

สำนักหอสมุดกลาง [พระจอมเกล้าลาดกระบัง]

บริการสารสนเทศเชิงภูมิศาสตร์ผ่านเว็บเซอร์วิสบน

ระบบปฏิบัติการวินโดวส์

Windows Based GIS Web Services



นาย รัฐพล สัตยารัฐ
นาย รัฐสิทธิ์ ดินสุธารักษ์

เลขหมู่.....
เลขทะเบียน..... 62334
วัน,เดือน,ปี..... 16 ส.ค. 2549

b..... 11621221
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บริการสารสนเทศเชิงภูมิศาสตร์ผ่านเว็บเซอร์วิสบน
ระบบปฏิบัติการวินโดวส์
Windows Based GIS Web Services



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ.2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ปีการศึกษา 2548

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง บริการสารสนเทศทางภูมิศาสตร์ผ่านเว็บเซอร์วิสบนระบบปฏิบัติการวินโดวส์

Windows based GIS Web Services

ผู้จัดทำ

1. นาย รัฐพล สัตยารัฐ เลขประจำตัว 45010645

2. นาย รัฐสิทธิ์ สิ้นสุธารักษ์ เลขประจำตัว 45010647



ชุตินเมษกุล ศรีนิลทา

(ดร. ชุตินเมษกุล ศรีนิลทา)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บริการสารสนเทศเชิงภูมิศาสตร์ผ่าน Web Service บน ระบบปฏิบัติการ Windows

นายรัฐพล	สัตยารัฐ	45010645
นายรัฐสิทธิ์	สินสุธารักษ์	45010647
ดร.ชุตินเมษฐ์	ศรีนิตทา	อาจารย์ที่ปรึกษา
ปีการศึกษา 2548		

บทคัดย่อ

ปริญญาโทฉบับนี้เป็นงานนำเสนอการพัฒนาแมพเซอร์เวอร์ตามมาตรฐานของ WMS (Web Map Service) ของ OGC (OpenGIS Consortium) ที่ให้บริการผ่านเว็บเซอร์วิสบนระบบปฏิบัติการวินโดวส์ โดยให้บริการเชิงพื้นที่ (Location based services) เป็นหลัก โดยมีบริการต่าง ๆ เช่น บริการเรียกแผนที่ บริการค้นหาสถานที่ตามชื่อสถานที่ บริการค้นหาสถานที่ตามที่อยู่ บริการค้นหาสถานที่ภายในรัศมีจากจุดอ้างอิง บริการค้นหาสถานที่ที่ใกล้กับจุดอ้างอิงมากที่สุด บริการสร้างแผนที่ในรูปแบบ SVG บริการคำนวณระยะขจัดจากจุดอ้างอิงและบริการแปลงที่อยู่เป็นค่าพิกัด เป็นต้น โดยผู้ใช้สามารถเรียกใช้บริการได้จากทุกแพลตฟอร์ม ข้อมูลที่ใช้มี 2 ประเภทคือ ข้อมูลเชิงพื้นที่และข้อมูลเชิงอธิบาย

นอกเหนือจากการพัฒนาแมพเซอร์เวอร์แล้ว ได้มีการพัฒนาโปรแกรมเพื่อเรียกใช้บริการต่างๆที่พัฒนาขึ้น รวมทั้งแสดงถึงการทำงานร่วมกันของแมพเซอร์เวอร์ต่างๆ ที่อยู่บนระบบแพลตฟอร์ม

เทคโนโลยีที่นำมาใช้ในการพัฒนาเว็บเซอร์วิสในโครงการนี้ประกอบด้วย C#, .Net, XML, SVG และ PostGIS เทคโนโลยีเหล่านี้ทำให้การพัฒนาเว็บเซอร์วิสเป็นไปอย่างมีประสิทธิภาพ

Windows Based GIS Web Services

Mr. Rattapon Sattayarat 45010645

Mr. Rathasit Sinsutharak 45010647

Dr. Chutimet Srinilta Advisor

Academic Year 2005

ABSTRACT

This objective of the project is to develop a Windows based map server that conforms to OGC (OpenGIS Consortium)'s WMS (Web Map Service) standard that provide services via Web Services. This project mainly provides location based services (LBS) such as get map service, find by name service, find by address service, find nearby point of interest service, find nearest point of interest service, create map in format of Scalable Vector Graphic (SVG) service, calculate displacement between points of interest service, and convert address to coordinate service etc. Users can access services from all platforms. The map server provides both spatial and non-spatial data.

This project has developed a prototype application that access each method of the Web Services provided by the map server. In addition, collaboration between map servers of different platforms is demonstrated via the application.

The technology adopted in development of Web Services in this project are C#, .Net, XML, SVG and PostGIS. All of these technologies help make Web Services development more efficient.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้อย่างดี ด้วยคำแนะนำ และคำปรึกษาจาก คร.ชุตติเมษภู ศรีนิลทา ซึ่งเป็นอาจารย์ที่ปรึกษา คอยให้ความช่วยเหลือต่าง ๆ จนทำให้ปริญญาานิพนธ์ฉบับนี้สำเร็จลุล่วง ซึ่งต้องขอขอบพระคุณเป็นอย่างสูง

ขอขอบคุณภาควิชาวิศวกรรมคอมพิวเตอร์ที่อำนวยความสะดวกต่าง ๆ ทั้งในเรื่องของสถานที่ และอินเทอร์เน็ต ทำให้สามารถทำการวิจัยและการพัฒนา ได้อย่างต่อเนื่องและรวดเร็ว

ขอขอบคุณพี่ๆ เพื่อนๆ น้องๆ ในห้องปฏิบัติการที่คอยให้คำแนะนำ และให้ความช่วยเหลือต่างๆ ทำให้สามารถทำงานได้อย่างราบรื่น

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้าที่คอยให้การสนับสนุน ให้กำลังใจ และที่สำคัญได้ให้โอกาสในการศึกษาเล่าเรียน ข้าพเจ้าขอกราบขอบพระคุณมา ณ ที่นี้ด้วย



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VIII
สารบัญภาพ.....	IX
บทที่ 1 บทนำ.....	1
1.1 ความสำคัญและที่มาของโครงการ.....	1
1.2 วัตถุประสงค์ของโครงการ.....	1
1.3 ขอบเขตของโครงการ.....	2
1.4 วิธีการดำเนินการ.....	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	2
1.6 ส่วนประกอบของปริิณญาานิพนธ์.....	3
บทที่ 2 ทฤษฎี.....	4
2.1 ระบบสารสนเทศภูมิศาสตร์ (Geographic Information System).....	4
2.1.1 องค์ประกอบของ GIS (Components of GIS).....	4
2.1.1.1 ข้อมูล (Data)	5
2.1.1.2 เครื่องคอมพิวเตอร์และอุปกรณ์ต่างๆ (Hardware).....	7
2.1.1.3 โปรแกรมหรือระบบซอฟต์แวร์ (Software).....	7
2.1.1.4 บุคลากร (People).....	7
2.1.1.5 วิธีการหรือขั้นตอนการทำงาน (Procedures/Methods).....	7
2.1.2 ระบบพิกัดบนแผนที่.....	9
2.1.2.1 ระบบพิกัดภูมิศาสตร์ (Geographic Coordinate System).....	9
2.1.2.2 ระบบพิกัดกริด UTM (Universal Transvers Mercator coordinate System).....	10
2.2 OGC WMS.....	12
2.3 Web Services	13
2.3.1 Web Application และ Web Services.....	13
2.3.2 ประโยชน์ของการทำงานร่วมกันระหว่าง Application กับ Web Services.....	14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
2.3.3 เทคโนโลยีที่ใช้ในการพัฒนา Web Services.....	14
2.3.3.1 XML (The Extensible Markup Language 1.0).....	14
2.3.3.2 SOAP (Simple Object Access Protocol).....	15
2.3.3.3 WSDL (Web Services Description Language).....	15
2.3.3.4 UDDI (Universal Description, Discovery and Integration).....	16
2.4 ความรู้เบื้องต้นของภาษา XML	16
2.4.1 XML	16
2.4.2 XSLT และ XPath	20
2.4.3 DTD.....	21
2.4.4 กระบวนการ parsing	22
2.4.4.1 DOM (Document Object Model).....	22
2.4.4.2 SAX (Simple API for XML).....	22
2.4.5 GML (Geography Markup Language).....	22
2.4.6 SVG (Scalable Vector Graphics).....	22
2.4.6.1 คุณสมบัติของ SVG.....	24
2.4.6.2 ประโยชน์ในการใช้ SVG.....	25
2.5 PostgreSQL/PostGIS	26
2.5.1 PostgreSQL	26
2.5.2 PostGIS	26
2.6 เทคโนโลยีดอทเน็ต (.Net Technology).....	29
2.6.1 .Net Framework.....	29
2.6.2 ASP.Net.....	30
2.6.3 การใช้ Visual Studio.Net สร้างเว็บเซอร์วิส.....	32
บทที่ 3 การออกแบบโครงการ.....	34
3.1 โครงสร้างของโครงการ.....	34
3.1.1 GIS Web Services.....	35
3.1.2 GIS Application.....	35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า	
3.2	ฐานข้อมูล.....	35
3.3	การออกแบบ GIS Web Services.....	37
3.3.1	GetCapabilities.....	37
3.3.2	GetMap.....	38
3.3.3	CreateMap.....	41
3.3.4	Find.....	44
3.3.5	FindAddress.....	45
3.3.6	FindNearby.....	46
3.3.7	FindNearest.....	48
3.3.8	CalculateDistance.....	49
3.3.9	ConvertAddressToCoordinate.....	50
3.4	การออกแบบ GIS Application.....	51
3.4.1	โครงสร้างการทำงานของ GIS Application.....	51
3.4.2	การเรียกใช้บริการต่างๆ จาก GIS Web Services.....	52
3.4.2.1	การเรียกใช้บริการ GetCapabilities.....	53
3.4.2.2	การเรียกใช้บริการรับแผนที่ในรูปแบบ SVG (GetMap).....	53
3.4.2.3	การเรียกใช้บริการสร้างแผนที่ในรูปแบบ SVG (CreateMap).....	55
3.4.2.4	การเรียกใช้บริการค้นหาสถานที่ (Find).....	58
3.4.2.5	การเรียกใช้บริการค้นหาสถานที่โดยระบุเป็นที่อยู่ (FindAddress).....	59
3.4.2.6	การเรียกใช้บริการค้นหาสถานที่ภายในรัศมีรอบจุดอ้างอิง (FindNearby)...	60
3.4.2.7	การเรียกใช้บริการค้นหาสถานที่ที่อยู่ใกล้จุดอ้างอิงมากที่สุด (FindNearest)61	
3.4.2.8	การเรียกใช้บริการคำนวณระยะห่างระหว่าง 2 จุด (CalculateDisplacement)62	
3.4.2.9	การเรียกใช้บริการแปลงที่อยู่เป็นพิกัด (ConvertAddressToCoordinate)....	63
บทที่ 4	ผลการทดลอง.....	64
4.1	ผลการทดลองในส่วนของ GIS Application.....	64
4.1.1	การทดลองการแสดงผลแผนที่ตามเลเยอร์ที่ได้เลือกไว้.....	65
4.1.2	การทดลองการแสดงผลแผนที่ตามเลเยอร์ที่ได้เลือกพร้อมกับสถานที่ใหม่ที่ สร้างขึ้น.....	66

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
4.1.3 การทดลองการค้นหาสถานที่จากชื่อของสถานที่.....	67
4.1.4 การทดลองการค้นหาสถานที่จากที่อยู่ของสถานที่.....	68
4.1.5 การทดลองการแสดงผลแผนที่จากจุดอ้างอิงเป็นรัศมีโคจรรอบ.....	69
4.1.6 การทดลองการค้นหาสถานที่ที่ใกล้ที่สุดจากจุดอ้างอิง.....	70
4.1.7 การทดลองคำนวณระยะทางจากจุดเริ่มต้นถึงจุดสิ้นสุด.....	71
4.1.8 การทดลองวางสัญลักษณ์ต่างๆ ลงในแผนที่.....	72
4.2 ผลการทดลองในส่วนของ GIS Web Services.....	73
บทที่ 5 บทวิจารณ์และสรุป.....	74
5.1 บทสรุป.....	74
5.2 สิ่งที่ได้จากโครงการ.....	74
5.3 ปัญหาและอุปสรรค.....	75
5.4 แนวทางการพัฒนาต่อ.....	75
ภาคผนวก ก.....	76
บรรณานุกรม.....	85

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 ความแตกต่างระหว่าง XML กับ HTML	16
2.2 คำอธิบายและสัญลักษณ์ที่ใช้ของแท็ก circle	24
2.3 คำอธิบายและสัญลักษณ์ที่ใช้ของแท็ก path	24
3.1 อธิบายพารามิเตอร์ของ Getmap Service	38
3.2 อธิบายพารามิเตอร์ของ CreateMap Service.....	41
3.3 อธิบายพารามิเตอร์ของ Find Service.....	45
3.4 อธิบายพารามิเตอร์ของ FindAddress Service.....	46
3.5 อธิบายพารามิเตอร์ของ FindNearby Service.....	47
3.6 อธิบายพารามิเตอร์ของ FindNearest Service.....	49
3.7 อธิบายพารามิเตอร์ของ CalculateDisplacement Service.....	50
3.8 อธิบายพารามิเตอร์ของ ConvertAddressToCoordinate Service.....	51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

รูปที่	หน้า
2.1 องค์ประกอบระบบสารสนเทศภูมิศาสตร์ GIS	4
2.2 ประเภทของข้อมูล.....	5
2.3 ข้อมูลประเภท Vector แบบจุด เส้น และ รูปปิด ตามลำดับ.....	5
2.4 ข้อมูลประเภท Raster แบบจุด เส้น และ รูปปิด เทียบกับ Vector ตามลำดับ.....	6
2.5 แสดงระบบพิกัดภูมิศาสตร์.....	10
2.6 แสดงระบบพิกัดกริด UTM	12
2.7 แสดงโครงสร้างของเอกสาร XML ในมุมมอง Logical Structure.....	17
2.8 แสดงโครงสร้างของเอกสาร XML ในมุมมอง Physical Structure	17
2.9 แสดงตัวอย่างของเอกสาร XML	18
2.10 แสดงโครงสร้างของเอกสาร.....	19
2.11 แสดงส่วนประกอบของ Element	20
2.12 ขั้นตอนการแสดงผลใน SVG	23
3.1 โครงสร้างของโครงการ.....	34
3.2 ตาราง Geometry_Columns ของ KMITL.....	36
3.3 ตาราง Geometry_Columns ของ USA.....	36
3.4 การทำงานของ GetCapability Service.....	37
3.5 การทำงานของ Getmap Service.....	38
3.6 แสดงขั้นตอนการให้บริการ GetMap.....	40
3.7 การทำงานของ CreateMap Service.....	41
3.8 แสดงขั้นตอนการให้บริการ CreateMap.....	44
3.9 การทำงานของ Find Service.....	44
3.10 การทำงานของ FindAddress Service.....	45
3.11 การทำงานของ FindNearby Service.....	46
3.12 แสดงการขยายขอบเขตของพื้นที่ออกเป็นสี่เหลี่ยมจัตุรัสในรัศมีที่ต้องการ.....	47
3.13 แสดงการขยายขอบเขตของพื้นที่ออกเป็นวงกลมในพื้นที่สี่เหลี่ยมจัตุรัสในรัศมีที่ต้องการ.....	48
3.14 การทำงานของ FindNearest Service.....	48

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญญภาพ (ต่อ)

รูปที่	หน้า
3.15 การทำงานของ CalculateDisplacement Service	50
3.16 การทำงานของ ConvertAddressToCoordinate Service	51
3.17 แสดง Interface โดยรวมของแอปพลิเคชัน	52
3.18 แสดงขั้นตอนการเรียกใช้บริการ GetCapabilities.....	53
3.19 ขั้นตอนในการเรียกใช้บริการ GetMap.....	54
3.20 แสดง Interface ของการเรียกใช้บริการ GetMap.....	54
3.21 ขั้นตอนในการเรียกใช้บริการ CreateMap.....	56
3.22 แสดง Interface ของการเรียกใช้บริการ CreateMap.....	56
3.23 ขั้นตอนในการเรียกใช้บริการ Find.....	58
3.24 Interface ของการเรียกใช้บริการ Find.....	58
3.25 แสดงขั้นตอนการเรียกใช้บริการ FindAddress.....	59
3.26 interface การเรียกใช้บริการ FindAddress.....	59
3.27 แสดงขั้นตอนการทำงานการเรียกใช้บริการ FindNearby.....	60
3.28 แสดง interface การเรียกใช้บริการ FindAddress.....	60
3.29 แสดงขั้นตอนการเรียกใช้บริการ FindNearest.....	61
3.30 แสดง interface การเรียกใช้บริการ FindAddress.....	61
3.31 แสดงขั้นตอนการทำงานการเรียกใช้บริการ Calculate Displacement.....	62
3.32 แสดง interface การเรียกใช้บริการ Calculate Displacement.....	62
3.33 แสดงขั้นตอนการเรียกใช้เซอร์วิส ConvertAddressToCoordinate.....	63
4.1 แสดงส่วนต่างๆ ของ Application.....	64
4.2 การแสดงผลแผนที่.....	65
4.3 แสดงการเพิ่มสถานที่ใหม่เข้าไปในแผนที่.....	66
4.4 แสดงการค้นหาสถานที่ตามชื่อที่ระบุ.....	67
4.5 แสดงการค้นหาสถานที่ตามที่อยู่ที่ระบุ.....	68
4.6 แสดงการเลือกสถานที่ที่จะกำหนดเป็นจุดอ้างอิง.....	69
4.7 การแสดงแผนที่จากจุดอ้างอิง โดยกำหนดครีสมิในการแสดงผล.....	69
4.8 แสดงการค้นหาสถานที่ที่ใกล้ที่สุดจากจุดอ้างอิง.....	70
4.9 แสดงผลลัพธ์จากการค้นหา จะมีดาวปรากฏขึ้น.....	70

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญญภาพ (ต่อ)

รูปที่	หน้า
4.10 แสดงการคำนวณระยะทางจากจุดเริ่มต้นถึงจุดสิ้นสุด.....	71
4.11 แสดงการแสดงผลจะมีพินระบุจุดเริ่มต้นและจุดสิ้นสุด มีเส้นสีแดงเชื่อมทั้ง 2 จุด.....	71
4.12 แสดงการวางสัญลักษณ์ต่างๆ ลงในแผนที่.....	72
4.13 ผลลัพธ์จากการวางสัญลักษณ์.....	72
4.14 GIS Web Services.....	73



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความสำคัญและที่มาของโครงการ

ระบบสารสนเทศภูมิศาสตร์ (Geographic Information System: GIS) เป็นหนึ่งในความก้าวหน้าทางด้านเทคโนโลยีสารสนเทศเชิงพื้นที่ หรือภูมิสารสนเทศ (Geo - Informatics) ที่มีศักยภาพสูงในการบริหารจัดการข้อมูลเชิงพื้นที่ในหลากหลายมิติ การวิเคราะห์ และการนำเสนอในรูปแบบของแผนที่ (Maps) รูปภาพ (Graphics) และแบบจำลอง (Models) ในมุมมองและเงื่อนไขต่าง ๆ ได้อย่างเป็นพลวัต (Dynamics) ได้มีการใช้งานระบบสารสนเทศภูมิศาสตร์อย่างแพร่หลาย ตัวอย่างเช่น ทางทหาร การคมนาคม การสำรวจทรัพยากรทางธรรมชาติและธุรกิจ เป็นต้น

ปัจจุบันระบบอินเทอร์เน็ตได้เข้ามามีบทบาทในการสื่อสารรับส่งข้อมูลมากขึ้น และการพัฒนาแอปพลิเคชันมีแนวโน้มไปในทิศทางเชิงวัตถุแบบกระจาย มีแนวโน้มไปในทิศทางโปรแกรมเชิงวัตถุแบบกระจาย (Distributed object) มากขึ้น เทคโนโลยีที่น่าสนใจและมีความเหมาะสมในการให้บริการ GIS คือ เว็บเซอร์วิส

โครงการนี้จึงได้นำเสนอทฤษฎี โครงสร้าง และการทำงานพื้นฐานของเว็บเซอร์วิส ที่ให้บริการด้าน GIS

1.2 วัตถุประสงค์ของโครงการ

1. เพื่อศึกษาความรู้พื้นฐานเกี่ยวกับระบบสารสนเทศภูมิศาสตร์
2. เพื่อศึกษาคุณลักษณะของ map server ตามมาตรฐาน WMS (Web Map Service) ของ OGC (OpenGIS Consortium)
3. เพื่อศึกษาหลักการทำงานของเว็บเซอร์วิส ที่ให้บริการเชิงพื้นที่ (Location Based Services : LBS) เป็นหลัก
4. เพื่อศึกษาระบบฐานข้อมูลเชิงพื้นที่ (Spatial Database)

1.3 ขอบเขตของการโครงการ

โครงการนี้จัดทำขึ้นเพื่อพัฒนา map server ตามมาตรฐานของ OGC ซึ่งให้บริการผ่านเว็บเซอร์วิสบนระบบปฏิบัติการวินโดวส์และพัฒนาแอปพลิเคชันที่เรียกใช้เว็บเซอร์วิสนั้น ๆ ใช้ Microsoft Visual Studio .Net เป็นเครื่องมือในการพัฒนา บริการต่าง ๆ เป็นบริการพื้นฐานของ WMS และบริการเชิงพื้นที่ (LBS) โดยข้อมูลจะถูกแบ่งออกเป็นเลเยอร์ต่าง ๆ ซึ่งเก็บอยู่ในฐานข้อมูล PostgreSQL 8.0 with GIS extension

1.4 วิธีการดำเนินงาน

1. ศึกษาเกี่ยวกับระบบสารสนเทศภูมิศาสตร์และ map server
2. ศึกษาเกี่ยวกับเว็บเซอร์วิส
3. ศึกษาเกี่ยวกับ XML
4. ศึกษาเกี่ยวกับ PostgreSQL และ PostGIS
5. ออกแบบและพัฒนาเว็บเซอร์วิสที่ให้บริการสารสนเทศเชิงภูมิศาสตร์(GIS)
6. ออกแบบและพัฒนาแอปพลิเคชันเพื่อเรียกใช้เว็บเซอร์วิสที่สร้างขึ้น

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้รับความรู้เกี่ยวกับระบบสารสนเทศเชิงภูมิศาสตร์
2. ได้รับความรู้เกี่ยวกับเทคโนโลยีของเว็บเซอร์วิส
3. ได้รับความรู้เกี่ยวกับภาษา XML , GML , SVG
4. ได้รับความรู้เกี่ยวกับการพัฒนาเว็บเซอร์วิสด้วยเทคโนโลยี .Net
5. สามารถให้บริการระบบสารสนเทศเชิงภูมิศาสตร์ผ่านเว็บเซอร์วิสกับ client ที่อยู่บนหลาย platform ทั้ง client ที่เป็น enduser และ client ที่เป็นเว็บเซอร์วิสด้วยกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.6 ส่วนประกอบของปริญญานิพนธ์

ปริญญานิพนธ์ฉบับนี้ได้แบ่งเนื้อหาออกเป็น 5 บทด้วยกันคือ

บทที่ 1 กล่าวถึงความสำคัญและที่มาของโครงการ วัตถุประสงค์ของโครงการ ขอบเขตของโครงการ วิธีการดำเนินการ ประโยชน์ที่คาดว่าจะได้รับ และส่วนประกอบของปริญญานิพนธ์

บทที่ 2 กล่าวถึงทฤษฎีพื้นฐานที่ใช้ในโครงการ ซึ่งประกอบด้วย ระบบสารสนเทศ ภูมิศาสตร์ Web Services ความรู้เบื้องต้นของภาษา XML PostgreSQL/PostGIS และเทคโนโลยีคอทเน็ต (.Net Technology)

บทที่ 3 กล่าวถึงการออกแบบโครงการ ประกอบด้วย การออกแบบโครงสร้างของโครงการ การออกแบบฐานข้อมูล การออกแบบ GIS Web Services และ การออกแบบ GIS Application

บทที่ 4 กล่าวถึงการทดลองและผลการทดลอง ทั้งในส่วนของ GIS Web Services และ GIS Application

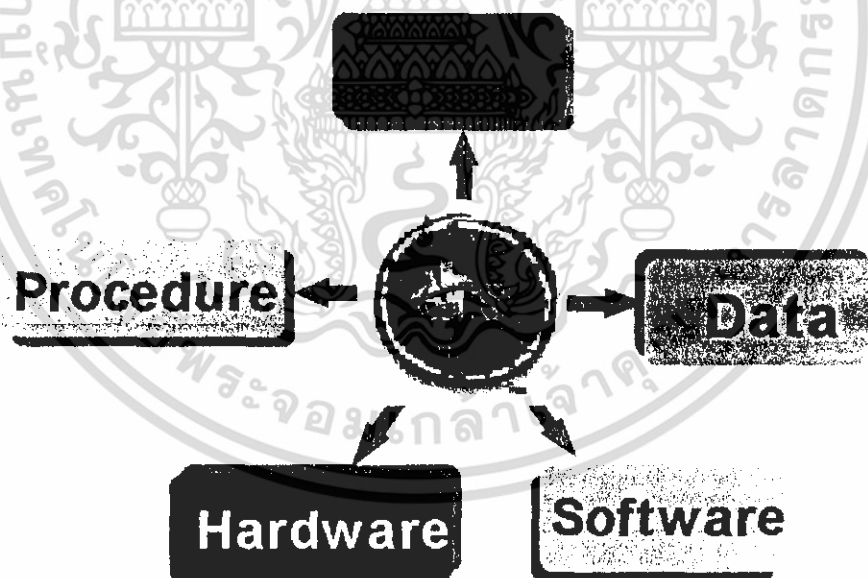
บทที่ 5 เป็นบทวิจารณ์และสรุป ซึ่งกล่าวถึงบทสรุปของโครงการ วิจารณ์สิ่งที่ได้รับจากโครงการ ปัญหาและอุปสรรค และแนวทางในการพัฒนาต่อ

บทที่ 2 ทฤษฎี

2.1 ระบบสารสนเทศภูมิศาสตร์ (Geographic Information System : GIS)

ระบบสารสนเทศภูมิศาสตร์ หมายถึง กระบวนการทำงานเกี่ยวกับข้อมูลในเชิงพื้นที่ด้วยระบบคอมพิวเตอร์ คือซอฟต์แวร์ทางด้านกราฟฟิกที่มีความสามารถในการเก็บข้อมูลด้านแผนที่หรือข้อมูลในลักษณะที่เป็นภาพต่างๆ เช่น ภาพดาวเทียม (Satellite images) ภาพถ่ายทางอากาศ (Aerial photographs) เป็นต้น ซึ่งซอฟต์แวร์ดังกล่าว นี้สามารถนำ เข้าข้อมูลแผนที่หรือข้อมูลภาพต่างๆของพื้นที่ใดพื้นที่หนึ่งซึ่งข้อมูลแต่ละด้านจะถูกจัดเก็บไว้ใน โปรแกรมในลักษณะของข้อมูลเฉพาะเรื่อง (Layer) แล้วสามารถนำเอาข้อมูลเหล่านี้มาวิเคราะห์ประมวลผลร่วมกัน เพื่อหาคำตอบเกี่ยวกับข้อมูลในพื้นที่

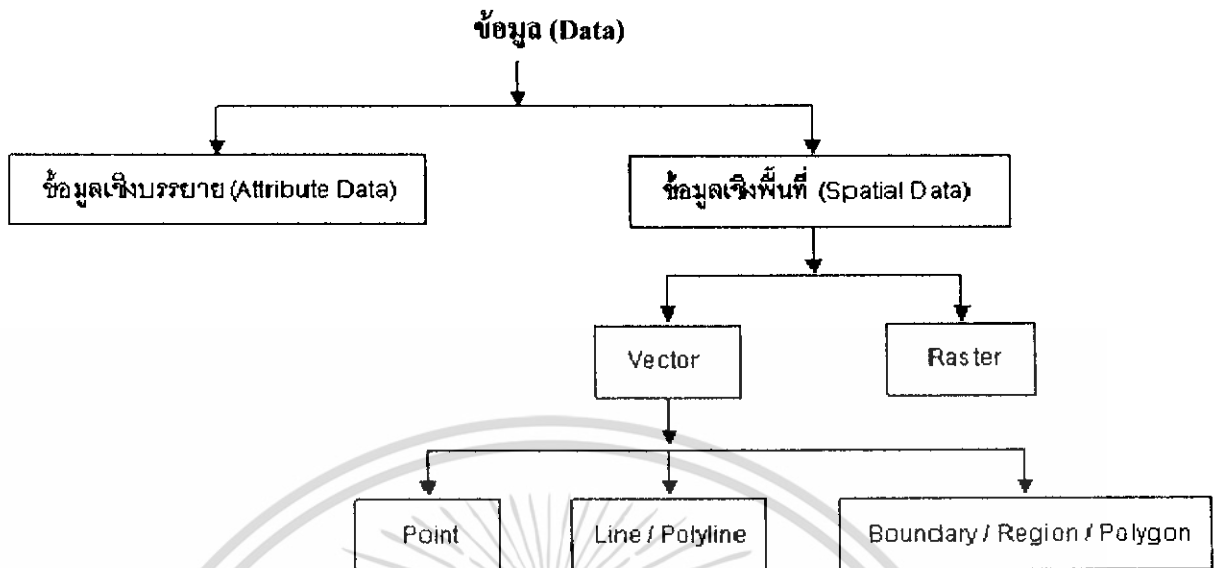
2.1.1 องค์ประกอบของ GIS (Components of GIS)



รูปที่ 2.1 องค์ประกอบระบบสารสนเทศภูมิศาสตร์ GIS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.1.1 ข้อมูล (Data)



รูปที่ 2.2 ประเภทของข้อมูล

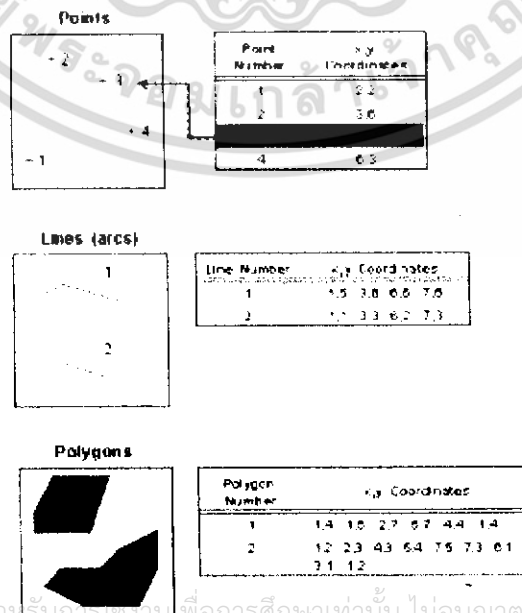
ข้อมูลทางด้าน GIS จะแบ่งออกเป็น 2 ประเภทใหญ่ๆ คือ

- ส่วนของข้อมูลเชิงพื้นที่ (Spatial Data)
- ส่วนของข้อมูลเชิงบรรยาย (Attribute Data)

ส่วนของข้อมูลเชิงพื้นที่ (Spatial Data) คือ ข้อมูลที่แสดงตำแหน่งที่ตั้งทางภูมิศาสตร์ (geo-referenced data) ของรูปลักษณะของพื้นที่ (graphic feature) ซึ่งมี 2 แบบ คือ ข้อมูลที่แสดงทิศทาง (Vector Data) และ ข้อมูลที่แสดงเป็นตารางกริด (Raster Data)

ข้อมูลประเภทแสดงทิศทาง (Vector Data)

ประกอบด้วยลักษณะ 3 แบบ คือ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น

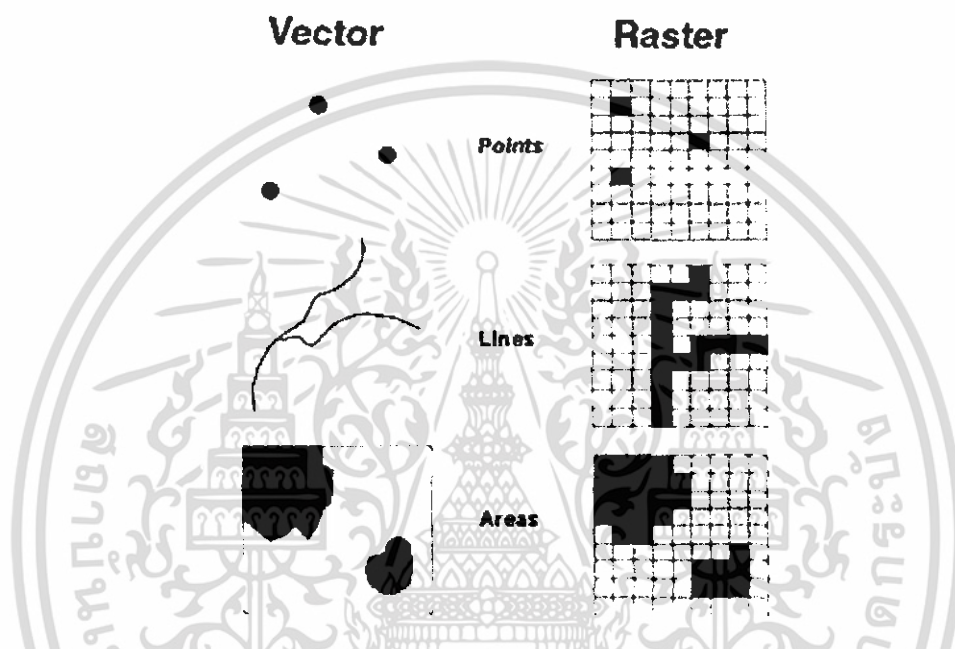
รูปที่ 2.3 ข้อมูลประเภท Vector แบบจุด เส้น และ รูปปิด ตามลำดับที่มีการนำไปใช้

