

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง
จีไอเอส-เพื่อการบรรเทาสาธารณภัยฉุกเฉิน
GIS for Public Emergency Rescue



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จีไอเอส-เพื่อการบรรเทาสาธารณภัยฉุกเฉิน
GIS for Public Emergency Rescue



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโท ปีการศึกษา 2549

ภาควิชา วิศวกรรมคอมพิวเตอร์

คณะ วิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง จีไอเอส-เพื่อการบรรเทาสาธารณภัยฉุกเฉิน

GIS for Public Emergency Rescue

คณะผู้จัดทำ

1. นาย ชัยยา กงศรี รหัสนักศึกษา 47015317
2. นาย วิวัฒน์ หาญจิต รหัสนักศึกษา 47015336




 อาจารย์ที่ปรึกษา
 (ผศ.ดร. วิศิษฐ์ หิรัญกิตติ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จีไอเอส-เพื่อการบรรเทาสาธารณภัยฉุกเฉิน

นาย ชัยยา กงศรี 47015317
 นาย วิวัฒน์ หาญจิต 47015336
 ผศ.ดร.วิศิษฎ์ หิรัญกิตติ อาจารย์ที่ปรึกษา
 ปีการศึกษา 2549

บทคัดย่อ

ปริญญาานิพนธ์ฉบับนี้ มุ่งพัฒนาระบบให้ความช่วยเหลือฉุกเฉิน ซึ่งเป็นการนำเอา GIS มาประยุกต์ใช้เพื่อช่วยเหลือประชาชนที่เดือดร้อนในเมืองใหญ่อย่างเช่น กรุงเทพมหานคร เนื่องจากในปัจจุบันเส้นทาง การเดินทางในกรุงเทพมหานครนั้นมีความซับซ้อนมากขึ้น ทำให้หน่วยงานของรัฐที่เกี่ยวข้องยากลำบากและเสียเวลามากในการเดินทางไปช่วยเหลือ ซึ่งประชาชนที่อาศัยอยู่อาจจะมี ความเดือดร้อนที่ต้องการบริการความช่วยเหลือ โดยเร่งด่วนจากหน่วยงานของรัฐ ได้แก่ รถดับเพลิง รถตำรวจ รถพยาบาล เป็นต้น การช่วยเหลือประชาชนที่ต้องการความช่วยเหลืออย่างเร่งด่วน จะเริ่มต้นด้วยประชาชนผู้เดือดร้อนต้องการความช่วยเหลือ ทำการแจ้งขอความช่วยเหลือผ่านทาง โทรศัพท์ระบบ CDMA หรือแจ้งขอความช่วยเหลือผ่านทาง Web site ที่ใช้แผนที่ที่เป็น GUI เมื่อการร้องขอถูกส่งไปยัง Server ส่วนกลางแล้ว Server ก็จะทำการส่งการร้องขอความช่วยเหลือไปยังหน่วยงานของรัฐที่เกี่ยวข้องที่อยู่ใกล้บริเวณจุดที่ผู้แจ้ง ขอความช่วยเหลือมากที่สุด พร้อมกับมีการระบุตำแหน่งของผู้เดือดร้อนให้กับหน่วยงานที่รับผิดชอบ จากนั้นผู้ที่จะทำการช่วยเหลือก็จะส่งความช่วยเหลือไปทางยานพาหนะ ไปยังจุดที่แจ้งขอความช่วยเหลือได้อย่างมีประสิทธิภาพ เพื่อทำการช่วยเหลือได้อย่างรวดเร็ว และในขณะที่มีการนำส่งผู้ป่วย ระบบสามารถแสดงภาพคนไข้ไปให้หมอเพื่อทำการวินิจฉัยล่วงหน้าได้ด้วย

โดยเราได้นำเอาเทคโนโลยี Location-based Service มาใช้ในการพัฒนาระบบในโครงการนี้ซึ่งประกอบไปด้วย XML , GML (Geography Markup Language) , SVG (Scalable Vector Graphics Language) , ฐานข้อมูลเชิงพื้นที่ (Spatial Database), AJAX (Asynchronous JavaScript + XML) , กล้อง IP และ Platform BREW ที่ใช้พัฒนาโปรแกรม Application บน โทรศัพท์มือถือระบบ CDMA เป็นต้น

GIS for Public Emergency Rescue

Mr. Chaiya	Kongsri	47015317
Mr. Wiwat	Hanjit	47015336
Asst.Prof.Dr. Visit	Hirankitti	Advisor

Academic Year 2006

ABSTRACT

In this thesis we have developed a GIS (Geographic Information System) system for public emergency rescue, which aims to help people living in a large city like Bangkok, where the road network is very complicated. It would take a lot of time and effort for a rescue team to travel by an ambulance, fire engine, or police car, to be able to reach people urgently needing help. The emergency rescue service provided by our system has the following steps. An emergency request is posed by a person who urgently needs help via an application on a CDMA phone or via a web application on our website. These mobile and web applications utilize GUI to locate the position of this person using a digital map. The position as that referenced by the map is sent together with the urgent request to the central server which then decides to pass the request to the nearest ambulance, fire engine, or police car, so that it can send a rescue team to the site relevant to the position and problems received from the person. The rescue team in the ambulance, fire engine, or police car employs a navigating and tracking system to locate and reach this person in a very express way. In the case of an rescue by an ambulance, while taking the patient to the hospital, our system can televise the video of the patient current conditions to a doctor for diagnosis in advance.

To develop this system, a location-based service technology was employed together with other technologies such as XML, GML (Geography Markup Language), SVG (Scalable Vector Graphics Language), spatial databases, AJAX (Asynchronous JavaScript + XML), an IP camera, and BREW platform for software development on a CDMA phone.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญาานิพนธ์ฉบับนี้สำเร็จลุล่วงได้อย่างดี ด้วยคำแนะนำ คำปรึกษาและคอยดูแลจากหลายๆ ฝ่ายด้วยกัน โดยเฉพาะอาจารย์ที่ปรึกษาที่ให้โอกาสกับข้าพเจ้าได้ทำปริญญาานิพนธ์ฉบับนี้ คอยให้ความเอาใจใส่ แนะนำ และให้ความช่วยเหลือเสมอมา คือ ผศ.ดร.วิศิษฎ์ หิรัญกิตติ ซึ่งต้องขอบพระคุณเป็นอย่างสูง

ขอกราบพระคุณคณาจารย์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกๆ ท่านที่ได้ประสิทธิ์ประสาทวิชาให้แก่ข้าพเจ้า ขอขอบคุณพี่ๆ เพื่อนๆ และน้องๆ ในห้องปฏิบัติการที่คอยให้แนะนำเป็นกำลังใจเสมอมา ในการทำงานและสร้างความถี่ครั้งน้่างบางครั้งบางคราว

ขอขอบคุณภาควิชาวิศวกรรมคอมพิวเตอร์ที่ได้จัดเตรียมสิ่งอำนวยความสะดวก เพื่อให้การวิจัยและพัฒนาเป็นไปได้อย่างสะดวกและรวดเร็ว รวมทั้งยังมีอินเทอร์เน็ตให้บริการ สำหรับการค้นคว้าหาความรู้ต่างๆ ซึ่งท้ายที่สุดแล้วก็ประกอบกันเป็นส่วนหนึ่งของโครงการนี้

และต้องขอขอบคุณบุคคลที่สำคัญที่สุดในชีวิตที่ทำให้ข้าพเจ้ามีวันนี้ นั่นคือ บิดา มารดา และบุคคลในครอบครัว อันเป็นที่เคารพรัก ซึ่งได้เลี้ยงดู คอยตั้งสอนข้าพเจ้ามาเป็นอย่างดี พร้อมทั้งให้โอกาสในการศึกษาอย่างเต็มที่และยังให้กำลังใจ ความรักเสมอมา ข้าพเจ้าขอกราบขอบพระคุณมา ณ ที่นี้ด้วย สุดท้ายนี้คุณค่าและประโยชน์ที่ได้มาจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอบแต่ผู้มีพระคุณทุกท่าน

ชัชยา กงศรี

วิวัฒน์ หาญจิต

สารบัญ

	หน้าที่
บทคัดย่อภาษาไทย	III
บทคัดย่อภาษาอังกฤษ	IV
กิตติกรรมประกาศ	V
สารบัญ	VI
สารบัญตาราง	VIII
สารบัญรูปภาพ	IX
บทที่ 1 บทนำ	
1.1 ความสำคัญและที่มา	1
1.2 วัตถุประสงค์ของโครงการ	2
1.3 ประโยชน์ที่คาดว่าจะได้รับ	3
1.4 ขอบเขตของโครงการ	3
1.5 วิธีการดำเนินงาน	4
1.6 ส่วนประกอบของรายงาน	5
บทที่ 2 ภาษา XML และภาษา Markup ที่เกี่ยวข้องกับการพัฒนาแผนที่	
2.1 ภาษา XML (Extensible Markup Language)	6
2.1.1 ลักษณะโครงสร้างของภาษา XML	6
2.1.2 ตัวกำหนดโครงสร้างของ XML ด้วย DTD	8
2.1.3 การแปลงรูปแบบเอกสาร XML	10
2.1.4 DOM (Document Object Model)	10
2.2 ภาษา Markup Language ที่เกี่ยวข้องกับการพัฒนาแผนที่	12
2.2.1 ภาษา GML (Geography Markup Language)	12
2.2.2 ภาษา SVG (Scalable Vector Graphics Language)	17
2.2.3 ภาษาจาวาสคริปต์ (Java Script)	22
2.2.4 ภาษา Ajax (Asynchronous JavaScript And XML)	31
บทที่ 3 GIS และ Spatial database	
3.1 ระบบสารสนเทศภูมิศาสตร์ (Geographic Information System : GIS)	38
3.2 ฐานข้อมูลเชิงพื้นที่ (Spatial database)	40
3.2.1 Model of spatial information	40
3.2.2 Spatial Data Type	41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้าที่
3.2.3 Spatial Relationships	42
3.3 postgresSQL	43
บทที่ 4 BREW Platforms	
4.1 BREW (Binary Runtime Environment for Wireless)	46
4.1.1 หลักการพื้นฐานของ BREW Platform	46
4.1.2 ระบบ BREW	47
4.1.3 สถาปัตยกรรมและระบบการทำงานของ BREW	47
4.1.4 BREW SDK COMPONENTS	48
4.1.5 MODULE INFORMATION FILE (MIF)	48
4.1.6 ความแตกต่างระหว่าง Handset กับ Emulator	48
4.2 การเขียน BREW Application	48
บทที่ 5 การออกแบบโครงสร้างของระบบ	
5.1 การออกแบบโครงสร้างระบบเรียกรถฉุกเฉิน	56
5.2 การสร้างส่วนประกอบต่างๆ ของระบบ	57
5.3 การพัฒนาในส่วนของรถฉุกเฉิน	63
5.4 การออกแบบและพัฒนาโปรแกรมเรียกใช้บริการรถฉุกเฉินบน Handset	65
5.5 ส่วนของการใช้งานกล้อง	67
บทที่ 6 ผลการทดลอง	
6.1 การทดลองการทำงานหลักๆ ของ Web Application	70
6.2 การทำงานของ Program ที่อยู่บนรถฉุกเฉิน	77
6.3 การทำงานของโทรศัพท์เคลื่อนที่	80
บทที่ 7 บทวิจารณ์และสรุป	
7.1 บทสรุป	83
7.2 ประโยชน์ที่ได้จากโครงงาน	83
7.3 ปัญหาที่พบในการพัฒนา	84
7.4 แนวทางการแก้ไข	84
ภาคผนวก ก.การติดตั้งโปรแกรมทั้งหมด	85
ภาคผนวก ข.รายละเอียดไฟล์โปรแกรมและโมดูลทั้งหมด	94
บรรณานุกรม	105

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้าที่
ตารางที่ 2-1 แสดงชนิดของแอตทริบิวต์ที่สามารถใช้ในการประกาศแอตทริบิวต์	9
ตารางที่ 2-2 แสดงค่าที่สามารถได้ในส่วนที่เป็นค่าดีฟอลต์ในการประกาศแอตทริบิวต์	9
ตารางที่ 2-3 แสดงรูปทรงเรขาคณิตพื้นฐานที่ถูกระบุไว้ใน GML	13
ตารางที่ 2-4 แสดงคำสั่งที่ใช้ในการสร้างรูปร่าง	20
ตารางที่ 2-5 แสดง SVG Element	21
ตารางที่ 2-6 แสดง SVG Attribute	22
ตารางที่ 5-1 แสดงส่วนควบคุมการให้บริการ	63
ตารางที่ 5-2 แสดงส่วนควบคุมแผนที่	64
ตารางที่ 5-3 แสดงส่วนควบคุมส่วนอื่นๆ	64
ตารางที่ 5-4 แสดงการติดต่อระหว่างเซิร์ฟเวอร์และโปรแกรมบนรถจักรยาน	65
ตารางที่ 5-5 แสดง Query String ที่ส่งไปเมื่อคลิกปุ่ม	69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

	หน้าที่
รูปที่ 2-1 การใช้งาน DOM	11
รูปที่ 2-2 แสดงวิธีการเข้าถึงข้อมูลโดยวิธี DOM Parsing	11
รูปที่ 2-3 The geometry model for simple feature (Open GIS Consortium, 2001)	12
รูปที่ 2-4 Features and properties	13
รูปที่ 2-5 แสดงรูปที่สร้างโดย Point	15
รูปที่ 2-6 แสดงรูปที่สร้างโดย LineString	15
รูปที่ 2-7 แสดงรูปที่สร้างโดย Polygon	15
รูปที่ 2-8 Ajax architecture	33
รูปที่ 2-9 เปรียบเทียบระบบการทำงานของเว็บแบบมาตรฐานและแบบใช้ Ajax	34
รูปที่ 2-10 ระบบเว็บแอปพลิเคชันแบบมาตรฐาน (Synchronize)	34
รูปที่ 2-11 ระบบเว็บแอปพลิเคชันแบบที่ใช้ Ajax (Asynchronous)	35
รูปที่ 2-12 Ajax Model: Partial UI updates and asynchronous communications	36
รูปที่ 3-1 องค์ประกอบของ GIS	38
รูปที่ 3-2 ลักษณะข้อมูลที่แสดงทิศทาง (vector data)	39
รูปที่ 3-3 ข้อมูลตารางกริด (raster data)	39
รูปที่ 3-4 โครงสร้างพื้นฐานที่ใช้ใน Object-Based Model	41
รูปที่ 3-5 โครงสร้างพื้นฐานที่ใช้ใน Field-Base Model	41
รูปที่ 3-6 โครงสร้างและความสัมพันธ์ภายในของ spatial geometry ในรูปแบบของ UML	42
รูปที่ 3-7 ลักษณะการทำงานของ PostgreSQL ติดต่อกับ Web page	43
รูปที่ 3-8 แสดงตัวอย่างการใช้ภาษา SQL ใน PostgreSQL	45
รูปที่ 4-1 แสดง Device Architecture (Layer) ของ BREW	47
รูปที่ 4-2 แสดง ระบบการทำงานของ BREW	47
รูปที่ 4-3 แสดง BREW Application Wizard	49
รูปที่ 4-4 สรุป Information	49
รูปที่ 4-5 แสดง Workspace	50
รูปที่ 4-6 แสดง Code ที่ถูก Generated โดย BREW Application Wizard	50
รูปที่ 4-7 แสดงการ Generate Local class ID	52
รูปที่ 4-8 แสดง Specify setting privileges	53
รูปที่ 4-9 แสดงการเพิ่ม String resource	53

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ(ต่อ)

	หน้าที่
รูปที่ 4-10 แสดงการเพิ่ม Image resource	53
รูปที่ 4-11 แสดงการ Compile resource file	54
รูปที่ 4-12 แสดงการ Setting output file .dll	54
รูปที่ 4-13 แสดง Application Menu	55
รูปที่ 5-1 แสดงโครงสร้างของระบบเรียกรถฉุกเฉิน	56
รูปที่ 5-2 แสดง ER Diagram ของฐานข้อมูลรถฉุกเฉินและการเรียกขอความช่วยเหลือ	60
รูปที่ 5-3 โปรแกรมที่อยู่บนรถฉุกเฉิน ที่ทำการตอบรับการเรียกขอความช่วยเหลือแล้ว	63
รูปที่ 5-4 แสดงโครงสร้างในส่วนของโทรศัพท์มือถือ	65
รูปที่ 5-5 แสดงลักษณะของข้อมูลที่ทำกรรับ-ส่งระหว่าง Handset กับ เซิร์ฟเวอร์	66
รูปที่ 5-6 แสดงตัวอย่าง Query string	66
รูปที่ 5-7 แสดงรูปโครงสร้างของกล้อง	67
รูปที่ 5-8 แสดงหน้าจอที่แสดงภาพที่ได้มาจากกล้อง	68
รูปที่ 6-1 แสดงหน้าเว็บเพจของ Web Emergency Rescue	71
รูปที่ 6-2 แสดงการนำเทคโนโลยี Ajax มาใช้กับเว็บ	72
รูปที่ 6-3 แสดงแผนที่ก่อนการกดปุ่ม Zoom in (+) เพื่อทำการขยายขนาดของแผนที่	73
รูปที่ 6-4 แสดงแผนที่หลังการกดปุ่ม Zoom in (+) เพื่อทำการขยายขนาดของแผนที่	73
รูปที่ 6-5 แสดงแผนที่ก่อนการกดปุ่ม Zoom out (-) เพื่อทำการย่อขนาดของแผนที่	74
รูปที่ 6-6 แสดงแผนที่หลังการกดปุ่ม Zoom out (-) เพื่อทำการย่อขนาดของแผนที่	74
รูปที่ 6-7 แสดงแผนที่ก่อนการกดปุ่มเลื่อน เพื่อทำการเลื่อนแผนที่	75
รูปที่ 6-8 แสดงแผนที่หลังการกดปุ่มเลื่อน เพื่อทำการเลื่อนแผนที่	75
รูปที่ 6-9 แสดงการทดลองค้นหาสถานที่บนแผนที่	76
รูปที่ 6-10 แสดงแสดง โปรแกรมที่อยู่ในส่วนของรถฉุกเฉิน	77
รูปที่ 6-11 แสดงผลบนเว็บเพจ	78
รูปที่ 6-12 แสดงอยู่บน Program ที่อยู่บนรถฉุกเฉิน	79
รูปที่ 6-13 แสดงอยู่บน Program ที่ใช้ติดตามรถฉุกเฉินทุกคัน	79
รูปที่ 6-14 แสดงไอคอน Program Application on Mobile ของ Emergency Rescue	80
รูปที่ 6-15 แสดงหน้าจอหน้าแรกของการทำงานเมนูหลัก	81
รูปที่ 6-16 แสดงหน้าเมนูของการเรียกรถพยาบาลแบบฉุกเฉิน	82
รูปที่ 6-17 แสดงหน้าเมนูของผลการเรียกรถพยาบาลแบบฉุกเฉินว่าสำเร็จ	82

บทที่ 1

บทนำ

1.1 ความสำคัญและที่มา

ในการทำงานและการดำเนินกิจกรรมต่างๆในชีวิตประจำวัน ไม่ว่าจะเป็นงานที่เกี่ยวข้องกับการช่วยเหลือสังคม บรรเทาสาธารณภัยหรืองานในด้านธุรกิจนั้น จำเป็นที่จะต้องอาศัยการติดต่อสื่อสาร รวมไปถึงการเดินทาง ในปัจจุบันด้วยความก้าวหน้าทางเทคโนโลยีการสื่อสาร ได้แก่ เทคโนโลยีการติดต่อสื่อสารบนอินเทอร์เน็ต , เทคโนโลยีโทรศัพท์เคลื่อนที่ และเทคโนโลยีคอมพิวเตอร์แบบพกพา ทำให้การติดต่อสื่อสารมีความสะดวกรวดเร็ว ซึ่งทั้งงานที่เกี่ยวข้องกับการช่วยเหลือสังคม เช่น การช่วยเหลือผู้บาดเจ็บหรือผู้ป่วย , การช่วยเหลือผู้ประสบอัคคีภัย หรือ การช่วยเหลือผู้ประสบกับเหตุร้าย และรวมทั้งการติดต่อทางธุรกิจมีความคล่องตัวขึ้นมาก อย่างไรก็ตาม สิ่งที่เราไม่ได้คือการเดินทาง ดังนั้นการพัฒนาเทคโนโลยีที่จะช่วยให้การเดินทางมีความสะดวกรวดเร็ว จึงเป็นสิ่งที่ไม่ยิ่งหย่อนไปกว่าการสื่อสาร องค์ประกอบสำคัญประการหนึ่งของการเดินทางคือ แผนที่ โดยที่แผนที่จะเป็นแหล่งเก็บข้อมูลของตำแหน่ง ถนน ตรอก ซอย สถานที่ต่างๆ ฯลฯ ซึ่งมีปริมาณข้อมูลเป็นจำนวนมากทำให้ยากแก่การค้นหา ซึ่งแผนที่ที่มีประโยชน์ในการใช้ในการอ้างอิงตำแหน่งของสถานที่ต่างๆ โดยแผนที่ที่มีอยู่ไม่สามารถตอบสนองความต้องการของผู้ใช้ได้อย่างเต็มที่เช่น การค้นหาสถานที่ต่างๆต้องใช้ตารางระบุพิกัดของสถานที่ที่ต้องการ จึงได้นำคอมพิวเตอร์มาแก้ปัญหาเหล่านั้น และนอกจากนี้ข้อมูลยังมีการเปลี่ยนแปลงอยู่ตลอดเวลา ทำให้แผนที่กระดาษไม่อำนวยความสะดวกกับผู้ใช้ และไม่สามารถรองรับการพัฒนาการเดินทางในอนาคตที่ต้องการความสะดวกรวดเร็ว และต้องการข้อมูลที่มีความถูกต้องทันสมัยอยู่ตลอดเวลาได้ ปัจจุบันได้มีการพัฒนาแผนที่คอมพิวเตอร์หรือที่เรียกว่า “แผนที่ดิจิทัล” ขึ้นและเริ่มมีความนิยมใช้งานกันเพิ่มมากขึ้น แผนที่ดิจิทัลนั้นสามารถให้ข้อมูลของ ถนน สถานที่สำคัญต่างๆ ฯลฯ สามารถค้นหาข้อมูลตำแหน่งของถนน หรือสถานที่สำคัญ โดยบอกเพียงชื่อถนน หรือสถานที่ที่ต้องการค้นหา ก็สามารถให้ผลลัพธ์ได้รวดเร็วและแม่นยำ

ในปัจจุบันระบบสารสนเทศภูมิศาสตร์ (Geometry Information System: GIS) เริ่มมีใช้งานกันอย่างแพร่หลาย ตัวอย่างเช่น ทางทหาร การคมนาคม การสำรวจทรัพยากรธรรมชาติและธุรกิจ เป็นต้น ระบบสารสนเทศภูมิศาสตร์เป็นกระบวนการทำงานเกี่ยวกับข้อมูลในเชิงพื้นที่ด้วยระบบคอมพิวเตอร์ ที่ใช้กำหนดข้อมูลและสารสนเทศที่มีความสัมพันธ์กับตำแหน่งในเชิงพื้นที่ ซึ่งรูปแบบและความสัมพันธ์ของข้อมูลเชิงพื้นที่ทั้งหลายจะสามารถนำมาวิเคราะห์ ซึ่งถือว่าเป็นข้อได้เปรียบที่เด่นชัดเมื่อเทียบกับการใช้แผนที่ตามปกติเดิมเมื่อองค์กรต้องการที่จะได้ข้อมูลจากระบบสารสนเทศเชิงภูมิศาสตร์ก็จะต้องพัฒนาระบบของตนเองและเก็บข้อมูลต่างๆไว้ที่ตนเอง เมื่อส่วน

ต่างๆภายในองค์กรหรือองค์กรอื่นต้องการใช้ระบบก็จะต้องปฏิบัติตามรูปแบบที่ได้ออกแบบระบบนั้นไว้ซึ่งเป็นเรื่องที่ทำได้ยากที่จะทำให้ผู้คนอื่นทราบและเข้าใจระบบที่ออกแบบไว้และอาจทำให้องค์กรอื่นตัดสินใจที่จะพัฒนาระบบขึ้นมาเป็นของตนเองทำให้มีการเก็บข้อมูลไว้หลายที่เกินความจำเป็นซึ่งข้อมูลแต่ละที่อาจมีความคลาดเคลื่อนได้

ดังนั้นโครงการนี้จึงเอาข้อดีของเทคโนโลยีที่มีความก้าวหน้า เช่น เทคโนโลยีการติดต่อสื่อสารบนอินเทอร์เน็ต หรือเทคโนโลยีโทรศัพท์เคลื่อนที่ (Mobile) ที่ทำให้การติดต่อสื่อสารมีความสะดวกและรวดเร็วมาประยุกต์ร่วมกับ ระบบสารสนเทศภูมิศาสตร์ (Geometry Information System: GIS) เพื่อสร้างระบบที่มีส่วนร่วมในการช่วยเหลือสังคมบรรเทาสาธารณภัยที่ต้องการความช่วยเหลือโดยเร่งด่วน เช่น การช่วยเหลือผู้บาดเจ็บหรือผู้ป่วย, การช่วยเหลือผู้ประสบอัคคีภัย หรือ การช่วยเหลือผู้ประสบกับเหตุร้าย เป็นต้น

1.2 วัตถุประสงค์ของโครงการ

- 1.2.1 เพื่อศึกษาและพัฒนาระบบที่มีส่วนในการช่วยเหลือสังคมบรรเทาสาธารณภัย
- 1.2.2 เพื่อศึกษาและนำเอาเทคโนโลยีภาษา XML มาใช้เขียนสร้างระบบแผนที่ พร้อมกับการปรับปรุงแผนที่คอมพิวเตอร์ที่ให้บริการบนเว็บปัจจุบัน ให้มีขีดความสามารถเพิ่มขึ้น
- 1.2.3 เพื่อศึกษาและนำเอาระบบสารสนเทศภูมิศาสตร์ (GIS) มาช่วยในการสร้างแผนที่ที่มีมาตรฐาน
- 1.2.4 เพื่อศึกษาและนำเอาเทคโนโลยีของ Spatial Database มาใช้เป็นระบบฐานข้อมูลของแผนที่เพื่อให้มีการเก็บข้อมูลแผนที่ที่เป็นมาตรฐาน
- 1.2.5 เพื่อศึกษาและนำเอาภาษา JavaScript มาเป็นตัวที่ช่วยในการแสดงผลภาพแผนที่ที่เป็นแบบ SVG บนเว็บเบราว์เซอร์
- 1.2.6 เพื่อศึกษาและพัฒนาเทคโนโลยีการสื่อสารทางด้าน โทรศัพท์เคลื่อนที่ (Mobile) ให้อำนวยความสะดวก รวดเร็ว และสามารถรู้ตำแหน่งได้อย่างถูกต้องและแม่นยำในการร้องขอความช่วยเหลือได้ในกรณีเร่งด่วน
- 1.2.7 เพื่อศึกษาและนำเอาเทคโนโลยี MapSever มาช่วยทำการสร้าง Web Server เพื่อทำการสร้างแผนที่และทำเป็นรูปภาพแล้วให้บริการแก่ โทรศัพท์เคลื่อนที่ (Mobile) เมื่อมีการร้องขอแผนที่

