

คํานักหลสมดกลาง พระจอมเกล้าลาดกระบัง

ระบบให้บริการระบุตำแหน่ง โดยใช้ภาษาเอ็กซ์เอ็มแอลและเว็บเซอร์วิส  
Location-based Services using XML and Web Services



นาย ฐานันตร์ พงศ์ภัทรวิฒน  
นาย ประภักดิ์ รุ่งเรืองอนันต์

เลขหมู่.....  
เลขทะเบียน..... 62115  
วัน,เดือน,ปี 31 ก.ค. 2549

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

ปริญญาบัตรนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
ภาควิชาวิศวกรรมคอมพิวเตอร์  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบให้บริการระบุตำแหน่ง โดยใช้ภาษาเอ็กซ์เอ็มแอลและเว็บเซอร์วิส  
Location-based Services using XML and Web Services



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
ภาควิชาวิศวกรรมคอมพิวเตอร์  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ ปีการศึกษา 2547

ภาควิชาวิศวกรรมคอมพิวเตอร์


คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบให้บริการระบุตำแหน่ง โดยใช้ภาษาเอ็กซ์เอ็มแอลและเว็บเซอร์วิส  
Location-based Services using XML and Web Services

คณะผู้จัดทำ

1. นาย ฐานันตร์ พงศ์ภัทรวัฒน์ รหัสนักศึกษา 44010125
2. นาย ประภักดิ์ รุ่งเรืองอนันต์ รหัสนักศึกษา 44010284



  
อาจารย์ที่ปรึกษา  
(ผศ. ดร. วิศิษฎ์ หิรัญกิตติ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ระบบให้บริการระบุตำแหน่ง โดยใช้ภาษาเอ็กซ์เอ็มแอลและเว็บเซอร์วิส

นาย ฐานันตร์ พงศ์ภัทรวัฒน์	44010125
นาย ประภักดิ์ รุ่งเรืองอนันต์	44010284
ผศ.ดร. วิศิษฎ์ หิรัญกิตติ	อาจารย์ที่ปรึกษา

### บทคัดย่อ

ปฏิญานิพนธ์ฉบับนี้นำเสนอการพัฒนาแผนที่ด้วยการออกแบบและสร้างระบบการให้บริการด้วยเทคโนโลยีของเว็บเซอร์วิส (Web Services) และ ออกแบบและสร้างฐานข้อมูลด้วยฐานข้อมูลในเชิงพื้นที่ (Spatial Database) ซึ่งในการพัฒนาแผนที่นั้นข้อมูลแผนที่ถูกออกแบบให้อยู่ในรูปแบบของภาษา GML (Geography Markup Language) เมื่อต้องการจะแสดงผลแผนที่ก็จะทำการแปลงข้อมูลจากแผนที่ในภาษา GML ให้ไปอยู่ในรูปแบบอื่นๆ ที่เหมาะแก่การแสดงผลมากกว่า เช่น แปลงให้อยู่ในรูปแบบของภาษา SVG (Scalable Vector Graphics) หรือแปลงให้อยู่ในรูปแบบของรูปภาพ JPEG (Joint Photographic Experts Group) ซึ่งการแปลงให้ไปอยู่ในรูปแบบภาษา SVG นั้นจะทำให้มีความสามารถในการปฏิสัมพันธ์กับผู้ใช้มากขึ้น โดยการใช้ความสามารถของภาษาจาวาสคริปต์เข้าไปทำงานร่วมกับ SVG ส่วนการแสดงผลในรูปแบบของรูปภาพ JPEG นั้นจะทำให้สามารถแสดงผลได้ในทุกๆ อุปกรณ์แสดงผล เพราะอุปกรณ์แสดงผลบางอย่างไม่สามารถที่จะแสดงผลของ SVG ได้

เทคโนโลยีที่นำมาใช้ในการพัฒนาระบบให้บริการแผนที่ในโครงการนี้ ประกอบด้วย XML, GML, SVG, JavaScript, Java Server Page, Spatial Database และ Web Services ทำให้แผนที่นี้สามารถทำงานได้กับเว็บทุก Platform และเทคโนโลยีจาวาทำให้ระบบนี้สามารถใช้งานได้บนทุกระบบปฏิบัติการ ดังนั้นจึงเป็นวิธีการที่เหมาะสมที่จะนำไปใช้ในการพัฒนาระบบให้บริการแผนที่ต่อไปในอนาคต

## Location-based Services using XML and Web Services

Thanan Pongpatwatana	44010125
Prapat Rungruenganan	44010284
Visit Hirankitti	Advisor

### ABSTRACT

This thesis proposed a development of a system providing location-based services using several technologies. Firstly, the system provided the services using Web Services technology. Secondly, the map data is represented logically in GML (Geography Markup Language) whereas the raw map data is stored in spatial databases and managed by a spatial database management system. Since the map data represented in GML cannot be displayed, this is mainly because GML is not designed for the display purpose but only to describe map data, we have to employ a presentation method in order to display the GML content. SVG (Scalable Vector Graphics Markup Language) and JPEG (Joint Photographic Experts Group) have been selected for this purpose. SVG contents can be embedded with JavaScript codes in order to manipulate the SVG data dynamically. The result of this is a more interactive map presentation in the form of SVG codes embedded with some JavaScript codes. For the other map presentation, maps in form JPEG can be easily viewed by any web browser.

The technologies adopted in this work are XML, GML, SVG, JavaScript, Java Server Page, Spatial Database, and Web Services. With the cross-platform map data represented in the open markup languages and the development of the system based on the Java platform, our map services can be made available in all of platforms. This approach is relevant for building a system providing location-based services in the future.

### กิตติกรรมประกาศ

ปริญญาบัตรฉบับนี้คงไม่อาจสำเร็จได้ด้วยดี หากไม่ได้รับความช่วยเหลือ และความร่วมมือ จากหลายๆ ฝ่ายด้วยกัน บุคคลแรกที่จะขาดเสียไม่ได้ในความสำเร็จดังกล่าวนี้ คือ อาจารย์ วิศิษฎ์ หิรัญ กิตติ อาจารย์ที่ปรึกษาปริญญาบัตรที่不僅แต่ตลอดเวลาเท่านั้น แต่ยังสละแรงกายและแรงใจในการ เอาใจใส่ดูแล แนะนำและช่วยเหลือเสมอมาจนปริญญาบัตรฉบับนี้สำเร็จลุล่วงด้วยดี อีกทั้งคุณาจารย์ ทุกท่านที่ถ่ายทอดวิชาความรู้มาให้ ซึ่งในความกรุณาของอาจารย์ที่มีต่อขอขอบคุณเป็นอย่างสูง

และต้องขอขอบคุณบุคคลสำคัญที่สุดที่ทำให้วันนี้ ก็คือ บิดา มารดา อันเป็นที่เคารพรักยิ่ง ซึ่งได้เลี้ยงดูมาเป็นอย่างดี พร้อมทั้งให้โอกาสในการศึกษาอย่างเต็มที่ และยังให้กำลังใจ เอาใจใส่เสมอ มาในทุกๆ ด้าน อันที่หาที่เปรียบมิได้ ข้าพเจ้าขอระลึกในพระคุณ และขอกราบขอขอบคุณมา ณ ที่นี้

ฐานันตร์ พงศ์ภัทรวัฒน์  
ประภัสร์ รุ่งเรืองอนันต์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

	หน้าที่
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VII
สารบัญรูปภาพ	VIII
บทที่ 1 บทนำ	
1.1 ความเป็นมาและความสำคัญ	1
1.2 วัตถุประสงค์ของปริญญานิพนธ์	1
1.3 ขอบเขตของปริญญานิพนธ์	2
1.4 วิธีการดำเนินงาน	2
บทที่ 2 ภาษา Markup Language ที่เกี่ยวข้องกับการพัฒนาแผนที่	
2.1 ภาษา Markup Language ที่เกี่ยวข้องกับการพัฒนาแผนที่	4
2.2 ภาษา XML (Extensible Markup Language)	4
2.2.1 ลักษณะโครงสร้างของ XML	4
2.2.1.1 Tag	5
2.2.1.2 Element	6
2.2.1.3 Content	6
2.2.1.4 Attribute	7
2.2.2 ตัวกำหนดโครงสร้างของ XML ด้วย DTD (Document Type Declaration)	7
2.2.2.1 รูปแบบของ DTD	7
2.2.3 การแปลงรูปแบบเอกสาร XML (Transforming XML)	9
2.2.3.1 The XSL Transformation Language (XSLT)	9
2.2.4 DOM (Document Object Model)	10
2.3 ภาษา GML (Geography Markup Language)	10
2.3.1 ข้อดีของการใช้ GML ในการแสดงแผนที่	10
2.3.2 GML Basic concepts	11
2.3.3 Encoding Geometry	13
2.3.3.1 GML Feature Collections	13
2.3.3.2 GML application schema	14
2.4 ภาษา SVG (Scalable Vector Graphics Language)	15
2.4.1 การสร้างภาพโดยใช้ภาษา SVG	15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ(ต่อ)

	หน้าที่
2.4.2 การแสดงภาพโดยใช้ภาษา SVG บนเบราว์เซอร์	17
2.5 จาวาสคริปต์ (JavaScript)	18
<b>บทที่ 3 Spatial Database System</b>	
3.1 ฐานข้อมูลเชิงพื้นที่ (Spatial database system)	19
3.2 Model of spatial information	19
3.3 Spatial Data Type	20
3.4 Spatial Relationships	21
3.5 ขั้นตอนการออกแบบฐานข้อมูลเชิงพื้นที่	21
3.6 ฐานข้อมูลเชิงพื้นที่กับข้อมูลในรูปแบบ WKT	22
<b>บทที่ 4 Web Services</b>	
4.1 เทคโนโลยีเว็บเซอร์วิส (Web Services)	23
4.2 องค์ประกอบของเว็บเซอร์วิส	23
4.3 ขั้นตอนการเรียกใช้บริการของเว็บเซอร์วิส	24
4.4 มาตรฐานที่ใช้ในการพัฒนาเว็บเซอร์วิส	25
4.4.1 SOAP (Simple Object Access Protocol)	25
4.4.1.1 โครงสร้างของ SOAP	26
4.4.1.2 Element ต่างๆ ของ SOAP	27
4.4.2 WSDL (Web Services Description Language)	29
4.4.2.1 โครงสร้างเอกสาร WSDL	30
4.4.3 UDDI (Universal Description and Discovery Integration)	31
4.4.5 ตัวอย่างมาตรฐานที่ใช้ในเว็บเซอร์วิส	31
<b>บทที่ 5 หลักการออกแบบ</b>	
5.1 บทนำ	34
5.2 โครงสร้างและการทำงานของระบบระบุตำแหน่งบนเว็บเบราว์เซอร์	34
5.3 การทำงานติดต่อกันของ เว็บเซอร์วิส เว็บเซิร์ฟเวอร์ และ เครื่องไคลเอนท์	35
5.4 การออกแบบระบบในส่วนต่างๆ ของระบบระบุตำแหน่ง	36
5.4.1 การออกแบบในส่วนของเว็บเซอร์วิส	36
5.4.1.1 เว็บเซอร์วิสที่ให้บริการแผนที่ (Map Services)	36
5.4.1.1.1 การออกแบบฐานข้อมูลแผนที่	36
5.4.1.1.2 ชนิดของข้อมูลในระบบจัดการฐานข้อมูลเชิงพื้นที่ MySQL Server	36
5.4.1.1.3 รูปแบบของฟังก์ชัน (Function) ที่ใช้ในการจัดเก็บข้อมูล	37
5.4.1.1.4 เริ่มต้นออกแบบและสร้างฐานข้อมูล	37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ(ต่อ)

	หน้าที่
5.4.1.1.5 การพัฒนาแผนที่โดยใช้ภาษา GML	45
5.4.1.1.5.1 การสร้างไฟล์ GML	45
5.4.1.1.6 การสร้างไฟล์ SVG	46
5.4.1.1.6.1 การเพิ่มความสามารถของแผนที่โดยใช้ภาษา SVG DOM	46
5.4.1.1.7 การออกแบบบริการต่างๆ	47
5.4.1.2 เว็บเซอร์วิสที่ให้บริการข้อมูลบุคคล (Yellow Page Services)	49
5.4.1.2.1 การออกแบบฐานข้อมูลแผนที่	49
5.4.1.2.2 การออกแบบบริการต่างๆ	51
5.4.2 การออกแบบในส่วนของเว็บเซิร์ฟเวอร์	53
บทที่ 6 ผลการทำลอง	
6.1 การใช้งาน	55
6.2 ผลการทดลองใช้งานระบบให้บริการแผนที่บนเว็บ	56
6.2.1 เมื่อเปิดเข้าสู่หน้าแรกของระบบ	56
6.2.2 ทดลองใช้ Map Tool	56
6.2.3 การทดลองใช้บริการต่างๆ	61
บทที่ 7 บทวิจารณ์และสรุป	
7.1 บทวิจารณ์และสรุป	69
7.2 ปัญหาที่เกิดขึ้นในการพัฒนาและแนวทางการแก้ปัญหา	69
7.3 ข้อจำกัดของระบบ	70
7.4 แนวทางการพัฒนา	70
ภาคผนวก ก. การติดตั้ง Apache Axis	71
ภาคผนวก ข. การแปลงข้อมูลแผนที่ให้อยู่ในรูปแบบ SVG	74
ภาคผนวก ค. การแปลงแผนที่ให้อยู่ในรูปแบบ JPEG	77
บรรณานุกรม	82

## สารบัญตาราง

	หน้าที่
ตารางที่ 2-1 SVG Element	16
ตารางที่ 2-2 SVG Attribute	17
ตารางที่ 4-1 Fault Element	29
ตารางที่ 4-2 Fault code Sub Element	29
ตารางที่ 5-1 เขตข้อมูลและคำอธิบายของตาราง geom.	37
ตารางที่ 5-2 เขตข้อมูลและคำอธิบายของตาราง Hospital	38
ตารางที่ 5-3 เขตข้อมูลและคำอธิบายของตาราง Police	39
ตารางที่ 5-4 เขตข้อมูลและคำอธิบายของตาราง Bank	40
ตารางที่ 5-5 เขตข้อมูลและคำอธิบายของตาราง Mall	40
ตารางที่ 5-6 เขตข้อมูลและคำอธิบายของตาราง Restaurant	41
ตารางที่ 5-7 เขตข้อมูลและคำอธิบายของตาราง TourPoint	42
ตารางที่ 5-8 เขตข้อมูลและคำอธิบายของตาราง add_place	43
ตารางที่ 5-9 เขตข้อมูลและคำอธิบายของตาราง user	44
ตารางที่ 5-10 ความสัมพันธ์ของเขตของข้อมูลจากตาราง Hospital และ GML	46
ตารางที่ 5-11 เขตข้อมูลและคำอธิบายของตาราง Yellow_Page	49
ตารางที่ 5-12 เขตข้อมูลและคำอธิบายของตาราง add_person	50
ตารางที่ 5-13 เขตข้อมูลและคำอธิบายของตาราง user	51

## สารบัญรูปภาพ

หน้าที่

รูปที่ 2-1 การใช้งาน DOM	10
รูปที่ 2-2 The geometry model for simple feature (Open GIS Consortium, 2001)	12
รูปที่ 2-3 Features and properties	12
รูปที่ 2-4 Point	13
รูปที่ 2-5 LineString	14
รูปที่ 2-6 Polygon	14
รูปที่ 3-1 โครงสร้างพื้นฐานที่ใช้ใน Object-Based Model	20
รูปที่ 3-2 โครงสร้างพื้นฐานที่ใช้ใน Field-Based Model	20
รูปที่ 3-3 โครงสร้างและความสัมพันธ์ภายในของ Spatial geometry ในรูปแบบของ UML	21
รูปที่ 3-4 ตัวอย่างข้อมูลคิบบของฐานข้อมูลเชิงพื้นที่ใน MySQL Server	22
รูปที่ 3-5 ตัวอย่างข้อมูลในรูปแบบ WKT ของฐานข้อมูลเชิงพื้นที่	22
รูปที่ 4-1 การทำงานของ Web Services	23
รูปที่ 4-2 องค์ประกอบของ Web Services	24
รูปที่ 4-3 ขั้นตอนการเรียกใช้บริการของ Web Services	25
รูปที่ 4-4 ขั้นตอนการส่ง SOAP message	26
รูปที่ 4-5 โครงสร้างของเอกสาร SOAP	27
รูปที่ 5-1 โครงสร้างการทำงานของระบบระบุตำแหน่งบนเว็บเบราว์เซอร์	34
รูปที่ 5-2 การติดต่อกันของ Web Services, Server และ Client	35
รูปที่ 5-3 ชนิดของข้อมูลและ Primary key ของตาราง geom.	38
รูปที่ 5-4 ชนิดของข้อมูลและ Primary key ของตาราง Hospital	38
รูปที่ 5-5 ชนิดของข้อมูลและ Primary key ของตาราง Police	39
รูปที่ 5-6 ชนิดของข้อมูลและ Primary key ของตาราง Bank	40
รูปที่ 5-7 ชนิดของข้อมูลและ Primary key ของตาราง Mall	41
รูปที่ 5-8 ชนิดของข้อมูลและ Primary key ของตาราง Restaurant	42
รูปที่ 5-9 ชนิดของข้อมูลและ Primary key ของตาราง Tourpoint	43
รูปที่ 5-10 ชนิดของข้อมูลและ Primary key ของตาราง add_place	44
รูปที่ 5-11 ชนิดของข้อมูลและ Primary key ของตาราง user	44
รูปที่ 5-12 ชนิดของข้อมูลและ Primary key ของตาราง Yellow_Page	50
รูปที่ 5-13 ชนิดของข้อมูลและ Primary key ของตาราง add_person	51
รูปที่ 5-14 ชนิดของข้อมูลและ Primary key ของตาราง user	51
รูปที่ 6-1 เว็บเพจของระบบให้บริการแผนที่	56
รูปที่ 6-2 ภาพแผนที่ก่อนการเลื่อนแผนที่ใน Overview Map	57

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป(ต่อ)

	หน้าที่
รูปที่ 6-3 ภาพแผนที่หลังการเลื่อนแผนที่ใน Overview Map	57
รูปที่ 6-4 ภาพแผนที่หลังทำการขยาย	58
รูปที่ 6-5 ภาพแผนที่หลังการคลิก Pan Tools เลื่อนภาพไปทางขวา	59
รูปที่ 6-6 ภาพแผนที่ที่ไม่มีแสดงเลขเอร์ของ Main Road	60
รูปที่ 6-7 ภาพแผนที่หลังจากการหมุนแผนที่ทวนเข็มนาฬิกา	60
รูปที่ 6-8 หน้าเว็บเพจที่แสดงการขอบริการค้นหาสถานที่	61
รูปที่ 6-9 หน้าเว็บเพจหลังจากขอบริการค้นหาสถานที่	62
รูปที่ 6-10 ตัวอย่างผลลัพธ์ของเอกสารในรูปแบบ GML	62
รูปที่ 6-11 ตัวอย่างผลลัพธ์ของเอกสารในรูปแบบ SVG	63
รูปที่ 6-12 ตัวอย่างผลลัพธ์ของเอกสารในรูปแบบ JPG	63
รูปที่ 6-13 หน้าเว็บเพจที่แสดงการขอบริการเพิ่มข้อมูลสถานที่	64
รูปที่ 6-14 หน้าเว็บเพจหลังจากขอบริการเพิ่มข้อมูลสถานที่	64
รูปที่ 6-15 หน้าเว็บเพจที่แสดงการขอบริการแก้ไขข้อมูลสถานที่	65
รูปที่ 6-16 หน้าเว็บเพจหลังจากขอบริการแก้ไขสถานที่	65
รูปที่ 6-17 หน้าเว็บเพจที่แสดงการขอบริการค้นหาบุคคล	66
รูปที่ 6-18 หน้าเว็บเพจหลังจากขอบริการค้นหาบุคคล	66
รูปที่ 6-19 หน้าเว็บเพจที่แสดงการขอบริการเพิ่มข้อมูลบุคคล	67
รูปที่ 6-20 หน้าเว็บเพจหลังจากขอบริการเพิ่มข้อมูลบุคคล	67
รูปที่ 6-21 หน้าเว็บเพจที่แสดงการขอบริการแก้ไขข้อมูลบุคคล	68
รูปที่ 6-22 หน้าเว็บเพจหลังจากขอบริการแก้ไขข้อมูลบุคคล	68

## บทที่ 1

### บทนำ

#### 1.1 ความเป็นมาแล้วความสำคัญ

กิจกรรมต่างๆ ในชีวิตประจำวัน ในการทำงานและในการดำเนินงานในด้านธุรกิจนั้น จำเป็นที่จะต้องอาศัยการติดต่อสื่อสาร รวมไปถึงการเดินทาง ในปัจจุบันด้วยความก้าวหน้าทางเทคโนโลยีการสื่อสาร ได้แก่ เทคโนโลยีการติดต่อสื่อสารบนอินเทอร์เน็ต เทคโนโลยีโทรศัพท์เคลื่อนที่ และเทคโนโลยีคอมพิวเตอร์แบบพกพา ทำให้การติดต่อสื่อสารมีความสะดวกรวดเร็ว การติดต่อทางธุรกิจมีความคล่องตัวแต่อย่างไรก็ตาม สิ่งที่เราขาดไม่ได้คือการเดินทาง เนื่องจากผู้คนจำเป็นต้องมีการพบปะหรือ การประชุมรวมกระทั่งการขนส่งสินค้า ดังนั้นการพัฒนาเทคโนโลยีที่จะช่วยให้การเดินทางมีความสะดวกรวดเร็ว จึงเป็นสิ่งที่สำคัญไม่ยิ่งหย่อนไปกว่าการสื่อสาร

องค์ประกอบสำคัญประการหนึ่งของการเดินทางคือ แผนที่ โดยที่แผนที่จะเป็นแหล่งเก็บข้อมูลของตำแหน่ง ถนน ตรอก ซอย สถานที่ต่างๆ ฯลฯ ซึ่งมีปริมาณข้อมูลเป็นจำนวนมากทำให้ยากแก่การค้นหา และนอกจากนี้ข้อมูลยังมีการเปลี่ยนแปลงอยู่ตลอดเวลา ทำให้แผนที่กระดาษไม่อำนวยความสะดวกกับผู้ใช้และไม่สามารถรองรับการพัฒนาการเดินทางในอนาคตซึ่งต้องการความสะดวกรวดเร็ว และต้องการข้อมูลที่มีความถูกต้องทันสมัยอยู่ตลอดเวลาได้

ปัจจุบันได้มีการพัฒนาแผนที่คอมพิวเตอร์ หรือที่เรียกว่า “แผนที่ดิจิทัล” ขึ้น และเริ่มมีความนิยมใช้งานกันเพิ่มมากขึ้น แผนที่ดิจิทัลนั้นสามารถให้ข้อมูลของ ถนน สถานที่ ฯลฯ สามารถค้นหาข้อมูลตำแหน่งของถนน หรือสถานที่ โดยบอกเพียงชื่อ ถนน หรือสถานที่ และอนุญาตให้ผู้ใช้สามารถเพิ่มเติมสถานที่เข้าไปในแผนที่ได้

และในปัจจุบันได้มีเทคโนโลยีในการติดต่อสื่อสารบนอินเทอร์เน็ตที่มีความสามารถในการพัฒนาโดยไม่คำนึงถึงภาษาที่จะใช้ในการพัฒนา และสามารถใช้งานได้โดยไม่คำนึงถึง Platform ที่ใช้อยู่ เทคโนโลยีการสื่อสารนั้นคือ “เทคโนโลยีของ Web Services” ซึ่งมีผลทำให้การติดต่อสื่อสารนั้นสามารถทำได้อย่างในวงกว้างกว่าอดีตที่ ต้องติดอยู่กับภาษาและ Platform ที่ใช้

ดังนั้น เมื่อนำเอาเทคโนโลยีการติดต่อสื่อสาร Web Services มาพัฒนาเพื่อให้บริการแผนที่ดิจิทัล จึงทำให้การใช้งานแผนที่นั้น สามารถทำได้ในหลายๆ อุปกรณ์ อาทิเช่น การใช้งานแผนที่บนเว็บเบราว์เซอร์ การใช้งานแผนที่บนโทรศัพท์มือถือ การใช้งานแผนที่บนคอมพิวเตอร์แบบพกพา เป็นต้น

#### 1.2 วัตถุประสงค์ของปริญญานิพนธ์

1. เพื่อพัฒนาระบบที่จะให้ข้อมูลแผนที่และ การเดินทางบนเว็บเบราว์เซอร์
2. เพื่อศึกษาและนำเอาเทคโนโลยีภาษา XML มาใช้สร้างระบบให้บริการแผนที่ที่สามารถให้บริการได้ในหลาย Platform

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เพื่อศึกษาและนำเอาเทคโนโลยีของ Spatial Database มาใช้เป็นระบบฐานข้อมูลของแผนที่เพื่อให้มีการเก็บข้อมูลแผนที่ที่เป็นมาตรฐาน ทำให้สามารถนำมาแสดงผลได้อย่างถูกต้อง
4. เพื่อศึกษาและนำเอาเทคโนโลยีของ Web Services มาใช้เซอร์วิสเพื่อให้บริการแก่ผู้ร้องขอ ซึ่งการใช้เทคโนโลยี Web Services นั้นสามารถพัฒนาและติดต่อกันได้ในหลายภาษาและหลาย Platform
5. เพื่อเป็นการบริการให้กับนักท่องเที่ยวและบุคคลทั่วไปที่ต้องเดินทางอยู่ตลอดเวลา ทำให้มีความสะดวกสบายในการเดินทาง และช่วยลดการสูญเสียเวลาจากการหลงทางอีกด้วย
6. พัฒนาแผนที่สมัยใหม่ให้มีความสามารถในการ Interactive กับผู้ใช้งานมากขึ้น เพื่อสะดวกในการใช้งาน และเพื่อช่วยในการค้นหาตำแหน่งต่างๆ ในแผนที่

### 1.3 ขอบเขตของปริญญาณิพนธ์

ปริญญาณิพนธ์นี้มุ่งเน้นการพัฒนาาระบบระบุตำแหน่งแผนที่ที่สามารถทำงานได้ ดังนี้

1. แสดงแผนที่กรุงเทพในตำแหน่งต่างๆ พร้อมทั้งสถานที่ที่อยู่ในบริเวณนั้นๆ
2. ค้นหาตำแหน่งของสถานที่ที่ต้องการ เพื่อแสดงแผนที่ในตำแหน่งของสถานที่นั้น และบริเวณใกล้เคียงด้วย
3. ให้ข้อมูลเกี่ยวกับรายละเอียดของสถานที่ต่างๆ ที่ต้องการ อาทิเช่น หมายเลขโทรศัพท์ เวลาทำการของสถานที่นั้นๆ หรือ สิ่งที่จะเชื่อมต่อไปยังหน้าเว็บของสถานที่นั้นๆ เป็นต้น
4. ค้นหาตำแหน่งของสถานที่ที่เกี่ยวข้องกับบุคคลที่ต้องการ พร้อมทั้งรายละเอียดของบุคคลที่ต้องการ อาทิเช่น ที่อยู่ เบอร์โทรศัพท์มือถือ เป็นต้น

### 1.4 วิธีการดำเนินงาน

1. ศึกษาเทคโนโลยีและ การทำงานของระบบในโครงการเก่า
2. ศึกษาเทคโนโลยีที่เกี่ยวข้องและจำเป็น สำหรับการนำไปใช้ในโครงการปัจจุบัน
3. ทดลองนำแผนที่ที่ได้รับมา มาแปลงให้อยู่ในรูปแบบของภาษา GML
4. ทดลองแสดงผลแผนที่โดยใช้ภาษา SVG พร้อมทั้งเขียนฟังก์ชันการทำงานที่ใช้ในการแสดงผลของ SVG ด้วย JavaScript
5. ออกแบบฐานข้อมูลจากข้อมูลแผนที่ที่ได้รับมา พร้อมทั้งข้อมูลเกี่ยวกับสถานที่ต่างๆ โดยใช้หลักการของ Spatial Database
6. ทดลองเขียนโปรแกรมทำการติดต่อระหว่างกับฐานข้อมูลแผนที่ที่เป็น Spatial Database
7. ออกแบบบริการต่างๆ ของเว็บเซอร์วิส พร้อมทั้งออกแบบฟังก์ชันการทำงานต่างๆ ของเว็บเซอร์วิส
8. ทดลองเขียนฟังก์ชันที่มีการติดต่อกันระหว่าง เว็บเซอร์วิสกับระบบฐานข้อมูล
9. ทดลองเขียนหน้าเว็บ เพื่อร้องขอใช้บริการของเว็บเซอร์วิส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10. ออกแบบหน้าเว็บต่างๆ ของเว็บเซิร์ฟเวอร์ที่มีการติดต่อกันระหว่างเว็บเซิร์ฟเวอร์และเว็บเซอร์วิส
11. ทดลองการทำงานที่เว็บเซอร์วิสตัวหนึ่งจะต้องไปร้องขอใช้บริการของเว็บเซอร์วิสอีกตัวหนึ่ง
12. ปรับปรุงรูปแบบการแสดงผลของแผนที่มีความถูกต้องและความเหมาะสมมากขึ้น
13. พัฒนาความสามารถในการให้บริการต่างๆ ของเว็บเซอร์วิส เช่น สามารถกำหนดแผนที่ในรูปแบบที่ต้องการได้ คือ GML , SVG และ JPEG ,ความสามารถในการค้นหาตำแหน่งสถานที่ต่างๆ เป็นต้น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

# ภาษา Markup Language ที่เกี่ยวข้องกับการพัฒนาแผนที่

### 2.1 ภาษา Markup Language ที่เกี่ยวข้องกับการพัฒนาแผนที่

องค์กร OGC (OpenGIS Consortium) ได้กำหนดรูปแบบในการจัดเก็บและการแลกเปลี่ยนข้อมูลทางด้านภูมิศาสตร์ พร้อมทั้งความสัมพันธ์และลักษณะประจำตัวของข้อมูล ในรูปแบบของ GML (Geography Markup Language) ซึ่งเป็นภาษา Markup Language รูปแบบหนึ่งที่มีพื้นฐานโครงสร้างที่อยู่ในรูปแบบของ XML (Extensible Markup Language) ทั้งนี้ทั้งนั้น รูปแบบการจัดเก็บและแลกเปลี่ยนข้อมูลทางด้านภูมิศาสตร์ในรูปแบบของ GML นั้นก็เป็นเพียงแค่ตัวข้อมูลที่ใช้ในการจัดเก็บเท่านั้น ไม่สามารถที่จะนำมาแสดงผลได้ ซึ่งการจะนำมาแสดงผลบน web browser ได้นั้น จำเป็นที่จะต้องแปลงรูปแบบซะก่อน ซึ่งหนึ่งวิธีในการแสดงผลของ GML บน web browser นั้นคือการแปลงให้อยู่ในรูปแบบของ SVG (Scalable Vector Graphics Language) ซึ่งเป็นภาษา Markup Language อีกรูปแบบหนึ่งที่ใช้ในการแสดงผลได้

### 2.2 ภาษา XML (Extensible Markup Language)

XML ถูกกำหนดโดยกลุ่มทำงาน XML ของสถาบัน World Wide Web Consortium (W3C) กลุ่มทำงานกลุ่มนี้ได้บรรยายถึงภาษา XML ไว้ว่า Extensible Markup Language เป็นฟอร์แมตที่อธิบายถึงรายละเอียดของโครงสร้างและแบบของข้อมูลเป็นภาษาหรือชุดคำสั่งเกี่ยวกับข้อมูลบนเว็บที่ให้การพัฒนาและมีศักยภาพในส่วนของโครงสร้างข้อมูลจากหลากหลายแอปพลิเคชันมานำเสนอบนเครื่องเดสก์ทอปด้วย XML จะทำให้การจัดการข้อมูลหรือเรียกใช้ข้อมูลจากแอปพลิเคชันต่างๆ จะเข้าสู่มาตรฐานเดียวกัน

XML และ HTML จะเป็นส่วนหนึ่งของภาษา SGML ซึ่ง XML จะให้รายละเอียดเกี่ยวกับข้อมูล เช่น ชื่อเมือง อุณหภูมิ ความกดอากาศ ส่วน HTML เป็นการกำหนด Tag ต่างๆ ที่จะกำหนดรูปแบบการแสดงผลของข้อมูลบนหน้าเว็บ ซึ่งข้อมูลจะสามารถแสดงออกมาได้หลายรูปแบบขึ้นอยู่กับข้อกำหนดของ HTML และในปัจจุบันนี้ ด้วย XML จะมีการให้รายละเอียดของเนื้อหาเอกสารที่เรียกว่า Document Type Definition (DTD) ที่ให้รายละเอียดเกี่ยวกับตัวเอกสารว่าจะแสดงหรือซ่อนส่วนไหนของเอกสารบ้างซึ่ง DTD จะเป็นส่วนที่เพิ่มเติมสำหรับ XML ถ้าหากมีการส่งข้อมูลในรูปแบบ DTD ก็จะสามารถรู้กันว่าเป็น XML มีความหมายหลายๆ คำที่อธิบายลักษณะของ XML

#### 2.2.1 ลักษณะโครงสร้างของ XML

XML เป็นการใช้อ็ควมเพื่อบ่งบอกโครงสร้างของเอกสาร พิจารณาตัวอย่างรูปแบบโครงสร้างของหนังสือ เมื่อนักเขียนประกอบด้วยจำนวนบท 2 บท ในแต่ละบทประกอบด้วยเนื้อความ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Begin Book
  Begin Chapter 1
    Text for Chapter 1
  End Chapter 1
  Begin Chapter 2
    Text for Chapter 2
  End Chapter 2
End Book

```

หนังสือที่มีอยู่ในปัจจุบันจะมีโครงสร้างที่มีรายละเอียดที่ซับซ้อนมากกว่านี้ เช่น บทนำ, สารบัญ เป็นต้น เช่นเดียวกัน ภายในส่วนเนื้อความยังประกอบด้วยโครงสร้างย่อยคือ ย่อหน้า (Paragraph) แต่ละย่อหน้ายังประกอบขึ้นจากประโยค คำและตัวอักษรด้วย

ลักษณะของเอกสาร XML นั้น สามารถอธิบายโดยใช้ตัวอย่างที่ 1 ได้ ดังนี้

ตัวอย่างที่ 1