1. จุด (Point) เช่น ที่ตั้งสำนักงานประปา ที่ตั้งร้านค้า วัด โรงเรียน มาตรฐานน้ำ ประตูน้ำ และ หัวดับเพลิง

2. ปลายเส้น (Polyline) เช่น ถนน แม่น้ำ ท่อประปา และ เส้นชั้นความสูง

3. ขอบเขตพื้นที่หรือเส้นรอบรูปปิด (Polygon) เช่น ผู้ใช้น้ำ พื้นที่ป่าไม้ พื้นที่เมือง พื้นที่พาณิชย์ และ พื้นที่แหล่งน้ำ

ข้อมูลประเภทแสดงเป็นตารางกริด (Raster Data)



รูปที่ 2.4 ข้อมูลประเภท Raster แบบจุด เส้น และ รูปปิด เทียบกับ Vector ตามลำดับ

จะเป็นลักษณะตารางสี่เหลี่ยมเล็กๆ (grid cell or pixel) เท่ากันและต่อเนื่องกัน ซึ่งสามารถอ้างอิงค่าพิกัดทางภูมิศาสตร์ได้ ขนาดของตารางกริด หรือความละเอียด (resolution) ในการเก็บข้อมูล จะใหญ่หรือเล็กขึ้นอยู่กับการจัดแบ่งจำนวนแถว (row) และจำนวนคอลัมน์ (column) ตัวอย่างข้อมูลที่จัดเก็บ โดยใช้ตารางกริด เช่น ภาพถ่ายดาวเทียม ข้อมูลระดับค่าความสูง เป็นต้น

2.1.1.2 เครื่องคอมพิวเตอร์และอุปกรณ์ต่างๆ (Hardware)

เครื่องคอมพิวเตอร์รวมกันเรียกว่า ระบบฮาร์ดแวร์ (Hardware) โดยจะประกอบด้วย อุปกรณ์นำเข้า เช่น Mouse, Digitizer, อุปกรณ์จัดเก็บข้อมูล เช่น แผ่นดิสก์ อุปกรณ์แสดงผลข้อมูล เช่น จอแสดงผล, Printer, Plotter เป็นต้น ซึ่งอุปกรณ์แต่ละชนิดจะมีหน้าที่และคุณภาพแตกต่างกันออกไปทำให้ระบบฮาร์ดแวร์ทั้งหมดจะต้องมีสมรรถนะเพียงพอสำหรับการจัดเก็บและการจัดการข้อมูลที่มีปริมาณมาก รวมทั้งต้องสามารถรองรับกับการทำงานของซอฟต์แวร์ได้

2.1.1.3 โปรแกรมหรือระบบซอฟต์แวร์ (Software)

หมายถึง โปรแกรมที่ใช้ในการจัดการระบบและสิ่งงานต่างๆ เพื่อให้ระบบฮาร์ดแวร์ทำงาน โดยมีความสามารถในการนำเข้าข้อมูล จัดเก็บข้อมูล วิเคราะห์ แผลงข้อมูลได้

2.1.1.4 บุคลากร (People)

คือ ผู้ใช้ข้อมูล Hardware, Software ซึ่งถือได้ว่าบุคลากรเป็นสิ่งสำคัญในระบบงาน GIS ซึ่งประกอบด้วยผู้ใช้ระบบ (analyst) และผู้ใช้สารสนเทศ (user) ผู้ใช้ระบบ หรือผู้ชำนาญการ GIS จะต้องมีความชำนาญในหน้าที่ และได้รับการฝึกฝนมาแล้วเป็นอย่างดี พร้อมทั้งจะทำงานได้เต็มความสามารถ โดยทั่วไปผู้ใช้ระบบจะเป็นผู้เลือกระบบฮาร์ดแวร์ และระบบซอฟต์แวร์ เพื่อให้ตรงตามวัตถุประสงค์ และสนองตอบความต้องการ ของหน่วยงาน ส่วนผู้ใช้สารสนเทศ (user) คือนักวางแผน หรือผู้มีอำนาจตัดสินใจ (decision-maker) เพื่อนำข้อมูลมาใช้ในการแก้ไขปัญหาต่างๆ

2.1.1.5 วิธีการหรือขั้นตอนการทำงาน (Procedures/Methods)

คือ การทำงานของระบบสารสนเทศทางภูมิศาสตร์ ประกอบด้วย 5 ขั้นตอนหลัก คือ

1. การกำหนดวัตถุประสงค์ (Objective)
2. การจัดเตรียมฐานข้อมูล (Data Preparation)
3. การนำเข้าข้อมูล (Data Input)
4. การวิเคราะห์ข้อมูล (Data Analysis)
5. การแสดงผล (Output)

1. การกำหนดวัตถุประสงค์ (Objective)

การกำหนดวัตถุประสงค์ เป็นขั้นตอนแรกและสำคัญที่สุดในการดำเนินงานที่เกี่ยวข้องกับระบบสารสนเทศทางภูมิศาสตร์ ทั้งนี้ นักวิเคราะห์ GIS ต้องทราบวัตถุประสงค์ที่ชัดเจนก่อนการดำเนินงานในขั้นตอนต่างๆ ว่าต้องการสร้างงาน GIS เพื่อนำไปใช้อย่างไรบ้าง นำไปใช้แบบใด และใครเป็นผู้ใช้ หรือจะนำไปแก้ไขปัญหาอะไรบ้าง ปัญหาดังกล่าวสามารถหาคำตอบได้โดยเอกสารที่เป็นเอกสารที่ส่งงานไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GIS หรือไม่ และผลที่คาดว่าจะได้รับจากการวิเคราะห์คืออะไร และใครจะเป็นผู้นำผลการวิเคราะห์ ไปใช้ในขั้นต่อไป

2. การจัดเตรียมฐานข้อมูล (Data Preparation)

ข้อมูลที่เราจัดเตรียมมีอยู่ 2 ประเภท คือ ข้อมูลเชิงพื้นที่ (Spatial Data) เช่น แผนที่ต่าง ๆ และข้อมูลอธิบายพื้นที่ (Attribute Data) ซึ่งข้อมูลเหล่านี้อาจได้มาจากหน่วยงานที่เกี่ยวข้อง หรือได้มาจากการสำรวจข้อมูลภาคสนาม (Field Data Collection)

3. การนำเข้าข้อมูล (Data Input)

สามารถแบ่งออกได้เป็น 2 ประเภท คือ การนำเข้าข้อมูลเชิงพื้นที่ และข้อมูลทั่วไป การนำเข้าข้อมูลเชิงพื้นที่เป็นการแปลงข้อมูลเชิงพื้นที่ให้เป็นข้อมูลเชิงตัวเลข(Digital data)ซึ่งสามารถนำเข้าได้หลายวิธี เช่น Digitize table, Keyboard , Scanner นำเข้าข้อมูลแผ่นฟิล์ม (film importation) และ แปลงค่าพิกัดทางภูมิศาสตร์ที่จัดเก็บจากเครื่อง Global Positioning System (GPS) ทั้งนี้โปรแกรมที่ใช้ในการนำเข้ามีหลายโปรแกรม เช่น MapInfo, ARC/INFO, PAMAP, Microstation, SPAN, ERDAS, Arc View เป็นต้น ส่วนการนำเข้าข้อมูลอธิบายพื้นที่สามารถนำเข้าโดยโปรแกรม Spreadsheet หรือ โปรแกรมทั่วไป เช่น Excel , Lotus , FoxPro , Word หรือ โปรแกรม GIS

4. การวิเคราะห์ข้อมูล (Data Analysis)

ระบบสารสนเทศภูมิศาสตร์ (GIS) มีความสามารถในการนำเข้าข้อมูลเชิงพื้นที่ที่หลาย ๆ ชั้นข้อมูล (layer) แล้วสามารถนำชั้นข้อมูลเหล่านั้นมาซ้อนทับกัน (overlay) เพื่อทำการวิเคราะห์ และกำหนดเงื่อนไขต่าง ๆ เช่น การวิเคราะห์เกี่ยวกับผลกระทบหลายของดิน โดยการวิเคราะห์ข้อมูลจากแผนที่ดิน องค์ประกอบในการกักร่อนดิน เส้นชั้นระดับความสูง แผนที่การใช้ที่ดิน ข้อมูลจากดาวเทียม รวมทั้งข้อมูลน้ำฝนในช่วงระยะเวลาหนึ่ง เพิ่มข้อมูลแต่ละเพิ่มจะถูกประมวลผลตามเกณฑ์ที่ตั้งไว้แล้วถูกนำซ้อนกัน

การวิเคราะห์ จะวิเคราะห์โดยใช้คอมพิวเตอร์ ตามวัตถุประสงค์ หรือตามแบบจำลอง (model) ต่างๆ ซึ่งอาจเป็นการเรียกค้นข้อมูลอย่างง่าย ๆ หรือซ้อนทับ เช่น โมเดลทางสถิติ หรือ โมเดลทางคณิตศาสตร์ เป็นต้น ทั้งนี้ เนื่องจากชั้นข้อมูลต่างๆ ถูกจัดเก็บโดยอ้างอิงค่าพิกัดทางภูมิศาสตร์ และมีการจัดเก็บอย่างมีระบบ ประมวลผลโดยใช้คอมพิวเตอร์ ผลที่ได้รับจากการวิเคราะห์ จะเป็นชั้นข้อมูลอีกลักษณะหนึ่งที่แตกต่างไปจากชั้นข้อมูลเดิม

5. การแสดงผล (Output)

ผลที่ได้รับจากการวิเคราะห์ข้อมูลสามารถนำเสนอหรือแสดงผลได้ทั้งบนจอคอมพิวเตอร์ (monitor) หรือผลิตออกเป็นเอกสาร(แผนที่และตาราง) โดยใช้ Printer , Potter หรือสามารถแปลงข้อมูลเหล่านั้นไปสู่ระบบการทำงานใน โปรแกรมอื่น ๆ ในรูปแบบของแผนที่ (map) แผนภูมิ (chart) หรือตาราง (table)

2.1.2 ระบบพิกัดบนแผนที่

ระบบพิกัด (Coordinate System) เป็นระบบที่สร้างขึ้นสำหรับใช้อ้างอิง ในการกำหนด ตำแหน่ง หรือ บอกตำแหน่งพื้น โลกจากแผนที่ มีลักษณะเป็นตาราง โครงข่ายที่เกิดจากตัดกันของ เส้นตรงสองชุดที่ถูกกำหนดให้วางตัวใน แนวเหนือ-ใต้ และ แนวตะวันออก- ตะวันตก ตามแนว ของจุดศูนย์กำเนิด (Origin) ที่กำหนดขึ้น ค่าพิกัดที่ใช้อ้างอิงในการบอกตำแหน่งต่าง ๆ จะใช้ค่า ของหน่วยที่นับออกจากจุดศูนย์กำเนิดเป็นระยะเชิงมุม (Degree) หรือเป็นระยะทาง (Distance) ไป ทางเหนือหรือใต้และตะวันออกหรือตะวันตก ตามตำแหน่งของค่าพิกัดที่ต้องการหาค่าพิกัดที่ กำหนดตำแหน่งต่างๆ จะถูกเรียกอ้างอิงเป็นตัวเลขในแนวตั้งและแนวนอนตามหน่วยวัดระยะ

สำหรับระบบพิกัดที่ใช้อ้างอิงกำหนดตำแหน่งบนแผนที่ที่นิยมใช้กับแผนที่ในปัจจุบัน มี อยู่ด้วยกัน 2 ระบบ คือ

- 1) ระบบพิกัดภูมิศาสตร์ (Geographic Coordinate)
- 2) ระบบพิกัดกริด (Grid Coordinate) ในที่นี้จะกล่าวถึง พิกัดกริดแบบ UTM (Universal Transverse Mercator)

2.1.2.1 ระบบพิกัดภูมิศาสตร์ (Geographic Coordinate System)

เป็นระบบพิกัดที่กำหนดตำแหน่งต่าง ๆ บนพื้นโลก ด้วยวิธีการอ้างอิงบอกตำแหน่งเป็น ค่าระยะเชิงมุมของละติจูด (Latitude) และ ลองจิจูด (Longitude) ตามระยะเชิงมุมที่ห่างจากศูนย์ กำเนิด (Origin) ของละติจูดและลองจิจูด ที่กำหนดขึ้นสำหรับศูนย์กำเนิดของละติจูด (Origin of Latitude) นั้นกำหนดขึ้นจากแนวระดับ ที่ตัดผ่านศูนย์กลางของโลกและตั้งฉากกับแกนหมุน เรียก แนวระนาบศูนย์กำเนิดนั้นว่า เส้นศูนย์สูตร (Equator) ซึ่งแบ่งโลกออกเป็นซีกโลกเหนือ และซีกโลกใต้ ฉะนั้นค่าระยะเชิงมุมของละติจูด จะเป็นค่าเชิงมุมที่เกิดจากมุมที่ศูนย์กลางของโลก กับ แนวระดับฐานกำเนิดมุมที่เส้นศูนย์สูตร ที่วัดค่าของมุมออกไปทั้งซีกโลกเหนือและซีกโลกใต้ ค่า ของมุมจะสิ้นสุดที่ขั้วโลกเหนือและขั้วโลกใต้มีค่าเชิงมุม 90 องศาพอดี ดังนั้น การใช้ค่าระยะเชิงมุมของละติจูดอ้างอิง บอกตำแหน่งต่างๆ นอกจากจะกำหนดเรียกค่าวัดเป็น องศา ลิปดา

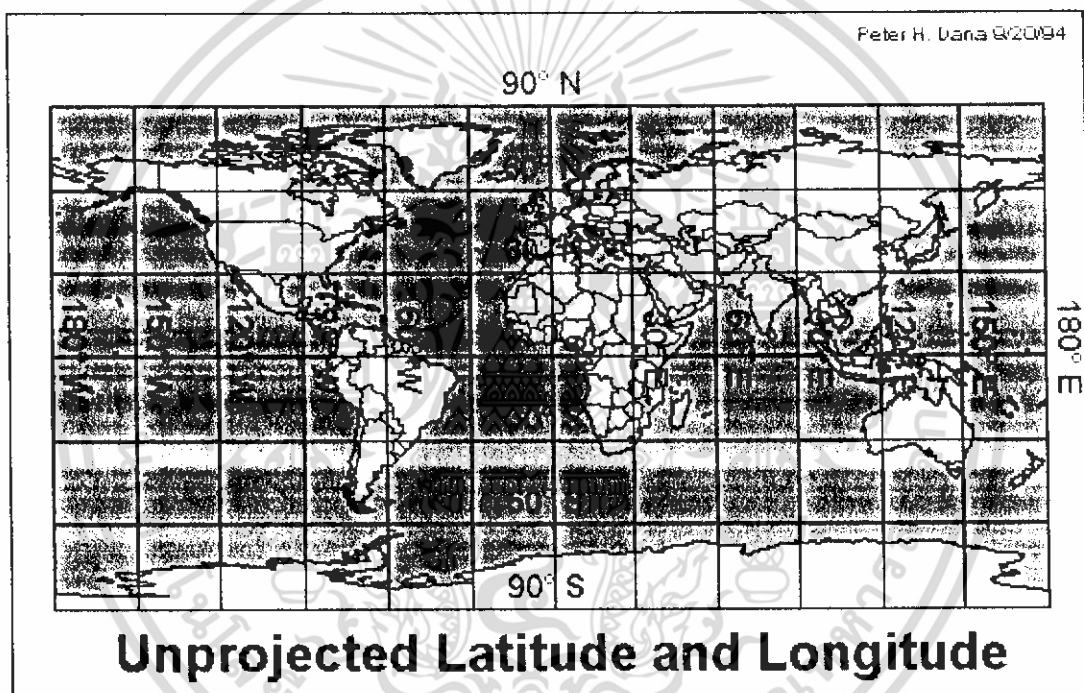
และฟิลิปดา แล้วจะบอก ซีกโลกเหนือหรือใต้กำกับด้วยเสมอ เช่น ละติจูดที่ 30 องศา 00 ลิปดา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

15 ฟิลิปดาเหนือ

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนศูนย์กำหนดของลองจิจูด (Origin of Longitude) นั้นก็กำหนดขึ้นจากแนวระนาบทางตั้งที่ผ่านแกนหมุนของโลกตรงบริเวณตำแหน่งบนพื้นโลกที่ผ่านหอดูดาวเมืองกรีนวิช (Greenwich) ประเทศอังกฤษ เรียกศูนย์กำเนิดนี้ว่า เส้นเมริเดียนเริ่มแรก (Prime Meridian) เป็นเส้นที่แบ่งโลกออกเป็นซีกโลกตะวันตกและซีกโลกตะวันออกค่าระยะเชิงของลองจิจูดเป็นค่าที่วัดมุมออกไปทางตะวันตก และตะวันออกของเส้นเมริเดียนเริ่มแรก วัดจากศูนย์กลางของโลกตามแนวระนาบ ที่มีเมริเดียนเริ่มแรกเป็นฐานกำเนิดมุมค่าของมุมจะสิ้นสุดที่เส้นเมริเดียนตรงข้ามเส้นเมริเดียนเริ่มแรกมีค่าของมุมซีกโลกละ 180 องศา การใช้ค่าอ้างอิงบอกตำแหน่งก็เรียกกำหนดเช่นเดียวกับละติจูด แต่ต่างกันที่จะต้องบอกเป็นซีกโลกตะวันตก หรือตะวันออกแทน เช่น ลองจิจูดที่ 90 องศา 00 ลิปดา 00 ฟลิปดาตะวันตก



รูปที่ 2.5 แสดงระบบพิกัดภูมิศาสตร์

2.1.2.2 ระบบพิกัดกริด UTM (Universal Transvers Mercator coordinate System)

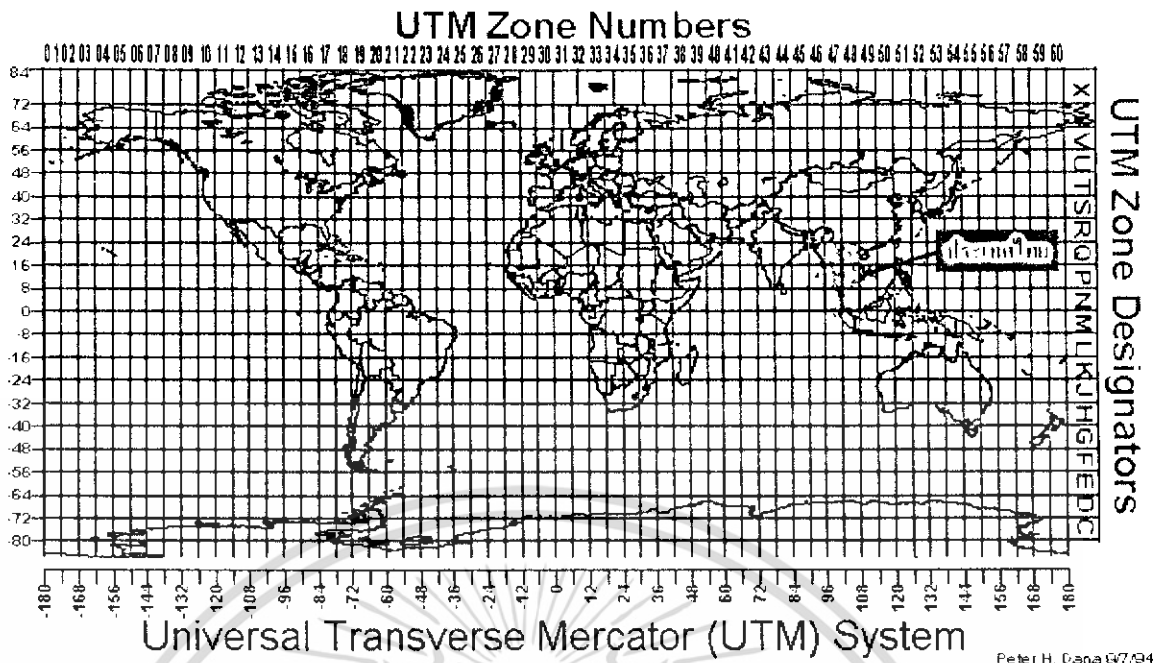
พิกัดกริด UTM (Universal Transvers Mercator) เป็นระบบตารางกริดที่ใช้ช่วยในการกำหนดตำแหน่งและใช้อ้างอิง ในการบอกตำแหน่ง ที่นิยมใช้กับแผนที่ในกิจการทหารของประเทศต่าง ๆ เกือบทั่วโลกในปัจจุบัน เพราะเป็นระบบตารางกริดที่มีขนาดรูปร่างเท่ากันทุกตาราง และมีวิธีการกำหนดบอกค่าพิกัดที่ง่ายและถูกต้องเป็นระบบกริดที่นำเอาเส้นโครงแผนที่แบบ Universal Transvers Mercator Projection ของ Gauss Krugger มาใช้ตัดแปลงการถ่ายทอด

รายละเอียดของพื้นผิวโลกให้รูปทรงกระบอก Mercator Projection อยู่ในตำแหน่ง Mercator Projection (แกนของรูปทรงกระบอกจะทับกับแนวเส้นเอคเวเตอร์ และตั้งฉากกับแนวแกนของขั้ว

โลก) ประเทศไทยเราได้นำเอาเส้นโครงแผนที่แบบ UTM นี้มาใช้กับการทำแผนที่กิจการทหาร ภายในประเทศจากรูปถ่ายทางอากาศในปี 1953 ร่วมกับสหรัฐอเมริกา เป็นแผนที่มาตราส่วน 1:50,000 ชุด 708 และปรับปรุงใหม่เป็นชุด L 7017 ที่ใช้ในปัจจุบัน

แผนที่ระบบพิกัดกริด ที่ใช้เส้นโครงแผนที่แบบ UTM เป็นระบบเส้นโครงชนิดหนึ่งที่ใช้ ผีวรูปทรงกระบอกเป็นผิวแสดงเส้นเมริเดียน (หรือเส้นลองจิจูด) และเส้นละติจูดของโลก โดยใช้ ทรงกระบอกตัดโลกระหว่างละติจูด 84 องศาเหนือ และ 80 องศาใต้ในลักษณะแกนรูป ทรงกระบอก ทำมุมกับแกนโลก 90 องศารอบโลก แบ่งออกเป็น 60 โซนๆ ละ 6 องศา โซน ที่ 1 อยู่ระหว่าง 180 องศา กับ 174 องศาตะวันตก และมีลองจิจูด 177 องศาตะวันตก เป็น เมริเดียนข่านกลาง (Central Meridian) มีเลขกำกับแต่ละโซนจาก 1 ถึง 60 โดยนับจากซ้าย ไป ทางขวาระหว่างละติจูด 84 องศาเหนือ 80 องศาใต้ แบ่งออกเป็น 2 ช่อง ช่องละ 8 องศา ยกเว้น ช่องสุดท้ายเป็น 12 องศา โดยเริ่มนับตั้งแต่ละติจูด 80 องศาใต้ ขึ้นไปทางเหนือ ให้ช่องแรกเป็น อักษร C และช่องสุดท้ายเป็นอักษร X (ยกเว้น I และ O) จากการแบ่งตามที่กล่าวแล้วจะเห็น พื้นที่ในเขตลองจิจูด 180 องศาตะวันตก ถึง 180 องศาตะวันออก และละติจูด 80 องศาใต้ ถึง 84 องศาเหนือ จะถูกแบ่งออกเป็นรูปสี่เหลี่ยมผืนผ้า 1,200 รูป แต่ละรูปมีขนาดกว้างยาว 6 องศา x 8 องศา จำนวน 1,140 รูป และกว้างยาว 6 องศา x 12 องศา จำนวน 60 รูป รูปสี่เหลี่ยมนี้ เรียกว่า Grid Zone Designation (GZD) การเรียกชื่อ Grid Zone Designation ประเทศไทยมีพื้นที่ อยู่ระหว่างละติจูด 5 องศา 30 ลิปดาเหนือ ถึง 20 องศา 30 ลิปดาเหนือ และลองจิจูดประมาณ 97 องศา 30 ลิปดา ตะวันออก ถึง 105 องศา 30 ลิปดา ตะวันออก ดังนั้น ประเทศไทยจึงตกอยู่ใน GZD 47N 47P 47Q 48N 48P และ 48Q การอ่านค่าพิกัดกริดเพื่อให้พิกัดค่ากริดในโซนหนึ่งๆ มี ค่าเป็นบวกเสมอ จึงกำหนดให้มีศูนย์สมมุติขึ้น 2 แห่ง ดังนี้

- ในบริเวณที่อยู่เหนือเส้นศูนย์สูตร : เส้นศูนย์สูตรมีระยะห่างจากศูนย์สมมุติเท่ากับ 0 เมตร และเส้นเมริเดียนข่านกลางห่างจากศูนย์สมมุติ 500,000 เมตร ทางตะวันออก
- ในบริเวณที่อยู่ใต้เส้นศูนย์สูตร : เส้นศูนย์สูตรมีระยะห่างจากศูนย์สมมุติไปทางเหนือ 10,000,000 เมตร และเมริเดียนข่านกลางห่างจากศูนย์สมมุติ 500,000 เมตร ทางตะวันออก



รูปที่ 2.6 แสดงระบบพิกัดกริด UTM

2.2 OGC WMS

OGC เป็นองค์กรระหว่างประเทศที่ประกอบด้วยบริษัท รัฐบาล และมหาวิทยาลัยต่างๆ โดยมีเป้าหมายที่จะร่วมกันพัฒนาเทคโนโลยีเกี่ยวกับข้อมูลทางด้านพื้นที่และสถานที่

The OpenGIS Consortium (OGC) เป็นองค์กรเกี่ยวกับมาตรฐานโพรโตคอลการเรียกดูแผนที่ผ่านเว็บที่เรียกว่า Web Map Service implementation Specification (WMS) เพื่อให้ผู้ใช้สามารถเรียกข้อมูลแผนที่จากเครื่องแม่ข่ายแผนที่หลายเครื่องพร้อมกัน และเครื่องแม่ข่ายแผนที่เป็นระบบได้ก็ได้ที่ปฏิบัติตามโพรโตคอลมาตรฐานของ OGC มาตรฐานเดียวกันนี้ เพื่อสร้างเครื่องแม่ข่ายแผนที่ชนิด Interoperable จากการริเริ่มของ OGC ก็กำลังได้รับการร่างให้เป็นมาตรฐานหนึ่งของ ISO ภายใต้ Working Item 19128

ตามมาตรฐาน WMS ได้กำหนดไว้ว่าผู้ที่จะสามารถให้บริการตัวเองเป็น map server ได้จะต้องมีบริการอย่างน้อย 2 บริการ คือ

- 1.) GetCapabilities บริการขอข้อมูลที่บอกว่า map server นั้นมีบริการอะไรบ้าง มีข้อมูลเชิงพื้นที่เลขอร์ใดบ้าง ในรูปแบบของเอกสาร XML
- 2.) GetMap บริการเรียกแผนที่ ซึ่งแสดงเป็นรูปภาพ ในรูปแบบต่างๆ และสามารถมีบริการที่เป็นตัวเลือก (Optional) คือ GetFeatureInfo บริการข้อมูลเกี่ยวกับสถานที่ที่แสดงอยู่บนแผนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 Web Services

Web Services เป็นบริการยุคใหม่ในวงการเว็บ ผู้ใช้เพียงแค่ดึงบริการต่างๆ บนเว็บมาใช้งาน ภาษาที่ใช้เป็นแกนกลางในการพัฒนาเว็บเซอร์วิส คือ XML

Web Services (WS) คือ แอปพลิเคชันหรือโปรแกรมที่ทำงานอย่างใดอย่างหนึ่งในลักษณะให้บริการ โดยจะถูกเรียกใช้งานจากโปรแกรมอื่นๆ (php, asp, java, python) ผ่านทาง HTTP Protocol การให้บริการของ Web Services นั้นจะถูกอธิบายด้วยเอกสาร WSDL ซึ่งจะมีคุณสมบัติของการบริการเขียนกำกับไว้ และ มีการนำเสนอให้สาธารณะชนรับทราบตามมาตรฐาน UDDI ผู้ใช้จึงสามารถค้นหาบริการ ได้โดยไม่ต้องรู้ที่อยู่จริงของแอปพลิเคชันหรือโปรแกรมนั้นๆ

2.3.1 Web Application และ Web Services

เทคโนโลยีในการกระจายข่าวสารข้อมูลทาง Internet ในปัจจุบันก็คือ เว็บเพจ แต่จากการที่มันมีความสามารถที่จะทำงานได้ด้วยการรวมภาษาทั้ง Client และ Server Side Script ไว้ในตัวเองเช่นภาษา VBScript, Java Script หรือ ASP, PHP, JSP นั้นทำให้เว็บเพจมีลักษณะคล้าย Application จึงถูกเรียกรวมกันว่า Web Application

Web Application สามารถตอบสนองความคิด Distributed Processing ได้ในระดับหนึ่ง ซึ่งก็คือ การแบ่งการประมวลผลไว้ที่ฝั่ง Client และฝั่ง Server และมักจะมีการใช้ database ควบคู่กับการทำ Web Application ไปด้วยตามความต้องการในการทำ E-Business และ E-Commerce ที่กำลังเป็นที่นิยมในปัจจุบัน และเกิดปัญหาที่ตามมาคือ เรื่องของการจ่ายเงินหรือที่เรียกว่า E-payment หรือ Payment-Gateway ซึ่ง Web Application ที่ทำ E-Commerce ต้องใช้บริการจากธนาคาร Online ในการจัดเก็บเงินกับลูกค้า เพราะด้วยเทคโนโลยีนี้การให้บริการเก็บเงินจากธนาคาร Online จำเป็นที่ผู้ค้าต้องไปทำการตกลงกับธนาคาร และเขียนโปรแกรมให้ตรงตามมาตรฐานที่ธนาคาร Online กำหนดไว้ด้วยปัญหายุ่ยากในการค้นหา ติดต่อและตกลงในการขอใช้บริการเก็บเงินจากธนาคาร Online แนวคิด Web Service จึงดูเหมือนเป็นทางออกของปัญหานี้ ความเด่นของเทคโนโลยี Web Service นี้ก็คือ การทำให้ Web กับ Web สามารถติดต่อสื่อสารกันได้ด้วยเอกสาร XML ที่ทั้งคนและคอมพิวเตอร์เข้าใจ และคอมพิวเตอร์ยังสามารถนำข้อมูลนั้นไปประมวลผลต่อได้ ด้วยเอกสาร XML นี้เองทำให้ Web สามารถส่งข้อมูลที่เป็นไปให้อีก Web หนึ่งทำงานบางอย่างให้ หรือให้บริการนั่นเองทำให้เป็นการง่ายที่จะเขียนโปรแกรม ที่จะติดต่อสื่อสารหรือขอใช้บริการเก็บเงินจากธนาคาร Online แต่สำหรับ Web Application ที่ใช้การส่งข้อมูลเป็น html ทำให้ข้อมูลนั้นไม่สามารถนำไปใช้ต่อได้ การเขียนโปรแกรมจึงยุ่งยากตามที่กล่าวด้านบน

แนวคิดของ Web Service ก็คือ Web ที่สามารถทำงานอะไรบางอย่างหรือก็คือให้บริการบางอย่างจากการร้องขอจากต่าง Server ด้วยเหตุนี้ทำให้เทคโนโลยี Web Service ถือคือแนวคิดเอกสารเป็นศูนย์กลางที่ช่วยให้การสื่อสารข้อมูลระหว่างระบบต่างๆ สามารถทำได้ง่ายขึ้นโดยไม่ต้องอาศัยการตั้งค่าใดๆ ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Distributed Processing มากกว่า Web Application และเมื่อประกอบกับการที่ Web Service มี UDDI ทำให้ Web Service สามารถค้นหาบริการต่างๆ ที่ต้องการได้จากทั่วทุกมุมโลก ในอนาคต อาจเป็นไปได้ว่า Application อาจเป็นเพียงแค่การรวม Service ที่แต่ละ Web Service มีบริการมาให้ใช้เท่านั้นเอง

