

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบให้บริการแผนที่บนเว็บและพีดีเอ
MAP SERVICE ON THE WEB AND PDA



โดย

นาย นัฐพล ภคพงศ์พันธุ์

นางสาว ธิมาพร พงษ์รัตน์

อาจารย์ที่ปรึกษา

ดร. วิศิษฎ์ หิรัญกิตติ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2546

เลขที่..... เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า

เลขทะเบียน 55109..... ห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีครุภัณฑ์ไปใช้

วัน,เดือน,ปี - 8 เม.ย. 2548

.....

ปริญญาานิพนธ์ ปีการศึกษา 2546

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบให้บริการแผนที่บนเว็บและพีดีเอ

MAP SERVICE ON THE WEB AND PDA

คณะผู้จัดทำ นางสาว ธิมาพร พงษ์รัตน์ รหัส 43010184

นาย นัฐพล ภคพงศ์พันธุ์ รหัส 43010211



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบให้บริการแผนที่เว็บและพีดีเอ

นางสาว ชีมาพร พงษ์รัตน์	43010184
นัฐพล ภคพงษ์พันธุ์	43010211
ดร. วิศิษฎ์ หิรัญกิตติ	อาจารย์ที่ปรึกษา
ปีการศึกษา 2546	

บทคัดย่อ

ปัญญานิพนธ์นี้ต้องการพัฒนาระบบให้บริการข้อมูลแผนที่ดิจิทัลซึ่งสามารถให้บริการได้ทั้งบนเว็บและบนเครื่องพีดีเอซึ่งเป็นอุปกรณ์พกพา เพื่ออำนวยความสะดวกให้กับผู้ที่ต้องการใช้บริการแผนที่ โดยมุ่งเน้นพัฒนาแผนที่ที่สามารถปฏิสัมพันธ์กับผู้ใช้ได้มากกว่าแผนที่ดิจิทัลโดยทั่วไป เช่นสามารถแสดงข้อมูลของต่างๆของสถานที่นั้นเมื่อนำเมาส์ไปคลิกบนรูปสถานที่ หรือสามารถเชื่อมต่อไปยังเว็บไซต์ของสถานที่นั้นได้และยังสามารถค้นหาสถานที่ต่างๆที่ผู้ใช้ต้องการ

ปัญญานิพนธ์ฉบับนี้นำเสนอการพัฒนาแผนที่ด้วยการออกแบบและสร้างฐานข้อมูลเชิงพื้นที่ (spatial database) ในการพัฒนาแผนที่นั้นข้อมูลแผนที่ถูกออกแบบให้อยู่ในรูปแบบของภาษา GML (Geography Markup Language) สำหรับการแสดงแผนที่ทำโดยการแปลงข้อมูลแผนที่ในภาษา GML ไปเป็นภาษา SVG (Scalable Vector Graphics) ด้วย XSLT (eXtensible Stylesheet Language Transformation) และการเพิ่มความสามารถของภาษาจาวาสคริปต์เข้าไปในข้อมูล SVG ทำให้ไฟล์เอกสาร SVG ที่ได้จากการแปลงข้างต้นสามารถแสดงแผนที่ที่สามารถปฏิสัมพันธ์กับผู้ใช้ได้ดียิ่งขึ้น ผลลัพธ์แผนที่ในรูปแบบของภาษา SVG สามารถแสดงผลบนเว็บเบราว์เซอร์ทั่วไปได้ รวมทั้งแสดงผลในบาคีทเบราว์เซอร์ (Batik Browser) ซึ่งเป็นเอสวีจีเบราว์เซอร์ (SVG browser) บน platform Java

เทคโนโลยีที่นำมาใช้ในการพัฒนาระบบให้บริการแผนที่ในโครงงานนี้ประกอบด้วย XML, GML, SVG, GPS, และ JavaServer Pages ทำให้แผนที่นี้สามารถใช้งานได้กับเว็บทุก Platform และเทคโนโลยีจาวาทำให้ระบบนี้สามารถใช้งานได้บนทุกระบบปฏิบัติการ ดังนั้นจึงเป็นวิธีการที่เหมาะสมที่จะนำไปใช้พัฒนาระบบให้บริการแผนที่ในอนาคตได้เป็นอย่างดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Map service on the web and PDA

Timaporn	Pongrat	43010184
Nattapol	Pakapongpun	43010211
Dr. Visit	Hirankitti	Advisor

ABSTRACT

The purpose of this thesis is to develop a system which can provide (digital) map data services for the clients using an internet browser and a PDA (Personal Digital Assistant). The map data provided by our system is easier and more interactive to use. Just move the mouse over a sensitive place on the map, the map will pop-up a message giving more information of that place. With one-mouse click on an object in the map, some multi-media content related to the object will be presented to the user, alternatively the system may follow the link to a website related to the object.

This thesis has proposed a development of a system providing map services using different technologies. First, the map data is represented with its meaning using GML (Geography Markup Language) whereas the low level data is stored in spatial databases and managed by a spatial database management system. Since the map data represented in GML cannot be displayed, this is mainly because GML is not designed for the display purpose but only to describe map data, we have to employ another markup language in order to display the GML content. SVG (Scalable Vector Graphics Markup Language) has been selected for this purpose. Therefore, to display map we have to transform GML data to SVG data using some simple rules for the mapping; and the mapping is performed by XSLT (eXtensible Stylesheet Language TRSformation); the resulting SVG data after the transformation is then displayed on a web browser or a SVG browser. To make the SVG content more interactive, we add JavaScript codes into the SVG data. The result of this is a more interactive map represented in SVG codes together with some JavaScript codes. This interactive map can be viewed on a normal web browser with SVG plug-in and on any SVG viewer, especially Batik, a Java SVG viewer. We also use Batik as one way to view our map. Since Batik can be ported to different operating system, this makes our map service even more widely supported.

The technologies that have been adopted in this work are XML, GML, SVG, GPS (Global Positioning Satellites) and JavaServer Page. With the cross-platform map data represented in the open markup languages and the development of the system based on the Java platform, our map service can be made available in a wide-range of platforms, this approach is suitable for building a system providing map services in the future.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญาานิพนธ์ฉบับนี้คงไม่อาจสำเร็จได้ด้วยดี หากไม่ได้รับความช่วยเหลือ และร่วมมือจากหลายๆ ฝ่ายด้วยกัน บุคคลแรกที่จะขาดเสียไม่ได้ในความสำเร็จดังกล่าวนี้ คือ อาจารย์ วิศิษฎ์ หิรัญกิตติ อาจารย์ที่ปรึกษาปริญญาานิพนธ์ที่ไม่เพียงแต่สละเวลาเท่านั้น แต่ยังสละแรงกายและแรงใจในการเอาใจใส่ดูแล แนะนำ และช่วยเหลือเสมอมาจนปริญญาานิพนธ์ฉบับนี้สำเร็จลุล่วงด้วยดี อีกทั้งคุณอาจารย์ทุกท่านที่ถ่ายทอดวิชาความรู้ มาให้ ซึ่งในความกรุณาของอาจารย์ที่มีต่อขอขอบคุณเป็นอย่างสูง

นอกจากนี้ต้องขอขอบคุณบริษัท ThaiMapGuide ที่ให้ความอนุเคราะห์ในด้านฐานข้อมูลของ กรุงเทพมหานครและขอขอบคุณสำนักงานกองทุนสนับสนุนการวิจัยฝ่ายอุตสาหกรรม โครงการโครงการงาน อุตสาหกรรมสำหรับปริญญาตรีปีการศึกษา 2546 สำหรับทุนสนับสนุนการวิจัย ขอขอบคุณเพื่อนๆ ทุกคนที่ได้ ช่วยเหลือและให้คำปรึกษาทางด้านเทคนิคต่างๆ ที่ใช้ในการทำงาน คอยให้กำลังใจและการสนับสนุนต่างๆ อย่างดียิ่ง

สุดท้ายต้องขอขอบคุณบุคคลสำคัญที่สุดที่ทำให้อันเป็นที่เคารพภักดี ซึ่งได้ เลี้ยงดูมาเป็นอย่างดี พร้อมทั้งให้โอกาสในการศึกษาอย่างเต็มที่ และยังให้กำลังใจ เอาใจใส่เสมอมาในทุกๆ ด้าน อันที่หาที่เปรียบมิได้ ข้าพเจ้าขอระลึกในพระคุณ และขอกราบขอขอบคุณมา ณ ที่นี้

ธิดาพร พงษ์รัตน์

นัฐพล ภคพงศ์พันธุ์

สารบัญ

	หน้าที่
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VII
สารบัญรูปภาพ	VIII
บทที่ 1 บทนำ	
1.1.ความเป็นมาและความสำคัญ	1
1.2 วัตถุประสงค์ของปริญญานิพนธ์	2
1.3 ขอบเขตของปริญญานิพนธ์	2
1.4 วิธีการดำเนินงาน	2
บทที่ 2 ทฤษฎี	
2.1. ภาษา XML (Extensive Markup Language)	4
2.1.1 ลักษณะ โครงสร้างของ XML	4
2.1.1.1 Tag	5
2.1.1.2 Element	6
2.1.1.3 Content	6
2.1.1.4 Attribute	6
2.1.2 การแปลงรูปแบบเอกสาร XML (Transforming XML)	6
2.1.2.1 The XSL Transformation Language (XSLT)	7
2.1.3 DOM (Document Object Model)	7
2.2. ภาษา GML (Geography Markup Language)	8
2.2.1. ข้อดีของการใช้ GML ในการแสดงแผนที่	8
2.2.2 GML BASIC CONCEPTS	9
2.2.3 ENCODING GEOMETRY	11
2.2.3.1 GML Feature Collections	11
2.2.3.2 GML application schema	13
2.3. ภาษา SVG (Scalable markup Language)	14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้าที่
2.3.1 การสร้างภาพโดยใช้ภาษา SVG	14
2.3.2 การแสดงภาพโดยใช้ภาษา SVG บนบราวเซอร์	17
2.4 จาวาสคริปต์ (JavaScript)	17
2.5 ระบบฐานข้อมูลเชิงพื้นที่ (Spatial database system)	18
2.5.1 Model of spatial information	18
2.5.2 Spatial Data Type	20
2.5.3 Spatial Relationships	20
2.5.4 ขั้นตอนการออกแบบฐานข้อมูลเชิงพื้นที่	21
2.6 การกำหนดตำแหน่งด้วยระบบจีพีเอส	21
2.6.1 มาตรฐานในการสื่อสารข้อมูล NMEA-0183	22
บทที่3 หลักในการออกแบบ	
3.1 บทนำ	23
3.2 โครงสร้างและการทำงานของระบบให้บริการแผนที่บนเว็บ	23
3.3 การออกแบบระบบในส่วนของเซิร์ฟเวอร์	24
3.3.1 การออกแบบฐานข้อมูลแผนที่	24
3.3.2 การพัฒนาแผนที่โดยใช้ภาษา GML	31
3.3.2.1 การสร้าง GML Application Schema	32
3.3.2.2 การสร้างไฟล์ GML	35
3.3.3 การนำแผนที่ในโครงสร้างของภาษา GML ไปแสดงผลโดยใช้ภาษา SVG	36
3.3.3.1 การเพิ่มความสามารถของแผนที่โดยใช้ภาษา SVG DOM	38
3.3.4 การออกแบบเว็บแอปพลิเคชัน	41
3.3.4.1 ส่วนติดต่อกับฐานข้อมูล	41
3.3.4.2 ส่วนประมวลผล	43
3.3.4.3 ส่วนติดต่อกับผู้ใช้งาน	46
3.5 ส่วนของเครื่องไคลเอนท์	46
3.5.1 ส่วนของบราวเซอร์ที่ใช้ในการเรียกดูเว็บไซต์ของระบบ	46
3.5.2 ส่วนค้นหาค่าพิกัดของผู้ใช้	46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้าที่
บทที่ 4 การออกแบบระบบให้บริการแผนที่บนเครื่องพีดีเอ	
4.1 โครงสร้างและการทำงานของระบบให้บริการแผนที่บนเครื่องพีดีเอ	49
4.2 การออกแบบระบบให้บริการแผนที่บนเครื่องพีดีเอ	50
4.3 สิ่งที่ต้องการในการติดตั้งโปรแกรม Batik	50
บทที่ 5 ผลการทดลอง	
5.1 การใช้งาน	51
5.2 ผลการทดลองใช้งานระบบให้บริการแผนที่บนเว็บ	52
5.2.1 เมื่อเปิดเข้าสู่หน้าแรกของเว็บ	52
5.2.2 การทดลองหาค่าพิกัดที่ผู้ใช้อยู่โดยผ่านเครื่องรับสัญญาณจีพีเอส	52
5.2.3 การทดลองทำการย่อภาพแผนที่	53
5.2.4 การทดลองทำการเลื่อนแผนที่	53
5.2.5 การทดลองแสดงผลเฉพาะ Layer	54
5.2.6 การแสดงข้อมูลของสถานที่ต่างๆบนแผนที่	54
5.2.7 การทดลองค้นหาสถานที่	55
5.2.8 การทดลองค้นหาสถานที่ที่ใกล้ที่สุดจากตำแหน่งที่ผู้ใช้อยู่	56
5.2.9 การทดลองค้นหาถนน	56
5.3 การทดลองใช้บราวเซอร์บาติก (Batik) ในการแสดงผลหน้าเว็บ	57
บทที่ 6 บทวิจารณ์และสรุป	
6.1 บทวิจารณ์และสรุป	58
6.2 ปัญหาที่เกิดขึ้นในการพัฒนาและแนวทางการแก้ปัญหา	58
6.3 ข้อจำกัดของระบบ	59
6.4 แนวทางการพัฒนา	59
ภาคผนวก ก XSLT Stylesheet ที่ใช้ในการแปลง GML เป็น SVG	60
ภาคผนวก ข GML Application Schema (Bangkok.xsd)	69
บรรณานุกรม	74

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้าที่
ตารางที่ 2-1 SVG Element	16
ตารางที่ 2-2 SVG Attribute	17
ตารางที่ 2-3 ความหมายของเขตข้อมูล NMEA GGA, Global Positioning System Fix Data	22
ตารางที่ 3-1 เขตข้อมูลและคำอธิบายของตาราง Path	26
ตารางที่ 3-2 เขตข้อมูลและคำอธิบายของตาราง Road_Name	26
ตารางที่ 3-3 เขตข้อมูลและคำอธิบายของตาราง Hospital	27
ตารางที่ 3-4 เขตข้อมูลและคำอธิบายของตาราง Police	28
ตารางที่ 3-5 เขตข้อมูลและคำอธิบายของตาราง Bank	28
ตารางที่ 3-6 เขตข้อมูลและคำอธิบายของตาราง Mall	29
ตารางที่ 3-7 เขตข้อมูลและคำอธิบายของตาราง Restaurant	30
ตารางที่ 3-8 เขตข้อมูลและคำอธิบายของตาราง TourPoint	31
ตาราง 3-9 ความสัมพันธ์ของเขตของข้อมูลจากตาราง Hospital และ GML	36
ตารางที่ 3-10 ความสัมพันธ์ระหว่าง Element ของ GML และ SVG	37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

	หน้าที่
รูปที่ 2-1 การใช้งาน DOM	8
รูปที่ 2-2 The geometry model for simple features (Open GIS Consortium, 2001)	10
รูปที่ 2-3 Features and properties	11
รูปที่ 2-4 Points	12
รูปที่ 2-5 LineString	12
รูปที่ 2-6 Polygon	13
รูปที่ 2-7 โครงสร้างพื้นฐานที่ใช้ใน Object-Base Model	19
รูปที่ 2-8 โครงสร้างพื้นฐานที่ใช้ใน Field-Base Model	19
รูปที่ 2-9 โครงสร้างและความสัมพันธ์ภายในของ Spatial geometry ในรูปแบบของ UML	20
รูปที่ 3-1 โครงสร้างและการทำงานของระบบ. ให้บริการแผนที่บนเว็บ	23
รูปที่ 3-2 การทำงานของระบบ. ให้บริการแผนที่บนเว็บ	24
รูปที่ 3-3 ER-Diagram ของระบบ	25
รูปที่ 3-4 ชนิดของข้อมูลและ Primary key ของตาราง path	26
รูปที่ 3-5 ชนิดของข้อมูลและ Primary key ของตาราง Hospital	26
รูปที่ 3-6 ชนิดของข้อมูลและ Primary key ของตาราง Police	27
รูปที่ 3-7 ชนิดของข้อมูลและ Primary key ของตาราง Bank	28
รูปที่ 3-8 ชนิดของข้อมูลและ Primary key ของตาราง Mall	29
รูปที่ 3-9 ชนิดของข้อมูลและ Primary key ของตาราง Restaurant	29
รูปที่ 3-10 ชนิดของข้อมูลและ Primary key ของตาราง TourPoint	30
รูปที่ 3-11 ชนิดของข้อมูลและ Primary key ของตาราง Bangkok.xsd	31
รูปที่ 3-12 Model ของ Bangkok.xsd	32
รูปที่ 3-13 ขั้นตอนในการแปลงไฟล์ GML ไปเป็น SVG	38
รูปที่ 3-14 ขั้นตอนการทำงานของฟังก์ชัน showInfo	40
รูปที่ 3-15 ขั้นตอนการทำงานของฟังก์ชัน Animate	41
รูปที่ 3-16 ขั้นตอนการทำงานของโปรแกรมค้นหาสถานที่	44
รูปที่ 3-17 ขั้นตอนการทำงานของโปรแกรมค้นหาสถานที่	45
รูปที่ 4-1 โครงสร้างของระบบให้บริการแผนที่บนเครื่องพีซี	49
รูปที่ 4-2 การทำงานของระบบ. ให้บริการแผนที่บนเครื่องพีซี	50
รูปที่ 5.1 หน้าแรกของระบบ	52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ(ต่อ)

	หน้าที่
รูปที่ 5.2 การแสดงผลการอ่านค่าพิกัดของผู้ใช้จากเครื่องรับสัญญาณจีพีเอส	52
รูปที่ 5.3 การแสดงผลแผนที่ที่ทำการย่อภาพ	53
รูปที่ 5.4 การแสดงผลการเลื่อนภาพแผนที่	53
รูปที่ 5.5 Layer Control Tool	54
รูปที่ 5.6 การแสดงผลแผนที่เฉพาะLayer ที่ต้องการ	55
รูปที่ 5.7 การแสดงข้อมูลของสถานที่ต่าง ๆ บนแผนที่	55
รูปที่ 5.8 แสดงค้นหาสถานที่โดยเลือกประเภทของสถานที่ที่ต้องการ	56
รูปที่ 5.9 การค้นหาสถานที่ที่อยู่ใกล้กับผู้ใช้งานที่สุด	57
รูปที่ 5.10 การใช้บราวเซอร์บาติก (Batik) ในการแสดงผลแผนที่	57



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญ

กิจกรรมต่างๆ ในชีวิตประจำวัน ในการทำงาน และ ในการดำเนินธุรกิจ จำเป็นต้องอาศัยการติดต่อสื่อสาร รวมไปถึงการเดินทาง ในปัจจุบันด้วยความก้าวหน้าทางเทคโนโลยีการสื่อสาร ได้แก่ เทคโนโลยีโทรศัพท์เคลื่อนที่และเทคโนโลยีคอมพิวเตอร์แบบพกพา ทำให้การติดต่อสื่อสารมีความสะดวกรวดเร็ว การติดต่อทางธุรกิจมีความคล่องตัว แต่อย่างไรก็ตาม สิ่งที่จะขาดไม่ได้คือการเดินทาง เนื่องจากผู้คนจำเป็นต้องมีการพบปะหรือ การประชุม รวมกระทั่งการขนส่งสินค้า ดังนั้นการพัฒนาเทคโนโลยีที่จะช่วยให้การเดินทางมีความสะดวกรวดเร็ว จึงเป็นสิ่งที่สำคัญไม่ยิ่งหย่อนไปกว่าการสื่อสาร

องค์ประกอบสำคัญประการหนึ่งของการเดินทางคือ แผนที่ โดยที่แผนที่จะเป็นแหล่งเก็บข้อมูลของตำแหน่ง ถนน ตรอก ซอย สถานที่ ฯลฯ ซึ่งมีปริมาณข้อมูลเป็นจำนวนมากทำให้ยากแก่การค้นหา และนอกจากนี้ข้อมูลยังมีการเปลี่ยนแปลงอยู่ตลอดเวลา ทำให้แผนที่กระดาษไม่อำนวยความสะดวกกับผู้ใช้และไม่สามารถรองรับการพัฒนาการเดินทางในอนาคตซึ่งต้องการความสะดวกรวดเร็วและต้องการข้อมูลที่มีความถูกต้องทันสมัยอยู่ตลอดเวลาได้

ปัจจุบันได้มีการพัฒนาแผนที่คอมพิวเตอร์ หรือที่ เรียกว่า “แผนที่ดิจิทัล” ขึ้น และเริ่มมีความนิยมใช้งานกันเพิ่มมากขึ้น แผนที่ดิจิทัลในท้องตลาดสามารถให้ข้อมูลของ ถนน สถานที่ ฯลฯ สามารถค้นหาข้อมูลตำแหน่งของถนน หรือสถานที่ โดยบอกเพียงชื่อ ถนน หรือสถานที่ และอนุญาติให้ผู้ใช้สามารถเพิ่มเติมสถานที่เข้าไปในแผนที่ได้ อย่างไรก็ตาม แผนที่ดิจิทัลที่จะพัฒนาขึ้นในโครงการนี้ มีความพิเศษกว่าแผนที่ดิจิทัลที่มีจำหน่ายในท้องตลาด คือ

1. สามารถใช้งานบนอุปกรณ์คอมพิวเตอร์ขนาดพกพาได้ เช่น คอมพิวเตอร์จำพวกพีดีเอ และ Pocket PC โดยที่ข้อมูลแผนที่ที่สมบูรณ์จะเก็บอยู่ในเซิร์ฟเวอร์ ซึ่งสามารถเก็บข้อมูลได้เป็นปริมาณมาก
2. มีการนำเอาข้อมูลเกี่ยวกับการท่องเที่ยวเก็บไว้ร่วมกับแผนที่ด้วย เช่น ข้อมูลร้านอาหาร สถานที่ท่องเที่ยว ธนาคาร ฯลฯ ซึ่งรวมไปถึงคำแนะนำเกี่ยวกับการติดต่อหน่วยงานราชการ ข้อมูลเฉพาะของสถานที่นั้นๆเช่น เวลาเปิดหรือปิดทำการของสถานที่นั้น เบอร์โทรศัพท์ที่ใช้ในการติดต่อเว็บไซต์ของสถานที่นั้น หรือข้อมูลกิจกรรมพิเศษต่างๆ ซึ่งข้อมูลเหล่านี้จะให้บริการกับผู้ใช้ผ่านโทรศัพท์เคลื่อนที่หรือพีดีเอ ด้วย พร้อมกับข้อมูลแผนที่
3. มีการใช้เทคโนโลยี จีพีเอส เพื่อใช้ระบุตำแหน่งของโทรศัพท์เคลื่อนที่ที่กำลังใช้งานอยู่ว่าอยู่ ณ ตำแหน่งใดบนเครือข่ายการสื่อสาร ซึ่งข้อมูลตำแหน่งนี้จะถูกเปลี่ยนแปลงไปเป็นตำแหน่งบน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แผนที่ได้ ทำให้ระบบให้บริการแผนที่ที่ทราบได้ว่า ณ ขณะนั้นผู้ใช้โทรศัพท์เคลื่อนที่กำลังอยู่ ณ บริเวณใด

4. แผนที่ที่พัฒนาขึ้นเป็นแผนที่ของกรุงเทพฯ และสามารถขยายไปสู่จังหวัดที่เป็นแหล่งท่องเที่ยวสำคัญในประเทศไทยได้ในภายหลัง
5. แผนที่พัฒนาขึ้นโดยอาศัยภาษา GML และภาษา SVG ซึ่งเป็นภาษาที่เป็น Text Based และเป็นมาตรฐานซึ่งสามารถย้ายไปทำงานบนเครื่องคอมพิวเตอร์อื่นๆ ได้ง่าย

1.2 วัตถุประสงค์ของปริญญาานิพนธ์

1. เพื่อพัฒนาระบบที่จะให้ข้อมูลแผนที่และการเดินทางบนเว็บและเครื่อง พีดีเอ
2. เพื่อศึกษาและนำเอาเทคโนโลยี XML มาใช้สร้างระบบบริการแผนที่บนหลาย Platform
3. เพื่อเป็นการบริการให้กับนักท่องเที่ยวและบุคคลทั่วไปที่ต้องเดินทางอยู่ตลอดเวลา
4. พัฒนาแผนที่สมัยใหม่ที่มีความสามารถในการ Interactive กับผู้ใช้งาน

1.3 ขอบเขตของปริญญาานิพนธ์

ปริญญาานิพนธ์นี้มุ่งเน้นการพัฒนาเว็บแผนที่ที่สามารถแสดง

1. ตำแหน่งต่างๆของกรุงเทพและแผนที่บริเวณใกล้เคียงกับตำแหน่งที่ผู้ใช้งานอยู่
2. ค้นหาสถานที่ที่ต้องการแสดงข้อมูลของสถานที่นั้น
3. ค้นหาถนนสายหลักและแสดงถนนเส้นนั้น
4. ให้ข้อมูลเกี่ยวกับรายละเอียดของสถานที่ที่ต้องการ เช่นหมายเลขโทรศัพท์ เวลาทำการของสถานที่นั้นหรือบริการพิเศษต่างๆ
5. สามารถค้นหาสถานที่ที่ต้องการซึ่งอยู่ใกล้ที่สุดจากตำแหน่งปัจจุบันที่ผู้ใช้อยู่