```

<?xml version="1.0" encoding="windows-874"?>
  <mali>
    <malisorn>ขึ้นต้นด้วยมะลิซ้อน</malisorn>
    <malila>พอแตกใบอ่อนเป็นมะลิลา</malila>
  </mali>
<?xml-stylesheet type="text/xsl" href=""/>

```

บรรทัดที่ 1 นั้นหมายความว่าเราประกาศเอกสารนี้เป็นเอกสาร XML และมีการเข้ารหัสอักขระเป็น windows-874 เพื่อให้ใช้ภาษาไทยได้ จากนั้นเราจะเห็นสิ่งที่เราคุ้นเคยที่เราเรียกว่า tag แต่จริงๆ แล้วในภาษา XML จะแบ่งโครงสร้างเป็น 2 ส่วนใหญ่ๆ คือ Tag และ Element สามารถอธิบายเพิ่มเติมได้จากตัวอย่างที่ 2

ตัวอย่างที่ 2

```

<root>
  <element>
    <tag></tag>
  </element>
</root>

```

ความหมายของ Tag กับ Element

### 2.2.1.1 Tag

สำหรับใน XML แล้ว tag มีความหมายในลักษณะเดียวกับที่ใช้ใน HTML Tag คือข้อความที่อยู่ระหว่างสัญลักษณ์ < และ >

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Tagเปิด (Start  
<book>

จากตัวอย่างที่แสดง ด้านบนถูกเรียกว่า Tag เปิด ดังนั้น Tag เปิดจึงมีสัญลักษณ์คือ <...>

Tagปิด (End  
</book>

Tag ที่ถูกเรียกว่า Tag ปิด ต่อเมื่อใน Tag มีเครื่องหมาย / อยู่หลังสัญลักษณ์ < ดังนั้นลักษณะของ Tag ปิดจึงมีรูปแบบคือ </...> หากพิจารณาระหว่าง Tag เปิดกับ Tag ปิดแล้ว ข้อแตกต่างอีกข้อหนึ่งคือ Tag เปิดเป็น Tag ที่สามารถใส่ข้อมูล attribute ลงไปภายใน tag ได้ แต่ Tag ปิดจะไม่ทำกัน

### 2.2.1.2 Element

ในที่นี้คือโครงสร้างหลักของ XML ซึ่งอยู่ในรูปของ Tag เช่นเดียวกันตามตัวอย่างข้างบน Element คือ

```
<root>
  <element>
  </element>
</root>
```

จะมีลักษณะซ้อนกันเป็นชั้นๆ โดย root element เป็น element แรกสุดของเอกสาร XML

Element ใช้เป็นส่วนประกอบของเนื้อหาของเอกสาร XML และ Attribute เพื่อให้ผู้อ่านได้เข้าใจความหมายของศัพท์หลายๆ คำศัพท์ก่อนที่จะนำไปใช้ในการสร้าง XML ซึ่งจำเป็นอย่างมากพิจารณาจำกัดความของ element

```
<chap number="1">Text for Chapter 1</chap>
```

สังเกตจากลักษณะรูปทางด้านบนทั้งหมดตั้งแต่ <chap...จนถึง </chap> ถูกเรียกว่า element หรือถ้ามองง่าย ๆ คือ element เริ่มต้นที่ Tag เปิด และสิ้นสุดที่ Tag ปิดใน Tag คำสั่งเดียวกัน

### 2.2.1.3 Content

เนื้อหาหรือ Content ถือได้ว่าเป็นข้อมูลเพื่อใช้ในการแสดงให้ผู้อ่านเอกสารได้เห็น หรือกล่าวอีกนัยหนึ่งคือ Content อยู่หลัง Tag เปิด และจบก่อนที่จะถึง Tag ปิดนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.2.1.4 Attribute

Attribute คือข้อมูลความหมายเพิ่มเติม แน่นนอนถ้ากล่าวถึงสิ่งของหากเราไม่ได้ให้ความหมายเพิ่มเติมเราก็จะไม่มีทางรู้หรือแยกแยะสิ่งที่คล้ายๆ กันออกจากกันได้ เช่น ถ้าพูดถึง บท ในหนังสือ แค่นี้เป็นความหมายโดยรวม แต่ถ้าบอกว่าบทที่ 1 ในหนังสือ เลข 1 ในที่นี้คือความหมายเพิ่มเติมให้กับบท

Attribute จะถูกบรรจุอยู่ใน Tag เปิด และ Attribute นี้ในบางครั้งอาจจะมีหรือไม่มีก็ได้ รวมถึงถ้ามีอาจมีได้มากกว่าหนึ่งตัวใน Tag เปิด

## 2.2.2 ตัวกำหนดโครงสร้างของ XML ด้วย DTD (Document Type Declaration)

Document Type Declaration ใช้ในการกำหนดโครงสร้างของเอกสาร XML โดยเมื่อเปิดเอกสาร XML ด้วย DTD ตัวประมวลผลจะตรวจสอบเอกสารว่าถูกต้องตรงกับ DTD ที่ได้กำหนดโครงสร้างไว้หรือไม่ เช่น ถ้าไม่ประกาศ Element หรือ Attribute ไว้ใน DTD แล้วจะไม่สามารถใช้ Element หรือ Attribute นั้นๆ ในเอกสาร XML ได้

### 2.2.2.1 รูปแบบของ DTD

DTD มีรูปแบบทั่วไปตามนี้

```
<!DOCTYPE Name DTD>
```

Name ตามตัวอย่างเป็นที่กำหนดชื่อของ Document Element (หรือ Root Element) ชื่อของ Document Element จะต้องตรงกับชื่อที่กำหนดไว้ที่นั่นเท่านั้น

DTD สามารถบรรจุชนิดของการประกาศ Markup ไว้ดังนี้

- การประกาศ **Element Type**

เป็นการกำหนดชนิด (Type) ของ Element ที่เอกสารสามารถบรรจุ เช่น ข้อมูล และคำสั่งของ Element โดยมีรูปแบบการประกาศ Element Type ดังนี้

```
<!ELEMENT Name contentspec>
```

ในตัวอย่าง Name คือชื่อของ Element Type ที่ประกาศ และ Contentspec คือ Content Specification เป็นที่กำหนดว่าภายใน Element สามารถบรรจุอะไรได้บ้าง ด้านล่างนี้จะเป็นตัวอย่างของเอกสาร XML ที่มี Element Type 2 ชนิด และ Element Type ที่ประกาศชื่อ COLLECTION แสดงให้เห็นว่าสามารถบรรจุ Element ที่ชื่อ CD ได้ มากกว่า 1 Element และ Element Type ที่ชื่อ CD กำหนดว่าสามารถบรรจุได้เฉพาะข้อมูลประเภท Character Data เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<?xml version="1.0"?>

<!DOCTYPE COLLECTION
[
  <!ELEMENT COLLECTION (CD) +>
  <!ELEMENT CD
      (#PCDATA)>
]
>

<COLLECTION>
  <CD>Mozart Violin Concertos 1, 2, and 3</CD>
  <CD>Telemann Trumpet Concertos</CD>
  <CD>Handel Concerti Grossi</CD>
</COLLECTION>
```

- การประกาศ Attribute

เป็นการกำหนดชื่อของ Attribute ที่สามารถใช้ได้เฉพาะใน Element Type Declaration เช่น ชนิดข้อมูลและค่าปริยาย (Default Value) ของ Attribute โดยมีรูปแบบทั่วไปดังนี้

```
<!ATTLIST Name AttDefs>
```

Name แสดงชื่อ Attribute ที่สัมพันธ์กับ Attribute ตั้งแต่ 1 ตัวขึ้นไป และ AttDefs เป็นชุดของ Attribute Definition ตั้งแต่ 1 ชุดขึ้นไป โดย AttDefs จะกำหนดค่าให้ Attribute 1 ตัว

Attribute Definition มีรูปแบบดังต่อไปนี้

```
Name AttType DefaultDecl
```

ในที่นี้ Name เป็นชื่อของ Attribute, AttType คือ Attribute Type เป็นชนิดของค่าที่สามารถกำหนดให้กับ Attribute และ DefaultDecl คือ Default Declaration ซึ่งจะแสดงข้อมูลที่ Attribute ต้องการ และเตรียมข้อมูลอื่นๆ

ตัวอย่างต่อไปนี้เป็นเอกสาร XML ที่ประกอบด้วย Attribute และ Element ที่ชื่อ FILM

```
<!DOCTYPE FILM>
[
  <!ELEMENT FILM (TITLE, (STAR | NARRATOR | INSTRUCTOR))>
  <!ELEMENT FILM Class CDATA "fictional" Year CDATA #REQUIRED>
  <!ELEMENT TITLE (#PCDATA)>
  <!ELEMENT STAR (#PCDATA)>
  <!ELEMENT NARRATOR (#PCDATA)>
  <!ELEMENT INSTRUCTOR (#PCDATA)>
]
>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<FILM Year="1984">
  <TITLE>The Morning After</TITLE>
  <STAR>Morgan Attenbury</STAR>
</FILM>
```

### 2.2.3 การแปลงรูปแบบเอกสาร XML (Transforming XML)

XML Stylesheet Language (XSL) เป็นภาษาที่ใช้ในการแปลงจากรูปแบบ (Transformation) ของเอกสาร XML ไปอยู่ในรูปแบบอื่นๆ และในขณะเดียวกันก็ทำการประยุกต์รูปแบบการจัดข้อความ (style) ด้วย XSL stylesheet (สไตลชีต) ถูกเขียนในรูปแบบที่เหมาะสมกับ XML (XML-compliant format) แต่ต้องทำการกำหนดลำดับของ Element ไว้ล่วงหน้าในการกำหนดสิ่งที่ต้องการให้กระทำในการแปลงรูปแบบ (อีกประการหนึ่งคือเพื่อให้ทราบไว้ว่าจะเลือกใช้อันไหนทำงาน) โดยความหมายจะถูกกำหนดโดยตัวประมวลผล XSL เอง ซึ่งควรจะเป็นไปตามมาตรฐานของ W3C ดังนั้นจาก XSL ซึ่งเป็นแอปพลิเคชันหนึ่งของ XML และอยู่ในรูปแบบที่เหมาะสมกับ XML ดังนั้นจึงสามารถประมวลผล XSL สไตลชีตได้เหมือนกับเอกสาร XML

ครั้งแรก XSL เป็นข้อเสนอที่ถูกระบุทั้งทางที่การแปลงรูปแบบ XML และไวยากรณ์ในการจัดการรูปแบบบนแพลตฟอร์มที่เป็นอิสระกัน (platform-independent styling grammar) แต่อย่างไรก็ตามในส่วนของการแปลงรูปแบบก้าวหน้าอย่างรวดเร็วและเป็นประโยชน์อย่างมาก ขณะที่ไม่มีแอปพลิเคชันในส่วนของการจัดรูปแบบมากนักและส่วนใหญ่จะกล่าวว่ามีประโยชน์ทางการทำงานน้อย ดังนั้นจากข้อเสนอในตอนแรกจึงสามารถแบ่งออกได้เป็น 2 ส่วนที่แตกต่างกันคือ ส่วนในการแปลงรูปแบบเอกสาร ซึ่งปัจจุบันถูกเรียกว่า XSLT (XSL Transformation Language) และในส่วนของการจัดรูปแบบจะถูกเรียกอย่างเป็นทางการว่าวัตถุในการจัดรูปแบบของ XSL (XSL Formatting Object) หรือ XSL-FO แต่ในที่นี้จะพิจารณาเฉพาะในส่วนของการแปลงรูปแบบเท่านั้นเพื่อแปลงรูปแบบจากเอกสาร XML แบบหนึ่งไปเป็น XML อีกแบบหนึ่ง ซึ่งในโครงการนี้ใช้ในการแปลงเอกสาร GML ไปเป็น SVG

#### 2.2.3.1 The XSL Transformation Language (XSLT)

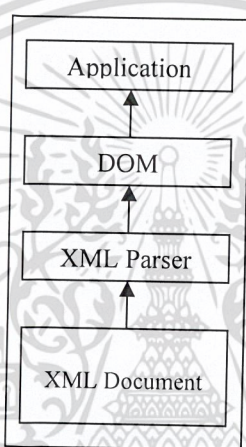
XSLT เป็นข้อเสนอแนะจาก W3C ในการกำหนดเขต Element ของ XML ที่สามารถจะถูกใช้ในการสร้างสไตลชีตที่แปลงเอกสาร XML ให้อยู่ในรูปแบบใดๆ ซึ่งในปัจจุบันนี้ใช้ในการแปลง XML รูปแบบหนึ่งไปเป็นอีกรูปแบบหนึ่ง (ในโครงการนี้ใช้ในการแปลงเอกสาร GML ไปเป็น SVG) โดยการทำงานจะทำโดยนำเอกสาร XML ที่เหมาะสมกับ XML นั้นไปทำการแปลงรูปแบบเพื่อสร้างเอกสารเอาต์พุต ซึ่งขึ้นอยู่กับคำสั่งและเนื้อหาภายใน สไตลชีตจึงสามารถสร้างรูปแบบเอาต์พุตใดๆ ก็ได้ตามที่ต้องการในการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.2.4 DOM (Document Object Model)

การใช้งาน XML นั้นเราสามารถที่จะทำอะไรกับข้อมูลในแอปพลิเคชันที่เราเขียนขึ้นได้ ไม่ใช่เพียงการเข้าถึงเท่านั้นแต่ยังรวมถึงการแก้ไขและเพิ่มเติมเอกสาร XML ที่เรามีอยู่อีกด้วย จึงเกิดมี Document Object Model (DOM) ขึ้นเพื่อช่วยในการทำงานกับเอกสาร XML ทั้งหมด

DOM มักจะถูกใส่เข้าไปเป็นชั้นที่กั้นกลางระหว่าง XML parser กับแอปพลิเคชันที่ต้องการใช้ข้อมูลในเอกสาร XML ซึ่งหมายความว่า parser จะอ่านข้อมูลจากเอกสาร XML ให้กับ DOM จากนั้น DOM จะถูกใช้โดยแอปพลิเคชันระดับสูงกว่า แอปพลิเคชันสามารถทำงานกับเอกสาร XML ได้ตามที่ต้องการ รวมถึงการใส่ DOM เข้าไปใน DOM ตัวอื่น ซึ่งในแต่ละภาษาก็จะมีแบบจำลอง DOM เป็นของตนเอง



รูปที่ 2-1 การใช้งาน DOM

## 2.3 ภาษา GML (Geography Markup Language)

ภาษา GML เป็นรูปแบบหนึ่งของเอกสาร XML ที่ถูกออกแบบขึ้นเพื่อการจัดเก็บและการแลกเปลี่ยนข้อมูลทางภูมิศาสตร์ ทั้งทางด้านความสัมพันธ์และลักษณะประจำตัวของข้อมูล ในรูปแบบของ feature ที่แตกต่างกันและสามารถกำหนดวิธีที่จะแสดงข้อมูลทางภูมิศาสตร์นั้นโดยใช้ข้อกำหนดของ GML ที่ถูกรับรองโดยองค์กร OGC (OpenGIS Consortium) ในการอธิบายข้อมูลเชิงพื้นที่ เพื่อให้ผู้ที่ทำงานด้าน GIS (Geographic Information System) ใช้เป็นพื้นฐานในการออกแบบพัฒนาแผนที่ในรูปแบบต่างๆ และผู้ใช้จะสามารถดูแผนที่ที่ถูกแสดงนั้นผ่านทาง browser มาตรฐานที่ใช้กันอยู่ทั่วไป โดยรายละเอียดเพิ่มเติมสามารถศึกษาได้จาก [http://portal.opengeospatial.org/files/?artifact\\_id=4700](http://portal.opengeospatial.org/files/?artifact_id=4700)

### 2.3.1 ข้อดีของการใช้ GML ในการแสดงแผนที่

1. ได้แผนที่ที่มีคุณภาพดีกว่า เนื่องจาก GML จะเปลี่ยนข้อมูลทางภูมิศาสตร์ทั้งหมดให้อยู่ในรูปของ feature หรือ object ซึ่งจะสามารถนำไปแสดงผลในรูปแบบที่ไม่จำกัดรายละเอียดในการแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

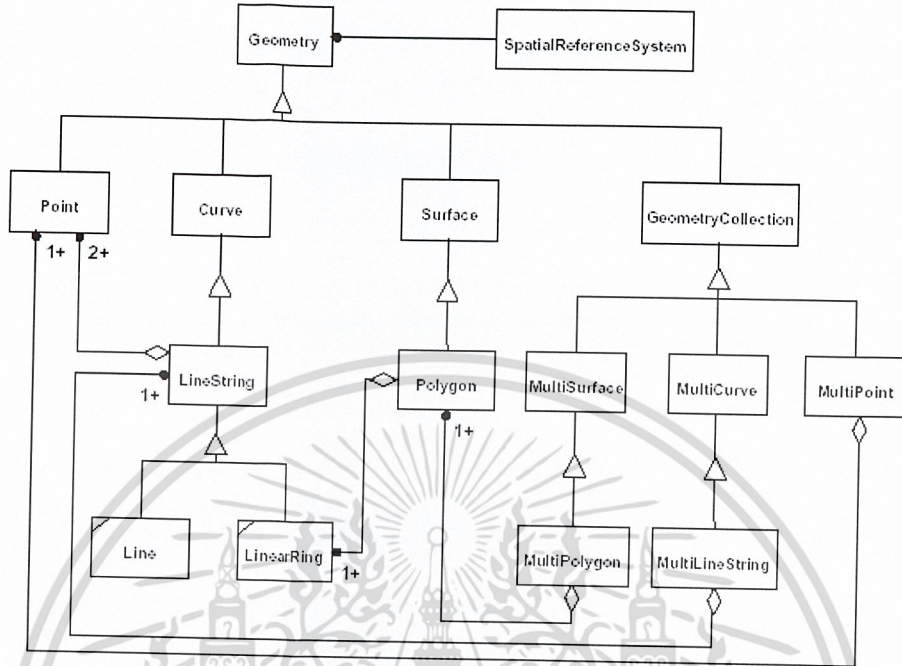
2. สามารถใช้งานได้บนเบราว์เซอร์โดยไม่ต้องซื้อโปรแกรมทางที่ใช้แสดงผลทางฝั่งไคลเอนต์ เมื่อข้อมูล GML ถูกรับโดยฝั่งไคลเอนต์จะถูกเปลี่ยนให้อยู่ในรูปของภาพแผนที่บนเบราว์เซอร์โดยใช้ภาษา SVG (Scalable Vector Graphics) ในการแสดงออกมาเป็นภาพ ซึ่งเราสามารถแสดงแผนที่บนเบราว์เซอร์โดยไม่จำเป็นต้องติดตั้งซอฟต์แวร์อื่นๆ เพิ่มเติมหากเบราว์เซอร์สามารถรองรับภาพแบบเวกเตอร์ (vector graphics) ได้ และเราสามารถดาวน์โหลด SVG Viewer ซึ่งเป็นปลั๊กอินที่ใช้สำหรับดูภาพแบบ SVG ของบริษัท Adobe ได้โดยไม่ต้องเสียค่าใช้จ่ายใดๆ
3. สามารถสร้างแผนที่ที่มีรูปแบบเฉพาะได้ เนื่องจาก GML จะเก็บข้อมูลของแผนที่ เช่น ที่ตั้งของ feature นั้น, ลักษณะทางภูมิศาสตร์, ชนิดของ feature และ attribute ต่างๆ โดยไม่กำหนดข้อมูลหรือวิธีในการแสดงผลของข้อมูลแผนที่นั้นๆ ซึ่งทำให้ผู้ใช้สามารถสร้างแผนที่ให้มีรูปแบบได้ตามที่ต้องการ
4. สามารถทำการแก้ไขแผนที่ได้ แผนที่ที่เขียนโดย GML จะถูกดาวน์โหลดและทำการ render บนเบราว์เซอร์ซึ่งเมื่อ GML ถูกเปลี่ยนเป็น SVG ผู้ใช้จะสามารถใช้โปรแกรมบนไคลเอนต์ในการเพิ่มเติมตัวอักษรหรือทำการเปลี่ยนแปลงภาพบนแผนที่ และสามารถบันทึกเก็บเป็นไฟล์ภาพเพื่อนำไปใช้ในกรณีอื่น เช่น ทำการพิมพ์ออกทางเครื่องพิมพ์หรือส่ง email ได้
5. สามารถเชื่อมโยง link ลงไปยังในแผนที่ได้ ประโยชน์ที่สำคัญของ GML คือผู้ใช้สามารถฝัง URL ลงใน feature ที่มีในแผนที่ได้ ความสามารถนี้ทำให้เราสามารถออกแบบแผนที่ให้มีความแปลกใหม่และทันสมัยยิ่งขึ้น
6. ทำการ query ได้ดียิ่งขึ้น หากผู้ใช้ต้องการที่จะคลิกไปบน feature บนแผนที่และค้นหาข้อมูลเพิ่มเติมเกี่ยวกับ feature นั้นๆ (เช่น ชื่อหรือความลึกของแม่น้ำ) ซึ่งไม่สามารถทำได้บนแผนที่ที่สร้างขึ้นโดยใช้ไฟล์ภาพ (gif/jpg) สำหรับแผนที่ที่ใช้ GML นั้น เราจะสามารถระบุ feature ที่ถูกคลิกนั้นได้
7. สามารถกำหนดเนื้อหาที่ต้องการแสดงหรือต้องการได้รับ เพราะ GML เป็นภาษาที่อ้างอิงถึง feature ทำให้ผู้ใช้สามารถกำหนดได้ว่าต้องการให้มี feature ใดแสดงบนแผนที่บ้าง ซึ่งจะช่วยลดระยะเวลาในการรับ
8. สามารถแสดงผลได้บนทุกอุปกรณ์ที่ใช้เบราว์เซอร์ได้ ข้อมูลทางภูมิศาสตร์ใน GML นั้นสามารถส่งไปยังอุปกรณ์ชนิดใดก็ได้ที่สามารถรองรับ XML การแสดงผลแบบ XML ได้ เช่น เครื่องพีดีเอหรือโทรศัพท์มือถือ ทำให้แผนที่นั้นสามารถใช้งานได้อย่างกว้างขวางและหลากหลายยิ่งขึ้น

### 2.3.2 GML Basic concepts

GML เป็นรูปแบบหนึ่งของเอกสาร XML ที่ถูกออกแบบขึ้นเพื่อการจัดเก็บและการแลกเปลี่ยนข้อมูลทางภูมิศาสตร์ รองรับการแสดงผลข้อมูลทางภูมิศาสตร์ในรูปแบบต่างๆ

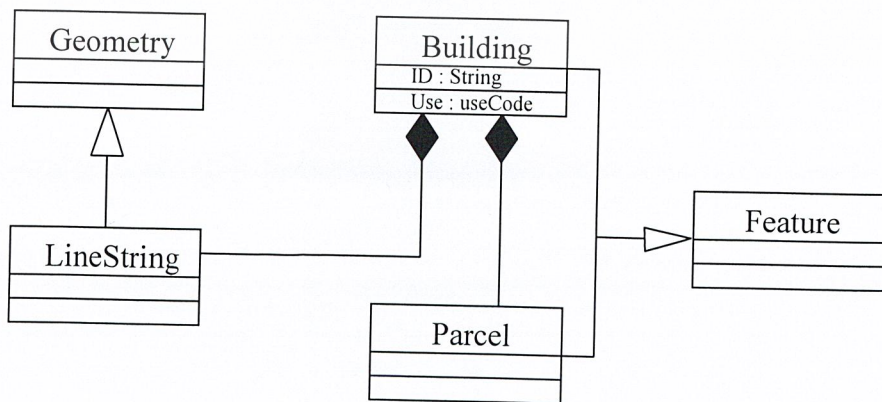
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยทั่วไปสเก็มา (Schemas) จะอธิบายถึงลักษณะของ Class ของอ็อบเจ็กต์ แต่ใน XML Schema จะอธิบายว่าข้อมูลจะถูกกำหนด (mark up) ได้อย่างไร



รูปที่ 2-2 The geometry model for simple feature (Open GIS Consortium, 2001)

องค์ประกอบทางเรขาคณิต (Geographic feature) จะถูกอธิบายด้วยคุณสมบัติต่างๆ ซึ่งคุณสมบัติเหล่านี้จะประกอบด้วยคุณสมบัติที่เกี่ยวข้องกับระยะทาง โดยอธิบายตำแหน่งและรูปร่างของ feature นั้นๆ แต่ละ feature จะมีชนิดของ feature ซึ่งเทียบเท่ากับ Class ในอ็อบเจ็กต์ model ดังนั้น Class definition จึงเป็นส่วนที่กำหนดคุณสมบัติที่แต่ละ feature จำเป็นต้องมี เช่น อาคารถูกกำหนดให้มีชื่อของอาคาร ซึ่งเป็นข้อมูลประเภท String ประเภทของการใช้งานอาคารซึ่งกำหนดให้ชนิดข้อมูลเป็น text ที่สามารถเลือกจาก list ที่กำหนดไว้ เส้นรอบนอกซึ่งกำหนดให้เป็นรูปวงแหวน (LinearRing) ซึ่งเป็นคุณสมบัติทางภูมิศาสตร์ โดยรูปด้านบนอธิบายถึง Geographic feature โดยใช้ UML



รูปที่ 2-3 Features and properties

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.3.3 Encoding Geometry

สิ่งที่สำคัญที่สุดในมาตรฐานของการแปลงข้อมูลทางภูมิศาสตร์ คือความสามารถในการแสดงรูปร่างทางเรขาคณิต ในการแสดงภาพทางภูมิศาสตร์ที่เกี่ยวข้องกับระยะทางโดยใช้ GML 2.0 นั้น จำเป็นต้องอ้างอิงถึง XML Schema ทั้งหมด 2 สกีมาคือ GML Feature Schema และ GML Geometry Schema

#### 2.3.3.1 GML Feature Collections

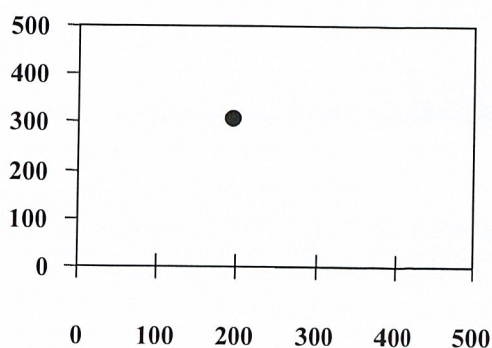
1. ภาษา GML ใช้ feature อธิบายองค์ประกอบต่างๆ ของภูมิประเทศในรูปแบบของ Geographic entities
2. feature ประกอบด้วย list ของคุณสมบัติต่างๆไป ซึ่งประกอบด้วยชื่อของข้อมูล ชนิดของข้อมูล ค่าของข้อมูล และคุณสมบัติต่างๆทางภูมิศาสตร์ เช่น lines, points, curves, polygons
3. feature สามารถสร้างขึ้นจาก feature อื่นๆ หรือกำหนดชนิดของข้อมูลเป็น feature อื่นๆ ได้

ตามหลักของ OGC Simple Feature Model, GML จะรองรับการแสดงผลขององค์ประกอบทางเรขาคณิต คือ จุด (point), เส้น (LineString), วงแหวน (LinearRing), รูปหลายเหลี่ยม (Polygons)

และโดยรูปแบบพื้นฐานที่มีมาให้นี้ เราสามารถเพิ่มเติม MultiPoint, MultiString, MultiPolygon และ MultiGeometry นอกจากองค์ประกอบทางเรขาคณิตเหล่านี้แล้ว GML ยังประกอบด้วย <coordinates> หรือ <coord> ซึ่งเป็น tag ที่ใช้ในการอ้างอิงถึงจุดคู่ลำดับหรือตำแหน่งและ <Box> ใช้ในการนิยามองค์ประกอบที่มีพื้นที่เป็นรูปสี่เหลี่ยม

ส่วนที่สำคัญของ GIS คือวิธีการในการอ้างอิงถึงองค์ประกอบทางภูมิศาสตร์ต่างๆ บนพื้นโลก คู่ลำดับ (coordinate) สำหรับรูปเรขาคณิตที่เราใช้แสดงแทนบนแผนที่นั้นจะอ้างอิงโดย Spatial Reference System (SRS) และทุกองค์ประกอบจะต้องระบุ SRS หรืออีกวิธีหนึ่งคือระบุ GID Attribute ซึ่งเป็น Attribute ที่จะแสดงค่าเฉพาะตัวทางภูมิศาสตร์ของแต่ละองค์ประกอบเรขาคณิตนั้น

**Points** เป็นรูปเรขาคณิตพื้นฐานที่ประกอบด้วยคู่ลำดับเพียงคู่ลำดับเดียว multipoint เป็นเซตที่ประกอบด้วยจุดหลายๆ จุด

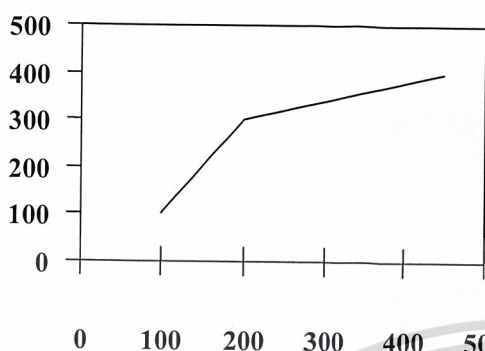


```
<Point GID="" srsName="gml_srs">
  <coord><x>200.0</x><y>300.0</y></coord>
</Point>
```

รูปที่ 2-4 Point

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**LineString** ประกอบด้วย list ของคู่ลำดับแสดงที่ส่วนของเส้นตรงหลายเส้นเชื่อมต่อกัน รูปด้านล่าง LineString จะต้องประกอบด้วยคู่ลำดับอย่างน้อย 2 คู่

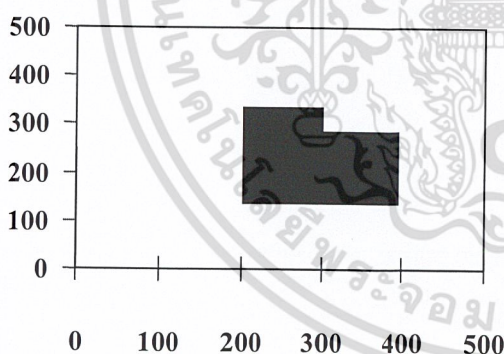


```
<LineString gid="2468" srsName="gml_srs">
  <coord><x>100.0</x><y>150.0</y></coord>
  <coord><x>200.0</x><y>300.0</y></coord>
  <coord><x>450.0</x><y>400.0</y></coord>
</LineString>
```

รูปที่ 2-5 LineString

**LinearRing** เป็นรูปปิด ประกอบด้วย list ของคู่ลำดับที่แสดงถึงส่วนของเส้นตรง คู่ลำดับสุดท้ายต้องมีค่าเท่ากับคู่ลำดับแรก และต้องประกอบด้วยคู่ลำดับอย่างน้อย 4 คู่

**Polygons** เป็นพื้นที่ผิวที่ผิวของรูปเป็นเซตของ LinearRing โดยขอบนอกของ Polygon คือ LinearRing วงแรกและ LinearRing วงที่เหลือจะซ้อนเรียงกันภายในของวงแรก



```
<Polygon gid="1234" srsName="gml_srs">
  <outerBoundaryIs>
    <LinearRing>
      <coord><x>200.0</x><y>150.0</y>
      <coord><x>200.0</x><y>350.0</y>
      <coord><x>300.0</x><y>350.0</y>
      <coord><x>300.0</x><y>250.0</y>
      <coord><x>400.0</x><y>250.0</y>
      <coord><x>400.0</x><y>150.0</y>
      <coord><x>200.0</x><y>150.0</y>
    </LinearRing>
  </outerBoundaryIs>
</Polygon>
```

รูปที่ 2-6 Polygon

### 2.3.3.2 GML application schema

Schema คือรูปแบบการจัดโครงสร้างและความสัมพันธ์ของข้อมูล โดยภาษา GML นั้นกำหนดให้ผู้ใช้ต้องกำหนด GML application schema ซึ่งเขียนโดยใช้หลักการของ XML Schema และเป็นไปตามข้อบังคับของ GML

GML Application Schema ประกอบด้วย

1. Annotations ซึ่งเป็นส่วนประกาศหรือส่วนที่ใช้การอธิบาย schema นั้น

2. Simple Type คือชนิดของข้อมูลที่ไว้รูปแบบของข้อมูลที่ GML กำหนดมาให้แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเฉพาะบุคคลเท่านั้น ไม่อนุญาตให้เผยแพร่ไปยังหน่วยงานอื่นด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. Complex Type คือชนิดของข้อมูลที่สร้างขึ้นใหม่โดยอาจจะเกิดจากการใช้ simple type หลายๆ ตัวมาประกอบกัน หรือเกิดจาก complex type ชนิดอื่นก็ได้
4. Element
5. Attribute

ตัวอย่าง GML Feature Type Schema

```
<element name="RANET Radio Station" type="RadioSchema"/>
<complexType name="RadioSchema">
  <sequence>
    <element name="villageName" type="string"/>
    <element name="countryName" type="string"/>
    <element name="numWorldSpace" type="integer"/>
    <element name="numFreeplay" type="integer"/>
  </sequence>
</complexType>
```

## 2.4 ภาษา SVG (Scalable Vector Graphics Language)

คือภาษาที่ใช้อธิบาย GML ในรูปของภาพ 2 มิติได้ เนื่องจากเราสามารถอ่านและจัดการข้อมูลในเอกสาร XML ได้ง่าย จึงมีความพยายามที่จะสร้างมาตรฐานในการบอกลักษณะการแสดงผลรูปภาพด้วย XML ทำให้เกิดมาตรฐาน SVG (Scalable Vector Graphics) ซึ่งได้รับการรับรองโดยองค์กร W3C

SVG จะกำหนดรูปแบบของวัตถุสามอย่างคือ vector graphic shape (เช่น ส่วนของเส้นตรงและเส้นโค้ง), รูป และตัวอักษร โดยสามารถที่จะกำหนด grouped, styled, transformed หรือ compo sited ให้แก่วัตถุใน SVG ได้ อีกทั้งยังเพิ่มคุณสมบัติ transformations, clipping paths, alpha masks, filter effects, template อีอบเจกต์และอีกมากมาย

