำหนดของ SOAP และ WSDL ก็ได้

สามารถใช้ข้อกำหนดอื่น แต่ต้องอยู่บนพื้นฐานของเทคโนโลยี XML เหมือนกัน และเป็นที่ยอมรับในการสื่อสารร่วมกัน

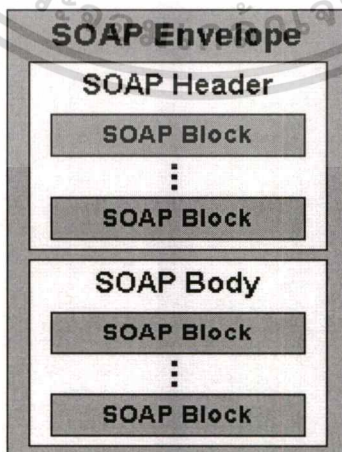
ในการพัฒนาเว็บเซอร์วิสเพื่อการคำนวณคุณค่าทางโภชนาการของอาหารนี้ จะมุ่งเน้นการศึกษาองค์ประกอบต่างๆ และการทำงานของเว็บเซอร์วิส ที่ใช้ SOAP เป็นโปรโตคอลในการส่งข้อมูลระหว่างกัน และใช้ WSDL ในการอธิบายการลักษณะของเว็บเซอร์วิส ต่อไปจะเป็นรายละเอียดของเทคโนโลยีที่เป็นองค์ประกอบหลักในการทำงานของเว็บเซอร์วิส ที่ทำให้เว็บเซอร์วิสสามารถบรรลุเป้าหมายในการเรียกใช้เซอร์วิส คอมโพเนนต์ หรือแอปพลิเคชัน โดยอยู่ต่างสถานที่และต่างระบบปฏิบัติการได้

**SOAP (Simple Object Access Protocol)**

SOAP เป็นกลไกการแลกเปลี่ยนข้อมูลภายใต้สภาพแวดล้อมแบบกระจาย โดยอาศัยเทคโนโลยีของ XML เป็นพื้นฐาน SOAP ไม่ใช่ภาษาในการเขียนโปรแกรม แต่เป็นเพียงเครื่องมืออย่างหนึ่งสำหรับส่งข้อมูล มีลักษณะคล้ายซองจดหมายในการส่งข้อมูลการเรียกใช้ไปยังแอปพลิเคชัน เซอร์วิสปลายทาง จึงมีการนำ SOAP ไปใช้ในระบบหลากหลายรูปแบบตั้งแต่ Messaging System จนถึง Remote Procedure calls (RPC)

SOAP มีองค์ประกอบหลักที่ทำงานประสานกันอยู่ 4 องค์ประกอบด้วยกันดังนี้

1. SOAP Envelope เปรียบเสมือนซองจดหมาย มีขึ้นเพื่อเป็นกรอบการอธิบายการทำงาน ต้องทำงานอย่างไรกับใคร มีอะไรใน message บ้าง และสิ่งที่ทำนั้นเป็นเพียงตัวเลือกหนึ่งหรือเป็นสิ่งที่จำเป็นต้องทำ และสิ่งที่อยู่ภายในส่วน Body ก็จะเป็นข้อมูลค่า Argument ที่จำเป็นต่อการประมวลผลในเว็บเซอร์วิส ที่จะเรียกใช้ หรือเป็นค่าที่ส่งกลับมาจากการประมวลผลในกรณีตอบกลับ



รูปที่ 2.1 แสดงลักษณะตามแนวคิดของ SOAP message

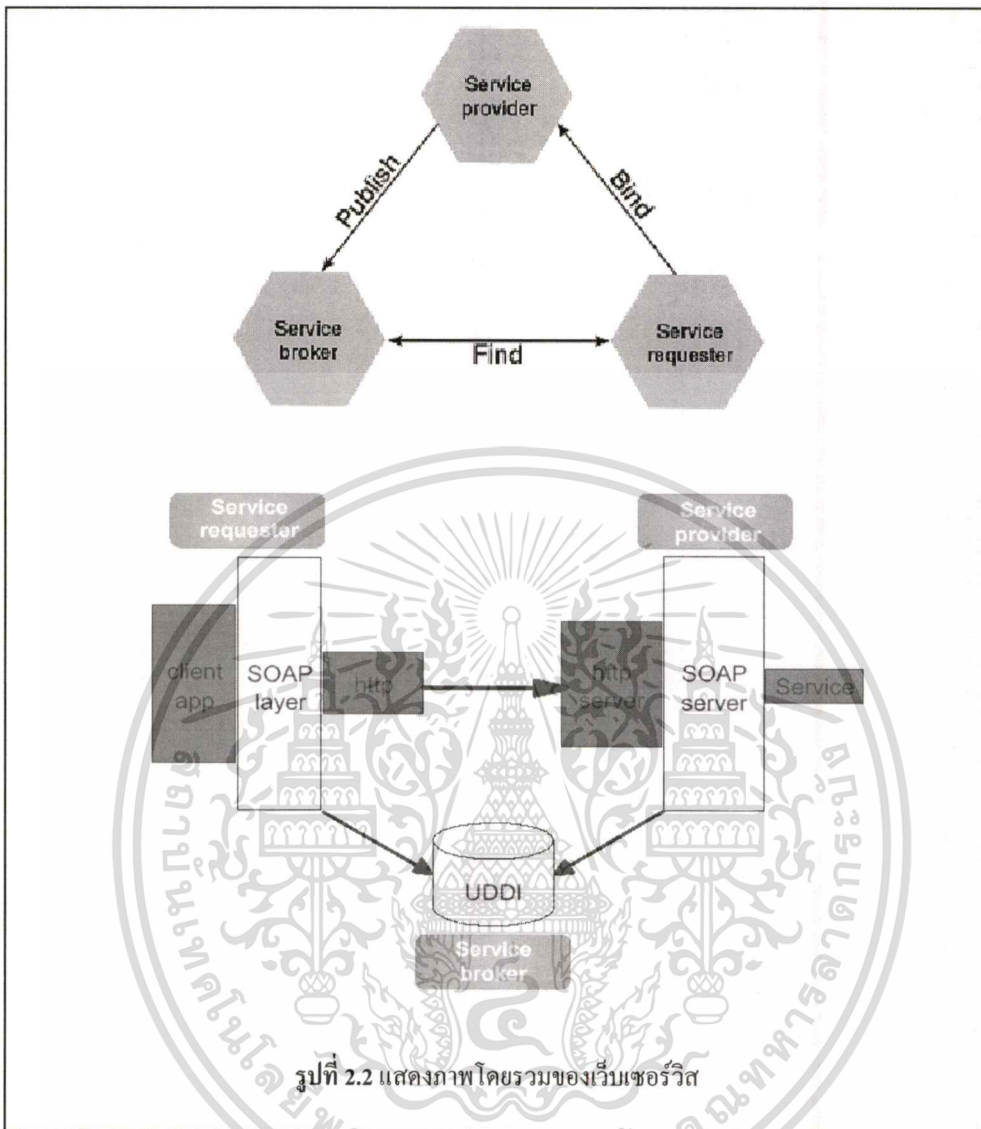
2. SOAP Encoding Rules หรือ กฎการเข้ารหัสของข้อมูล SOAP เป็นส่วนของการกำหนดกลไกที่ใช้ในการแลกเปลี่ยนชนิดของโครงสร้างข้อมูลที่ใช้ในแอปพลิเคชันที่เรียกใช้
3. SOAP RPC เป็นการนิยามรูปแบบการใช้ remote procedural calls และการตอบกลับ
4. SOAP Binding เป็นข้อกำหนดการแลกเปลี่ยน SOAP envelope ระหว่างกันภายใต้โปรโตคอลที่ใช้ในการสื่อสาร ที่มีอยู่เดิมในระบบอินเทอร์เน็ตปัจจุบัน ซึ่งในปัจจุบันนิยมทำงานอยู่บนโปรโตคอล HTTP

### WSDL (Web Services Description Language)

ในการพัฒนาเว็บเซอร์วิส คอมโพเนนต์ที่พัฒนาขึ้นนั้นจะมีลักษณะข้อเว็บเซอร์วิสได้ก็คือเมื่อมีเอกสาร WSDL เป็นเอกสารที่ใช้โครงสร้างตามรูปแบบภาษา XML ที่ได้รับการควบคุมมาตรฐานโดย W3C มีลักษณะคล้ายเป็น Interface Definition Language เพื่อแสดงคุณสมบัติของเว็บเซอร์วิสว่าคอมโพเนนต์ที่ประกาศขึ้นเป็นเว็บเซอร์วิสนั้น สามารถให้บริการอะไรได้บ้าง ซึ่งในรายละเอียดของเอกสาร WSDL จะเป็นการบอกถึงชื่อของเว็บเซอร์วิสที่ต้องใช้อ้างเมื่อต้องการใช้บริการ, ชื่อของ methods ที่ให้บริการ, arguments ต่างๆ ที่จะใช้ในการประมวลผลของแต่ละ methods, ลักษณะรูปแบบข้อมูลที่จะตอบกลับมาจากเซอร์วิส, ที่อยู่ของ SOAP Listener ซึ่งจะเป็นตัวที่คอยจับว่า มีการเรียกใช้บริการเว็บเซอร์วิสหรือไม่ ตลอดจนรายละเอียดเพิ่มเติมอื่นๆ เกี่ยวกับเซอร์วิส

### UDDI (Universal Description, Discovery and Integration)

UDDI เป็นเครื่องมือที่ช่วยทำให้เกิดการใช้เว็บเซอร์วิส ร่วมกัน ในการสืบค้นหาเว็บเซอร์วิส (Service Discovery) ที่เปิดให้บริการอยู่ในเครือข่ายอินเทอร์เน็ต เว็บเซอร์วิสใดที่ต้องการเผยแพร่ให้ผู้ใช้งานนอกสามารถเข้ามาใช้บริการ ก็สามารถขึ้นทะเบียนเว็บเซอร์วิสกับ UDDI ได้ ซึ่งจะเป็นประโยชน์กับผู้พัฒนาแอปพลิเคชัน เนื่องจากจะได้ไม่ต้องพัฒนาในสิ่งที่มีอยู่แล้ว ให้สามารถติดต่อร้องขอเรียกใช้ได้ตามข้อตกลง วิธีการทำงานของ UDDI นั้น ก็จะใช้วิธีการแลกเปลี่ยนข้อมูลหาข้อมูล และที่อยู่ของเว็บเซอร์วิสกับ UDDI ด้วยกัน โดยอาศัยความสามารถในการทำงานข้ามระบบปฏิบัติการของ SOAP และ XML เป็นพื้นฐาน ผลที่ได้จากการสืบค้นด้วย UDDI นั้น จะถูกตอบกลับมาเป็นค่าต่างๆ เช่นสิ่งที่ได้จากการใช้บริการ รายละเอียดที่จะใช้ในการอ้างอิงเชื่อมโยงกับเว็บเซอร์วิสนั้น และการติดต่อกับผู้ให้บริการเว็บเซอร์วิส ซึ่ง UDDI ก็ได้รวบรวมรายละเอียดเหล่านี้จาก เอกสาร WSDL ที่ประกาศขึ้นมาจากเว็บเซอร์วิสนั่นเอง



เมื่อนำองค์ประกอบหลักๆ ของเว็บเซอร์วิส มามองอย่างภาพกว้าง จะได้ลักษณะดังรูปที่ 2 ซึ่งอธิบายได้ว่า การทำงานโดยทั่วไปของเว็บเซอร์วิส นั้น จะมีอยู่ด้วยกัน 3 ส่วนใหญ่ๆ ซึ่งมีหน้าที่แตกต่างกันคือ

- Service provider : ทำหน้าที่เป็น Server ให้บริการเว็บเซอร์วิส คอมพิวเตอร์ โดยที่คอมพิวเตอร์ที่จะให้บริการ จะมีการเผยแพร่ (Publish) ลักษณะการให้บริการ และวิธีการเรียกใช้ เช่น เอกสาร WSDL ไปลงทะเบียนไว้กับ Service broker

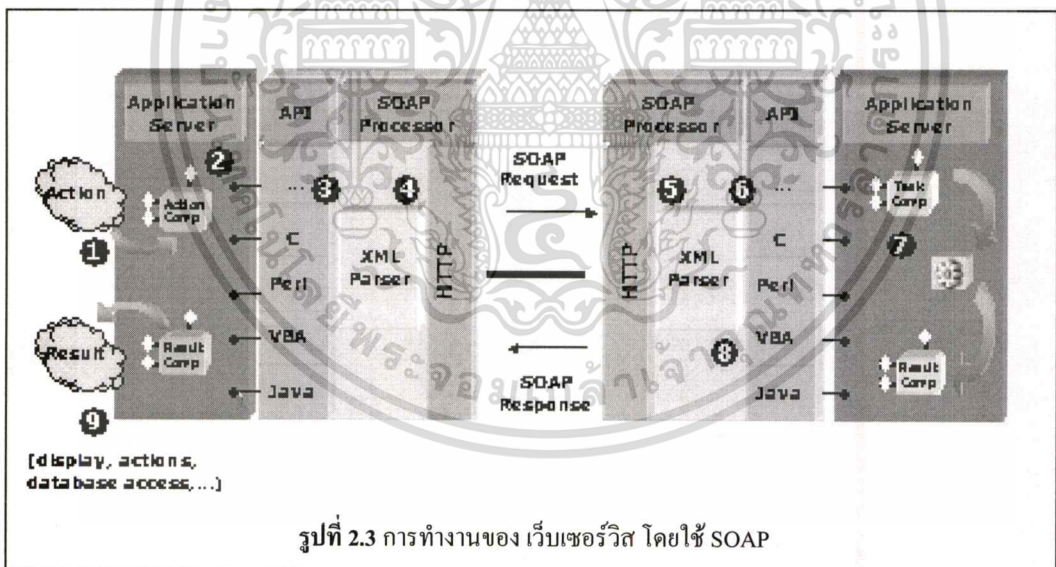
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Service requester : ทำหน้าที่เป็น Client เรียกใช้บริการจาก Service Provider ถ้าหากยังไม่รู้วิธีการเรียกใช้ หรือลักษณะการให้บริการ ก็สามารถทำการค้นหา (Find) ได้จาก Service broker
- Service broker : ทำหน้าที่เป็นตัวแทน หรือเครื่องมือในการค้นหาเว็บเซอร์วิสที่มีการลงทะเบียนไว้เพื่อให้บริการ ซึ่งเป็นลักษณะการทำงานของ UDDI

ซึ่งการสื่อสารทั้งหมดภายในส่วนต่างๆ ไม่ว่าจะเป็นการ Bind, Find หรือ Publish จะใช้โปรโตคอล SOAP ซึ่งเป็นมาตรฐานการสื่อสารร่วมกัน

### 2.2.3 การทำงานของเว็บเซอร์วิสด้วย SOAP

การทำงานของเว็บเซอร์วิสที่อาศัย SOAP เป็นข้อกำหนดหรือโปรโตคอลหลักในการแลกเปลี่ยนข้อมูล โดยทำงานร่วมกับโปรโตคอลสื่อสารเดิมอย่าง HTTP แสดงตามรูปที่ 3 ซึ่งอธิบายรายละเอียดการทำงานได้ดังนี้



- (1) เมื่อมีการเรียกใช้ Application ที่เป็นไปตามข้อกำหนดของ SOAP Message คำสั่งต่างๆ จะถูกตีความ แล้วสร้าง Action บน Application Server
- (2) ดำเนินการประมวลผลคำสั่งเหล่านั้น ซึ่งผลของการดำเนินการคำสั่งนั้นจะปรากฏอยู่ใน Application Interface ตามรูปแบบของ Application ที่พัฒนาขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- (3) Message หรือข้อมูลที่ต้องการส่งไปประมวลผลทั้งหมด จะถูกแปลให้อยู่ในรูปเอกสาร XML ในลักษณะที่เป็น SOAP message แล้วส่งไปยัง Service Provider
- (4) ตัวแปรเอกสาร XML (XML Parser) จะทำการตรวจสอบไวยากรณ์ของเอกสาร XML แล้วส่ง SOAP Messages นั้นไปบน HTTP
- (5) เมื่อ SOAP Messages มาถึงยัง Server ของฝั่งรับ XML Parser จะทำการตรวจสอบความถูกต้องของ Message โดยพิจารณาจาก Header ของ HTTP และ XML แล้วดำเนินการต่อไป หรือ ยกเลิกการทำงาน
- (6) จากนั้น Message จะถูกส่งไปยัง Application Server ที่ถูกต้องตรงกับการร้องขอ เมื่อ SOAP Application ได้รับความ SOAP Message มาแล้วจะทำการดำเนินการตามลำดับดังนี้ ขั้นแรก จะทำการตรวจสอบ SOAP Message ทุกส่วน โดยพิสูจน์ว่า SOAP Message นั้นมีลักษณะตรงกันกับ Application ที่ให้บริการ ขั้นต่อมาจะทำการตรวจสอบ Application ว่าสามารถทำงานร่วมกับ SOAP Message ตามที่ได้ร้องขอมาได้หรือไม่ จากนั้นจึงเริ่มประมวลผล หากไม่เป็นดังที่กล่าวมา Application จะทำการยกเลิก SOAP Message นั้น และสิ้นสุดท้าย ถ้า Application ที่ทำการประมวลผลนี้ ไม่สามารถตอบคำถามตามที่ร้องขอได้ทั้งหมด กล่าวคือ อาจมีการระบุให้ส่ง Message ไปประมวลผลยัง Application อื่นต่อไป จะต้องมีการนำส่วนที่ทำการได้ตรวจสอบและประมวลผลไปแล้วออกก่อนที่จะทำการส่งต่อไป
- (7) แล้วโปรแกรมจะดำเนินการไปตามการร้องขอแล้ว สร้าง ผลลัพธ์ขึ้น
- (8) การส่งข้อมูลกลับจะเป็นไปในทางเดียวกัน โดยอาศัย โพรโทคอล HTTP
- (9) ผลลัพธ์ที่ได้จากการดำเนินการทั้งหมดจะถูกแสดงในรูปแบบที่เหมาะสมตรงกับการร้องขอแต่ละประเภท ตามที่ได้ร้องขอออกไป

## บทที่ 3

### การออกแบบเว็บเซอร์วิสเพื่อการคำนวณคุณค่าทางโภชนาการของอาหาร

การคำนวณคุณค่าทางโภชนาการของอาหารในลักษณะของเว็บเซอร์วิสนั้น ต้องแบ่งการทำงานออกเป็น 2 ฝั่ง คือฝั่ง Client ซึ่งเป็นผู้เรียกใช้บริการ และฝั่ง Server ซึ่งเป็นผู้ให้บริการ โดยที่ทั้ง 2 ฝั่งนั้น ไม่จำเป็นต้องถูกพัฒนาขึ้นด้วยภาษาในการเขียนโปรแกรมเดียวกัน หรืออยู่ในระบบปฏิบัติการรูปแบบเดียวกัน และสามารถสรุปการทำงานได้ดังนี้

#### 3.1 การออกแบบเชิง Client/Server

##### 3.1.1 ฝั่งผู้ให้บริการ (Server)

ผู้ที่จะให้บริการต้องทำการสร้าง Component ซึ่งจะเป็นตัวคอยให้บริการ การเรียกใช้ Method หรือ Function การคำนวณคุณค่าทางโภชนาการของอาหาร จากนั้นทำการประกาศให้ Component นั้น มีลักษณะเป็นเว็บเซอร์วิส โดยมีเอกสารที่ใช้บอกคุณลักษณะของเว็บเซอร์วิสต่างๆ เช่น WSDL เป็นต้น เพื่อให้ผู้ที่ต้องการใช้บริการเว็บเซอร์วิสนี้ รู้ได้ว่าจะต้องเรียกใช้เว็บเซอร์วิสนี้ อย่างไร

ลักษณะการทำงานในการให้บริการเว็บเซอร์วิสเพื่อการคำนวณคุณค่าทางโภชนาการจะเป็นดังนี้

- การร้องขอที่ส่งมายัง Server นั้น มีลักษณะเป็น SOAP Message เอกสาร XML ซึ่ง XML Parser จะตีความเอกสารนั้น เมื่อรู้ว่าเป็นงานทำงานของ SOAP เพื่อเรียกใช้เว็บเซอร์วิส จึงเรียกการทำงานของ SOAP ขึ้นมาดำเนินการต่อไป
- เมื่อมีการเรียกใช้บริการ SOAP Listener ซึ่งเป็นเครื่องมือที่มีไว้เพื่อคอยฟังการเรียกใช้งานเว็บเซอร์วิส จะทำการเรียกการทำงานของ Component ที่เป็นเว็บเซอร์วิสนั้นทำการประมวลผล โดยทำการส่งค่า Parameters ที่เป็นส่วนผสมของอาหาร ไปยัง Web Service Component
- โปรแกรมจะทำการตรวจสอบว่า ข้อมูลส่วนผสมของอาหารเหล่านั้น ว่าหน่วยในการชั่งตวงของส่วนผสมต่างๆ มีหน่วยเป็นกรัมแล้วหรือยัง ถ้ายังจะถูกส่งไปคำนวณแปลงค่าและหน่วยให้มีเป็นกรัมเสียก่อน เพื่อเป็นมาตรฐานเดียวกันในการคำนวณ

- จากนั้นข้อมูลส่วนผสมแต่ละอย่าง จะถูกปรับให้มีค่าเป็นจำนวนเทียบต่อ 100 กรัม แล้วนำไปเรียกค่าจากตารางแสดงคุณค่าทางโภชนาการ ซึ่งถ้าข้อมูลส่วนผสมที่ไม่มีในตาราง จะมีการบันทึกเก็บไว้เพื่อบอกกับผู้ใช้บริการต่อไปว่า ไม่ได้คำนวณส่วนผสมใดบ้าง และอาจจะทำการส่งข้อความแจ้งไปยังผู้ดูแลระบบถึงส่วนผสมของอาหารที่ขาดไป เพื่อการหามาเพิ่มเติมในอนาคต
- ค่าที่ได้จากส่วนผสมทั้งหมดจะถูกคำนวณรวม ปริมาณสารอาหารแต่ละชนิด
- ปริมาณสารอาหารแต่ละชนิด จะถูกนำไปคำนวณเทียบร้อยละกับปริมาณความต้องการ หรือระดับความเหมาะสมในการบริโภคของร่างกาย
- นำค่าปริมาณ และจำนวนร้อยละต่อปริมาณความต้องการ ของสารอาหารแต่ละชนิด ไปสร้างเป็น SOAP Message แล้วส่งกลับไปยัง Client
- สำหรับการเพิ่ม หรือแก้ไข ข้อมูลทางโภชนาการ หรือปริมาณสารอาหารที่ร่างกายต้องการ นั้น จะต้องมีการตรวจสอบสิทธิ์ของผู้เข้าใช้ก่อน ว่าเป็นผู้ได้รับสิทธิ์ในการเพิ่มหรือแก้ไข ข้อมูล จากผู้ดูแลระบบเท่านั้น

### 3.1.2 ผู้เรียกใช้บริการ (Client)

Client ที่ต้องการใช้บริการคำนวณคุณค่าทางโภชนาการของอาหาร ต้องทราบข้อมูลเบื้องต้นในการเรียกใช้เว็บเซอร์วิสก่อน ซึ่งมี URI สำหรับเรียกใช้บริการ, ชื่อของ method ที่จะเรียก, และรูปแบบการส่งค่า Parameters ต่างๆ ที่จำเป็นต้องใช้ในการคำนวณ ซึ่งก็คือส่วนผสมในการประกอบอาหารนั่นเอง จากนั้นก็ทำการสร้าง Application ในการเรียกใช้เว็บเซอร์วิส โดยอ้างอิงกับข้อมูลต่างๆ ดังที่ได้กล่าวไปแล้ว

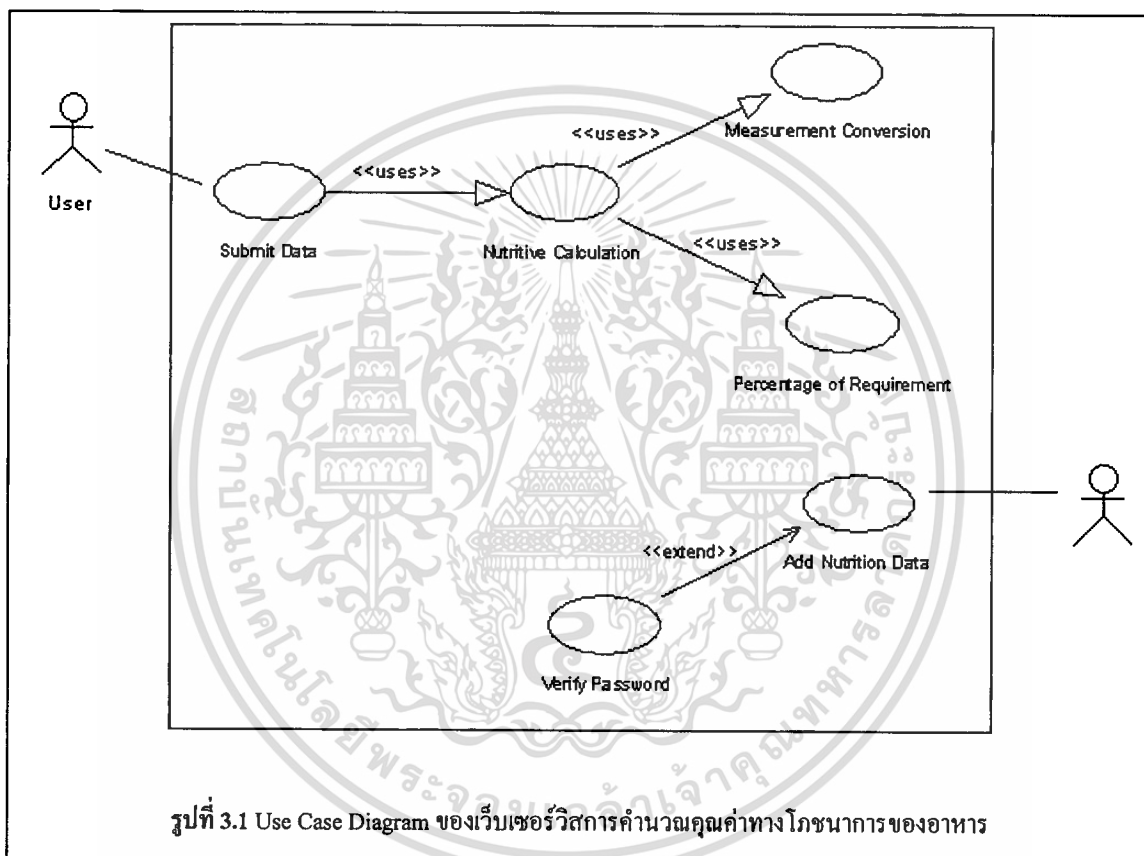
เมื่อมี Application ในการเรียกใช้เว็บเซอร์วิสการคำนวณนี้แล้ว ลักษณะการทำงานจะเป็นดังนี้