1.4 วิธีการดำเนินงาน

1. ศึกษาเทคโนโลยีที่เกี่ยวข้องที่จำเป็นสำหรับการนำไปใช้ในโครงการ
2. ทดลองเขียนแผนที่โดยใช้ภาษา GML
3. ทดลองนำแผนที่ที่เขียนโดยภาษาGML มาแสดงผล โดยใช้ XSLT
4. ทดลองการแสดงผลแผนที่โดยใช้ภาษา SVG
5. ทดลองการแสดงผลแผนที่ GML บนเว็บไซต์โดยใช้ภาษา JavaScript ร่วมกับ SVG และ XSLT
6. ออกแบบฐานใหม่จากข้อมูลแผนที่เดิมโดยใช้หลักการของ Spatial database
7. เขียนโปรแกรมนำข้อมูลแผนที่ในฐานข้อมูลขึ้นมาแสดงผลบนเว็บไซต์และทดลองการค้นหาข้อมูลในแผนที่โดยใช้ ภาษา จาวาสคริปต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. ปรับปรุงรูปแบบการแสดงผลแผนที่ให้ถูกต้องและเหมาะสม
9. พัฒนาความสามารถต่างๆ ของแผนที่ เช่น ความสามารถในการค้นหาข้อมูลของสถานที่ต่างๆ
10. พัฒนาโปรแกรมที่ทำการวิเคราะห์ข้อมูลที่ได้รับจากเครื่องรับสัญญาณจีพีเอส (GPS receiver) เพื่อหาตำแหน่งของผู้ใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎี

2.1. ภาษา XML (Extensive Markup Language)

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

XML และ HTML จะเป็นส่วนหนึ่งของภาษา SGML ซึ่ง XML จะให้รายละเอียดเกี่ยวกับข้อมูล เช่น ชื่อเมือง อุณหภูมิ ความกดอากาศ ส่วน HTML เป็นการกำหนด Tag ต่างๆ ที่จะกำหนดรูปแบบการแสดงผลของข้อมูลบนหน้าเว็บ ซึ่งข้อมูลจะสามารถแสดงออกมาได้หลายรูปแบบ ขึ้นอยู่กับการกำหนดของ HTML และในปัจจุบันนี้ ด้วย XML จะมีการให้รายละเอียดของเนื้อหาเอกสารที่เรียกว่า Document Type Definition (DTD) ที่ให้รายละเอียดเกี่ยวกับตัวเอกสารว่าจะแสดงหรือซ่อนส่วนไหนของเอกสารบ้าง ซึ่ง DTD จะเป็นส่วนที่เพิ่มเติมสำหรับ XML ถ้าหากมีการส่งข้อมูลในรูปแบบ DTD ก็จะสามารถรู้กันว่าเป็น XML มีความหมายหลาย ๆ คำที่อธิบายลักษณะของ XML

2.1.1 ลักษณะโครงสร้างของ XML

XML เป็นการใช้ข้อความเพื่อบ่งบอกโครงสร้างของเอกสาร พิจารณาตัวอย่างรูปแบบ โครงสร้างของหนังสือ เมื่อหนังสือประกอบด้วยจำนวนบท 2 บท ในแต่ละบทประกอบด้วยเนื้อความ

Begin Book

Begin Chapter 1

Text for Chapter 1

End Chapter 1

Begin Chapter 2

Text for Chapter 2

End Chapter 2

End Book

หนังสือที่มีอยู่ในปัจจุบัน จะมีโครงสร้างที่มีรายละเอียดที่ซับซ้อนมากกว่านี้ เช่น บทนำ, สารบัญ, เป็นต้น เช่นเดียวกัน ภายในส่วนเนื้อความ ยังประกอบด้วย โครงสร้างย่อย คือ ย่อหน้า (Paragraph) แต่ละย่อหน้ายังประกอบขึ้นจาก ประโยค คำ และตัวอักษรด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะของเอกสาร XML นั้น สามารถอธิบายโดยใช้ตัวอย่างที่ 1 ได้ ดังนี้
ตัวอย่างที่ 1

```
<?xmlversion="1.0"encoding="windows-874"?>
  <mail>
    <malisorn>ขึ้นต้นด้วยมะลิซ้อน</malisorn>
    <malila>พอแตกใบอ่อนเป็นมะลิลา</malila>
  </mail>

<?xmlstylesheet type="text/xsl" href=""?>
```

บรรทัดที่ 1 นั้นหมายความว่าเราประกาศเอกสารนี้เป็นเอกสาร XML และมีการเข้ารหัสอักขระเป็น windows-874 เพื่อให้ใช้ภาษาไทยได้ จากนั้นเราจะเห็น สิ่งที่เราคุ้นเคยที่เราเรียกว่า tag แต่จริงๆแล้วในภาษา XML จะแบ่ง โครงสร้างเป็น 2 ส่วนใหญ่ ๆ คือ Tag และ Element สามารถอธิบายเพิ่มเติมได้จากตัวอย่างที่ 2
ตัวอย่างที่ 2

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

ความหมายของ Tag กับ Element

2.1.1.1 Tag

สำหรับใน XML แล้ว tag มีความหมายในลักษณะเดียวกับที่ใช้ใน HTML Tag คือข้อความที่อยู่ระหว่างสัญลักษณ์<และ>

```
Tagเปิด(Start tag)
<book>
จากตัวอย่างที่แสดง ด้านบนถูกเรียกว่า Tag เปิด ดังนั้น Tag เปิดจึงมีสัญลักษณ์คือ <...>
Tagปิด(End Tag)
</book>
```

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.1.2 Element

ในที่นี้คือ โครงสร้างหลักของ XML ซึ่งอยู่ในรูปของ Tag เช่นเดียวกัน ตามตัวอย่างข้างบน Element คือ

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

จะมีลักษณะซ้อนกันเป็นชั้นๆ โดย Root Element เป็น Element แรกสุดของเอกสาร XML

Element ใช้เป็นส่วนประกอบของเนื้อหาของเอกสาร XML และ Attribute เพื่อให้ผู้อ่านได้เข้าใจ ความหมายของศัพท์หลายๆ คำศัพท์ ก่อนที่จะนำไปใช้ในการสร้าง XML ซึ่งจำเป็นอย่างมาก พิจารณา คำจำกัด ความของElement

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

สังเกตจากลักษณะ รูปทางด้านบน ทั้งหมดตั้งแต่ <chap...จนถึง </chap> ถูกเรียกว่า Element หรือถ้า มองง่ายๆคือ Element เริ่มต้นที่ Tag เปิด และสิ้นสุดที่ Tag ปิดใน Tag คำสั่งเดียวกัน

2.1.1.3 Content

เนื้อความ หรือ Content ถือได้ว่าเป็นข้อมูลเพื่อใช้ในการแสดง ให้ผู้อ่านเอกสารได้เห็น หรือกล่าวอีก นัยหนึ่งคือ Content อยู่หลัง Tag เปิด และจบที่ก่อนถึง Tag ปิดนั่นเอง

2.1.1.4 Attribute

Attribute คือข้อมูลความหมายเพิ่มเติม แน่นนอน ถ้ากล่าวถึงสิ่งของ หากเราไม่ได้ให้ความหมาย เพิ่มเติมเราก็จะไม่รู้ และแยกแยะสิ่งที่เหมือนกันออกจากกันได้ เช่น ถ้าพูดถึง บท ในหนังสือ แค่นี้เป็น ความหมายโดยรวม แต่ถ้าบอกว่า บทที่ 1 ในหนังสือ เลข 1 ในที่นี้คือความหมายเพิ่มเติมให้กับบท

Attribute จะ ถูกบรรจุอยู่ภายใน Tag เปิด และ Attribute นี้ในบางครั้ง อาจจะมีหรือไม่มีก็ได้ รวมถึงถ้า มีอาจมีได้มากกว่าหนึ่งตัวในTagเปิด

2.1.2 การแปลงรูปแบบเอกสาร XML (Transforming XML)

XML Stylesheet Language (XSL) เป็นภาษาที่ใช้ในการแปลงจากรูปแบบ (Transformation) ของ เอกสาร XML ไปอยู่ในรูปแบบอื่น ๆ และในขณะเดียวกันก็ทำการประยุกต์รูปแบบการจัดข้อความ (style) ด้วย XSL stylesheet (สไปล์ชีต) ถูกเขียนในรูปแบบที่เหมาะสมกับ XML (XML-compliant format) แต่ต้องทำการ กำหนดลำดับลำดับของอิลิเมนต์ไว้ล่วงหน้าในการกำหนดสิ่งที่ต้องการให้กระทำในการแปลงรูปแบบ (อีก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประการหนึ่งคือเพื่อให้ทราบไว้ว่าจะเลือกใช้อันไหนทำงาน) โดยความหมายจะถูกกำหนดโดยตัวประมวลผล XSL เอง ซึ่งควรจะเป็นไปตามมาตรฐาน ของ W3C ดังนั้นจาก XSL ซึ่งเป็นแอปพลิเคชันหนึ่งของ XML และอยู่ในรูปแบบที่เหมาะสมกับ XML ดังนั้นจึงสามารถประมวลผล XSL สไตลชีตได้เหมือนกับเอกสาร XML ครั้งแรก XSL เป็นข้อเสนอที่ถูกระบุทั้งทางที่การแปลงรูปแบบ XML และไวยากรณ์ในการจัดการรูปแบบบนแพลตฟอร์มที่เป็นอิสระกัน (platform-independent styling grammar) แต่อย่างไรก็ตามในส่วนของการแปลงรูปแบบกำหนดอย่างรวดเร็และเป็นประโยชน์อย่างมาก ขณะที่ไม่มีแอปพลิเคชันในส่วนของการจัดรูปแบบมากนักและส่วนใหญ่จะกล่าวว่ามีประโยชน์ทางด้านการทำงานน้อย ดังนั้นจากข้อเสนอในตอนแรกจึงสามารถแบ่งออกได้เป็น 2 ส่วนที่แตกต่างกันคือ ส่วนในการแปลงรูปแบบเอกสาร ซึ่งปัจจุบันถูกเรียกว่า XSLT (XSL Transformation Language) และในส่วนของการจัดรูปแบบจะถูกเรียกอย่างเป็นทางการว่าวัตถุในการจัดรูปแบบของ XSL (XSL Formatting Object) หรือ XSL-FO แต่ในที่นี้จะพิจารณาเฉพาะในส่วนของการแปลงรูปแบบเท่านั้นเพื่อแปลงรูปแบบจากเอกสาร XML แบบหนึ่งไปเป็น XML อีกแบบหนึ่ง ซึ่งในโครงการนี้ใช้ในการแปลงเอกสาร GML ไปเป็น SVG

2.1.2.1 The XSL Transformation Language (XSLT)

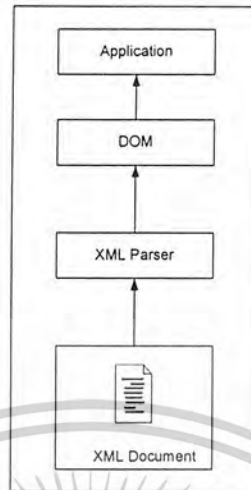
XSLT เป็นข้อเสนอแนะจาก W3C ในการกำหนดเซต Element ของ XML ที่สามารถจะถูกใช้ในการสร้างสไตลชีตที่แปลงเอกสาร XML ให้อยู่ในรูปแบบใดๆ ซึ่งในปัจจุบันนี้ใช้ในการแปลง XML รูปแบบหนึ่งไปเป็นอีกรูปแบบหนึ่ง (ในโครงการนี้ใช้ในการแปลงเอกสาร GML ไปเป็น SVG) โดยการทำงานจะทำโดยนำเอกสาร XML ที่เหมาะสมกับ XML นั้นไปทำการแปลงรูปแบบเพื่อสร้างเอกสารเอาต์พุต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

2.1.3 DOM (Document Object Model)

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

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

1 เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-1 การใช้งาน DOM

2.2. ภาษา GML (Geography Markup Language)

ภาษา GML เป็นรูปแบบหนึ่งของเอกสาร XML ที่ถูกออกแบบขึ้นเพื่อการจัดเก็บและการแลกเปลี่ยนของข้อมูลทางภูมิศาสตร์ ทั้งทางด้าน ความสัมพันธ์ และลักษณะประจำตัวของข้อมูล ในรูปแบบของ feature ที่แตกต่างกันและสามารถกำหนดวิธีที่จะแสดงข้อมูลทางภูมิศาสตร์นั้น โดยใช้ข้อกำหนดของ GML ที่ถูกรับรองโดย องค์กร OGC (OpenGIS Consortium) ในการอธิบายข้อมูลเชิงพื้นที่ เพื่อให้ผู้ที่ทำงานด้าน GIS (Geographic Information System) ใช้เป็นพื้นฐานในการออกแบบพัฒนาแผนที่ในรูปแบบต่างๆ และผู้ใช้จะสามารถดูแผนที่ ที่ถูกแสดงนั้น ผ่านทาง browser มาตรฐานที่ใช้กันอยู่ทั่วไป โดยรายละเอียดเพิ่มเติมสามารถศึกษาได้จาก <http://www.opengis.net/gml/01-029/GML2.pdf>

2.2.1. ข้อดีของการใช้ GML ในการแสดงแผนที่

1. ได้แผนที่ที่มีคุณภาพดีกว่า เนื่องจาก GML จะเปลี่ยนข้อมูลทางภูมิศาสตร์ทั้งหมดให้อยู่ในรูปแบบของ feature หรืออ็อบเจ็กต์ ซึ่งสามารถจะนำไปแสดงผลในรูปแบบที่ไม่จำกัดความละเอียดในการแสดงผล
2. สามารถใช้งานได้บน บราวเซอร์โดยไม่ต้องซื้อโปรแกรมทางที่ใช้แสดงผลทางฝั่ง ไคลเอนท์ เมื่อข้อมูล GML ถูกรับโดยฝั่งไคลเอนท์จะถูกเปลี่ยนให้อยู่ในรูปแบบของภาพแผนที่บนบราวเซอร์โดยใช้ภาษา SVG (Scalable Vector Graphics) ในการแสดงออกมาเป็นภาพ ซึ่งเราสามารถแสดงแผนที่บนบราวเซอร์ โดยไม่จำเป็นต้องติดตั้งซอฟต์แวร์อื่นๆ เพิ่มเติมหากบราวเซอร์สามารถรองรับภาพแบบเวกเตอร์ (vector graphics) ได้ และเราสามารถดาวน์โหลด SVG ซึ่งเป็นโปรแกรมเสริมของ Adobe Corporation ได้โดยไม่ต้องเสียค่าใช้จ่ายใดๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. สามารถสร้างแผนที่ ที่มีรูปแบบเฉพาะได้ เนื่องจาก GML จะเก็บข้อมูลของแผนที่ เช่น ที่ตั้งของ Feature นั้น, ลักษณะทางภูมิศาสตร์, ชนิดของ Feature และ Attribute ต่างๆ โดยไม่กำหนดข้อมูลหรือวิธีในการแสดงผลของข้อมูล แผนที่นั้นๆ ซึ่งทำให้ผู้ใช้สามารถสร้างแผนที่ให้มีรูปแบบได้ตามที่ต้องการ
4. สามารถทำการแก้ไขแผนที่ได้ แผนที่ที่เขียนโดย GML จะถูกดาวน์โหลดและทำการ render บนเบราว์เซอร์ซึ่งเมื่อ GML ถูกเปลี่ยนเป็น SVG ผู้ใช้จะสามารถใช้โปรแกรมบนไคลเอนท์ในการเพิ่มเติมตัวอักษร หรือ ทำการเปลี่ยนแปลงภาพบนแผนที่ และสามารถบันทึกเก็บเป็นไฟล์ภาพเพื่อนำไปใช้ในกรณีอื่นเช่นทำการพิมพ์ออกทางเครื่องพิมพ์ หรือส่ง E mail ได้
5. สามารถเชื่อมโยง link ไปยังลงในแผนที่ได้ ประโยชน์ที่สำคัญของ GML คือผู้ใช้สามารถฝัง URL ลงใน feature ที่มีในแผนที่ ความสามารถนี้ทำให้เราสามารถออกแบบแผนที่ให้มีความแปลกใหม่และทันสมัยยิ่งขึ้น
6. ทำการ query ได้ดียิ่งขึ้น หากผู้ใช้ต้องการที่จะ Click ไปบน Feature บนแผนที่และค้นหาข้อมูลเพิ่มเติมเกี่ยวกับ Feature นั้นๆ (เช่น ชื่อหรือความลึกของแม่น้ำ) ซึ่งไม่สามารถทำได้บนแผนที่ที่สร้างขึ้นโดยใช้ไฟล์ภาพ (GIF/JPG) สำหรับแผนที่ที่ใช้ GML นั้น เราจะสามารถระบุ feature ที่ถูก Click นั้นได้
7. สามารถกำหนดเนื้อหาที่ต้องการแสดงหรือต้องการได้รับ เพราะ GML เป็นภาษาที่อ้างอิงถึง Feature ทำให้ผู้ใช้สามารถกำหนดได้ว่า ต้องการให้มี Feature ใด แสดงบน แผนที่ บ้าง ซึ่งจะช่วยลดระยะเวลาในการรับ
8. สามารถแสดงผลได้บนทุกอุปกรณ์ที่ใช้เบราว์เซอร์ได้ ข้อมูลทางภูมิศาสตร์ใน GML สามารถส่งไปยังอุปกรณ์ชนิดใดก็ได้ที่สามารถรองรับ XML การแสดงผลแบบ XML ได้ เช่นเครื่องพีดีเอหรือโทรศัพท์มือถือทำให้แผนที่นั้นสามารถใช้งานได้อย่างกว้างขวางและหลากหลายยิ่งขึ้น

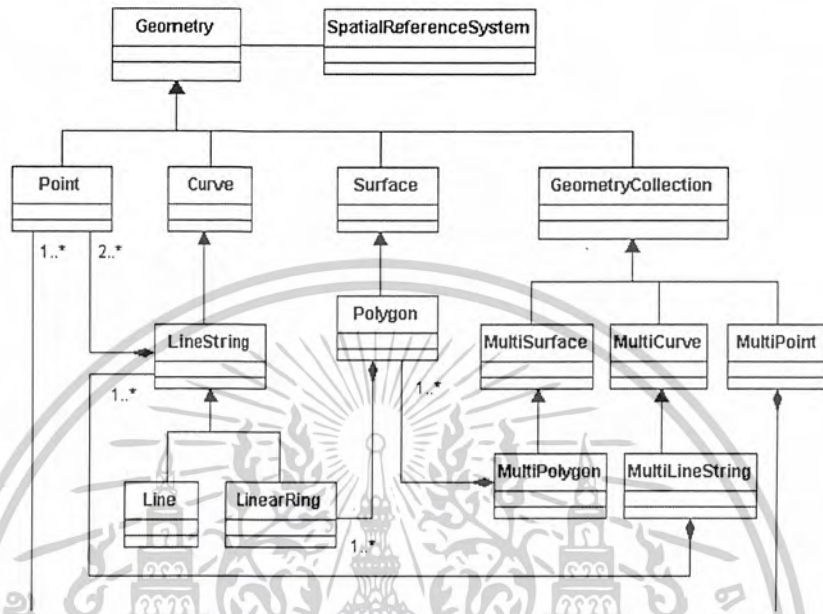
2.2.2 GML BASIC CONCEPTS

GML เป็นรูปแบบหนึ่งของเอกสาร XML ที่ถูกออกแบบขึ้นเพื่อการจัดเก็บและการแลกเปลี่ยนของข้อมูลทางภูมิศาสตร์รองรับการแสดงผลข้อมูลทางภูมิศาสตร์ในรูปแบบต่างๆ

GML version แรกได้ถูกพัฒนาขึ้นโดย OGC และได้รับการพัฒนามาจนถึง version 3.0 แต่ที่ใช้ในโครงการนี้เป็น GML เวอร์ชัน 2.0 ซึ่งมีพื้นฐานทั้งหมดอยู่บน XML Schema ซึ่งเป็นข้อแตกต่างที่สำคัญจาก เวอร์ชัน แรกคือ specification ของ GML 2.0 ถูกกำหนดโดย OGC เรียกว่า simple features รูปที่ 1 โดยแต่ละ simple feature จะมี abstract base geometry class และระบบอ้างอิงระยะทาง (spatial reference system) ที่ อธิบายตำแหน่งในบริเวณที่ อีอบเจกต์ ของ feature นั้น ถูกอ้างอิงถึง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

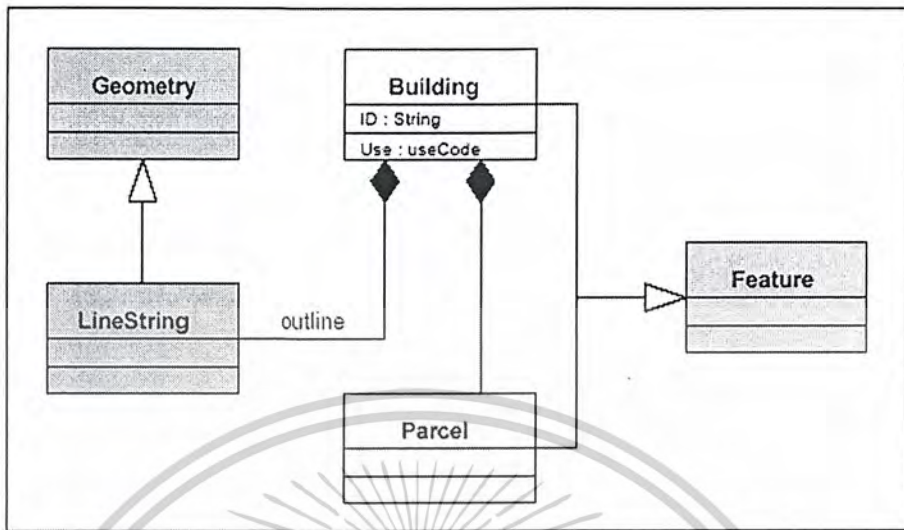
โดยทั่วไปสเกิมา Schemasจะอธิบายถึงลักษณะของ Class ของ อ็อบเจกต์ส แต่ใน XML Schema จะอธิบายว่าข้อมูลจะถูกกำหนด (mark up) ได้อย่างไร



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

องค์ประกอบทางเรขาคณิต (Geographic feature) จะถูกอธิบายด้วยคุณสมบัติ ต่างๆ ซึ่งคุณสมบัติเหล่านี้จะประกอบด้วย คุณสมบัติที่เกี่ยวข้องกับระยะทาง โดยอธิบาย ตำแหน่งและรูปร่างของ feature นั้นๆ แต่ละ feature จะมีชนิดของfeature ซึ่งเทียบเท่ากับ Class ใน อ็อบเจกต์ model ดังนั้น Class definition จึงเป็นส่วนที่กำหนด คุณสมบัติของแต่ละ feature จำเป็นต้องมี เช่น อาคารถูกกำหนดให้มี ชื่อของอาคาร ซึ่งเป็น -ข้อมูลประเภท String ประเภทของการใช้งานอาคาร ซึ่งกำหนดให้ชนิดข้อมูลเป็น Text ที่สามารถเลือกจาก List ที่กำหนดไว้ เส้นรอบนอกซึ่งกำหนดให้เป็น รูปวงแหวน (LinearRing) ซึ่งเป็น คุณสมบัติทางภูมิศาสตร์ รูปที่ 2.2 อธิบายถึง Geographic feature โดยใช้ UML

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-3 Features and properties

2.2.3 ENCODING GEOMETRY

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

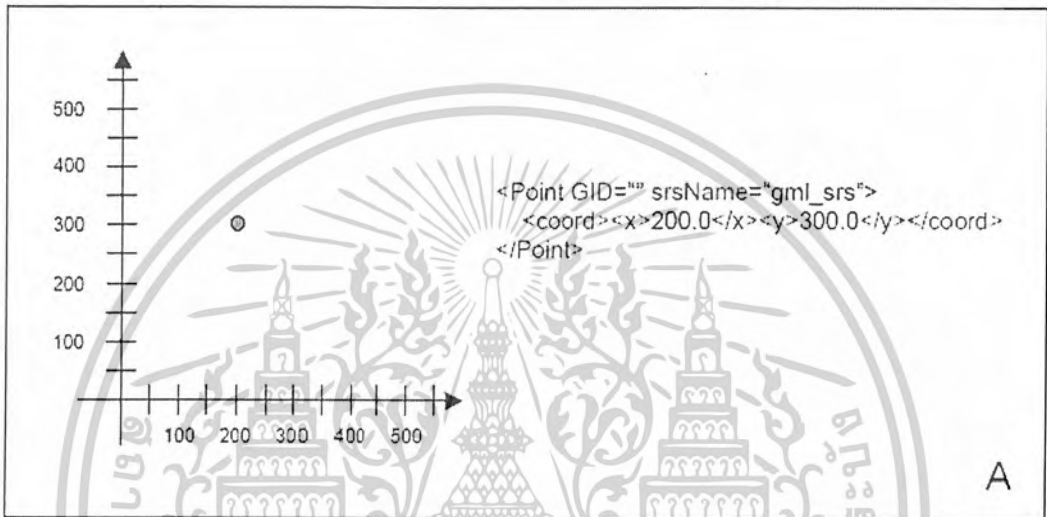
2.2.3.1 GML Feature Collections

- 1) ภาษา GML ใช้ Feature อธิบายองค์ประกอบต่างๆของภูมิประเทศ ในรูปแบบของ Geographic entities
- 2) Feature ประกอบด้วย List ของคุณสมบัติต่างๆไป (simple properties) ซึ่งประกอบด้วยชื่อของข้อมูล ชนิดของข้อมูล และค่าของข้อมูล และ คุณสมบัติทางภูมิศาสตร์ต่างๆเช่น lines, points, curves, polygons
- 3) Feature สามารถสร้างขึ้นจาก Feature อื่นๆ หรือกำหนดชนิดของข้อมูลเป็น Feature อื่นๆได้ ตามหลักของ OGC Simple Feature Model, GML จะรองรับการแสดงผลขององค์ประกอบทางเรขาคณิต คือ จุด (Point), เส้น (LineString), วงแหวน (LinearRing), รูปหลายเหลี่ยม (Polygons) และโดยรูปแบบพื้นฐานที่มีมาให้ เราสามารถเพิ่มเติม MultiPoint, MultiLineString, MultiPolygon และ MultiGeometry. นอกจากองค์ประกอบทางเรขาคณิตเหล่านี้แล้ว GML ยังประกอบด้วย <coordinates> หรือ <coord> ซึ่งเป็น Tag ใช้ในการอ้างอิงถึงจุดคู่ลำดับหรือตำแหน่งและ <Box> ใช้ในการนิยามองค์ประกอบที่มีพื้นผิวเป็นรูปสี่เหลี่ยม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