เนื่องจาก SVG เป็น XML ดังนั้นจึงมีการสร้าง DOM สำหรับ SVG ขึ้นมาโดยอ้างอิงตามมาตรฐานของ W3C ซึ่งทำให้ SVG มีคุณสมบัติ dynamic และ interactive ตามไปด้วย แม้แต่ event handlers อย่าง onmouseover( ) หรือ onclick( ) ก็สามารถกำหนดลงในวัตถุของ SVG ได้อีกด้วย อีกทั้งการอ้างอิงตามมาตรฐานทำให้ SVG สามารถติดต่อกับ XML อื่นๆ ในเว็บไซต์เดียวกันได้ โดยรายละเอียดเพิ่มเติมสามารถศึกษาได้จาก <http://www.w3.org/Graphics/SVG>

### 2.4.1 การสร้างภาพโดยใช้ภาษา SVG

ลักษณะการเขียน Code ภาษา SVG มีลักษณะเป็น XML (คล้ายกับ HTML) ซึ่งต้องมี tag เปิดและปิด และมี tag ที่อยู่ใน tag ด้วย ซึ่งจะสังเกตได้ว่า SVG ไฟล์ จะต้องมี <svg> อยู่เป็น tag นอกสุดซึ่งมีตัวอย่างดังต่อไปนี้

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 20000303 Stylable//EN"
"http://www.w3.org/TR/2003/WD-SVG-20000303/DTD/svg-20000303-
stylable.dtd">
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<svg xml:space="preserve" width="5.5in" height=".5in">
...
...
...
</svg>
```

ในบรรทัดแรกจะเป็นการบอกให้ผู้รู้ที่นำ SVG ไฟล์นี้ไปอ่านทราบว่า code ต่อไปนี้อ้างอิงจากมาตรฐาน XML version 1.0 โดยข้อมูลที่อยู่ในแต่ละ tag จะใช้มาตรฐาน iso-8859-1 ในการเขียน บรรทัดที่สองจะอ้างอิงถึง DTD หรือ Document Type Definition (DTD เป็นเอกสารที่ใช้บอกถึง ชนิดของข้อมูลที่จะใช้ใน XML และยังระบุอีกด้วยว่าเอกสารนั้นๆ มีโครงสร้างเป็นอย่างไร) ใน <svg> element สามารถที่จะบรรจุ text, shapes และ paths ได้ เช่น

1. การเขียน code SVG ในการแสดงผลข้อมูลแบบ Text

```
<text x="2.5cm" y="1.5cm" style="font-family:Verdana; font-size:20pt; fill:blue">
Hello world
</text>
```

2. การเขียน code SVG ในการแสดงผลข้อมูลแบบเส้น

```
<line x1="90" y1="20" x2="110" y2="100" style="stroke-width:25;stroke:purple"/>
```

4. เขียน code SVG ในการแสดงผลการวาดภาพสี่เหลี่ยม

```
<rect x="2cm" y="1cm" width="4cm" height="2cm" fill="none" stroke="red"
stroke-width=".1cm"/>
```

โดย element และ attribute ของ SVG ที่ใช้ในงานข่อยมีดังต่อไปนี้

### SVG Element

SVG	Action
<svg>	เป็น tag ที่ใช้ในการกำหนดพื้นที่ทั้งหมดของภาพ และเป็น tag ที่เก็บ tag ต่างๆ ของ SVG
<path>	ใช้ในการวาดเส้น
<rect>	ใช้ในการวาดสี่เหลี่ยม
<g>	ใช้ในการรวม tag ต่างๆ ให้อยู่ในกลุ่มเดียวกัน
<desc>	ใช้ในการเพิ่มเติมข้อมูลหรือคำอธิบาย
<defs>	ใช้ในการกำหนดค่าต่างๆ
<script>	ใช้ในการเขียน script จะเขียนใน tag นี้
<text>	ใช้ในการแสดงผลแบบอักษร
<cursor>	ใช้ในการเปลี่ยนรูปของ mouse cursor
<line>	ใช้ในการวาดเส้นอย่างง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<textPath>	ใช้ในการเขียนอักษรให้เป็นไปตามเส้นทางที่กำหนด
<image>	ใช้ในการแสดงรูปภาพในรูปแบบต่างๆ เช่น jpeg, gif, png เป็นต้น

ตารางที่ 2 -1 SVG Element

### SVG Attribute

Attribute	Action
fill	จัดการกับสีของอ็อบเจ็กต์นั้นๆ
id	ใช้ในการแยกแยะอ็อบเจ็กต์ต่างๆ
Width	ขนาดความกว้างของอ็อบเจ็กต์
Height	ขนาดความสูงของอ็อบเจ็กต์
X	ตำแหน่งพิกัด x
Y	ตำแหน่งพิกัด y
Xmlns	ใช้ในการอ้างอิง namespace
Visibility	กำหนดการปรากฏอยู่ของอ็อบเจ็กต์
Style	กำหนดลักษณะของอ็อบเจ็กต์
point-event	กำหนดการปฏิสัมพันธ์กับอ็อบเจ็กต์
viewBox	กำหนดขนาดในการแสดงผลของภาพ
stroke-width	กำหนดขนาดของเส้น
Stroke	กำหนดสีของเส้น
zoomAndPan	กำหนดความสามารถในการย่อหรือขยาย
OnClick	mouse event
onmouseover	mouse event
onmouseout	mouse event
Transform	เคลื่อนย้ายหรือหมุนปรับอัตราส่วนของภาพ
xlink:href	เป็นการอ้างอิงถึง URL

ตารางที่ 2 -2 SVG Attribute

#### 2.4.2 การแสดงภาพโดยใช้ภาษา SVG บนเบราว์เซอร์

การที่จะทำให้ เบราวเซอร์ เข้าใจ SVG DOM มีอยู่หลายวิธี แต่หนึ่งในวิธีที่นิยม และจะใช้ในการทำโครงงานนี้ก็คือ การติดตั้ง plug-in ให้แก่เบราว์เซอร์ โดยในโครงงานนี้อ้างอิงกับ SVG plug-in ของบริษัท Adobe โดยสามารถดาวน์โหลดได้ที่ <http://www.adobe.com/svg>

หลังจากที่เราทำการบันทึกเป็น File นามสกุล \*.svg แล้ว การแทรก svg ลงในเอกสาร HTML สามารถทำการแทรกได้โดยใช้ Tag ต่อไปนี้ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<EMBED SRC="path/svgfile.svg" Name="SVGEmbed" HEIGHT="36"
WIDTH="396" TYPE="image/svg+xml"
PLUGINSOURCE="http://www.adobe.com/svg/viewer/install"/>
</EMBED>
```

## 2.5 จาวาสคริปต์ (JavaScript)

จาวาสคริปต์ เป็นภาษาสคริปต์ ที่สมบูรณ์ใกล้เคียงกับภาษาในการเขียนโปรแกรมปกติ จาวาสคริปต์มีคุณสมบัติพื้นฐานต่างๆ เช่นเดียวกับภาษาปกติ ไม่ว่าจะเป็นการกำหนดตัวแปร การกำหนดฟังก์ชันการควบคุมการทำงานของโปรแกรมสแตทเมนต์ และชุดของโอเปอเรเตอร์ต่างๆ นอกจากนี้จาวาสคริปต์ยังมีคุณสมบัติในการเขียนโปรแกรมแบบอ็อบเจกต์บางอย่างไว้ด้วย ดังเช่น จาวาสคริปต์สามารถเข้าไปใช้กับอ็อบเจกต์ ซึ่งปรากฏอยู่บนเว็บหรือจะแก้ไข Attribute รวมทั้งอีเวนต์ไว้ให้เรียกใช้งานด้วย

จาวาสคริปต์นั้นช่วยให้เพจ HTML สามารถทำงานได้ดังเช่นนักพัฒนาเว็บอาจจะสร้างให้เว็บเพจของตนสามารถตอบสนองต่อเหตุการณ์หรืออีเวนต์ของผู้ใช้ได้ไม่ว่าจะเป็นการคลิกเมาส์ หรือว่าการเรียกฟอร์มของ HTML ข้อดีของจาวาสคริปต์ที่สำคัญอีกประการหนึ่งคือ โครงสร้างของภาษาคัดลอกภาษาจาวา ซึ่งง่ายต่อการเรียนรู้และใช้งาน

ความแตกต่างระหว่างภาษาจาวาและจาวาสคริปต์ที่เห็นได้ชัดเจนคือ จาวาสคริปต์มีโครงสร้างของภาษาที่ไม่ซับซ้อนเหมือนกับจาวา นักพัฒนาเว็บที่คุ้นเคยกับ Tag HTML และเคยใช้งานภาษาในการเขียนภาษาซี สามารถเรียนรู้และใช้งานภาษาสคริปต์ได้ในเวลาไม่นาน นอกจากนี้โปรแกรมที่เขียนด้วยภาษาสคริปต์นั้นไม่จำเป็นที่จะต้องคอมไพล์เหมือนโปรแกรมจาวาทั่วไป ทั้งนี้เนื่องจากเป็นภาษาสคริปต์ โปรแกรมเว็บเบราว์เซอร์จึงมีตัวทำหน้าที่ปฏิบัติตามสคริปต์ที่เขียนไว้ทันที

ในการพัฒนาแผนที่โดยใช้ภาษา GML เราสามารถใช้ภาษาจาวาสคริปต์ในการเข้าถึง Element หรือ Attribute ของโครงสร้างภาษา SVG เพื่อสร้างส่วนของการปฏิสัมพันธ์กับผู้ใช้แผนที่เช่น การย่อหรือ ขยายแผนที่ การเลื่อนแผนที่หรือ การนำข้อมูลรายละเอียดของสถานที่ต่างๆ ขึ้นมาแสดงโดยใช้ภาษา SVG

## บทที่ 3

## Spatial Database System

## 3.1 ฐานข้อมูลเชิงพื้นที่ (Spatial database system)

ระบบฐานข้อมูลเชิงพื้นที่ (Spatial database system) เป็นระบบฐานข้อมูลที่ใช้ในการจัดการข้อมูลทางเรขาคณิต ข้อมูลที่เกี่ยวข้องกับภูมิศาสตร์และข้อมูลเชิงพื้นที่ โดยตัวอย่างของพื้นที่ที่เราสนใจ อาจจะเป็น พื้นที่ 2 มิติต่างๆ, ส่วนของพื้นผิวโลก หรือตัวอย่างอื่นๆ เช่น ข้อมูลแสดงโครงสร้างของโมเลกุลของโปรตีน ซึ่งการนำวิธีการจัดการกับฐานข้อมูลโดยใช้วิธีการจัดการกับระบบฐานข้อมูลในรูปแบบเก่าที่เคยมีอยู่เช่น การจัดการระบบฐานข้อมูลเชิงสัมพันธ์ (Relational database system) มาจัดการกับข้อมูลในรูปแบบที่ได้กล่าวมานั้นทำให้เกิดความไม่เหมาะสมเช่น

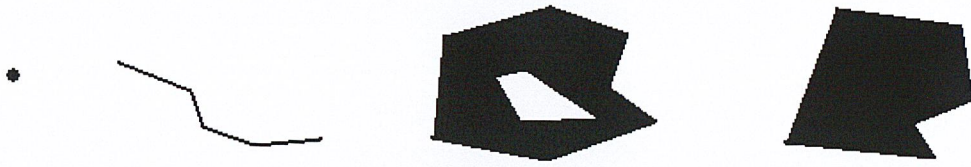
1. ข้อมูลในเชิงพื้นที่ (Spatial data) มีรูปแบบความสัมพันธ์ที่ซับซ้อนกว่าข้อมูลทางธุรกิจโดยทั่วไป ซึ่งทำให้โครงสร้างของฐานข้อมูลในรูปแบบเดิมไม่มีความเหมาะสมในการจัดการข้อมูล
2. การออกแบบโครงสร้างและการสร้างฐานข้อมูลโดยทั่วไปจะเป็นถูกดูแลโดยนักคอมพิวเตอร์ในขณะที่ข้อมูลเชิงพื้นที่จะถูกกำหนดโดยนักภูมิศาสตร์ซึ่งโดยทั่วไปจะมีรูปแบบและวิธีการที่แตกต่างกัน

## 3.2 Model of spatial information

แอปพลิเคชันหลักที่มีส่วนในการผลักดันการวิจัยระบบฐานข้อมูลเชิงพื้นที่คือเทคโนโลยีสำหรับ GIS (geographic information systems) ดังนั้นการใช้งานระบบฐานข้อมูลในเชิงพื้นที่ส่วนใหญ่จะเกี่ยวข้องกับข้อมูลทางด้านภูมิศาสตร์ ซึ่งอาจจะอยู่ในรูปของข้อมูลในรูปแบบ 2 มิติ หรือ 3 มิติ ซึ่งมีรูปแบบการจำลอง (Modeling) 2 ทางในการจำลองแบบจำลองของข้อมูลเชิงพื้นที่คือ

1. **Objects Base Model** ในการจำลองข้อมูลรูปแบบนี้ เราให้ความสนใจในวัตถุหรือองค์ประกอบต่างๆ ที่อยู่ในพื้นที่นั้น และแต่ละวัตถุหรือองค์ประกอบจะมีข้อมูลที่บอกรายละเอียดทางเรขาคณิตของตัวเองเช่น เมือง, ป่า หรือแม่น้ำ ซึ่งเป็นองค์ประกอบของประเทศ เป็นต้น

ในการจำลองวัตถุหรือองค์ประกอบที่เราสนใจ โครงสร้างพื้นฐานที่ใช้ในการจำลองโดยทั่วไปมักจะเป็นจุด (point) เส้น (line) หรือพื้นที่ (region) โดยจุด (point) จะใช้แสดงถึงวัตถุหรือองค์ประกอบที่เราต้องการแสดงเพียงที่ตั้งของมันบนพื้นที่ เส้น (line) โดยทั่วไปจะใช้แสดงถึงวัตถุหรือองค์ประกอบที่มีการเคลื่อนผ่านพื้นที่ที่เราสนใจหรือการเชื่อมต่อบนพื้นที่ เช่น แม่น้ำ ถนน หรือการวางสายโทรศัพท์ พื้นที่ (region) จะใช้แสดงถึงวัตถุหรือองค์ประกอบที่มีข้อมูลซึ่งแสดงรายละเอียดทาง 2 มิติ เช่นพื้นที่ของอุทยานแห่งชาติ ประเทศ หรือทะเลสาบ และอาจจะประกอบด้วยชิ้นส่วนย่อยหลายชิ้นมาเชื่อมต่อกัน ซึ่งโครงสร้างพื้นฐานที่ใช้ใน Object-Based Model แสดงได้ในรูปที่ 3-1



รูปที่ 3-1 โครงสร้างพื้นฐานที่ใช้ใน Object-Based Model

2. **Field-Based Model** ในการจำลองข้อมูลรูปแบบนี้เราให้ความสนใจในตัวพื้นที่นั้น หรือ ต้องการแสดงข้อมูลเกี่ยวกับตัวพื้นที่นั้น เช่นข้อมูลการใช้พื้นที่ หรือการแบ่งเขตต่างๆ ของประเทศ ซึ่งตัวแทนในการแสดงความสัมพันธ์เชิงพื้นที่ที่เกี่ยวข้องกันของข้อมูลคือ พาร์ทิชัน (Partitions) และ เน็ตเวิร์ค (Network) โดยพาร์ทิชันจะใช้ในการแสดงถึงส่วนของพื้นที่ที่ต้องมีการเชื่อมต่อกันและส่วนที่ติดกันนั้นเป็นส่วนที่เราให้ความสนใจ และเน็ตเวิร์คจะเป็นกราฟที่ถูกฝังลงในพื้นที่ ประกอบด้วยเซตของ จุดซึ่งแสดงตัวเป็นแต่ละ โหนดของกราฟหรือเซตของเส้นซึ่งแสดงถึงขอบของพื้นที่ที่เราสนใจ



รูปที่ 3-2 โครงสร้างพื้นฐานที่ใช้ใน Field-Based Model

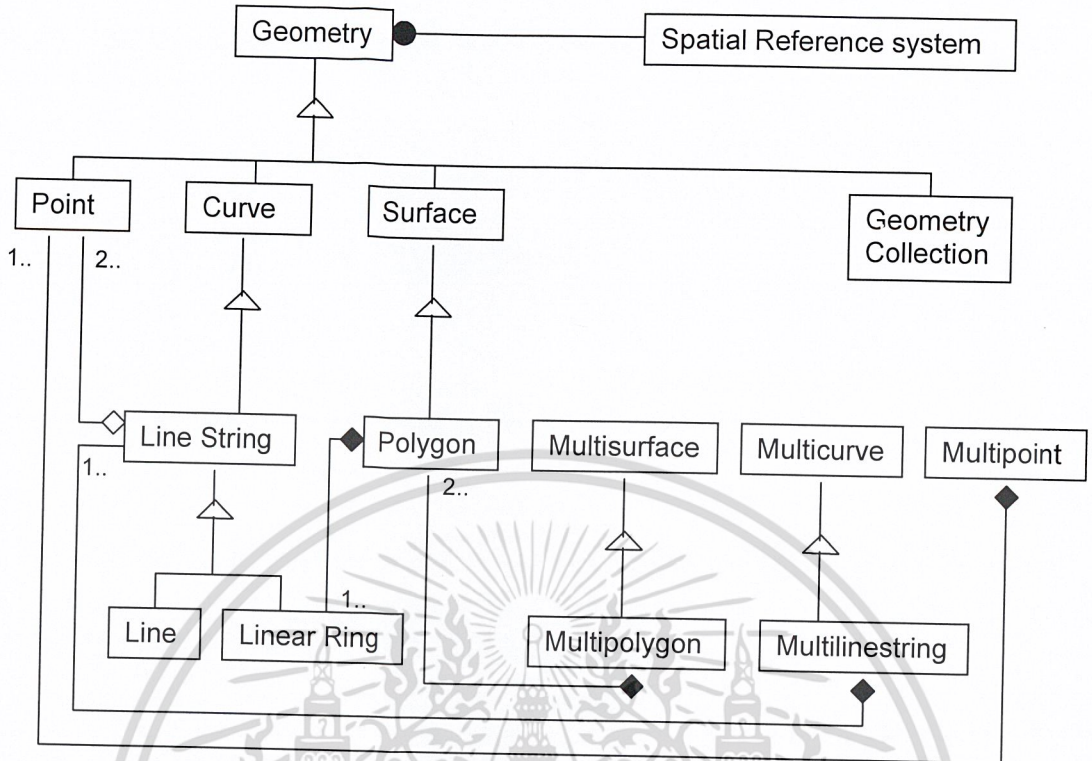
เราสามารถใช้งานการจำลองข้อมูลทั้งสองแบบรวมกันได้โดยการจำลองโครงสร้างข้อมูลของ วัตถุ และกำหนดความสัมพันธ์ในเชิงพื้นที่ที่เกี่ยวข้องกับวัตถุนั้น

### 3.3 Spatial Data Type

ระบบของชนิดของข้อมูลเชิงพื้นที่ (Spatial data Type or spatial algebras) คือระบบที่ใช้ ในการรวมนิยามพื้นฐานสำหรับจุด (point), เส้น (line) และพื้นที่ (region) เข้าด้วยกันกับ ความสัมพันธ์และวิธีการจัดการขององค์ประกอบนั้นๆ เช่น การกำหนดรูปแบบการตัดผ่านหรือซ้อนทับ กัน (intersection) ของพื้นที่ เป็นต้น

ทางเลือกของชนิดของข้อมูลที่ใช้กันใน Objects-Based Model ของข้อมูลเชิงพื้นที่คือชนิด ของข้อมูลพื้นฐานทั่วไปที่ใช้ในระบบแผนที่ ซึ่งในรูปที่ 3-3 แสดงถึงโครงสร้างและความสัมพันธ์ภายใน ของ spatial geometry ในรูปแบบของ UML

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-3 โครงสร้างและความสัมพันธ์ภายในของ Spatial geometry ในรูปแบบของ UML

### 3.4 Spatial Relationships

ในกลุ่มของกระบวนการ (Operation) ที่นำเสนอโดยพีชคณิตเชิงพื้นที่ (spatial algebras) นั้น ความสัมพันธ์เชิงพื้นที่ (Spatial Relationship) ถือเป็นสิ่งที่สำคัญที่สุดเนื่องจากความสัมพันธ์ที่ทำให้เราสามารถค้นหาวัตถุที่เราต้องการจากความสัมพันธ์ที่เรากำหนดให้โดยผ่านการสืบค้น โดยแบ่งความสัมพันธ์เป็น 3 แบบดังนี้

1. Topology Relationships เช่น อยู่ติดกัน (adjacent) หรืออยู่ภายใน (inside)
2. Direction Relationships เช่น อยู่ด้านบน (above), อยู่ด้านล่าง (below), อยู่ทางเหนือ (north\_of) หรือทางใต้ของ (south\_of)
3. Metric Relationships เช่น วัตถุที่อยู่ห่างกันมากกว่า 100 กิโลเมตร เป็นต้น

### 3.5 ขั้นตอนการออกแบบฐานข้อมูลเชิงพื้นที่

ในการออกแบบฐานข้อมูลเชิงพื้นที่นั้น เราใช้หลักการของการออกแบบฐานข้อมูลโดยทั่วไปคือ ในขั้นตอนแรกจะเป็นการนำข้อมูลทั้งหมดที่เกี่ยวข้องกับแอปพลิเคชันนั้นมาจัดโดยใช้หลักการของ Conceptual data model ซึ่งในขั้นตอนนี้จะให้ความสนใจในการจัดแบ่งชนิดของข้อมูลโดยผ่านแบบจำลองของฐานข้อมูลในแบบต่างๆ เช่น Entity Relationship (ER) Model ซึ่งเราใช้ในการออกแบบฐานข้อมูลของโครงการนี้

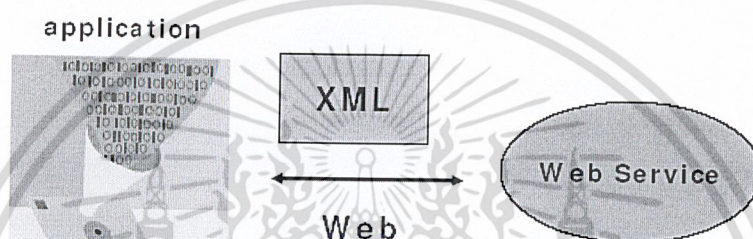


## บทที่ 4

### Web Services

#### 4.1 เทคโนโลยีเว็บเซอร์วิส (Web Services)

Web Services คือแอปพลิเคชันหรือโปรแกรมที่ทำงานอย่างใดอย่างหนึ่งในลักษณะให้บริการ โดยจะถูกเรียกใช้งานแอปพลิเคชันจากโปรแกรมอื่น ๆ (php, asp, java, python) ผ่านทางหน้าเว็บ การให้บริการของ Web Services จะมีเอกสารที่อธิบายคุณสมบัติของการบริการกำกับไว้ และมีการนำเสนอให้สาธารณะชนรับทราบ ผู้ใช้จึงสามารถค้นหา Web Services ได้โดยไม่ต้องรู้ที่อยู่จริงของแอปพลิเคชันหรือโปรแกรมนั้นๆ



รูปที่ 4-1 การทำงานของ Web Services

จากรูปจะเห็นได้ว่าการทำงานของ Web services นั้นจะใช้ภาษา XML เป็นมาตรฐานในการติดต่อแลกเปลี่ยนข้อมูลระหว่างผู้ให้บริการ และผู้ใช้บริการบนอินเทอร์เน็ต โดยการติดต่อข้อมูลกันนั้นจะผ่านทางโปรโตคอลของอินเทอร์เน็ต เช่น โปรโตคอล HTTP เป็นต้น

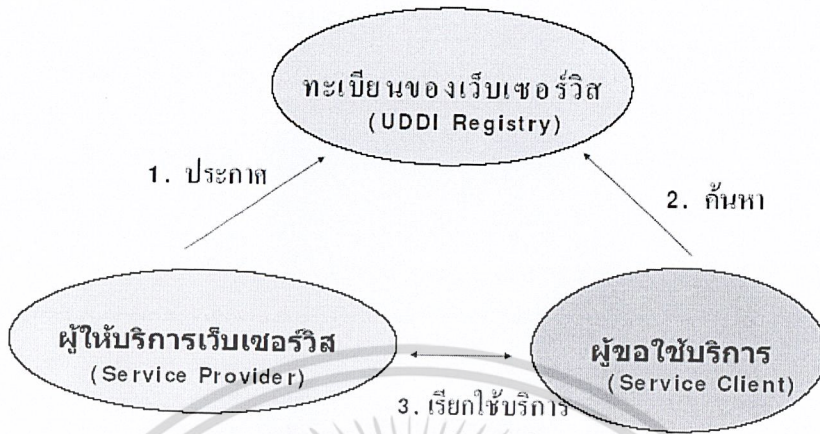
#### 4.2 องค์ประกอบของเว็บเซอร์วิส

องค์ประกอบของเทคโนโลยีเว็บเซอร์วิส ประกอบด้วย

- ผู้ให้บริการเว็บเซอร์วิส (Service Provider)  
เป็นผู้ให้บริการ มีหน้าที่ในการเปิดบริการเพื่อรองรับการขอใช้บริการจาก Requestor ที่เรียกเข้ามาขอใช้
- ทะเบียนเว็บเซอร์วิส (UDDI Registry)  
ทำหน้าที่เป็นตัวกลางให้ Provider มาลงทะเบียนไว้ โดยใช้ WSDL ไฟล์ บอกรายละเอียดของบริษัทและบริการที่มีให้ ซึ่งอาจจะใช้หรือไม่ใช้ก็ได้
- ผู้ใช้บริการ (Service Client)  
เป็นใครก็ตามที่ต้องการเรียกใช้บริการจาก Provider ซึ่งสามารถค้นหาบริการที่ต้องการได้จาก UDDI registry หรือ Service Registry หรือติดต่อจาก Provider โดยตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แสดงได้ดังรูป



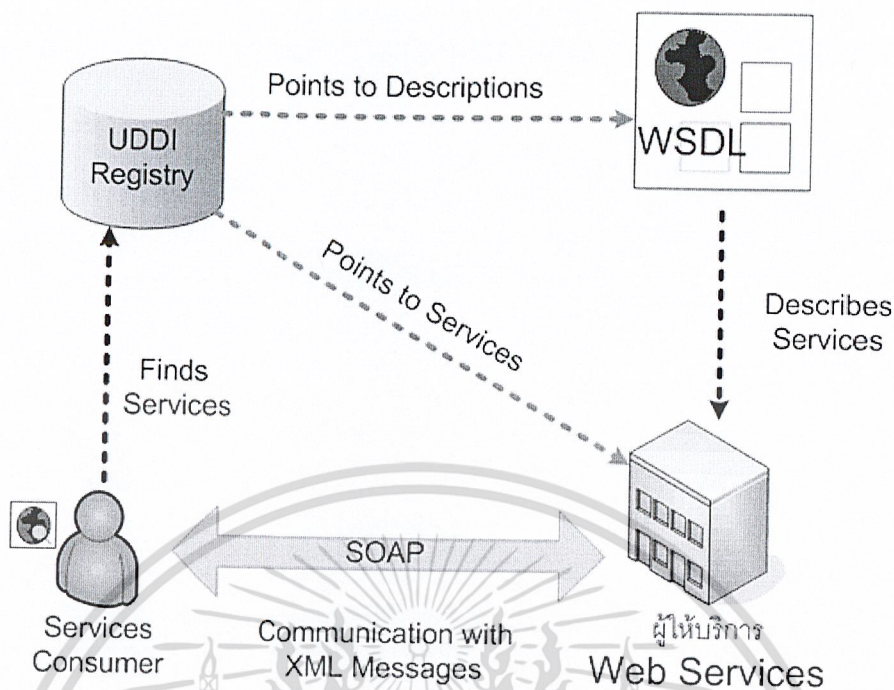
รูปที่ 4-2 องค์ประกอบของ Web Services

4.3 ขั้นตอนการเรียกใช้บริการของเว็บเซอร์วิส

ผู้ขอใช้บริการจะมีขั้นตอนการขอใช้บริการ ดังนี้คือ

1. ผู้ขอใช้บริการจะทำการค้นหาบริการที่ทะเบียนของเว็บเซอร์วิส เพื่อหาเซอร์วิสที่เปิดให้บริการอยู่ เมื่อได้เลือกเซอร์วิสที่ต้องการได้แล้ว ก็จะสามารถรู้ที่อยู่ของคำอธิบายในการใช้เซอร์วิสได้ทันทีจากทะเบียนของเว็บเซอร์วิส
2. ผู้ขอใช้บริการทำการขอคำอธิบายการใช้เซอร์วิส (WSDL) จากที่อยู่ที่ทะเบียนของเว็บเซอร์วิสให้มา
3. ผู้ขอใช้บริการทำการติดต่อขอใช้บริการกับผู้ให้บริการเว็บเซอร์วิส โดยติดต่อกันด้วยมาตรฐาน SOAP

แสดงดังรูป



รูปที่ 4-3 ขั้นตอนการเรียกใช้บริการของ Web Services

#### 4.4 มาตรฐานที่ใช้ในการพัฒนาเว็บเซอร์วิส

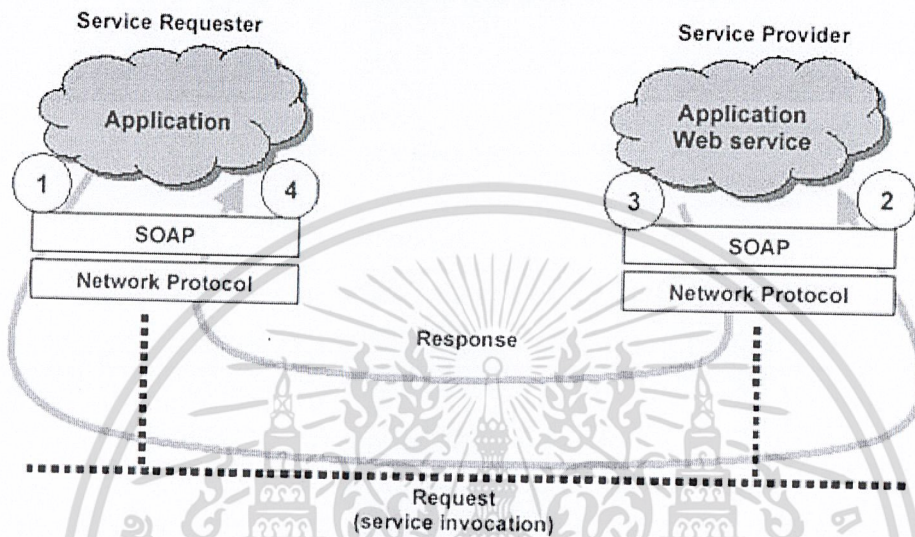
##### 4.4.1 SOAP (Simple Object Access Protocol)

เป็นมาตรฐานที่ใช้เพื่อกำหนดการส่งข้อมูลของผู้ให้บริการและผู้รับบริการ เป็นโปรโตคอลที่ผู้จัดหา Web Services เลือกใช้ที่จะส่ง message ระหว่าง Web Services SOAP เป็น Transport Protocol ที่มี XML เป็นพื้นฐานและใช้ HTTP เป็นโปรโตคอลร่วมในการส่งผ่านเครือข่าย SOAP จะระบุวิธีในการเข้ารหัสส่วนหัว (Header Encoding) ของทั้ง HTTP และไฟล์ XML ให้อย่างชัดเจนทั้งใน ส่วนของการติดต่อไปยังคอมพิวเตอร์เครื่องหนึ่งและส่งผ่านข้อมูลไปให้รวมถึงระบุวิธีที่โปรแกรมซึ่งถูกเรียกนั้นจะส่งค่าคืนกลับมาด้วย

SOAP (Simple Object Access Protocol) เป็น XML-based โปรโตคอล (lightweight protocol) และใช้ HTTP เป็นโปรโตคอลร่วม สำหรับการแลกเปลี่ยนข้อมูลในสภาวะแวดล้อมแบบกระจายศูนย์ (decentralized, distributed environment) SOAP ได้ กำหนดเมสเซจจิงโปรโตคอล (Messaging Protocol) ระหว่างผู้ขอบริการ (requestor) กับผู้ให้บริการ (provider) เช่น ผู้ขอบริการสามารถติดต่อแลกเปลี่ยนข้อมูลกับผู้ให้บริการโดยใช้ RMI (Remote Method Invocation) ตามวิธีการของ โปรแกรมแบบออปเจ็ค บริษัทไมโครซอฟท์, ออปีเอ็ม, โลตัส, ยูสเซอร์แลนด์ (UserLand) และ ดีเวลลอปเปอร์เมนเตอร์ (DeveloperMenter) ได้ร่วมกันกำหนดมาตรฐานของ SOAP ขึ้น ซึ่งต่อมาได้มีบริษัทอีก 30 กว่าบริษัทเข้าร่วมและ จัดตั้งเป็น W3C XML Protocol Workgroup ขึ้น SOAP ได้กำหนดรูปแบบพื้นฐานของการสื่อสารแบบกระจายขึ้นโดย การพัฒนา SOA แม้ว่า SOA จะไม่ได้กำหนดเมสเซจจิงโปรโตคอล (Messaging Protocol) ไว้ แต่ SOAP ได้ถูก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำหนด ให้เป็น Services-Oriented Architecture Protocol เรียบร้อยแล้ว เนื่องจากมันได้ถูกใช้ในการพัฒนา SOA อย่างแพร่หลายแล้วนั่นเอง จุดเด่นของ SOAP ก็คือเป็นโปรโตคอลที่เป็นกลาง กล่าวคือ ไม่มีใครเป็นเจ้าของและเป็นโปรโตคอล ที่ทำงานกับโปรโตคอลอื่นหลายชนิด การพัฒนาที่อนุญาตให้ทำได้อย่างอิสระตามแพลตฟอร์มระบบปฏิบัติการ แบบจำลองทางวัตถุ (Object model) และ ภาษาโปรแกรมของผู้ที่ทำการพัฒนา โดยมีการทำงานดังรูป



รูปที่ 4-4 ขั้นตอนการส่ง SOAP message

มีขั้นตอนในการส่ง SOAP เพื่อทำการติดต่อระหว่างผู้เรียกใช้บริการกับผู้ให้บริการดังนี้