- เมื่อ User ต้องการคำนวณคุณค่าทางโภชนาการของอาหาร จากส่วนผสมของอาหาร ก็จะทำการป้อนข้อมูลส่วนผสมเหล่านั้นเข้าไปใน Application เพื่อทำการคำนวณ
- จากนั้น Application ก็จะทำการ รวบรวมข้อมูลที่ต้องใช้ในการคำนวณทั้งหมด แล้วสร้าง SOAP message ซึ่งเป็นเอกสาร XML ส่งไปประมวลผล ยัง Server ที่ให้บริการเว็บเซอร์วิสนั้น
- เมื่อ Server ตอบกลับมาแล้วจะทำการนำค่าที่ได้มาตีความตามไวยากรณ์ของเอกสาร XML ที่มีลักษณะเป็น SOAP Message แล้วแสดงผลตามรูปแบบของ Client Application

### 3.2 การออกแบบเชิงวัตถุโดยใช้ UML

สำหรับการออกแบบเว็บเซอร์วิสเพื่อการคำนวณคุณค่าทางโภชนาการของอาหาร จะใช้การออกแบบในเชิงวัตถุ โดยใช้ UML เป็นตัวแบบอธิบายการทำงาน ซึ่ง Diagram ที่นำมาใช้ในการออกแบบนั้น ได้แก่ Use Case Diagram, Class Diagram และ Sequence Diagram

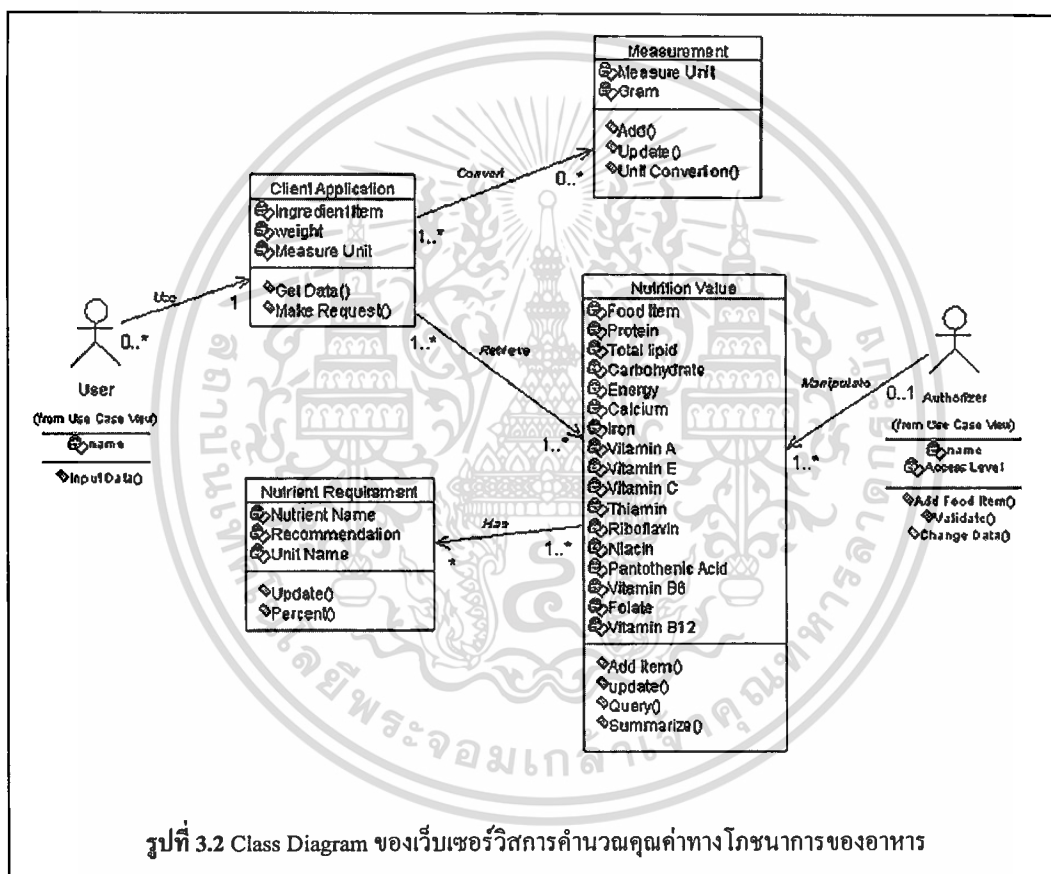
#### 3.2.1 Use Case Diagram



เป็นสิ่งที่ใช้อธิบายการทำงานโดยรวมในลักษณะ Conceptual view สำหรับเว็บเซอร์วิสการคำนวณคุณค่าทางโภชนาการของอาหารจะมี Use Case หรือการทำงานที่เกี่ยวข้องจำนวน 6 Use Case โดยแบ่งเป็น ส่วนที่มีการ activate การทำงานโดย User ซึ่งได้แก่ การรับข้อมูล (Submit Data) ซึ่งจะเป็นตัวรับข้อมูลมาจาก User จากนั้นมีการเรียกใช้ไปยัง การคำนวณคุณค่าทางโภชนาการ (Nutritive Calculation) จากนั้นการคำนวณคุณค่าทางโภชนาการ จะมีการเรียกใช้การทำงานอีก 2 Use Case ขึ้นมาทำงาน นั่นคือการเปลี่ยนแปลงมาตราการชั่ง ตวง (Measurement Conversion) และ

การคำนวณเทียบร้อยละของความต้องการของร่างกายในแต่ละวัน (Percentage of Requirement) โดยที่การทำงานที่เป็นส่วนที่เชื่อมโยงกันนั้นเปรียบเสมือนการเรียกใช้งานระหว่างกันของเว็บเซอร์วิส และอีกส่วนหนึ่งคือการได้รับ activate การทำงานโดย Authorizer ได้แก่ การเพิ่มข้อมูลคุณค่าทางโภชนาการให้กับอาหารหรือส่วนประกอบอาหารใหม่ๆ (Add Nutrition Data) ซึ่งต้องมี การตรวจสอบ Password หรือสิทธิ์ในการเข้าใช้ก่อน (Verify Password)

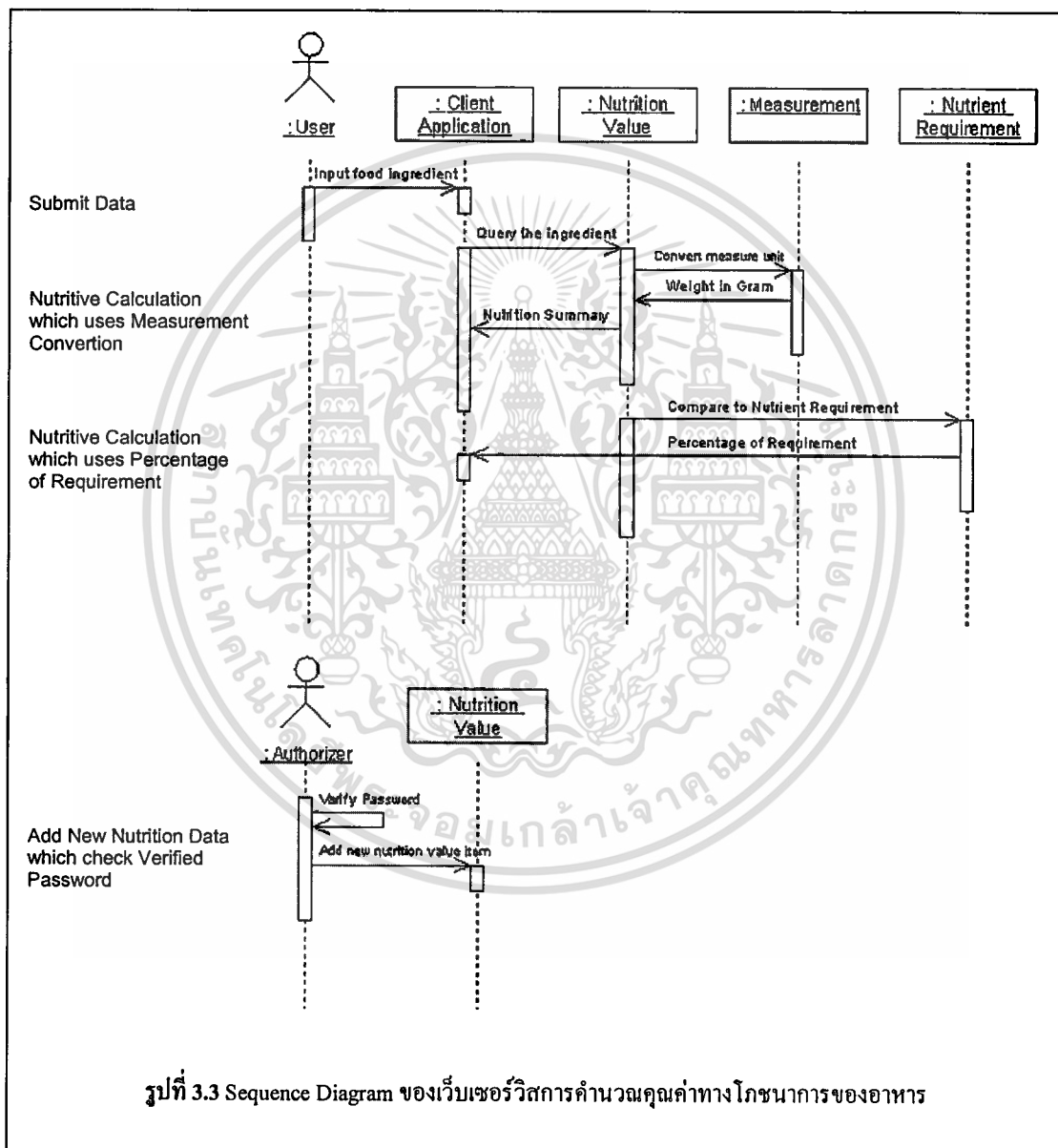
### 3.2.2 Class Diagram



เป็นแผนภาพที่แสดงกลุ่มของสิ่งที่กำลังสนใจหรือศึกษาอยู่ รวมทั้งความสัมพันธ์ระหว่าง Class เหล่านั้น สำหรับเว็บเซอร์วิสการคำนวณคุณค่าทางโภชนาการของอาหารจะมี Class ที่เกี่ยวข้องกันอยู่ 6 Class ได้แก่ User ซึ่งเป็นผู้ที่มาใช้งาน Client Application เป็นกลุ่มของสิ่งที่ User ส่งเข้ามาในระบบนั่นก็คือ ข้อมูลอาหารหรือส่วนผสมของอาหารต่างๆ ที่ต้องการคำนวณหาคุณค่าทางโภชนาการ Nutrition Value เป็นกลุ่มของสารอาหารต่างๆที่มีอยู่ในอาหารหรือส่วนผสมของ

อาหาร Measurement เป็นกลุ่มของมาตราซึ่ง ตวงอาหาร Nutrient Requirement เป็นกลุ่มของ ปริมาณความต้องการในสารอาหารต่างๆ ของร่างกายในแต่ละวัน และสุดท้ายคือ Authorizer ซึ่ง เป็นผู้ที่ยกยอดูแลข้อมูลโภชนาการ

### 3.2.3 Sequence Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นแผนภาพจำลองการทำงานของเว็บเซอร์วิสในลักษณะ Logical View ขยายความการทำงานของ Use Case Diagram เพื่อให้เห็นความสัมพันธ์ของ Class ภายใน โดยเส้นที่ลากลงมาจากร Class นั้นจะแสดงลำดับเวลา ก่อนหลัง และเส้นที่ลากเชื่อมกันระหว่าง Class นั้นจะแสดงลำดับกิจกรรมการทำงานระหว่าง Class หนึ่งไปยังอีก Class หนึ่ง

### 3.3 วิเคราะห์ Input /Process/ Output

รายละเอียดของการพัฒนาเว็บเซอร์วิสเพื่อการคำนวณคุณค่าทางโภชนาการของอาหารจะแบ่งการพัฒนาออกเป็น 2 ส่วนคือส่วนที่เป็นการใช้เว็บเซอร์วิส และส่วนที่เป็นตัวเซอร์วิสคอมพิวเตอร์ที่จะให้บริการ

#### 3.3.1 Interface Applications

##### (1) Client Application

Service : เรียกใช้บริการเว็บเซอร์วิสการคำนวณคุณค่าทางโภชนาการของอาหาร

Input :

- ชื่อของอาหาร
- รายการอาหาร หรือรายการเครื่องปรุงของอาหาร ที่ต้องการคำนวณ
- น้ำหนักของอาหารนั้น
- หน่วยในการชั่ง ตวงของอาหารนั้น

Process:

- รับข้อมูลป้อนเข้าจากผู้ใช้
- เตรียมข้อมูลตามรูปแบบการเรียกใช้บริการ
- ส่งข้อมูลไปคำนวณยัง เว็บเซอร์วิสการคำนวณคุณค่าทางโภชนาการของอาหาร
- รับข้อมูลตอบกลับ แล้วตีความ
- แสดงผลตามรูปแบบที่ต้องการ

Output :

- ชื่อของอาหาร
- จำนวนพลังงานและสารอาหารต่างๆ ที่จะได้รับจากอาหาร (Protein, Total lipid(fat), Carbohydrate , Energy, Calcium, Iron, Vitamin A, E, C, B1, B2, B3, B5, B6, Folate, B12)
- ร้อยละของสารอาหารต่างๆ ที่ร่างกายต้องการในแต่ละวันที่มีในอาหารนั้น

## (2) Admin Application

Service : ตรวจสอบสิทธิการเข้าใช้

Input :

- Username
- Password

Process:

- รับข้อมูล Username และ Password ที่ป้อนเข้ามาเก็บไว้เพื่อรอการผนวกไปกับ SOAP message ที่จะเรียกเว็บเซอร์วิส ที่มีความจำเป็นต้องใช้รหัสผ่านในขั้นขั้นสิทธิในการเข้าใช้
- ส่งข้อมูลไปตรวจสอบสิทธิการเข้าใช้และรอการตอบกลับ

Output :

- สิทธิในการเข้าแก้ไขข้อมูลต่างๆ

Service : อัปเดตข้อมูลคุณค่าทางโภชนาการของอาหาร

Input :

- ชื่อของอาหารหรือเครื่องปรุงอาหาร
- จำนวนของพลังงานและสารอาหารต่างๆ ต่ออาหาร 100 กรัม (Protein, Total lipid(fat), Carbohydrate , Energy, Calcium, Iron, Vitamin A, E, C, B1, B2, B3, B5, B6, Folate, B12)

Process:

- ป้อนชื่ออาหารหรือเครื่องปรุงอาหาร เพื่อตรวจสอบว่าเป็นรายการใหม่หรือไม่
- เตรียมข้อมูลสิทธิการเข้าใช้ และข้อมูลคุณค่าทางโภชนาการทั้งหมดที่ต้องการบันทึก ลงฐานข้อมูล ส่งไปยังเว็บเซอร์วิสการอัปเดตหรือเพิ่มข้อมูลรายการใหม่
- แสดงผลการปรับปรุงข้อมูลต่อผู้ใช้

Output :

- ผลการปรับปรุงข้อมูล

### 3.3.2 Web Services Components

#### (1) Nutritive Value Calculation Component

Service : คำนวณคุณค่าทางโภชนาการของอาหาร

**Input :**

- รายการอาหารหรือเครื่องปรุงอาหาร
- น้ำหนักของอาหารนั้น
- หน่วยที่ใช้ชั่งตวง

**Process:**

- รับข้อมูลการเรียกใช้บริการจากการทำงานของ SOAP แล้วตีความ
- ตรวจสอบว่ามีรายการอาหารใดบ้างที่ไม่ได้มีหน่วยในการชั่งตวงเป็นกรัม
- เตรียมรายการอาหารที่ยังไม่มีหน่วยเป็นกรัมนั้นตามรูปแบบการเรียกใช้บริการเว็บเซอร์วิสการแปลงหน่วยการชั่งตวง(2) แล้วส่งรายการอาหารเหล่านั้นไปทำการแปลงหน่วยในการชั่งตวงให้เป็นกรัมก่อน
- เรียกข้อมูลจำนวนพลังงานและสารอาหารต่อ 100 กรัมของรายการอาหารเหล่านั้น จากข้อมูลคุณค่าทางโภชนาการของอาหาร
- คำนวณเปรียบเทียบอัตราส่วนของพลังงานและสารอาหารต่างๆ ตามน้ำหนักจริงของอาหาร
- ทำการรวมจำนวนพลังงานและสารอาหารต่างๆ จากทุกรายการอาหาร
- เตรียมข้อมูลจำนวนพลังงานและสารอาหารตามรูปแบบการเรียกใช้บริการเว็บเซอร์วิสการคำนวณร้อยละของความต้องการ ในพลังงานและสารอาหารต่างๆ ในแต่ละวัน (3) แล้วทำการส่งไปคำนวณ
- เตรียมข้อมูลที่ได้อัตราส่วนแล้วทั้งหมด แล้วส่งกลับไปยังผู้เรียกใช้บริการ

**Output :**

- จำนวนพลังงานและสารอาหารต่างๆ ที่จะได้รับจากอาหาร
- ร้อยละของพลังงานและสารอาหารต่างๆ ที่ร่างกายต้องการในแต่ละวันที่มีในอาหารนั้นรายการ

**(2) Measurement Conversion**

**Service :** แปลงหน่วยน้ำหนักของอาหารเป็นกรัม

**Input :**

- รายการอาหารหรือเครื่องปรุง
- น้ำหนักของอาหารหรือเครื่องปรุงนั้นๆ

- หน่วยในการชั่งตวงอาหารหรือเครื่องปรุงนั้นๆ

Process:

- รับข้อมูลการเรียกใช้บริการจากการทำงานของ SOAP แล้วตีความ
- ตรวจสอบว่ามีรายการอาหารใดบ้างที่ไม่ได้มีหน่วยในการชั่งตวงเป็นกรัม
- ตรวจสอบว่ารายการที่ไม่ได้มีหน่วยในการชั่งตวงเป็นกรัม นั้น มีหน่วยในการชั่งตวงเป็นอะไร แล้วจึงเรียกข้อมูลอัตราส่วนการชั่งตวงของหน่วยนั้นขึ้นมา
- คำนวณตามอัตราส่วน แปลงหน่วยให้เป็นกรัม
- เตรียมข้อมูลที่ได้อัตราส่วนแล้วทั้งหมด แล้วส่งกลับไปยังผู้เรียกใช้บริการ

Output :

- รายการอาหารหรือเครื่องปรุง
- น้ำหนักของอาหารหรือเครื่องปรุงนั้นๆ ที่เป็นกรัม
- หน่วยกรัม

### (3) Percentage of Requirement

Service : คำนวณร้อยละของสารอาหารที่ร่างกายต้องการในแต่ละวัน ในอาหารที่คำนวณ

Input :

- รายการพลังงานหรือสารอาหารต่างๆ
- จำนวนของพลังงานหรือสารอาหารที่มี

Process:

- รับข้อมูลการเรียกใช้บริการจากการทำงานของ SOAP แล้วตีความ
- ตรวจสอบข้อมูลรายการพลังงานและสารอาหารที่รับมา แล้วทำการเรียกจำนวนข้อมูลมาตรฐานความต้องการพลังงานและสารอาหารต่างๆ เหล่านั้น
- คำนวณเปรียบเทียบอัตราส่วนร้อยละของข้อมูลกับมาตรฐาน
- เตรียมข้อมูลที่ได้อัตราส่วนแล้วทั้งหมด แล้วส่งกลับไปยังผู้เรียกใช้บริการ

Output :

- รายการพลังงานและสารอาหาร
- ร้อยละของพลังงานและสารอาหารที่ได้จากการคำนวณ

#### (4) Nutrient Data Admin

Service : เป็นตัวแทนฐานข้อมูลในการปรับปรุงหรือเพิ่มข้อมูลรายการอาหารหรือในฐานข้อมูลคุณค่าทางโภชนาการของอาหาร

Input :

- ชื่อของอาหารหรือเครื่องปรุงที่ต้องการบันทึก
- จำนวนพลังงานและสารอาหารต่างๆ ของอาหารนั้นๆ ต่อ 100 กรัม

Process :

- รับข้อมูล SOAP message แล้วแยกข้อมูลเกี่ยวกับการจัดการสิทธิ์การบันทึกข้อมูล ไปตรวจสอบว่ามีสิทธิ์การแก้ไข ปรับปรุงข้อมูลหรือไม่
- หลังจากที่ได้รับสิทธิ์การแก้ไขแล้ว ตรวจสอบข้อมูลที่ได้รับเข้ามา ว่าเป็นรายการใหม่หรือไม่ ถ้าเป็นรายการใหม่จะทำการเพิ่มข้อมูลรายการใหม่ในฐานข้อมูล แต่ถ้ามีรายการนั้นอยู่แล้วก็จะปรับปรุงรายการ
- บันทึกข้อมูลทางโภชนาการต่างๆ และข้อมูลเกี่ยวกับชื่อของรายการนั้นลงฐานข้อมูล

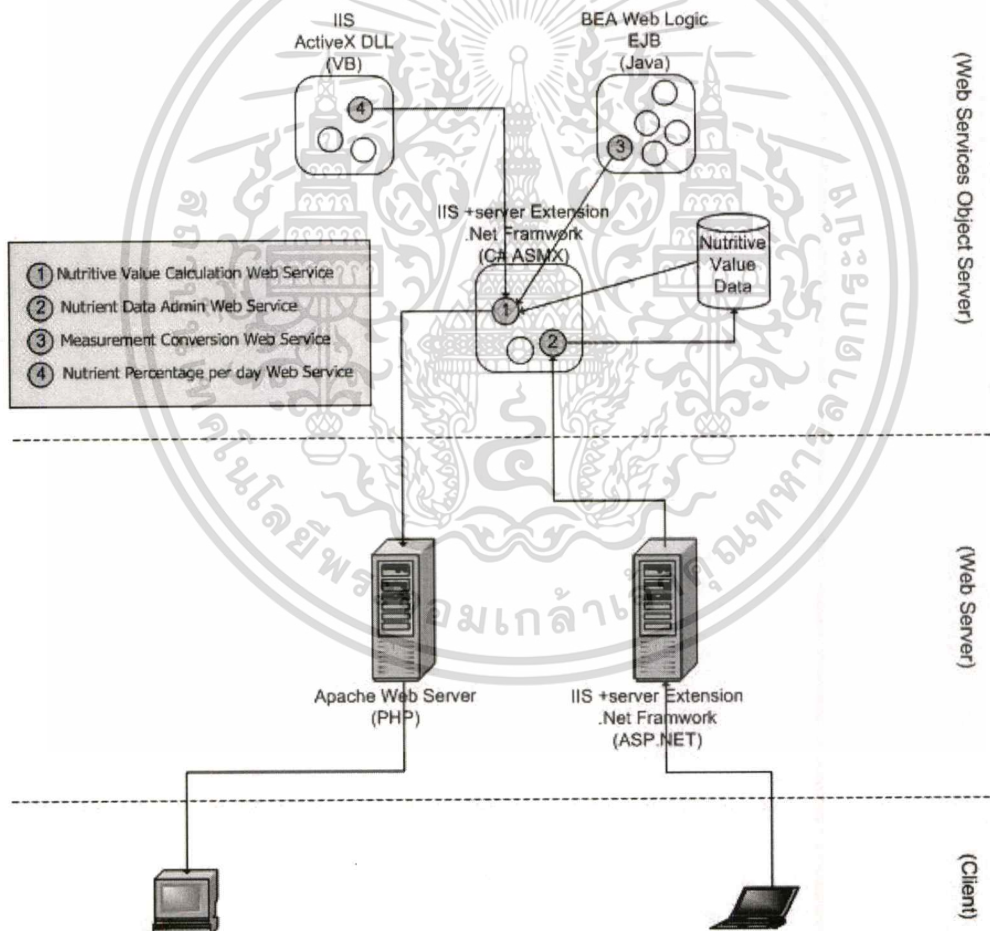
Output :

- ส่งผลการปรับปรุงข้อมูลคืนไปยัง Application ที่เรียกใช้

## บทที่ 4

### การพัฒนาเว็บเซอร์วิสเพื่อการคำนวณคุณค่าทางโภชนาการ

หลังจากได้ทำการออกแบบเรียบร้อยแล้ว เนื้อหาในบทนี้จะเป็นการนำสิ่งที่ได้ทำการออกแบบแล้วมาพัฒนาขึ้นเป็นแอปพลิเคชันในรูปแบบเว็บเซอร์วิส เพื่อทดสอบการทำงานจริง ซึ่งสิ่งที่ได้จากการออกแบบทั้งหมดนั้น เมื่อนำมาผนวกกับฮาร์ดแวร์ และสิ่งแวดล้อมที่จะใช้ในการพัฒนา สามารถสรุปเป็นสถาปัตยกรรมของระบบโดยรวมได้ดังนี้



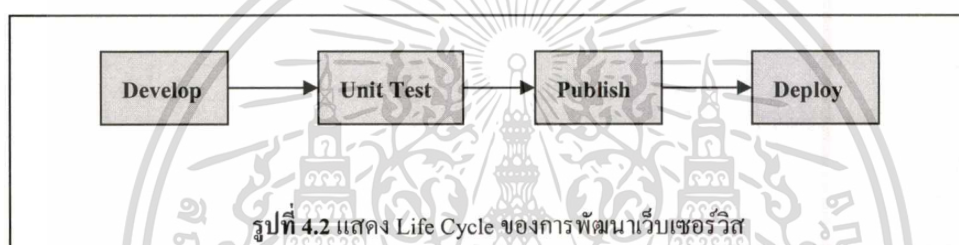
รูปที่ 4.1 แสดงสถาปัตยกรรมที่จะใช้ในการพัฒนาเว็บเซอร์วิสเพื่อการคำนวณคุณค่าทางโภชนาการของอาหาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดของการพัฒนาเว็บเซอร์วิสเพื่อการคำนวณคุณค่าทางโภชนาการของอาหารตามที่ได้ทำการศึกษาและออกแบบมานั้น แบ่งการพัฒนาออกเป็น 2 ส่วน โดยที่ส่วนแรกเป็นส่วนกระบวนการทำงานภายใน เป็นการพัฒนา เว็บเซอร์วิส คอมโพเนนต์ที่จะให้บริการ และส่วนที่สองจะเป็นส่วนในการเชื่อมต่อเรียกใช้งานเว็บเซอร์วิสซึ่งมีรายละเอียดอยู่ในบทต่อไป