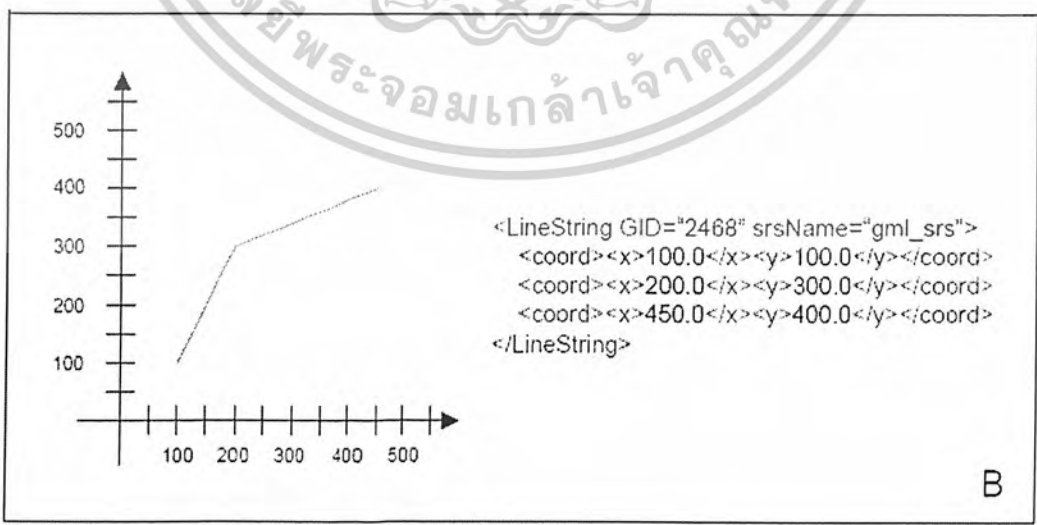
ส่วนที่สำคัญของ GIS คือวิธีการ ในการอ้างอิงถึงองค์ประกอบทางภูมิศาสตร์ต่างๆบนพื้นโลก คู่ลำดับ (coordinate) สำหรับรูปเรขาคณิตที่เราใช้แสดงแทนบนแผนที่นั้น จะอ้างอิงโดย Spatial Reference System (SRS) และทุกองค์ประกอบจะต้องระบุ SRS หรืออีกวิธีหนึ่งคือระบุ GID Attribute ซึ่งเป็น Attributeที่จะแสดงค่าเฉพาะตัวทางภูมิศาสตร์ของแต่ละ องค์ประกอบเรขาคณิตนั้น

Points เป็นรูปเรขาคณิตพื้นฐานที่ประกอบด้วย คู่ลำดับเพียงคู่ลำดับเดียว multipoint เป็น เซตที่ประกอบด้วยจุดหลายๆจุด ดังรูปที่ 2-4



รูปที่ 2-4 Points

LineString ประกอบด้วย List ของ คู่ลำดับแสดงที่ส่วนของเส้นตรงหลายเส้นเชื่อมต่อกัน รูป 2-5 LineString จะต้องประกอบด้วยคู่ลำดับอย่างน้อย 2 คู่

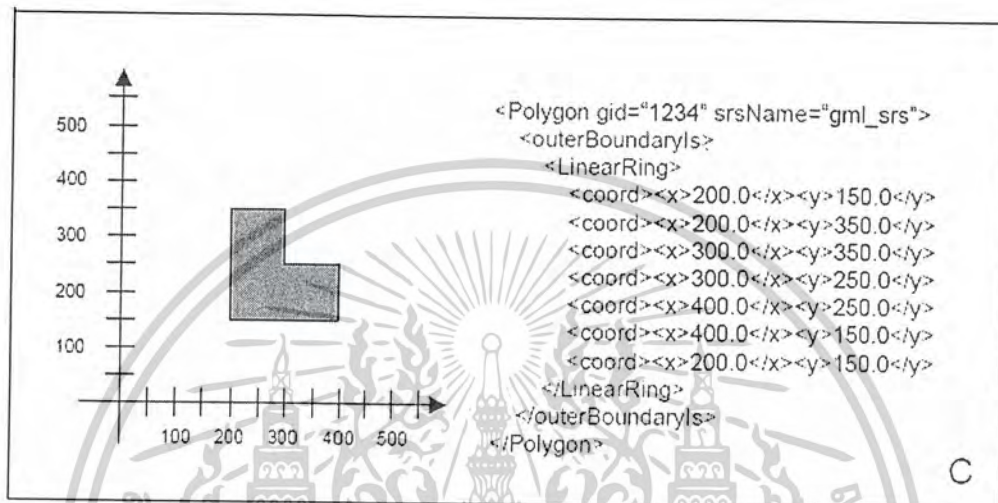


รูปที่ 2-5 LineString

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LinearRing เป็นรูปปิด ประกอบด้วย List ของคู่ลำดับที่แสดงถึงส่วนของเส้นตรง คู่ลำดับสุดท้ายต้องมีค่าเท่ากับคู่ลำดับแรก และต้องประกอบด้วยคู่ลำดับอย่างน้อย 4 คู่

Polygons เป็นพื้นผิวที่ผิวของรูปเป็นเซตของ LinearRing โดยขอบนอกของ Polygon คือ LinearRing วงแรกและ LinearRing วงที่เหลือจะซ้อนเรียงกันภายในของวงแรก.



รูปที่ 2-6 Polygon

2.2.3.2 GML application schema

Schema คือ รูปแบบการจัด โครงสร้างและความสัมพันธ์ของข้อมูล โดยภาษา GML นั้น กำหนดให้ผู้ใช้ต้องกำหนด GML application schema ซึ่งเขียนโดยใช้หลักการของ XML Schema และเป็นไปตามข้อบังคับของ GML .

GML Application Schema ประกอบด้วย

- 1 Annotations ซึ่งเป็นส่วนประกาศหรือส่วนที่ใช้การอธิบาย schema นั้น
2. Simple Type คือชนิดของข้อมูลที่ใช้รูปแบบของข้อมูลที่ GML กำหนดมาให้แล้ว
3. Complex Type คือชนิดของข้อมูลที่สร้างขึ้นใหม่โดยอาจจะเกิดจากการใช้ simple type หลายนๆ ด้มา ประกอบกันหรือเกิดจาก complex type ชนิดอื่นก็ได้
4. Element
5. Attribute

ตัวอย่าง GML Feature Type Schema

```
<element name="RANET Radio Station" type="RadioSchema" />
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

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

```

2.3 ภาษา SVG (Scalable Vector Language)

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

SVG จะกำหนดรูปแบบของวัตถุสามอย่างคือ vector graphic shape (เช่น ส่วนของเส้นตรง และเส้นโค้ง), รูป และ ตัวอักษร โดยสามารถที่จะกำหนด grouped, styled, transformed หรือ compo sited ให้แก่ วัตถุ ใน SVG ได้ อีกทั้งยังเพิ่มคุณสมบัติ transformations, clipping paths, alpha masks, filter effects , template อี้อบเจ็คและอีกมากมาย

เนื่องจาก SVG เป็น XML ดังนั้นจึงมีการสร้าง DOM สำหรับ SVG ขึ้นมาโดยอ้างอิงตามมาตรฐานของ W3 ซึ่งทำให้ SVG มีคุณสมบัติ dynamic และ interactive ตามไปด้วย แม้แต่ event handlers อย่าง onmouseover หรือ onclick ก็สามารถกำหนดลงในแต่ละวัตถุของ SVG ได้ด้วย อีกทั้งการอ้างอิงตามมาตรฐานทำให้ SVG สามารถติดต่อกับ XML อื่นๆ ในเว็บไซต์เดียวกันได้ โดยรายละเอียดเพิ่มเติมสามารถศึกษาได้จาก <http://www.w3.org/TR/2003/REC-SVG11-20030114/>

2.3.1 การสร้างภาพโดยใช้ภาษา SVG

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

```

<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 20000303 Stylable//EN"
"http://www.w3.org/TR/2000/03/WD-SVG-20000303/DTD/svg-20000303-stylable.dtd">
<svg xml:space="preserve" width="5.5in" height=".5in">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

.....
</svg>

ในบรรทัดแรกจะเป็นการบอกให้ผู้ที่น่า SVG ไฟล์นี้ไปอ่านทราบว่า code ต่อไปนี้อ้างอิงจากมาตรฐาน XML version 1.0 โดยข้อมูลที่อยู่ในแต่ละ tag จะใช้มาตรฐาน iso-8859-1 ในการเขียน บรรทัดที่สองจะอ้างอิงถึง DTD หรือ Document Type Definition (DTD เป็นเอกสารที่ใช้บอกถึงชนิดของข้อมูลที่จะใช้ใน XML และ ยังระบุอีกด้วยว่าเอกสารนั้นๆ มีโครงสร้าง เป็นอย่างไร) ใน<svg> element สามารถที่จะบรรจุ text, shapes และ paths ได้ เช่น

1. การเขียน code SVG ในการแสดงผลข้อมูลแบบ Text

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

Hello world