1.3 ประโยชน์ที่จะได้รับ

- 1.3.1 ได้รับความรู้และความเข้าใจเกี่ยวกับภาษา XML , GML และ SVG
- 1.3.2 เป็นประโยชน์ต่อบุคคลทั่วไปที่ต้องการความช่วยเหลือจากหน่วยงานของรัฐที่เกี่ยวข้อง ดังเช่น รถดับเพลิง , รถตำรวจ และรถพยาบาล เป็นต้น
- 1.3.3 เป็นประโยชน์ต่อหน่วยงานของรัฐที่เกี่ยวข้อง โดยจะช่วยลดงบประมาณแผ่นดิน จากค่าใช้จ่ายด้านน้ำมันเชื้อเพลิง
- 1.3.4 ได้รับความรู้และความเข้าใจเกี่ยวกับภาษาที่ใช้ในการเขียนเพื่อพัฒนาโทรศัพท์เคลื่อนที่ (Mobile)
- 1.3.5 ได้รับความรู้และความเข้าใจเกี่ยวกับระบบสารสนเทศภูมิศาสตร์
- 1.3.6 ได้รับความรู้และความเข้าใจเกี่ยวกับ MapSever ซึ่งใช้ในการแสดงผลที่เป็นแผนที่บนเว็บเบราว์เซอร์ได้ โดยได้ข้อมูลมาจากที่มีอยู่

1.4 ขอบเขตโครงการ

ปฏิญานพันธันี้มุ่งเน้นการพัฒนาาระบบที่ให้ความช่วยเหลือแก่สังคมหรือบรรเทาสธารณะภัย โดยอาศัยความก้าวหน้าของเทคโนโลยีทั้งเทคโนโลยีทางการติดต่อสื่อสารบนอินเทอร์เน็ต และเทคโนโลยีทางด้านโทรศัพท์เคลื่อนที่ ซึ่งระบบบรรเทาสธารณะภัยนี้สามารถทำงานได้ดังนี้

- 1.4.1 สามารถแสดงแผนที่กรุงเทพฯ ในตำแหน่งต่างๆ บนเว็บเบราว์เซอร์ได้
- 1.4.2 สามารถแสดงผลแผนที่แบบ SVG ได้
- 1.4.3 สามารถย่อหรือขยายแผนที่ได้
- 1.4.4 สามารถเลื่อนขอบเขตการแสดงผลแผนที่ได้
- 1.4.5 แผนที่มิปฏิสัมพันธ์กับผู้ใช้ เช่น สามารถแสดงข้อมูลสถานที่เมื่อมีการโฟกัสเมาส์บริเวณสถานทีนั้นๆ สักพัก
- 1.4.6 สามารถเลือกการแสดงผลแผนที่ของกรุงเทพฯ ให้แสดงเฉพาะรายละเอียดต่างๆ เช่น ต้องการให้แผนที่แสดงแค่ ถนนหลัก ซอย หรือ สถานที่ ก็ได้
- 1.4.7 สามารถค้นหาสถานที่ว่าอยู่ที่ตำแหน่งใด จากการป้อนชื่อสถานทีนั้น
- 1.4.8 สามารถเรียกรถฉุกเฉินได้ทั้งผ่าน โทรศัพท์เคลื่อนที่และส่วนของ Callcenter ที่เป็นเว็บเพจหรือโทรเข้ามาที่ Callcenter
- 1.4.9 สามารถหารถคันที่อยู่ใกล้ที่สุดได้ จากจุดที่มีการร้องขอความช่วยเหลือ ไปยังจุดที่หน่วยงานของรัฐที่เกี่ยวข้อง ดังเช่น รถดับเพลิง รถตำรวจ และรถพยาบาล เป็นต้น
- 1.4.10 สามารถสังเกตการณ์เส้นทางของ รถดับเพลิง , รถตำรวจ และรถพยาบาล ในกรณีที่เกิดเหตุฉุกเฉินได้บนเว็บเบราว์เซอร์ ว่าในขณะนั้น ได้เดินทางถึงตำแหน่งใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 วิธีการดำเนินงาน

- ปริญญาโทเริ่มต้นด้วยการศึกษาทฤษฎีพื้นฐานต่างๆที่เกี่ยวข้องกับปริญญาโท คือ
- 1.5.1 ศึกษาเทคโนโลยีและ การทำงานของระบบใน โครงการเก่า ที่มีข้อมูลที่เกี่ยวข้องกับระบบต่างๆ ที่ใช้ใน โครงการนี้
 - 1.5.2 ศึกษาเทคโนโลยีที่เกี่ยวข้องและจำเป็น สำหรับการนำไปใช้ในโครงการปัจจุบัน
 - 1.5.3 กำหนดขอบเขตของโครงการ
 - 1.5.4 ออกแบบโครงสร้างของโครงการ
 - 1.5.5 ทดลองนำแผนที่มาแสดง โดยใช้ MapServer
 - 1.5.6 ทดลองนำแผนที่ที่ได้รับมา มาแปลงให้อยู่ในรูปแบบของภาษา GML
 - 1.5.7 ทดลองเขียน โปรแกรมแสดงผลต่างๆ บน โทรศัพท์มือถือ
 - 1.5.8 ทดลองแสดงค่าพิกัด GPS จากโทรศัพท์มือถือ
 - 1.5.9 ทดลองแสดงผลแผนที่โดยใช้ภาษา SVG พร้อมทั้งเขียนฟังก์ชันการทำงานที่ใช้ในการแสดงผลของ SVG ด้วย JavaScript
 - 1.5.10 ออกแบบฐานข้อมูลจากข้อมูลแผนที่ที่ได้รับมา พร้อมทั้งข้อมูลเกี่ยวกับสถานที่ต่างๆ โดยใช้หลักการของ Spatial Database
 - 1.5.11 ทดลองเขียน โปรแกรมทำการติดต่อระหว่างกับฐานข้อมูลแผนที่ที่เป็น Spatial Database
 - 1.5.12 ทดลองเขียนหน้าเว็บ เพื่อแสดงผลแผนที่โดยใช้ภาษา SVG พร้อมกับดึงข้อมูลจาก Spatial Database
 - 1.5.13 ออกแบบหน้าเว็บต่างๆ ของเว็บเซิร์ฟเวอร์ที่มีการติดต่อกันระหว่างเว็บเซิร์ฟเวอร์กับฐานข้อมูลแผนที่ที่เป็น Spatial Database
 - 1.5.14 ปรับปรุงข้อผิดพลาดของแอปพลิเคชันและระบบที่เกี่ยวข้อง และปรับปรุงรูปแบบการแสดงผลของแผนที่ให้มีความถูกต้องและความเหมาะสมมากขึ้น
 - 1.5.15 เพิ่มประสิทธิภาพให้กับแอปพลิเคชันเช่น ปรับปรุงให้แผนที่ในรูปแบบ SVG ให้มีประสิทธิภาพกับผู้ใช้งานมากขึ้น

หลังจากนั้น จะเป็นการทดสอบการทำงานของระบบให้บริการบรรเทาสาธารณภัย โดยจะมีการทดสอบทั้งบนเว็บเบราว์เซอร์และบนโทรศัพท์เคลื่อนที่ ตามหน้าที่ต่างๆที่กำหนดไว้ซึ่งจะอยู่ในบทที่ 5 และสุดท้ายบทวิเคราะห์และสรุปจะอยู่ในบทที่ 6 เป็นบทวิจารณ์ และสุดท้ายทำการสรุปผลและจัดทำเอกสารประกอบ

1.6 ส่วนประกอบของรายงาน

รายงานฉบับนี้ได้แบ่งเนื้อหาออกเป็น 5 บทด้วยกันคือ

เนื้อหาในบทที่ 1 กล่าวถึงความสำคัญและที่มาของโครงการ วัตถุประสงค์ของโครงการ ประโยชน์ที่คาดว่าจะได้รับ ขอบเขตของโครงการ วิธีการดำเนินการ และส่วนประกอบของรายงานฉบับนี้

เนื้อหาในบทที่ 2 กล่าวถึงภาษา Markup Language ที่เกี่ยวข้องกับการพัฒนาแผนที่ โดยมีรายละเอียดย่อยๆ ดังเช่น ภาษา XML (Extensible Markup Language) ภาษา GML (Geography Markup Language) ภาษา SVG (Scalable Vector Graphics Language) เป็นต้น

เนื้อหาในบทที่ 3 กล่าวถึงระบบฐานข้อมูลเชิงพื้นที่ (Spatial database system) Model of spatial information Spatial Data Type Spatial Relationships

เนื้อหาในบทที่ 4 กล่าวถึงภาษา BREW ซึ่งเป็นภาษาที่ใช้ในการพัฒนา Application บนระบบโทรศัพท์เคลื่อนที่ ที่เป็นมีระบบ CDMA โดยมีรายละเอียดย่อยๆ ดังเช่น หลักการพื้นฐานของ BREW Platform ระบบ BREW โครงสร้างของ BREW ความแตกต่างระหว่าง Handset กับ Emulator เป็นต้น หลักการเขียนโปรแกรม BREW Application และส่วนสุดท้ายจะกล่าวถึงเรื่อง MapServer ซึ่งเป็นการแสดงผลแผนที่บนเว็บเบราว์เซอร์ เล็กน้อย

เนื้อหาในบทที่ 5 กล่าวถึงการออกแบบระบบทั้งหมด ของระบบเรียลไทม์จิกจิก พร้อมทั้งการสร้างในแต่ละระบบ

เนื้อหาในบทที่ 6 กล่าวถึงการทดลองและผลการทดลอง ของส่วนที่ได้ทำการพัฒนาขึ้น รวมทั้งผลการทดลองของระบบทั้งหมด ซึ่งส่วนหลักๆ มีอยู่ 2 ส่วนคือ ส่วนของการทำงานหลักๆ ของ Web Application และอีกส่วนคือส่วนการทำงานของโทรศัพท์เคลื่อนที่

เนื้อหาในบทที่ 7 เป็นการกล่าวถึงบทวิจารณ์และสรุป ซึ่งกล่าวถึงบทสรุปของโครงการ วิจารณ์สิ่งที่ได้รับจากโครงการ ปัญหาที่พบในการพัฒนาและแนวทางการแก้ไข

บทที่ 2

ภาษา XML และภาษา Markup ที่เกี่ยวข้องกับการพัฒนาแผนที่

2.1 ภาษา XML (Extensible Markup Language)

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

XML และ HTML เป็นส่วนหนึ่งของภาษา SGML ซึ่ง XML จะให้รายละเอียดเกี่ยวกับข้อมูล เช่น ชื่อเมือง อุณหภูมิ ความกดอากาศ ส่วน HTML เป็นการกำหนด Tag ต่างๆ ที่จะกำหนดรูปแบบการแสดงผลของข้อมูลบนหน้าเว็บ ซึ่งข้อมูลจะสามารถแสดงออกมาได้หลายรูปแบบขึ้นอยู่กับข้อกำหนดของ HTML

2.1.1 ลักษณะโครงสร้างของ XML

XML เป็นการใช้อ็ควมเพื่อบ่งบอกโครงสร้างของเอกสาร พิจารณาตัวอย่างรูปแบบโครงสร้างของหนังสือ 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
```

หนังสือที่มีอยู่ในปัจจุบันจะมีโครงสร้างที่มีรายละเอียดที่ซับซ้อนมากกว่านี้ เช่น บทนำ, สารบัญ เป็นต้น เช่นเดียวกัน ภายในส่วนเนื้อความยังประกอบด้วยโครงสร้างย่อยคือ ย่อหน้า (Paragraph) แต่ละย่อหน้ายังประกอบขึ้นจากประโยค คำและตัวอักษรด้วยลักษณะของเอกสาร XML นั้น สามารถอธิบายโดยใช้ตัวอย่างที่ 1 ได้ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างที่ 1

```
<?xml version="1.0" encoding="windows-874"?>
<note>
  <to> Tiny </to>
  <from>Michael</from>
  <heading>กรุณาติดต่อ Mr.Michael Tiny</heading>
  <body>Mr.Tiny โทรศัพท์ทางไกลมาบอกว่ามีปัญหาเรื่องภาษีศุลกากร ติดต่อกลับ
</body>
</note>
```

บรรทัดที่ 1 นั้นหมายความว่าเราประกาศเอกสารนี้เป็นเอกสาร XML เวอร์ชันที่ 1.0 และมีการเข้ารหัสอักขระเป็น windows-874 เพื่อให้ใช้ภาษาไทยได้ บรรทัดที่ 2 คือ Element ตัวแรกหรือตัวแม่ของเอกสาร หรือที่เรียกว่า root element ที่ทุกๆ เอกสาร XML ต้องมีโดยที่เอกสารหนึ่งเอกสารมีได้เพียง root element เดียวเท่านั้น บรรทัดที่ 3 ถึง 6 มี element 4 ตัวคือ to, from, heading และ body ซึ่งเป็น element ลูกของ element note ส่วนบรรทัดสุดท้ายก็คือแท็กปิดของ element แม่ตัวเองและทุกๆ element ในเอกสาร XML ต้องมีทั้งแท็กเปิดและแท็กปิด

2.1.1.1 Tag

tag ใน XML มีความหมายในลักษณะเดียวกับที่ใช้ใน HTML Tag คือข้อความที่อยู่ระหว่างสัญลักษณ์ < และ > ส่วนชนิดของ tag จะมีทั้ง tag เปิดและ tag ปิดแต่ก็จะมีที่เป็นแท็กเปิดและแท็กปิดอยู่ในตัวเดียวกันด้วยเช่น <ชื่อแท็ก /> ซึ่งแท็กเปิดเป็นตัวบอกว่าเริ่ม element และแท็กปิดเป็นตัวบอกว่าจบ element และความแตกต่างระหว่างแท็กเปิดกับแท็กปิดคือแท็กเปิดสามารถใส่ข้อมูลอธิบายเพิ่มเติมได้ซึ่งเรียกว่า attribute แต่แท็กปิดไม่สามารถทำได้

2.1.1.2 Element

Element เป็นส่วนขยายอธิบายความหมายเพิ่มเติมอีกและมีความสัมพันธ์แบบ element แม่กับ element ลูกใน element สามารถใส่ข้อมูลเข้าไปได้โดยข้อมูลที่ใส่เข้าไประหว่างแท็กเปิด กับแท็กปิดเรียกว่า Content และข้อมูลที่ใส่เพิ่มเข้าไปในแท็กเปิดเพื่ออธิบายคุณสมบัติลักษณะของ element เพิ่มเรียกว่า Attribute ทุก element ในเอกสารต้องมีทั้งแท็กเปิดและแท็กปิด

2.1.1.4 Content

เนื้อความหรือ Content ถือได้ว่าเป็นข้อมูลเพื่อใช้ในการแสดงให้ผู้อ่านเอกสารได้เห็น หรือ กล่าวอีกนัยหนึ่งคือ Content อยู่หลัง Tag เปิด และจบก่อนที่จะถึง Tag ปิดนั่นเอง

2.1.1.3 Attribute

Element ในเอกสาร XML สามารถมี Attribute ในแท็กเปิดได้และมีได้มากกว่า 1 ตัวเหมือนกับในเอกสาร HTML โดยที่ Attribute จะอธิบายถึงคุณสมบัติหรือบอกลักษณะของ element นั้นเพิ่ม คือถ้ากล่าวถึงสิ่งของหากเราไม่ได้ให้ความหมายเพิ่มเติมเราก็จะไม่มีทางรู้หรือแยกแยะสิ่งที่คล้ายๆ กันออกจากกันได้ เช่น ถ้าพูดถึง บท ในหนังสือ แค่นี้เป็นความหมายโดยรวม แต่ถ้าบอกว่าบทที่ 1 ในหนังสือ เลข 1 ในที่นี้คือความหมายเพิ่มเติมให้กับบท

2.1.2 ตัวกำหนดโครงสร้างของ XML ด้วย DTD (Document Type Definition)

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

2.1.2.1 รูปแบบของ DTD

DTD มีรูปแบบทั่วไปตามนี้ `<!DOCTYPE Name DTD>` Name ตามตัวอย่างเป็นที่กำหนดชื่อของ Document Element (หรือ Root Element) ชื่อของ Document Element จะต้องตรงกับชื่อที่กำหนดไว้ที่นั่น DTD สามารถบรรจุชนิดของการประกาศ Markup ไว้ดังนี้ การประกาศ Element Type

โดยมีรูปแบบการประกาศ Element Type ดังนี้

```
<! ELEMENT ชื่อelement (รายละเอียดในelement)>
```

รายละเอียดใน Element เป็นที่กำหนดว่าภายใน Element สามารถบรรจุอะไรได้บ้าง ด้านล่างนี้จะเป็นอย่างของเอกสาร XML ที่มี Element Type 2 ชนิด และ Element Type ที่ประกาศชื่อ COLLECTION แสดงให้เห็นว่าสามารถบรรจุ Element ที่ชื่อ CD ได้ มากกว่า 1 Element และ Element Type ที่ชื่อ CD กำหนดว่าสามารถบรรจุได้เฉพาะข้อมูลประเภท Character Data เท่านั้น

```
<?xml version="1.0"?>
<!DOCTYPE COLLECTION
[
    <!ELEMENT COLLECTION (CD) +>
    <!ELEMENT CD (#PCDATA)
]
>
<COLLECTION>
    <CD>Mozart Violin Concertos 1, 2, and 3</CD>
    <CD>Telemann Trumpet Concertos</CD>
    <CD>Handel Concerti Grossi</CD>
</COLLECTION>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การประกาศ Attribute

โดยมีรูปแบบของการประกาศดังนี้

<! ATTLIST ชื่อของ element ชื่อของแอตทริบิวต์ ชนิดของแอตทริบิวต์ ค่าดีฟอลต์ >

ชนิดของแอตทริบิวต์มีได้หลายอย่าง ดังตารางที่ 2-1

ชนิดของแอตทริบิวต์	คำอธิบาย
CDATA	ค่าที่เป็นข้อมูลแบบ Character Data (ไม่กระจายค่าในพาสเซอร์)
(eval eval ...)	ค่านี้จะป็นกลุ่มของค่าที่เตรียมไว้ให้เลือก
ID	ค่านี้เป็นเลขประจำตัวที่ไม่ซ้ำกับใคร
IDREF	ค่านี้เป็นเลขประจำตัวของelementหนึ่ง
IDREFS	ค่านี้เป็นลิสต์ของเลขประจำตัวอื่นๆ
NMTOKEN	ค่านี้เป็นชื่อที่ถูกต้องตามหลัก XML
NMTOKENS	ค่านี้เป็นลิสต์ของชื่อที่ถูกต้องตามหลัก XML
ENTITY	ค่านี้เป็นเอนติตี
ENTITIES	ค่านี้เป็นลิสต์ของเอนติตี
NOTATION	ค่านี้เป็นชื่อของ Notation ที่ได้ประกาศไว้
Xml:	ค่านี้เป็นค่าที่มีการกำหนดไว้ก่อนแล้ว

ตารางที่ 2-1 แสดงชนิดของแอตทริบิวต์ที่สามารถใช้ในการประกาศแอตทริบิวต์

ส่วนที่เป็นค่าดีฟอลต์สามารถกำหนดได้ดังตารางที่ 2-2

ค่าที่กำหนด	คำอธิบาย
#DEFAULT value	แอตทริบิวต์นี้มีค่าดีฟอลต์และระบุมาให้
#REQUIRED	ต้องมีค่าแอตทริบิวต์ให้กับelement
#IMPLIED	ค่าแอตทริบิวต์อาจไม่จำเป็นต้องให้ไว้ก็ได้
#FIXED value	ค่าของแอตทริบิวต์เป็นค่าที่กำหนดไว้ตายตัว ผู้ใช้เปลี่ยนไม่ได้

ตารางที่ 2-2 แสดงค่าที่สามารถได้ในส่วนที่เป็นค่าดีฟอลต์ในการประกาศแอตทริบิวต์

2.1.3 การแปลงรูปแบบเอกสาร XML (Transformation XML)

XSL ซึ่งย่อมาจาก Extensible Style sheet Language (XSL) เป็นภาษาที่ใช้ในการแปลงจากรูปแบบ(Transformation) ของเอกสาร XML ไปอยู่ในรูปแบบอื่นๆ และในขณะเดียวกันก็ทำการประยุกต์รูปแบบการจัดข้อความ(style) ด้วย XSL stylesheet (สไตล์ชีต) ถูกเขียนในรูปแบบที่เหมาะสมกับ XML แต่ต้องทำการกำหนดลำดับของ Element ไว้ล่วงหน้าในการกำหนดสิ่งที่ต้องการให้กระทำในการแปลงรูปแบบโดยความหมายจะถูกกำหนดโดยตัวประมวลผล XSL เอง ซึ่งควรจะเป็นไปตามมาตรฐานของ W3C ดังนั้นจาก XSL ซึ่งเป็นแอปพลิเคชันหนึ่งของ XML และอยู่ในรูปแบบที่เหมาะสมกับ XML ดังนั้นจึงสามารถประมวลผล XSL สไตล์ชีตได้เหมือนกับเอกสาร XML ครั้งแรก XSL เป็นข้อเสนอที่ถูกระบุทั้งทางที่การแปลงรูปแบบ XML และไวยากรณ์ในการจัดการรูปแบบบนแพลตฟอร์มที่เป็นอิสระกัน (platform-independent styling grammar) แต่อย่างไรก็ตามในส่วนของการแปลงรูปแบบกำหนดอย่างรวดเร็วจึงเป็นประโยชน์อย่างมาก ขณะที่ไม่มีแอปพลิเคชันในส่วนของการจัดรูปแบบมากนักและส่วนใหญ่จะกล่าวว่ามีประโยชน์ทางการทำงานน้อย ดังนั้นจากข้อเสนอในตอนแรกจึงสามารถแบ่งออกได้เป็น 2 ส่วนที่แตกต่างกันคือ ส่วนในการแปลงรูปแบบเอกสาร ซึ่งปัจจุบันถูกเรียกว่า XSLT (XSL Transformation Language) แต่ในที่นี้จะพิจารณาเฉพาะในส่วนของการแปลงรูปแบบเท่านั้นเพื่อแปลงรูปแบบจากเอกสาร XML แบบหนึ่งไปเป็น XML อีกแบบหนึ่ง เช่น ใช้ในการแปลงเอกสาร GML ไปเป็น SVG

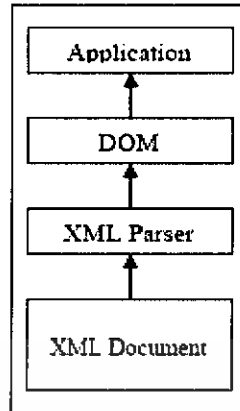
2.1.3.1 The XSL Transformation Language (XSLT)

XSLT ถูกกำหนดโดยองค์กร W3C ในการกำหนดชุด Element ของ XML ที่สามารถจะถูกใช้ในการสร้างสไตล์ชีตที่แปลงเอกสาร XML ให้อยู่ในรูปแบบใดๆ ซึ่งในปัจจุบันนี้ใช้ในการแปลง XML รูปแบบหนึ่งไปเป็นอีกรูปแบบหนึ่ง โดยการทำงานจะทำโดยนำเอกสาร XML ที่เหมาะสมกับ XML นั้นไปทำการแปลงรูปแบบเพื่อสร้างเอกสารเอาต์พุต ซึ่งขึ้นอยู่กับคำสั่งและเนื้อหาภายใน สไตล์ชีตจึงสามารถสร้างรูปแบบเอาต์พุตใดๆ ก็ได้ตามที่ต้องการในการทำงาน

2.1.4 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

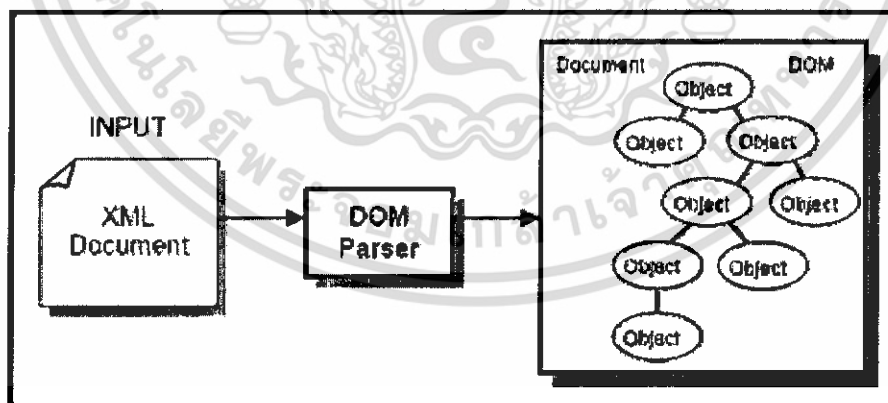
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-1 การใช้งาน DOM

DOM Parsing

DOM (Document Object Model) เป็นการทำงานแบบ Tree-Based parser และใช้การเข้าถึงข้อมูลด้วยวิธีการ Random-Access คือจะประมวลโครงสร้างของเอกสาร XML ให้เป็นโครงสร้างต้นไม้(Tree structure) แบบ hierarchical object model ซึ่งในโครงสร้างต้นไม้จะประกอบไปด้วยอีลิเมนต์โหนด (element node) โดยภายในแต่ละอีลิเมนต์โหนด ประกอบไปด้วยข้อมูลที่เกี่ยวข้องกับอีลิเมนต์นั้นตามโครงสร้างและข้อมูลในเอกสาร XML ซึ่งจะประกอบไปด้วยข้อมูลทั้งหมดของเอกสาร XML เพื่อให้แอปพลิเคชันสามารถเข้าหาจุดต่างๆ ของ tree structure ได้ โดยที่ DOM จะโหลด XML ทั้งหมด (ข้อมูลในRoot Node ทั้งหมด) เข้ามาเป็น Tree ในหน่วยความจำหลักของคอมพิวเตอร์ (memory) ก่อนจึงทำงานได้ดังแสดงในรูปที่ 2-2



รูปที่ 2-2 แสดงวิธีการเข้าถึงข้อมูลโดยวิธี DOM Parsing

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ภาษา Markup Language ที่เกี่ยวข้องกับการพัฒนาแผนที่