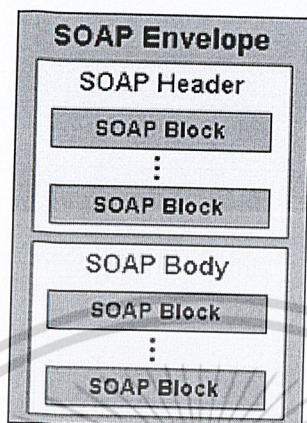
1. แอปพลิเคชันของผู้ร้องขอบริการสร้าง SOAP message เพื่อเรียกใช้บริการของเว็บเซอร์วิส
2. เว็บเซอร์วิสของผู้ให้บริการ ผู้ให้บริการได้รับ SOAP message ของผู้ร้องขอ ซึ่งอยู่ในรูปของ XML
3. เว็บเซอร์วิสประมวลผลต่างคอมพิวเตอร์ที่ให้บริการเว็บเซอร์วิส ส่งผลลัพธ์มาแล้วผู้ให้บริการก็จะสร้าง SOAP message ที่มีผลลัพธ์นั้นส่งกลับมายังผู้ร้องขอบริการ
4. แอปพลิเคชันของผู้ร้องขอบริการได้รับผลลัพธ์ที่เป็น SOAP message แล้วทำการแปลงให้อยู่ในรูปแบบที่ต้องการ เพื่อนำไปประมวลผลต่อ

#### 4.4.1.1 โครงสร้างของ SOAP

เอกสาร SOAP นั้นมีโครงสร้างในรูปแบบ XML ซึ่งเราสามารถแบ่งเป็นส่วนของเอกสารได้เป็น 3 ส่วนหลักดังนี้คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- SOAP envelop เนื้อหาสาระ (Content) ของเอกสารทั้งหมด
- SOAP header ส่วนเพิ่มเติมของเอกสาร SOAP ซึ่งจะมีก็ได้ หรือไม่มีก็ได้
- SOAP body ส่วนที่ใช้ในการเรียกใช้งานเซอร์วิส และผลลัพธ์ที่ได้จากเซอร์วิส



รูปที่ 4-5 โครงสร้างของเอกสาร SOAP

ตัวอย่างเอกสาร SOAP อย่างง่ายของการสอบถามราคา ดอกกุหลาบ จำนวน 100 ดอก

```
<soap:Envelope>
  <soap:Body>
    <GetPrice>
      <Item>Rose</Item>
      <Quantity>100</Quantity>
    </GetPrice>
  </soap:Body>
</soap:Envelope>
```

#### 4.4.1.2 Element ต่างๆ ของ SOAP

##### Element <Envelope>

จะเป็น Root ของเอกสาร XML SOAP เสมอ โดยจากตัวอย่างจะใช้ Namespaces จาก "http://schemas.xmlsoap.org/soap/envelope/" (Schema for the SOAP/1.1 envelope)

ดังนั้น Element ต่างๆ ในตัวอย่างนี้ที่อยู่ในเอกสาร XML ส่วนที่เป็นมาตรฐานของ SOAP เราจะมี Namespaces ที่เราตั้งเป็น soap (<soap:Envelope>, <soap:Header>, <soap:Body>, <soap:Fault>

##### Attribute ใน Envelop

Attribute encodingStyle ในตัวอย่างมีการอ้างการ encoding จาก "http://schemas.xmlsoap.org/soap/encoding/" (Schema for the SOAP/1.1 encoding) จะใช้ในกรณีที่เรามีการใช้ Parameter หลายชนิดข้อมูลในเอกสาร (Data Type) เช่น Boolean, String, Integer เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### Element <Header>

จากตัวอย่างมีการเพิ่มเนื้อหาของเอกสาร SOAP เข้าไปในส่วนของ Header ในที่นี้คือ สกุลเงิน (Currency) ซึ่งส่วนนี้จะใช้ในส่วนของ Application ไม่ได้เป็นส่วนของมาตรฐานของ SOAP แต่ผู้ใช้งานเป็นผู้กำหนดค่าเอง (User-defined)

### Attribute ใน Header

ใน <Header> Element อาจจะมีการใส่ Attribute mustUnderstand เพื่อให้ฝั่งที่รับเอกสาร SOAP ให้ทำการประมวลผลในส่วน Header ด้วยโดยกำหนดค่าเป็น Boolean เช่น

```
<i:local xmlns:i="http://www.i3t.or.th/ws/">
  <i:country mustUnderstand="1">Thailand</i:country>
  <i:currency mustUnderstand="0">Bath</i:currency>
</i:local>
```

ค่าของ mustUnderstand ถ้าเป็น "0" แสดงว่าไม่ต้องประมวลผล ถ้าเป็น "1" : true คือจำเป็นต้องมีการประมวลผล (โดยค่า default = "0" : fault)

### Element <Body>

โดยปกติแล้วเอกสาร SOAP จะต้องมีส่วนของ Body เพราะเป็นส่วนเป็นเนื้อหาสาระจริงๆ ของ SOAP จากตัวอย่างเราต้องการสอบถามราคาของดอกกุหลาบจำนวน 100 ดอก ซึ่ง <GetPrice>, <Item> และ <Quantity> เป็น Element ที่ใช้งานใน Application ไม่ใช่มาตรฐานของ SOAP

### Element <Fault>

เป็นส่วนที่จะถูกใช้เมื่อมีข้อผิดพลาดในการประมวลผลของเอกสาร SOAP ซึ่งโดยปกติจะเห็นเฉพาะที่เป็นเอกสารตอบกลับเท่านั้น (Reply Message หรือ Answer Message) ซึ่งตัวอย่างจะเป็นเอกสารที่เป็นการร้องขอบริการ (Request Message) ซึ่งไม่น่าจะมีส่วนของ <Fault> Element แต่ที่เขียนไว้เพื่อให้เห็นภาพรวมของโครงสร้างเอกสาร SOAP ทั้งหมดก่อน ตัวอย่างข้อมูลใน <Fault> Element ที่มีการเกิดข้อผิดพลาด

```
<soap:Fault>
  <faultcode>Client</faultcode>
  <faultstring>Invalid Request</faultstring>
</soap:Fault>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ตารางอธิบาย Fault Element

Sub Element	คำอธิบาย
<faultcode>	ชื่อของการเกิดข้อผิดพลาด (ดูตารางด้านล่าง)
<faultstring>	คำอธิบายของข้อผิดพลาด
<faultactor>	ชื่อของแหล่งที่เกิดข้อผิดพลาด
<detail>	ข้อมูลรายละเอียดเพิ่มเติมของข้อผิดพลาดนั้น

ตารางที่ 4-1 Fault Element

### ตารางอธิบาย faultcode Sub Element

Error (faultcode)	คำอธิบาย
VersionMismatch	มีข้อผิดพลาดของการอ้าง Namespace ของ Element
MustUnderstand	ในการประมวลผลของ Sub Element ใน Header ที่มีการกำหนด attribute mustUnderstand เป็น “1” ไม่สามารถประมวลผลได้ (not understood)
Client	เอกสาร SOAP มีข้อผิดพลาดอันเนื่องมาจากมีโครงสร้างทางผิด หรือมีข้อมูลที่ไม่ถูกต้อง
Server	มีข้อผิดพลาดจาก Server ไม่สามารถประมวลผลเอกสารได้

ตารางที่ 4-2 Faultcode Sub Element

ท่านจะเห็นว่า SOAP Envelope ก็เป็นเอกสารที่อยู่ในรูปแบบ XML ทั่วไป แต่สิ่งที่ทำให้ SOAP มีความสามารถมากขึ้นก็คือ SOAP ได้มีการกำหนดโครงสร้างของเอกสารเป็นส่วนๆ โดยอธิบายว่าส่วนใดมีหน้าที่อะไร (What) มีข้อมูลอะไรอยู่ในส่วนนั้น และใคร (Who) คือผู้ที่จะต้องสนใจในส่วนนั้น ๆ และกล่าวถึงว่าส่วนใดจำเป็นที่ต้องมีในเอกสาร และส่วนใดเป็นส่วนที่เพิ่มเติมอาจจะมีหรือไม่มีส่วนนั้นก็ขึ้นอยู่กับความจำเป็นในการใช้งาน ทำให้การพัฒนาโปรแกรมมีทิศทางที่ค่อนข้างชัดเจน แต่สามารถเพิ่มขยายได้ในอนาคตอีกด้วย

#### 4.4.2 WSDL (Web Services Description Language)

เป็นภาษาที่ใช้อธิบายคุณลักษณะการใช้งานบริการของ Web Services และวิธีการติดต่อกับ Web Services ความต้องการของนิยามนี้เกี่ยวเนื่องกับความต้องการของ distributed system ที่ จะกำหนด Interface Definition Language (IDL) โดยใช้ภาษา XML, WSDL เกิดจากการรวมแนวคิดของ NASSL (The Network Accessible Service Specification Language), WDS (Well-Defined Services) ของบริษัทไอบีเอ็ม, SDL (The Service Description Language) และ SCL (the SOAP Contract Language) ของบริษัทไมโครซอฟท์ ปัจจุบัน WSDL เป็นภาษาที่เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อยู่ในการดูแลของ W3C (World Wide Web Consortium) ซึ่งยังไม่เป็นมาตรฐานที่สมบูรณ์ เวอร์ชันที่ใช้งานอยู่ใน ปัจจุบันคือ WSDL 1.1 (รายละเอียดเพิ่มเติมเกี่ยวกับ WSDL สามารถศึกษาเพิ่มเติมได้ที่ <http://www.w3c.org/TR/wsdl>)

#### 4.4.2.1 โครงสร้างเอกสาร WSDL

WSDL เป็นภาษาที่อยู่ในความดูแลขององค์กร W3C (World Wide Web Consortium) version ที่มีอยู่ในปัจจุบัน คือ WSDL 1.1 ในการใช้งานจริง หากเราสร้างบริการ Web Services ก็จะมีเครื่องมือช่วยสร้างเอกสาร WSDL สำหรับ Web Services อย่างอัตโนมัติ จุดภายในเอกสารที่เราควรรู้เกี่ยวกับการติดต่อและเรียกใช้บริการของ Web Services มีจุดที่ควรรู้ ดังนี้

##### Element <portType>

เป็นส่วนที่สำคัญที่สุดใน WSDL element อธิบาย operations ที่ web service มีให้บริการและ messages ที่เกี่ยวข้อง เทียบได้กับ function library หรือ module หรือ class ในการเขียนโปรแกรม

##### Element <operation>

อธิบาย method ที่ให้บริการ Web Services หนึ่งจะมี method จำนวนกี่ method ก็ได้

##### Element <message>

อธิบาย data elements ของ operation แต่ละ message อาจมีมากกว่าหนึ่งส่วน เทียบได้กับ parameter ของ function ในการเขียนโปรแกรม

##### Element <types>

อธิบายชนิดข้อมูลที่ web service ใช้ เพื่อความเป็นกลาง WSDL ใช้ XML Schema syntax ในการระบุชนิดข้อมูล

##### Element <binding>

อธิบาย format ของ message และ protocol details ในแต่ละ port

##### Element <service>

สำหรับ web server จะมี Web Services จำนวนกี่บริการก็ได้ และ ชื่อ Web Services ก็เป็นตัวจำแนกและบ่งบอกแต่ละบริการซึ่งห้ามมีชื่อซ้ำกัน

ตามทฤษฎีแล้ว ไฟล์เอกสาร WSDL แต่ละไฟล์ สามารถอธิบายคุณลักษณะของบริการ Web Services ได้มากกว่า 1 บริการโดยแต่ละ Web Services จะมี port สื่อสารเฉพาะตัว ซึ่งบ่งบอกไว้ในเอกสาร WSDL อยู่แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

เป็นมาตรฐานที่ให้ชุดพื้นฐาน APIs (Application Programming Interface) ของ SOAP ที่สามารถนำมาใช้ในการพัฒนาเป็นตัวแทนของผู้ให้บริการ (Service broker) UDDI ใช้สำหรับค้นหา Service ที่ต้องการและเมื่อได้มาแล้ว UDDI ยังจัดหาข้อตกลงในวิธีการที่จะใช้งานเปรียบได้กับสมุดหน้าเหลือง เป็นมาตรฐานที่จัดตั้งขึ้นโดยบริษัทไอบีเอ็ม บริษัทไมโครซอฟต์ และบริษัทอาร์riba (Ariba) ปัจจุบันมีบริษัทที่ร่วม กันกำหนดมาตรฐานของ UDDI มากกว่า 70 บริษัท ซึ่งมาตรฐานของ UDDI ถูกกำหนดให้เป็นมาตรฐานสำหรับ B2B interoperability (รายละเอียดเพิ่มเติมเกี่ยวกับ UDDI สามารถศึกษาเพิ่มเติมได้ที่ <http://uddi.org>)

#### 4.5 ตัวอย่างมาตรฐานที่ใช้ในเว็บเซอร์วิส

- SOAP

ตัวอย่างต่อไปนี้เป็นการส่ง SOAP เพื่อไปขอแผนที่ โดยมี Host อยู่ที่ <http://161.246.5.68/axis/services/MapServices> และทำการเรียกใช้บริการ getMap ซึ่งต้องใส่พารามิเตอร์ 5 ตัว ได้แก่ ละติจูดและลองจิจูดของตำแหน่งเริ่มต้นกับตำแหน่งสุดท้าย และรูปแบบของแผนที่ที่ต้องการ โดยมี SOAP ดังนี้

```
POST /InStock HTTP/1.1
```

```
Host: 161.246.5.68/axis/services/MapServices
```

```
Content-Type: application/soap+xml; charset=utf-8
```

```
Content-Length: nnn
```

```
<?xml version="1.0"?>
```

```
<soap:Envelope
```

```
xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
```

```
soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
```

```
<soap:Body xmlns:m="http://161.246.5.68/axis/services/MapServices">
```

```
<m:getMap>
```

```
<m:lat1>100.0</m:lat1>
```

```
<m:lon1>13.5</m:lon1>
```

```
<m:lat2>100.5</m:lat2>
```

```
<m:lon2>13.75</m:lon2>
```

```
<m:outtype>SVG</m:outtype>
```

```
</m:getMap>
```

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

```
</soap:Envelope>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **WSDL**

ตัวอย่างนี้เป็นการแสดง WSDL ของ Map Services ที่มีฟังก์ชัน getMap เอาไว้ให้บริการแผนที่

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions targetNamespace="urn:servicesmap"
xmlns:impl="urn:servicesmap" xmlns:intf="urn:servicesmap"
xmlns:apacheSOAP="http://xml.apache.org/xml-soap"
xmlns:wsdlSOAP="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:SOAPENC="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:XSD="http://www.w3.org/2001/XMLSchema"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:types>
    <schema xmlns="http://www.w3.org/2001/XMLSchema"
targetNamespace="urn:servicesmap">
      <import namespace="http://schemas.xmlsoap.org/soap/encoding/"/>
    </schema>
  </wsdl:types>

  <wsdl:message name="getMapResponse">
    <wsdl:part name="getMapReturn" type="xsd:string"/>
  </wsdl:message>

  <wsdl:message name="getMapRequest">
    <wsdl:part name="user" type="xsd:string"/>
    <wsdl:part name="password" type="xsd:string"/>
    <wsdl:part name="lat_tl" type="xsd:string"/>
    <wsdl:part name="lon_tl" type="xsd:string"/>
    <wsdl:part name="lat_rb" type="xsd:string"/>
    <wsdl:part name="lon_rb" type="xsd:string"/>
    <wsdl:part name="outtype" type="xsd:string"/>
  </wsdl:message>

  <wsdl:portType name="MapServices">
    <wsdl:operation name="getMap" parameterOrder="user password lat_tl
lon_tl lat_rb lon_rb outtype">
      <wsdl:input name="getMapRequest" message="impl:getMapRequest"/>
      <wsdl:output name="getMapResponse"
message="impl:getMapResponse"/>
    </wsdl:operation>
  </wsdl:portType>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<wsdl:binding name="MapServicesSoapBinding"
type="impl:MapServices">
  <wsdlsoap:binding style="rpc"
transport="http://schemas.xmlsoap.org/soap/http"/>
  <wsdl:operation name="getMap">
    <wsdlsoap:operation soapAction=""/>
    <wsdl:input name="getMapRequest">
      <wsdlsoap:body use="encoded"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="urn:servicesmap"/>
    </wsdl:input>
    <wsdl:output name="getMapResponse">
      <wsdlsoap:body use="encoded"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="urn:servicesmap"/>
    </wsdl:output>
  </wsdl:operation>
</wsdl:binding>
<wsdl:service name="MapServicesService">
  <wsdl:port name="MapServices"
binding="impl:MapServicesSoapBinding">
    <wsdlsoap:address
location="http://localhost/axis/services/MapServices"/>
  </wsdl:port>
</wsdl:service>
</wsdl:definitions>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### การออกแบบระบบระบุตำแหน่ง

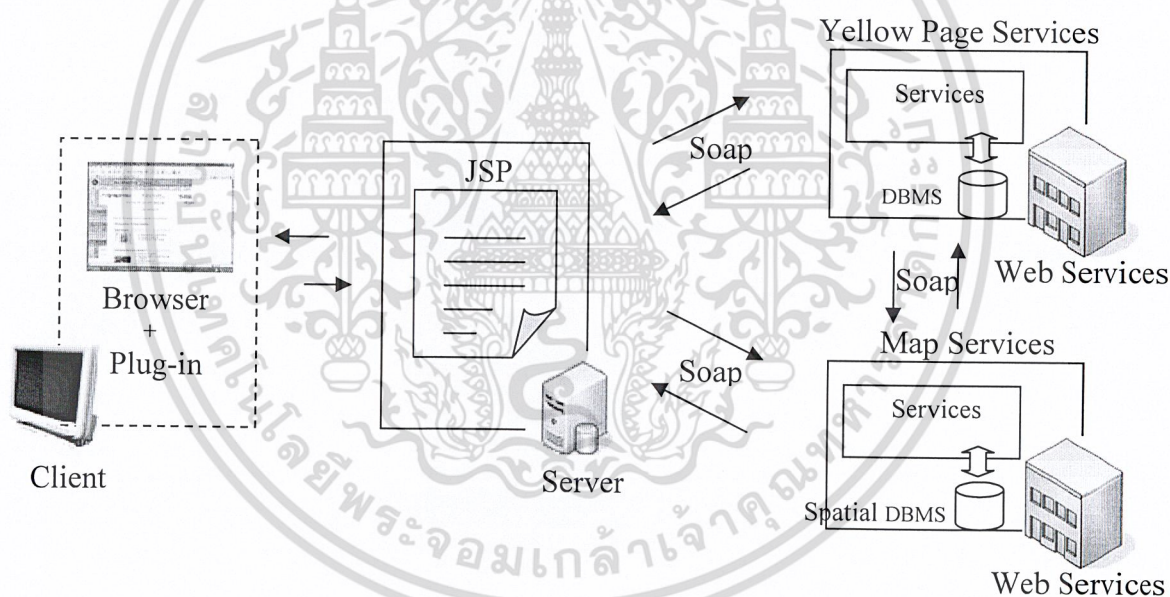
#### 5.1 บทนำ

ระบบระบุตำแหน่งนี้ จะเป็นการออกแบบเพื่อให้เห็นผลและทำงานได้บนเว็บเบราว์เซอร์ ซึ่งสามารถแบ่งตามลักษณะการทำงานได้เป็น 3 ส่วน คือ

1. การทำงานในส่วนของเว็บเซิร์ฟเวอร์
2. การทำงานในส่วนของเว็บเบราว์เซอร์
3. การทำงานในส่วนของไคลเอนท์

โดยแต่ละส่วนของการทำงานนั้น จะมีการทำงานที่ต่างกัน ซึ่งการติดต่อกันของแต่ละส่วนนั้นจะแสดงในหัวข้อถัดไป

#### 5.2 โครงสร้างและการทำงานของระบบระบุตำแหน่งบนเว็บเบราว์เซอร์



รูปที่ 5-1 โครงสร้างการทำงานของระบบระบุตำแหน่งบนเว็บเบราว์เซอร์

โครงสร้างของระบบระบุตำแหน่งที่ให้บริการบนเว็บเบราว์เซอร์นั้น แต่ละส่วนมีการทำงานดังนี้

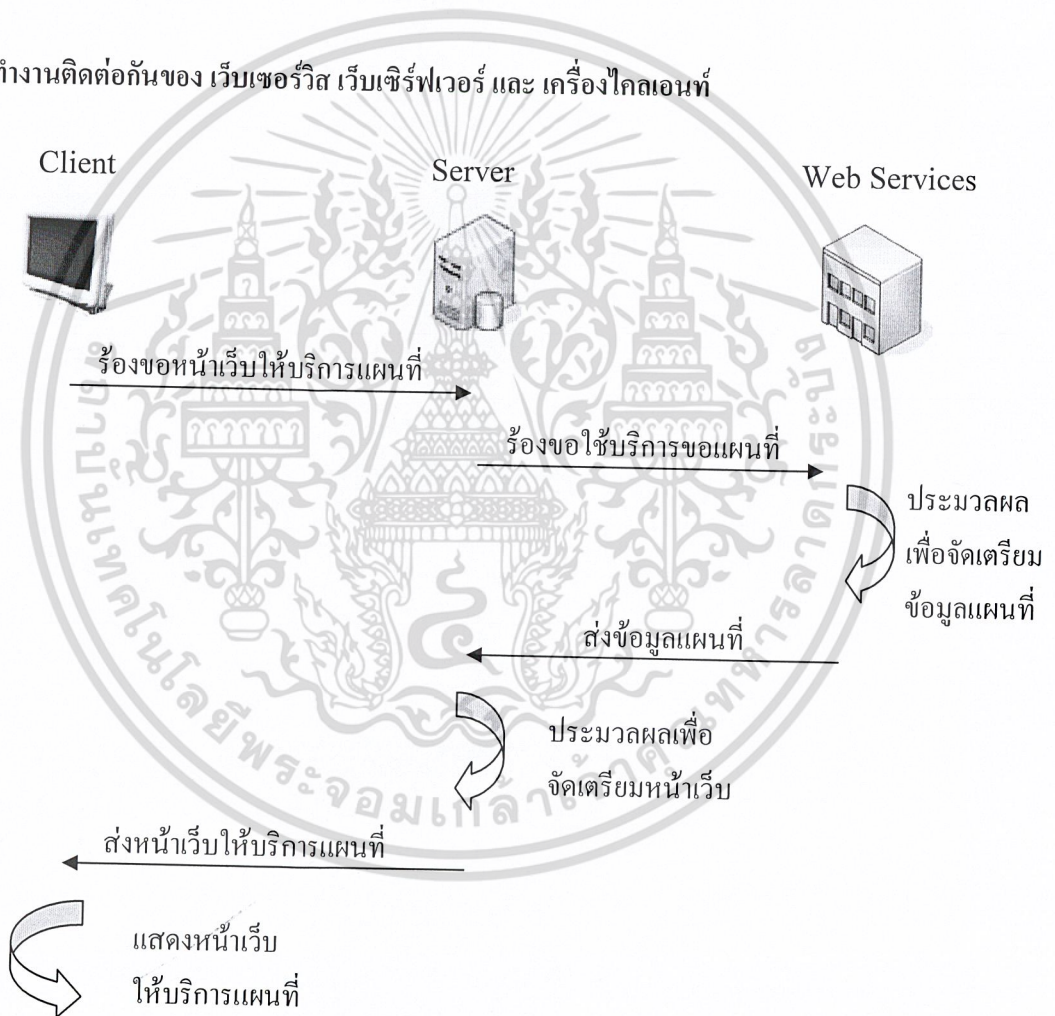
1. ส่วนของเว็บเซิร์ฟเวอร์ จะทำหน้าที่ในการให้บริการเซิร์ฟเวอร์ต่างๆ ซึ่งเว็บเซิร์ฟเวอร์จะมีการติดต่อกับฐานข้อมูลเพื่อใช้ในการเก็บและคำนวณข้อมูลต่างๆ เพื่อนำมาให้บริการ โดยเมื่อเว็บเซิร์ฟเวอร์ต้องการร้องขอใช้บริการ จะทำการส่ง SOAP message ไปยังเว็บเซิร์ฟเวอร์ จากนั้นเว็บเซิร์ฟเวอร์จะทำการประมวลผล แล้วส่งผลลัพธ์ที่ประมวลผลได้กลับไปให้ทางเว็บเซิร์ฟเวอร์โดยผ่านทาง SOAP message เช่นเดียวกัน ซึ่งเว็บเซิร์ฟเวอร์สามารถที่จะติดต่อกับเว็บเซิร์ฟเวอร์อื่นๆ ได้โดยผ่านทาง SOAP message เช่นเดียวกับการติดต่อระหว่าง เว็บเซิร์ฟเวอร์กับเว็บเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ส่วนของเว็บเซิร์ฟเวอร์ จะทำหน้าที่ในการเป็นตัวกลางในการเชื่อมต่อกันระหว่างเครื่องไคลเอนท์กับเว็บเซอร์วิส โดยเครื่องไคลเอนท์ที่ติดต่อเรียกใช้ เว็บเพจจากเว็บเซิร์ฟเวอร์ ทางเว็บเซิร์ฟเวอร์จะไปทำการเรียกใช้บริการจากเว็บเซอร์วิส เพื่อขอข้อมูลแปลงข้อมูลให้อยู่ในรูปแบบที่เครื่องไคลเอนท์ต้องการแล้ว ส่งไปให้เครื่องไคลเอนท์เพื่อทำการแสดงผลอีกที

3. ส่วนของเครื่องไคลเอนท์ ซึ่งจะเรียกใช้บริการจากอีกเว็บเซิร์ฟเวอร์ โดยเรียกผ่านทางเว็บเบราว์เซอร์ เมื่อเว็บเซิร์ฟเวอร์ส่งผลมาให้เครื่องไคลเอนท์ เครื่องไคลเอนท์จะนำข้อมูลที่ได้มาทำการแสดงผลต่อไป โดยถ้าเครื่องไคลเอนท์ที่ต้องการจะแสดงผลไฟล์ SVG เว็บเบราว์เซอร์จำเป็นที่จะต้อง SVG plug-in ซึ่งเป็น viewer ในการแสดงผลไฟล์ SVG โดยในโครงการนี้ใช้ SVG plug-in ของบริษัท Adobe

### 5.3 การทำงานติดต่อกันของ เว็บเซอร์วิส เว็บเซิร์ฟเวอร์ และ เครื่องไคลเอนท์



รูปที่ 5-2 การติดต่อกันของ Web Services, Server และ Client

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 5.4 การออกแบบระบบในส่วนต่างๆ ของระบบระบุตำแหน่ง

### 5.4.1 การออกแบบในส่วนของเว็บเซอร์วิส

ในส่วนของเว็บเซอร์วิสนั้น ประกอบด้วย

1. ฐานข้อมูลแผนที่
2. บริการต่างๆ

โดยในระบบระบุตำแหน่งออกแบบเว็บเซอร์วิสไว้ 2 เว็บเซอร์วิส เพื่อให้บริการที่ต่างกัน ดังนี้

#### 5.4.1.1 เว็บเซอร์วิสที่ให้บริการแผนที่ (Map Services)

เว็บเซอร์วิสที่ให้บริการแผนที่นั้น มีหน้าที่ในการให้บริการเกี่ยวกับแผนที่ และสถานที่ต่างๆ โดยมีการออกแบบ ดังนี้

##### 5.4.1.1.1 การออกแบบฐานข้อมูลแผนที่

ฐานข้อมูลที่ใช้ในการเก็บข้อมูลรายละเอียดต่างๆ ของจังหวัดกรุงเทพมหานคร เดิมทีโครงการที่แล้วได้เลือกใช้ Microsoft SQL Server 2000 เป็นระบบในการจัดการฐานข้อมูล (Database Management System: DBMS) แต่ในโครงการนี้ต้องการเก็บข้อมูลในรูปแบบของฐานข้อมูลเชิงพื้นที่ (Spatial database) ซึ่งตัวระบบในการจัดการฐานข้อมูลเดิมไม่สนับสนุนความสามารถนี้ ทางผู้จัดทำจึงได้เลือกใช้ MySQL Server เป็นระบบในการจัดการฐานข้อมูล โดยได้รับความอนุเคราะห์ข้อมูลสถานที่และถนนของกรุงเทพมหานครจากบริษัท ThaiMapGuide

##### 5.4.1.1.2 ชนิดของข้อมูลในระบบจัดการฐานข้อมูลเชิงพื้นที่ MySQL Server

ในการที่เราจะสร้างฐานข้อมูลเชิงพื้นที่บน MySQL Server นั้น เราจำเป็นต้องทราบถึงชนิดของข้อมูลที่ MySQL Server สนับสนุนได้ ซึ่ง MySQL Server ได้ทำการออกแบบไว้แล้วโดยยึดมาตรฐานของ OpenGis Consortium ลักษณะชนิดของข้อมูลดังที่กล่าวมามีดังนี้

1. GEOMETRY
2. POINT
3. LINESTRING
4. POLYGON

GEOMETRY นั้นเป็นชนิดของข้อมูลที่จะสามารถเก็บชนิดของข้อมูลตั้งแต่ข้อ 2. ถึงข้อ 4. ได้ และในแต่ละชนิดตั้งแต่ข้อ 2. ถึงข้อ 4. นั้นก็จะจัดการกับชนิดข้อมูลของตนเองเท่านั้น

5. MULTIPOINT
6. MULTILINESTRING
7. MULTIPOLYGON
8. GEOMETRYCOLLECTION

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GEOMETRYCOLLECTION เป็นชนิดของข้อมูลที่สามารถจะเก็บกลุ่มของข้อมูลของทุกชนิดข้อมูล และในแต่ละชนิดตั้งแต่ข้อ 6. ถึงข้อ 8. นั่นก็จะจัดการกับกลุ่มชนิดข้อมูลของตนเองเท่านั้น

#### 5.4.1.1.3 รูปแบบของฟังก์ชัน (Function) ที่ใช้ในการจัดเก็บข้อมูล

ใน MySQL Server นั้นจะมีฟังก์ชันที่ใช้ในการจัดเก็บข้อมูลคิบบนฐานข้อมูลเชิงพื้นที่อยู่ 3 ฟังก์ชัน ได้แก่

1. การสร้างข้อมูลเชิงพื้นที่ที่ใช้ฟังก์ชัน WKT เป็นการสร้างข้อมูลเชิงพื้นที่โดยรับค่าที่อยู่ในรูปแบบของ Well-Known Text representation และสามารถเพิ่มในส่วนของตัวอ้างอิงระบบข้อมูลเชิงพื้นที่ (spatial reference system identifier: SRID) เป็นส่วนเสริมได้
2. การสร้างข้อมูลเชิงพื้นที่ที่ใช้ฟังก์ชัน WKB เป็นการสร้างข้อมูลเชิงพื้นที่โดยรับค่าที่อยู่ในรูปแบบของ Well-Known Binary representation และสามารถเพิ่มในส่วนของตัวอ้างอิงระบบข้อมูลเชิงพื้นที่ (spatial reference system identifier: SRID) เป็นส่วนเสริมได้
3. การสร้างข้อมูลเชิงพื้นที่ที่ใช้ฟังก์ชันเฉพาะของ MySQL Server ซึ่งฟังก์ชันในข้อนี้เป็นรายละเอียดปลีกย่อยที่ไม่เกี่ยวกับเนื้อหาในโครงการนี้จึงจะไม่กล่าวถึง

#### 5.4.1.1.4 เริ่มต้นออกแบบและสร้างฐานข้อมูล

เราจะมีตารางข้อมูลทั้งหมด 9 ตารางดังนี้













1. ตาราง geom เป็นตารางที่เก็บข้อมูลชนิด GEOMETRY ซึ่งจะเป็นตัวบอกว่า ข้อมูลชนิดนั้นเป็นแบบ POIN, LINESTRING หรือ POLYGON และมีตำแหน่งพิกัดเป็นเท่าไร โดยตารางนี้จะประกอบไปด้วยเขตข้อมูล (field) ดังนี้

เขตข้อมูล (Field)	คำอธิบาย
g	ชนิดของข้อมูลแบบ GEOMETRY
layer	ค่าเลเยอร์ของชนิดข้อมูลนั้น



ตารางที่ 5-1 เขตข้อมูลและคำอธิบายของตาราง geom

ซึ่งในตารางข้อมูลมีการกำหนดชนิดของข้อมูลในแต่ละเขตของข้อมูลและ Primary key ของตารางดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟิลด์	ชนิด	การเรียงลำดับ	แอตทริบิวต์	ว่างเปล่า (null)	ค่าปริยาย	เพิ่มเติม	กระทำการ
<input type="checkbox"/> g	geometry			ไม่			     
<input type="checkbox"/> layer	int(4)			ใช่	NULL		     

ดัชนี: ๖

ชื่อคีย์	ชนิด	Cardinality	กระทำการ	ฟิลด์
g	INDEX	ไม่มี	 	g 32

















































รูปที่ 5-3 ชนิดของข้อมูลและ Primary key ของตาราง geom

2. ตาราง Hospital เป็นตารางที่เก็บข้อมูลเกี่ยวกับสถานพยาบาลในกรุงเทพฯ ซึ่งจะแบ่งออกเป็นประเภทย่อยๆ ได้เป็นโรงพยาบาลและคลินิก ประกอบไปเขตข้อมูลต่างๆ ดังนี้



เขตข้อมูล (Field)	คำอธิบาย
Place_ID	หมายเลขอ้างอิงของโรงพยาบาลนั้น
Hospital_Name	ชื่อโรงพยาบาล
Hospital_Type	ประเภทของโรงพยาบาล (โรงพยาบาล / คลินิก)
Tel	หมายเลขโทรศัพท์ของโรงพยาบาล
Website	URL ของเว็บของโรงพยาบาล
OpenTime	เวลาทำการ
Lat	ละติจูด
Lon	ลองจิจูด

ตารางที่ 5-2 เขตข้อมูลและคำอธิบายของตาราง Hospital

ซึ่งในตารางข้อมูลมีการกำหนดชนิดของข้อมูลในแต่ละเขตของข้อมูลและ Primary key ของตารางดังนี้

