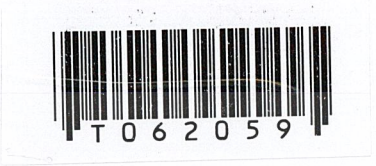
สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เว็บเซอร์วิสเชิงพื้นที่สำหรับระบบสารสนเทศเชิงภูมิศาสตร์

GIS Web Services : Location Based Services



นาย วิศิษฎ์ นวอิทธิพร

นาย สมพล แซ่ปึง

นางสาว สิริวรรณ พรกิตติวัฒนากุล

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เลขหมู่.....
เลขทะเบียน 62059
วัน,เดือน,ปี 27 ก.ค. 2549

b 11608924
.....
.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เว็บเซอร์วิสเชิงพื้นที่สำหรับระบบสารสนเทศเชิงภูมิศาสตร์

GIS Web Services : Location Based Services

โดย

นาย วิศิษฎ์ นวอิทธิพร

นาย สมพล แซ่ปึง

นางสาว สิริวรรณ พรกิตติวัฒนากุล

อาจารย์ที่ปรึกษา

ดร. ชุติเมษฎ์ ศรีนิลทา

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ ปีการศึกษา 2547

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เว็บเซอร์วิสเชิงพื้นที่สำหรับระบบสารสนเทศเชิงภูมิศาสตร์

GIS Web Services : Location Based Services

คณะผู้จัดทำ นาย วิศิษฏ์ นวอิทธิพร รหัส 44010461

นาย สมพล แซ่ปึง รหัส 44010506

นางสาว สิริวรรณ พรกิตติวัฒนากุล รหัส 44010536

ชุติเมษฎ์ ศรีนิลทา อาจารย์ที่ปรึกษา

(ดร. ชุติเมษฎ์ ศรีนิลทา)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เว็บเซอร์วิสเชิงพื้นที่สำหรับระบบสารสนเทศเชิงภูมิศาสตร์

นายวิศิษฎ์ นวอิทธิพร 44010461
นายสมพล แซ่ปึง 44010506
นางสาวสิริวรรณ พรกิตติวัฒนากุล 44010536
ดร.ชุติเมษฎ์ ศรีนิลทา อาจารย์ที่ปรึกษา
ปีการศึกษา 2547

บทคัดย่อ

ปริญญานิพนธ์นี้เป็นการนำเสนอการพัฒนาเว็บเซอร์วิสเชิงพื้นที่สำหรับระบบสารสนเทศเชิงภูมิศาสตร์ที่สามารถให้บริการแผนที่และข้อมูลรายละเอียดของสถานที่ต่างๆ รวมทั้งบริการค้นหาสถานที่ภายในรัศมีขอบเขตที่ต้องการซึ่งมีวัตถุประสงค์เพื่อให้ผู้ที่ต้องการข้อมูลดังกล่าวได้รับข้อมูลที่ถูกต้องและเป็นข้อมูลในปัจจุบันที่สามารถให้ผู้ใช้แก้ไขข้อมูลให้มีความทันสมัยมากที่สุดได้โดยผู้ใช้สามารถเรียกใช้บริการได้จากทุก ๆ แพลตฟอร์มโดยในฐานข้อมูลจะมีการจัดเก็บข้อมูล 2 ประเภท คือ ข้อมูลเชิงพื้นที่และข้อมูลอธิบายโดยข้อมูลเชิงพื้นที่จะถูกนำมาใช้ในการแสดงแผนที่ในรูปแบบ SVG และข้อมูลอธิบายใช้ในการให้รายละเอียดของสถานที่ต่างๆ ประกอบกับข้อมูลเชิงพื้นที่

ในด้านของผู้เรียกใช้บริการได้มีการพัฒนาตัวอย่างโปรแกรมที่มีการเรียกใช้บริการดังกล่าวโดยมีการแสดงผลแผนที่ที่ต้องการจากเว็บเซอร์วิสซึ่งสามารถทำการย่อ ขยาย (Zoom) เลื่อนขอบเขตการแสดงผล (Pan) และแสดงข้อมูลรายละเอียดของสถานที่ต่างๆ ในแผนที่ได้รวมทั้งยังสามารถส่งข้อมูลที่ต้องการปรับปรุงผ่านทางเว็บเซอร์วิสได้อีกด้วย

เทคโนโลยีที่นำมาใช้ในการพัฒนาเว็บเซอร์วิสเชิงพื้นที่สำหรับระบบสารสนเทศเชิงภูมิศาสตร์ในโครงงานนี้ประกอบด้วย Java, Axis, XML, SVG, GPS และ PostGIS ซึ่งเทคโนโลยีเหล่านี้เป็นส่วนสำคัญที่ช่วยในการพัฒนาเว็บเซอร์วิสที่ให้บริการเกี่ยวกับระบบสารสนเทศเชิงภูมิศาสตร์ได้เป็นอย่างดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GIS Web Services : Location Based Services

Mr. Wisid Nawaitthiporn
Mr. Sompon Sae-Pung
Mrs. Siriwan Pornkittiwattanakul
Dr. Chutimet Srinilta Advisor
Academic Year 2004

ABSTRACT

The propose of this thesis was to develop Location-Based Services (LBS) in the form of Web Services. LBS provides access to map and information of places on the map. Features of LBS include proximity search and find closest point of interest. In addition to LBS, we developed data collection upload and update services where user could collect new data, make changes of existing data and finally upload them to the server. We took into account both spatial and non-spatial data. SVG were used to represent output.

The sample of program was also developed for the user by performing the requested map from our Web Services. Zoom and pan were basic map viewing function.User could also search for point of interest, add new data as well as modify existing data.

The technologies that have been adopted in all was done using Web Services we had developed this project are Java, Axis, XML (Extensibles Markup Language), SVG (Scalables Vector Graghics), GPS (Global Positioning System) and PostGIS.All of these technologies are significant factors in the Web Services development for Location Based Services.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้อย่างดี ด้วยคำแนะนำ คำปรึกษาและคอยดูแลจากหลายๆ ฝ่ายด้วยกัน โดยเฉพาะอาจารย์ที่ปรึกษาที่ให้โอกาสให้ข้าพเจ้าได้ทำปริญญานิพนธ์ฉบับนี้ คอยให้ความเอาใจใส่ แนะนำ และให้ความช่วยเหลือเสมอมา คือ ดร.ชุติเมษฎ์ ศรีนิลทาซึ่งต้องขอขอบพระคุณเป็นอย่างสูง

ขอขอบคุณภาควิชาวิศวกรรมคอมพิวเตอร์ที่ได้จัดเตรียมสิ่งอำนวยความสะดวก เพื่อให้การวิจัยและพัฒนาเป็นไปได้ด้วยความสะดวกและรวดเร็ว รวมทั้งยังมีอินเทอร์เน็ตความเร็วสูงให้บริการ สำหรับการค้นคว้าหาความรู้ต่างๆ ซึ่งท้ายที่สุดแล้วก็ประกอบกันเป็นส่วนหนึ่งของโครงงานนี้

ขอขอบคุณพี่ๆ เพื่อนๆ น้องๆ ในห้องปฏิบัติการที่คอยสร้างความคึกคึ้นยามอยู่ในห้องเป็นกำลังใจเสมอมาในการทำงาน

และสุดท้ายต้องขอขอบคุณบุคคลที่สำคัญที่สุดในชีวิตที่ทำให้ข้าพเจ้ามีวันนี้ นั่นคือ บิดา มารดา และบุคคลในครอบครัว อันเป็นที่เคารพรัก ซึ่งได้เลี้ยงดู คอยสั่งสอนข้าพเจ้ามาเป็นอย่างดี พร้อมให้โอกาสในการศึกษาอย่างเต็มที่และยังให้กำลังใจ ความรักเสมอมา ข้าพเจ้าขอกราบขอบพระคุณมา ณ ที่นี้ด้วย

วิศิษฎ์ นวอิทธิพร
สมพล แซ่ปึง
สิริวรรณ พรกิตติวัฒนากุล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้าที่
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VIII
สารบัญภาพ	IX
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญ	1
1.2 วัตถุประสงค์ของปริญญานิพนธ์	1
1.3 ขอบเขตของการพัฒนา	2
1.4 วิธีการดำเนินงาน	3
1.5 ผลที่คาดว่าจะได้รับ	3
บทที่ 2 ทฤษฎี	4
2.1 ความรู้เบื้องต้นของระบบสารสนเทศภูมิศาสตร์ (Geographic Information System)	4
2.1.1 องค์ประกอบของระบบสารสนเทศภูมิศาสตร์	4
2.1.2 ความสามารถของระบบสารสนเทศภูมิศาสตร์	6
2.1.3 การทำงานของระบบสารสนเทศภูมิศาสตร์ (GIS Operation)	7
2.1.4 ระบบพิกัดบนแผนที่	7
2.1.4.1 ระบบค่าพิกัดทางภูมิศาสตร์ (Geo-referenced Coordinate System)	8
2.1.4.2 ระบบพิกัดกริด UTM (Universal Transvers Mercator Coordinate System)	9
2.2 ความรู้เบื้องต้นของเว็บเซอร์วิส	10
2.2.1 เว็บเซอร์วิสและเว็บแอพพลิเคชั่น	10
2.2.2 ประโยชน์ของการทำงานร่วมกันระหว่างแอพพลิเคชั่นกับเว็บเซอร์วิส	11
2.2.3 สถาปัตยกรรมของเว็บเซอร์วิส	11
2.2.4 โมเดลของเว็บเซอร์วิส	12
2.2.5 เทคโนโลยีที่ใช้ในการพัฒนาเว็บเซอร์วิส	12
2.2.5.1 XML (The Extensible Markup Language 1.0)	12
2.2.5.2 SOAP (Simple Object Access Protocol)	12
2.2.5.3 WSDL (Web Services Description Language)	14
2.2.5.4 UDDI (Universal Description, Discovery and Integration protocol)	15
2.2.6 โครงสร้างและการทำงานของเว็บเซอร์วิส	17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้าที่

2.3 ความรู้เบื้องต้นของภาษา XML 17
2.3.1 XML (Extensive Markup Language) 17
2.3.2 เอกสาร XML (XML Document) 17
2.3.3 DTD 18
2.3.4 วิธีการเข้าถึงข้อมูลในภาษา XML 18
2.3.4.1 DOM Parsing 18
2.3.4.2 SAX Parsing 19
2.4 ความรู้เบื้องต้นเกี่ยวกับ SVG 19
2.4.1 SVG (Scalables Vector Graphics) 19
2.4.2 รูปแบบโครงสร้างของเอกสาร SVG 20
2.4.3 ประโยชน์ในการใช้ SVG 23
2.4.4 Batik SVG Toolkit 23
2.5 การกำหนดตำแหน่งด้วยระบบจีพีเอส 25
2.5.1 GPS (Global Positioning System) 25
2.5.2 มาตรฐานในการสื่อสารข้อมูล NMEA – 0183 25
บทที่ 3 การออกแบบโครงงาน 27
3.1 โครงสร้างของโครงงาน 27
3.2 การออกแบบส่วนของฐานข้อมูล GIS 28
3.2.1 การจัดเก็บข้อมูลเชิงพื้นที่ด้วย PostGIS 37
3.2.1.1 วิธีการจัดเก็บข้อมูลเชิงภูมิศาสตร์ 38
3.2.2 ฟังก์ชั่นการทำงานของ PostGIS ที่เกี่ยวข้องกับข้อมูลเชิงพื้นที่ 39
3.3 การออกแบบส่วนของ GIS เว็บเซอร์วิส 40
3.3.1 การนำข้อมูลจากฐานข้อมูลมาสร้างเป็นเอกสาร XML 40
3.3.2 การสร้างเว็บเซอร์วิส 43
3.3.3 ส่วนติดต่อกับฐานข้อมูล 43
3.3.4 การนำข้อมูลจากฐานข้อมูลมาสร้างเป็นแผนที่ในรูปแบบภาษา SVG 44
3.3.5 การจัดการกับเอกสาร XML 45
3.4 การออกแบบส่วนของ GIS แอปพลิเคชั่น 45
3.5 การออกแบบบริการต่างๆบนระบบ GIS เว็บเซอร์วิส 45
3.5.1 บริการข้อมูลเลเยอร์ที่มีในฐานข้อมูล 47
3.5.2 บริการข้อมูลรายละเอียดและตำแหน่งที่ตั้งของสถานที่ 47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้าที่

3.5.3 บริการแผนที่ในรูปแบบ SVG 48
3.5.4 บริการค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการ 49
3.5.5 บริการค้นหาสถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุด 51
3.5.6 บริการนำข้อมูลจากผู้ใช้ไปเพิ่มและแก้ไขในฐานข้อมูล 53
บทที่ 4 การออกแบบส่วนของ GIS แอปพลิเคชั่น 54
4.1 โครงสร้างการทำงานของ GIS แอพพลิเคชัน 54
4.1.1 ส่วนแสดงผลแผนที่และเครื่องมือต่างๆในการจัดการกับแผนที่ 54
4.1.2 ส่วนแสดงผลข้อมูลของสถานที่ต่างๆ 54
4.1.3 ส่วนค้นหาพิกัดตำแหน่งของผู้ใช้ 54
4.1.4 ส่วนที่เรียกใช้บริการจาก GIS เว็บเซอร์วิส 55
4.2 การสร้างส่วนการเรียกใช้งานเว็บเซอร์วิส 55
4.3 การเรียกใช้บริการต่างๆจาก GIS เว็บเซอร์วิส 56
4.3.1การเรียกใช้บริการข้อมูลเลเยอร์ที่มีในฐานข้อมูล 56
4.3.2 การเรียกใช้บริการข้อมูลรายละเอียดและตำแหน่งที่ตั้งของสถานที่ 56
4.3.3 การเรียกใช้บริการแผนที่ในรูปแบบ SVG 57
4.3.4 การเรียกใช้บริการค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการ 58
4.3.5 การเรียกใช้บริการค้นหาสถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุด 58
4.3.6 การเรียกใช้บริการนำข้อมูลจากผู้ใช้ไปเพิ่มและแก้ไขในฐานข้อมูล 59
4.3.6.1 การเพิ่มเรคคอร์ดของข้อมูล 59
4.3.6.2 การแก้ไขเรคคอร์ดของข้อมูล 60
4.3.6.3 บันทึกข้อมูลที่เพิ่มและแก้ไขลงไฟล์ 60
บทที่ 5 ผลการทดลอง 61
5.1 ผลการทดลองใช้งานในส่วนของ GIS แอพพลิเคชัน 61
5.1.1 การทดลองเปิดแผนที่ในรูปแบบ SVG 62
5.1.2 การทดลองบันทึกแผนที่ในรูปแบบ SVG 63
5.1.3 การทดลองย่อ ขยายแผนที่ 64
5.1.4 การทดลองเลื่อนแผนที่ 65
5.1.5 การทดลองแสดงผลเฉพาะเลเยอร์ 66
5.1.6 การทดลองการคลิ๊กเมาส์บนแผนที่เพื่อแสดงรายละเอียด 67
5.1.7 การทดลองหาค่าพิกัดของผู้ใช้โดยผ่านเครื่องรับสัญญาณจีพีเอส 68
5.2 ผลการทดลองใช้งานในส่วนของ GIS เว็บเซอร์วิส 69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้าที่

5.2.1. ทดลองเรียกใช้บริการข้อมูลเลเยอร์ที่มีในฐานข้อมูล 71

5.2.2. ทดลองเรียกใช้บริการขอรายละเอียดและตำแหน่งที่ตั้งของสถานที่ 73

5.2.3. ทดลองเรียกใช้บริการแผนที่ในรูปแบบ SVG 73

5.2.4. ทดลองเรียกใช้บริการค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการ 74

5.2.5. ทดลองเรียกใช้บริการค้นหาสถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุด 75

5.2.6. ทดลองเรียกใช้บริการนำข้อมูลจากผู้ใช้ไปเพิ่มและแก้ไขในฐานข้อมูล 76

5.2.6.1 การทดลองเพิ่มเรคคอร์ดของข้อมูล 76

5.2.6.2 การทดลองแก้ไขเรคคอร์ดของข้อมูล 77

5.2.6.3 การทดลองเรียกใช้บริการบันทึกข้อมูลที่เพิ่มและแก้ไขลงไฟล์ 78

บทที่ 6 บทวิจารณ์และสรุป 79

6.1 บทวิจารณ์และสรุป 79

6.2 ปัญหาที่เกิดขึ้นในการพัฒนาและแนวทางการแก้ปัญหา 79

6.3 ข้อจำกัดของระบบ 80

6.4 แนวทางการพัฒนา 80

ภาคผนวก ก 81

บรรณานุกรม 87

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้าที่

ตารางที่ 2-1 แสดงอิลิเมนต์เบื้องต้นของเอกสาร WSDL 15
ตารางที่ 2-2 แสดงความหมายในแต่ละฟิลด์ของโปรโตคอล NMEA-0183 26
ตารางที่ 3-1 แสดงรายละเอียดของตาราง atm 29
ตารางที่ 3-2 แสดงรายละเอียดของตาราง gatm 29
ตารางที่ 3-3 แสดงรายละเอียดของตาราง bank 30
ตารางที่ 3-4 แสดงรายละเอียดของตาราง gbank 30
ตารางที่ 3-5 แสดงรายละเอียดของตาราง canteen 31
ตารางที่ 3-6 แสดงรายละเอียดของตาราง gcanteen 31
ตารางที่ 3-7 แสดงรายละเอียดของตาราง library 32
ตารางที่ 3-8 แสดงรายละเอียดของตาราง glibrary 32
ตารางที่ 3-9 แสดงรายละเอียดของตาราง carpark 33
ตารางที่ 3-10 แสดงรายละเอียดของตาราง gcarpark 33
ตารางที่ 3-11 แสดงรายละเอียดของตาราง foodshop 34
ตารางที่ 3-12 แสดงรายละเอียดของตาราง gfoodshop 34
ตารางที่ 3-13 แสดงรายละเอียดของตาราง phoneboot 35
ตารางที่ 3-14 แสดงรายละเอียดของตาราง gphoneboot 35
ตารางที่ 3-15 แสดงรายละเอียดของตาราง postbox 36
ตารางที่ 3-16 แสดงรายละเอียดของตาราง gpostbox 36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

หน้าที่

รูปที่ 2-1 องค์ประกอบของ GIS 4
รูปที่ 2-2 ลักษณะข้อมูลที่แสดงทิศทาง (vector data) 5
รูปที่ 2-3 ข้อมูลตารางกริด (raster data) 5
รูปที่ 2-4 ขั้นตอนการดำเนินงานด้าน GIS 7
รูปที่ 2-5 Spherical Coordinate System 8
รูปที่ 2-6 Planar Coordinate 8
รูปที่ 2-7 แสดงการแบ่งกริดโซนระบบพิกัดกริด UTM 10
รูปที่ 2-8 โมเดลของเว็บเซอร์วิส 12
รูปที่ 2-9 XML messaging using SOAP 13
รูปที่ 2-10โครงสร้างของเอกสาร SOAP 14
รูปที่ 2-11 เว็บเซอร์วิส กับ SOAP, UDDI และ WSDL 16
รูปที่ 2-12 โครงสร้างการทำงานของเว็บเซอร์วิส 17
รูปที่ 2-13 แสดงวิธีการเข้าถึงข้อมูลโดยวิธี DOM Parsing 18
รูปที่ 2-14 แสดงวิธีการเข้าถึงข้อมูลโดยวิธี SAX Parsing 19
รูปที่ 2-15 โครงสร้างสถาปัตยกรรมของ Batik 24
รูปที่ 3-1 โครงสร้างของระบบ GIS เว็บเซอร์วิส 27
รูปที่ 3-2 ความสัมพันธ์ระหว่างตาราง atm และ gatm 29
รูปที่ 3-3 ความสัมพันธ์ระหว่างตาราง bank และ gbank 30
รูปที่ 3-4 ความสัมพันธ์ระหว่างตาราง canteen และ gcanteen 31
รูปที่ 3-5 ความสัมพันธ์ระหว่างตาราง library และ glibrary 32
รูปที่ 3-6 ความสัมพันธ์ระหว่างตาราง carpark และ gcarpark 33
รูปที่ 3-7 ความสัมพันธ์ระหว่างตาราง loodshop และ gfoodshop 34
รูปที่ 3-8 ความสัมพันธ์ระหว่างตาราง phoneboot hและ gphonebooth 35
รูปที่ 3-9 ความสัมพันธ์ระหว่างตาราง postbox และ gpostbox 36
รูปที่ 3-10 แสดงขั้นตอนการให้บริการข้อมูลเลเยอร์ที่มีในฐานข้อมูล 47
รูปที่ 3-11 แสดงขั้นตอนการให้บริการข้อมูลรายละเอียดและตำแหน่งที่ตั้งของสถานที่ 48
รูปที่ 3-12 แสดงขั้นตอนการให้บริการแผนที่ในรูปแบบ SVG 49
รูปที่ 3-13 แสดงการขยายขอบเขตของพื้นที่ออกเป็นสี่เหลี่ยมจัตุรัสในรัศมีที่ต้องการ 50
รูปที่ 3- 14 แสดงการขยายขอบเขตของพื้นที่ออกเป็นวงกลมในพื้นที่สี่เหลี่ยมจัตุรัสในรัศมีที่ต้องการ 51
รูปที่ 3-15 แสดงขั้นตอนการให้บริการค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการ 51
รูปที่ 3-16 แสดงตำแหน่งที่อยู่ใกล้กับจุดอ้างอิงมากที่สุด 52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

หน้าที่

รูปที่ 3-17 แสดงขั้นตอนการให้บริการค้นหาสถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุด 53
รูปที่ 3-18 แสดงขั้นตอนการให้บริการนำข้อมูลจากผู้ใช้ไปเพิ่มและแก้ไขในฐานข้อมูล 53
รูปที่ 4-1 แสดงขั้นตอนการเรียกใช้บริการข้อมูลเลเยอร์ที่มีในฐานข้อมูล 56
รูปที่ 4-2 แสดงขั้นตอนการเรียกใช้บริการข้อมูลรายละเอียดและตำแหน่งที่ตั้งของสถานที่ 57
รูปที่ 4-3 แสดงขั้นตอนการเรียกใช้บริการแผนที่ในรูปแบบ SVG 57
รูปที่ 4-4 แสดงการเรียกใช้บริการค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการ 58
รูปที่ 4-5 แสดงขั้นตอนการเรียกใช้บริการค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการ 58
รูปที่ 4-6 แสดงการเรียกใช้บริการค้นหาสถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุด 59
รูปที่ 4-7 แสดงขั้นตอนการเรียกใช้บริการค้นหาสถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุด 59
รูปที่ 4-8 แสดงขั้นตอนการเรียกใช้บริการนำข้อมูลจากผู้ใช้ไปเพิ่มและแก้ไขในฐานข้อมูล 60
รูปที่ 5-1 แสดงหน้าจอแรกของ GIS แอพพลิเคชัน 61
รูปที่ 5-2 แสดงหน้าจอส่วนแสดงผลแผนที่ 62
รูปที่ 5-3 แสดงหน้าจอส่วนบันทึกแผนที่ 63
รูปที่ 5-4 แสดงหน้าจอส่วนย่อและขยายแผนที่ 64
รูปที่ 5-5 แสดงหน้าจอส่วนเลื่อนแผนที่ 65
รูปที่ 5-6 แสดงหน้าจอส่วนแสดงผลแผนที่ในแต่ละเลเยอร์ 66
รูปที่ 5-7 แสดงหน้าจอส่วนแสงรายละเอียดเมื่อมีการคลิ้กเม้าส์บนแผนที่ 67
รูปที่ 5-8 แสดงหน้าจอส่วนการหาค่าพิกัดของผู้ใช้โดยผ่านเครื่องรับสัญญาณจีพีเอส 68
รูปที่ 5-9 แสดงหน้าเว็บเพื่อเข้าดูบริการต่างๆที่เปิดบน GIS เว็บเวอร์วิส(1) 69
รูปที่ 5-10 แสดงหน้าเว็บเพื่อเข้าดูบริการต่างๆที่เปิดบน GIS เว็บเซอร์วิส(2) 70
รูปที่ 5-11 แสดงหน้าเว็บที่แสดง WSDLเพื่อเข้าดูบริการต่างๆที่เปิดบน GIS เว็บเซอร์วิส 70
รูปที่ 5-12 แสดงหน้าจอในส่วนของการใช้บริการข้อมูลเลเยอร์ในฐานข้อมูล 71
รูปที่ 5-13 แสดงหน้าจอ TCP Monitor 72
รูปที่ 5-14 แสดงหน้าจอรายชื่อเลเยอร์ที่ได้รับการเลือก 72
รูปที่ 5-15 แสดงหน้าจอข้อมูลของเลเยอร์ต่างๆ 73
รูปที่ 5-16 แสดงส่วนเรียกใช้บริการค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการ 75
รูปที่ 5-17 แสดงข้อมูลที่ได้จากการเรียกใช้บริการค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการ 75
รูปที่ 5-18 แสดงส่วนเรียกใช้บริการค้นหาสถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุด 76
รูปที่ 5-19 แสดงข้อมูลที่ได้จากการเรียกใช้บริการค้นหาสถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุด 76

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

หน้าที่

รูปที่ 5-20 แสดงหน้าจอเพิ่มเรคคอร์ดของข้อมูล 77
รูปที่ 5-21 แสดงหน้าจอแก้ไขเรคคอร์ดของข้อมูล 77
รูปที่ 5-22 แสดงหน้าจอบันทึกข้อมูลที่เพิ่มและแก้ไขลงไฟล์ 78

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญ

ในปัจจุบันระบบสารสนเทศภูมิศาสตร์ (Geometry Information System: GIS) เริ่มมีใช้งานกันอย่างแพร่หลาย ตัวอย่างเช่น ทางการทหาร การคมนาคม การสำรวจทรัพยากรธรรมชาติและธุรกิจ เป็นต้น ระบบสารสนเทศภูมิศาสตร์เป็นกระบวนการทำงานเกี่ยวกับข้อมูลในเชิงพื้นที่ด้วยระบบคอมพิวเตอร์ ที่ใช้กำหนดข้อมูลและสารสนเทศที่มีความสัมพันธ์กับตำแหน่งในเชิงพื้นที่ ซึ่งรูปแบบและความสัมพันธ์ของข้อมูลเชิงพื้นที่ทั้งหลายจะสามารถนำมาวิเคราะห์ ซึ่งถือว่าเป็นข้อได้เปรียบที่เด่นชัดเมื่อเทียบกับการใช้แผนที่ตามปกติ

เดิมเมื่อองค์กรต้องการที่จะได้ข้อมูลจากระบบสารสนเทศเชิงภูมิศาสตร์ก็จะต้องพัฒนาระบบของตนและเก็บข้อมูลต่างๆไว้ที่ตนเอง เมื่อส่วนต่างๆภายในองค์กรหรือองค์กรอื่นต้องการใช้ระบบก็จะต้องปฏิบัติตามรูปแบบที่ได้ออกแบบระบบนั้นไว้ซึ่งเป็นสิ่งที่ทำได้ยากที่จะให้ผู้คนอื่นทราบและเข้าใจระบบที่ออกแบบไว้และอาจทำให้องค์กรอื่นตัดสินใจที่จะพัฒนาระบบขึ้นมาเป็นของตนเองทำให้มีการเก็บข้อมูลไว้หลายที่เกินความจำเป็นซึ่งข้อมูลแต่ละที่อาจมีความคลาดเคลื่อนได้

เราสามารถใช้เว็บเซอร์วิสมาช่วยในการแก้ไขปัญหาเหล่านี้ได้เนื่องจากหากเราพัฒนาระบบสารสนเทศเชิงภูมิศาสตร์โดยให้บริการข้อมูลในรูปแบบของเว็บเซอร์วิสก็จะทำให้ผู้ใช้ที่ต้องการใช้ข้อมูลจากระบบเราสามารถเข้าใจบริการของเราได้ง่ายเนื่องจาก เว็บเซอร์วิสมีส่วนที่ใช้อธิบายรูปแบบของบริการที่เป็นมาตรฐานสามารถเข้าใจได้ตรงกัน รูปแบบการติดต่อขอใช้บริการและรับบริการมีการกำหนดเป็นมาตรฐาน อีกทั้งรูปแบบการติดต่อสื่อสารยังสื่อสารกันด้วยโปรโตคอล HTTP ซึ่งเป็นที่นิยมใช้กันอย่างกว้างขวาง และด้วยการพัฒนาระบบในลักษณะนี้จะทำให้โปรแกรมที่อยู่บนแพลตฟอร์มที่ต่างกันสามารถเรียกใช้บริการได้หมด เมื่อผู้ต้องการข้อมูลสามารถเรียกใช้บริการได้จากที่เดียวกันก็จะทำให้สามารถเก็บรวบรวมข้อมูลไว้ที่เดียวได้ ทำให้สามารถแก้ไขข้อมูลให้มีความถูกต้องตรงกับความต้องการได้ง่ายขึ้น

1.2 วัตถุประสงค์ของปริญญานิพนธ์

1. ศึกษาระบบสารสนเทศภูมิศาสตร์ (GIS)
2. ศึกษาหลักการทำงานและส่วนประกอบต่างๆของเว็บเซอร์วิส
3. ออกแบบระบบสารสนเทศภูมิศาสตร์ โดยมีขอบเขตอยู่ในพื้นที่ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
4. ออกแบบเว็บเซอร์วิสเพื่อให้บริการข้อมูลเชิงพื้นที่จากระบบสารสนเทศภูมิศาสตร์
5. สร้างแอปพลิเคชั่นเพื่อทดสอบการเรียกใช้บริการของเว็บเซอร์วิส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของการพัฒนา

ปริญญานิพนธ์นี้มุ่งเน้นการพัฒนาเว็บเซอร์วิสที่ให้บริการเกี่ยวกับระบบสารสนเทศภูมิศาสตร์ โดยมีขอบเขตของข้อมูลอยู่ในบริเวณพื้นที่เขตสถาบันสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังซึ่งการพัฒนาแบ่งออกเป็น 3 ส่วนดังนี้

1. ระบบฐานข้อมูล GIS เป็นส่วนที่ใช้ในการเก็บข้อมูล 2 ประเภท คือ ข้อมูลเชิงพื้นที่ (spatial data) และข้อมูลเชิงอธิบาย (Non-Spatial data) โดยใช้ PostgreSQL เป็น DBMS และได้มีการติดตั้ง PostGIS ซึ่งเป็นส่วนเพิ่มเติมของ PostgreSQL เพื่อเป็นการเพิ่มประสิทธิภาพในการจัดการข้อมูลเชิงพื้นที่ โดยระบบฐานข้อมูลนี้จะติดตั้งอยู่บนระบบปฏิบัติการ LINUX ดิสทริบิวชั่นของ Fedora Core 3 ซึ่งจะมีการจัดเก็บข้อมูลโดยแบ่งเป็นเลเยอร์ต่างๆ ทั้งหมด 8 เลเยอร์เพื่อใช้เป็นข้อมูลในการให้บริการ ดังนี้

- เครื่องถอนเงินอัตโนมัติ
- ธนาคาร
- โรงอาหาร
- ที่จอดรถ
- ร้านค้าประเภทอาหารและเครื่องดื่ม
- ตู้โทรศัพท์สาธารณะ
- ห้องสมุด
- ตู้ไปรษณีย์

2. GIS เว็บเซอร์วิสเซิร์ฟเวอร์ เป็นเว็บเซอร์วิสเชิงพื้นที่สำหรับระบบสารสนเทศเชิงภูมิศาสตร์ (GIS) ซึ่งสภาพแวดล้อมบนเซิร์ฟเวอร์ที่เลือกใช้ในการพัฒนาระบบนี้ผู้จัดทำได้เลือกใช้ Jakarta Tomcat 5.0.27 เป็นเว็บคอนเทนเนอร์ และติดตั้ง Apache Axis 1.1 เป็น SOAP เซิร์ฟเวอร์โดยเว็บคอนเทนเนอร์นี้จะติดตั้งอยู่บนระบบปฏิบัติการของ Microsoft Windows XP โดยจะมีการให้บริการต่างๆดังนี้

- บริการข้อมูลเลเยอร์ที่มีในฐานข้อมูล
- บริการข้อมูลรายละเอียดและตำแหน่งที่ตั้งของสถานที่
- บริการแผนที่ในรูปแบบ SVG
- บริการค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการ
- บริการค้นหาสถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุด
- บริการนำข้อมูลจากผู้ใช้ไปเพิ่มและแก้ไขข้อมูลในฐานข้อมูล

3. GIS แอพพลิเคชั่น เป็นส่วนที่เรียกใช้บริการต่างๆ จากเว็บเซิร์ฟเวอร์ และนำข้อมูลจากการเรียกใช้บริการทั้งข้อมูลรายละเอียด และแผนที่นำมาแสดงผลในแอพพลิเคชั่น การแสดงผลแผนที่นั้นจะใช้ Batik SVG ToolKit1.5.1 ของค่าย Apache โดยจะใช้ API ของ Apache Axis ในการจัดการกับการรับและส่ง SOAP แมสเสจโดย GIS แอพพลิเคชันจะมีความสามารถหลักๆ ดังนี้

- ส่วนแสดงผลแผนที่ และเครื่องมือต่างๆในการจัดการกับแผนที่
  - แสดงแผนที่ในรูปแบบ SVG ที่ร้องขอมาจากเว็บเซอร์วิส
  - สามารถเลือกแสดงเลเยอร์ต่างๆได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สามารถย่อหรือขยายแผนที่
- สามารถเลื่อนขอบเขตการแสดงแผนที่ได้
- แผนที่มีปฏิสัมพันธ์กับผู้ใช้ เช่น สามารถแสดงข้อมูลสถานที่เมื่อคลิกเมาส์บริเวณสถานที่นั้นๆ

- ส่วนแสดงผลข้อมูลของสถานที่ต่างๆ
- ส่วนค้นหาพิกัดตำแหน่งของผู้ใช้
- ส่วนที่เรียกใช้บริการจาก GIS เว็บเซอร์วิส

1.4 วิธีการดำเนินงาน

1. ศึกษาเทคโนโลยีที่เกี่ยวข้องกับโครงงาน
2. กำหนดขอบเขตของโครงงาน
3. ออกแบบโครงสร้างของโครงงาน
4. ทดสอบสร้างการให้บริการและการเรียกใช้บริการเว็บเซอร์วิส ทั่วๆไป
5. ออกแบบฐานข้อมูลที่ใช้เก็บข้อมูลในระบบ
6. สร้างเว็บเซอร์วิสที่ให้บริการต่างๆ
7. สร้างแอปพลิเคชั่นมาเรียกใช้บริการต่างๆ จากเว็บเซอร์วิส
8. ปรับปรุงข้อผิดพลาดของเว็บเซอร์วิสและแอปพลิเคชั่น
9. เพิ่มประสิทธิภาพให้กับแอปพลิเคชั่นเช่น ปรับปรุงให้แผนที่ในรูปแบบ SVG ให้มีปฏิสัมพันธ์กับผู้ใช้มากขึ้นและสามารถรับค่าพิกัดจากเครื่องรับสัญญาณ GPS ได้

1.5 ผลที่คาดว่าจะได้รับ

1. ได้รับความรู้เกี่ยวกับระบบสารสนเทศภูมิศาสตร์
2. ได้รับความรู้เกี่ยวกับส่วนต่างของเว็บเซอร์วิส
3. ได้รับความรู้เกี่ยวกับภาษา XML , SVG
4. ได้รับความรู้เกี่ยวกับการพัฒนาแอปพลิเคชั่นด้วยภาษาจาวา
5. สามารถเปิดให้บริการข้อมูลสารสนเทศภูมิศาสตร์ผ่านเว็บเซอร์วิส

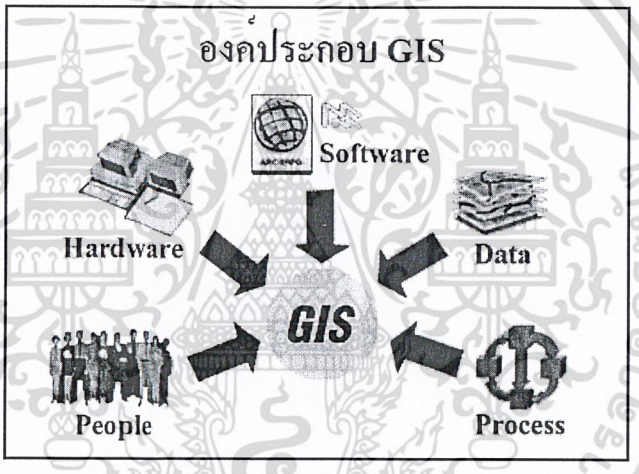
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎี

2.1 ความรู้เบื้องต้นของระบบสารสนเทศภูมิศาสตร์ (Geographic Information System)

ระบบสารสนเทศภูมิศาสตร์ (Geographic Information System : GIS) หมายถึง เครื่องมือที่ใช้ระบบคอมพิวเตอร์ เพื่อใช้ในการนำเข้า จัดเก็บ จัดเตรียม ดัดแปลง แก้ไข จัดการ และวิเคราะห์ พร้อมทั้งแสดงผลข้อมูลเชิงพื้นที่ตามวัตถุประสงค์ต่างๆที่ได้กำหนดไว้ ดังนั้น GIS จึงเป็นเครื่องมือที่มีประโยชน์เพื่อใช้ในการจัดการและบริหารการใช้ทรัพยากรธรรมชาติและสิ่งแวดล้อม และสามารถติดตามการเปลี่ยนแปลงข้อมูลด้านพื้นที่ให้เป็นไปอย่างมีประสิทธิภาพ

2.1.1 องค์ประกอบของระบบสารสนเทศภูมิศาสตร์ (GIS)



รูปที่ 2-1 องค์ประกอบของ GIS

- **ข้อมูล (Data/Information)** ข้อมูลที่จะนำเข้าระบบสารสนเทศภูมิศาสตร์ ควรเป็นข้อมูลเฉพาะเรื่อง(theme) และเป็นข้อมูลที่สามารถนำมาใช้ในการตอบคำถามต่างๆ ได้ตรงตามวัตถุประสงค์ เป็นข้อมูลที่มีความถูกต้องและเชื่อถือได้และเป็นปัจจุบันมากที่สุด ข้อมูลหรือสารสนเทศสามารถแบ่งออกได้เป็น 2 ประเภท ดังนี้

1. ข้อมูลที่มีลักษณะเชิงพื้นที่ (Spatial Data) ข้อมูลเชิงพื้นที่เป็นข้อมูลที่แสดงตำแหน่งที่ตั้งทางภูมิศาสตร์ (geo-referenced data) ของรูปลักษณ์ของพื้นที่ (graphic feature) ซึ่งแบ่งเป็น 2 ลักษณะ ดังนี้

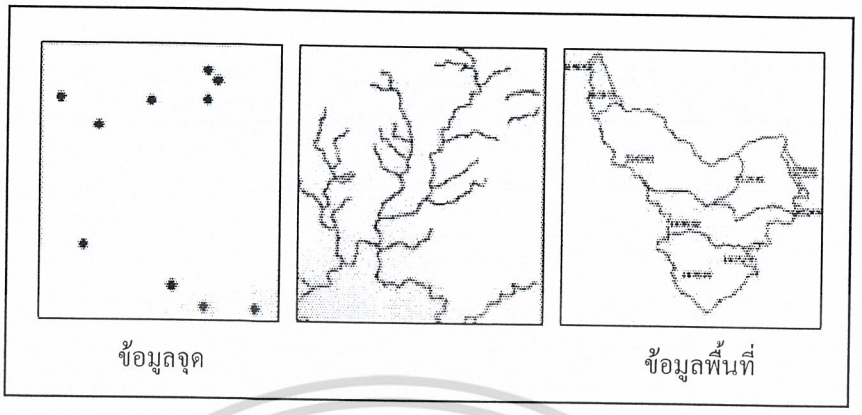
1.1. ข้อมูลที่แสดงทิศทาง (vector data) ประกอบด้วยลักษณะ 3 อย่าง ดังนี้

1.1.1 ข้อมูลจุด (Point)ใช้อ้างอิงถึงตำแหน่งที่ตั้งของสิ่งต่างๆในแผนที่ เช่น โรงเรียน เป็นต้

1.1.2 ข้อมูลเส้น (Arc หรือ Line) เป็นชุดของจุดที่เรียงต่อกันโดยใช้แทนลักษณะที่เป็นเส้น เช่น ถนน เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

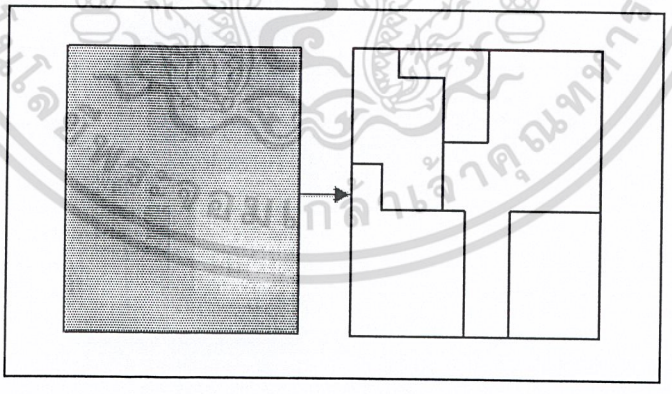
1.1.3 ข้อมูลพื้นที่ หรือเส้นรอบรูป (Polygon) เป็นเส้นรอบรูปปิด ใช้แทนลักษณะที่พื้นที่ เช่น พื้นที่ป่าไม้ ตัวเมือง เป็นต้น



รูปที่ 2-2 ลักษณะข้อมูลที่แสดงทิศทาง (vector data)

1.2. ข้อมูลที่แสดงเป็นตารางกริด (Raster Data)

ข้อมูลที่มีลักษณะเป็นกริด (Raster Data) จะเป็นลักษณะตารางสี่เหลี่ยมเล็กๆ (grid cell or pixel) เท่ากันและต่อเนื่องกัน ซึ่งสามารถอ้างอิงค่าพิกัดทางภูมิศาสตร์ได้ ขนาดของตารางกริดหรือความละเอียด (resolution) ในการเก็บข้อมูลจะใหญ่หรือเล็กขึ้นอยู่กับการจัดแบ่งจำนวนแถว (row) และจำนวนคอลัมน์ (column) ตัวอย่างข้อมูลที่จัดเก็บโดยใช้ตารางกริด เช่น ภาพถ่ายดาวเทียม Land sat หรือข้อมูลระดับค่าความสูง (digital elevation model : DEM) เป็นต้น ดังรูปที่ 2-3



รูปที่ 2-3 ข้อมูลตารางกริด (raster data)

2. ข้อมูลอธิบายพื้นที่ (Non-Spatial Data หรือ Attribute Data) เป็นข้อมูลบอกคุณลักษณะต่างๆของฟีเจอร์ เช่น ชื่อถนน และความกว้างของถนน เป็นต้น

- **เครื่องคอมพิวเตอร์ (Hardware)** เครื่องคอมพิวเตอร์และอุปกรณ์ต่างๆ เครื่องคอมพิวเตอร์รวมกันเรียกว่า ระบบฮาร์ดแวร์ (Hardware) ซึ่งประกอบด้วยคอมพิวเตอร์ และอุปกรณ์นำเข้า เช่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

digitizer scanner อุปกรณ์อ่านข้อมูล เก็บรักษาข้อมูล และแสดงผลข้อมูล เช่น printer plotter เป็นต้น ซึ่งอุปกรณ์แต่ละชนิดจะมีหน้าที่และคุณภาพแตกต่างกันออกไป

- **โปรแกรมหรือระบบซอฟท์แวร์ (Software)** ซอฟท์แวร์ หมายถึง โปรแกรมที่ใช้ในการจัดการระบบ สั่งงานต่างๆ เพื่อให้ระบบฮาร์ดแวร์ทำงาน และสามารถสร้างความสัมพันธ์ระหว่างข้อมูลที่เป็นกราฟฟิกและแอตทริบิว รวมทั้งการเพิ่มเติม แก้ไขข้อมูล และเรียกดึงข้อมูลที่จัดเก็บในระบบฐานข้อมูลมาใช้ได้ง่าย สะดวก และรวดเร็ว อีกทั้งมีความสามารถในการวิเคราะห์ข้อมูลและแสดงผลข้อมูลในรูปแบบที่เข้าใจได้ง่าย เช่น รายงาน ตาราง หรือแผนที่ เป็นต้น โดยทั่วไปชุดคำสั่งหรือโปรแกรมของสารสนเทศทางภูมิศาสตร์จะประกอบด้วยหน่วยนำเข้าข้อมูล หน่วยเก็บข้อมูลและการจัดการข้อมูล หน่วยวิเคราะห์ แสดงผล หน่วยแปลงข้อมูล และหน่วยโต้ตอบกับผู้ใช้
- บุคลากร **(Human Resources)** บุคลากรจะประกอบด้วยผู้ใช้ระบบ (Analyst) และผู้ใช้สารสนเทศ (User)
- กระบวนการ **(Process)** กระบวนการหรือขั้นตอนการทำงานที่ออกแบบไว้สำหรับ GIS ขององค์กรหรืองานนั้นเป็นไปอย่างมีแบบแผน ถูกต้อง และสามารถปฏิบัติงานได้ต่อไป

2.1.2 ความสามารถของระบบสารสนเทศภูมิศาสตร์

ระบบสารสนเทศภูมิศาสตร์มีข้อได้เปรียบมากกว่าการใช้แผนที่ในเรื่องการจัดเก็บข้อมูลเชิงพื้นที่ให้ทันสมัยอยู่เสมอ รวมถึงการรวมข้อมูลเชิงพื้นที่ทั้งหมดให้อยู่ในฐานข้อมูลเดียว ดังนั้น GIS จึงเป็นระบบสารสนเทศที่รวบรวมข้อมูลเชิงพื้นที่ (Spatial Data) และข้อมูลอธิบายต่างๆ (แอตทริบิวต์ Data) ทำให้มีประสิทธิภาพในการวิเคราะห์ข้อมูลในรูปแบบต่างๆ และตอบคำถามเกี่ยวกับความสัมพันธ์ด้านพื้นที่ได้หลายประการทั้งในเรื่องของระยะเวลา และต้นทุนในการจัดทำ ระบบ GIS ช่วยในการวิเคราะห์พื้นที่ในหลายรูปแบบสำหรับแผนงานที่ต่างๆกันเพื่อตอบคำถามประเภท what-if question ซึ่งสามารถแบ่งออกได้เป็น 5 ประเภท ดังนี้

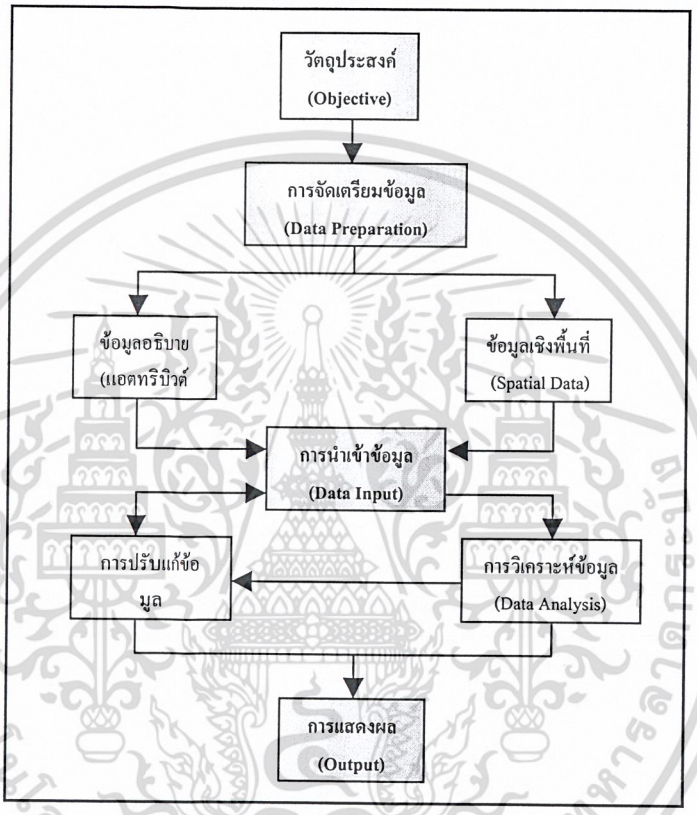
- **Location : What is at...?** มีอะไรอยู่ที่ไหน คำถามแรกที่ GIS สามารถตอบได้ คือ มีอะไรอยู่ที่ไหน หากผู้ถามรู้ตำแหน่งที่แน่นอน เช่น ทราบชื่อหมู่บ้าน ตำบล หรืออำเภอ แต่ต้องการรู้ว่าตำแหน่งนั้นๆ มีรายละเอียดข้อมูลเป็นอย่างไรบ้าง เป็นต้น
- **Condition : Where is it?** สิ่งที่อยากทราบอยู่ที่ไหน และต้องมีการวิเคราะห์ข้อมูลในการค้นหา
- **Trends : What has changed since...?** ในช่วงระยะที่ผ่านมามีอะไรเปลี่ยนแปลงบ้าง คำถามที่สามารถเป็นการวิเคราะห์การเปลี่ยนแปลงในระยะช่วงเวลาใดเวลาหนึ่ง
- **Patterns : What spatial patterns exist?** ความสัมพันธ์ด้านพื้นที่เป็นอย่างไร
- **Modeling : What if...?** จะมีอะไรเกิดขึ้นหาก คำถามนี้จะเกี่ยวข้องกับการคาดการณ์ว่าจะมีอะไรเกิดขึ้นหากปัจจัยอิสระ (independence factor) ซึ่งเป็นตัวกำหนดการเปลี่ยนแปลงไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3 การทำงานของระบบสารสนเทศภูมิศาสตร์ (GIS Operation)

การทำงานของระบบสารสนเทศภูมิศาสตร์ ประกอบด้วย 4 ขั้นตอนหลัก ดังนี้

1. กำหนดวัตถุประสงค์ (Determine Objective)
2. จัดเตรียมข้อมูล (Database Preparation)
3. การนำเข้าข้อมูล (Data Input)
4. แสดงผล (Data Display)



รูปที่ 2-4 ขั้นตอนการดำเนินงานด้าน GIS

2.1.4 ระบบพิกัดบนแผนที่

ระบบพิกัด (Coordinate System) เป็นระบบที่สร้างขึ้นสำหรับใช้อ้างอิงในการกำหนดตำแหน่งหรือระบุตำแหน่งพื้นโลกจากแผนที่ซึ่งมีลักษณะเป็นตารางโครงข่ายที่เกิดจากการตัดกันของเส้นตรงสองชุดที่ถูกกำหนดให้วางตัวในแนวเหนือ-ใต้ และแนวตะวันออก-ตะวันตกตามแนวของจุดศูนย์กำเนิด (Origin) ที่กำหนดขึ้น ค่าพิกัดที่ใช้อ้างอิงในการบอกตำแหน่งต่างๆ จะใช้ค่าของหน่วยที่นับออกจากจุดศูนย์กำเนิดเป็นระยะเชิงมุม (Degree) หรือเป็นระยะทาง (Distance)ไปทางเหนือหรือใต้และตะวันออกหรือตะวันตกตามตำแหน่งของพื้นที่ที่ต้องการหาค่าพิกัดที่กำหนดตำแหน่งต่างๆจะถูกเรียกอ้างอิงเป็นตัวเลขในแนวตั้งและแนวนอนตามหน่วยวัดระยะใช้วัด สำหรับระบบพิกัดที่ใช้อ้างอิงกำหนดตำแหน่งบนแผนที่ที่นิยมใช้กับแผนที่ในปัจจุบันมี 2 ระบบ ได้แก่ ระบบค่าพิกัดทางภูมิศาสตร์และระบบพิกัดกริด

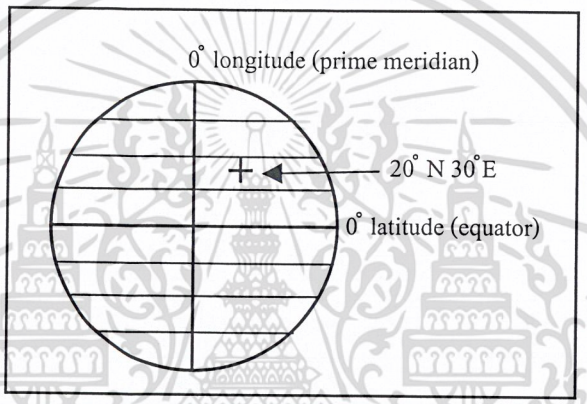
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4.1 ระบบค่าพิกัดทางภูมิศาสตร์ (Geo-referenced Coordinate System)

ระบบค่าพิกัดทางภูมิศาสตร์ สามารถแบ่งออกได้เป็น 2 ระบบใหญ่ๆ ดังนี้

1. Spherical Coordinate System

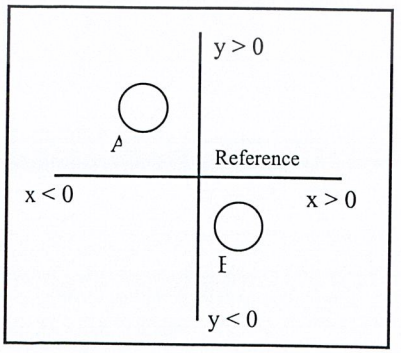
เป็นระบบค่าพิกัดที่อ้างอิงเส้นรุ้ง (latitude) และเส้นแวง (longitude) โดยตั้งอยู่บนพื้นฐานว่าโลกมีลักษณะกลมซึ่งเป็นภาพ 3 มิติ เส้นแวง (longitude or meridians) จะลากจากขั้วโลกเหนือมายังขั้วโลกใต้ เส้นรุ้งบางครั้งเรียกว่า parallels เนื่องจากจะมีระยะห่างที่เท่ากันตลอดเส้นรุ้งที่ลากผ่านเส้นศูนย์สูตร (equator) จะมีค่า 0 องศา เส้นรุ้งที่อยู่ทางทิศเหนือของเส้นศูนย์สูตรจะมีค่าจาก 0-90 องศา ถึงขั้วโลกเหนือ และเส้นรุ้งที่อยู่ทางทิศใต้ของเส้นศูนย์สูตรจะมีค่าจาก 0-(-90) องศา ถึงขั้วโลกใต้ ดังนั้นค่าพิกัดหนึ่งๆของระบบเส้นรุ้ง เส้นแวงจะมีเพียงตำแหน่งเดียวบนพื้นโลก ดังรูปที่ 2-5



รูปที่ 2-5 Spherical Coordinate System

2. Map Projection

เรียกอีกอย่างหนึ่งว่า Cartesian Coordinate or Planar Coordinate System เป็นระบบค่าพิกัดทางภูมิศาสตร์ที่แปลงค่า (projection) เส้นรุ้ง เส้นแวง ที่เป็นรูป 3 มิติ ให้เป็นแผนที่ในลักษณะพื้นราบ 2 มิติ โดยค่า x แทนค่าระยะทางจากจุดอ้างอิงสมมติ ในแนวแกนนอน (horizontal axis) และ y แทนค่าระยะทางจากจุดอ้างอิงสมมติในแนวแกนตั้ง (vertical axis) ดังรูปที่ 2-6



รูปที่ 2-6 Planar Coordinate

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4.2 ระบบพิกัดกริด UTM (Universal Transvers Mercator Coordinate System)

พิกัดกริด UTM (Universal Transvers Mercator) เป็นระบบตารางกริดที่ใช้ช่วยในการกำหนดตำแหน่งและใช้อ้างอิงในการบอกตำแหน่งที่นิยมใช้กับแผนที่ในกิจการทหารของประเทศต่างๆ เกือบทั่วโลกในปัจจุบัน เพราะ เป็นระบบตารางกริดที่มีขนาดรูปร่างเท่ากันทุกตาราง และมีวิธีการกำหนดบอกค่าพิกัดที่ง่ายและถูกต้อง

เป็นระบบกริดที่นำเอาเส้นโครงแผนที่แบบ Universal Transvers Mercator Projection ของ Gauss Krugger มาใช้ดัดแปลงการถ่ายทอดรายละเอียดของพื้นผิวโลกโดยให้รูปทรงกระบอก Mercator Projection อยู่ในตำแหน่ง Mercator Projection (แกนของรูปทรงกระบอกจะทับกับแนวเส้นอิเควเตอร์ และตั้งฉากกับแนวแกนของขั้วโลก)

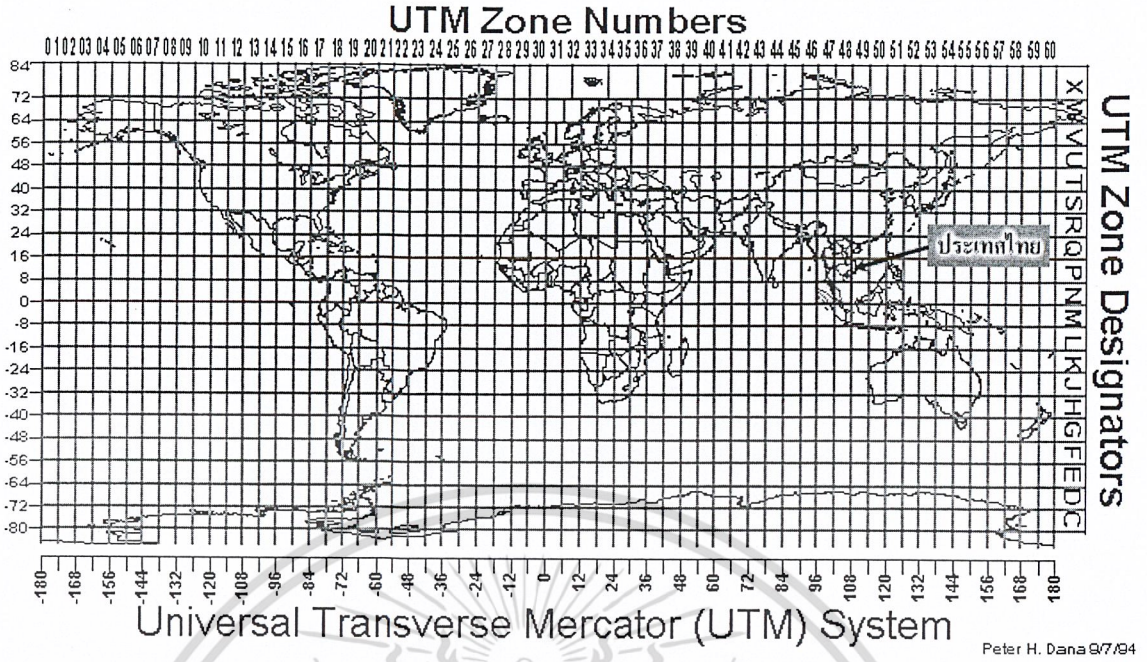
แผนที่ระบบพิกัดกริดที่ใช้เส้นโครงแผนที่แบบ UTM เป็นระบบเส้นโครงชนิดหนึ่งที่ใช้ผิวรูปทรงกระบอกเป็นผิวแสดงเส้นเมริเดียน (หรือเส้นลองกิจูด) และเส้นละติจูดของโลก โดยใช้ทรงกระบอกตัดโลกระหว่างละติจูด 84 องศาเหนือและ 80 องศาใต้ในลักษณะแกนรูปทรงกระบอกทำมุมกับแกนโลก 90 องศารอบโลก แบ่งออกเป็น 60 โซนๆ ละ 6 องศา โดยโซนที่ 1 อยู่ระหว่าง 180 องศา กับ 174 องศาตะวันตก และมีลองกิจูด 177 องศาตะวันตก เป็นเมริเดียนย่านกลาง (Central Meridian) ซึ่งมีเลขกำกับแต่ละโซนจาก 1 ถึง 60 โดยนับจากด้านซ้ายไปยังด้านขวาระหว่างละติจูด 84 องศาเหนือ 80 องศาใต้ แบ่งออกเป็น 2 ช่อง ช่องละ 8 องศา โดยจะยกเว้นช่องสุดท้ายเป็น 12 องศา โดยเริ่มนับตั้งแต่ละติจูด 80 องศาใต้ ขึ้นไปทางเหนือ กำหนดให้ช่องแรกเป็นอักษร C และช่องสุดท้ายเป็นอักษร X (ยกเว้น I และ O) จากการแบ่งตามที่กล่าวแล้วจะเห็นพื้นที่ในเขตลองกิจูด 180 องศาตะวันตกถึง 180 องศาตะวันออก และละติจูด 80 องศาใต้ถึง 84 องศาเหนือ จะถูกแบ่งออกเป็นรูปสี่เหลี่ยมผืนผ้าทั้งหมด 1,200 รูป ซึ่งแต่ละรูปมีขนาดกว้าง 6 องศา ยาว 8 องศา จำนวน 1,140 รูป และกว้าง 6 องศา ยาว 12 องศา จำนวน 60 รูป โดยรูปสี่เหลี่ยมนี้เรียกว่า Grid Zone Designation (GZD)

ตัวอย่างการเรียกชื่อ Grid Zone Designation เช่น ประเทศไทยมีพื้นที่อยู่ ระหว่างละติจูด 5 องศา 30 ลิปดาเหนือ ถึง 20 องศา 30 ลิปดาเหนือ และลองกิจูดประมาณ 97 องศา 30 ลิปดาตะวันออก ถึง 105 องศา 30 ลิปดาตะวันออก ดังนั้นประเทศไทยจึงตกอยู่ใน GZD 47N 47P 47Q 48N 48P และ 48Q เป็นต้น

การอ่านค่าพิกัดกริดเพื่อให้พิกัดค่ากริดในโซนหนึ่งๆ มีค่าเป็นบวกเสมอ จึงกำหนดให้มีศูนย์สมมุติขึ้น 2 แห่ง ดังนี้

- ในบริเวณที่อยู่เหนือเส้นศูนย์สูตร : เส้นศูนย์สูตรมีระยะห่างจากศูนย์สมมุติเท่ากับ 0 เมตร และเส้นเมริเดียนย่านกลางห่างจากศูนย์สมมุติ 500,000 เมตร ทางตะวันออก
- ในบริเวณที่อยู่ใต้เส้นศูนย์สูตร : เส้นศูนย์สูตรมีระยะห่างจากศูนย์สมมุติไปทางเหนือ 10,000,000 เมตร และเมริเดียนย่านกลางห่างจากศูนย์สมมุติ 500,000 เมตร ทางตะวันออก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-7 แสดงการแบ่งกริดโซนระบบพิกัดกริด UTM

2.2 ความรู้เบื้องต้นของเว็บเซอร์วิส

เว็บเซอร์วิส คือ แอพพลิเคชั่น หรือโปรแกรมที่ทำงานอย่างใดอย่างหนึ่งในลักษณะให้บริการ โดยจะถูกเรียกใช้งานจากแอพพลิเคชั่นอื่นๆในรูปแบบ RPC (Remote Procedure Call) ซึ่งการให้บริการจะมีเอกสารที่อธิบายคุณสมบัติของบริการกำกับไว้ โดยภาษาที่ถูกใช้เป็นสื่อในการแลกเปลี่ยนคือ XML ทำให้เราสามารถเรียกใช้คอมโพเน้นท์ใดๆ ก็ได้ ในแพลตฟอร์มใดๆ ก็ได้ บนโปรโตคอล HTTP ซึ่งเป็นโปรโตคอลสำหรับ World Wide Web ซึ่งเป็นช่องทางที่ได้รับการยอมรับทั่วโลกที่ใช้ในการติดต่อสื่อสารกันระหว่างแอพพลิเคชั่นกับแอพพลิเคชั่นในปัจจุบัน

2.2.1 เว็บเซอร์วิสและเว็บแอพพลิเคชั่น

เทคโนโลยีในการกระจายข่าวสารข้อมูลทางอินเทอร์เนตในปัจจุบันคือ เว็บเพจ เนื่องจากเว็บเพจมีความสามารถที่จะทำงานได้ด้วยการรวมภาษาทั้งไคลเอนท์และ Server Side Script ไว้ในตัวเอง เช่น ภาษา VBScript, Java Script หรือ ASP, PHP และ JSP ทำให้เว็บเพจมีลักษณะคล้ายแอพพลิเคชั่น จึงถูกเรียกรวมกันว่าเว็บแอพพลิเคชั่น

เว็บแอพพลิเคชั่นสามารถตอบสนองความคิด Distributed Processing ได้ในระดับหนึ่งซึ่งก็คือการแบ่งการประมวลผลไว้ที่ฝั่งไคลเอนท์และฝั่งเซิร์ฟเวอร์ และส่วนใหญ่มีการใช้ดาตาเบสควบคู่กับการทำเว็บแอพพลิเคชั่นด้วยตามความต้องการในการทำ e-Business และ e-Commerce ที่เป็นที่นิยมในปัจจุบัน

ความเด่นของเทคโนโลยี เว็บเซอร์วิส คือ การทำให้เว็บกับเว็บสามารถติต่อสื่อสารกันได้ด้วยเอกสาร XML ที่ทั้งมนุษย์และคอมพิวเตอร์เข้าใจ และคอมพิวเตอร์ยังสามารถนำข้อมูลนั้นไปประมวลผลต่อได้ ด้วยเอกสาร XML นี้เองทำให้เว็บสามารถส่งข้อมูลที่จำเป็นไปให้อีกเว็บหนึ่งทำงานบางอย่างให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือใช้บริการนั่นเอง แต่สำหรับเว็บแอพพลิเคชั่นที่ใช้การส่งข้อมูลเป็น html ทำให้ข้อมูลนั้นไม่สามารถนำไปใช้งานต่อได้

แนวคิดของเว็บเซอร์วิส คือ เว็บที่สามารถทำงานอะไรบางอย่างซึ่งก็คือให้บริการบางอย่างจากการร้องขอจากต่างเซิร์ฟเวอร์ ด้วยเหตุนี้ทำให้เทคโนโลยีเว็บเซอร์วิสเอื้อต่อแนวคิด Distributed Processing มากกว่าเว็บแอพพลิเคชั่นและเมื่อประกอบกับการที่เว็บเซอร์วิส มี UDDI ทำให้เว็บเซอร์วิสสามารถค้นหาบริการต่างๆ ที่ต้องการได้

2.2.2 ประโยชน์ของการทำงานร่วมกันระหว่างแอพพลิเคชั่นกับเว็บเซอร์วิส

เว็บเซอร์วิสช่วยให้การเข้าถึงข้อมูลสารสนเทศจากแอพพลิเคชันที่ต่างกันเป็นไปโดยง่าย โดยแอพพลิเคชันนั้นๆ สามารถเขียนด้วยจาวาและรันอยู่บน Sun Solaris Application Server หรืออาจจะเขียนด้วย C++ และรันอยู่บน Windows NT หรืออาจะเขียนด้วย Perl และรันอยู่บนเครื่อง Linux ซึ่งมาตรฐานของเว็บเซอร์วิสทำให้อินเทอร์เฟซของแอพพลิเคชันเหล่านี้ ถูกอธิบายโดย WSDL และทำให้อยู่ในมาตรฐานของ UDDI หลังจากนั้น จึงสามารถติดต่อสื่อสารถึงกันโดย XML ผ่าน SOAP อินเตอร์เฟซ

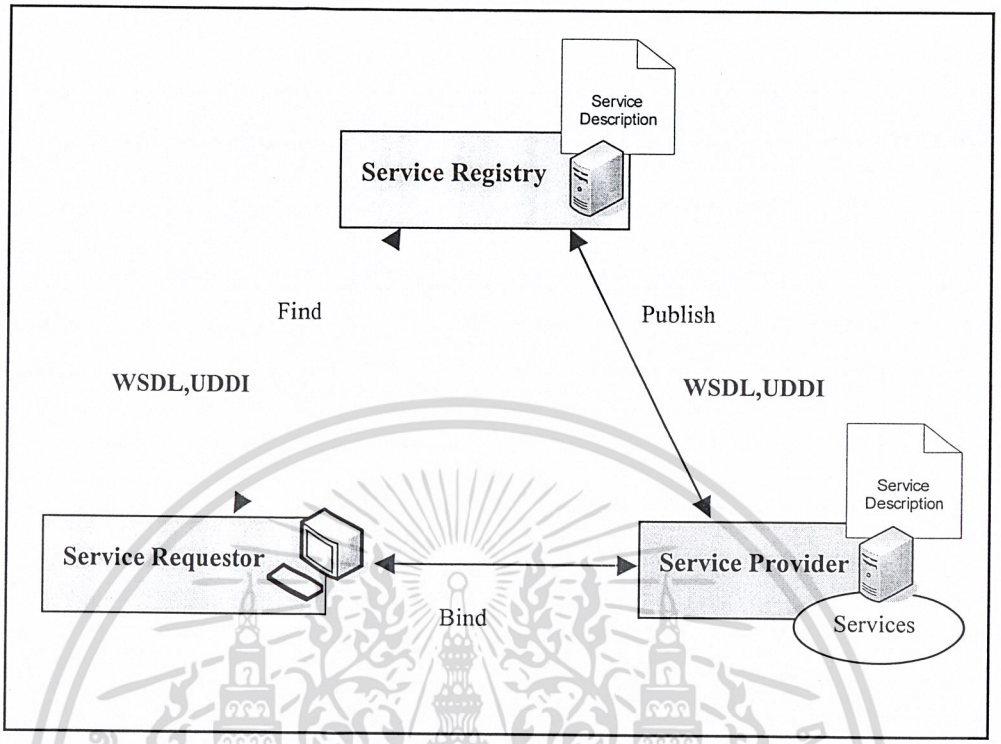
2.2.3 สถาปัตยกรรมของเว็บเซอร์วิส

โครงสร้างสถาปัตยกรรมของระบบที่เน้นการให้บริการเป็นหลัก เรียกว่า **Service - Oriented Architecture : SOA** ประกอบด้วย 3 ส่วนหลัก คือ ผู้ให้บริการ (Service Provider) ผู้ขอบริการ (Service Requester) และตัวแทนของผู้ให้บริการ (Service Broker) ซึ่งส่วนประกอบหลักทั้ง 3 ส่วนนี้ติดต่อถึงกันโดยใช้ฟังก์ชันพื้นฐาน คือ การประกาศ (publish) การค้นหา (find) และการเรียกใช้ (bind)

- **Service Provider** คือ ผู้ให้บริการที่ทำการประกาศ (Publish) บริการขององค์กรไปยังไดเรคทอรี่ที่เก็บทะเบียนของการบริการ
- **Service Requestor** คือ ผู้ขอใช้บริการซึ่งจะทำการค้นหา (find) บริการที่ต้องการ และเมื่อพบเห็นก็จะทำการเรียกใช้ (bind) ไปยังผู้ให้บริการนั้น
- **Service Broker** คือ ตัวแทนของผู้ให้บริการหรือที่อาจเรียกว่า "ไดเรคทอรี่ของบริการ"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4 โมเดลของเว็บเซอร์วิส



รูปที่ 2-8 โมเดลของเว็บเซอร์วิส

- **Requestor** เป็นผู้ที่ต้องการเรียกใช้บริการจาก Provider ซึ่งสามารถค้นหาบริการที่ต้องการได้จาก UDDI registry หรือ Service Registry หรือติดต่อจาก Provider โดยตรง
- **Registry** ทำหน้าที่เป็นตัวกลางให้ Provider มาลงทะเบียนไว้ โดยใช้ WSDL ไฟล์ บอกรายละเอียดของบริษัทและบริการที่มีให้ ซึ่งอาจจะใช้หรือไม่ใช้ก็ได้
- **Provider** เป็นผู้ให้บริการ มีหน้าที่ในการเปิดบริการเพื่อรองรับการขอใช้บริการจาก Requestor ที่เรียกเข้ามาขอใช้บริการ

2.2.5 เทคโนโลยีที่ใช้ในการพัฒนาเว็บเซอร์วิส

2.2.5.1 XML (The Extensible Markup Language 1.0)

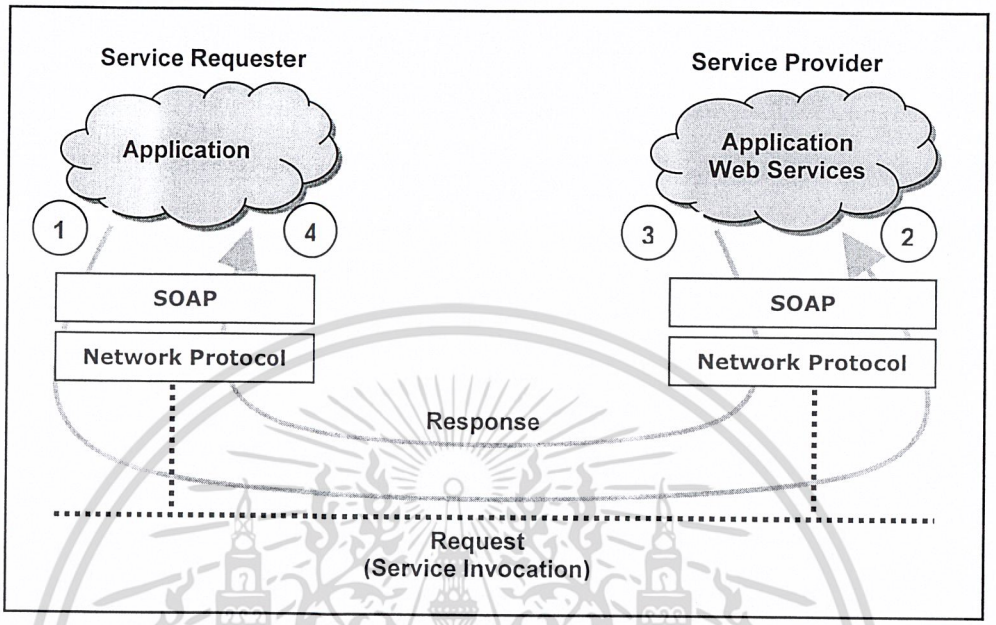
XML เป็นภาษามาร์คอัพที่เป็น text-based ซึ่งทำให้เป็นมาตรฐานในการแลกเปลี่ยนข้อมูลบนอินเทอร์เน็ตอย่างรวดเร็ว ความแตกต่างระหว่าง XML กับ HTML คือ XML เป็นภาษาที่ใช้สำหรับอธิบายข้อมูลต่างๆ แต่ภาษา HTML นั้นใช้ในการแสดงผลเว็บเพจ