2.3.2 ประโยชน์ของการทำงานร่วมกันระหว่าง Application กับ Web Services

Web Service ช่วยให้การเข้าถึงข้อมูลสารสนเทศจากแอปพลิเคชันที่ต่างกันเป็นไปได้โดยง่าย โดยแอปพลิเคชันนั้นๆ สามารถเขียนด้วย Java และรันอยู่บน Sun Solaris Application Server หรืออาจจะเขียนด้วย C++ และรันอยู่บน Windows NT หรืออาจจะเขียนด้วย Perl และรันอยู่บนเครื่อง Linux ซึ่งมาตรฐานของ Web Service ทำให้อินเทอร์เน็ตเฟรมเวิร์กของแอปพลิเคชันเหล่านี้ ถูกอธิบายโดย WSDL และทำให้อยู่ในมาตรฐานของ UDDI หลังจากนั้น จึงสามารถติดต่อสื่อสารถึงกันโดย XML ผ่าน SOAP อินเทอร์เน็ต

Web Service สามารถถูกเรียกใช้ภายในองค์กรเองหรือจากภายนอกองค์กรโดยผ่าน ไฟร์วอลล์ ดังนั้นจึงมีองค์กรใหญ่ๆ มากมาย กำลังพัฒนาระบบที่มีอยู่ของตน ให้เข้ากับ Web Service ซึ่งนับเป็นการลงทุนที่คุ้มค่า เนื่องจาก Web Service สามารถเพิ่มศักยภาพในการทำงานขององค์กร อีกทั้งลดค่าใช้จ่ายในการจัดการทรัพยากรขององค์กรได้อีกทางหนึ่ง

นอกจากนั้น Web Service ยังสามารถใช้ร่วมกับ Web Application โดยส่งผ่านข้อมูลทางอินเทอร์เน็ตได้อีกด้วยซึ่งนับเป็นวิธีที่มีประสิทธิภาพในการติดต่อสื่อสารกับลูกค้า หรือหุ้นส่วน ถึงแม้จะต้องคำนึงถึงระบบรักษาความปลอดภัย และการจัดการรายการของข้อมูลอยู่ก็ตาม แต่ Web Service ได้ใช้มาตรฐานทั่วไปของ internet เรื่องดังกล่าวจึงนับเป็นเรื่องธรรมดาของการสื่อสารผ่านระบบอิเล็กทรอนิกส์

2.3.3 เทคโนโลยีที่ใช้ในการพัฒนา Web Services

2.3.3.1 XML (The Extensible Markup Language 1.0)

XML เป็นภาษา Markup ที่เป็น text-based ซึ่งทำให้เป็น มาตรฐานในการแลกเปลี่ยนข้อมูลบนอินเทอร์เน็ตอย่างรวดเร็ว ผู้ที่ทำหน้าที่รับผิดชอบ และกำหนดมาตรฐานของ XML คือ World Wide Web Consortium (W3C) ความแตกต่างระหว่าง XML กับ HTML คือ HTML ถูกนำมาใช้ในการสร้าง เว็บเพจ ที่สามารถแสดงผลได้โดยโปรแกรมเบราว์เซอร์ แต่ XML จะใส่ tags ได้อย่างอิสระ แล้วทำการส่ง XML ชุดนี้ไปประมวลผลยังแอปพลิเคชันใด ๆ ที่สามารถใช้ข้อมูลใน XML นี้

2.3.3.2 SOAP (Simple Object Access Protocol)

SOAP กลายเป็นสิ่งที่มีความสำคัญสำหรับ Web Services อย่างรวดเร็ว เป็นโปรโตคอลที่ผู้จัดทำ Web Services เลือกใช้ที่จะส่ง message ระหว่าง Web Services SOAP เป็น Transport Protocol ที่มี XML เป็นพื้นฐานและใช้ HTTP เป็นโปรโตคอลร่วมในการส่งผ่านเครือข่าย SOAP จะระบุวิธีในการเข้ารหัสส่วนหัว (Header Encoding) ของทั้ง HTTP และไฟล์ XML ไว้อย่างชัดเจนทั้งใน ส่วนของการติดต่อไปยังคอมพิวเตอร์เครื่องหนึ่งและส่งผ่านข้อมูลไปให้ รวมถึงระบุวิธีที่โปรแกรมซึ่งถูกเรียกนั้นจะส่งค่าคืนกลับมาด้วย

SOAP (Simple Object Access Protocol) เป็น XML-based โปรโตคอล (lightweight protocol) และใช้ HTTP เป็นโปรโตคอลร่วม สำหรับการแลกเปลี่ยนข้อมูลในสภาวะแวดล้อมแบบกระจายศูนย์ (decentralized, distributed environment) SOAP ได้กำหนดเมสเซจิงโปรโตคอล (Messaging Protocol) ระหว่างผู้ขอบริการ (requestor) กับผู้ให้บริการ (provider) เช่น ผู้ขอบริการสามารถติดต่อแลกเปลี่ยนข้อมูลกับผู้ให้บริการโดยใช้ RMI (Remote Method Invocation) ตามวิธีการของโปรแกรมแบบออปเจ็ค บริษัทไมโครซอฟท์ ไอบีเอ็ม โลตัส ซุสเซอร์แลนด (UserLand) และ ดีเวลลอปเปอร์เมนเตอร์ (DeveloperMenter) ได้ร่วมกันกำหนดมาตรฐานของ SOAP ขึ้น ซึ่งต่อมาได้มีบริษัทอีก 30 กว่าบริษัทเข้าร่วมและ จัดตั้งเป็น W3C XML Protocol Workgroup ขึ้น SOAP ได้กำหนดรูปแบบพื้นฐานของการสื่อสารแบบกระจายขึ้นโดย การพัฒนา SOA แม้ว่า SOA จะไม่ได้กำหนดเมสเซจิงโปรโตคอล (Messaging Protocol) ไว้แต่ SOAP ได้ถูกกำหนดให้เป็น Services-Oriented Architecture Protocol เรียบร้อยแล้ว เนื่องจากมันได้ถูกใช้ในการพัฒนา SOA อย่างแพร่ หลายแล้วนั่นเอง จุดเด่นของ SOAP ก็คือเป็นโปรโตคอลที่เป็นกลาง กล่าวคือ ไม่มีใครเป็นเจ้าของและเป็นโปรโตคอล ที่ทำงานกับโปรโตคอลอื่นหลายชนิด การพัฒนาที่อนุญาตให้ทำได้อย่างอิสระตามแพลตฟอร์มระบบปฏิบัติการ แบบจำลองทางวัตถุ (Object model) และ ภาษาโปรแกรมของผู้ที่ทำการพัฒนา

2.3.3.3 WSDL (Web Services Description Language)

WSDL เป็นภาษาที่ใช้อธิบายคุณลักษณะการใช้บริการของ Web Services และวิธีการติดต่อกับ Web Services ความต้องการของนิยามนี้เกี่ยวข้องกับความต้องการของ distributed system ที่จะกำหนด 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 เป็นภาษา ที่อยู่ใน การดูแลของ W3C (World Wide Web Consortium) ซึ่งยังไม่เป็นมาตรฐานที่สมบูรณ์ เวอร์ชันที่ใช้งานอยู่ใน ปัจจุบันคือ WSDL 1.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.3.4 UDDI (Universal Description, Discovery and Integration)

UDDI เป็นมาตรฐานที่ให้ชุดพื้นฐาน APIs (Application Programming Interfaces) ของ SOAP ที่สามารถนำมาใช้ในการพัฒนาเป็นตัวแทนของผู้ให้บริการ (Service broker) UDDI ใช้สำหรับค้นหา Service ที่ต้องการและเมื่อได้มาแล้ว UDDI ยังจัดหาข้อตกลงในวิธีการที่จะใช้งานเปรียบได้กับสมุดหน้าเหลือง เป็นมาตรฐานที่จัดตั้งขึ้นโดยบริษัทไอบีเอ็ม บริษัทไมโครซอฟต์ และบริษัทอริบา (Arriba) ปัจจุบันมีบริษัทที่ร่วม กันกำหนดมาตรฐานของ UDDI มากกว่า 70 บริษัท ซึ่งมาตรฐานของ UDDI ถูกกำหนดให้เป็นมาตรฐานสำหรับ B2B interoperability

2.4 ความรู้เบื้องต้นของภาษา XML

2.4.1 XML

XML ย่อมาจาก “Extensible Markup Language” ถูกพัฒนาโดย W3C (The World Wide Web Consortium) เป็นภาษา Markup Language เช่นเดียวกับ HTML Markup ประกอบด้วยโค้ดหรือที่เรียกว่า แท็ก (tag) เป็นพื้นฐานสำคัญที่ใช้เพิ่มเข้าไปในข้อความ ในการปรับเปลี่ยนรูปแบบมุมมองหรือความหมาย ข้อความที่แท็กห่อหุ้มอยู่เรียกว่า ซอร์สโค้ด ตัวอย่างเช่น โปรแกรม Word Processing หรือ โปรแกรมพิมพ์งานประเภทแก้ไขข้อความ ต่างก็มีความสามารถในการจัดรูปแบบข้อความ ด้วยภาษา Markup

XML ไม่ได้ถูกพัฒนาขึ้นเพื่อมาแทนที่ HTML แต่ถูกพัฒนาขึ้นด้วยจุดมุ่งหมายที่แตกต่างจาก HTML ซึ่งความแตกต่างระหว่าง XML กับ HTML คือ

XML	HTML
ออกแบบขึ้นมาเพื่อใช้ในการอธิบายข้อมูล	ออกแบบขึ้นเพื่อใช้ในการแสดงข้อมูล
ให้ความสนใจว่าข้อมูลคืออะไร	ให้ความสนใจในรูปแบบของการแสดงข้อมูล
มีประโยชน์ในการแลกเปลี่ยนข้อมูลในระบบที่แตกต่างกัน (Incompatible System)	

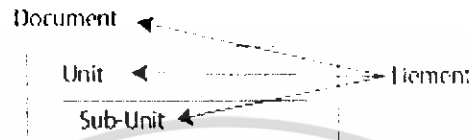
ตารางที่ 2.1 ความแตกต่างระหว่าง XML กับ HTML

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้างของ XML

1. Logical Structure

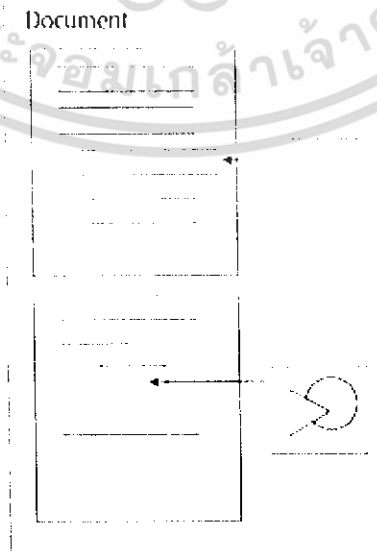
ภายในเอกสาร XML หนึ่ง จะถูกแบ่งออกเป็นหน่วย (Units) และ ในแต่ละหน่วย จะสามารถแบ่งย่อย ลงไปอีกได้ เรียกว่า หน่วยย่อย (Sub-Units) เราเรียกส่วนประกอบในเอกสารนี้ว่า Element ดังรูป



รูปที่ 2.7 แสดงโครงสร้างของเอกสาร XML ในมุมมอง Logical Structure

2. Physical Structure

Physical Structure จะอนุญาตให้เก็บส่วนประกอบของเอกสาร ซึ่งเรียกว่า Entities แยกออกจากตัวเอกสาร ได้ ในกรณีที่มีข้อมูลนั้นเป็นข้อมูลที่ถูกเรียกประจำ รวมถึงเอกสารที่ไม่ใช่ XML เช่นข้อมูลที่เป็นรูปภาพ เป็นต้น โดย Entities เหล่านี้จะถูกเรียกใช้ได้โดยการอ้างอิง ดังรูป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 รูปที่ 2.8 แสดงโครงสร้างของเอกสาร XML ในมุมมอง Physical Structure
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้างของเอกสาร XML มีลักษณะ โครงสร้างเป็นลำดับชั้นเหมือนต้นไม้ โดยมี Element ที่ซ้อนกัน อยู่ภายใน Element อื่นๆ



รูปที่ 2.9 แสดงตัวอย่างของเอกสาร XML

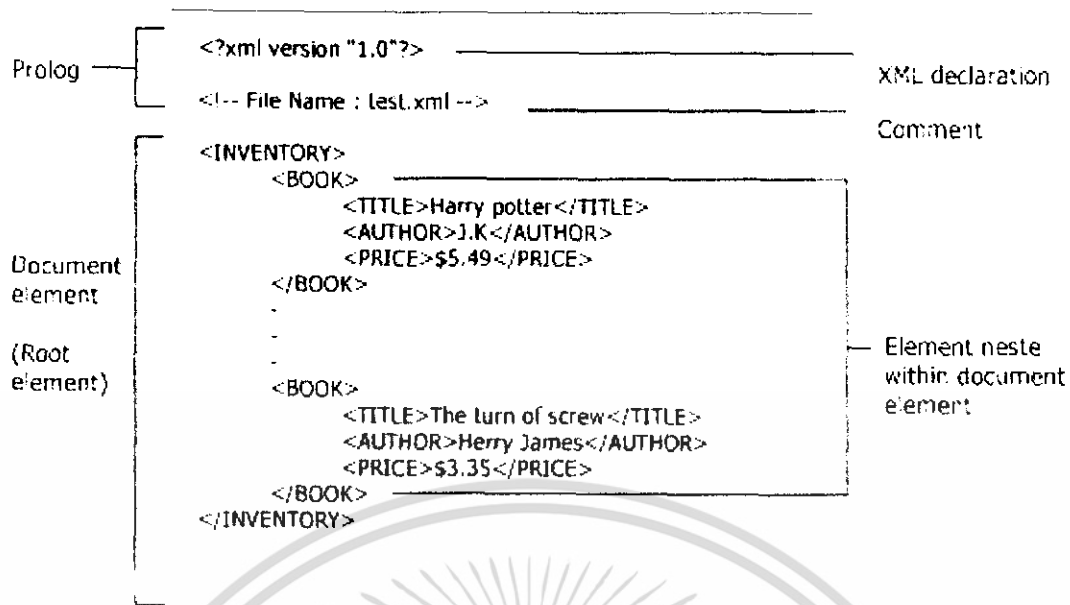
Parser

เอกสาร XML จะถูกตรวจสอบโดยใช้ Parser ซึ่งจะสามารถรายงานความผิดพลาดต่างๆ ที่เกิดขึ้น ในตัวเอกสาร

Meta-language

XML เป็น Meta-language นั่นคือ XML เป็นภาษาที่ใช้อธิบายภาษาอื่น การกำหนด Element สามารถกำหนดได้ตามความต้องการของผู้พัฒนา เพื่อให้สอดคล้องกับโปรแกรมการทำงานของผู้พัฒนาเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 แสดงโครงสร้างของเอกสาร

โครงสร้างของเอกสาร XML ประกอบด้วย

1. ส่วนแรก (Prolog) ในส่วนนี้ จะประกอบด้วย

- XML declaration เป็นการประกาศให้รู้ว่าเอกสารนี้คือ XML และเป็นการประกาศเวอร์ชันของ XML (ในตัวอย่างเป็นเวอร์ชัน 1.0) การใส่ค่า XML declaration จะประกาศหรือไม่ประกาศก็ได้ แต่ควรมีข้อกำหนดนี้ในเอกสาร

- บรรทัดว่าง เพื่อช่วยให้เอกสารนำอ่านขึ้น ตัวประมวลผลของ XML (XML Processor) จะข้ามและไม่นำบรรทัดว่างเหล่านั้นมาประมวลผล

- หมายเหตุ (Comment) เพื่อให้สามารถพิมพ์ข้อความที่ต้องการ อาจเป็นข้อความ ที่ใช้อธิบายจุดประสงค์ของเอกสาร เป็นต้น จะมีหรือไม่ก็ได้ เช่นเดียวกับบรรทัดว่าง ตัวประมวลผลของ XML จะข้ามและไม่นำหมายเหตุมาประมวลผล

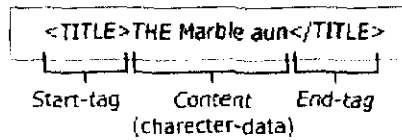
2. ส่วนที่สอง (Document element)

ในส่วนที่สองเรียกว่า Document element หรือ Root element ซึ่งสามารถบรรจุ Element เพื่อเติม ในเอกสาร XML ได้ ในเอกสาร XML นั้น Element จะแสดงลักษณะโครงสร้างของเอกสาร และแสดงส่วนประกอบของเนื้อหาเอกสารอยู่ภายใน จากรูปที่ 2.11 เป็นตัวอย่างของข้อมูลใน Book Element ซึ่งประกอบด้วย TITLE, AUTHOR, PRICE

Element

สัญลักษณ์ Element ประกอบด้วยแท็กเริ่มต้น (Start-tag), เนื้อหาภายใน Element และแท็กปิดท้าย (End-tag)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 แสดงส่วนประกอบของ Element

Attributes

ลักษณะของการกำหนด Attributes ในเอกสาร XML จะมีลักษณะเดียวกับ HTML โดยมี การกำหนด Attribute ในส่วน Start-tag

Attributes ไม่ได้เป็นส่วนหนึ่งของข้อมูล แต่ถูกใช้เพื่อเป็นส่วนอธิบายเพิ่มเติมให้กับ Element แต่ละตัว ในเอกสาร HTML

```
<IMG SRC="computer.gif">
```

จะเห็นว่า SRC คือ Attributes ที่ถูกกำหนดขึ้นเพื่อใช้อธิบายให้กับ Element IMG ใน เอกสาร XML สามารถเขียนได้เป็น

```
<file type="gif">computer.gif</file>
```

ค่าของ Attributes จะอยู่ในเครื่องหมายคำพูด ("...") จากตัวอย่าง File คือ Element หนึ่ง ซึ่งมี Attributes ชื่อ type ซึ่งเป็นส่วนอธิบายว่าไฟล์นี้ เป็นชนิดใด ดังในตัวอย่าง มีรูปแบบเป็น รูปภาพ (type="gif")

2.4.2 XSLT และ XPath

ด้วยเทคโนโลยี XML ทำให้เราสามารถพัฒนารูปแบบการเก็บข้อมูลของเรา กำหนด Tag ต่างๆ ได้เอง ต่างคนก็ต่างกำหนดตามใจชอบ ไม่ว่าจะเป็นองค์กรต่างๆ บริษัทห้างร้านไหนๆ หรือ แม้แต่ W3C เอง สิ่งเหล่านี้คือที่มาของความจำเป็นในการที่จะต้องแปลงลักษณะข้อมูลรูปแบบ XML แบบหนึ่งไปเป็นอีกแบบหนึ่ง

ทาง W3C จึงได้แนะนำให้ใช้ XSLT (Extensible Stylesheet Language Transformation) และก็คือ XPath (XML Path Language) ซึ่งถือเป็นเครื่องมืออันทรงพลังที่จะใช้แปลงรูปแบบ ข้อมูล XML ให้อยู่ในรูปแบบต่างๆ เช่น HTML หรือ XML หรือรูปแบบใดๆ ก็ได้ เราจึงต้องใช้ XSLT ซึ่งตัวมันเองใช้ XPath เพื่อจะสร้างผลลัพธ์ที่ต้องการจากข้อมูล mark-up ที่เป็น XML

XSLT สไลต์ซึ่งถูกสร้างขึ้นบนโครงสร้างที่เรียกว่า เทมเพลต (Template) เทมเพลต จะเป็นตัวกำหนดสิ่งที่ต้องค้นหาจากต้นฉบับ

XSLT เขียนขึ้นด้วย XML หมายความว่า มีอิลิเมนต์และแอตทริบิวต์พิเศษของ XSLT ที่ ใช้เพื่อสร้างสไลต์ชนิดในรูปแบบเฉพาะตัวได้

ตัวอย่าง

```
<xsl:template match="first"> First tag found! </xsl:template>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่ไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใน XSLT เท็มเพลตถูกกำหนดขึ้น โดยใช้อิลิเมนต์ `<xsl:template>` ที่มี 2 ส่วนสำคัญคือ แอคตริวิตี `match` และเนื้อหาของเท็มเพลต

แอคตริวิตี `match` จะกำหนดรูปแบบในต้นไม้ต้นฉบับดั้งเดิมในตัวอย่างข้างต้น จะใช้เท็มเพลตนี้กับทุกโหนดในต้นไม้ต้นฉบับที่จับคู่กับรูปแบบนั้น ๆ ซึ่งในกรณีนี้ก็จะใช้เท็มเพลตนี้กับทุกอิลิเมนต์ที่มีชื่อว่า `first`

คุณได้ระบุสิ่งที่จะได้ออกมาเป็นต้นไม้ผลลัพธ์เอาไว้ในเนื้อหาของอิลิเมนต์ `template` ในที่นี้เราได้กำหนดข้อความบางอย่างขึ้นมาง่าย ๆ ถ้าเท็มเพลตนี้ถูกรวมเข้าไปในสไตลชีตแล้วทุก ๆ ครั้งที่กลไก XSLT ค้นพบอิลิเมนต์ `<first>` ในต้นไม้ต้นฉบับมันก็จะให้เอาต์พุตออกมาเป็นข้อความว่า “First tag found!”

XPath ใช้ในการกำหนดรายละเอียดของภาษาที่ใช้ในการสืบค้นข้อมูลบางส่วนของเอกสาร XML ทำให้แอปพลิเคชันต่างๆ สามารถค้นหาข้อมูลที่ต้องการได้ แทนที่จะต้องนำเอาข้อมูลทั้งหมดของเอกสารมาใช้งาน

2.4.3 DTD

DTD คือเพิ่มข้อมูล (หรือหลายเพิ่มข้อมูลที่ใช้งานร่วมกัน) ซึ่งบรรจุข้อกำหนด และกฎเกณฑ์ของเอกสาร ชุดข้อกำหนดเหล่านี้ สำหรับการกำหนดรูปแบบ `element` ตัวอย่างเช่น หากต้องการเอกสารที่มี `element <LIST>` ที่มี `element <ITEM>` บรรจุอยู่ภายใน ข้อกำหนดในเพิ่มข้อมูล DTD จะมีรูปแบบดังนี้

```
<!ELEMENT item (#pcdata)>
```

```
<!ELEMENT list (item)+>
```

ซึ่งอธิบายความหมายคือ `element items` บรรจุข้อความใดๆ และ `element list` บรรจุ `element item` อีกที ดังนั้น DTD เป็นรูปแบบภาษา ซึ่งทำให้สามารถตรวจสอบเอกสาร ที่นำเอาข้อกำหนด DTD ไปใช้ ว่าถูกจัดสร้างตามความต้องการหรือไม่ ทำให้ระบบการ `rendering` สามารถเข้าใจตัวเอกสารได้ดี และดึงไปใช้งานได้อย่างถูกต้อง

ในการจัดสร้าง DTD ทำให้เอกสารมีความเป็นรูปแบบ `well-formed` เรียบง่าย ๆ กันว่า เอกสารที่ถูกต้อง ในบางครั้งหากเรา ต้องการสร้างเอกสาร XML ที่สามารถแลกเปลี่ยนกับคนอื่น ๆ หรือส่วนอื่นๆ และให้สามารถเข้าใจโครงสร้างเดียวกัน นั่นก็คือ เอกสาร XML ของเรา และส่วนอื่นๆ จำเป็นต้องมีรูปแบบโครงสร้างเดียวกัน นั่นก็คือควรที่จะเลือกใช้ DTD เดียวกัน เพื่อบังคับโครงสร้างให้เหมือนกัน

2.4.4 กระบวนการ parsing

โดยปกติทั่วไปแล้ว การที่ web server สามารถ load file XML แล้วนำมาแสดงเป็น HTML ได้นั้นจะเข้าสู่กระบวนการ parsing จะมีอยู่ด้วยกัน 2 วิธีหลัก ๆ คือ

2.4.4.1 DOM (Document Object Model)

เป็นวิธีที่ประมวลโครงสร้างของเอกสาร XML ให้เป็นโครงสร้างแบบต้นไม้ เพื่อให้แอปพลิเคชันสามารถเข้าหาจุดต่างๆ ของโครงสร้างต้นไม้ได้ โดยที่ DOM จะโหลดไฟล์ XML ทั้งไฟล์ใน Memory ซึ่งก็มีทั้งข้อดีและข้อเสีย คือ

ข้อดี คือ ครั้งแรกสุดที่มีการเรียกใช้งานจะช้า แต่หลังจากนั้นการเข้าถึงจุดต่างๆ ของไฟล์จะเร็วเพราะถูกเก็บใน memory แล้ว

ข้อเสีย คือ ไม่เหมาะกับ file XML ที่มีขนาดใหญ่ โดยที่ขอบเขตจะถูกจำกัดด้วยหน่วยความจำที่เรียกใช้ในขณะนั้น

2.4.4.2 SAX (Simple API for XML)

เป็นการทำงานแบบ event-based API คือจะรายงานข้อมูลตั้งแต่จุดเริ่มต้นและสิ้นสุดของ elements ต่างๆ ไปให้แอปพลิเคชัน โดยไม่ต้องมีการสร้างโครงสร้างแบบต้นไม้ขึ้นมา ซึ่งจะเข้าถึงเอกสาร XML ทำงานได้ง่ายและไม่ซับซ้อน และที่สำคัญผู้ใช้สามารถทำ parsing เอกสารที่มีขนาดใหญ่กว่าปริมาณหน่วยความจำได้

2.4.5 GML (Geography Markup Language)

GML เป็นรูปแบบหนึ่งของเอกสาร XML ที่ถูกออกแบบขึ้นเพื่อจัดเก็บและแลกเปลี่ยนของข้อมูลทางภูมิศาสตร์ทั้งทางด้านความสัมพันธ์ และคุณลักษณะประจำตัวของข้อมูล ในรูปแบบที่แตกต่างกันและสามารถกำหนดวิธีที่จะแสดงข้อมูลทางภูมิศาสตร์นั้น โดยใช้ข้อกำหนดของ GML ที่ได้รับการรับรองโดยองค์กร (Opengeospatial Consortium) ในการอธิบายข้อมูลเชิงพื้นที่ เพื่อให้ผู้ที่ทำงานด้าน GIS ใช้เป็นพื้นฐานในการออกแบบและพัฒนาแผนที่ในรูปแบบต่างๆ และผู้ใช้สามารถดูแผนที่ ที่ถูกแสดงนั้นผ่านทาง browser มาตรฐานที่ใช้กันอยู่ทั่วไป

2.4.6 SVG (Scalable Vector Graphics)

ภาษาที่มีพื้นฐานจาก XML ถูกออกแบบมาเพื่อการแสดงผล กราฟิกบนเว็บเพจโดยเฉพาะ คล้ายกับ HTML ที่ถูกออกแบบ มาเพื่อควบคุมลักษณะหน้าตาของเว็บเพจ เช่น ลักษณะตัวอักษร การจัดรูปแบบหน้า ลักษณะตาราง สีพื้น เป็นต้น ในทำนองเดียวกับ HTML เราสามารถใช้ภาษา SVG ในการควบคุมลักษณะการ แสดงผลกราฟิกในหน้าเว็บเพจได้ เช่น แสดงเส้นตรง เส้นโค้ง รูปทรงทางเรขาคณิตต่างๆ หรือ รูปร่างอื่นๆ ที่เราสร้างขึ้นเอง ตลอดจนสีหรือการแรเงา และ

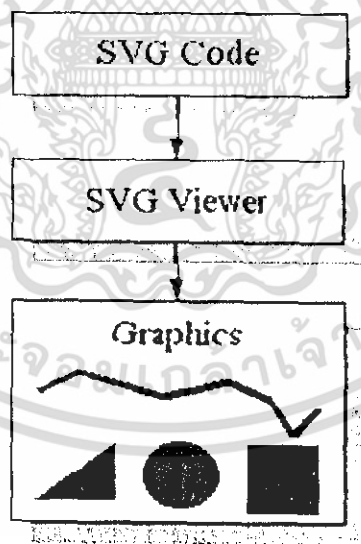
เอกสารนี้เป็นเอกสารที่สงวนไว้ว่าห้ามมิให้ดัดแปลงเนื้อหา และการตีความของเนื้อหาในเอกสารนี้ให้แตกต่างไปจากเอกสารต้นฉบับโดยไม่ได้รับอนุญาตจากเจ้าของลิขสิทธิ์

พีลเตอร์ต่างๆ นอกจากนี้เรายังสามารถเขียน โปรแกรม JavaScript เพื่อเข้าทำการควบคุม SVG ให้แสดงผลแบบ พลวัตหรือโต้ตอบกับผู้ใช้ได้ตามต้องการ

ข้อมูลส่วนใหญ่ที่ มนุษย์เราเกี่ยวข้องกับข้อมูลมักจะมีเชื่อมโยงกับข้อมูล เชิงตำแหน่งเสมอ เช่น ข้อมูลโรงเรียน โรงพยาบาลซึ่งสัมพันธ์กับตำแหน่งที่ตั้ง ข้อมูล สถานที่ท่องเที่ยว หรือ ประชากร ซึ่งมีการกระจายตัวสัมพันธ์กับลักษณะภูมิศาสตร์ ข้อมูลยอดขายสินค้ากับลักษณะการกระจายตัวของจุดกระจายสินค้า และปริมาณความหนาแน่นของเที่ยวรถบรรทุกสินค้าในแต่ละเส้นทาง เป็นต้น

SVG สามารถนำมาใช้ช่วยแสดงผลกราฟิกข้อมูลเชิง ตำแหน่งที่ต้องการมีปฏิสัมพันธ์กับผู้ใช้สูง รองรับการประมวลผลบนฝั่งเครื่องผู้ใช้ได้มากกว่าแผนที่ ที่เป็นรูปภาพธรรมดา พร้อมทั้งแสดงผลร่วมกับสารสนเทศอื่น ได้อย่างมีประสิทธิภาพการแสดงผลหรือ วิเคราะห์ร่วมกันระหว่างสารสนเทศทั้งสองแบบจะช่วยให้ระบบสารสนเทศใหม่มีความสามารถตอบสนองความต้องการ ผู้ใช้งาน ได้กว้างและลึกซึ้ง

เมื่อเรามีภาษาแล้วก็แน่นอนว่าเราต้องมีเครื่องมือ หรือโปรแกรมที่ทำหน้าที่อ่านภาษา SVG แล้วแปลออกมาเป็นเวกเตอร์กราฟิก ตามเจตนาารมณ์ที่ SVG ถูกออกแบบมา โปรแกรมที่สามารถเข้าใจ และแสดงผลข้อมูล SVG ให้อยู่ในรูปกราฟิกได้เรียกว่า โปรแกรม SVG Viewer ในกรณีที่ผู้ใช้ต้องการให้เว็บเบราว์เซอร์ดูข้อมูลกราฟิกที่อยู่ในรูป SVG ได้นั้น ผู้ใช้จะต้อง ติดตั้งโปรแกรมเสริมเพิ่มเติมประเภทดังกล่าว



รูปที่ 2.12 ขั้นตอนการแสดงผลใน SVG

2.4.6.1 คุณสมบัติของ SVG

เนื่องจาก SVG เกิดจากการใช้ความสามารถของภาษา XML ที่อนุญาตให้ผู้ใช้สร้างแท็ก (tag) และกำหนดความหมายของแท็กต่างๆ เพิ่มเติมได้เอง แท็กที่สร้างขึ้นจึงมีจุดประสงค์เพื่อใช้รองรับ หรืออธิบายพฤติกรรมต่างๆ ของข้อมูลเวกเตอร์กราฟิก โดยเฉพาะ เช่น แท็กที่ชื่อว่า circle ในภาษา SVG จะมีรูปแบบดังนี้

```
<circle cx="250" cy="250" r="80" fill="blue"></circle>
```

ความหมายของสัญลักษณ์ดังกล่าวหมายถึงรูปวงกลมซึ่งมีคุณสมบัติดังนี้

คำอธิบาย	สัญลักษณ์ที่ใช้
มีจุดศูนย์กลางที่พิกัด (x,y) ที่ (250,250)	cx="250" cy="250"
รัศมี 80 หน่วย	r="80"
ระบายสีน้ำเงินไว้ภายในวงกลม	fill="blue"

ตารางที่ 2.2 คำอธิบายและสัญลักษณ์ที่ใช้ของแท็ก circle

หรือแท็กที่ชื่อว่า path จะมีรูปแบบดังนี้