ฟิลด์	ชนิด	การเรียงลำดับ	แอตทริบิวต์	ว่างเปล่า (null)	ค่าปริยาย	เพิ่มเติม	กระทำการ
<input type="checkbox"/> Place_ID	int(4)			ไม่	0		     
<input type="checkbox"/> Hospital_Name	varchar(100)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> Hospital_Type	varchar(30)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> Tel	decimal(9,0)			ใช่	NULL		     
<input type="checkbox"/> Website	varchar(50)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> OpenTime	varchar(50)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> Lat	float			ใช่	NULL		     
<input type="checkbox"/> Lon	float			ใช่	NULL		     

ดัชนี: ๖

ชื่อคีย์	ชนิด	Cardinality	กระทำการ	ฟิลด์
PRIMARY	PRIMARY	0	 	Place_ID

รูปที่ 5-4 ชนิดของข้อมูลและ Primary key ของตาราง Hospital





































เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


3. ตาราง PoliceStation เป็นตารางที่เก็บข้อมูลเกี่ยวกับสถานีตำรวจในกรุงเทพฯ ซึ่งจะแบ่งออกเป็นประเภทย่อยๆ คือสถานีตำรวจและป้อมตำรวจ ประกอบไปด้วยเขตข้อมูลต่างๆ ดังนี้



เขตข้อมูล (Field)	คำอธิบาย
Place_ID	หมายเลขอ้างอิงของสถานีตำรวจ
Police_Name	ชื่อสถานีตำรวจ
Police_Type	ประเภทของสถานีตำรวจ (สถานีตำรวจ / ป้อม)
Tel	หมายเลขโทรศัพท์ของสถานีตำรวจ
Lat	ละติจูด
Lon	ลองจิจูด

ตารางที่ 5-3 เขตข้อมูลและคำอธิบายของตาราง Police

ซึ่งในตารางข้อมูลมีการกำหนดชนิดของข้อมูลในแต่ละเขตของข้อมูลและ Primary key ของตารางดังนี้

ฟิลด์	ชนิด	การเรียงลำดับ	แอตทริบิวต์	ว่างเปล่า (null)	ค่าปริยาย	เพิ่มเติม	กระทำการ
<input type="checkbox"/> Place_ID	int(4)			ไม่	0		     
<input type="checkbox"/> Police_Name	varchar(255)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> Police_Type	varchar(20)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> Tel	decimal(9,0)			ใช่	NULL		     
<input type="checkbox"/> Lat	float			ใช่	NULL		     
<input type="checkbox"/> Lon	float			ใช่	NULL		     

ดัชนี: 

ชื่อคีย์	ชนิด	Cardinality	กระทำการ	ฟิลด์
PRIMARY	PRIMARY	0	 	Place_ID

รูปที่ 5-5 ชนิดของข้อมูลและ Primary key ของตาราง Police

4. ตาราง Bank เป็นตารางที่เก็บข้อมูลเกี่ยวกับธนาคารในกรุงเทพฯ ซึ่งจะแบ่งออกเป็นประเภทย่อย คือ ธนาคารและตู้บริการเงินด่วน (ATM) ประกอบไปด้วยเขตข้อมูลต่างๆ ดังนี้

















































เขตข้อมูล (Field)	คำอธิบาย
Place_ID	หมายเลขอ้างอิงของธนาคารหรือตู้บริการเงินด่วน
Bank_Name	ชื่อธนาคารหรือตู้บริการเงินด่วน
Bank_Type	ชนิด (ธนาคาร / ตู้บริการเงินด่วน)
Tel	หมายเลขโทรศัพท์ของธนาคาร
Website	URL ของเว็บธนาคาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



OpenTime	เวลาทำการ
Lat	ละติจูด
Lon	ลองจิจูด

ตารางที่ 5-4 เขตข้อมูลและคำอธิบายของตาราง Bank

ซึ่งในตารางข้อมูลมีการกำหนดชนิดของข้อมูลในแต่ละเขตของข้อมูลและ Primary key ของตารางดังนี้

ฟิลด์	ชนิด	การเรียงลำดับ	แฮทริบิวต์	ว่างเปล่า (null)	ค่าปริยาย	เพิ่มเติม	กระทำการ
<input type="checkbox"/> Place_ID	int(4)			ไม่	0		     
<input type="checkbox"/> Bank_Name	varchar(100)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> Bank_Type	varchar(10)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> Tel	decimal(9,0)			ใช่	NULL		     
<input type="checkbox"/> Website	varchar(50)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> OpenTime	varchar(50)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> Lat	float			ใช่	NULL		     
<input type="checkbox"/> Lon	float			ใช่	NULL		     

ดัชนี: ①

ชื่อคีย์	ชนิด	Cardinality	กระทำการ	ฟิลด์
PRIMARY	PRIMARY	0	 	Place_ID

รูปที่ 5-6 ชนิดของข้อมูลและ Primary key ของตาราง Bank

















































5. ตาราง Mall เป็นตารางที่เก็บข้อมูลเกี่ยวกับห้างสรรพสินค้าในกรุงเทพฯ ประกอบไปด้วยเขตข้อมูลดังนี้

เขตข้อมูล (Field)	คำอธิบาย
Place_ID	หมายเลขอ้างอิงของห้างสรรพสินค้า
Mall_Name	ชื่อห้างสรรพสินค้า
Tel	หมายเลขโทรศัพท์ของห้างสรรพสินค้า
Website	URL ของเว็บของห้างสรรพสินค้า
OpenTime	เวลาทำการ
เขตข้อมูล (Field)	คำอธิบาย
Promotion	รายการส่งเสริมการขายของห้างสรรพสินค้า
Lat	ละติจูด
Lon	ลองจิจูด



ตารางที่ 5-5 เขตข้อมูลและคำอธิบายของตาราง Mall

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งในตารางข้อมูลมีการกำหนดชนิดของข้อมูลในแต่ละเขตของข้อมูลและ Primary key ของตารางดังนี้

ฟิลด์	ชนิด	การเรียงลำดับ	แฮทริวิต	ว่างเปล่า (null)	ค่าปริยาย	เพิ่มเติม	กระทำการ
<input type="checkbox"/> Place_ID	int(4)			ไม่	0		     
<input type="checkbox"/> Mall_Name	varchar(255)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> OpenTime	varchar(20)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> Tel	decimal(9,0)			ใช่	NULL		     
<input type="checkbox"/> Website	varchar(50)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> Promotion	varchar(255)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> Lat	float			ใช่	NULL		     
<input type="checkbox"/> Lon	float			ใช่	NULL		     

ดัชนี: ๒

ชื่อคีย์	ชนิด	Cardinality	กระทำการ	ฟิลด์
PRIMARY	PRIMARY	0	 	Place_ID

รูปที่ 5-7 ชนิดของข้อมูลและ Primary key ของตาราง Mall





























































6. ตาราง Restaurant เป็นตารางที่เก็บข้อมูลเกี่ยวกับร้านอาหารในกรุงเทพฯ ประกอบไปด้วยเขตข้อมูลดังนี้

เขตข้อมูล (Field)	คำอธิบาย
Place_ID	หมายเลขอ้างอิงของร้านอาหาร
Restaurant_Name	ชื่อร้านอาหาร
Restaurant_Type	ชนิดของร้านอาหาร
Tel	หมายเลขโทรศัพท์ของร้านอาหาร
Website	URL ของเว็บของร้านอาหาร
OpenTime	เวลาทำการ
Promotion	รายการส่งเสริมการขายของร้านอาหาร
Menu	รายการอาหารแนะนำ
Lat	ละติจูด
Lon	ลองจิจูด



ตารางที่ 5-6 เขตข้อมูลและคำอธิบายของตาราง Restaurant

ซึ่งในตารางข้อมูลมีการกำหนดชนิดของข้อมูลในแต่ละเขตของข้อมูลและ Primary key ของตารางดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟิลด์	ชนิด	การเรียงลำดับ	แอตทริบิวต์	ว่างเปล่า (null)	ค่าปริยาย	เพิ่มเติม	กระทำการ
<input type="checkbox"/> Place_ID	int(4)			ไม่	0		     
<input type="checkbox"/> Restaurant_Name	varchar(100)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> Restaurant_Type	varchar(10)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> OpenTime	varchar(50)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> Tel	decimal(9,0)			ใช่	NULL		     
<input type="checkbox"/> Website	varchar(50)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> Promotion	varchar(255)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> Menu	varchar(255)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> Lat	float			ใช่	NULL		     
<input type="checkbox"/> Lon	float			ใช่	NULL		     

ดัชนี: ②

ชื่อคีย์	ชนิด	Cardinality	กระทำการ	ฟิลด์
PRIMARY	PRIMARY	0	 	Place_ID

รูปที่ 5-8 ชนิดของข้อมูลและ Primary key ของตาราง Restaurant

7. ตาราง Tour point เป็นตารางที่เก็บข้อมูลเกี่ยวกับสถานที่ท่องเที่ยวในกรุงเทพฯ ประกอบไปด้วยเขตข้อมูลดังนี้

เขตข้อมูล (Field)	คำอธิบาย
Place_ID	หมายเลขอ้างอิงของสถานที่ท่องเที่ยว
Toup_Name	ชื่อสถานที่ท่องเที่ยว
Tour_Type	ชนิดของสถานที่ท่องเที่ยว
Tel	หมายเลขโทรศัพท์ของสถานที่ท่องเที่ยว
Website	URL ของเว็บของสถานที่ท่องเที่ยว
OpenTime	เวลาทำการ
Promotion	รายการส่งเสริมการขายของสถานที่ท่องเที่ยว
Fee	ราคาค่าบริการของสถานที่ท่องเที่ยว
Lat	ละติจูด
Lon	ลองจิจูด

ตารางที่ 5-7 เขตข้อมูลและคำอธิบายของตาราง TourPoint

ซึ่งในตารางข้อมูลมีการกำหนดชนิดของข้อมูลในแต่ละเขตข้อมูลและ Primary key ของตารางดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟิลด์	ชนิด	การเรียงลำดับ	แฮทริบิต	ว่างเปล่า (null)	ค่าปริยาย	เพิ่มเติม	กระทำการ
<input type="checkbox"/> Place_ID	int(4)			ไม่	0		
<input type="checkbox"/> Tour_Name	varchar(50)	utf8_general_ci		ใช่	NULL		
<input type="checkbox"/> Tour_Type	varchar(10)	utf8_general_ci		ใช่	NULL		
<input type="checkbox"/> Tel	decimal(9,0)			ใช่	NULL		
<input type="checkbox"/> Website	varchar(50)	utf8_general_ci		ใช่	NULL		
<input type="checkbox"/> OpenTime	varchar(50)	utf8_general_ci		ใช่	NULL		
<input type="checkbox"/> Promotion	varchar(50)	utf8_general_ci		ใช่	NULL		
<input type="checkbox"/> Fee	int(4)			ใช่	NULL		
<input type="checkbox"/> Lat	float			ใช่	NULL		
<input type="checkbox"/> Lon	float			ใช่	NULL		

ดัชนี: ๖

ชื่อคีย์	ชนิด	Cardinality	กระทำการ	ฟิลด์
PRIMARY	PRIMARY	0		Place_ID

รูปที่ 5-9 ชนิดของข้อมูลและ Primary key ของตาราง Tourpoint





























































8. ตาราง add\_place เป็นตารางที่เก็บข้อมูลเกี่ยวกับสถานที่ที่ผู้ใช้งานเพิ่มเข้ามา ประกอบไปด้วยเขตข้อมูลดังนี้

เขตข้อมูล (Field)	คำอธิบาย
Place_ID	หมายเลขอ้างอิงของสถานที่
Place_Name	ชื่อสถานที่
Place_Type	ชนิดของสถานที่
Tel	หมายเลขโทรศัพท์ของสถานที่
Website	URL ของเว็บของสถานที่
OpenTime	เวลาทำการ
Lat	ละติจูด
Lon	ลองจิจูด
Detail	รายละเอียดของสถานที่
User_Name	ผู้ใช้งาน



ตารางที่ 5-8 เขตข้อมูลและคำอธิบายของตาราง add\_place

ซึ่งในตารางข้อมูลมีการกำหนดชนิดของข้อมูลในแต่ละเขตข้อมูลและ Primary key ของตาราง ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	ฟิลด์	ชนิด	การเรียงลำดับ	แอตทริบิวต์	ว่างเปล่า (null)	ค่าปริยาย	เพิ่มเติม	กระทำการ
<input type="checkbox"/>	Place_ID	int(11)			ไม่	0		     
<input type="checkbox"/>	Place_Name	varchar(100)	latin1_swedish_ci		ใช่	NULL		     
<input type="checkbox"/>	Place_Type	varchar(10)	latin1_swedish_ci		ใช่	NULL		     
<input type="checkbox"/>	Tel	varchar(20)	latin1_swedish_ci		ใช่	NULL		     
<input type="checkbox"/>	Website	varchar(50)	latin1_swedish_ci		ใช่	NULL		     
<input type="checkbox"/>	OpenTime	varchar(20)	latin1_swedish_ci		ใช่	NULL		     
<input type="checkbox"/>	Lat	double			ใช่	NULL		     
<input type="checkbox"/>	Lon	double			ใช่	NULL		     
<input type="checkbox"/>	Detail	text	latin1_swedish_ci		ใช่	NULL		     
<input type="checkbox"/>	User_Name	varchar(10)	latin1_swedish_ci		ไม่			     

ดัชนี: ?

ชื่อคีย์	ชนิด	Cardinality	กระทำการ	ฟิลด์
PRIMARY	PRIMARY	0	 	Place_ID User_Name













รูปที่ 5-10 ชนิดของข้อมูลและ Primary key ของตาราง add\_place

9. ตาราง user เป็นตารางที่เก็บชื่อเข้าใช้งานและรหัสผ่านของผู้ใช้งาน ประกอบไปด้วยเขตข้อมูลดังนี้



เขตข้อมูล (Field)	คำอธิบาย
User_Name	ชื่อเข้าใช้งาน
Password	รหัสผ่าน

ตารางที่ 5-9 เขตข้อมูลและคำอธิบายของตาราง user

ซึ่งในตารางข้อมูลมีการกำหนดชนิดของข้อมูลในแต่ละเขตข้อมูลและ Primary key ของตารางดังนี้

	ฟิลด์	ชนิด	การเรียงลำดับ	แอตทริบิวต์	ว่างเปล่า (null)	ค่าปริยาย	เพิ่มเติม	กระทำการ
<input type="checkbox"/>	User_Name	varchar(10)	latin1_swedish_ci		ไม่			     
<input type="checkbox"/>	Password	varchar(10)	latin1_swedish_ci		ไม่			     

ดัชนี: ?

ชื่อคีย์	ชนิด	Cardinality	กระทำการ	ฟิลด์
PRIMARY	PRIMARY	0	 	User_Name

รูปที่ 5-11 ชนิดของข้อมูลและ Primary key ของตาราง user

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 5.4.1.1.5 การพัฒนาแผนที่โดยใช้ภาษา GML

เมื่อทำการออกแบบฐานข้อมูลให้มีความเหมาะสมกับการใช้งานของระบบแล้ว เราจะนำข้อมูลจากฐานข้อมูลนั้นมาทำให้อยู่ในรูปแบบของโครงสร้างของภาษา GML ซึ่งแบ่งออกเป็นขั้นตอนต่างๆ ดังนี้

1. การสร้าง GML Application Schema
2. การสร้างไฟล์ GML ซึ่งอยู่ในรูปของไฟล์ .XML

##### 5.4.1.1.5.1 การสร้างไฟล์ GML

หลังจากกำหนดโครงสร้างของข้อมูล GML เรียบร้อยแล้ว ในส่วนถัดมาคือการนำข้อมูลจากฐานข้อมูลขึ้นมาใส่ใน Tag ของ GML ตามโครงสร้างที่เราได้กำหนดขึ้นโดยทำการเขียนโปรแกรมอ่านข้อมูลจากฐานข้อมูลที่มีที่ละตารางและตั้งให้เขียนข้อมูลออกไปยังไฟล์ซึ่งมีนามสกุลของไฟล์เป็น .xml โดยในที่นี้กำหนดให้เป็นไฟล์ชื่อ map.xml ตัวอย่างบางส่วนของไฟล์ GML ที่ใช้ในการทำงานจริง

```
<?xml version='1.0' encoding='windows-874'standalone='no'?>
<!--File:Bangkok.xml -->
<CityModel xmlns:gml='http://www.opengis.net/gml'
  xmlns:xlink='http://www.w3.org/1999/xlink'
  xmlns:xsi='http://www.w3.org/2000/10/XMLSchema-instance'
  xsi:noNamespaceSchemaLocation='file:///c:/tomcat-
4.1.18/webapps/ROOT/project/Bangkok.xsd'>
```

ในส่วนแรกเป็นการประกาศนามสเปซต่างๆ ที่เกี่ยวข้องในการเขียนไฟล์ GML โดยตัวอักษรที่เป็นตัวหนาเป็นการระบุถึง GML Application Schema ที่เราเขียนขึ้นโดยเป็นการระบุว่า Application Schema ที่เราอ้างอิงนี้ไม่ได้ระบุนามสเปซและมีตำแหน่งที่เก็บไฟล์คือ c:/tomcat-4.1.18/webapps/ROOT/project/Bangkok.xsd สังเกตว่า Element ที่เป็น Element Root ของไฟล์นี้คือ <CityModel> ตามที่เรากำหนดไว้ใน Application Schema

```
<cityMember>
  <Hospital fid='H2' Classified='1'>
    <gml:name>รพ ราชวิถี</gml:name>
    <Location>
      <gml:Point
srsName='http://www.opengis.net/gml/srs/epsg.xml#4362'>
        <gml:coordinates>13.7649,100.537</gml:coordinates>
      </gml:Point>
    </Location>
  </Hospital>
</cityMember>
--
--
</CityModel>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนถัดมาเป็นการนำข้อมูลจากฐานข้อมูลขึ้นมาทำให้อยู่ในรูปของภาษา GML โดยดึงขึ้นมาที่ละเขตของข้อมูลและใส่ข้อมูลลงไปตามลำดับของ Element ที่เรากำหนดไว้ ซึ่งในที่นี้จะพบว่าข้อมูลของโรงพยาบาลรายชื่อที่อยู่ในตาราง Hospital นั้นมีข้อมูลเพียง 4 column คือ Place\_ID, Hospital\_Name, Lat, Lon ซึ่งเรานำมาเป็นข้อมูลใน Attribute fid และเป็นข้อมูลใน Element <gml:name> และ <Location> ซึ่งใน Element <Location> นั้น เราได้ประกาศ Element ย่อยภายในไว้เป็น <gml:point> ซึ่งมี <gml:coordinate> เป็น Element ย่อยภายในสำหรับเก็บค่าละติจูดกับลองจิจูดซึ่งเรานำขึ้นมาจากเขตของข้อมูล Lat, Lon ในตาราง

ในการนำเขตของข้อมูลในแต่ละตารางมาใส่ใน Element ของ GML แสดงได้ดังตาราง

ตาราง	เขตข้อมูล	GML Element	Attribute
Hospital	Place_ID	<Hospital>	Fid="H"+Place_ID
	Place_Name	<gml:name>	
	Hospital_Type	<Hospital>	Classified
	Tel	<tel>	
	Website	<Website>	
	OpenTime	<OpenTime>	
	Lat	<gml:coordinates>	
	Lon	<gml:coordinates>	

ตารางที่ 5-10 ความสัมพันธ์ของเขตของข้อมูลจากตาราง Hospital และ GML

#### 5.4.1.1.6 การสร้างไฟล์ SVG

เช่นเดียวกับการสร้างไฟล์ GML โดยเราจะนำข้อมูลจากฐานข้อมูลขึ้นมาใส่ใน Tag ของ SVG โดยทำการเขียนโปรแกรมอ่านข้อมูลจากฐานข้อมูลที่มีที่ละตารางและสั่งให้เขียนข้อมูลออกไปยังไฟล์ซึ่งมีนามสกุลของไฟล์เป็น .svg โดยในที่นี้กำหนดให้เป็น map.svg ตัวอย่างบางส่วนของไฟล์ SVG ที่ใช้ในการทำงานจริง

##### 5.4.1.1.6.1 การเพิ่มความสามารถของแผนที่โดยใช้ภาษา SVG DOM

แผนที่ SVG ที่สร้างขึ้นมานั้น เป็นเพียงแค่ภาพแผนที่ที่เป็นภาพนิ่งเท่านั้น ซึ่งในการพัฒนาเราสามารถใช้ความสามารถของภาษา SVG ในการเพิ่มความสามารถของแผนที่ ซึ่งทำได้โดยการใช้อ็อบเจกต์ร่วมกับ SVG DOM เพื่อเพิ่มความสามารถในการปฏิสัมพันธ์กับผู้ใช้ โดยมีฟังก์ชันการทำงาน ดังนี้

1. การย่อขยายแผนที่ เป็นการย่อหรือขยายแผนที่ให้เป็นไปตามสเกลที่ต้องการได้ ไม่จอมนเห็นส่วนที่เกินแผนที่ หรือขยายจนมองไม่เห็นเป็นแผนที่เลย
2. การเลื่อนแผนที่ เป็นการเลื่อนการมองแผนที่ โดยสามารถที่จะเลือกทิศทางที่จะเลือกดูได้ โดยจะมีลูกศรเป็นตัวบอกว่า ถ้าลูกศรชี้ไปทางไหน แสดงว่าต้องการจะดูแผนที่ในทางนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. การเลือกเฉพาะเลเยอร์ที่ต้องการได้ โดยแผนที่ SVG จะมีการแบ่งเลเยอร์ออกเป็น ส่วนต่างๆ ซึ่งสามารถคลิกเพื่อเลือกเลเยอร์ที่ต้องการ หรือไม่ต้องการจะแสดงผลได้
4. การหมุนแผนที่ โดยฟังก์ชันนี้จะทำการหมุนแผนที่ไปตามทิศทางทวนเข็มนาฬิกา หรือตามเข็มนาฬิกา ไปทีละ 5 องศา
5. แสดงแผนที่ขนาดเล็กด้านข้าง ทำให้สามารถคลิกเมาส์แล้วเลื่อนไปแสดงผลในส่วน ที่ต้องการได้ทันที
6. การแสดงพิกัด เมื่อนำเมาส์ไปวางบนแผนที่

#### 5.4.1.1.7 การออกแบบบริการต่างๆ

ในส่วนของบริษัทต่างๆ ที่เกี่ยวข้องกับแผนที่ และสถานที่นั้น จะมีบริการดังนี้

- บริการขอแผนที่ (getMap)
  - เป็นบริการที่ใช้สำหรับขอแผนที่ โดยมีค่าอินพุตและเอาต์พุต ดังนี้
    - ค่าอินพุตที่ต้องการ
      - ค่าละติจูดของตำแหน่งเริ่มต้น
      - ค่าลองจิจูดของตำแหน่งเริ่มต้น
      - ค่าละติจูดของตำแหน่งสุดท้าย
      - ค่าลองจิจูดของตำแหน่งสุดท้าย
      - รูปแบบของแผนที่ที่ต้องการ โดยสามารถกำหนดได้ 3 รูปแบบ คือ GML, SVG และ JPEG
    - ค่าเอาต์พุตที่ออกมา
      - แผนที่ในรูปแบบที่ได้กำหนดไว้
  - บริการขอแผนที่ของบุคคล (getPersonMap)
    - เป็นบริการที่ใช้สำหรับขอแผนที่สถานที่ที่เกี่ยวข้องกับบุคคลที่ต้องการ โดยจำเป็นต้องระบุ ID ของสถานที่ และ ชื่อของสถานที่ ซึ่งผลลัพธ์ที่ได้จะเป็นแผนที่ในรูปแบบของ SVG เท่านั้น โดยมีค่าอินพุตและเอาต์พุต ดังนี้
      - ค่าอินพุตที่ต้องการ
        - ID ของสถานที่
        - ชื่อของสถานที่
      - ค่าเอาต์พุตที่ออกมา
        - แผนที่ในรูปแบบของ SVG
    - บริการขอข้อมูลสถานที่ (getPlaceDetail)
      - เป็นบริการที่ใช้สำหรับขอข้อมูลสถานที่ใดๆ โดยจำเป็นต้องระบุ ID ของสถานที่ หรือ ชื่อของสถานที่อย่างใดอย่างหนึ่ง เพื่อใช้ในการค้นหาในฐานะข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยมีค่าอินพุทและเอาท์พุท ดังนี้

ค่าอินพุทที่ต้องการ

- ID ของสถานที่ที่ต้องการจะรู้ข้อมูล
- ชื่อของสถานที่ที่ต้องการจะรู้ข้อมูล
- สถานที่นั้นๆ เป็นอะไร เช่น ธนาคาร, โรงพยาบาล เป็นต้น

ค่าเอาท์พุทที่ออกมา

- ข้อมูลของสถานที่ที่ต้องการ เช่น ค่าละติจูด, ค่าลองจิจูด, เปิดทำงานกี่โมง เป็นต้น

- บริการเพิ่มสถานที่ในแผนที่ (setPlaceDetail)

เป็นบริการที่ใช้สำหรับเพิ่มสถานที่ใหม่ เข้าไปในระบบ โดยข้อมูลที่มีความจำเป็นที่จะต้องใส่ก็คือ ชื่อของสถานที่, ค่าละติจูด, ค่าลองจิจูด และ type ของสถานที่ โดยมีค่าอินพุทและเอาท์พุท ดังนี้

ค่าอินพุทที่ต้องการ

- ชื่อของสถานที่
- ค่าละติจูดของสถานที่
- ค่าลองจิจูดของสถานที่
- Type ของสถานที่ อาทิเช่น โรงเรียน, โรงพยาบาล, ห้างสรรพสินค้า เป็นต้น
- หน้าเว็บที่จะเชื่อมต่อไปยังสถานที่นั้น
- เวลาเปิดทำการ – ปิดทำการ
- รายละเอียดปลีกย่อยอื่น

ค่าเอาท์พุทที่ออกมา

- ค่ายืนยัน การเพิ่มสถานที่ในระบบ

- บริการแก้ไขข้อมูลสถานที่ (editPlaceDetail)

เป็นบริการที่ใช้สำหรับแก้ไขข้อมูลสถานที่ โดยมีค่าอินพุทและเอาท์พุท ดังนี้

ค่าอินพุทที่ต้องการ

- ชื่อของสถานที่
- ค่าละติจูดของสถานที่
- ค่าลองจิจูดของสถานที่
- Type ของสถานที่ อาทิเช่น โรงเรียน, โรงพยาบาล, ห้างสรรพสินค้า เป็นต้น
- หน้าเว็บที่จะเชื่อมต่อไปยังสถานที่นั้น
- เวลาเปิดทำการ – ปิดทำการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- รายละเอียดปลีกย่อยอื่น
- ค่าเอาท์พุทที่ออกมา
- ค่ายืนยัน การแก้ไขข้อมูลของสถานที่
- ค้นหาสถานที่ (searchPlace)
- เป็นบริการที่ใช้สำหรับค้นหาสถานที่ที่มีชื่อตรงตามที่ผู้ใช้ระบุ โดยมีค่าอินพุทและเอาท์พุท ดังนี้
- ค่าอินพุทที่ต้องการ
- ชื่อสถานที่
  - ชนิดของสถานที่ที่ต้องการค้นหา
- ค่าเอาท์พุทที่ออกมา
- รายชื่อสถานที่ต้องมีชื่อสอดคล้องกับอินพุท

#### 5.4.1.2 เว็บไซต์ที่ให้บริการข้อมูลบุคคล (Yellow Page Services)

เว็บไซต์ที่ให้บริการข้อมูลบุคคลนั้น มีหน้าที่ในการให้บริการข้อมูล เกี่ยวกับบุคคล อาทิเช่น ที่อยู่ เบอร์โทรศัพท์ สถานที่ทำงาน เป็นต้น อีกทั้งยังสามารถของแผนที่ที่ทำงานจะเว็บไซต์นี้ได้อีกด้วย

##### 5.4.1.2.1 การออกแบบฐานข้อมูลแผนที่

เราจะมีตารางข้อมูลทั้งหมด 3 ตาราง ดังนี้











































1. ฐานข้อมูลที่ใช้เก็บรายละเอียดต่างๆ ของบุคคล ประกอบด้วยเขตข้อมูลดังนี้

เขตข้อมูล (Field)	คำอธิบาย
Pid	หมายเลขอ้างอิงของบุคคล
Name	ชื่อบุคคล
Surname	นามสกุลของบุคคล
Bid	หมายเลขอ้างอิงของสถานที่ที่เกี่ยวข้องกับบุคคล
Buliding_name	ชื่อสถานที่ที่เกี่ยวข้องกับบุคคล
Tel	เบอร์โทรศัพท์
Detail	รายละเอียดอื่นๆ



ตารางที่ 5-11 เขตข้อมูลและคำอธิบายของตาราง Yellow\_Page

ซึ่งในตารางข้อมูลมีการกำหนดชนิดของข้อมูลในแต่ละเขตข้อมูลและ Primary key ของตาราง ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟิลด์	ชนิด	การเรียงลำดับ	แฮทริวิต	ว่างเปล่า (null)	ค่าปริยาย	เพิ่มเติม	กระทำการ
<input type="checkbox"/> pid	int(10)			ไม่	0		     
<input type="checkbox"/> name	varchar(20)	utf8_general_ci		ไม่			     
<input type="checkbox"/> surname	varchar(20)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> bid	int(10)			ใช่	NULL		     
<input type="checkbox"/> building_name	varchar(100)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> tel	varchar(15)	utf8_general_ci		ใช่	NULL		     
<input type="checkbox"/> detail	text	utf8_general_ci		ใช่	NULL		     

ดัชนี: 0























































ชื่อคีย์	ชนิด	Cardinality	กระทำการ	ฟิลด์
PRIMARY	PRIMARY	0	 	pid
สร้างดัชนีโดยคอมส์มัน์ 1 <input type="text"/> <input type="button" value="ลงมือ"/>				

รูปที่ 5-12 ชนิดของข้อมูลและ Primary key ของตาราง Yellow\_Page



2. ฐานข้อมูลที่ใช้เก็บรายละเอียดต่างๆ ของบุคคลที่ผู้ใช้งานเพิ่มเข้ามา ประกอบไปด้วย เขตข้อมูลดังนี้

เขตข้อมูล (Field)	คำอธิบาย
Person_ID	หมายเลขอ้างอิงของบุคคล
Person_Name	ชื่อบุคคล
Person_Surname	นามสกุลของบุคคล
Place_ID	หมายเลขอ้างอิงของสถานที่ที่เกี่ยวข้องกับบุคคล
Place_Name	ชื่อสถานที่ที่เกี่ยวข้องกับบุคคล
Address	ที่อยู่ของสถานที่ที่เกี่ยวข้องกับบุคคล
Tel	เบอร์โทรศัพท์
Detail	รายละเอียดของบุคคล
User_Name	ผู้ใช้งาน

ตารางที่ 5-12 เขตข้อมูลและคำอธิบายของตาราง add\_person

ฟิลด์	ชนิด	การเรียงลำดับ	แฮทริวิต	ว่างเปล่า (null)	ค่าปริยาย	เพิ่มเติม	กระทำการ
<input type="checkbox"/> Person_ID	varchar(11)	latin1_swedish_ci		ไม่			     
<input type="checkbox"/> Person_Name	varchar(20)	latin1_swedish_ci		ใช่	NULL		     
<input type="checkbox"/> Person_Surname	varchar(20)	latin1_swedish_ci		ใช่	NULL		     
<input type="checkbox"/> Place_ID	varchar(11)	latin1_swedish_ci		ใช่	NULL		     
<input type="checkbox"/> Place_Name	varchar(50)	latin1_swedish_ci		ใช่	NULL		     
<input type="checkbox"/> Address	text	latin1_swedish_ci		ใช่	NULL		     
<input type="checkbox"/> Tel	varchar(20)	latin1_swedish_ci		ใช่	NULL		     
<input type="checkbox"/> Detail	text	latin1_swedish_ci		ใช่	NULL		     
<input type="checkbox"/> User_Name	varchar(10)	latin1_swedish_ci		ไม่			     

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้











ชื่อคีย์	ชนิด	Cardinality	กระทำการ	ฟิลด์
PRIMARY	PRIMARY	0	 	Person_ID User_Name



รูปที่ 5-13 ชนิดของข้อมูลและ Primary key ของตาราง add\_person

3. ฐานข้อมูลที่ใช้เก็บชื่อเข้าใช้งานและรหัสผ่านของผู้ใช้งาน ประกอบด้วยเขตข้อมูลดังนี้

เขตข้อมูล (Field)	คำอธิบาย
User_Name	ชื่อเข้าใช้งาน
Password	รหัสผ่าน

ตารางที่ 5-13 เขตข้อมูลและคำอธิบายของตาราง user  
ซึ่งในตารางข้อมูลมีการกำหนดชนิดของข้อมูลในแต่ละเขตข้อมูลและ Primary key ของตารางดังนี้

ฟิลด์	ชนิด	การเรียงลำดับ	แอตทริบิวต์	ว่างเปล่า (null)	คำอธิบาย	เพิ่มเติม	กระทำการ
<input type="checkbox"/> User_Name	varchar(10)	latin1_swedish_ci		ไม่			   
<input type="checkbox"/> Password	varchar(10)	latin1_swedish_ci		ไม่			   

ชื่อคีย์	ชนิด	Cardinality	กระทำการ	ฟิลด์
PRIMARY	PRIMARY	0	 	User_Name

รูปที่ 5-14 ชนิดของข้อมูลและ Primary key ของตาราง user

#### 5.4.1.2.2 การออกแบบบริการต่างๆ

ในส่วนของการบริการต่างๆ ที่เกี่ยวข้องกับข้อมูลส่วนบุคคล มีดังนี้

- บริการขอข้อมูลส่วนบุคคล (getPersonDetail)  
เป็นบริการที่ใช้สำหรับขอข้อมูลของบุคคลที่ต้องการ โดยสามารถใส่ ID หรือชื่อ-นามสกุลของคนๆ นั้นก็ได้ ในการขอข้อมูลของบุคคลที่ต้องการ โดยมีค่าอินพุทและเอาต์พุท ดังนี้

ค่าอินพุทที่ต้องการ

- ID ของบุคคลที่ต้องการจะรู้ข้อมูลส่วนบุคคล
- ชื่อของบุคคลที่ต้องการจะรู้ข้อมูลส่วนบุคคล
- นามสกุลของบุคคลที่ต้องการจะรู้ข้อมูลส่วนบุคคล

ค่าเอาต์พุทที่ออกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ข้อมูลของบุคคลที่ต้องการ อาทิเช่น สถานที่ทำงาน, ที่อยู่, เบอร์โทรศัพท์ เป็นต้น
- บริการขอแผนที่สถานที่ที่เกี่ยวข้องกับบุคคลที่ต้องการ (getPersonMap) เป็นบริการที่ใช้สำหรับขอแผนที่สถานที่ที่เกี่ยวข้องกับบุคคลที่ต้องการได้ อาจจะเป็นบ้านหรือ สถานที่ทำงาน โดยสามารถใส่ ID หรือชื่อ-นามสกุลของบุคคลนั้นๆ ก็ได้ เพื่อใช้ในการขอแผนที่สถานที่ทำงาน โดยมีค่าอินพุตและเอาต์พุต ดังนี้
  - ค่าอินพุตที่ต้องการ
    - ID ของบุคคลที่ต้องการแผนที่สถานที่ทำงาน
    - ชื่อของบุคคลที่ต้องการจะหาสถานที่ทำงาน
    - นามสกุลของบุคคลที่ต้องการจะหาสถานที่ทำงาน
  - ค่าเอาต์พุตที่ออกมา
    - แผนที่ในรูปแบบที่กำหนดไว้
- บริการเพิ่มข้อมูลส่วนบุคคล (setPersonDetail) เป็นบริการที่ใช้สำหรับเพิ่มข้อมูลของส่วนบุคคลของบุคคลใหม่ โดยมีค่าอินพุตและเอาต์พุต ดังนี้
  - ค่าอินพุตที่ต้องการ
    - ชื่อของบุคคลที่ต้องการจะเพิ่มข้อมูล
    - นามสกุลของบุคคลที่ต้องการจะเพิ่มข้อมูล
    - ชื่อสถานที่ทำงานของบุคคลที่ต้องการจะเพิ่มข้อมูล
    - ที่อยู่ของบุคคลที่ต้องการจะเพิ่มข้อมูล
    - เบอร์โทรศัพท์ของบุคคลที่ต้องการจะเพิ่มข้อมูล
    - รายละเอียดปลีกย่อยอื่น
  - ค่าเอาต์พุตที่ออกมา
    - ค่ายืนยัน การเพิ่มข้อมูลของบุคคลที่ต้องการจะเพิ่มข้อมูล
- บริการแก้ไขข้อมูลส่วนบุคคล (editPersonDetail) เป็นบริการที่ใช้สำหรับแก้ไขข้อมูลส่วนบุคคล โดยมีค่าอินพุตและเอาต์พุต ดังนี้
  - ค่าอินพุตที่ต้องการ
    - ชื่อของบุคคลที่ต้องการจะแก้ไขข้อมูล
    - นามสกุลของบุคคลที่ต้องการจะเพิ่มข้อมูล
    - ชื่อสถานที่ทำงานของบุคคลที่ต้องการจะแก้ไขข้อมูล
    - ที่อยู่ของบุคคลที่ต้องการจะแก้ไขข้อมูล
    - เบอร์โทรศัพท์ของบุคคลที่ต้องการจะแก้ไขข้อมูล
    - รายละเอียดปลีกย่อยอื่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าเอาต์พุตที่ออกมา

- ค่ายืนยัน การเพิ่มข้อมูลของบุคคลที่ต้องการจะแก้ไขข้อมูล
- ค้นหาสถานที่ (searchPerson)  
เป็นบริการที่ใช้สำหรับค้นหาบุคคลที่มีชื่อตรงตามที่ใช้ระบุ โดยมีค่าอินพุตและเอาต์พุต ดังนี้

ค่าอินพุตที่ต้องการ

- ชื่อของบุคคลที่ต้องการค้นหา
- นามสกุลของบุคคลที่ต้องการค้นหา

ค่าเอาต์พุตที่ออกมา

- รายชื่อบุคคลต้องมีชื่อสอดคล้องกับอินพุต

#### 5.4.2 การออกแบบในส่วนของเว็บเซิร์ฟเวอร์

การออกแบบส่วนของเว็บเซิร์ฟเวอร์นั้น เป็นการออกแบบในส่วนของเว็บแอปพลิเคชัน ซึ่งทำหน้าที่ในการตัวกลางในการติดต่อระหว่างผู้ใช้ และเว็บเซิร์ฟวิส โดยเมื่อผู้ใช้งานร้องขอใช้หน้าเว็บที่ให้บริการแผนที่ หรือข้อมูลต่างๆ ทางเว็บเซิร์ฟเวอร์ก็จะไปทำการเรียกใช้บริการจากเว็บเซิร์ฟวิส โดยการเรียกใช้บริการจากเว็บเซิร์ฟวิสสามารถทำได้ ดังนี้

1. ต้อง import library ที่ใช้ในการเรียก โดยมี library ดังนี้