2.2.5.2 SOAP (Simple Object Access Protocol)

SOAP เป็นโปรโตคอลที่ใช้ส่งข้อความระหว่างเว็บเซอร์วิส ซึ่ง SOAP เป็น Transport Protocol ที่มี XML เป็นพื้นฐานและใช้ HTTP เป็นโปรโตคอลร่วมในการส่งผ่านเครือข่าย SOAP จะระบุวิธีในการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เข้ารหัสส่วนหัว (Header Encoding) ของทั้ง HTTP และไฟล์ XML ไว้อย่างชัดเจนทั้งในส่วนของการติดต่อไปยังคอมพิวเตอร์อีกเครื่องหนึ่งและส่งผ่านข้อมูลไปให้ รวมถึงระบุวิธีที่โปรแกรมซึ่งถูกเรียกนั้นจะส่งค่าคืนกลับมาด้วย



รูปที่ 2-9 XML messaging using SOAP

จากรูปที่ 2-9 เป็นตัวอย่างแสดงขั้นตอนในการเรียกใช้บริการจากเว็บเซอร์วิสโดยใช้ SOAP โปรโตคอล ทั้งหมด 4 ขั้นตอน ดังนี้

1. แอพพลิเคชันของผู้ร้องขอบริการสร้าง SOAP แมสเสจ เพื่อเรียกใช้บริการของเว็บเซอร์วิส
2. เว็บเซอร์วิสของผู้ให้บริการได้รับ SOAP แมสเสจจากผู้ร้องขอซึ่งอยู่ในรูปแบบเอกสาร XML
3. เว็บเซอร์วิสประมวลผลตามคอมโพเนนต์ที่ให้บริการ จากนั้นเว็บเซอร์วิสส่งผลลัพธ์กลับมา แล้วผู้ให้บริการก็จะสร้าง SOAP แมสเสจที่มีผลลัพธ์นั้นส่งกลับมายังผู้ร้องขอบริการ
4. แอพพลิเคชันของผู้ร้องขอบริการได้รับผลลัพธ์ที่เป็น SOAP แมสเสจ แล้วทำการแปลงให้อยู่ในรูปแบบที่ต้องการเพื่อนำเข้าไปประมวลผลต่อ

เนื่องจากจุดประสงค์หลักของการใช้งานเว็บเซอร์วิสคือต้องการให้แอพพลิเคชั่นมีการทำงานกับแอพพลิเคชั่นที่ทำงานอยู่ในเครื่องอื่นโดยผ่านทางเครือข่าย ซึ่งเทคโนโลยีที่มีอยู่ในปัจจุบันที่ใช้มีการสื่อสารระหว่างวัตถุในระยะไกล (Remote Procedure Calls : RPC) เช่น DCOM, EJB หรือ CORBA นั้นไม่ได้ถูกออกแบบมาให้ใช้สำหรับโปรโตคอล HTTP ดังนั้นเทคนิค RPC ของเทคโนโลยีที่กล่าวข้างต้นนั้นต่างก็มีปัญหาในด้านการนำมาใช้งานในแง่ของความเข้ากันได้ของการเรียกใช้งานข้ามเทคโนโลยี เนื่องจากเป็นเทคโนโลยีเฉพาะของแต่ละค่าย ยกเว้น CORBA ที่ผู้พัฒนาระบบจะต้องพัฒนาโปรแกรมที่มีความซับซ้อนและยังมีปัญหาในส่วนของไฟร์วอลล์และพรอคซีเซิร์ฟเวอร์ด้วยเนื่องจากโดยปกติเซิร์ฟเวอร์จะปิดการสื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

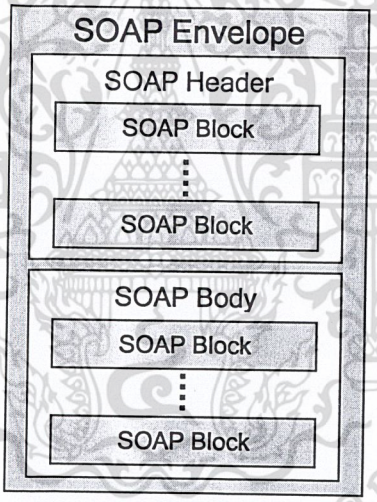
สารที่ไม่ใช่โปรโตคอล HTTP ออกไป เพื่อความปลอดภัยของระบบที่มีการติดต่อสื่อสารกับภายนอก

ดังนั้นทางเลือกของการสื่อสารที่จะนำมาใช้ในการทำบริการเว็บเซอร์วิส ก็คือให้ทำงานอยู่บน HTTPโปรโตคอล ซึ่ง SOAP นอกจากจะทำงานบนโปรโตคอล HTTP แล้วยังเป็นมาตรฐานเปิดที่จะทำให้สามารถติดต่อสื่อสารกับเครื่องคอมพิวเตอร์ที่มีความแตกต่างกันทั้งระบบปฏิบัติการ เทคโนโลยี รวมไปถึงภาษาที่ใช้ในการพัฒนาด้วยได้

- **โครงสร้างของเอกสาร SOAP**

เอกสาร SOAP นั้นมีโครงสร้างในรูปแบบ XML ซึ่งเราสามารถแบ่งเป็นส่วนของเอกสารได้เป็น 3 ส่วนหลักดังนี้คือ

1. SOAP envelope เป็นส่วนเนื้อหาสาระ (Content) ของเอกสารทั้งหมด
2. SOAP header ส่วนเพิ่มเติมของเอกสาร SOAP ซึ่งจะมีก็ได้ หรือไม่มีก็ได้
3. SOAP body ส่วนที่ใช้ในการเรียกใช้งานเซอร์วิส และผลลัพธ์ที่ได้จากเซอร์วิส



รูปที่ 2-10 โครงสร้างของเอกสาร SOAP

2.2.5.3 WSDL(Web Services Description Language)

WSDL เป็นภาษาที่ใช้อธิบายคุณลักษณะการใช้บริการของเว็บเซอร์วิสและวิธีการติดต่อกับเว็บเซอร์วิส ความต้องการของนิยามนี้เกี่ยวเนื่องกับความต้องการของ Distributed System ที่จะกำหนด Interface Definition Language (IDL) โดยใช้ภาษา XML ซึ่ง 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 เป็นภาษาที่อยู่ในการดูแลของ W3C (World Wide Web Consortium) ซึ่งยังไม่เป็นมาตรฐานที่สมบูรณ์ เวอร์ชันที่ใช้งานอยู่ใน ปัจจุบันคือ WSDL 1.1 WSDL คือ มาตรฐานสำหรับการประกาศกระบวนการที่จำเป็นในการเรียกใช้เซอร์วิส SOAP (Simple Object Access Protocol)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **โครงสร้างเอกสาร WSDL**

ในการใช้งานจริงหากเราสร้างบริการ Web Services ก็จะมีเครื่องมือช่วยสร้างเอกสาร WSDL สำหรับเว็บเซอร์วิสอย่างอัตโนมัติ โดยโครงสร้างหลักภายในเอกสาร WSDL ที่ควรรู้เกี่ยวกับการติดต่อและเรียกใช้บริการของเว็บเซอร์วิสมีดังนี้

Element	Definition
<portType>	เป็นส่วนที่สำคัญที่สุดใน WSDL ซึ่งเป็นอิลิเมนต์อธิบายโอเปอเรชั่นที่เว็บเซอร์วิส มีให้บริการและแมสเสจที่เกี่ยวข้องเทียบได้กับ ฟังก์ชั่นไลบารีหรือโมดูลหรือคลาสในการเขียนโปรแกรม
<operation>	อธิบายวิธีที่ให้บริการ โดยเว็บเซอร์วิสหนึ่งจะมีวิธีจำนวนกี่วิธีก็ได้
<message>	อธิบายดาต้าอิลิเมนต์ของโอเปอเรชันแต่ละแมสเสจอาจมีมากกว่าหนึ่งส่วนเทียบได้กับพารามิเตอร์ของฟังก์ชั่นในการเขียนโปรแกรม
<binding>	อธิบายฟอร์แมตของแมสเสจและรายละเอียดของโปรโตคอลในแต่ละพอร์ต
<service>	สำหรับเว็บเซิร์ฟเวอร์จะมีเว็บเซอร์วิสจำนวนกี่บริการก็ได้และชื่อเว็บเซอร์วิสก็เป็นตัวจำแนกและบ่งบอกแต่ละบริการซึ่งห้ามมีชื่อซ้ำกัน

ตารางที่ 2-1 แสดงอิลิเมนต์เบื้องต้นของเอกสาร WSDL

ตามทฤษฎีแล้ว ไฟล์เอกสาร WSDL แต่ละไฟล์ สามารถอธิบายคุณลักษณะของบริการเว็บเซอร์วิสได้มากกว่าหนึ่งบริการ โดยแต่ละเว็บเซอร์วิสจะมีพอร์ตสื่อสารเฉพาะตัว ซึ่งบ่งบอกไว้ในเอกสาร WSDL อยู่แล้ว

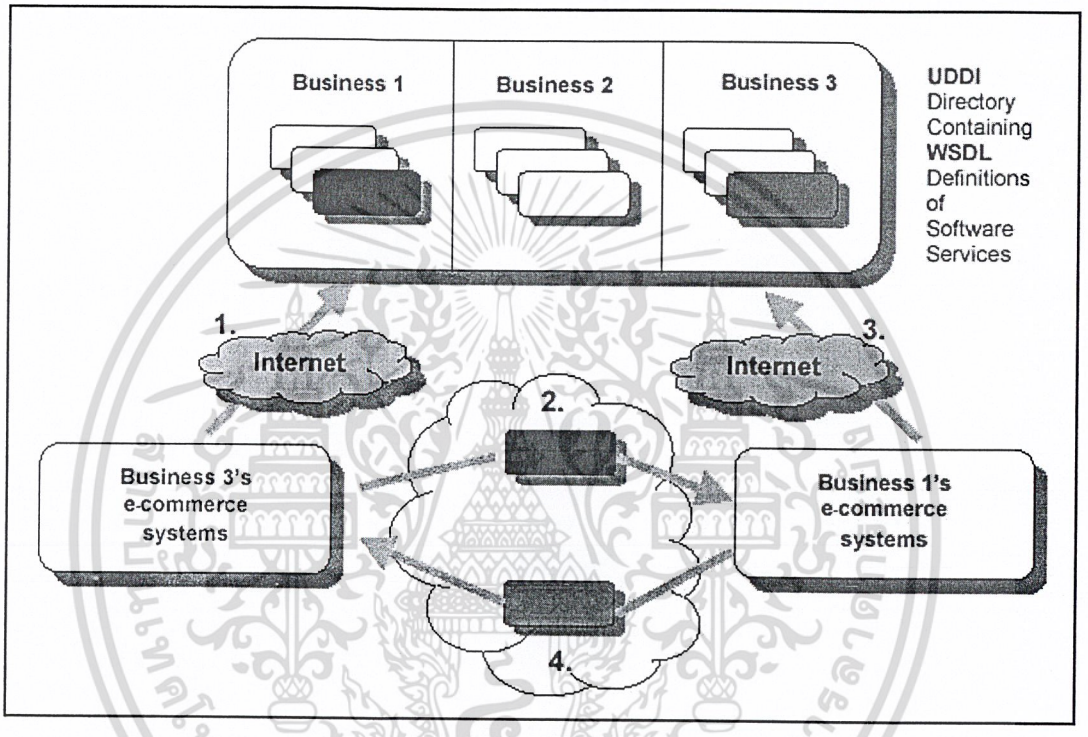
2.2.5.4 UDDI (Universal Description, Discovery and Integration protocol)

UDDI เป็นวิธีการมาตรฐานที่ให้ชุดพื้นฐาน APIs (Application Programming Interface) ของ SOAP ที่สามารถนำมาใช้ในการพัฒนาเป็นตัวแทนของผู้ให้บริการ (Service broker) UDDI ใช้สำหรับจัดเก็บและรวบรวมบริการต่างๆที่ให้บริการในรูปของ Directory Service และค้นหา Service ที่ต้องการและเมื่อได้มาแล้ว UDDI ยังจัดหาข้อตกลงในวิธีการที่จะใช้งานเปรียบได้กับสมุดหน้าเหลือง เป็นมาตรฐานที่จัดตั้งขึ้นโดยบริษัทไอบีเอ็ม บริษัทไมโครซอฟต์ และบริษัทอาริบา (Ariba) ปัจจุบันมีบริษัทที่ร่วมกันกำหนดมาตรฐานของ UDDI มากกว่า 70 บริษัท ซึ่งมาตรฐานของ UDDI ถูกกำหนดให้เป็นมาตรฐานสำหรับ B2B (Business to Business)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<u>การจำลองภาพการทำงานของเว็บเซอร์วิส</u>

เว็บเซอร์วิส คือ ซอฟต์แวร์คอมโพเนนท์ (Software Component) ที่สามารถนำมาสร้างเป็นแอพพลิเคชันสำหรับให้บริการการทำงาน ๆ หนึ่งให้แก่ผู้ร้องขอบนอินเทอร์เน็ต หรือสามารถที่จะนำเว็บเซอร์วิสแต่ละตัวมาประกอบกันตามกระบวนการทางธุรกิจ เพื่อร่วมกันทำงานในลักษณะที่เป็นการทำงานร่วมกันข้ามระบบที่แตกต่างกันได้ (Interoperability) รวมกันเป็น "เว็บเซอร์วิส" ซึ่งสามารถจำลองภาพการทำงานของเว็บเซอร์วิสที่เกิดจากการทำงานร่วมกันของ SOAP, UDDL และ WSDL ในรูปที่ 2-11



รูปที่ 2-11 เว็บเซอร์วิสกับ SOAP, UDDI และ WSDL

จากรูปที่ 2-11 เป็นตัวอย่างการใช้งานเว็บเซอร์วิสของ Business 1 จากพนักงานไอทีของ Business 3 ซึ่งมีขั้นตอนดังนี้

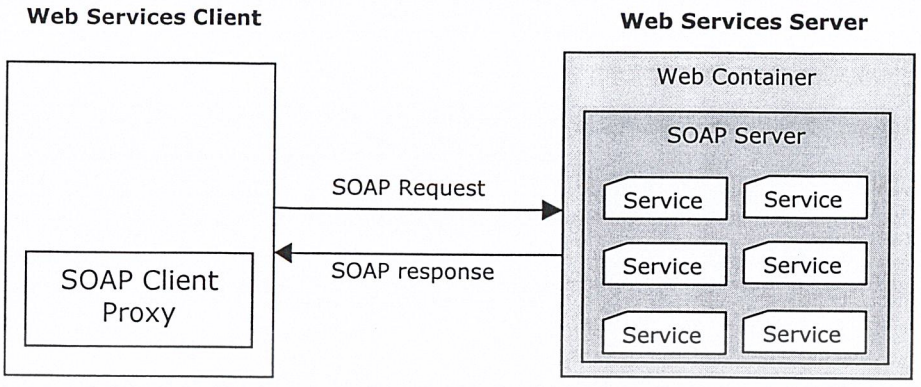
1. ค้นหาข้อกำหนดการให้บริการใน WSDL ผ่านอินเทอร์เฟสของ Business 1
2. ติดต่อผ่านอินเทอร์เฟสของแอพพลิเคชันที่ให้บริการด้วย SOAP ผ่านทางอินเทอร์เน็ต

ในขั้นตอนที่ 3 และ 4 Business 1 จะรวมหรือบูรณาการกับ Business 3 ด้วยวิธีเดียวกันกับกรณีที่ Business 3 ติดต่อกับ Business 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

2.2.6 โครงสร้างและการทำงานของ เว็บเซอร์วิส



รูปที่ 2-12 โครงสร้างการทำงานของเว็บเซอร์วิส

จากรูปที่ 2-12 แสดงโครงสร้างการทำงานของระบบเว็บเซอร์วิสจะแบ่งเป็น 2 ส่วนหลักๆ ดังนี้

1. เว็บเซอร์วิสเซิร์ฟเวอร์ เป็นส่วนที่เปิดให้บริการต่างๆ โดยเซิร์ฟเวอร์จะทำงานบนสภาพแวดล้อมซึ่งประกอบด้วย เว็บคอนเทนเนอร์ และ SOAP เซิร์ฟเวอร์
2. เว็บเซอร์วิสไคลเอนท์ เป็นส่วนที่เรียกใช้บริการจากเว็บเซอร์วิสที่เปิดให้บริการ ซึ่งไคลเอนท์นี้จะต้องมีส่วน SOAP Client Proxy เพื่อทำหน้าที่ในการสร้างและรับ SOAP Message

2.3 ความรู้เบื้องต้นของภาษา XML

2.3.1 XML (Extensibles Markup Language)

XML เป็นฟอร์แมตที่อธิบายถึงรายละเอียดของโครงสร้างและแบบของข้อมูลเป็นภาษาหรือชุดคำสั่งเกี่ยวกับข้อมูลบนเว็บ ที่ให้การพัฒนาและมีศักยภาพในส่วนของโครงสร้างข้อมูลจากหลากหลายแอพพลิเคชันมานำเสนอบนเครื่องเดสก์ทอป โดยมีโครงสร้างและรูปแบบที่เปิดให้แอพพลิเคชันต่างๆ สามารถเรียกไปใช้งานได้โดยที่ XML จะทำให้การจัดการข้อมูลหรือเรียกใช้ข้อมูลจากแอพพลิเคชันต่างๆ จะเข้าสู่มาตรฐานเดียวกันด้วย

XML เป็น Meta-language นั่นคือ XML เป็นภาษาที่ใช้อธิบายภาษาอื่น การกำหนดอิลิเมนต์สามารถกำหนดได้ตามความต้องการของผู้พัฒนา เพื่อให้สอดคล้องกับโปรแกรมการทำงานของผู้พัฒนาเอกสาร

2.3.2 เอกสาร XML (XML Document)

Well Formed XML Documents

เอกสาร well-formed คือ เอกสาร XML ที่มีความถูกต้องตามกับกฎเกณฑ์พื้นฐานของภาษา XML และมีความสอดคล้องกับข้อบังคับของการเป็นเอกสาร well-formed ที่เรียกว่า well-formedness constraint

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
62059

Valid XML Documents

เอกสาร valid นั้นนอกจากจะต้องเป็นเอกสาร well-formed แล้วยังต้องมีความถูกต้องและสอดคล้องของโครงสร้างเอกสาร XML กับการประกาศโครงสร้างทางตรรกะของเอกสาร XML และ มีความสอดคล้องกับข้อบังคับของการเป็นเอกสาร valid ที่เรียกว่า validity constraint

2.3.3 DTD

เป็นภาษาที่ทำหน้าที่กำหนดกฎ กติกา ข้อตกลงเพื่อใช้ในการตรวจสอบความถูกต้องและเป็นข้อบังคับให้กับอีลิเมนต์ แอตทริบิวต์ และเอนทิตี้ที่อยู่ในแหล่งข้อมูล XML

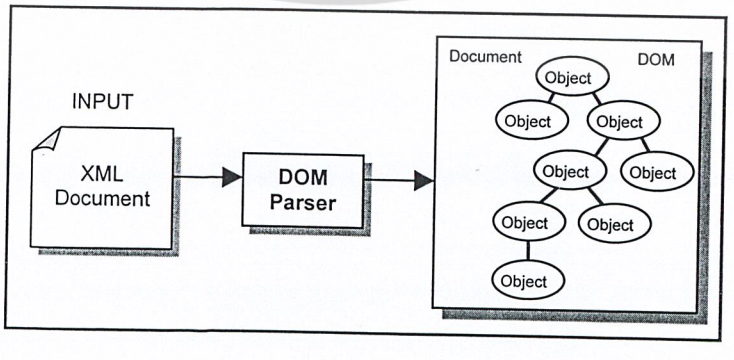
2.3.4 วิธีการเข้าถึงข้อมูลในภาษา XML

เพื่อให้สามารถเข้าถึงข้อมูลและดึงข้อมูลนำมาใช้จากเอกสาร XML ได้วิธีการเข้าถึงข้อมูลในเอกสารXML โดยใช้ตัววิเคราะห์โครงสร้างของเอกสาร (Parser) ซึ่งปัจจุบันมีวิธีการที่นิยมใช้อยู่ 2 วิธี คือ

- DOM (Document Object Model)
- SAX (Simple API for XML)

2.3.4.1 DOM Parsing

DOM (Document Object Model) เป็นการทำงานแบบ Tree-Based parser และใช้การเข้าถึงข้อมูลด้วยวิธีการ Random-Access คือจะประมวลโครงสร้างของเอกสาร XML ให้เป็นโครงสร้างต้นไม้ (tree structure) แบบ hierarchical object model ซึ่งในโครงสร้างต้นไม้จะประกอบไปด้วยอิลิเมนต์โหนด (element node) โดยภายในแต่ละอิลิเมนต์โหนด ประกอบไปด้วยข้อมูลที่เกี่ยวข้องกับอิลิเมนต์นั้นตามโครงสร้างและข้อมูลในเอกสาร XML ซึ่งจะประกอบไปด้วยข้อมูลทั้งหมดของเอกสาร XML เพื่อให้แอพพลิเคชั่นสามารถเข้าหาจุดต่าง ๆ ของ tree structure ได้ โดยที่ DOM จะโหลด XML ทั้งชุด (ข้อมูลใน Root Node ทั้งหมด) เข้ามาเป็น Tree ในหน่วยความจำหลักของคอมพิวเตอร์ (memory) ก่อนจึงทำงานได้ดังแสดงในรูปที่ 2-13

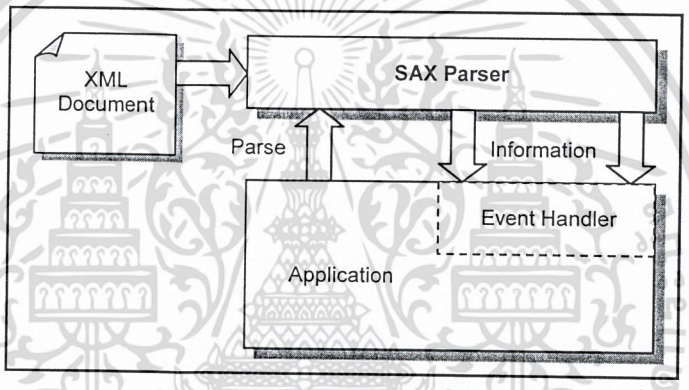


รูปที่ 2-13 แสดงวิธีการเข้าถึงข้อมูลโดยวิธี DOM Parsing

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.4.2 SAX Parsing

SAX (Simple API for XML) เป็นการทำงานแบบ Event-Based API และใช้การเข้าถึงข้อมูลด้วยวิธีการ Sequential-Access คือ จะเข้าถึงข้อมูลโดยใช้เหตุการณ์เป็นตัวกำหนดด้วยการอ่านเอกสาร XML และตอบสนองต่อ Markup ที่อ่านพบโดยถือว่าเป็นเสมือนเหตุการณ์ต่างๆ ซึ่งจะรายงานข้อมูลตั้งแต่จุดเริ่มต้นจนถึงจุดสิ้นสุดของอิลิเมนต์ต่าง ๆ ไปให้แอพพลิเคชั่นโดยไม่ต้องมีการสร้างโครงสร้างต้นไม้ขึ้นมา ส่วนการเข้าถึงข้อมูลจะอาศัยกระบวนการ Scan Node ซึ่งจะทำการ scan ไปที่ละ line ทีละ tag เริ่มต้นจาก Root Node มายังส่วนที่เราต้องการติดต่อจากนั้นก็ใช้งานฟังก์ชั่นของ SAX ต่อไป ส่วนใหญ่วิธีนี้มักใช้ในงานที่เกี่ยวข้องกับการรับ-ส่งข้อมูลเอกสาร XML ผ่านระบบครือข่ายเนื่องจากวิธีนี้ให้ความรวดเร็วและสิ้นเปลืองหน่วยความจำน้อย แต่ในทางโปรแกรมมิ่งมักไม่ใช้เนื่องจากวิธีนี้เป็นวิธีที่ยากต่อการใช้งานเพราะการเข้าถึงข้อมูลเป็นไปตามลำดับเหตุการณ์ซึ่งไม่สอดคล้องกับวิธีการทางโปรแกรมในการเข้าถึงข้อมูลดังแสดงในรูปที่ 2-14



รูปที่ 2-14 แสดงวิธีการเข้าถึงข้อมูลโดยวิธี SAX Parsing

2.4 ความรู้เบื้องต้นเกี่ยวกับ SVG

2.4.1 SVG (Scalable Vector Graphics)

SVG เป็นภาษาที่ใช้อธิบาย GML ในรูปภาพ 2 มิติได้ เนื่องจากเราสามารถอ่านและจัดการข้อมูลในเอกสาร XML ได้ง่าย จึงมีความพยายามที่จะสร้างมาตรฐานในการบอกลักษณะการแสดงผลรูปภาพด้วย XML ทำให้เกิดมาตรฐาน SVG ซึ่งเป็นภาษาที่ได้รับการรับรองโดยองค์กร World Wide Web Consortium (W3C)

SVG จะกำหนดรูปแบบของวัตถุสามอย่างคือ vector graphic shape (เช่น ส่วนของเส้นตรง และเส้นโค้ง) รูป และ ตัวอักษร โดยสามารถที่จะกำหนด grouped, styled, transformed หรือ composite ให้แก่วัตถุใน SVG ได้ อีกทั้งยังเพิ่มคุณสมบัติ transformations, clipping paths, alpha masks, filter effects , template object และอีกมากมาย

เนื่องจาก SVG เป็นเอกสาร XML ดังนั้นจึงมีการสร้าง DOM สำหรับ SVG ขึ้นมาโดยอ้างอิงตามมาตรฐานของ W3C ซึ่งทำให้ SVG มีคุณสมบัติ dynamic และ interactive ตามไปด้วย แม้แต่ event handlers' อย่าง on mouse over หรือ on click ก็สามารถกำหนดลงในแต่ละวัตถุของ SVG ได้ด้วย อีกทั้งการอ้างอิงตามมาตรฐาน ทำให้ SVG สามารถติดต่อ XML อื่นๆ ในเวบเพจเดียวกันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2 รูปแบบโครงสร้างของเอกสาร SVG

ตัวอย่างของเอกสาร SVG

```
<?xml version="1.0"  encoding="iso-8859-1"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
"http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg width="300" height="300" x="0" y="0">
        ...
        …
</svg>
```

จากตัวอย่างเอกสาร SVG ที่กล่าวมาข้างต้น โดยบรรทัดแรกเป็นการประกาศว่าเอกสารนี้มีรูปแบบเอกสารเป็น XML โดยอ้างอิงมาตรฐาน XML เวอร์ชัน 1.0 เนื่องจาก SVG ก็ถือได้ว่าเป็นเอกสาร XML ประเภทหนึ่งเหมือนกัน และเอกสารนี้ใช้การเข้ารหัสข้อมูลแบบ iso-8859-1

บรรทัดที่สองเป็นการบอกเกี่ยวกับ DTD ที่ใช้ในเอกสารนี้ซึ่ง DTD นั้นเป็นเอกสารที่ใช้บอกชนิดของข้อมูลที่ใช้ในเอกสาร XML และบอกว่าเอกสาร XML นั้นมีลักษณะโครงสร้างเป็นอย่างไร

ในส่วนของ <svg> อิลิเมนต์เป็นส่วนที่สามารถกำหนดรูปแบบของวัตถุได้ดังนี้

- **Vector Graphic Shapes**

<rect> : ใช้สำหรับสร้างและปรับแต่งรูปสี่เหลี่ยม

ตัวอย่าง

```
< rect x="20" y="20" rx="20" ry="20" width="250"
        height="100" style="fill:red;stroke:black;stroke-
        width:5;opacity:0.5"/>
```

คำอธิบาย แอตทริบิวต์

x, y ใช้กำหนดตำแหน่งของรูปสี่เหลี่ยมที่สร้าง

rx, ry ใช้กำหนดส่วนโค้งบริเวณมุมสี่เหลี่ยม

width, height ใช้กำหนดขนาดความกว้างและความสูงตามลำดับ

style ใช้ในการกำหนดลักษณะของวัตถุ

fill, stroke ใช้กำหนดสีของวัตถุและสีของเส้นขอบวัตถุตามลำดับ

stroke-width ใช้กำหนดความหนาของเส้น

opacity ใช้กำหนดความโปร่งใสของวัตถุมีค่าระหว่าง 0-1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<circle> : ใช้สำหรับสร้างและปรับแต่งรูปวงกลม

ตัวอย่าง

```
<circle cx="100" cy="50" r="40" stroke="black"
        stroke-width="2" fill="red"/>
```

คำอธิบายแอตทริบิวต์

cx, cy ใช้กำหนดจุดศูนย์กลางของวงกลม หากไม่กำหนดจะมีค่าเป็น 0, 0

r ใช้กำหนดรัศมีของวงกลม

<ellipse> : ใช้สำหรับสร้างและปรับแต่งรูปวงรี

ตัวอย่าง

```
< ellipse cx="300" cy="150" rx="200" ry="80"
        style="fill:rgb(200,100,50);stroke:rgb(0,0,100);
        stroke-width:2"/>
```

คำอธิบาย แอตทริบิวต์

cx, cy ใช้กำหนดจุดศูนย์กลางของวงกลม หากไม่กำหนดจะมีค่าเป็น 0, 0

rx, ry ใช้กำหนดรัศมีในแนวนอนและแนวตั้งตามลำดับ

<line> : ใช้สำหรับสร้างและปรับแต่งเส้นตรง

ตัวอย่าง

```
<line x1="0" y1="0" x2="300" y2="300"
        style="stroke:rgb(99,99,99);stroke-width:2"/>
```

คำอธิบาย แอตทริบิวต์

x1, y1 ใช้กำหนดตำแหน่งจุดเริ่มต้นของเส้นตรง

x2, y2 ใช้กำหนดตำแหน่งจุดสิ้นสุดของเส้นตรง

<polygon> : ใช้สำหรับสร้างและปรับแต่งรูปหลายเหลี่ยม

ตัวอย่าง

```
<polygon points="220,100 300,210 170,250"
        style="fill:#cccccc;stroke:#000000;stroke-width:1"/>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำอธิบาย แอตทริบิวต์

point ใช้กำหนดตำแหน่งของมุมแต่ละมุม (x,y)

<polyline> : ใช้สำหรับสร้างและปรับแต่งรูปเส้นตรงหลายเส้นเชื่อมต่อกัน

ตัวอย่าง

```
<polyline points="0,0 0,20 20,20 20,40 40,40 40,60"
        style="fill:#cccccc;stroke:#000000;stroke-width:1"/>
```

คำอธิบาย แอตทริบิวต์

point ใช้กำหนดตำแหน่งจุดเชื่อมต่อของเส้น (x,y)

<path> : ใช้สำหรับสร้างและปรับแต่งรูปร่างต่างๆโดยจะมีคำสั่งที่ใช้ในการสร้างรูปร่างดังนี้

- M = moveto
- Q = quadratic Belzier curve
- A = elliptical Arc
- H = horizontal lineto
- V = vertical lineto
- S = smooth curveto
- C = curveto
- L = lineto
- Z = closepath
- T = smooth quadratic Belzier curveto

ตัวอย่าง