### การพัฒนาเว็บเซอร์วิส คอมโพเนนต์ (Web Services Components)

ขั้นตอนในการพัฒนาเว็บเซอร์วิสโดยทั่วไปนั้น ส่วนใหญ่จะแบ่งออกเป็น 4 ขั้นตอนด้วยกัน ได้แก่ การสร้างเว็บเซอร์วิส การทดสอบการทำงาน การประกาศขึ้นเป็นเว็บเซอร์วิส และการเตรียมพร้อมให้บริการ ซึ่งแสดงได้ดังรูป



จากการออกแบบที่ผ่านมาจะเห็นว่า มีการออกแบบให้มีคอมโพเนนต์ที่ทำงานเป็นเว็บเซอร์วิสทั้งหมด 4 คอมโพเนนต์ด้วยกัน เพื่อความยืดหยุ่นในการพัฒนาและทดสอบการทำงานร่วมกันระหว่างเว็บเซอร์วิสในหลายๆ รูปแบบ แต่เพื่อความสะดวกในการเชื่อมต่อกับฝั่ง Client ที่ต้องการเรียกใช้ จึงให้สามารถเรียกการทำงานทั้งหมด ผ่านเว็บเซอร์วิสเพียงคอมโพเนนต์เดียว และเว็บเซอร์วิสหลักในการเชื่อมต่อนั้นจะทำการคำนวณให้ โดยที่ Client เลือกใช้ Method เพื่อแสดงผลสารอาหารในแต่ละตัวได้ตามต้องการ

โดยปรกตินั้น การทำงานกับเว็บเซอร์วิสนั้นจะเป็นไปในลักษณะ เรียกใช้บริการไปยังเว็บเซอร์วิสแล้ว คอมโพเนนต์จะทำการประมวลผล แล้วตอบกลับ ในลักษณะนี้ เรียกว่า Synchronous Web Services แต่ถ้าเว็บเซอร์วิส ไม่สามารถประมวลผลตอบกลับได้ทันที อาจต้องมีการส่งต่อการทำงานให้กับเว็บเซอร์วิสอื่น ในลักษณะนี้เรียก Asynchronous Web Services ซึ่งการทำงานในลักษณะนี้ ต้องอาศัยความสามารถของเครื่องมือที่ใช้ในการพัฒนาในการสร้าง Thread เพื่อรองรับการร้องขอไปยังอีกการทำงานหนึ่งในขณะที่การทำงานหลักทำงานอยู่ ซึ่งเครื่องมือที่มีความสามารถในการสร้าง Asynchronous method นี้ได้มีประสิทธิภาพ เช่น Visual Studio .Net เป็นต้น และดังที่ได้กล่าวมาแล้วว่า การพัฒนาเว็บเซอร์วิสนี้ ต้องการศึกษากonstruktion การทำงาน และ

การเรียกใช้เว็บเซอร์วิส ในสภาวะแวดล้อมที่ต่างกัน ดังนั้น จึงเลือกสภาวะแวดล้อมการพัฒนา ที่ใช้สร้างเว็บเซอร์วิส คอมโพเนนต์ที่แตกต่างกัน โดยให้คอมโพเนนต์ที่พัฒนาจาก ภาษา C# ใน Visual Studio .Net เป็นเว็บเซอร์วิสหลักในการเรียกใช้เว็บเซอร์วิสอื่น ที่พัฒนาจาก ActiveX DLL ด้วย Visual Basic 6 และ EJB(Enterprise JavaBean) ที่มีพื้นฐานจาก J2EE

#### 4.1 Nutritive Value Calculation

เป็นเว็บเซอร์วิสหลักที่ Client จะเชื่อมต่อด้วยทำหน้าที่รับข้อมูลวัตถุดิบที่ใช้ปรุงอาหาร หรืออาหารที่ต้องการคำนวณ น้ำหนัก และหน่วยที่ใช้ชั่งตวง ที่ส่งมาจาก Client พิจารณาว่าถ้าหน่วยของวัตถุดิบยังไม่เป็นกรัม ให้ส่งไปยังเว็บเซอร์วิสที่ใช้เพื่อการแปลงมาตราการชั่งตวง จากนั้นดึงข้อมูลวัตถุดิบที่ต้องการคำนวณขึ้นมาจากฐานข้อมูล แล้วคำนวณเทียบสัดส่วนคุณค่าทางโภชนาการ ถ้า Client มีการร้องขอให้มีการคำนวณเทียบร้อยละกับความต้องการบริโภคในแต่ละวัน ก็จะส่งไปยังเว็บเซอร์วิสที่ใช้คำนวณนั้น แล้วตอบกลับให้กับ Client นอกจากนี้ภายใน Solution นี้ ยังมีเว็บเซอร์วิสอีกส่วนหนึ่งเพื่อใช้จัดการกับฐานข้อมูล เพื่อให้ การทำงานของ Data Administrator ในการปรับปรุงข้อมูลคุณค่าทางโภชนาการของอาหารได้สะดวกขึ้นด้วย

##### เครื่องมือที่ใช้พัฒนา

- Microsoft Visual Studio .Net
- Microsoft Internet Information Services 5.1
- Microsoft Access

##### ขั้นตอนในการพัฒนา

1. รวบรวมข้อมูลเพื่อทำการสร้างเว็บเซอร์วิสคำนวณคุณค่าทางโภชนาการ แล้วทำการสร้างตาราง ฐานข้อมูลเพื่อเก็บรายการอาหารและวัตถุดิบที่ใช้ประกอบอาหารต่างๆ รวมถึง ปริมาณสารอาหารที่มีในรายการอาหารนั้นๆ
2. เลือกการสร้างโปรเจคใหม่ ด้วย Visual C# แล้วเลือกรูปแบบการพัฒนาเป็นแบบ ASP.NET Web Service เลือก Web Server ที่ต้องการเก็บเว็บเซอร์วิส
3. เพิ่มเว็บเซอร์วิสที่เกี่ยวข้องในการประมวลผลเข้ามาในโปรเจค โดยให้ทำการตรวจสอบดูว่า เว็บเซอร์วิสที่ต้องการเรียกใช้นั้นคือ เว็บเซอร์วิสการแปลงหน่วยมาตราการชั่งตวง และเว็บเซอร์วิสการคำนวณร้อยละของสารอาหารที่ต้องการในแต่ละ

ละวัน เก็บเอกสาร WSDLไว้ที่ใด แล้วให้ทำการ Add Web Reference เข้ามาในโปรเจก

4. จากนั้นเขียนโปรแกรมการทำงานในการคำนวณคุณค่าทางโภชนาการ โดยตรวจสอบว่า ข้อมูลอาหารที่ถูกส่งเข้ามาประมวลผลนั้น มีหน่วยในการชั่งตวง เป็นกรัมหรือไม่ ถ้ายัง ให้ส่ง ไปแปลงที่เว็บเซอร์วิสแปลงหน่วยการชั่งตวง หลังจากนั้นนำรายการอาหารหรือวัตถุดิบที่รับเข้ามา ดึงค่าสารอาหารของรายการอาหารเหล่านั้นมาจากฐานข้อมูล แล้วคำนวณตามสัดส่วนน้ำหนักของอาหาร สุดท้ายจึงนำสารอาหารที่ได้ทั้งหมด ส่งไปยังเว็บเซอร์วิสที่ใช้คำนวณเปรียบเทียบร้อยละที่แนะนำให้บริโภคในแต่ละวัน และพัฒนาเว็บเซอร์วิสในส่วนที่จัดการกับข้อมูลการปรับปรุงรายการจาก Client ที่เป็น Data Admin
5. เมื่อเขียนโปรแกรมเสร็จแล้ว จะทำการทดสอบการทำงานซึ่งสามารถ Start Service ได้เลย เนื่องจากขณะที่สร้างนั้น Visual Studio . Net จะทำการสร้าง Listener และ WSDL ให้โดยอัตโนมัติ

#### รายละเอียดในการพัฒนา

##### 4.1.1 การพัฒนาฐานข้อมูลคุณค่าทางโภชนาการของอาหาร

ในส่วนของฐานข้อมูลคุณค่าทางโภชนาการของอาหารนี้ เลือกใช้ Microsoft Access เป็น DBMS เนื่องจาก Microsoft Access มีความสามารถเหมาะสมกับงานที่ไม่มีความซับซ้อนมากนัก ซึ่งรายละเอียดในการพัฒนาส่วนฐานข้อมูลนี้ แบ่งออกเป็น 2 ส่วนคือ ส่วนของตาราง(Tables) และแบบสอบถาม(Query) หรือ Stored Procedure

##### ส่วนที่ 1 : ตาราง(Tables)

ตารางที่เกี่ยวกับข้อมูลคุณค่าทางโภชนาการนี้ มีอยู่ด้วยกัน 2 ตาราง และตารางเพื่อใช้จัดการสิทธิ์ในการเข้าใช้ระบบอีก 1 ตาราง ซึ่งมีคุณสมบัติต่างๆ ดังนี้

1. ตาราง NutritiveValue – ใช้เก็บข้อมูลรายการอาหารหรือเครื่องส่วนผสมของอาหาร และจำนวนคุณค่าทางโภชนาการต่างๆ เทียบต่อปริมาณอาหาร 100 กรัม

	ชื่อคอลัมน์	ชนิดของข้อมูล
(PK)	ID	Auto Number(Long Integer)
	FoodItem	Text
	Protein	Number(Double)
	Fat	Number(Double)

Carbohydrate	Number(Double)
Energy	Number(Double)
Calcium	Number(Double)
Iron	Number(Double)
VitA	Number(Double)
VitE	Number(Double)
VitC	Number(Double)
VitB1	Number(Double)
VitB2	Number(Double)
VitB5	Number(Double)
VitB6	Number(Double)
VitB12	Number(Double)
Niacin	Number(Double)
Folate	Number(Double)

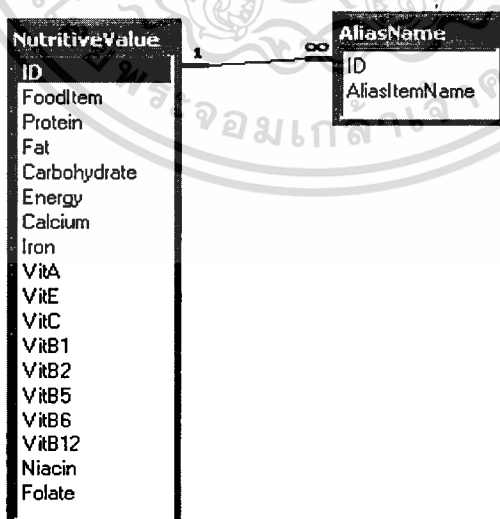
2. ตาราง AliasName – ใช้เก็บชื่อของอาหาร ในกรณีที่อาหารชนิดนั้นๆ สามารถเรียกได้หลายอย่าง

ชื่อคอลัมน์	ชนิดของข้อมูล
ID	Number(Long Integer)
AliasItemName	Text

3. ตาราง User – ใช้เก็บข้อมูล รหัสการเข้าใช้ของผู้ใช้ ที่มีสิทธิ์ แต่ละคน

ชื่อคอลัมน์	ชนิดของข้อมูล
(PK) Username	Text
Password	Text

ความสัมพันธ์ระหว่างตาราง – มีการกำหนดความสัมพันธ์ระหว่างตาราง NutritiveValue และตาราง AliasName ไว้ในลักษณะ 1:N (Cascade Update Related Fields และ Cascade Delete Related Records)



รูปที่ 4.3 แสดงความสัมพันธ์ของตาราง NutritiveValue และตาราง AliasName

## ส่วนที่ 2 : แบบสอบถาม หรือ Stored Procedure

เพื่อให้การทำงานกับฐานข้อมูลเป็นไปอย่างมีประสิทธิภาพมากยิ่งขึ้น จึงได้ทำการสร้าง Stored Procedure หรือ Query ใน Microsoft Access ไว้ดูแลการเรียกใช้ข้อมูลจากฐานข้อมูล ซึ่งสร้างไว้ทั้งหมดจำนวน 5 Query ดังนี้

1. Query : NutrientSP – มีไว้เพื่อค้นหารายการอาหารที่อยู่ในฐานข้อมูลตามที่ต้องการ ว่ามีจำนวนคุณค่าทางโภชนาการต่างๆ เป็นจำนวนเท่าใด ซึ่งมีรูปแบบภาษา SQL ดังนี้

```
SELECT NutritiveValue.*
FROM NutritiveValue
WHERE NutritiveValue.ID = (SELECT AliasName.ID
FROM AliasName
WHERE AliasName.AliasItemName = [FoodName]);
```

2. Query : getAlias – มีไว้เพื่อค้นหาชื่อเรียกของรายการอาหารที่ต้องการค้นหาว่าสามารถเรียกรายการอาหารนั้น เป็นชื่ออื่นๆ อะไรได้บ้าง ซึ่งมีรูปแบบภาษา SQL ดังนี้

```
SELECT AliasName.AliasItemName
FROM AliasName
WHERE AliasName.ID = (SELECT AliasName.ID
FROM AliasName
WHERE AliasName.AliasItemName = [FoodName]);
```

3. Query : getID – มีไว้เพื่อค้นหา ID ของรายการอาหารที่ต้องการค้นหาว่ามี ID เป็นเท่าใด เพื่อไปใช้อ้างอิงกับส่วนอื่นๆ ต่อไป ซึ่งมีรูปแบบภาษา SQL ดังนี้

```
SELECT ID
FROM NutritiveValue
WHERE FoodItem = [FoodName];
```

4. Query : countItem – มีไว้เพื่อนับจำนวนรายการอาหารที่มีอยู่ในฐานข้อมูล ซึ่งมีรูปแบบภาษา SQL ดังนี้

```
SELECT count(ID)
FROM NutritiveValue;
```

5. Query : getUser – มีไว้เพื่อค้นหา Password ของ Username ที่ต้องการค้นหา ซึ่งมีรูปแบบภาษา SQL ดังนี้

```
SELECT Password
FROM User
WHERE Username=[User];
```

#### 4.1.2 รายละเอียดการพัฒนา Nutritive Value Calculation Web Services Components

ในส่วนของเว็บเซอร์วิสที่หลักในการคำนวณคุณค่าทางโภชนาการของอาหารนี้ แบ่งออกเป็น 2 Web Services ย่อย ได้แก่ MainServices และ NutrientDataAdmin ซึ่งทั้งสองเว็บเซอร์วิสต่างก็อยู่บน .Net Platform ที่พัฒนาขึ้นจากภาษา C#

**4.1.2.1 MainServices** : เป็นเว็บเซอร์วิสหลักในการติดต่อกับส่วน Client เพื่อคำนวณคุณค่าทางโภชนาการของอาหารจากข้อมูลรายการอาหารที่รับเข้ามาจาก Client โดยที่ภายใต้การเรียกใช้ MainServices นี้ จะมี Method ที่ให้บริการอยู่เพียง Method เดียว นั่นคือ getNutrient

```
[WebMethod(Description="get nutrient values from arrays of
items and return in complex type object")]
public NutrientResult getNutrient(String [] FoodItem,float []
Value,String [] UnitName){...
```

ซึ่งจากการประกาศ WebMethod getNutrient จะเห็นว่า Method นี้จะรับข้อมูลเข้ามาประมวลผล ในลักษณะ Array ของข้อมูล 3 กลุ่ม ได้แก่ FoodItem ซึ่งเป็นรายการอาหารที่ต้องการคำนวณ ปริมาณน้ำหนักของแต่ละรายการนั้น และหน่วยที่ใช้ชั่งน้ำหนักของแต่ละรายการนั้นๆ แล้วส่งข้อมูลกลับในลักษณะ Object ของ Class NutrientResult ที่มีรายละเอียดดังนี้

```
public class NutrientResult
{
    public double Protein;
    public double Fat;
    public double Carbo;
    public double Energy;
    public double Calcium;
    public double Iron;
    public double VitA;
    public double VitE;
    public double VitC;
    public double VitB1;
    public double VitB2;
    public double VitB5;
    public double VitB6;
    public double VitB12;
    public double Niacin;
    public double Folate;
    public double ProteinPercent;
    public double FatPercent;
    public double CarboPercent;
    public double EnergyPercent;
    public double CalciumPercent;
    public double IronPercent;
    public double VitAPercent;
    public double VitEPercent;
    public double VitCPercent;
```

```

public double VitB1Percent;
public double VitB2Percent;
public double VitB5Percent;
public double VitB6Percent;
public double VitB12Percent;
public double NiacinPercent;
public double FolatePercent;
public String err;
}

```

ในการประกาศ class เพื่อใช้รับส่งข้อมูล กับเว็บเซอร์วิส นั้น สามารถทำได้ในลักษณะที่มอง class นั้นเป็นชนิดข้อมูลหนึ่งเท่านั้น กล่าวคือ class ที่จะใช้นี้ จะต้องประกาศให้ properties ต่างๆ ของ class เป็น public เพื่อที่ เมื่อข้อมูลถูกส่งกลับไปยัง ผู้ที่ทำการเรียกเว็บเซอร์วิส ใช้นี้จะได้มองเห็นค่าที่ส่งกลับมา และนำค่านั้นไปใช้ได้ ซึ่งจะสามารถนำค่าจาก properties ไปใช้ได้ เท่านั้น ไม่สามารถนำค่าจาก method ไปใช้ได้ หากมีการประกาศ method ในคลาส ถึงแม้ว่าจะให้ method นั้นเป็น public ก็ตาม

เนื่องด้วยข้อกำหนดของ SOAP encoding ที่ว่าด้วยเรื่องของการแปลงโครงสร้างของข้อมูล ให้เป็นมาตรฐานเดียวกัน เพื่อจะได้มีความสามารถในการทำงานร่วมกัน ไม่ยึดติดกับรูปแบบหรือ Platform ใดๆ อย่างใดอย่างหนึ่ง ได้กล่าวถึงรูปแบบของข้อมูลที่จะใช้แลกเปลี่ยนกันระหว่างเว็บเซอร์วิส โดยที่ชนิดของข้อมูลที่จะสามารถแลกเปลี่ยนกันได้โดยตรงคือข้อมูลชนิด Primitive หรือ Array ของ Primitive นั่นคือข้อมูลธรรมดา เช่น Integer, float, double, string เป็นต้น หรือ Array ของข้อมูลเหล่านี้ นอกจากนี้ ยังสามารถส่งข้อมูลที่เรียกว่า Complex Type ได้ ซึ่งความหมายของ Complex Type ก็นิยามความครอบคลุมข้อมูลทุกประเภท ที่มีลักษณะที่เป็น โครงสร้างมีข้อมูลหลายๆ ประเภทอยู่ภายใน เช่น Object ของ Class หรือ Structure เป็นต้น ซึ่งหมายถึงข้อมูลที่มีลักษณะดังกล่าวจะต้องถูกแปลงให้มีลักษณะเป็น Complex Type เสียก่อน ที่จะส่งข้อมูลนั้นออกไป

หลังจากที่ได้รับข้อมูลเข้ามาแล้ว จะเริ่มการประมวลผลด้วยการตรวจสอบ ขนาดของ Array ที่ส่งเข้ามาก่อนว่า มีขนาดเท่ากันหรือไม่ เพื่อป้องกันความผิดพลาดในการประมวลผล ถ้าไม่เท่ากันจะจบการทำงานด้วยการส่ง object ของ class NutrientResult กลับโดยที่ภายในมีค่าของ err ที่ทำหน้าที่คอยแจ้งข้อผิดพลาดในการทำงาน ซึ่งในกรณีนี้คือ แจ้งเตือนว่าไม่สามารถคำนวณค่าได้ เนื่องจากขนาดของ Array ที่รับเข้ามาไม่เท่ากัน

ก่อนที่จะเข้าสู่การคำนวณหลัก ต้องมีการตรวจสอบก่อนว่าข้อมูลรายการอาหารทุกรายการ นั้นมีหน่วยในการชั่งตวงเป็นกรัมหรือไม่ ถ้าไม่จะต้องส่งไปคำนวณแปลงหน่วยการชั่งตวงให้เป็นกรัมที่เว็บเซอร์วิส MeasurementConversion ที่ได้ทำการ Add Web Reference เข้ามาเป็นส่วนหนึ่งของการคำนวณไว้ในตอนต้นแล้ว ซึ่งเมื่อทำการ Add Web References ใดๆ เข้ามาใน Project นั้น

Microsoft Visual Studio.NET จะทำการสร้าง Asynchronously Method ให้ทันที ซึ่งลักษณะการทำงานของ Asynchronously Method นี้คือ ในขณะที่เรียกใช้ Method ของ Web Reference จะเสมือนเป็นการทำงานผ่าน Asynchronously Method ที่สร้างขึ้น นั่นคือ ไม่จำเป็นต้องรอให้ Method ที่เรียกใช้ ทำงานเสร็จก่อน สามารถทำงานต่อไปได้เลย แล้วเมื่อ Asynchronously Method ได้รับการตอบกลับการทำงานเรียบร้อยแล้ว จะส่งผลกลับมาให้การทำงานหลักเอง

จากนั้นจะเข้าสู่การคำนวณโดยการวนรอบการทำงานจนกระทั่งหมดทุกรายการอาหารที่รับเข้ามา โดยจะทำการติดต่อกับฐานข้อมูลผ่านเทคโนโลยีของ ADO.NET ซึ่งจะต้องมีการประกาศการใช้ namespace ที่เกี่ยวข้องกับการทำงานกับฐานข้อมูลก่อน

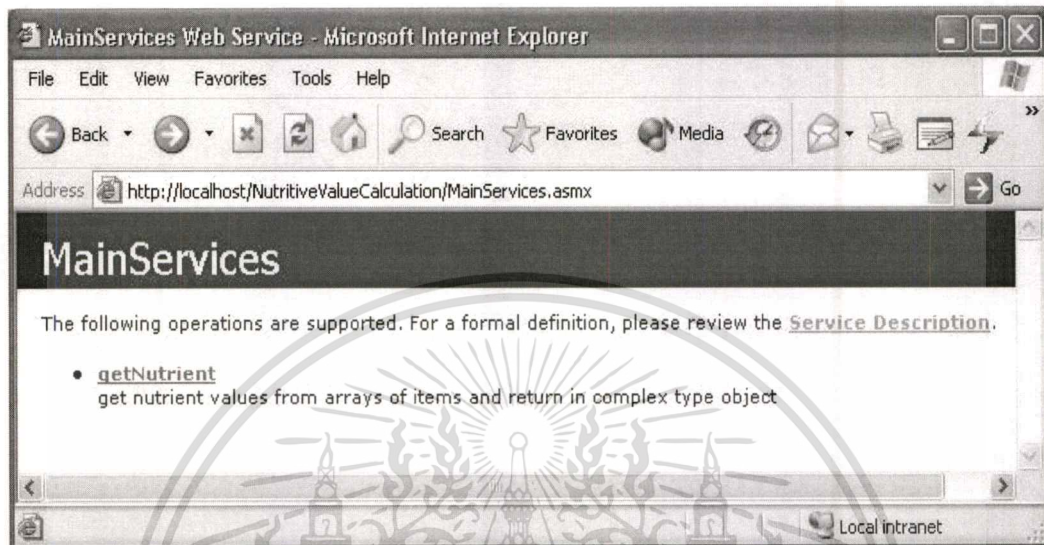
```
using System.Data; // เป็น namespace หลักของ ADO.NET
using System.Data.Common; // เกี่ยวกับการติดต่อและเรียกใช้ข้อมูลจากฐานข้อมูล
using System.Data.OleDb; //ใช้สำหรับ การติดต่อฐานข้อมูลผ่าน OLE DB Provider
```

โดย OleDb Provider เรียกใช้ stored procedure NutrientSP ซึ่งเทคนิคการติดต่อกับฐานข้อมูล Microsoft Access โดย stored procedure ในภาษา C# มีลักษณะดังนี้

```
string connectionString = "Provider=Microsoft.JET.OLEDB.4.0;
data source=" + Server.MapPath("NutritiveValue.mdb");
OleDbConnection conn = new OleDbConnection(connectionString);
OleDbCommand cmd = new OleDbCommand();
cmd.Connection=conn;
cmd.CommandText="NutrientSP";
cmd.CommandType=CommandType.StoredProcedure;
OleDbParameter param = _
    cmd.Parameters.Add("FoodName",OleDbType.VarChar);
param.Direction =ParameterDirection.Input;
param.Value=FoodItem[i];
```

เมื่อเชื่อมต่อฐานข้อมูลเรียบร้อยแล้ว จะอ่านข้อมูลที่ได้อ่านมาจากฐานข้อมูลด้วย OleDbDataReader เพื่อนำค่าที่คืนกลับมาคำนวณต่อไป โดยข้อมูลคุณค่าทางโภชนาการในแต่ละสารอาหารจะถูก คำนวณตามอัตราส่วน ปริมาณของอาหารนั้นๆ แล้วจึงนำค่าที่ได้ เก็บลงตัวแปรสารอาหารต่างๆ ใน object ของ class NutritiveValue ถ้ารายการอาหารใด ไม่พบในฐานข้อมูลจะบันทึกเป็น Exception ไว้ที่ตัวแปร err เพื่อแจ้งให้ผู้ใช้ทราบ เมื่อทำการคำนวณหมดจนทุกรายการอาหารแล้ว จะส่ง object ของ class ที่เก็บผลการคำนวณนี้ส่งกลับไปยังแอฟพลิเคชันที่เรียกใช้

และเมื่อทำการสร้าง NutritiveValueCalculation/MainServices เรียบร้อยแล้ว จะได้ไฟล์ .asmx ที่เป็นตัวดำเนินการในการประมวลผลเว็บเซอร์วิส ซึ่งเมื่อทดลองเรียกการทำงานจะได้ผลดังรูป



รูปที่ 4.4 แสดงผลการเรียกใช้ MainServices.asmx

#### การวิเคราะห์เอกสาร WSDL ของ MainService

Microsoft Visual Studio.Net ได้ทำการสร้าง เอกสาร WSDL เพื่ออธิบายคุณลักษณะของเว็บเซอร์วิส รวมถึงวิธีการเรียกใช้เว็บเซอร์วิสที่ ..MainServices.asmx?WSDL ซึ่งมีข้อมูลสำคัญที่ควรรู้เพื่อการเรียกใช้เว็บเซอร์วิสนี้ได้ถูกต้องดังนี้

(1) ข้อมูลที่จะบอกว่า Service นี้มีชื่อว่าอะไร และมี Listener ที่จะคอยเรียกการทำงานของเว็บเซอร์วิสนี้อยู่ที่ใด ซึ่งจะอยู่ส่วนท้ายของเอกสาร WSDL

```
- <service name="MainServices">
- <port name="MainServicesSoap" binding="s0:MainServicesSoap">
  <soap:address location="http://localhost/NutritiveValueCalculation/MainServices.asmx" />
</port>
```