```
<path style="opacity:0.3;fill:#CC0000;stroke:none;" d="M 0 260,L140 260, L140 60">
</path>
```

ความหมายของสัญลักษณ์ดังกล่าวหมายถึงรูปปิดซึ่งมีคุณสมบัติดังนี้

คำอธิบาย	สัญลักษณ์ที่ใช้
ความเข้มสี (ความโปร่งแสง) 30%	opacity:0.3
ระบายสีภายในรูปปิด (สีแดง)	fill:#CC0000
สีของเส้นขอบรูป (ไม่มี)	stroke:none
เริ่มต้นลากจากจุด 0,260 ไปยัง 140,260 สิ้นสุดที่ 140,60	d="M 0 260, L140 260, L140 60"

ตารางที่ 2.3 คำอธิบายและสัญลักษณ์ที่ใช้ของแท็ก path

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Element อื่นๆของ SVG ที่มีการใช้งานบ่อย

<g>	ใช้ในการ tag ต่างๆให้อยู่ในกลุ่มเดียวกัน
<desc>	ใช้ในการเพิ่มเติมข้อมูลหรือคำอธิบาย
<defs>	ใช้ในการกำหนดค่าต่าง
<script>	ใช้ในการเขียน script
<cursor>	ใช้ในการเปลี่ยนรูป mouse cursor
<textpath>	ใช้ในการเขียนตัวอักษรให้เป็นไปตามเส้นทางที่กำหนด
<image>	ใช้ในการแสดงรูปภาพในรูปแบบอื่นๆเช่น jpeg, gif, png เป็นต้น

แอตทริบิวต์ อื่นๆของ SVG ที่มีการใช้งานบ่อย

id	ใช้ในการแยกแยะวัตถุต่างๆ
Xmlns	ใช้ในการอ้างอิง namespace
Visibility	กำหนดว่าให้วัตถุนั้นๆปรากฏอยู่หรือไม่
Point-event	ใช้กำหนดปฏิสัมพันธ์ที่มีกับวัตถุ
ViewBox	ใช้กำหนดขนาดการแสดงผลภาพ
zoomAndPan	ใช้กำหนดความสามารถในการย่อและขยายภาพ
OnClick	ใช้กำหนดพฤติกรรมของวัตถุเมื่อคลิกเมาส์
onmouseover	ใช้กำหนดพฤติกรรมของวัตถุเมื่อเมาส์อยู่บนวัตถุ
onmouseout	ใช้กำหนดพฤติกรรมของวัตถุเมื่อเลื่อนเมาส์ออกจากวัตถุ
Transform	ใช้ในการเคลื่อนย้าย, หมุน, ปรับอัตราส่วนของภาพ
Xlink:href	ใช้ในการอ้างอิง URL

2.4.6.2 ประโยชน์ในการใช้ SVG

1. SVG สามารถอ่านและแก้ไขได้โดยโปรแกรมหลายโปรแกรม เช่น notepad
2. สามารถเปลี่ยนขนาดของภาพได้โดยคุณภาพของภาพไม่ลดลง
3. SVG สามารถทำงานร่วมกับภาษาจาวาได้
4. SVG เป็นมาตรฐานที่เปิดเผยมให้คนทั่วไปทราบ
5. ข้อความใน SVG มีความสามารถในการเลือกและค้นหาจึงเหมาะแก่การนำมาทำเป็นแผนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 PostgreSQL/PostGIS

2.5.1 PostgreSQL

PostgreSQL คือ ระบบจัดการฐานข้อมูลเชิงวัตถุ-สัมพันธ์ (Object-Relational Database Management System หรือ ORDBMS) ซึ่งปรับปรุงจากต้นแบบระบบฐานข้อมูล POSTGRES 4.2 ของมหาวิทยาลัยแคลิฟอร์เนีย วิทยาเขตเบอร์กลีย์ (UC Berkeley) ภายใต้การควบคุมของ Professor Michael Stonebraker

PostgreSQL เป็น Open Source ที่สามารถรองรับมาตรฐาน SQL:2003 และมีความสามารถที่ทันสมัยต่างๆ เช่น

- Complex Query
- Foreign Key
- Triggers
- Views
- Transactional Integrity
- Multiversion Concurrency Control

2.5.2 PostGIS

PostGIS เป็นส่วนขยายเพิ่มเติมจาก PostgreSQL ซึ่งเป็น Object-Relational Database เพื่อให้สามารถเก็บข้อมูลที่เป็นข้อมูลเชิงพื้นที่ได้ นอกจากนั้น PostGIS ยังสนับสนุนการทำ GiST-based R-Tree Spatial Indexes และ มีฟังก์ชันการทำงานที่ช่วยในการวิเคราะห์ข้อมูลเชิงพื้นที่ได้อีกด้วย ซึ่งข้อมูลเชิงพื้นที่ที่ PostGIS สามารถจัดเก็บได้นั้น ได้แก่ Point, Line, Polygon, Multipoint, Multiline, Multipolygon และ Geometry Collections ซึ่ง Opengeospatial ได้มีการกำหนดรูปแบบขึ้นและเรียกรูปแบบนี้ว่า Well-Known Text (WKT) ซึ่งมีลักษณะดังนี้

- POINT(0 0)
- LINESTRING(0 0,1 1,1 2)
- POLYGON((0 0,4 0,4 4,0 4),(1 1, 2 1, 2 2, 1 2,1 1))
- MULTIPOINT(0 0,1 2)
- MULTILINESTRING((0 0,1 1,1 2),(2 3,3 2,5 4))
- MULTIPOLYGON(((0 0,4 0,4 4,0 4,0 0)),((11,21,22,12,11)),((-1-1,-1-2,-2-2,-2-1,-1-1)))
- GEOMETRYCOLLECTION(POINT(2 3),LINESTRING((2 3,3 4)))

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อให้การเก็บข้อมูลนั้นเป็นไปตามมาตรฐานการเก็บข้อมูลเชิงพื้นที่ที่ Opeengeospatial ได้กำหนดเอาไว้จึงต้องมีตาราง 2 ตารางเพื่อใช้ในการเก็บเมตาเดตา ซึ่งได้แก่ ตาราง SPATIAL_REF_SYS และตาราง GEOMETRY_COLUMNS โดยมีรายละเอียดในแต่ละตาราง ดังนี้

ตาราง SPATIAL_REF_SYS

ใช้สำหรับเก็บ SRID (Spatial Reference Identifier) และข้อความที่ใช้อธิบายรายละเอียดของระบบพิกัดที่ใช้ในการเก็บข้อมูลเชิงพื้นที่ประกอบด้วยคอลัมน์ต่างๆดังต่อไปนี้

PROJ4TEXT : PostGIS ใช้ Proj4 library ในการแปลงระบบพิกัด ซึ่งคอลัมน์นี้จะเก็บคำจำกัดความของแต่ละระบบอ้างอิงเพื่อใช้กับ Proj4 ตัวอย่างเช่น

```
+proj=utm +zone=10 +ellps=clrk66 +datum=NAD27 +units=m
```

SRID : ใช้ในการระบุและแยกแยะ Spatial Referencing System (SRS) ในฐานข้อมูล

AUTH_NAME : เก็บชื่อของมาตรฐานหรือผู้ที่ออกระบบอ้างอิงนั้น ตัวอย่างเช่น "EPSG"

AUTH_SRID : เก็บตัวเลขของระบบอ้างอิงข้อมูลเชิงพื้นที่ซึ่งกำหนดโดยผู้ออกระบบนั้นเช่นกรณีของ EPSG คอลัมน์นี้จะเก็บ EPSG projection code

SRTEXT : เก็บ Well-Known Text (WKT) ที่แทนระบบอ้างอิงนั้น ตัวอย่างเช่น

```
PROJCS["NAD83 / UTM Zone 10N", GEOGCS["NAD83",
DATUM["North_American_Datum_1983",
SPHEROID["GRS 1980",6378137,298.257222101] ],
PRIMEM["Greenwich",0],UNIT["degree",0.0174532925199433] ],
PROJECTION["Transverse_Mercator"],
PARAMETER["latitude_of_origin",0],
PARAMETER["central_meridian",-123],
PARAMETER["scale_factor",0.9996],
PARAMETER["false_casting",500000],
PARAMETER["false_northing",0],
UNIT["metre",1]]
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง GEOMETRY_COLUMNS

ใช้สำหรับเก็บรายละเอียดของ Geometry Column ทั้งหมดในฐานข้อมูล ประกอบด้วย คอลัมน์ต่างๆดังต่อไปนี้

F_TABLE_NAME	: ใช้เก็บรายละเอียดของตารางที่มี Geometry Column นั้นอยู่
F_TABLE_SCHEMA	: ใช้เก็บรายละเอียดของตารางที่มี Geometry Column นั้นอยู่
F_TABLE_CATALOG	: ใช้เก็บรายละเอียดของตารางที่มี Geometry Column นั้นอยู่
F_GEOMETRY_COLUMN	: ใช้เก็บชื่อของ Geometry Column นั้น
COORD_DIMENSION	: ใช้เก็บมิติของข้อมูลเชิงพื้นที่ (2 หรือ 3 มิติ)
SRID	: ใช้เก็บ เลข SRID เพื่อระบุว่าใช้ระบบอ้างอิงพิกัดใด
TYPE	: เพื่อบอกว่าข้อมูลที่เก็บเป็นรูปแบบใดโดยถ้ามีชนิดเดียวก็เลือกใช้ POINT, LINESTRING, POLYGON, MULTIPOINT, MULTILINESTRING, MULTIPOLYGON, GEOMETRYCOLLECTION แต่ถ้าเก็บข้อมูลหลายชนิดรวมกันกำหนดให้ใช้ GEOMETRY เป็นชนิดของข้อมูล

ฟังก์ชันของ PostGIS ที่เกี่ยวข้องกับข้อมูลเชิงพื้นที่

ฟังก์ชันของ PostGIS ที่เกี่ยวข้องกับฐานข้อมูลเชิงพื้นที่ ที่สามารถนำมาใช้ประโยชน์ได้ในโครงการนี้ เช่น

- (1) **AddGeometryColumn(varchar,varchar,varchar,integer,varchar,integer)**
AddGeometryColumn(ชื่อฐานข้อมูล ชื่อตาราง ชื่อคอลัมน์ SRID ชนิดของพื้นที่ มิติของข้อมูล)
- (2) **GeometryFromText(varchar,integer)**
GeometryFromText(text,<srld>
ฟังก์ชันที่ใช้แปลง Well-Known Text เป็น geometry object
- (3) **intersects(geometry,geometry)**
ฟังก์ชันที่ให้ทดสอบการซ้อนทับกันของรูปทรงเรขาคณิต
- (4) **touches(geometry,geometry)**
ฟังก์ชันทดสอบการมีพื้นที่ติดต่อกันแต่ไม่มีการซ้อนทับกัน
- (5) **crosses(geometry,geometry)**
ฟังก์ชันทดสอบการตัดกันของพื้นที่รูปทรงเรขาคณิต
- (6) **within(geometry,geometry)**
ฟังก์ชันทดสอบว่ารูปทรงเรขาคณิตอยู่ภายในรูปทรงเรขาคณิตอื่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(7) contain(geometry, geometry)

ฟังก์ชันทดสอบว่ารูปทรงเรขาคณิตมีรูปทรงเรขาคณิตอื่นอีกอยู่ภายในนั้น

2.6 เทคโนโลยีดอทเน็ต (.Net Technology)

.Net เป็นระบบที่พัฒนาขึ้นมาโดยบริษัท Microsoft เพื่อใช้ในการพัฒนาระบบการให้บริการ Web Service

2.6.1 .Net Framework

.Net Framework เป็นเสมือนโปรแกรมหนึ่งที่จะสามารถสร้างสถานะแวดล้อมหนึ่ง ซึ่งสามารถทำงานในระบบ .Net นี้ได้ มีส่วนประกอบภายในแบ่งออกเป็น 3 ชั้นใหญ่ๆคือ

1. **Programming Language** : เป็นรูปแบบของภาษาที่ออกแบบมาเพื่อให้สามารถทำงานในสถานะที่เป็น .Net ได้โดยที่ทางไมโครซอฟท์ได้เปิดตัวภาษาหลักๆที่จะใช้พัฒนาบน .Net นี้ 3 ภาษา

- C# เป็นภาษาใหม่ที่ไม่โครซอฟท์พัฒนามาจาก C++ กับ JAVA เป็นหลัก
- VB.Net เป็นภาษาที่พัฒนามาจาก Visual Basic ในเวอร์ชัน 6.0
- JScript.Net เป็นภาษาที่พัฒนามาจาก JScript ซึ่งเป็น JavaScript ในเวอร์ชันของไมโครซอฟท์

2. **Base Classes Library** : Library นั้นเปรียบเสมือนชุดคำสั่งสำเร็จรูปย่อยๆที่เพิ่มเข้ามา ซึ่งส่วนใหญ่จะเป็นชุดคำสั่งที่ต้องใช้งานอยู่เป็นประจำ ดังนั้นจึงมีผู้คิดค้นเพื่ออำนวยความสะดวกในการเขียนโปรแกรม ซึ่ง Library ในภาษาต่างๆส่วนใหญ่จะอยู่ในรูปแบบไฟล์ include แต่ถ้าเป็น ASP สิ่งที่เป็น library ก็คือ คอมโพเนนต์ต่างๆนั่นเอง ซึ่งภายในระบบ .Net จะสร้างสิ่งที่เรียกว่าเป็น Library พื้นฐานขึ้น ทำให้ไม่ว่าจะใช้ภาษาใดในการพัฒนาโปรแกรมก็สามารถที่จะเรียกใช้ Library ที่เป็นตัวเดียวกันได้หมด

3. **Common Language Runtime (CLR)** : นับเป็นสิ่งสำคัญแทบจะที่สุดของระบบ .Net นี้ก็ว่าได้ เพราะ CLR ที่ว่านี้มีหน้าที่ทำให้โปรแกรมที่เขียนขึ้นมาด้วยภาษาต่างๆ กัน กลายเป็นภาษารูปแบบมาตรฐานเดียวกัน ทั้งหมด เราเรียกภาษาที่ว่านี้ว่า Intermediate language (IL) ซึ่งเมื่อต้องการที่จะรันโปรแกรมใด CLR ที่ว่านี้จะตรวจสอบเครื่องที่รันว่ามีสถานะแวดล้อมการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เช่นใดหลังจากนั้นก็จะมีคอมไพล์เป็นโปรแกรมที่เหมาะสมต่อการทำงานของเครื่องนั้น ทำให้เราสามารถใช้งานโปรแกรมต่างๆ ได้อย่างมีประสิทธิภาพสูงสุดในแต่ละเครื่อง

ประโยชน์และข้อดีของ .Net Framework

1. เป็นระบบที่มีไลบรารีที่เป็นมาตรฐานเดียวกัน : เนื่องจากมีไลบรารีที่เป็นมาตรฐานเดียวกันทั้งหมดทำให้เราไม่ต้องกังวลว่าภาษาที่ใช้เขียนนั้นมีไลบรารีตัวนั้นตัวนี้หรือไม่ รวมทั้งไม่ต้องคอยกังวลว่าถ้าใช้ไลบรารีของภาษาหนึ่งแล้วอีกภาษาหนึ่งจะไม่มีไลบรารีตัวนั้น
2. ไม่ขึ้นกับระบบปฏิบัติการ (OS) : เนื่องจากระบบปฏิบัติการ ที่แต่ละบุคคลหรือองค์กรใช้นั้นย่อมไม่เหมือนกัน แต่ภายใน .Net Framework จะไม่มีปัญหานี้ของเพียงแค่มีระบบ .Net Framework ก็จะทำให้สามารถใช้งานโปรแกรมต่างๆ ได้ ซึ่งเป็นข้อดีตรงที่เราจะสามารถใช้โปรแกรมต่างๆ ได้ทุกระบบปฏิบัติการ
3. ใช้ในการพัฒนาได้ทุกภาษา : ทำให้เราไม่ต้องคอยมาศึกษาภาษาใหม่ๆ เมื่อต้องการสร้างโปรแกรมในแต่ละครั้ง นอกจากนั้นเรายังสามารถเลือก ใช้ภาษาที่เราถนัดที่สุดในการพัฒนาโปรแกรมต่างๆ ได้ด้วย
4. มีการควบคุมสิ่งแวดล้อมในการทำงานเป็นอย่างดี : เนื่องจากเป็นระบบที่เป็นมาตรฐาน ทำให้การควบคุมจัดสรรระบบต่างๆ ทำได้ง่ายขึ้น ไม่ว่าจะเป็นการจัดสรรหน่วยความจำ ด้านการใช้งานเครื่องก็มีความรวดเร็วมากขึ้น ลดโอกาสที่เครื่องจะแองก์ได้เป็นอย่างดี
5. ความปลอดภัยที่มีมากขึ้น : .Net Framework สามารถกำหนดสิทธิ์การใช้งานหรือ permission ของผู้ใช้งาน ได้มากขึ้นทำให้สามารถกำหนดว่า จะให้โปรแกรมในส่วนใดใช้งานได้หรือไม่ได้ แล้วแต่เฉพาะบุคคล

2.6.2 ASP.Net

ภาษา ASP และ ASP.Net ย่อมาจาก Active Server Page เป็นโปรแกรมประเภท Server-Side Script (โปรแกรมที่ทำงานบนเครื่อง Server) ที่นิยมใช้อย่างแพร่หลาย สร้างโดย บริษัท ไมโครซอฟท์ จำกัด บนระบบปฏิบัติการ ไมโครซอฟท์ วินโดวส์ ซึ่งใช้ร่วมกับโปรแกรม Internet Information Service หรือ IIS โดยเว็บเพจที่ใช้ ASP หรือ ASP.Net เขียน จะระบุเป็นตระกูลไฟล์ที่ลงท้ายด้วย .asp และ .aspx ซึ่งภาษา ASP ยังสามารถใช้เขียนโปรแกรมเพื่อควบคุมฐานข้อมูลต่างๆ ผ่าน ODBC, ADO, DAO JET และอื่นๆ อีกมากมาย

ASP.Net เป็นภาษา script ที่พัฒนาต่อมาจาก ASP ตัวเก่า เพื่อให้โปรแกรมเมอร์สามารถพัฒนา web application ให้ใช้ .Net ได้สะดวกขึ้น ASP.Net นี้ถึงแม้จะอ้างอิงมาจาก ASP ตัวเก่า แต่ก็มี syntax หลายส่วนที่เปลี่ยนแปลงไป คือ

1. ใช้ภาษาใดๆ ในการเขียนสคริปต์ก็ได้ : จากเดิมที่เราสามารถใช้ได้เฉพาะภาษาที่เป็นสคริปต์ของ VBScript และ JScript แต่ใน ASP.Net เราสามารถใช้ภาษาที่มีรูปแบบของภาษาเต็มๆ ซึ่ง ในเบื้องต้น มี 3 ภาษาคือ C#, VB.Net และ JScript.Net ที่ออกมาเป็นมาตรฐาน แต่ในอนาคตไมโครซอฟท์มีแผนที่จะเพิ่มตัวแปลภาษาให้ครบทุกภาษา

2. มีความยืดหยุ่นในการเขียนโปรแกรมมากขึ้น : โดยที่เราสามารถใช้ภาษาในการเขียน ASP.Net ได้มากกว่า 1 ภาษาภายในไฟล์เดียวกัน ทำให้สามารถเลือกรูปแบบของภาษาที่ง่ายที่สุดต่อการเขียน ในแต่ละส่วนได้

3. ลักษณะการแปลภาษาและนามสกุลไฟล์เปลี่ยนไป : ใน ASP เวอร์ชันก่อนๆ มีลักษณะการแปลภาษาเป็นแบบอินเตอร์พรีเตอร์ (Interpreter) คือการจะทำคำสั่งใดค้อยแปลคำสั่งนั้น แต่ในเวอร์ชัน .Net นี้จะมี ลักษณะเป็นคอมไพเลอร์ (Compiler) คือการแปลคำสั่งรวมทั้ง โปรแกรม นอกจากนี้นามสกุลของไฟล์ก็มีการเปลี่ยนแปลงจากเดิมที่ใช้นามสกุลไฟล์เป็น " *.asp " เป็น " *.aspx "

4. รูปแบบและการใช้งานคอมโพเนนต์ที่ง่ายขึ้น : รูปแบบของคอมโพเนนต์จะเน้นไปที่ XML มากที่สุด และที่สำคัญคือการใช้งานคอมโพเนนต์ใน ASP.Net นั้นเราสามารถอัปโหลดไฟล์ไปไว้ในไคลเอนต์ที่ผู้ดูแลเซิร์ฟเวอร์ (Admin) กำหนดหลังจากนั้นคอมโพเนนต์จะติดตั้งตัวเองโดยอัตโนมัติ ลดปัญหาที่เกิดจาก ASP เวอร์ชันก่อนๆ ได้เป็นอย่างดี เนื่องจากใน ASP เวอร์ชันก่อนนั้น การติดตั้งคอมโพเนนต์กระทำได้เพียงผู้ดูแลเซิร์ฟเวอร์เพียงคนเดียวเท่านั้น ทำให้เวลาต้องการใช้คอมโพเนนต์ต่างๆที่เซิร์ฟเวอร์ไม่มี จึงเป็นเรื่องที่ลำบาก

5. มีไลบรารีให้เลือกใช้ได้มากขึ้น : ใน ASP เวอร์ชันก่อนๆนั้นแอปพลิเคชันบางอย่างสร้างได้ไม่สะดวกนัก ต้องอาศัยคอมโพเนนต์ต่างๆมากมาย แต่ใน ASP.Net นั้นได้เพิ่มไลบรารีในส่วนเหล่านี้ให้กลายเป็นพื้นฐานของการใช้งาน

6. มีคอนโทรลทำให้การใช้งานในบางสิ่งง่ายขึ้น : เป็นส่วนพิเศษที่เพิ่มเติมมาจาก ASP รุ่นก่อนๆ ที่ไม่มีส่วนที่เรียกว่า คอนโทรล ซึ่งคอนโทรลนี้จะช่วยให้เราสามารถสร้างเว็บไซต์ได้อย่างง่ายและมีประสิทธิภาพมากขึ้น จึงไม่ต้องกังวลว่าบราวเซอร์รุ่นนั้นรุ่นนี้จะรองรับกับภาษาที่เราเขียนหรือไม่

7. สามารถเรียกขอข้อมูลจากเซิร์ฟเวอร์ได้ : ใน ASP เวอร์ชันก่อนๆ เซิร์ฟเวอร์สามารถเรียกขอข้อมูลได้จากเครื่องผู้ใช้เท่านั้นแต่ใน ASP.Net เครื่องเซิร์ฟเวอร์สามารถเรียกขอข้อมูลจากเครื่องเซิร์ฟเวอร์ด้วยกันได้

8. ไม่ต้องต่อ Hardware : เนื่องจากเป็นระบบใน .Net Framework ดังนั้นจึงมีคุณสมบัติของ Common Language Runtime (CLR) ทำให้มีการคอมไพล์โปรแกรมเป็นภาษามาตรฐานที่เรียกว่า IL ก่อน ดังนั้นไม่ว่าคุณจะเล่นเครื่องปาล์มหรือ โน้ตบุ๊ก PDA ก็ไม่เกิดปัญหา

9. ง่ายต่อการหาจุดผิดพลาดในการเขียนโปรแกรม : หากเป็น ASP รุ่นก่อนเวลาเกิดความผิดพลาด (error) เครื่องจะบอกแค่ว่าเป็นความผิดพลาดชนิดใดบรรทัดไหน แต่ใน ASP.Net นี้การแจ้งเตือนจะเป็นเอกสารที่แสดงรายละเอียดของข้อผิดพลาด

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องจะแสดงรายละเอียดที่มากขึ้น พร้อมแนวทางแก้ไข

10. มีการตรวจสอบเหตุการณ์ต่างๆ ได้ภายในเว็บเพจ : มีการตรวจสอบเหตุการณ์ต่างๆ ตั้งแต่โหลดหน้าเว็บเพจไปจนถึงปิดหน้าเว็บเพจลง ทำให้เราสามารถเขียนโปรแกรมกำหนดเหตุการณ์ต่างๆ ได้ง่ายขึ้น

11. แยกส่วนที่เป็น HTML กับ ASP ออกมาอย่างชัดเจน : ในเวอร์ชันก่อนๆ ส่วนที่เป็น HTML กับ ASP จะเขียนปนกันไปมา แต่ในเวอร์ชันนี้จะแยกส่วนกันอย่างชัดเจนว่าส่วนไหนเป็น HTML และส่วนไหนเป็น ASP

2.6.3 การใช้ Visual Studio.Net สร้างเว็บเซอร์วิส

ใน Visual Studio.Net Framework เว็บเซอร์วิสคือคลาสที่สืบทอดมาจากคลาส System.Web.Services.WebService สามารถสร้าง public method เป็น Web Method แล้วตัว ASP.Net runtime จะทำการ publish และ invoke method เหล่านี้เอง ตัวอย่างเช่น

```
[WebService (Namespace= "http://you.org/")]
```

```
Public class
```

```
yourService : System.Web.Services.WebService {
```

```
    [WebMethod]
```

```
    Public string echo(string name) {
```

```
        Return name;}
    }
```

ทาง client อ้างอิง method ได้ดังนี้เช่น

```
yourService ys = new yourService();
```

```
string ans = ys.echo("Hello World");
```

พารามิเตอร์ทั้งหมดจะถูกจัดลงใน XML SOAP message framework จะเตรียมการจัดการกับ data type มาตรฐานอย่าง (number , string , date , array , list , . . .) แต่สำหรับสิ่งที่มีโครงสร้างที่ซับซ้อนอย่าง dataset นั้นยังไม่ได้เป็นมาตรฐานใน W3C ในปัจจุบัน

การสร้างเว็บเซอร์วิสโดยใช้ Visual Studio .Net นั้นสามารถทำได้ง่าย โดยสร้างโปรเจกต์ใหม่ แล้วเลือก Web Service Project แล้วมันจะทำการสร้าง class definition ให้เรา ซึ่งสืบทอดมาจากคลาส System.Web.Services.WebService

Visual Studio .Net จะสร้างโปรเจกต์ซึ่งบรรจุด้วยกลุ่มของซอร์สไฟล์ มี 2 ไฟล์ที่ทำให้เว็บเซอร์วิสกับแอปพลิเคชันอื่นของ .Net แยกต่างหากคือ ไฟล์ ".asmx" และ ".asmx.cs" สำหรับไฟล์ ".asmx.cs" จะบรรจุ Class definition ที่เป็นชื่อเว็บเซอร์วิสของเราและสืบทอดมาจากคลาส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

System.Web.Services ส่วนไฟล์ “.asmx” เป็นไฟล์ที่เรียกใช้งานง่ายที่ถูกใช้โดย ASP.Net runtime ในการระบุที่ตั้งของเว็บเซอร์วิสเมื่อมีการอ้างถึงโดย SOAP request จากฝั่ง client

สรุปแล้วบนแพลตฟอร์มของ Microsoft .Net เว็บเซอร์วิสคือ คลาสที่ประกอบด้วย public method ที่ประกาศอยู่ภายใต้แอตทริบิวต์ [WebMethod] แอตทริบิวต์นี้จะซ่อนรายละเอียดทั้งหมดของการสร้าง proxy และ stub , การ binding กับเว็บเซอร์วิส , และการสร้าง metadata ที่ client ต้องการในการอ้างถึงเว็บเซอร์วิส ถ้าต้องการให้ทำงานร่วมกันได้ระหว่างแพลตฟอร์มทั้งหมด ผู้เขียนเว็บเซอร์วิสจะต้องทำตามมาตรฐานของ W3C ในเรื่อง datatype และ โครงสร้างของ SOAP เมจเซจในการส่งพารามิเตอร์และคืนค่ากลับ

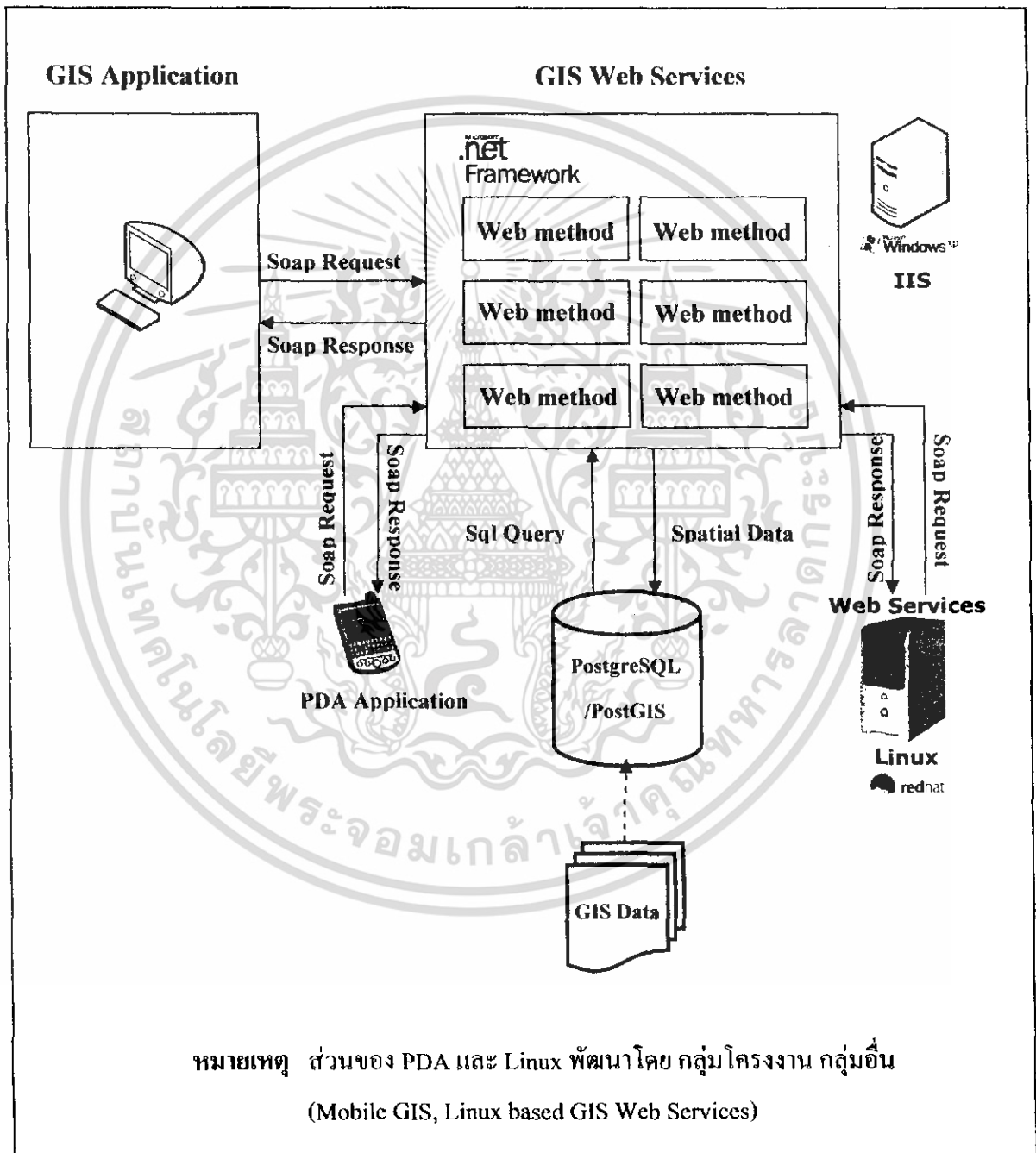


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบโครงงาน

3.1 โครงสร้างของโครงงาน



รูปที่ 3.1 โครงสร้างของโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงการนี้แบ่งออกเป็น 2 ส่วนหลัก คือ

3.1.1 GIS Web Services

ให้บริการเกี่ยวกับข้อมูลเชิงพื้นที่ และรายละเอียดต่างๆ ของสถานที่ต่างๆ และแผนที่ในรูปแบบ SVG ซึ่งสภาพแวดล้อมในการพัฒนาใช้ .net Framework เครื่องมือที่ใช้ในการพัฒนา คือ Microsoft Visual Studio .Net 2005 ภาษาที่ใช้ในการพัฒนาได้แก่ C# และใช้ DBMS สำหรับจัดการกับข้อมูลเชิงพื้นที่ คือ PostgreSQL และ PostGIS โดยทั้งหมดติดตั้งอยู่บนระบบปฏิบัติการ Microsoft Windows XP

ระบบมีการทำงานร่วมกันกับ map server ซึ่งให้บริการผ่านเว็บเซิร์ฟเวอร์ที่อยู่บน platform Linux โดยมีการร้องขอข้อมูลเชิงพื้นที่ที่ไม่มีอยู่ใน map server ของตนเอง ได้แก่ ข้อมูลในส่วนของประเทศสหรัฐอเมริกา (USA) 3 เลเยอร์ ดังนี้

1. รัฐ (states)
2. ถนน (roads)
3. สวนสาธารณะ (parks)