```
<%@ page import = "org.apache.axis.client.Call" %>
<%@ page import = "org.apache.axis.client.Service" %>
<%@ page import = "javax.xml.namespace.QName" %>
```

2. ประกาศค่า parameter ต่างๆ และเซตที่อยู่ของเว็บเซิร์ฟวิส ดังนี้

```
String endpoint = "http://localhost/axis/services/MapServices";
Service service = new Service();
Call call = (Call) service.createCall();
```

```
call.setTargetEndpointAddress(new java.net.URL(endpoint));
```

3. เซตบริการที่จะเรียกใช้ และ ทำการเรียกใช้

```
call.setOperationName(new
QName("http://localhost/axis/services/MapServices", "getMap"));
String result = call.invoke(new Object[] {new String("664500"), new
String("1575400"), new String("674500"), new String("1576400"), new
String("SVG")});
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อีกหน้าที่หนึ่งของเว็บเซิร์ฟเวอร์ คือ การนำข้อมูลที่รับจากเว็บเซอร์วิสมาใส่ไว้ในหน้าเว็บเพจ เพื่อนำไปแสดงผลในเครื่องไคลเอนท์ โดยรูปแบบในการส่งแผนที่ของเว็บเซอร์วิสสามารถส่งออกมาได้ 3 แบบ ซึ่งจะแสดงวิธีการแสดงผลในหน้าเว็บเพจ

### 1. ผลลัพธ์ที่ได้เป็น GML

GML เป็นรูปแบบมาตรฐานของแผนที่ ดังนั้นจึงมีรูปแบบการแสดงผลที่ไม่แน่นอน จะทำการแสดงผลเป็น XML ก็ได้ หรือจะแปลงไปเป็น SVG ก่อนแล้วจึงแสดงผลก็ได้

### 2. ผลลัพธ์ที่ได้เป็น SVG

SVG เป็นรูปแบบที่ใช้ในการแสดงผลอยู่แล้ว ดังนั้น เมื่อต้องการจะนำไปแสดงก็ทำได้ โดยการเพิ่ม Tag <embed> </embed> ดังนี้

```
<EMBED>
SRC="path/svgfile.svg" NAME="SVGEmbed"
HEIGHT="90%" WIDTH="90%" TYPE="image/svg+xml"
PLUGINSOURCE="http://www.adobe.com/svg/viewer/install/"
</EMBED>
```

### 3. ผลลัพธ์ที่ได้เป็น JPEG

JPEG เป็นรูปภาพ ดังนั้น จึงเป็นการง่ายที่จะนำไปแสดงผล ซึ่งทำได้ดังนี้

```
<IMG height=100 src="path/jpegfile.jpeg" width=100 >
```

## บทที่ 6

### ผลการทดลอง

#### 6.1 การใช้งาน

ในการใช้งานระบบให้บริการแผนที่นั้นเราจะทำการใช้งานโดยการเรียกหน้าเว็บผ่านทาง URL <http://161.246.5.51/index.jsp> โดยผู้ที่เข้ามาใช้งานจะต้องเปิดดูผ่านเว็บเบราว์เซอร์ Internet Explorer เมื่อเข้าสู่หน้าแรกจะเป็นหน้าที่แสดงการใช้งานบริการต่างๆ ซึ่งในตอนนี้นี้จะมีบริการที่เรียกผ่านเว็บเซอร์วิสอยู่ 2 ชนิดคือ

##### 5. MapServices

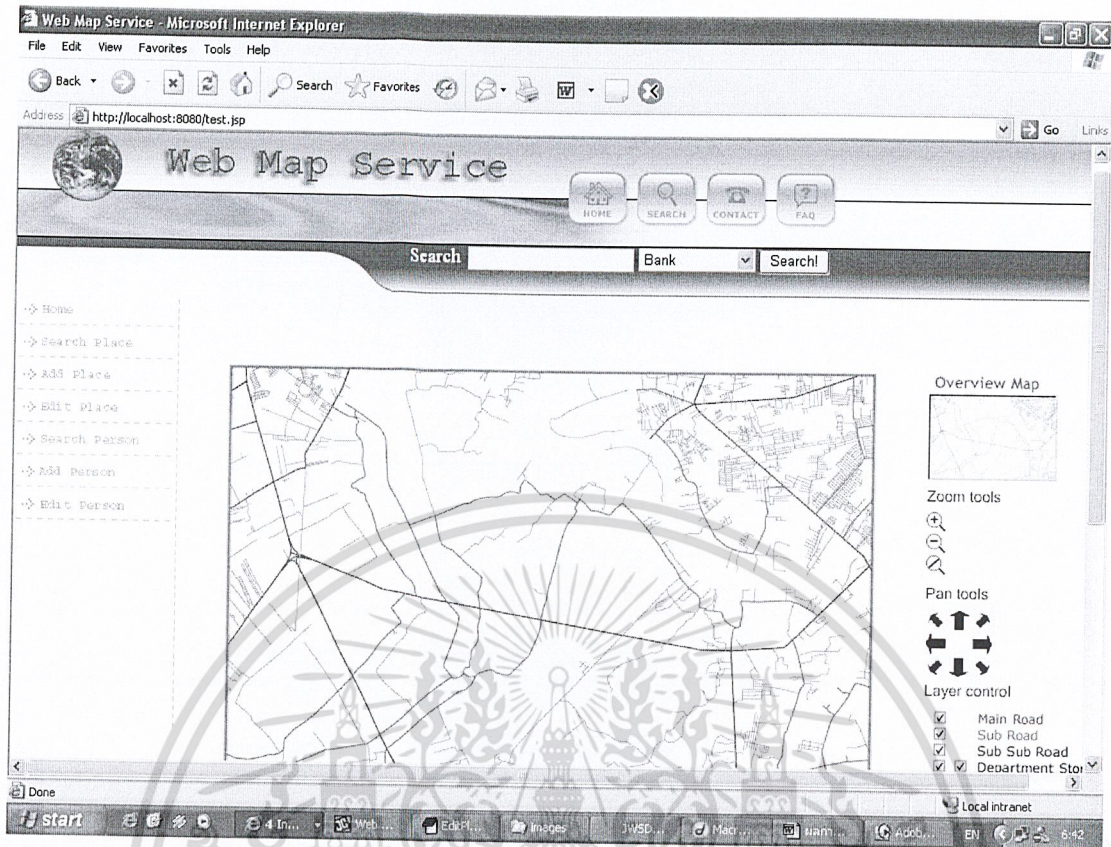
จะให้บริการเกี่ยวกับการค้นหาข้อมูลแผนที่ซึ่งจะแสดงภาพและชื่อของสถานที่ต่างๆ บนแผนที่ รวมทั้งแก้ไขเพิ่มเติมรายละเอียดต่างๆ ได้ด้วย โดยจะมีบริการต่างๆ ที่เกี่ยวข้องดังนี้

- getMap เป็นการเรียกดูข้อมูลแผนที่ตามพิกัดที่เรากำหนด
- getPlaceDetail เป็นการเรียกดูข้อมูลต่างๆ ของสถานที่
- setPlaceDetail เป็นการกำหนดรายละเอียดของสถานที่ต่างๆ เพิ่มเติมลงไป
- editPlaceDetail เป็นการแก้ไขรายละเอียดของสถานที่ต่างๆ ที่เราเพิ่ม
- getMapInfo เป็นการบอกรายละเอียดของข้อมูลต่างๆ ที่ใช้ในการขอแผนที่
- getPlaceDetailInfo เป็นการบอกรายละเอียดของข้อมูลต่างๆ ที่ใช้ในการขอรายละเอียดของสถานที่

##### 5. YellowPageServices

จะให้บริการคล้ายๆ กับสมุดหน้าเหลืองซึ่งจะมีข้อมูลของบุคคลต่างๆ, เบอร์โทรศัพท์ และรายละเอียดอื่นๆ

- getPersonDetail เป็นการเรียกดูข้อมูลของบุคคล
- getPersonMap เป็นการเรียกดูแผนที่ที่เกี่ยวข้องกับบุคคลนั้น
- setPersonDetail เป็นการกำหนดรายละเอียดของบุคคล
- editPersonDetail เป็นการแก้ไขรายละเอียดของบุคคล
- getPersonDetailInfo เป็นการบอกรายละเอียดของข้อมูลที่ใช้บริการเกี่ยวกับข้อมูลส่วนบุคคล



รูปที่ 6-1 เว็บเพจของระบบให้บริการแผนที่

## 6.2 ผลการทดลองใช้งานระบบให้บริการแผนที่บนเว็บ

### 6.2.1 เมื่อเปิดเข้าสู่หน้าแรกของระบบ

เมื่อเราเปิดเข้าสู่หน้าแรกของเว็บเพจจะพบกับหน้าแรกซึ่งจะประกอบด้วยส่วนต่างๆ ดังนี้

#### 1. ส่วนของแผนที่

เป็นส่วนที่แสดงหน้าตาของแผนที่

#### 2. ส่วนควบคุมการแสดงผลแผนที่ (Map Tool)

3. ส่วน Search Bar ที่ใช้สำหรับค้นหาข้อมูลสถานที่ต่างๆ อย่างรวดเร็ว

และสุดท้ายคือเมนูต่างๆ ในหน้าเว็บเพจซึ่งจะมีรายละเอียดของบริการต่างๆ ที่สามารถเลือกได้

โดยเราจะแบ่งออกเป็น 2 กลุ่มคือ Map Service และ Yellow Page Service

### 6.2.2 ทดลองใช้ Map Tool

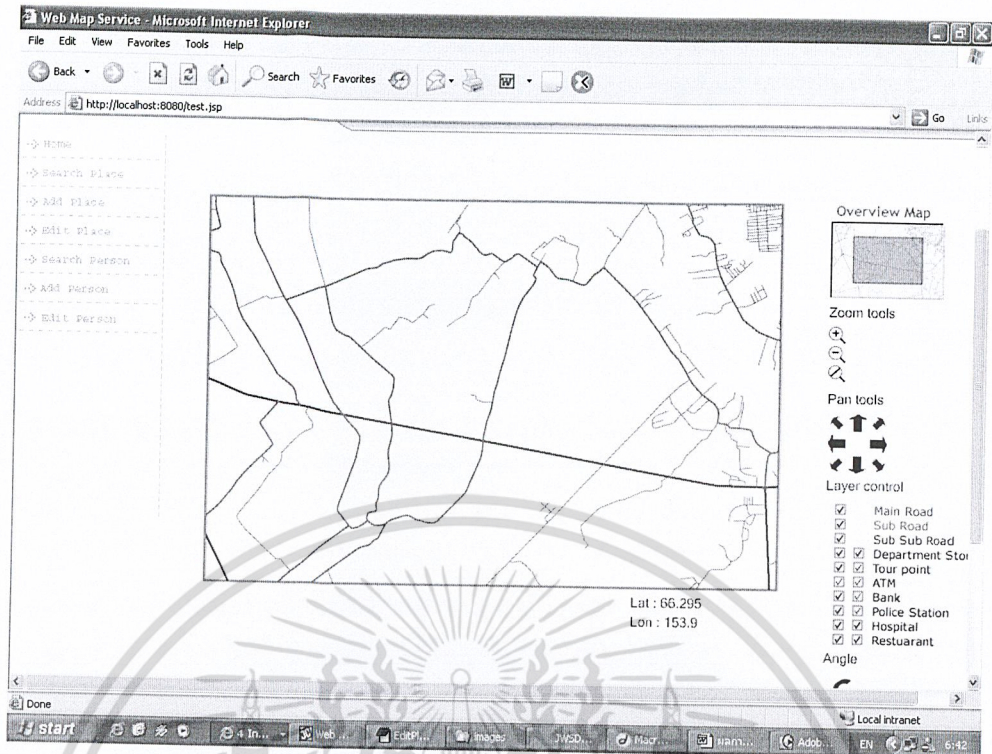
ใน Map Tool จะมีส่วนที่ใช้จัดการกับหน้าแผนที่มากมาย โดยจะประกอบไปด้วยส่วนต่างๆ ดังนี้

#### 1. Overview Map

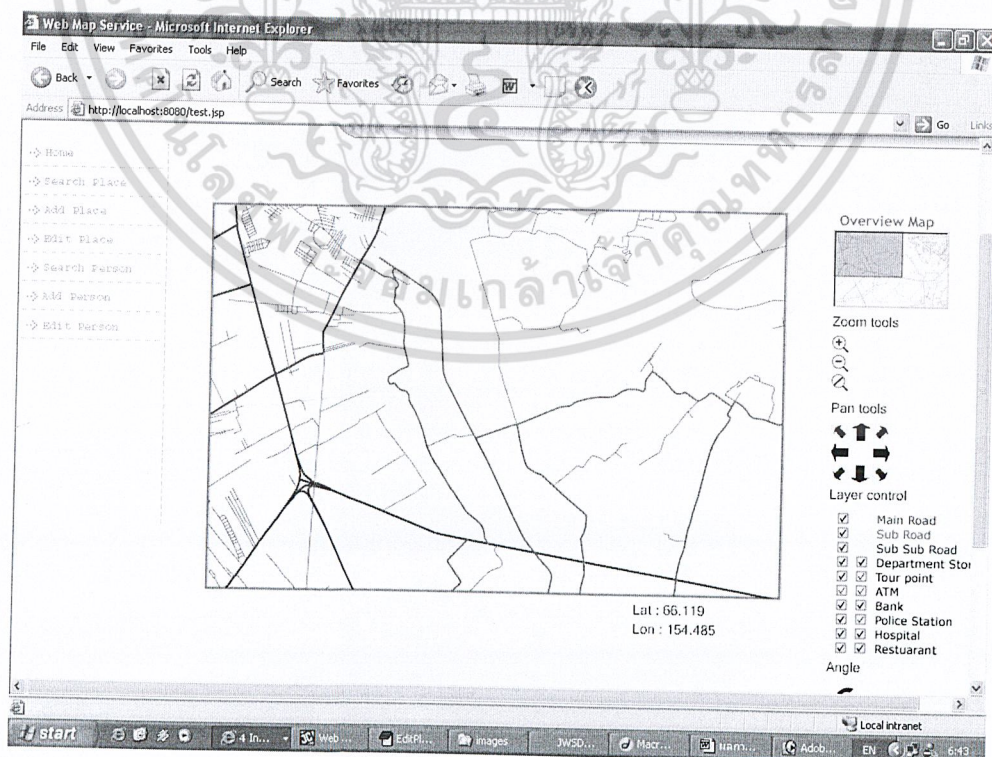
เป็นส่วนที่ใช้ในการเลื่อนหน้าแผนที่ไปยังจุดต่างๆ ที่เราต้องการได้อย่างรวดเร็ว โดยให้

สังเกตรูปที่ 6-2 ซึ่งยังไม่มีการทำอะไรทั้งสิ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



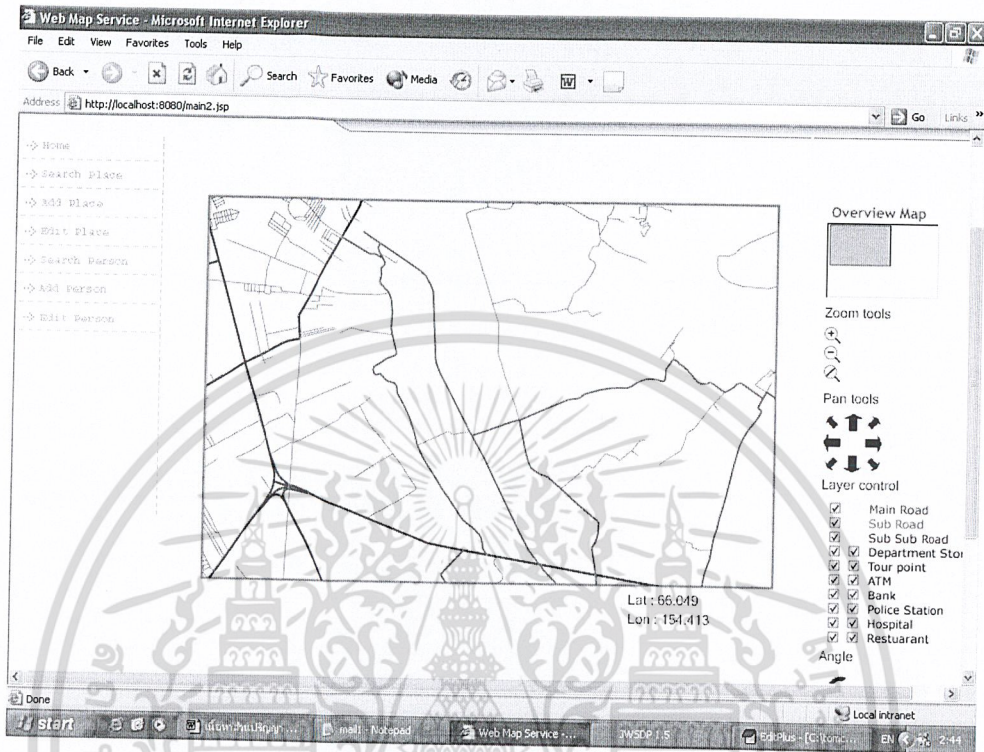
รูปที่ 6-2 ภาพแผนที่ก่อนการเลื่อนแผนที่ใน Overview Map และเมื่อทำการคลิกเมาส์บนสี่เหลี่ยมเล็กๆ ใน Overview Map แล้วลากสี่เหลี่ยมไปยังจุดที่เราต้องการให้แผนที่แสดงภาพ แผนที่ก็จะเปลี่ยนไปตามนั้นทันทีตามรูปที่ 6-3



รูปที่ 6-3 ภาพแผนที่หลังการเลื่อนแผนที่ใน Overview Map เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2. Zoom tools

เมื่อต้องการขยายหรือย่อภาพแผนที่ก็สามารถทำได้ โดยเราจะทำการขยายภาพแผนที่ที่อยู่ในรูป 6-3 โดยคลิกที่รูปแว่นขยายที่มีเครื่องหมาย “+” ก็จะได้ผลลัพธ์ดังรูปที่ 6-4 ดังนี้

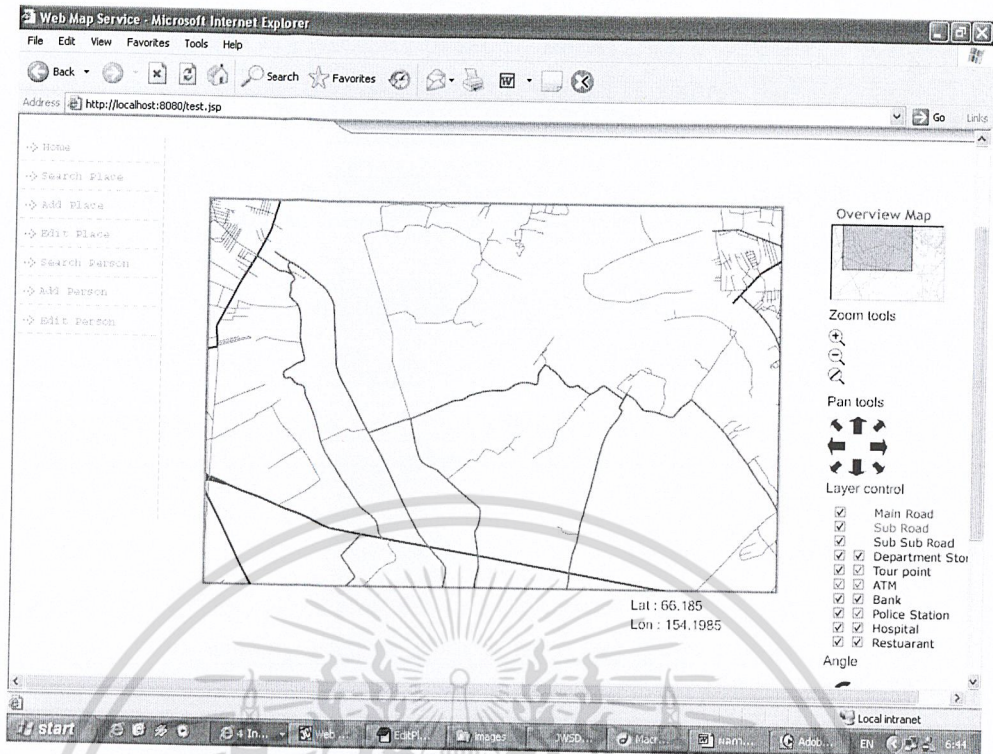


รูปที่ 6-4 ภาพแผนที่หลังทำการขยาย

## 3. Pan tools

เป็นส่วนที่ใช้ในการเลื่อนภาพแผนที่ โดยจะสามารถเลื่อนแผนที่ไปได้ทั้งหมด 8 ทิศทางตามลักษณะการวางของรูปมือ โดยให้สังเกตรูปที่ 6-4 ซึ่งเป็นรูปก่อนการเลื่อนภาพ จากนั้นเราจะทำการเลื่อนภาพไปทางด้านขวา ผลลัพธ์เป็นดังรูปที่ 6-5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



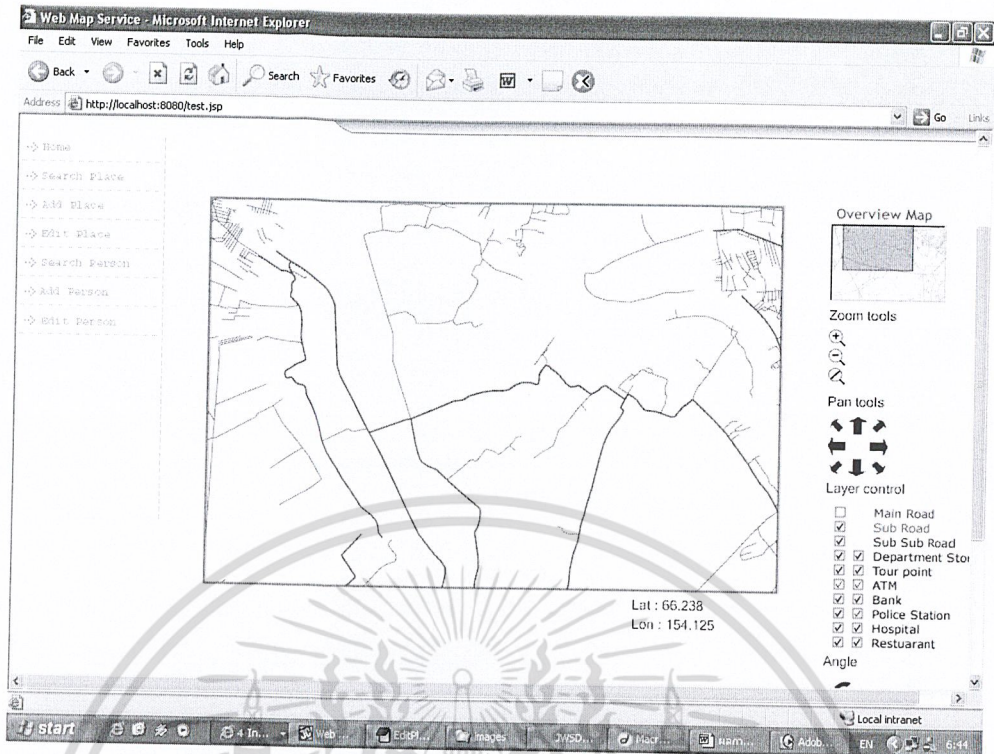
รูปที่ 6-5 ภาพแผนที่หลังการคลิก Pan Tools เลื่อนภาพไปทางขวา

#### 4. Layer control

เป็นส่วนที่ควบคุมการแสดงผลของเลเยอร์ต่างๆ ซึ่งมีดังนี้

- Main Road
- Sub Road
- Sub Sub Road
- Department Store
- Tour Point
- ATM
- Bank
- Police Station
- Hospital
- Restaurant

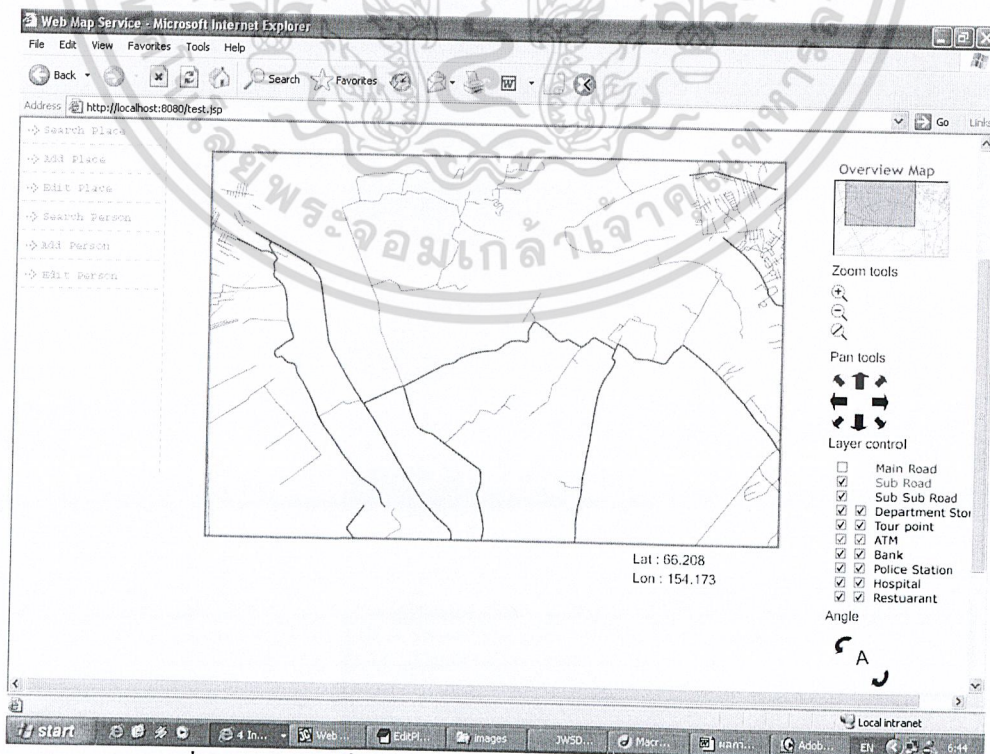
เราจะทำการทดลองโดยการคลิกเมาส์ไปยังช่องสี่เหลี่ยมของ Main Road เพื่อปิดการแสดงผลในส่วนของถนนโดยผลลัพธ์เมื่อหลังจากปิดเลเยอร์ Main Road เป็นดังรูปที่ 6-6



รูปที่ 6-6 ภาพแผนที่ที่ไม่มีแสดงเลเยอร์ของ Main Road

## 5. Angle

สามารถปรับเปลี่ยนมุมของแผนที่ที่เราแสดงได้ เราจะทดลองทำการหมุนภาพแผนที่ที่ทวนเข็มนาฬิกา โดยผลลัพธ์เมื่อเปรียบเทียบกับรูปที่ 6-6 จะเป็นดังรูปที่ 6-7 ดังนี้



รูปที่ 6-7 ภาพแผนที่หลังจากการหมุนแผนที่ทวนเข็มนาฬิกา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

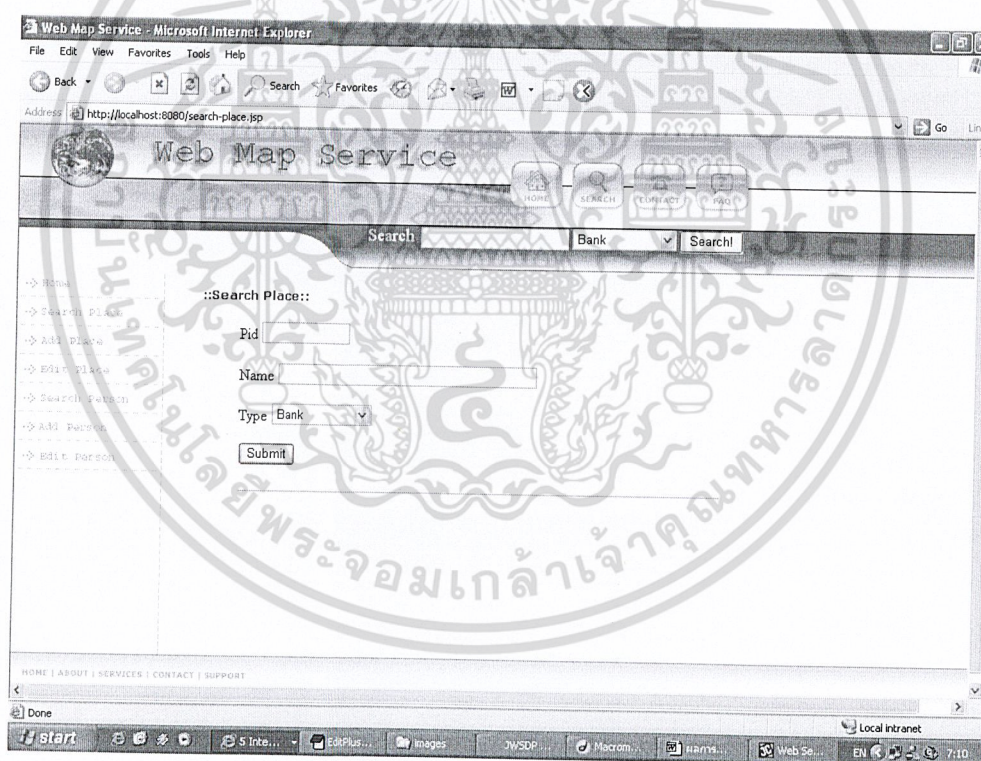
### 6.2.3 การทดลองใช้บริการต่างๆ

ในเว็บเพจของเราจะมีเรียกใช้ความสามารถของการให้บริการของเว็บเซอร์วิสทั้ง 2 แบบได้ โดยจะมีเมนูต่างๆ ที่มีไว้ใช้งานบริการทั้ง 2 ตามแต่ความต้องการของผู้ใช้ดังนี้