```
</text>
```

2. การเขียน code SVG ในการแสดงผลข้อมูลแบบเส้น

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

3. เขียน code SVG ในการแสดงผลการการวาดภาพสี่เหลี่ยม

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

โดย Element และ Attribute ของ SVG ที่ใช้ในงานบ่อยมีดังต่อไปนี้

SVG Element

SVG	Action
<svg>	เป็น Tag ที่ใช้ในการกำหนดพื้นที่ทั้งหมดของภาพ และเป็น Tag ที่เก็บ Tag ต่างๆของ SVG
<path>	ใช้ในการวาดเส้น
<rect>	ใช้ในการวาดสี่เหลี่ยม
<g>	ใช้ในการรวม Tag ต่างๆให้อยู่ในกลุ่มเดียวกัน
<desc>	ใช้ในการเพิ่มเติมข้อมูล หรือคำอธิบาย
<defs>	ใช้ในการกำหนดค่าต่างๆ
<script>	ในการเขียน script จะเขียนใน Tag นี้
<text>	ใช้ในการแสดงผลแบบอักษร
<cursor>	ใช้ในการเปลี่ยนรูปของ mouse cursor
<line>	ใช้ในการวาดเส้นอย่างง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

ตารางที่ 2-1 SVG Element

SVG Attribute

Attribute	Action
fill	จัดการกับสีของ อ็อบเจกต์ นั้นๆ
id	ใช้ในการแยกแยะ อ็อบเจกต์ ต่างๆ
Width	ขนาดความกว้างของ อ็อบเจกต์
Height	ขนาดความสูงของ อ็อบเจกต์
X	ตำแหน่งพิกัด x
Y	ตำแหน่งพิกัด y
Xmlns	ใช้ในการอ้างอิงถึง namespace
Visibility	กำหนดการปรากฏอยู่ของ อ็อบเจกต์
Style	กำหนดลักษณะของ อ็อบเจกต์
point-event	กำหนดการปฏิสัมพันธ์กับ อ็อบเจกต์
viewBox	กำหนดขนาดในการแสดงผลของภาพ
stroke-width	กำหนดขนาดของเส้น
Stroke	กำหนดสีของเส้น
zoomAndPan	กำหนดความสามารถในการย่อหรือขยาย
OnClick	mouse event
onmouseover	mouse event
onmouseout	mouse event
Transform	เคลื่อนย้ายหรือหมุนหรือปรับอัตราส่วนของภาพ
xlink:href	เป็นการอ้างอิงถึง URL

ตารางที่ 2-2 SVG Attribute

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2 การแสดงภาพโดยใช้ภาษา SVG บนเบราว์เซอร์

การที่จะทำให้ เบราวเซอร์ เข้าใจ SVG DOM มีอยู่หลายวิธี แต่หนึ่งในวิธีที่นิยม และจะใช้ในการทำโครงการนี้ก็คือ การติดตั้ง plugin ให้แก่เบราว์เซอร์ โดยในโครงการนี้อ้างอิงกับ SVG plugin ของบริษัท Adobe โดยสามารถจะ ดาวน์โหลดได้ที่ <http://www.adobe.com/svg>

หลังจากที่เราทำการบันทึกเป็น File นามสกุล *.svg แล้ว การแทรก svg ลงในเอกสาร HTML สามารถทำแทรกได้โดยใช้ Tag ต่อไปนี้

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

2.4 จาวาสคริปต์ (JavaScript)

จาวาสคริปต์ เป็นภาษาสคริปต์ ที่สมบูรณ์ใกล้เคียงกับภาษาในการเขียน โปรแกรมปกติ จาวาสคริปต์ มีคุณสมบัติพื้นฐานต่างๆเช่นเดียวกับภาษาปกติ ไม่ว่าจะเป็นการกำหนดตัวแปร การกำหนดฟังก์ชัน การควบคุมการทำงานของ โปรแกรมสคริปต์ และชุดของโอเปอเรเตอร์ต่างๆนอกจากนี้จาวาสคริปต์ยังมีคุณสมบัติในการเขียน โปรแกรมแบบ อ็อบเจกต์บางอย่างไว้ด้วย ดังเช่น จาวาสคริปต์สามารถเข้าไปใช้อ็อบเจกต์ซึ่งปรากฏอยู่บนเว็บหรือจะแก้ไข Attribute รวมทั้งอีเวนต์ไว้ให้เรียกใช้งานด้วย

จาวาสคริปต์นั้นช่วยให้เพจ HTML สามารถทำงานได้ดั่งเช่นนักพัฒนาเว็บอาจจะสร้างให้เว็บเพจของตนสามารถตอบสนองต่อเหตุการณ์หรืออีเวนต์ของผู้ใช้ได้ไม่จำเป็นการคลิกเมาส์ หรือว่าการเรียกฟอร์มของ HTML ข้อดีของจาวาสคริปต์ที่สำคัญอีกประการหนึ่งคือ โครงสร้างของภาษาคำสั่งภาษาจาวาซึ่งง่ายต่อการเรียนรู้และใช้งาน

ความแตกต่างระหว่างภาษาจาวาและจาวาสคริปต์ที่เห็นได้ชัดเจนคือ จาวาสคริปต์มีโครงสร้างของภาษาที่ไม่ซับซ้อนเหมือนกับจาวา นักพัฒนาเว็บที่คุ้นเคยกับ Tag HTML และเคยใช้งานภาษาในการเขียนภาษาซี สามารถเรียนรู้และใช้งานภาษาสคริปต์ได้ในเวลาไม่นาน นอกจากนั้น โปรแกรมที่เขียนด้วยภาษาสคริปต์นั้นไม่จำเป็นที่จะต้องคอมไพล์เหมือน โปรแกรมจาวาทั่วไป ทั้งนี้เนื่องจากเป็นภาษาสคริปต์ โปรแกรมเว็บเบราว์เซอร์จึงมีตัวทำหน้าที่ปฏิบัติตามสคริปต์ที่เขียนไว้ทันที

ในการพัฒนาแผนที่โดยใช้ภาษา GML เราสามารถใช้ภาษาจาวาสคริปต์ในเข้าถึง Element หรือ Attribute ของ โครงสร้างภาษา SVG เพื่อสร้างส่วนของการปฏิสัมพันธ์กับผู้ใช้แผนที่เช่น การย่อหรือขยายแผนที่ การเลื่อนแผนที่หรือการนำข้อมูลรายละเอียดของสถานที่ต่างๆ ขึ้นมาแสดงโดยใช้ภาษา SVG

2.5 ระบบฐานข้อมูลเชิงพื้นที่ (Spatial database system)

ระบบฐานข้อมูลเชิงพื้นที่ (Spatial database system) เป็นระบบฐานข้อมูลที่ใช้ในการจัดการข้อมูลทางเรขาคณิต ข้อมูลที่เกี่ยวข้องกับภูมิศาสตร์และข้อมูลเชิงพื้นที่ โดยตัวอย่างของพื้นที่ที่เราสนใจอาจจะเป็น พื้นที่ 2 มิติต่างๆ, ส่วนของพื้นผิวโลก หรือตัวอย่างอื่นๆ เช่น ข้อมูลแสดงโครงสร้างของโมเลกุลของโปรตีน ซึ่งการนำวิธีการจัดการกับฐานข้อมูลโดยใช้วิธีจัดการกับระบบฐานข้อมูลในรูปแบบเก่าที่เคยมีอยู่เช่น การจัดการระบบฐานข้อมูลเชิงสัมพันธ์ (Relational database system) มาจัดการข้อมูลในรูปแบบที่ได้กล่าวมานั้นทำให้เกิดความไม่เหมาะสมเช่น

1. ข้อมูลในเชิงพื้นที่ (Spatial data) มีรูปแบบความสัมพันธ์ที่ซับซ้อนกว่าข้อมูลทางธุรกิจโดยทั่วไป ซึ่งทำให้โครงสร้างของฐานข้อมูลในรูปแบบเดิมไม่มีความเหมาะสมในการจัดการข้อมูล
2. การออกแบบโครงสร้างและการสร้างฐานข้อมูลโดยทั่วไปจะเป็นถูกดูแลโดยนักคอมพิวเตอร์ ในขณะที่ข้อมูลเชิงพื้นที่จะถูกกำหนดโดยนักภูมิศาสตร์ซึ่งโดยทั่วไปจะมีรูปแบบและวิธีการที่แตกต่างกัน

2.5.1 Model of spatial information

แอปพลิเคชันหลักที่มีส่วนในการผลักดันการวิจัยระบบฐานข้อมูลเชิงพื้นที่ คือเทคโนโลยีสำหรับ GIS (geographic information systems) ดังนั้นการใช้งานระบบฐานข้อมูลเชิงพื้นที่ส่วนใหญ่จะเกี่ยวข้องกับข้อมูลทางด้านภูมิศาสตร์ ซึ่งอาจอยู่ในรูปของข้อมูลในรูปแบบ 2 มิติหรือ 3 มิติ ซึ่งมีรูปแบบการจำลอง (Modeling) 2 ทางในการจำลองแบบจำลองของข้อมูลเชิงพื้นที่คือ

1. **Objects Base Model** ในการจำลองข้อมูลรูปแบบนี้เราให้ความสนใจในวัตถุหรือองค์ประกอบต่างๆที่อยู่ในพื้นที่นั้น และแต่ละวัตถุหรือองค์ประกอบจะมีข้อมูลที่บอกรายละเอียดทางเรขาคณิตของตัวเอง เช่น เมือง, ป่า, หรือ แม่น้ำ ซึ่งเป็นองค์ประกอบของประเทศเป็นต้น

ในการจำลองวัตถุหรือองค์ประกอบที่เราสนใจ โครงสร้างพื้นฐานที่ใช้ในการจำลองโดยทั่วไปมักจะเป็น จุด (point) เส้น (line) หรือ พื้นที่ (region) โดย จุด (point) จะใช้แสดงถึงวัตถุหรือองค์ประกอบที่เราต้องการแสดงเพียงที่ตั้งของมันบนพื้นที่ เส้น (line) โดยทั่วไปจะใช้แสดงถึงวัตถุหรือองค์ประกอบที่มีการเคลื่อนที่ผ่านพื้นที่ที่เราสนใจหรือการเชื่อมต่อบนพื้นที่ เช่น แม่น้ำ ถนน หรือการวางสายโทรศัพท์ พื้นที่ (region) จะใช้แสดงถึงวัตถุหรือองค์ประกอบที่มีข้อมูลซึ่งแสดงรายละเอียดทาง 2 มิติ เช่น พื้นที่ของ อุทยานแห่งชาติ ประเทศ หรือทะเลสาบ และอาจประกอบด้วยชิ้นส่วนย่อยหลายชิ้นมาเชื่อมต่อกัน ซึ่งโครงสร้างพื้นฐานที่ใช้ใน Object-Base Model แสดงได้ในรูปที่ 2-7



รูปที่ 2-7 โครงสร้างพื้นฐานที่ใช้ใน Object-Based Model

2. Field-Based Model ในการจำลองข้อมูลรูปแบบนี้เราให้ความสนใจในตัวพื้นที่นั้น หรือต้องการแสดงข้อมูลเกี่ยวกับตัวพื้นที่นั่นเอง เช่นข้อมูลการใช้พื้นที่ หรือการแบ่งเขตต่างๆ ของประเทศ ซึ่งตัวแทนในการแสดงความสัมพันธ์เชิงพื้นที่ที่เกี่ยวข้องกันของข้อมูลคือ พาร์ทิชัน (Partitions) และ เน็ตเวิร์ค (Network) โดยพาร์ทิชันจะใช้ในการแสดงถึงส่วนของพื้นที่ที่ต้องมีการเชื่อมต่อกันและส่วนที่ติดกันนั้นเป็นส่วนที่เราให้ความสนใจ และเน็ตเวิร์คจะเป็นกราฟที่ถูกฝังลงในพื้นที่ ประกอบด้วยเซตของจุดซึ่งแสดงตัวเป็น แต่ละโหนดของกราฟหรือเซตของเส้นซึ่งแสดงถึงขอบของพื้นที่ที่เราสนใจ

Partitions and networks



รูปที่ 2-8 โครงสร้างพื้นฐานที่ใช้ใน Field-Based Model

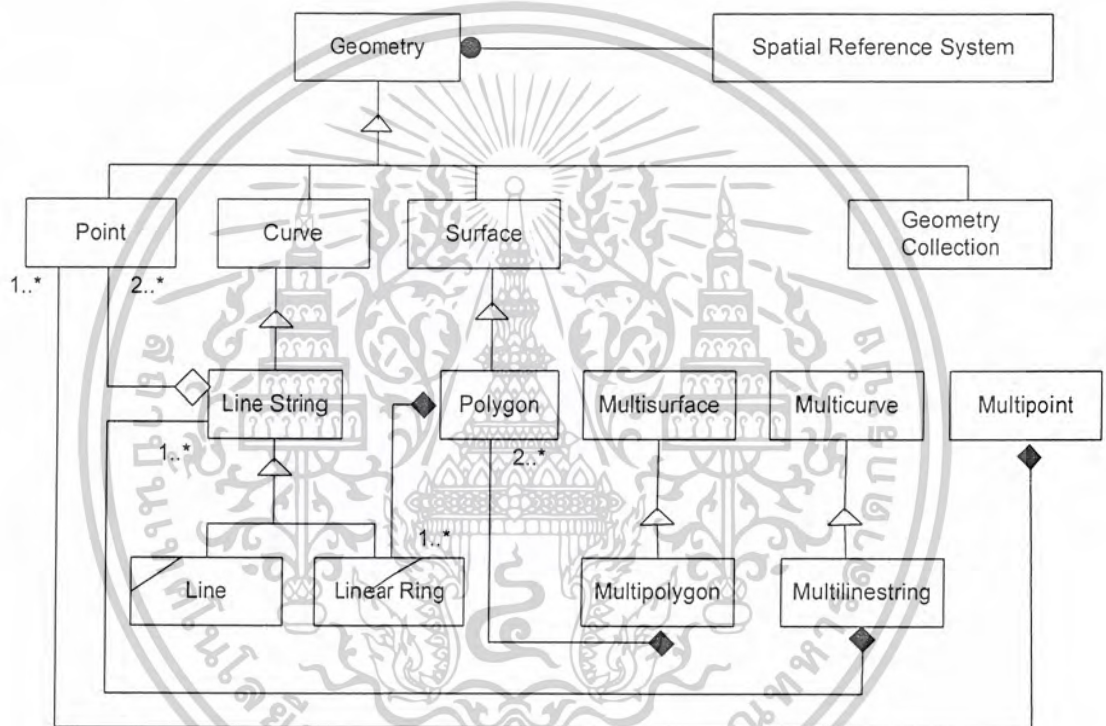
เราสามารถใช้งานการจำลองข้อมูลทั้งสองแบบรวมกันได้โดยการ จำลองโครงสร้างข้อมูลของวัตถุ และกำหนดความสัมพันธ์ในเชิงพื้นที่ที่เกี่ยวข้องกับวัตถุนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.2 Spatial Data Type

ระบบของชนิดของข้อมูลเชิงพื้นที่ (Spatial data Type or spatial algebras) คือระบบที่ใช้ในการรวมนิยามพื้นฐาน สำหรับ จุด (point), เส้น (line) และ พื้นที่ (region) เข้าด้วยกันกับความสัมพันธ์และวิธีการจัดการขององค์ประกอบนั้นๆ เช่น การกำหนดรูปแบบการตัดผ่านหรือซ้อนทับกัน (intersection) ของพื้นที่ เป็นต้น

ทางเลือกของชนิดของข้อมูลที่ใช้กันใน Objects-Base Model ของข้อมูลเชิงพื้นที่คือชนิดของข้อมูลพื้นฐานทั่วไปที่ใช้ในระบบแผนที่ ซึ่งในรูปที่ 2-9 แสดงถึง โครงสร้างและความสัมพันธ์ภายในของ spatial geometry ในรูปแบบของ UML



รูปที่ 2-9 โครงสร้างและความสัมพันธ์ภายในของ Spatial geometry ในรูปแบบของ UML

2.5.3 Spatial Relationships

ในกลุ่มของ Operation ที่นำเสนอโดย spatial algebras นั้นความสัมพันธ์เชิงพื้นที่ (Spatial relationship) ถือเป็นสิ่งที่สำคัญที่สุด เนื่องจากความสัมพันธ์เชิงพื้นที่ทำให้เราสามารถค้นหา อีอบเจกต์ที่เราต้องการจากความสัมพันธ์ที่เรากำหนดให้ โดยผ่านการ query โดยแบ่งความสัมพันธ์เป็น 3 แบบดังนี้

1. Topology Relationships เช่น อยู่ติดกัน (adjacent) หรือ อยู่ภายใน (inside)
2. Direction Relationships เช่น อยู่ด้านบน (above), อยู่ด้านล่าง (below) , อยู่ทางเหนือ (north_of) หรือทางใต้ของ (south_of)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. Metric Relationships เช่น หาอ็อบเจกต์ที่อยู่ห่างมากกว่า 100 กิโลเมตร เป็นต้น

2.5.4 ขั้นตอนการออกแบบฐานข้อมูลเชิงพื้นที่

ในการออกแบบฐานข้อมูลเชิงพื้นที่นั้นเราใช้หลักการของการออกแบบฐานข้อมูลโดยทั่วไปคือ ในขั้นตอนแรก จะเป็นการนำข้อมูลทั้งหมดที่เกี่ยวข้องกับแอปพลิเคชันนั้นมาจัดโดยใช้หลักการของ Conceptual data model ซึ่งในขั้นตอนนี้จะให้ความสนใจในการจัดแบ่งชนิดของข้อมูลของแอปพลิเคชัน, ความสัมพันธ์ และข้อกำหนดต่างๆของข้อมูล โดยจะแสดงรูปแบบความสัมพันธ์ของข้อมูลโดยผ่านแบบจำลองของฐานข้อมูลในแบบต่างๆ เช่น Entity Relationship (ER) Model ซึ่งเราใช้ในการออกแบบฐานข้อมูลของโครงการนี้

ในขั้นตอนที่สองจะเป็นขั้นตอนการพัฒนาและทำการจัดการกับข้อมูลต่างๆตามหลักของระบบจัดการฐานข้อมูล (Database management system)

และในขั้นตอนที่สามจะเป็นการออกแบบทางกายภาพ (Physical design) โดยเกี่ยวข้องกับการจัดการแอปพลิเคชันของฐานข้อมูล การทำอินเด็กซ์ และการจัดการหน่วยความจำของเครื่องคอมพิวเตอร์ที่ใช้ในการทำงานกับฐานข้อมูล เป็นต้น

2.6 การกำหนดตำแหน่งด้วยระบบจีพีเอส

ระบบบอกตำแหน่งบนพื้นดินหรือระบบจีพีเอสจะบอกตำแหน่งของผู้ใช้งาน โดยใช้เครื่องรับจีพีเอส อ่านตำแหน่งละติจูดลองจิจูดของทุกจุดบนพื้นโลก โดยรับสัญญาณจากดาวเทียมบอกตำแหน่งซึ่งทำการโคจรรอบโลกอย่างน้อย 4 ดวงจากทั้งหมด 24 ดวง ระบบจีพีเอสใช้ดาวเทียมนาฟสตาร์ (NAVSTAR) โดยการส่งคลื่นวิทยุจากดาวเทียมในอวกาศมายังภาคพื้นดิน และใช้ความต่างของเวลาในการรับส่งสัญญาณระหว่างดาวเทียมกับตัวรับสัญญาณ โดยการคำนวณหาตำแหน่ง, ความเร็ว และ เวลาให้กับผู้ใช้ ตามปรกติระบบจีพีเอสจะมีการใส่รหัสเพื่อให้เกิดความผิดพลาดเล็กน้อยทั้งนี้เนื่องจากระบบนี้ถูกพัฒนาขึ้นเพื่อให้กระทรวงกลาโหมของสหรัฐอเมริกา นำระบบนี้ไปใช้ในทางการทหาร แต่สัญญาณดาวเทียมที่ถูกส่งออกมานั้นสามารถถูกรับได้โดยผู้ใช้ทั่วไป ดังนั้นจึงมีการใส่รหัสเพื่อผู้ใช้ที่ได้รับอนุญาตเท่านั้นใช้งานได้ ในการใช้งานระบบจีพีเอสในโหมดมาตรฐานจะมีความเที่ยงตรงโดยเฉลี่ย 100 เมตร ในแนวอนและ 156 เมตรในแนวตั้งในงานที่ต้องการความเที่ยงตรงมากๆจึงต้องใช้ DGPS(Difference Global Positioning System) โดยรูปแบบของ DGPS จะประกอบด้วยเครื่องรับที่เรียกว่า "เครื่องรับอ้างอิง" ซึ่งทราบตำแหน่งที่ถูกต้องอย่างแท้จริงอยู่แล้ว ส่วนเครื่องรับจีพีเอสอื่นๆจะเป็นของผู้ใช้งาน โดยเครื่องรับอ้างอิงหรือสถานีอ้างอิงนี้จะนำสัญญาณดาวเทียมจีพีเอสมาคำนวณพิสัยของดาวเทียมแต่ละดวง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.1 มาตรฐานในการสื่อสารข้อมูล NMEA-0183

หน่วยงานที่ทำหน้าที่ในการกำหนดมาตรฐานในการสื่อสารข้อมูลจีพีเอส คือหน่วยงาน NMEA (National Marine Electronics Association) ได้กำหนดโปรโตคอล NMEA-183 เป็นโปรโตคอลมาตรฐานในการสื่อสารข้อมูลจากเครื่องรับจีพีเอสไปสู่อุปกรณ์เชื่อมต่อภายนอก ซึ่งรูปแบบเหล่านี้จะอยู่ในรูปของรหัสแอสกี (ASCII Codes) และสามารถเชื่อมต่อกับเครื่องคอมพิวเตอร์ได้โดยผ่านพอร์ตอนุกรมของเครื่องคอมพิวเตอร์ (RS-232) โดยกำหนดอัตราเร็วในการส่งข้อมูลอยู่ที่ 4800 บิตต่อวินาที โดยใช้การส่งข้อมูล 8 บิตแบบไม่มีพาริตี แต่มีบิตเริ่มต้น (Start bit) และบิตสิ้นสุด (Stop bit)

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

\$GPGGA,183805,3722.3622,N,12159.8274,W,2,03,02.8,16.6,M,20.2,M,5.800*XX

และมีความหมายของแต่ละเขตข้อมูลดังตารางที่ 2-3

NMEA GGA, Global Positioning System Fix Data

Header	Field	Description	Format
\$GPGGA	1	Time in GMT	HHMMSS.SS
	2	Latitude	IIII.IIII
	3	Latitude Hemisphere	'N' or 'S'
	4	Longitude	yyyyy.yyyy
	5	Longitude Hemisphere	'E' or 'W'
	6	Receiver Mode: 0=fix not available 1=non-differential 2=differential	'0', '1' or '2'
	7	Number of satellites used in position fix	xx
	8	HDOP	xx.x
	9	Altitude	+/-xxxxx.x
	10	Altitude Units: Always meters	'M'
	11	Geoidal Separation	+/-xxxx.x
	12	Separation Units: Always meters	'M'
	13	Age of differential corrections in seconds	xxx
	14	Differential reference station ID	xxxx
	15	NMEA Checksum	*hh

ตารางที่ 2-3 ความหมายของเขตข้อมูล NMEA GGA, Global Positioning System Fix Data

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบระบบให้บริการแผนที่

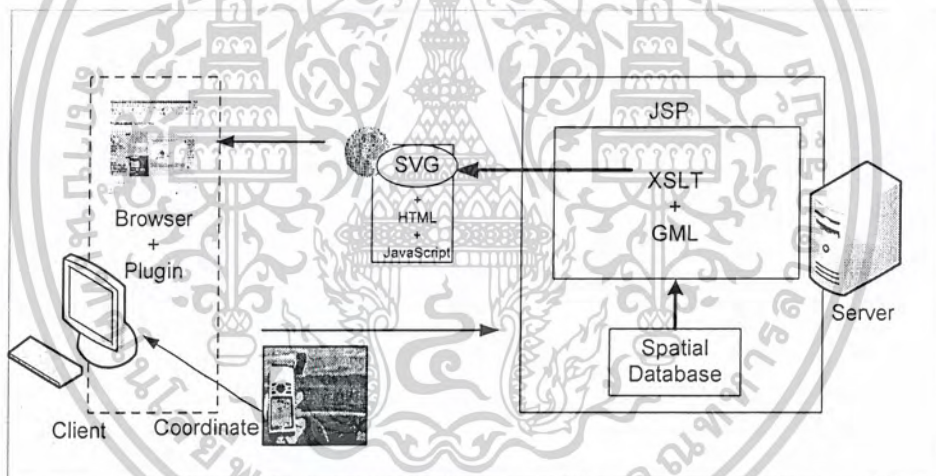
3.1 บทนำ

ระบบให้บริการแผนที่แบ่งออกเป็น 2 รูปแบบคือ

- 1.ระบบให้บริการแผนที่บนเว็บ
- 2.ระบบให้บริการแผนที่บนเครื่องพีซี

เนื้อหาในบทนี้จะกล่าวถึงการออกแบบของระบบให้บริการแผนที่บนเว็บ และในส่วนของ การออกแบบของระบบให้บริการแผนที่บนเครื่องพีซีจะกล่าวถึงในบทถัดไป

3.2 โครงสร้างและการทำงานของระบบให้บริการแผนที่บนเว็บ



รูปที่ 3-1 โครงสร้างของระบบให้บริการแผนที่บนเว็บ

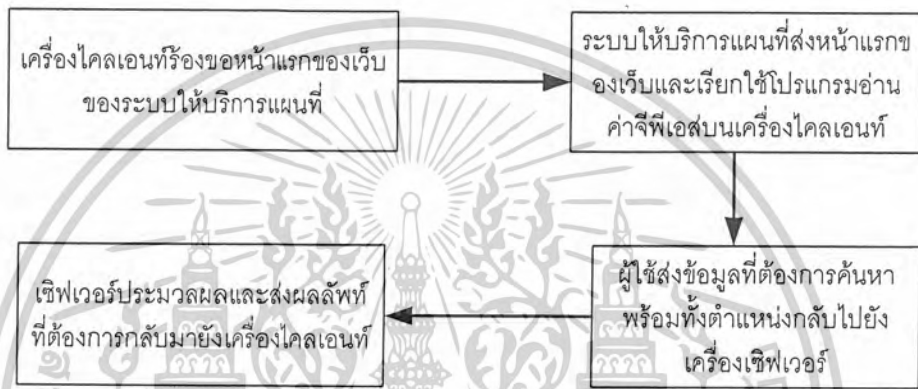
โครงสร้างของระบบให้บริการแผนที่บนเว็บแบ่งออกเป็น 2 ส่วนคือส่วนของเครื่องเซิร์ฟเวอร์และ ส่วนของเครื่องไคลเอนท์

1. ส่วนของเครื่องเซิร์ฟเวอร์ทำหน้าที่สร้างและประมวลผลแผนที่โดยนำข้อมูลแผนที่ซึ่งเก็บอยู่ในฐานข้อมูลเชิงพื้นที่ มาทำให้อยู่ในรูปแบบของภาษา GML และแปลงข้อมูลแผนที่ซึ่งอยู่ในรูปแบบของภาษา GML ให้เป็น ภาษา SVG โดยใช้ XSLT หลังจากนั้นจึงนำไฟล์ SVG ที่ได้ฝังลงในหน้าเว็บซึ่งเป็นภาษา HTML และ จาวาสคริปต์ส่งไปแสดงผลยังเครื่องไคลเอนท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ส่วนของเครื่องไคลเอนท์ ซึ่งจะเรียกใช้บริการของเซิร์ฟเวอร์ ได้โดยเรียกผ่านบราวเซอร์ที่ติดตั้ง plugin ในการดูไฟล์ซึ่งเป็นไฟล์ .SVG ซึ่งในโครงการนี้อ้างอิงกับ SVG plugin ของบริษัท Adobe โดยไคลเอนท์จะร้องขอใช้บริการจากเครื่องเซิร์ฟเวอร์โดยส่งพิกัดที่เครื่องไคลเอนท์อยู่และข้อมูลที่ต้องการร้องขอไปให้กับ เซิร์ฟเวอร์ ซึ่งพิกัดของเครื่องไคลเอนท์นั้นจะได้อ่านค่าข้อมูลของเครื่องรับสัญญาณ จีพีเอส (GPS receiver) ซึ่งต่ออยู่กับพอร์ตอนุกรมของเครื่องไคลเอนท์

โดยระบบให้บริการให้บริการแผนที่บนเว็บมีการทำงานของระบบซึ่งแสดงได้ดังรูปต่อไปนี้



รูปที่ 3-2 การทำงานของระบบให้บริการแผนที่บนเว็บ

3.3 การออกแบบระบบในส่วนของเซิร์ฟเวอร์

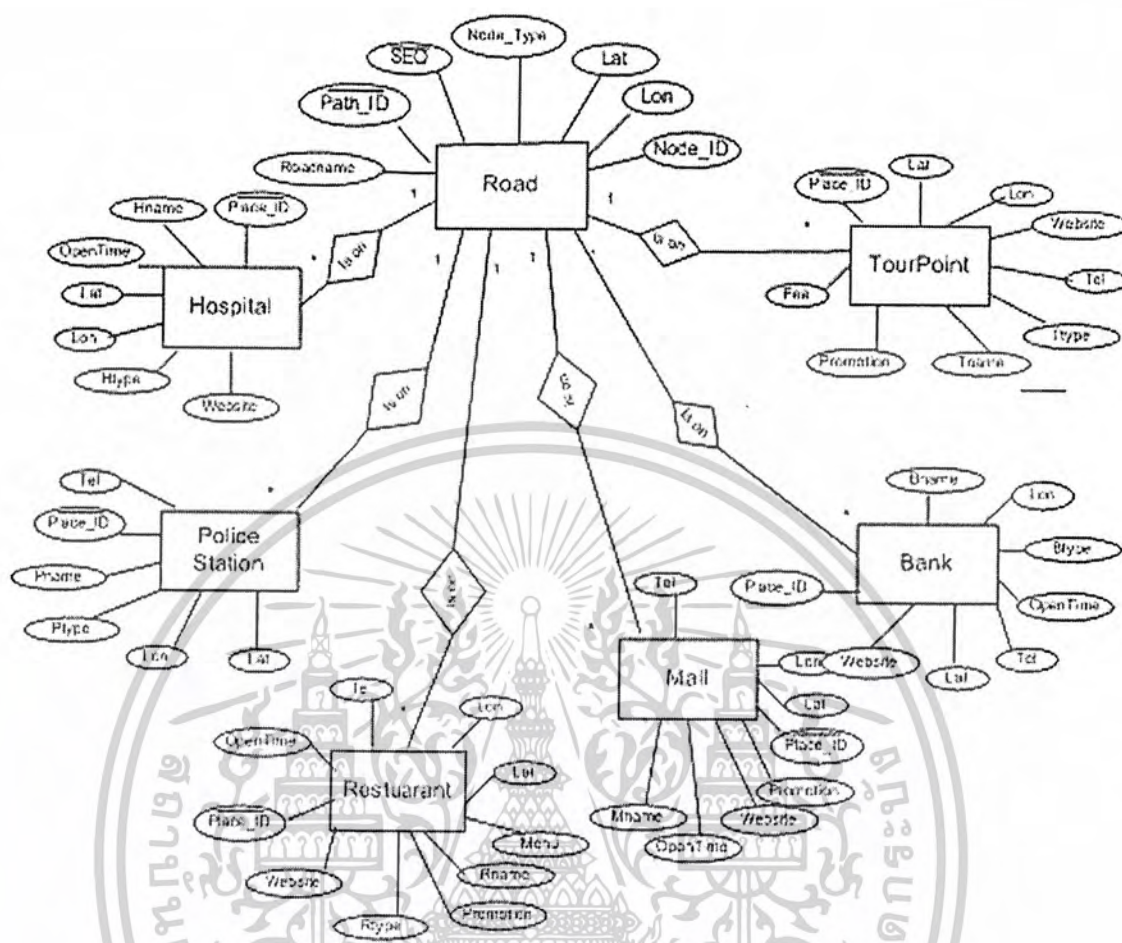
ในส่วนของเซิร์ฟเวอร์ประกอบด้วย

1. ฐานข้อมูลแผนที่
2. ส่วนของเว็บแอปพลิเคชันที่ทำการแสดงและประมวลผลแผนที่

3.3.1 การออกแบบฐานข้อมูลแผนที่

ฐานข้อมูลที่ใช้ในการเก็บข้อมูลรายละเอียดต่างๆของจังหวัดกรุงเทพมหานคร ในการพัฒนาระบบนี้ ผู้จัดทำได้เลือกใช้ไมโครซอฟท์เอสคิวแอลเซิร์ฟเวอร์2000 (MicrosoftSQLServer2000) เป็นระบบจัดการฐานข้อมูล (Database Management System: DBMS) และได้รับความอนุเคราะห์ข้อมูลสถานที่และถนนของกรุงเทพมหานครจากบริษัท ThaiMapGuide ซึ่งฐานข้อมูลเดิมนั้นมีไว้เพื่อใช้กับแอปพลิเคชันของทางบริษัทซึ่งยังไม่เหมาะสมกับระบบที่เราจะพัฒนาดังนั้นจึงมีการปรับรูปแบบของฐานข้อมูลให้เหมาะสมโดยได้เริ่มออกแบบจาก ER-Diagram ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่3-3 ER-Diagram ของระบบ

หลังจากได้ ER-Diagram แล้วเราจึงทำการ Normalization ซึ่งจะได้ตารางข้อมูลทั้งหมด 8 ตารางซึ่งมีรายละเอียดของตารางดังต่อไปนี้

1. ตาราง Path เป็นตารางที่เก็บข้อมูลเกี่ยวกับส่วนของถนน(Path) แต่ละเส้นว่าประกอบด้วยจุดใดบ้างและจุดนั้นมีพิกัดเป็นเท่าไรซึ่งประกอบไปด้วยเขตข้อมูล(Field) ดังต่อไปนี้

เขตข้อมูล (Field)	คำอธิบาย
Path_ID	หมายเลขอ้างอิงส่วนของถนน
SEQ	ลำดับของจุดพิกัดที่อยู่ในส่วนของถนนนั้น
Lat	ค่าละติจูดของพิกัดนั้น
Lon	ค่าลองจิจูดของพิกัดนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เขตข้อมูล (Field)	คำอธิบาย
Node_Type	ชนิดของพิกัดนั้น เป็น(Start/end/null)
Node_ID	หมายเลขอ้างอิงของพิกัดนั้น

ตารางที่ 3-1 เขตข้อมูลและคำอธิบายของตาราง Path

ซึ่งในตารางข้อมูลมีการกำหนดชนิดของข้อมูลในแต่ละเขตของข้อมูลและ Primary key ของตารางดังนี้

Column Name	Data Type	Length	Allow Nulls
Path_ID	int	4	
SEQ	char	10	
Lon	float	8	✓
Lat	float	8	✓
Node_Type	char	10	✓
Node_ID	int	4	✓

รูปที่ 3-4 ชนิดของข้อมูลและ Primary key ของตาราง path

2. ตาราง Road_Name เป็นตารางที่เก็บข้อมูลเกี่ยวกับหมายเลขที่ใช้ระบุส่วนของถนน (Path_ID) และชื่อถนนที่เกี่ยวข้องกับส่วนของถนนนั้น ใ้บอกว่า ส่วน (Path) ของถนนนั้น อยู่ในถนนเส้นใดประกอบไปด้วยเขตข้อมูลดังนี้

เขตข้อมูล (Field)	คำอธิบาย
Path_ID	หมายเลขอ้างอิงส่วนของถนน
Road_Name	ชื่อถนน

ตารางที่ 3-2 เขตข้อมูลและคำอธิบายของตาราง Road_Name

ซึ่งในตารางข้อมูลมีการกำหนดชนิดของข้อมูลในแต่ละเขตของข้อมูลและ Primary key ของตารางดังนี้

Column Name	Data Type	Length	Allow Nulls
Path_ID	int	4	
Road_Name	char	10	✓

รูปที่ 3-5 ชนิดของข้อมูลและ Primary key ของ Road_Name

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ตาราง Hospital เป็นตารางที่เก็บข้อมูลเกี่ยวกับสถานพยาบาลในกรุงเทพฯซึ่งจะแบ่งออกเป็นประเภทย่อยๆได้เป็นโรงพยาบาล และ คลินิกประกอบไปด้วยเขตข้อมูลต่างๆดังนี้

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

ตารางที่ 3-3 เขตข้อมูลและคำอธิบายของตาราง Hospital

ซึ่งในตารางข้อมูลมีการกำหนดชนิดของข้อมูลในแต่ละเขตของข้อมูลและ Primary key ของตารางดังนี้

Column Name	Data Type	Length	Allow Nulls
Place_ID	int	4	✓
Hospital_Name	nvarchar	100	✓
Hospital_Type	nvarchar	30	✓
Tel	decimal	9	✓
Website	nvarchar	50	✓
OpenTime	nvarchar	50	✓
Lat	float	8	✓
Lon	float	8	✓

รูปที่ 3-6 ชนิดของข้อมูลและ Primary key ของตาราง Hospital

4. ตาราง PoliceStation เป็นตารางที่เก็บข้อมูลเกี่ยวกับสถานีตำรวจในกรุงเทพฯซึ่งจะแบ่งออกเป็นประเภทย่อยๆคือสถานีตำรวจและป้อมตำรวจประกอบด้วยเขตข้อมูลต่างๆดังนี้

เขตข้อมูล (Field)	คำอธิบาย
Place_ID	หมายเลขอ้างอิงของสถานีตำรวจ
Police_Name	ชื่อสถานีตำรวจ
Police_Type	ประเภทของสถานีตำรวจ(สถานีตำรวจ/ป้อมตำรวจ)
Tel	หมายเลขโทรศัพท์ของสถานีตำรวจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เขตข้อมูล (Field)	คำอธิบาย
Lat	ละติจูด
Lon	ลองจิจูด

ตารางที่ 3-4 เขตข้อมูลและคำอธิบายของตาราง Police

ซึ่งในตารางข้อมูลมีการกำหนดชนิดของข้อมูลในแต่ละเขตของข้อมูลและ Primary key ของตาราง ดังนี้

Column Name	Data Type	Length	Allow Nuls
Place_ID	int	4	
Police_Name	nvarchar	255	✓
Police_Type	char	20	✓
Tel	decimal	9	✓
Lat	float	8	✓
Lon	float	8	✓

รูปที่ 3-7 ชนิดของข้อมูลและ Primary key ของตาราง Police

5. ตาราง Bank เป็นตารางที่เก็บข้อมูลเกี่ยวกับธนาคารในกรุงเทพฯ ซึ่งจะแบ่งออกเป็นประเภทย่อยคือ ธนาคาร และ ตู้บริการเงินด่วน (ATM) ประกอบไปด้วยเขตข้อมูลต่างๆดังนี้

เขตข้อมูล (Field)	คำอธิบาย
Place_ID	หมายเลขอ้างอิงของธนาคารหรือตู้บริการเงินด่วน
Bank_Name	ชื่อธนาคารหรือตู้บริการเงินด่วน
Bank_Type	ชนิด(ธนาคาร/ตู้บริการเงินด่วน)
Tel	หมายเลขโทรศัพท์ของธนาคาร
Website	URL ของเว็บของธนาคาร
OpenTime	เวลาทำการ
Lat	ละติจูด
Lon	ลองจิจูด

ตารางที่ 3-5 เขตข้อมูลและคำอธิบายของตาราง Bank

ซึ่งในตารางข้อมูลมีการกำหนดชนิดของข้อมูลในแต่ละเขตของข้อมูลและ Primary key ของตาราง ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Column Name	Data Type	Length	Allow Nulls
Place_ID	int	4	
Bank_Name	nvarchar	100	✓
Bank_Type	char	10	✓
Tel	decimal	9	✓
Website	nvarchar	50	✓
OpenTime	nvarchar	50	✓
LAT	float	8	✓
LON	float	8	✓

รูปที่ 3-8 ชนิดของข้อมูลและ Primary key ของตาราง Bank

6. ตาราง Mall เป็นตารางที่เก็บข้อมูลเกี่ยวกับห้างสรรพสินค้าในกรุงเทพ ประกอบด้วยเขตข้อมูลดังนี้

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

ตารางที่ 3-6 เขตข้อมูลและคำอธิบายของตาราง Mall

ซึ่งในตารางข้อมูลมีการกำหนดชนิดของข้อมูลในแต่ละเขตของข้อมูลและ Primary key ของตาราง ดังนี้

Column Name	Data Type	Length	Allow Nulls
Place_ID	int	4	
Mall_Name	nvarchar	255	✓
OpenTime	nchar	20	✓
Tel	decimal	9	✓
Website	nvarchar	50	✓
Promotion	nvarchar	255	✓
LAT	float	8	✓
LON	float	8	✓

รูปที่ 3-9 ชนิดของข้อมูลและ Primary key ของตาราง Mall

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. ตาราง Restaurant เป็นตารางที่เก็บข้อมูลเกี่ยวกับร้านอาหารในกรุงเทพ ประกอบไปด้วยเขตข้อมูลดังนี้

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

ตารางที่ 3-7 เขตข้อมูลและคำอธิบายของตาราง Restaurant

ซึ่งในตารางข้อมูลมีการกำหนดชนิดของข้อมูลในแต่ละเขตของข้อมูลและ Primary key ของตารางดังนี้

Column Name	Date Type	Length	Allow Nulls
Place_ID	int	4	✓
Restuarant_Name	nvarchar	100	✓
Restuarant_Type	char	10	✓
OpenTime	nvarchar	50	✓
Tel	decimal	9	✓
Website	nvarchar	50	✓
Promotion	nvarchar	255	✓
Menu	nvarchar	255	✓
Lat	float	8	✓
Lon	float	8	✓

รูปที่ 3-10 ชนิดของข้อมูลและ Primary key ของตาราง Restaurant

8. ตาราง Tour point เป็นตารางที่เก็บข้อมูลเกี่ยวกับสถานที่ท่องเที่ยวในกรุงเทพ ประกอบไปด้วยเขตข้อมูลดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เขตข้อมูล (Field)	คำอธิบาย
Planeside	หมายเลขอ้างอิงของสถานที่ท่องเที่ยว
Restaurant_Name	ชื่อสถานที่ท่องเที่ยว
Restaurant_Type	ชนิดของสถานที่ท่องเที่ยว
Tel	หมายเลขโทรศัพท์ของสถานที่ท่องเที่ยว
Website	URL ของเว็บของสถานที่ท่องเที่ยว
OpenTime	เวลาทำการ
Promotion	รายการส่งเสริมการขายของสถานที่ท่องเที่ยว
Fee	ราคาค่าบริการของสถานที่ท่องเที่ยว
Lat	ละติจูด
Lon	ลองจิจูด

ตารางที่ 3-8 เขตข้อมูลและคำอธิบายของตาราง TourPoint

ซึ่งในตารางข้อมูลมีการกำหนดชนิดของข้อมูลในแต่ละเขตของข้อมูลและ Primary key ของตาราง

ดังนี้

Column Name	Data Type	Length	Allow Nulls
Place_ID	int	4	✓
Tour_Name	nvarchar	50	✓
Tour_Type	char	10	✓
Tel	decimal	9	✓
Website	nvarchar	50	✓
OpenTime	nvarchar	50	✓
Promoton	nvarchar	50	✓
Fee	int	4	✓
Lat	float	8	✓
Lon	float	8	✓

รูปที่ 3-11 ชนิดของข้อมูลและ Primary key ของตาราง TourPoint

3.3.2 การพัฒนาแผนที่โดยใช้ภาษา GML

เมื่อทำการออกแบบฐานข้อมูลให้มีความเหมาะสมกับการใช้งานของระบบแล้ว เราจะนำข้อมูลจากฐานข้อมูลที่มีอยู่นั้นมาทำให้อยู่ในรูปแบบของโครงสร้างของภาษา GML ซึ่งแบ่งออกเป็นขั้นตอนต่างๆ ดังนี้

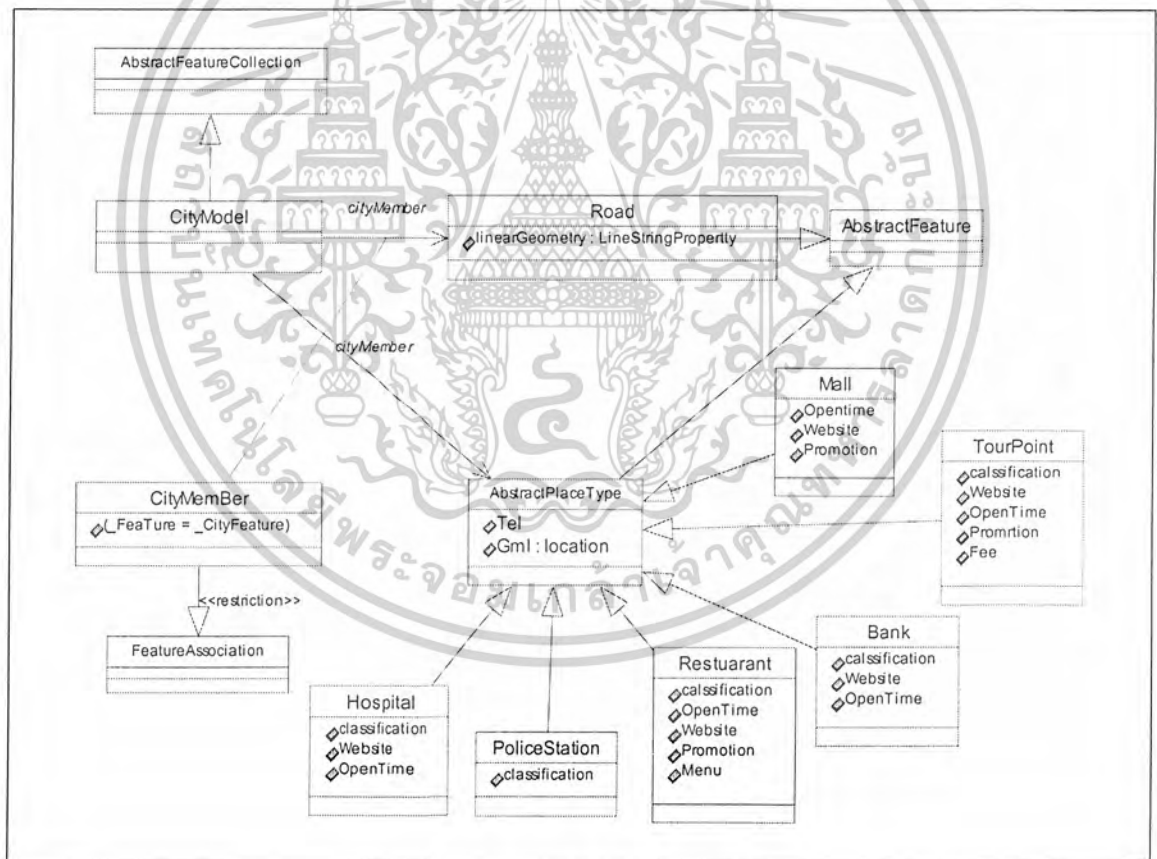
1. การสร้าง GML Application Schema

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.การสร้างไฟล์ GML ซึ่งอยู่ในรูปของไฟล์ .XML

3.3.2.1 การสร้าง GML Application Schema

สเกิมา(Schema) คือ การรูปแบบการจัด โครงสร้างและความสัมพันธ์ของข้อมูล โดยในภาษา GML จะประกอบด้วย สเกิมาพื้นฐานของ XML คือ Feature.xsd ซึ่งจะให้ข้อกำหนดของคุณสมบัติและโครงสร้างของ Feature พื้นฐาน (feature-property model) geometry.xsd ซึ่งจะกล่าวถึงองค์ประกอบทางเรขาคณิต (Geometry Component) และ xlink.xsd กำหนด Attribute ของ Xlink ซึ่งใช้ในการเชื่อมโยง (link) ฟังก์ชันต่างๆของเอกสารเข้าด้วยกัน แต่สเกิมาที่มีอยู่นี้ยังไม่เพียงพอเนื่องจากสเกิมาเหล่านี้จะกำหนดเฉพาะ Feature พื้นฐาน สำหรับการใช้งาน GML เท่านั้นดังนั้นการที่จะทำให้แผนที่ที่เราจะสร้างมีโครงสร้างและความสัมพันธ์ขององค์ประกอบต่างๆเป็นไปตามที่เราต้องการ เราจึงต้องทำการกำหนด GML application schema เพิ่มเติมขึ้นมา ซึ่งเขียนโดยใช้หลักการของ XML Schema และเป็นไปตามข้อบังคับของ GML ซึ่งสเกิมาที่ใช้ในกำหนดโครงสร้างของข้อมูล GML ที่เราใช้มีโครงสร้างดังนี้



รูปที่ 3-12 Model ของ Bangkok.xsd

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำหนดให้ <CityModel> อติเมตต์เป็น Instance ของชนิดข้อมูลที่เรากำหนดขึ้นมาเองและได้รับการถ่ายทอดโดยขยายมาจาก gml: AbstractFeatureCollectionType โดยมีสมาชิกของ CityModel เพียง 2 Feature คือ Road และ Place โดยชนิดข้อมูล PlaceType นั้นเรากำหนดให้เป็นข้อมูลชนิด Abstract DataType สามารถใช้งานได้โดยการถ่ายทอดไปใช้เท่านั้นและได้รับการถ่ายทอดไปเป็น HospitalType, PoliceStationType, BankType, ReatuarantType, TourPointType และ MallType

ซึ่งจากโมเดลที่เราออกแบบไว้นำมาเขียนเป็น สกิมมาได้ดังต่อไปนี้