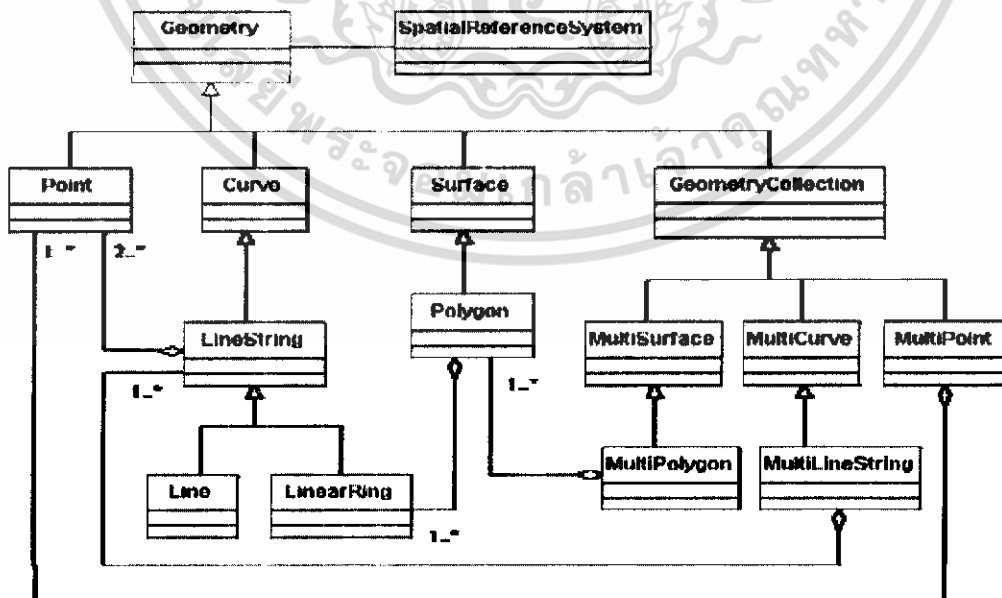
องค์กร OGC (OpenGIS Consortium) ได้กำหนดรูปแบบในการจัดเก็บและการแลกเปลี่ยนข้อมูลทางด้านภูมิศาสตร์ พร้อมทั้งความสัมพันธ์และลักษณะประจำตัวของข้อมูล ในรูปแบบที่เรียกว่า GML(Geography Markup Language) ซึ่งเป็นภาษา Markup Language รูปแบบหนึ่งที่มีพื้นฐานโครงสร้างที่อยู่ในรูปแบบของ XML (Extensible Markup Language) แต่รูปแบบการจัดเก็บและแลกเปลี่ยนข้อมูลทางด้านภูมิศาสตร์ในรูปแบบของ GML นั้นก็เป็นเพียงแค่ตัวข้อมูลที่ใช้ในการจัดเก็บเท่านั้น ไม่สามารถที่จะนำมาแสดงผลได้ ซึ่งการจะนำมาแสดงผลบน web browser ได้ นั้นจำเป็นต้องแปลงรูปแบบให้ไปอยู่ในรูปแบบที่สามารถแสดงผลได้ก่อน ซึ่งหนึ่งวิธีการในการแสดงผลของ GML บน web browser นั้นคือการแปลงให้อยู่ในรูปแบบของ SVG (Scalable Vector Graphics Language) ซึ่งเป็นภาษา Markup Language อีกรูปแบบหนึ่งที่ใช้ในการแสดงผลได้

2.2.1 ภาษา GML (Geography Markup Language)

ภาษา GML เป็นรูปแบบหนึ่งของเอกสาร XML ที่ถูกออกแบบขึ้นเพื่อการจัดเก็บและการแลกเปลี่ยนข้อมูลทางภูมิศาสตร์ ทั้งทางด้านความสัมพันธ์และลักษณะประจำตัวของข้อมูลในรูปแบบของ feature ที่แตกต่างกันและสามารถกำหนดวิธีที่จะแสดงข้อมูลทางภูมิศาสตร์นั้นโดยใช้ข้อกำหนดของ GML ที่ถูกรับรองโดยองค์กร OGC (OpenGIS Consortium) ในการอธิบายข้อมูลเชิงพื้นที่เพื่อให้ผู้ที่ทำงานด้าน GIS (Geographic Information System) ใช้เป็นพื้นฐานในการออกแบบพัฒนาแผนที่ในรูปแบบต่างๆ

2.2.1.1 GML Basic concepts

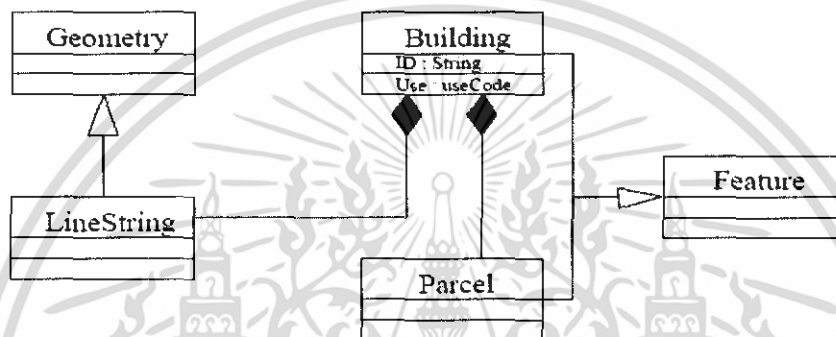
GML เป็นรูปแบบหนึ่งของเอกสาร XML ที่ถูกออกแบบขึ้นเพื่อการจัดเก็บและการแลกเปลี่ยนข้อมูลทางภูมิศาสตร์



รูปที่ 2-3 The geometry model for simple feature (Open GIS Consortium, 2001)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

องค์ประกอบทางเรขาคณิต (Geographic feature) จะถูกอธิบายด้วยคุณสมบัติต่างๆ ซึ่งคุณสมบัติเหล่านี้อาจจะประกอบด้วยคุณสมบัติที่เกี่ยวข้องกับระยะทาง โดยอธิบายตำแหน่งและรูปร่างของ feature นั้นๆ แต่ละ feature จะมีชนิดของ feature ซึ่งเทียบเท่ากับคลาสใน object โมเดล ดังนั้น Class definition จึงเป็นส่วนที่กำหนดคุณสมบัติของแต่ละ feature จำเป็นต้องมี เช่น อาคารที่ต้องมีการกำหนดให้มีชื่อของอาคาร ซึ่งเป็นข้อมูลประเภท String และเส้นรอบนอกซึ่งกำหนดให้เป็นรูปวงแหวน (LinearRing) ซึ่งเป็นคุณสมบัติทางภูมิศาสตร์ โดยรูปด้านบนอธิบายถึง Geographic feature โดยใช้ UML



รูปที่ 2-4 Features and properties

2.2.1.2 Basic geometric properties

รูปทรงเรขาคณิตพื้นฐานที่มีการนิยามไว้ใน GML มีดังตารางที่ 2-3

ชื่อ	อธิบาย	ชนิดของเรขาคณิต
boundedBy	-	Box
pointProperty	location, position, centerOf	Point
lineStringProperty	CenterLineOf	LineString
polygonProperty	coverage	Polygon
geometryProperty	-	-
multiPointProperty	multiLocation, multiPosition, multiCenterOf	MultiPoint
multiLineStringProperty	multiCenterLineOf	MultiLineString
multiPolygonProperty	multiCoverage	MultiPolygon
multiGeometryProperty	-	MultiGeometry

ตารางที่ 2-3 แสดงรูปทรงเรขาคณิตพื้นฐานที่ถูกนิยามไว้ใน GML

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1.3 Encoding Geometry

สิ่งที่สำคัญที่สุดในมาตรฐานของการแปลงข้อมูลทางภูมิศาสตร์ คือความสามารถในการแสดงรูปร่างทางเรขาคณิต ในการแสดงภาพทางภูมิศาสตร์ที่เกี่ยวข้องกับระยะทางโดยใช้ GML 2.0 นั้นจำเป็นที่จะต้องอ้างอิงถึง XML Schema ทั้งหมด 2 Schema คือ GML Feature Schema และ GML Geometry Schema

1. GML Feature Collections

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

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

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

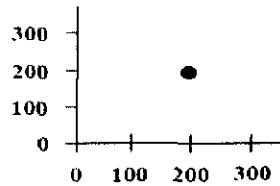
ส่วนที่สำคัญของ GIS คือวิธีการในการอ้างอิงถึงองค์ประกอบทางภูมิศาสตร์ต่างๆ บนพื้นโลกคู่ลำดับ (coordinate) สำหรับรูปเรขาคณิตที่เราใช้แสดงแทนบนแผนที่นั้นจะอ้างอิงโดย Spatial Reference System (SRS) และทุกองค์ประกอบประกอบจะต้องระบุ SRS หรืออีกวิธีหนึ่งคือระบุ GID Attribute ซึ่งเป็น Attribute ที่จะแสดงค่าเฉพาะตัวทางภูมิศาสตร์ของแต่ละองค์ประกอบเรขาคณิตนั้น

Points เป็นรูปเรขาคณิตพื้นฐานที่ ประกอบด้วยคู่ลำดับเพียงคู่ลำดับเดียว multipoint เป็นเซตที่ประกอบด้วยจุดหลายๆ จุด

ตัวอย่าง Code

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

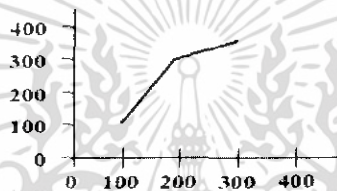


รูปที่ 2-5 แสดงรูปที่สร้างโดย Point

LineString ประกอบด้วย list ของคู่ลำดับแสดงที่ส่วนของเส้นตรงหลายเส้นเชื่อมต่อกัน รูปด้านล่าง LineString จะต้องประกอบด้วยคู่ลำดับอย่างน้อย 2 คู่

ตัวอย่าง Code

```
<LineString gid="2468" srsName="gml_srs">
  <coord><x>100.0</x><y>100.0</y></coord>
  <coord><x>200.0</x><y>300.0</y></coord>
  <coord><x>300.0</x><y>350.0</y></coord>
</LineString>
```



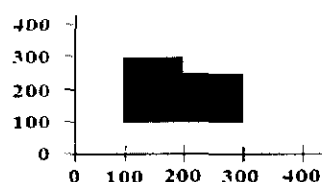
รูปที่ 2-6 แสดงรูปที่สร้างโดย LineString

LinearRing เป็นรูปปิด ประกอบด้วย list ของคู่ลำดับที่แสดงถึงส่วนของเส้นตรง คู่ลำดับสุดท้ายจะต้องเท่ากับคู่ลำดับแรก และต้องประกอบด้วยคู่ลำดับอย่างน้อย 4 คู่

Polygons เป็นพื้นที่ผิวที่ผิวของรูปเป็นเซตของ LinearRing โดยขอบนอกของ Polygon คือ LinearRing วงแรกและ LinearRing วงที่เหลือจะซ้อนเรียงกันภายในของวงแรก

ตัวอย่าง Code

```
<Polygon gid="1234" srsName="gml_srs">
  <outerBoundaryIs>
    <LinearRing>
      <coord><x>100.0</x><y>100.0</y>
      <coord><x>100.0</x><y>300.0</y>
      <coord><x>200.0</x><y>300.0</y>
      <coord><x>200.0</x><y>200.0</y>
      <coord><x>300.0</x><y>200.0</y>
      <coord><x>300.0</x><y>100.0</y>
      <coord><x>100.0</x><y>100.0</y>
    </LinearRing>
  </outerBoundaryIs>
</Polygon>
```



รูปที่ 2-7 แสดงรูปที่สร้างโดย Polygon

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. GML application schema

Schema คือรูปแบบการจัดโครงสร้างและความสัมพันธ์ของข้อมูล โดยภาษา GML นั้นกำหนดให้ผู้ใช้ต้องกำหนด GML Application schema ซึ่งเขียนโดยใช้หลักการของ XML Schema และเป็นไปตามข้อบังคับของ GML Application Schema ประกอบด้วย

- Annotations ซึ่งเป็นส่วนประกาศหรือส่วนที่ใช้การอธิบาย schema นั้น
- Simple Type คือชนิดของข้อมูลที่ใช้รูปแบบของข้อมูลที่ GML กำหนดมาให้แล้ว
- Complex Type คือชนิดของข้อมูลที่สร้างขึ้นใหม่โดยอาจจะเกิดจากการใช้ simple type หลายๆ ตัวมาประกอบกัน หรือเกิดจาก Complex type ชนิดอื่นก็ได้
- Element
- Attribute

ตัวอย่าง GML Feature Type Schema

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

ข้อดีของการใช้ GML ในการอธิบายแผนที่

1. ได้แผนที่ที่มีคุณภาพดีกว่า เนื่องจาก GML จะเปลี่ยนข้อมูลทางภูมิศาสตร์ทั้งหมดให้อยู่ใน รูปของ Feature หรือ object ซึ่งสามารถนำไปแสดงผลในรูปแบบที่ไม่จำกัดความละเอียดในการแสดงผลได้
2. สามารถใช้งานได้บนบราวเซอร์โดยไม่ต้องซื้อ โปรแกรมที่ใช้แสดงผลทางฝั่งไคลเอนท์ เมื่อข้อมูล GML ถูกรับโดยฝั่งไคลเอนท์จะถูกเปลี่ยนให้อยู่ในรูปของภาพแผนที่บนบราวเซอร์โดยใช้ภาษา SVG (Scalable Vector Graphics) ในการแสดงออกมาเป็นภาพ ซึ่งเราสามารถแสดงแผนที่บนบราวเซอร์โดยไม่จำเป็นต้องติดตั้งซอฟต์แวร์อื่นๆ เพิ่มเติมหากบราวเซอร์สามารถรองรับภาพแบบเวกเตอร์ (Vector graphics) ได้ และเราสามารถดาวน์โหลด SVG Viewer ซึ่งเป็นปลั๊กอินที่ใช้สำหรับดูภาพแบบ SVG ของบริษัท Adobe ได้โดยไม่ต้องเสียค่าใช้จ่ายใดๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. สามารถสร้างแผนที่ที่มีรูปแบบเฉพาะได้ เนื่องจาก GML จะเก็บแค่ข้อมูลของแผนที่ เช่นที่ตั้งของสถานที่นั้น, ลักษณะทางภูมิศาสตร์ ฉะนั้นแล้วการแสดงผลก็ขึ้นอยู่กับความพอใจของผู้พัฒนาระบบนั้นๆ
4. สามารถทำการแก้ไขแผนที่ได้ แผนที่ที่เขียนโดย GML จะถูกดาวน์โหลดและทำการ render บนเบราว์เซอร์ซึ่งเมื่อ GML ถูกเปลี่ยนเป็น SVG ผู้ใช้จะสามารถใช้โปรแกรมบนไคลเอนท์ในการเพิ่มเติมตัวอักษรหรือทำการเปลี่ยนแปลงภาพบนแผนที่ และสามารถบันทึกเก็บเป็นไฟล์ภาพเพื่อนำไปใช้ในกรณีอื่นได้
5. สามารถกำหนดเนื้อหาที่ต้องการแสดงหรือต้องการได้รับ เพราะ GML เป็นภาษาที่อ้างอิงถึง feature ทำให้ผู้ใช้สามารถกำหนดได้ว่าต้องการให้มี feature ใดแสดงบนแผนที่บ้างซึ่งจะช่วยลดระยะเวลาในการรับ
6. สามารถแสดงผลได้บนทุกอุปกรณ์ที่ใช้เบราว์เซอร์ได้ ข้อมูลทางภูมิศาสตร์ใน GML นั้นสามารถส่งไปยังอุปกรณ์ชนิดใดก็ได้ที่สามารถรองรับ XML การแสดงผลแบบ XML ได้ เช่น เครื่องพีดีเอหรือโทรศัพท์มือถือ ทำให้แผนที่นั้นสามารถใช้งานได้อย่างกว้างขวางและหลากหลายยิ่งขึ้น
7. สามารถเชื่อมโยง link ลงไปยังในแผนที่ได้ ประโยชน์ที่สำคัญของ GML คือผู้ใช้สามารถฝัง URL ลงใน feature ที่มีในแผนที่ได้ ความสามารถนี้ทำให้เราสามารถออกแบบแผนที่ให้มีความแปลกใหม่และทันสมัยยิ่งขึ้น
8. ทำการ query ได้ดียิ่งขึ้น หากผู้ใช้ต้องการที่จะคลิกไปบน feature บนแผนที่และค้นหาข้อมูลเพิ่มเติมเกี่ยวกับ feature นั้นๆ (เช่น ชื่อหรือความลึกของแม่น้ำ) ซึ่งไม่สามารถทำได้บนแผนที่ที่สร้างขึ้นโดยใช้ไฟล์ภาพ (gif/jpg) สำหรับแผนที่ที่ใช้ GML นั้น เราจะสามารถระบุ feature ที่ถูกคลิกนั้นได้

2.2.2 ภาษา SVG (Scalable Vector Graphics Language)

SVG เป็นภาษาที่ใช้อธิบายภาพกราฟิก 2 มิติและอยู่ในรูปแบบ XML และกราฟิกที่อธิบายนั้นเป็นแบบเวกเตอร์ด้วย สามารถจัดกลุ่ม กำหนดลักษณะ และอื่นๆ ของการอธิบายรูปกราฟิก และสามารถที่จะทำให้กราฟิกนั้นมีการทำงานที่มีการปฏิสัมพันธ์กับผู้ใช้งาน โดยใช้ภาษาจาวาสคริปต์ต่างๆ เข้าช่วยและสามารถทำงานในรูปแบบแอนิเมชันได้อีกด้วย

และเนื่องจาก SVG เป็น XML ดังนั้นจึงมีการสร้าง DOM สำหรับ SVG ขึ้นมาโดยอ้างอิงตามมาตรฐานของ W3C ซึ่งทำให้ SVG มีคุณสมบัติไดนามิกและการปฏิสัมพันธ์ตามไปด้วย แม้แต่การจัดการเหตุการณ์ที่เกิดขึ้นที่ตัวกราฟิก อย่างเช่น onmouseover() หรือ onclick() ก็สามารถกำหนดลงในวัตถุของ SVG ได้อีกด้วย โดยรายละเอียดเพิ่มเติมสามารถศึกษาได้จาก

<http://www.w3.org/TR/SVG/index.html#minitoc>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปะเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2.1 การอธิบายภาพโดยใช้ภาษา SVG

ลักษณะการเขียนภาษา SVG มีลักษณะเป็น XML ซึ่งต้องมี tag เปิดและปิด และมี tag ที่อยู่ใน tag ด้วย ซึ่งจะสังเกตได้ว่า SVG ไฟล์ จะต้องมี <svg> อยู่เป็น tag นอกสุดหรือเป็น root element ซึ่งมีตัวอย่างดังต่อไปนี้