(2) ข้อมูลที่จะบอกว่า Web Service นี้ให้บริการ Method ที่มีชื่อว่าอะไร ซึ่งผู้ให้บริการอาจจะแทรกข้อความที่อธิบายความหมายของ Method นี้ในเบื้องต้นมาด้วย

```
- <operation name="getNutrient">
  <documentation>get nutrient values from arrays of items and return in complex type object</documentation>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(3) ข้อมูลที่จะบอกว่า Web Service ที่ให้บริการนี้มีข้อมูลนำเข้าเป็นอย่างไร ซึ่ง Web Method getNutrient จะรับข้อมูลนำเข้าเป็น Array ของข้อมูล 3 ชนิด

```
- <s:element name="getNutrient">
- <s:complexType>
- <s:sequence>
  <s:element minOccurs="0" maxOccurs="1" name="FoodItem" type="s0:ArrayOfString" />
  <s:element minOccurs="0" maxOccurs="1" name="Value" type="s0:ArrayOfFloat" />
  <s:element minOccurs="0" maxOccurs="1" name="UnitName" type="s0:ArrayOfString" />
</s:sequence>
</s:complexType>
</s:element>
```

(4) ข้อมูลที่จะบอกลักษณะข้อมูลที่จะได้รับการตอบกลับมาจาก Web Service ซึ่ง Web Method getNutrient จะตอบกลับข้อมูลในรูปแบบ object ของ class NutrientResult

```
- <s:element name="getNutrientResponse">
- <s:complexType>
- <s:sequence>
  <s:element minOccurs="0" maxOccurs="1" name="getNutrientResult" type="s0:NutrientResult" />
</s:sequence>
</s:complexType>
</s:element>
- <s:complexType name="NutrientResult">
- <s:sequence>
  <s:element minOccurs="1" maxOccurs="1" name="Protein" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="Fat" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="Carbo" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="Energy" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="Calcium" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="Iron" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitA" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitE" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitC" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitB1" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitB2" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitB5" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitB6" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitB12" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="Niacin" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="Folate" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="ProteinPercent" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="FatPercent" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="CarboPercent" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="EnergyPercent" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="CalciumPercent" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="IronPercent" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitAPercent" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitEPercent" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitCPercent" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitB1Percent" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitB2Percent" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitB5Percent" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitB6Percent" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitB12Percent" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="NiacinPercent" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="FolatePercent" type="s:double" />
  <s:element minOccurs="0" maxOccurs="1" name="err" type="s:string" />
</s:sequence>
</s:complexType>
```

และสุดท้ายคือการทดสอบการทำงานของเว็บเซอร์วิส การเรียกการทำงานของเว็บเซอร์วิส แอปพลิเคชันที่เรียก จะส่ง SOAP message มากระตุ้นการทำงานของเว็บเซอร์วิส ซึ่งจะมีลักษณะดังนี้

```
POST /NutritiveValueCalculation/MainServices.asmx HTTP/1.1
Host: localhost
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://localhost/NutritiveValueCalculation/getNutrient"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <getNutrient xmlns="http://localhost/NutritiveValueCalculation/">
      <FoodItem>
        <string>string</string>
        <string>string</string>
      </FoodItem>
      <Value>
        <float>float</float>
        <float>float</float>
      </Value>
      <UnitName>
        <string>string</string>
        <string>string</string>
      </UnitName>
    </getNutrient>
  </soap:Body>
</soap:Envelope>
```

หลังจากเว็บเซอร์วิสถูกเรียกให้ทำงานแล้ว ก็จะตอบกลับไปในรูปแบบของ SOAP message เช่นเดียวกัน ซึ่งจะมีลักษณะดังนี้

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <getNutrientResponse
xmlns="http://localhost/NutritiveValueCalculation/">
      <getNutrientResult>
        <Protein>double</Protein>
        <Fat>double</Fat>
        <Carbo>double</Carbo>
        <Energy>double</Energy>
        <Calcium>double</Calcium>
        <Iron>double</Iron>
        <VitA>double</VitA>
        <VitE>double</VitE>
        <VitC>double</VitC>
        <VitB1>double</VitB1>
        <VitB2>double</VitB2>
        <VitB5>double</VitB5>
        <VitB6>double</VitB6>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<VitB12>double</VitB12>
<Niacin>double</Niacin>
<Folate>double</Folate>
<ProteinPercent>double</ProteinPercent>
<FatPercent>double</FatPercent>
<CarboPercent>double</CarboPercent>
<EnergyPercent>double</EnergyPercent>
<CalciumPercent>double</CalciumPercent>
<IronPercent>double</IronPercent>
<VitAPercent>double</VitAPercent>
<VitEPercent>double</VitEPercent>
<VitCPercent>double</VitCPercent>
<VitB1Percent>double</VitB1Percent>
<VitB2Percent>double</VitB2Percent>
<VitB5Percent>double</VitB5Percent>
<VitB6Percent>double</VitB6Percent>
<VitB12Percent>double</VitB12Percent>
<NiacinPercent>double</NiacinPercent>
<FolatePercent>double</FolatePercent>
<err>string</err>
</getNutrientResult>
</getNutrientResponse>
</soap:Body>
</soap:Envelope>

```

**4.1.2.2 NutrientDataAdmin** : เป็นเว็บเซอร์วิสที่เป็นเสมือนตัวแทนในการติดต่อกับฐานข้อมูลในการปรับปรุงรายการในฐานข้อมูลคุณค่าทางโภชนาการของอาหาร เพื่อป้องกันไม่ให้เกิดการทำงานกับฐานข้อมูลโดยตรง ซึ่งสามารถเพิ่มความปลอดภัยให้กับฐานข้อมูลได้ในระดับหนึ่ง ซึ่ง Method ที่มีให้บริการในเว็บเซอร์วิสนี้มีด้วยกัน 5 Methods ได้แก่ searchNutrient, getAliasItemName, countFoodItem, updateFoodItem และ UpdateAliasName

- **Web Method** : searchNutrient เป็น Method ที่ใช้เพื่อค้นหาอาหารหรือเครื่องปรุงอาหารที่ต้องการว่ามีอยู่ในฐานข้อมูลหรือไม่ และมีจำนวนคุณค่าทางโภชนาการของอาหารต่อ 100 กรัม เป็นจำนวนเท่าใด

```

[WebMethod]
public NutrientData searchNutrient(string FoodName)

```

จากการประกาศ WebMethod จะเห็นว่ามีบริการรับข้อมูลเข้ามาเพื่อที่จะประมวลผลเป็น string และส่งค่ากลับในรูปแบบ object ของ class NutrientData ซึ่งมีลักษณะดังนี้

```

public class NutrientData
{
    public string FoodName;
    public double Protein;
    public double Fat;
    public double Carbo;
    public double Energy;
    public double Calcium;
    public double Iron;
}

```

```

public double VitA;
public double VitE;
public double VitC;
public double VitB1;
public double VitB2;
public double VitB5;
public double VitB6;
public double VitB12;
public double Niacin;
public double Folate;
public string err;
public bool Record;
}

```

จะเห็นว่า class NutrientData จะมีลักษณะคล้ายกับ class NutrientResult ที่ใช้ใน MainService แต่จะลดในส่วนของข้อมูลที่เป็นผลการคำนวณร้อยละ และเพิ่มข้อมูลขึ้นมาอีก 2 properties คือ FoodName เพื่อบอกว่าชื่อของข้อมูลอาหารที่ค้นหา และ Record เพื่อใช้บอกว่า มีข้อมูลอาหารนี้อยู่ในฐานข้อมูลหรือไม่

การพัฒนา Web Method นี้จะเริ่มด้วยการรับข้อมูลเข้ามาเป็นชื่ออาหารที่ต้องการค้นหา จากนั้น ทำการติดต่อกับฐานข้อมูลผ่าน Stored Procedure NutrientSP แล้ว นำข้อมูลที่อ่านได้จาก ฐานข้อมูลเก็บเข้าตัวแปรใน object แล้วส่งค่า object นั้นกลับไปยังแอฟพลิเคชันที่เรียกใช้ แต่ถ้าไม่พบข้อมูลชื่ออาหารที่ค้นหา จะทำการส่ง object กลับ โดยบันทึกค่าตัวแปร Record ให้เป็น False

- **Web Method** : getAliasItemName เป็น Method ที่มีไว้เพื่อค้นหารายการอาหาร ว่า รายการอาหารที่ทำการค้นหานั้นสามารถ มีชื่อที่บันทึกไว้หรือไม่ อย่างไร

```

[WebMethod]
public DataSet getAliasItemName(string FoodName)

```

จากการประกาศ Web Method จะเห็นว่า getAliasName นั้นรับข้อมูลเข้ามาประมวลผลในรูปของ string ซึ่งเป็นชื่อของอาหารที่ต้องการค้นหา แล้วตอบกลับเป็นชื่อเรียกของอาหารชนิดนั้นที่บันทึกไว้ในฐานข้อมูล ซึ่งอยู่ในรูป DataSet

DataSet เป็นรูปแบบการจัดการข้อมูลที่ได้จากการติดต่อกับฐานข้อมูลแบบใหม่ ที่ได้รับการแนะนำภายใต้เทคโนโลยี ADO.NET ซึ่ง DataSet นี้มีประสิทธิภาพกว่าการจัดการรูปแบบเดิม หรือที่รู้จักกันในชื่อ RecordSet ซึ่งเพิ่มความสามารถขึ้นอย่างมากมาย อาทิเช่น การติดต่อกับฐานข้อมูลได้พร้อมๆ กันทีละหลายตาราง มีความสามารถในการเชื่อมต่อฐานข้อมูลแบบ Disconnect oriented กล่าวคือจะทำการเก็บข้อมูลในฐานข้อมูลที่เราทำการ Query นั้นเก็บลงฐานข้อมูลแล้วทำการเลิกการติดต่อกับฐานข้อมูลทันที ซึ่งแบบเดิมคือจะทำการเปิดฐานข้อมูลทิ้งไว้

จนกว่าจะจัดการกับฐานข้อมูลเสร็จ แล้วจึงปิดการติดต่อ ซึ่ง WebMethod นี้ไม่ได้มีจุดมุ่งหมาย เพื่อที่จะทดสอบความสามารถในการทำงานของ DataSet เพียงแต่ต้องการที่จะทดสอบ และดู รูปแบบของข้อมูล ในกรณีส่งค่ากลับจากเว็บเซอร์วิสมีลักษณะเป็น DataSet

ในการพัฒนา หลังจากที่มีการเรียกใช้ Method และได้รับข้อมูลเข้ามาเป็นชื่อของอาหาร แล้ว จะมีการเชื่อมต่อกับฐานข้อมูล โดยเรียกการทำงานของ Stored Procedure getAlias ขึ้นทำงาน ซึ่งจากข้างต้น Method นี้จะส่งค่ากลับเป็น DataSet ดังนั้นการเรียกอ่านข้อมูลที่ได้รับกลับมาผ่าน Object ตัวกลางที่เป็น OleDbDataAdapter เพื่อทำการปรับรูปแบบของข้อมูลให้อยู่ในรูปแบบ DataSet แล้วส่งกลับแอฟพลิเคชันที่เรียกใช้ต่อไป

การจัดการกับข้อมูลที่เป็น DataSet นั้น เนื่องจากเป็นรูปแบบที่ถูกสร้างขึ้นโดยเฉพาะใน .Net Framework เพื่อให้สื่อสารกับแอฟพลิเคชันอื่นๆ ได้จึงต้องมีการปรับรูปแบบให้อยู่ใน มาตรฐานโครงสร้างของ XML ซึ่งเมื่อสร้างเว็บเซอร์วิสแล้วลองทดสอบการทำงานจะเห็นว่า โครงสร้างของการส่งข้อมูลกลับนั้นจะอยู่ในรูปแบบที่มีข้อมูลที่ได้จากการทำงานของฐานข้อมูล ส่งไปพร้อมกับ XML Schema หรือเอกสาร XSD เพื่อเป็นการระบุโครงสร้างของข้อมูลให้กับข้อมูลที่ ส่งไปนั้น ทำให้การนำข้อมูลนี้ไปใช้ต่อทำได้อย่างถูกต้อง

- **Web Method** : countFoodItem เป็น Method ที่ใช้เพื่อนับจำนวนรายการอาหารที่มีอยู่ใน ฐานข้อมูลในปัจจุบันว่ามีจำนวนรายการอาหารเป็นเท่าใด

```
[WebMethod]
public long countFoodItem()
```

จากการประกาศ Web Method จะเห็นว่า เมื่อมีการเรียกทำงาน countFoodItem นี้ จะทำงาน ทันทีโดยไม่ต้องมี parameter ส่งเข้ามาเพื่อทำการประมวลผล แล้วส่งค่ากลับอยู่ในรูปแบบ long

ซึ่งในส่วนของการพัฒนานั้นเริ่มจากการติดต่อฐานข้อมูล ผ่าน Stored Procedure countItem เพื่อให้ทำการประมวลผลคำสั่ง SQL ให้แล้วสรุปค่าที่ได้ว่ามีจำนวนรายการอาหารที่อยู่ใน ฐานข้อมูลเป็นจำนวนเท่าใด จากนั้นจะใช้ OleDbDataReader อ่านข้อมูลที่ส่งกลับมา แล้ว ส่งกลับไปยังแอฟพลิเคชันที่เรียกใช้

- **Web Method** : UpdateFoodItem เป็น Method ที่ใช้เพื่อบันทึกเพิ่มหรือปรับปรุง รายการข้อมูลอาหารลงในฐานข้อมูลคุณค่าทางโภชนาการ ซึ่งจะต้องมีการจำกัดสิทธิ์การเข้าใช้ Method ให้เข้าใช้ Method นี้ได้เฉพาะผู้ที่มีสิทธิ์เข้าใช้ตามทะเบียน User ที่กำหนดไว้เท่านั้น

```
[WebMethod(Description="A secure method requires authentication by soap header")]
[SoapHeader("sHeader")]
public string updateFoodItem(NutrientData nData)
```

จากการประกาศ Web Method จะเห็นว่าข้อมูลที่รับเข้ามาจะอยู่ในรูป object ของ class NutrientData เก็บข้อมูลรายละเอียดปริมาณคุณค่าทางโภชนาการต่างๆ ของรายการอาหารนั้นและส่งกลับข้อมูลเป็น string ซึ่งจะเป็นข้อความบอกผลที่ได้จากการปรับปรุงข้อมูล แต่จะมีส่วนเพิ่มเติมพิเศษ นั่นคือการเรียกใช้ให้มีส่วนของการจัดการ SoapHeader ผสมเข้าไปกับ Method นี้ด้วย

จุดมุ่งหมายของการพัฒนา Web Method นั้นนอกจากจะเป็นสร้าง Method เพื่อใช้ในการปรับปรุงข้อมูลในฐานข้อมูล ผ่านรูปแบบการให้บริการเว็บเซอร์วิสแล้ว ยังเป็นการศึกษาทดสอบการทำงานของ SOAP Header ในส่วนของการทำ Authentication อีกด้วย เนื่องจากทางพิจารณาคุณลักษณะของ SOAP Message ที่ W3C ประกาศไว้นั้น จะพบว่ามีส่วนที่เป็นตัวเลือก หรือ optional ไว้ส่วนหนึ่งนั่นคือ SOAP Header ซึ่งเราจะ Implement หรือไม่ก็ได้ ซึ่งวัตถุประสงค์หลักที่มี SOAP Header ไว้เพื่อความยืดหยุ่นในการพัฒนา กล่าวคือเป็นการจัดการต่างๆ นอกเหนือจากการประมวลผลหลัก เช่น การจัดการเรื่อง Transaction หรือการจัดการ Authentication กำหนดสิทธิ์ผู้เข้าใช้ Method ดังที่จะได้ทดสอบใน Web Method นี้เป็นต้น

ซึ่งการ Implement ส่วน Header ให้กับ SOAP Message นั้น แต่ละเครื่องมือก็จะมีวิธีการที่แตกต่างกันไป แต่การทำงานหลักๆ ของมันก็คือการสร้าง tag ในรูปแบบเอกสาร xml เพิ่มส่วนที่บอกว่าเป็น Header อย่างเช่นใน Web Method นี้ เป็น Header ที่มีไว้เพื่อการทำ Authentication เพราะฉะนั้น แอปพลิเคชันที่เรียกใช้ Web Method นี้จำเป็นต้องทำการ Implement SOAP Header มากับ SOAP Message ด้วย โดยส่วนการทำงานของ Web Method เองก็จะทำการแยก ส่วนของ Header ออกไปประมวลผลพิจารณาสิทธิ์การเข้าใช้ แยกจากการทำงานหลักของ Method ซึ่งถ้าแอปพลิเคชันที่เรียกใช้ไม่มีการ Implement ส่วน SOAP Header จะเกิด Exception เป็น error ฟ้องความผิดพลาดว่าไม่มีการ Implement ส่วนของ Header และจะไม่ยอมให้เข้าไปใน Method ที่เรียกใช้ ดังเช่น

```
at
System.Web.Services.Protocols.SoapHeaderHandling.GetHeaderMembers(SoapHeaderCollection headers, Object target, SoapHeaderMapping[] mappings, SoapHeaderDirection direction, Boolean client)
at System.Web.Services.Protocols.SoapHttpClientProtocol.BeforeSerialize(WebRequest request, String methodName, Object[] parameters)
at System.Web.Services.Protocols.SoapHttpClientProtocol.Invoke(String methodName, Object[] parameters)
```

เริ่มการพัฒนาด้วยการเพิ่ม namespace ที่เกี่ยวข้องกับการ Implement SOAP Header นั้น หมายถึงเราจะต้องเข้าไปยุ่งเกี่ยวกับ SOAP Message หรือตัว Protocol เพราะฉะนั้น namespace ที่จะเพิ่มเข้ามามีดังนี้

```
using System.Web.Services; // เป็น namespace หลักในการสร้างเว็บเซอร์วิส ซึ่งได้ตั้งแต่ตอน
สร้างเว็บเซอร์วิส
using System.Web.Services.Protocols; // เป็น namespace ที่เพิ่มเข้ามาเพื่อจัดการกับ
SOAP Header
```

หลังจากนั้นจะทำการประกาศคลาสที่จะใช้เก็บ Username และ Password ที่จะนำเข้ามา เพื่อ Authentication ซึ่งมีลักษณะดังนี้

```
public class AuthHeader : SoapHeader
{
    public string Username;
    public string Password;
}
```

ซึ่งเป็นการประกาศ class AuthHeader ซึ่งเป็นการถ่ายทอดคุณสมบัติมาจาก class SoapHeader ภายในมีตัวแปรที่ใช้เก็บข้อมูล Username และ Password จากนั้น เมื่อจะใช้คุณสมบัติต่างๆ class AuthHeader จะต้องสร้าง Instance หรือ object ของ class นั้นขึ้นมา เพื่อใช้งานดังนี้

```
public AuthHeader sHeader;
```

ซึ่งจะเห็นว่าเป็นการประกาศ object sHeader ที่มีคุณสมบัติตาม class AuthHeader และถ้าย้อนกลับไปดูตอนที่ประกาศ WebMethod จะเห็นว่าเราได้ Implement ชื่อหรือโครงสร้างของ Header ในชื่อ sHeader นี้เอง

```
[SoapHeader ("sHeader")]
```

และเมื่อต้องการนำข้อมูลจาก Header ไปประมวลผล ก็ใช้ dot operation ในการดำเนินการ เรียกค่าของตัวแปรที่เก็บอยู่ออกมาใช้ได้ทันที เช่น sHeader.Username หรือ sHeader.Password เป็นต้น ซึ่งจะนำค่าที่ได้ ไปตรวจสอบกับข้อมูล User ในฐานข้อมูล โดยติดต่อกับฐานข้อมูลผ่าน Stored Procedure getUser นำค่า Password กลับมาตรวจสอบว่าเป็นผู้ใช้ที่มีสิทธิ์ให้เข้าแก้ไข ปรับปรุง ข้อมูลจริงหรือไม่ ถ้าจริงจึงจะเรียกการทำงานของ Method จริงๆ ขึ้นมาทำงาน แต่ถ้าไม่มีสิทธิ์ก็จะจบการทำงานและตอบกลับไปยังแอปพลิเคชันที่เรียกใช้ว่า ไม่สามารถเข้าแก้ไขข้อมูลใดๆ ได้ เนื่องจากไม่ได้รับอนุญาต

หลังจากที่ได้รับสิทธิ์ในการเข้าแก้ไข ปรับปรุงข้อมูลแล้ว ข้อมูลที่ได้รับเข้ามาซึ่งอยู่ในรูปแบบ class NutrientData นำค่า FoodName ที่ถูกส่งเข้ามาส่งไปเรียกการทำงานของ Stored

Procedure NutrientSP เพื่อตรวจสอบว่า มีรายการอาหารชนิดนี้อยู่ในฐานข้อมูลแล้วหรือยัง ถ้ามีแล้ว จะส่งคำสั่ง Update ข้อมูล แต่ถ้ายังไม่มี หมายถึงความต้องการที่จะเพิ่มรายการใหม่ จึงจะทำการส่งคำสั่ง Insert ข้อมูลต่างๆ ลงฐานข้อมูล และถ้าเป็นการเพิ่มรายการใหม่ สิ่งที่จะต้องทำต่อมาคือการเพิ่มชื่อของรายการอาหาร บันทึกเป็นรายการใหม่ลงในตาราง AliasName ด้วยเพื่อเก็บการเรียกชื่ออาหารนั้นไว้ เมื่อทุกอย่างเสร็จสิ้นลงจะส่งข้อความคืนกลับไปบอกแอฟพลิเคชันที่เรียกใช้ว่าปรับปรุงรายการเรียบร้อยแล้ว

- **Web Method** : updateAliasName เป็น Method ที่มีไว้เพื่อเพิ่มการเรียกชื่อให้กับรายการอาหารหรือส่วนผสมอาหารต่างๆ ในกรณีที่ รายการอาหารนั้นสามารถทำการเรียกได้หลายชื่อ หรือเป็นการเพิ่มข้อมูลการเรียกชื่ออาหารนั้นเป็นภาษาอื่น ซึ่ง Web Method นี้ก็จะต้องมีการตรวจสอบสิทธิ์การเข้าใช้ก่อน จึงจะสามารถปรับปรุงรายการในฐานข้อมูลได้เช่นเดียวกัน

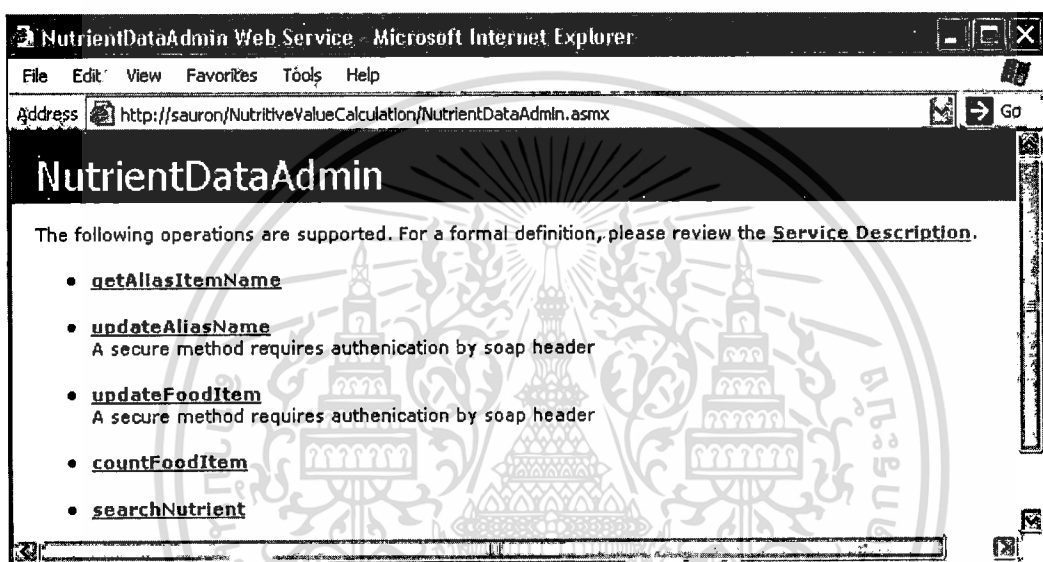
```
[WebMethod(Description="A secure method requires authentication by soap header")]
[SoapHeader("sHeader")]
public string updateAliasName(string FoodName, string AliasName)
```

จากการประกาศ Web Method จะเห็นว่า Method นี้มีส่วนที่ต้องทำการ Authentication ก่อนเข้าใช้ Method ในส่วนเริ่มต้นซึ่งใช้ object sHeader เป็นตัวแทนของ SOAP Header จากนั้นการประกาศการเรียกใช้ updateAliasName นี้จะต้องส่งค่า string เข้ามา 2 ค่า ซึ่งตัวแปรแรกจะเป็นชื่อเรียกที่มีอยู่แล้วในฐานข้อมูล และตัวแปรหนึ่งจะเป็นชื่อเรียกใหม่ ที่ต้องการเพิ่มเข้าไปยังฐานข้อมูล โดยที่เมื่อทำการประมวลผลเสร็จแล้ว Method นี้จะส่งค่ากลับเป็น string ซึ่งเป็นข้อความที่จะบอกผลการทำการของ Method

การพัฒนาเริ่มต้นขึ้นด้วย นำค่าที่ได้จาก Header ไปตรวจสอบข้อมูลผู้ใช้จากฐานข้อมูลว่ามีสิทธิ์เข้าปรับปรุงข้อมูลในฐานข้อมูลหรือไม่ เมื่อได้รับสิทธิ์แล้ว จะนำ AliasName หรือชื่ออาหารที่ต้องการเพิ่มเข้าไปในฐานข้อมูล ไปตรวจสอบว่ามีรายการนี้อยู่ในฐานข้อมูลแล้วหรือไม่ โดยเรียกการทำงานของ Method searchNutrient ขึ้นมาทำงาน ถ้าปรากฏว่ามีชื่อนั้นอยู่แล้วในฐานข้อมูลก็จะส่งข้อความกลับไปว่า มีข้อมูลรายการอาหารนั้นในฐานข้อมูลแล้ว แต่ถ้าไม่มีจะดำเนินการต่อ โดยจะนำชื่อของรายการอาหารนั้น (FoodName) ไปตรวจสอบว่ามี ID เป็นเท่าใด โดยทำการติดต่อกับฐานข้อมูลเรียก Stored Procedure getID ขึ้นมาทำงาน ถ้าไม่มีการตอบกลับมามีรายการอยู่แสดงว่าไม่มีชื่อของรายการอาหารนี้อยู่ในฐานข้อมูลมาก่อน ซึ่งก็จะไม่สามารถทำการ update ฐานข้อมูลได้ เพราะเป็นรายการใหม่ ต้องทำการเพิ่มค่าข้อมูลคุณค่าทางโภชนาการของอาหารก่อน ซึ่งจะส่งข้อความกลับไปว่า ไม่สามารถปรับปรุงรายการได้ เนื่องจากไม่มีข้อมูลอยู่ในฐานข้อมูล แต่ถ้า