```
<?xml version="1.0" encoding="UTF-16"?>
<!--File: Bangkok.xsd -->
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xlinks="http://www.w3.org/1999/xlink"
xmlns:gml="http://www.opengis.net/gml" elementFormDefault="qualified">
<xs:import namespace="http://www.opengis.net/gml" schemaLocation="feature.xsd"/>
<xs:import namespace="http://www.opengis.net/gml" schemaLocation="geometry.xsd"/>
<xs:import namespace="http://www.w3.org/1999/xlink" schemaLocation="xlinks.xsd"/>
```

ในส่วนแรกนั้นเป็นการประกาศสกีมาตามหลักของ XML Schema และกำหนดเนมสเปซ (Namespace) ต่างๆที่ใช้อ้างอิงในการสร้างสกีมา โดยเนมสเปซที่ใช้ระบุสกีมาพื้นฐานที่ใช้ในการสร้างสกีมา คือ

```
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

ส่วนเนมสเปซอื่นๆเป็นเนมสเปซที่ใช้เพิ่มเติมในการสร้าง GML Application Schema

```
<!--=====
global element declaration
=====-->
<xs:element name="CityModel" type="CityModelType"
substitutionGroup="gml:_FeatureCollection"/>
<xs:element name="cityMember" type="CityMemberType" substitutionGroup="gml:featureMember"/>
<xs:element name="AbstractPlace" type="AbstractPlaceType" abstract="true"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

substitutionGroup="_CityFeature"/>
<xs:element name="_CityFeature" type="gml:AbstractFeatureType" abstract="true"
substitutionGroup="gml:_Feature"/>
<xs:element name="Hospital" type="HospitalType" substitutionGroup="_CityFeature"/>

```

ส่วนถัดมาเป็นตัวอย่างการประกาศ Global Element ให้เป็นไปตามที่เรากำหนดไว้จากตัวอย่างจะเห็นว่ามีการประกาศให้ Element AbstractPlace ให้ มีชนิดเป็น AbstractPlaceType และกำหนดให้ Element นี้เป็น abstract โดยการกำหนด abstract="true" ซึ่งหมายความว่า Element นี้ไม่สามารถนำไปใช้ได้โดยตรงแต่จะสามารถทำการขยายเป็น Element อื่นเพื่อนำไปใช้ได้ซึ่งเราได้ขยายไปเป็น Element Hospital นั่นเอง

```

-----
type definition for Bangkok model
----->
<xs:complexType name="AbstractPlaceType" abstract="true">
  <xs:annotation>
    <xs:documentation>All Place elements are derived from this abstract supertype; a Place element
    may have an identifying attribute (TEL, Location). It may be associated with a spatial reference
    system.</xs:documentation>
  </xs:annotation>
  <xs:complexContent>
    <xs:extension base="gml:AbstractFeatureType">
      <xs:sequence minOccurs="0">
        <xs:element name="Location" type="PointPropertyType" minOccurs="0"/>
        <xs:element name="Tel" type="xs:string" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
</xs:schema>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนสุดท้ายเป็นการกำหนดชนิดของ Element ว่าในแต่ละ Element ที่เรากำหนดขึ้นมานั้นจะประกอบด้วยโครงสร้างภายในเป็นอย่างไร มี Element ย่อยภายในอะไรบ้าง และแต่ละ Element นั้นมีชนิดของข้อมูลเป็นแบบใด โดยข้อมูลสกีมาฉบับเต็มที่ใช้ในการทำงานจริงสามารถดูได้จากภาคผนวก ข

3.3.2.2 การสร้างไฟล์ GML

หลังจากกำหนดโครงสร้างของข้อมูล GML เรียบร้อยแล้วในส่วนถัดมาคือการนำข้อมูลจากฐานข้อมูลขึ้นมาใส่ใน Tag ของ GML ตามโครงสร้างที่เราได้กำหนดขึ้นโดยทำการเขียนโปรแกรมอ่านข้อมูลจากฐานข้อมูลที่มีที่ละตาราง และสั่งให้เขียนข้อมูลออกไปยังไฟล์ซึ่งมีนามสกุลของไฟล์เป็น .xml โดยในที่นี้กำหนดให้เป็น ไฟล์ชื่อ Bangkok.xml ตัวอย่างบางส่วนของไฟล์ GML ที่ใช้ในการทำงานจริง

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

ในส่วนแรกเป็นการประกาศ นามสเปซต่างๆที่เกี่ยวข้องในการเขียน ไฟล์ GML โดยตัวอักษรที่เป็นตัวหนาเป็นการระบุถึง GML Application Schema ที่เราเขียนขึ้น โดยเป็นการระบุว่า Application Schema ที่เราอ้างถึงนี้ไม่ได้ระบุ นามสเปซ และมีตำแหน่งที่เก็บไฟล์คือ `c:/tomcat-4.1.18/webapps/ROOT/project/Bangkok.xsd` สังเกตว่า Element ที่เป็น Element Root ของไฟล์นี้คือ `<CityModel>` ตามที่เรากำหนดไว้ใน Application Schema

```
<cityMember>
  <Hospital fid='H2' Classified='1'>
    <gml:name>รพ ราชวิถี </gml:name>
    <Location>
      <gml:Point srsName='http://www.opengis.net/gml/srs/epsg.xml#4326'>
        <gml:coordinates>13.7649,100.537</gml:coordinates>
      </gml:Point>
    </Location>
  </Hospital>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

</cityMember>
--
--
</CityModel>

```

ส่วนถัดมาเป็นการนำข้อมูลจากฐานข้อมูลขึ้นมาทำให้อยู่ในรูปของภาษา GML โดยดึงขึ้นมาทีละเขตของข้อมูลและใส่ข้อมูลลงไปตามลำดับของ Element ที่เรากำหนดไว้ ซึ่งในที่นี้จะพบว่า ข้อมูลของโรงพยาบาลราชวิถีที่อยู่ใน ตาราง Hospital นั้น มีข้อมูลเพียง 4 column คือ Place_ID, Hospital_Name, Lat, Lon ซึ่งเรานำมาเป็นข้อมูลใน Attribute fid และเป็นข้อมูลใน Element <gml:name> และ <Location> ซึ่ง ใน Element <Location> นั้น เราได้ประกาศ Element ย่อยภายในไว้ เป็น <gml:point> ซึ่งมี <gml:coordinate> เป็น Element ย่อยภายในสำหรับเก็บค่าละติจูดกับลองจิจูดซึ่งเรานำขึ้นมาจากเขตของข้อมูล Lat, Lon ในตาราง

ในการนำเขตของข้อมูลในแต่ละตารางมาใส่ใน Element ของ GML แสดงได้ดังตาราง

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

ตาราง 3-9 ความสัมพันธ์ของเขตของข้อมูลจากตาราง Hospital และ GML

3.3.3 การนำแผนที่ในโครงสร้างของภาษา GML ไปแสดงผลโดยใช้ภาษา SVG

ในการจะแสดงผลแผนที่ซึ่งอยู่ในรูปแบบของภาษา GML นั้นจะต้องทำการแปลงข้อมูลซึ่งอยู่ในรูปแบบของ GML ไปเป็นภาษา SVG เพื่อแสดงผลโดยใช้ XSLT Stylesheet การแปลงนั้นจะใช้ XSLT ในการเข้าถึงแต่ละ Element ของ GML และให้ผลลัพธ์ออกมาเป็น Tag ของ SVG เช่น

```
<cityMember>
```

```
<Restaurant fid='R15225'>
```

```
<gml: name>โรงพยาบาลพระราม 2</gml: name>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<Location>
  <gml:Point srsName='http://www.opengis.net/gml/srs/epsg.xml#4362'>
    <gml:coordinates>13.7349, 100.538</gml:coordinates>
  </gml:Point>
</Location>
</Restaurant>
</cityMember>

```

เมื่อ XSLT เข้าถึง Element ของ <gml: name>โรงพยาบาลพระราม 2</gml:name> XSLT จะให้ผลลัพธ์เป็น

```

<text x="13.7349" y="100.538">โรงพยาบาลพระราม 2</text>
<circle cx="13.7349" cy="100.538" r="10" fill="yellow"/>

```

ซึ่งจะเป็นการเขียนอักษรลงบนแผนที่ ในการแปลงนี้จะเพิ่มรูปร่างกลมเข้าไปเพื่อให้ทราบตำแหน่งของจุดนั้นๆ ได้ชัดเจนยิ่งขึ้น

```

ในส่วนของถนนนั้นจะใช้ Element <path> แทน Element <circle> และใช้ <textPath> เข้ามาช่วย <text> เช่น
เขตของข้อมูล<path id="123" d="M0,300 L40,260 L44,255 L48,250 L53,240 L57,230 L59,220 L59,0 "
style="fill:none;stroke:black;stroke-width:1;"/>
<text>
  <textPath xlink:href="#123" >ถนนลาดพร้าว</textPath>
</text>

```

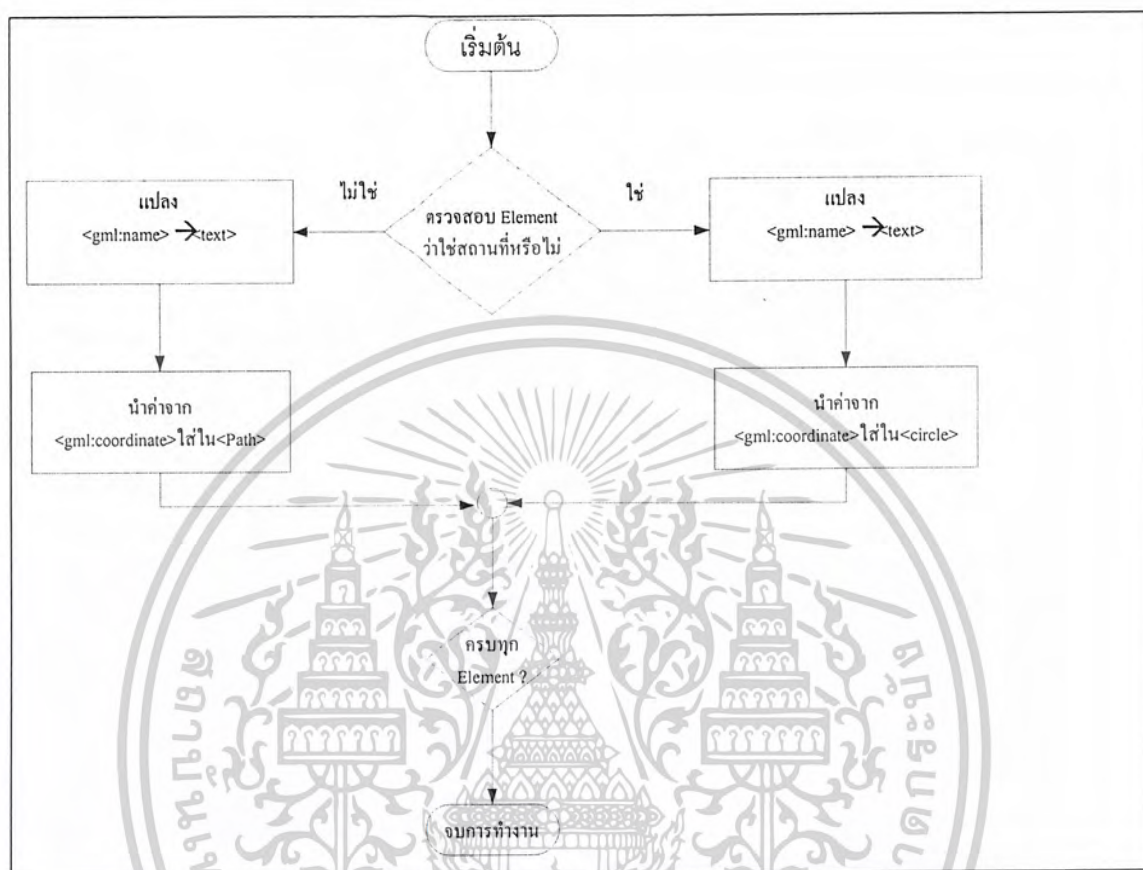
โดยความสัมพันธ์ระหว่าง Element ของ GML และ SVG เป็นดังตาราง

GML	SVG
<gml:coordinates>(1 จุด)	<line>, <path> กำหนดให้จุดเริ่มต้นและจุดปลายเป็นจุดเดียวกัน
<gml:coordinates>	<path>
<gml:name>	<text>
<gml:description>	<desc>

ตารางที่ 3-10 ความสัมพันธ์ระหว่าง Element ของ GML และ SVG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนในการแปลงไฟล์ GML ไปเป็น SVG แสดงได้ดังรูป



รูปที่ 3-13 ขั้นตอนในการแปลงไฟล์ GML ไปเป็น SVG

และ stylesheet ที่ใช้จริงในโครงการนี้สามารถศึกษาได้จากภาคผนวก ก

3.3.3.1 การเพิ่มความสามารถของแผนที่โดยใช้ภาษา SVG DOM

แผนที่ SVG ที่สร้างขึ้นมาจากการแปลง ภาษา GML นั้นสร้างได้เพียงภาพแผนที่ซึ่งเป็นภาพนิ่งเท่านั้น ซึ่งในการพัฒนาเราสามารถใช้ความสามารถของภาษา SVG ในการ เพิ่มความสามารถของแผนที่ ซึ่ง ทำได้ โดยการใช้ภาษาจาวาสคริปต์ร่วมกับ SVG DOM เพื่อเพิ่มความสามารถในการปฏิสัมพันธ์กับผู้ใช้ เช่น

1. การแสดงรายละเอียดต่างๆของสถานที่ โดยแสดงเมื่อผู้ใช้นำเมาส์มาที่ตำแหน่งของสถานที่นั้นๆ ในที่นี้เราเขียน โดยใช้ชื่อฟังก์ชันว่า showinfo โดยมีตัวอย่างของ code คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//=====
                                showInfo                                =
//=====

function showInfo(evt, info) {
    elementTarget = get_target(evt);
    var objText = SVGDoc.getElementById("info");
    var objBox = SVGDoc.getElementById("infoBoxRect");

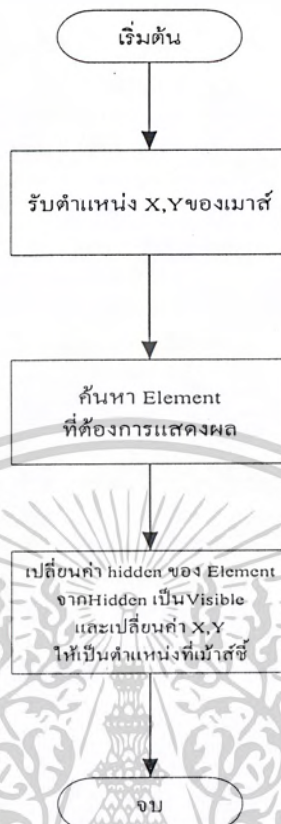
    objText.firstChild.nodeValue = info;
    objText.setAttribute("x", x + 20); //x + 5 + 15
    objText.setAttribute("y", y - 3); //y + 12 - 15
    //objText.getStyle().setProperty("visibility", "visible",""); //batik
    objText.setAttribute("visibility", "visible");

    objBox.setAttribute("x", x + 15); // x +15
    objBox.setAttribute("y", y - 15); // y - 15
    //objBox.getStyle().setProperty("visibility", "visible",""); //batik
    objBox.setAttribute("visibility", "visible");
}

```

ขั้นเราสามารถอธิบายขั้นตอนการทำงานของฟังก์ชัน showinfo ได้ดังไดอะแกรมในหน้าถัดไป

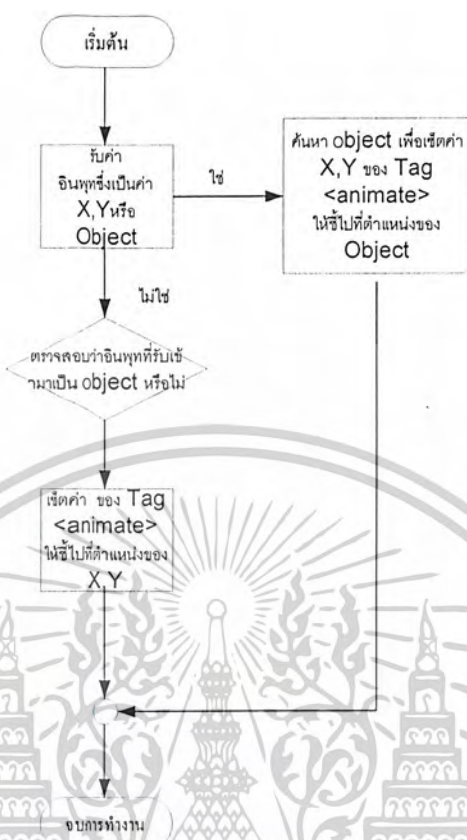
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-14 ขั้นตอนการทำงานของฟังก์ชัน showInfo