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 20000303 Stylable//EN"
"http://www.w3.org/TR/2003/WD-SVG-20000303/DTD/svg-20000303-
stylable.dtd">
<svg xml:space="preserve" width="5.5in" height=".5in">
...
...
...
</svg>
```

ในบรรทัดแรกจะเป็นการบอกให้ผู้ที่น่า SVG ไฟล์นี้ไปอ่านทราบว่าเอกสารต่อไปนี้อ้างอิงจากมาตรฐาน XML เวอร์ชัน 1.0 โดยข้อมูลที่อยู่ในแต่ละแท็กจะใช้มาตรฐาน iso-8859-1 ในการเขียนบรรทัดที่สองจะอ้างอิงถึง DTD หรือ Document Type Definition (DTD เป็นเอกสารที่ใช้บอกถึงชนิดของข้อมูลที่จะใช้ใน XML และยังระบุอีกด้วยว่าเอกสารนั้นๆ มีโครงสร้างเป็นอย่างไร) ใน <svg> Element สามารถที่จะบรรจุ text, shapes และ paths ได้ เช่น

Vector Graphic Shapes

<rect> : ใช้สำหรับสร้างและปรับแต่งรูปสี่เหลี่ยม

ตัวอย่าง

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

คำอธิบาย แอตทริบิวต์

x, y : ใช้กำหนดตำแหน่งของรูปสี่เหลี่ยมที่สร้าง

rx, ry : ใช้กำหนดส่วนโค้งบริเวณมุมสี่เหลี่ยม

width, height : ใช้กำหนดขนาดความกว้างและความสูงตามลำดับ

style : ใช้ในการกำหนดลักษณะของวัตถุ

fill, stroke : ใช้กำหนดสีของวัตถุและสีของเส้นขอบวัตถุตามลำดับ

stroke-width : ใช้กำหนดความหนาของเส้น

opacity : ใช้กำหนดความโปร่งใสของวัตถุมีค่าระหว่าง 0-1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<circle> : ใช้สำหรับสร้างและปรับแต่งรูปวงกลม

ตัวอย่าง

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

คำอธิบายแอตทริบิวต์

cx, cy : ใช้กำหนดจุดศูนย์กลางของวงกลม หากไม่กำหนดจะมีค่าเป็น 0, 0

r : ใช้กำหนดรัศมีของวงกลม

<ellipse> : ใช้สำหรับสร้างและปรับแต่งรูปวงรี

ตัวอย่าง

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

คำอธิบาย แอตทริบิวต์

cx, cy : ใช้กำหนดจุดศูนย์กลางของวงกลม หากไม่กำหนดจะมีค่าเป็น 0, 0

rx, ry : ใช้กำหนดรัศมีในแนวนอนและแนวตั้งตามลำดับ

<line> : ใช้สำหรับสร้างและปรับแต่งเส้นตรง

ตัวอย่าง

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

คำอธิบาย แอตทริบิวต์

x1, y1 : ใช้กำหนดตำแหน่งจุดเริ่มต้นของเส้นตรง

x2, y2 : ใช้กำหนดตำแหน่งจุดสิ้นสุดของเส้นตรง

<polygon> : ใช้สำหรับสร้างและปรับแต่งรูปหลายเหลี่ยม

ตัวอย่าง

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

คำอธิบาย แอตทริบิวต์

point ใช้กำหนดตำแหน่งของมุมแต่ละมุม (x,y)

<polyline> : ใช้สำหรับสร้างและปรับแต่งรูปเส้นตรงหลายเส้นเชื่อมต่อกัน

ตัวอย่าง

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

คำอธิบาย แอตทริบิวต์

point ใช้กำหนดตำแหน่งจุดเชื่อมต่อของเส้น (x,y)

<path> : ใช้สำหรับสร้างและปรับแต่งรูปร่างต่างๆ โดยจะมีคำสั่งที่ใช้ในการสร้างรูปร่างดังนี้

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

ตารางที่ 2-4 แสดงคำสั่งที่ใช้ในการสร้างรูปร่าง

ตัวอย่าง

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

คำอธิบาย

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

2.2.2.2 Element ที่ใช้อธิบายรูปร่างพื้นฐาน

Element ที่ใช้อธิบายรูปร่างพื้นฐานมีดังนี้

- rect Element ใช้อธิบายรูปสี่เหลี่ยมซึ่งในแท็กเปิดของ element นี้ก็จะมีแอตทริบิวต์ที่จะคอยอธิบายถึงลักษณะของรูปสี่เหลี่ยมเพิ่มเติมว่ามีลักษณะอย่างไรบ้างเช่น
- circle Element ใช้อธิบายรูปวงกลมว่าจะมีลักษณะ ขนาดของรัศมี สี และอื่นๆ
- ellipse Element บอกถึงลักษณะของรูปวงรี
- line Element ใช้อธิบายถึงลักษณะของเส้นขนาดเส้นใหญ่ขนาดไหนมีสีอะไร
- polygon Element ใช้อธิบายถึงลักษณะของรูปหลายเหลี่ยม

2.2.2.3 ประโยชน์ในการใช้ SVG

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

โดย element และ attribute ของ SVG ที่ใช้งานบ่อยมีดังต่อไปนี้

SVG Element

SVG	Action
<svg>	เป็น tag ที่ใช้ในการกำหนดพื้นที่ทั้งหมดของภาพและเป็น tag ที่เก็บ tag ต่างๆ ของ SVG
<path>	ใช้ในการวาดเส้น
<rect>	ใช้ในการวาดสี่เหลี่ยม
<g>	ใช้ในการรวม tag ต่างๆ ให้อยู่ในกลุ่มเดียวกัน
<desc>	ใช้ในการเพิ่มเติมข้อมูลหรือคำอธิบาย
<defs>	ใช้ในการกำหนดค่าต่างๆ
<script>	ใช้ในการเขียน script จะเขียนใน tag นี้
<text>	ใช้ในการแสดงผลแบบอักษร
<cursor>	ใช้ในการเปลี่ยนรูปของ mouse cursor
<line>	ใช้ในการวาดเส้นอย่างง่าย
<textPath>	ใช้ในการเขียนอักษรให้เป็นไปตามเส้นทางที่กำหนด
<image>	ใช้ในการแสดงรูปภาพในรูปแบบต่างๆ เช่น jpeg, gif, png เป็นต้น

ตารางที่ 2 -5 SVG Element

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SVG Attribute

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

ตารางที่ 2-6 SVG Attribute

2.2.3 ภาษา จาวาสคริปต์ (JavaScript)

จาวาสคริปต์ เป็นภาษาสคริปต์ ที่สมบูรณ์ใกล้เคียงกับภาษาในการเขียนโปรแกรมปกติ จาวาสคริปต์มีคุณสมบัติพื้นฐานต่างๆ เช่นเดียวกับภาษาปกติ ไม่ว่าจะเป็นการกำหนดตัวแปร การกำหนดฟังก์ชันการควบคุมการทำงานของโปรแกรมสแตทเมนต์ และชุดของโอเปอเรเตอร์ต่างๆ นอกจากนี้จาวาสคริปต์ยังมีคุณสมบัติในการเขียนโปรแกรมแบบอ็อบเจกต์บางอย่างไว้ด้วย ดังเช่น จาวาสคริปต์สามารถเข้าไปใช้กับอ็อบเจกต์ ซึ่งปรากฏอยู่บนเว็บหรือจะแก้ไข Attribute รวมทั้งอีเวนต์ไว้ให้เรียกใช้งานด้วย จาวาสคริปต์นั้นช่วยให้เพจ HTML สามารถทำงานได้ ดังเช่นนักพัฒนาเว็บอาจจะสร้างให้เว็บเพจของตนสามารถตอบสนองต่อเหตุการณ์ หรืออีเวนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของผู้ใช้ได้ไม่ว่าจะเป็นคลิกเมาส์ หรือว่าการเรียกฟอร์มของ HTML ข้อดีของจาวาสคริปต์ที่สำคัญอีกประการหนึ่งคือ โครงสร้างของภาษากลายภาษากาวา ซึ่งง่ายต่อการเรียนรู้และใช้งาน

ความแตกต่างระหว่างภาษากาวาและจาวาสคริปต์ที่เห็นได้ชัดเจนคือ จาวาสคริปต์มีโครงสร้างของภาษาที่ไม่ซับซ้อนเหมือนกับจาวา นักพัฒนาเว็บที่คุ้นเคยกับ Tag HTML และเคยใช้งานภาษาในการเขียนภาษาซี สามารถเรียนรู้และใช้งานภาษาสคริปต์ได้ในเวลาไม่นาน นอกจากนั้นโปรแกรมที่เขียนด้วยภาษาสคริปต์นั้น ไม่จำเป็นที่จะต้องคอมไพล์เหมือนโปรแกรมจาวาทั่วไป ทั้งนี้เนื่องจากเป็นภาษาสคริปต์ โปรแกรมเว็บเบราว์เซอร์จึงมีตัวทำหน้าที่ปฏิบัติตามสคริปต์ที่เขียนไว้ทันที ในการพัฒนาแผนที่โดยใช้ภาษา GML เราสามารถใช้ภาษากาวาสคริปต์ในการเข้าถึง Element หรือ Attribute ของโครงสร้างภาษา SVG เพื่อสร้างส่วนของการปฏิสัมพันธ์กับผู้ใช้แผนที่เช่น การย่อ หรือ ขยายแผนที่ การเลื่อนแผนที่หรือ การนำข้อมูลรายละเอียดของสถานที่ต่างๆ ขึ้นมาแสดงโดยใช้ภาษา SVG

2.2.3.1 JavaScript คืออะไร

JavaScript เป็นภาษายุคใหม่สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ตที่กำลังได้รับความนิยมอย่างสูง เราสามารถเขียน โปรแกรม JavaScript เพิ่มเข้าไปในเว็บเพจเพื่อใช้ประโยชน์สำหรับงานด้านต่าง ๆ ทั้งการคำนวณ การแสดงผล การรับ-ส่งข้อมูล และที่สำคัญคือสามารถโต้ตอบกับผู้ใช้ได้อย่างทันทีทันใด นอกจากนี้ยังมีความสามารถด้านอื่น ๆ อีกหลายประการที่ช่วยสร้างความน่าสนใจให้กับเว็บเพจของเราได้อย่างดี

2.2.3.2 ลักษณะการทำงานของ JavaScript

JavaScript เป็นภาษาสคริปต์เชิงวัตถุ หรือเรียกว่า Object Oriented Programming ที่มีเป้าหมายในการออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ต สำหรับผู้เขียนเอกสารด้วยภาษา HTML สามารถทำงานข้ามแพลตฟอร์มได้ ทำงานร่วมกับ ภาษา HTML และภาษากาวาได้ ทั้งทางฝั่งไคลเอนต์ (Client) และ ทางฝั่งเซิร์ฟเวอร์ (Server) โดยมีลักษณะการทำงานดังนี้

1. Navigator JavaScript เป็น Client-Side JavaScript ซึ่งหมายถึง JavaScript ที่ถูกแปลทางฝั่งไคลเอนต์ (หมายถึงฝั่งเครื่องคอมพิวเตอร์ของผู้ใช้ไม่ว่าจะเป็นเครื่องพีซี เครื่องแมคอินทอช หรือ อื่นๆ) จึงมีความเหมาะสมต่อการใช้งานของผู้ใช้ทั่วไปเป็นส่วนใหญ่
2. LiveWire JavaScript เป็น Server-Side JavaScript ซึ่งหมายถึง JavaScript ที่ถูกแปลทางฝั่งเซิร์ฟเวอร์ (หมายถึงฝั่งเครื่อง คอมพิวเตอร์ของผู้ให้บริการเว็บ โดยอาจจะเป็นเครื่องของซัน ซิลิคอมกราฟิก หรือ อื่น ๆ) สามารถใช้ได้เฉพาะกับ LiveWire ของเน็ตสเคป โดยตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวแปร

ตัวแปร (Variable) หมายถึง ชื่อหรือสัญลักษณ์ที่ตั้งขึ้นสำหรับการเก็บค่าใดๆ ที่ไม่คงที่โดยการจองเนื้อที่ในหน่วยความจำของระบบเครื่องที่เก็บข้อมูลซึ่งสามารถอ้างอิงได้ มีขนาดขึ้นอยู่กับชนิดของข้อมูลและค่าของข้อมูล ซึ่งค่าในตัวแปรนี้สามารถเปลี่ยนแปลงได้ตามคำสั่งในการประมวลผล

การตั้งชื่อ

การตั้งชื่อ (Identifier or Name) เป็นชื่อที่ตั้งขึ้นมาเพื่อกำหนดให้เป็นชื่อของโปรแกรมหลัก, ฟังก์ชัน, ตัวแปร, ค่าคงที่, คำสั่ง และคำสงวน โดยมีหลักการตั้งชื่อว่า

- เริ่มต้นด้วยตัวอักษรในภาษาอังกฤษ ตามด้วยตัวอักษรหรือตัวเลขใด ๆ ก็ได้
- ห้ามเว้นช่องว่าง
- ห้ามใช้สัญลักษณ์พิเศษ ยกเว้นขีดล่าง (_) และ ดอลลาร์ (\$)
- สำหรับความยาวของชื่อใน JavaScript จะมีความยาวเท่าใดก็ได้ แต่ที่นิยมใช้ ไม่เกิน 20 ตัวอักษร
- การตั้งชื่อมีข้อพึงระวังว่า จะต้องไม่ซ้ำกับคำสงวน (Reserve word) และตัวอักษรของชื่อจะจำแนกแตกต่างกันระหว่างอักษรตัวพิมพ์เล็กกับอักษรตัวพิมพ์ใหญ่
- ควรจะตั้งชื่อโดยให้ชื่อนั้นมีสื่อความหมายให้เข้ากับข้อมูล สามารถอ่านและเข้าใจได้

คำสงวน

คำสงวน (Reserve word) เป็นคำที่มีความหมายเฉพาะตัวในภาษา JavaScript สงวนไม่ให้มีการตั้งชื่อซ้ำกับชื่อ โปรแกรม, ฟังก์ชัน, ตัวแปร, ค่าคงที่ และคำสั่ง คำสงวน สามารถเรียกใช้ได้ทันทีโดยไม่ต้องมากำหนดความหมายใหม่แต่อย่างใด

ชนิดข้อมูลของตัวแปร

ชนิดของข้อมูลของตัวแปร (Data Type) เป็นการกำหนดประเภทค่าของข้อมูลให้กับตัวแปร เพื่อให้เหมาะสมกับการอ้างอิงข้อมูลจากตัวแปรในการใช้งาน ชนิดข้อมูลของตัวแปรนั้นมีอยู่ด้วยกัน 4 ชนิด ได้แก่

- Number หมายถึง ข้อมูลชนิดตัวเลข ประกอบด้วย เลขจำนวนเต็ม (Integer) และเลขจำนวนจริง (Floating)
- Logical หมายถึง ข้อมูลทางตรรกะ มี 2 สถานะ คือ จริง (True) และเท็จ (False)
- string หมายถึง ข้อมูลที่เป็นข้อความ ซึ่งจะต้องกำหนดไว้ในเครื่องหมายคำพูด ("...")
- Null หมายถึง ไม่มีค่าข้อมูลใดๆ ซึ่งค่า null ใช้สำหรับการยกเลิกพื้นที่เก็บค่าของตัวแปรออกจากหน่วยความจำ

การประกาศตัวแปร

การประกาศตัวแปร (Declarations) เป็นการกำหนดชื่อและชนิดข้อมูลให้กับตัวแปรเพื่อนำไปใช้ในโปรแกรม โดยการตั้งชื่อจะต้องคำนึงถึงค่าของข้อมูลและ ชนิดของข้อมูลที่อ้างอิง นอกจากนี้การตั้งชื่อควรให้สื่อความหมายของข้อมูล และอักษรของชื่อจะจำแนกแตกต่างกันระหว่างอักษรตัวพิมพ์เล็กกับอักษรตัวพิมพ์ใหญ่

รูปแบบ

Var ชื่อตัวแปร; เป็นรูปแบบการประกาศตัวแปรปกติ หรือ

Var ชื่อตัวแปร = ข้อมูล; เป็นรูปแบบการกำหนดค่าเริ่มต้น

ในกรณีที่ต้องการกำหนดตัวแปรหลายตัวในบรรทัดเดียวกันให้ใช้เครื่องหมาย คอมม่า (,) คั่นระหว่างชื่อตัวแปรและปิดท้ายด้วยเครื่องหมายเซมิโคลอน (;)

การกำหนดค่าให้กับตัวแปร

รูปแบบ

ชื่อตัวแปร = ค่าของข้อมูล

โดยที่

ค่าของข้อมูล ได้แก่

1. ข้อมูลที่เป็นตัวเลข โดยกำหนดตัวเลขไปได้เลย เช่น `num = 500;`
2. ข้อมูลในทางตรรกะ ได้แก่ จริง (True) หรือ เท็จ (False) เช่น `test = True;`
3. ข้อมูลสตริง ให้กำหนดอยู่ในเครื่องหมายคำพูด ("...") เช่น `name = "Adisak";`

ตัวแปร มี 2 จำพวก หากเรากำหนดชื่อตัวแปรไว้ที่โปรแกรมหลักโดยไม่ได้อยู่ภายในขอบเขตฟังก์ชันใด ๆ เราเรียกว่าเป็นตัวแปรแบบ โกลบัล (global) ตัวแปรจำพวกนี้จะมีค่าคงอยู่ในหน่วยความจำตลอดการทำงาน of โปรแกรม ทำให้สามารถเรียกใช้ได้จากทุก ๆ ส่วนของโปรแกรม รวมถึงภายในฟังก์ชันต่าง ๆ ด้วย

แต่ถ้ากำหนดตัวแปรไว้ภายในขอบเขตฟังก์ชันใด ๆ เราจะเรียกว่าเป็นตัวแปรแบบ โลคัล (Local) เพราะจะเป็นตัวแปรที่มีค่าคงอยู่ และสามารถเรียกใช้ได้เฉพาะ ภายในขอบเขตของฟังก์ชันนั้น ๆ เท่านั้น

ตัวแปรแบบอาร์เรย์

ตัวแปรแบบอาร์เรย์ (Array) หมายถึงตัวแปรซึ่งมีค่าได้หลายค่าโดยใช้ชื่ออ้างอิงเพียงชื่อเดียว ด้วยการให้หมายเลขลำดับเป็นตัวจำแนกความแตกต่างของค่าตัวแปรแต่ละตัว ถ้าเราจะเรียกตัวแปรชนิดนี้ว่า "ตัวแปรชุด" ก็เห็นจะไม่ผิดนัก ตัวแปรชนิดนี้มีประโยชน์มาก ลองคิดถึงค่าข้อมูลจำนวน 100 ค่า ที่ต้องการเก็บไว้ในตัวแปรจำนวน 100 ตัว อาจทำให้ต้องกำหนดตัวแปรที่แตกต่างกันมากถึง 100 ชื่อ กรณีอย่างนี้ควรจะทำอย่างไรดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่ด้วยการใช้คุณสมบัติอาร์เรย์ เราสามารถนำตัวแปรหลาย ๆ ตัวมาอยู่รวมเป็นชุดเดียวกันได้ และสามารถเรียกใช้ตัวแปรทั้งหมดโดยระบุผ่านชื่อเพียงชื่อเดียวเท่านั้น ด้วยการระบุหมายเลขลำดับ หรือ ดัชนี(index) กำกับตามหลังชื่อตัวแปร ตัวแปรเพียงชื่อเดียวจึงมีความสามารถเทียบได้กับตัวแปรนับร้อยตัว พันตัว (ตัวที่ 1) ในตัวแปรแบบอาร์เรย์มีดัชนีเป็น 0 ส่วนตัวแปรต่อ ๆ ไปก็จะมีดัชนีเป็น 1,2,3,... ไปตามลำดับ เมื่อต้องการระบุชื่อตัวแปรแบบอาร์เรย์แต่ละตัว ก็จะใช้รูปแบบ name[0], name[1],... เรียงต่อกันไปเรื่อยๆ เราสามารถสร้างตัวแปรอาร์เรย์ใหม่ด้วย myArray = new Array() ดังนี้

```
myArray[0] = 17;
myArray[1] = "Nun";
myArray[2] = "Stop";
```

ค่าคงที่

ค่าคงที่ (Literal หรือ Constant) หมายถึง ค่าของข้อมูลที่เมื่อกำหนดแล้วจะทำการเปลี่ยนแปลงค่าเป็นอย่างอื่นไม่ได้ ชนิดข้อมูลของค่าคงที่ได้แก่

- เลขจำนวนเต็ม (Integer) เป็นตัวเลขที่ไม่มีเศษทศนิยม สามารถเขียนให้อยู่ในแบบเลขฐานสิบ (0-9), เลขฐานสิบหก (0-9, A-F) หรือ เลขฐานแปด (0-7) โดยการเขียนเลขฐานแปดให้ นำหน้าด้วย O (Octonary) ส่วนการเขียนเลขฐานสิบหกให้นำหน้าด้วย Ox หรือ OX (Hexadecimal)
- เลขจำนวนจริง (Floating) ใช้รูปแบบการเขียนโดยประกอบไปด้วยตัวเลข จุดทศนิยม และตัวเลขยกกำลัง E (Exponential) เช่น
 - 5.00E2 จะหมายถึงค่า 5.00 คูณด้วย 10 ยกกำลัง 2 จะมีค่าเป็น 500
 - 3.141E5 จะหมายถึงค่า 3.141 คูณด้วย 10 ยกกำลัง 5 จะมีค่าเป็น 314,1000
- ค่าบูลีน (Boolean) เป็นค่าคงที่เชิงตรรกะ ก็มีค่าเป็น จริง(True) และ เท็จ(False) เท่านั้น
- ข้อความสตริง (String) เป็นค่าคงที่แบบข้อความที่อยู่ภายในเครื่องหมายคำพูด ("..." หรือ '...') เช่น "บริษัท เอ็กซ์ทริม จำกัด", 'นางนฤมล เวชตระกูล'

นิพจน์

นิพจน์ (Expression) เป็นข้อคำสั่งที่ใช้กำหนดค่าของข้อมูล เช่น การบวกตัวเลข การเปรียบเทียบข้อมูล โดยการกำหนดชื่อของตัวแปร ตามด้วยเครื่องหมายที่ต้องการกระทำ b (Operations) ต่อข้อมูลเป็นผลให้เกิดค่าข้อมูลใหม่ค่าหนึ่งให้กับตัวแปรเพื่อนำไปใช้งาน

นิพจน์ JavaScript มีด้วยกัน 3 ชนิดดังนี้

1. นิพจน์คณิตศาสตร์ (Arithmetic) เป็นนิพจน์ที่ใช้เครื่องหมายทางคณิตศาสตร์เป็นตัวกระทำ ผลลัพธ์ที่ได้จะมีค่าเป็นตัวเลขให้กับตัวแปร เช่น ให้ตัวแปร num เก็บตัวเลข 5000 จะเขียนได้ดังนี้ `num = 5000;`
2. นิพจน์ตรรกะ (Logical) เป็นนิพจน์ในการเปรียบเทียบข้อมูล โดยใช้เครื่องหมายในการเปรียบเทียบเพื่อตรวจสอบข้อมูลในการเปรียบเทียบว่าจริงหรือเท็จ เช่น กำหนดให้ `a = 50; b = 70; c = b > a;` ผลลัพธ์ที่ได้คือ c จะมีค่าเป็นจริง (True)
3. นิพจน์ข้อความ (String) เป็นนิพจน์เกี่ยวกับการกำหนดข้อความ การเชื่อมโยงข้อความ ใช้ประมวลผลข้อความในลักษณะต่าง ผลลัพธ์ที่ได้จึงมีค่าเป็นตัวอักษรหรือข้อความเสมอ เช่น ให้ตัวแปร name เก็บชื่อ Adisak จะเขียนได้ดังนี้ `name = "Adisak";`

ตัวดำเนินการ

ตัวดำเนินการ (Operator) หมายถึง เครื่องหมายกำหนดกรรมวิธีทางคณิตศาสตร์, พีชคณิต, บูลีน, การเปรียบเทียบ ระหว่างข้อมูล 2 ตัว ซึ่งเรียกว่า Operand โดยอาจมีค่าเป็นตัวเลข ข้อความ ค่าคงที่ หรือตัวแปรต่าง ๆ

ชนิดของตัวดำเนินการ

ตัวดำเนินการคณิตศาสตร์ (Arithmetic operator) หมายถึง ใช้สำหรับคำนวณ Operand ที่เป็นค่าคงที่หรือตัวแปรก็ได้ โดยให้ค่าผลลัพธ์เป็นตัวเลขค่าเดียว Operator เชิงคณิตศาสตร์ที่คนส่วนใหญ่รู้จักคุ้นเคยกันมากที่สุดได้แก่ `+`, `-`, `*`, `/`, `%`, `++`, `--` และ `(-)` เป็นต้น

รูปแบบ

เช่น `x = 20 % 3;` ผลลัพธ์คือ x จะมีค่าเป็น 2 ,

ถ้า `x = -100` ดังนั้น `-x` จะมีค่าเท่ากับ 100 เป็นต้น

ตัวดำเนินการเชิงเปรียบเทียบ (Comparison operator) หมายถึง เครื่องหมายในการเปรียบเทียบข้อมูล ผลลัพธ์ที่ได้จะมีค่าตรรกะบูลีนเป็น จริง (True) และ เท็จ (False) ได้แก่ `==`, `!=`, `>`, `>=`, `<` และ `<=` เป็นต้น

ตัวดำเนินการเชิงตรรกะ (Logical operator) เป็นเครื่องหมายที่ให้ค่าจริง (True) และ เท็จ (False) ในการเปรียบเทียบ ประกอบด้วยเครื่องหมาย `&&` (AND), `||` (OR), `!` (NOT)

ตัวดำเนินการเชิงข้อความ (String operator) เป็นการเชื่อมประโยคข้อความเข้าด้วยกัน (concatenation) โดยใช้เครื่องหมายบวก (+) เป็นตัวกระทำเช่น

`Name = "Bodin"; Say = "Hey "+Name;` ผลลัพธ์ที่ได้ Say จะมีข้อความ เป็น Hey Bodin

การสร้างตัวแปรอาร์เรย์

ก่อนที่จะมีการใช้งานอาร์เรย์นั้น เราจะต้องทำการประกาศตัวแปรที่มีลักษณะเป็นอาร์เรย์เสียก่อน เพื่อให้โปรแกรมรู้จักชื่อของตัวแปรที่จะกำหนดเป็นอาร์เรย์ รวมถึงการกำหนดขนาดของพื้นที่ในหน่วยความจำสำหรับเก็บค่าของข้อมูล รายละเอียดของการประกาศสร้างตัวแปรอาร์เรย์ มีดังต่อไปนี้

การประกาศตัวแปรอาร์เรย์ ชื่อตัวแปรอาร์เรย์ = new Array (จำนวนสมาชิก);
รายละเอียดมีดังนี้ จำนวนสมาชิก หมายถึง การกำหนดการจองพื้นที่ในหน่วยความจำ ให้กับตัวแปรเพื่อรองรับข้อมูลที่กำหนด โดยปกติจะกำหนดหรือไม่ก็ได้ เพราะอาร์เรย์ของจาวาสคริปต์มีความยืดหยุ่นมากสำหรับในการรับจำนวนสมาชิก การกำหนดค่าให้กับตัวแปรอาร์เรย์ ตัวแปร [Index] = ข้อมูล; รายละเอียดมีดังนี้ Index หมายถึง หมายเลขลำดับของพื้นที่ที่เก็บข้อมูล โดยเริ่มนับตั้งแต่ 0, 1, 2 ... เป็นต้นไป

ฟังก์ชันและเมธอด

ฟังก์ชัน (Function) โดยเนื้อแท้แล้วก็คือ โปรแกรมย่อย เป็นส่วนของโปรแกรมที่ถูกกำหนดให้ทำงานใดงานหนึ่งจนสำเร็จ มีประโยชน์ตรงที่ช่วยเหลือการทำงาน หรือตอบสนองการทำงานต่อ โปรแกรมหลัก เมื่อมีการเรียกใช้งาน และมีประโยชน์ทำให้โครงสร้างของโปรแกรม มีขนาดเล็กกลง เมื่อมีการเรียกใช้งานเดิมซ้ำๆ หลายๆ ครั้ง แทนที่จะเขียนคำสั่งเดิมเพิ่มขึ้นใหม่ซ้ำๆ หลายครั้ง ทำให้ดูสั้นเปลืองเนื้อที่ ซ้ำซ้อนและเสียเวลาการทำงาน นอกจากนี้ฟังก์ชันยังทำให้โปรแกรมอ่านได้ง่ายขึ้น สะดวกในการหาจุดที่ผิดและง่ายต่อการปรับปรุงแก้ไขพัฒนาโปรแกรมให้ดียิ่งขึ้น ชนิดของฟังก์ชัน

ฟังก์ชันใน ภาษาJavaScript มีอยู่ด้วยกัน 2 แบบ คือ

- ฟังก์ชันมาตรฐาน (Standard Function) เป็นแบบชื่อของฟังก์ชันที่มีอยู่แล้วในภาษา JavaScript เราสามารถนำเอาไปใช้งานได้ทันที
- ฟังก์ชันสร้างขึ้นเอง (User-defined Function) เป็นแบบชื่อของฟังก์ชันที่ผู้ใช้สร้างขึ้นมาใช้เอง เพื่อกำหนดให้ทำงานใดงานหนึ่งจนสำเร็จ เราอาจจะเรียกฟังก์ชันสร้างขึ้นเองนี้สั้นๆ ว่าฟังก์ชัน (Function) ก็ได้

การเรียกใช้ฟังก์ชัน

เป็นการกำหนดทิศทางการทำงานของคำสั่ง หรือ กำหนดส่วนของโปรแกรมให้ทำงานใดงานหนึ่งจนสำเร็จ โดยอ้างอิงชื่อฟังก์ชันของการทำงานที่ต้องการผลของการเรียกใช้ฟังก์ชันจะมีการส่งค่าคืนกลับไปยังโปรแกรมส่วนที่เรียก โดยใช้ชื่อฟังก์ชันเป็นตัวเก็บค่าเปรียบเสมือนหนึ่งกับเป็นตัวแปร เมื่อต้องการใช้งานก็ให้พิมพ์ชื่อฟังก์ชันนี้ลงไปตรงๆ แต่จะต้องจำให้แม่นว่าต้องพิมพ์ให้เหมือนทั้งชื่อและตัวอักษรตัวพิมพ์ใหญ่-เล็ก มีรูปแบบการเรียกใช้ดังนี้

รูปแบบ

ตัวแปร = ชื่อฟังก์ชัน();

โดยกำหนดให้

ตัวแปร ทำหน้าที่เก็บผลลัพธ์ที่ได้จากการอ้างอิงเรียกใช้ฟังก์ชัน เพื่อให้ทำงานใดงานหนึ่งจนสำเร็จ การสร้างฟังก์ชันขึ้นใช้เอง

การสร้างฟังก์ชันขึ้นใช้เอง (User-defined Function) เป็นแบบชื่อของฟังก์ชันที่ผู้ใช้สร้างขึ้นมาเพื่อกำหนดให้ทำงานใดงานหนึ่งจนสำเร็จ การสร้างฟังก์ชันนั้นจะขึ้นต้นด้วยคำว่า function ตามด้วยชื่อของฟังก์ชันที่ต้องการ มีรายละเอียดดังนี้

รูปแบบ

function ชื่อฟังก์ชัน (พารามิเตอร์1, พารามิเตอร์2, ...)

{

 ชื่อคำสั่ง

}

โดยกำหนดให้

- ชื่อฟังก์ชัน (function name) - หมายถึง ชื่อของฟังก์ชันที่สร้างขึ้น ที่ไม่ไปซ้ำกับชื่อของฟังก์ชันอื่น
- พารามิเตอร์ (parameter) - หมายถึง ค่าของข้อมูลหรือตัวแปรที่อ้างอิง (argument) แล้วส่งผ่านไปยังฟังก์ชัน ต้องระบุอยู่ในเครื่องหมายวงเล็บเท่านั้น โดยจะมีพารามิเตอร์เพียงตัวเดียว, หลายตัว หรือไม่มีเลยก็ได้ กรณีที่มีพารามิเตอร์หลาย ๆ ตัว แต่ละตัวจะต้องเขียนแยกออกจากกันด้วยเครื่องหมายจุลภาค (,)
- ชื่อคำสั่ง (statements) - หมายถึง คำสั่งต่าง ๆ ที่ประกอบขึ้นเพื่อให้ดำเนินงานภายในฟังก์ชัน ต้องกำหนดคำสั่งต่าง ๆ ภายใต้อันตรึงวงเล็บปีกกา {.....}

การวางตำแหน่งฟังก์ชัน

สำหรับการวางตำแหน่งฟังก์ชันในภาษาJavaScript ก็มีลักษณะเช่นเดียวกับการวางตำแหน่งสคริปต์ นั่นคือจะวางไว้ในส่วนของ <HEAD> หรือวางไว้ในส่วนของ <BODY>อย่างไรก็ขึ้นอยู่กับว่าต้องการให้ฟังก์ชันนั้นถูกโหลดใช้งานก่อนหรือหลังตามลำดับการเรียกใช้งานอย่างไร ในกรณีที่ฟังก์ชันนั้นมีการถูกเรียกใช้บ่อยครั้งจากส่วนอื่น ๆ ของโปรแกรม ทางคณะผู้จัดทำแนะนำว่า ควรจะกำหนดฟังก์ชันไว้ในส่วนของ <HEAD> เพราะเมื่อมีการเรียกใช้โหลดเว็บเพจขึ้นมา ฟังก์ชันต่าง ๆ ที่กำหนดในส่วน <HEAD> จะถูกโหลดเข้ามาเก็บไว้ในหน่วยความจำก่อนเป็นอันดับแรก ทำให้เราสามารถเรียกใช้ฟังก์ชันจากตำแหน่งใดๆ บนเอกสาร HTML หรือบนขอบเขต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<SCRIPT> ได้อย่างต่อเนื่อง และนอกจากนี้ฟังก์ชันยังสามารถเรียกใช้ฟังก์ชันอื่นๆ ที่กำหนดในส่วน <HEAD> ทำงานร่วมกันได้อีกด้วย

รูปแบบ