- Search Place มีไว้สำหรับค้นหาสถานที่ที่เราต้องการจะค้นหาข้อมูล
- Add Place สำหรับเพิ่มสถานที่ที่ผู้ใช้สนใจลงไปในฐานะข้อมูล
- Edit Place สำหรับแก้ไขข้อมูลสถานที่ต่างๆ ของผู้ใช้
- Search Person สำหรับค้นหารายละเอียดของบุคคลต่างๆ
- Add Person สำหรับเพิ่มบุคคลและรายละเอียดที่ผู้ใช้สนใจลงไปในฐานะข้อมูล
- Edit Person สำหรับแก้ไขข้อมูลบุคคลต่างๆ ของผู้ใช้

#### 1. Search Place

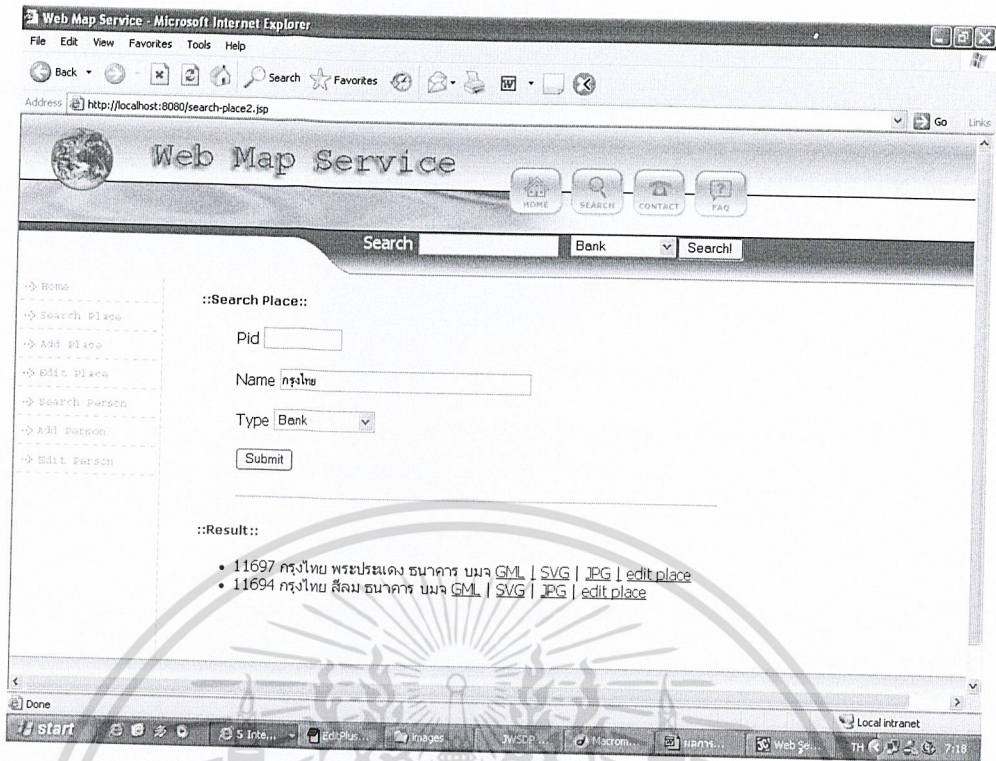
ในการค้นหาสถานที่นั้นให้เราเลือกไปที่เมนู Search Place จากนั้นก็จะเข้าสู่หน้าเว็บเพจตามรูปที่ 6-8 จากนั้นให้เราใส่ข้อมูลที่เรานำสนใจลงไป



รูปที่ 6-8 หน้าเว็บเพจที่แสดงการขอบริการค้นหาสถานที่

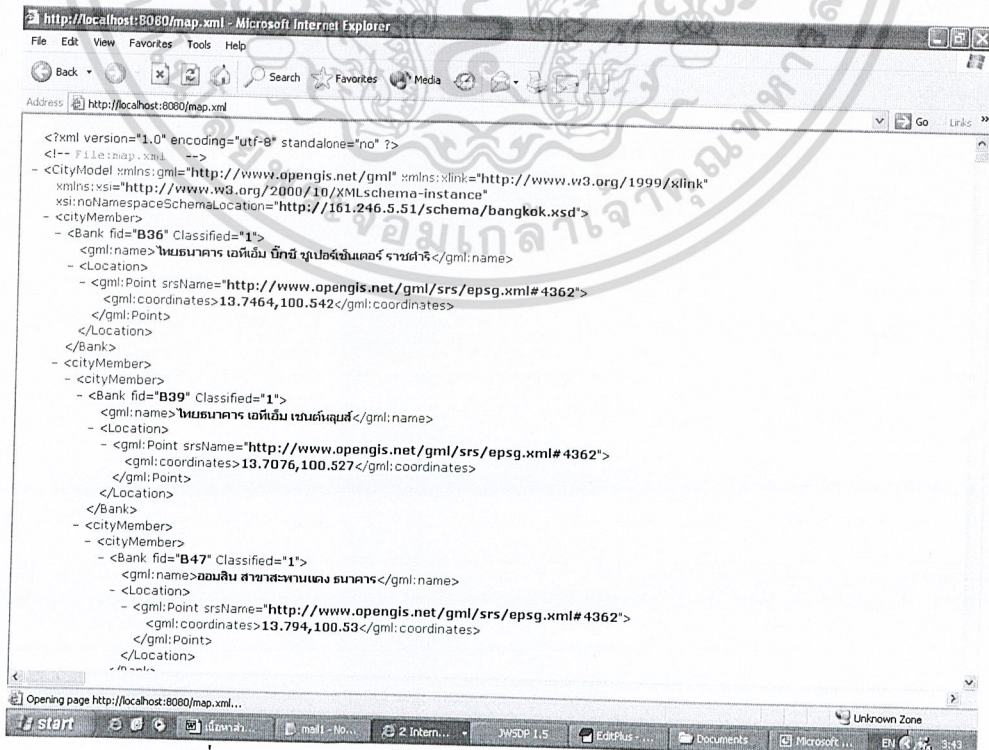
หลังจากกดปุ่ม Submit แล้วข้อมูลที่ได้อาจจะแสดงดังในรูปที่ 6-9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



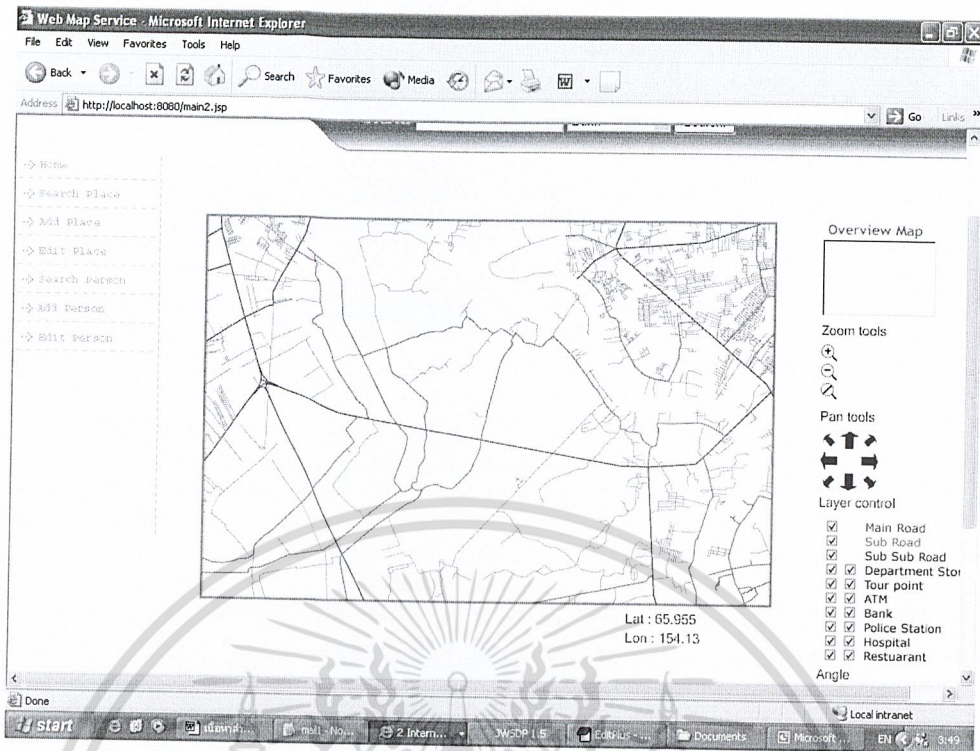
รูปที่ 6-9 หน้าเว็บเพจหลังจากขอบริการค้นหาสถานที่

จะสังเกตเห็นว่าผลลัพธ์ที่ระบบส่งกลับมานั้นจะมีส่วนเชื่อมโยงไปยังหน้าเอกสารอื่นๆ ในรูปแบบต่างๆ ด้วย ซึ่งได้แก่ GML, SVG, JPG ส่วน “edit place” นั้นจะใช้สำหรับการแก้ไขข้อมูลต่างๆ ของผลลัพธ์นั้น ซึ่งจะแสดงตัวอย่างของผลลัพธ์ดังรูปต่อไปนี้



รูปที่ 6-10 ตัวอย่างผลลัพธ์ของเอกสารในรูปแบบ GML

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6-11 ตัวอย่างผลลัพธ์ของเอกสารในรูปแบบ SVG

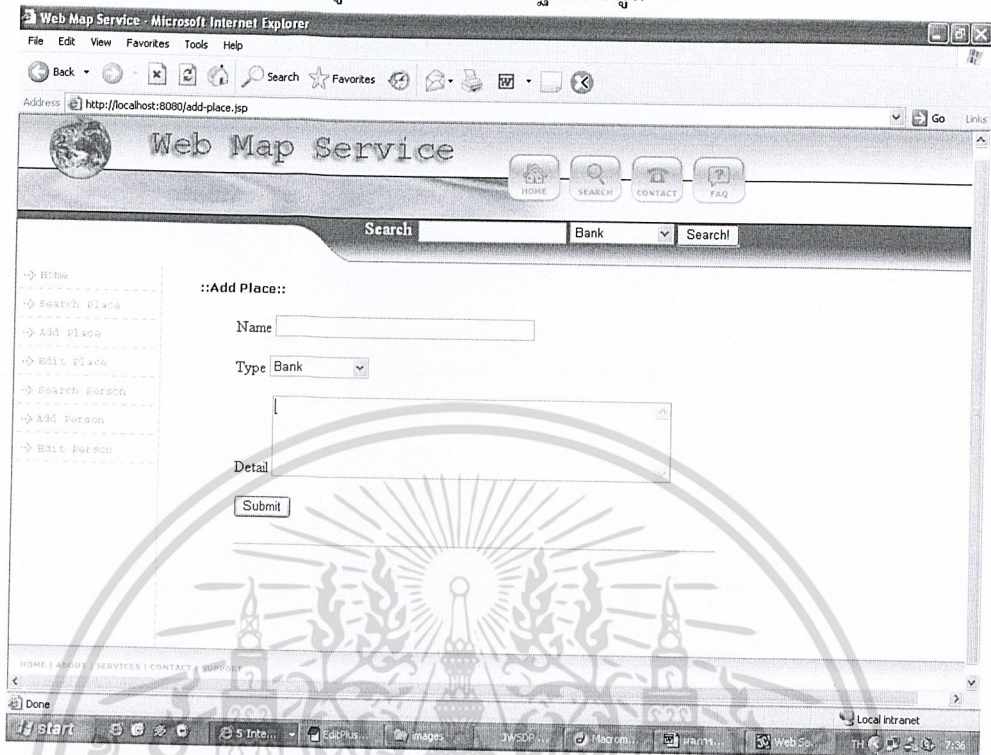


รูปที่ 6-12 ตัวอย่างผลลัพธ์ของเอกสารในรูปแบบ JPG

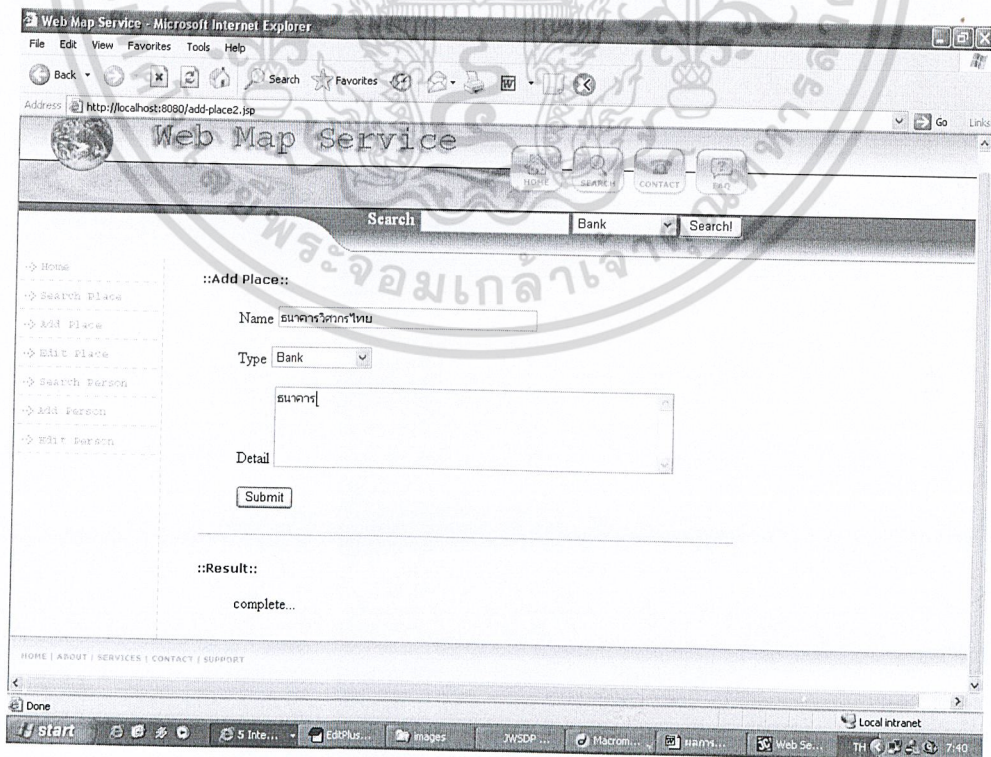
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2. Add Place

จะทำการเพิ่มส่วนของข้อมูลสถานที่ลงไปบนฐานข้อมูล



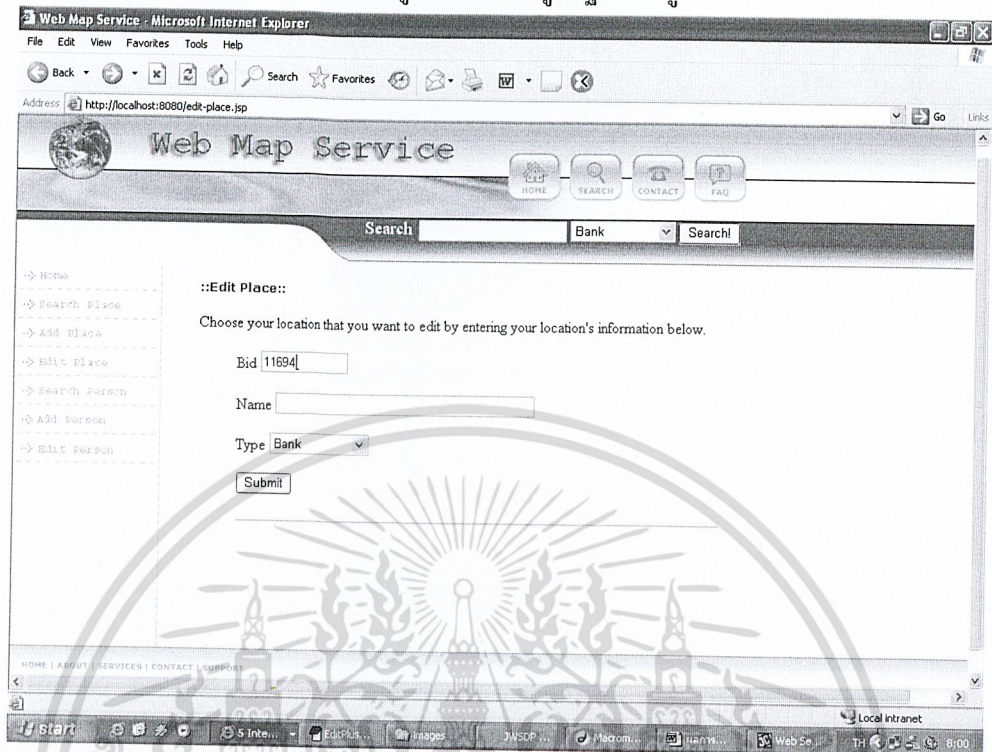
รูปที่ 6-13 หน้าเว็บเพจที่แสดงการขอบริการเพิ่มข้อมูลสถานที่ หลังจากกดปุ่ม Submit แล้ว บราวเซอร์ก็จะแสดงหน้าเว็บเพจตามรูปที่ 6-14



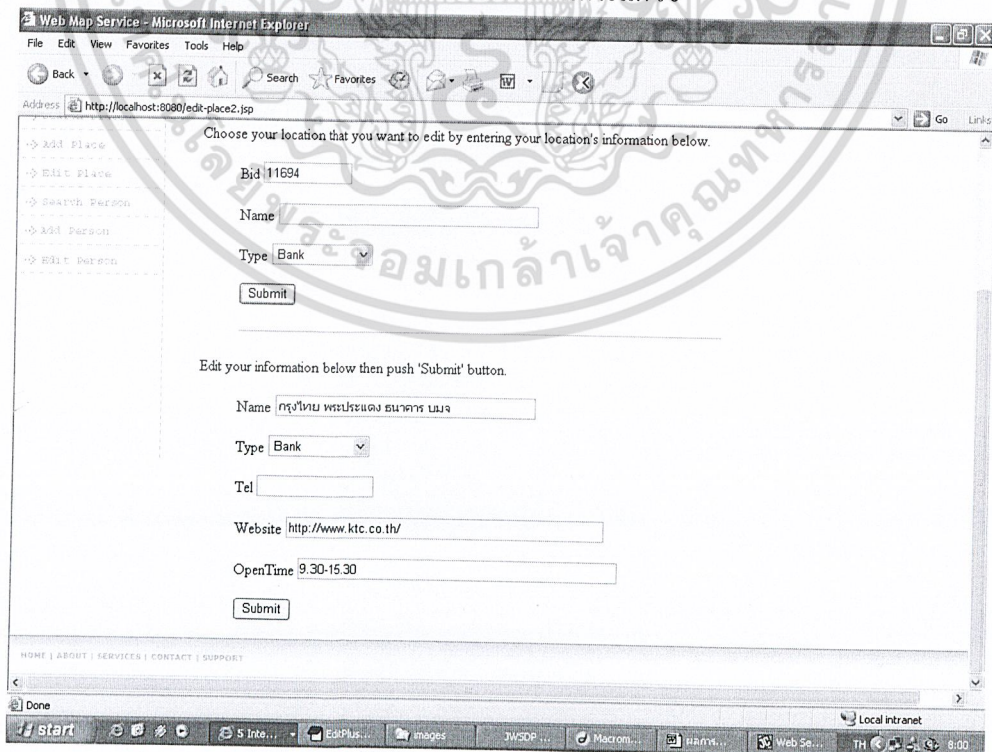
รูปที่ 6-14 หน้าเว็บเพจหลังจากขอบริการเพิ่มข้อมูลสถานที่  
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3. Edit Place

จะทำการแก้ไขส่วนของข้อมูลสถานที่ที่อยู่ในฐานข้อมูล



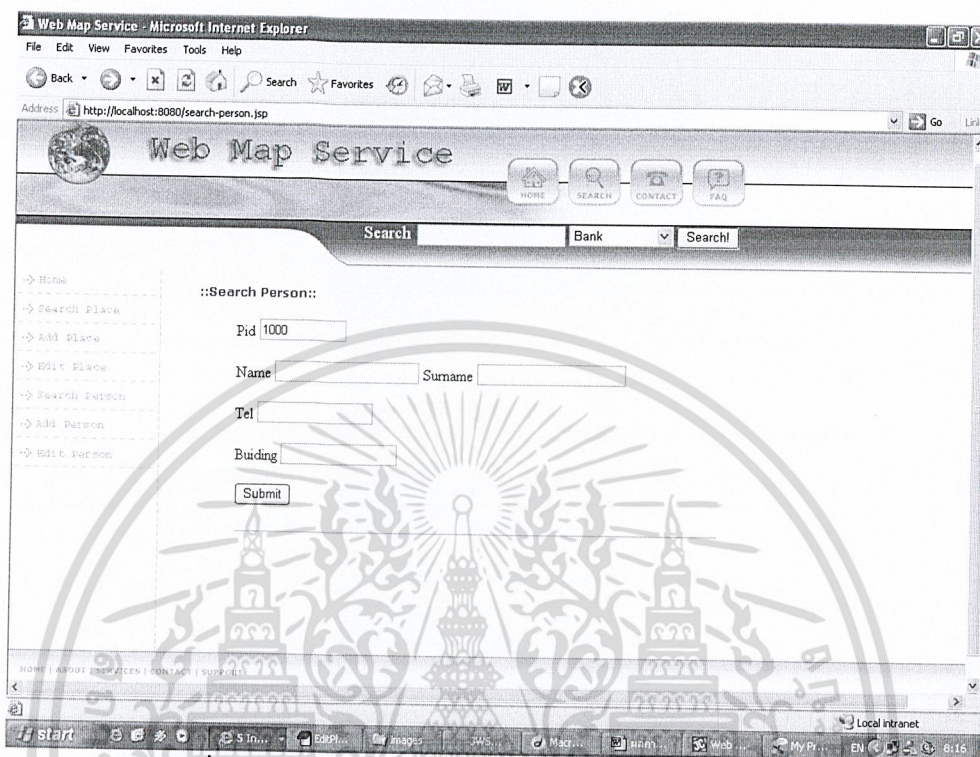
รูปที่ 6-15 หน้าเว็บเพจที่แสดงการขอบริการแก้ไขข้อมูลสถานที่  
จากนั้นก็ Server ก็จะส่งข้อมูลกลับมาและแสดงผลดังในรูปที่ 6-16 ซึ่งผู้ใช้งานจะสามารถ  
เปลี่ยนแปลงแก้ไขได้ก่อนที่จะกด Submit เพื่อยืนยันการแก้ไข



รูปที่ 6-16 หน้าเว็บเพจหลังจากขอบริการแก้ไขสถานที่  
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

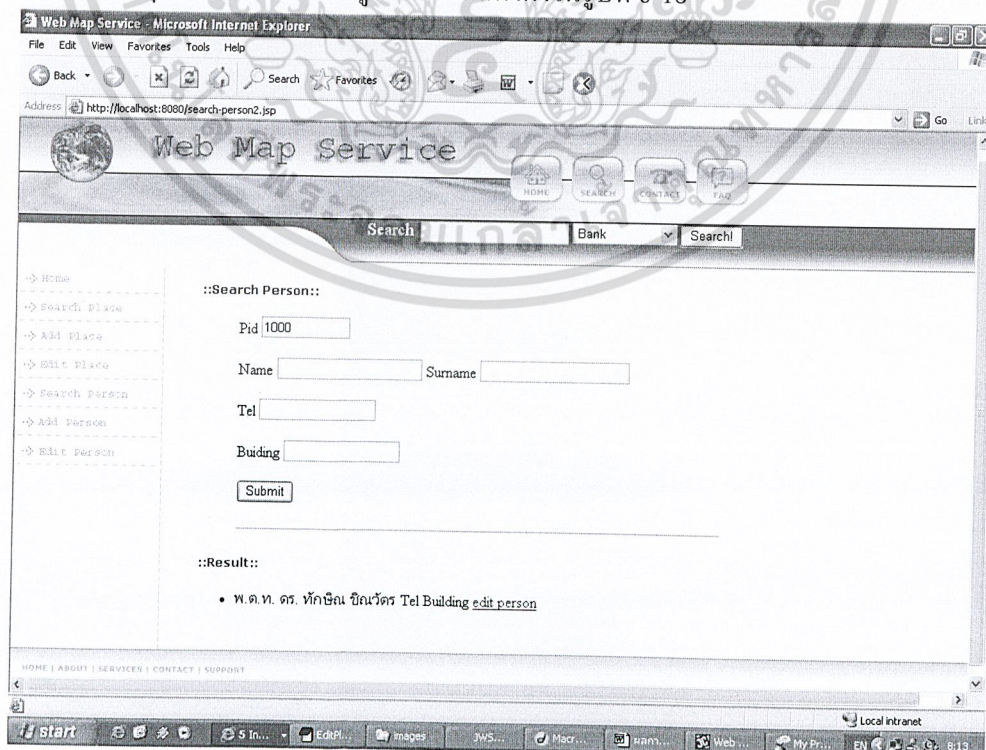
#### 4. Search Person

ในการค้นหาบุคคลนั้นให้เราเลือกไปที่เมนู Search Person จากนั้นก็จะเข้าสู่หน้าเว็บเพจตามรูปที่ 6-17 จากนั้นให้เราใส่ข้อมูลที่เราสงใจลงไป



รูปที่ 6-17 หน้าเว็บเพจที่แสดงการขอบริการค้นหาบุคคล

หลังจากกดปุ่ม Submit แล้วข้อมูลที่ได้อาจจะแสดงดังในรูปที่ 6-18

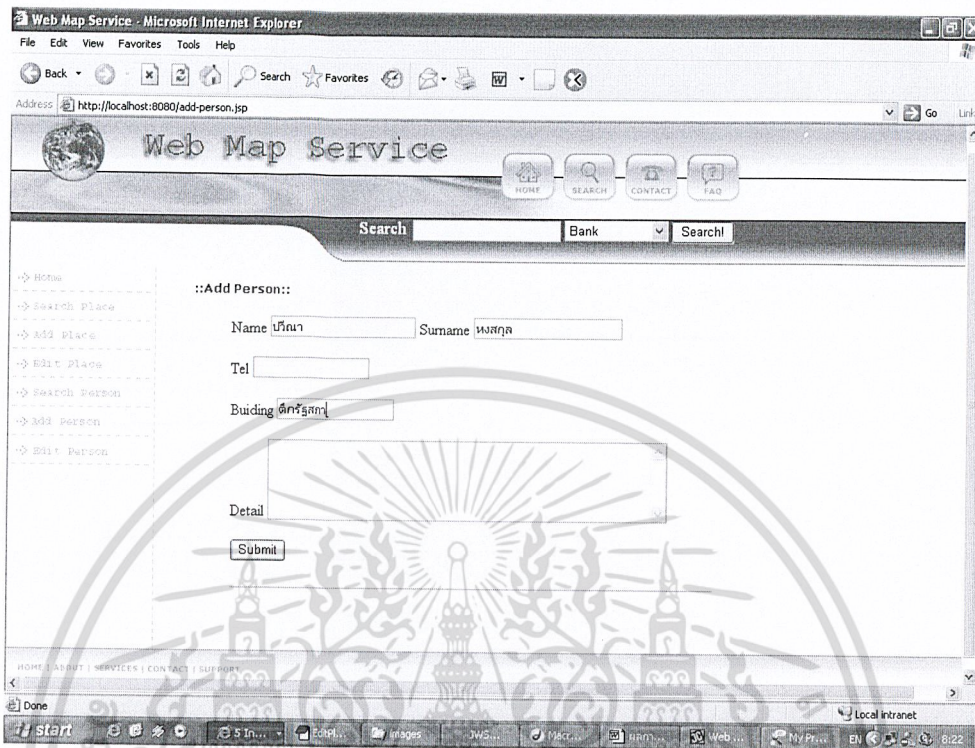


รูปที่ 6-18 หน้าเว็บเพจหลังจากขอบริการค้นหาบุคคล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

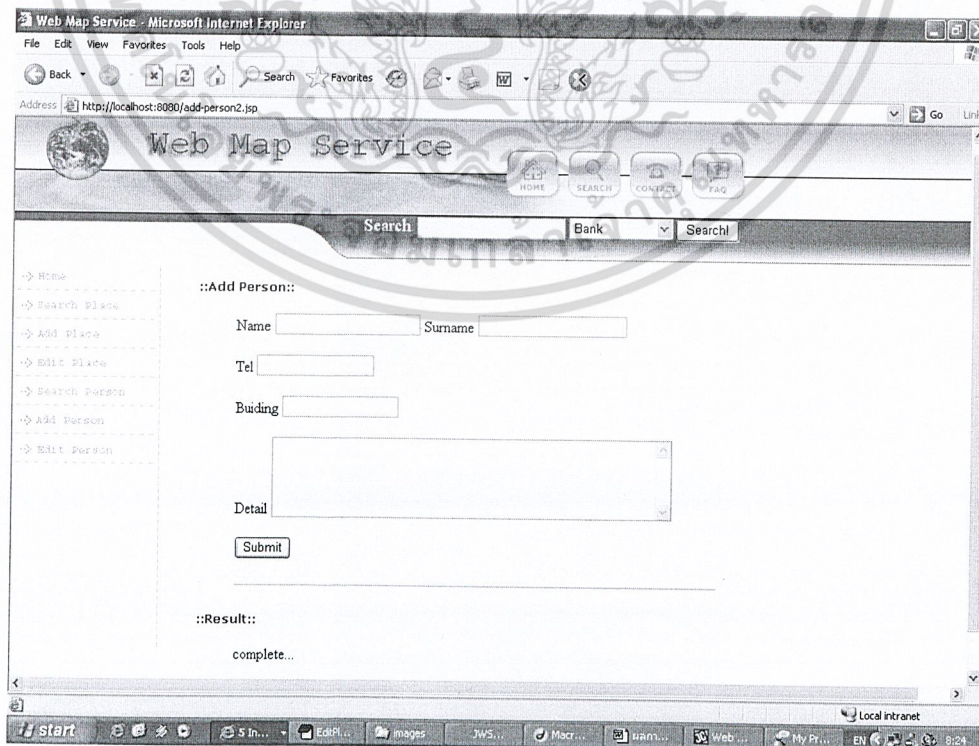
### 5. Add Person

จะทำการเพิ่มส่วนของข้อมูลสถานที่ลงไปในฐานข้อมูล



รูปที่ 6-19 หน้าเว็บเพจที่แสดงการขอบริการเพิ่มข้อมูลบุคคล

หลังจากกดปุ่ม Submit แล้ว บราวเซอร์ก็จะแสดงหน้าเว็บเพจตามรูปที่ 6-20

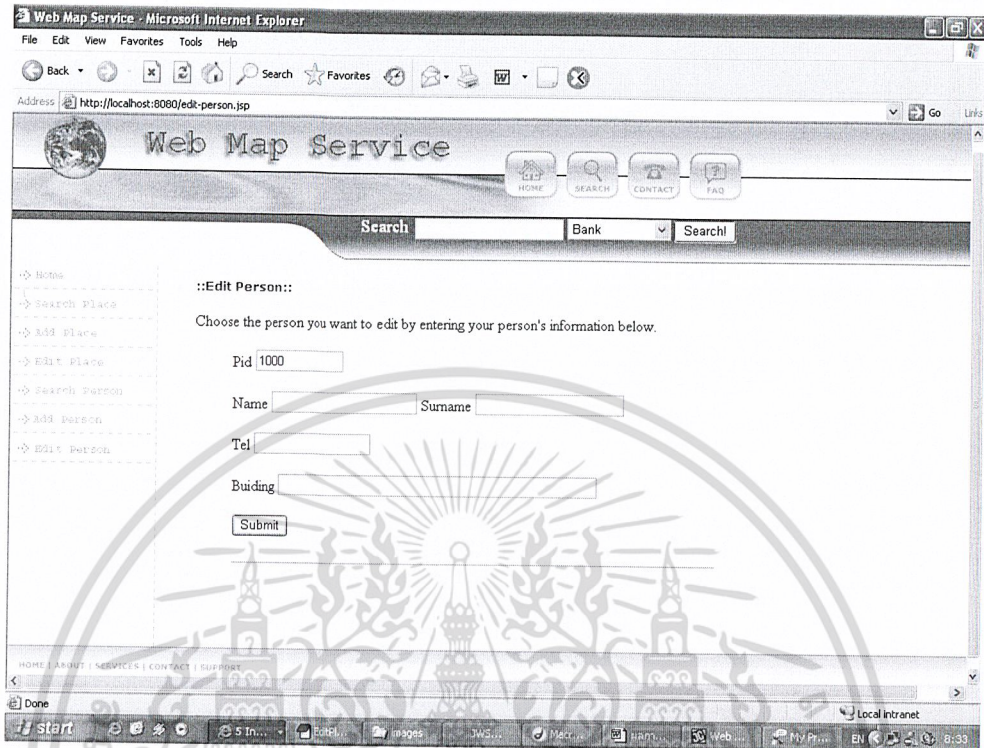


รูปที่ 6-20 หน้าเว็บเพจหลังจากขอบริการเพิ่มข้อมูลบุคคล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