เงื่อนไขทุกอย่างสมบูรณ์ กล่าวคือ FoodName เป็นรายการเก่าที่มีอยู่ในฐานข้อมูลจริง และ AliasName ก็เป็นรายการใหม่ที่ไม่ได้อยู่ในฐานข้อมูล และจะเป็นชื่อเรียกใหม่ให้กับ FoodName โปรแกรมจะทำการบันทึกข้อมูลเพิ่มรายการ AliasName ใหม่ในตาราง AliasName แล้วส่งข้อความกลับไปยังแอปพลิเคชันที่เรียกใช้ว่า ปรับปรุงรายการเรียบร้อยแล้ว

ถึงจุดนี้ ก็จะถือว่าได้ทำการพัฒนา NutrientDataAdmin สำเร็จ ซึ่งเมื่อทำการทดสอบเรียกไฟล์ NutrientDataAdmin.asmx ซึ่งจะได้ผลดังนี้



รูปที่ 4.5 แสดงผลการเรียกใช้ NutrientDataAdmin.asmx

#### การวิเคราะห์เอกสาร WSDL ของ NutrientDataAdmin

หลังจากพัฒนาเว็บเซอร์วิส ด้วย Microsoft Visual Studio.Net แล้วสิ่งที่จะได้ตามมานอกจากจะได้ File .asmx ที่ทำหน้าที่เป็นเว็บเซอร์วิส แล้วยังได้เอกสาร WSDL ที่ใช้อธิบายคุณลักษณะต่างๆ ของเว็บเซอร์วิสที่ได้สร้างขึ้นนี้ด้วย ซึ่งสามารถวิเคราะห์ส่วนต่างๆ ที่สำคัญเพื่อการเรียกใช้เว็บเซอร์วิส ดังนี้

(1) ข้อมูลที่มีไว้เพื่อบอกชื่อและที่อยู่ของเว็บเซอร์วิส NutrientDataAdmin ที่จะให้บริการและเรียกใช้ได้

```
<service name="NutrientDataAdmin">
  <port name="NutrientDataAdminSoap" binding="s0:NutrientDataAdminSoap">
    <soap:address
      location="http://sauron/NutritiveValueCalculation/NutrientDataAdmin.asmx" />
  </port>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## (2) ข้อมูลชื่อของ Method ที่เปิดให้บริการภายในเว็บเซอร์วิส NutrientDataAdmin

**Web Method : updateFoodItem**

```

- <operation name="updateFoodItem">
  <soap:operation      soapAction="http://localhost/NutritiveValueCalculation/updateFoodItem"
style="document" />
  - <input>
    <soap:body use="literal" />
    <soap:header      d5p1:required="true"      message="s0:updateFoodItemAuthHeader"
part="AuthHeader" use="literal" xmlns:d5p1="http://schemas.xmlsoap.org/wsdl/" />
  </input>
  - <output>
    <soap:body use="literal" />
  </output>
</operation>

```

**Web Method : updateAliasItem**

```

- <operation name="updateAliasName">
  <soap:operation      soapAction="http://localhost/NutritiveValueCalculation/updateAliasName"
style="document" />
  - <input>
    <soap:body use="literal" />
    <soap:header      d5p1:required="true"      message="s0:updateAliasNameAuthHeader"
part="AuthHeader" use="literal" xmlns:d5p1="http://schemas.xmlsoap.org/wsdl/" />
  </input>
  - <output>
    <soap:body use="literal" />
  </output>
</operation>

```

**Web Method : searchNutrient**

```

- <operation name="searchNutrient">
  <soap:operation      soapAction="http://localhost/NutritiveValueCalculation/searchNutrient"
style="document" />
  - <input>
    <soap:body use="literal" />
  </input>
  - <output>
    <soap:body use="literal" />
  </output>
</operation>

```

**Web Method : getAliasItemName**

```

- <operation name="getAliasItemName">
  <soap:operation      soapAction="http://localhost/NutritiveValueCalculation/getAliasItemName"
style="document" />
  - <input>
    <soap:body use="literal" />
  </input>
  - <output>
    <soap:body use="literal" />
  </output>
</operation>

```

**Web Method : countFoodItem**

```

- <operation name="countFoodItem">
  <soap:operation          soapAction="http://localhost/NutritiveValueCalculation/countFoodItem"
style="document" />
  <input>
  <soap:body use="literal" />
  </input>
- <output>
  <soap:body use="literal" />
  </output>
</operation>

```

จะเห็นว่า Web Method ที่ได้ทำการ Implement ส่วน SOAP header จะมีส่วน soap:header เพิ่มมาเพื่อเป็นการบอกแอฟพลิเคชันที่จะทำการเรียกใช้ Web Method updateFoodItem และ updateAliasName ให้ต้องทำการ Implement ส่วน SOAP header ด้วย

**(3) ส่วนที่แสดงข้อมูลนำเข้า เพื่อที่จะใช้ประมวลผลในเว็บไซต์ NutrientDataAdmin**

**Web Method : updateFoodItem** ซึ่งมีข้อมูลนำเข้าในรูปแบบ object ของ class

**NutrientData**

```

- <s:element name="updateFoodItem">
- <s:complexType>
- <s:sequence>
  <s:element minOccurs="0" maxOccurs="1" name="nData" type="s0:NutrientData" />
</s:sequence>
</s:complexType>
</s:element>

```

**Web Method : updateAliasItem** ซึ่งมีข้อมูลนำเข้าเป็น string ของชื่อรายการอาหารและชื่อเรียกอาหารที่ต้องการเพิ่มเข้าไปในฐานข้อมูล

```

- <s:element name="updateAliasName">
- <s:complexType>
- <s:sequence>
  <s:element minOccurs="0" maxOccurs="1" name="FoodName" type="s:string" />
  <s:element minOccurs="0" maxOccurs="1" name="AliasName" type="s:string" />
</s:sequence>
</s:complexType>
</s:element>

```

**Web Method : searchNutrient** ซึ่งมีข้อมูลนำเข้าเป็น string ของชื่ออาหาร

```

- <s:element name="searchNutrient">
- <s:complexType>
- <s:sequence>
  <s:element minOccurs="0" maxOccurs="1" name="FoodName" type="s:string" />
</s:sequence>
</s:complexType>
</s:element>

```

**Web Method** : getAliasItemName ซึ่งมีข้อมูลนำเข้าเป็น string ของชื่ออาหาร

```
- <s:element name="getAliasItemName">
- <s:complexType>
- <s:sequence>
  <s:element minOccurs="0" maxOccurs="1" name="FoodName" type="s:string" />
</s:sequence>
</s:complexType>
</s:element>
```

**Web Method** : countFoodItem ซึ่งเป็น Web Method ที่ไม่ต้องมีข้อมูลนำเข้าสามารถ

Invoke ได้ทันที

```
- <s:element name="countFoodItem">
  <s:complexType />
</s:element>
```

(4) ส่วนที่แสดงข้อมูลที่ได้รับการตอบกลับจากเว็บเซอร์วิส NutrientDataAdmin

**Web Method** : updateFoodItem ซึ่งมีข้อมูลตอบกลับเป็นแบบ string

```
- <s:element name="updateFoodItemResponse">
- <s:complexType>
- <s:sequence>
  <s:element minOccurs="0" maxOccurs="1" name="updateFoodItemResult" type="s:string" />
</s:sequence>
</s:complexType>
</s:element>
```

**Web Method** : updateAliasItem ซึ่งมีข้อมูลตอบกลับเป็น string

```
- <s:element name="updateAliasNameResponse">
- <s:complexType>
- <s:sequence>
  <s:element minOccurs="0" maxOccurs="1" name="updateAliasNameResult" type="s:string" />
</s:sequence>
</s:complexType>
</s:element>
```

**Web Method** : searchNutrient ซึ่งมีข้อมูลนำเข้าเป็น string ของชื่ออาหาร

```
- <s:element name="searchNutrientResponse">
- <s:complexType>
- <s:sequence>
  <s:element minOccurs="0" maxOccurs="1" name="searchNutrientResult"
  type="s0:NutrientData" />
</s:sequence>
</s:complexType>
</s:element>
```

**Web Method** : getAliasItemName ซึ่งมีข้อมูลนำเข้าเป็น string ของชื่ออาหาร

```
- <s:element name="getAliasItemNameResponse">
- <s:complexType>
- <s:sequence>
  <s:element minOccurs="0" maxOccurs="1" name="getAliasItemNameResult">
- <s:complexType>
- <s:sequence>
  <s:element ref="s:schema" />
<s:any />
</s:sequence>
</s:element>
```

```

</s:complexType>
</s:element>
</s:sequence>
</s:complexType>
</s:element>

```

**Web Method** : countFoodItem ซึ่งเป็น Web Method ที่ไม่ต้องมีข้อมูลนำเข้าสามารถ

Invoke ได้ทันที

```

- <s:element name="countFoodItemResponse">
- <s:complexType>
- <s:sequence>
  <s:element minOccurs="1" maxOccurs="1" name="countFoodItemResult" type="s:long" />
</s:sequence>
</s:complexType>
</s:element>

```

นอกจากนั้นภายในเอกสาร WSDL ยังได้อธิบายรูปแบบข้อมูลที่มีการอ้างอิงใช้ภายในเว็บเซอร์วิสนี้ไว้ด้วย

- โครงสร้างของข้อมูลรูปแบบ Nutrient Data ที่ได้มาจาก Class NutrientData

```

- <s:complexType name="NutrientData">
- <s:sequence>
  <s:element minOccurs="0" maxOccurs="1" name="FoodName" type="s:string" />
  <s:element minOccurs="1" maxOccurs="1" name="Protein" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="Fat" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="Carbo" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="Energy" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="Calcium" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="Iron" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitA" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitE" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitC" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitB1" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitB2" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitB5" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitB6" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="VitB12" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="Niacin" type="s:double" />
  <s:element minOccurs="1" maxOccurs="1" name="Folate" type="s:double" />
  <s:element minOccurs="0" maxOccurs="1" name="err" type="s:string" />
  <s:element minOccurs="1" maxOccurs="1" name="Record" type="s:boolean" />
</s:sequence>
</s:complexType>

```

- โครงสร้างของข้อมูลรูปแบบ AuthHeader ที่ได้มาจาก Class AuthHeader เพื่อใช้กับการ

จัดการ Authentication ในส่วน Header

```

- <s:complexType name="AuthHeader">
- <s:sequence>
  <s:element minOccurs="0" maxOccurs="1" name="Username" type="s:string" />
  <s:element minOccurs="0" maxOccurs="1" name="Password" type="s:string" />
</s:sequence>
</s:complexType>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- โครงสร้างของข้อมูลรูปแบบ DataSet ซึ่งมีการอ้างอิงต่อไปยัง xml schema หรือเอกสาร xsd ที่จะแสดงโครงสร้างข้อมูลที่ได้มาจากชนิดข้อมูล DataSet ซึ่งใช้จัดการกับข้อมูลที่ได้รับ

ตอบกลับมาจากฐานข้อมูล

```
- <s:element name="DataSet" nillable="true">
- <s:complexType>
- <s:sequence>
  <s:element ref="s:schema" />
  <s:any />
</s:sequence>
</s:complexType>
</s:element>
```

### ลักษณะ SOAP Message

**Web Method : updateFoodItem**

(SOAP Message ที่รับเข้า)

POST /NutritiveValueCalculation/NutrientDataAdmin.asmx HTTP/1.1

Host: localhost

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

Content-Length: length

SOAPAction: "http://localhost/NutritiveValueCalculation/updateFoodItem"

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <AuthHeader xmlns="http://localhost/NutritiveValueCalculation/">
      <Username>string</Username>
      <Password>string</Password>
    </AuthHeader>
  </soap:Header>
  <soap:Body>
    <updateFoodItem xmlns="http://localhost/NutritiveValueCalculation/">
      <nData>
        <FoodName>string</FoodName>
        <Protein>double</Protein>
        <Fat>double</Fat>
        <Carbo>double</Carbo>
        <Energy>double</Energy>
        <Calcium>double</Calcium>
        <Iron>double</Iron>
        <VitA>double</VitA>
        <VitE>double</VitE>
        <VitC>double</VitC>
        <VitB1>double</VitB1>
        <VitB2>double</VitB2>
        <VitB5>double</VitB5>
        <VitB6>double</VitB6>
        <VitB12>double</VitB12>
        <Niacin>double</Niacin>
        <Folate>double</Folate>
        <err>string</err>
        <Record>boolean</Record>
      </nData>
    </updateFoodItem>
  </soap:Body>
</soap:Envelope>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(SOAP Message ที่ตอบกลับ)

HTTP/1.1 200 OK  
Content-Type: text/xml; charset=utf-8  
Content-Length: length

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <updateFoodItemResponse xmlns="http://localhost/NutritiveValueCalculation/">
      <updateFoodItemResult>string</updateFoodItemResult>
    </updateFoodItemResponse>
  </soap:Body>
</soap:Envelope>
```

**Web Method : updateAliasName**

(SOAP Message ที่รับเข้า)

POST /NutritiveValueCalculation/NutrientDataAdmin.asmx HTTP/1.1  
Host: localhost  
Content-Type: text/xml; charset=utf-8  
Content-Length: length  
SOAPAction: "http://localhost/NutritiveValueCalculation/updateAliasName"

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <AuthHeader xmlns="http://localhost/NutritiveValueCalculation/">
      <Username>string</Username>
      <Password>string</Password>
    </AuthHeader>
  </soap:Header>
  <soap:Body>
    <updateAliasName xmlns="http://localhost/NutritiveValueCalculation/">
      <FoodName>string</FoodName>
      <AliasName>string</AliasName>
    </updateAliasName>
  </soap:Body>
</soap:Envelope>
```

(SOAP Message ที่ตอบกลับ)

HTTP/1.1 200 OK  
Content-Type: text/xml; charset=utf-8  
Content-Length: length

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <updateAliasNameResponse xmlns="http://localhost/NutritiveValueCalculation/">
      <updateAliasNameResult>string</updateAliasNameResult>
    </updateAliasNameResponse>
  </soap:Body>
</soap:Envelope>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Web Method : searchNutrient**

(SOAP Message ที่รับเข้า)

POST /NutritiveValueCalculation/NutrientDataAdmin.asmx HTTP/1.1

Host: localhost

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

Content-Length: length

SOAPAction: "http://localhost/NutritiveValueCalculation/searchNutrient"

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <searchNutrient xmlns="http://localhost/NutritiveValueCalculation/">
      <FoodName>string</FoodName>
    </searchNutrient>
  </soap:Body>
</soap:Envelope>

```

(SOAP Message ที่ตอบกลับ)

HTTP/1.1 200 OK

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

Content-Length: length

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <searchNutrientResponse xmlns="http://localhost/NutritiveValueCalculation/">
      <searchNutrientResult>
        <FoodName>string</FoodName>
        <Protein>double</Protein>
        <Fat>double</Fat>
        <Carbo>double</Carbo>
        <Energy>double</Energy>
        <Calcium>double</Calcium>
        <Iron>double</Iron>
        <VitA>double</VitA>
        <VitE>double</VitE>
        <VitC>double</VitC>
        <VitB1>double</VitB1>
        <VitB2>double</VitB2>
        <VitB5>double</VitB5>
        <VitB6>double</VitB6>
        <VitB12>double</VitB12>
        <Niacin>double</Niacin>
        <Folate>double</Folate>
        <err>string</err>
        <Record>boolean</Record>
      </searchNutrientResult>
    </searchNutrientResponse>
  </soap:Body>
</soap:Envelope>

```

**Web Method : getAliasItemName**

(SOAP Message ที่รับเข้า)

POST /NutritiveValueCalculation/NutrientDataAdmin.asmx HTTP/1.1

Host: localhost

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

Content-Length: length

SOAPAction: "http://localhost/NutritiveValueCalculation/getAliasItemName"

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <getAliasItemName xmlns="http://localhost/NutritiveValueCalculation/">
      <FoodName>string</FoodName>
    </getAliasItemName>
  </soap:Body>
</soap:Envelope>
```

(SOAP Message ที่ตอบกลับ)

HTTP/1.1 200 OK

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

Content-Length: length

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <getAliasItemNameResponse xmlns="http://localhost/NutritiveValueCalculation/">
      <getAliasItemNameResult>
        <xsd:schema>schema</xsd:schema>xml</getAliasItemNameResult>
      </getAliasItemNameResponse>
    </soap:Body>
  </soap:Envelope>
```

**Web Method : countFoodItem**

(SOAP Message ที่รับเข้า)

POST /NutritiveValueCalculation/NutrientDataAdmin.asmx HTTP/1.1

Host: localhost

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

Content-Length: length

SOAPAction: "http://localhost/NutritiveValueCalculation/countFoodItem"

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <countFoodItem xmlns="http://localhost/NutritiveValueCalculation/" />
  </soap:Body>
</soap:Envelope>
```

(SOAP Message ที่ตอบกลับ)

HTTP/1.1 200 OK

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

Content-Length: length

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <countFoodItemResponse xmlns="http://localhost/NutritiveValueCalculation/">
      <countFoodItemResult>long</countFoodItemResult>
    </countFoodItemResponse>
  </soap:Body>
</soap:Envelope>
```

## 4.2 Measurement Conversion

เว็บเซอร์วิสนี้จะถูกเรียกใช้ให้ทำงานเมื่อต้องการแปลงมาตรการชั่งตวงอาหารหรือ วัตถุประสงค์ในการปรุงอาหาร จากมาตราในการชั่งตวงต่างๆ ให้เป็นหน่วยกรัม ตามอัตราส่วน เปรียบเทียบที่ได้กำหนดไว้ โดยข้อมูลที่จะต้องนำมาพิจารณาคำนวณคือ หน่วยในการชั่งตวงเดิม และน้ำหนักของอาหารที่ต้องการคำนวณนั้นๆ

### เครื่องมือที่ใช้พัฒนา

- Borland JBuilder 7 Enterprise - WebLogic Edition
- BEA WebLogic Server 7.0

### ขั้นตอนในการพัฒนา

1. ทำการติดตั้ง Web Logic Server ให้กับ JBuilder เพื่อให้ JBuilderรู้ว่าจะต้องทำงานกับ Application Server ไດ
2. สร้างโปรเจคใหม่ แล้วตั้งค่า Properties ให้กับโปรเจคนั้น โดยให้เลือก Server เป็น Web Logic Server แล้วเพิ่ม WebLogic Webservices Library เข้าไปใน Project Required Libraries และ Ant Build ซึ่ง Ant นี้จะมีหน้าที่ในนำ Class ทั้งหมดที่เกี่ยวข้อง หรือไฟล์ EJB Jar ไปสร้างคอมโพเนนต์ทั้งหมดที่จำเป็นต้องใช้กับเว็บ เซอร์วิส แล้วจะทำการ package สิ่งที่ได้มาทั้งหมดให้กลายเป็น EAR (Enterprise Archive) เพื่อรอกการนำไป Deploy ต่อไป
3. สร้าง EJB Module แล้วสร้าง Session ให้กับ EJB แล้ว เขียนโปรแกรม สร้าง Methods การทำงานทั้งหมดของการคำนวณเปรียบเทียบมาตราการชั่งตวงใน Bean

4. หลังจากนั้นทำการ Build Project
5. เสร็จแล้วให้นำ ไฟล์ .xml ที่ทำหน้าที่เป็น Ant เข้ามาในโปรเจก จากนั้นให้ Ant สร้าง EAR
6. เมื่อได้ ไฟล์ EAR แล้ว ให้เปิดเซอรัวิส การทำงานของ Weblogic Server แล้วทำการ Deploy ไฟล์ EAR ให้กลายเป็นเว็บเซอรัวิส บน Server
7. สุดท้ายคือการทดสอบการทำงานของเว็บเซอรัวิส

#### รายละเอียดในการพัฒนา

ในการสร้างเว็บเซอรัวิสคอมโพเนนต์ รูปแบบ EJB ด้วยเครื่องมือ Borland Jbuilder นั้น จะเริ่มต้นด้วยการสร้างโปรเจกใหม่ ซึ่งจะเป็น file .jpx เพื่อเก็บข้อมูลเกี่ยวกับโปรเจกทั้งหมด ซึ่งเพื่อให้โปรเจกที่สร้างใหม่นี้ เมื่อทำการคอมไพล์แล้วสามารถทำงานได้ในลักษณะ EJB เว็บเซอรัวิส จึงต้องมีการตั้งค่า required libraries ใน project properties ซึ่ง WebLogic WebServices library ที่จะเพิ่มเข้าไปนี้ได้มาจากการติดตั้ง BEA WebLogic Server นอกจากจะติดตั้ง library ให้กับโปรเจกสำหรับการคอมไพล์แล้ว ยังต้องทำการติดตั้ง library นี้ให้กับ Ant ในการ build โปรเจกให้มีลักษณะเป็น EJB ด้วย ซึ่ง Ant จะทำหน้าที่รวบรวม library, component หรือไฟล์ที่เกี่ยวข้องต่างๆ จัดอยู่ในรูปแบบ EJB แล้ว package อย่างเป็นระเบียบ เพื่อให้การพัฒนาทำได้ง่ายขึ้น โดยที่เราไม่ต้องรวบรวมสิ่งต่างๆ เหล่านั้นเอง

หลังจากทำการสร้างโปรเจกและติดตั้งสิ่งที่ต้องการต่างๆ เรียบร้อย ให้ทำการสร้าง EJB Module ซึ่งชื่อของ EJB Module นี้จะกลายเป็นส่วนหนึ่งของ URL ที่เป็นที่อยู่ของเว็บเซอรัวิสหรือ SOAP Listener เช่น EJB Module ชื่อ MeasurementConversion เมื่อทำการสร้างเว็บเซอรัวิส เรียบร้อยแล้วก็จะได้ URL <http://localhost:7001/MeasurementConversion/service> ซึ่งภายใน EJB Module นี้จะทำการสร้าง Session Bean เพื่อเป็นส่วนกำหนดขอบเขตการทำงานทั้งหมดของ EJB นี้ โดยที่โปรแกรมจะให้ทำการตั้งชื่อ Bean และ Package ซึ่งชื่อของ Bean จะกลายเป็นชื่อของ file source code.java และ ชื่อ class ที่สร้างขึ้น และ Package จะเป็นที่เก็บ class เหล่านั้นอย่างเป็นระเบียบ โดยที่เมื่อถึงจุดนี้แล้ว โปรแกรมจะทำการสร้างไฟล์ เพื่อใช้การทำงานภายในเว็บเซอรัวิส ขึ้น 3 ไฟล์ ด้วยกัน ซึ่งเว็บเซอรัวิสนี้ได้ตั้งชื่อ session Bean เป็น ConversionService ซึ่งก็จะได้ไฟล์สำหรับการทำงาน ดังนี้

ConversionService.java

ConversionServiceBean.java

ConversionServiceHome.java

หลังจากนั้นจะเป็นการเขียนโปรแกรมเพื่อให้เกิดการคำนวณเปลี่ยนแปลงหน่วยในการชั่งตวงอาหาร ซึ่งเริ่มต้นจากการการปรับปรุง ส่วนของ ConversionService.java ก่อน เนื่องจากจะเป็นการสร้าง Interface ที่ชื่อ ConversionService ที่ถ่ายทอดคุณสมบัติมาจาก class ที่ใช้ในการสร้าง EJB Object ซึ่งในเว็บเซอร์วิสนี้จะใช้ Interface นี้ในการประกาศว่าเว็บเซอร์วิสนี้สามารถให้บริการ Method ใดๆ ได้บ้าง

```
package ws.ejb.service;

import javax.ejb.*;
import java.util.*;
import java.rmi.*;

public interface ConversionService extends javax.ejb.EJBObject {
    public float ToGram(String unit, float value) throws
RemoteException;
}
```

จาก Code ด้านบนจะเห็นว่ามีวิธีการสร้าง Method หนึ่งที่ชื่อ ToGram โดยรับข้อมูลเป็น String และ float แล้วมีการส่งค่ากลับเป็น float

ในส่วนการทำงานคำนวณจริงทั้งหมดจะอยู่ใน ConversionServiceBean.java ซึ่งจะเป็นการสร้าง Session ในการทำงาน โดยเครื่องมือจะทำการสร้าง Class และ Implement Method ที่จำเป็นให้ แล้วหลังจากนั้นจะเป็นการเขียนโปรแกรมในส่วน Method ที่จะใช้คำนวณเปลี่ยนแปลงหน่วยการชั่งตวงอาหารจริง

```
package ws.ejb.service;

import javax.ejb.*;

public class ConversionServiceBean implements SessionBean {
    SessionContext sessionContext;
    public void ejbCreate() throws CreateException {}
    public void ejbRemove() {}
    public void ejbActivate() {}
    public void ejbPassivate() {}
    public void setSessionContext(SessionContext sessionContext) {
        this.sessionContext = sessionContext;
    }

    public float ToGram(String unit, float value){
        ...
    }
}
```

การคำนวณจะเริ่มจากการรับข้อมูลนำเข้าซึ่งเป็น หน่วยในการชั่งตวงและปริมาณที่ต้องการเปลี่ยนแปลงหน่วยในการชั่งตวงเป็นกรัม โดยจะนำ unit หรือหน่วยที่ยังไม่ได้แปลงนี้ ไปเปรียบเทียบกับหน่วยในการชั่งตวงทั้งหมดที่ได้มีการบันทึกไว้ ซึ่งถ้าตรงกับหน่วยใด ก็จะทำการเรียกค่าของหน่วยนั้นขึ้นมา ทำการคำนวณเปรียบเทียบกับ ค่าของ value หรือปริมาณเดิม แล้วส่งผลการคำนวณที่ได้กลับไปยังแอปพลิเคชันที่เรียกใช้เว็บเซอร์วิสต่อไป

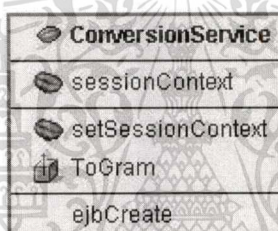
ส่วนไฟล์ ConversionServiceHome.java จะเป็นส่วนที่เครื่องมือสร้างให้ทั้งหมด ซึ่งเป็น การสร้าง Interface ที่ใช้ในสร้าง Measurement Conversion Web Service นี้

```
package ws.ejb.service;

import javax.ejb.*;
import java.util.*;
import java.rmi.*;

public interface ConversionServiceHome extends javax.ejb.EJBHome {
    public ConversionService create() throws CreateException,
RemoteException;
}
```

เมื่อถึงจุดนี้ถือได้ว่าการพัฒนาเว็บเซอร์วิสเพื่อใช้การการเปลี่ยนแปลงหน่วยของอาหารใน ส่วนของการเขียน โปรแกรมสำเร็จแล้วเหลือเพียงส่วนของการคอมไพล์โปรเจกต์ทั้งหมด สร้างขึ้น เป็น EJB และการติดตั้ง EJB บน BEA WebLogic Application Server ซึ่งลักษณะของ Session Bean หลังจากที่ได้ทำการเขียน โปรแกรมแล้วจะเป็นดังนี้