```
<path d="M250 150 L150 350 L350 350 Z" />
```

คำอธิบาย

จากตัวอย่างเป็นการสร้าง path โดยเริ่มต้นที่จุด 250,150 จากนั้นลากเส้นตรงไปยังจุด 150,350 จากนั้นลากเส้นตรงไปยังจุด 350,350 จากนั้นทำการปิด path โดยกลับไปที่จุด 250,150

- **การเขียนตัวอักษรใน SVG**

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Element อื่นๆของ SVG ที่มีการใช้งานบ่อย**

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

- **แอตทริบิวต์ อื่นๆของ SVG ที่มีการใช้งานบ่อย**

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

2.4.3 ประโยชน์ในการใช้ SVG

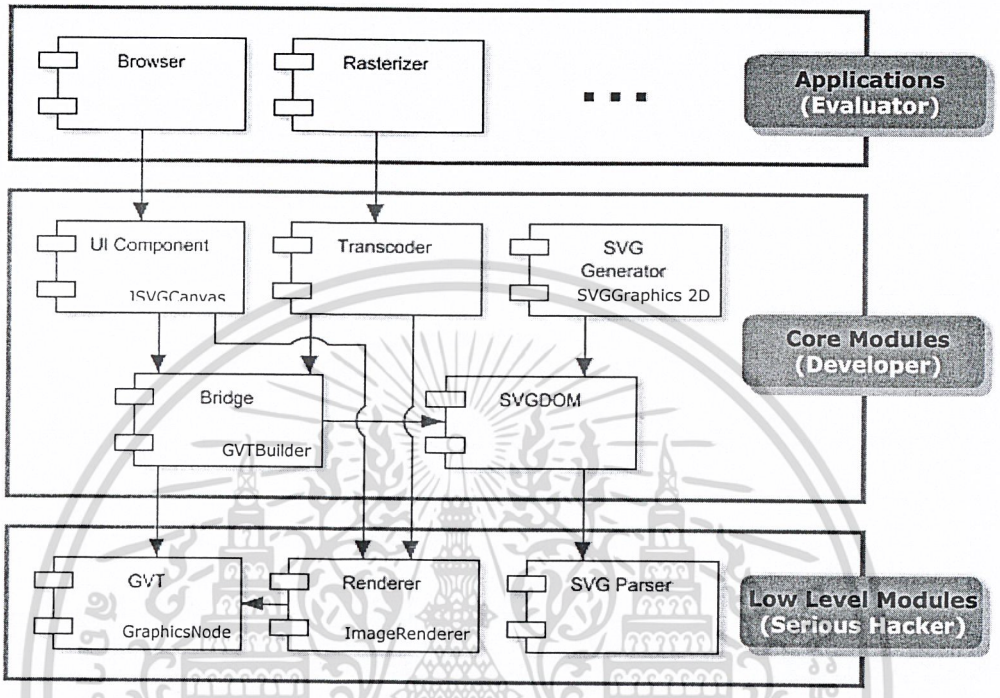
- SVG สามารถอ่านและแก้ไขได้โดยโปรแกรมหลายโปรแกรม เช่น notepad
- สามารถเปลี่ยนขนาดของภาพได้โดยคุณภาพของภาพไม่ลดลง
- SVG สามารถทำงานร่วมกับภาษาจาวาได้
- SVG เป็นมาตรฐานที่เปิดเผยให้คนทั่วไปทราบ
- ข้อความใน SVG มีความสามารถในการเลือกและค้นหาจึงเหมาะแก่การนำมาทำเป็นแผนที่

2.4.4 Batik SVG Toolkit

Batik คือ ชุดเครื่องมือภาษาจาวาที่ใช้สำหรับ แอปพลิเคชั่นหรือ applets ที่ต้องการใช้รูปภาพในรูปแบบ SVG เพื่อใช้ในจุดประสงค์ต่างๆ เช่น แสดงภาพ สร้างภาพในรูปแบบ SVG และจัดการกับรูปภาพ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วัตถุประสงค์ของโครงการนี้เพื่อสร้างโมดูลที่สามารถทำงานได้ด้วยตัวเองหรือทำงานรวมกับส่วนอื่นๆเพื่อช่วยเหลือในการพัฒนาโปรแกรมที่เกี่ยวกับ SVG ตัวอย่างเช่น โมดูลในการสร้าง SVG, โมดูล SVG Parser, โมดูล SVG DOM และสิ่งที่สำคัญที่สุดคือ สร้างความยืดหยุ่นในการใช้งาน เช่น อนุญาตให้นักพัฒนาสามารถจัดการกับแท็ก SVG ที่กำหนดเพิ่มเติมได้เอง



รูปที่ 2-15 โครงสร้างสถาปัตยกรรมของ Batik

โครงสร้างสถาปัตยกรรมของ **Batik** ประกอบด้วย 3 ส่วนดังนี้

- **Applications Modules** คือ ส่วนที่แสดงตัวอย่างวิธีการใช้ Core Modules เพื่อให้ผู้ใช้ทราบถึงความสามารถของโปรแกรมที่พัฒนาด้วย Batik ตัวอย่างเช่น SVG browser, SVG Pretty Printer, SVGFont Converter
- **Core Modules** คือ ส่วนหลักของ Batik เป็นโมดูลที่นักพัฒนาสามารถนำไปใช้ในการสร้าง จัดการดัดแปลง และแสดงผลเอกสาร SVG ตัวอย่างเช่น
  - SVG Generator ใช้สำหรับแปลงรูปภาพรูปแบบให้อยู่ในรูปแบบ SVG
  - SVG DOM ใช้สำหรับให้จาวาจัดการกับเอกสาร SVG
  - JSVGCanvas เป็นส่วนที่ใช้ในการแสดง SVG โดยอนุญาตให้ผู้ใช้สามารถ มีปฏิสัมพันธ์กับรูปภาพได้ เช่น ย่อ/ขยาย, เลื่อนส่วนแสดงรูปภาพ, หมุนรูป, เลือกตัวอักษร
- **Low Level Modules** คือ ส่วนที่ถูกเรียกใช้โดย Core Modules โดยปกตินักพัฒนาจะไม่ได้เรียกใช้งานในส่วนนี้โดยตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 การกำหนดตำแหน่งด้วยระบบจีพีเอส

2.5.1 GPS (Global Positioning System)

GPS (Global Positioning System) คือ ระบบที่ระบุตำแหน่งทุกแห่งบนโลก ซึ่งถ้าเรามีอุปกรณ์รับข้อมูลติดตั้งอยู่ จะทำให้สามารถแสดงตำแหน่งนั้นอย่างแม่นยำโดยใช้เครื่องรับจีพีเอสอ่านตำแหน่งละติจูดลองติจูดของทุกจุดบนพื้นโลก โดยรับสัญญาณจากดาวเทียมบอกตำแหน่งซึ่งทำการโคจรอยู่รอบโลกอย่างน้อย 4 ดวงจากทั้งหมด 24 ดวง ระบบจีพีเอสใช้ดาวเทียมนาฟสตาร์ (NAVSTAR) โดยการส่งคลื่นวิทยุจากดาวเทียมในอวกาศมายังภาคพื้นดิน และใช้ความต่างของเวลาในการรับส่งสัญญาณระหว่างดาวเทียมกับตัวรับสัญญาณ โดยการคำนวณหาตำแหน่ง ความเร็วและเวลาให้กับผู้ใช้ ตามปกติระบบจีพีเอสจะมีการใส่รหัสเพื่อให้เกิดความผิดพลาดเล็กน้อยทั้งนี้เนื่องจากระบบนี้ถูกพัฒนาขึ้นเพื่อให้กระทรวงกลาโหมของสหรัฐอเมริกานำระบบนี้ไปใช้ในทางการทหาร แต่สัญญาณดาวเทียมที่ถูกส่งออกมานั้นสามารถถูกรับได้โดยผู้ใช้ทั่วไป ดังนั้นจึงมีการใส่รหัสเพื่อให้ผู้ใช้ที่ได้รับอนุญาตเท่านั้นใช้งานได้ ในการใช้งานระบบจีพีเอสในโหมดมาตรฐานจะมีความเที่ยงตรงโดยเฉลี่ย 100 เมตร ในแนวนอนและ 156 เมตรในแนวตั้งในงานที่ต้องการความเที่ยงตรงมากๆ จึงต้องใช้ DGPS (Difference Global Positioning System)

โดยรูปแบบของ DGPS จะประกอบด้วยเครื่องรับที่เรียกว่า “ เครื่องรับอ้างอิง ” ซึ่งทราบตำแหน่งที่ถูกต้องอย่างแท้จริงอยู่แล้ว ส่วนเครื่องรับจีพีเอสอื่นๆ จะเป็นของผู้ใช้งาน โดยเครื่องรับอ้างอิงหรือสถานีอ้างอิงนี้จะนำสัญญาณดาวเทียมจีพีเอสมาคำนวณพิสัยของดาวเทียมแต่ละดวง

2.5.2 มาตรฐานในการสื่อสารข้อมูล NMEA-0183

หน่วยงานที่ทำหน้าที่ในการกำหนดมาตรฐานในการสื่อสารข้อมูลจีพีเอส คือ หน่วยงาน NMEA (National Marine Electromisc Association) ได้กำหนดโปรโตคอล NMEA – 183 เป็นโปรโตคอลมาตรฐานในการสื่อสารข้อมูลจากเครื่องรับจีพีเอสไปสู่อุปกรณ์เชื่อมต่อภายนอก ซึ่งรูปแบบเหล่านี้จะอยู่ในรูปของรหัสแอสกี้ (ASCII Codes) และสามารถเชื่อมต่อกับเครื่องคอมพิวเตอร์ได้โดยผ่านพอร์ทอนุกรมของเครื่องคอมพิวเตอร์ (RS – 232) โดยกำหนดอัตราเร็วในการส่งข้อมูลอยู่ที่ 4800 บิตต่อวินาที โดยใช้การส่งข้อมูล 8 บิต แบบไม่มีพาริตี้ แต่มีบิตเริ่มต้น (Start bit) และบิตสิ้นสุด (Stop bit)

ตามโปรโตคอล NMEA – 183 สัญลักษณ์ข้อมูลที่ใช้ในการสื่อสารจะอยู่ในรูปแบบประโยคโดยที่อักขระพิเศษหมายถึงขึ้นบรรทัดใหม่ (<CR> <LF>) แต่ละส่วนในประโยคมีการคั่นด้วยเครื่องหมายจุลภาคและมีการแบ่งประโยคเป็นกลุ่มรูปแบบที่แตกต่างกัน โดยมีตัวอย่างกลุ่มรูปแบบประโยคได้แก่ GLL,GGA,STN,TRF และ VRW เป็นต้น ซึ่งมีตัวอย่างของรูปแบบประโยคดังต่อไปนี้

กรณีที่ยังไม่ได้รับสัญญาณ :

$GPGGA,235947.000,0000.0000,N,00000.0000,E,0,00,0.0,0.0,M,,,,0000*00

กรณีที่ได้รับสัญญาณแล้ว :

$GPGGA,092204.999,4250.5589,S,14718.5084,E,1,04,24.4,19.7,M,,,,0000*1F

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

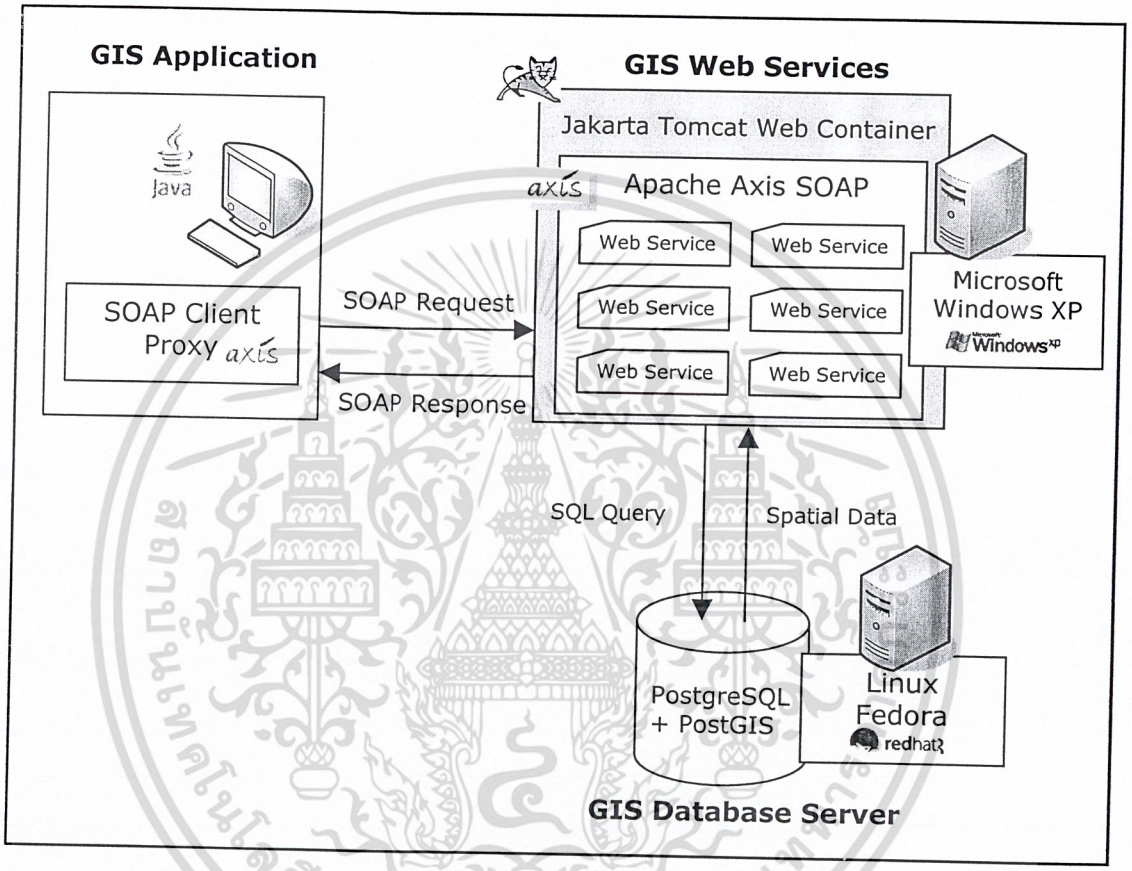
Field	Example	Comments
Sentence ID	$GPGGA	
UTC Time	092204.999	hhmmss.sss
Latitude	4250.5589	ddmm.mmmm
N/S Indicator	S	N = North, S = South
Longitude	14718.5084	dddmm.mmmm
E/W Indicator	E	E = East, W = West
Position Fix	1	0 = Invalid, 1 = Valid SPS, 2 = Valid DGPS, 3 = Valid PPS
Satellites Used	04	Satellites being used (0-12)
HDOP	24.4	Horizontal dilution of precision
Altitude	19.7	Altitude in meters according to WGS-84 ellipsoid
Altitude Units	M	M = Meters
Geoid Seperation		Geoid seperation in meters according to WGS-84 ellipsoid
Seperation Units		M = Meters
DGPS Age		Age of DGPS data in seconds
DGPS Station ID	0000	
Checksum	*1F	
Terminator	CR/LF	

ตารางที่ 2-2 แสดงความหมายในแต่ละฟิลด์ของโปรโตคอล NMEA-0183

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบโครงงาน

3.1 โครงสร้างของโครงงาน



รูปที่ 3-1 โครงสร้างของโครงงาน

โครงสร้างของโครงงานแบ่งออกเป็น 3 ส่วน คือ

1. ระบบฐานข้อมูล GIS

ผู้จัดทำเลือกใช้ PostgreSQL เป็น DBMS ในการเก็บข้อมูล 2 ประเภท คือ ข้อมูลเชิงพื้นที่ (Spatial data) และข้อมูลเชิงอธิบาย (Non-Spatial data) และได้มีการติดตั้ง PostGIS ซึ่งเป็นส่วนเพิ่มเติมของ PostgreSQL เพื่อเป็นการเพิ่มประสิทธิภาพในการจัดการข้อมูลเชิงพื้นที่โดยระบบฐานข้อมูลนี้จะติดตั้งอยู่บนระบบปฏิบัติการลีนุกซ์ ซึ่งผู้จัดทำเลือกใช้ลีนุกซ์ดิสทริบิวชั่นของ Fedora Core 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. GIS เว็บเซอร์วิส

เป็นเว็บเซอร์วิสสำหรับให้บริการเชิงพื้นที่ (Location-Based Services) สำหรับระบบสารสนเทศเชิงภูมิศาสตร์ (GIS) โดยจะให้บริการในส่วนของข้อมูลรายละเอียดของสถานที่ต่างๆ และแผนที่ในรูปแบบภาษา SVG ซึ่งสภาพแวดล้อมบนเซิร์ฟเวอร์ที่เลือกใช้ในการพัฒนาระบบนี้ผู้จัดทำได้เลือกใช้ Jakarta Tomcat 5.0.27 เป็นเว็บคอนเทนเนอร์ และติดตั้ง Apache Axis 1.1 เป็น SOAP เซิร์ฟเวอร์ ซึ่งมีหน้าที่ในการจัดการกับ SOAP แมสเสจโดยจะทำการแปลงจากจาวาออปเจกต์เป็น SOAP แมสเสจเพื่อใช้ในการส่งและแปลงจาก SOAP แมสเสจเป็นจาวา ออปเจกต์ในกรณีของการรับ SOAP แมสเสจ โดยเว็บคอนเทนเนอร์นี้จะติดตั้งอยู่บนระบบปฏิบัติการ Microsoft Windows XP

3. GIS แอพพลิเคชั่น

เป็นส่วนที่เรียกใช้บริการต่างๆ จากเว็บเซิร์ฟเวอร์ และนำข้อมูลจากการเรียกใช้บริการทั้งข้อมูลรายละเอียด และแผนที่นำมาแสดงผลในแอพพลิเคชั่น การแสดงผลแผนที่นั้นจะใช้ Batik SVG ToolKit1.5.1 ของค่าย Apache นำมาทำเป็นส่วนแสดงผลแผนที่ที่อยู่ในรูปแบบภาษา SVG อีกทั้งยังมีความสามารถพื้นฐานในการแสดงผลแผนที่ คือ สามารถย่อ/ขยาย เลื่อนแผนที่ และเลือกแสดงผลในแต่ละเลเยอร์ได้ โดยใช้ API ของ Apache Axis ในการจัดการกับการรับ และส่ง SOAP แมสเสจ

3.2 การออกแบบส่วนของฐานข้อมูล GIS

ข้อมูลที่เก็บอยู่ในฐานข้อมูลนี้เป็นข้อมูลที่แสดงตำแหน่ง และรายละเอียดของสถานที่ที่อยู่ในบริเวณพื้นที่ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง โดยเป็นข้อมูลของสถานที่ที่มีลักษณะเป็นจุด (point) และแบ่งการจัดเก็บข้อมูลเป็นเลเยอร์ต่างๆ ทั้งหมด 8 เลเยอร์เพื่อใช้เป็นข้อมูลในการให้บริการดังนี้

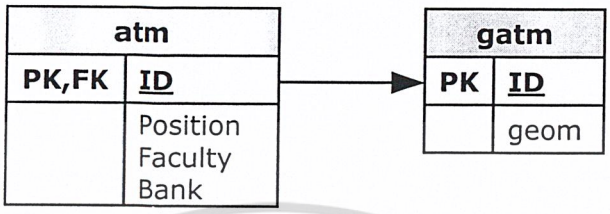
- เครื่องถอนเงินอัตโนมัติ
- ธนาคาร
- โรงอาหาร
- ที่จอดรถ
- ร้านค้าประเภทอาหารและเครื่องดื่ม
- ตู้โทรศัพท์สาธารณะ
- ห้องสมุด
- ตู้ไปรษณีย์

ซึ่งในแต่ละเลเยอร์มีการเก็บข้อมูลโดยแบ่งออกเป็น 2 ตาราง คือ ตารางข้อมูลเชิงพื้นที่ (Spatial data) และตารางข้อมูลเชิงอธิบาย (Non-Spatial data) ซึ่งทั้งสองตารางของแต่ละเลเยอร์นั้นจะใช้คอลัมน์ ID เป็นส่วนที่เชื่อมความสัมพันธ์ระหว่างตาราง 2 ตารางดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ตารางเก็บข้อมูลเครื่องถอนเงินอัตโนมัติ

เป็นตารางที่ใช้สำหรับเก็บข้อมูลรายละเอียดและตำแหน่งที่ตั้งของเครื่องถอนเงินอัตโนมัติ (ATM) ซึ่งแบ่งออกเป็น 2 ตาราง ได้แก่ ตาราง atm สำหรับเก็บข้อมูลเชิงอธิบาย และตาราง gatm สำหรับเก็บข้อมูลเชิงพื้นที่



รูปที่ 3-2 ความสัมพันธ์ระหว่างตาราง atm และ gatm

รายละเอียดของตาราง atm

ชื่อคอลัมน์	ความหมาย	key
ID	หมายเลขอ้างอิงตู้ ATM	PK
Position	คำอธิบายรายละเอียดตำแหน่งที่ตั้ง	
Faculty	คณะที่ตู้ ATM ตั้งอยู่	
Bank	ชื่อธนาคาร	

ตารางที่ 3-1 แสดงรายละเอียดของตาราง atm

รายละเอียดของตาราง gatm

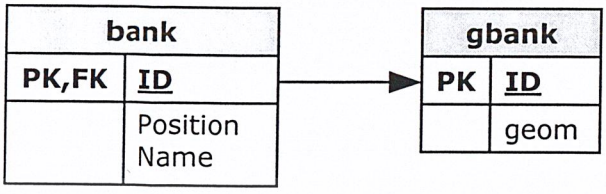
ชื่อคอลัมน์	ความหมาย	key
ID	หมายเลขอ้างอิงตู้ ATM	PK
geom	ข้อมูลเชิงภูมิศาสตร์ของ ATM (point)	

ตารางที่ 3-2 แสดงรายละเอียดของตาราง gatm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ตารางเก็บข้อมูลธนาคาร

เป็นตารางที่ใช้สำหรับเก็บข้อมูลรายละเอียดและตำแหน่งที่ตั้งของธนาคารต่างๆ ซึ่งแบ่งออกเป็น 2 ตาราง ได้แก่ ตาราง bank สำหรับเก็บข้อมูลเชิงอธิบาย และตาราง gbank สำหรับเก็บข้อมูลเชิงพื้นที่



รูปที่ 3-3 ความสัมพันธ์ระหว่างตาราง bank และ gbank

รายละเอียดของตาราง bank

ชื่อคอลัมน์	ความหมาย	key
ID	หมายเลขอ้างอิงธนาคาร	PK
Position	คำอธิบายรายละเอียดตำแหน่งที่ตั้ง	
Name	ชื่อธนาคาร	

ตารางที่ 3-3 แสดงรายละเอียดของตาราง bank

รายละเอียดของตาราง gbank

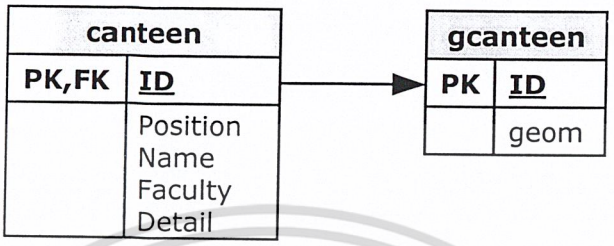
ชื่อคอลัมน์	ความหมาย	key
ID	หมายเลขอ้างอิงธนาคาร	PK
geom	ข้อมูลเชิงภูมิศาสตร์ของธนาคาร (point)	

ตารางที่ 3-4 แสดงรายละเอียดของตาราง gbank

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ตารางเก็บข้อมูลโรงอาหาร

เป็นตารางที่ใช้สำหรับเก็บข้อมูลรายละเอียดและตำแหน่งที่ตั้งของโรงอาหารต่างๆซึ่งแบ่งออกเป็น 2 ตาราง ได้แก่ ตาราง canteen สำหรับเก็บข้อมูลเชิงอธิบาย และตาราง gcanteen สำหรับเก็บข้อมูลเชิงพื้นที่



รูปที่ 3-4 ความสัมพันธ์ระหว่างตาราง canteen และ gcanteen

รายละเอียดของตาราง **canteen**

ชื่อคอลัมน์	ความหมาย	key
ID	หมายเลขอ้างอิงโรงอาหาร	PK
Position	คำอธิบายรายละเอียดตำแหน่งที่ตั้ง	
Name	ชื่อโรงอาหาร	
Faculty	คณะที่โรงอาหารตั้งอยู่	
DetailInformation	ข้อมูลรายละเอียดเกี่ยวกับโรงอาหาร	

ตารางที่ 3-5 แสดงรายละเอียดของตาราง canteen

รายละเอียดของตาราง **gcanteen**

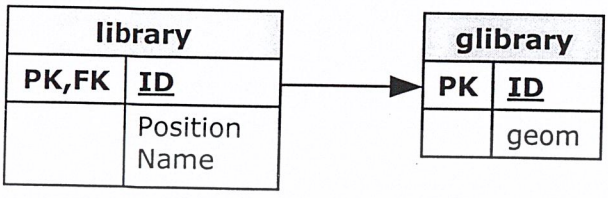
ชื่อคอลัมน์	ความหมาย	key
ID	หมายเลขอ้างอิงโรงอาหาร	PK
geom	ข้อมูลเชิงภูมิศาสตร์ของโรงอาหาร(point)	

ตารางที่ 3-6 แสดงรายละเอียดของตาราง gcanteen

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ตารางเก็บข้อมูลห้องสมุด

เป็นตารางที่ใช้สำหรับเก็บข้อมูลรายละเอียดและตำแหน่งที่ตั้งของห้องสมุดต่างๆ ซึ่งแบ่งออกเป็น 2 ตาราง ได้แก่ ตาราง library สำหรับเก็บข้อมูลเชิงอธิบาย และตาราง glibrary สำหรับเก็บข้อมูลเชิงพื้นที่



รูปที่ 3-5 ความสัมพันธ์ระหว่างตาราง library และ glibrary

รายละเอียดตาราง library

ชื่อคอลัมน์	ความหมาย	key
ID	หมายเลขอ้างอิงห้องสมุด	PK
Position	คำอธิบายรายละเอียดตำแหน่งที่ตั้ง	
Name	ชื่อห้องสมุด	

ตารางที่ 3-7 แสดงรายละเอียดของตาราง library

รายละเอียดของตาราง glibrary

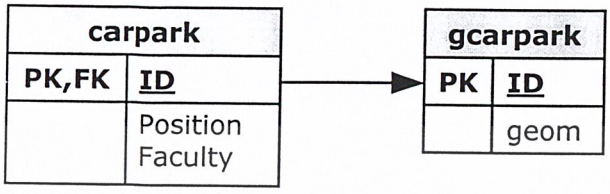
ชื่อคอลัมน์	ความหมาย	key
ID	หมายเลขอ้างอิงห้องสมุด	PK
geom.	ข้อมูลเชิงภูมิศาสตร์ของห้องสมุด(point)	

ตารางที่ 3-8 แสดงรายละเอียดของตาราง glibrary

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ตารางเก็บข้อมูลที่จอดรถ

เป็นตารางที่ใช้สำหรับเก็บข้อมูลรายละเอียดและตำแหน่งที่ตั้งของที่จอดรถต่างๆ ซึ่งแบ่งออกเป็น 2 ตาราง ได้แก่ ตาราง carpark สำหรับเก็บข้อมูลเชิงอธิบาย และตาราง gcarpark สำหรับเก็บข้อมูลเชิงพื้นที่



รูปที่ 3-6 ความสัมพันธ์ระหว่างตาราง carpark และ gcarpark

รายละเอียดของตาราง **carpark**

ชื่อคอลัมน์	ความหมาย	key
ID	หมายเลขอ้างอิงที่จอดรถ	PK
Position	คำอธิบายรายละเอียดตำแหน่งที่ตั้ง	
Faculty	ชื่อคณะ	

ตารางที่ 3-9 แสดงรายละเอียดของตาราง carpark

รายละเอียดของตาราง **gcarpark**

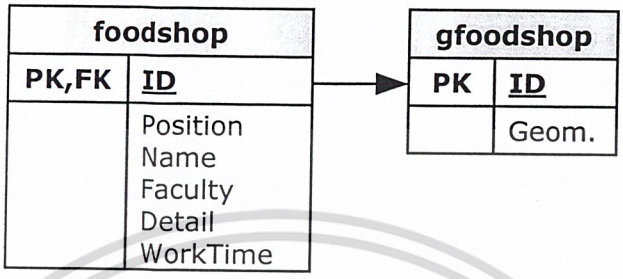
ชื่อคอลัมน์	ความหมาย	key
ID	หมายเลขอ้างอิงที่จอดรถ	PK
geom	ข้อมูลเชิงภูมิศาสตร์ของที่จอดรถ(point)	

ตารางที่ 3-10 แสดงรายละเอียดของตาราง gcarpark

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ตารางเก็บข้อมูลร้านขายอาหารและเครื่องดื่ม

เป็นตารางที่ใช้สำหรับเก็บข้อมูลรายละเอียดและตำแหน่งที่ตั้งของร้ายขายอาหารและเครื่องดื่มต่างๆ ซึ่งแบ่งออกเป็น 2 ตาราง ได้แก่ ตาราง foodshop สำหรับเก็บข้อมูลเชิงอธิบาย และตาราง gfoodshop สำหรับเก็บข้อมูลเชิงพื้นที่



รูปที่ 3-7 ความสัมพันธ์ระหว่างตาราง foodshop และ gfoodshop

รายละเอียดของตาราง foodshop

ชื่อคอลัมน์	ความหมาย	key
ID	หมายเลขอ้างอิงที่ร้านค้า	PK
Position	คำอธิบายรายละเอียดตำแหน่งที่ตั้ง	
Name	ชื่อร้าน	
Faculty	ชื่อคณะ	
Detail	ข้อมูลเกี่ยวกับร้าน	
WorkTime	เวลาเปิด-ปิด	

ตารางที่ 3-11 แสดงรายละเอียดของตาราง foodshop

รายละเอียดของตาราง gfoodshop

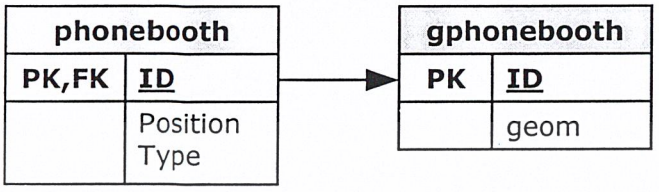
ชื่อคอลัมน์	ความหมาย	key
ID	หมายเลขอ้างอิงร้านค้า	PK
geom.	ข้อมูลเชิงภูมิศาสตร์ของร้าน(point)	

ตารางที่ 3-12 แสดงรายละเอียดของตาราง gfoodshop

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. ตารางเก็บข้อมูลตู้โทรศัพท์

เป็นตารางที่ใช้สำหรับเก็บข้อมูลรายละเอียดและตำแหน่งที่ตั้งของตู้โทรศัพท์ต่างๆ ซึ่งแบ่งออกเป็น 2 ตาราง ได้แก่ ตาราง phonebooth สำหรับเก็บข้อมูลเชิงอธิบาย และตาราง gphonebooth สำหรับเก็บข้อมูลเชิงพื้นที่



รูปที่ 3-8 ความสัมพันธ์ระหว่างตาราง phonebooth และ gphonebooth

รายละเอียดของตาราง **phonebooth**

ชื่อคอลัมน์	ความหมาย	key
ID	หมายเลขอ้างอิงตู้โทรศัพท์	PK
Position	คำอธิบายรายละเอียดตำแหน่งที่ตั้ง	
Type	ชนิดของตู้(เหรียญ,บัตร)	

ตารางที่ 3-13 แสดงรายละเอียดของตาราง phonebooth

รายละเอียดของตาราง **gphonebooth**

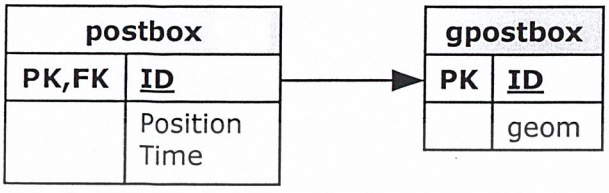
ชื่อคอลัมน์	ความหมาย	key
ID	หมายเลขอ้างอิงตู้โทรศัพท์	PK
geom	ข้อมูลเชิงภูมิศาสตร์ของตู้โทรศัพท์(point)	

ตารางที่ 3-14 แสดงรายละเอียดของตาราง gphonebooth

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. ตารางเก็บข้อมูลตู้ไปรษณีย์

เป็นตารางที่ใช้สำหรับเก็บข้อมูลรายละเอียดและตำแหน่งที่ตั้งของตู้ไปรษณีย์ต่างๆซึ่งแบ่งออกเป็น 2 ตาราง ได้แก่ ตาราง postbox สำหรับเก็บข้อมูลเชิงอธิบาย และตาราง gpostbox สำหรับเก็บข้อมูลเชิงพื้นที่



รูปที่ 3-9 ความสัมพันธ์ระหว่างตาราง postbox และ gpostbox

รายละเอียดของตาราง **postbox**

ชื่อคอลัมน์	ความหมาย	key
ID	หมายเลขอ้างอิงตู้ไปรษณีย์	PK
Position	คำอธิบายรายละเอียดตำแหน่งที่ตั้ง	
Time	เวลาไขตู้	

ตารางที่ 3-15 แสดงรายละเอียดของตาราง postbox

รายละเอียดของตาราง **gpostbox**

ชื่อคอลัมน์	ความหมาย	key
ID	หมายเลขอ้างอิงตู้ไปรษณีย์	PK
geom	ข้อมูลเชิงภูมิศาสตร์ของตู้ไปรษณีย์ (point)	

ตารางที่ 3-16 แสดงรายละเอียดของตาราง gpostbox

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1 การจัดเก็บข้อมูลเชิงพื้นที่ด้วย PostGIS

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

- POINT(0 0)
- LINESTRING(0 0,1 1,1 2)
- POLYGON((0 0,4 0,4 4,0 4,0 0),(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)))

เพื่อให้การเก็บข้อมูลนั้นเป็นไปตามมาตรฐานการเก็บข้อมูลเชิงพื้นที่ที่ OpenGIS ได้กำหนดเอาไว้จึงต้องมีตาราง 2 ตารางเพื่อใช้ในการเก็บเมต้าดาต้า 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 : เก็บตัวเลขของระบบอ้างอิงข้อมูลเชิงพื้นที่ซึ่งกำหนดโดยผู้ออกระบบนั้นเช่น กรณีของ 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",

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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_easting",500000],
PARAMETER["false_northing",0],
UNIT["metre",1]]

สามารถศึกษาข้อมูลเพิ่มเติมของ EPGS projection และ WKT ได้จาก

http://www.opengis.org/techno/interop/EPSG2WKT.TXT

http://www.opengis.org/techno/specs.htm

- ตาราง **GEOMETRY_COLUMNS**

ใช้สำหรับเก็บรายละเอียดของ Geometry Column ทั้งหมดในฐานข้อมูล ประกอบด้วยคอลัมน์ต่างๆดังต่อไปนี้

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

3.2.1.1 วิธีการจัดเก็บข้อมูลเชิงภูมิศาสตร์

1. **สร้างตารางสำหรับเก็บข้อมูลเชิงพื้นที่**

การเรียกใช้ฟังก์ชั่น

```
CREATE TABLE gatm ( ID int4 );
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. เพิ่มคอลัมน์ geom เพื่อเก็บข้อมูลเชิงพื้นที่

รูปแบบของฟังก์ชั่น

```
AddGeometryColumn(<ชื่อฐานข้อมูล>, <ชื่อตาราง>, <ชื่อคอลัมน์>,
                  <SRID>, <ชนิดของพื้นที่>, <มิติของข้อมูล>)
```

การเรียกใช้ฟังก์ชั่น

```
SELECT AddGeometryColumn('gis_db','gatm','geom', -1 , 'POINT', 2);
```

3. การจัดเก็บข้อมูลเชิงพื้นที่เข้าสู่ตาราง

รูปแบบของฟังก์ชั่น

```
GeomFromText(text,[<srid>])
```

การเรียกใช้ฟังก์ชั่น

```
INSERT INTO gatm (ID, GEOM) VALUES
        (atm001, GeomFromText('POINT(1512365 698511)', -1));
```

ขั้นตอนดังกล่าวเป็นการสร้างข้อมูลเชิงพื้นที่โดยใช้รูปแบบของ Well-Known Text และกำหนด SRID (ถ้าไม่กำหนดจะมีค่าเป็น -1)

3.2.2 ฟังก์ชั่นการทำงานของ PostGIS ที่เกี่ยวข้องกับข้อมูลเชิงพื้นที่

1. && (Intersects Operation)

เป็นฟังก์ชั่นที่ใช้ในการทดสอบว่ารูปทรงเรขาคณิตทั้งสองมีส่วนที่ซ้อนทับกันโดยการใช้โอเปอร์เรชั่น && ช่วยให้การค้นหาข้อมูลให้เร็วขึ้น เพราะ หากมีการทำอินเด็กซ์ไว้ก็จะทดสอบได้โดยใช้การใช้อินเด็กซ์

2. Expand(geometry, float)

เป็นฟังก์ชั่นที่ให้ผลลัพธ์เป็นการขยายขอบเขตออกจากตำแหน่งอ้างอิงออกไปในทุกทิศทางในรูปแบบของรูปทรงสี่เหลี่ยมจัตุรัสโดยใช้พารามิเตอร์ตัวที่สองเป็นรัศมีในการขยายขอบเขต

3. Distance(geometry,geometry)

เป็นฟังก์ชั่นที่ช่วยในการหาระยะห่างระหว่างรูปทรงเรขาคณิต 2 รูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการใช้งาน

```
SELECT *
FROM GEOTABLE
WHERE  GEOCOLUMN && Expand(GeomFromText('POINT(1000 1000)',-1),100)
        AND
        Distance(GeomFromText('POINT(1000 1000)',-1),GEOCOLUMN) < 100;
```

จากตัวอย่าง เป็นการค้นหาวัตถุที่อยู่ห่างจากจุด 1000,1000 ในรัศมี 100 (ในการค้นหาข้อมูลเชิงพื้นที่มักจะมีการใช้ทั้ง อินเตอร์เซกโอเปอร์เรชั่นและ spatial ฟังก์ชั่น โดยใช้โอเปอร์เรชั่นในการทดสอบจากอินเด็กซ์เพื่อจำกัดจำนวนเรคคอร์ดก่อนจะตรวจสอบด้วยฟังก์ชั่นเพื่อความถูกต้องแน่นอน การทำเช่นนี้จะทำให้การค้นหาข้อมูลเร็วขึ้น) จากจุดที่อ้างอิงอยู่

3.3 การออกแบบส่วนของ GIS เว็บเซอร์วิส

ในส่วนของ GIS เว็บเซอร์วิสเป็นส่วนที่ให้บริการเชิงพื้นที่สำหรับระบบสารสนเทศเชิงภูมิศาสตร์ (GIS) โดยจะให้บริการข้อมูลรายละเอียดของสถานที่ต่างๆ และแผนที่ในรูปแบบของภาษา SVG ซึ่งจะมีหน้าที่หลักในการทำงาน ดังต่อไปนี้

1. ติดต่อกับฐานข้อมูลเพื่อนำข้อมูลมาแสดงผลในรูปแบบต่างๆ
2. สร้างแผนที่ในรูปแบบเอกสาร SVG
3. ให้บริการข้อมูลในรูปแบบของเอกสาร XML
4. นำข้อมูลที่ได้รับมาจากผู้ใช้ซึ่งอยู่ในรูปแบบของเอกสาร XML ไปเก็บในฐานข้อมูล

3.3.1 การนำข้อมูลจากฐานข้อมูลมาสร้างเป็นเอกสาร XML

การเก็บข้อมูลต่างๆลงเอกสาร XML นั้นจะต้องมีความถูกต้องตามกฎเกณฑ์ที่นิยามไว้ในเอกสาร DTD โดยเอกสาร XML นั้นจะต้องมีการอ้างอิงถึงเอกสาร DTD ด้วยเพื่อตรวจสอบความถูกต้อง ในระบบนี้จะกำหนดกฎเป็นเลเยอร์ ทั้งหมดที่มีอยู่ในฐานข้อมูล และคอลัมน์ทุกคอลัมน์ในแต่ละเลเยอร์ โดยมีตัวอย่างของเอกสาร DTD ที่ใช้งาน ดังนี้