2. ภาพเคลื่อนไหวหรือกระพริบแสดงจุดบนแผนที่ ซึ่งสามารถทำได้โดยการใช้ Element `<animate>` โดยทำการเซตที่ค่า Object นั้น ซึ่งขั้นตอนการทำงานของฟังก์ชันอธิบายได้ดังไดอะแกรมต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-15 ขั้นตอนการทำงานของฟังก์ชัน Animate

3.3.4 การออกแบบเว็บแอปพลิเคชัน

ในส่วนของเว็บแอปพลิเคชันเป็นส่วนที่ทำหน้าที่ในการติดต่อกับผู้ใช้ และทำการประมวลผลการร้องขอสถานะที่จากผู้ใช้ และแสดงผลที่ให้กับผู้ใช้ ในการพัฒนาในส่วนเว็บแอปพลิเคชันโปรแกรม โดยออกแบบตามหลักของ Three-Tier Systems ซึ่งแบ่งเป็น 3 ส่วน คือ ส่วนแสดงผล, ส่วนประมวลผล และส่วนติดต่อกับฐานข้อมูล

3.3.4.1 ส่วนติดต่อกับฐานข้อมูล

ออกแบบโดยใช้ JavaBean มาช่วยในการพัฒนา ซึ่งกำหนดให้ 1 ตารางของฐานข้อมูล จะมี 1 คลาส ที่ทำหน้าที่ติดต่อกับตารางนั้น ด้วยการส่ง SQL Statement และมี 1 คลาส ที่ทำหน้าที่ระบุในส่วนของที่อยู่, ชื่อผู้ใช้, รหัสผ่าน ของฐานข้อมูล และจะให้คลาสอื่น เรียกใช้ ทำให้เมื่อมีการเปลี่ยนแปลงเกี่ยวกับฐานข้อมูล จึงแก้ไข และคอมไพล์ ไฟล์นี้ ไฟล์เดียวกันนั้น ไม่ต้องแก้ไขทั้งหมด และทำการนำเอาคลาสทั้งหมดนี้ ไว้ใน แพคเกจ ที่ชื่อ database เพื่อความเป็นระเบียบในการพัฒนาและนำไปใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดสำคัญของการเขียนโปรแกรมในการติดต่อฐานข้อมูล โดยใช้ภาษาจาวา มีดังต่อไปนี้

1. กำหนดชนิดของไดร์เวอร์

```
String driver = "com.microsoft.jdbc.sqlserver.SQLServerDriver";
Class.forName (driver);
```

2. สร้างการเชื่อมต่อไปยังฐานข้อมูล

```
String url = "jdbc:microsoft:sqlserver://161.246.5.69";
String user = "sa";
String password = "1817";
Connection connection = DriverManager.getConnection (url, user, password);
```

3. สร้างสเตตเมนต์

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

4. ประมวลผล เอสคิวแอล และจัดการกับข้อมูล

```
select_hospital="SELECT * FROM MAP.dbo.HOSPITAL WHERE Hospital_Type='HOSPITAL'";
ResultSet rs = statement.executeQuery(sql);
rs.next();
rs.getString("Place_ID");
```

ข้อมูลที่ได้ขึ้นมาจะเป็นอักขระที่เป็นแบบ ASCII ทำให้ถ้ามีอักขระที่ไม่ใช่ภาษาอังกฤษ อาจทำให้มีปัญหาได้ จึงต้องทำการเข้ารหัสให้ข้อมูลดังกล่าวเป็น Unicode ก่อนนำไปใช้

```
String (rs.getString("Place_ID")).getBytes("ISO8859_1").toString();
```

5. ปิดการเชื่อมต่อ

```
statement.close();
connection.close();
```

โดยนำเอาส่วนต่างมาเขียนให้เป็น เมธอด และคลาส เพื่อสะดวกในการทำงาน ลดความซ้ำซ้อนที่จะเกิดขึ้นในการเขียน โปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.4.2 ส่วนประมวลผล

เป็นส่วนสำคัญที่สุดของระบบ เพราะจะทำหน้าที่นำข้อมูลในส่วนต่างๆ มาประมวลผล ทั้งในของข้อมูลสถานที่นำมาจากฐานข้อมูล ส่วนของพิกัดและข้อมูลสถานที่ที่ต้องการค้นหาที่นำมาจากผู้ใช้ โดยแบ่งตามหน้าที่การทำงานคือ

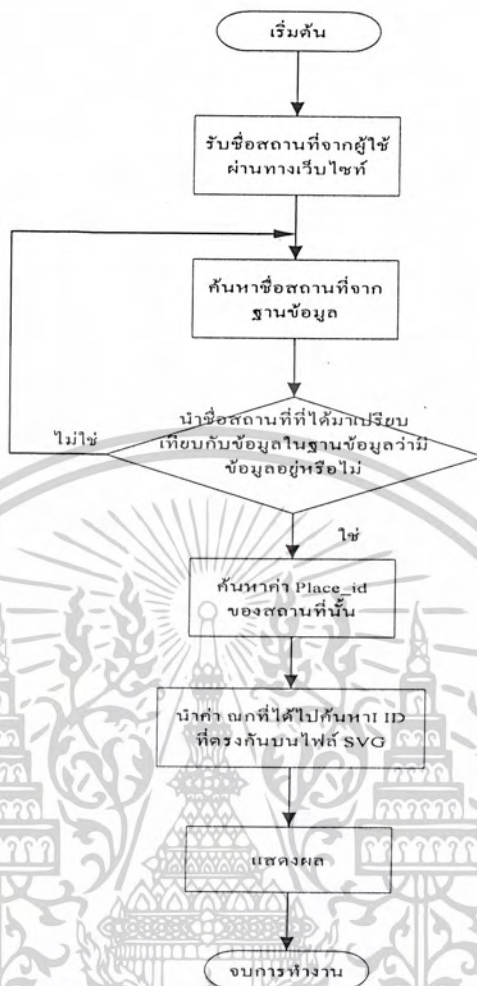
1.การค้นหาสถานที่ จะเป็นการค้นหาสถานที่โดยให้ผู้ใช้เลือกประเภทของสถานที่ที่ต้องการค้นหา และระบุชื่อของสถานที่นั้นซึ่งแผนที่จะแสดงตำแหน่งและข้อมูลของสถานที่นั้น โดยมีหลักการทำงานคือ เมื่อผู้ใช้กรอกชื่อสถานที่ที่ต้องการค้นหาและกดปุ่ม SEARCH โปรแกรมก็จะทำการนำชื่อสถานที่ไปค้นหา ID ของสถานที่นั้นและนำไปเขียนต่อท้ายไฟล์ จากนั้นจาวาสคริปต์ก็จะนำ ID นั้นไปค้นหา tag ของสถานที่นั้น เพื่อที่จะหาค่าละติจูดและลองจิจูด ซึ่งภาษาจาวาสคริปต์ที่ใช้ในการค้นหา tag ของสถานที่คือ

```
SVGDoc.getElementById("ID ของสถานที่");
```

จากนั้นจะเลื่อนหน้าแสดงผลไปยังตำแหน่งของสถานที่นั้นๆ มีโดยโปรแกรมมีขั้นตอนการทำงานดัง

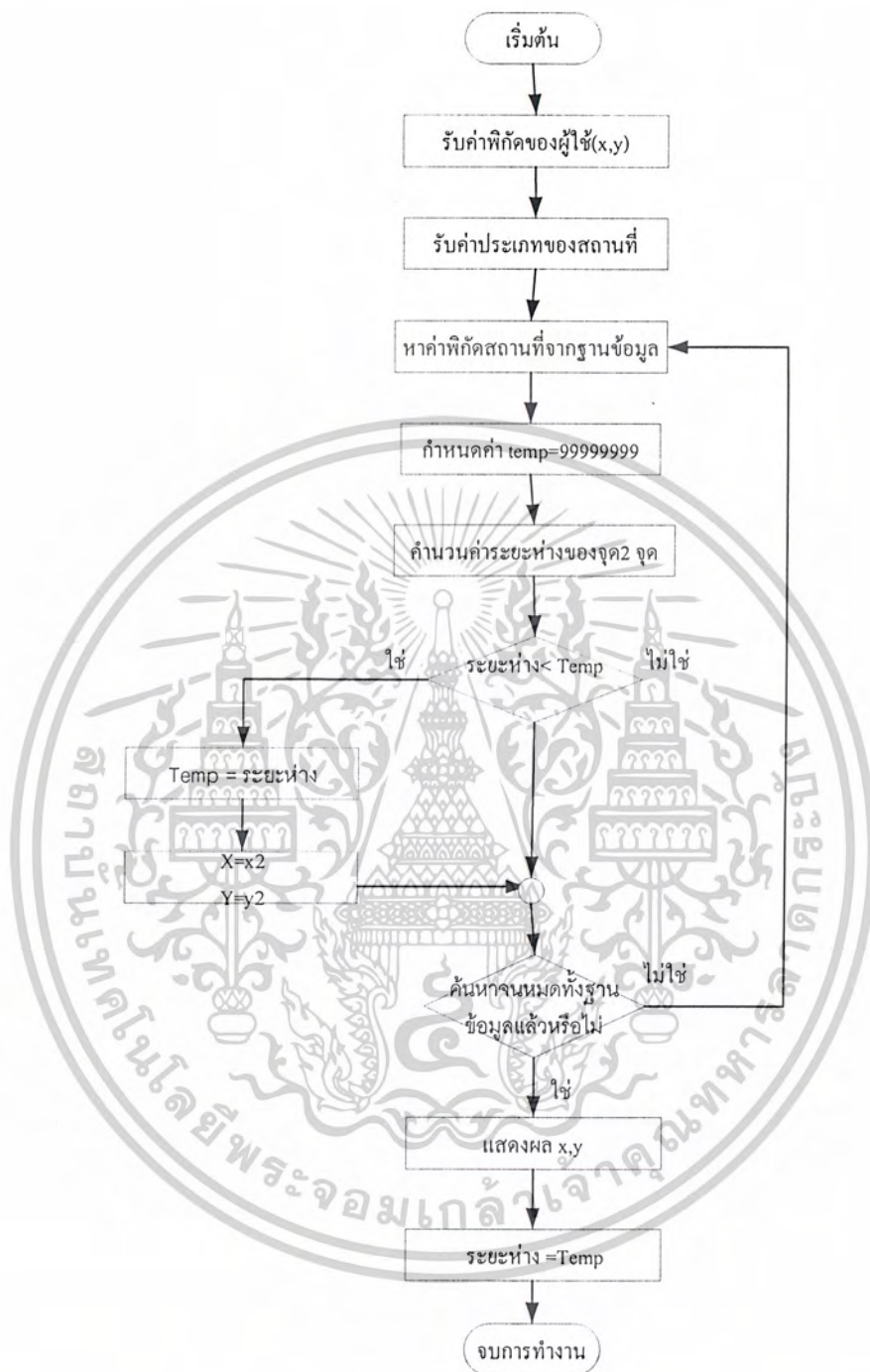
รูปที่ 3-16





รูปที่ 3-16 ขั้นตอนการทำงานของโปรแกรมค้นหาสถานที่

2. การค้นหาสถานที่ที่ใกล้ที่สุดจากจุดที่ผู้ใช้งานอยู่ จะเป็นการค้นหาสถานที่โดยให้ผู้ใช้เลือกประเภทของสถานที่ที่ต้องการค้นหา โปรแกรมจะนำตำแหน่งปัจจุบันของผู้ใช้ไปคำนวณกับตำแหน่งของประเภทสถานที่ที่ผู้ใช้เลือกไว้โดยใช้สูตรคณิตศาสตร์ที่ใช้หาระยะห่างระหว่างจุดทั้ง 2 บนระนาบ x,y เมื่อได้จุดที่ระยะห่างสั้นที่สุด จุดนั้นก็คือสถานที่ที่ใกล้ที่สุด



รูปที่ 3-17 ขั้นตอนการทำงานของโปรแกรมค้นหาสถานที่

ในการค้นหาจุดที่ใกล้ที่สุดนั้นเราจะทำการคำนวณหาระยะทางโดยใช้สูตรการหาระยะห่างระหว่างจุด 2 จุด ซึ่งมีค่าเท่ากับรากที่สองของ $((x_1-x_2)^2 + (y_1-y_2)^2)$ เมื่อกำหนดให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

X1, y1 คือค่าพิกัดของจุดที่ 1

X2, y2 คือค่าพิกัดของจุดที่ 2

3. การค้นหาถนนสายหลัก จะหลักการทำงานคล้ายกับการค้นหาสถานที่ โดยให้ผู้ใช้ใส่ชื่อของถนนที่ผู้ใช้ต้องการค้นหา โปรแกรมจะทำการนำชื่อถนนไปค้นหา ID ของถนนนั้นและนำไปเขียนต่อท้ายไฟล์ จากนั้น จาวาสคริปท์ก็จะนำ ID นั้นไปค้นหา tag ของถนนนั้น ซึ่งภาษาจาวาสคริปท์ที่ใช้ในการค้นหา tag ของถนนคือ SVGDoc.getElementById("ID ของถนน") ซึ่งจะเป็นข้อมูลที่บอก path ของถนนทั้งเส้น แล้วนำมาแสดงผล

เมื่อเซิร์ฟเวอร์ทำการประมวลผลหาพิกัดของสถานที่และชื่อของสถานที่ที่ผู้ใช้ต้องการได้แล้ว ก็จะนำพิกัดของสถานที่ที่ได้ไปแสดงผลบนไฟล์ SVG โดยเราใช้ SVG DOM ช่วยในการเพิ่ม Node หรือ Element ลงไปบนไฟล์ SVG เดิม

3.3.4.3 ส่วนติดต่อกับผู้ใช้งาน

ส่วนติดต่อกับผู้ใช้งานซึ่งเป็นหน้าเว็บจะแบ่งออกเป็น 2 ส่วนคือ

1. ส่วนรับข้อมูลจากผู้ใช้ ซึ่งจะใช้ <Form> ของภาษา HTML ในการสร้างและทำหน้าที่ในการส่งข้อมูลที่ใช้กรอก เช่น ชื่อสถานที่ที่ผู้ใช้ต้องการค้นหา
2. ส่วนแสดงผลแผนที่ ในส่วนนี้เป็นไฟล์ SVG ซึ่งทำการฝังลงในหน้าเพจ HTML โดยใช้ Tag

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

3.5 ส่วนของเครื่องไคลเอนท์

ในส่วนของเครื่องไคลเอนท์สำหรับระบบให้บริการแผนที่บนเว็บนั้นเราแบ่งเป็น 2 ส่วนคือ

3.5.1 ส่วนของบราวเซอร์ที่ใช้ในการเรียกดูเว็บไซต์ของระบบ

บราวเซอร์ที่ใช้นั้นเป็นบราวเซอร์ที่มีใช้งานโดยทั่วไป เพียงติดตั้ง Plugin ซึ่งมีชื่อว่า SVG Viewer ซึ่งเป็นของบริษัท Adobe เข้าไปก็สามารถใช้งานได้

3.5.2 ส่วนค้นหาค่าพิกัดของผู้ใช้

ส่วนค้นหาค่าพิกัดของผู้ใช้ประกอบด้วย GPS receiver ต่ออยู่กับ serial port ของเครื่อง ไคลเอนท์ โดย GPS receiver โดยส่วนนี้จะรับข้อมูลจาก GPS receiver ทาง serial port หลังจากนั้นจึงทำการวิเคราะห์เอา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เฉพาะข้อมูลส่วนที่เป็น ตำแหน่งของผู้ใช้งานส่งให้กับเซิร์ฟเวอร์เพื่อให้เซิร์ฟเวอร์นำไปประมวลผลต่อไป โดยโปรแกรมที่เขียนขึ้นนี้จะใช้เทคโนโลยีของ Java Servlet ซึ่งมีขั้นตอนการทำงานดังนี้

1. รอรับการเรียกใช้จากหน้าแรกของเว็บในฝั่งของเครื่องเซิร์ฟเวอร์ โดยที่หน้าแรกของเว็บบนฝั่งของเซิร์ฟเวอร์เรากำหนดให้มี ฟอর্ম(<form>) ของ HTML ซึ่งประกอบด้วย ปุ่มกดให้ผู้ใช้ทำการกดเพื่อหาค่าพิกัดของตัวเองโดยฟอর্মนี้จะกำหนด Action ไปที่ โปรแกรมหาอ่านค่าจากจีพีเอสซึ่งอยู่บนเครื่องไคลเอนท์

```
<form action="http://161.246.5.69:8080/servlet/serial" >
<input type="submit" name="update" value="Update Position"/>
</form>
```

2. เมื่อโปรแกรมอ่านค่าตำแหน่งจีพีเอสได้รับการเรียกจากหน้าแรกโปรแกรมนี้จะทำการอ่านค่าจากพอร์ทอนุกรมของเครื่องไคลเอนท์ โดยคำสั่งภาษาจาวาที่ใช้ในการอ่านค่าจากพอร์ทอนุกรมนั้น อยู่ใน Class SerialPort ซึ่งอยู่ใน packet javax.comm.*; ดังนั้นจึงต้องทำการ import packet นี้เข้ามา โดยใช้คำสั่ง

```
import javax.comm.*;
```

3. กำหนดอ็อบเจกต์ของ InputStream ที่ใช้ในการรับค่าที่อ่านเข้ามาจาก พอร์ทอนุกรม และกำหนดอ็อบเจกต์ของ class SerialPort

```
InputStream inputStream;
SerialPort serialPort;
```

4. เปิดพอร์ทอนุกรมอ่านค่าเข้ามาเก็บไว้ใน InputStream และกำหนดค่าตั้งต้นของพอร์ทอนุกรม

```
try {
    serialPort = (SerialPort) portId.open("testserialApp", 2000);
} catch (PortInUseException e) {}

try {
    if(beginning==true){
        inputStream = serialPort.getInputStream();
        beginning=false;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

serialPort.notifyOnDataAvailable(true);

serialPort.setSerialPortParams(4800,

    SerialPort.DATABITS_8,

    SerialPort.STOPBITS_1,

    SerialPort.PARITY_NONE);

serialPort.addListener(this);

}

```

5. นำค่าที่อ่านได้มาทำการวิเคราะห์ค่าละติจูดและลองจิจูดซึ่งค่าที่อ่านมาจากพอร์ทอนุกรมนั้นจะเป็นสตริงซึ่งมีรูปแบบตามโปรโตคอล NMEA GGA, Global Positioning System Fix Data โดยในที่นี้เรานำค่าที่อ่านจาก Inputstream มาเก็บในตัวแปรชนิดสตริงชื่อ str และใช้ ฟังก์ชัน indexOf เพื่อหาค่าแห่งเริ่มต้นของประโยคหลังจากนั้นจึงใช้ฟังก์ชัน substring ตัดเฉพาะข้อมูลที่เป็นละติจูดและลองจิจูด

```

index=str.indexOf("SGPGGA");

Lat=str.substring(index+14,index+23);

Lon=str.substring(index+27,index+36);