```
<html>
```

```
<head><title> ชื่อความแถบเรื่อง </title>
```

```
<script Language = "JavaScript">
```

```
<!-- ชื่อความหมายเหตุ ...
```

```
function ชื่อฟังก์ชัน (พารามิเตอร์1, พารามิเตอร์2, ...)
```

```
{ ชื่อคำสั่ง ;
```

```
.....
```

```
}
```

```
กลุ่มโค้ดคำสั่งของภาษาจาวาสคริปต์
```

```
// ชื่อความหมายเหตุ -->
```

```
</script>
```

```
</head>
```

```
<body>
```

```
ข้อความเอกสาร html
```

```
.....
```

```
<script Language = "JavaScript">
```

```
กลุ่มโค้ดคำสั่งของภาษาจาวาสคริปต์
```

```
.....
```

```
</script>
```

```
</body>
```

```
</html>
```

โดยกำหนดให้

กลุ่มโค้ดคำสั่งของภาษาจาวาสคริปต์ หมายถึง คำสั่งใด ๆ หรือ ฟังก์ชัน หรือ การเรียกใช้ฟังก์ชันที่ต้องการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4 Ajax (Asynchronous JavaScript and XML)

ปัจจุบันนี้ ลักษณะการทำงานแบบ Client - Server เริ่มถูกนำมาใช้งานอย่างแพร่หลายในลักษณะการติดต่อสื่อสารผ่านทาง Web browser ซึ่งการทำงานแบบนี้ จะมีการทำงานโดย Client จะร้องขอและต้องการข้อมูลบางอย่างจาก Server ดังนั้นการโหลดและการรีเฟรชหน้าจอก็ เป็นสิ่งที่หลีกเลี่ยงไม่ได้ จึงเป็นผลให้การทำงานของฝั่ง Client นี้ทำให้ผู้ใช้ต้องหยุดรอการโหลดและการรีเฟรชหน้าจอ ซึ่งถือว่าเป็นการทำงานที่ไม่มีประสิทธิภาพ

2.2.4.1 Ajax คืออะไร

Ajax ไม่ใช่ชื่อของการเขียนโปรแกรม หรือเป็นชื่อของภาษาที่ใช้ในการ โปรแกรม แต่เป็นชุดของเทคโนโลยีต่างๆ Ajax ย่อมาจาก Asynchronous JavaScript And XML; ซึ่งหมายถึงการทำงานร่วมกันของ JavaScript และ XML แบบ Asynchronous มีหลักการการทำงาน 2 ประเด็น คือ การ update หน้าจอแบบบางส่วน และการติดต่อสื่อสารกับ Server โดยใช้หลักการ Asynchronous ทำให้ผู้ใช้ไม่ต้องหยุดการทำงาน เพื่อรอการประมวลผลจาก Server รวมถึงการโหลดและการรีเฟรชหน้าจอ ของเบราว์เซอร์ทางฝั่ง Client มีการใช้ Ajax โดยการเพิ่มเลเยอร์ระหว่าง user browser กับ server ทำให้ผู้ใช้สามารถทำงานได้โดยไม่ต้องรอให้ Client ติดต่อไปยัง Server รวมถึงการโหลดและการรีเฟรชหน้าจอตลอดด้วย ดังนั้นผู้ใช้สามารถใช้งาน application ได้อย่างมีประสิทธิภาพมากขึ้น

สรุปก็คือ Ajax จึงไม่ใช่เทคโนโลยีในตัวของมันเองและ ไม่ใช่เทคโนโลยีล่าสุดของการพัฒนาเว็บแอปพลิเคชัน แต่ Ajax เป็นการนำเทคโนโลยีหลายๆ ตัวมารวมกันเช่น JavaScript, DHTML, XML, Css, DOM และ XMLHttpRequest มันประกอบไปด้วยเทคโนโลยีที่เราคุ้นเคยกันดีอยู่แล้ว แต่ถูกจับมารวมอยู่ด้วยกัน Ajax ในวันนี้มีแนวโน้มที่จะเติบโตอย่างรวดเร็วในอนาคตอันใกล้ เพราะแม้แต่ Google ที่ได้รับการยอมรับว่ามีการพัฒนาเรื่องของเทคโนโลยีทางเว็บก้าวหน้าอย่างมากยังนำ Ajax มาใช้ในแอปพลิเคชันของ

Ajax engine ทำหน้าที่เป็นตัวกลางระหว่าง client และ server ฉะนั้นเมื่อ client มี request แทนที่จะส่ง HTTP request ไปยัง server โดยตรง client จะส่ง JavaScript call ไปยัง Ajax engine เพื่อโหลดข้อมูลที่ user ต้องการ และหาก Ajax engine ต้องการข้อมูลเพิ่มเติมในการตอบสนองต่อ user Ajax engine จะส่ง request ไปยัง server โดยใช้ XML เทคโนโลยีต่าง ๆ ที่เป็นส่วนประกอบของ Ajax ซึ่งได้แก่

- HTML/XHTML เป็นภาษาในการจัดแสดงข้อมูล
- CSS เป็นรูปแบบการจัดแต่ง XHTML
- Document Object Model (DOM) สำหรับ dynamic display and interaction

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- XML เป็นรูปแบบการแลกเปลี่ยนข้อมูล
- XSLT สำหรับ แปลง XML เป็น XHTML
- XMLHttpRequest สำหรับ asynchronous data retrieval
- JavaScript เป็นภาษาในการใช้งาน Ajax engine

โดยส่วนประกอบจำเป็นขั้นพื้นฐานที่ขาดไม่ได้ใน Ajax ได้แก่ HTML/XHTML DOM และ JavaScript เพราะ XHTML

เนื่องจากแอปพลิเคชันที่ใช้งานในปัจจุบันนี้ มีหลักการที่ทำงานแล้วเกิดการสูญเสียเวลาและทรัพยากรของผู้ใช้ในการรอคอยการทำงานต่างๆ ทำให้ผู้ใช้ต้องหยุดคอย ดังนั้นการทำงานของผู้ใช้จึงเป็นไปอย่างไม่ต่อเนื่อง ซึ่งหลักการดังกล่าวคือ

- **"Click, wait, and refresh" user interaction paradigm**

การที่เบราว์เซอร์ตอบสนองต่อการทำงานของผู้ใช้ โดยจะทิ้งหน้าเว็บที่แสดงอยู่ในขณะนั้นแล้วไปทำการส่ง HTTP request กลับไปยัง server แทน ซึ่งทำให้ผู้ใช้ไม่สามารถทำอะไรได้เลยในขณะนั้น นอกจากการรอคอย เมื่อ server ทำการประมวลผลเสร็จก็จะส่งหน้า HTML กลับมายังเบราว์เซอร์ ต่อจากนั้นเบราว์เซอร์ก็จะรีเฟรชและแสดงหน้า HTML หน้าใหม่ และนี่เองที่ทำให้ผู้ใช้สามารถใช้งานต่อไปได้ จะเห็นว่า ผู้ใช้มีช่วงเวลาของการหยุดรอคอยเป็นเวลานานสำหรับการประมวลผลของ Server และการรีเฟรชหน้า HTML ใหม่ทั้งหน้า ซึ่งเป็นสิ่งที่ไม่มีประสิทธิภาพในเชิง Dynamic ของการทำงานบนเว็บแอปพลิเคชัน

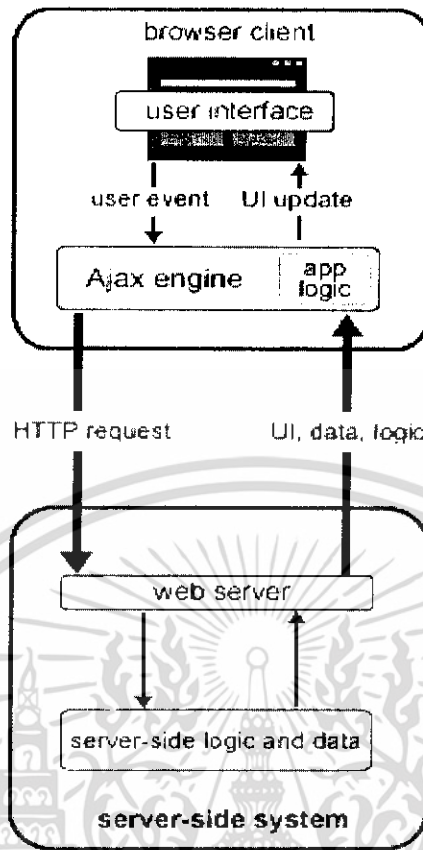
- **Synchronous "request/response" communication mode**

การที่เบราว์เซอร์เริ่มทำการร้องขอข้อมูล และ server ก็ตอบสนองเฉพาะการร้องขอที่เบราว์เซอร์ร้องขอมา server จะไม่สามารถส่งข้อมูลได้ถ้าเบราว์เซอร์ไม่ได้ร้องขอข้อมูลในขณะนั้น ซึ่งถือว่าเป็นการติดต่อสื่อสารเป็นแบบทิศทางเดียว วงจรการ request/response แบบ synchronous คือ การทำงานแบบประสานจังหวะระหว่างเบราว์เซอร์กับ Server ทำให้เกิดความล่าช้าในการทำงาน ทำให้ผู้ใช้ทำอะไรไม่ได้อีก นอกจากการคอยการตอบสนองกลับมาจาก server เมื่อ server ประมวลผลเสร็จ

2.2.4.2 สถาปัตยกรรมของ Ajax

มุมมองของโครงสร้างทาง Software ของ Ajax ต่างจากเว็บแอปพลิเคชันในทุกวันนี้ เนื่องจากมีการเพิ่ม engine ทางฝั่ง client

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



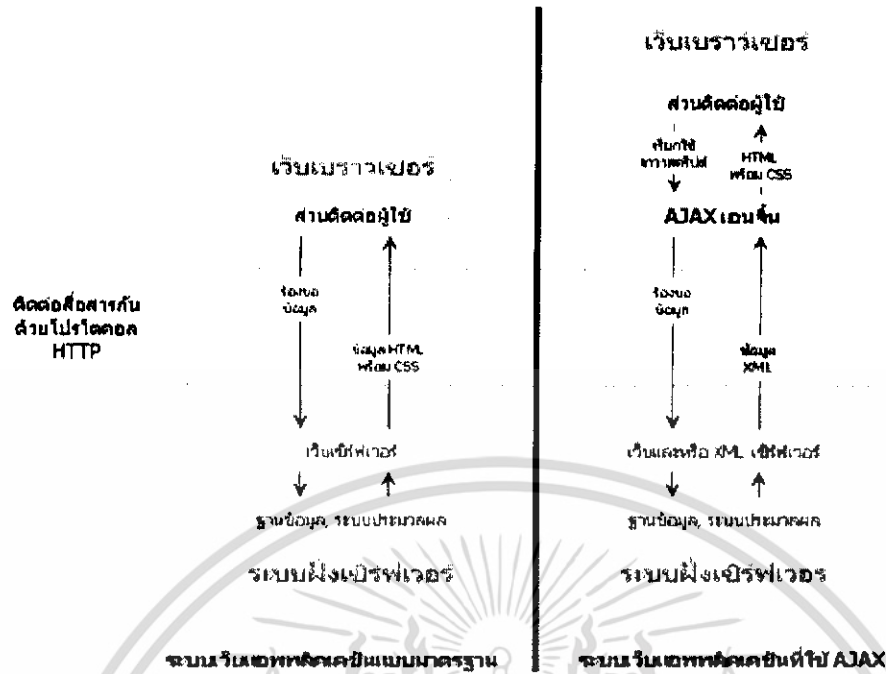
รูปที่ 2-8 Ajax architecture

จากรูป Ajax engine นี้ อยู่ระหว่าง User Interface กับ server ซึ่งจะมองว่าเป็นการทำงานที่ Client การทำงานต่างๆของผู้ใช้ โปรแกรมจะไปเรียก Ajax engine ตัวนี้ขึ้นมา แทนที่การร้องขอ หน้าเว็บจาก server โดยตรง และจะใช้โครงสร้างข้อมูลแบบ XML ในการขนย้ายข้อมูลระหว่าง server กับ Ajax engine เมื่อเบราว์เซอร์ทำการร้องขอข้อมูลจาก server นอกจากนี้ Ajax engine ไม่ต้องการติดตั้ง ไม่ใช่ plug-in และไม่สามารถ download ได้ เพราะ Ajax เป็นแนวคิดในการ แก้ปัญหาการหยุดชะงักการ ทำงานของผู้ใช้

2.2.4.3 หลักการและการทำงานของ Ajax

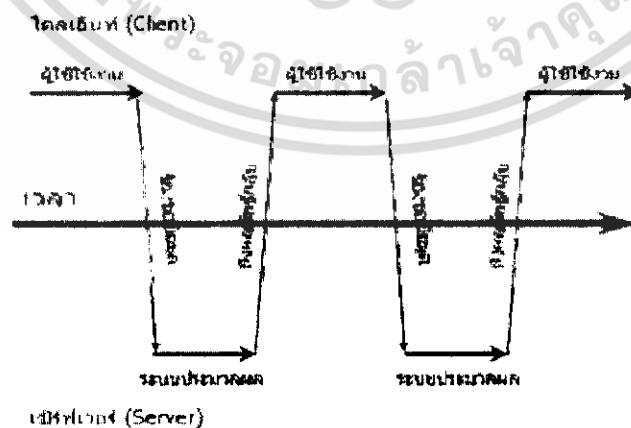
Ajax (Asynchronous JavaScript and XML) คือเทคโนโลยีที่รวมเอาความสามารถของ JavaScript, XML, CSS และ XHTML เอาไว้ด้วยกัน Ajax เป็นการประยุกต์เอาเทคโนโลยีเก่า มาผสมผสานจนได้เทคโนโลยีใหม่ที่นำศึกษาและนำมาใช้งาน แต่ก่อนอื่นมาทำความเข้าใจ หลักการทำงานของ Web กันก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



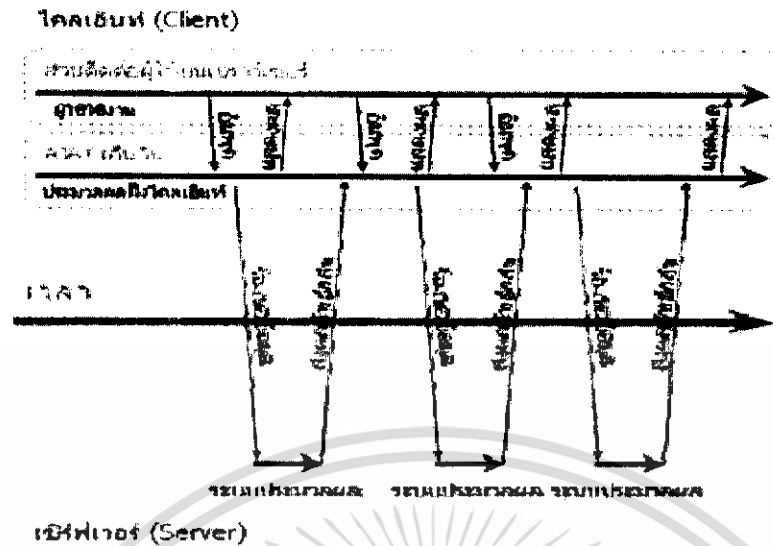
รูปที่ 2-9 เปรียบเทียบระบบการทำงานของเว็บแบบมาตรฐานและแบบใช้ Ajax

ตามปกติเมื่อเราเปิด Web Browser และพิมพ์ URL ของเว็บเพจที่ต้องการ เราจะเรียกผู้ใช้ว่า client-side browser ก็จะส่งค่าไปยัง server เพื่อขอเปิดหน้า url ที่เราพิมพ์ลงไป เช่น <http://www.google.co.th> และเมื่อทาง server ได้รับค่าที่ส่งมาก็จะส่งหน้าเว็บเพจกลับมาให้ โดยเราจะเรียก server ว่าผู้ให้บริการหรือ server side เมื่อฝั่งผู้ใช้ได้รับข้อมูลจาก server ที่ส่งมาให้ browser ก็จะนำข้อมูลนั้นขึ้นหน้าจอ จากนั้นเมื่อเราคลิกเว็บหน้าอื่นๆก็จะเริ่มขั้นตอนทั้งหมดใหม่อีกครั้ง วิธีการนี้หน้าจอ webpage จะต้องมีการ refresh ใหม่ และการรับผลที่ส่งกลับมาจะเป็นการส่งมาแบบทั้งหน้า webpage แบบเต็มๆ ทำให้กิน bandwidth มาก



รูปที่ 2-10 ระบบเว็บแอปพลิเคชันแบบมาตรฐาน (Synchronize)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-11 ระบบเว็บแอปพลิเคชันแบบที่ใช้ Ajax (Asynchronous)

แล้ว Ajax คือว่าตรงที่การทำงานของ Ajax นั้นจะส่งเฉพาะข้อมูลที่ต้องการไปยัง server และส่งกลับมาเฉพาะข้อมูลที่ต้องการ ไม่ใช่การส่งทั้งหน้า webpage ใหม่ โดย Ajax อาศัย object ที่ชื่อ XMLHttpRequest เมื่อผู้ใช้เปิดหน้าเว็บแล้วมีการส่งข้อมูล Ajax ก็จะใช้ XMLHttpRequest ส่งค่าไปให้ server แล้วให้ server จัดการข้อมูลนั้นตามเงื่อนไขแล้วส่งข้อมูลนั้นกลับมาในรูปแบบ XML ซึ่งก็จะใช้ JavaScript เป็นตัวจัดการข้อมูลที่ได้รับให้แสดงผลได้อย่างถูกต้องในหน้าเว็บเพจเดิม

AJAX จะช่วยลดการติดต่อกันระหว่าง Client กับ Server โดยในการโหลดหน้าเว็บนั้นเบราว์เซอร์จะโหลดข้อมูลจาก AJAX engine แทนการร้องขอข้อมูลจาก server โดยตรง ดังนั้น Ajax จะทำหน้าที่ทั้งการ render ส่วนติดต่อกับผู้ใช้และติดต่อไปยัง server แล้ว AJAX engine อนุญาตให้ทำการกระทำต่างๆ ใน web application เป็นแบบ Asynchronous คือความเป็นอิสระในการติดต่อไปยัง server นั่นเอง ดังนั้นผู้ใช้จะไม่พบกับเบราว์เซอร์หน้าขาวๆ อีกต่อไป และไม่ต้องการการโหลดข้อมูลต่างๆ จาก server

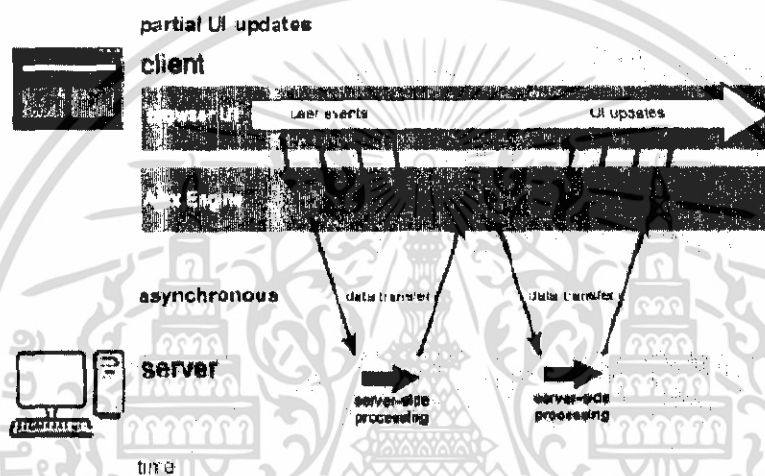
- "Partial screen update" replaces the "click, wait, and refresh" user interaction model

การ Update หน้าจอบางส่วน แทนที่การ "click, wait, and refresh" ระหว่างที่เกิด การทำงานแบบการติดต่อสื่อสารของผู้ใช้ user interface ที่ต้องนำมาแสดงซ้ำในหน้าเว็บที่ร้องขอไปยัง server จะถูกจัดเป็นข้อมูลใหม่เมื่อถูก update แล้ว การหยุดชะงักของ user interface จึงไม่เกิดขึ้น เพราะหน้าเว็บนั้นยังคงถูกแสดงอยู่และสามารถใช้งานได้ โดยปราศจากการหยุดชะงักการทำงานของ ผู้ใช้ การ update หน้าเว็บบางส่วนสามารถทำให้หน้าเว็บทำงานต่อไปได้ ถึงจะไม่ใช้ทั้งหมด แต่อย่างน้อยก็ทำให้การทำงานไม่จำเป็นต้องหยุดชะงักเลย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

■ **Asynchronous communication replaces "synchronous request/response model"**

การติดต่อแบบ Asynchronous เข้ามาแทนที่การ "synchronous request/response model" สำหรับ Ajax การ request/response จะทำแบบ asynchronous ซึ่งคือการติดต่อสื่อสารกับ server แบบอิสระโดยทำการลดการติดต่อระหว่างบราวเซอร์ กับ server ผลที่ได้ก็คือผู้ใช้สามารถใช้งานเว็บแอปพลิเคชันได้ในขณะที่ client ทำการร้องขอข้อมูลจาก server อยู่เบื้องหลัง(การทำงานแบบพร้อมกันแต่มองเป็น 2 ฟาก เช่นหน้าร้านกับหลังร้าน) เมื่อข้อมูลเดินทางมาถึงบราวเซอร์ ก็จะ update หน้า user interface ที่ต้องการข้อมูลใหม่ ส่วนหน้า user interface ที่ไม่ต้องการ update ก็จะแสดงส่วนนั้นต่อไป



รูปที่ 2-12 Ajax Model: Partial UI updates and asynchronous communications

รูปการทำงานแบบ Asynchronous และการ update หน้าเว็บแบบบางส่วน ที่ทำให้การทำงานของผู้ใช้มีประสิทธิภาพมากขึ้น

บราวเซอร์ที่สนับสนุนอย่างที่กล่าวมาแล้วว่า Ajax เป็นเทคนิคที่อยู่บนพื้นฐานของเทคโนโลยีที่มีอยู่ในปัจจุบัน ไม่มีอะไรใหม่ จึงทำให้โปรแกรมบราวเซอร์ที่เป็นที่นิยมกันอยู่ในปัจจุบันสามารถทำงานร่วมกับ Ajax ได้

- Apple Safari 1.2 หรือใหม่กว่า
- Microsoft Internet Explorer 4.0 หรือใหม่กว่า
- Mozilla Firefox 1.0 หรือใหม่กว่า
- Netscape 7.1 หรือใหม่กว่า
- Opera 7.6
- Konqueror

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อดีของ Ajax

1. ตอบสนองต่อผู้ใช้ได้อย่างรวดเร็วเนื่องจากการ Update แบบบางส่วน
2. การแสดงผลที่เร็วกว่า ไม่ต้อง Refresh หน้าจอใหม่ทุกครั้งอีกทั้งข้อมูลที่ส่งไป-กลับไม่ได้ส่งไปทั้งหน้า ทำให้กิน bandwidth น้อยกว่า
3. ผู้ใช้ไม่ต้องหยุดรอคอยการประมวลผลของ Server เนื่องจากการติดต่อแบบ Asynchronous รองรับกับบราวเซอร์หลักๆที่สามารถใช้ JavaScript ได้
4. ทำให้การประมวลผลที่ Server มีความรวดเร็วขึ้นเนื่องจากการประมวลผลที่ Server ลดลงไม่ต้องทำการติดตั้ง หรือใช้ Plugs-in
5. ไม่ยึดติดกับ Platform หรือภาษาที่ใช้ในการเขียนโปรแกรม
6. เป็นเทคโนโลยีใหม่ที่ไม่ได้เป็นของนักพัฒนาเว็บแอปพลิเคชันคนใด นั่นคือทุกคนมีสิทธิ์เข้ามาพัฒนาแอปพลิเคชันตัวนี้

สรุปก็คือ Ajax ไม่ใช่เทคโนโลยีล่าสุดของการพัฒนาเว็บแอปพลิเคชัน แต่มันประกอบไปด้วยเทคโนโลยีที่เราคุ้นเคยกันดีอยู่แล้ว แต่ถูกจับมารวมอยู่ด้วยกัน Ajax ในวันนี้มีแนวโน้มที่จะเติบโตอย่างรวดเร็วในอนาคตอันใกล้ เพราะแม้แต่ Google ที่ได้รับการยอมรับว่ามีการพัฒนาเรื่องของเทคโนโลยีทางเว็บก้าวหน้าอย่างมากยังนำ Ajax มาใช้ในแอปพลิเคชันของตน เราในฐานะของนักพัฒนาชาวไทย เราคงต้องหันมาสนใจและนำ Ajax มาใช้มากขึ้น

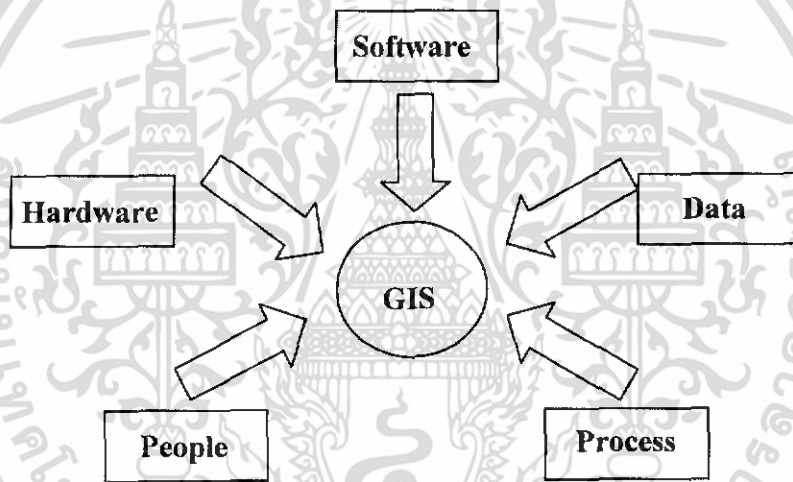
บทที่ 3

GIS และ Spatial Database

3.1 ระบบสารสนเทศภูมิศาสตร์ (Geographic Information System)

ระบบสารสนเทศภูมิศาสตร์ หมายถึง เครื่องมือที่ใช้ระบบคอมพิวเตอร์ เพื่อใช้ในการนำเข้า จัดเก็บ จัดเตรียม คัดแปลง แก้ไข จัดการ และวิเคราะห์ พร้อมทั้งแสดงผลข้อมูลเชิงพื้นที่ตาม วัตถุประสงค์ต่างๆ ที่ได้กำหนดไว้ ดังนั้น GIS จึงเป็นเครื่องมือที่มีประโยชน์เพื่อใช้ในการจัดการ และบริหารการใช้ทรัพยากรธรรมชาติและสิ่งแวดล้อม และสามารถติดตามการเปลี่ยนแปลงข้อมูล ด้านพื้นที่ให้เป็นไปอย่างมีประสิทธิภาพ

องค์ประกอบของระบบสารสนเทศภูมิศาสตร์ (GIS)



รูปที่ 3-1 องค์ประกอบของ GIS

- ข้อมูล (Data/Information) ข้อมูลที่จะนำเข้าสู่ระบบสารสนเทศภูมิศาสตร์ ควรเป็นข้อมูล เฉพาะเรื่อง(theme) และเป็นข้อมูลที่สามารถนำมาใช้ในการตอบคำถามต่างๆ ได้ตรงตาม วัตถุประสงค์เป็นข้อมูลที่มีความถูกต้องและเชื่อถือได้และเป็นปัจจุบันมากที่สุด ข้อมูลหรือ สารสนเทศ

สามารถแบ่งออกได้เป็น 2 ประเภท ดังนี้

1. ข้อมูลที่มีลักษณะเชิงพื้นที่ (Spatial Data) ข้อมูลเชิงพื้นที่เป็นข้อมูลที่แสดงตำแหน่งที่ตั้ง ทางภูมิศาสตร์ (geo-referenced data) ของรูปลักษณะของพื้นที่ (graphic feature) ซึ่งแบ่งเป็น 2 ลักษณะ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 1.1. ข้อมูลที่แสดงทิศทาง (vector data) ประกอบด้วยลักษณะ 3 อย่าง ดังนี้
- 1.1.1 ข้อมูลจุด (Point) ใช้อ้างอิงถึงตำแหน่งที่ตั้งของสิ่งต่างๆ ในแผนที่ เช่น โรงเรียน เป็นต้น
 - 1.1.2 ข้อมูลเส้น (Arc หรือ Line) เป็นชุดของจุดที่เรียงต่อกันโดยใช้แทนลักษณะที่เป็นเส้น เช่น ถนน เป็นต้น
 - 1.1.3 ข้อมูลพื้นที่ หรือเส้นรอบรูป (Polygon) เป็นเส้นรอบรูปปิด ใช้แทนลักษณะที่พื้นที่ เช่น พื้นที่ป่าไม้ ตัวเมือง เป็นต้น



รูปที่ 3-2 ลักษณะข้อมูลที่แสดงทิศทาง (vector data)

1.2. ข้อมูลที่แสดงเป็นตารางกริด (Raster Data)

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



รูปที่ 3-3 ข้อมูลตารางกริด (raster data)

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เครื่องคอมพิวเตอร์ (Hardware) เครื่องคอมพิวเตอร์และอุปกรณ์ต่างๆ เครื่องคอมพิวเตอร์รวมกันเรียกว่า ระบบฮาร์ดแวร์ (Hardware) ซึ่งประกอบด้วยคอมพิวเตอร์ และอุปกรณ์นำเข้า เช่น digitizer scanner อุปกรณ์อ่านข้อมูล เก็บรักษาข้อมูล และแสดงผลข้อมูล เช่น printer plotter เป็นต้น ซึ่งอุปกรณ์แต่ละชนิดจะมีหน้าที่และคุณภาพแตกต่างกันออกไป
- โปรแกรมหรือระบบซอฟต์แวร์ (Software) ซอฟต์แวร์ หมายถึง โปรแกรมที่ใช้ในการจัดการระบบ สิ่งงานต่างๆ เพื่อให้ระบบฮาร์ดแวร์ทำงาน และสามารถสร้างความสัมพันธ์ระหว่างข้อมูลที่เป็นกราฟิกและแอตทริบิวต์ รวมทั้งการเพิ่มเติม แก้ไขข้อมูล และเรียกดึงข้อมูลที่จัดเก็บในระบบฐานข้อมูลมาใช้ได้ง่าย สะดวก และรวดเร็ว อีกทั้งมีความสามารถในการวิเคราะห์ข้อมูลและแสดงผลข้อมูลในรูปแบบที่เข้าใจได้ง่าย เช่น รายงาน ตาราง หรือแผนที่ เป็นต้น โดยทั่วไป ชุดคำสั่งหรือโปรแกรมของสารสนเทศทางภูมิศาสตร์จะประกอบด้วยหน่วยนำเข้าข้อมูล หน่วยเก็บข้อมูลและการจัดการข้อมูล หน่วยวิเคราะห์แสดงผล หน่วยแปลงข้อมูล และหน่วยโต้ตอบกับผู้ใช้
- บุคลากร (Human Resources) บุคลากรจะประกอบด้วยผู้ใช้ระบบ (Analyst) และผู้ใช้สารสนเทศ (User)
- กระบวนการ (Process) กระบวนการหรือขั้นตอนการทำงานที่ออกแบบไว้สำหรับ GIS ขององค์กรหรืองานนั้นเป็นไปอย่างมีแบบแผน ถูกต้อง และสามารถปฏิบัติงานได้ต่อไป

3.2 ฐานข้อมูลเชิงพื้นที่ (Spatial database)

ระบบฐานข้อมูลเชิงพื้นที่ เป็นระบบฐานข้อมูลที่ใช้ในการจัดการข้อมูลทางเรขาคณิต ข้อมูลที่เกี่ยวข้องกับภูมิศาสตร์และข้อมูลเชิงพื้นที่ โดยตัวอย่างของพื้นที่ที่เราสนใจอาจจะเป็น พื้นที่ 2 มิติต่างๆ, ส่วนของพื้นผิวโลก หรือตัวอย่างอื่นๆ เช่น ข้อมูลแสดง โครงสร้างของโมเดลของโปรตีน ซึ่งการนำวิธีการจัดการกับฐานข้อมูล โดยใช้วิธีการจัดการกับระบบฐานข้อมูลในรูปแบบเก่าที่เคยมีอยู่เช่น การจัดการระบบฐานข้อมูลเชิงสัมพันธ์ (Relational database system) มาจัดการกับข้อมูลในรูปแบบที่ได้กล่าวมานั้นทำให้เกิดความไม่เหมาะสมเช่น

1. ข้อมูลในเชิงพื้นที่ (Spatial data) มีรูปแบบความสัมพันธ์ที่ซับซ้อนกว่าข้อมูลทางธุรกิจ โดยทั่วไป ซึ่งทำให้โครงสร้างของฐานข้อมูลในรูปแบบเดิมไม่มีความเหมาะสมในการจัดการข้อมูล
2. การออกแบบโครงสร้างและการสร้างฐานข้อมูล โดยทั่วไปจะถูกดูแลโดยนักคอมพิวเตอร์ในขณะที่ข้อมูลเชิงพื้นที่จะถูกกำหนด โดยนักภูมิศาสตร์ซึ่งโดยทั่วไปจะมีรูปแบบและวิธีการที่ต่างกัน

3.2.1 Model of spatial information

แอปพลิเคชันหลักที่มีส่วนในการผลักดันการวิจัยระบบฐานข้อมูลเชิงพื้นที่คือเทคโนโลยีสำหรับ GIS (geographic information systems) ดังนั้นการใช้งานระบบฐานข้อมูลในเชิงพื้นที่ส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใหญ่จะเกี่ยวข้องกับข้อมูลทางด้านภูมิศาสตร์ ซึ่งอาจจะอยู่ในรูปของข้อมูลในรูปแบบ 2 มิติ หรือ 3 มิติ ซึ่งมีรูปแบบการจำลอง (Modeling) 2 ทางในการจำลองแบบจำลองของข้อมูลเชิงพื้นที่ก็คือ

1. Objects Base Model ในการจำลองข้อมูลรูปแบบนี้ เราให้ความสนใจในวัตถุหรือองค์ประกอบต่างๆ ที่อยู่ในพื้นที่นั้น และแต่ละวัตถุหรือองค์ประกอบจะมีข้อมูลที่บอกรายละเอียดทางเรขาคณิตของตัวมันเองเช่น เมือง, ป่า หรือแม่น้ำ ซึ่งเป็นองค์ประกอบของประเทศ เป็นต้น ในการจำลองวัตถุหรือองค์ประกอบที่เราสนใจ โครงสร้างพื้นฐานที่ใช้ในการจำลองโดยทั่วไปมักจะ เป็นจุด (point) เส้น (line) หรือพื้นที่ (region) โดยจุด (point) จะใช้แสดงถึงวัตถุหรือองค์ประกอบที่เราต้องการแสดงเพียงที่ตั้งของมันบนพื้นที่ เส้น (line) โดยทั่วไปจะใช้แสดงถึงวัตถุหรือองค์ประกอบที่มีการเคลื่อนผ่านพื้นที่ที่เราสนใจหรือการเชื่อมต่อบนพื้นที่ เช่น แม่น้ำ ถนน หรือการวางสายโทรศัพท์ พื้นที่ (region) จะใช้แสดงถึงวัตถุหรือองค์ประกอบที่มีข้อมูลซึ่งแสดงรายละเอียดทาง 2 มิติ เช่นพื้นที่ของอุทยานแห่งชาติ ประเทศ หรือทะเลสาบ และอาจจะประกอบด้วยชิ้นส่วนย่อยหลายชิ้นมาเชื่อมต่อกัน ซึ่งโครงสร้างพื้นฐานที่ใช้ใน Object-Based Model แสดงได้ในรูปที่ 3-1



รูปที่ 3-4 โครงสร้างพื้นฐานที่ใช้ใน Object-Based Model

2. Field-Base Model ในการจำลองข้อมูลรูปแบบนี้เราให้ความสนใจในตัวพื้นที่นั้น หรือต้องการแสดงข้อมูลเกี่ยวกับตัวพื้นที่นั้น เช่นข้อมูลการใช้พื้นที่ หรือการแบ่งเขตต่างๆ ของประเทศ ซึ่งตัวแทนในการแสดงความสัมพันธ์เชิงพื้นที่ที่เกี่ยวข้องกันของข้อมูลคือ พาร์ทิชัน (Partitions) และเน็ตเวิร์ค (Network) โดยพาร์ทิชันจะใช้ในการแสดงถึงส่วนของพื้นที่ที่ต้องมีการเชื่อมต่อกัน และส่วนที่ติดกันนั้นเป็นส่วนที่เราให้ความสนใจ และเน็ตเวิร์คจะเป็นกราฟที่ถูกฝังลงในพื้นที่ ประกอบด้วยเซตของจุดซึ่งแสดงตัวเป็นแต่ละโนดของกราฟหรือเซตของเส้นซึ่งแสดงถึงขอบของพื้นที่ที่เราสนใจ



รูปที่ 3-5 โครงสร้างพื้นฐานที่ใช้ใน Field-Base Model

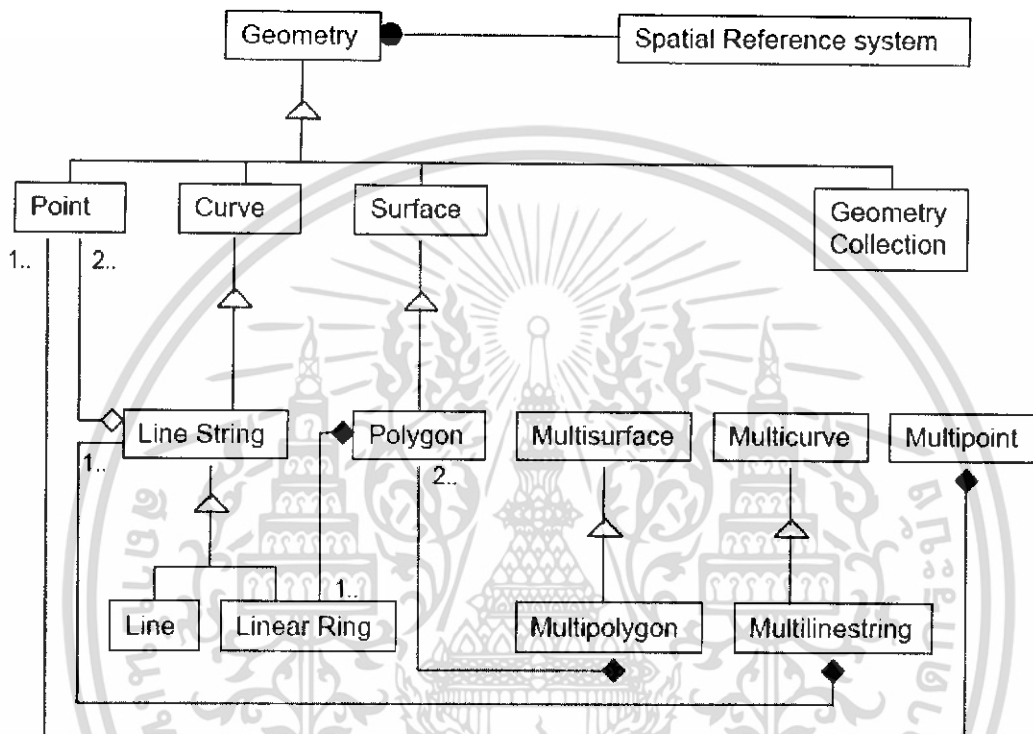
3.2.2 Spatial Data Type

ระบบของชนิดของข้อมูลเชิงพื้นที่ (Spatial data Type or spatial algebras) คือระบบที่ใช้ในการรวมนิยามพื้นฐานสำหรับจุด (point), เส้น (line) และพื้นที่ (region) เข้าด้วยกันกับความสัมพันธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และวิธีการจัดการขององค์ประกอบนั้นๆ เช่น การกำหนดรูปแบบการตัดผ่านหรือซ้อนทับกัน (intersection) ของพื้นที่ เป็นต้น

ทางเลือกของชนิดของข้อมูลที่ใช้กันใน Objects-Based Model ของข้อมูลเชิงพื้นที่คือชนิดของข้อมูลพื้นฐานทั่วไปที่ใช้ในระบบแผนที่ ซึ่งในรูปที่ 3-3 แสดงถึงโครงสร้างและความสัมพันธ์ภายในของ spatial geometry ในรูปแบบของ UML



รูปที่ 3-6 โครงสร้างและความสัมพันธ์ภายในของ Spatial geometry ในรูปแบบของ UML

3.2.3 Spatial Relationships

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

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

3.3 postgresSQL

เป็น Object-Relational DBMS โดยสามารถใช้รูปแบบของภาษา SQL ได้เกือบทั้งหมด และสามารถให้ subselects , transactions , user-defined types และ functions ได้ อีกทั้งเป็น Database ซึ่งให้ Source code ฟรี ด้วย(Open Source) ซึ่ง PostgreSQL เป็น DBMS ตัวหนึ่ง ที่มี คุณสมบัติและคุณลักษณะ ที่ไม่ได้ด้อยไปกว่า DBMS ราคาแพงอื่นๆ เลย ในทางกลับกัน หลายๆ คุณลักษณะอาจหาไม่ได้บน DBMS เิงการค้าทั้งหลายและ Postgres ถูกจัดอยู่ในกลุ่มของ object-relational ซึ่งมีความแตกต่างไปจาก object-oriented

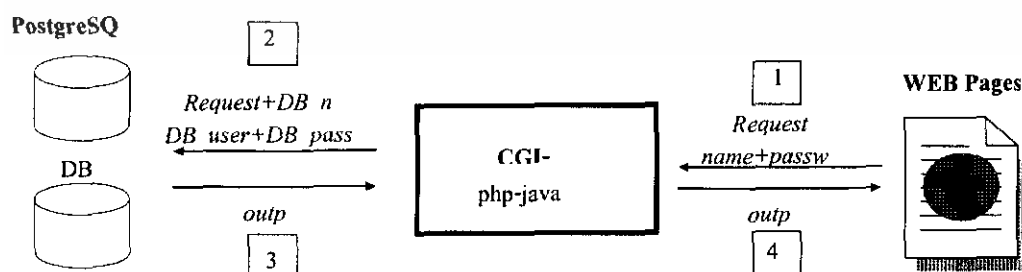
ลักษณะโครงสร้าง

ระบบที่ใช้ PostgreSQL จะติดตั้ง PostgreSQL ไว้ที่เครื่อง Server ซึ่งเป็นที่เก็บ database ด้วย และยังสามาร ติดตั้ง PostgreSQL ได้มากกว่า 1 ชุดใน Server เครื่องเดียว ผู้ดูแลระบบ PostgreSQL จะใช้ชื่อว่า postgres ซึ่งเป็นผู้ดูแลทั้ง ตัวโปรแกรม และ database ซึ่งสามารถทำงานกับบางคำสั่งเฉพาะ เพื่อจัดการ database และ ผู้ใช้บริการ (user) ซึ่ง ผู้ดูแลระบบ database (postgres) จะคล้ายการทำงานของ superuser ในระบบ Unix หน้าทีของ postgres สามารถ สร้างชื่อ user และกำหนดสิทธิและระดับการใช้งานต่างๆ ได้ PostgreSQL ใช้รูปแบบการทำงาน แบบ Client/Server ซึ่งในการทำงานจะประกอบด้วย 3 process ทำงานร่วมกัน คือ

1. Postmaster เป็น supervisory daemon process ซึ่งจัดการติดต่อระหว่าง Frontend กับ Backend process ในการ allocate share buffer , จัดการค่าเริ่มต้นต่างๆในระหว่างเริ่มทำงาน และเก็บบันทึกการเข้าใช้ระบบและความผิดพลาดต่างๆที่เกิดขึ้น
2. Postgres เป็น backend process เพื่อจัดการ database ถือว่าส่วนนี้เป็น process ที่ทำงานจริงๆ เช่น ทำงานตาม query โดย Postmaster จะสั่งให้สร้าง Backend process สำหรับทุกๆ การเชื่อมต่อกับ Frontend ดังนั้น Postgres นี้จะทำงานที่ server
3. Frontend เป็น application ซึ่งจะทำงานที่เครื่อง client และจะส่งคำสั่งการเชื่อมต่อหรือคำสั่งต่างๆ มาที่ Postmaster แล้ว Postmaster จึงส่งต่อการทำงานไปที่ Postgres

การใช้ PostgreSQL กับ Web Page

ลักษณะการทำงานจะเป็นดังรูป



รูปที่ 3-7 ลักษณะการทำงานของ PostgreSQL ติดต่อกับ Web page

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้คำสั่ง สร้าง Database และ สร้าง ผู้ใช้

การสร้าง ผู้ใช้ใหม่ ทำโดยใช้ Login postgres และใช้คำสั่ง createuser จากนั้นป้อน รายละเอียดของผู้ใช้ โดยสามารถใช้ชื่อเดียวกับ ชื่อของ unix ก็ได้แต่ไม่ควร เหตุผลในเรื่องความปลอดภัย และไม่ควรถูกกำหนดให้ผู้ใช้ นั้น สามารถสร้าง ผู้ใช้อื่น ต่อได้อีก การสร้าง database ใหม่ ทำโดยใช้คำสั่ง createdb โดยควร จะ login เป็น ชื่อผู้ใช้ที่จะเป็นเจ้าของ database นั้น

■ การสร้างตารางและกำหนดสิทธิ์ของผู้ใช้

การกำหนด ตารางที่ใช้ จะกำหนดขึ้นภายหลัง และจะต้อง grant สิทธิการใช้จากตาราง โดยมีข้อกำหนดดังนี้ admin ของแต่ละหน่วยงาน สามารถ เพิ่ม/ลบ/แก้ไข ตารางได้, สามารถ เรียกดู/เพิ่ม/ลบ/แก้ไข ข้อมูลในตารางได้ ทำโดย admin สร้าง table ใน database จะมีสิทธิในการ ในเพิ่ม/ลบ table อยู่แล้ว จากนั้น admin กำหนดสิทธิ์ในการเข้าใช้ตาราง คือ

- REVOKE {select,insert,update,delete,rule,all} ON {table,view,sequence,index} FROM {public | GROUP group | username} เช่น