3.1.2 GIS Application

เป็นส่วนที่เรียกใช้บริการต่างๆ จาก map server โดยข้อมูลรายละเอียดต่างๆ ของสถานที่ และข้อมูลที่ถูกเก็บอยู่ในรูปของ SVG จะถูกนำมาแสดงผลเป็นภาพ อีกทั้งยังมีบริการต่างๆ คอยอำนวยความสะดวก เช่น สามารถเลือกแสดงผลตามเลเยอร์ที่ต้องการ ค้นหาสถานที่ตามชื่อหรือที่อยู่ หาสถานที่ที่เป็นรัศมีจากจุดอ้างอิง หาสถานที่ที่ใกล้กับจุดอ้างอิงมากที่สุด หาพิกัดของสถานที่ที่ต้องการ และแปลงที่อยู่เป็นค่าพิกัด เป็นต้น

3.2 ฐานข้อมูล

ฐานข้อมูลของโครงการนี้ เป็นฐานข้อมูลที่เก็บข้อมูลเชิงพื้นที่ ซึ่งแบ่งข้อมูลออกเป็น 2 ส่วน คือ

ข้อมูลส่วนของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (KMITL) มีทั้งหมด 9 เลเยอร์ ดังนี้

1. คณะวิศวกรรมศาสตร์ (buildineng)
2. คณะวิทยาศาสตร์ (buildingsci)
3. คณะเทคโนโลยีการเกษตร (buildingagriind)
4. สิ่งก่อสร้างของสถาบัน (institute)

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (KMITL) ไม่อนุญาติให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ถนน (road)
7. รางรถไฟ (rail)
8. เอทีเอ็ม (atm)
9. แม่น้ำ (river)

รายละเอียดของเลขอร์ต่าง ๆ จะถูกเก็บตาราง Geometry_Columns ซึ่งจะบอกว่าตารางที่เก็บข้อมูลของแต่ละเลขอร์มีชื่อว่าอะไร มี Geometry column ชื่ออะไร ใช้ระบบพิกัดแบบใด และมีชนิดเป็นอะไร (POINT, MULTIPOLYGON, MULTILINESTRING etc.)

	oid	table_catalog	table_schema	table_name	geometry_column_name	coord_dimension	srid	type
		varchar	varchar	varchar	varchar	int4	int4	varchar
1	27761	public	public	atm	the_geom	2	-1	POINT
2	27762	public	public	building	the_geom	2	-1	MULTIPOLYGON
3	27763	public	public	buildingagrind	the_geom	2	-1	MULTIPOLYGON
4	27764	public	public	buildingeng	the_geom	2	-1	MULTIPOLYGON
5	27765	public	public	buildingsci	the_geom	2	-1	MULTIPOLYGON
6	28057	public	public	institute	the_geom	2	-1	MULTIPOLYGON
7	27766	public	public	rail	the_geom	2	-1	MULTILINESTRING
8	27767	public	public	road	the_geom	2	-1	MULTILINESTRING
9	28183	public	public	water	the_geom	2	-1	MULTIPOLYGON

รูปที่ 3.2 ตาราง Geometry_Columns ของ KMITL

ข้อมูลส่วนของประเทศสหรัฐอเมริกา (USA) มี 3 เลขอร์ ดังนี้

1. เมือง (cities)
2. โรงพยาบาล (hospitals)
3. สนามบิน (airports)

	oid	table_catalog	table_schema	table_name	geometry_column_name	coord_dimension	srid	type
		varchar	varchar	varchar	varchar	int4	int4	varchar
1	46102	public	public	tb_airport	the_geom	2	-1	POINT
2	34687	public	public	tb_cities	the_geom	2	-1	POINT
3	49586	public	public	tb_hospital	the_geom	2	-1	POINT

รูปที่ 3.3 ตาราง Geometry_Columns ของ USA

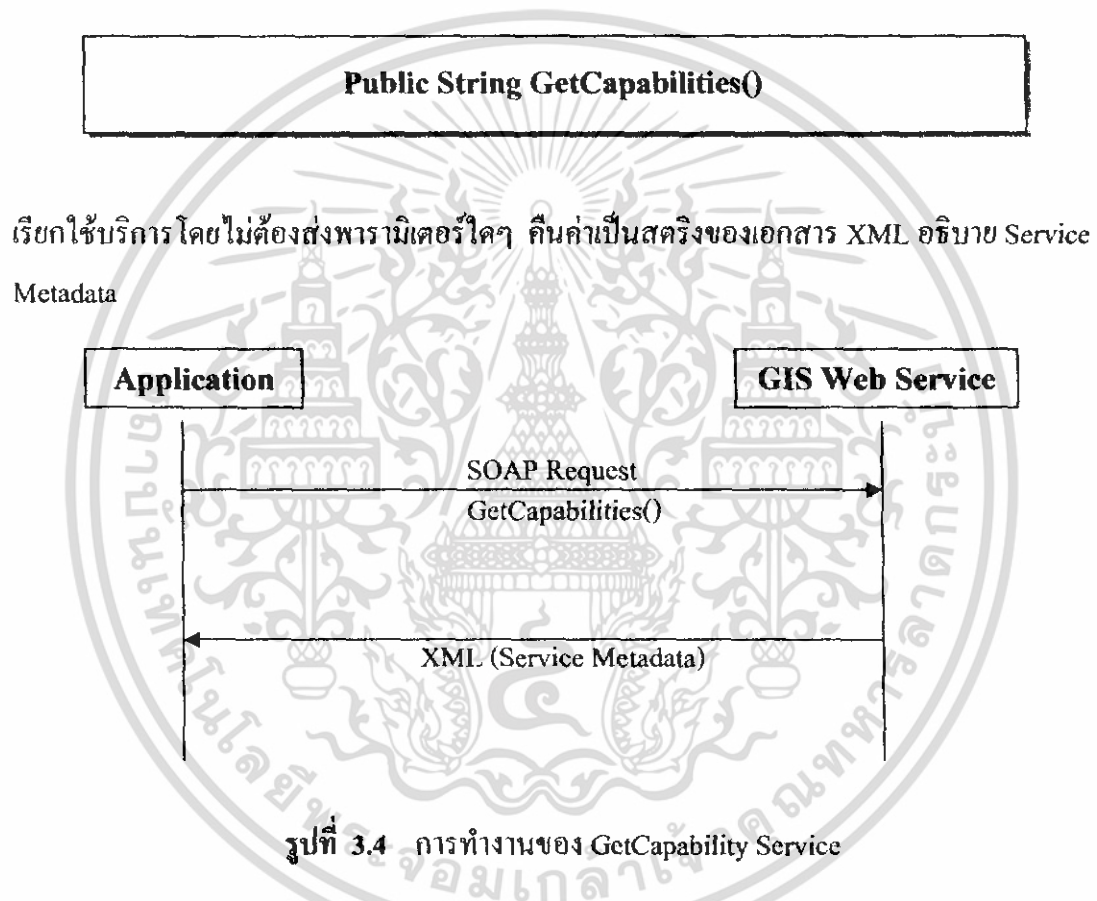
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การออกแบบ GIS Web Services

ส่วนของบริการเชิงพื้นที่สำหรับระบบสารสนเทศเชิงภูมิศาสตร์ ซึ่งจะมีบริการต่าง ๆ ดังนี้

3.3.1 GetCapabilities

บริการข้อมูลซึ่งจะบอกว่า Server เรามีบริการและข้อมูลเชิงพื้นที่อะไรบ้าง (Service metadata) ในรูปแบบของเอกสาร XML



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 GetMap

บริการแสดงแผนที่ในรูปแบบที่สามารถนำมาแสดงในรูปแบบ SVG

Public XmlDocument GetMap(Parameters)

รับพารามิเตอร์ เป็นเอกสาร XML ซึ่งจะระบุรูปแบบของแผนที่ที่ต้องการให้แสดงผล เช่น เลเยอร์ที่ต้องการให้แสดงผล สีเส้นขอบและสีเติม แสดงข้อความบนแผนที่ ขนาดตัวอักษร ลักษณะตัวอักษร ความเข้มของแผนที่ ขนาดของแผนที่ ขนาดของ view box จุดเริ่มต้นของ view box เป็นต้น



รูปที่ 3.5 การทำงานของ Getmap Service

ชนิด	พารามิเตอร์	คำอธิบาย
String	getMapRequest XML	เอกสาร XML ที่ระบุเงื่อนไข รูปแบบของแผนที่ที่ต้องการ

ตารางที่ 3.1 อธิบายพารามิเตอร์ของ Getmap Service

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะของเอกสาร getMapRequest XML

```

<?xml version="1.0" encoding="windows-874"?>
<xs:schema attributeFormDefault="unqualified"
elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="GetMap">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="viewbox">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="vbox1" type="xs:decimal" />
              <xs:element name="vbox2" type="xs:decimal" />
              <xs:element name="vbox3" type="xs:decimal" />
              <xs:element name="vbox4" type="xs:decimal" />
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="width" type="xs:unsignedShort" />
        <xs:element name="height" type="xs:unsignedShort" />
        <xs:element name="opacity" type="xs:unsignedByte" />
        <xs:element name="text">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="text-visible" type="xs:bool" />
              <xs:element name="text-family" type="xs:string" />
              <xs:element name="text-weight" type="xs:string" />
              <xs:element name="text-style" type="xs:string" />
              <xs:element name="text-size" type="xs:unsignedByte" />
              <xs:element name="text-color" type="xs:string" />
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="mouse-event" type="xs:bool" />
        <xs:element maxOccurs="unbounded" name="layer">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="name" type="xs:string" />
              <xs:element name="fill-color" type="xs:string" />
              <xs:element name="stroke-color" type="xs:string" />
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

getMapRequest XML จะต้อง มี root element ที่ชื่อว่า GetMap และความหมายของแต่ละแท็กดังนี้

- vbox1 คือ พิกัดเริ่มต้นของ view box ตามแนวแกน X
- vbox2 คือ พิกัดเริ่มต้นของ view box ตามแนวแกน Y
- vbox3 คือ ขนาดความกว้างของ view box ในระบบพิกัด
- vbox4 คือ ขนาดความสูงของ view box ในระบบพิกัด

เอกสารนี้เป็นเอกสารของกรมแผนที่ทหารบก อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

width คือ ขนาดความกว้างของเอกสาร SVG ที่จะให้แสดง

height คือ ขนาดความสูงของเอกสาร SVG ที่จะให้แสดง

opacity คือ ความเข้มสีที่จะให้แสดง

text-visible คือ ต้องการให้แสดงข้อความบนรูป SVG หรือไม่ มีค่าเป็น True หรือ False

text-family คือ ชนิดของตัวอักษร

text-style คือ ตัวอักษรเป็นตัวเอียงหรือตัวปกติ มีค่าเป็น italic หรือ normal

text-width คือ รูปแบบตัวอักษรเป็นตัวหนาหรือปกติ มีค่าเป็น bold หรือ normal

text-color คือ สีตัวอักษร

text-size คือ ขนาดตัวอักษร

mouse-event คือ ต้องการให้มีเมาส์อีเวนต์ คือ เมื่อเอาเมาส์ไปวางบน SVG แล้วให้แสดงรายละเอียดหรือไม่ มีค่าเป็น True หรือ False

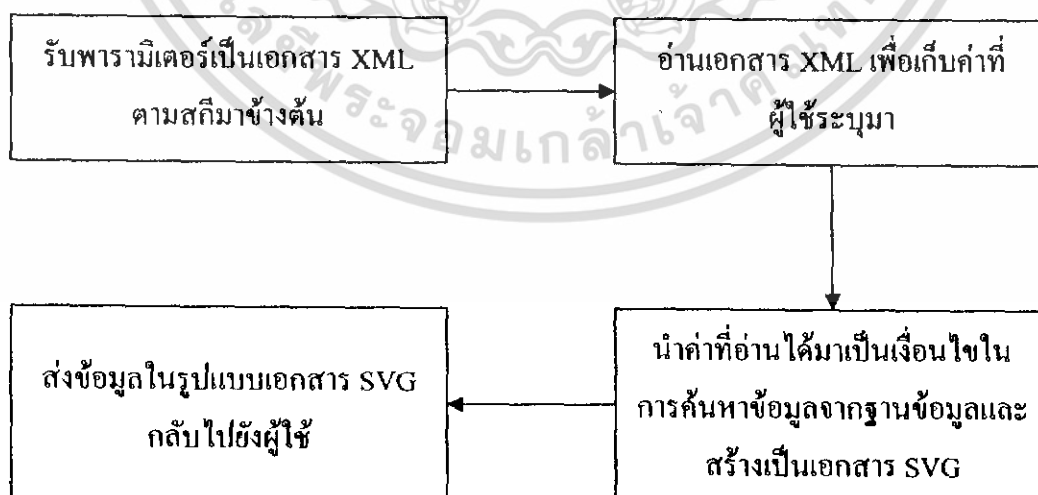
name คือ ชื่อเลขอร์ที่ต้องการให้แสดงผล

fill-color คือ สีที่ต้องการเติมลงในเลขอร์นั้น

stroke-color คือ สีขอบที่ต้องการเติมลงในเลขอร์นั้น

เมื่อผู้ใช้เรียกใช้บริการ โดยส่งพารามิเตอร์เป็นเอกสาร XML ตามสกีมาดังกล่าวมายังเว็บเซอร์วิส เว็บเซอร์วิสจะเข้าไปอ่านเอกสาร XML แล้วนำค่าที่ได้ไปเป็นเงื่อนไขในการค้นหาข้อมูลจากฐานข้อมูลเพื่อสร้างเอกสาร SVG ตามที่ผู้ใช้ระบุ แล้วคืนค่ากลับไปยังผู้ใช้เป็นเอกสาร SVG

สามารถสรุปขั้นตอนการทำงานได้ดังนี้



รูปที่ 3.6 แสดงขั้นตอนการให้บริการ GetMap

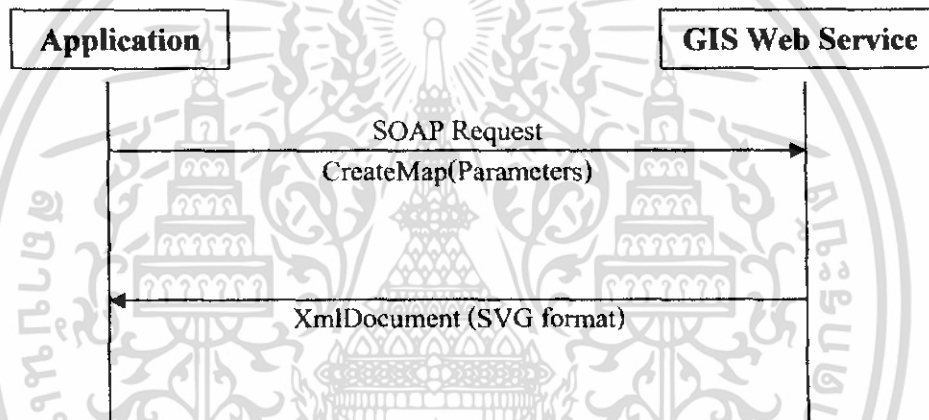
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.3 CreateMap

บริการ CreateMap เป็นบริการที่ผู้ใช้สามารถสร้างแผนที่โดยผู้ใช้ระบุพิกัดของสถานที่ที่ต้องการสร้างในรูปแบบ SVG โดยสามารถเลือกเลขอร์ของข้อมูลที่มีอยู่ในเซอร์เวอร์ที่ต้องการให้แสดงในแผนที่นั้นได้ด้วย

CreateMap(Parameters)

รับพารามิเตอร์ เป็นเอกสาร XML ซึ่งจะระบุรูปแบบของแผนที่ที่ต้องการให้แสดงผล โดยมีเท็กของสถานที่ใหม่ที่จะสร้างเพิ่มเข้าไปด้วย



รูปที่ 3.7 การทำงานของ CreateMap Service

ชนิด	พารามิเตอร์	คำอธิบาย
String	createMapRequestXML	เอกสาร XML ที่ระบุรูปแบบของแผนที่ที่ต้องการให้แสดงผล และรายละเอียดของสิ่งที่ต้องการสร้างเพิ่มในแผนที่

ตารางที่ 3.2 อธิบายพารามิเตอร์ของ CreateMap Service

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบของเอกสาร createMapRequestXML

XML Schema

```

<?xml version="1.0" encoding="windows-874"?>
<xs:schema attributeFormDefault="unqualified"
elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="CreateMap">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="viewbox">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="vbox1" type="xs:decimal" />
              <xs:element name="vbox2" type="xs:decimal" />
              <xs:element name="vbox3" type="xs:decimal" />
              <xs:element name="vbox4" type="xs:decimal" />
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="width" type="xs:unsignedShort" />
        <xs:element name="height" type="xs:unsignedShort" />
        <xs:element name="opacity" type="xs:unsignedByte" />
        <xs:element name="text">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="text-visible" type="xs:bool" />
              <xs:element name="text-family" type="xs:string" />
              <xs:element name="text-weight" type="xs:string" />
              <xs:element name="text-style" type="xs:string" />
              <xs:element name="text-size" type="xs:unsignedByte" />
              <xs:element name="text-color" type="xs:string" />
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="mouse-event" type="xs:bool" />
        <xs:element maxOccurs="unbounded" name="layer">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="name" type="xs:string" />
              <xs:element name="fill-color" type="xs:string" />
              <xs:element name="stroke-color" type="xs:string" />
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="item">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="item-coordinate" type="xs:string" />
              <xs:element name="item-name" type="xs:string" />
              <xs:element name="item-reference" type="xs:string" />
              <xs:element name="item-detail" type="xs:string" />
              <xs:element name="item-fill" type="xs:string" />
              <xs:element name="item-stroke" type="xs:string" />
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

เอกสารนี้เป็นสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่าจะด้วยวิธีใดก็ตามมิให้ตีพิมพ์เปลี่ยนแปลงเนื้อหา และห้องข้อมูลอิเล็กทรอนิกส์ของเอกสารทุกครั้งที่มีการนำไปใช้

`createMapRequest XML` จะต้องมี `root element` ที่ชื่อว่า `GetMap` และความหมายของแต่ละแท็กดังนี้

`vbox1` คือ พิกัดเริ่มต้นของ `view box` ตามแนวแกน X

`vbox2` คือ พิกัดเริ่มต้นของ `view box` ตามแนวแกน Y

`vbox3` คือ ขนาดความกว้างของ `view box` ในระบบพิกัด

`vbox4` คือ ขนาดความสูงของ `view box` ในระบบพิกัด

`width` คือ ขนาดความกว้างของเอกสาร SVG ที่จะให้แสดง

`height` คือ ขนาดความสูงของเอกสาร SVG ที่จะให้แสดง

`opacity` คือ ความเข้มสีที่จะให้แสดง

`text-visible` คือ ต้องการให้แสดงข้อความบนรูป SVG หรือไม่ มีค่าเป็น `True` หรือ `False`

`text-family` คือ ชนิดของตัวอักษร

`text-style` คือ ตัวอักษรเป็นตัวเอียงหรือตัวปกติ มีค่าเป็น `italic` หรือ `normal`

`text-width` คือ รูปแบบตัวอักษรเป็นตัวหนาหรือปกติ มีค่าเป็น `bold` หรือ `normal`

`text-color` คือ สีตัวอักษร

`text-size` คือ ขนาดตัวอักษร

`mouse-event` คือ ต้องการให้มีเมาส์เอนต์ คือ เมื่อเอาเมาส์ไปวางบน SVG แล้วให้แสดง

รายละเอียดหรือไม่ มีค่าเป็น `True` หรือ `False`

`name` คือ ชื่อเลขอร์ที่ต้องการให้แสดงผล

`fill-color` คือ สีที่ต้องการเติมลงในเลขอร์นั้น

`stroke-color` คือ สีขอบที่ต้องการเติมลงในเลขอร์นั้น

`item-coordinate` คือ ค่าพิกัดของสถานที่ที่ต้องการสร้างเพิ่ม ซึ่งอาจจะเป็นชนิด `polygon` ,

`point` หรือ `line`

`item-name` คือ ชื่อของสถานที่ที่ต้องการให้แสดงในแผนที่

`item-reference` คือ ลิงก์ไปยังเว็บไซต์หรือไฟล์

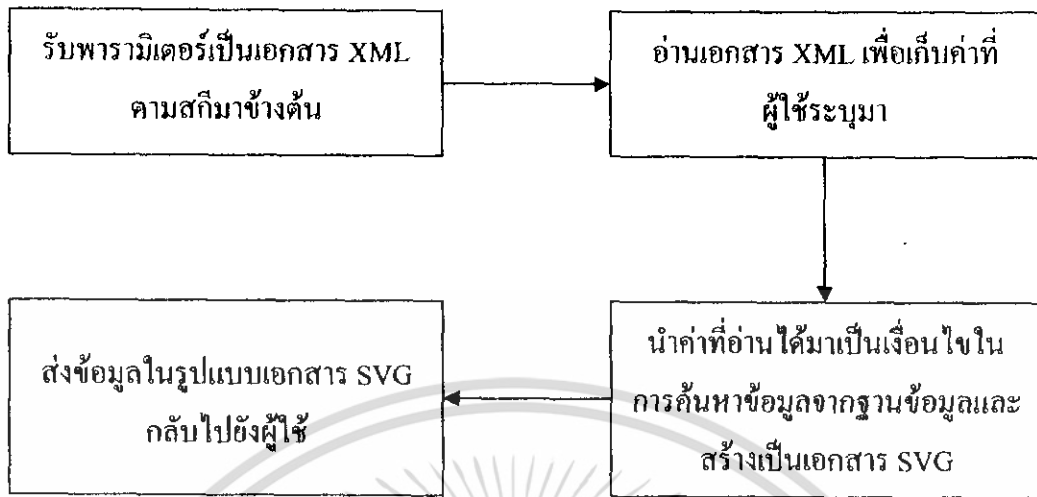
`item-detail` คือ รายละเอียดของสถานที่ที่ต้องการให้สร้าง

`item-fill` คือ สีที่เติมของสถานที่ที่ต้องการสร้าง

`item-stroke` คือ สีเส้นขอบของสถานที่ที่ต้องการสร้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถสรุปขั้นตอนการทำงาน ได้ดังนี้



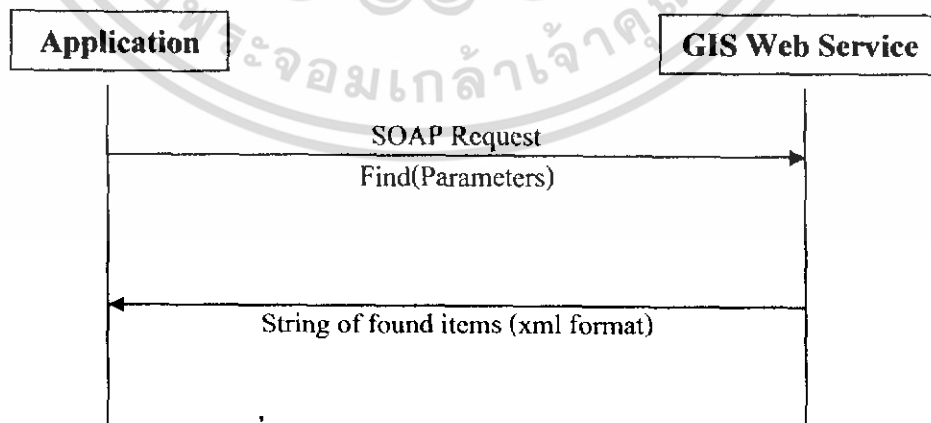
รูปที่ 3.8 แสดงขั้นตอนการให้บริการ CreateMap

3.3.4 Find

บริการค้นหาสถานที่ใน map server

Public string Find (Parameters)

ค้นหาโคออร์ดิเนตที่ต้องการทำการค้นหา และชื่อสถานที่ โดยจะคืนค่าเป็นเอกสาร XML. บอกรายละเอียดของสถานที่ที่ค้นหาพบ



รูปที่ 3.9 การทำงานของ Find Service

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชนิด	พารามิเตอร์	คำอธิบาย
String	Name	ระบุสถานชื่อสถานที่ที่ต้องการค้นหา
String[]	Layer	ระบุเลขที่ที่ต้องการค้นหา

ตารางที่ 3.3 อธิบายพารามิเตอร์ของ Find Service

รูปแบบของเอกสาร XML ที่คืนค่าในเซอร์วิส Find

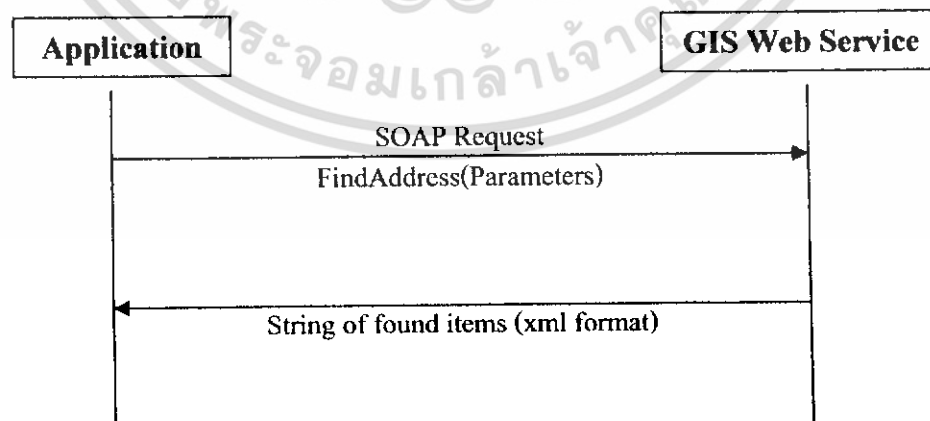
```
<?xml version="1.0" encoding="utf-8"?>
<findResult>
  <foundItem>
    <item>
      <name>.....</name>
      <coordinateX>.....</coordinateX>
      <coordinateY>.....</coordinateY>
    </item>
  </foundItem>
</findResult>
```

3.3.5 FindAddress

บริการค้นหาสถานที่ใน map server ซึ่งสามารถระบุเป็นที่อยู่ได้ (addressable) เช่น ระบุที่อยู่, ถนน, อำเภอ, จังหวัด เป็นต้น

Public String FindAddress (Parameters)

รับพารามิเตอร์ ซึ่งจะระบุสถานที่ที่ต้องการค้นหาเป็นที่อยู่ได้ เช่น เลขที่ ชื่อถนน ตำบล อำเภอ จังหวัด รหัสไปรษณีย์ เป็นต้น



รูปที่ 3.10 การทำงานของ FindAddress Service

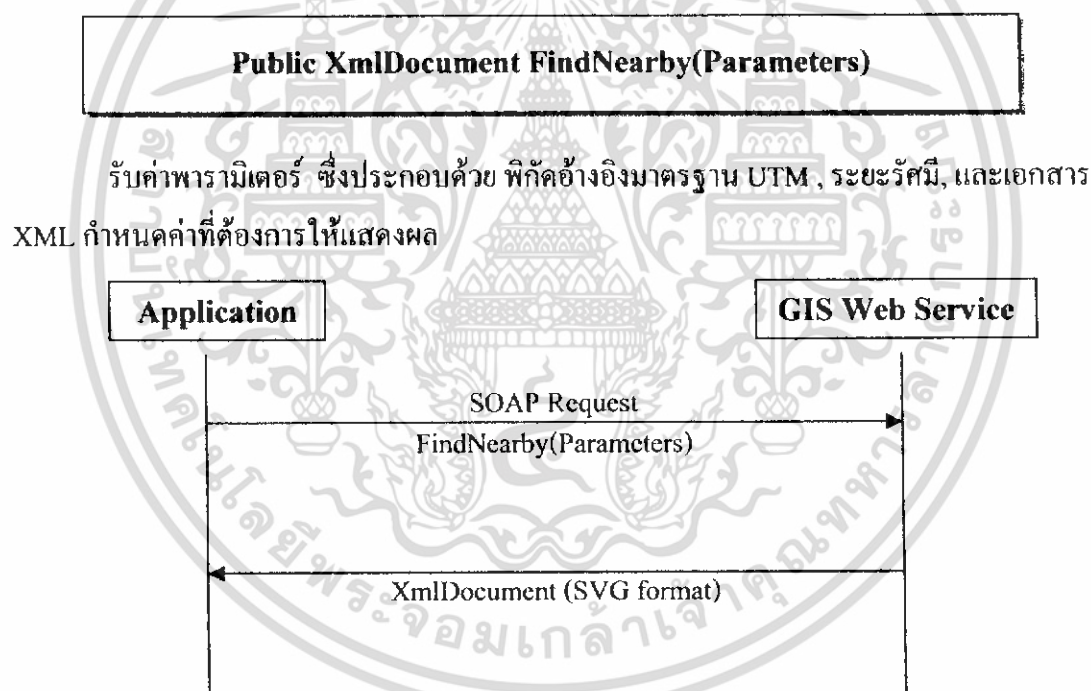
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชนิด	พารามิเตอร์	คำอธิบาย
String	address	บ้านเลขที่
String	road	ชื่อถนน
String	district	ชื่ออำเภอ
String	province	ชื่อจังหวัด
String	postcode	รหัสไปรษณีย์

ตารางที่ 3.4 อธิบายพารามิเตอร์ของ FindAddress Service

3.3.6 FindNearby

บริการค้นหาสถานที่ในละแวกใกล้เคียงกับจุดอ้างอิง ผู้ใช้สามารถค้นหาสถานที่ที่ต้องการแสดงในเลเยอร์ต่างๆ ที่อยู่ภายในรัศมีที่กำหนด



รูปที่ 3.11 การทำงานของ FindNearby Service

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

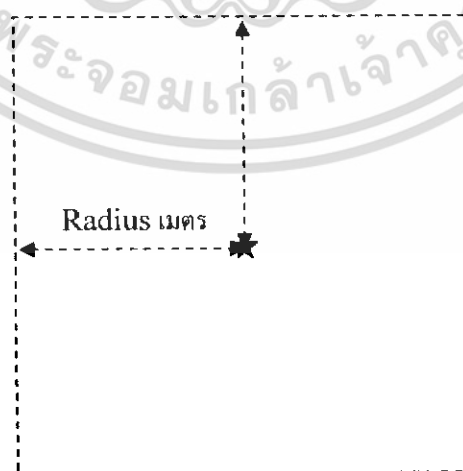
ชนิด	พารามิเตอร์	คำอธิบาย
double	coordinateX	พิกัดอ้างอิงมาตรฐาน UTM
double	coordinateY	พิกัดอ้างอิงมาตรฐาน UTM
double	radius	รัศมีที่ต้องการค้นหาหน่วยเป็น เมตร
string	getMapRequest XML	เอกสาร XML กำหนดค่าต่างๆของแผนที่ที่ต้องการให้แสดงผลเหมือนกับของเซอร์วิส GetMap

ตารางที่ 3.5 อธิบายพารามิเตอร์ของ FindNearby Service

ในการเรียกใช้บริการผู้ใช้จะต้องส่งพารามิเตอร์ข้างต้น จากนั้นระบบจะไปทำการค้นหาข้อมูลในฐานข้อมูลใช้ภาษา SQL โดยมีคำสั่งค้นหาสิ่งที่อยู่ภายในรัศมีจากจุดอ้างอิงดังนี้

```
SELECT Geometry Column FROM Table WHERE Geometry Column &&
Expand(GeometryFromText('POINT(coordinateX coordinatY)',-1),radius) AND
Distance(GeometryFromText('POINT(coordinateX coordinateY)',-1),Geometry Column)
< radius;
```

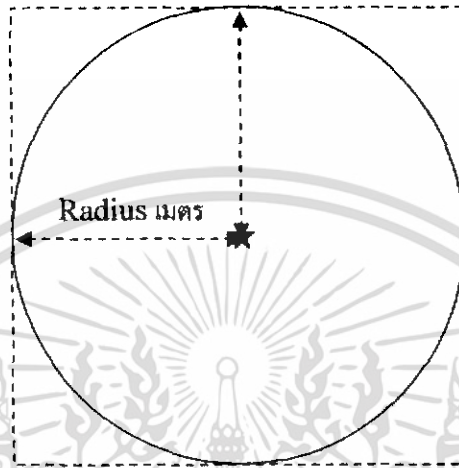
ใช้ฟังก์ชัน Expand(geometry,bound) ซึ่งเป็นฟังก์ชันของ PostGIS เพื่อขยายพื้นที่โดยใส่อาร์กิวเมนต์เป็นตำแหน่งที่ระบุในมาตรฐาน UTM และรัศมีที่ต้องการ จะได้ผลลัพธ์เป็นการขยายขอบเขตพื้นที่ออกเป็นสี่เหลี่ยมจัตุรัสในรัศมีนั้น ๆ ดังรูปที่