```

6. ส่งข้อมูลนั้นกลับไปให้กับเครื่องเซิร์ฟเวอร์ โดยการ redirect กลับไปยังหน้าแรกของเว็บอีกครั้ง

```

response.sendRedirect("http://161.246.5.56/index.jsp?Lat=" + Lat + "&Lon=" + Lon);

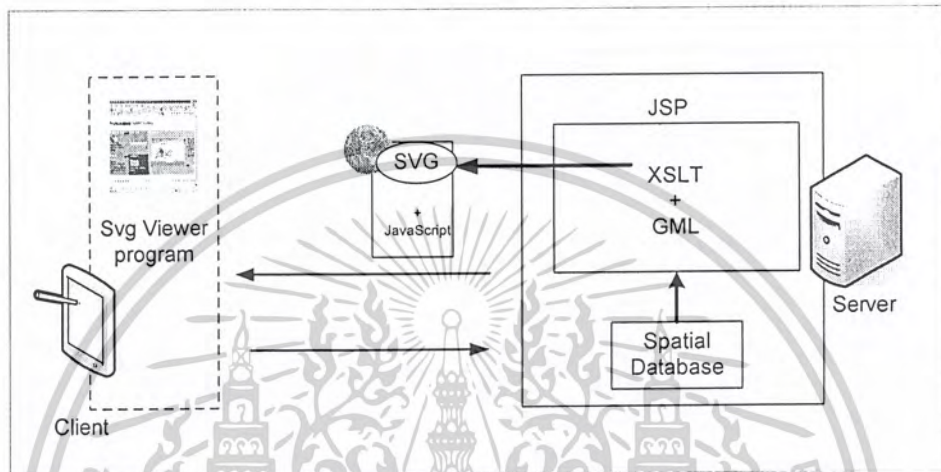
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การออกแบบระบบให้บริการแผนที่บนเครื่องพีดีเอ

4.1 โครงสร้างและการทำงานของระบบให้บริการแผนที่บนเครื่องพีดีเอ



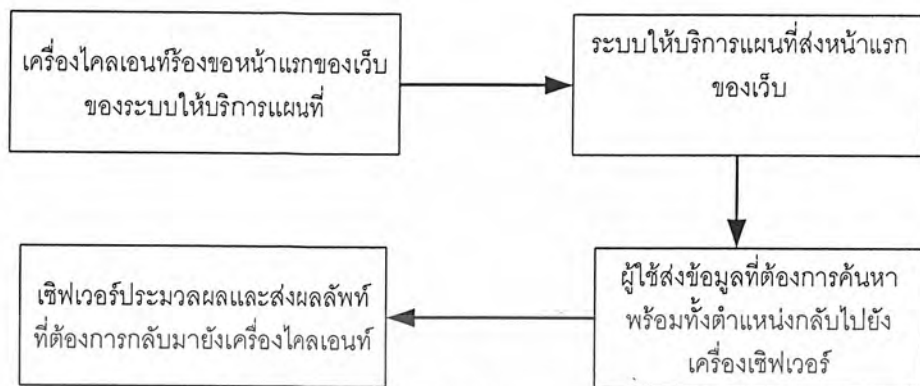
รูปที่ 4-1 โครงสร้างของระบบให้บริการแผนที่บนเครื่องพีดีเอ

ระบบให้บริการแผนที่บนเครื่องพีดีเอแบ่งออกเป็น 2 ส่วน

1. ส่วนของเครื่องเซิร์ฟเวอร์ ทำหน้าที่สร้างและประมวลผลแผนที่โดยนำข้อมูลแผนที่ซึ่งเก็บอยู่ในฐานข้อมูล มาทำให้อยู่ในรูปแบบของภาษา GML และแปลงข้อมูลแผนที่ซึ่งอยู่ในรูปแบบของภาษา GML ให้เป็น ภาษา SVG โดยใช้ XSLT ส่วนที่แตกต่างจากระบบให้บริการแผนที่บนเว็บคือ หน้าเพจที่สร้างขึ้นจากเซิร์ฟเวอร์ ซึ่งจะส่งไปให้ไคลเอนท์นั้นจะถูกสร้างขึ้นจากภาษา SVG ทั้งหมดเป็นไฟล์ .SVG

2. ส่วนของเครื่องไคลเอนท์ ซึ่งจะเรียกใช้บริการของเซิร์ฟเวอร์ ได้โดยเรียกผ่านโปรแกรมที่แสดงผล SVG ซึ่งในโครงการนี้อ้างอิงโปรแกรมภาษา จาวาชื่อว่า Batik

โดยระบบให้บริการให้บริการแผนที่บนพีดีเอมีการทำงานของระบบซึ่งแสดงได้ดังรูปต่อไปนี้



รูปที่ 4-2การทำงาน of ระบบ.ให้บริการแผนที่บนเครื่องพีดีเอ

4.2 การออกแบบระบบให้บริการแผนที่บนเครื่องพีดีเอ

ในการออกแบบส่วนของการให้บริการแผนที่บนเครื่องพีดีเอนั้นข้อแตกต่างสำคัญที่สุดของทั้งสองระบบคือ โปรแกรมบราวเซอร์ที่ใช้ติดตั้งบนเครื่องพีดีเอซึ่งในโครงการนี้อ้างอิง โปรแกรมภาษาจาวาชื่อว่า Batik ซึ่ง โปรแกรม Batik เป็นบราวเซอร์บนเครื่องพีดีเอ ซึ่งสามารถแสดงผลไฟล์ที่เป็นภาษา SVG ได้ แต่ไม่สามารถแสดงผล HTML ได้ดังนั้นเราจึงไม่สามารถส่งหน้าเว็บเพจซึ่งเป็นรูปแบบทั่วไปที่ใช้บนเว็บขึ้นมาแสดงผลบนเครื่องพีดีเอโดยผ่านโปรแกรม Batik ซึ่งทำให้เราต้องมีการเปลี่ยนแปลง ส่วนติดต่อกับผู้ใช้ในส่วนที่รับอินพุตจากผู้ใช้ โดยจากเดิมที่รับอินพุต โดยใช้ <form> ของ HTML ก็ทำการแก้ไข โดยให้ส่วนที่รับอินพุตเป็นภาษาจาวาสคริปต์และ ใช้ร่วมกับ

สำหรับหน้าแรกของเว็บนั้น เราได้ออกแบบให้เป็นภาษา SVG ทั้งหมด เพื่อให้สามารถใช้งานได้ทั้งบนบราวเซอร์บนเครื่องพีซี และบราวเซอร์บนเครื่อง พีดีเอ

4.3 สิ่งที่ต้องการในการติดตั้งโปรแกรม Batik

เนื่องจาก Batik เป็นโปรแกรมภาษาจาวา ดังนั้นในการติดตั้งโปรแกรม Batik จึงต้องการ Java Virtual Machine version 1.3 หรือสูงกว่านั้น และสามารถดาวน์โหลดโปรแกรม Batikได้ที่ <http://xml.apache.org/batik>

บทที่ 5

ผลการทดลอง

5.1 การใช้งาน

ในการใช้งานระบบให้บริการแผนที่ที่สามารถใช้งานได้โดย เรียกหน้าแรกของเว็บ ผ่านทาง URL <http://161.246.5.56/index.svg> โดยเมื่อเข้าสู่หน้าแรกจะเป็นหน้าที่ให้เลือกชนิดของการแสดงผล ซึ่งมีให้เลือก 2 แบบ คือ

1. การแสดงผลบนเว็บ ซึ่งเปิดดูโดยใช้บราวเซอร์ Internet explorer ที่ติดตั้ง plugin ซึ่งถ้าเลือกการแสดงผลแบบนี้ หน้าเพจที่จะได้รับจะเป็น HTML + SVG
2. การแสดงผลที่ดีเอ ซึ่งเปิดดูโดยใช้บราวเซอร์บาติก (batik) ซึ่งถ้าเลือกการแสดงผลแบบนี้ หน้าเพจที่ได้รับจะอยู่ในรูปแบบของไฟล์ SVG เพียงอย่างเดียว

เมื่อเข้าสู่ระบบจะมีฟังก์ชัน ในการใช้งานดังต่อไปนี้

1. การหาค่าพิกัดที่ผู้ใช้อยู่โดยผ่านเครื่องรับสัญญาณจีพีเอส
2. การย่อขยายแผนที่
3. การเลื่อนภาพแผนที่
4. การแสดงผลแผนที่โดยแบ่งตามเลเยอร์
5. การแสดงข้อมูลของสถานที่ต่าง ๆ บนแผนที่
6. การค้นหาสถานที่ ซึ่งสามารถเลือกได้ 2 แบบคือค้นหาสถานที่ที่ต้องการหรือสถานที่ที่ใกล้ที่สุด
7. การค้นหาถนน.

5.2 ผลการทดลองใช้งานระบบให้บริการแผนที่บนเว็บ

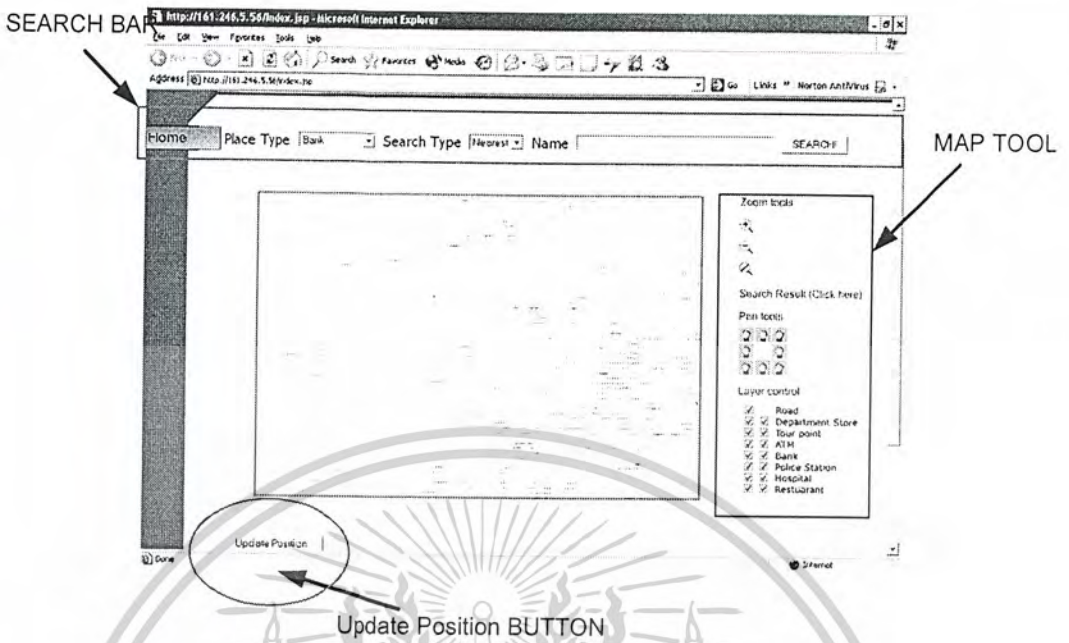
ในการทดลองนี้เราได้ทำการทดลองแสดงผลแผนที่และใช้ฟังก์ชันต่างๆที่ช่วยในการแสดงผลแผนที่ เช่นการย่อ ขยายภาพ หรือการเลื่อนภาพ และการค้นหาสถานที่ต่างๆ เป็นต้น

5.2.1 เมื่อเปิดเข้าสู่หน้าแรกของระบบ

เมื่อเปิดเข้าสู่หน้าแรกของเว็บเพจจะพบกับหน้าแรกซึ่งจะประกอบด้วย 3 ส่วนคือ

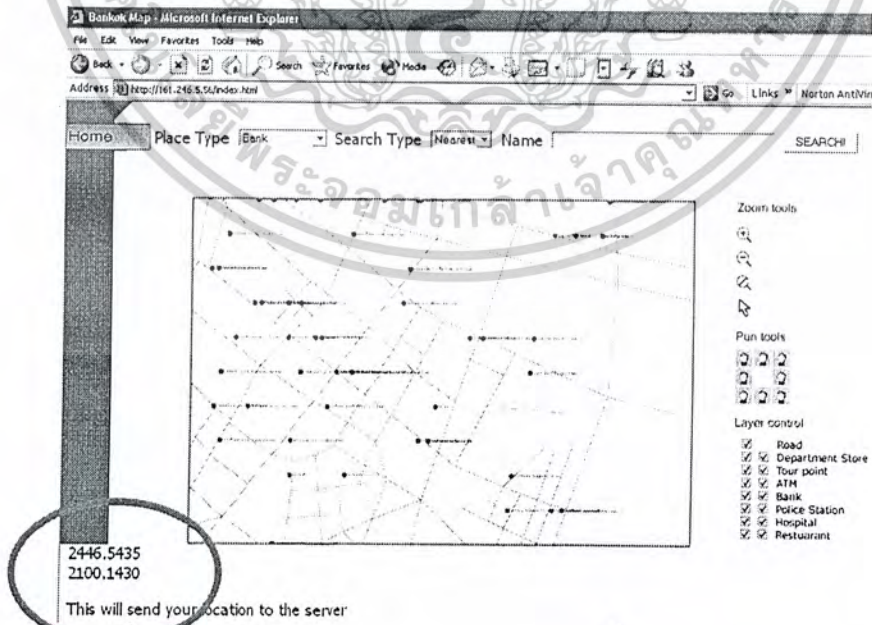
1. ส่วนควบคุมการแสดงผลแผนที่ (MAP TOOL)
2. ส่วน SEARCH BAR ซึ่งใช้ในการ ค้นหาถนนหรือสถานที่ที่ต้องการ
3. ปุ่มกดเพื่อส่งค่าพิกัดของผู้ใช้กลับไปให้กลับเครื่องเซิร์ฟเวอร์

โดยหน้าแรกของ ระบบสามารถแสดง ได้ดังภาพที่ 5.1



รูปที่ 5.1 หน้าแรกของระบบ



5.2.2 การทดลอง หากค่าพิกัดที่ผู้ใช้อยู่โดยผ่านเครื่องรับสัญญาณจีพีเอส เมื่อทำการกดปุ่มเพื่อหาค่าพิกัดของผู้ใช้งาน โปรแกรมจะทำการอ่านค่าพิกัดจากเครื่องรับสัญญาณจีพีเอส ซึ่งเรากำหนดค่าตั้งต้นให้ผู้ใช้ต่อเข้ากับพอร์ท COM11 ซึ่งเป็นพอร์ทอนุกรมของเครื่องคอมพิวเตอร์ ที่ผู้ใช้ใช้งาน

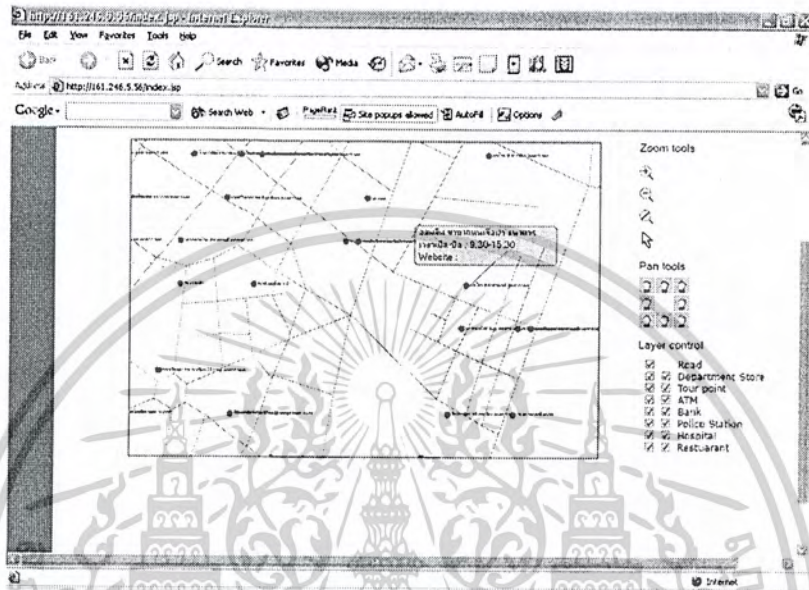


รูปที่ 5.2 การแสดงผลการอ่านค่าพิกัดของผู้ใช้จากเครื่องรับสัญญาณจีพีเอส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.3 การทดลองทำการย่อภาพแผนที่

เราสามารถทำการย่อหรือขยายภาพแผนที่ได้ โดยได้โดย กดปุ่ม  เพื่อทำการขยายแผนที่ หรือคลิกที่ปุ่ม  เพื่อทำการย่อภาพแผนที่ โดยทั้งสองปุ่มจะอยู่ในส่วนของ MAP TOOL

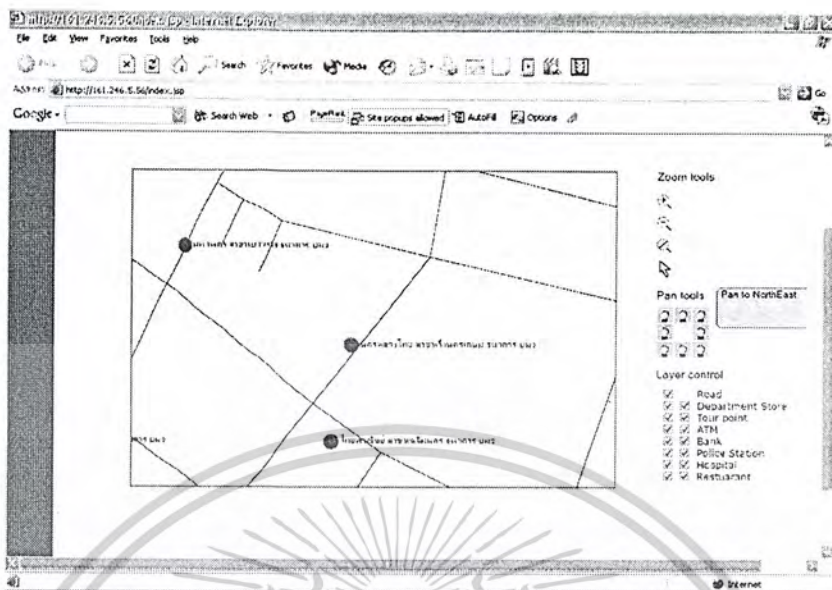


รูปที่ 5.3 การแสดงผลแผนที่ที่ทำการย่อภาพ

5.2.4 การทดลองทำการเลื่อนแผนที่

เราสามารถทำการ โดยการคลิกที่ปุ่ม PAN TOOL ซึ่งอยู่ใต้ปุ่มสำหรับย่อหรือขยายแผนที่ ซึ่งอยู่ในส่วนของ ส่วนควบคุมการแสดงผลแผนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.4 การแสดงผลการเลื่อนภาพแผนที่

5.2.5 การทดลองแสดงผลเฉพาะ Layer

การแสดงผลแผนที่แบบเป็นเลเยอร์เพื่อให้ผู้ใช้สามารถเลือกแสดงเฉพาะรายละเอียดของแผนที่ที่ต้องการ เราสามารถแสดงผลแผนที่เฉพาะเลเยอร์ที่เราต้องการได้โดยเลือกจาก Layer Control Tool ซึ่งอยู่ได้ปุ่ม PAN TOOL ซึ่งเราสามารถเลือกแสดงเฉพาะข้อมูลที่เราต้องการได้โดยการนำเครื่องหมายถูกออกจากเลเยอร์ที่เราต้องการ โดยเครื่องหมายถูกทางซ้ายจะเป็นการควบคุมการแสดง จุดของสถานที่ และเครื่องหมายถูกทางด้านขวาจะการควบคุมการแสดง ชื่อของสถานที่ ซึ่งผู้ใช้สามารถเลือกให้แสดงเฉพาะแบบใดแบบหนึ่งได้ ซึ่ง Layer Control Tool มีลักษณะดังรูปที่ 5.5

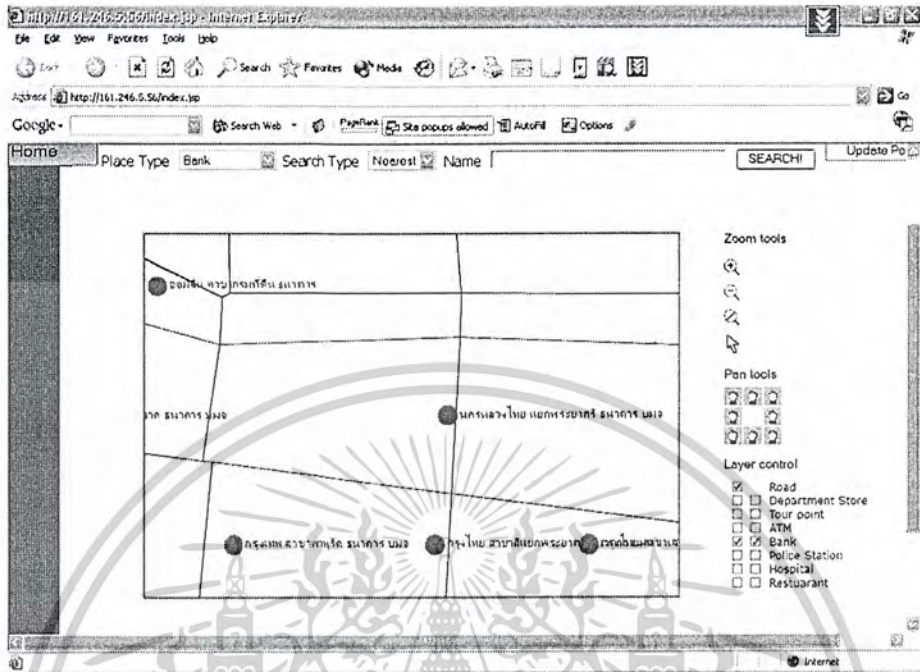
Layer control

- Road
- Department Store
- Tour point
- ATM
- Bank
- Police Station
- Hospital
- Restuarant

รูปที่ 5.5 Layer Control Tool

และผลการทดลองการแสดงผลเฉพาะเลเยอร์ที่ต้องการแสดงได้ดังรูปที่ 5.6 โดยเลือกแสดงเฉพาะเลเยอร์ของธนาคาร

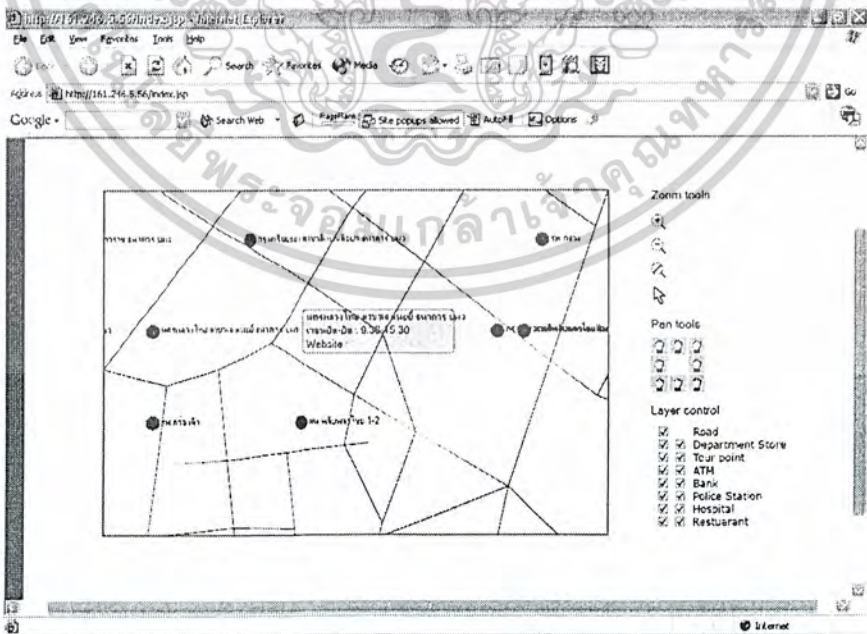
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.6 การแสดงผลแผนที่เฉพาะ Layer ที่ต้องการ

5.2.6 การแสดงข้อมูลของสถานที่ต่าง ๆ บนแผนที่

เป็นการแสดงผลรายละเอียดของสถานที่นั้นโดยใช้ภาษาจาวาสคริปต์ร่วมกับ SVG DOM เพื่อดึงข้อมูลของสถานที่ขึ้นมาแสดง

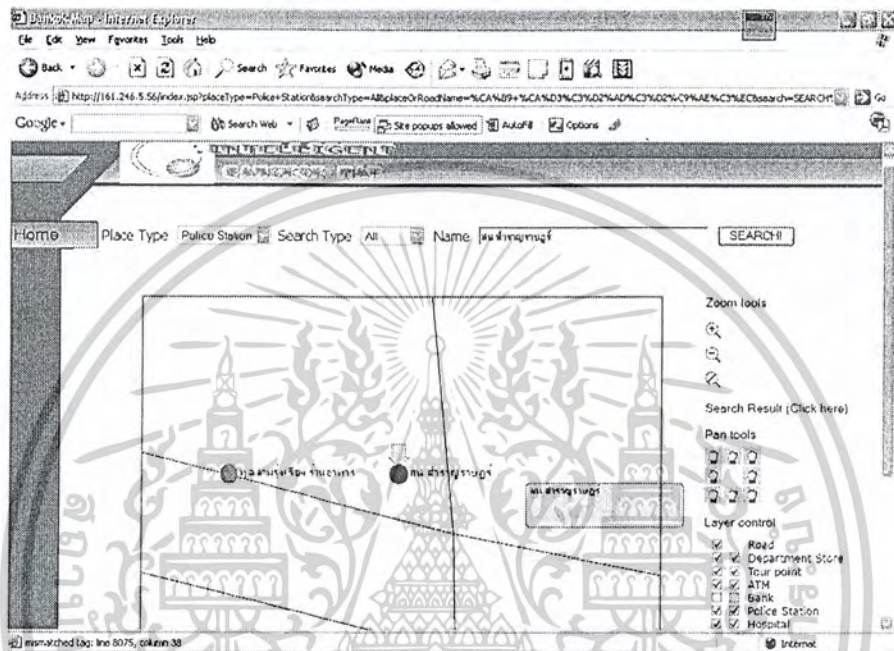


รูปที่ 5.7 การแสดงข้อมูลของสถานที่ต่าง ๆ บนแผนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.7 การทดลองค้นหาสถานที่

เราสามารถค้นหาสถานที่ที่ต้องการได้โดยเลือกประเภทของสถานที่ที่ต้องการค้นหาและใส่ชื่อสถานที่ที่ต้องการค้นหา และเลือกประเภทของการค้นหา เป็น “ALL” จากนั้นจึงกดปุ่ม SEARCH บนหน้าจอ

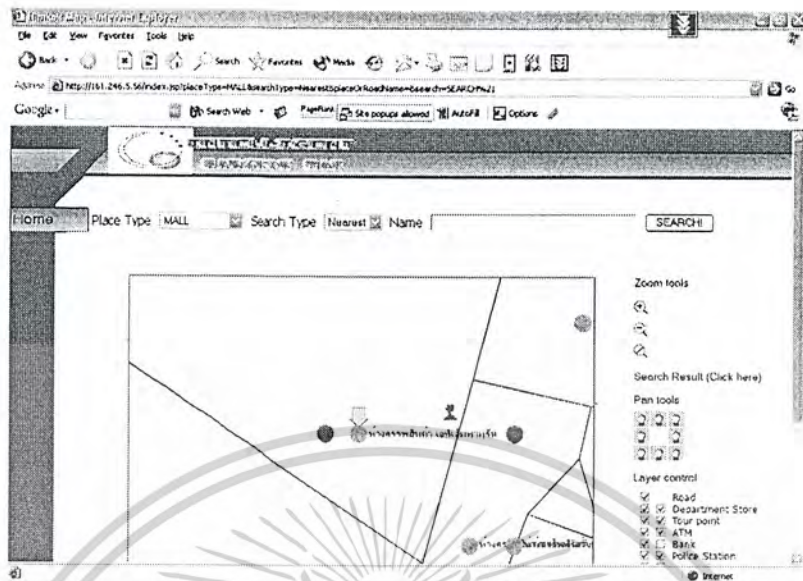


รูปที่ 5.8 แสดงค้นหาสถานที่โดยเลือกประเภทของสถานที่ที่ต้องการ

5.2.8 การทดลองค้นหาสถานที่ที่ใกล้ที่สุดจากตำแหน่งที่ผู้ใช้อยู่

หลังจากทำการกดปุ่มเพื่อส่งค่าพิกัดตำแหน่งปัจจุบันกลับไปให้เซิร์ฟเวอร์แล้ว เราสามารถค้นหาสถานที่ที่ต้องการได้โดยเลือกประเภทของสถานที่ที่ต้องการค้นหาและเลือกประเภทของการค้นหา เป็น “Nearest” จากนั้นจึงกดปุ่ม SERACH บนหน้าจอ

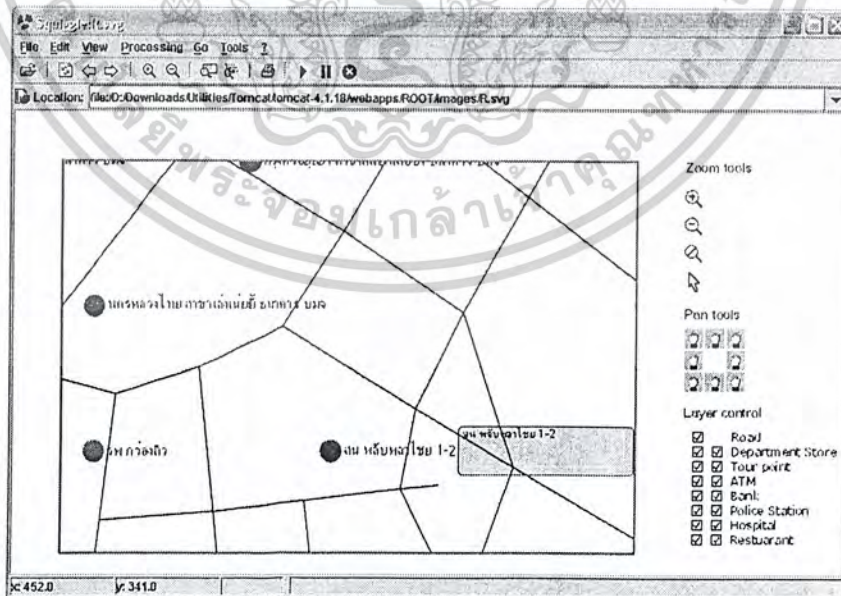
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.9 การค้นหาสถานที่ที่อยู่ใกล้กับผู้ใช้งานที่สุด

5.3 การทดลองใช้บราวเซอร์บาติก (Batik) ในการแสดงผลหน้าเว็บ

ในการแสดงผลบนเครื่องพีซีเดิมนั้น เราสามารถใช้บริการระบบให้บริการแผนที่ได้โดยใช้บราวเซอร์บาติก ซึ่งเป็น SVG บราวเซอร์ที่พัฒนาขึ้นจากภาษาจาวา แต่มีข้อจำกัดคือไม่สามารถแสดงผลบน HTML ได้ ดังนั้นหน้าเพจที่ใช้ในการแสดงผลบนบาติกถ้าเลือกการแสดงผลแบบนี้หน้าเพจที่ได้รับจะอยู่ในรูปแบบของไฟล์ SVG เพียงอย่างเดียว



รูปที่ 5.10 การใช้บราวเซอร์บาติก (Batik) ในการแสดงผลแผนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

บทวิจารณ์และสรุป

6.1 บทวิจารณ์และสรุป

ระบบให้บริการแผนที่บนเว็บและเครื่องพีดีเอที่พัฒนาขึ้นมาทำงานบนเว็บเซิร์ฟเวอร์และสามารถทำการประมวลผลฐานความรู้แผนที่ได้อย่างดี โดยมีให้บริการได้ดังนี้ สอบถามตำแหน่งของสถานที่สำคัญได้ สอบถามหาสถานที่ใกล้เคียง สอบถามตำแหน่งของถนน ซึ่งผลลัพธ์ที่ได้จะส่งกลับไปให้ผู้ใช้เป็นภาพแผนที่ ซึ่งมีการเคลื่อนไหวของสถานที่ที่ผู้ใช้ต้องการค้นหา และสามารถแสดงรายละเอียดของสถานที่นั้นได้เมื่อนำเมาส์ไปวางบนรูปสถานที่ ซึ่งรายละเอียดที่ได้ขึ้นอยู่กับชนิดของสถานที่นั้น

เมื่อเปรียบเทียบกับเว็บไซต์ที่ให้บริการแผนที่ในปัจจุบัน เว็บไซต์ที่พัฒนาขึ้นมาได้มีการเพิ่มความสามารถของภาษาจาวาสคริปต์เข้าไปในข้อมูล SVG ทำให้ไฟล์เอกสาร SVG ที่ได้จากการแปลงข้างต้นสามารถแสดงแผนที่ที่สามารถปฏิสัมพันธ์กับผู้ใช้ได้ดียิ่งขึ้น ผลลัพธ์แผนที่ในรูปของภาษา SVG สามารถแสดงผลบนเว็บเบราว์เซอร์ทั่วไปได้ รวมทั้งแสดงผลในบราวเซอร์ (Batik Browser) ซึ่งเป็นเอสวีบีบราวเซอร์ (SVG browser) บน platform Java

เทคโนโลยีที่นำมาใช้ในการพัฒนาระบบให้บริการแผนที่ในโครงการนี้ประกอบด้วย XML, GML, SVG, GPS, และ JavaServer Pages ทำให้แผนที่นี้สามารถใช้งานได้กับเว็บทุก Platform และเทคโนโลยีจาวาทำให้ระบบนี้สามารถใช้งานได้บนทุกระบบปฏิบัติการ ดังนั้นจึงเป็นวิธีการที่เหมาะสมที่จะนำไปใช้พัฒนาระบบให้บริการแผนที่ในอนาคตได้เป็นอย่างดี

6.2 ปัญหาที่เกิดขึ้นในการพัฒนาและแนวทางการแก้ปัญหา

1. เทคโนโลยีที่ใช้ในการพัฒนาแผนที่นี้เป็นเทคโนโลยีที่ยังใหม่อยู่มาก ทำให้เกิดปัญหาในการค้นหาเอกสารอ้างอิงและ เซอร์ฟแวร์ที่รองรับการทำงานของระบบ
2. ขั้นตอนนำข้อมูลแผนที่กรุงเทพมหานครเก็บลงในฐานข้อมูลของเครื่องเซิร์ฟเวอร์นั้น เนื่องจากฐานข้อมูลที่ไว้ในตอนแรกมีข้อมูลไม่สมบูรณ์และฐานข้อมูลที่สมบูรณ์นั้นได้มาในตอนหลังจึงใช้วิธีการเขียน โปรแกรมในส่วนแสดงผลแผนที่ไปก่อนและเมื่อได้ฐานข้อมูลถูกต้องมาแล้วจึงมาทำการออกแบบฐานข้อมูลใหม่ทำให้เกิดความล่าช้าและต้องกลับมาแก้ไข โปรแกรมหลายส่วน
3. แผนที่ที่ได้รับมาเป็นแผนที่ที่มีการออกแบบไว้สำหรับ แอปพลิเคชันของทางบริษัทจึงมีรูปแบบไม่เหมาะสมกับการใช้งานโดยตรงทำให้ต้องมีการปรับแก้ ฐานข้อมูลซึ่งมีข้อมูลเป็นจำนวนมากก่อให้เกิดความล่าช้าในการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ไม่มี plugin สำหรับการแสดงผล SVG ที่สามารถใช้กับ Internet Explorer บน windowCE ได้ทำให้ไม่สามารถใช้เบราว์เซอร์ที่มีอยู่ทั่วไปบนเครื่องพีดีเอ จึงต้องเปลี่ยนมาใช้ Batik ซึ่งสามารถแสดงผล SVG ได้ แต่ไม่สามารถแสดงผล HTML ได้จึงต้องทำการออกแบบส่วนแสดงผลแผนที่สำหรับแสดงบนเครื่องพีดีเอโดยเฉพาะ
5. Batik เป็นโปรแกรม ภาษาจาวาซึ่งต้องใช้ Java Virtual Machine ในการ run โปรแกรม ทำให้เกิดปัญหาความไม่เข้ากันของ Java Virtual Machine บนเครื่องพีดีเอที่ยังมีเวอร์ชันต่ำกว่า
6. ปัญหาในการใช้ DOM บน Batik เนื่องจาก Batik ไม่รองรับฟังก์ชันพื้นฐานของ DOM ในบางฟังก์ชันทำให้ต้องมีการแก้ไขโปรแกรมให้สามารถทำงานได้อย่างเหมาะสม
7. ปัญหาในการรับส่งค่า บน Batik เนื่องจาก Batik ไม่รองรับการใช้งาน HTML จึงมีความยุ่งยากในการส่ง Input จากผู้ใช้กลับไปยัง เครื่อง เซิร์ฟเวอร์ เนื่องจากจาวาสคริปต์ไม่สามารถรองรับความสามารถในการส่งค่าที่รับมาไปให้กับเซิร์ฟเวอร์จึงแก้ไขโดยการนำ Cookies เข้ามาช่วยในการส่งค่า

6.3 ข้อจำกัดของระบบ

ข้อจำกัดของระบบส่วนใหญ่เกิดจากปัญหาในการแสดงผล SVG ซึ่งยังไม่เป็นที่แพร่หลาย และโปรแกรมที่แสดงผล SVG ได้มีค่อนข้างจำกัดหรือไม่ตรงกับความต้องการที่จะใช้งาน โดยเฉพาะบนเครื่องพีดีเอ ซึ่งไม่มี plugin สำหรับ Internet Explorer ของ windowCE ทำให้ต้องใช้ Batik ซึ่งเป็นโปรแกรมที่ยังไม่แพร่หลายนักในการใช้งาน และยังมีปัญหาค่อนข้างมากในด้านความสามารถในการรองรับไฟล์ประเภทต่างๆ

6.4 แนวทางการพัฒนา

1. สามารถเพิ่มข้อมูลต่างของแผนที่เพื่อให้ครอบคลุมพื้นที่ทั้งหมดของกรุงเทพและเพิ่มรายละเอียดของข้อมูลสถานที่
2. สามารถเพิ่มเติมในส่วนของฟังก์ชันต่างๆของแผนที่ให้มีความสามารถมากขึ้น เช่น แสดงเส้นทางรถประจำทาง
3. เพิ่มเติมความสามารถในด้านการตอบสนองการใช้งานเช่น เพิ่มเติมภาพเคลื่อนไหวในส่วนต่างๆให้มากขึ้น หรือเพิ่มการแสดงผลที่หลากหลายโดยใช้ความสามารถของภาษา SVG
4. สามารถเพิ่มเติมในส่วนของ ฟังก์ชันที่สามารถช่วยในด้านธุรกิจเช่นสามารถเลือกร้านอาหารแล้วสามารถดาวน์โหลดคู่มือรถราตามาใช้ได้เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

XSLT Stylesheet ที่ใช้ในการแปลง GML เป็น SVG