```
REVOKE all ON rainfall_table FROM rid_user ;
```

- GRANT {select,insert,update,delete,rule,all} ON {table,view,sequence,index} TO {public | GROUP group | username} เช่น

```
GRANT insert,update,delete,select ON rainfall_table TO rid_user ; หรือ
```

```
GRANT select ON rainfall_table TO rid_nobody ;
```

ในโครงการนี้ได้ใช้ความสามารถทางด้าน Object-relational ของ PostgreSQL ร่วมกับ โมดูล PostGIS เพื่อจัดการกับข้อมูลแผนที่ PostGIS เป็นตัวที่จะมาสนับสนุนการทำงานของ PostgreSQL ที่เกี่ยวกับข้อมูลเชิงพื้นที่ งานที่นิยมนำไปใช้ก็จะเป็นฐานข้อมูลเชิงพื้นที่สำหรับ GIS ในโครงการนี้ได้นำมาใช้เก็บข้อมูลแผนที่

ตัวอย่าง GIS Object อย่างง่ายที่ support โดย PostGIS ซึ่งถูกกำหนดโดยองค์กร OpenGIS Consortium (OGC) รวมทั้งการกำหนดฟังก์ชันต่างๆ ไว้คอยจัดการกับ GIS Object และอื่น ๆ

- POINT(0 0 0)
- LINESTRING(0 0,1 1,1 2)
- POLYGON(((0 0 0,4 0 0,4 0 0,0 4 0,0 0 0),(1 1 0,2 1 0,2 2 0,1 2 0,1 1 0)))
- MULTIPOINT(0 0 0,1 2 1)
- MULTILINESTRING((0 0 0,1 1 0,1 2 1),(2 3 1,3 2 1,5 4 1))
- MULTIPOLYGON((((0 0 0,4 0 0,4 0 0,0 4 0,0 0 0),(1 1 0,2 1 0,2 2 0,1 2 0,1 1 0)),((-1 -1 0,-1 -2 0,-2 -2 0,-2 -1 0,-1 -1 0)))
- GEOMETRYCOLLECTION(POINT(2 3 9),LINESTRING((2 3 4,3 4 5)))

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการใช้งานจะมี 2 ตารางมาตรฐานที่มากับ PostGIS ดังนี้ คือ

1. SPATIAL_REF_SYS
2. GEOMETRY_COLUMNS

สองตารางนี้ เก็บ Numeric IDs และ คำอธิบายระบบพิกัด (coordinate systems) ที่ใช้ในฐานข้อมูลเชิงพื้นที่
ตัวอย่างการใช้ภาษา SQL