รูปที่ 3.12 แสดงการขยายขอบเขตของพื้นที่ออกเป็นสี่เหลี่ยมจัตุรัสในรัศมีที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยจะนำ Geometry Column มาทำการ && กับฟังก์ชัน Expand เพื่อเลือกอาณาเฉพาะแถวที่มีข้อมูลอยู่ในบริเวณนั้น และเพิ่มเงื่อนไขตรวจสอบว่าจุดใด ๆ ที่มีระยะขจัดจากจุดนั้น ๆ ถึงจุดที่ผู้ใช้ระบุมีระยะขจัดน้อยกว่ารัศมีที่ผู้ใช้ต้องการ คือ ฟังก์ชัน Distance(geometry,geometry)
 สูตรท้ายผลลัพธ์ที่ได้เมื่อเรียกคำสั่ง SQL นี้ คือ



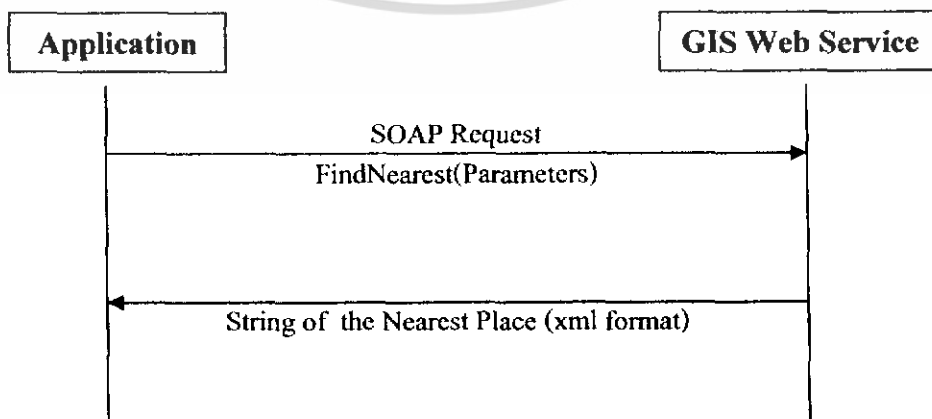
รูปที่ 3.13 แสดงการขยายขอบเขตของพื้นที่ออกเป็นวงกลมในพื้นที่สี่เหลี่ยมจัตุรัสในรัศมีที่ต้องการ

3.3.7 FindNearest

บริการค้นหาสถานที่ที่ใกล้กับจุดอ้างอิงมากที่สุด

```
Public string FindNearest(Parameters)
```

คืนค่าเป็นสตริงของเอกสาร XML ที่ระบุรายละเอียดของสถานที่ที่ใกล้กับจุดอ้างอิงมากที่สุด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

รูปที่ 3.14 การทำงานของ FindNearest Service

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชนิด	พารามิเตอร์	คำอธิบาย
Double	coordinateX	พิกัดอ้างอิงมาตรฐาน UTM
Double	coordinateY	พิกัดอ้างอิงมาตรฐาน UTM
String	Layer	เลเยอร์ที่ต้องการค้นหา

ตารางที่ 3.6 อธิบายพารามิเตอร์ของ FindNearest Service

ในการเรียกใช้บริการผู้ใช้จะต้องส่งพารามิเตอร์มา จากนั้นระบบจะทำการค้นหาข้อมูลจากฐานข้อมูล โดยใช้คำสั่งภาษา SQL ดังนี้

```
SELECT Geometry Column FROM Table
WHERE Distance('POINT(coordinateX coordinateY)',Geometry Column) <=
ALL(SELECT Distance('POINT(coordinateX coordinateY)',Geometry Column)
FROM Table)
```

การค้นหาข้อมูลที่อยู่ใกล้กับจุดอ้างอิงมากที่สุด จะใช้หลักการของ Sub Queries ในการหาระยะจัดโดยเทียบกันระหว่างจุดอ้างอิงกับพิกัดในแต่ละแถว

เมื่อได้แถวที่ใกล้กับจุดอ้างอิงมากที่สุดก็จะนำค่าที่ได้มาสร้างเป็นเอกสาร XML แสดงรายละเอียดของสถานที่ที่ค้นหาพบ แล้วคืนค่ากลับไปยังผู้ใช้

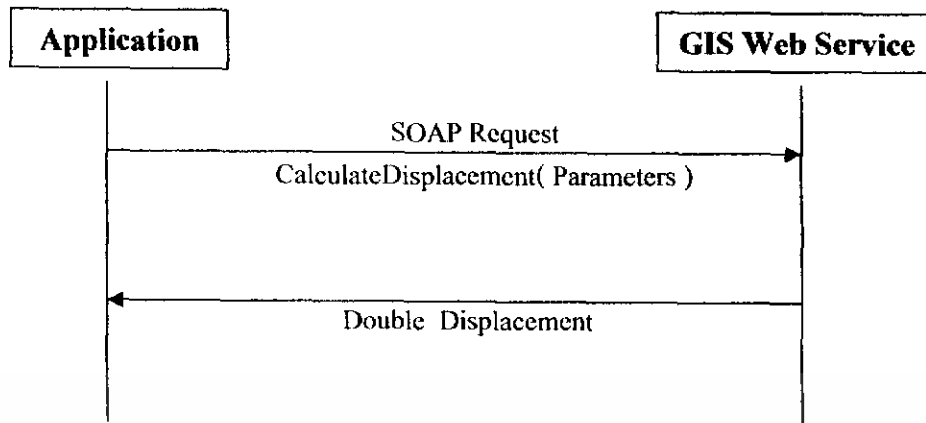
3.3.8 CalculateDisplacement

บริการคำนวณระยะจัดตามพิกัดระหว่าง 2 จุด

Public Double CalculateDisplacement(Parameter)

รับพารามิเตอร์เป็นพิกัด 2 ตำแหน่งซึ่งจะระบุพิกัดที่จะให้คำนวณหาระยะจัดระหว่างจุด 2 จุด คืนค่าเป็นค่าระยะจัดระหว่างจุด 2 จุดที่คำนวณได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.15 การทำงานของ CalculateDisplacement Service

ชนิด	พารามิเตอร์	คำอธิบาย
Double	startCoordinateX	พิกัดเริ่มต้นมาตรฐาน UTM
Double	startCoordinateY	พิกัดเริ่มต้นมาตรฐาน UTM
Double	endCoordinateX	พิกัดสุดท้ายมาตรฐาน UTM
Double	endCoordinateY	พิกัดสุดท้ายมาตรฐาน UTM

ตารางที่ 3.7 อธิบายพารามิเตอร์ของ CalculateDisplacement Service

การคำนวณหาระยะขจัดระหว่างจุด 2 จุด สามารถใช้คำสั่งภาษา SQL ได้ดังนี้

```
SELECT Distance ('POINT(startCoordinateX startCoordinateY)',
                 'POINT(endCoordinateX endCoordinateY)')
```

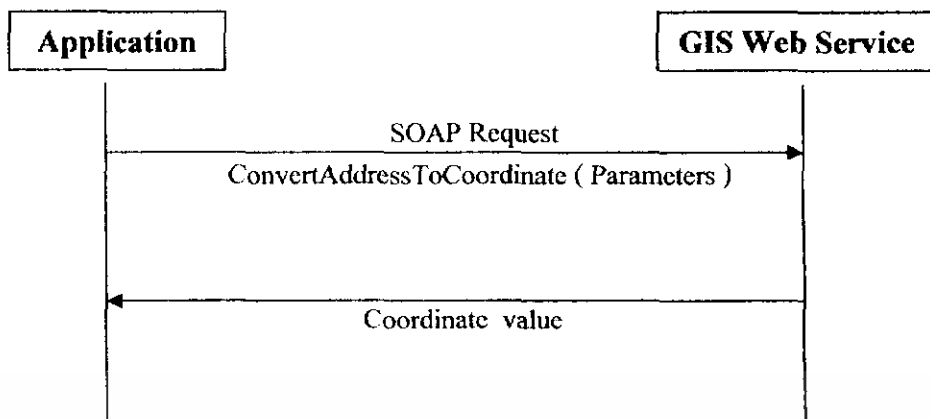
3.3.9 ConvertAddressToCoordinate

บริการแปลงที่อยู่เป็นค่าพิกัดระบุตำแหน่งมาตรฐาน UTM

```
Public Double[] ConvertAddressToCoordinate (Parameters)
```

รับพารามิเตอร์ ซึ่งระบุที่อยู่เช่น เลขที่ ชื่อถนน อำเภอ จังหวัด รหัสไปรษณีย์ เป็นต้น
คืนค่าเป็นพิกัดที่ตรงกับที่อยู่ที่อยู่ (Latitude , Longitude)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.16 การทำงานของ ConvertAddressToCoordinate Service

ชนิด	พารามิเตอร์	คำอธิบาย
String	address	บ้านเลขที่
String	road	ชื่อถนน
String	district	ชื่ออำเภอ
String	province	ชื่อจังหวัด
String	postcode	รหัสไปรษณีย์

ตารางที่ 3.8 อธิบายพารามิเตอร์ของ ConvertAddressToCoordinate Service

3.4 การออกแบบ GIS Application

3.4.1 โครงสร้างการทำงานของ GIS Application

แอปพลิเคชันสร้างเพื่อทดสอบการเรียกใช้บริการต่างๆ จาก GIS เว็บเซอร์วิส ซึ่งใช้ .Net Framework เป็นตัวจัดการการติดต่อกับเว็บเซอร์วิส ซึ่งแอปพลิเคชันจะมีความสามารถทั่วไปเกี่ยวกับการแสดงผลแผนที่ในรูปแบบ SVG โดยใช้ไลบรารีของ Adobe SVG Viewer 3.0 ในการแสดงผล SVG โดยมีความสามารถในการจัดการแผนที่ SVG หลักๆ ดังนี้

- Zoom In
- Zoom Out
- Original View
- Copy SVG
- View SVG
- View Source
- Save SVG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากนั้นยังมีส่วนที่เรียกใช้บริการจาก GIS เว็บเซอร์วิส ซึ่งมีบริการต่างๆ ดังที่กล่าวมาแล้วข้างต้น

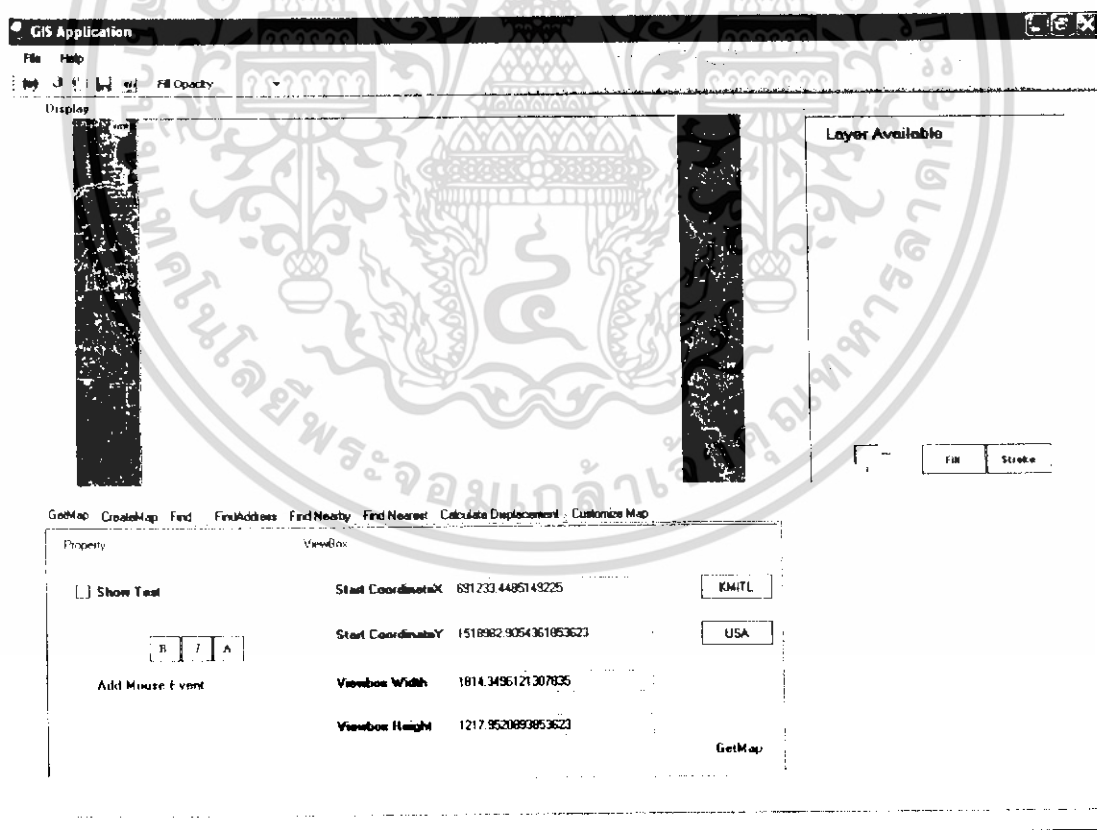
3.4.2 การเรียกใช้บริการต่างๆ จาก GIS Web Services

การที่ผู้ใช้งานจะสามารถเรียกใช้บริการจากเว็บเซอร์วิสได้นั้นผู้ใช้งานจะต้องรู้ถึงบริการต่างๆ ที่มีในเว็บเซอร์วิสนั้นๆ ก่อน โดยดูจากประกาศบนเซิร์ฟเวอร์ที่เราต้องการใช้บริการ และยังสามารถรู้ถึงการเรียกใช้ และรายละเอียดต่างๆ ของบริการผ่านทาง WSDL อีกด้วย

แอปพลิเคชันที่พัฒนาโดยใช้เครื่องมือ Visual Studio .Net นั้นสามารถเรียกใช้บริการจากเว็บเซอร์วิสที่ต้องการ โดยการเพิ่ม Web Reference ไปยังเว็บเซอร์วิสที่ต้องการ และสามารถเรียกใช้เว็บเซอร์วิสผ่านทาง Web Reference ที่สร้างขึ้นมา

รูปแบบการสร้าง Web Reference ดังนี้

```
WebReference.Service Service = new WebReference.Service();
```

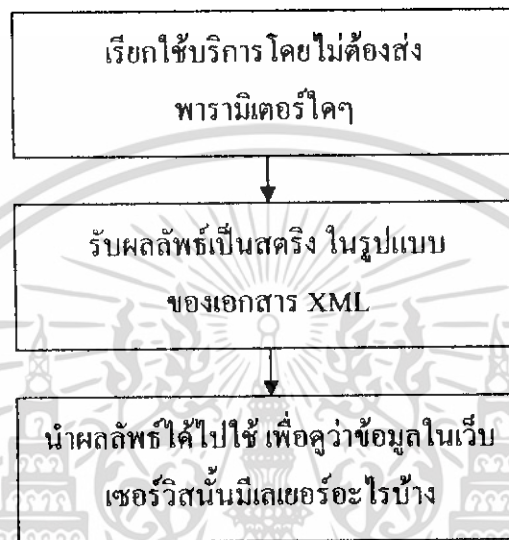


รูปที่ 3.17 แสดง Interface โดยรวมของแอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.2.1 การเรียกใช้บริการ GetCapabilities

บริการ GetCapabilities ผู้ใช้สามารถเรียกใช้โดยที่ไม่ต้องส่งค่าพารามิเตอร์ใดมากับการเรียกใช้บริการ สำหรับแอปพลิเคชันนี้ส่วนที่เรียกใช้บริการ GetCapabilities ก็คือ toolbar และ menu (Connect to map server) ซึ่งเมื่อกด ตัวแอปพลิเคชันจะทำการเรียกใช้บริการ GetCapabilities เพื่อขอรู้ว่า ในเซิร์ฟเวอร์นั้นมีข้อมูลอะไรบ้าง มีเลเยอร์อะไรบ้าง เพื่อที่จะนำมาใส่ใน Check List Box เพื่อให้ผู้ใช้สามารถเลือกเลเยอร์ที่มีใน map server ได้

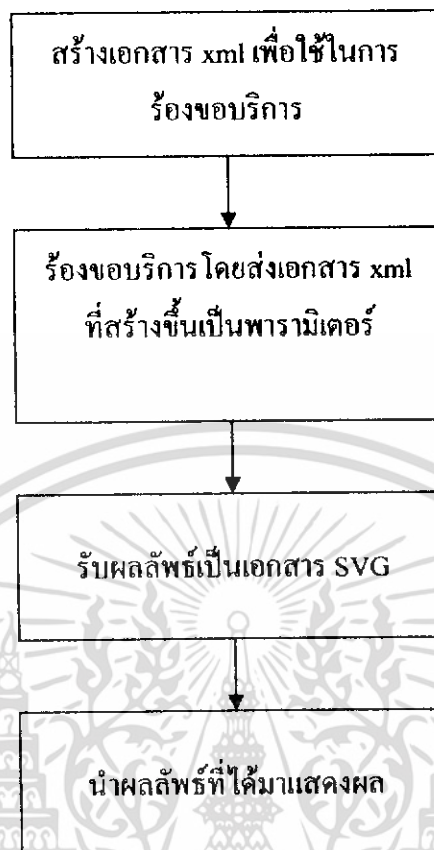


รูปที่ 3.18 แสดงขั้นตอนการเรียกใช้บริการ GetCapabilities

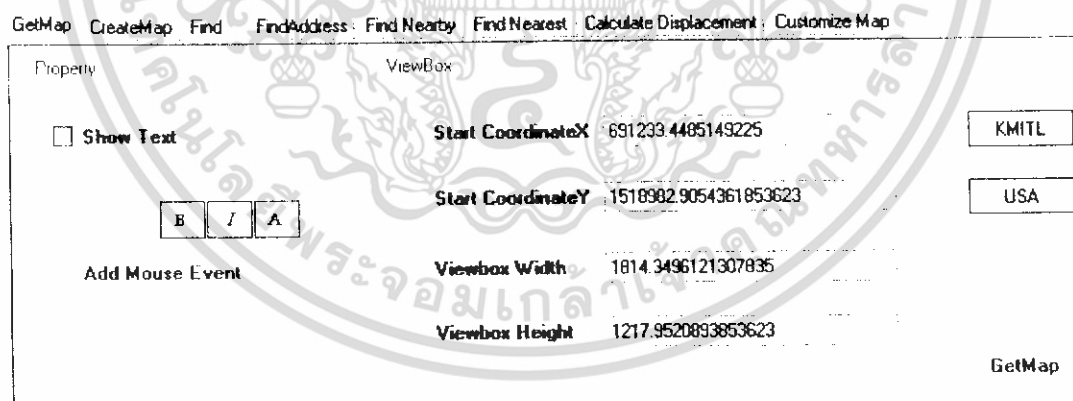
3.4.2.2 การเรียกใช้บริการรับแผนที่ในรูปแบบ SVG (GetMap)

การเรียกใช้บริการ GetMap ผู้ใช้ต้องส่งพารามิเตอร์ตาม getMapRequest XML Schema เพื่อเรียกใช้บริการ GetMap ซึ่งผู้ใช้สามารถระบุเลเยอร์ สีของแต่ละเลเยอร์ ต้องการแสดงข้อความหรือไม่ต้องการแสดงข้อความ รูปแบบของตัวอักษร สีของตัวอักษร ระบุขนาดของแผนที่ SVG ระบุค่า ใน View box ของไฟล์ SVG ซึ่งได้แก่ พิกัดเริ่มต้น ขนาดของ View box เป็นต้นให้ครบและถูกต้อง

ซึ่งสามารถสรุปขั้นตอนการทำงาน ได้ดังนี้



รูปที่ 3.19 ขั้นตอนในการเรียกใช้บริการ GetMap



รูปที่ 3.20 แสดง Interface ของการเรียกใช้บริการ GetMap

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่าง getMapRequest XML

```

<GetMap>
  <viewbox>
    <vbox1>691233.448514923</vbox1>
    <vbox2>1518982.90543619</vbox2>
    <vbox3>1814.34961213078</vbox3>
    <vbox4>1217.95208938536</vbox4>
  </viewbox>
  <width>508</width>
  <height>345</height>
  <opacity>1</opacity>
  <text>
    <text-visible>True</text-visible>
    <text-family>arial</text-family>
    <text-weight>normal</text-weight>
    <text-style>normal</text-style>
    <text-size>15</text-size>
    <text-color>black</text-color>
  </text>
  <mouse-event>True</mouse-event>
  <layer>
    <name>Road</name>
    <fill-color>white</fill-color>
    <stroke-color>black</stroke-color>
  </layer>
  <layer>
    <name>Building</name>
    <fill-color>white</fill-color>
    <stroke-color>black</stroke-color>
  </layer>
  <layer>
    <name>Engineering </name>
    <fill-color>white</fill-color>
    <stroke-color>black</stroke-color>
  </layer>
  <layer>
    <name>Science </name>
    <fill-color>white</fill-color>
    <stroke-color>black</stroke-color>
  </layer>
  <layer>
    <name>Agriculture Industrial </name>
    <fill-color>white</fill-color>
    <stroke-color>black</stroke-color>
  </layer>
</GetMap>

```

3.4.2.3 การเรียกใช้บริการสร้างแผนที่ในรูปแบบ SVG (CreateMap)

เป็นบริการแสดงแผนที่โดยสามารถเพิ่มสถานที่ใหม่ ตามตำแหน่งพิกัดที่สามารถระบุได้ โดยส่งเอกสาร XML ที่มีแท็กของสถานที่ใหม่ร่วมกับแท็กของสถานที่ต่างๆ จากเลเยอร์ที่ได้ทำการเลือกไว้ การเรียกใช้บริการต้องสร้างเอกสาร XML เพื่อส่งเป็นพารามิเตอร์ ซึ่งถูกต้องตาม createMapRequest XML Schema ซึ่งได้ระบุไว้ในส่วน GIS เว็บเซอร์วิส

ผู้ใช้งานจะได้รับเอกสาร SVG ซึ่งแสดงผลเลเยอร์ที่ได้ทำการเลือกและพิกัดที่เพิ่มเข้าไป ซึ่งเอกสารนี้เป็นเอกสารที่สร้างขึ้นไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.22 แสดง Interface ของการเรียกใช้บริการ CreateMap

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่าง createMapRequest XML

```

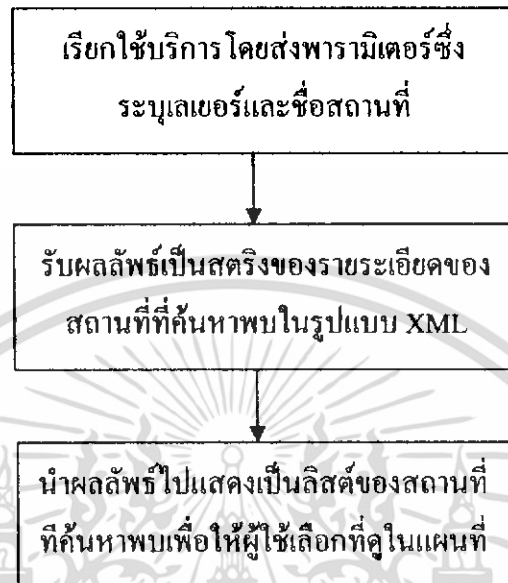
<?xml version="1.0" encoding="UTF-8"?>
<CreateMap>
  <viewbox>
    <vbox1>691233.448514923</vbox1>
    <vbox2>1518982.90543619</vbox2>
    <vbox3>1814.34961213078</vbox3>
    <vbox4>1217.95208938536</vbox4>
  </viewbox>
  <width>508</width>
  <height>345</height>
  <opacity>1</opacity>
  <text>
    <text-visible>True</text-visible>
    <text-family>Tahoma</text-family>
    <text-weight>bold</text-weight>
    <text-style>italic</text-style>
    <text-size>15</text-size>
    <text-color>rgb(0 0 255)</text-color>
  </text>
  <mouse-event>True</mouse-event>
  <layer>
    <name>Road</name>
    <fill-color>white</fill-color>
    <stroke-color>black</stroke-color>
  </layer>
  <layer>
    <name>Building</name>
    <fill-color>white</fill-color>
    <stroke-color>black</stroke-color>
  </layer>
  <layer>
    <name>Engineering </name>
    <fill-color>white</fill-color>
    <stroke-color>black</stroke-color>
  </layer>
  <item>
    <item-coordinate>691300 1518740,691400 1518740,691400
1518640,691300 1518640,691300 1518740</item-coordinate>
    <item-name>Ahashop</item-name>
    <item-reference>http://www.aha.com</item-reference>
    <item-detail>aha shop sell coffee</item-detail>
    <item-fill>rgb(255 128 64)</item-fill>
    <item-stroke>rgb(128 0 0)</item-stroke>
  </item>
</CreateMap>

```

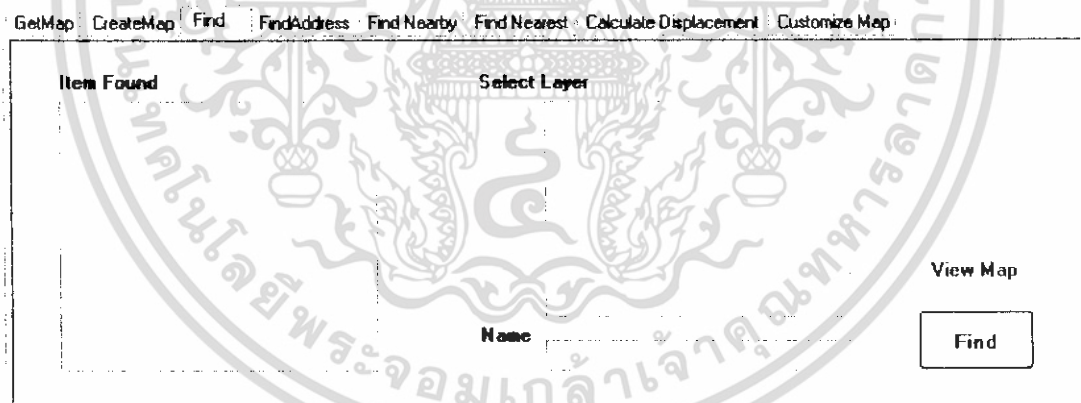
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.2.4 การเรียกใช้บริการค้นหาสถานที่ (Find)

บริการ Find เป็นบริการค้นหาสถานที่โดยผู้ใช้งานจะต้องระบุเลขเซอร์และสามารถชื่อสถานที่ที่ต้องการค้นหา



รูปที่ 3.23 ขั้นตอนในการเรียกใช้บริการ Find

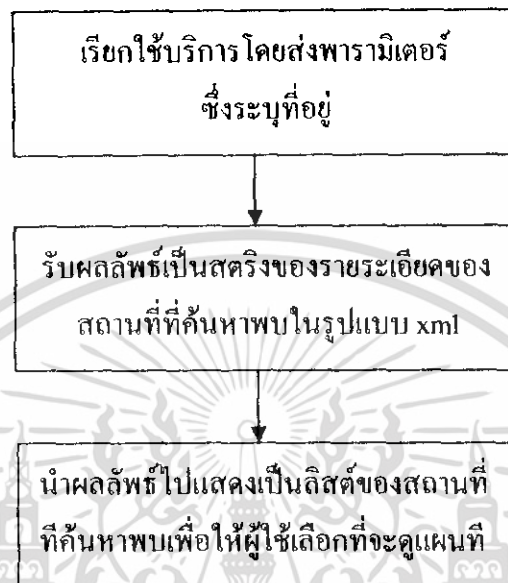


รูปที่ 3.24 Interface ของการเรียกใช้บริการ Find

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.2.5 การเรียกใช้บริการค้นหาสถานที่โดยระบุเป็นที่อยู่ (FindAddress)

ผู้ใช้สามารถเรียกใช้บริการค้นหาสถานที่ที่สามารถระบุที่อยู่ได้ โดยระบุตามพารามิเตอร์ที่จำเป็นต้องส่งไปเพื่อเรียกใช้บริการ



รูปที่ 3.25 แสดงขั้นตอนการเรียกใช้บริการ FindAddress

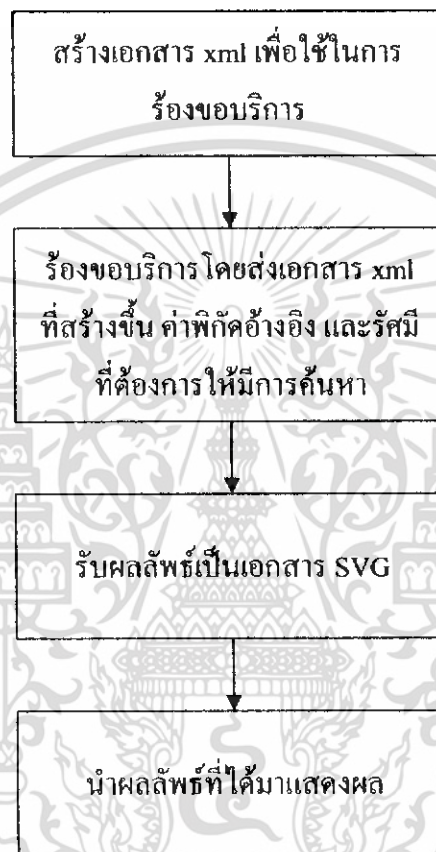
Specify Address	
Address	
Street	View Map
District	Convert To Coordinate
Province	Find Address
Post Code	

รูปที่ 3.26 interface การเรียกใช้บริการ FindAddress

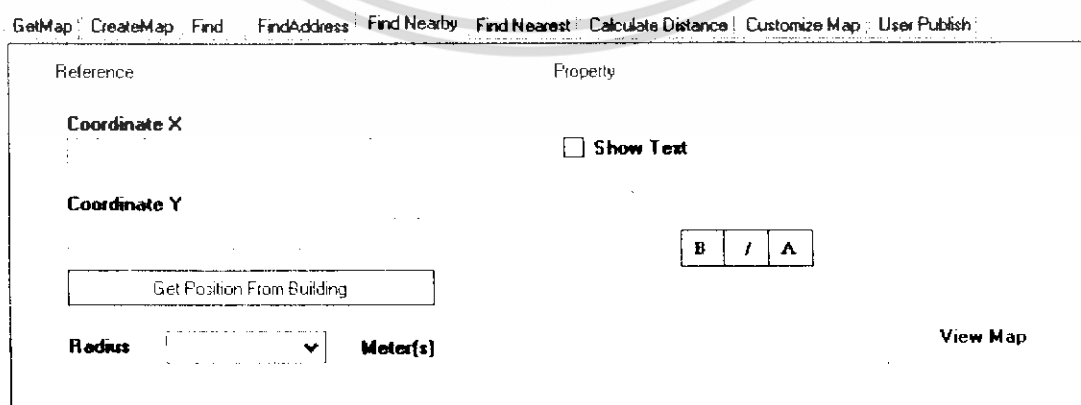
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.2.6 การเรียกใช้บริการค้นหาสถานที่ภายในรัศมีรอบจุดอ้างอิง (FindNearby)

เป็นบริการแสดงแผนที่โดยแสดงแผนที่จากจุดอ้างอิงที่กำหนดและบริเวณรัศมีรอบๆ ตามที่ได้ระบุไว้ โดยผู้ใช้ต้องสามารถระบุพิกัดอ้างอิงมาตรฐาน UTM และรัศมีขอบเขตที่ต้องการให้ค้นหาและสร้างแผนที่(หน่วยเป็นเมตร) ผู้ใช้ร้องขอบริการจะต้องส่งพารามิเตอร์ตามที่กำหนดข้างต้นผ่านไปยังเว็บเซอวิซ และจะได้รับเอกสาร SVG กลับเพื่อนำไปใช้ในการแสดงผลต่อไป สามารถสรุปขั้นตอนการทำงานการเรียกใช้บริการค้นหาสถานที่ภายในรัศมีรอบจุดอ้างอิงได้ดังนี้



รูปที่ 3.27 แสดงขั้นตอนการทำงานการเรียกใช้บริการ FindNearby

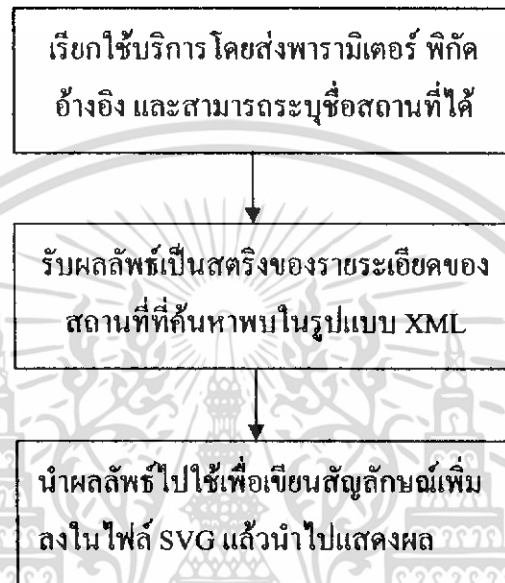


รูปที่ 3.28 แสดง interface การเรียกใช้บริการ FindNearby เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้ในเพื่อการศึกษาเท่านั้น เมื่อผู้ผู้เห็นได้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.2.7 การเรียกใช้บริการค้นหาสถานที่ที่อยู่ใกล้จุดอ้างอิงมากที่สุด (FindNearest)