รูปที่ 4.6 แสดงลักษณะของ Session Bean

เมื่อทำการคอมไพล์โปรเจกต์ทั้งหมด โดยคำสั่ง Make Project แล้วจะได้ไฟล์ชื่อ ConversionService.jar ซึ่งจะเก็บรวบรวมคลาสต่างๆ ไว้ภายในสามารถนำไปใช้งานในลักษณะ คอมโพเนนต์โดยทั่วไปได้ แต่ยังไม่ถือว่าเป็น EJB ที่จะใช้ทำงานในลักษณะเว็บเซอร์วิส จะต้องมีการสร้างไฟล์ที่จะทำหน้าที่เป็น Ant หรือ ตัวที่จะรวบรวมสิ่งที่เกี่ยวข้องทั้งหมด Package ให้อยู่ใน ลักษณะ EJB Web Service โดยที่ไฟล์ที่ทำหน้าที่เป็น Ant จะใช้ชื่อ build.xml ซึ่งมีลักษณะดังนี้

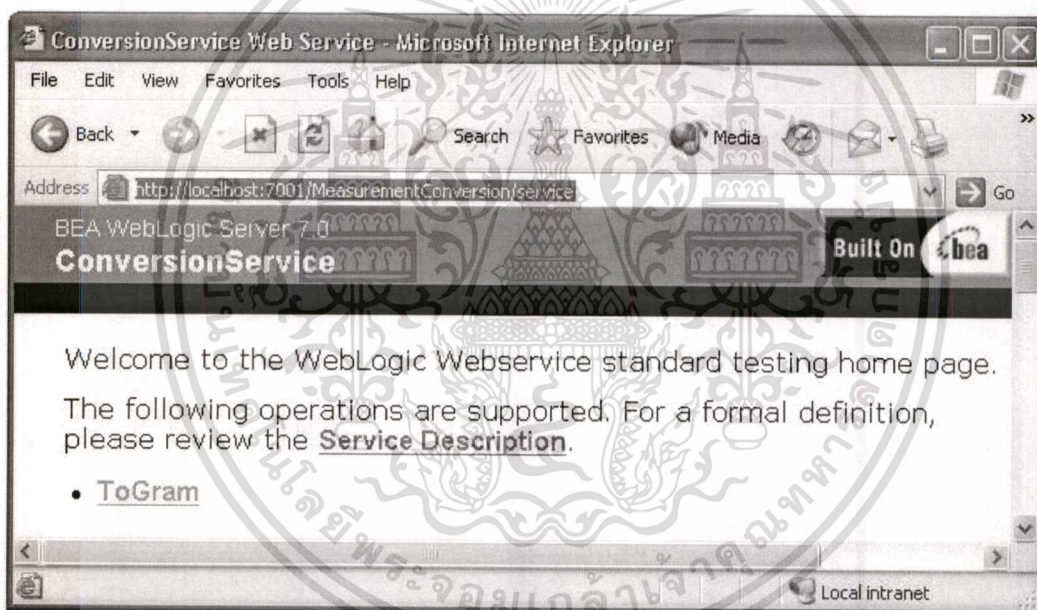
```
<project name="servicegen" default="servicegen" basedir=".">
  <property file="build.properties"/>
  <target name="servicegen">
    <servicegen destEar="{service.ear}"
      warName="{service.war}">
      <service.ejbJar="{service.jar}"
        targetNamespace="{service.ns}"
        serviceName="{service.name}"
        serviceURI="{service.uri}"
        generateTypes="True"
        expandMethods="True" />
    </servicegen>
  </target>
</project>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ Ant นี้จะทำการสร้าง EJB Web Service ให้มีคุณสมบัติต่างๆ ต่างที่กำหนดไว้ ซึ่งจะเห็นว่าคุณสมบัติต่างๆ เช่น ชื่อของ Service, ที่อยู่ของไฟล์ jar เป็นต้น มีการอ้างอิงจากไฟล์ที่ชื่อ build.properties ซึ่งจะเก็บคุณสมบัติต่างๆ เหล่านั้น โดยมีลักษณะดังนี้

```
service.name = ConversionService
service.jar = MeasurementConversion.jar
service.ear = MeasurementConversion.ear
service.war = MeasurementConversion.war
service.ns = http://ConversionService.net
service.uri = /service
```

ซึ่งเมื่อทำการ build EJB Web Service นี้โดย Ant แล้วจะได้ไฟล์ .ear ซึ่งเมื่อนำไฟล์ MeasurementConversion.ear นี้ ไปทำการ Deployment บน Application Server ก็ถือว่าการพัฒนานั้นเสร็จเรียบร้อย และให้บริการเว็บเซอร์วิสได้ทันที



รูปที่ 4.7 แสดงการ Measurement Conversion Web Service เมื่อได้ Deployment บน BEA WebLogic Server

#### การวิเคราะห์เอกสาร WSDL ของ MeasurementConversion

เช่นเดียวกับการพัฒนาเว็บเซอร์วิส ด้วยวิธีการอื่นๆ เมื่อทำการสร้างเว็บเซอร์วิสเรียบร้อยแล้ว สิ่งที่จะได้ตามมาด้วยคือ เอกสาร WSDL ที่จะใช้อธิบายคุณลักษณะ และวิธีการเรียกใช้เว็บเซอร์วิสต่างๆ ซึ่งเมื่อทำการ Deployment EJB Web Service บน Application Server แล้วสามารถเข้าถึงเอกสาร WSDL ได้โดย ../service?WSDL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(1) ข้อมูลที่จะบอกว่า Service นี้มีชื่อว่าอะไร และมี Listener ที่จะคอยเรียกการทำงานของเว็บเซอร์วิสที่อยู่ใด ซึ่งจะอยู่ส่วนท้ายของเอกสาร WSDL

```
- <service name="ConversionService">
  <documentation>todo: add your documentation here</documentation>
- <port name="ConversionServicePort" binding="tns:ConversionServicePortSoapBinding">
  <soap:address location="http://localhost:7001/MeasurementConversion/service" />
  </port>
</service>
```

(2) ข้อมูลที่บอกชื่อของ Operation หรือ Method ที่มีให้บริการภายใต้ ConversionService

```
- <operation name="ToGram">
  <soap:operation soapAction="" style="rpc" />
```

(3) ส่วนแสดงข้อมูลที่คือนำเข้าเพื่อประมวลผลใน Method ToGram ได้แก่ ข้อมูลหน่วยในการชั่งตวงที่ต้องการแปลงเป็นกรัมซึ่งเป็น String และปริมาณที่ต้องการคำนวณเปรียบเทียบ ซึ่งเป็น float

```
<message name="ToGram">
<part name="string" xmlns:partns="http://www.w3.org/2001/XMLSchema" type="partns:string" />
<part name="floatVal" xmlns:partns="http://www.w3.org/2001/XMLSchema"
type="partns:float" />
</message>
```

(4) ส่วนแสดงข้อมูลที่คืนกลับมาจากการประมวลผล Method ToGram ซึ่งเป็น float

```
<message name="ToGramResponse">
<part name="result" xmlns:partns="http://www.w3.org/2001/XMLSchema" type="partns:float" />
</message>
```

### ลักษณะ SOAP Message

(SOAP Message ที่รับเข้า)

```
<env:Envelope xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:env="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<env:Header>
</env:Header>
<env:Body>
<m:ToGram xmlns:m="http://ConversionService.net"
env:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
<string xsi:type="xsd:string">ถ้วยตวง</string>
<floatVal xsi:type="xsd:float">0.75</floatVal>
</m:ToGram>
</env:Body>
</env:Envelope>
```

(SOAP Message ที่ตอบกลับ)

```
<env:Envelope xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:env="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<env:Header>
</env:Header>
<env:Body>
<m:ToGramResponse xmlns:m="http://ConversionService.net"
env:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
<result xsi:type="xsd:float">177.4425</result>
</m:ToGramResponse>
</env:Body>
</env:Envelope>
```

### 4.3 Percentage of Requirement

เว็บเซอร์วิสนี้ใช้ในการคำนวณร้อยละของคุณค่าทางโภชนาการของอาหารเมื่อเปรียบเทียบกับปริมาณที่เป็นมาตรฐาน ที่แนะนำให้บริโภคในแต่ละวัน โดยที่ข้อมูลที่จะต้องนำเข้ามาเพื่อคำนวณคือ ชื่อสารอาหารที่ต้องการคำนวณ และปริมาณที่มีของสารอาหารนั้นๆ

#### เครื่องมือที่ใช้พัฒนา

- Microsoft Visual Basic 6
- Microsoft Internet Information Services 5.1
- Microsoft SOAP Toolkit 3

#### ขั้นตอนในการพัฒนา

1. สร้างโปรเจกใหม่ให้เป็นในลักษณะ ActiveX DLL
2. สร้าง Method หรือ Function การทำงานต่างๆ ทั้งหมดในการคำนวณสัดส่วนร้อยละของการบริโภคแต่ละสารอาหารในแต่ละวัน
3. เมื่อเขียนโปรแกรมทั้งหมดเสร็จแล้ว ให้ทำการคอมไพล์สร้างให้เป็นไฟล์ DLL ซึ่งจะ  
ได้เป็น Component เพื่อนำไปใช้งานได้
4. จากนั้นอาจมีการตรวจสอบการทำงานของ Component ที่สร้างขึ้น โดยเขียนโปรแกรม  
ง่ายๆ ในการทำงานกับ Component ที่สร้างขึ้นมานั้น
5. เรียกโปรแกรม Microsoft SOAP Toolkit ในส่วนที่เป็น WSDL Generator ขึ้นมา  
ทำงาน เพื่อทำการสร้างเอกสาร WSDL ประกอบกับ Component ที่สร้างขึ้น เพื่อให้  
กลายเป็นเว็บเซอร์วิส ซึ่งการทำงานของ WSDL Generator จะทำการถามว่าต้องการนำ

ไฟล์ที่สร้างทั้งหมดไปไว้ที่ใด ซึ่งจะต้องนำไฟล์ที่เป็น SOAP Listener และ WSDL เก็บไว้ที่ Web Server เพื่อรอการเรียกใช้บริการ

6. เมื่อได้ไฟล์ .wsdl แล้ว ให้ทดสอบการทำงานของเว็บเซอร์วิสที่สร้างขึ้น

### รายละเอียดในการพัฒนา

การสร้างเว็บเซอร์วิสที่มาจากคอมโพเนนต์แบบ ActiveX DLL โดยใช้ Visual Basic นั้น วิธีที่ง่ายที่สุดคือสร้างการทำงานอย่างใดอย่างหนึ่งในลักษณะการทำงานแบบ function ซึ่ง function เหล่านี้ที่ได้มีการเขียนขึ้นเหล่านี้จะกลายเป็น Operation หรือ Method ของเว็บเซอร์วิส

```
Public Function getNutrientPercent(ByVal NutrientName As String, ByVal srcValue As Double) As Double
```

จากการประกาศฟังก์ชัน ซึ่งต้องเป็นการประกาศประกาศแบบ Public เท่านั้น เพื่อให้แอปพลิเคชันอื่นสามารถเรียกใช้การทำงานได้ ส่วนที่สำคัญอีกประการหนึ่ง คือเพื่อให้การใช้งานตัวแปรกับเว็บเซอร์วิสนี้ ทำงานได้อย่างถูกต้องนั้น ต้องมีการประกาศตัวแปรที่จะนำเข้าไปในแบบ ByVal หรือการนำเฉพาะค่าของตัวแปรนั้นมาใช้เท่านั้น ซึ่งถ้าไม่ประกาศในลักษณะ ByVal แล้ว Microsoft Visual Basic 6 ได้กำหนดค่าเริ่มต้น หรือค่าหลักสำหรับการเรียกใช้ตัวแปรในแบบ ByRef หรือการอ้างอิงค่าของตัวแปรจากหน่วยความจำจริง อาจทำให้การทำงานในลักษณะของเว็บเซอร์วิสซึ่งเป็นการทำงานข้ามเครือข่ายนั้นทำงานผิดพลาดได้

เว็บเซอร์วิสเพื่อการคำนวณร้อยละของคุณค่าทางโภชนาการที่ต้องการในแต่ละวันนั้น มีการรับข้อมูลเข้ามา 2 อย่างด้วยกัน คือชื่อของสารอาหารที่ต้องการคำนวณซึ่งเป็น string และ ปริมาณของสารอาหารชนิดนั้นที่ต้องการคำนวณเปรียบเทียบซึ่งเป็น double แล้วส่งค่ากลับการทำงานเป็น double

เมื่อได้รับข้อมูลนำเข้ามาแล้ว จะทำการเปิด text file ที่เก็บข้อมูลมาตรฐานความต้องการของร่างกายที่มีต่อสารอาหารต่างๆ ในแต่ละวัน โดยที่ลักษณะของข้อมูลที่เก็บใน text file นี้มีลักษณะดังตัวอย่างเช่น

```
[NutrientPerDay Data]
Format :: NutrientName=StandardValue

Protein=54
lipid=510
Fat=510
Carbohydrate=510
Energy=2550
Calcium=500
...
```

โดยที่จะทำการวนรอบการทำงานตั้งแต่ต้นไฟล์ ไปจนถึงจุดสิ้นสุดไฟล์ เพื่อเปรียบเทียบว่ามีข้อมูลสารอาหารที่ต้องการคำนวณเปรียบเทียบนั้นบันทึกไว้หรือไม่ ซึ่งถ้าไม่มีบันทึกไว้ในไฟล์นี้ ค่าที่ได้จะถูกปรับให้เป็น 0 แต่ถ้าพบก็จะทำการดึงค่ามาตรฐานของสารนั้นมาคำนวณเปรียบเทียบเป็นร้อยละ กับค่าที่ถูกส่งเข้ามาคำนวณ จากนั้นก็จะส่งค่าที่ได้จากการคำนวณส่งกลับไปยังแอปพลิเคชันที่เรียกใช้คอมโพเนนต์นี้

เมื่อทำการเขียนโปรแกรมการทำงานหลักเสร็จเรียบร้อยแล้ว จึงทำการสร้างฟังก์ชันการทำงานที่สร้างขึ้นให้เป็น ActiveX DLL โดยคำสั่ง Make NutrientPerDay.dll ซึ่งเมื่อได้ไฟล์ .dll มาแล้วนั้นหมายถึงได้คอมโพเนนต์ที่สามารถทำการคำนวณร้อยละความต้องการในสารอาหารต่างๆ ในแต่ละวันแล้ว แต่คอมโพเนนต์ที่สร้างขึ้นนี้ ยังไม่สามารถให้บริการได้ในลักษณะเว็บเซอร์วิส ซึ่งต้องการสร้างเอกสาร WSDL ให้กับคอมโพเนนต์นี้เสียก่อน โดยใช้เครื่องมือ Microsoft SOAP Toolkit Version 3 : WSDL Generator

ซึ่งโปรแกรมจะให้ตั้งชื่อเซอร์วิสที่จะให้บริการ และเลือกคอมโพเนนต์ หรือไฟล์ .dll ที่จะให้บริการในลักษณะเว็บเซอร์วิส จากนั้นจะให้กำหนดลักษณะของไฟล์ ที่จะทำหน้าที่เป็น SOAP Listener ซึ่งจะมีลักษณะเป็น script ที่จะไปเรียกการทำงานของคอมโพเนนต์ที่ให้บริการ โดยสามารถเลือกได้ทั้งแบบ ASP และ ISAPI ซึ่งสุดท้ายก็จะให้กำหนดสถานที่ที่จะเก็บไฟล์ต่างๆ ที่ได้สร้างขึ้นเหล่านี้ โดยที่เพื่อความสะดวกในการจัดการและการทำงานอาจสร้างเป็น Directory ภายใน wwwroot ซึ่งเมื่อสร้างเอกสาร WSDL ขึ้นมาแล้ว จึงถือได้ว่า ActiveX DLL คอมโพเนนต์นี้มีลักษณะเป็นเว็บเซอร์วิส และสามารถให้บริการในลักษณะเว็บเซอร์วิสได้

#### การวิเคราะห์เอกสาร WSDL ของ NutrientPerDay

เพื่อให้การเรียกใช้เว็บเซอร์วิสที่สร้างขึ้นนี้เป็นไปอย่างถูกต้อง จึงต้องทำการเข้าใช้ส่วนต่างๆ ที่สำคัญในการเรียกใช้เว็บเซอร์วิส NutrientPerDay ดังนี้

- (1) ชื่อของเซอร์วิสที่ให้บริการและที่อยู่ของไฟล์ที่ทำหน้าที่เป็น SOAP Listener

```
<service name='NutrientPerDay' >
<port name='PercentageSoapPort' binding='wsdl:PercentageSoapBinding'>
<soap:address location='http://localhost/NutrientService/NutrientPerDay.ASP'>
</port>
</service>
```

- (2) ชื่อของ Operation หรือ Method ที่ให้บริการภายในเว็บเซอร์วิส

```
<operation name='getNutrientPercent' parameterOrder='NutrientName srcValue'>
<input message='wsdl:Percentage.getNutrientPercent'>
<output message='wsdl:Percentage.getNutrientPercentResponse'>
</operation>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(3) ส่วนข้อมูลที่ใช้มาเพื่อมาประมวลใน Method `getNutrientPercent` ซึ่งมีข้อมูลชื่อสารอาหารที่เป็น String และปริมาณสารอาหารที่ต้องการคำนวณ ซึ่งเป็น double

```
<message name='Percentage.getNutrientPercent'>
  <part name='NutrientName' type='xsd:string'/>
  <part name='srcValue' type='xsd:double'/>
</message>
```

(4) ส่วนข้อมูลที่ตอบกลับการทำงานจาก Method `getNutrientPercent` ซึ่งเป็น double

```
<message name='Percentage.getNutrientPercentResponse'>
  <part name='Result' type='xsd:double'/>
</message>
```

### ลักษณะของ SOAP Message

(SOAP Message ที่รับเข้า)

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
- <SOAP-ENV:Envelope xmlns:SOAPSDK1="http://www.w3.org/2001/XMLSchema"
xmlns:SOAPSDK2="http://www.w3.org/2001/XMLSchema-instance"
xmlns:SOAPSDK3="http://schemas.xmlsoap.org/soap/encoding/" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/">
- <SOAP-ENV:Body
  SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
- <SOAPSDK4:getNutrientPercent
  xmlns:SOAPSDK4="http://localhost/NutrientPerDay/message/">
  <NutrientName>Protein</NutrientName>
  <srcValue>58.784</srcValue>
</SOAPSDK4:getNutrientPercent>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

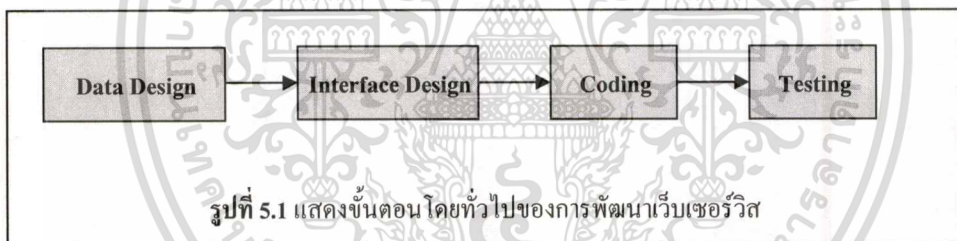
(SOAP Message ที่ตอบกลับ)

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
- <SOAP-ENV:Envelope xmlns:SOAPSDK1="http://www.w3.org/2001/XMLSchema"
xmlns:SOAPSDK2="http://www.w3.org/2001/XMLSchema-instance"
xmlns:SOAPSDK3="http://schemas.xmlsoap.org/soap/encoding/" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/">
- <SOAP-ENV:Body
  SOAP- ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
- <SOAPSDK4:getNutrientPercentResponse
  xmlns:SOAPSDK4="http://localhost/NutrientPerDay/message/">
  <Result>108.859259259259</Result>
</SOAPSDK4:getNutrientPercentResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

## บทที่ 5

### การพัฒนาแอปพลิเคชันเพื่อเรียกใช้เว็บเซอร์วิสเพื่อการคำนวณคุณค่าทางโภชนาการ

ในส่วนของการสร้างแอปพลิเคชันในลักษณะเพื่อไปเชื่อมต่อ เรียกใช้เว็บเซอร์วิสที่ได้ให้บริการเกี่ยวกับการคำนวณคุณค่าทางโภชนาการของอาหาร ซึ่งจะแยกเป็น 2 แอปพลิเคชันย่อยด้วยกันคือการประมวลผลคำนวณคุณค่าทางโภชนาการของอาหาร และเพื่อดูแลปรับปรุงข้อมูลคุณค่าทางโภชนาการของอาหาร โดยที่เทคนิคที่ใช้ในการพัฒนาทั้ง 2 แอปพลิเคชันนี้ เป็นในรูปแบบการพัฒนาเว็บแอปพลิเคชัน ซึ่งการพัฒนาเว็บแอปพลิเคชันเพื่อเรียกใช้เว็บเซอร์วิส นั้น ก็มีขั้นตอนการพัฒนาโดยทั่วไปคือ ขั้นตอนรวบรวมข้อมูลต่างๆ ที่จะใช้ในการเรียกเว็บเซอร์วิส ขั้นที่สองออกแบบส่วนและสร้างโครงสร้างเว็บที่จะติดต่อกับผู้ใช้ ต่อมาคือการเขียนโปรแกรมการทำงานรับข้อมูล เชื่อมต่อการทำงานกับเว็บเซอร์วิส และสุดท้ายคือการทดสอบการทำงานทั้งหมด



#### 5.1 Client Interface Application

เป็นเว็บแอปพลิเคชันที่ทำหน้าที่รับข้อมูลจากผู้ใช้ ตรวจสอบแล้วรวบรวมข้อมูลอาหารหรือวัตถุดิบที่จะใช้ปรุงอาหารทั้งหมด ส่งไปยังเว็บเซอร์วิสหลักในการคำนวณคุณค่าทางโภชนาการของอาหาร แล้วรอการตอบกลับหลังจากที่คำนวณเสร็จ แสดงผลข้อมูลคุณค่าทางโภชนาการของอาหารที่คำนวณได้ทั้งหมดแก่ผู้ใช้

##### เครื่องมือที่ใช้พัฒนา

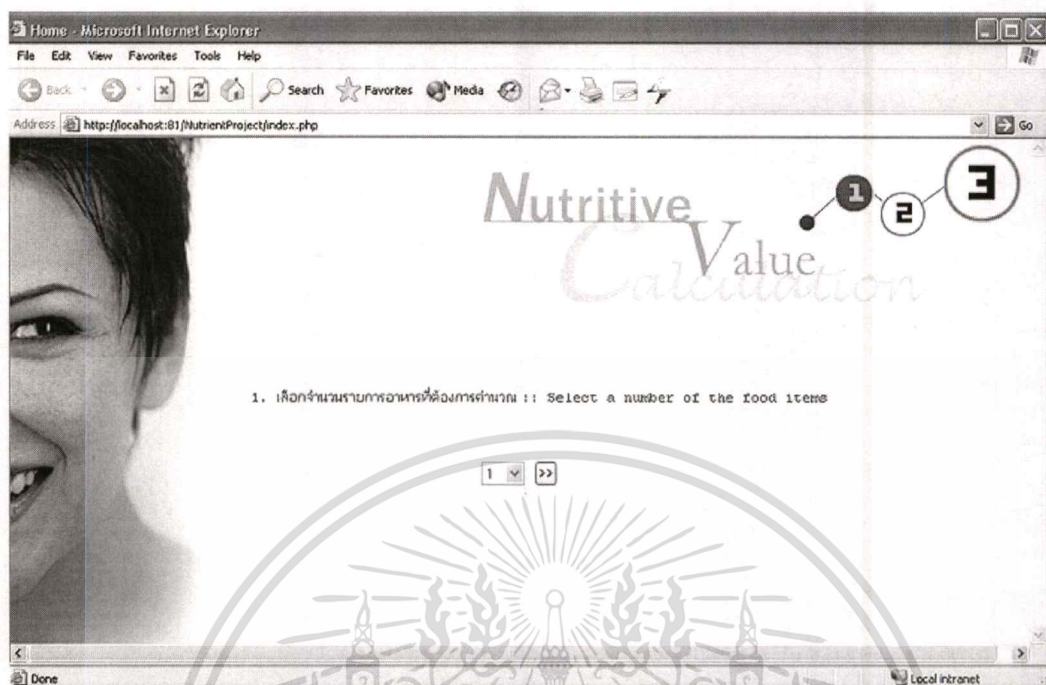
- PHP 4.2.3
- Apache Server 2.0.43

### ขั้นตอนในการพัฒนา

1. หลังจากทำการติดตั้ง PHP ให้กับ Apache แล้วจึงทำการสร้างเว็บเพจด้วยภาษา PHP ให้สามารถรับข้อมูลวัตถุดิบในการปรุงอาหารหรืออาหารที่ต้องการคำนวณ น้ำหนัก และหน่วยในการชั่งตวงน้ำหนักนั้นจากผู้ใ้
2. สร้าง Action การทำงานเพื่อเรียกเว็บเซอร์วิส แต่เนื่องจากในปัจจุบันการพัฒนาด้วย ภาษา PHP ให้มีความสามารถในการเรียกใช้เว็บเซอร์วิส นั้น ยังไม่มีประสิทธิภาพไม่เพียงพอ เช่นมีการสร้างสคริปต์ที่ใช้เชื่อมต่อกับเว็บเซอร์วิส ทำงานในลักษณะ SOAP Parser แต่การรับส่ง Parameter ที่มีความซับซ้อนนั้นทำได้ยาก ดังนั้นในการพัฒนานี้จึงเลือกใช้วิธีการอื่นที่จะทำให้ PHP สามารถทำงานเรียกใช้เว็บเซอร์วิสได้ ซึ่งปรกติแล้ว PHP นั้นจะมีความสามารถในการเรียกใช้ Component ไม่ว่าจะเป็น COM DLL ของ วินโดว์ หรือ Bean ของ Java ซึ่งการพัฒนานี้จะให้ PHP ติดต่อเรียกใช้ Component ในการเรียกใช้ SOAP ของวินโดว์
3. นำค่า Parameter ทั้งหมดที่จะให้เว็บเซอร์วิสประมวลผล ส่งไปยังคอมพิวเตอร์ ที่ทำงานในลักษณะการจัดการ SOAP Message ของวินโดว์ ซึ่งต้องมีการส่งค่า Parameter ที่เป็นที่อยู่ของเอกสาร WSDL ของเว็บเซอร์วิสการคำนวณคุณค่าทางโภชนาการให้คอมพิวเตอร์นั้นรู้จักด้วย แล้วนำค่าที่ตอบกลับมาจากการทำงาน แสดงผลต่อผู้ใ้
4. จัดรูปแบบเว็บเพจทั้งหมด แล้วทดสอบการทำงานทั้งระบบ