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT Atm (lat, long, position, bank, faculty)>
<!ATTLIST Atm
        id CDATA #REQUIRED >
<!ELEMENT Bank (lat, long, position, name)>
<!ATTLIST Bank
        id CDATA #REQUIRED >
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<!ELEMENT Canteen (lat, long, position, name, faculty, detail)>
<!ATTLIST Canteen
        id CDATA #REQUIRED >
<!ELEMENT Carpark (lat, long, position, faculty)>
<!ATTLIST Carpark
        id CDATA #REQUIRED >
<!ELEMENT Phoneboot (lat, long, position, type, amount)>
<!ATTLIST Phoneboot
        id CDATA #REQUIRED >
<!ELEMENT Place (Atm, Bank, Canteen, Carpark, Phoneboot, Shop)>
<!ELEMENT Shop (lat, long, position, name, type, faculty, detail)>
<!ATTLIST Shop
        id CDATA #REQUIRED >
<!ELEMENT amount EMPTY>
<!ELEMENT bank EMPTY>
<!ELEMENT detail EMPTY>
<!ELEMENT faculty EMPTY>
<!ELEMENT lat EMPTY>
<!ELEMENT long EMPTY>
<!ELEMENT name EMPTY>
<!ELEMENT position EMPTY>
<!ELEMENT type EMPTY>
```

ตัวอย่างของเอกสาร XML ที่จัดเก็บข้อมูลที่ได้มาจากฐานข้อมูล

```
<?xml version="1.0" encoding="UTF-8"?>
<Layer>
        <Atm id="atm003">
                <lat>691733</lat>
                <long>1518500</long>
                <position>in front of Canteen A</position>
                <Bank>Thaipanith</Bank>
                <faculty>Engineer</faculty>
        </Atm>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<Bank id=" bk001">
        <lat>691544</lat>
        <long>1517999</long>
        <position>in front of science</position>
        <name>Thaipanith_ladkrabung</name>
</Bank>
<Canteen id="ct001">
        <lat>691999</lat>
        <long>1517999</long>
        <position>beside of building A</position>
        <name>Canteen A</name>
        <faculty>Agricultural Technology</faculty>
        <detail>no detail now</detail>
</Canteen>
<Carpark id="cp001">
        <lat>691654</lat>
        <long>1517999</long>
        <position>at the orange gate</position>
        <faculty>Agricultural Industry</faculty>
</Carpark>
<Phoneboot id="ph011">
        <lat>691244</lat>
        <long>1517852</long>
        <position>infront of libraly</position>
        <type>coin&amp;card</type>
        <amount>4</amount>
</Phoneboot>
</Layer>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 การสร้างเว็บเซอร์วิส

ในเว็บเซอร์วิสนั้นจะต้องมีส่วนที่ใช้ในการจัดการกับ SOAP แมสเสจ ซึ่งในระบบนี้ผู้จัดทำได้เลือกใช้ Apache Axis สำหรับภาษาจาวาเป็นเครื่องมือในการช่วยสร้างเว็บเซอร์วิส โดยจะนำฟังก์ชันการให้บริการต่างๆที่สร้างขึ้นนำมาสร้างเป็นคลาสของอินเตอร์เฟสสำหรับฟังก์ชันเหล่านั้น

```
public interface WsInf {
        public boolean ws_UploadFile(String fileName);
        public byte[] ws_GetListLayerFile();
        public byte[] ws_GetLayerDataFile(Vector vSelPlace);
        public byte[] ws_GetSVG(Vector vSelPlace);
        public String[] ws_GetNearest(String place,float lat,float lon);
        public String[] ws_GetAreaPlace(String place,int area,float lat,float  lon);
}
```

โดยใช้ Java2WSDL ของ Apache Axis เป็นเครื่องในการช่วยสร้าง WSDL ไฟล์ จากอินเตอร์เฟซคลาสที่สร้างขึ้น และใช้ WSDL2Java สร้างคลาสที่ทำหน้าในการสร้าง และรับ SOAP แมสเสจ โดยสร้างจาก WSDL ไฟล์ จากนั้นจะประกาศบริการต่างๆไว้บนเซิร์ฟเวอร์เพื่อแจ้งให้ผู้ใช้บริการรับทราบบริการทั้งหมดที่มีใน GIS เว็บเซอร์วิส

3.3.3 ส่วนติดต่อกับฐานข้อมูล

การติดต่อกับฐานข้อมูล PostgreSQL โดยการใช้ภาษาจาวานั้นจะใช้ PostgreSQL JDBC เป็นไดรเวอร์ โดยติดตั้งแพคเกจ pg74.215.jdbc3.jar ไว้ภายในแอพพลิเคชันเพื่อให้สามารถเรียกใช้งานไดรเวอร์ได้ ซึ่งรายละเอียดสำคัญของการเขียนโปรแกรมติดต่อฐานข้อมูล โดยใช้ภาษา Java มีดังต่อไปนี้

1. กำหนดชนิดของไดรเวอร์

```
String driver = "org.postgresql.Driver";      // กำหนดชนิดของไดรเวอร์
Class.forName(driver);                        // โหลดไดรเวอร์
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. สร้างการเชื่อมต่อไปยังฐานข้อมูล

```
String url = "jdbc:postgresql://161.246.6.207:5432;
String database = “gis”;
String username = “postgresql”;
String password = “postgresql”;
Connection connection =
        DriverManager.getConnection(url+database,username,password);
```

3. สร้างสเตทเมนท์

```
Statement statement = connection.createStatement( );
```

4. ประมวลผลด้วยภาษา SQL และรับผลลัพธ์กลับมาเป็น Result Set

```
String SQL = "SELECT * FROM atm;
ResultSet rs = statement.executeQuery(SQL);
Rs.next();
```

5. ปิดการเชื่อมต่อกับฐานข้อมูล

```
Statement.close();
Connection.close();
```

3.3.4 การนำข้อมูลจากฐานข้อมูลมาสร้างเป็นแผนที่ในรูปแบบภาษา SVG

ในการสร้างแผนที่ SVG ของระบบจะเริ่มด้วยผู้ใช้ส่งค่าอาร์กิวเมนต์มาบอกยังเว็บเซอร์วิสว่าต้องการแผนที่ในเลเยอร์ใดบ้างโดยจะเป็นเลเยอร์ที่มีรูปร่างเป็นจุดจากนั้นเว็บเซอร์วิสจะทำการค้นหาข้อมูลเชิงพื้นที่จากฐานข้อมูลเพื่อให้ทราบตำแหน่งพิกัดของสถานที่ต่างแล้วจึงนำข้อมูลที่ได้ไปสร้างเป็นแผนที่ SVG โดยการเขียนจุดลงบนแผนที่หลักซึ่งแผนที่หลักนี้จะประกอบด้วย อาคาร(polygon), ถนน(line) และทางรถไฟ(line)เมื่อเติมจุดตามพิกัดที่ค้นมาได้จากฐานข้อมูลจนครบก็จะได้แผนที่ SVG ที่ผู้ใช้ต้องการ ซึ่งรายละเอียดของการเติมจุดลงในเอกสาร SVG มีดังนี้

```
<symbol overflow="visible" preserveAspectRatio="xMidYMid meet" id="point_atm">
        <circle r="10" style="fill: rgb(255,0,0); fill-opacity: 0.8;" cx="0" cy="0" />
</symbol>
```

ประกาศ อีลิเมนต์ symbol ลงในเอกสาร SVG เพื่อใช้กำหนดลักษณะของสถานที่ที่จะปรากฏในแผนที่SVGโดยสถานที่แต่ละเลเยอร์ก็จะมีสีของจุดที่ต่างกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<g visibility="hidden" id="Atm">
        <use transform="translate(692977 1517999)" xlink:href="#point_atm"
        xlink:type="simple" xlink:actuate="onRequest" id="atm001"
        xlink:show="replace" />
</g>
```

นำข้อมูลที่ได้จากจากฐานข้อมูลมาสร้างจุดโดยสร้างให้แต่ละเลเยอร์รวมกันอยู่ในกลุ่มเดียวกัน โดยใช้ <g> จากนั้นเรียกใช้งาน symbol ที่ได้ประกาศไว้โดยใช้ <use> โดยจะย้ายจุดไปยังตำแหน่งต่างๆ บนแผนที่ตามพิกัดที่ค้นได้มาโดยใช้ แอททริบิว transform และใช้ xlink:href เพื่อระบุว่าใช้ symbol ใด

3.3.5 การจัดการกับเอกสาร XML

ในกระบวนการนำข้อมูล XML มาใช้งานในแอพพลิเคชั่น นั้นจะมี XML Parser เป็นตัวกลางในการดึงข้อมูลจากเอกสาร XML และแอพพลิเคชั่น ซึ่งเป็น API ชนิดหนึ่ง โดย API ที่เลือกใช้ในระบบนี้ คือ JDOM ซึ่งเป็นไลบรารีที่ใช้ในภาษา Java โดย JDOM นั้นจะช่วยให้การเขียนโปรแกรมสำหรับการอ่าน การเข้าถึงโหนดต่างๆ และการเขียนเอกสาร XML มีความสะดวก และง่ายมากขึ้น โดย JDOM นั้นได้รวมข้อดีของ API ดั้งเดิม คือ DOM และ SAX เข้าไว้ด้วยกัน

3.4 การออกแบบส่วนของ GIS แอปพลิเคชั่น

ในส่วนของ GIS แอปพลิเคชั่น จะออกแบบให้มีความสามารถพื้นฐานในการแสดงผลและจัดการแผนที่ในรูปแบบ SVG และเพิ่มเติมส่วนที่มีความสามารถในการเรียกใช้บริการต่างๆจาก GIS เว็บเซอร์วิส ซึ่งรายละเอียดของส่วนแอปพลิเคชั่นจะกล่าวถึงในบทถัดไป

3.5 การออกแบบบริการต่างๆบนระบบ GIS เว็บเซอร์วิส

บริการทั้งหมดที่พัฒนาขึ้นในโครงงานนี้เป็นการให้บริการเชิงพื้นที่ (Location-Based Service) บนระบบสารสนเทศเชิงภูมิศาสตร์โดยได้มีการพัฒนาและออกแบบการให้บริการเพื่อให้เป็นประโยชน์แก่ผู้ใช้ทั้งหมด 6 บริการด้วยกัน ดังต่อไปนี้

1. บริการข้อมูลเลเยอร์ที่มีในฐานข้อมูล
2. บริการข้อมูลรายละเอียดและตำแหน่งที่ตั้งของสถานที่
3. บริการแผนที่ในรูปแบบ SVG
4. บริการค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการ
5. บริการค้นหาสถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุด
6. บริการนำข้อมูลจากผู้ใช้ไปเพิ่มและแก้ไขในฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งบริการทั้งหมดนี้ผู้ใช้สามารถเรียกดูบริการทั้งหมดได้โดยเข้าไปดูจาก http://161.246.6.209/axis/index.html ซึ่งเป็น URL ของเซิร์ฟเวอร์ต่อจากนั้นเลือกในหมวด View ซึ่งจะแสดงบริการทั้งหมดที่มีอยู่ในเซิร์ฟเวอร์ โดยแต่ละหัวข้อการให้บริการผู้ใช้สามารถรับทราบรายละเอียดของบริการต่างๆได้จาก WSDL ที่เซิร์ฟเวอร์สร้างขึ้นมาได้อย่างสะดวก

ตัวอย่างการแจ้งบริการใน WSDL

```
<wsdl:operation name="ws_GetNearest" parameterOrder="in0 in1 in2">
        <wsdl:input message="impl:ws_GetNearestRequest" name="ws_GetNearestRequest"/>
        <wsdl:output message="impl:ws_GetNearestResponse" name="ws_GetNearestResponse"/>
</wsdl:operation>
```

ระบุว่าบริการที่ชื่อ ws_GetNearest และต้องส่งผ่านค่าพารามิเตอร์มาในฟังก์ชัน 3 ค่า จากนั้นในบรรทัดถัดมาจะแจ้งว่าข้อความขาเข้าในการเรียกใช้บริการมีชื่อว่า ws_GetNearestRequest และข้อความขาออกในการส่งผลลัพธ์มีชื่อว่า ws_GetNearestResponse

```
<wsdl:message name="ws_GetNearestRequest">
        <wsdl:part name="in0" type="xsd:string"/>
        <wsdl:part name="in1" type="xsd:float"/>
        <wsdl:part name="in2" type="xsd:float"/>
</wsdl:message>
```

ต่อมาจะประกาศว่าเมื่อมีการเรียกใช้ฟังก์ชันจะต้องส่งผ่านค่าพารามิเตอร์มากี่ตัว และแต่ละตัวมีชนิดเป็นอะไรบ้าง ในที่นี้เป็นฟังก์ชันที่ต้องส่งค่าพารามิเตอร์เข้ามาทั้งหมด 3 ตัว และมีชนิดของตัวแปรเป็น string, float และ float ตามลำดับ

```
<wsdl:message name="ws_GetNearestResponse">
        <wsdl:part name="ws_GetNearestReturn" type="impl:ArrayOf_xsd_string"/>
</wsdl:message>
```

จากนั้นจะเป็นส่วนที่บอกว่าผลลัพธ์ในการเรียกใช้ฟังก์ชันจะคืนค่าเป็นตัวแปรชนิดอะไร ในที่นี้จะคืนค่าเป็นชนิดสตริงอาร์เรย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5.1 บริการข้อมูลเลเยอร์ที่มีในฐานข้อมูล

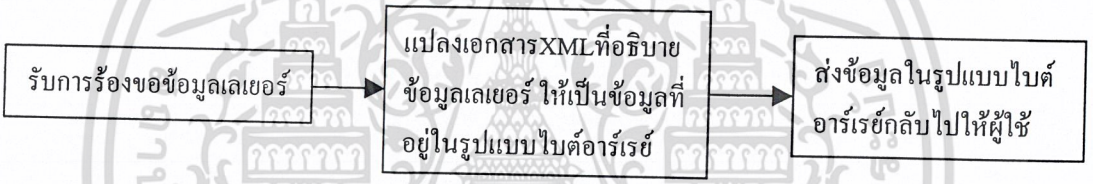
เซิร์ฟเวอร์จะสร้างโครงสร้างของเลเยอร์จากฐานข้อมูล โดยจะเก็บข้อมูลเฉพาะตารางที่มี geometry คอลัมน์ที่มีลักษณะเป็น point เท่านั้น ซึ่งจะเก็บไว้เป็นรูปแบบของเอกสาร XML จากนั้นเมื่อผู้ใช้ต้องการทราบว่าระบบมีข้อมูลของเลเยอร์ใดบ้างในฐานข้อมูล บริการนี้ก็จะให้บริการโดยการส่งไฟล์เอกสาร XML กลับไปให้กับผู้ใช้โดยเลเยอร์ที่มีในฐานข้อมูลมีทั้งหมด 8 เลเยอร์ ดังที่กล่าวไว้ในหัวข้อ 3.2

โดยผู้ใช้เรียกใช้บริการนี้ได้จากฟังก์ชัน ต่อไปนี้ ผ่านทางเว็บเซอร์วิส

```
public byte [ ] ws_GetListLayerFile ( )
```

ซึ่งผู้ใช้จะได้รับเอกสาร XML ที่อยู่ในรูปแบบของไบต์อาร์เรย์ จากนั้นผู้ใช้ต้องนำไบต์อาร์เรย์ที่ได้รับ ไปแปลงให้อยู่ในรูปแบบของไฟล์เอกสาร XML จึงจะได้ข้อมูลเลเยอร์ในฐานข้อมูลทั้งหมด

โดยสามารถสรุปขั้นตอนการให้บริการข้อมูลเลเยอร์ที่มีในฐานข้อมูลได้ดังนี้



รูปที่ 3-10 แสดงขั้นตอนการให้บริการข้อมูลเลเยอร์ที่มีในฐานข้อมูล

3.5.2 บริการข้อมูลรายละเอียดและตำแหน่งที่ตั้งของสถานที่

ผู้ใช้สามารถรับทราบรายละเอียด และตำแหน่งที่ตั้งของสถานที่ที่อยู่ในฐานข้อมูลได้ โดยผู้ใช้ต้องส่งค่าอาร์กิวเมนต์มาเป็นรายชื่อเลเยอร์ที่ผู้ใช้ต้องการรายละเอียดเป็นตัวแปรชนิดเวคเตอร์จากนั้นระบบจะทำการติดต่อกับฐานข้อมูลเพื่อดึงข้อมูลออกมาจากตารางทุกตารางที่ผู้ใช้เลือก และนำข้อมูลที่ได้มาสร้างเป็นเอกสาร XML เก็บไว้ก่อนโดยจะมีการตั้งชื่อให้กับเอกสาร XML นั้นๆ จะใช้ค่าของ timestamp ซึ่งเป็นเวลาที่ผู้ใช้แต่ละคนได้เรียกใช้บริการ

เหตุผลที่ตั้งชื่อไฟล์เป็นค่าของ timestamp เนื่องจากเมื่อมีการสร้างไฟล์เสร็จแล้วต้องมีการอ่านไฟล์นั้นอีกครั้งเพื่อส่งไฟล์กลับไปให้กับผู้ใช้ แต่ถ้าหากมีการกำหนดชื่อไฟล์ไว้ตายตัวจะทำให้เกิดความผิดพลาดขึ้นได้ในการอ่านและสร้างไฟล์ในกรณีที่มีผู้ใช้เรียกใช้บริการในเวลาที่ใกล้เคียงกันซึ่งอาจทำให้เกิดการซ้อนทับกันของไฟล์ทำให้ผู้ใช้ได้รับข้อมูลที่ผิดพลาด

กำหนดชื่อไฟล์เป็นค่าของ timestamp โดยรับค่า timestamp มาจากเวลาของเครื่องเซิร์ฟเวอร์

```
Timestamp ts = new Timestamp System.currentTimeMillis());
String tsfile   = Long.toString(ts.getTime( )) + ".xml";
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

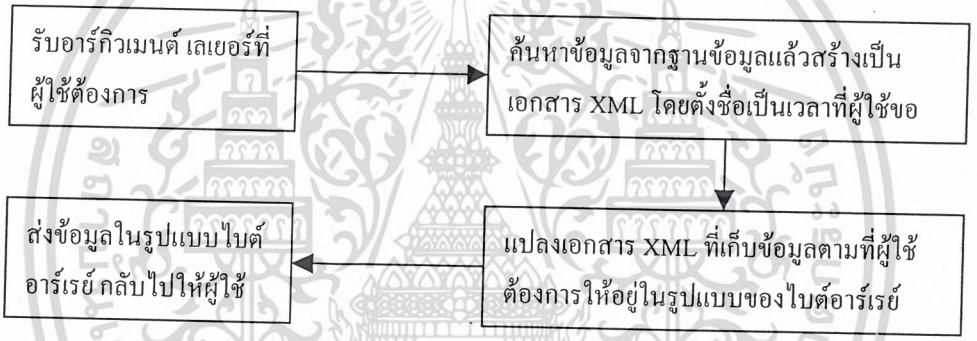
ตัวอย่างเช่น ถ้าผู้ใช้เรียกใช้บริการนี้ในเวลา 3:08 นาฬิกา 52 วินาที ก็จะได้ชื่อไฟล์เป็น "1110658109281.xml"

เมื่อสร้างไฟล์ XML เสร็จแล้วเซิร์ฟเวอร์จะกลับไปอ่านไฟล์ XML อีกครั้งหนึ่งเพื่อแปลงเอกสารที่สร้างขึ้นมาให้อยู่ในรูปแบบของไบต์อาร์เรย์เพื่อทำการส่งกลับไปเป็นผลลัพธ์ให้กับผู้ใช้นำไปแปลงเป็นเอกสาร XML ซึ่งบรรจุข้อมูลในทุกเลเยอร์ที่ผู้ใช้ต้องการ หลังจากนั้นเซิร์ฟเวอร์จะลบไฟล์นั้นทิ้งเพราะจะไม่มีการใช้ไฟล์นั้นอีก

โดยผู้ใช้เรียกใช้บริการนี้ได้จากฟังก์ชัน ต่อไปนี้ผ่านทางเว็บเซอร์วิส

```
public byte[ ] ws_GetPlaceDataFile(Vector vSelPlace)
```

ซึ่งสามารถสรุปขั้นตอนการบริการข้อมูลรายละเอียดและตำแหน่งที่ตั้งของสถานที่ได้ดังนี้



รูปที่ 3-11 แสดงขั้นตอนการให้บริการข้อมูลรายละเอียดและตำแหน่งที่ตั้งของสถานที่

3.5.3 บริการแผนที่ในรูปแบบ SVG

บริการนี้เป็นบริการที่ช่วยให้ผู้ใช้สามารถเลือกเลเยอร์ที่ต้องการเพื่อให้เว็บเซอร์วิสสร้างแผนที่ให้ในรูปแบบ SVG เช่น

- ตึกต่างๆภายในสถาบัน มี shape เป็น polygon
- ถนนภายในสถาบัน มี shape เป็น line
- ทางรถไฟภายในสถาบัน มี shape เป็น line
- สถานที่ต่างๆ ที่มีลักษณะของ shape เป็น point ได้แก่ เครื่องถอนเงินอัตโนมัติ, ธนาคาร, โรงอาหาร,ที่จอดรถ, ร้านค้าประเภทอาหารและเครื่องดื่ม, ห้องสมุด, ตู้โทรศัพท์สาธารณะ และตู้ไปรษณีย์

ในการเรียกบริการผู้ใช้ต้องส่งค่าอาร์กิวเมนต์มาเป็นเลเยอร์ที่ต้องการ และจำกัดว่าต้องเป็นเลเยอร์ที่มีรูปล่างเป็นจุด (point) เท่านั้นซึ่งได้แก่ ตู้ ATM ธนาคาร โรงอาหาร ที่จอดรถ ร้านค้าประเภทอาหารและเครื่องดื่ม ห้องสมุด ตู้โทรศัพท์สาธารณะและตู้ไปรษณีย์ จากนั้น เซิร์ฟเวอร์จะทำการติดต่อกับฐานข้อมูลเพื่อดึงข้อมูลมาสร้างเป็นแผนที่ SVG โดยใช้แผนที่ SVG ต้นฉบับที่ประกอบไปด้วย shape ที่เป็น

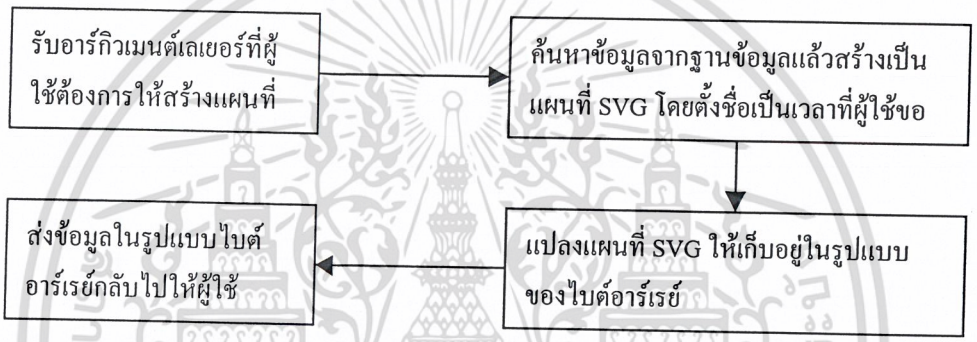
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

polygon และ line เท่านั้นจากนั้นจะเพิ่มเติมในส่วนที่เป็นจุดที่ผู้ใช้ต้องการในเลเยอร์ต่างที่ได้กล่าวไปแล้วข้างต้นเข้าไป จากนั้นตั้งชื่อแผนที่ SVG ที่ได้โดยใช้ชื่อเป็น timestamp เหมือนบริการข้อมูลรายละเอียดและตำแหน่งที่ตั้งของสถานที่ และส่งผลลัพธ์กลับไปให้กับผู้ใช้ในรูปแบบไบต์อาร์เรย์ ซึ่งผู้ใช้ต้องนำไบต์อาร์เรย์ที่ได้แปลงเป็น SVG เช่นเดียวกับการแปลงเอกสาร XML

โดยผู้ใช้เรียกใช้บริการนี้ได้จากฟังก์ชัน ต่อไปนี้ ผ่านทางเว็บเซอร์วิส

```
public byte[] ws_GetSVG(Vector vSelPlace);
```

ซึ่งสามารถสรุปขั้นตอนการบริการแผนที่ในรูปแบบ SVG ได้ดังนี้



รูปที่ 3-12 แสดงขั้นตอนการให้บริการแผนที่ในรูปแบบ SVG

3.5.4 บริการค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการ

บริการนี้จะช่วยให้ผู้ใช้สามารถค้นหาสถานที่ที่ต้องการในเลเยอร์ต่างๆ ว่าสามารถอยู่ในตำแหน่งใดได้บ้างภายในบริเวณรัศมี 100 , 200 , 500 และ 1000 เมตรรอบตัวผู้ใช้ เพื่อช่วยเป็นข้อมูลในการตัดสินใจในการเดินทางที่จะทำให้เกิดความสะดวกที่สุด

ในการเรียกบริการผู้ใช้ต้องส่งค่าอาร์กิวเมนต์มาเป็นเลเยอร์ที่ผู้ใช้ต้องการค้นหา ซึ่งตำแหน่งที่ผู้ใช้อยู่และบริเวณรัศมีรอบตำแหน่งที่ผู้ใช้อยู่ จะเป็นค่าในพิกัดมาตรฐาน UTM จากนั้นระบบจะไปค้นหาในฐานข้อมูล โดยใช้ภาษา SQL เป็นคำสั่งในการค้นหาข้อมูล ดังนี้

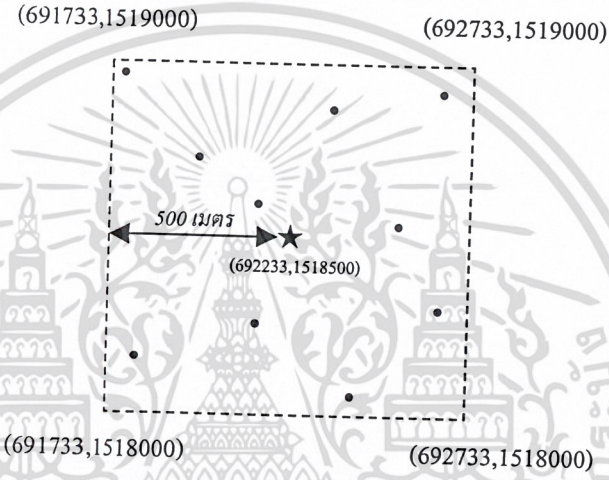
```
SQL = "SELECT *
        FROM Atm p1, gAtm p2
        WHERE (p2.geom &&
                Expand(GeometryFromText ('POINT(692233  1518500)',-1),  500 )
                AND Distance(GeometryFromText('POINT(692233 1518500)',-1),p2.geom)
                < 500 )
                AND (p1.id = p2.id)";
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำอธิบายภาษา SQL

ในการค้นหาจะใช้ข้อมูลจาก 2 ตาราง คือ ตารางเชิงอธิบาย กับตารางเชิงพื้นที่ ได้แก่ Atm p1, gAtm p2 ตามลำดับ โดยกำหนดให้แทนตารางเชิงอธิบายด้วย p1 และตารางเชิงพื้นที่แทนด้วย p2 ซึ่งจะมีการเชื่อมความสัมพันธ์ของทั้งสองตารางด้วย คอลัมน์ id ดังนี้ p1.id = p2.id

หลังจากนั้นจะทำการขยายขอบเขตออกจากตำแหน่งที่ผู้ใช้อยู่เป็นพื้นที่สี่เหลี่ยมจัตุรัสโดยการใช้ฟังก์ชั่น Exapand (gemotry,bound) ซึ่งเป็นฟังก์ชันการทำงานของ PostGIS มาช่วยในการขยายพื้นที่โดยใส่อาร์กิวเมนต์เป็นตำแหน่งบริเวณที่ผู้ใช้อยู่ และรัศมีที่ผู้ใช้ต้องการก็จะได้ผลลัพธ์เป็นการขยายขอบเขตพื้นที่ออกเป็นสี่เหลี่ยมจัตุรัสในรัศมีนั้นๆ ดังรูปที่ 3-13



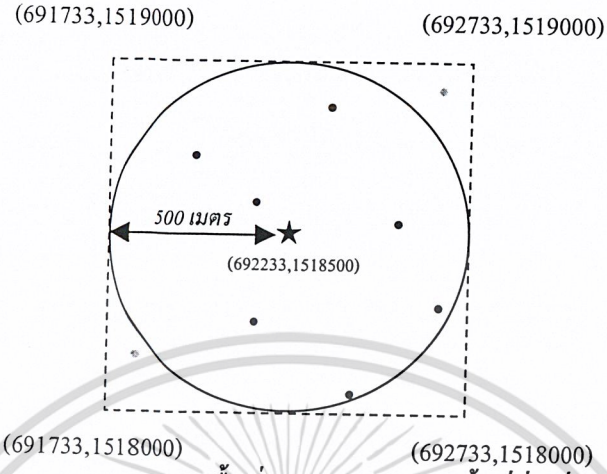
รูปที่ 3-13 แสดงการขยายขอบเขตของพื้นที่ออกเป็นสี่เหลี่ยมจัตุรัสในรัศมีที่ต้องการ

โดยจะนำคอลัมน์ geom (Geometry Column) จากตารางเชิงพื้นที่มา && กับฟังก์ชัน Expand เพื่อเลือกแถวที่มีข้อมูลตำแหน่งอยู่ในบริเวณ (intersect) ที่ขยายออก แต่จะสังเกตเห็นได้ว่าพื้นที่ที่ขยายออกมานั้นไม่ใช่ขอบเขตพื้นที่ที่ถูกต้อง ซึ่งในจุดประสงค์ของบริการนี้ต้องการให้เป็นบริเวณวงกลมรอบตำแหน่งของผู้ใช้ ดังนั้น จึงเพิ่มเงื่อนไขเข้าไปอีก 1 เงื่อนไข คือตรวจสอบระยะทางว่าจุดใดๆที่มีระยะทางจากจุดนั้นๆถึงจุดที่ผู้ใช้อยู่มีระยะทางน้อยกว่ารัศมีที่ผู้ใช้ต้องการ ฉะนั้นจุดที่ตรงตามเงื่อนไขเหล่านั้นจะอยู่ในบริเวณรัศมีที่ผู้ใช้ต้องการ โดยใช้ฟังก์ชัน Distance (geometry , geometry) ในการหาระยะทางระหว่างจุดสองจุด ซึ่งมีวิธีการใช้ ดังนี้

```
Distance(GeometryFromText('POINT(692233 1518500)',-1),p2.geom) < 500
```

สุดท้ายผลลัพธ์ที่ได้เมื่อเรียกใช้ภาษา SQL นี้ คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



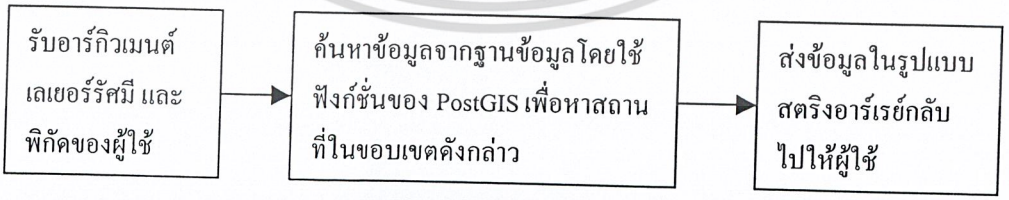
รูปที่ 3- 14 แสดงการขยายขอบเขตของพื้นที่ออกเป็นวงกลมในพื้นที่สี่เหลี่ยมจัตุรัสในรัศมีที่ต้องการ

ดังนั้น เราจะได้สถานที่ทุกแห่งที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการ โดยมีตำแหน่งที่ผู้ใช้อยู่เป็นจุดศูนย์กลางของรัศมีในวงกลม เมื่อประมวลผลเสร็จแล้วบริการนี้จะนำ Result Set ที่ได้นำมาสร้างเป็นสตริงอาร์เรย์ เพื่อเก็บผลลัพธ์ของการค้นหาที่ได้ส่งกลับไปยังผู้ใช้เพื่อให้ผู้ใช้นำไปแสดงผลเป็นตารางข้อมูลต่อไป

โดยผู้ใช้เรียกใช้บริการนี้ได้จากฟังก์ชัน ต่อไปนี้ ผ่านทางเว็บเซอร์วิส

```
public String[ ] ws_GetAreaPlace (String place,int area,float lat,float lon)
```

ซึ่งสามารถสรุปขั้นตอนการบริการค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการได้ดังนี้



รูปที่ 3-15 แสดงขั้นตอนการให้บริการค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการ

3.5.5 บริการค้นหาสถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุด

บริการนี้เป็นบริการที่ช่วยให้ผู้ใช้สามารถค้นหาสถานที่ที่ต้องการในเลเยอร์ต่างๆ ที่อยู่ใกล้กับตำแหน่งที่ผู้ใช้อยู่มากที่สุดได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการเรียกบริการผู้ใช้ต้องส่งค่าอาร์กิวเมนต์มาเป็นเลเยอร์ที่ผู้ใช้ต้องการค้นหา ซึ่งตำแหน่งที่ผู้ใช้อยู่ จะเป็นค่าในพิกัดมาตรฐาน UTM จากนั้นระบบจะไปค้นหาในฐานข้อมูล โดยใช้ภาษา SQL เป็นคำสั่งในการค้นหาข้อมูล ดังนี้

```
SQL =  "SELECT *
        FROM Atm p1 , gAtm p2
        WHERE  p1.id = ANY (SELECT id
                            FROM gAtm
                            WHERE Distance('POINT(692233 1518500)',geom) <=
                            ALL (SELECT Distance('POINT(692233 1518500)',geom)
                            FROM     gAtm))
                        AND p2.id = p1.id";
```

คำอธิบายภาษา SQL

ในการค้นหาจะใช้ข้อมูลจาก 2 ตาราง คือ ตารางเชิงอธิบาย กับตารางเชิงพื้นที่ ได้แก่ Atm p1, gAtm p2 ตามลำดับ โดยกำหนดให้แทนตารางเชิงอธิบายด้วย p1 และตารางเชิงพื้นที่แทนด้วย p2 ซึ่งจะมีการเชื่อมความสัมพันธ์ของทั้งสองตารางด้วย คอลัมน์ id ดังนี้ p1.id = p2.id

โดยจะใช้หลักการของ Sub Queries ในการหาระยะทางโดยเทียบกันระหว่างจุดที่ผู้ใช้อยู่กับพิกัดในแต่ละแถว เมื่อได้ระยะทางของทุกแถวแล้ว จากนั้นจะเลือกว่าแถวไหนมีระยะทางน้อยที่สุดเมื่อเทียบกับทุกๆ แถวของข้อมูลก็จะได้สถานที่ที่มีตำแหน่งอยู่ใกล้กับจุดที่ผู้ใช้อยู่หรือจุดอ้างอิงมากที่สุด ดังรูปที่ 3-16

รูปที่ 3-16 แสดงตำแหน่งที่อยู่ใกล้กับจุดอ้างอิงมากที่สุด

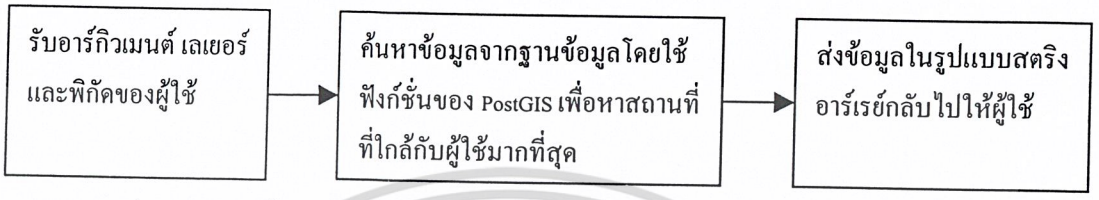
ดังนั้น เราจะได้สถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุดต้องการ เมื่อประมวลผลเสร็จแล้ว บริการนี้จะนำ Resultset ที่ได้นำมาสร้างเป็นสตริงอาร์เรย์ เพื่อเก็บผลลัพธ์ของการค้นหาที่ได้ส่งกลับไปยังผู้ใช้เพื่อให้ผู้ใช้นำไปแสดงผลเป็นตารางข้อมูลต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยผู้ใช้เรียกใช้บริการนี้ได้จากฟังก์ชัน ต่อไปนี้ ผ่านทางเว็บเซอร์วิส

```
public String[ ] ws_GetNearest(String place,float lat,float lon)
```

ซึ่งสามารถสรุปขั้นตอนการบริการค้นหาสถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุดได้ดังนี้



รูปที่ 3-17 แสดงขั้นตอนการให้บริการค้นหาสถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุด

3.5.6 บริการนำข้อมูลจากผู้ใช้ไปเพิ่มและแก้ไขในฐานข้อมูล

บริการนี้ทำให้ผู้ใช้สามารถเพิ่มและแก้ไขข้อมูลที่ได้รับไปจากฐานข้อมูลได้ จากนั้นผู้ใช้จะส่งข้อมูลที่ได้รับการเพิ่มและแก้ไขแล้วซึ่งอยู่ในรูปแบบของเอกสาร XML มาให้กับระบบ เพื่อฝากให้ระบบเว็บเซอร์วิสบริการนำข้อมูลไปเพิ่มและแก้ไขในฐานข้อมูลให้