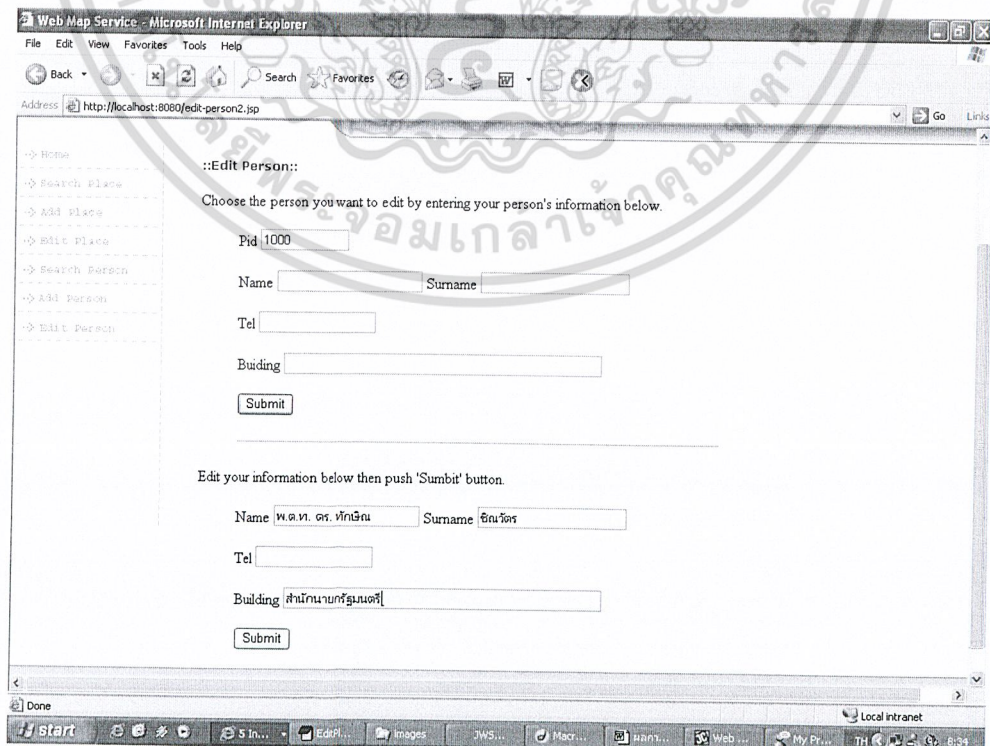
## 6. Edit Person

จะทำการแก้ไขส่วนของข้อมูลบุคคลที่อยู่ในฐานข้อมูล



รูปที่ 6-21 หน้าเว็บเพจที่แสดงการขอรับการแก้ไขข้อมูลบุคคล

จากนั้นก็ Server ก็จะส่งข้อมูลกลับมาและแสดงผลดังในรูปที่ 6-22 ซึ่งผู้ใช้งานจะสามารถเปลี่ยนแปลงแก้ไขได้ก่อนที่จะกด Submit เพื่อยืนยันการแก้ไข



รูปที่ 6-22 หน้าเว็บเพจหลังจากขอรับการแก้ไขข้อมูลบุคคล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 7

### บทวิจารณ์และสรุป

#### 7.1 บทวิจารณ์และสรุป

ระบบระบุตำแหน่ง โดยใช้ภาษาเอ็กซ์เอ็มแอลและเว็บเซอร์วิส ที่พัฒนาขึ้นมาสามารถทำงานบนเว็บเบราว์เซอร์ได้เป็นอย่างดี โดยสามารถติดต่อกับเว็บเซิร์ฟเวอร์เพื่อขอใช้บริการหน้าเว็บที่ให้บริการทั้งแผนที่และข้อมูล และนำมาแสดงผลได้เป็นอย่างดี โดยในส่วนของเว็บเซิร์ฟเวอร์นั้น ก็สามารถติดต่อเรียกใช้บริการของเว็บเซอร์วิสได้เป็นอย่างดี สามารถร้องขอแผนที่ และข้อมูลได้ ไม่ว่าจะเป็นข้อมูลของสถานที่ต่างๆ หรือข้อมูลของบุคคล พร้อมทั้งจัดรูปแบบข้อมูลที่ได้มาให้อยู่ในรูปแบบที่เหมาะสมแก่การแสดงผลได้อย่างดี ในส่วนของเว็บเซอร์วิสนั้น สามารถให้บริการเซอร์วิสต่างๆ ได้เป็นอย่างดี สามารถติดต่อกับฐานข้อมูลเพื่อนำค่าไปส่งให้เว็บเซิร์ฟเวอร์ได้อย่างถูกต้อง และการติดต่อกันระหว่างเว็บเซอร์วิสด้วยกันนั้น ก็สามารถทำได้โดยไม่มีปัญหา

เมื่อเปรียบเทียบกับเว็บไซต์ที่ให้บริการแผนที่ในปัจจุบันแล้ว ระบบที่พัฒนามานี้มีความหลากหลายในการใช้งานมากกว่า โดยสามารถให้บริการได้ ไม่ว่าจะเป็นแอปพลิเคชันที่ใช้แสดงผลจะเขียนด้วยภาษาใดก็ตาม ไม่จำเป็นที่จะต้องเป็นเว็บเซิร์ฟเวอร์ และระบบยังได้เพิ่มความสามารถในการแสดงผลของข้อมูล SVG ด้วยความสามารถของ JavaScript ทำให้การแสดงผลนั้นมีลูกเล่นมากขึ้น ทำให้สามารถปฏิสัมพันธ์กับผู้ใช้งานได้ดียิ่งขึ้น

เทคโนโลยีที่นำมาใช้ในการพัฒนาระบบให้บริการแผนที่ในโครงการนี้ ประกอบด้วย XML, GML, SVG และ Java Server Page ทำให้แผนที่นี้สามารถทำงานได้กับเว็บทุก Platform และเทคโนโลยีจาวาทำให้ระบบนี้สามารถใช้งานได้บนทุกระบบปฏิบัติการ และเทคโนโลยีของ Spatial Database นั้นทำให้ระบบมีการจัดเก็บข้อมูลแผนที่ที่ถูกต้อง สามารถนำมาใช้งานและแสดงผลได้หลากหลายรูปแบบ และเทคโนโลยีของ Web Services ทำให้ระบบสามารถติดต่อกันได้ ไม่ว่าจะทำงานด้วยภาษาไหน หรือบน Platform ใดก็ได้ ดังนั้นจึงเป็นวิธีการที่เหมาะสมที่จะนำไปใช้ในการพัฒนาระบบให้บริการแผนที่ต่อไปในอนาคต

#### 7.2 ปัญหาที่เกิดขึ้นในการพัฒนาและแนวทางการแก้ปัญหา

1. มีปัญหาเรื่องของ time out ในการส่งข้อมูลระหว่างเว็บเซอร์วิสกับเว็บเซิร์ฟเวอร์ หรือระหว่างเว็บเซอร์วิสกับเว็บเซอร์วิส เนื่องจากข้อมูลแผนที่ที่ใช้ในการส่งมีจำนวนมาก จึงทำให้เกิด time out ซึ่งสามารถแก้ไขได้โดยการเซตค่า time out ให้มีค่ามากกว่าค่าเริ่มต้นที่ได้กำหนดไว้
2. ฐานข้อมูลที่ได้รับมาในตอนแรกนั้น ไม่สมบูรณ์ แต่เนื่องจากไม่มีฐานข้อมูลใหม่ จึงทำให้ต้องแสดงผลด้วยฐานข้อมูลนั้น แต่เมื่อได้ฐานข้อมูลใหม่ที่ดีกว่ามานั้น ทำให้ต้องเขียนฟังก์ชันการแสดงผลใหม่
3. ฐานข้อมูลที่ได้มานั้น ไม่สมบูรณ์ ไม่สื่อความหมายว่าเป็นอะไร ทำให้ต้องเอาไปเปรียบเทียบกับฐานข้อมูลอื่น เพื่อให้แน่ใจว่าเป็นถนนจริงๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ฐานข้อมูลที่ได้รับมานั้น ไม่มีการกำหนดสถานที่ต่างๆ เอาไว้ จึงทำให้ต้องไปเอาตำแหน่งสถานที่ต่างๆ จากฐานข้อมูลอื่น

### 7.3 ข้อจำกัดของระบบ

ข้อมูลแผนที่ที่ใช้ในการส่งแผนที่ที่ทำให้แสดงผลได้อย่างเหมาะสมนั้น มีปริมาณมาก ซึ่งในปัจจุบันความเร็วในการติดต่อสื่อสารยังไม่เร็วพอ มีผลทำให้ต้องใช้ระยะเวลาในการติดต่อและแสดงผลนาน จึงทำให้ต้องส่งข้อมูลแผนที่ในแต่ละครั้งในปริมาณที่น้อยลง ทำให้แสดงผลภาพได้ไม่ติดนัก

### 7.4 แนวทางการพัฒนา

1. สามารถเพิ่มข้อมูลต่างๆ ของแผนที่เพื่อให้ครอบคลุมพื้นที่ทั้งหมดของกรุงเทพและเพิ่มรายละเอียดของข้อมูลสถานที่ต่างๆ
2. สามารถเพิ่มเติมในส่วนของฟังก์ชันต่างๆ ของแผนที่ให้มีความสามารถมากขึ้น เช่น แสดงเส้นทางรถประจำทาง
3. เพิ่มความสามารถในด้านการตอบสนองการใช้งาน เช่น เพิ่มภาพเคลื่อนไหวในส่วนต่างๆ ให้มากขึ้น หรือเพิ่มการแสดงผลที่หลากหลายโดยใช้ความสามารถของภาษา SVG
4. สามารถเพิ่มเติมในส่วนของบริษัทที่สามารถช่วยในด้านธุรกิจ เช่น สามารถเลือกร้านอาหารแล้วสามารถดาวน์โหลดคู่มือรถราสามารถใช้ได้ เป็นต้น
5. เพิ่มอุปกรณ์แสดงผล โดยทำให้สามารถใช้งานได้บน โทรศัพท์มือถือ คอมพิวเตอร์พกพา เป็นต้น

ภาคผนวก ก  
การติดตั้ง Apache Axis

ขั้นตอนการติดตั้ง Apache Axis มีดังนี้

**1. ติดตั้ง Axis Application ใน Jakarta Tomcat**

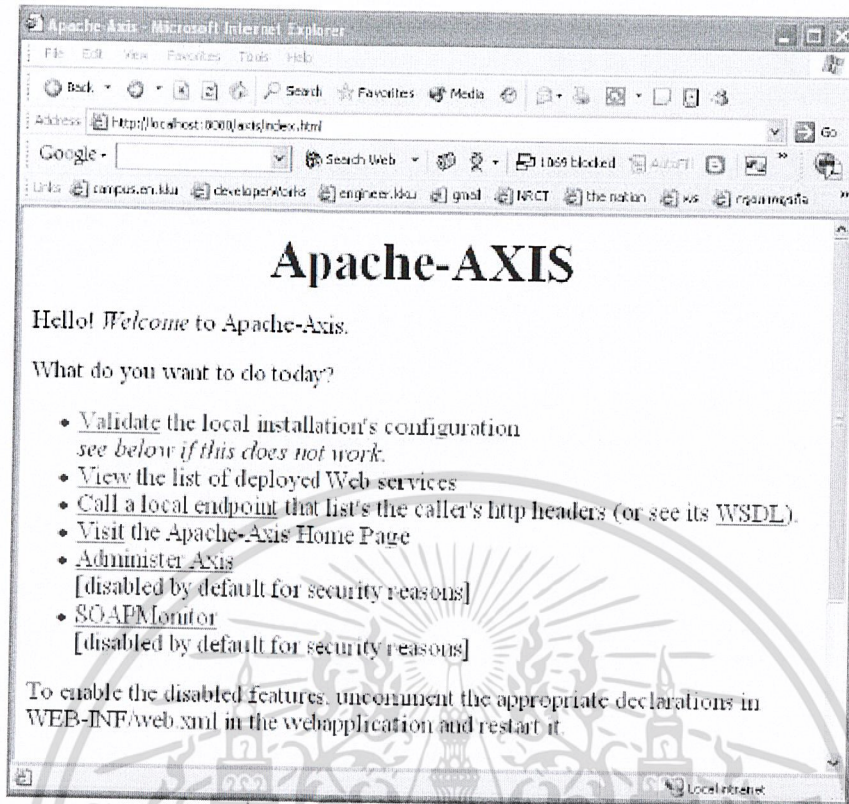
- 1.1 ดาวน์โหลด Apache Axis และ Jakarta Tomcat จาก <http://gear.kku.ac.th/~krunapon/178375/tool.html> จะได้ไฟล์ axis-1\_1.zip และ Jakarta-tomcat-5.0.27.exe
- 1.2 Unzip axis-1\_1.zip
- 1.3 run Jakarta-tomcat-5.0.27.exe สมมุติว่า directory ที่ Apache Axis อยู่คือ %AXIS\_HOME% ซึ่งอาจจะเป็น C:\axis-1\_1 และ directory ที่ Jakarta-tomcat อยู่คือ %TOMCAT\_HOME% ซึ่งอาจจะเป็น C:\tomcat5.0
- 1.4 copy folder “%AXIS\_HOME%\webapps\axis” ไปใส่ใน “%TOMCAT\_HOME%\webapps\”

**2. ติดตั้ง Libraries**

- 2.1 ดาวน์โหลด \*.jar file จาก <http://gear.kku.ac.th/~krunapon/178375/tool.html>
- 2.2 Copy \*.jar files ใส่ไว้ใน %TOMCAT\_HOME%\webapps\axis\WEB-INF\lib

**3. เปิด Jakarta Tomcat Server**

- 3.1 start Tomcat service โดย run file %TOMCAT\_HOME%\bin\startup.bat
- 3.2 เปิด browser เพื่อทดสอบ <http://localhost:8080/axis/index.html> สิ่งที่เราควรจะได้รับปรากฏในรูป



#### 4. ทดสอบการติดตั้ง Apache Axis

4.1 กด Validate เพื่อทดสอบว่าเราลง libraries ครบหรือเปล่า

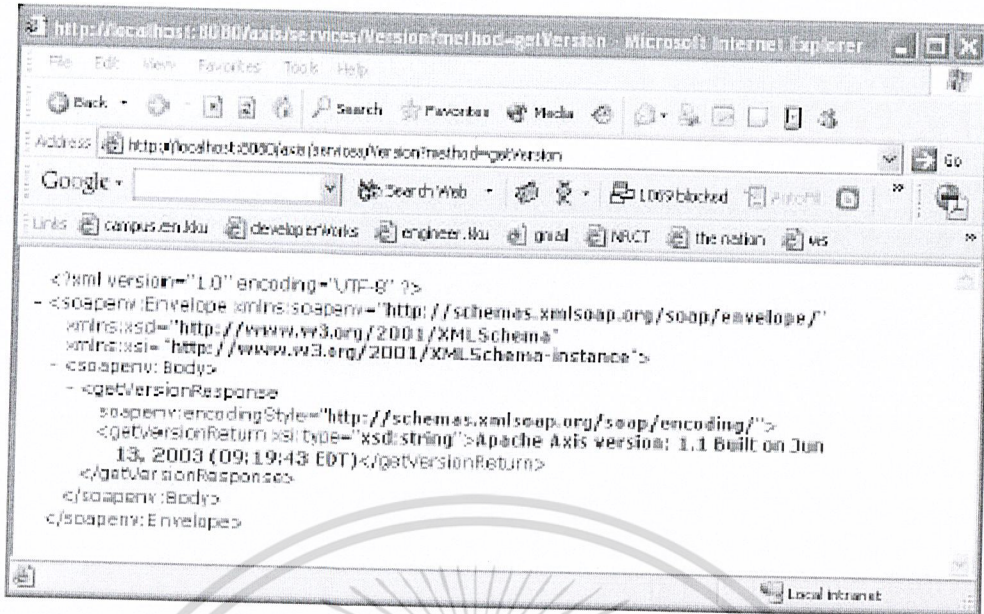
4.2 ถ้า libraries ไม่ครบ จะเกิด error หรือ warning ถ้าหากว่า Apache Axis ต้องการจะได้ไฟล์ activation.jar, mail.jar, และ xmlsec.jar สามารถดาวน์โหลดไฟล์ jar\_happy.zip ได้จาก <http://gear.kku.ax.th/~kruapon/178375/tools.html> และ copy \*.jar file พวกนี้ที่ %TOMCAT\_HOME%\webapps\axis\WEB-INF\lib Stop Tomcat service และ Start Tomcat service เพื่อจะทดสอบว่าตอนนี้ Apache Axis Happy หรือยัง ถ้า Axis Happy จะไม่มี Warning และ Error message

#### 5. ตรวจสอบและเรียกใช้ Service ที่มีอยู่แล้วกดที่ View เพื่อจะรู้ว่า Service อะไรอยู่ในระบบ

5.1 ทดสอบการทำงานของ SOAP Endpoint โดยเรียก URL

<http://localhost:8080/axis/services/Version?method=getVersion> ได้ผลลัพธ์ดังปรากฏในรูป

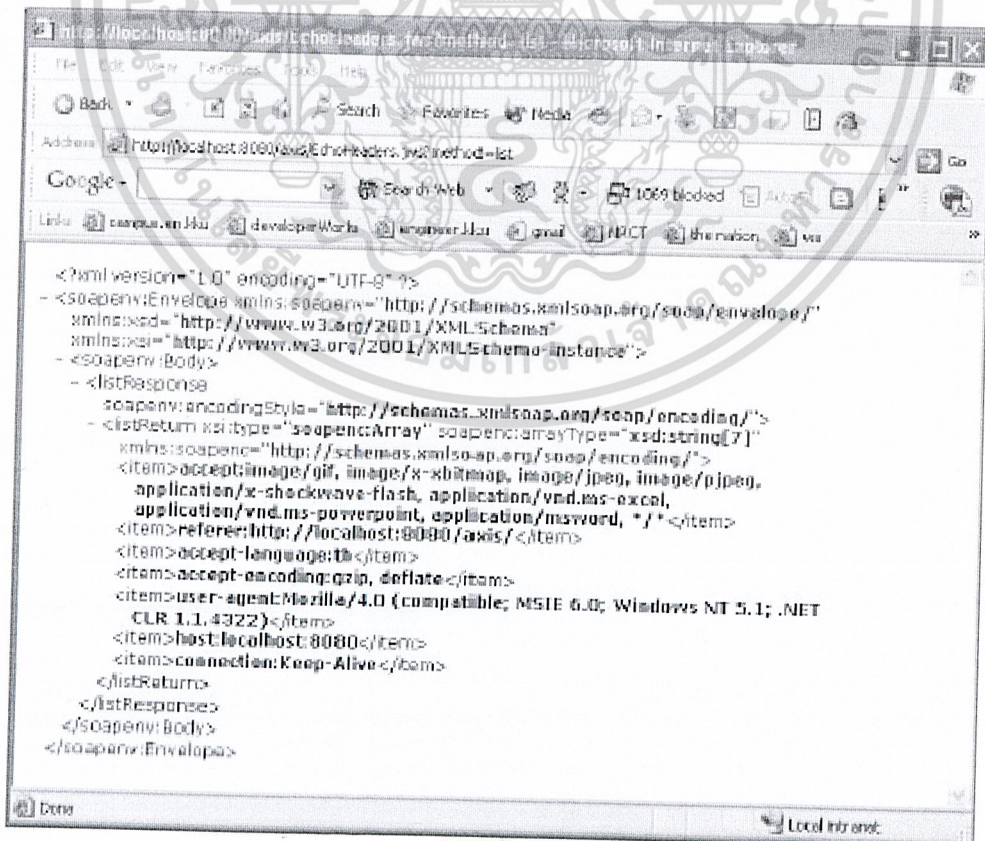
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



### 5.2 ทดสอบ a JWS Endpoint

Axis' JWS Web Services คือ java ไฟล์ที่เก็บไว้ใน the WEB-INF tree ชื่อไฟล์มี .jws extension เมื่อเราเรียกไฟล์โดยการพิมพ์ URL

<http://localhost:8080/axis/EchoHeader.jws?method=list> ไฟล์ก็จะถูก compiled and executed สิ่งที่ได้กลับมาปรากฏในรูป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

## การแปลงข้อมูลแผนที่ให้อยู่ในรูปแบบ SVG

```
public void exportToSVG(double lon_tl, double lat_tl, double lon_rb, double lat_rb) {
    String BBox =
        "Polygon((" +
        Double.toString(lon_tl) + " " + Double.toString(lat_tl) +
        ", " + Double.toString(lon_rb) + " " + Double.toString(lat_tl) +
        ", " + Double.toString(lon_rb) + " " + Double.toString(lat_rb) +
        ", " + Double.toString(lon_tl) + " " + Double.toString(lat_rb) +
        ", " + Double.toString(lon_tl) + " " + Double.toString(lat_tl) +
        "))";

    String width = Double.toString(lon_rb - lon_tl);
    String height = Double.toString(lat_rb - lat_tl);
    String layer[] = new String[20];

    try {
        FileOutputStream f = new FileOutputStream(SVGfile);
        PrintStream ps = new PrintStream(f);

        // print header
        ps.println("<?xml version='1.0' encoding='UTF-8' standalone='no'?'>");
        ps.println("<!DOCTYPE svg PUBLIC '-//W3C//DTD SVG 20000303
Styable//EN' 'http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd'>");
        ps.println();
        ps.println("<svg width='100%' height='100%' " +
            "viewBox='" + lon_tl + " " + lat_tl + " " + width + " " + height +
            "'>");
        ps.println();

        //***** print graphical layer *****
        try {
            int i=0;

            rs = stmt.executeQuery(
                "SELECT DISTINCT(layer) FROM geom ORDER BY 1");
            while(rs.next()) {
                layer[i] = rs.getObject(1).toString();
                i++;
            }
        }
        catch(SQLException e) {}

        // print layer 30,31,32,34
        for (int i=0;i<layer.length;i++) {
            if (layer[i] != null) {
                if (layer[i].equalsIgnoreCase("30") |
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

layer[i].equalsIgnoreCase("31") |
layer[i].equalsIgnoreCase("32") |
layer[i].equalsIgnoreCase("34")) {

if (layer[i].equalsIgnoreCase("30") | layer[i].equalsIgnoreCase("34"))
    ps.println("\t\t\t\t\t<g id=\"Layer1\">");
if (layer[i].equalsIgnoreCase("31"))
    ps.println("\t\t\t\t\t<g id=\"Layer2\">");
if (layer[i].equalsIgnoreCase("32"))
    ps.println("\t\t\t\t\t<g id=\"Layer3\">");

try {
    rs = stmt.executeQuery(
"SELECT ASTEXT(g) FROM geom WHERE layer=" + layer[i] + " " +
"AND MBRContains(GeomFromText(" + BBox + "), g);
        while(rs.next()) {
            String s = rs.getObject(1).toString();
            ps.println("\t\t" + WKTtoSVG(s, layer[i]));
        }
    } catch(SQLException e) {}

    ps.println("\t</g>");
    ps.println();
}
}

//*****

//***** print all place's text *****
try {
    // Bank
    ps.println("\t\t\t\t\t<g id=\"Bank\">");

    ps.println("\t\t\t\t\t<g id=\"BankPoint\">");
    rs = stmt.executeQuery(
"SELECT Place_ID, Lat, Lon FROM bank WHERE " +
"Lon > " + lon_tl + " AND " + "Lon < " + lon_rb + " AND " +
"Lat > " + lat_tl + " AND " + "Lat < " + lat_rb);
    while(rs.next()) {
        String id = "B" + rs.getObject(1).toString();
        String lat = rs.getObject(2).toString();
        String lon = rs.getObject(3).toString();

        ps.println("\t\t\t\t\t\t\t<circle id=\"" + id + "\" cx=\"" + lon + "\"
cy=\"" + lat + "\" r=\"0.0001\" style=\"fill:darkorchid\"/>");
    }
    ps.println("\t\t\t\t\t\t\t</g>");
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ps.println("\t\t\t\t\t<g id=\"BankT\">");
rs = stmt.executeQuery(
    "SELECT Place_ID, Bank_Name, Lat, Lon FROM bank WHERE " +
    "Lon > " + lon_tl + " AND " + "Lon < " + lon_rb + " AND " +
    "Lat > " + lat_tl + " AND " + "Lat < " + lat_rb);
while(rs.next()) {
    String id = "B" + rs.getObject(1).toString();
    String name = new String(rs.getObject(2).toString().getBytes("ISO-
        8859-1"), "TIS-620");
    //String name = new String(temp.getBytes("UTF-8"));
    String lat = rs.getObject(3).toString();
    String lon = rs.getObject(4).toString();

    ps.println("\t\t\t\t\t<text id=\"" + id + "\" x=\"" + lon + "\"
y=\"" + lat + "\" dx=\"0.0002\" dy=\"0.0002\" style=\"font-family:Microsoft Sans
    Serif;font-size:0.0003\">" + name + "</text>");
    }
ps.println("\t\t\t\t\t</g>");

ps.println("\t\t\t\t\t</g>")
}
catch(Exception e) {}

//*****

ps.println("</svg>");
ps.close();
}
catch(FileNotFoundException e) {}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ค.

## การแปลงแผนที่ให้อยู่ในรูปแบบ JPEG

```

public void exportToJPG(double lon_tl, double lat_tl, double lon_rb, double lat_rb,
int width, int height) {
    String BBox =
        "Polygon((" +
        Double.toString(lon_tl) + " " + Double.toString(lat_tl) +
        ", " + Double.toString(lon_rb) + " " + Double.toString(lat_tl) +
        ", " + Double.toString(lon_rb) + " " + Double.toString(lat_rb) +
        ", " + Double.toString(lon_tl) + " " + Double.toString(lat_rb) +
        ", " + Double.toString(lon_tl) + " " + Double.toString(lat_tl) +
        "))";
    String layer[] = new String[20];
    boolean print_text = true;

    BufferedImage bi = new BufferedImage(width, height,
BufferedImage.TYPE_INT_RGB);

    try {
        FileOutputStream f = new FileOutputStream(JPGfile);
        Graphics g = bi.createGraphics();
        //set background's color = white
        g.setColor(Color.white);
        g.fillRect(0, 0, width, height);

        try {
            int i=0;

            rs = stmt.executeQuery(
                "SELECT DISTINCT(layer) FROM geom ORDER BY 1");
            while(rs.next()) {
                layer[i] = rs.getObject(1).toString();
                i++;
            }
        }
        catch(SQLException e) {}

        // draw layer 30,31,32,34
        for (int i=0;i<layer.length;i++) {
            if (layer[i] != null) {
                if (layer[i].equalsIgnoreCase("30") |
                    layer[i].equalsIgnoreCase("31") |
                    layer[i].equalsIgnoreCase("32") |
                    layer[i].equalsIgnoreCase("34")) {

                    if (layer[i].equalsIgnoreCase("30") |
                        layer[i].equalsIgnoreCase("34")) g.setColor(Color.red);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if (layer[i].equalsIgnoreCase("31")) g.setColor(Color.green);
if (layer[i].equalsIgnoreCase("32")) g.setColor(Color.blue);

try {
    // print geographic information
    rs = stmt.executeQuery(
        "SELECT ASTEXT(g) FROM geom WHERE layer=" +
layer[i] + " " + "AND MBRContains(GeomFromText('" + BBox + "'), g)");
    while(rs.next()) {
        String s = rs.getObject(1).toString();
        int x[] = getX(s,lon_tl,lon_rb,width);
        int y[] = getY(s,lat_tl,lat_rb,height);
        int n = getN(x);

        g.drawPolyline(x, y, n);
    }

    double llon = lon_rb - lon_tl;
    double llat = lat_rb - lat_tl;
    System.out.println(llon + " " + llat);

    // check width & height size
    if (llon >= llat & llat > 0.05) print_text = false;
    else if (llat > llon & llon > 0.05) print_text = false;

    if (print_text) {
        // ----- print text -----
        String name, lat, lon;
        int x[] = new int[1];
        int y[] = new int[1];
        double temp = 0.4/(lon_rb - lon_tl);
        int fsize = (int)temp;

        Font font = new Font("Micro Sans Serif", Font.PLAIN,
            fsize);
        g.setFont(font);
        g.setColor(Color.black);

        // print Bank
        rs = stmt.executeQuery(
            "SELECT Bank_Name, Lat, Lon FROM bank WHERE " +
"Lon > " + lon_tl + " AND " + "Lon < " + lon_rb + " AND " + "Lat > " + lat_tl +
" AND " + "Lat < " + lat_rb);
        while (rs.next()) {
            name = "[]" + new
String(rs.getObject(1).toString().getBytes("ISO-8859-1"), "TIS-620");
            lat = "LAT" + rs.getObject(2).toString();
            lon = "LON" + rs.getObject(3).toString();
            x = getX(lon,lon_tl,lon_rb,width);
            y = getY(lat,lat_tl,lat_rb,height);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        g.drawString(name, x[0], y[0]);
    }

    // print Hospital
    rs = stmt.executeQuery(
        "SELECT Hospital_Name, Lat, Lon FROM hospital
WHERE " + "Lon > " + lon_tl + " AND " + "Lon < " + lon_rb + " AND " + "Lat > " +
lat_tl + " AND " + "Lat < " + lat_rb);
    while (rs.next()) {
        name = "[" + new
String(rs.getObject(1).toString().getBytes("ISO-8859-1"), "TIS-620");
        lat = "LAT" + rs.getObject(2).toString();
        lon = "LON" + rs.getObject(3).toString();
        x = getX(lon,lon_tl,lon_rb,width);
        y = getY(lat,lat_tl,lat_rb,height);

        g.drawString(name, x[0], y[0]);
    }

    // print Mall
    rs = stmt.executeQuery(
        "SELECT Mall_Name, Lat, Lon FROM mall WHERE " +
"Lon > " + lon_tl + " AND " + "Lon < " + lon_rb + " AND " + "Lat > " + lat_tl + "
AND " + "Lat < " + lat_rb);
    while (rs.next()) {
        name = "[" + new
String(rs.getObject(1).toString().getBytes("ISO-8859-1"), "TIS-620");
        lat = "LAT" + rs.getObject(2).toString();
        lon = "LON" + rs.getObject(3).toString();
        x = getX(lon,lon_tl,lon_rb,width);
        y = getY(lat,lat_tl,lat_rb,height);

        g.drawString(name, x[0], y[0]);
    }

    // print Police station
    rs = stmt.executeQuery(
        "SELECT Police_Name, Lat, Lon FROM police WHERE "
+"Lon > " + lon_tl + " AND " + "Lon < " + lon_rb + " AND " + "Lat > " + lat_tl + "
AND " + "Lat < " + lat_rb);
    while (rs.next()) {
        name = "[" + new
String(rs.getObject(1).toString().getBytes("ISO-8859-1"), "TIS-620");
        lat = "LAT" + rs.getObject(2).toString();
        lon = "LON" + rs.getObject(3).toString();
        x = getX(lon,lon_tl,lon_rb,width);
        y = getY(lat,lat_tl,lat_rb,height);

        g.drawString(name, x[0], y[0]);
    }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }

    // print Restaurant
    rs = stmt.executeQuery(
        "SELECT Restaurant_Name, Lat, Lon FROM restaurant
WHERE " + "Lon > " + lon_tl + " AND " + "Lon < " + lon_rb + " AND " + "Lat > " +
lat_tl + " AND " + "Lat < " + lat_rb);
    while (rs.next()) {
        name = "[" + new
String(rs.getObject(1).toString().getBytes("ISO-8859-1"), "TIS-620");
        lat = "LAT" + rs.getObject(2).toString();
        lon = "LON" + rs.getObject(3).toString();
        x = getX(lon,lon_tl,lon_rb,width);
        y = getY(lat,lat_tl,lat_rb,height);

        g.drawString(name, x[0], y[0]);
    }

    // print Tourpoint
    rs = stmt.executeQuery(
        "SELECT Tour_Name, Lat, Lon FROM tour WHERE " +
"Lon > " + lon_tl + " AND " + "Lon < " + lon_rb + " AND " + "Lat > " + lat_tl + "
AND " + "Lat < " + lat_rb);
    while (rs.next()) {
        name = "[" + new
String(rs.getObject(1).toString().getBytes("ISO-8859-1"), "TIS-620");
        lat = "LAT" + rs.getObject(2).toString();
        lon = "LON" + rs.getObject(3).toString();
        x = getX(lon,lon_tl,lon_rb,width);
        y = getY(lat,lat_tl,lat_rb,height);

        g.drawString(name, x[0], y[0]);
    }

    // print Add_place
    rs = stmt.executeQuery(
        "SELECT Place_Name, Lat, Lon FROM add_place
WHERE " + "Lon > " + lon_tl + " AND " + "Lon < " + lon_rb + " AND " + "Lat > " +
lat_tl + " AND " + "Lat < " + lat_rb);
    while (rs.next()) {
        name = "[" + new
String(rs.getObject(1).toString().getBytes("ISO-8859-1"), "TIS-620");
        lat = "LAT" + rs.getObject(2).toString();
        lon = "LON" + rs.getObject(3).toString();
        x = getX(lon,lon_tl,lon_rb,width);
        y = getY(lat,lat_tl,lat_rb,height);

        g.drawString(name, x[0], y[0]);
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }
  }
  catch(SQLException e) {}
}
}

g.dispose();
JPEGImageEncoder encoder = JPEGCodec.createJPEGEncoder(f);
JPEGEncodeParam param = encoder.getDefaultJPEGEncodeParam(bi);
param.setQuality((float)1.0, true);
encoder.encode(bi, param);
f.close();
}
catch(IOException e) {}
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม

- [1] David Hunter, Curt Cagle, Dave Gibbons, Nikola Ozu, Jon Pinnock, Paul Spencer; “คัมภีร์การใช้ XML ฉบับสมบูรณ์”, “บริษัท ซีเอ็ดยูเคชั่น จำกัด”, 2002, 262
- [2] ศุภชัย สมพานิช; “เข้าใจและใช้งานภาษา XML ฉบับโปรแกรมเมอร์”, “อินโฟเพรส”, 2001, 61
- [3] Jagoe, Andrew; “Mobile location service: the definitive guide”, “Practice Hall Professionnal Technical Reference”, 2003, 147
- [4] สาธิต ชัยวิวัฒน์ตระกูล; “เก่ง JSP ให้ครบสูตร”, “วิศศึกษา จำกัด”, 2002, 287
- [5] Jason Cranford Teague, Marc Campbell; “SVG for Web Designers”, “Wiley Publishing, Inc.”, 2003
- [6] Shashi Shekhar, Sanjay Chawla; “Spatial Database: A TOUE”, “Prentice Hall”, 2003
- [7] James McGovern, Sameer Tyagi, Michael E. Stevens, Sunil Mathew; “Java Web Services Architecture”, “Morgan Kaufmann Publishers”, 2003
- [8] Simon Cox; “Geography Markup Language (GML)”  
,”<http://www.opengis.net/gml/01-029/GML2.pdf>”, 2001
- [9] Jon Ferraiolo; “Scalable Verctor Graphics (SVG) 1.1 Specification”,”<http://www.w3.org/TR/2003/REC-SVG11-20030114/>”, 2003
- [10] Philippe Le Hegaret; “Document Object Model (DOM)”,”<http://www.w3.org/DOM/>”, 2002
- [11] Arnut; “What’s Web Services – Web services Blogs”,”<http://www.webservice.cmsthailand.com>”, 2004
- [12] <http://www.adobe.com/svg/viewer/install>
- [13] <http://gear.kku.ac.th/~krunapon/178375/>
- [14] <http://www.wsiam.com>