### รายละเอียดในการพัฒนา

เว็บแอปพลิเคชันเพื่อใช้ในการคำนวณคุณค่าทางโภชนาการของอาหาร การทำงานหลักจะเป็นการส่งข้อมูลไปยัง MainServices ของ Nutritive Value Calculation ซึ่งจากเอกสาร WSDL ที่ได้วิเคราะห์ในบทที่ผ่านมามองเห็นว่าข้อมูลนำเข้าของเว็บเซอร์วิสที่จะใช้ในการคำนวณคุณค่าทางโภชนาการของอาหารนี้มีลักษณะเป็น Array ของข้อมูล 3 อย่างด้วยกัน นั่นคือ ข้อมูลรายการอาหารหรือเครื่องปรุงที่ต้องการคำนวณ น้ำหนักของรายการอาหารแต่ละอย่างนั้น และหน่วยที่ใช้ชั่งตวงของรายการอาหารแต่ละรายการ ซึ่งเมื่อทราบว่าจะต้องส่งไปประมวลผลการทำงานนั้นเป็น Array เพื่อความสะดวกในการทำงานจึงควรมีการสอบถามจำนวนรายการอาหารที่ผู้ใ้ต้องการคำนวณเสียก่อน เพื่อจะได้สร้างแบบฟอร์มในการกรอกข้อมูลได้อย่างเหมาะสม



รูปที่ 5.2 แสดงการรับจำนวนรายการอาหารที่จะทำการคำนวณค่าทางโภชนาการ

การพัฒนาเว็บแอปพลิเคชันเพื่อใช้คำนวณคุณค่าทางโภชนาการของอาหารนี้ จะใช้ PHP เป็นภาษาหลักในการพัฒนา ซึ่ง PHP เป็นภาษาหนึ่งที่มีความสามารถในการทำงานร่วมกับภาษา HTML ได้เป็นอย่างดี โดยการเขียนสคริปต์ตามโครงสร้างภาษา PHP แทรกไปในเอกสาร HTML เมื่อเว็บเพจนั้นถูกเรียกขึ้นทำงาน ตัวแปรคำสั่งภาษา PHP ก็จะทำให้การแปลคำสั่งเหล่านั้นประมวลผลตามสคริปต์ที่ได้เขียนไว้ แล้วจัดให้อยู่ในรูปเอกสาร HTML เพื่อให้ Browser สามารถตีความและแสดงผลได้

การออกแบบส่วนติดต่อกับผู้ใช้ในเว็บแอปพลิเคชันนี้นั้น มีการสร้างเว็บเพจหลักๆ ขึ้น 3 หน้า และเว็บเพจย่อยส่วนขยายอีก 1 หน้า สำหรับหน้าแรกของการคำนวณคุณค่าทางโภชนาการนั้น จะให้มีการเลือกจำนวนรายการอาหารที่จะคำนวณ โดยจะมีการเขียนสคริปต์แทรกไว้บริเวณ `<option>` ภายใต้ `<select>` ซึ่งมีลักษณะเป็น combo box โดยที่ `<option>` นั้นจะเป็นค่าที่ที่จะปรากฏเป็นตัวเลือก ในกรณีนี้จะทำการวนรอบการทำงานสร้างตัวเลือกตั้งแต่ 1 ถึง 30 โดยอาศัยสมมติฐานที่ว่าโดยปรกติอาหารโดยทั่วไปนั้น มักมีเครื่องปรุงอาหารไม่เกิน 20 ชนิด ในที่นี้จึงได้กำหนดไว้อย่างมากให้มีได้ประมาณ 30 ชนิด ซึ่งน่าจะเป็นจำนวนที่เหมาะสมต่อการประมวลผลผ่านเครือข่ายและการแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการสร้างฟอร์มรับข้อมูล ลักษณะเว็บแอปพลิเคชันนี้จะมีวิธีการส่งข้อมูลไปประมวลผลอยู่ 2 แบบ นั่นคือ Post และ Get ซึ่งการส่งค่า parameter ในรูปแบบโครงสร้างข้อมูลที่ซับซ้อนไปยังเว็บเซิร์ฟเวอร์ มักจะยอมให้ส่งข้อมูลในลักษณะ Post เท่านั้น สำหรับการส่งค่าตัวแปรในภาษา PHP สามารถกำหนดชื่อให้กับ control ต่างๆ เมื่อผู้ใช้ได้ทำงานกำหนดค่าให้กับ control เหล่านั้นแล้วมีการ submit ข้อมูล ชื่อของ control เหล่านั้นจะเป็นเสมือนตัวแปรที่สามารถนำไปประมวลผลยังสคริปต์ภาษา PHP ในเว็บเพจต่อไปได้ทันที เช่นเดียวกันในหน้าเว็บแอปพลิเคชันหน้าแรกนี้ เมื่อมีการเลือกจำนวนรายการอาหาร แล้วทำการ submit ข้อมูลโดยการกดปุ่ม ค่าของ combo box จะถูกส่งไปเป็นตัวแปรเพื่อทำการประมวลผลสร้าง ฟอร์มในการป้อนข้อมูลในหน้าต่อไป



รูปที่ 5.3 แสดงการรับรายการอาหารที่จะทำการคำนวณคุณค่าทางโภชนาการ

เมื่อหน้าเว็บเพจที่ใช้สำหรับการป้อนข้อมูลรายการอาหารเพื่อส่งไปคำนวณยังเว็บเซิร์ฟเวอร์ การคำนวณคุณค่าทางโภชนาการถูกเรียกขึ้นทำงาน การสร้างแบบฟอร์มสำหรับป้อนข้อมูลจะนำค่าจากตัวแปร หรือ combo box ที่รับจำนวนรายการจากเว็บเพจก่อนหน้านี้ มาทำการวนloopสร้างช่องในการป้อนข้อมูลรายการอาหาร ปริมาณ และหน่วยในการชั่งตวงรายการอาหารนั้นๆ ซึ่งเมื่อผู้ใช้ป้อนรายการอาหารเรียบร้อยแล้ว การ submit ข้อมูลจะทำให้โดยการกดปุ่มด้านขวา เพื่อทำการประมวลผลต่อไป โดยที่ข้อมูลที่จะทำการส่งไปยังเว็บเพจหน้าต่อไปเพื่อส่งต่อไปคำนวณยังเว็บเซิร์ฟเวอร์การคำนวณคุณค่าทางโภชนาการ และแสดงผลการคำนวณนั้น มีดังนี้

- ชื่อของอาหารที่ต้องการคำนวณ เช่น เนื้อผัดน้ำมันหอย ข้าวผัด เป็นต้น ซึ่งเป็นส่วนที่ผู้ใช้จะป้อนหรือไม่ก็ได้ เนื่องจากไม่ได้ถูกรวมกับการคำนวณ
- จำนวนรายการอาหารที่ผู้ใช้ต้องการคำนวณ
- ชื่อของรายการอาหารทั้งหมดที่ต้องการคำนวณ ซึ่งเก็บไว้ในรูปแบบ Array
- ปริมาณหรือน้ำหนักของแต่ละรายการอาหาร ซึ่งอยู่ในรูปแบบ Array
- หน่วยที่ใช้ในการชั่งตวงน้ำหนักของแต่ละรายการอาหารเหล่านั้น ซึ่งอยู่ในรูปแบบ Array

โดยเมื่อข้อมูลตัวแปรต่างๆ ถูกส่งมายังเว็บเพจที่ใช้แสดงผลแล้ว จะมีการตรวจสอบตัวแปรต่างๆ ว่าถูกต้องหรือไม่ ตัวแปรที่จะถูกนำมาพิจารณาาก่อนคือ ชื่อของรายการอาหาร เนื่องจากในการคำนวณนั้น อย่างน้อยที่สุดต้องมีชื่อของรายการอาหารที่จะคำนวณ ซึ่งจะมีการเขียนสคริปต์เพื่อจัดเรียงข้อมูลใหม่เพื่อการส่งไปคำนวณจริง โดยจะรับเฉพาะรายการที่มีการป้อนข้อมูลชื่อรายการอาหารเท่านั้น และถ้าไม่มีการป้อนข้อมูล ซึ่งจะทำให้ Array มีขนาดเป็น 0 จะมีการบอกผู้ใช้ว่าพบข้อผิดพลาด ให้ตรวจสอบการป้อนข้อมูลใหม่อีกครั้ง



รูปที่ 5.4 แสดงการเตือนผู้ใช้เมื่อพบข้อผิดพลาดในการป้อนข้อมูลรายการอาหาร หรือไม่มีการป้อนรายการอาหาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนปริมาณของน้ำหนักของอาหารแต่ละรายการนั้นจะมีค่าเริ่มต้นเป็น 0 นั้นหมายถึง ถ้าไม่มีการป้อนข้อมูลปริมาณน้ำหนักหรือป้อนในรูปแบบที่ไม่ถูกต้องเช่น ตัวอักษร ค่าของปริมาณหรือน้ำหนักนั้นจะมีค่าเป็น 0

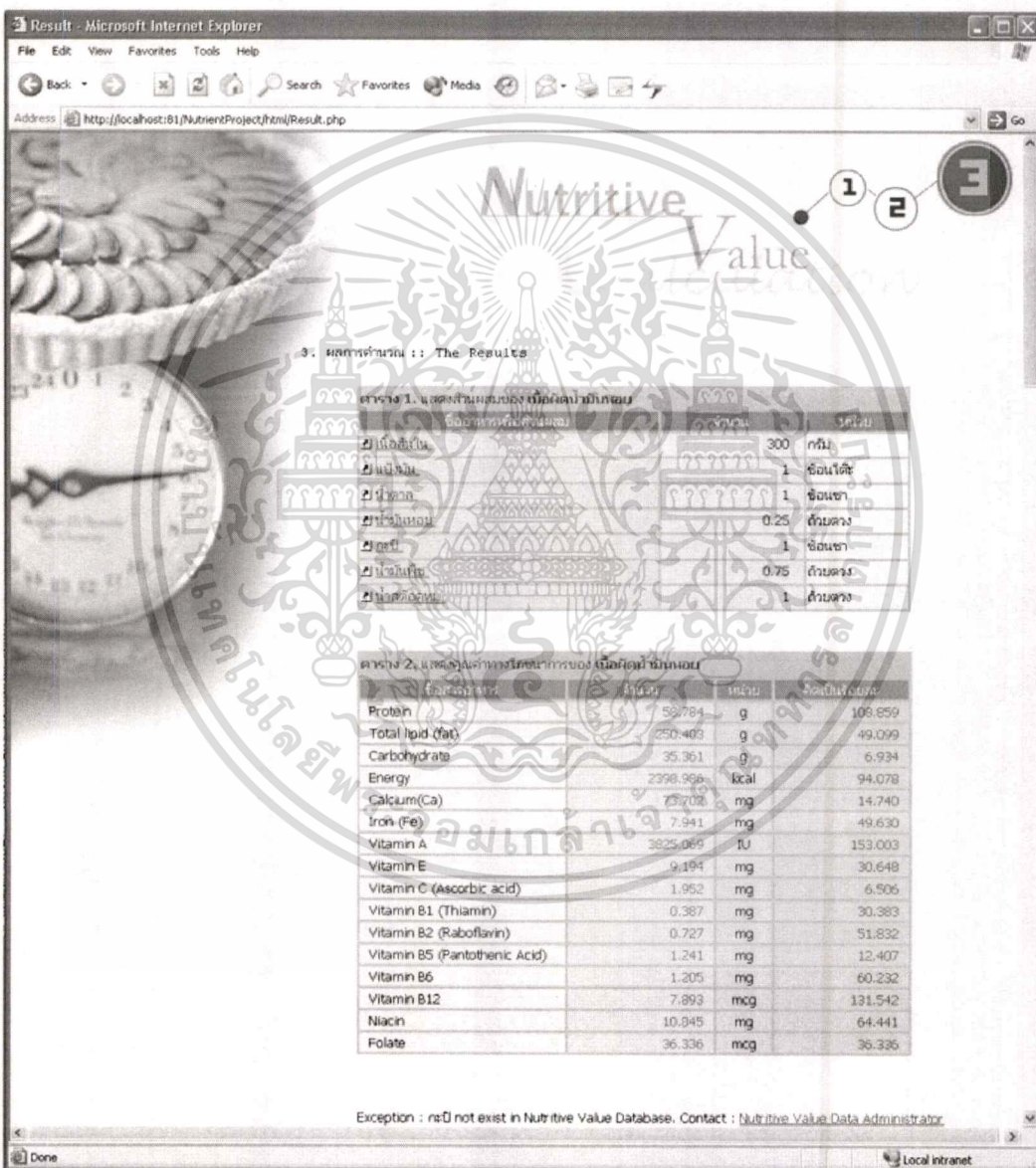
หลังจากได้ทำการรวบรวมข้อมูลรายการอาหารต่างๆ พร้อมทั้งจะทำการส่งข้อมูลแล้ว จะทำการแสดงรายการอาหารที่ผู้ใช้ป้อนและมีการตรวจสอบความถูกต้องต่างๆ เรียบร้อยแล้ว ซึ่งจะมีการสร้างเป็น Hypertext สำหรับรายการอาหารต่างๆ ที่มีปริมาณมากกว่า 0 ด้วย เพื่อที่หากผู้ใช้ต้องการทราบคุณค่าทางโภชนาการของอาหารในแต่ละรายการ แล้วจึงทำการส่งข้อมูลไปยังเว็บเซอร์วิส ซึ่งการส่งข้อมูลไปยังเว็บเซอร์วิสเพื่อการคำนวณคุณค่าทางโภชนาการ สามารถทำได้หลายวิธี โดยที่การทำงานหลักๆ คือการสร้าง SOAP Message เพื่อส่งค่า Parameters ต่างๆ ทำงานบน internet protocol เช่น HTTP ไปเรียกใช้เว็บเซอร์วิส Method ในการคำนวณคุณค่าทางโภชนาการ และสามารถอ่านค่า SOAP Message ที่ได้รับการตอบกลับการทำงานมา โดยที่วิธีการอาจจะเป็นการเรียกใช้สคริปต์ภาษา PHP ที่มีความสามารถดังที่ได้กล่าวมา แต่ยังมีวิธีการเรียกใช้ที่มีประสิทธิภาพและง่ายกว่านั้นคือ การใช้ความสามารถในการเรียกใช้คอมโปเนนต์ของภาษา PHP ไม่ว่าจะเป็น COM ของ Microsoft หรือ EJB ของ Sun โดยที่ใน Microsoft Windows จะมี COM หนึ่งที่มีความสามารถในการทำงานเกี่ยวกับ SOAP สำหรับติดต่อเรียกใช้ Web Services นั่นคือ MSSOAP ซึ่งเป็น COM ที่มากับ Microsoft Windows หรือสามารถเพิ่มได้ด้วยการติดตั้ง Microsoft SOAP Toolkit ซึ่งวิธีในการใช้งานเป็นดังนี้

```
<? php
$myCom = new COM("MSSOAP.SoapClient30");
$wsdl
"http://localhost/NutritiveValueCalculation/MainServices.asmx?WSDL";
$serviceName = "MainServices";
$myCom->msoapinit ($wsdl, $serviceName);
$o=$myCom->getNutrient ($FoodItem, $Value, $UnitName);
?>
```

เมื่อได้รับการตอบกลับการทำงานมาแล้วผลจะถูกเก็บอยู่ในรูปแบบ COM object หรือ Complex Type ตามโครงสร้าง SOAP Message ซึ่งถือว่าเป็นรูปแบบที่ซับซ้อนในการนำข้อมูลออกมาใช้ ดังนั้นต้องอาศัยความรู้โครงสร้างของเอกสาร XML ใน SOAP Message ที่ได้รับการตอบกลับมา โดยที่การใช้ข้อมูลในเอกสาร XML นั้น มีวิธีการเข้าถึงข้อมูลผ่าน API อยู่ 2 ชนิดนั่นคือ SAX และ DOM ซึ่ง การใช้ข้อมูลแบบ SAX นั้นจะเป็นการวิ่งเข้าไปดึงค่าเฉพาะส่วนที่ต้องการเท่านั้น ส่วน DOM นั้นจะเป็นการมองเอกสาร XML นั้นเป็น Tree ภายใน Tree ก็จะมี Node ข้อมูลต่างๆ ซึ่งการทำงานแบบ DOM จะได้ข้อมูลทั้งหมด แล้วเลือกใช้ข้อมูลภายใน Node ต่างๆ เอง ซึ่งสำหรับการคำนวณคุณค่าทางโภชนาการของอาหารนี้ ต้องการแสดงสิ่งที่ตอบกลับมาทั้งหมด จึงเลือกการนำออกมาแสดงผลตามลักษณะวิธีการของ DOM API ซึ่งมีลักษณะดังนี้

```
<? php
    $x=$o->Item(0);
    echo $x->nodeTypedValue();
?>
```

จาก Code ข้างต้นก่อนหน้าจะเป็นการนำค่าที่ได้จากการเรียกใช้เว็บเซอร์วิส เก็บไว้ในตัวแปร \$o จากนั้นจะทำการกำหนดค่าตัวแปร \$x เพื่อชี้ไปยัง Node หรือ Item ต่างๆ และแสดงผลค่าใน node ต่างๆ นั้นด้วยคำสั่ง nodeTypedValue ()



รูปที่ 5.5 แสดงผลการคำนวณคุณค่าทางโภชนาการของอาหาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งหน้าเว็บเพจที่ใช้แสดงผลการคำนวณนั้น จะแบ่งข้อมูลออกเป็น 3 ส่วนด้วยกัน ส่วนแรก จะแสดงข้อมูลที่ผู้ใช้ป้อนเข้ามา ซึ่งเป็นรายการอาหารที่ต้องการคำนวณ ซึ่งในแต่ละรายการอาหาร ที่มีปริมาณมากกว่า 0 นั้นสามารถเข้าไปดูรายละเอียดคุณค่าทางโภชนาการในแต่ละรายการได้ ส่วนที่สองจะเป็นผลการคำนวณคุณค่าทางโภชนาการที่ได้รับการตอบกลับมาจากเว็บเซอร์วิส ซึ่งด้านซ้ายจะเป็นจำนวนหรือปริมาณสารอาหารต่างๆ ที่มีในอาหารนั้น และด้านขวาจะเป็นข้อมูลแสดงปริมาณร้อยละของสารอาหารเมื่อเทียบกับความต้องการสารอาหารนั้นๆ ในแต่ละวัน ซึ่งถ้ามีปริมาณที่เกินความต้องการหรือเกิน 100 % จะมีการแสดงสีแดง เพื่อให้ผู้ใช้ สังเกตได้ง่าย และส่วนสุดท้ายจะเป็นส่วนที่จะบอกข้อมูลความผิดพลาดต่างๆ ที่เกิดขึ้นจากการคำนวณ เช่น รายการอาหารใดบ้างที่ไม่ได้รับการคำนวณ หรือมีเว็บเซอร์วิสใดที่เกี่ยวข้องกับการคำนวณไม่สามารถให้บริการได้บ้าง ทำให้ส่งผลกระทบต่อผลการคำนวณอย่างไร เป็นต้น

ตาราง แสดงคุณค่าทางโภชนาการของ เนื้อสับใน 300 กรัม

ชื่อสารอาหาร	จำนวน	หน่วย	คิดเป็นร้อยละ
Protein	53.280	g	98.667
Total lipid (fat)	69.060	g	13.541
Carbohydrate	0.000	g	0.000
Energy	849.000	kcal	38.294
Calcium(Ca)	21.000	mg	4.200
Iron (Fe)	6.870	mg	42.938
Vitamin A	0.000	IU	0.000
Vitamin E	0.000	mg	0.000
Vitamin C (Ascorbic acid)	0.000	mg	0.000
Vitamin B1 (Thiamin)	0.330	mg	25.882
Vitamin B2 (Raboflavin)	0.600	mg	42.781
Vitamin B5 (Pantothenic Acid)	0.900	mg	9.000
Vitamin B6	1.110	mg	55.500
Vitamin B12	7.650	mcg	127.500
Niacin	8.790	mg	52.228
Folate	18.000	mcg	18.000

Exception : -

close

รูปที่ 5.6 แสดงผลการคำนวณคุณค่าทางโภชนาการของรายการอาหารที่ต้องการทราบเฉพาะรายการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ItemResult - Microsoft Internet Explorer

ตาราง แสดงคุณค่าทางโภชนาการของ กล้วย 1 ช้อนชา

ชื่อสารอาหาร	จำนวน	หน่วย	คิดเป็นร้อยละ
Protein	0.000	g	0.000
Total lipid (fat)	0.000	g	0.000
Carbohydrate	0.000	g	0.000
Energy	0.000	kcal	0.000
Calcium(Ca)	0.000	mg	0.000
Iron (Fe)	0.000	mg	0.000
Vitamin A	0.000	IU	0.000
Vitamin E	0.000	mg	0.000
Vitamin C (Ascorbic acid)	0.000	mg	0.000
Vitamin B1 (Thiamin)	0.000	mg	0.000
Vitamin B2 (Raboflavin)	0.000	mg	0.000
Vitamin B5 (Pantothenic Acid)	0.000	mg	0.000
Vitamin B6	0.000	mg	0.000
Vitamin B12	0.000	mcg	0.000
Niacin	0.000	mg	0.000
Folate	0.000	mcg	0.000

Exception : nuti not exist in Nutritive Value Database. Contact : Nutritive Value Data Administrator

close

รูปที่ 5.7 แสดงผลการคำนวณเฉพาะรายการซึ่งเป็นรายการที่ไม่มีในฐานข้อมูล

## 5.2 Data Admin Application

เป็นแอปพลิเคชัน ที่ทำหน้าที่ดูแลรายการข้อมูลคุณค่าทางโภชนาการของอาหารต่างๆ สามารถปรับปรุง เพิ่ม และลบรายการออกจากฐานข้อมูลได้ ซึ่งจำเป็นต้องได้รับการยืนยันสิทธิ์ในการแก้ไขจากระบบก่อน ด้วยการตรวจสอบ Username และ password

### เครื่องมือที่ใช้พัฒนา

- Microsoft Visual Studio.NET
- Microsoft Internet Information Services 5.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ขั้นตอนในการพัฒนา

1. สร้างโปรเจกใหม่โดยเลือกรูปแบบ C# ASP.NET Web Application
2. สร้างฟอร์มป้อน Username, Password และการทำงานในส่วนการ ตรวจสอบสิทธิ์การเข้าใช้
3. สร้างฟอร์มเพื่อทำการติดต่อกับผู้ใช้ เพื่อรับการป้อนข้อมูลรายการอาหาร และปริมาณของสารอาหารที่มีในรายการอาหารนั้นๆ
4. เขียนโปรแกรมการทำงานเพื่อเรียกใช้เว็บเซอร์วิสที่ใช้ในการปรับปรุงข้อมูลคุณค่าทางโภชนาการ
5. แสดงผลการปรับปรุงข้อมูลต่อผู้ใช้
6. ทดสอบการทำงาน

### รายละเอียดในการพัฒนา

เว็บแอปพลิเคชันเพื่อเรียกใช้เว็บเซอร์วิสเพื่อการปรับปรุงข้อมูลคุณค่าทางโภชนาการนี้ ถูกพัฒนาขึ้นตามรูปแบบ ASP.NET ซึ่งใน .NET Framework นั้น ASP.NET สามารถทำงานได้โดยมีภาษาที่ใช้ในการเขียนโปรแกรมทำหน้าที่เป็นสคริปต์แทรกอยู่ในเอกสาร HTML หรือ ทำงานในลักษณะ code behind กล่าวคือจะมองเห็นหน้าเว็บเพจในลักษณะเช่นเดียวกับวินโดว์ฟอร์ม ซึ่งเรียกว่าเว็บฟอร์ม แล้วมีการเขียน โปรแกรมการทำงานให้กับ Event หรือเหตุการณ์ต่างๆ เมื่อมีการดำเนินการใดๆ กับฟอร์มนั้นอยู่ภายใน ซึ่งภาษาที่ใช้ในการเขียนโปรแกรมสำหรับ ASP.NET นี้ สามารถเลือกได้ 2 ภาษา นั่นคือ C# และ VB.NET

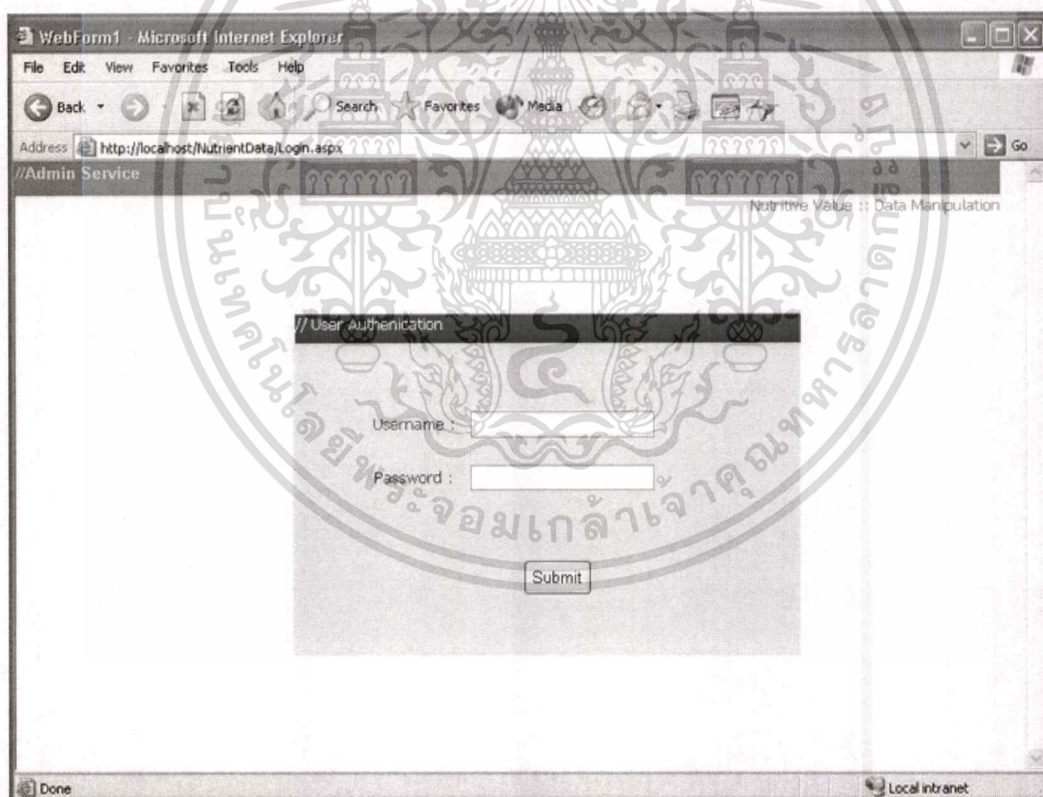
การพัฒนาเว็บแอปพลิเคชันนี้เริ่มต้นด้วยการเลือกภาษาที่ใช้ในการพัฒนาเป็น C# และเลือกรูปแบบ ASP.NET Web Application โดยทำงานในลักษณะ code behind และเนื่องจากเว็บแอปพลิเคชันนี้จะเป็นการรับข้อมูลจากผู้ใช้ผ่านเว็บฟอร์ม แล้วทำการรวบรวมข้อมูลส่งไปประมวลผลยังเว็บเซอร์วิสเพื่อการปรับปรุงข้อมูลคุณค่าทางโภชนาการ ซึ่งการพัฒนาด้วย Microsoft Visual Studio.NET นั้น การเรียกใช้เว็บเซอร์วิสต้องทำการ add Web Reference โดยค้นหาจากที่อยู่ของเอกสาร WSDL