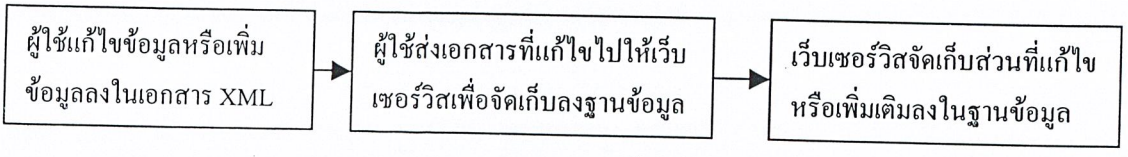
เมื่อเซิร์ฟเวอร์ได้รับไฟล์เอกสาร XML จากผู้ใช้แล้วเซิร์ฟเวอร์จะนำมาอ่านเอกสารด้วย JDOM จากนั้นจะทำการตรวจสอบว่าเรคคอร์ดไหนมีการเพิ่มก็จะใช้ SQL คิวรี่เป็นคำสั่งในการ INSERT ข้อมูลเข้าไปในดาตาเบส ส่วนเรคคอร์ดไหนที่เป็นการแก้ไขข้อมูลก็จะใช้ SQL คิวรี่เป็นคำสั่งในเป็นการ UPDATE ข้อมูลในฐานข้อมูล

โดยผู้ใช้เรียกใช้บริการนี้ได้จากฟังก์ชัน ต่อไปนี้ ผ่านทางเว็บเซอร์วิส

```
public boolean ws_UploadFile(String fileName)
```

ซึ่งทำให้ผู้ใช้สามารถทำการเพิ่มเติมและแก้ไขข้อมูลต่างๆ ให้มีความถูกต้องและให้มีความเป็นปัจจุบันมากที่สุดได้ง่ายสะดวกและง่ายต่อการนำไปพัฒนาในอนาคต

ซึ่งสามารถสรุปขั้นตอนการบริการนำข้อมูลจากผู้ใช้ไปเพิ่มและแก้ไขในฐานข้อมูลได้ดังนี้



รูปที่ 3-18 แสดงขั้นตอนการให้บริการนำข้อมูลจากผู้ใช้ไปเพิ่มและแก้ไขในฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การออกแบบ GIS แอพพลิเคชัน

4.1 โครงสร้างการทำงานของ GIS แอพพลิเคชัน

GIS แอพพลิเคชันสร้างขึ้นเพื่อนำมาทดสอบการเรียกใช้บริการต่างๆจาก GIS เว็บเซอร์วิส ซึ่งการติดต่อกับเว็บเซอร์วิสนั้นจะติดต่อผ่านส่วนการทำงานของ SOAP Proxy ซึ่งในโครงงานนี้ได้เลือกใช้เป็น Apache Axis API ในการจัดการเกี่ยวกับการรับส่ง SOAP แมสเสจ

โดย GIS แอพพลิเคชันจะมีความสามารถหลักๆ ดังนี้

1. ส่วนแสดงผลแผนที่ และเครื่องมือต่างๆในการจัดการกับแผนที่
2. ส่วนแสดงผลข้อมูลของสถานที่ต่างๆ
3. ส่วนค้นหาพิกัดตำแหน่งของผู้ใช้
4. ส่วนที่เรียกใช้บริการจาก GIS เว็บเซอร์วิส

4.1.1 ส่วนแสดงผลแผนที่ และเครื่องมือต่างๆในการจัดการกับแผนที่

ในส่วนการแสดงผลแผนที่นั้นจะใช้ Batik ซึ่งเป็นชุดเครื่องมือภาษาจาวาที่ช่วยในการใช้รูปภาพ SVG ในจุดประสงค์ต่างๆ โดยจะใช้คลาส JSVGCanvas ซึ่งเป็นคลาสๆหนึ่งในชุดของ Batik เป็นพาเนลในการแสดงผลแผนที่ในรูปแบบของ SVG

เครื่องมือต่างๆที่ใช้จัดการกับแผนที่ SVG ได้แก่

1. ส่วนการย่อ/ขยาย และเลื่อนแผนที่
2. ส่วนเปิด/ปิดการแสดงผลแต่ละเลเยอร์ในแผนที่
3. ส่วนเปิดและบันทึกแผนที่

4.1.2 ส่วนแสดงผลข้อมูลต่างๆของสถานที่

ข้อมูลต่างๆที่นำมาแสดงผลเป็นข้อมูลที่อ่านมาจากเอกสาร XML ซึ่งได้รับมาจากการเรียกใช้บริการขอข้อมูลสถานที่จาก GIS เว็บเซอร์วิส โดยแสดงผลทั้งอยู่ในรูปแบบของตาราง และรูปแบบพาเนล

4.1.3 ส่วนค้นหาพิกัดตำแหน่งของผู้ใช้

ส่วนค้นหาพิกัดของผู้ใช้ประกอบด้วย GPS รีซีฟเวอร์ต่ออยู่กับพอร์ทอนุกรมของเครื่องไคลเอนท์ ซึ่ง GIS แอพพลิเคชันจะรับข้อมูลจาก GPS รีซีฟเวอร์ ผ่านทางพอร์ทอนุกรมหลังจากนั้นจึงทำการวิเคราะห์เอาเฉพาะข้อมูลส่วนที่เป็นตำแหน่งของผู้ใช้งาน โดยคำสั่งภาษาจาวาที่ใช้ในการอ่านค่าจากพอร์ทอนุกรมอยู่ในคลาส SerialPort ซึ่งอยู่ในแพคเกจ javax.comm ดังนั้นจึงต้องทำการ import แพคเกจนี้เข้ามาในส่วนของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยค่าที่อ่านได้นั้นจะใช้ส่วนที่มีเฮดเดอร์เป็น $GPGGA เท่านั้น และทำการดึงข้อมูลในส่วนของพิกัดออกมาใช้งาน ซึ่งพิกัดที่ได้รับจาก GPS รีซีฟเวอร์ นั้นเป็นพิกัดที่อยู่ในมาตรฐานของ WGS-84 โดยมีหน่วยวัดเป็นละติจูด และลองติจูด แต่ในระบบฐานข้อมูลที่เก็บข้อมูลเชิงพื้นที่นั้นในส่วนของ geometry คอลัมน์ จะเก็บพิกัดมาตรฐานของ UTM ซึ่งมีหน่วยเป็นเมตร ดังนั้นจึงต้องเพิ่มการเขียนโปรแกรมในส่วนของการแปลงมาตรฐานพิกัดจาก WGS-84 เป็นมาตรฐาน UTM โดยในโครงงานนี้ได้เลือกใช้ไลบรารีของ openmap สามารถศึกษารายละเอียดเพิ่มเติมได้จากเว็บไซต์ www.openmap.org

รายละเอียดการเขียนโปรแกรมภาษาจาวาแปลงมาตรฐานพิกัด มีดังนี้

```
float latitude = (float) 13.728833;
float longitude = (float) 100.775583;
LatLonPoint latlon = new LatLonPoint(lat,lon);
UTMPoint utm = UTMPoint.LLtoUTM(latlon);
```

4.1.4 ส่วนที่เรียกใช้บริการจาก GIS เว็บเซอร์วิส

การติดต่อกับเว็บเซอร์วิสจากแอพพลิเคชันจะต้องติดต่อกันผ่าน SOAP Proxy ซึ่ง SOAP Proxy นั้นจะทำหน้าที่เป็นตัวกลางในการรับ และส่ง SOAP แมสเสจ โดยฝั่งขาออกจะแปลงจาวาออปเจกต์ให้อยู่ในรูปแบบของ SOAP Request และในฝาขาเข้าจะแปลงจาก SOAP Response ให้กลายเป็น จาวาออปเจกต์ เพื่อนำมาใช้งานต่อไป

4.2 การสร้างส่วนการเรียกใช้งานเว็บเซอร์วิส

การที่ผู้ใช้จะสามารถเรียกใช้บริการจากเว็บเซอร์วิสได้นั้นผู้ใช้จะต้องรับรู้ถึงบริการต่างๆที่มีในเว็บเซอร์วิสนั้นๆ ก่อนโดยดูจากประกาศบนเซิร์ฟเวอร์ที่เราต้องการใช้บริการ และยังสามารถรับรู้ถึงการเรียกใช้ และรายละเอียดต่างๆของบริการผ่านทาง WSDL ไฟล์ อีกด้วย

ในการสร้างส่วนติดต่อกับเว็บเซอร์วิสบนแอพพลิเคชันนั้นจะต้องนำ WSDL ไฟล์จากเว็บเซอร์วิสมาสร้าง SOAP Proxy โดยใช้เครื่องมือของ Apache Axis ชื่อว่า WSDL2Java เพื่อนำมาสร้างจาวาคลาสที่ใช้ในการจัดการกับ SOAP แมสเสจ ซึ่งมีวิธีการ ดังนี้

```
java org.apache.axis.wsdl.WSDL2Java -o . -d Session -s -S true -Nurn:lbs lbs.ws lbs.wsdl
```

หลังจากที่ใช้ WSDL2Java แล้วจะได้จาวาคลาสที่ทำหน้าที่เป็น SOAP Proxy ให้กับตัวไคลเอนท์ ดังนี้

LbsSoapBindingStub.java : ทำหน้าที่เป็น stub ให้ฝั่งไคลเอนท์

WsInf.java : เป็นไฟล์อินเทอร์เฟซตัวใหม่ที่มีการใช้ java.rmi.Remote

WsInfService.java : เป็นไฟล์จาวาที่เป็นอินเทอร์เฟซบริการด้านไคลเอนท์

WsInfServiceLocator.java : เป็นไฟล์จาวาที่ทำหน้าที่ระบุที่อยู่ของเซิร์ฟเวอร์ที่จะเรียกใช้บริการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การเรียกใช้บริการต่างๆจาก GIS เว็บเซอร์วิส

4.3.1 เรียกใช้บริการข้อมูลเลเยอร์ที่มีในฐานข้อมูล

ผู้ใช้สามารถรับทราบเกี่ยวกับข้อมูลเลเยอร์ที่มีทั้งหมดในระบบฐานข้อมูลได้ว่ามีเลเยอร์อะไรบ้าง ซึ่งอยู่ในรูปแบบของไบต์อาร์เรย์ โดยผู้ใช้จะต้องนำไบต์อาร์เรย์นี้มาแปลงให้อยู่ในรูปแบบเอกสาร XML โดยคำสั่งภาษาจาวาที่ใช้ในการเขียนไฟล์นั้นอยู่ในคลาส FileOutputStream ซึ่งมีรายละเอียดการเขียนโปรแกรม ดังนี้

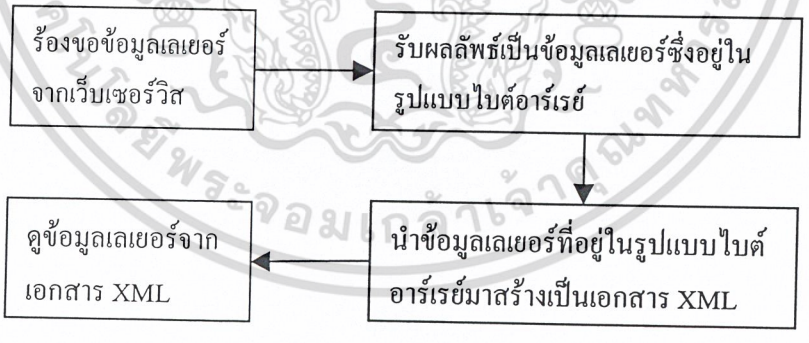
```
byte[ ] bytesLayer = ws_GetListLayerFile();
FileOutputStream fout = new FileOutputStream("layer.xml");
fout.write(bytesLayer,0,bytesLayer.length);
```

โดยผู้เรียกใช้บริการนี้จะส่งค่าอาร์กิวเมนต์ต่างๆ เข้าไปในฟังก์ชันผ่านทางเว็บเซอร์วิสดังนี้

```
public byte [ ] ws_GetListLayerFile ( )
```

สุดท้ายผู้ใช้จะได้รับเอกสาร XML ที่อยู่ในรูปแบบของไบต์อาร์เรย์ จากนั้นผู้ใช้ต้องนำไบต์อาร์เรย์ที่ได้รับไปแปลงให้อยู่ในรูปแบบของไฟล์เอกสาร XML จึงจะได้ข้อมูลเลเยอร์ในฐานข้อมูลทั้งหมด

ซึ่งสามารถสรุปขั้นตอนการเรียกใช้บริการข้อมูลเลเยอร์ที่มีในฐานข้อมูลได้ดังนี้



รูปที่ 4-1 แสดงขั้นตอนการเรียกใช้บริการข้อมูลเลเยอร์ที่มีในฐานข้อมูล

4.3.2 การเรียกใช้บริการข้อมูลรายละเอียดและตำแหน่งที่ตั้งของสถานที่

ผู้ใช้สามารถรับทราบรายละเอียด และตำแหน่งที่ตั้งของสถานที่ที่อยู่ในฐานข้อมูลในแต่ละเลเยอร์ที่ต้องการได้

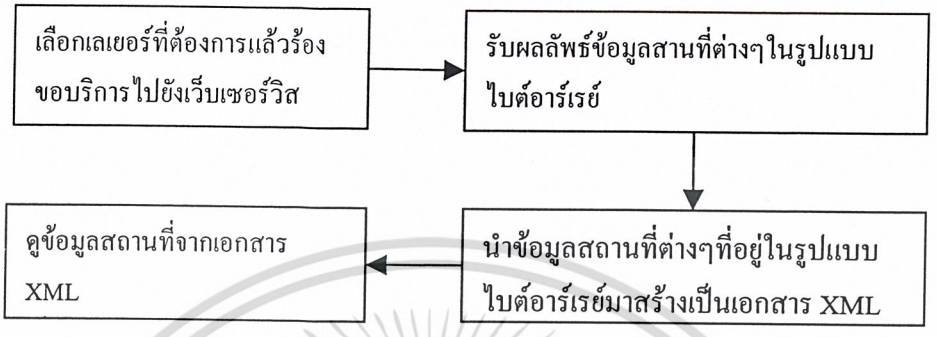
โดยผู้เรียกใช้บริการนี้จะส่งค่าอาร์กิวเมนต์ต่างๆ เข้าไปในฟังก์ชันผ่านทางเว็บเซอร์วิสดังนี้

```
public byte[ ] ws_GetPlaceDataFile(Vector vSelPlace)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สุดท้ายผู้ใช้จะได้รับเอกสาร XML ที่อยู่ในรูปแบบไบต์อาร์เรย์เช่นเดียวกับในบริการข้อมูลเลเยอร์ที่มีในฐานข้อมูลจากนั้นจึงนำไบต์อาร์เรย์นี้ไปแปลงเป็นเอกสาร XML เช่นกัน

ซึ่งสามารถสรุปขั้นตอนการเรียกใช้บริการข้อมูลรายละเอียดและตำแหน่งที่ตั้งของสถานที่ได้ดังนี้



รูปที่ 4-2 แสดงขั้นตอนการเรียกใช้บริการข้อมูลรายละเอียดและตำแหน่งที่ตั้งของสถานที่

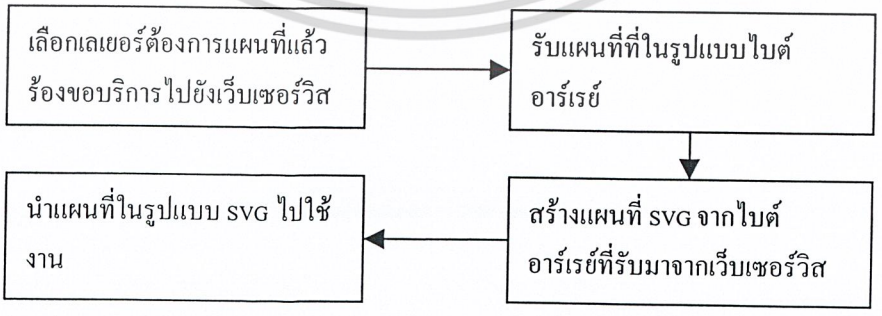
4.3.3 การเรียกใช้บริการแผนที่ในรูปแบบ SVG

ผู้ใช้สามารถเลือกเลเยอร์ที่ต้องการแสดงผลเป็นแผนที่เพื่อให้เว็บเซอร์วิสนำไปสร้างแผนที่ในรูปแบบ SVG ได้ซึ่งเลเยอร์ที่ผู้ใช้สามารถเลือกได้นั้นจำกัดว่าต้องเป็นเลเยอร์ที่มี shape เป็น point เท่านั้นซึ่งได้แก่ ตู้ ATM , ธนาคาร ,โรงอาหาร,ที่จอดรถ,ร้านค้าประเภทอาหารและเครื่องดื่ม,ห้องสมุด,ตู้โทรศัพท์สาธารณะและตู้ไปรษณีย์จากนั้นจะนำไปแสดงผลในส่วนที่แสดงผลแผนที่

โดยผู้เรียกใช้บริการนี้จะส่งค่าอาร์กิวเมนต์ต่างๆ เข้าไปในฟังก์ชันผ่านทางเว็บเซอร์วิสดังนี้

```
public byte[] ws_GetSVG(Vector vSelPlace);
```

ซึ่งสามารถสรุปขั้นตอนการเรียกใช้บริการแผนที่ในรูปแบบ SVG ได้ดังนี้

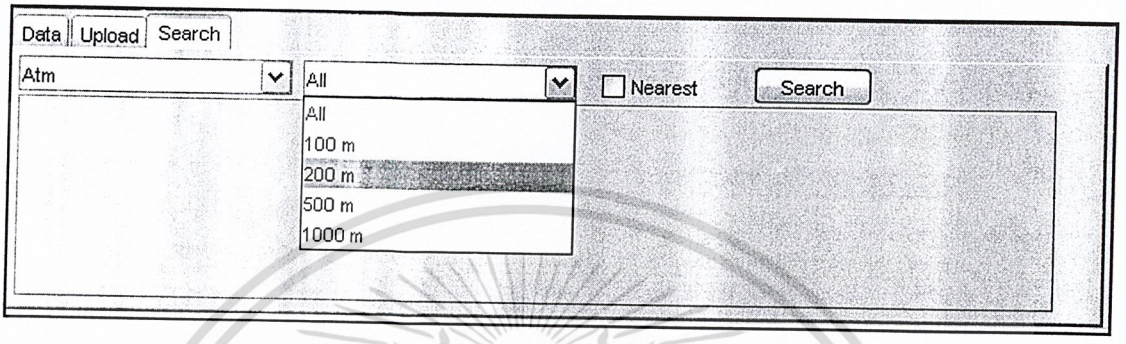


รูปที่ 4-3 แสดงขั้นตอนการเรียกใช้บริการแผนที่ในรูปแบบ SVG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.4 การเรียกใช้บริการค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการ

ผู้ใช้สามารถค้นหาสถานที่ที่ต้องการในเลเยอร์ต่างๆ ที่อยู่ภายในบริเวณขอบเขตรัศมี 100, 200,500 และ 1000 เมตร ที่ผู้ใช้ต้องการได้โดยการเลือกเลเยอร์ที่ต้องการค้นหาและขอบเขตรัศมีที่ต้องการในแท็บ Search ด้านล่างของเฟรมจากนั้นต้องกดปุ่ม LockGPS เพื่อเป็นตำแหน่งของจุดอ้างอิงของผู้ใช้ด้วยแล้วกดปุ่ม Search ดังรูปที่ 4-4



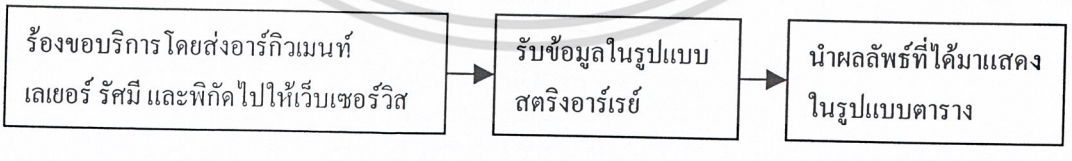
รูปที่ 4-4 แสดงการเรียกใช้บริการค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการ

โดยผู้เรียกใช้บริการนี้จะส่งค่าอาร์กิวเมนต์ต่างๆ เข้าไปในฟังก์ชันผ่านทางเว็บเซอร์วิสดังนี้

```
public String[ ] ws_GetAreaPlace (String place,int area,float lat,float lon)
```

ซึ่งผลลัพธ์ที่ได้รับกลับมาจะเป็นสตริงอาร์เรย์จากนั้นโปรแกรมจะนำผลลัพธ์ที่ได้นั้นมาสร้างเป็นตารางเพื่อแสดงผลการค้นหาให้กับผู้ใช้ต่อไป

โดยสามารถสรุปขั้นตอนการเรียกใช้บริการค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการได้ดังนี้

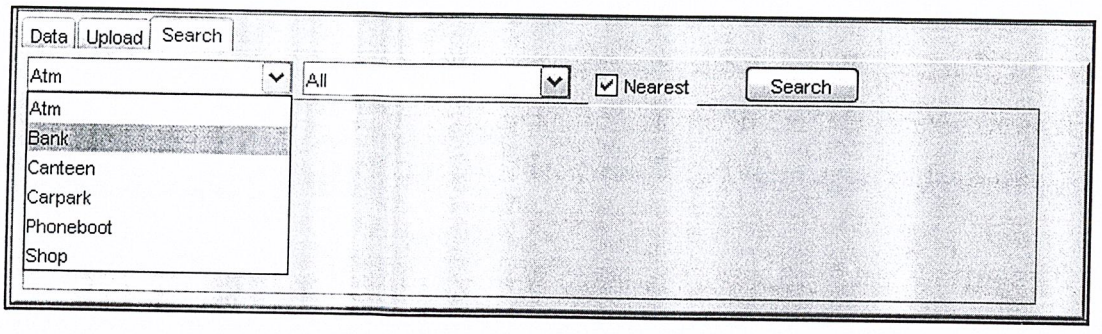


รูปที่ 4-5 แสดงขั้นตอนการเรียกใช้บริการค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการ

4.3.5 การเรียกใช้บริการค้นหาสถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุด

ผู้ใช้สามารถค้นหาสถานที่ที่ต้องการในเลเยอร์ต่างๆ ที่อยู่ใกล้กับตำแหน่งจุดอ้างอิงของผู้ใช้มากที่สุดโดยการเลือกเลเยอร์ที่ต้องการค้นหาที่อยู่ในแท็บ Search ด้านล่างของเฟรม จากนั้นคลิกที่กล่อง Nearest แล้วกดปุ่ม Search ดังรูปที่ 4-6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



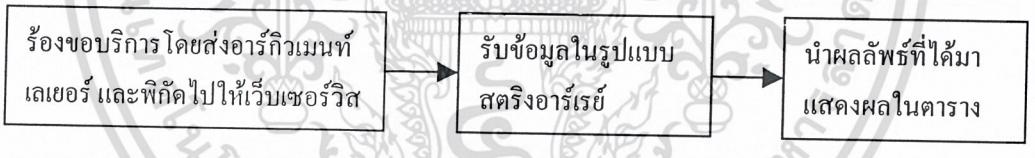
*รูปที่ 4-6 แสดง*การเรียกใช้บริการค้นหาสถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุด

โดยผู้เรียกใช้บริการนี้จะส่งค่าอาร์กิวเมนต์ต่างๆ เข้าไปในฟังก์ชันผ่านทางเว็บเซอร์วิสดังนี้

```
public String[ ] ws_GetNearest(String place,float lat,float lon)
```

ซึ่งผลลัพธ์ที่ได้รับกลับมาจะเป็นสตริงอาร์เรย์จากนั้นโปรแกรมจะนำผลลัพธ์ที่ได้นั้นมาสร้างเป็นตารางเพื่อแสดงผลการค้นหาให้กับผู้ใช้ต่อไป

โดยสามารถสรุปขั้นตอนการเรียกใช้บริการค้นหาสถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุดได้ดังนี้



รูปที่ 4-7 แสดงขั้นตอนการเรียกใช้บริการค้นหาสถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุด

4.3.6 การเรียกใช้บริการนำข้อมูลจากผู้ใช้ไปเพิ่มและแก้ไขในฐานข้อมูล

ผู้ใช้สามารถทำการเพิ่มข้อมูลหรือแก้ไขข้อมูลที่ได้รับไปจากฐานข้อมูลได้ผ่านทางหน้าต่างของการเพิ่ม และแก้ไขข้อมูล โดยโปรแกรมจะสร้างอินสแตนท์ขึ้นมาตัวหนึ่งจากคลาส Document ซึ่งอยู่ในไลบรารีของ JDOM ซึ่งคลาส Document นี้จะมีคุณสมบัติเสมือนเป็นเอกสาร XML ที่สร้างไว้แล้วก่อนที่จะเขียน Document นี้ลงไฟล์ให้เป็นเอกสาร XML จริงๆ

4.3.6.1 การเพิ่มเรคคอร์ดของข้อมูล

ผู้ใช้สามารถเพิ่มเรคคอร์ดของข้อมูลที่ต้องการได้ ซึ่งทาง GIS เว็บเซอร์วิสยินยอมให้เพิ่มเรคคอร์ดในส่วนที่เป็น point เท่านั้น โดยการเลือกเลเยอร์ที่ต้องการจะเพิ่มเรคคอร์ดของข้อมูลจากแท็บ Data ซึ่งแท็บนี้จะอยู่ด้านล่างของเฟรมของโปแกรม จากนั้นกดปุ่ม New จะปรากฏหน้าต่างขึ้นมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.6.2 การแก้ไขเรคคอร์ดของข้อมูล

ผู้ใช้สามารถแก้ไขเรคคอร์ดของข้อมูลที่ต้องการได้ซึ่งทาง GIS เว็บเซอร์วิสยินยอมให้เพิ่มเรคคอร์ดในส่วนที่เป็น point เท่านั้น โดยการเลือกเรคคอร์ดของข้อมูลที่ต้องการแก้ไขจากตารางที่อยู่ในแท็บ Data ด้านล่างของเฟรม จากนั้นกด Edit จะปรากฏหน้าต่างขึ้นมา

4.3.6.3 บันทึกข้อมูลที่เพิ่มและแก้ไขลงไฟล์

หลังจากที่จัดการกับข้อมูลเสร็จแล้ว ผู้ใช้ต้องบันทึก Document ลงไฟล์เพื่อจะนำไปอัพโหลดให้กับ GIS เว็บเซอร์วิสจัดการต่อไป ซึ่งการบันทึกนั้นสามารถทำได้โดยการกดปุ่ม Browse ที่อยู่ในแท็บ Upload เพื่อเลือกไฟล์ที่จะบันทึก จากนั้นกดปุ่ม Save เพื่อบันทึกข้อมูลต่างๆที่อยู่ใน Document ลงไฟล์ที่อยู่ในรูปแบบของเอกสาร XML โดยรายละเอียด การเขียนโปรแกรมในส่วนของการเขียนข้อมูลใน Document ลงไฟล์ XML มีดังนี้

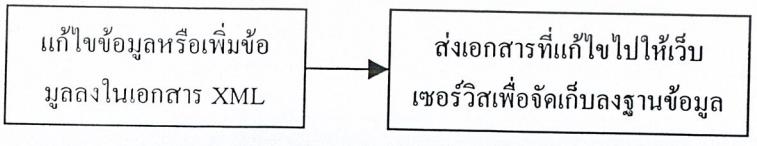
```
XMLOutputter outputter = new XMLOutputter();
FileOutputStream out = new FileOutputStream(file);
outputter.output(document,out);
```

เมื่อบันทึกข้อมูลเสร็จแล้วจากนั้นจะเรียกใช้บริการอัพโหลดไฟล์โดยกดปุ่ม Upload ซึ่งผู้ใช้จะส่งค่าอาร์กิวเมนต์ต่างๆ เข้าไปในฟังก์ชันผ่านทางเว็บเซอร์วิสดังนี้