```
db=# SELECT id, AsText(geom) AS geom, name FROM ROADS_GEOM;
id | geom | name
-----+-----+-----
1 | LINESTRING(191232 243110,191108 243242) | Jeff Rd
2 | LINESTRING(189141 244150,189265 244017) | Geordie Rd
3 | LINESTRING(192783 228138,192612 229814) | Paul St
4 | LINESTRING(189412 252431,189631 259122) | Graeme Ave
5 | LINESTRING(190131 224140,190871 228134) | Phil Tce
6 | LINESTRING(198231 263410,198213 260322) | Dave Cres
7 | LINESTRING(218421 284121,224123 241231) | Chris Way
(6 rows)
```

รูปที่ 3-8 แสดงตัวอย่างการใช้ภาษา SQL ใน PostgreSQL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

BREW

4.1 BREW (Binary Runtime Environment for Wireless)

หลายล้านคนทั่วโลกเห็นว่าโทรศัพท์เป็นเครื่องมือสื่อสารที่จำเป็นอีกอันหนึ่ง เทคโนโลยี BREW (Binary Runtime Environment for Wireless) ถือเป็น Hardware platform ใหม่โดยเฉพาะประเทศไทยและกำลังเริ่มนิยมในการเขียนโปรแกรมบน mobile phones แบบใหม่ซึ่งจะ กลายเป็นที่นิยมในโลกของการสื่อสารไร้สายและโทรศัพท์มือถือ มีผู้ผลิตมือถือรายใหญ่หลายรายสนใจที่จะผลิตอุปกรณ์มือถือที่มีความสามารถในการรองรับสนับสนุนเทคโนโลยี BREW นี้ ปัจจุบัน BREW มีการพัฒนามาแล้ว 4 version หลักได้แก่ 1.0 1.1 2.0 3.0 แต่เครื่องมือถือในประเทศไทยปัจจุบันรองรับแค่เวอร์ชัน 2.0 เท่านั้นเอง

อีกตัวหนึ่งที่เกี่ยวข้องกับเทคโนโลยี BREW คือ SDK ตัว BREW SDK นี้เป็น library ที่โปรแกรมเมอร์สามารถใช้ในการสร้าง application สำหรับ BREW Platform โดยใช้ภาษา C หรือ C++ แต่ส่วนมากแล้วและตัวอย่างส่วนมากเป็นภาษา C ซึ่ง SDK ได้มีฟังก์ชันที่ใช้ส่วนมากประกอบด้วยฟังก์ชันทางด้าน Graphics, Networking, Sound, Text Display, GUIs และอื่นๆอีกมากมายและจุดเด่นของมันคือมี Emulator ให้ใช้ในการทดลองโดยที่สามารถนำมาทดลองโปรแกรมที่เราเขียนขึ้นมาบนเครื่องคอมพิวเตอร์ที่มีระบบปฏิบัติการ Windows แทนที่จะต้องทดลองไปทดลองโปรแกรมกับอุปกรณ์ของจริงทำให้ง่ายต่อการทดลองและตรวจแก้ไขโปรแกรม มันสามารถที่จะกำหนดคุณสมบัติของ Emulator ได้เช่นกำหนดขนาดของหน่วยความจำ ความละเอียดของหน้าจอเป็นต้น แต่ Emulator ตัวนี้ก็ไม่สามารถที่จะทดลองได้เหมือนอุปกรณ์จริงทุกประการ

และสิ่งที่ BREW แตกต่างไปจากเทคโนโลยีอื่น เช่น J2ME คือการเผยแพร่และการกำหนดมาตรฐานของ Application ที่จะทำงานบนเทคโนโลยี BREW เป็นลักษณะที่จะมีองค์กรกลางที่คอยให้มาตรฐานของ Application นั้นที่จะเอาขึ้นไปทำงานบนอุปกรณ์จริงซึ่งจะระบบระเบียบของการขออนุญาตนี้เช่นต้องเป็นสมาชิกหรือต้องมีการทดสอบ Application กับทางองค์กรก่อน

4.1.1 หลักการพื้นฐานของ BREW Platform

ลักษณะของ BREW ในมุมมองของผู้ที่พัฒนา Application หลักๆ มีดังนี้

1. BREW เป็นกลุ่มของฟังก์ชันการทำงาน(APIs)ที่ใช้ในการเขียนโปรแกรมสำหรับ Mobile devices

2. มีฟังก์ชันการทำงานมากมายที่มีให้ใช้ทั้งเข้า access อุปกรณ์เช่น Display , การจัดการเกี่ยวกับ Memory , การจัดการเกี่ยวกับ File เป็นต้น
3. เป็นตัวกลางไม่ให้โปรแกรมเมอร์ต้องไปจัดการกับอุปกรณ์ภายในของ Mobile devices

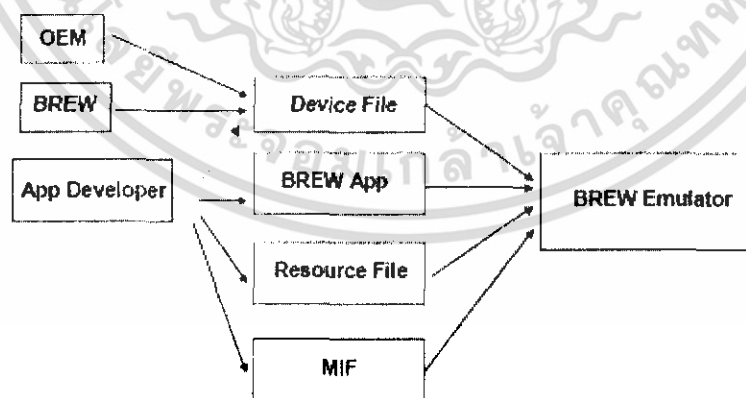
4.1.2 ระบบ BREW โดยรวม

1. BDS for Carriers
2. Open platform for device OEMs
3. BREW SDK application developers
4. Application for end-user download

4.1.3 สถาปัตยกรรมและระบบการทำงานของ BREW



รูปที่ 4-1 แสดง Device Architecture (Layer) ของ BREW



รูปที่ 4-2 แสดง ระบบการทำงานของ BREW

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.4 BREW SDK COMPONENTS

1. BREW Emulator ใช้ในการทดลองโปรแกรมที่เขียนบนเครื่องคอมพิวเตอร์
2. BREW Resource Editor เป็นโปรแกรมช่วยสร้าง Resource ที่จะใช้ใน Application ที่สร้าง
3. BREW MIF Editor เป็นโปรแกรมที่ช่วยสร้าง module information files(MIF) ที่จะเป็นตัวบอกว่า Application ที่จะสร้างขึ้นนั้นจะมีลักษณะแบบไหน
4. BREW Compressed Image Authoring Tool เป็นโปรแกรมที่ช่วยในการสร้างภาพที่เป็นภาพเคลื่อนไหวที่จะใช้ใน BREW เทคโนโลยี

4.1.5 MODULE INFORMATION FILE (MIF)

เป็นไฟล์ที่บอกถึงลักษณะของ Application ที่โปรแกรมเมอร์ที่จะสร้างและบอกว่าจะใช้งาน Module ไหนในการสร้าง Application แต่ละ Module มีการร้องขอสิ่งต่างๆ ดังนี้

1. หมายเลขของ Application
2. ClassID ของแต่ละ Module ที่มีใช้ Application
3. รวมทั้ง Title, Icons ที่จะใช้ในแต่ละ Application

4.1.6 ความแตกต่างระหว่าง Handset กับ Emulator

1. Speed เน้นในเรื่องความเร็ว Emulator จะทำงานเร็วกว่าอุปกรณ์จริง (Handset)
2. Appearance ความแตกต่างในการแสดงผลของ Emulator และ Handset
3. Memory เรื่องของหน่วยความจำ Emulator มีมากกว่า Handset ซึ่ง Emulator สามารถกำหนดได้

ดังนั้นในขั้นตอนการพัฒนาโปรแกรมต้องคำนึงถึงเรื่องนี้ด้วยเพราะอาจทดลองบน Emulator สามารถทำงานได้แต่พอเอาโปรแกรมมาลงบนอุปกรณ์จริงแล้วอาจไม่สามารถทำงานเนื่องจากเงื่อนไขต่างๆ ที่ Emulator มีความแตกต่างไปจากอุปกรณ์จริงเช่นเรื่องหน่วยความจำหรือจะเป็นเรื่องความเร็วของทั้งสองแบบที่มีความแตกต่างกัน

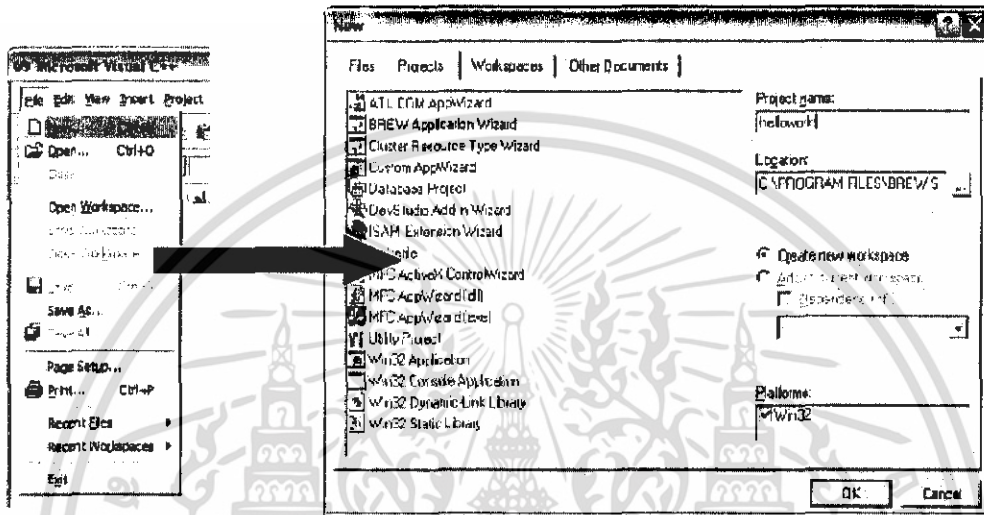
4.2 การเขียน BREW Application

การเขียน BREW Application ที่ถูกใช้ในการพัฒนาครั้งนี้ เราจะกระทำบน Microsoft Visual Studio 6.0 environment. ในส่วนนี้จะแสดงการแนะนำคร่าวๆ ของการพัฒนาและประมวลผลของ BREW รวมถึงขั้นตอนพื้นฐานที่จำเป็นถึงการเขียนเกี่ยวกับการทำหน้าที่ของ BREW Application

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

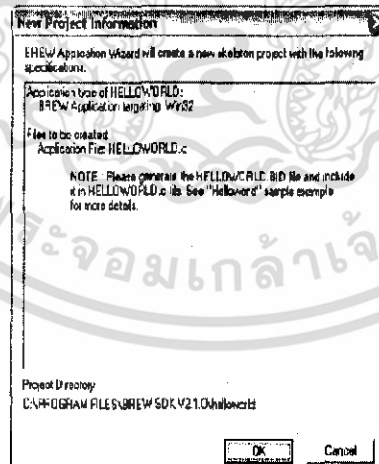
4.2.1 การสร้าง Application ใหม่ด้วย BREW Application Wizard

การใช้ BREW Application Wizard ทำให้สะดวกในการสร้าง Application ใหม่ เริ่มต้นใช้ BREW Application Wizard เข้าไปที่ Microsoft Visual Studio แล้วเลือกที่ File>New เพื่อสร้าง Project ใหม่ BREW Application Wizard จะแสดงให้ภายในชนิดเมนูโครงการ จากนั้นก็ใส่ชื่อของโครงการที่จะทำการพัฒนาแล้วก็เลือกที่ OK เพื่อเริ่มต้นการเขียน Application ดังรูปที่ 4-4



รูปที่ 4-3 แสดง BREW Application Wizard

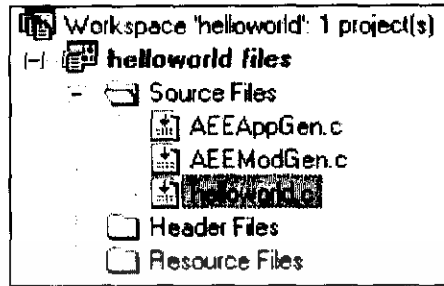
จากนั้นเมื่อการสร้าง Project ใหม่ก็จะมีกรรายนสรูปการกระทำต่างๆที่ได้เลือกไป



รูปที่ 4-4 สรุป Information

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นก็เปิดไฟล์ที่ได้ตั้งชื่อไว้ในตัวอย่างนี้มีไฟล์ที่ชื่อว่า helloworld.c (ชื่อไฟล์ยกตัวอย่าง) เป็นไฟล์หลักๆที่ใช้ในการพัฒนา Application



รูปที่ 4-5 แสดง Workspace

ส่วนไฟล์ AEEAppGen.c, AEEModGen.c และ library "ไฟล์เหล่านี้เป็นไฟล์ที่ใช้ในการสนับสนุนการพัฒนา BREW Application ตัวอย่างโครงสร้างของ Code ที่ถูก Generated โดย BREW Application Wizard

```
#include "AEEModGen.h" // Module interface definitions ①
#include "AEEAppGen.h" // Applet interface definitions
#include "AEEShell.h" // Shell interface definitions

#include "helloworld.bid" ②
typedef struct _helloworld { ③
    // add your own variables here...
    ...
} helloworld;
int AEEClsCreateInstance(AECLSID ClsId, IShell *pIShell, ④
    IModule *po, void **ppObj) {
    ...
}
static boolean helloworld_HandleEvent(helloworld* pMe, ⑤
    AEEEvent eCode, uint16 wParam, uint32 dwParam) {
    ...
}
boolean helloworld_InitAppData(helloworld* pMe) { ⑥
    ...
}
void helloworld_FreeAppData(helloworld* pMe) { ⑦
    ...
}
```

รูปที่ 4-6 แสดง Code ที่ถูก Generated โดย BREW Application Wizard

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความหมายและความสำคัญในแต่ละส่วนของ Code ที่ถูก Generated โดย BREW Application Wizard ซึ่งโดยทั่วไปแล้วจะมีอยู่ประมาณ 7 ส่วน ตามสัญลักษณ์หมายเลขที่แสดงอยู่ด้านหลัง code ดังนี้

1). API Library Includes

เป็นส่วนที่ประกาศว่าไฟล์ที่ถูกรวมเข้ามาซึ่งเป็นไฟล์ที่ใช้เฉพาะเจาะจงของ BREW API และไฟล์ Header (*.h) จำเป็นต้องถูกประกอบด้วยในโปรแกรม ซึ่งส่วนของข้อมูลบนไฟล์ Header นั้นจะถูกบรรจุคำอธิบายใน BREW API Reference. ข้อแนะนำในการป้องกันการใช้อินโฟร์เมชันที่ตัวแปร Global และ ตัวแปร Static ไม่ควรใช้ C Standard Library ใน BREW Application และช่วยเพิ่มเติมคือ การคำนวณเกี่ยวกับ floating-point นั้นควรใช้ floating-point helper function.

2). Class ID file

ก่อนการ Compile Application เรานั้นจะต้องมีการ Generate ไฟล์ Class ID. ไฟล์นี้จะถูกบรรจุไปด้วยเลขฐาน 16 จำนวน 32 บิต ที่เป็นตัวเลข Application ที่จะต้องไม่ซ้ำกัน ซึ่งถูกใช้สร้างเป็นกลไกในการติดต่อของ BREW สำหรับการทดสอบบนระบบคอมพิวเตอร์ ซึ่งมันพอเพียงต่อการ Generate Local Class ID ผ่านทาง MIF Editor.

3). Applet Structure

Objects ใดๆ ที่ถูกใช้ในระหว่างการทำงานของโปรแกรม ควรจะถูกวางในที่ applet structure. เช่นตัวแปร Static และตัวแปร global จะไม่ถูกยอมรับภายใน BREW, applet structure จัดเตรียมกลไกสำหรับสถานที่เก็บเอกสารสำคัญๆ Element ของ Applet ทั้งหมดจะถูก initialized โดย Method Initial data ของ Application นั้นๆ

4). AEEClsCreateInstance() Method

Method นี้เป็นที่ต้องการสำหรับ Application ทั้งหมด. BREW ถูกร้องขอ routine นี้โดยอัตโนมัติในการสร้าง Instance. ภายใน Method AEEAppletNew() นี้คือ เรียกว่าหน้าที่การ register event handler, กระบวนการ initialization ข้อมูลและการ deallocation ข้อมูล. กระบวนการ Initialization ทั้งหมด คือถูกกระทำเจาะจงโปรแกรมในการ initialization data. Method AEEClsCreateInstance() มันควรจะไม่ต้องถูกแก้ไข.

5). Event Handler

Application ทั้งหมดของ BREW จะต้องบรรจุ event handler function ที่ใช้ในการเรียกใช้ AEEAppletNew() ภายใน method AEEClsCreateInstance(). Event จะถูกส่งมายัง function นี้โดย BREW Layer. หลังจากได้รับ Event ควรจัดการให้มีการ return TRUE เพื่อที่เป็นการบอกว่าเป็นการจัดการที่สมบูรณ์หรือ return FALSE ถ้า event ไม่มีการถูกจัดการ. Event codes supports โดย BREW (AEE Event) ที่ถูกแสดงในส่วน data type ของ BREW API Reference.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6). Applet data init method

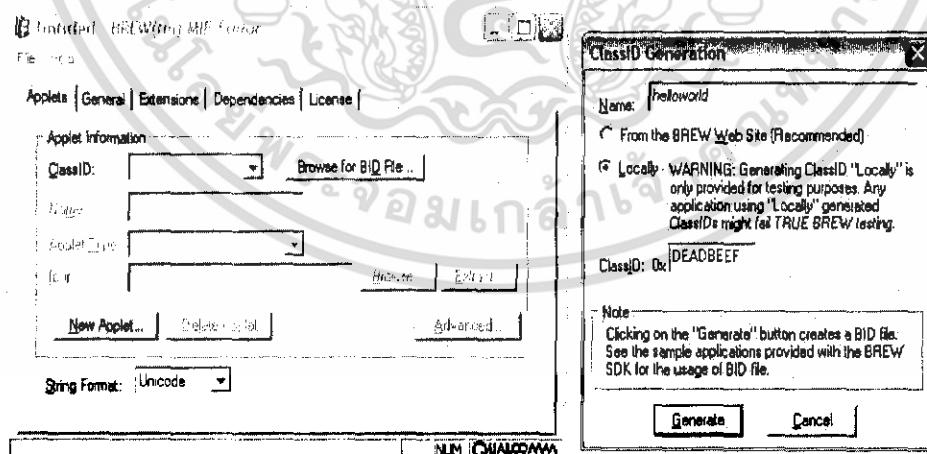
Method นี้ถูกใช้ในการ Initialized code ที่ถูกร้องขอโดย applet. โดยปกติแล้ว Function นี้จะถูกใช้ในการยกตัวอย่าง member applet structure หรือสำหรับการ allocate memory สำหรับ buffer. การ Return ค่าซึ่งเป็นค่า Boolean นั้นถูกใช้ในการประมวลผลไม่ว่าจะเสร็จสมบูรณ์หรือไม่สมบูรณ์. โดยที่ Place code การจัดสรรของคุณใน method เดียวจะลดการละเลยที่อาจจะมีผลต่อสมรรถภาพ member ของ applet structure.

7). Applet data free method

Applet data free method นี้ถูกใช้ในการ deallocate ของทรัพยากรทั้งหมดและการเพิ่มขึ้นตอนการถูกร้องขอสำหรับการ safety terminate application. ทรัพยากรทั้งหมดที่ Allocate มาจาก method initialization data ควรจะถูก deallocated ภายใน method นี้.

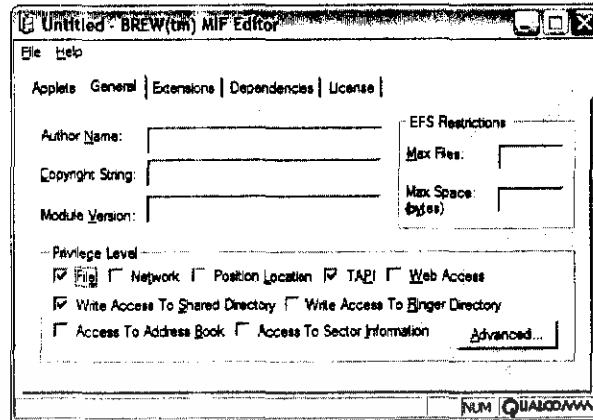
4.2.2 การใช้ BREW MIF Editor

จะต้องมีการสร้าง MIF ก่อนที่จะวาง Application ไปไว้ใน BREW Device หรือ BREW Simulator. MIF จะบรรจุสิ่งที่มีความสำคัญคือ Information ที่เกี่ยวกับ Application ดังเช่น ชื่อ, Icon, Class ID และ privilege levels. การวางไฟล์ต่างๆในโฟลเดอร์ ที่เป็นโฟลเดอร์ของ Application ก็ต้องวาง MIF ไฟล์ไว้ระดับเดียวกันกับโฟลเดอร์ที่เก็บไฟล์ที่เป็นไฟล์รันโปรแกรม BREW MIF Editor ถูกบรรยายลักษณะการใช้ในส่วนนี้เป็นเครื่องมือที่ช่วยให้ง่ายต่อการสร้างไฟล์ MIF. ดังแสดงเป็นขั้นตอนดังรูปต่อไปนี้



รูปที่ 4-7 แสดงการ Generate Local class ID

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

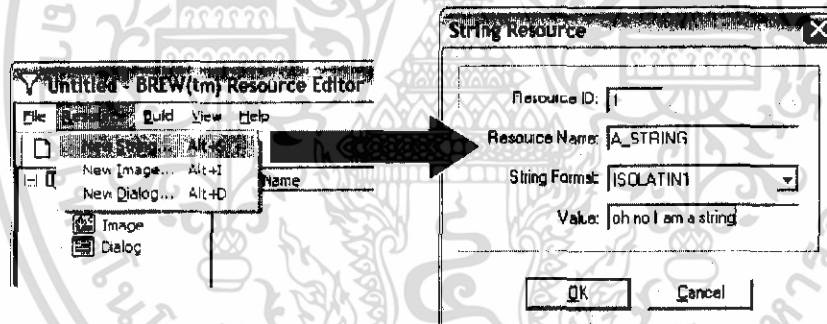


รูปที่ 4-8 แสดง Specify setting privileges

4.2.3 การใช้ BREW Resource Editor

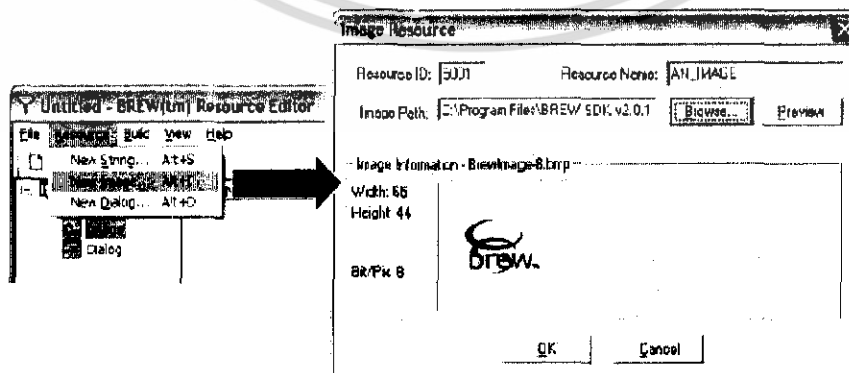
BREW Resource Editor เป็นเครื่องมือที่ใช้ในการสร้างไฟล์ที่เอาไว้เก็บ resource ที่เพื่อนำมาใช้ในโปรแกรม โดยจะประกอบไปด้วย resource 3 ชนิดคือ string resource , image resource และ dialog box.

- การสร้าง String resource



รูปที่ 4-9 แสดงการเพิ่ม string resource

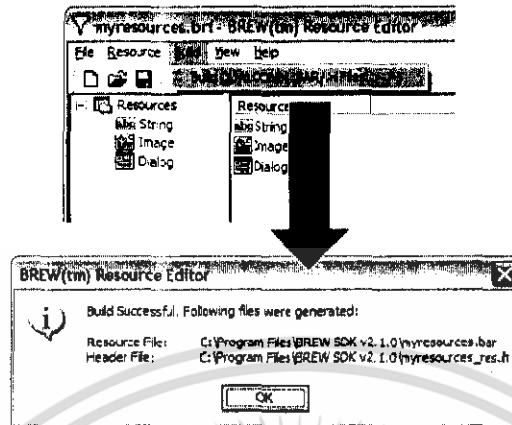
- การสร้าง Image resource



รูปที่ 4-10 แสดงการเพิ่ม Image resource

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

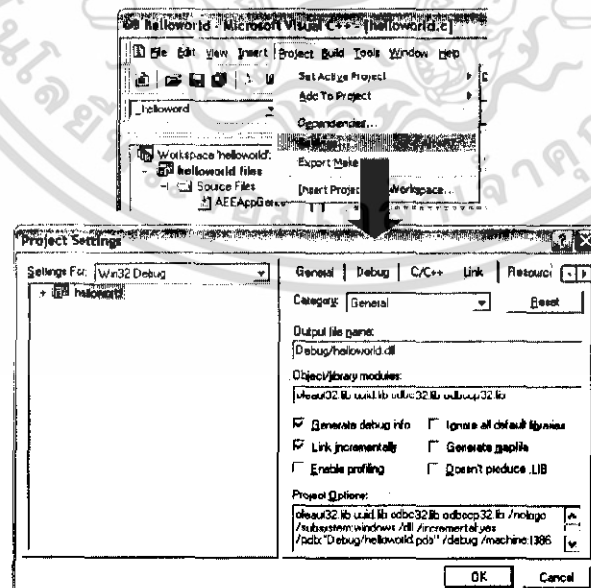
จากนั้นเมื่อสร้าง resource ทั้งหมดตามที่ต้องการแล้วจะต้องมีการ compile ไฟล์ที่ใช้ในการเก็บ resource นี้ด้วย



รูปที่ 4-11 แสดงการ Compile resource file

4.2.4 การ Compile BREW Application เพื่อใช้ในการจำลองการใช้งาน (Simulation)

หลังจากที่เราได้เขียน BREW Application แล้วจะต้องมีการสร้างไฟล์ DLL เพื่อเอามาใช้สำหรับ Simulator และอย่างแรกต้องแน่ใจก่อนว่าได้มีการสร้าง MIF ไฟล์แล้วเก็บไว้ในระดับพาร์ตเดียวกันของ Application และได้สร้างไฟล์ class ID แล้ว ซึ่งโดย default ของไฟล์ DLL นั้นจะถูกกระบอบอยู่ใน directory ที่มีชื่อว่า Debug (Debug/<projectname.dll>) ซึ่งจะทำให้ตัว Simulator นั้นไม่สามารถ execute application ได้ ดังนั้นจะต้องมีการระบุให้มีการสร้างไฟล์ DLL ใน directory ของ project นั้น โดยถูกแสดงไว้ดังนี้



รูปที่ 4-12 แสดงการ Setting output file .dll

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในช่องของ Output file name: นั้นให้ทำการลบคำว่า Debug/ ออก เพื่อให้การ compile program แล้วสร้างไฟล์ *.dll ให้อยู่ใน directory project นั้นๆ ซึ่งไฟล์ *.dll นี้ เป็นไฟล์ที่ซึ่งถูกเรียกใช้โดย Simulation ในการทดสอบการทำงานของ Application

4.2.5 การ Run BREW Application เพื่อใช้ในการจำลองการใช้งาน (Simulation)

หลังจากที่มีการ Configure ค่าต่างๆ ที่จะถูกใช้โดย Simulator แล้วและแน่ใจว่าได้สร้าง MIF file ที่เก็บไว้ใน directory ถูกต้องแล้วก็ทำการรัน Application ได้ทันที Build > Start debug > go หรือ กด F5



รูปที่ 4-13 แสดง Application Menu

การวางไฟล์ต่างๆ ในโฟลเดอร์ ที่เป็นโฟลเดอร์ของ Application คือต้องวาง MIF ไฟล์ไว้ระดับเดียวกันกับโฟลเดอร์ที่เก็บไฟล์ที่เป็นไฟล์รันโปรแกรม

บทที่ 5

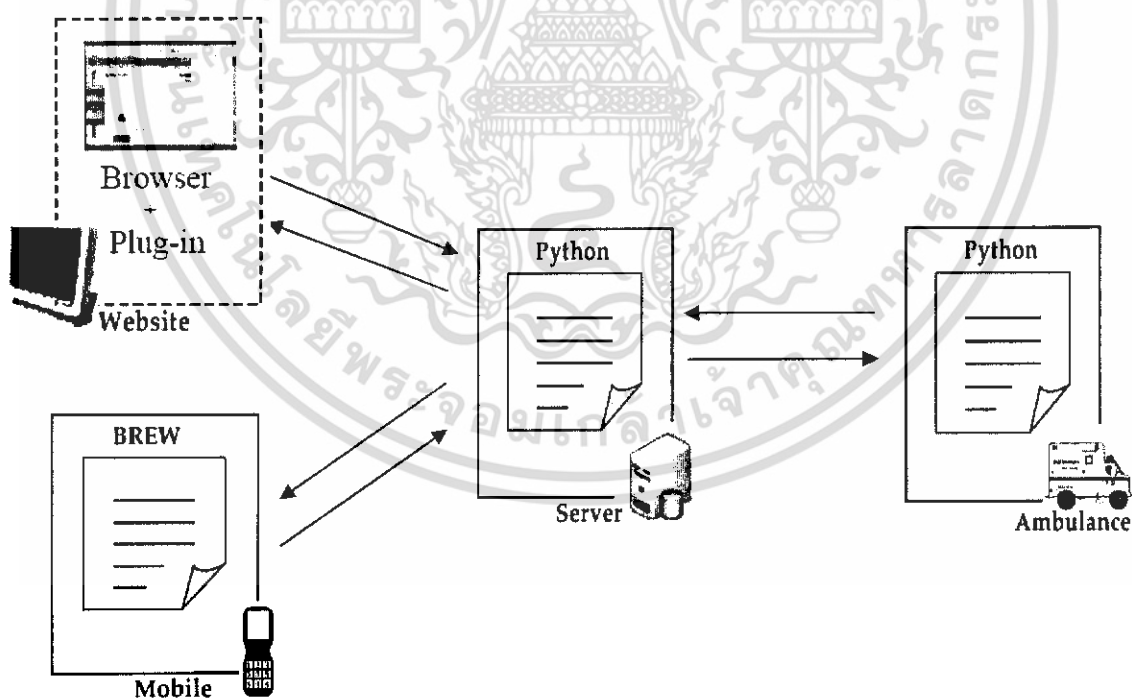
การออกแบบโครงสร้างของระบบ

ในบทนี้จะขอกล่าวถึงการออกแบบโครงสร้างของระบบเรียกรถฉุกเฉิน ตลอดจนการสร้างส่วนประกอบต่างของระบบ

5.1 การออกแบบโครงสร้างระบบเรียกรถฉุกเฉิน

ระบบเรียกรถฉุกเฉินที่พัฒนาขึ้นประกอบด้วย 5 ส่วน คือ

1. ส่วนของเว็บเซิร์ฟเวอร์ จะให้บริการข้อมูลแผนที่พร้อมทั้งคอยจัดสรรและติดตามรถฉุกเฉินให้เมื่อมีการเรียก
2. ส่วนของเว็บไซต์ใช้สำหรับการเรียกรถฉุกเฉินและการติดตามคูการเดินทางของรถฉุกเฉิน
3. ส่วนของรถฉุกเฉิน เป็นอุปกรณ์แจ้งตำแหน่งและแสดงผลสำหรับผู้ขับรถฉุกเฉิน
4. ส่วนของโทรศัพท์เคลื่อนที่ระบบ CDMA เป็นอุปกรณ์ที่ใช้โทรติดต่อเรียกรถเรียกรถฉุกเฉิน
5. ส่วนของกล้อง IP



รูปที่ 5-1 แสดงโครงสร้างของระบบเรียกรถฉุกเฉิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของระบบเรียกรถฉุกเฉินมีลำดับขั้นตอนดังต่อไปนี้

1. ผู้ที่ต้องการเรียกรถฉุกเฉินสามารถทำการเรียกรถฉุกเฉินได้ 2 แบบ ดังนี้
 1. เรียกผ่านทางเว็บไซต์ โดยระบุจุดที่เกิดเหตุหรือจุดที่จะให้ไปรับผู้ป่วย
 2. เรียกผ่านทางโทรศัพท์เคลื่อนที่โดยเรียกใช้โปรแกรมที่อยู่ในส่วนของโทรศัพท์เคลื่อนที่
2. ถ้าผู้เรียกทำการเรียกผ่านทางเว็บไซต์แล้วเว็บไซต์ก็จะส่งข้อมูลตำแหน่งที่ระบุไปยังเซิร์ฟเวอร์ หรือถ้าผู้เรียกทำการเรียกผ่านทางโทรศัพท์เคลื่อนที่โปรแกรมในส่วนของโทรศัพท์เคลื่อนที่ก็จะทำการส่งข้อมูลตำแหน่งที่อยู่ในขณะทำการเรียกไปยังเซิร์ฟเวอร์เพื่อขอความช่วยเหลือ
3. ทุกๆ ขณะ โปรแกรมที่อยู่ในส่วนของรถฉุกเฉินแต่ละคันจะส่งข้อมูลตำแหน่งของตน มาให้เซิร์ฟเวอร์เพื่อการติดตามแบบ Real-time
4. เซิร์ฟเวอร์เมื่อได้รับการเรียกขอความช่วยเหลือในขั้นตอน 2 แล้วจะค้นหารถฉุกเฉินคันที่ว่าง และส่งข้อมูลการเรียกขอความช่วยเหลือไปยังรถฉุกเฉินคันนั้น
5. รถฉุกเฉินคันที่ถูกเรียกขอความช่วยเหลือก็จะเอาข้อมูลของการเรียกมาทำการแสดงบนแผนที่ว่าตำแหน่งของผู้ที่ต้องการความช่วยเหลือนั้นอยู่ที่ไหนพร้อมทั้งแจ้งไปยังเซิร์ฟเวอร์ว่าเห็นตำแหน่งของผู้ที่ต้องการความช่วยเหลือและพร้อมที่จะไปให้ความช่วยเหลือ
6. เมื่อรถฉุกเฉินกำลังให้ความช่วยเหลือก็จะแจ้งไปยังเซิร์ฟเวอร์ว่ากำลังให้ความช่วยเหลือเพื่อผู้เรียกหรือญาติของผู้ที่ต้องการความช่วยเหลือสามารถใช้รหัสของการเรียกขอความช่วยเหลือล็อกอินเข้าไปติดตามสถานการณ์การให้ความช่วยเหลือผ่านทางเว็บไซต์ได้แบบ Real-time
7. เมื่อสิ้นสุดการให้ความช่วยเหลือรถฉุกเฉินคันนั้นก็แจ้งไปยังเซิร์ฟเวอร์ว่าสิ้นสุดการให้ความช่วยเหลือแล้วเซิร์ฟเวอร์ก็จะตั้งค่าสถานะของรถให้เป็นสถานะว่างและพร้อมจะให้ความช่วยเหลือครั้งต่อไป

ระบบจะทำ 7 ขั้นตอนนี้ซ้ำแล้วซ้ำอีกเมื่อมีผู้เรียกขอความช่วยเหลือผ่านทางโทรศัพท์เคลื่อนที่หรือผ่านทางเว็บไซต์ ลำดับต่อไปจะเป็นการอธิบายถึงการสร้างส่วนประกอบต่างๆ ของระบบ

5.2 การสร้างส่วนประกอบต่างๆ ของระบบ

5.2.1 ส่วนของเว็บเซิร์ฟเวอร์ ในส่วนนี้ก็สามารถแบ่งออกเป็น 2 ส่วนใหญ่ๆ ดังนี้

5.2.1.1. ส่วนที่ให้บริการข้อมูลแผนที่ในรูปแบบของ XML ซึ่งจะถูกรหัสเรียกใช้จากส่วนเว็บไซต์ของระบบโดยข้อมูลแผนที่มีหลายชั้นดังนี้คือ

- ชั้นข้อมูลที่ 1 เป็นข้อมูลถนนสายหลัก
- ชั้นข้อมูลที่ 2 เป็นข้อมูลถนนสายรอง
- ชั้นข้อมูลที่ 3 เป็นข้อมูลของคลอง
- ชั้นข้อมูลที่ 4 เป็นข้อมูลของโรงพยาบาล
- ชั้นข้อมูลที่ 5 เป็นข้อมูลของโรงเรียน
- ชั้นข้อมูลที่ 6 เป็นข้อมูลของสถานีตำรวจ
- ชั้นข้อมูลที่ 7 เป็นข้อมูลของสถานีอนามัย
- ชั้นข้อมูลที่ 8 เป็นข้อมูลของบ้านและหมู่บ้าน
- ชั้นข้อมูลที่ 9 เป็นข้อมูลของกรม
- ชั้นข้อมูลที่ 10 เป็นข้อมูลของตลาด
- ชั้นข้อมูลที่ 11 เป็นข้อมูลของสถานีรถไฟ
- ชั้นข้อมูลที่ 12 เป็นข้อมูลของมหาวิทยาลัย

ข้อมูลทั้งหมดนี้ถูกเก็บอยู่รูปของฐานข้อมูลเชิงพื้นที่ ในส่วนของการให้บริการข้อมูลแผนที่นี้จะใช้ภาษา Python เขียนเป็นฟังก์ชันสร้างเอกสาร XML ของแต่ละชั้นข้อมูล โดยใช้ไลบรารี GDAL (Geospatial Data Abstraction Library) ซึ่งสามารถใช้งานด้วยภาษา Python ได้ ไปดึงเอาข้อมูลแผนที่จากฐานข้อมูลเชิงพื้นที่มา

ข้อมูลแผนที่ที่ใส่เก็บอยู่ในระบบพิกัด UTM WGS84 เหตุผลที่เลือกระบบนี้เนื่องจากเป็นระบบพิกัดที่ใกล้เคียงกับระบบพิกัดที่ใช้ในเครื่อง GPS

การออกแบบเอกสาร XML ของชั้นข้อมูลแต่ละชั้นเป็นดังนี้คือ

- ชั้นข้อมูลที่ 1 ข้อมูลถนนสายหลัก