การทำงานเริ่มด้วยการป้อนข้อมูล Username และ Password เพื่อเก็บไว้รอการ Implement ในส่วน Header ของ SOAP Message เนื่องจากตามคุณลักษณะของเว็บเซอร์วิส NutrientDataAdmin นั้น บางเว็บเซอร์วิส Method ที่เกี่ยวกับการปรับปรุงข้อมูลในฐานข้อมูลคุณค่าทางโภชนาการ จำเป็นต้องมีการ Authentication ก่อนเข้าใช้งาน Method ซึ่งเมื่อผู้ใช้ป้อนข้อมูล

Username และ Password แล้วจะมีการส่งข้อมูลเก็บไว้ในตัวแปรส่วน Session ซึ่งต้องมีการกำหนดค่าเริ่มต้นไว้ก่อนในไฟล์ global.asax ซึ่งมีลักษณะดังนี้

```
protected void Session_Start(Object sender, EventArgs e)
{
    Session["username"] = "";
    Session["password"] = "";
    Session.Timeout=60;
}
```

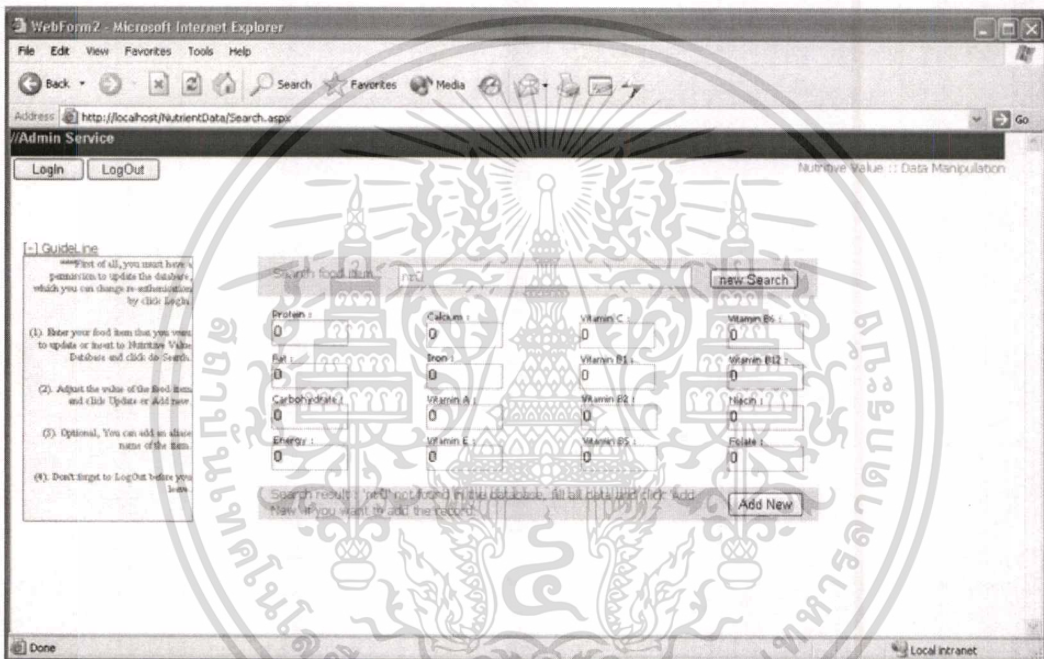
เมื่อผู้ใช้เข้ามาใช้งานเว็บแอปพลิเคชัน จะมีการสร้าง Session ให้กับผู้ใช้คนนั้นตลอดการใช้งาน ซึ่งจะทำการกำหนดค่าเริ่มต้นให้กับตัวแปร username และ password นอกจากนั้นยังมีการกำหนด Timeout ให้กับ session ไว้ นั่นหมายถึงว่า ตัวแปรต่างๆ ที่อยู่ภายใน session จะมีอายุการใช้งานในระยะเวลา 60 นาที ซึ่งถ้าผู้ใช้ทำการ Authentication ไว้แล้วมีการปรับปรุงข้อมูลเกินระยะเวลานั้น ต้องทำการ Authentication ใหม่อีกครั้ง



รูปที่ 5.8 แสดงเว็บเพจหน้าแรกสำหรับป้อนข้อมูล username และ password

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อผู้ใช้ submit ข้อมูล username และ password เรียบร้อยแล้ว จะเป็นการเรียกเว็บเพจที่ใช้ในการค้นหาและปรับปรุงข้อมูลคุณค่าทางโภชนาการของอาหารขึ้นมาทำงาน โดยที่ผู้ใช้จำเป็นต้องทำการค้นหาชื่อของรายการอาหารที่จะเพิ่ม หรือปรับปรุงในฐานข้อมูลก่อน ซึ่งชื่อรายการอาหารที่รับเข้ามานี้ เมื่อมีการกดปุ่ม search จะทำการเรียกใช้การทำงานเว็บเซอว์วิธ Method searchNutrient เพื่อดูว่ามีรายการอาหารนี้ในฐานข้อมูลหรือไม่ ถ้าไม่พบรายการ จะนำค่าสารอาหารต่างๆ ซึ่งเป็น 0 ไล่ลงไปในช่วงสำหรับการป้อนข้อมูลสารอาหาร แล้วตั้ง caption ของปุ่มที่ใช้ในการปรับปรุงข้อมูลเป็น “Add New”



รูปที่ 5.9 แสดงเว็บเพจสำหรับค้นหาข้อมูลเพื่อปรับปรุงรายการ และไม่ปรากฏข้อมูลที่ค้นหาในฐานข้อมูล

แต่ถ้าพบรายการที่ค้นหาในฐานข้อมูล ก็จะแสดงผลปริมาณ สารอาหารต่างๆ แล้วเปลี่ยน caption ของปุ่มเพื่อใช้ในการปรับปรุงข้อมูลเป็น “Update” นอกจากนี้จะมีการแสดง ฟอรั่มที่ใช้ในการป้อนข้อมูลเพิ่มเติม ซึ่งเป็นส่วนของการปรับปรุงข้อมูลชื่อเรียกของรายการอาหารนั้น ว่าสามารถเรียกได้อะไรบ้าง ซึ่งจะเป็นการนำชื่อของรายการอาหารนั้น ไปเรียกเว็บเซอว์วิธ Method getAliasItemName แล้วนำค่าที่ค้นกลับมาได้นั้น ซึ่งเป็น DataSet กลับมาแสดงใน Listbox ได้ทันที ซึ่งมีลักษณะดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 5.10 แสดงผลค้นหาข้อมูลเพื่อปรับปรุงรายการ

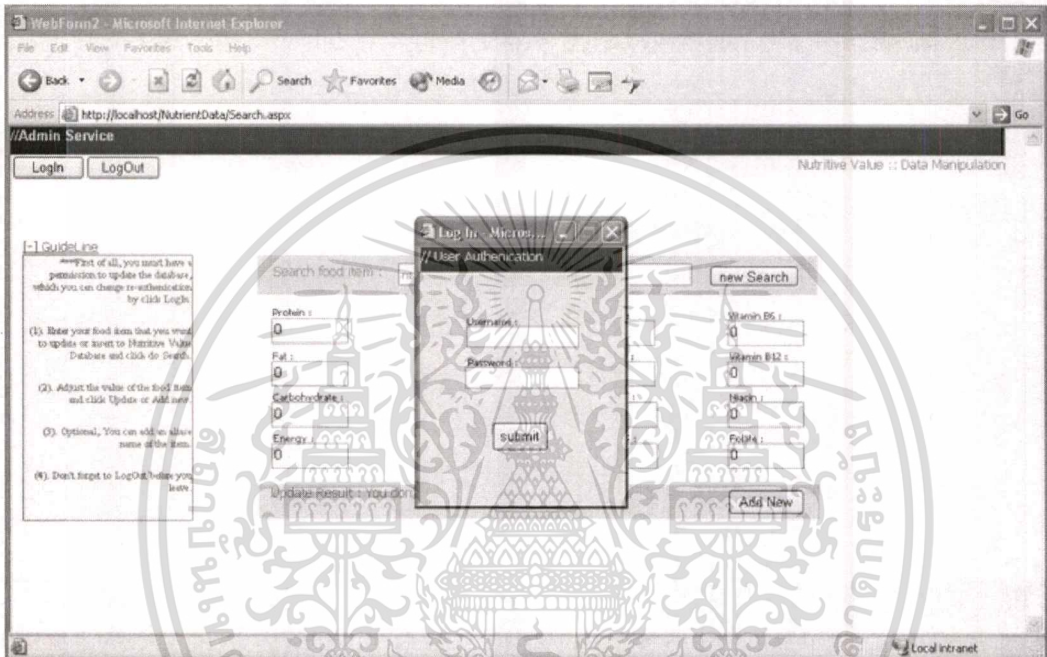
```
public DataSet ds = new DataSet();
ds=NDA.getAliasItemName(txtFoodName.Text.Trim());
ListAliasName.DataSource=ds;
ListAliasName.DataValueField="AliasItemName";
ListAliasName.DataBind();
```

ถ้าผู้ใช้ต้องการปรับปรุงข้อมูล หรือเพิ่มข้อมูลใหม่เข้าไปในฐานข้อมูลคุณค่าทางโภชนาการ การทำงานเริ่มด้วยกดปุ่ม Update หรือ Add New ซึ่งจะเป็นการเรียกใช้เว็บเซอร์วิส Method updateFoodItem ซึ่งการทำงานภายในจะทำการรวบรวมข้อมูลต่างๆ จัดให้อยู่ในรูปแบบเดียวกับ class NutrientData ตามลักษณะการเรียกใช้ Method นี้ นอกจากนั้น ต้องทำการกำหนดค่า Header ของ SOAP Message โดยทำการดึงค่า username และ password มาจากตัวแปร Session ซึ่งมีลักษณะดังนี้

```
WebReference.NutrientDataAdmin NDA = new WebReference.NutrientDataAdmin();
WebReference.AuthHeader authen=new WebReference.AuthHeader();
authen.Username=Session["username"].ToString();
authen.Password=Session["password"].ToString();
NDA.AuthHeaderValue=authen;
```

เมื่อรวบรวมข้อมูล Parameters ต่างๆ ที่ใช้ในการเรียกการทำงาน Method updateNutrient เรียบร้อยแล้ว ก็จะส่งค่า Parameter เหล่านั้นให้เว็บเซอร์วิส Method ที่เรียกใช้ทำงาน แล้วจะมีการ

ส่งผลการปรับปรุงข้อมูลกลับมาบอกยังผู้ใช้ ว่าสามารถปรับปรุงข้อมูลได้สำเร็จหรือไม่ โดยที่หากมีการตรวจพบว่าผู้ใช้ ไม่ได้เป็นผู้ที่มีสิทธิ์ในการเข้าปรับปรุงข้อมูล หรือมีการใส่ username และ password ที่ไม่ถูกต้อง เว็บเซอว์วิจจะแจ้งกลับมาบอกว่า ผู้ใช้เป็นผู้ไม่มีสิทธิ์ที่จะเข้าใช้ Method นี้ ซึ่งต้องทำการ Authentication ใหม่ ให้เป็นผู้ใช้ที่มีสิทธิ์ที่ถูกต้องจึงจะสามารถปรับปรุงข้อมูลได้ โดยผู้ใช้สามารถทำการ Authentication ใหม่อีกครั้ง ได้โดยกดปุ่ม LogIn แล้วจะมีหน้าต่าง pop up ขึ้นมาเพื่อทำการปรับปรุงค่าของ username และ password



รูปที่ 5.11 แสดง pop up window เพื่อทำการปรับปรุงข้อมูล Authentication



รูปที่ 5.12 แสดง pop up window หลังจาก submit ข้อมูล Authentication แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งถ้าเป็นผู้ใช้ที่มีสิทธิ์ปรับปรุงข้อมูล การทำงานเว็บเซอร์วิส Method ก็จะสามารถทำงานได้สำเร็จ โดยจะมีการบอกกับผู้ใช้ว่า สามารถปรับปรุงข้อมูลได้เรียบร้อยแล้ว ซึ่งจะมีลักษณะคล้ายรูป 5.10 ซึ่งมีการแสดงข้อมูลชื่อของรายการอาหารนั้น ที่สามารถเรียกเป็นอย่างอื่นได้ หรือ AliasName ซึ่งผู้ใช้สามารถเพิ่มข้อมูลในส่วนนี้ได้ โดยการป้อนข้อมูลในช่องสำหรับป้อนข้อมูล AliasName ใหม่สำหรับรายการอาหารนั้นๆ แล้วทำการคลิกปุ่ม >> ซึ่งการทำงานภายใน จะเป็นการนำชื่อที่ผู้ใช้ป้อนเข้ามา ส่งไปยังเว็บเซอร์วิส Method updateAliasName เพื่อทำการเพิ่มข้อมูลการเรียกชื่อรายการอาหารนั้น ซึ่งเช่นเดียวกันต้องมีการ Implement ส่วน Header เพื่อการ Authentication ให้กับ SOAP Message ด้วย และเมื่อเว็บเซอร์วิสนั้นได้ดำเนินการตามที่ร้องขอเรียบร้อยแล้วจะทำการส่งข้อความกลับมาบอกผลการทำงานกับผู้ใช้ โดยที่หากปรับปรุงข้อมูลสำเร็จก็จะมีการเรียกเว็บเซอร์วิส Method getAliasItemName ใหม่อีกครั้งเพื่อทำการปรับปรุงรายการใน ListBox ให้ผู้ใช้เห็นว่าได้มีการเพิ่มรายการการเรียกชื่อใหม่นั้นให้กับฐานข้อมูลแล้ว และส่วนสุดท้ายคือการ Log out เมื่อผู้ใช้คลิกปุ่ม LogOut จะทำการล้างข้อมูลตัวแปรใน Session ทั้งหมด แล้วเรียกเว็บเพจหน้าแรก (รูปที่ 5.8) ขึ้นมารองรับการทำงานจากผู้ใช้อีกครั้ง

ซึ่งจากการพัฒนาเว็บแอปพลิเคชันทั้งหมดจะเห็นว่า เว็บแอปพลิเคชันที่ได้ทำการพัฒนานี้จะเป็นเพียงส่วนที่ใช้เชื่อมต่อกับผู้ใช้เพื่อทำการรับข้อมูลและแสดงผลเท่านั้น ส่วนการทำงานประมวลผลต่างๆ นั้นจะเป็นหน้าที่ของเว็บเซอร์วิสทั้งหมด ซึ่งจะเป็นการแยก ส่วน Processing กับ ส่วน Presentation ออกจากกันอย่างแท้จริง

## บทที่ 6

### บทสรุปและข้อคิดเห็น

เว็บเซอร์วิส เป็นแนวคิดใหม่ในการพัฒนาแอปพลิเคชัน โดยเปลี่ยนมุมมองการพัฒนาให้เป็นในลักษณะ Service Oriented กล่าวคือ ในแต่ละหน่วยงานธุรกิจที่มีแอปพลิเคชัน คอมโพเนนต์เป็นของตัวเอง หรือมี Business Logic ของตัวเองที่เป็นประโยชน์ ต้องมีการคิดว่า เราจะสามารถให้บริการอะไรได้บ้าง ซึ่งการสร้างแอปพลิเคชันในลักษณะเว็บเซอร์วิส จะสร้างมูลค่าเพิ่มให้กับการพัฒนาที่ธุรกิจต้องลงทุนไปอยู่แล้ว อาจเป็นการสร้างรายได้เพิ่มให้กับธุรกิจ หรือเป็นการสร้างความร่วมมือในระดับองค์กร ใช้ทรัพยากรที่เป็นประโยชน์ร่วมกัน

ในส่วนของเทคโนโลยีนั้น เว็บเซอร์วิสได้ถูกออกแบบมาให้มีความยืดหยุ่นอย่างมาก เนื่องจากจะพยายามมองทุกอย่างให้เป็นกลาง ไม่ยึดติดกับผลิตภัณฑ์หรือเทคโนโลยีของผู้ค้ารายใดรายหนึ่งเท่านั้น ไม่ว่าจะคอมโพเนนต์ที่จะให้บริการนั้นอยู่บนรูปแบบ Platform ใด สามารถให้บริการ และใช้งานร่วมกันได้ด้วยมาตรฐานเดียวกัน ซึ่งมีพื้นฐานมาจากเทคโนโลยีภาษา XML ทั้งหมด โดยผู้ที่ต้องการใช้งานเว็บเซอร์วิส ไม่จำเป็นต้องรู้ว่าคอมโพเนนต์ที่จะใช้บริการนั้นอยู่ที่ใด พัฒนามาด้วยภาษาใด หรืออยู่ในระบบปฏิบัติการใด เพียงแต่ต้องรู้ว่า Interface ของเว็บเซอร์วิสคอมโพเนนต์นี้ต้องการ Parameters ใดบ้าง ในการส่งข้อมูลเข้ามาประมวลผล และเว็บเซอร์วิสจะตอบกลับการทำงานเหล่านั้นอย่างไร นอกจากนี้ยังมีความยืดหยุ่น สำหรับการพัฒนา กล่าวคือถ้ามีการเปลี่ยนแปลงกระบวนการทำงานภายใน หรือมีการเปลี่ยนไปใช้ผลิตภัณฑ์ของผู้ค้ารายอื่น เพื่อให้คอมโพเนนต์ที่ให้บริการมีประสิทธิภาพมากขึ้น การกระทำเหล่านั้นจะไม่ส่งผลกระทบต่อ Interface หากไม่มีการเปลี่ยนแปลงรูปแบบการส่งข้อมูลเข้าและออกจากเว็บเซอร์วิสที่ให้บริการ นั่นหมายถึงจะไม่ส่งผลกระทบต่อไปยังแอปพลิเคชัน หรือคอมโพเนนต์อื่นที่ทำการเรียกใช้เว็บเซอร์วิสนี้ด้วย

เทคโนโลยีพื้นฐานที่ทำให้เว็บเซอร์วิสสามารถทำให้เกิดการทำงานร่วมกันระหว่างคอมโพเนนต์ต่างๆ ได้นั้นมีเทคโนโลยีหลักอยู่ 3 อย่าง คือ SOAP เป็นแกนหลักในการสื่อสาร ส่งข้อมูลระหว่างคอมโพเนนต์ WSDL เป็นเอกสารที่ใช้อธิบายลักษณะของ Interface ของคอมโพเนนต์นั้นว่าสามารถเรียกใช้ได้อย่างไร และจะตอบกลับมาอย่างไร และ UDDI ซึ่งใช้สำหรับการค้นหาเว็บเซอร์วิสที่มีให้บริการ โดยที่ทั้ง 3 เทคโนโลยีนี้มีพื้นฐานมาจากมาตรฐานตามโครงสร้างภาษา XML เหมือนกัน

ดังที่ได้กล่าวมาข้างต้นจะเห็นว่า เว็บเซอร์วิสเป็นแนวโน้มเทคโนโลยีที่มีประโยชน์ ต่อการพัฒนาแอปพลิเคชันในอนาคตเป็นอย่างมาก จึงเป็นการดีที่จะศึกษาวิธีการในการทำงานและพัฒนาเว็บเซอร์วิสอย่างจริงจัง ซึ่งในเอกสารนี้จะเป็นการนำเอาการคำนวณคุณค่าทางโภชนาการของอาหารขึ้นมาเป็นกรณีศึกษาในการพัฒนาเว็บเซอร์วิสและการทำงานร่วมกันของเว็บเซอร์วิส ภายใต้สภาวะในการพัฒนาที่ต่างกัน

การคำนวณหาคุณค่าทางโภชนาการของอาหารนั้น เป็นสิ่งที่จะช่วยให้ผู้บริโภคสามารถดูแลภาวะทางโภชนาการของตนเองได้อย่างสะดวกมากขึ้น ในลักษณะของเว็บเซอร์วิสนั้น ผู้ให้บริการจะทำการเตรียม Component ที่มีลักษณะเว็บเซอร์วิสไว้คอยให้บริการ ผู้ที่ต้องการใช้บริการเพียงแค่สร้าง Application ที่มีการส่งข้อมูลตามรูปแบบที่กำหนด ซึ่งจะเป็น SOAP Message หรือข้อความที่มีลักษณะเป็นเอกสาร XML ไปร้องขอการบริการจากผู้ให้บริการให้ดำเนินการคำนวณให้ โดยที่การออกแบบพัฒนาเว็บเซอร์วิสเพื่อการคำนวณคุณค่าทางโภชนาการของอาหารนี้นั้นใช้การออกแบบในเชิงวัตถุ โดยใช้ UML เป็นตัวแบบ ซึ่งแสดงการทำงานและความสัมพันธ์ต่างๆ ด้วย Use Case Diagram, Class Diagram และ Sequence Diagram ตามลำดับ และในส่วนของการพัฒนานั้นมีการใช้เทคโนโลยีที่หลากหลาย ในการสร้างคอมโพเนนต์ ไม่ว่าจะเป็นในรูปแบบ EJB, ActiveX หรือ .NET แล้วมีการเรียกใช้เว็บเซอร์วิสเหล่านั้น ด้วยเว็บแอปพลิเคชัน

ซึ่งจากการพัฒนาพบว่า ในเบื้องต้นนั้น เว็บเซอร์วิสสามารถทำงานร่วมกันได้จริง โดยที่ไม่ยึดติดกับรูปแบบของการพัฒนาใดๆ แต่ผู้ใช้งานเว็บเซอร์วิส จำเป็นที่จะต้องมีความรู้เบื้องต้นในเรื่องเกี่ยวกับเว็บเซอร์วิส ไม่ว่าจะเป็นการอ่านข้อมูลที่อยู่ในเอกสาร WSDL เพื่อให้การเรียกใช้เว็บเซอร์วิสทำได้ถูกต้อง หรือต้องเข้าใจในเทคโนโลยี XML เพื่อให้รู้และเข้าใจวิธีการนำข้อมูลออกมาใช้ ในกรณีที่ไม่สามารถแสดงผลได้ตามปรกติ เนื่องจากแตกต่าง และความซับซ้อนของข้อมูลที่ส่งมาภายใน SOAP Message

วันนี้เว็บเซอร์วิส อาจจะยังไม่สามารถตอบสนองความต้องการของธุรกิจได้อย่างเต็มที่ เนื่องจากเป็นเทคโนโลยีที่อยู่ในระยะเริ่มต้น และยังมีข้อกำหนดที่ควรได้รับการแก้ไขอีกมาก แต่ก็ถือว่าเป็นเทคโนโลยีที่มีแนวโน้มที่ดี เนื่องจากผู้ผลิตซอฟต์แวร์รายใหญ่ของโลกต่างให้ความสนใจและมีการกำหนดมาตรฐานในการพัฒนาร่วมกัน แม้ว่าในรายละเอียดปลีกย่อยบางส่วนนั้น เริ่มมีการกำหนดเป็นรูปแบบเฉพาะของตนแล้ว เช่นการทำงานในรูปแบบ Transaction หรือมาตรฐานด้านความปลอดภัย ซึ่งสิ่งเหล่านี้ อาจก่อให้เกิดความขัดแย้งและไม่สามารถทำงานร่วมกันได้ในอนาคต ฉะนั้นเว็บเซอร์วิสจะสามารถเป็นสิ่งที่ เป็นมาตรฐานและสามารถใช้งานร่วมกันได้จริง ก็ต่อเมื่อที่ผู้คิดค้นเทคโนโลยีแต่ละราย ยังยึดถือในมาตรฐานเดียวกัน และยังคงร่วมมือในการกำหนดแนวทางของเทคโนโลยีนี้ร่วมกัน

## บรรณานุกรม

- กรมอนามัย. 2535. ตารางแสดงคุณค่าทางโภชนาการของอาหารไทย. กรุงเทพฯ: กองโภชนาการ กรมอนามัย กระทรวงสาธารณสุข.
- กิตติชัย พินิจคำ. 2544. การเขียนโปรแกรม XML และ SOAP สำหรับ BizTalk Server. กรุงเทพฯ: สามย่าน.COM
- ไพศาล โมลิสกุลมงคล, น.ท. 2545. Microsoft Visual C# .NET. กรุงเทพฯ: ดวงกลมสมัย.
- สราวุธ อ้อยศรีสกุล. 2544. ถอดรหัส .net + Web Services. กรุงเทพฯ: วิดีดี กรู๊ป.
- เสาวนีย์ จักรพิทักษ์. 2544. หลักโภชนาการปัจจุบัน. กรุงเทพฯ: ไทยวัฒนาพานิชย์.
- Border, W. and Dumser, J. 2000. SOAP: Simple Object Access Protocol. [Online]. Available: <http://www.intranetjournal.com/articles/200012/>.
- Chester, T. 2001. "Cross-platform integration with XML and SOAP." IEEE IT Pro. September-October: 26-34.
- Czerveny, A. 2001. Web Services. [Presentation]. Summa Technologies.
- Eriksson, H. and Penker, M. 1998. UML Toolkit. Wiley Computer.
- Forte, S. 2002. "Consume Web Services Asynchronously in VS.NET." ACCESS-VB-SQL ADVISOR. November: 20-25.
- Snell, J. et al. 2002. Programming Web Service with SOAP. California: O'reilly.
- Sturm, J. 1999. VB6 UML Design and Development. Birmingham: Wrox.
- U.S. Department of Agriculture. 2002. USDA Nutrient Database for Standard Reference. Release 15. Agricultural Research Service.
- Weaver, R. 2002. Creating Web Services with WebSphere Studio Application Developer. [Presentation]. IBM.

## ประวัติผู้เขียน

นายเลอชาติ ธรรมวีระเสถียร

เกิดเมื่อวันที่ 23 พฤศจิกายน พ.ศ. 2521

ภูมิลำเนา กรุงเทพมหานคร

ประวัติการศึกษา

- พ.ศ. 2543 จบการศึกษาระดับปริญญาตรี วิทยาการสารสนเทศบัณฑิต (ระบบสารสนเทศเพื่อการจัดการ) เกียรตินิยมอันดับสอง จาก มหาวิทยาลัยเทคโนโลยีสุรนารี



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้