```
public boolean ws_UploadFile(String fileName)
```

ซึ่งผลลัพธ์ที่ได้คือ การนำไฟล์ในรูปแบบของเอกสาร XML มาให้กับระบบ เพื่อฝากให้กับระบบเว็บเซอร์วิสบริการนำข้อมูลไปเพิ่มและแก้ไขในฐานข้อมูลให้

โดยสามารถสรุปขั้นตอนการเรียกใช้บริการนำข้อมูลจากผู้ใช้ไปเพิ่มและแก้ไขในฐานข้อมูลได้ดังนี้



รูปที่ 4-8 แสดงขั้นตอนการเรียกใช้บริการนำข้อมูลจากผู้ใช้ไปเพิ่มและแก้ไขในฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

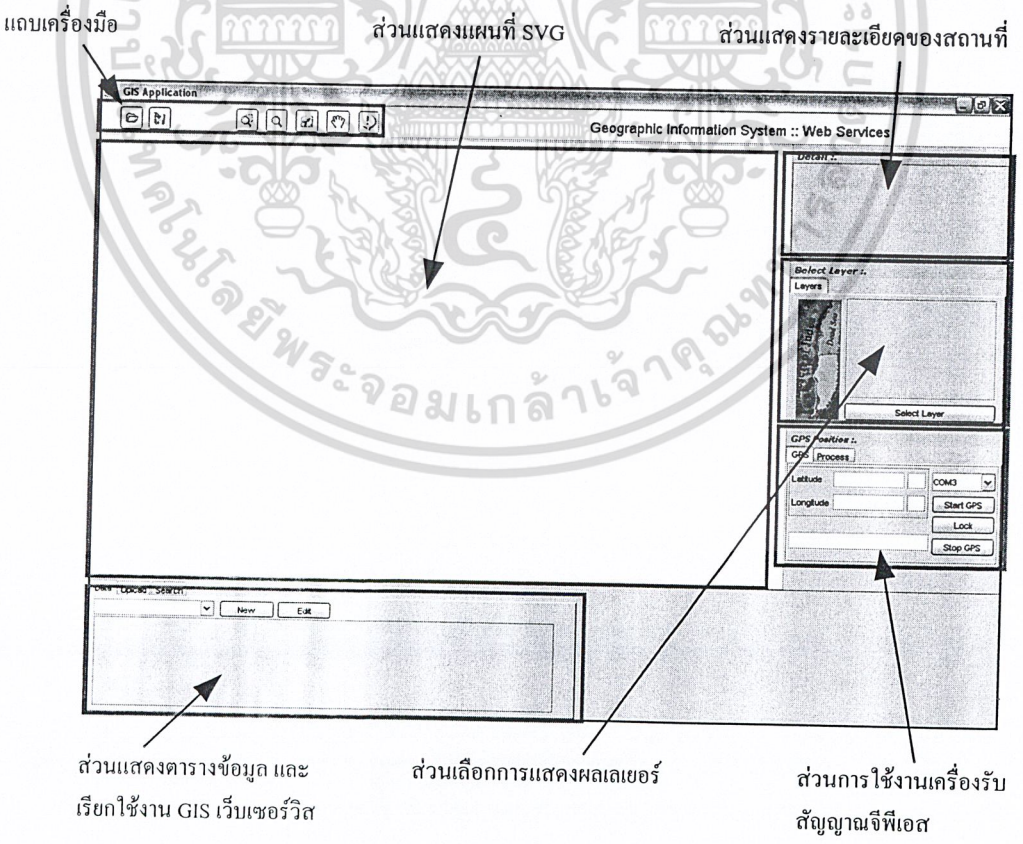
บทที่ 5
ผลการทดลอง

5.1 ผลการทดลองใช้งานในส่วนของ GIS แอพพลิเคชัน

ในการทดลองนี้เราได้ทำการทดลองแสดงผลแผนที่และใช้ฟังก์ชันต่างๆ ที่ช่วยในการแสดงผลแผนที่ โดยมีหัวข้อการทดลองดังนี้

1. การทดลองเปิดแผนที่ในรูปแบบ SVG
2. การทดลองบันทึกแผนที่ในรูปแบบ SVG
3. การทดลองย่อ ขยายแผนที่
4. การทดลองเลื่อนแผนที่
5. การทดลองแสดงผลเฉพาะเลเยอร์
6. การทดลองการคลิกส์เมาส์บนแผนที่เพื่อแสดงรายละเอียด
7. การทดลองหาค่าพิกัดของผู้ใช้โดยผ่านเครื่องรับสัญญาณจีพีเอส

เมื่อเปิดเข้าสู่หน้าแรกในส่วนของ GIS แอพพลิเคชันจะประกอบด้วย 6 ส่วน ดังนี้

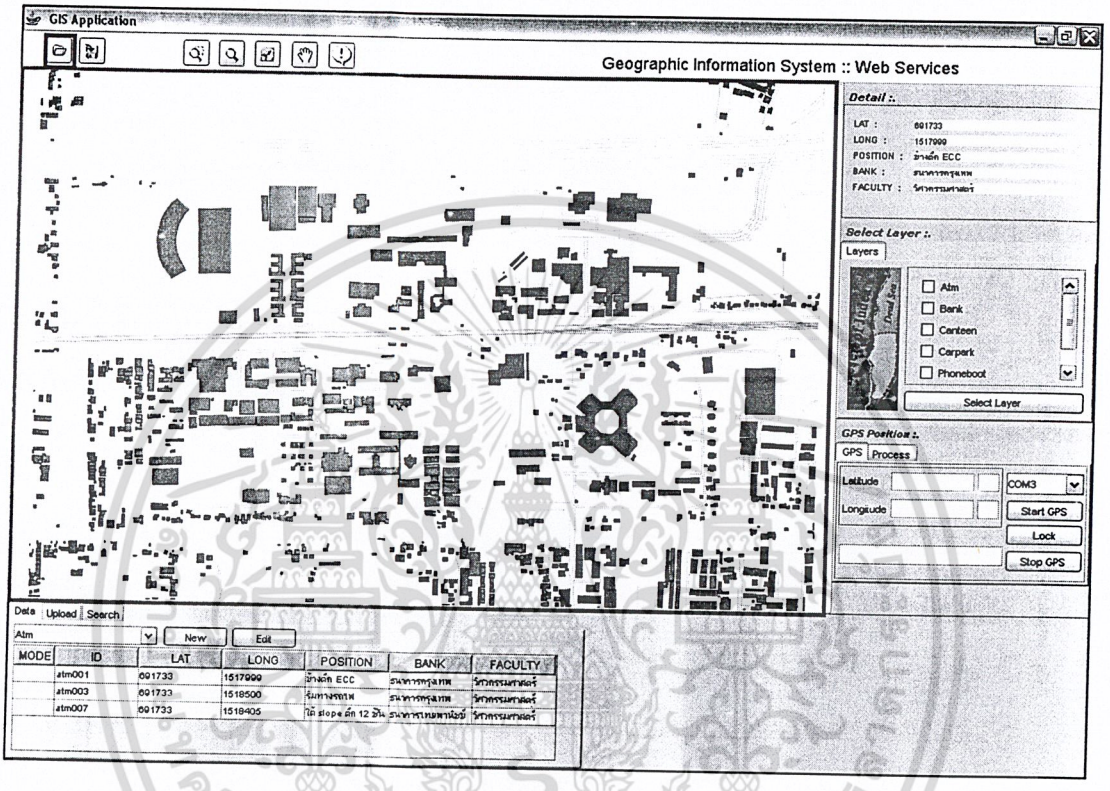


รูปที่ 5-1 แสดงหน้าจอแรกของ GIS แอพพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.1 การทดลองเปิดแผนที่ในรูปแบบ SVG

การทดลองนี้เป็นการทดลองเปิดแผนที่ SVG โดยกดปุ่ม ในส่วนของแถบเครื่องมือด้านบนซ้ายมือของหน้าจอ โดยเลือกเปิดจากไฟล์ที่มีนามสกุลเป็น SVG



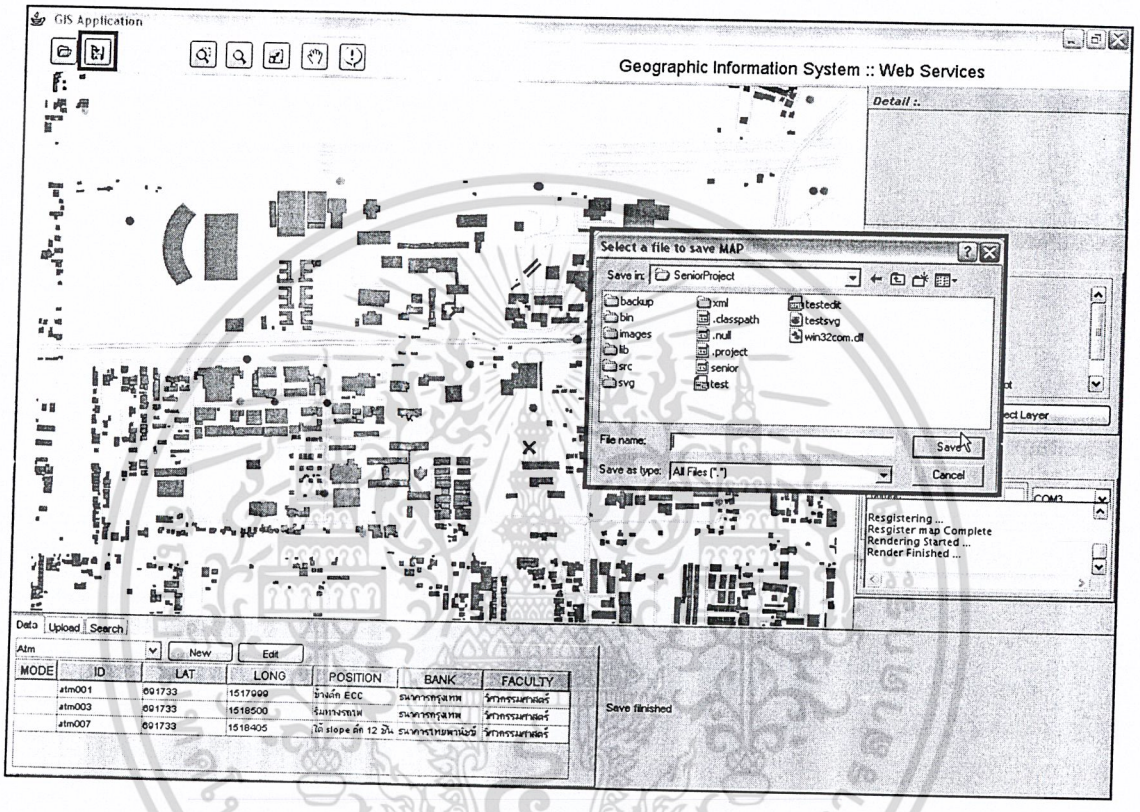
รูปที่ 5-2 แสดงหน้าจอส่วนแสดงผลแผนที่ SVG

จากรูปที่ 5-2 เมื่อทำการแสดงผลแผนที่ SVG แล้ว โปรแกรมจะโหลดข้อมูลจากไฟล์ XML ที่มีชื่อเดียวกับไฟล์แผนที่ SVG จากนั้นจะนำข้อมูลมาแสดงในส่วนตารางแสดงข้อมูลด้านล่างของหน้าจอ และจะทำการโหลดรายชื่อเลเยอร์ทั้งหมดขึ้นมาแสดงในส่วนเลือกการแสดงผลเลเยอร์ ที่อยู่ทางด้านขวามือของหน้าจอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.2 การทดลองบันทึกแผนที่ในรูปแบบ SVG

การทดลองนี้เป็นการทดลองบันทึกแผนที่ SVG โดยกดปุ่ม ในส่วนของแถบเครื่องมือด้านบนซ้ายมือของหน้าจอ โดยเลือกไฟล์ที่ต้องการบันทึกจากนั้นโปรแกรมจะบันทึกแผนที่ในสถานะปัจจุบันลงในไฟล์ .SVG ดังรูปที่ 5.3

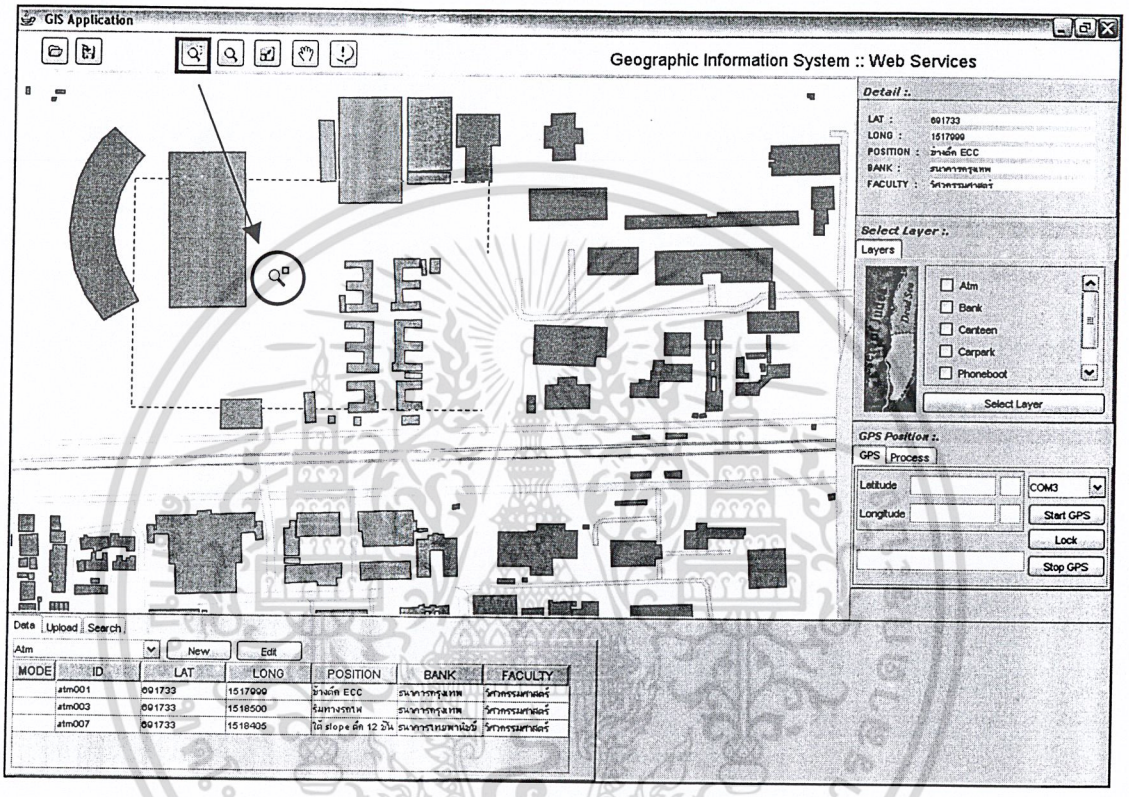


รูปที่ 5-3 แสดงหน้าจอส่วนบันทึกแผนที่ SVG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.3 การทดลองย่อ ขยายแผนที่

การทดลองนี้เป็นการทดลองย่อ ขยายแผนที่ โดยกดปุ่ม เพื่อทำการขยายแผนที่แบบเลือกขอบเขตสี่เหลี่ยม และกดปุ่ม เพื่อทำการย่อและขยายแผนที่แบบเอกซ์เทนด์โหมด (Extended mode) ในส่วนของแถบเครื่องมือด้านบนซ้ายมือของหน้าจอ ดังรูปที่ 5.4

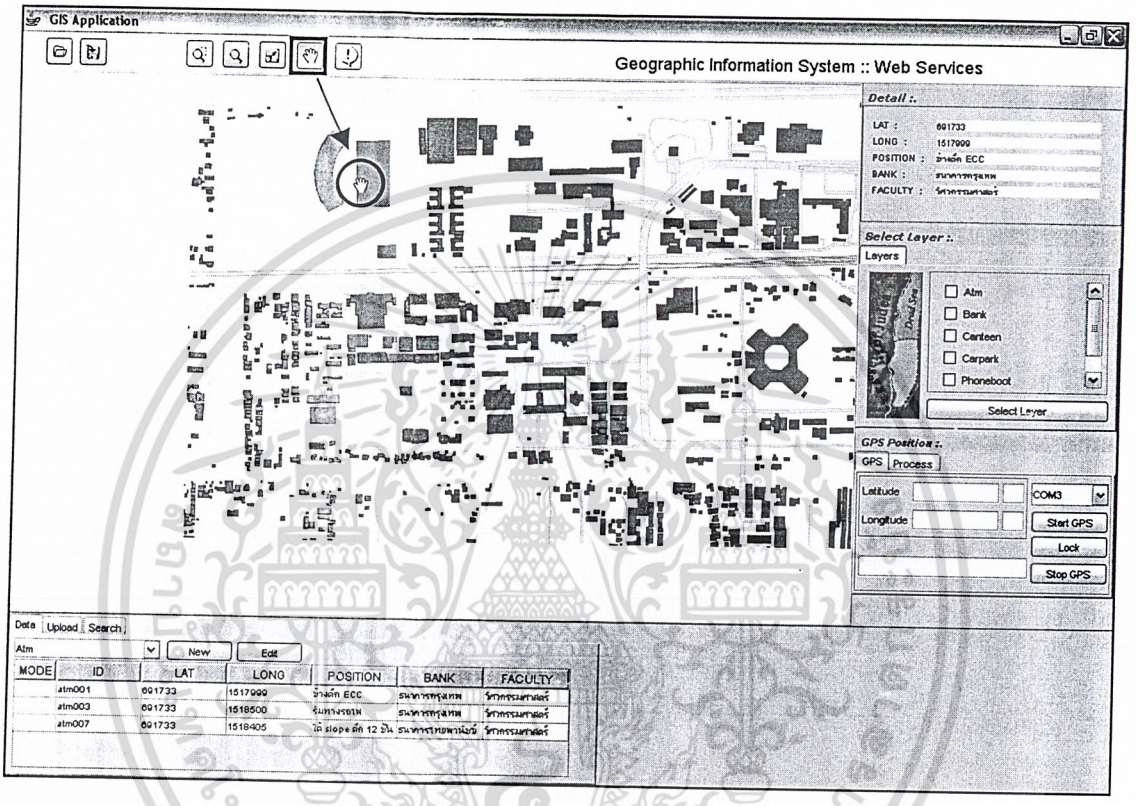


รูปที่ 5-4 แสดงหน้าจอส่วนย่อและขยายแผนที่ SVG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.4 การทดลองเลื่อนแผนที่

การทดลองนี้เป็นการทดลองเลื่อน (pan) แผนที่ โดยกดปุ่ม เพื่อทำการแผนที่ ในส่วนของแถบเครื่องมือด้านบนซ้ายมือของหน้าจอ ดังรูปที่ 5.5

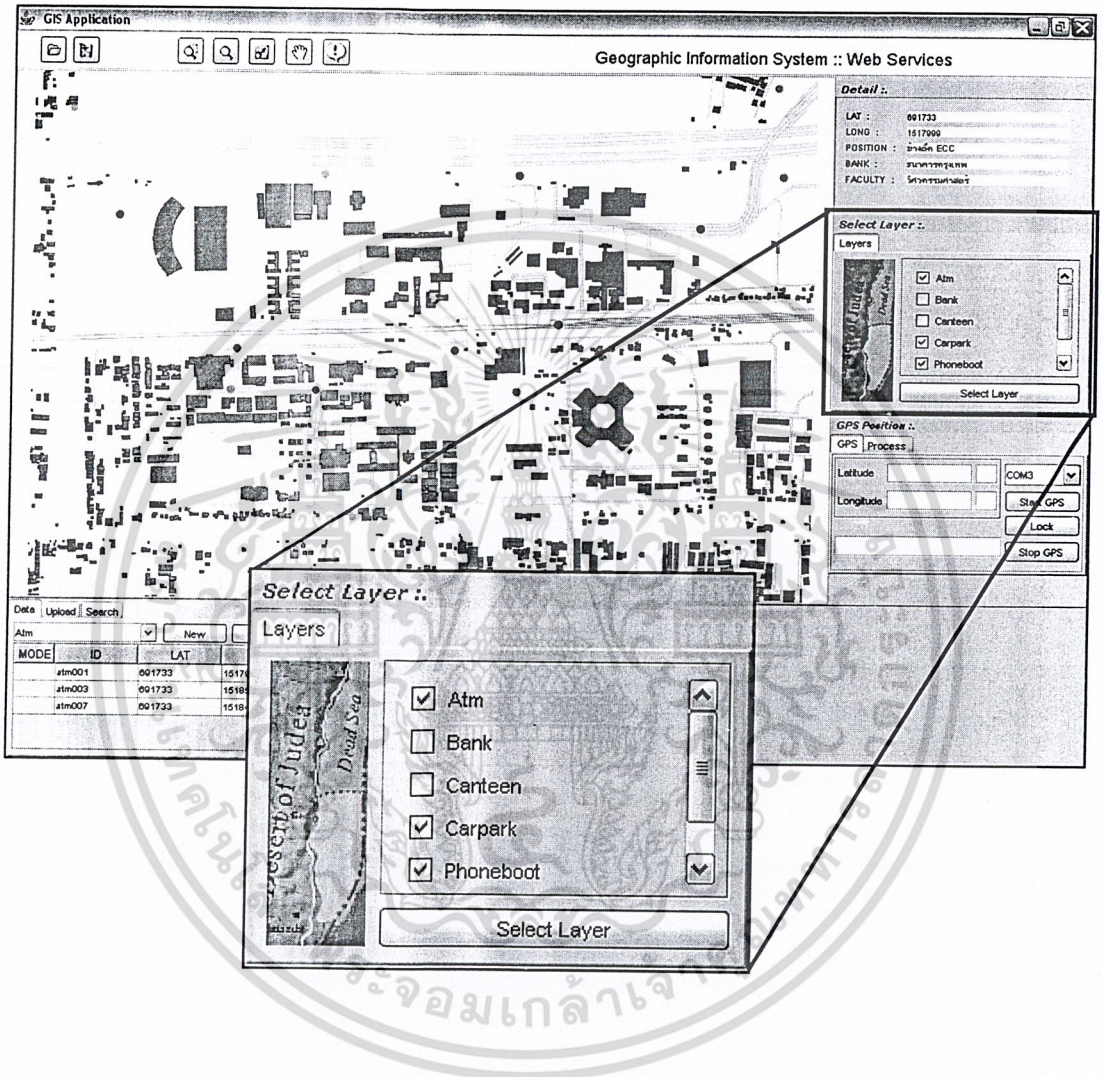


รูปที่ 5-5 แสดงหน้าจอส่วนเลื่อนแผนที่ SVG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.5 การทดลองแสดงผลเฉพาะเลเยอร์

การทดลองนี้เป็นการทดลองเลือกแสดงผลเลเยอร์ที่ต้องการ โดยเลือกจากเช็คบ๊อคซ์ที่อยู่ในส่วนของการแสดงผลเลเยอร์ ด้านขวาของหน้าจอ ดังรูปที่ 5.6



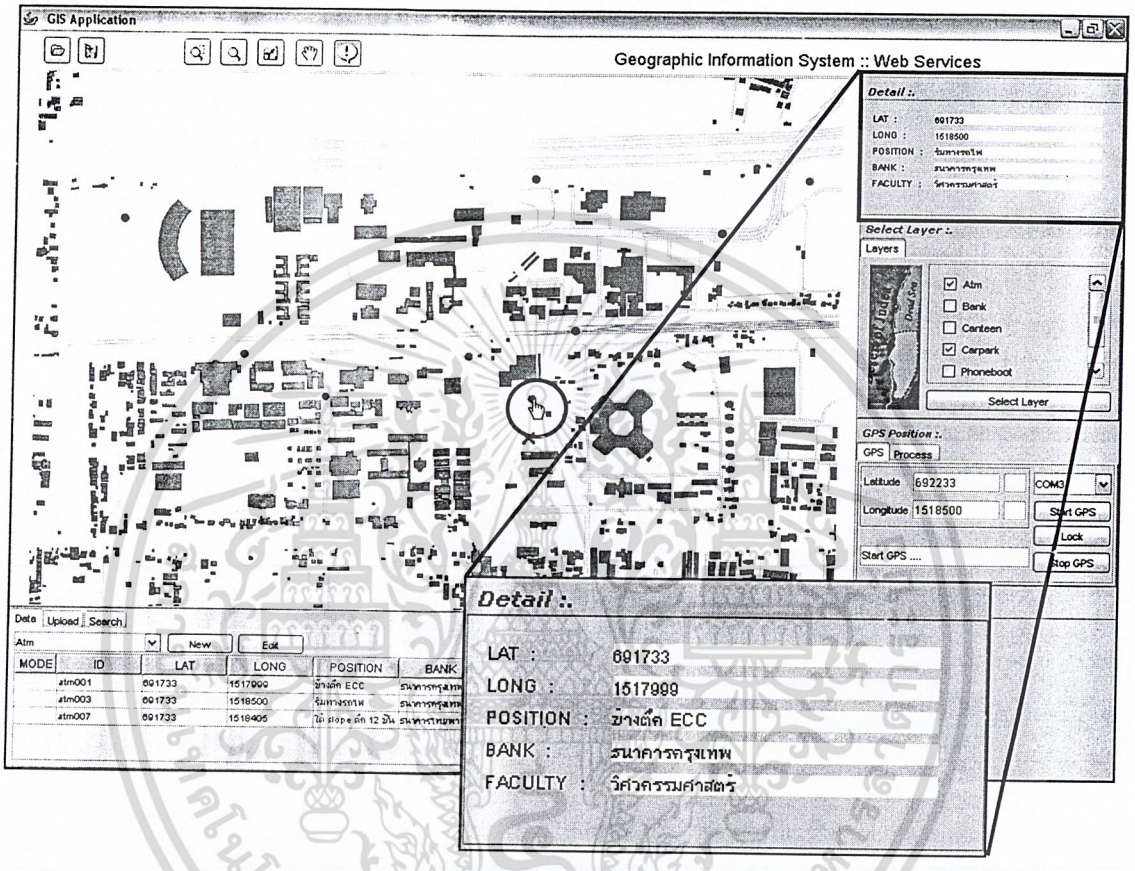
รูปที่ 5-6 แสดงหน้าจอส่วนแสดงผลแผนที่ SVGในแต่ละเลเยอร์

โดยเลเยอร์ที่เลือกนั้นจะปรากฏเป็นจุดวงกลมสีต่างๆบนแผนที่ SVG และสามารถคลิ๊กที่แต่ละจุดเพื่อแสดงรายละเอียดของนั้นๆในส่วนการแสดงรายละเอียดของสถานที่ ด้านบนขวามือของหน้าจอ ซึ่งจะกล่าวในการทดลองต่อไป โดยในตัวอย่างนี้จะเป็นการเลือกแสดงผลเลเยอร์ เอทีเอ็ม ธนาคาร และ ตู้โทรศัพท์สาธารณะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.6 การทดลองการคลิ๊กเมาส์บนแผนที่เพื่อแสดงรายละเอียด

การทดลองนี้เป็นการทดลองเลือกแสดงผลเลเยอร์ที่ต้องการ โดยเลือกจากเช็คบ๊อคซ์ที่อยู่ในส่วนของการแสดงผลเลเยอร์ ด้านขวาของหน้าจอ ดังรูปที่ 5.7

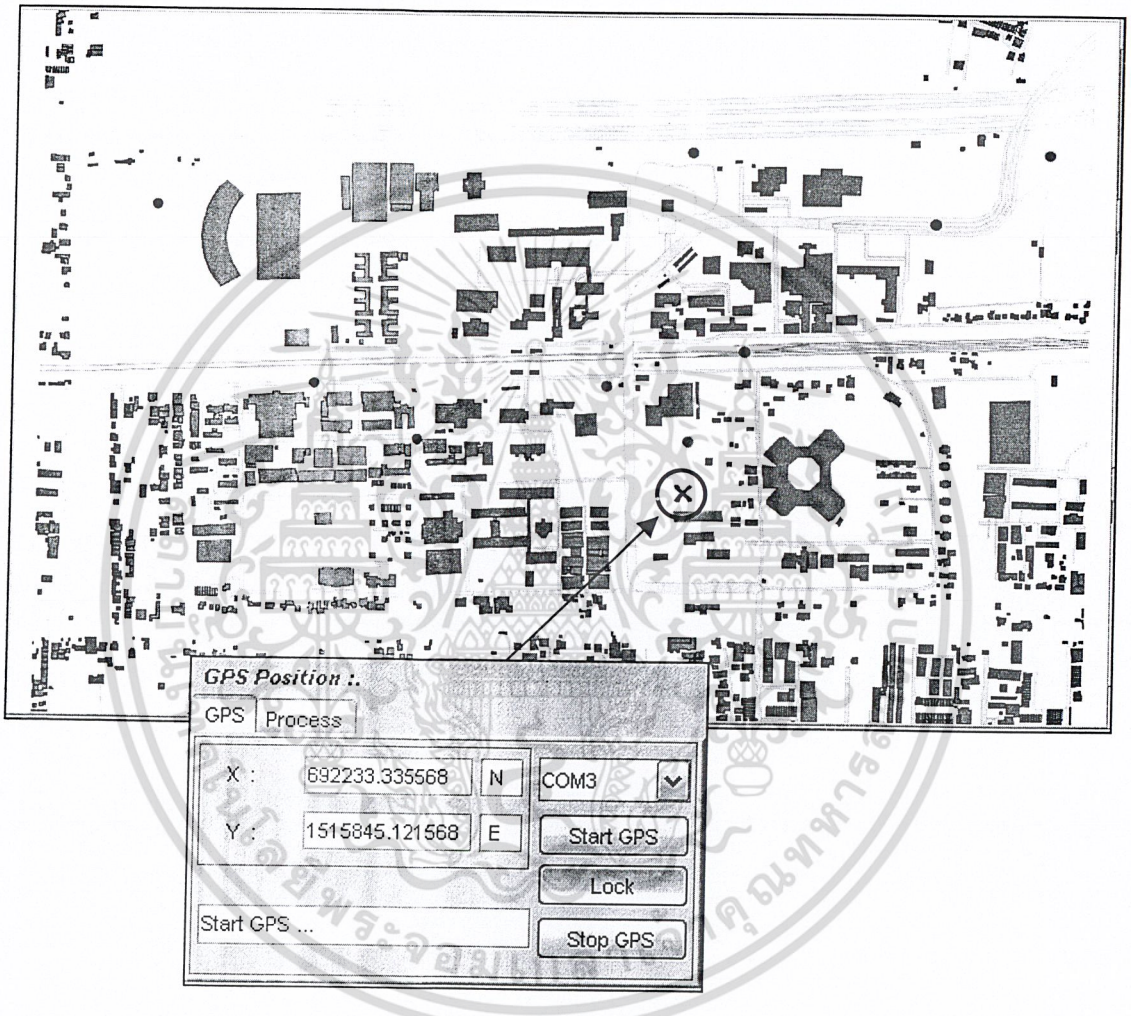


รูปที่ 5-7 แสดงหน้าจอส่วนแสงรายละเอียดเมื่อมีการคลิ๊กเม้าส์บนแผนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.7 การทดลองหาค่าพิกัดของผู้ใช้โดยผ่านเครื่องรับสัญญาณจีพีเอส

การทดลองนี้เป็นการทดลองหาตำแหน่งของผู้ใช้ผ่านทางเครื่องรับสัญญาณจีพีเอส โดยเลือกพอร์ตของการรับสัญญาณ ในที่นี้จะเลือกที่พอร์ต COM 3 และคลิ๊ก Lock ที่อยู่ในส่วนการใช้งานเครื่องรับสัญญาณจีพีเอสทางด้านขวามือของหน้าจอ ดังรูปที่ 5.8



รูปที่ 5-8 แสดงหน้าจอส่วนการหาค่าพิกัดของผู้ใช้โดยผ่านเครื่องรับสัญญาณจีพีเอส

จากนั้นโปรแกรมจะรับค่าพิกัดตำแหน่งที่ผู้ใช้อ้างอิงอยู่นำมาแสดงผลเป็นค่าพิกัดในมาตรฐาน UTM บนส่วนแสดงผลพิกัด X และ Y นอกจากนั้นยังนำพิกัดตำแหน่งของผู้ใช้ไปวาดเป็นรูปกากบาทบนแผนที่เพื่อแสดงให้เห็นถึงตำแหน่งที่ผู้ใช้อ้างอิงอยู่

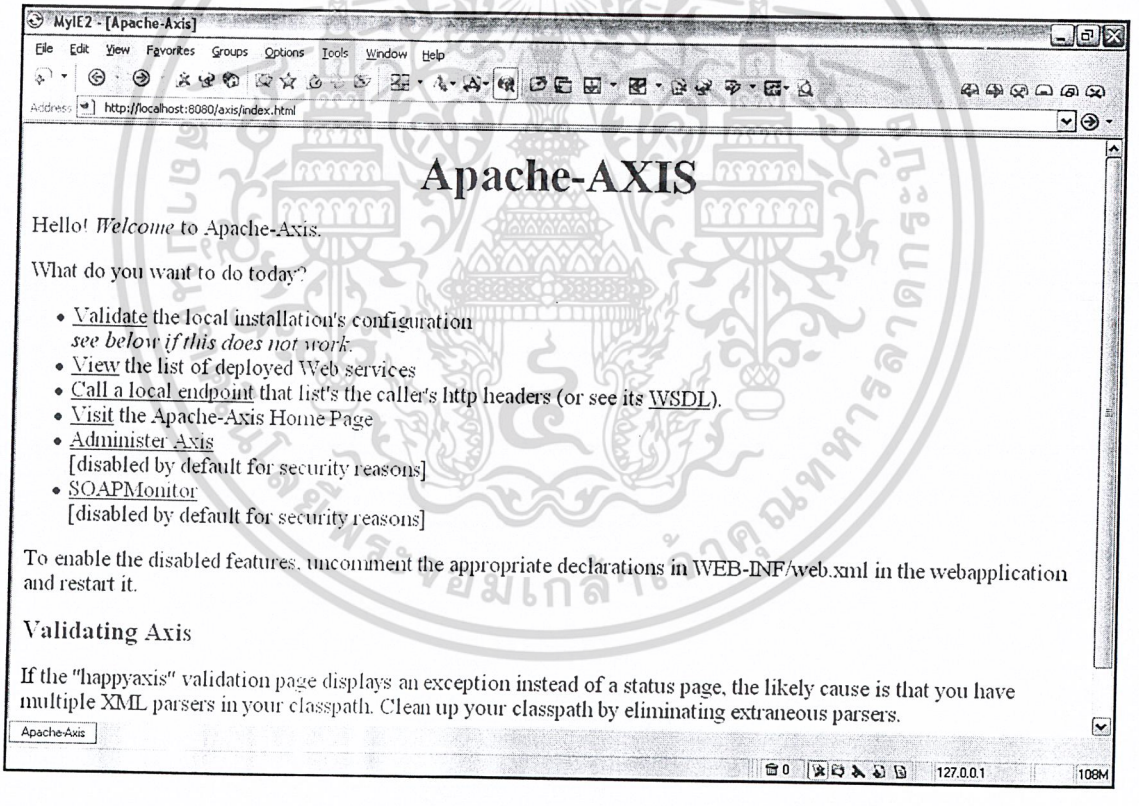
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 ผลการทดลองใช้งานในส่วนของ GIS เว็บเซอร์วิส

ในส่วนการทดลองนี้เป็นการทดลองเรียกใช้บริการต่างๆที่เปิดให้บริการบน GIS เว็บเซอร์วิส ผ่านทาง GIS แอพพลิเคชัน โดยจะทดลองเรียกใช้บริการทั้งหมด 6 บริการ ดังนี้

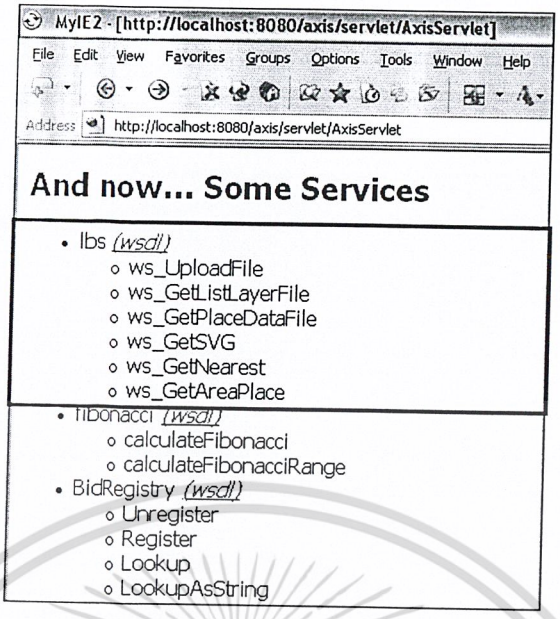
1. ทดลองเรียกใช้บริการข้อมูลเลเยอร์ที่มีในฐานข้อมูล
2. ทดลองเรียกใช้บริการขอรายละเอียดและตำแหน่งที่ตั้งของสถานที่
3. ทดลองเรียกใช้บริการแผนที่ในรูปแบบ SVG
4. ทดลองเรียกใช้บริการค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการ
5. ทดลองเรียกใช้ บริการค้นหาสถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุด
6. ทดลองเรียกใช้บริการนำข้อมูลจากผู้ใช้ไปเพิ่มและแก้ไขในฐานข้อมูล

โดยเริ่มทดลองจากการดูบริการต่างๆบน GIS เว็บเซอร์วิส ว่ามีบริการใดเปิดให้บริการบ้าง โดยการเปิดเว็บเบราเซอร์เพื่อดูบริการผ่านทาง http://161.246.6.209:8080/axis/index.html จากนั้นเมื่อโหลดหน้าเว็บขึ้นมาได้แล้วเลือกที่ View เพื่อดูบริการต่างๆ ดังรูปที่ 5-9



รูปที่ 5-9 แสดงหน้าเว็บเพื่อเข้าดูบริการต่างๆที่เปิดบน GIS เว็บเวอร์วิส(1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5-10 แสดงหน้าเว็บเพื่อเข้าดูบริการต่างๆที่เปิดบน GIS เว็บเซอร์วิส(2)

จากรูปที่ 5 -10 จะแสดงบริการทั้งหมดที่เปิดให้บริการบน GIS เว็บเซอร์วิส จากนั้นเลือกที่ WSDL เพื่อดูรายละเอียดการให้บริการผ่านทาง WSDL ดังรูปที่ 5-11

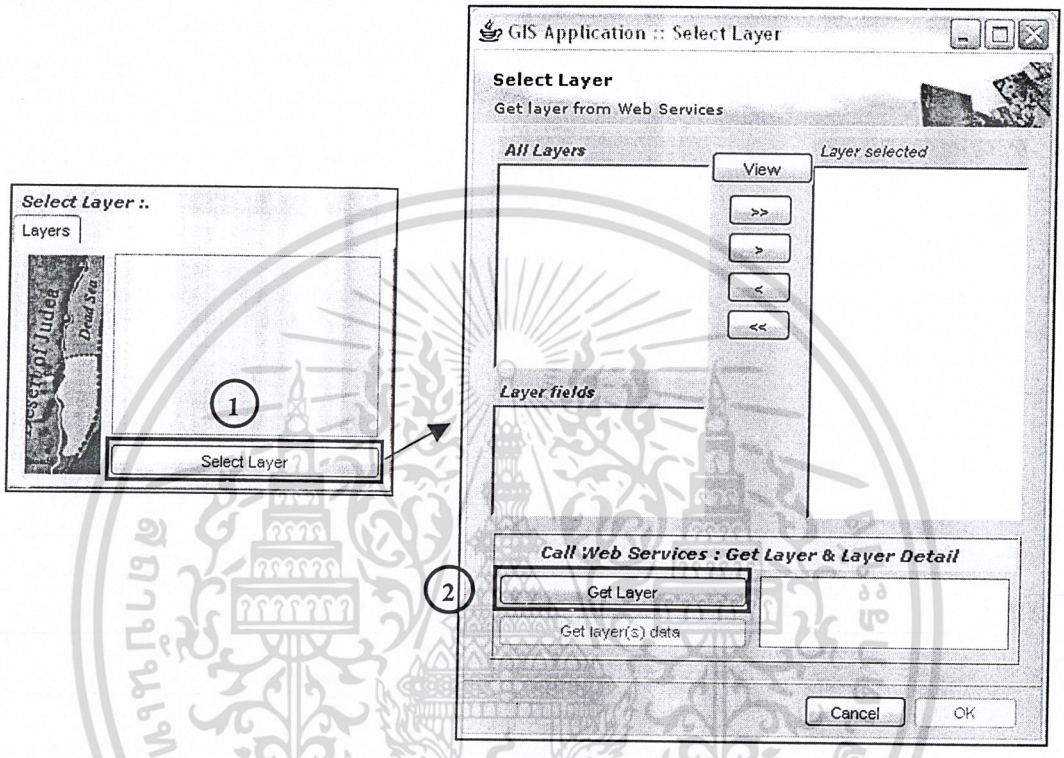
```
<?xml version="1.0" encoding="UTF-8" ?>
- <wsdl:definitions targetNamespace="urn:lbs"
    xmlns="http://schemas.xmlsoap.org/wsdl/"
    xmlns:apachesoap="http://xml.apache.org/xml-soap" xmlns:impl="urn:lbs"
    xmlns:intf="urn:lbs" xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
    xmlns:tns1="http://www.soapinterop.org/Bid"
    xmlns:tns2="http://www.soapinterop.org/Registry"
    xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
    xmlns:wsdlsoap="http://schemas.xmlsoap.org/wsdl/soap/"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  + <wsdl:types>
  - <wsdl:message name="ws_GetNearestResponse">
      <wsdl:part name="ws_GetNearestReturn" type="impl:ArrayOf_xsd_string" />
    </wsdl:message>
  - <wsdl:message name="ws_GetSVGResponse">
      <wsdl:part name="ws_GetSVGReturn" type="xsd:base64Binary" />
    </wsdl:message>
  - <wsdl:message name="ws_UploadFileResponse">
      <wsdl:part name="ws_UploadFileReturn" type="xsd:boolean" />
    </wsdl:message>
  - <wsdl:message name="ws_GetPlaceDataFileResponse">
      <wsdl:part name="ws_GetPlaceDataFileReturn" type="xsd:base64Binary" />
```

รูปที่ 5-11 แสดงหน้าเว็บที่แสดง WSDLเพื่อเข้าดูบริการต่างๆที่เปิดบน GIS เว็บเซอร์วิส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.1. ทดลองเรียกใช้บริการข้อมูลเลเยอร์ที่มีในฐานข้อมูล

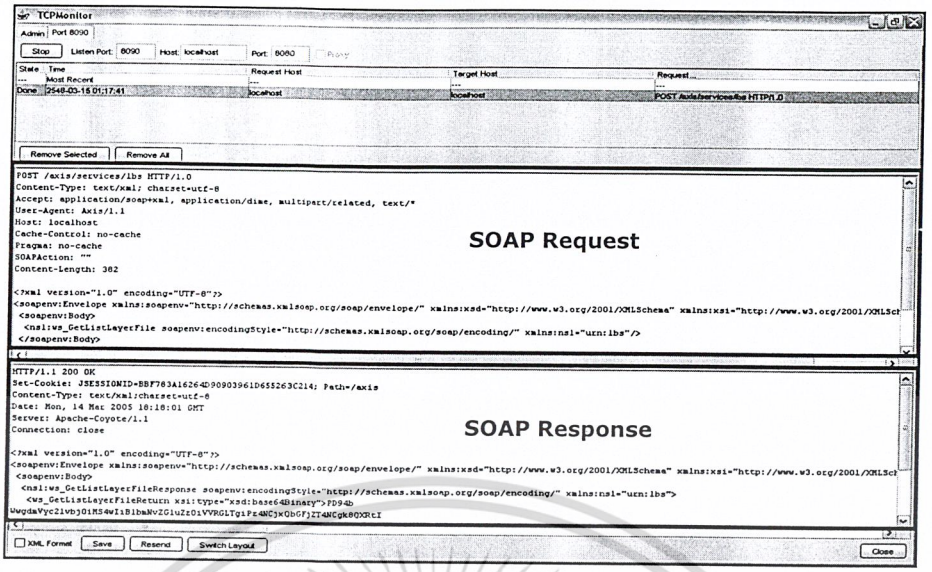
การทดลองเรียกใช้บริการข้อมูลเลเยอร์เริ่มจากกดปุ่ม Select Layer ที่อยู่ใน ส่วนเลือกการแสดงผลเลเยอร์ จากนั้นโปรแกรมจะเปิดเฟรมในการเรียกใช้บริการข้อมูลเลเยอร์ขึ้นมา และทำการเรียกใช้บริการจากเว็บเซอร์วิสโดยกดปุ่ม Get Layer ดังรูปที่ 5-12



รูปที่ 5-12 แสดงหน้าจอในส่วนของการใช้บริการข้อมูลเลเยอร์ในฐานข้อมูล

เราสามารถดูการเรียกใช้ และรับผลลัพธ์จากการให้บริการได้ซึ่งจะส่งข้อมูลผ่านทาง SOAP แมสเสจ ได้จาก TCPMonitor ดังรูปที่ 5-13

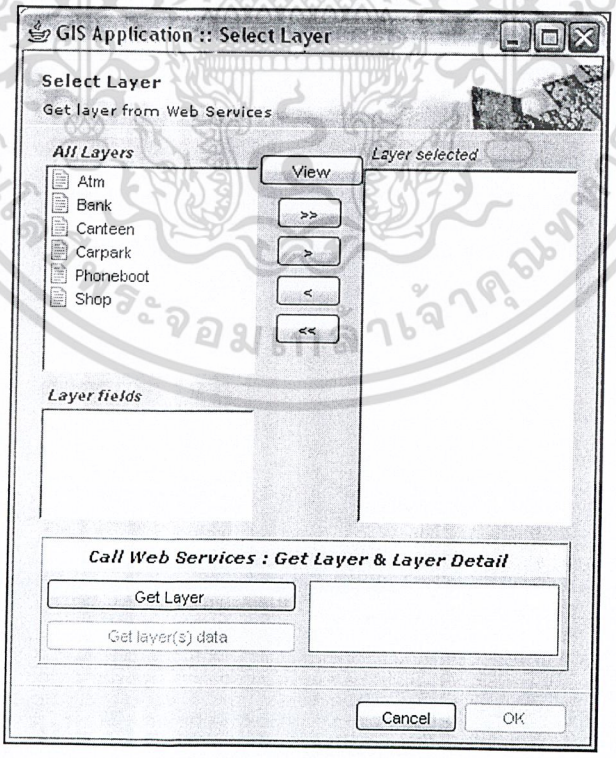
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5-13 แสดงหน้าจอ TCP Monitor

จากรูปที่ 5-13 จะแสดงถึงส่วนของ SOAP รีเควส ในการเรียกใช้บริการที่ชื่อ ws_GetListLayerFile และตอบกลับมาทาง SOAP เรสพอนด์ ซึ่งผลลัพธ์ที่ได้จะเป็นสายของสตริงซึ่งชนิดเป็นไบต์อาร์เรย์

เมื่อโปรแกรมรับผลลัพธ์กลับมาแล้ว จะนำมาแสดงผลเป็นรายชื่อเลเยอร์ทั้งหมด ดังรูปที่ 5-14

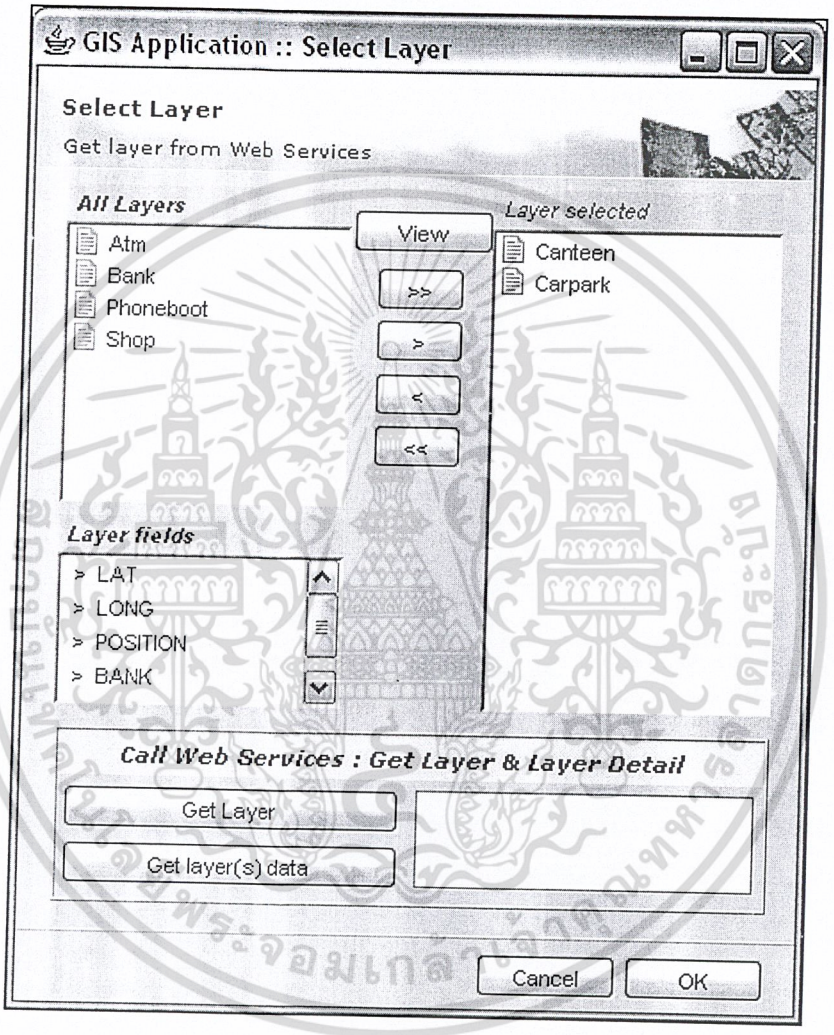


รูปที่ 5-14 แสดงหน้าจอรายชื่อเลเยอร์ที่ได้รับการเลือก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.2. ทดลองเรียกใช้บริการขอรายละเอียดและตำแหน่งที่ตั้งของสถานที่

การทดลองเรียกใช้บริการขอรายละเอียดและตำแหน่งที่ตั้งของสถานที่โดยเลือกเลเยอร์ที่ต้องการขอรายละเอียดก่อน และกดปุ่ม Get Layer Data จากนั้นโปรแกรมจะเรียกใช้บริการชื่อ ws_GetPlaceDataFile และ ws_GetSVG พร้อมกัน แต่บริการ ws_GetSVG จะกล่าวถึงในหัวข้อถัดไป ซึ่งสิ่งที่ได้รับจากการเรียกใช้บริการนี้คือข้อมูลของเลเยอร์ต่างๆที่ผู้ใช้ต้องการดังรูปที่ 5-15



รูปที่ 5-15 แสดงหน้าจอข้อมูลของเลเยอร์ต่างๆ

5.2.3. ทดลองเรียกใช้บริการแผนที่ในรูปแบบ SVG

จากการทดลองที่แล้วเมื่อเรียกใช้บริการชื่อ ws_GetSVG ซึ่งเป็นบริการร้องขอแผนที่ SVG จากเว็บเซอร์วิส ซึ่งสิ่งที่ได้รับกลับมาคือแผนที่ในรูปแบบ SVG ซึ่งมีจำนวนเลเยอร์ตามที่ผู้ใช้ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

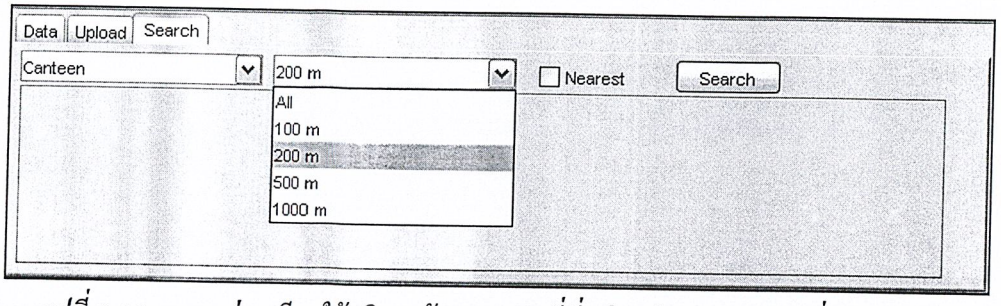
5.2.4. ทดลองเรียกใช้บริการค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการ

การทดลองเรียกใช้บริการค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการจะอยู่ในแท็บ Search จากนั้นเลือกเลเยอร์ และบริเวณรัศมีที่ต้องการค้นหาดังรูปที่ 5-16 โดยมีรูปแบบของ SOAP รีเควส ดังนี้

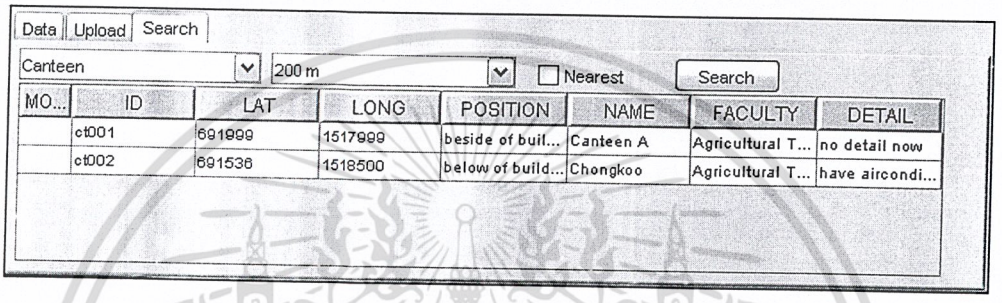
```
POST /axis/services/lbs HTTP/1.0
Content-Type: text/xml; charset=utf-8
Accept: application/soap+xml, application/dime, multipart/related, text/*
User-Agent: Axis/1.1
Host: localhost
Cache-Control: no-cache
Pragma: no-cache
SOAPAction: ""
Content-Length: 575