ผู้ใช้สามารถค้นหาสถานที่ที่ต้องการในเลขอร์ต่างๆ ที่อยู่ใกล้กับตำแหน่งจุดอ้างอิงผู้ใช้มากที่สุด

สามารถสรุปขั้นตอนการเรียกใช้บริการค้นหาสถานที่ที่อยู่ใกล้กับจุดอ้างอิงมากที่สุด (FindNearest) ได้ดังนี้



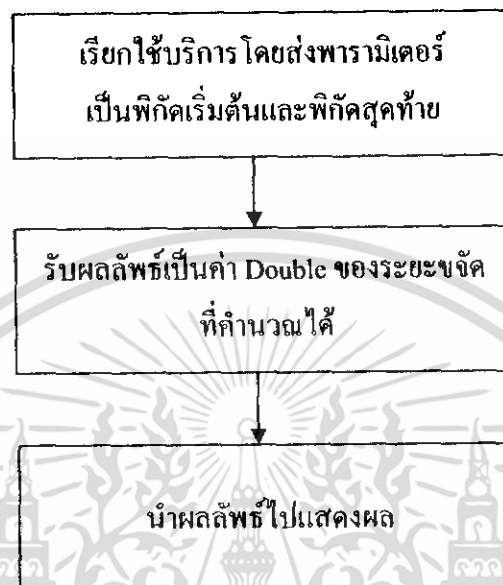
รูปที่ 3.29 แสดงขั้นตอนการเรียกใช้บริการ FindNearest

รูปที่ 3.30 แสดง interface การเรียกใช้บริการ FindNearest

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.2.8 การเรียกใช้บริการคำนวณระยะขจัดระหว่าง 2 จุด (CalculateDisplacement)

ผู้ใช้สามารถเรียกใช้บริการ โดยระบุพิกัดอ้างอิง 2 จุด คือจุดเริ่มต้นและจุดสิ้นสุดเพื่อคำนวณหาระยะขจัดระหว่างจุด 2 จุด พิกัดที่ระบุต้องอยู่ในรูปแบบพิกัดมาตรฐาน UTM สามารถดูขั้นตอนการเรียกใช้บริการคำนวณระยะห่างระหว่าง 2 จุดได้ดังนี้



รูปที่ 3.31 แสดงขั้นตอนการทำงานการเรียกใช้บริการ Calculate Displacement

GetMap CreateMap Find FindAddress Find Nearby Find Nearest Calculate Displacement Customize Map

Reference Point Displacement (Metres)

Coordinate X

Coordinate Y

Get position from entity

◀ ▶ edit Add Calculate Displacement

รูปที่ 3.32 แสดง interface การเรียกใช้บริการ Calculate Displacement

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

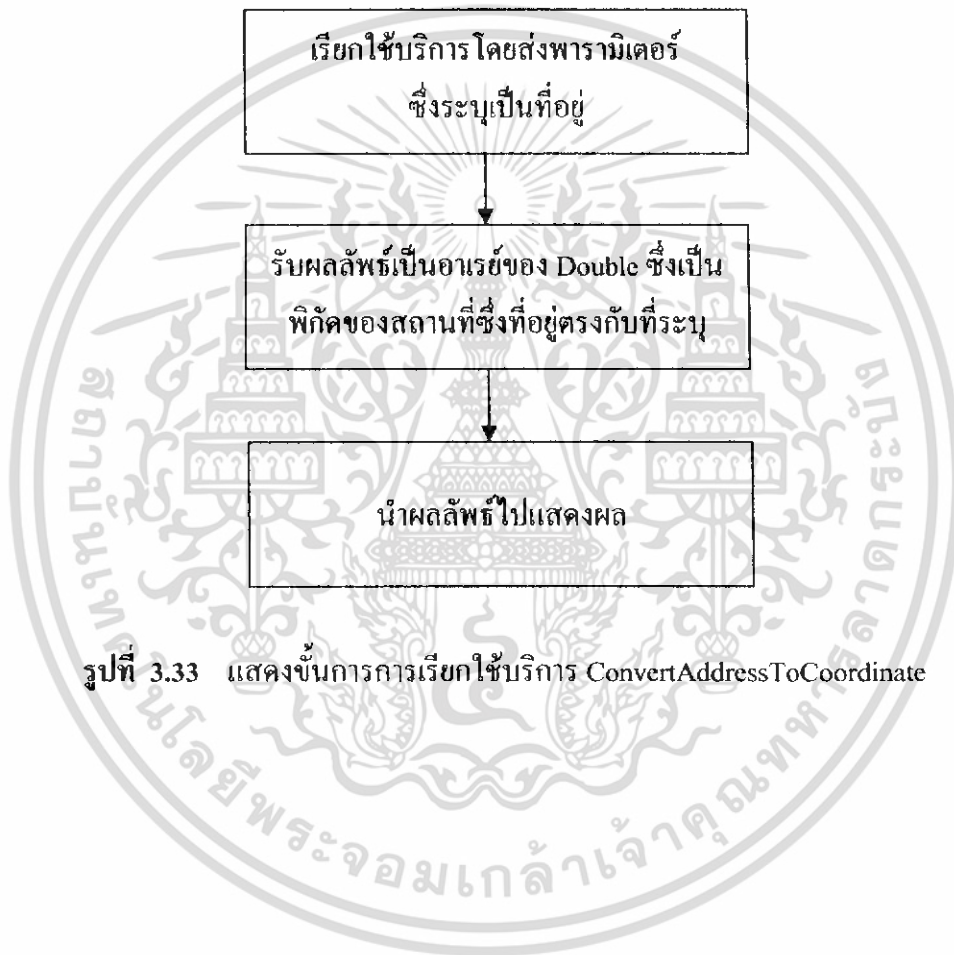
3.4.2.9 การเรียกใช้บริการแปลงที่อยู่เป็นพิกัด (ConvertAddressToCoordinate)

ส่วนนี้เป็นการเรียกใช้บริการเซอร์วิส ConvertAddressToCoordinate ซึ่งจะทำการค้นหาที่อยู่พื้นฐานข้อมูลแล้วแปลงเป็นพิกัด เพื่อส่งค่าคืนให้กับผู้ใช้

รูปแบบในการเรียกใช้งานเซอร์วิสดังนี้

```
Double[] coordinate = Service.ConvertAddressToCoordinate(Parameters)
```

สามารถสรุปขั้นตอนการเรียกใช้บริการแปลงที่อยู่เป็นพิกัดได้ดังนี้



รูปที่ 3.33 แสดงขั้นตอนการการเรียกใช้บริการ ConvertAddressToCoordinate

บทที่ 4

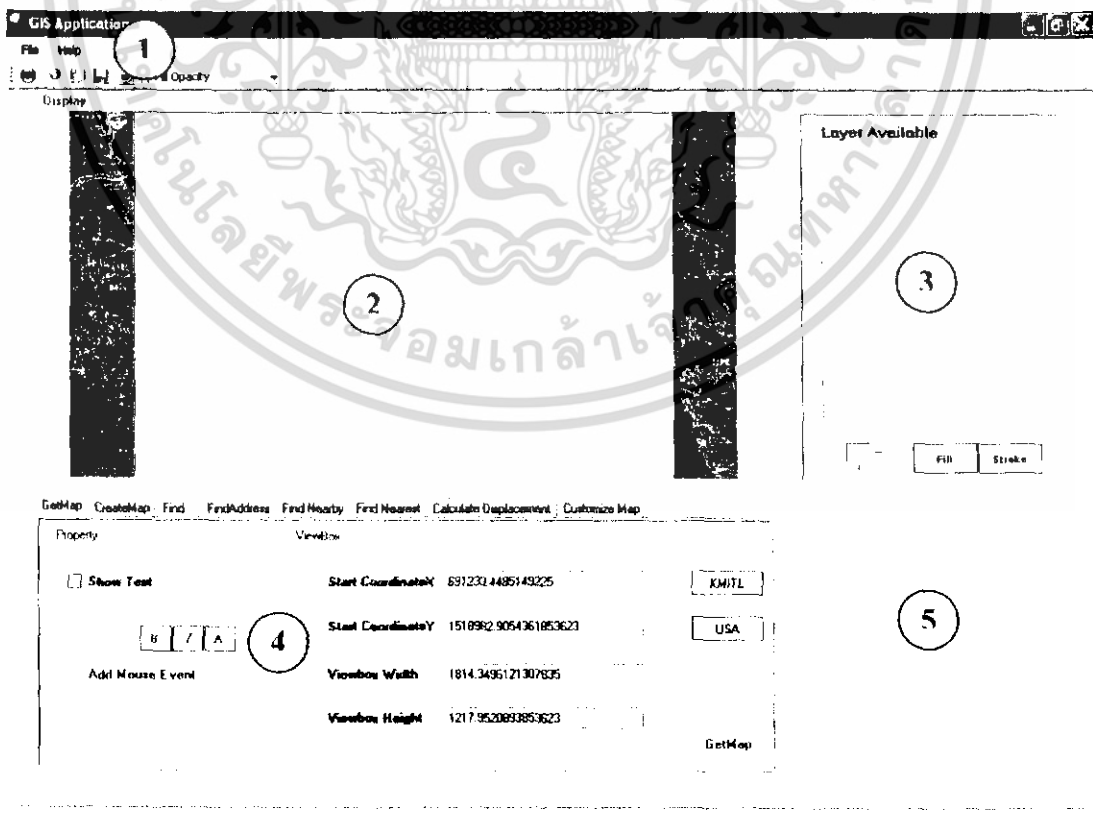
ผลการทดลอง

4.1 ผลการทดลองในส่วนของ GIS Application

ผลที่ได้ทำการทดลองในการแสดงผลแผนที่ ตามฟังก์ชันต่างๆ มีดังนี้

1. การทดลองการแสดงผลแผนที่ตามเลเยอร์ที่ได้เลือกไว้
2. การทดลองการแสดงผลแผนที่ตามเลเยอร์ที่ได้เลือกรวมกับสถานที่ใหม่ที่ได้สร้างขึ้น
3. การทดลองการค้นหาสถานที่จากชื่อของสถานที่
4. การทดลองการค้นหาสถานที่จากที่อยู่ของสถานที่
5. การทดลองการแสดงผลแผนที่จากจุดอ้างอิงเป็นรัศมีโดยรอบ
6. การทดลองการค้นหาสถานที่ที่ใกล้ที่สุดจากจุดอ้างอิง
7. การทดลองคำนวณระยะทางจากจุดเริ่มต้นถึงจุดสิ้นสุด
8. การทดลองวางสัญลักษณ์ต่างๆ ลงในแผนที่

เมื่อเข้าสู่ Application จะเป็นหน้าต่างของโปรแกรมโดยมีส่วนประกอบต่างๆ ดังนี้

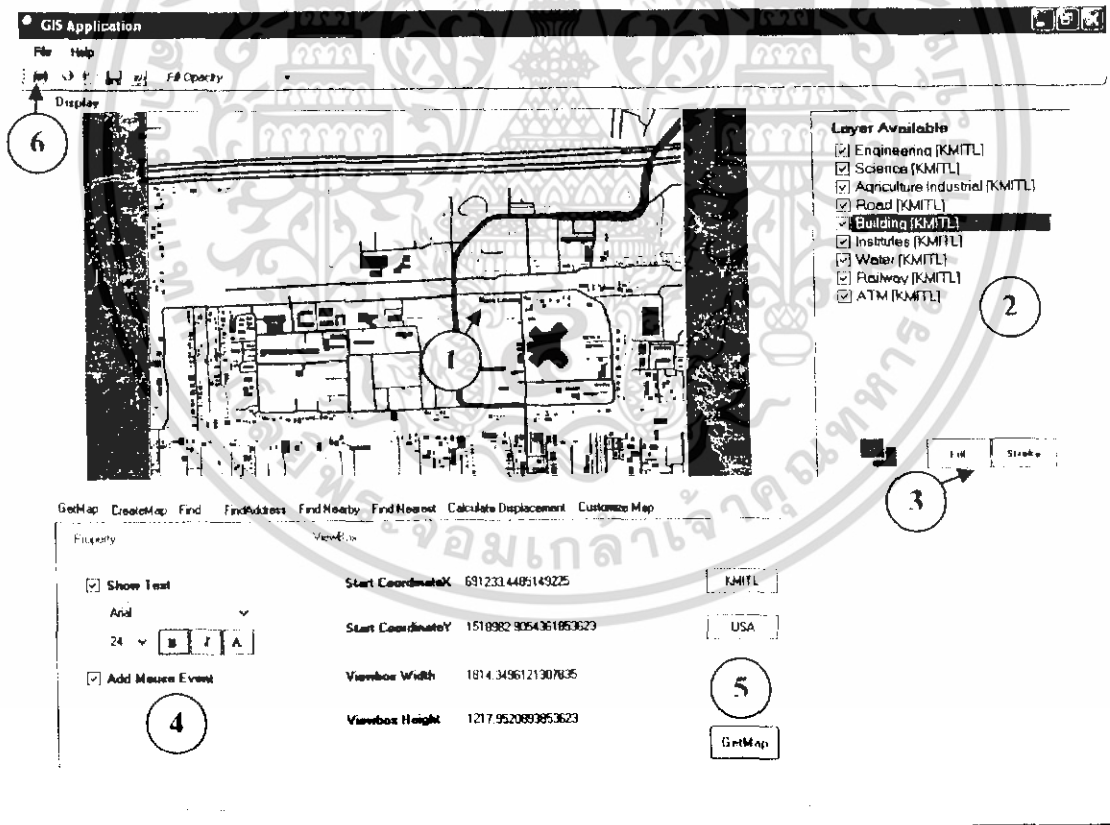


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 4.1 แสดงส่วนต่างๆ ของ Application นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ส่วนที่
1. เป็นส่วนของเมนูบาร์และทูลบาร์
 2. เป็นส่วนที่ใช้แสดงผลแผนที่
 3. เป็นส่วนสำหรับการเลือกเลเยอร์ และสีของเลเยอร์
 4. เป็นส่วนของฟังก์ชันการเรียกใช้บริการต่าง ๆ ของ map server
 5. เป็นส่วนของการแสดงผลข้อมูล

4.1.1 การทดลองการแสดงผลแผนที่ตามเลเยอร์ที่ได้เลือกไว้

การทดลองนี้เริ่มจากการคิดต่อกับ map server โดยกดที่หมายเลข 6 จากนั้นจะสามารถเลือกเลเยอร์และจะปรากฏที่บริเวณหมายเลข 2 จากนั้นทำการเลือกและสีแต่ละเลเยอร์ที่หมายเลข 3 เลือกลักษณะการแสดงผลชื่อของสถานที่ในบริเวณหมายเลข 4 สุดท้ายกดที่ปุ่ม GetMap จากนั้นแผนที่จะถูกแสดงขึ้น โดยสามารถแสดงชื่อ โดยนำมาใส่ไปวางเหนือรูปของสถานที่นั้น (หมายเลข 1)

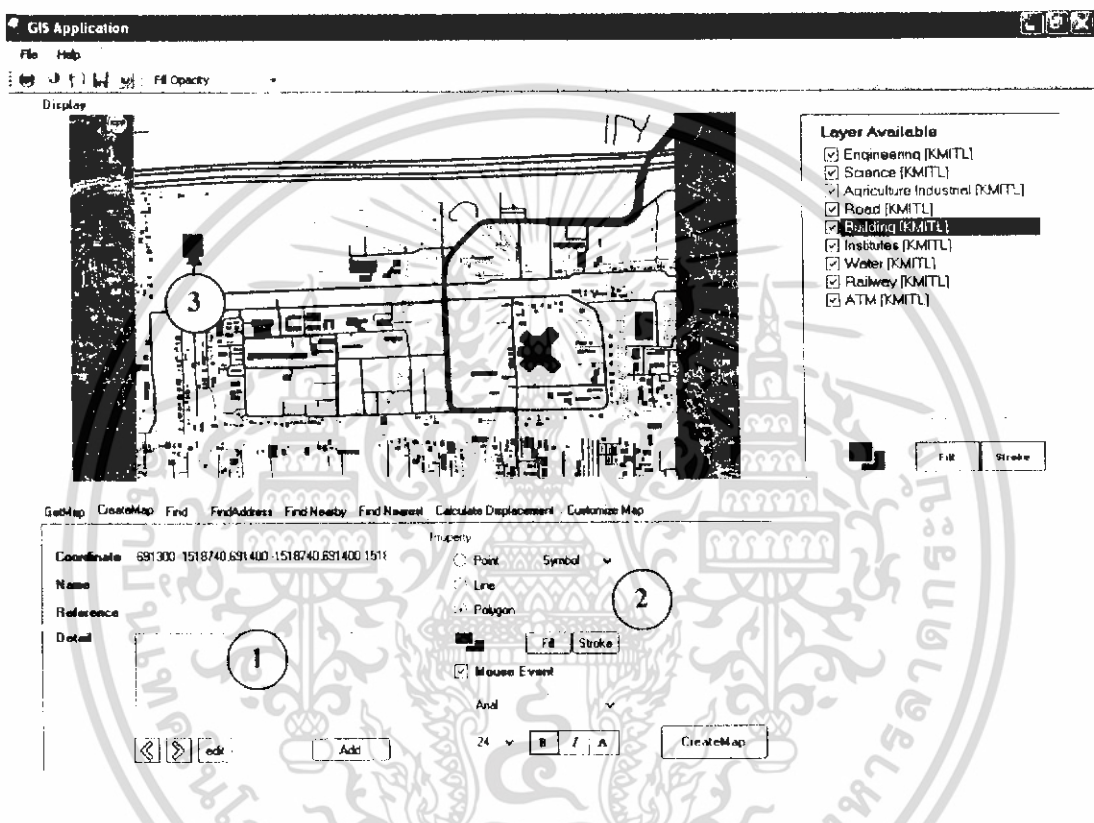


รูปที่ 4.2 การแสดงผลแผนที่

ในส่วนของ ViewBox จะแสดงตัวเลขที่เป็นพิกัดขอบเขตของแผนที่ในการแสดงผล เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 การทดลองการแสดงผลแผนที่ตามเลขอร์ที่ได้เลือกรวมกับสถานที่ใหม่ที่ได้สร้างขึ้น

การทดลองนี้ต้องใส่พิกัดและรายละเอียดต่างๆของสถานที่ที่ต้องการเพิ่มลงไปในพื้นที่ที่ได้รับมาจาก server โดยสามารถใส่ได้หลายสถานที่โดยเมื่อใส่เสร็จหนึ่งครั้งจะต้องทำการกดปุ่ม Add หนึ่งครั้ง และที่บริเวณหมายเลข 2 เป็นส่วนที่ใช้ในการปรับแต่งสี และการแสดงชื่อของสถานที่ที่ได้สร้างขึ้น สุดท้ายต้องกดปุ่ม CreateMap เพื่อแสดงสถานที่ทั้งหมดที่ได้ทำการเพิ่มเข้าไปในพื้นที่ (หมายเลข 3)

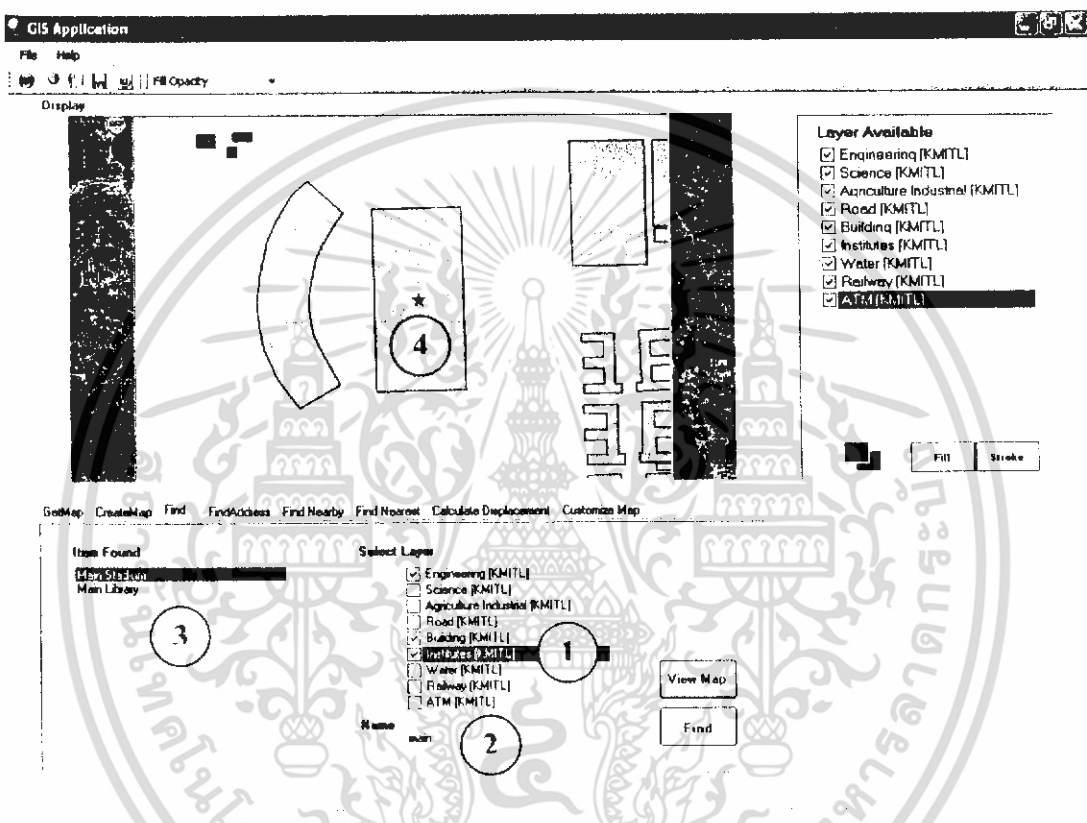


รูปที่ 4.3 แสดงการเพิ่มสถานที่ใหม่เข้าไปในพื้นที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.3 การทดลองการค้นหาสถานที่จากชื่อของสถานที่

การทดลองนี้ใช้ในการค้นหาสถานที่ โดยเริ่มจากบริเวณหมายเลข 1 ทำการเลือกเลเยอร์ที่ต้องการจะค้นหา ก่อน จากนั้นใส่ส่วนของชื่อของสถานที่ที่จะทำการค้นหาที่บริเวณหมายเลข 2 จากนั้นกดปุ่ม Find เพื่อเริ่มทำการค้นหาสถานที่ ชื่อสถานที่ที่ค้นหาจะปรากฏที่บริเวณหมายเลข 3 เมื่อเลือกรายชื่อสถานที่จากรายการ จุดทำยกดปุ่ม View Map เพื่อทำการแสดงสถานที่ที่ได้ทำการเลือกไว้โดยสถานที่นั้นจะมีรูปดาวปรากฏอยู่ หมายเลข 4

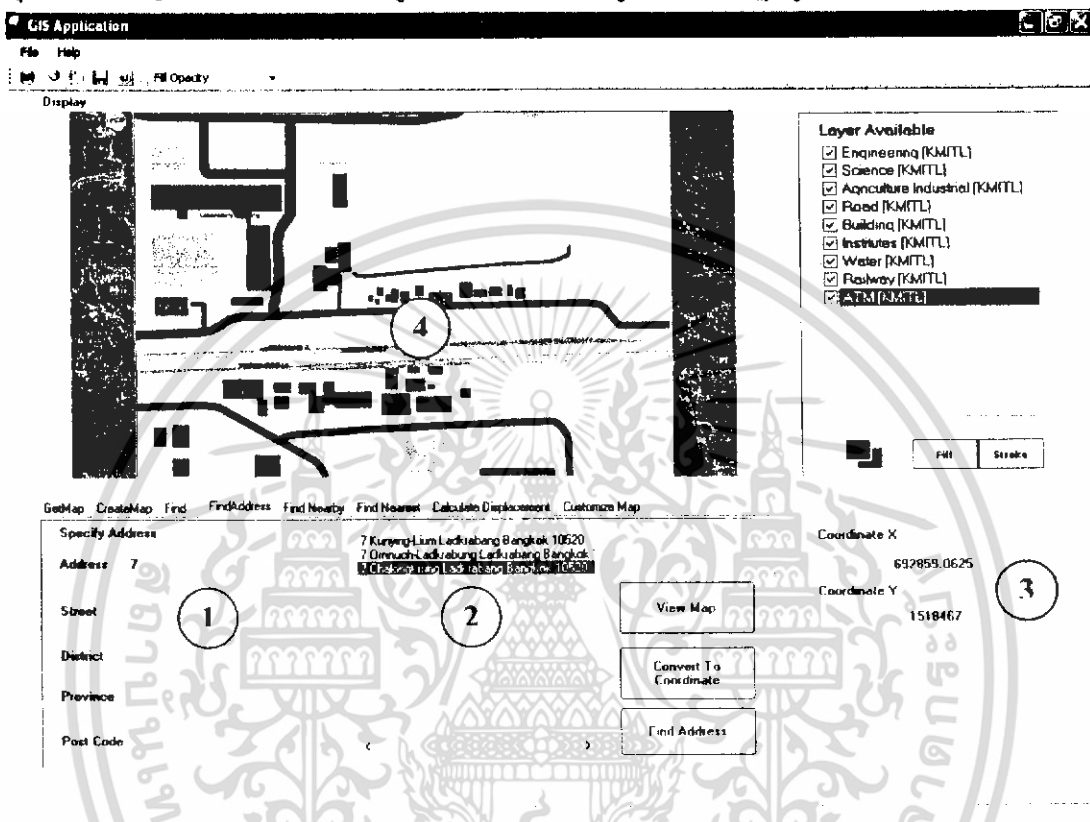


รูปที่ 4.4 แสดงการค้นหาสถานที่ตามชื่อที่ระบุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.4 การทดลองการค้นหาสถานที่จากที่อยู่ของสถานที่

การทดลองการค้นหาสถานที่โดยค้นหาจากที่อยู่ เริ่มจากใส่รายละเอียดเกี่ยวกับที่อยู่ของสถานที่ในบริเวณหมายเลข 1 จากนั้นกดปุ่ม Find Address ผลลัพธ์ในการค้นหาจะถูกแสดงในบริเวณหมายเลข 2 และสามารถกดปุ่ม Convert To Coordinate เพื่อแสดงพิกัดของสถานที่ และปุ่ม View Map เพื่อแสดงสถานที่ที่ถูกเลือกไว้ โดยจะมีรูปดาวปรากฏอยู่

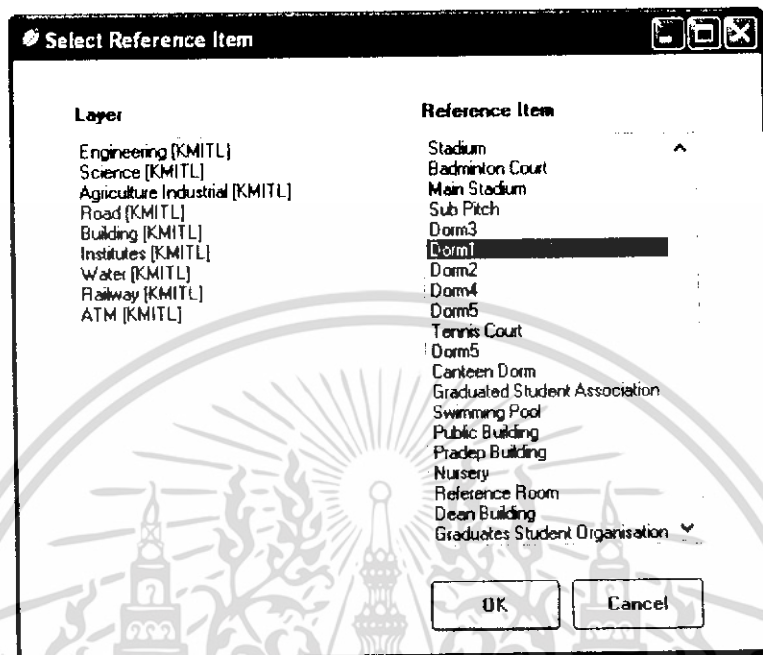


รูปที่ 4.5 แสดงการค้นหาสถานที่ตามที่อยู่ที่ระบุ

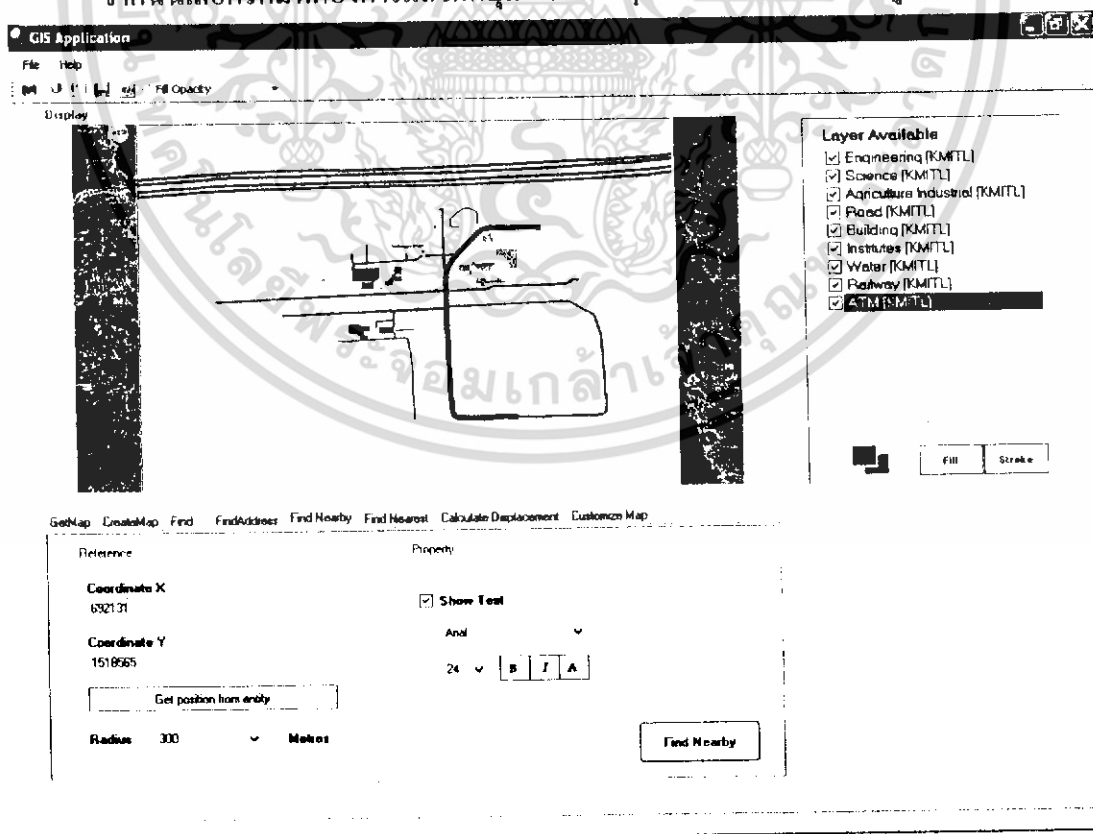
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.5 การทดลองการแสดงผลแผนที่จากจุดอ้างอิงเป็นรัศมีโดยรอบ

การทดลองการแสดงผลแผนที่ โดยเริ่มจากการกำหนดจุดอ้างอิงตรงบริเวณหมายเลข 1 ซึ่งสามารถคลิกปุ่ม Get position from entity เพื่อเลือกจุดอ้างอิงจากสถานที่ที่ได้