```

<?xml version='1.0'?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:gml="http://www.opengis.net/gml"
exclude-result-prefixes="xsl exp Extfun gml"
xmlns:Extfun="/org.opengis.gml.StyleExt"
xmlns:xlink="http://www.w3.org/1999/xl_u110 ?k"
xmlns:svg="http://www.w3.org/2000/svg"
xmlns:exp="/" >
<xsl:output indent="yes" />
<!-- Coordinates were flipped already in this Translate2SVG.xsl (x = Lat, y = Lon)-->
<!-- fill="rgb(0,0,0)" style="font-weight:normal; font-family:'Verdana';font-size:12;" -->
<xsl:template match="/">
<svg viewBox="13815000_100454000 2000 14000" width="600" height="400">
<!-- Hospital -->
<xsl:comment><![CDATA[Hospital]]</xsl:comment>
<g id="Hospital" visibility="visible">
<g id="hospitalT" visibility="visible">
<xsl:for-each select="CityModel/cityMember/Hosp_u116 ?al" >
<xsl:apply-templates select="Location/gml:Point"
mode="Hospital" />
</xsl:for-each>
</g>
<g id="hospitalPoint" visibility="visible">
<xsl:for-each select="CityModel/cityMember/Hosp_u116 ?al" >
<xsl:call-template name="buildPoint" />
</xsl:for-each>
</g>
</g>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<!-- Mall -->
<xsl:comment><![CDATA[Mall]]></xsl:comment>
<g id="Mall" visibility="visible">
  <g id="MallT" visibility="visible">
    <xsl:for-each select="CityModel/cityMember/Mall" >
      <xsl:apply-templates select="Location/gml:Point"
mode="Mall" />
    </xsl:for-each>
  </g>
  <g id="MallPoint" visibility="visible">
    <xsl:for-each select="CityModel/cityMember/Mall" >
      <xsl:call-template name="buildPoint" />
    </xsl:for-each>
  </g>
  <!-- Bank -->
  <xsl:comment><![CDATA[Bank]]></xsl:comment>
  <g id="Bank" visibility="visible">
    <g id="BankT" visibility="visible">
      <xsl:for-each select="CityModel/cityMember/Bank" >
        <xsl:apply-templates select="Location/gml:Point"
mode="Bank" />
      </xsl:for-each>
    </g>
  </g>
  <g id="BankPoint" visibility="visible">
    <xsl:for-each select="CityModel/cityMember/Bank" >
      <xsl:call-template name="buildPoint" />
    </xsl:for-each>
  </g>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

</g>

<!-- Restuarant -->
<xsl:comment><![CDATA[Restuarant]]></xsl:comment>
<g id="Restuarant" visibility="visible">
  <g id="RestuarantT" visibility="visible">
    <xsl:for-each select="CityModel/cityMember/Restuarant" >
      <xsl:apply-templates select="Location/gml:Point"
mode="Restuarant" />
    </xsl:for-each>
  </g>
  <g id="RestuarantPoint" visibility="visible">
    <xsl:for-each select="CityModel/cityMember/Restuarant" >
      <xsl:call-template name="buildPoint" />
    </xsl:for-each>
  </g>
</!-- PoliceStation -->
<xsl:comment><![CDATA[PoliceStation]]></xsl:comment>
<g id="PoliceStation" visibility="visible">
  <g id="PoliceStationT" visibility="visible">
    <xsl:for-each select="CityModel/cityMember/PoliceStation" >
      <xsl:apply-templates select="Location/gml:Point"
mode="PoliceStation" />
    </xsl:for-each>
  </g>
  <g id="PoliceStationPoint" visibility="visible">
    <xsl:for-each select="CityModel/cityMember/PoliceStation" >
      <xsl:call-template name="buildPoint" />
    </xsl:for-each>
  </g>
</g>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        </g>
    </g>

</svg>
</xsl:template>

<!-- Hospital -->
<xsl:template match="gml:Point" mode="Hospital">
    <xsl:element name="text">
        <xsl:attribute name="id">
            <xsl:value-of select="ancestor::Hospital/@fid" />
        </xsl:attribute>
        <xsl:attribute name="x">
            <xsl:for-each select="gml:coordinates" >
                <xsl:value-of select="Extfun:getPointX(text(),@decimal,@cs)" />
            </xsl:for-each>
        </xsl:attribute>
        <xsl:attribute name="y">
            <xsl:for-each select="gml:coordinates" >
                <xsl:value-of select="Extfun:getPointY(text(),@decimal,@cs)" />
            </xsl:for-each>
        </xsl:attribute>
        <xsl:attribute name="dx">90</xsl:attribute>
        <xsl:attribute name="dy">20</xsl:attribute>
        <xsl:attribute name="style">font-weight:normal;font-family:Microsoft Sans
serif;font-size:100;</xsl:attribute>
        <xsl:attribute name="fill">rgb(0,0,0)</xsl:attribute>
        <xsl:value-of select="ancestor::Hospital/gml:name" />
    </xsl:element>
</xsl:template>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<!-- Mall -->
<xsl:template match="gml:Point" mode="Mall">
  <xsl:element name="text">
    <xsl:attribute name="id">
      <xsl:value-of select="ancestor::Mall/@fid" />
    </xsl:attribute>
    <xsl:attribute name="x">
      <xsl:for-each select="gml:coordinates" >
        <xsl:value-of select="Extfun:getPointX(text(),@decimal,@cs)" />
      </xsl:for-each>
    </xsl:attribute>
    <xsl:attribute name="y">
      <xsl:for-each select="gml:coordinates" >
        <xsl:value-of select="Extfun:getPointY(text(),@decimal,@cs)" />
      </xsl:for-each>
    </xsl:attribute>
    <xsl:attribute name="dx">90</xsl:attribute>
    <xsl:attribute name="dy">20</xsl:attribute>
    <xsl:attribute name="style">font-weight:normal;font-family:Microsoft Sans
serif;font-size:100;</xsl:attribute>
    <xsl:attribute name="fill">rgb(0,0,0)</xsl:attribute>
    <xsl:value-of select="ancestor::Mall/gml:name" />
  </xsl:element>
</xsl:template>

<!-- Bank -->
<xsl:template match="gml:Point" mode="Bank">
  <xsl:element name="text">
    <xsl:attribute name="id">
      <xsl:value-of select="ancestor::Bank/@fid" />
    </xsl:attribute>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<xsl:attribute name="x">
  <xsl:for-each select="gml:coordinates" >
    <xsl:value-of select="Extfun:getPointX(text(),@decimal,@cs)" />
  </xsl:for-each>
</xsl:attribute>
<xsl:attribute name="y">
  <xsl:for-each select="gml:coordinates" >
    <xsl:value-of select="Extfun:getPointY(text(),@decimal,@cs)" />
  </xsl:for-each>
</xsl:attribute>
<xsl:attribute name="dx">90</xsl:attribute>
<xsl:attribute name="dy">20</xsl:attribute>
<xsl:attribute name="style">font-weight:normal;font-family:'Microsoft Sans
serif';font-size:100;</xsl:attribute>
<xsl:attribute name="fill">rgb(0,0,0)</xsl:attribute>
<xsl:value-of select="ancestor::Bank/gml:name" />
</xsl:element>
</xsl:template>

<!-- Restuarant -->
<xsl:template match="gml:Point" mode="Restuarant">
  <xsl:element name="text">
    <xsl:attribute name="id">
      <xsl:value-of select="ancestor::Restuarant/@fid" />
    </xsl:attribute>
    <xsl:attribute name="x">
      <xsl:for-each select="gml:coordinates" >
        <xsl:value-of select="Extfun:getPointX(text(),@decimal,@cs)" />
      </xsl:for-each>
    </xsl:attribute>
    <xsl:attribute name="y">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        <xsl:for-each select="gml:coordinates" >
            <xsl:value-of select="Extfun:getPointY(text(),@decimal,@cs)" />
        </xsl:for-each>
    </xsl:attribute>
    <xsl:attribute name="dx">90</xsl:attribute>
    <xsl:attribute name="dy">20</xsl:attribute>
    <xsl:attribute name="style">font-weight:normal;font-family:'Microsoft Sans
serif;font-size:100;</xsl:attribute>
    <xsl:attribute name="fill">rgb(0,0,0)</xsl:attribute>
    <xsl:value-of select="ancestor::R_14tuarant/gml:name" />
</xsl:element>
</xsl:template>
<!-- PoliceStation -->
<xsl:template match="gml:Point" mode="PoliceStation">
    <xsl:element name="text">
        <xsl:attribute name="id">
            <xsl:value-of select="ancestor::Pol_u99 ?eStation/@fid" />
        </xsl:attribute>
        <xsl:attribute name="x">
            <xsl:for-each select="gml:coordinates" >
                <xsl:value-of select="Extfun:getPointX(text(),@decimal,@cs)" />
            </xsl:for-each>
        </xsl:attribute>
        <xsl:attribute name="y">
            <xsl:for-each select="gml:coordinates" >
                <xsl:value-of select="Extfun:getPointY(text(),@decimal,@cs)" />
            </xsl:for-each>
        </xsl:attribute>
        <xsl:attribute name="dx">90</xsl:attribute>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<xsl:attribute name="dy">20</xsl:attribute>
<xsl:attribute name="style">font-weight:normal;font-family:'Microsoft Sans
serif;font-size:100;</xsl:attribute>
<xsl:attribute name="fill">rgb(0,0,0)</xsl:attribute>
<xsl:value-of select="ancestor::Pol_u99 ?eStation/gml:name" />
</xsl:element>
</xsl:template>

<xsl:template name="buildPoint">
  <xsl:element name="circle">
    <xsl:attribute name="id">
      <xsl:value-of select="concat(@fix,'point')"/>
    </xsl:attribute>
    <xsl:attribute name="cx">
      <xsl:for-each select="Location/gml:Point/gml:coordinates">
        <xsl:value-of select="Extfun:getPointX(text(),@decimal,@cs)"/>
      </xsl:for-each>
    </xsl:attribute>
    <xsl:attribute name="cy">
      <xsl:for-each select="Location/gml:Point/gml:coordinates">
        <xsl:value-of select="Extfun:getPointY(text(),@decimal,@cs)"/>
      </xsl:for-each>
    </xsl:attribute>
    <xsl:attribute name="r">80</xsl:attribute>
    <xsl:attribute name="fill">
      <xsl:choose>
        <xsl:when test="starts-with(@fid, 'H')">orangered</xsl:when>
        <xsl:when test="starts-with(@fid, 'M')">yellowgreen</xsl:when>
        <xsl:when test="starts-with(@fid, 'R')">darkturquoise</xsl:when>
        <xsl:when test="starts-with(@fid, 'B')">darkorchid</xsl:when>
        <xsl:when test="starts-with(@fid, 'P')">red</xsl:when>
      </xsl:choose>
    </xsl:attribute>
  </xsl:element>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        <xsl:when test="starts-with(@fid, 'P')">red</xsl:when>
      </xsl:choose>
    </xsl:attribute>
    <xsl:attribute name="stroke">none</xsl:attribute>
    <xsl:attribute name="stroke-width">0</xsl:attribute>
  </xsl:element>
</xsl:template>
</xsl:stylesheet>

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

GML Application Schema (Bangkok.xsd)

```

<?xml version="1.0" encoding="UTF-16"?>
<!--File: Bangkok.xsd -->
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xlinks="http://www.w3.org/1999/xlink"
xmlns:gml="http://www.opengis.net/gml" elementFormDefault="qualified">
<xs:import namespace="http://www.opengis.net/gml" schemaLocation="feature.xsd"/>
<xs:import namespace="http://www.opengis.net/gml" schemaLocation="geometry.xsd"/>
<xs:import namespace="http://www.w3.org/1999/xlink" schemaLocation="xlinks.xsd"/>
<!--
-----
global element declaration
----->
<xs:element name="CityModel" type="CityModelType"
substitutionGroup="gml:_FeatureCollection"/>
<xs:element name="cityMember" type="CityMemberType"
substitutionGroup="gml:featureMember"/>
<xs:element name="AbstractPlace" type="AbstractPlaceType" abstract="true"
substitutionGroup="_CityFeature"/>
<xs:element name="_CityFeature" type="gml:AbstractFeatureType" abstract="true"
substitutionGroup="gml:_Feature"/>
<xs:element name="Location" type="gml:PointPropertyType"
substitutionGroup="gml:pointProperty"/>
<xs:element name="Hospital" type="HospitalType" substitutionGroup="_CityFeature"/>
<xs:element name="Mall" type="MallType" substitutionGroup="_CityFeature"/>
<xs:element name="PoliceStation" type="PoliceStationType" substitutionGroup="_CityFeature"/>
<xs:element name="Bank" type="BankType" substitutionGroup="_CityFeature"/>
<xs:element name="Restuarant" type="RestuarantType" substitutionGroup="_CityFeature"/>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<!--=====

type definition for bangkok model

=====-->

```

<xs:complexType name="CityModelType">
  <xs:complexContent>
    <xs:extension base="gml:AbstractFeatureCollectionType"/>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="CityMemberType">
  <xs:annotation>
    <xs:documentation>
      </xs:documentation>
    </xs:annotation>
  <xs:complexContent>
    <xs:restriction base="gml:FeatureAssociationType">
      <xs:sequence minOccurs="0">
        <xs:element ref="_CityFeature" minOccurs="0"/>
      </xs:sequence>
      <xs:attributeGroup ref="gml:AssociationAttributeGroup" />
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="PointPropertyType">
  <xs:annotation>
    <xs:documentation>Encapsulates a single point to represent position, location, or centerOf
properties.</xs:documentation>
  </xs:annotation>
  <xs:complexContent>
    <xs:restriction base="gml:GeometryAssociationType">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        <xs:sequence minOccurs="0">
            <xs:element ref="gml:Point"/>
        </xs:sequence>
        <xs:attributeGroup ref="xlinks:simpleLink"/>
        <xs:attribute ref="gml:remoteSchema" use="optional"/>
    </xs:restriction>
</xs:complexContent>
</xs:complexType>

<xs:complexType name="PointType">
    <xs:annotation>
        <xs:documentation>A Point is defined by a single coordinate tuple.</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
        <xs:extension base="gml:AbstractGeometryType">
            <xs:sequence>
                <xs:element name="Point" type="PointType"/>
            </xs:sequence>
            <xs:choice>
                <xs:element ref="gml:coord"/>
                <xs:element ref="gml:coordinates"/>
            </xs:choice>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>

<xs:complexType name="AbstractPlaceType" abstract="true">
    <xs:annotation>
        <xs:documentation>All Place elements are derived from this abstract supertype; a Place element
may have an identifying attribute (TEL, Opentime, Website). It may be associated with a spatial reference
system.</xs:documentation>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

</xs:annotation>
<xs:complexContent>
  <xs:extension base="gml:AbstractFeatureType">
    <xs:sequence minOccurs="0">
      <xs:element name="Website" type="xs:anyURI" minOccurs="0"/>
      <xs:element name="Opentime" type="xs:date" minOccurs="0" maxOccurs="3"/>
      <xs:element name="Location" type="PointPropertyType" minOccurs="0"/>
      <xs:element name="Tel" type="xs:string" minOccurs="0"/>
    </xs:sequence>
  </xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="HospitalType">
  <xs:complexContent>
    <xs:extension base="AbstractPlaceType">
      <xs:attribute name="Classified" type="xs:string"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="MallType">
  <xs:complexContent>
    <xs:extension base="AbstractPlaceType">
      <xs:sequence minOccurs="0">
        <xs:element name="Promotion" type="xs:string" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<xs:complexType name="PoliceStationType">
  <xs:complexContent>
    <xs:extension base="AbstractPlaceType">
      <xs:attribute name="Classified" type="xs:string"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="BankType">
  <xs:complexContent>
    <xs:extension base="AbstractPlaceType">
      <xs:attribute name="Classified" type="xs:string"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="RestuarantType">
  <xs:complexContent>
    <xs:extension base="AbstractPlaceType">
      <xs:sequence minOccurs="0">
        <xs:element name="Promotion" type="xs:string" />
        <xs:element name="Menu" type="xs:string" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
</xs:schema>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- [1] David Hunter, Curt Cagle, Dave Gibbons, Nikola Ozu, Jon Pinnock, Paul Spencer; “คัมภีร์การใช้ XML ฉบับสมบูรณ์ “,”บริษัท ซีเอ็ดดูเคชั่น จำกัด” 2002 ,262
- [2] ศุภชัย สมพานิช; “เข้าใจและใช้งานภาษา XML ฉบับโปรแกรมเมอร์,” อินโฟเพรส”,2001,61
- [3] Jagoe, Andrew “Mobile location service: the definitive guide”,”Practice Hall Professional Technical Reference”2003,147
- [4] สาริต ชัยวิวัฒน์ตระกูล;”แก่ง JSP ให้ครบสูตร”,”วิดีดีกรุ๊ป จำกัด”,2002,287
- [5] นิยม โพธิ์ลาด;”DHTML&JavaScript”,”ซัคเซสมิเดีย”,
- [6] Simon Cox;”Geography Markup language”,<http://www.opengis.net/gml/01-029/GML2.pdf>,2001
- [7] วีระศักดิ์ สุโขตินันท์ ;” ทำไปลองไป จาวาสคริปต์”,”วิดีดีกรุ๊ป จำกัด”,2002
- [8] Tsichritzis D.C. and Klug A. (1978) : “*The ANSI/X3/SPARC DBMS Frame Work : Report of the Study Group on Data Base Management Systems*”, “Information System” ,3 1978.
- [9] Jason cranford Teaque,Marc Campbell;”SVG FOR WEB DESIGNER”,”Wiley Publishing”,2003
- [10] <http://www.galdosinc.com/technology-index.html>
- [11] Jon Ferraiolo;” Scalable Vector Graphics (SVG) 1.1 Specification”,”
<http://www.w3.org/TR/2003/REC-SVG11-20030114/> “,2003
- [12] <http://www.adobe.com/svg/viewer/install/>
- [13] Philippe Le Hegaret;” Document Object Model (DOM)”,” <http://www.w3.org/DOM/>”,2002
- [14] <http://www.stylusstudio.com/>
- [15] <http://saxon.sourceforge.net/>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้