<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv=http://schemas.xmlsoap.org/soap/envelope/
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <soapenv:Body>
        <ns1:ws_GetAreaPlace
        soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
        xmlns:ns1="urn:lbs">
            <in0 xsi:type="xsd:string">Atm</in0>
            <in1 xsi:type="xsd:int">200</in1>
            <in2 xsi:type="xsd:float">692248.0</in2>
            <in3 xsi:type="xsd:float">1518342.0</in3>
        </ns1:ws_GetAreaPlace>
    </soapenv:Body>
</soapenv:Envelope>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยจะนำผลของการค้นหามาแสดงผลในรูปของตารางข้อมูลดังรูปที่ 5-17



รูปที่ 5-16 แสดงส่วนเรียกใช้บริการค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการ



รูปที่ 5-17 แสดงข้อมูลที่ได้จากการเรียกใช้บริการค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการ

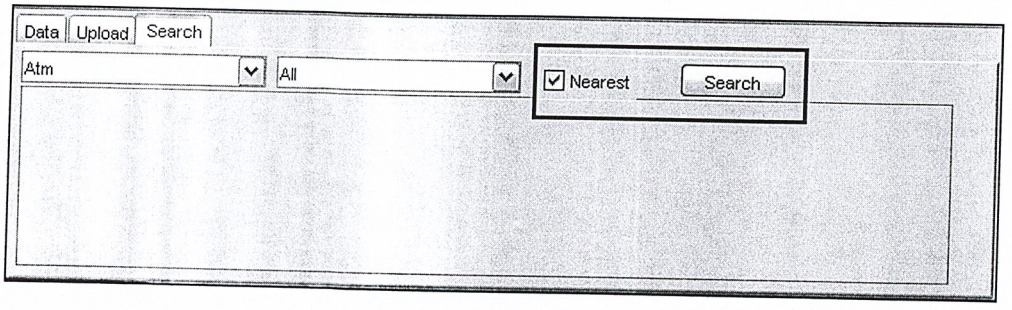
5.2.5. ทดลองเรียกใช้บริการค้นหาสถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุด

การทดลองเรียกใช้บริการค้นหาสถานที่ที่อยู่ใกล้กับผู้ใช้มากที่สุดจะอยู่ในแท็บ Search จากนั้นเลือก
เลเยอร์ และเลือกเช็คบ๊อคซ์ Nearest ดังรูปที่ 5-18 ซึ่งรูปแบบของ SOAP Request มีดังนี้

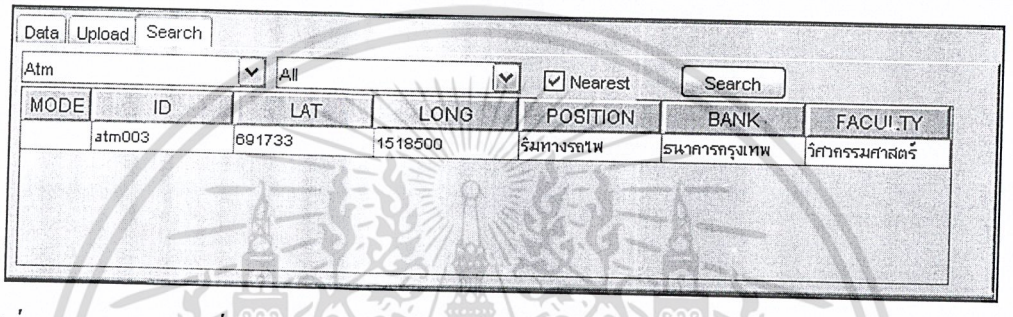
```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv=http://schemas.xmlsoap.org/soap/envelope/
                  xmlns:xsd=http://www.w3.org/2001/XMLSchema
                  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        <soapenv:Body>
                <ns1:ws_GetNearest
        soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
        xmlns:ns1="urn:lbs">
                        <in0 xsi:type="xsd:string">Atm</in0>
                        <in1 xsi:type="xsd:float">692248.0</in1>
                        <in2 xsi:type="xsd:float">1518342.0</in2>
                </ns1:ws_GetNearest>
        </soapenv:Body>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยจะนำผลของการค้นหามาแสดงผลในรูปของตารางข้อมูล ดังรูปที่ 5-19



รูปที่ 5-18 แสดงส่วนเรียกใช้บริการค้นหาสถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุด



รูปที่ 5-19 แสดงข้อมูลที่ได้จากการเรียกใช้บริการค้นหาสถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุด

5.2.6. ทดลองเรียกใช้บริการนำข้อมูลจากผู้ใช้ไปเพิ่มและแก้ไขในฐานข้อมูล

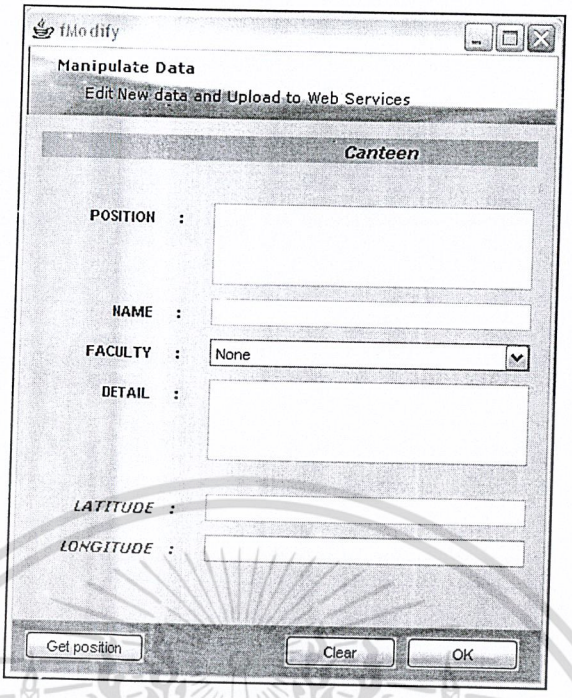
การทดลองนี้จะแบ่งการทดลองออกเป็น 3 ส่วน ดังนี้

1. การทดลองเพิ่มเรคคอร์ดของข้อมูล
2. การทดลองแก้ไขเรคคอร์ดของข้อมูล
3. การทดลองบันทึกข้อมูลที่เพิ่มและแก้ไขลงไฟล์

5.2.6.1 การทดลองเพิ่มเรคคอร์ดของข้อมูล

ทดลอง โดยการเลือกเลเยอร์ที่อยู่ในแท็บ Data จากนั้นกด New เพื่อเพิ่มเรคคอร์ดของข้อมูลดังรูปที่ 5-20

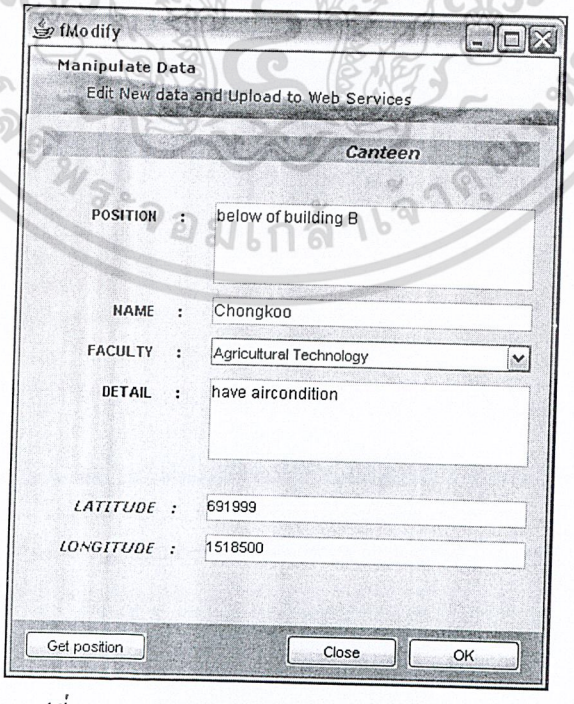
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5-20 แสดงหน้าจอเพิ่มเรคคอร์ดของข้อมูล

5.2.6.2 การทดลองแก้ไขเรคคอร์ดของข้อมูล

ทดลองโดยการเลือกเลเยอร์ที่อยู่ในแท็บ Data และเลือกเรคคอร์ดที่ต้องการแก้ไข จากนั้นกด Edit เพื่อแก้ไขข้อมูลจากเรคคอร์ดนั้นใหม่ ดังรูปที่ 5-21

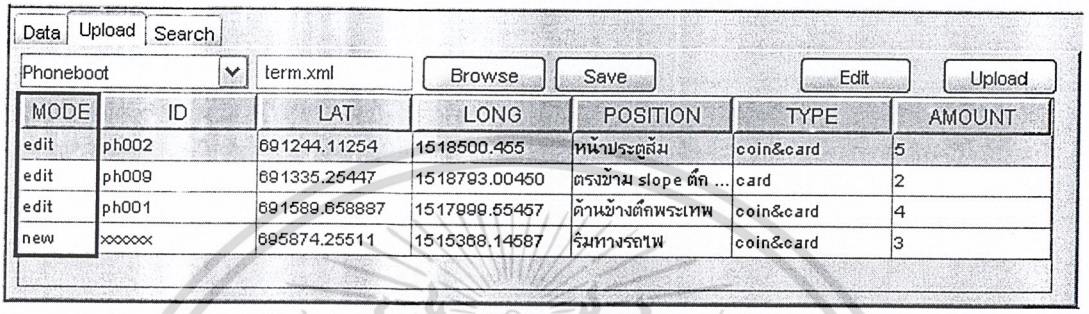


รูปที่ 5-21 แสดงหน้าจอแก้ไขเรคคอร์ดของข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.6.3 การทดลองเรียกใช้บริการบันทึกข้อมูลที่เพิ่มและแก้ไขลงไฟล์

การทดลองเรียกใช้บริการบันทึกข้อมูลที่เพิ่มและแก้ไขลงไฟล์ซึ่งข้อมูลที่ถูกเพิ่มและแก้ไขแล้วจะนำไปแสดงเป็นตารางข้อมูลที่อยู่ในแท็บ Upload โดยที่จะมีฟิลด์ MODE บอกว่าข้อมูลแต่ละแถวในตารางนั้นเป็นการเพิ่มเรคคอร์ดของข้อมูลหรือเป็นการแก้ไขเรคคอร์ดของข้อมูล ดังรูปที่ 5-22 และการกดปุ่ม Upload เพื่อทำการบันทึกข้อมูลลงไฟล์



รูปที่ 5-22 แสดงหน้าจอบันทึกข้อมูลที่เพิ่มและแก้ไขลงไฟล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
บทวิจารณ์และสรุป

6.1บทวิจารณ์และสรุป

เว็บเซอร์วิสเชิงพื้นที่สำหรับระบบสารสนเทศเชิงภูมิศาสตร์ที่พัฒนาขึ้นมานั้นสามารถให้บริการแสดงผลแผนที่ในรูปแบบ SVG , ข้อมูลรายละเอียดและตำแหน่งที่ตั้งของสถานที่,ค้นหาสถานที่ที่อยู่ภายในบริเวณรัศมีที่ผู้ใช้ต้องการหรือสถานที่ที่อยู่ใกล้กับจุดอ้างอิงของผู้ใช้มากที่สุดได้และบริการนำข้อมูลจากผู้ใช้ไปเพิ่มและแก้ไขในฐานข้อมูลซึ่งผู้ใช้สามารถเลือกแสดงเลเยอร์ที่ต้องการได้ในการแสดงผลแผนที่และสามารถจัดการเกี่ยวกับการแสดงผลแผนที่ได้ เช่น สามารถย่อหรือขยายแผนที่และเลื่อนขอบเขตการแสดงแผนที่ได้โดยแอปพลิเคชั่นที่ร้องขอบริการไม่จำเป็นต้องอยู่บน platform เดียวกันหรือพัฒนาจากภาษาเดียวกันทำให้ผู้ใช้บริการมีความสะดวกมากขึ้น อีกทั้งเทคโนโลยีของ เว็บเซอร์วิสมีมาตรฐานในการอธิบายรายละเอียดของแต่ละบริการทำให้ผู้เรียกใช้บริการสามารถเข้าใจลักษณะของบริการได้โดยง่าย โดยผู้ใช้อาจเรียกใช้เพียงบริการใดบริการหนึ่งก็ได้ เช่น ต้องการเพียงข้อมูลสถานที่ไม่ต้องการแผนที่ก็สามารถทำได้ หรือต้องการเพียงให้เว็บเซอร์วิสช่วยค้นหาสถานที่ก็ได้ โดยข้อมูลที่สถานที่หรือข้อมูลที่ผู้ใช้แก้ไขแล้วส่งมายังเว็บเซอร์วิสนั้นผู้จัดทำได้ออกแบบให้มีการส่งข้อมูลในรูปแบบเอกสาร XML เพื่อความสะดวกในการจัดการกับข้อมูลดังกล่าว ในส่วนของการแสดงผลแผนที่ได้ออกแบบให้มีการนำเสนอแผนที่ในรูปแบบ SVG ทำให้มีความสามารถในการย่อขยายแผนที่โดยที่คุณภาพของการแสดงผลแผนที่นั้นยังคงเดิม นอกจากนั้นเอกสาร SVG ยังมีรูปแบบพื้นฐานเป็น XML ด้วยทำให้ช่วยในการจัดการกับแผนที่ได้ง่ายขึ้นอีกด้วย

6.2 ปัญหาที่เกิดในการพัฒนาและแนวทางการแก้ไข

1. เทคโนโลยีเว็บเซอร์วิสเป็นเทคโนโลยีที่ใหม่และยังอยู่ในช่วงพัฒนา ทำให้เกิดปัญหาในการค้นหาเอกสารอ้างอิงและตำรา

2. การพัฒนาเว็บเซอร์วิสโดยภาษาจาวานั้นมีเครื่องมือในการพัฒนาเว็บเซอร์วิสมากมาย และเมื่อพัฒนาด้วยเครื่องมือใดแล้วการที่จะเปลี่ยนเครื่องมือในการพัฒนานั้นทำได้ยาก จึงต้องศึกษาข้อดีข้อเสียของเครื่องมือแต่ละอย่างให้ดีเพื่อที่จะสามารถเลือกใช้ในการพัฒนาได้อย่างเหมาะสม

3. การพัฒนาแอปพลิเคชั่นเกี่ยวกับระบบสารสนเทศภูมิศาสตร์นั้นมีรายละเอียดมากมายเช่น ลักษณะการเก็บข้อมูล, ระบบพิกัด, มาตรฐานต่างๆ ทำให้ต้องใช้ระยะเวลาในการศึกษามากพอสมควร

4. ข้อมูลในแต่ละเลเยอร์ต้องใช้เวลาในการเก็บข้อมูลมากเนื่องจากเครื่องจับค่าพิกัดใช้เวลาในการจับค่าพิกัดแต่ละจุดนาน

5. การส่งแผนที่และข้อมูลต่างๆ ผ่านทางเว็บเซอร์วิสนั้นจะจัดส่งในรูปแบบของไฟล์ซึ่งไม่สะดวกจึงใช้การส่งเป็นอาเรย์ชนิดไบท์แทนแล้วจึงนำไปเขียนลงไฟล์อีกทีหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. การใช้ภาษาไทยในฐานข้อมูล (PostgreSQL) สามารถแสดงผลได้ไม่ดี จึงแก้ไขโดยใช้การเข้ารหัสอักษร(encode)

6.3 ข้อจำกัดของระบบ

1. การใช้ Batik แสดงผลแผนที่ SVGไม่สามารถทำงานในส่วนแท็ก animated ของเอกสาร SVG ได้ ทำให้ไม่สามารถทำแอนนิเมชั่นบน SVG ได้

2. การพัฒนาเว็บเซอร์วิสด้วย Apache Axis ยังไม่สันบสนุนการคืนค่าจากฟังก์ชันเป็นชนิดไฟล์โดยตรงได้ ทำให้การส่งไฟล์จากเว็บเซอร์วิสมายังไคลเอนท์ต้องทำการแปลงไฟล์ให้อยู่ในรูปแบบของไบต์อาร์เรย์เสียก่อน ซึ่งทำให้ผู้ใช้ไม่สามารถรับรู้ได้ว่าบริการนี้จะคืนค่ากลับมาเป็นไฟล์

6.4 แนวทางการพัฒนา

1. สามารถเพิ่มข้อมูลลงในฐานข้อมูลเพื่อขยายขอบเขตการให้บริการข้อมูลออกนอกพื้นที่ได้อีก
2. สามารถเพิ่มจำนวนเลเยอร์ของสถานที่อื่นๆและเพิ่มข้อมูลรายละเอียดของสถานที่ต่างๆได้
3. สามารถเพิ่มบริการการแก้ไข หรือเพิ่มเลเยอร์ได้
4. เพิ่มความสามารถในการให้บริการแผนที่เช่น สามารถเลือกสีของแต่ละเลเยอร์ได้
5. พัฒนาแอปพลิเคชั่นบนอุปกรณ์ PDA เพื่อเรียกใช้บริการเว็บเซอร์วิส
6. พัฒนาให้เว็บเซอร์วิสสามารถค้นหาข้อมูลเกี่ยวกับระบบสารสนเทศภูมิศาสตร์จากเว็บเซอร์วิสอื่นๆที่ให้บริการเพื่อความสะดวกของผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

WSDL ที่ใช้อธิบายบริการบน GIS เว็บเซอร์วิส

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions targetNamespace="urn:lbs" xmlns="http://schemas.xmlsoap.org/wsdl/"
xmlns:apachesoap="http://xml.apache.org/xml-soap" xmlns:impl="urn:lbs" xmlns:intf="urn:lbs"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:wsdlsoap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
 <wsdl:types>
  <schema targetNamespace="http://xml.apache.org/xml-soap"
xmlns="http://www.w3.org/2001/XMLSchema">
   <import namespace="http://schemas.xmlsoap.org/soap/encoding/"/>
   <complexType name="Vector">
    <sequence>
     <element maxOccurs="unbounded" minOccurs="0" name="item" type="xsd:anyType"/>
    </sequence>
   </complexType>
  </schema>
  <schema targetNamespace="urn:lbs" xmlns="http://www.w3.org/2001/XMLSchema">
   <import namespace="http://schemas.xmlsoap.org/soap/encoding/"/>
   <complexType name="ArrayOf_xsd_string">
    <complexContent>
     <restriction base="soapenc:Array">
      <attribute ref="soapenc:arrayType" wsdl:arrayType="xsd:string[]"/>
     </restriction>
    </complexContent>
   </complexType>
  </schema>
 </wsdl:types>
  <wsdl:message name="ws_GetNearestResponse">
    <wsdl:part name="ws_GetNearestReturn" type="impl:ArrayOf_xsd_string"/>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    </wsdl:message>
    <wsdl:message name="ws_GetSVGResponse">
      <wsdl:part name="ws_GetSVGReturn" type="xsd:base64Binary"/>
    </wsdl:message>
    <wsdl:message name="ws_GetGMLResponse">
      <wsdl:part name="ws_GetGMLReturn" type="xsd:anyType"/>
    </wsdl:message>
    <wsdl:message name="ws_UploadFileResponse">
      <wsdl:part name="ws_UploadFileReturn" type="xsd:boolean"/>
    </wsdl:message>
    <wsdl:message name="ws_GetPlaceDataFileResponse">
      <wsdl:part name="ws_GetPlaceDataFileReturn" type="xsd:base64Binary"/>
    </wsdl:message>
    <wsdl:message name="ws_GetListLayerFileRequest">
    </wsdl:message>
    <wsdl:message name="ws_UploadFileRequest">
      <wsdl:part name="in0" type="xsd:string"/>
    </wsdl:message>
    <wsdl:message name="ws_GetSVGRequest">
    </wsdl:message>
    <wsdl:message name="ws_GetNearestRequest">
      <wsdl:part name="in0" type="xsd:string"/>
      <wsdl:part name="in1" type="xsd:float"/>
      <wsdl:part name="in2" type="xsd:float"/>
    </wsdl:message>
    <wsdl:message name="ws_GetGMLRequest">
    </wsdl:message>
    <wsdl:message name="ws_GetPlaceDataFileRequest">
      <wsdl:part name="in0" type="apachesoap:Vector"/>
    </wsdl:message>
    <wsdl:message name="ws_GetListLayerFileResponse">
      <wsdl:part name="ws_GetListLayerFileReturn" type="xsd:base64Binary"/>
    </wsdl:message>
    <wsdl:message name="ws_GetAreaPlaceResponse">
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    <wsdl:part name="ws_GetAreaPlaceReturn" type="impl:ArrayOf_xsd_string"/>
   </wsdl:message>
   <wsdl:message name="ws_GetAreaPlaceRequest">
    <wsdl:part name="in0" type="xsd:string"/>
    <wsdl:part name="in1" type="xsd:int"/>
    <wsdl:part name="in2" type="xsd:float"/>
    <wsdl:part name="in3" type="xsd:float"/>
   </wsdl:message>
   <wsdl:portType name="WsInf">
    <wsdl:operation name="ws_UploadFile" parameterOrder="in0">
     <wsdl:input message="impl:ws_UploadFileRequest" name="ws_UploadFileRequest"/>
     <wsdl:output message="impl:ws_UploadFileResponse" name="ws_UploadFileResponse"/>
    </wsdl:operation>
    <wsdl:operation name="ws_GetListLayerFile">
     <wsdl:input message="impl:ws_GetListLayerFileRequest"
name="ws_GetListLayerFileRequest"/>
     <wsdl:output message="impl:ws_GetListLayerFileResponse"
name="ws_GetListLayerFileResponse"/>
    </wsdl:operation>
    <wsdl:operation name="ws_GetPlaceDataFile" parameterOrder="in0">
     <wsdl:input message="impl:ws_GetPlaceDataFileRequest"
name="ws_GetPlaceDataFileRequest"/>
     <wsdl:output message="impl:ws_GetPlaceDataFileResponse"
name="ws_GetPlaceDataFileResponse"/>
    </wsdl:operation>
    <wsdl:operation name="ws_GetSVG">
     <wsdl:input message="impl:ws_GetSVGRequest" name="ws_GetSVGRequest"/>
     <wsdl:output message="impl:ws_GetSVGResponse" name="ws_GetSVGResponse"/>
    </wsdl:operation>
    <wsdl:operation name="ws_GetGML">
     <wsdl:input message="impl:ws_GetGMLRequest" name="ws_GetGMLRequest"/>
     <wsdl:output message="impl:ws_GetGMLResponse" name="ws_GetGMLResponse"/>
    </wsdl:operation>
    <wsdl:operation name="ws_GetNearest" parameterOrder="in0 in1 in2">
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        <wsdl:input message="impl:ws_GetNearestRequest" name="ws_GetNearestRequest"/>
        <wsdl:output message="impl:ws_GetNearestResponse" name="ws_GetNearestResponse"/>
      </wsdl:operation>
      <wsdl:operation name="ws_GetAreaPlace" parameterOrder="in0 in1 in2 in3">
        <wsdl:input message="impl:ws_GetAreaPlaceRequest" name="ws_GetAreaPlaceRequest"/>
        <wsdl:output message="impl:ws_GetAreaPlaceResponse" name="ws_GetAreaPlaceResponse"/>
      </wsdl:operation>
    </wsdl:portType>
    <wsdl:binding name="lbsSoapBinding" type="impl:WsInf">
      <wsdlsoap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>
      <wsdl:operation name="ws_UploadFile">
        <wsdlsoap:operation soapAction=""/>
        <wsdl:input name="ws_UploadFileRequest">
          <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
 namespace="urn:lbs" use="encoded"/>
        </wsdl:input>
        <wsdl:output name="ws_UploadFileResponse">
          <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
 namespace="urn:lbs" use="encoded"/>
        </wsdl:output>
      </wsdl:operation>
      <wsdl:operation name="ws_GetListLayerFile">
        <wsdlsoap:operation soapAction=""/>
        <wsdl:input name="ws_GetListLayerFileRequest">
          <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
 namespace="urn:lbs" use="encoded"/>
        </wsdl:input>
        <wsdl:output name="ws_GetListLayerFileResponse">
          <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
 namespace="urn:lbs" use="encoded"/>
        </wsdl:output>
      </wsdl:operation>
      <wsdl:operation name="ws_GetPlaceDataFile">
        <wsdlsoap:operation soapAction=""/>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        <wsdl:input name="ws_GetPlaceDataFileRequest">
          <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  namespace="urn:lbs" use="encoded"/>
        </wsdl:input>
        <wsdl:output name="ws_GetPlaceDataFileResponse">
          <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  namespace="urn:lbs" use="encoded"/>
        </wsdl:output>
      </wsdl:operation>
      <wsdl:operation name="ws_GetSVG">
        <wsdlsoap:operation soapAction=""/>
        <wsdl:input name="ws_GetSVGRequest">
          <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  namespace="urn:lbs" use="encoded"/>
        </wsdl:input>
        <wsdl:output name="ws_GetSVGResponse">
          <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  namespace="urn:lbs" use="encoded"/>
        </wsdl:output>
      </wsdl:operation>
      <wsdl:operation name="ws_GetGML">
        <wsdlsoap:operation soapAction=""/>
        <wsdl:input name="ws_GetGMLRequest">
          <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  namespace="urn:lbs" use="encoded"/>
        </wsdl:input>
        <wsdl:output name="ws_GetGMLResponse">
          <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  namespace="urn:lbs" use="encoded"/>
        </wsdl:output>
      </wsdl:operation>
      <wsdl:operation name="ws_GetNearest">
        <wsdlsoap:operation soapAction=""/>
        <wsdl:input name="ws_GetNearestRequest">
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
    namespace="urn:lbs" use="encoded"/>
          </wsdl:input>
          <wsdl:output name="ws_GetNearestResponse">
            <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
    namespace="urn:lbs" use="encoded"/>
          </wsdl:output>
        </wsdl:operation>
        <wsdl:operation name="ws_GetAreaPlace">
          <wsdlsoap:operation soapAction=""/>
          <wsdl:input name="ws_GetAreaPlaceRequest">
            <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
    namespace="urn:lbs" use="encoded"/>
          </wsdl:input>
          <wsdl:output name="ws_GetAreaPlaceResponse">
            <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
    namespace="urn:lbs" use="encoded"/>
          </wsdl:output>
        </wsdl:operation>
      </wsdl:binding>
      <wsdl:service name="WsInfService">
        <wsdl:port binding="impl:lbsSoapBinding" name="lbs">
          <wsdlsoap:address location="http://localhost:8080/axis/services/lbs"/>
        </wsdl:port>
      </wsdl:service>
    </wsdl:definitions>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] David Hunter, Curt Cagle, Dave Gibbons, Nikola Ozu, Jon Pinnock, Paul Spencer; "คัมภีร์การใช้ XML ฉบับสมบูรณ์" , "บริษัท ซีเอ็ดยูเคชั่น จำกัด", 2002

[2] ศุภชัย สมพานิช; "เข้าใจและใช้งานภาษา XML ฉบับโปรแกรมเมอร์", "อินโฟร์เพรส",2001

[3] ดร. วีระศักดิ์ ซึงถาวร,"JAVA PROGRAMMING VOLUME I", "บริษัท ซีเอ็ดยูเคชั่น จำกัด",2000

[4] ดร. วีระศักดิ์ ซึงถาวร,"JAVA PROGRAMMING VOLUME II", "บริษัท ซีเอ็ดยูเคชั่น จำกัด",2002

[5] ภัทรพงศ์ น้อยเรือง, ประภาพร ช่างไม้,"คู่มือการใช้งาน Linux ฉบับ Admin.","อินโฟร์เพรส" ,2001

[6] Richard Monson-Haefel,"J2EE Web Services", "Addison-Wesley",2004

[7] http://www.wsiam.com/

[8] http://www.postgresql.org/

[9] http://postgis.refractions.net/

[10] http://www.w3.org/TR/SVG/shapes.html#PointsBNF

[11] http://xml.apache.org/

[12] http://www.uk-valuations.com/gis01.html

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้