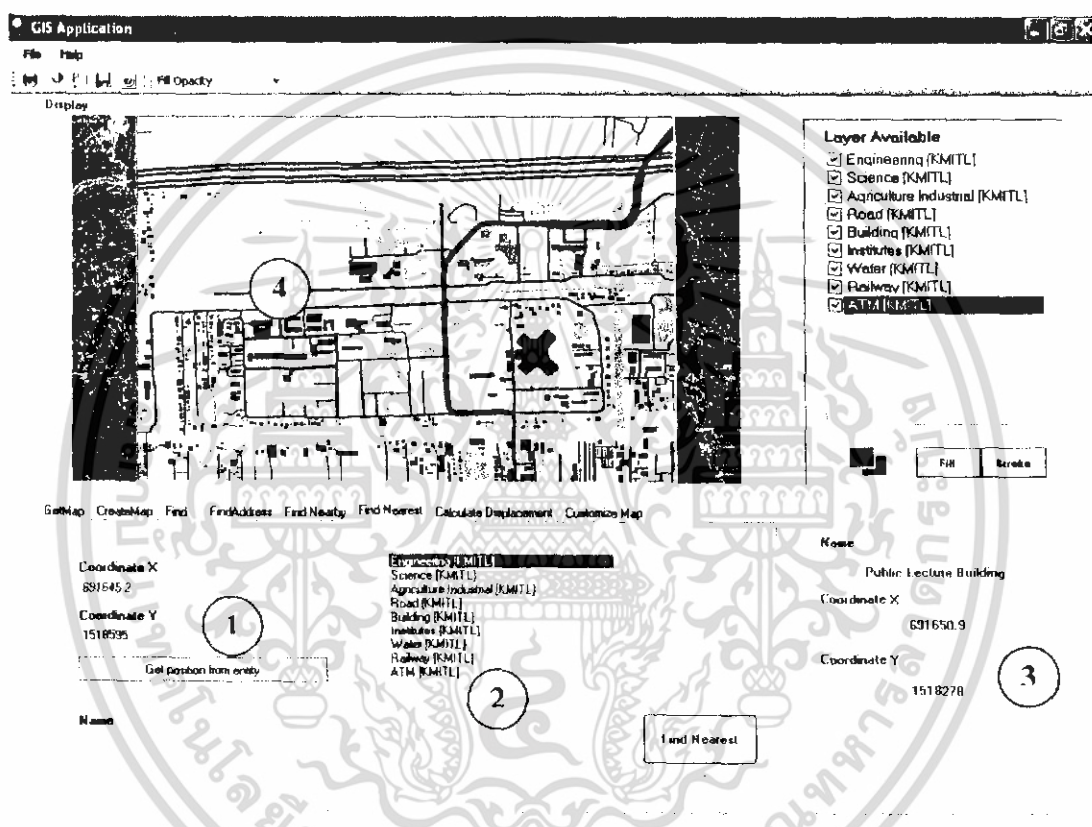
รูปที่ 4.6 แสดงการเลือกสถานที่ที่จะกำหนดเป็นจุดอ้างอิง จากนั้นเลือกรัศมีที่ต้องการแล้วคลิกปุ่ม View Map จะทำการแสดงผลผังรูป



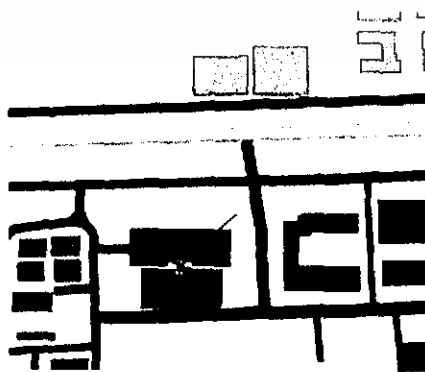
เอกสารนี้เป็นเอกสารที่รูปที่ 4.7 การแสดงผลแผนที่จากจุดอ้างอิง โดยกำหนดรัศมีในการแสดงผล ซึ่งด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.6 การทดลองการค้นหาสถานที่ที่ใกล้ที่สุดจากจุดอ้างอิง

การทดลองนี้จะต้องกำหนดจุดอ้างอิงซึ่งสามารถคลิกปุ่ม Get position from entity เพื่อเลือกสถานที่อ้างอิงได้ หลังจากนั้นจะต้องเลือกเลเยอร์ในบริเวณหมายเลข 2 จะเป็นเลเยอร์ของสถานที่ที่จะทำการค้นหา สูดท้ายกดปุ่ม Find Nearest เพื่อทำการค้นหาสถานที่จากเลเยอร์ที่กำหนดโดยจะเลือกสถานที่ที่ใกล้จุดอ้างอิงที่สุดจากเลเยอร์นั้น รายละเอียดของสถานที่ที่ใกล้จุดอ้างอิงมากที่สุดจะแสดงตรงหมายเลข 3 จากรูปตรงหมายเลข 4 คือสถานที่ผลลัพธ์จากการค้นหา



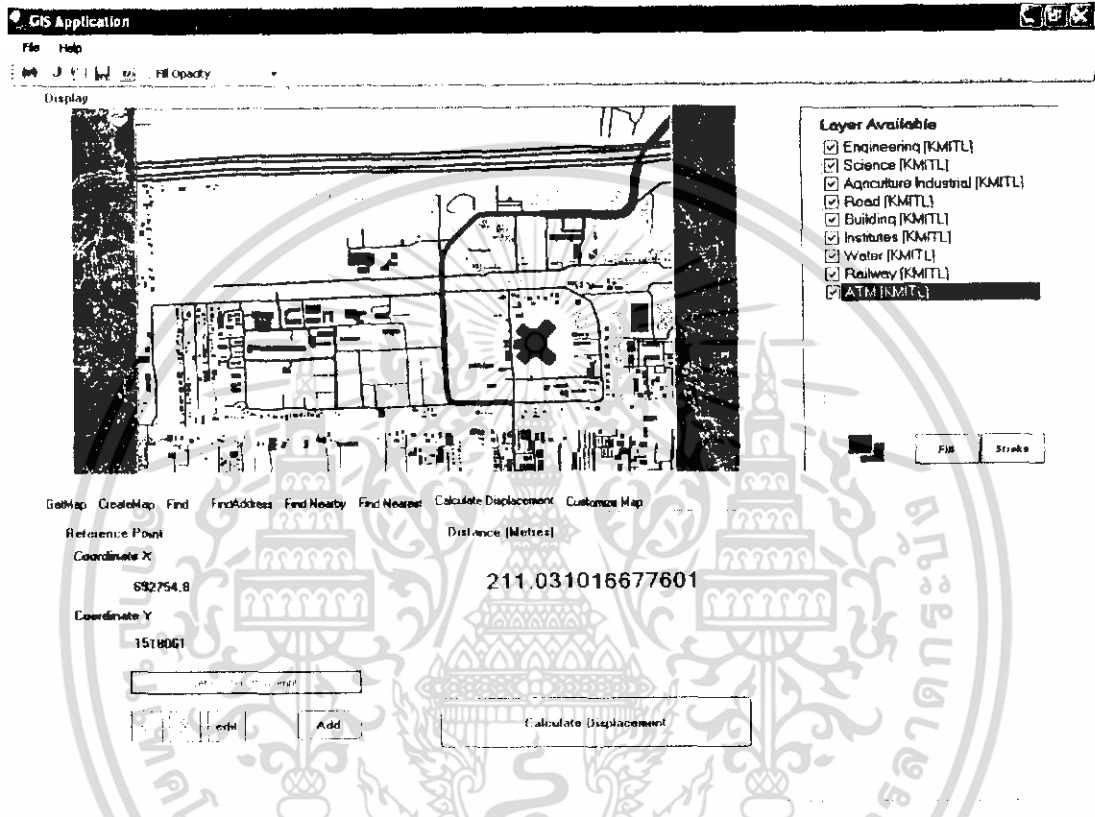
รูปที่ 4.8 แสดงการค้นหาสถานที่ที่ใกล้ที่สุดจากจุดอ้างอิง



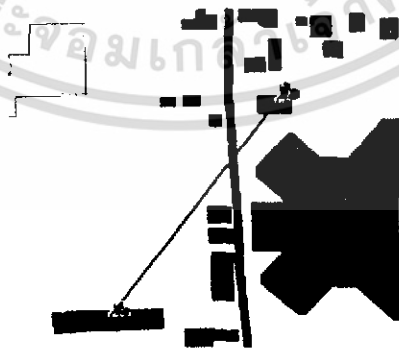
เอกสารนี้เป็นเอกสารที่สงวนรูปที่ 4.9 แสดงผลลัพธ์จากการค้นหา จะมีค่าปรากฏขึ้น ใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.7 การทดลองคำนวณระยะทางจากจุดเริ่มต้นถึงจุดสิ้นสุด

การทดลองคำนวณระยะทางต้องกำหนดพิกัดเริ่มต้นกับพิกัดสิ้นสุด โดยสามารถเลือกพิกัดจากสถานที่ได้จากปุ่ม Get position from entity สามารถเลือกได้หลายจุด สุดท้ายกดปุ่ม Calculate Displacement เพื่อแสดงระยะทางเป็นเส้นตรงจากจุดเริ่มต้นถึงจุดสิ้นสุด โดยจะมีการแสดงระยะทางที่คำนวณได้



รูปที่ 4.10 แสดงการคำนวณระยะทางจากจุดเริ่มต้นถึงจุดสิ้นสุด

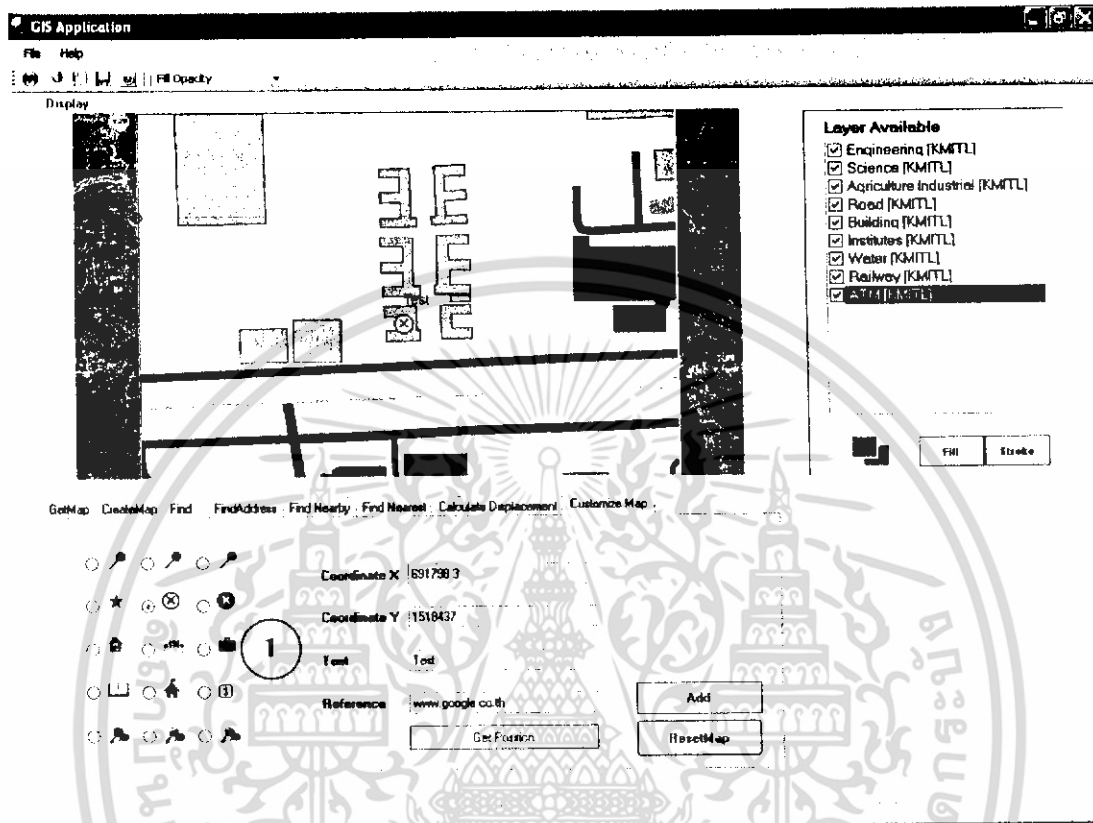


รูปที่ 4.11 แสดงการแสดงผลจะมีพินระบุจุดเริ่มต้นและจุดสิ้นสุด มีเส้นสีแดงเชื่อมทั้ง 2 จุด

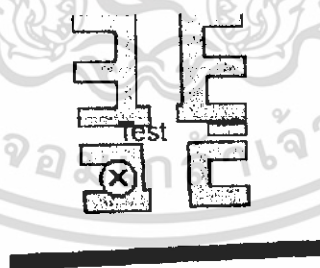
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.8 การทดลองวางสัญลักษณ์ต่างๆ ลงในแผนที่

การทดลองการวางสัญลักษณ์ จะต้องเลือกสัญลักษณ์จากบริเวณหมายเลข 1 หลังจากนั้น ต้องกำหนดพิกัดที่จะทำการวาง โดยสามารถคลิกปุ่ม Get Position เพื่อเลือกพิกัดจากสถานที่ได้



รูปที่ 4.12 แสดงการวางสัญลักษณ์ต่างๆ ลงในแผนที่



รูปที่ 4.13 ผลลัพธ์จากการวางสัญลักษณ์

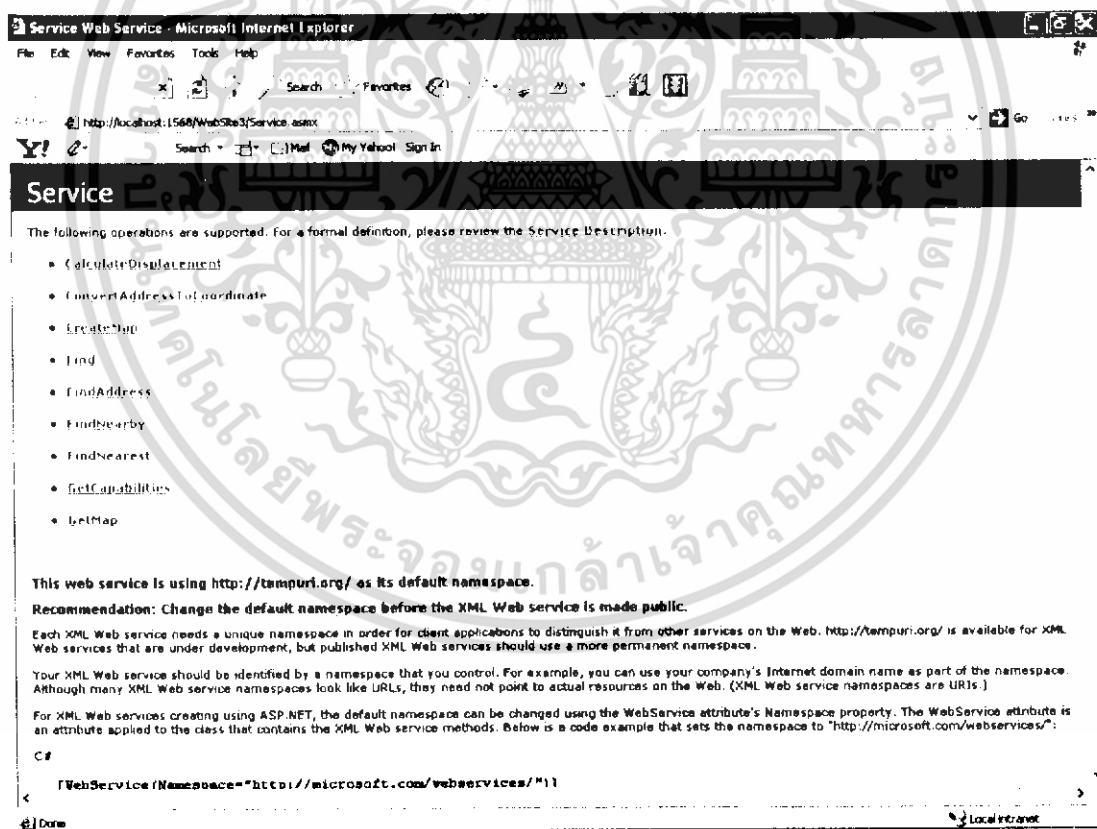
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ผลการทดลองในส่วนของ GIS Web Services

ในส่วนของ GIS Web Services ได้ทำการสร้างบริการต่างๆ ดังนี้

1. บริการ GetCapabilities
2. บริการ GetMap
3. บริการ CreateMap
4. บริการ Find
5. บริการ FindAddress
6. บริการ FindNearby
7. บริการ FindNearest
8. บริการ ConvertAddressToCoordinate
9. บริการ CalculateDisplacement

โดยเมื่อเข้าสู่เว็บเซอร์วิสจะสามารถเลือกใช้บริการต่างๆ ได้ดังรูป



รูปที่ 4.14 GIS Web Services

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทวิจารณ์และสรุป

5.1 บทสรุป

Map server ที่ทำการพัฒนาขึ้นมาเพื่อให้บริการผ่านทางเว็บเซอร์วิส โดยเน้นบริการทางพื้นที่ (Location based services) เป็นหลัก ได้จัดให้มีบริการต่างๆ ได้แก่ บริการเรียกแผนที่ บริการสร้างแผนที่ บริการค้นหาสถานที่ตามชื่อหรือที่อยู่ บริการค้นหาสถานที่ภายในรัศมีจากจุดอ้างอิง บริการค้นหาสถานที่ที่ใกล้กับจุดอ้างอิงมากที่สุด บริการคำนวณระยะขจัดระหว่างจุดอ้างอิง และบริการแปลงที่อยู่เป็นค่าพิกัด โดยทั้งบริการทั้งหมดได้มีการทดลองเรียกใช้งานจากแอปพลิเคชันที่พัฒนาขึ้นมาด้วย

ความได้เปรียบของการใช้เว็บเซอร์วิส คือ ความสามารถในการทำงานร่วมกัน โดยไม่คำนึงถึงแพลตฟอร์ม (collaboration cross platform) ทำให้เว็บเซอร์วิสสามารถเรียกใช้บริการจากเว็บเซอร์วิสด้วยกันได้ สำหรับโครงการที่พัฒนาขึ้นมาได้มีการทำงานร่วมกันระหว่างกลุ่ม Mobile GIS , Linux based GIS Web Services, และ Windows based GIS Web Services พบว่าสามารถทำงานร่วมกันข้ามแพลตฟอร์มได้

5.2 สิ่งที่ได้จากโครงการ

ทำให้เข้าใจถึงกระบวนการทำงานของเว็บเซอร์วิสเทคโนโลยีต่างๆ รวมถึงมาตรฐานสากล เพื่อที่จะสื่อสารข้อมูลระหว่างเว็บเซอร์วิสด้วยกันอย่างถูกต้อง และศึกษาเกี่ยวกับเทคโนโลยี .Net ซึ่งเป็นเทคโนโลยีที่ใช้ในการพัฒนาเว็บเซอร์วิส และการเรียนรู้เกี่ยวกับการสร้างและการส่งเอกสาร XML เพื่อใช้ในการติดต่อกับเว็บเซอร์วิส เนื่องจาก XML เป็นภาษาที่ใช้เป็นมาตรฐานและสามารถนำไปใช้กับการประมวลผลของแอปพลิเคชันได้ เนื่องจากเว็บเซอร์วิสที่ได้ทำการพัฒนาเป็นเว็บเซอร์วิสเกี่ยวกับระบบสารสนเทศทางภูมิศาสตร์ จึงต้องทำการศึกษาเรียนรู้เกี่ยวกับระบบสารสนเทศทางภูมิศาสตร์ และการจัดการทำฐานข้อมูลทางภูมิศาสตร์โดยจะใช้ PostgreSQL/PostGIS

5.3 ปัญหาและอุปสรรค

1. การศึกษาเกี่ยวกับมาตรฐานต่างๆ ในส่วนของเว็บเซอร์วิส XML และ PostgreSQL ซึ่งในแต่ละส่วนมีมาตรฐานหลายอย่าง ทำให้ต้องใช้เวลาในการศึกษามากพอสมควร
2. การนำข้อมูลจาก PostgreSQL มาใช้ใน Microsoft Visual Studio 2005 บางส่วนไม่สามารถนำมาใช้งานได้เนื่องจากจำนวนตัวอักษรมีความยาวเกินกว่าที่จะสามารถนำมาใช้งานได้
3. Control SVG ไม่สามารถรับ mouse even ได้ทำให้ไม่สามารถตอบสนองกับภาพได้ จึงทำให้การใช้งานสำหรับบริการบางอย่างไม่ได้รับความสะดวกอย่างเต็มที่

5.4 แนวทางการพัฒนาต่อ

1. สามารถเพิ่มเลขอร์และข้อมูลลงในฐานข้อมูลเพื่อขยายขอบเขตของแผนที่ในการให้บริการ
2. หาเว็บเซอร์วิสอื่นทั้งที่เกี่ยวข้องกับภูมิศาสตร์ และไม่เกี่ยวข้องกับภูมิศาสตร์เพื่อติดต่อและร่วมมือในการให้บริการต่างๆ ทำให้บริการมีความสามารถเพิ่มมากขึ้น

บรรณานุกรม

- [1] David Hunter, Curt Cagle, Dave Gibbons, Nikola Ozu, Jon Pinnock, and Paul Spencer; “คัมภีร์การใช้ XML ฉบับสมบูรณ์”, “บริษัท ซีเอ็ดดูเคชั่น จำกัด”, 2545
- [2] วิสุทธิ แซ่ตั้ง; “Open Source DBMS PostgreSQL”, “สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น)”, 2546
- [3] [Http://www.umn.edu/](http://www.umn.edu/)
- [4] <http://www.wsiam.com/>
- [5] <http://www.postgresql.org/>
- [6] <http://www.developerfusion.co.uk/>
- [7] <http://www.gis2me.com/>
- [8] <http://www.p2p.wrox.com/>
- [9] <http://www.adobe.com/>
- [10] http://kevlindev.com/dom/asv_sharp
- [11] <http://www.opengeospatial.org>
- [12] <http://www.microsoft.com>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

WSDL อธิบายบริการบน GIS เว็บเซอร์วิส

```

<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tm="http://microsoft.com/wsdl/mime/textMatching/"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
xmlns:tns="http://tempuri.org/"
xmlns:s="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
targetNamespace="http://tempuri.org/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:types>
    <s:schema elementFormDefault="qualified"
targetNamespace="http://tempuri.org/">
      <s:element name="GetCapabilities">
        <s:complexType />
      </s:element>
      <s:element name="GetCapabilitiesResponse">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1"
name="GetCapabilitiesResult" type="s:string" />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="GetMap">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="request"
type="s:string" />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="GetMapResponse">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1"
name="GetMapResult">
              <s:complexType mixed="true">
                <s:sequence>
                  <s:any />
                </s:sequence>
              </s:complexType>
            </s:element>
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="CreateMap">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="request"
type="s:string" />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:schema>
  </wsdl:types>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

</s:element>
<s:element name="CreateMapResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1"
name="CreateMapResult">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="Find">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="name"
type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="layer"
type="tns:ArrayOfString" />
      <s:element minOccurs="1" maxOccurs="1" name="startX"
type="s:double" />
      <s:element minOccurs="1" maxOccurs="1" name="startY"
type="s:double" />
      <s:element minOccurs="1" maxOccurs="1" name="width"
type="s:double" />
      <s:element minOccurs="1" maxOccurs="1" name="height"
type="s:double" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:complexType name="ArrayOfString">
  <s:sequence>
    <s:element minOccurs="0" maxOccurs="unbounded" name="string"
nillable="true" type="s:string" />
  </s:sequence>
</s:complexType>
<s:element name="FindResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="FindResult"
type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="FindAddress">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="address"
type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="street"
type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="district"
type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="province"
type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="postcode"
type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    </s:complexType>
  </s:element>
  <s:element name="FindAddressResponse">
    <s:complexType>
      <s:sequence>
        <s:element minOccurs="0" maxOccurs="1"
name="FindAddressResult" type="s:string" />
      </s:sequence>
    </s:complexType>
  </s:element>
  <s:element name="FindNearby">
    <s:complexType>
      <s:sequence>
        <s:element minOccurs="1" maxOccurs="1" name="CoordinateX"
type="s:double" />
        <s:element minOccurs="1" maxOccurs="1" name="CoordinateY"
type="s:double" />
        <s:element minOccurs="1" maxOccurs="1" name="radius"
type="s:double" />
        <s:element minOccurs="0" maxOccurs="1" name="request"
type="s:string" />
        <s:element minOccurs="1" maxOccurs="1" name="showText"
type="s:boolean" />
        <s:element minOccurs="1" maxOccurs="1" name="startX"
type="s:double" />
        <s:element minOccurs="1" maxOccurs="1" name="startY"
type="s:double" />
        <s:element minOccurs="1" maxOccurs="1" name="vwidth"
type="s:double" />
        <s:element minOccurs="1" maxOccurs="1" name="vheight"
type="s:double" />
      </s:sequence>
    </s:complexType>
  </s:element>
  <s:element name="FindNearbyResponse">
    <s:complexType>
      <s:sequence>
        <s:element minOccurs="0" maxOccurs="1"
name="FindNearbyResult">
          <s:complexType mixed="true">
            <s:sequence>
              <s:any />
            </s:sequence>
          </s:complexType>
        </s:element>
      </s:sequence>
    </s:complexType>
  </s:element>
  <s:element name="FindNearest">
    <s:complexType>
      <s:sequence>
        <s:element minOccurs="1" maxOccurs="1" name="CooridinateX"
type="s:double" />
        <s:element minOccurs="1" maxOccurs="1" name="CoordinateY"
type="s:double" />
        <s:element minOccurs="0" maxOccurs="1" name="name"
type="s:string" />
        <s:element minOccurs="0" maxOccurs="1" name="layer"
type="s:string" />
      </s:sequence>
    </s:complexType>
  </s:element>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

</s:element>
<s:element name="FindNearestResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1"
name="FindNearestResult" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="CalculateDistance">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="beginX"
type="s:double" />
      <s:element minOccurs="1" maxOccurs="1" name="beginY"
type="s:double" />
      <s:element minOccurs="1" maxOccurs="1" name="endX"
type="s:double" />
      <s:element minOccurs="1" maxOccurs="1" name="endY"
type="s:double" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="CalculateDistanceResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1"
name="CalculateDistanceResult" type="s:double" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="ConvertAddressToCoordinate">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="address"
type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="street"
type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="district"
type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="province"
type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="postcode"
type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="ConvertAddressToCoordinateResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1"
name="ConvertAddressToCoordinateResult" type="tns:ArrayOfDouble" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:complexType name="ArrayOfDouble">
  <s:sequence>
    <s:element minOccurs="0" maxOccurs="unbounded" name="double"
type="s:double" />
  </s:sequence>
</s:complexType>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    </s:schema>
  </wsdl:types>
  <wsdl:message name="GetCapabilitiesSoapIn">
    <wsdl:part name="parameters" element="tns:GetCapabilities" />
  </wsdl:message>
  <wsdl:message name="GetCapabilitiesSoapOut">
    <wsdl:part name="parameters" element="tns:GetCapabilitiesResponse" />
  </wsdl:message>
  <wsdl:message name="GetMapSoapIn">
    <wsdl:part name="parameters" element="tns:GetMap" />
  </wsdl:message>
  <wsdl:message name="GetMapSoapOut">
    <wsdl:part name="parameters" element="tns:GetMapResponse" />
  </wsdl:message>
  <wsdl:message name="CreateMapSoapIn">
    <wsdl:part name="parameters" element="tns:CreateMap" />
  </wsdl:message>
  <wsdl:message name="CreateMapSoapOut">
    <wsdl:part name="parameters" element="tns:CreateMapResponse" />
  </wsdl:message>
  <wsdl:message name="FindSoapIn">
    <wsdl:part name="parameters" element="tns:Find" />
  </wsdl:message>
  <wsdl:message name="FindSoapOut">
    <wsdl:part name="parameters" element="tns:FindResponse" />
  </wsdl:message>
  <wsdl:message name="FindAddressSoapIn">
    <wsdl:part name="parameters" element="tns:FindAddress" />
  </wsdl:message>
  <wsdl:message name="FindAddressSoapOut">
    <wsdl:part name="parameters" element="tns:FindAddressResponse" />
  </wsdl:message>
  <wsdl:message name="FindNearbySoapIn">
    <wsdl:part name="parameters" element="tns:FindNearby" />
  </wsdl:message>
  <wsdl:message name="FindNearbySoapOut">
    <wsdl:part name="parameters" element="tns:FindNearbyResponse" />
  </wsdl:message>
  <wsdl:message name="FindNearestSoapIn">
    <wsdl:part name="parameters" element="tns:FindNearest" />
  </wsdl:message>
  <wsdl:message name="FindNearestSoapOut">
    <wsdl:part name="parameters" element="tns:FindNearestResponse" />
  </wsdl:message>
  <wsdl:message name="CalculateDistanceSoapIn">
    <wsdl:part name="parameters" element="tns:CalculateDistance" />
  </wsdl:message>
  <wsdl:message name="CalculateDistanceSoapOut">
    <wsdl:part name="parameters"
  element="tns:CalculateDistanceResponse" />
  </wsdl:message>
  <wsdl:message name="ConvertAddressToCoordinateSoapIn">
    <wsdl:part name="parameters"
  element="tns:ConvertAddressToCoordinate" />
  </wsdl:message>
  <wsdl:message name="ConvertAddressToCoordinateSoapOut">
    <wsdl:part name="parameters"
  element="tns:ConvertAddressToCoordinateResponse" />
  </wsdl:message>
  <wsdl:portType name="ServiceSoap">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<wsdl:operation name="GetCapabilities">
  <wsdl:input message="tns:GetCapabilitiesSoapIn" />
  <wsdl:output message="tns:GetCapabilitiesSoapOut" />
</wsdl:operation>
<wsdl:operation name="GetMap">
  <wsdl:input message="tns:GetMapSoapIn" />
  <wsdl:output message="tns:GetMapSoapOut" />
</wsdl:operation>
<wsdl:operation name="CreateMap">
  <wsdl:input message="tns:CreateMapSoapIn" />
  <wsdl:output message="tns:CreateMapSoapOut" />
</wsdl:operation>
<wsdl:operation name="Find">
  <wsdl:input message="tns:FindSoapIn" />
  <wsdl:output message="tns:FindSoapOut" />
</wsdl:operation>
<wsdl:operation name="FindAddress">
  <wsdl:input message="tns:FindAddressSoapIn" />
  <wsdl:output message="tns:FindAddressSoapOut" />
</wsdl:operation>
<wsdl:operation name="FindNearby">
  <wsdl:input message="tns:FindNearbySoapIn" />
  <wsdl:output message="tns:FindNearbySoapOut" />
</wsdl:operation>
<wsdl:operation name="FindNearest">
  <wsdl:input message="tns:FindNearestSoapIn" />
  <wsdl:output message="tns:FindNearestSoapOut" />
</wsdl:operation>
<wsdl:operation name="CalculateDistance">
  <wsdl:input message="tns:CalculateDistanceSoapIn" />
  <wsdl:output message="tns:CalculateDistanceSoapOut" />
</wsdl:operation>
<wsdl:operation name="ConvertAddressToCoordinate">
  <wsdl:input message="tns:ConvertAddressToCoordinateSoapIn" />
  <wsdl:output message="tns:ConvertAddressToCoordinateSoapOut" />
</wsdl:operation>
</wsdl:portType>
<wsdl:binding name="ServiceSoap" type="tns:ServiceSoap">
  <soap:binding transport="http://schemas.xmlsoap.org/soap/http" />
  <wsdl:operation name="GetCapabilities">
    <soap:operation soapAction="http://tempuri.org/GetCapabilities"
style="document" />
    <wsdl:input>
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetMap">
    <soap:operation soapAction="http://tempuri.org/GetMap"
style="document" />
    <wsdl:input>
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="CreateMap">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    <soap:operation soapAction="http://tempuri.org/CreateMap"
style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="Find">
    <soap:operation soapAction="http://tempuri.org/Find"
style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="FindAddress">
    <soap:operation soapAction="http://tempuri.org/FindAddress"
style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="FindNearby">
    <soap:operation soapAction="http://tempuri.org/FindNearby"
style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="FindNearest">
    <soap:operation soapAction="http://tempuri.org/FindNearest"
style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="CalculateDistance">
    <soap:operation
soapAction="http://tempuri.org/CalculateDistance" style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ConvertAddressToCoordinate">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    <soap:operation
soapAction="http://tempuri.org/ConvertAddressToCoordinate"
style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
</wsdl:binding>
<wsdl:binding name="ServiceSoap12" type="tns:ServiceSoap">
    <soap12:binding transport="http://schemas.xmlsoap.org/soap/http"
/>
    <wsdl:operation name="GetCapabilities">
        <soap12:operation
soapAction="http://tempuri.org/GetCapabilities" style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetMap">
        <soap12:operation soapAction="http://tempuri.org/GetMap"
style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="CreateMap">
        <soap12:operation soapAction="http://tempuri.org/CreateMap"
style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="Find">
        <soap12:operation soapAction="http://tempuri.org/Find"
style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="FindAddress">
        <soap12:operation soapAction="http://tempuri.org/FindAddress"
style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="FindNearby">
    <soap12:operation soapAction="http://tempuri.org/FindNearby"
style="document" />
    <wsdl:input>
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="FindNearest">
    <soap12:operation soapAction="http://tempuri.org/FindNearest"
style="document" />
    <wsdl:input>
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="CalculateDistance">
    <soap12:operation
soapAction="http://tempuri.org/CalculateDistance" style="document" />
    <wsdl:input>
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ConvertAddressToCoordinate">
    <soap12:operation
soapAction="http://tempuri.org/ConvertAddressToCoordinate"
style="document" />
    <wsdl:input>
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
</wsdl:binding>
<wsdl:service name="Service">
    <wsdl:port name="ServiceSoap" binding="tns:ServiceSoap">
        <soap:address location="http://localhost/WebSite3/Service.asmx"
/>
    </wsdl:port>
    <wsdl:port name="ServiceSoap12" binding="tns:ServiceSoap12">
        <soap12:address
location="http://localhost/WebSite3/Service.asmx" />
    </wsdl:port>
</wsdl:service>
</wsdl:definitions>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้