```
<mainroad>
```

```
<road>
```

```
<line points="642019.046, 1547006.115 642045.027, 1546739.827
```

```
642065.822, 1546691.008 642097.313, 1546617.082"/>
```

```
<name>ชื่อถนน</name>
```

```
<length>ความยาวของถนน</length>
```

```
</road>
```

```
</mainroad>
```

แอดทริบิว points ของแท็ก line เป็นคู่ลำดับของจุด x1,y1 x2,y2 ...

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ชั้นข้อมูลที่ 2 ข้อมูลถนนสายรอง

<roads>

```
<road type="linestring" points="640214.053,1535775.177 640506.829,
1535775.177" />
```

</roads>

แอตทริบิวต์ points ของแท็ก road เป็นคู่ลำดับของจุด x1,y1 x2,y2 ...

แอตทริบิวต์ type ของแท็ก road เป็นชนิดของการแสดงผลซึ่งมี 2 ชนิดดังนี้

1. linestring เพื่อบอกว่าให้ทำการแสดงเป็นเส้น
2. polygon เพื่อบอกว่าให้ทำการแสดงเป็นพื้นที่

- ชั้นข้อมูลที่ 3 ข้อมูลของคลอง

<rivers>

```
<river type="linestring" points="649029.621,1546734.629 648746.376,
1546109.019 648677.618,1545952.546" />
```

</rivers>

แอตทริบิวต์ points ของแท็ก river เป็นคู่ลำดับของจุด x1,y1 x2,y2 ...

แอตทริบิวต์ type ของแท็ก river เป็นชนิดของการแสดงผลซึ่งมี 2 ชนิดดังนี้

1. linestring เพื่อบอกว่าให้ทำการแสดงเป็นเส้น
2. polygon เพื่อบอกว่าให้ทำการแสดงเป็นพื้นที่

- ชั้นข้อมูลที่ 4 ถึง ชั้นข้อมูลที่ 12 เป็นชั้นข้อมูลของสถานที่ซึ่งออกแบบเอกสาร XML ได้ดังนี้

<places>

<place>

```
<point pointx="701263.5" pointy="1532350.164"/>
```

```
<name>ชื่อสถานที่</name>
```

</place>

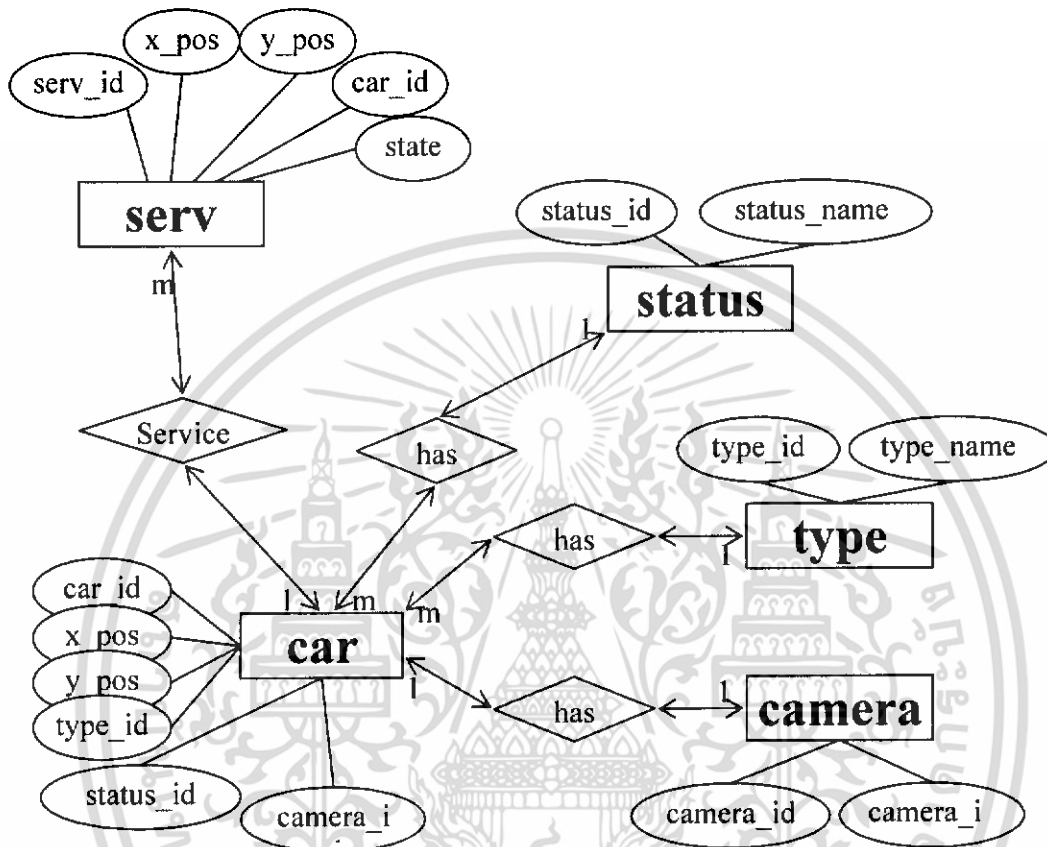
</places>

5.2.1.2 ส่วนของการจัดการรถฉุกเฉิน เมื่อมีการเรียกขอความช่วยเหลือเข้ามาในส่วนนี้ก็จะหารรถฉุกเฉินคันที่มีสถานะว่างและพร้อมที่จะให้ความช่วยเหลือแล้วส่งข้อมูลการเรียกขอความช่วยเหลือไปให้รถคันนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบฐานข้อมูล

รถฉุกเฉินมีด้วยกัน 3 ประเภท ได้แก่ รถพยาบาล รถดับเพลิง และรถตำรวจ ในการเรียกขอความช่วยเหลือแต่ละครั้งจะต้องมีรถฉุกเฉินประเภทที่ตรงกับผู้เรียกขอมาและมีสถานะที่ว่างเพื่อออกไปให้ความช่วยเหลือตามประเภทนั้นๆ ดังแสดง ER Diagram อย่างง่ายตามรูปที่ 5-2



รูปที่ 5-2 แสดง ER Diagram ของฐานข้อมูลรถฉุกเฉินและการเรียกขอความช่วยเหลือ

จากนั้นสร้างตารางได้ดังนี้

1. ตาราง car จะเก็บข้อมูลของรถฉุกเฉินทุกคันที่อยู่ในระบบรวมเช่นเก็บตำแหน่งของรถ เก็บสถานะเก็บประเภท

			FK	FK	FK
car_id	x_pos	y_pos	type_id	status_id	camera_id

- car_id ชนิดตัวอักษรความยาว 8 ตัวอักษร ใช้เป็นรหัสของรถแต่ละคัน
- x_pos ชนิดเลขจำนวนจริง ใช้เพื่อบอกพิกัดของรถในแกน X
- y_pos ชนิดเลขจำนวนจริง ใช้เพื่อบอกพิกัดของรถในแกน Y
- type_id ชนิดตัวอักษรความยาว 8 ตัวอักษร ใช้เพื่อบอกชนิดของรถ
- status_id ชนิดตัวอักษรความยาว 8 ตัวอักษร ใช้เพื่อบอกสถานะของรถ
- camera_id ชนิดตัวอักษรความยาว 5 ตัวอักษร ใช้รหัสของกล้องที่ติดบนรถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ตาราง status เก็บข้อมูลรายละเอียดเกี่ยวกับสถานะของรถฉุกเฉิน

status_id	status_name

- status_id ชนิดตัวอักษรความยาว 8 ตัวอักษร ใช้เพื่อบอกสถานะของรถ
- status_name ชนิด Text ใช้เพื่ออธิบายสถานะของรถ

3. ตาราง type เก็บข้อมูลรายละเอียดเกี่ยวกับประเภทของรถฉุกเฉิน

type_id	type_name

- type_id ชนิดตัวอักษรความยาว 8 ตัวอักษร ใช้เพื่อบอกชนิดของรถ
- type_name ชนิด Text ใช้เพื่ออธิบายชนิดของรถ

4. ตาราง camera เก็บข้อมูลของกล้อง

camera_id	camera_ip

- camera_id ชนิดตัวอักษรความยาว 5 ตัวอักษร ใช้เพื่อบอกรหัสของกล้อง
- camera_ip ชนิด Text ใช้เพื่อบอก IP Address ของกล้อง

5. ตาราง serv เก็บข้อมูลรายละเอียดของการเรียกขอความช่วยเหลือเช่นตำแหน่งที่เรียกและประเภทของการเรียก

serv_id	x_pos	y_pos	type_id	state

- serv_id ชนิดตัวอักษรความยาว 14 ตัวอักษร ใช้เป็นรหัสของการขอความช่วยเหลือ
- x_pos ชนิดเลขจำนวนจริง ใช้เพื่อบอกพิกัดของผู้เรียกในแกน X
- y_pos ชนิดเลขจำนวนจริง ใช้เพื่อบอกพิกัดของผู้เรียกในแกน Y
- type_id ชนิดตัวอักษรความยาว 8 ตัวอักษร ใช้เพื่อบอกชนิดของรถ
- state ชนิดตัวอักษรความยาว 2 ตัวอักษร ใช้บอกสถานะของการขอความช่วยเหลือ

ฟังก์ชันในส่วนของการจัดการรถฉุกเฉินมีดังนี้คือ

1. ฟังก์ชัน onaccept นี้จะเปลี่ยนสถานะของรถฉุกเฉินที่ตอบรับการเรียกจากสถานะว่างเป็นสถานะกำลังเดินทางไปยังจุดที่เรียก
2. ฟังก์ชัน onprocess จะเปลี่ยนสถานะของรถฉุกเฉินเป็นสถานะกำลังให้ความช่วยเหลือ
3. ฟังก์ชัน onready จะเปลี่ยนสถานะของรถฉุกเฉินเป็นสถานะว่างและพร้อมที่จะออกให้ความช่วยเหลือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ฟังก์ชัน online นี้จะเปลี่ยนตำแหน่งของรถฉุกเฉินเมื่อมีการเคลื่อนที่เพื่อการติดตามรถฉุกเฉินแบบ Real-time รถฉุกเฉินทุกคันจะส่งตำแหน่งของตนมาที่ฟังก์ชันนี้ตลอดเพื่อเปลี่ยนแปลงตำแหน่งของตนเอง
5. ฟังก์ชัน requests นี้จะคอยจัดการรถฉุกเฉินคันที่ว่างและตรงตามประเภทที่ผู้เรียกเรียกเข้ามาเพื่อให้ออกไปทำการช่วยเหลือตามประเภทนั้นๆ
6. ฟังก์ชัน ontrack เป็นฟังก์ชันที่คอยติดตามรถฉุกเฉินที่ออกไปให้การช่วยเหลือ
7. ฟังก์ชัน requestsfrommobile เป็นฟังก์ชันที่รอรับการเรียกจากโทรศัพท์เคลื่อนที่แล้วจัดการรถฉุกเฉินให้ตามประเภทที่เรียกขอเข้ามา

5.2.2. ส่วนของเว็บไซต์ใช้สำหรับการเรียกและการติดตามคูการเดินทางของรถฉุกเฉิน

1. การพัฒนาเว็บไซต์ส่วนของการแสดงผลแผนที่ ส่วนนี้เราได้ใช้ SVG ร่วมกับ Java Script แสดงผลแผนที่รวมทั้งการควบคุมแผนที่และใช้เทคโนโลยี AJAX (Asynchronous JavaScript And XML) ในการโหลดข้อมูลแผนที่จากเซิร์ฟเวอร์ส่วนที่ให้บริการข้อมูลแผนที่มาทำการแสดงผลที่ละชั้นข้อมูล
2. การพัฒนาเว็บไซต์ส่วนของการเรียกรถฉุกเฉิน ส่วนนี้จะเป็นการส่งข้อมูลการเรียกรถฉุกเฉินไปยังเซิร์ฟเวอร์ที่จัดการรถฉุกเฉินโดยรูปแบบการติดต่อสื่อสารจากเว็บไซต์ไปยังเซิร์ฟเวอร์ใช้การส่งผ่าน HTTP โพรโทคอลเพื่อให้ง่ายต่อการพัฒนาดังนั้นการส่งข้อมูลการเรียกจึงส่งไปในรูปแบบของ Query string ส่วนของข้อมูลการเรียกก็จะมีตำแหน่งที่ผู้เรียกระบุพร้อมทั้งประเภทของรถฉุกเฉิน
3. การพัฒนาเว็บไซต์สำหรับการติดตามการนำส่งผู้ป่วยหรือการให้ความช่วยเหลือ

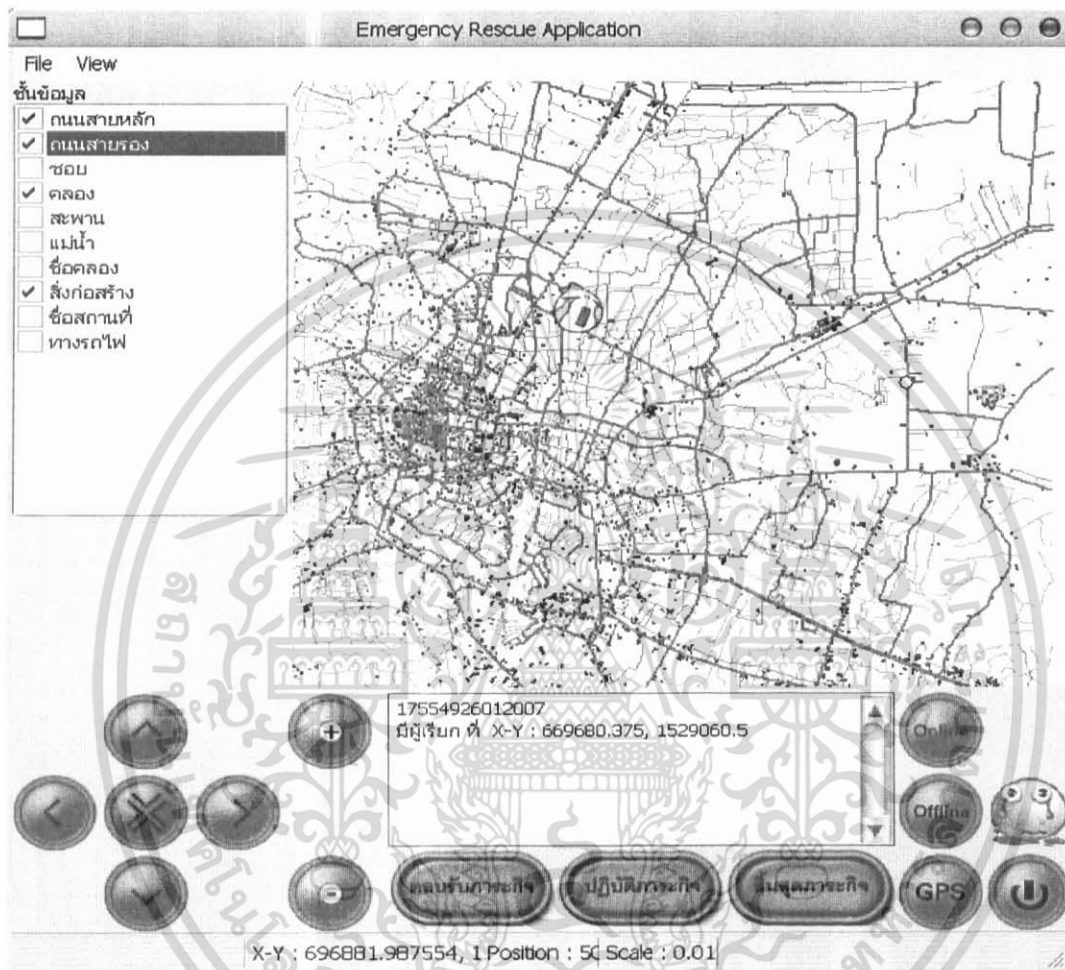
ในการพัฒนาเว็บไซต์สำหรับการติดตามนั้น หลักพื้นฐานในการออกแบบคือจำเป็นต้องสร้างเว็บไซต์ที่มีการทำงานแบบ Dynamic นั่นคือต้องมีความสามารถในการอัปเดตตัวเอง เพื่อแสดงผลการเดินทางในทุกๆช่วงเวลาหนึ่ง โดยความสามารถเหล่านี้ค่อนข้างจะเกินความสามารถที่ภาษา html จะทำได้ ดังนั้นเราจึงเลือก Java Script กับเทคโนโลยี AJAX เข้ามาช่วยในการแสดงผลคือ เราใช้ SVG กับ Java Script ในการแสดงผลแผนที่แล้วใช้ AJAX ไปขอตำแหน่งของรถฉุกเฉินคันที่กำลังให้ความช่วยเหลือตามรหัสการเรียกขอความช่วยเหลือจากผู้ใช้ที่ป้อนก่อนเข้าหน้าเว็บไซต์นี้

สำหรับการใช้งานเว็บไซต์นี้เริ่มจากผู้ใช้ซึ่งเป็นญาติมิตรของผู้ป่วยหรือผู้ประสพภัยที่ต้องการติดตามสถานการณ์การให้ความช่วยเหลือของเจ้าหน้าที่บนรถฉุกเฉินต้องป้อนรหัสการเรียกที่หน้าจอล็อกอิน โดยรหัสการเรียกนั้นจะเป็นรหัสที่ระบบได้สร้างไว้และส่งไปยังอีเมลที่กำหนด เมื่อล็อกอินเข้าไปแล้วเว็บไซต์นี้ก็จะไปเอาตำแหน่งของรถฉุกเฉินคันที่ตอบรับรหัสการเรียกที่ผู้ใช้ล็อกอินเข้ามาแสดงผลบนแผนที่แบบ Real-time

5.3 การพัฒนาในส่วนของรถฉุกเฉิน




โปรแกรมในส่วนของรถฉุกเฉินมีความสามารถ ดังนี้

1. สามารถติดต่อกับอุปกรณ์ GPS และนำตำแหน่งของรถมาแสดงบนแผนที่ได้
2. สามารถส่งข้อมูลตำแหน่งไปยังโปรแกรม เซิร์ฟเวอร์ ได้



รูปที่ 5-3 โปรแกรมที่อยู่บนรถฉุกเฉิน ที่ทำการตอบรับการเรียกขอความช่วยเหลือแล้ว

ตาราง ส่วนควบคุมการให้บริการ

ปุ่ม	ความหมาย
	ตอบรับการเรียกขอความช่วยเหลือ
	ขณะที่กำลังให้ความช่วยเหลือ
	เมื่อสิ้นสุดการให้ความช่วยเหลือ

ตารางที่ 5-1 แสดงส่วนควบคุมการให้บริการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง ส่วนควบคุมแผนที่

ปุ่ม	ความหมาย
	เลื่อนขึ้น
	เลื่อนลง
	เลื่อนซ้าย
	เลื่อนขวา
	ปรับแผนที่ให้เต็มหน้าจอ
	ขยายแผนที่
	ย่อแผนที่

ตารางที่ 5-2 แสดงส่วนควบคุมแผนที่

ตาราง ส่วนอื่นๆ

ปุ่ม	ความหมาย
	เริ่มส่งตำแหน่งรถฉุกเฉิน ไปเซิร์ฟเวอร์
	หยุดส่งตำแหน่งรถฉุกเฉิน ไปยังเซิร์ฟเวอร์
	กำหนดค่าเกี่ยวกับการติดต่อ GPS
	แสดงว่ายังไม่มี การเรียกขอความช่วยเหลือ
	แสดงว่ามีการเรียกขอความช่วยเหลือ

ตารางที่ 5-3 แสดงส่วนควบคุมส่วนอื่นๆ

5.3.1 การสื่อสาร และการเชื่อมต่อระหว่างเซิร์ฟเวอร์และโปรแกรมที่รถฉุกเฉิน

รูปแบบการติดต่อสื่อสาร ใช้การส่งผ่าน HTTP โพรโทคอลเพื่อให้ง่ายต่อการพัฒนา โปรแกรมบนโทรศัพท์เคลื่อนที่ซึ่งมีข้อจำกัดมากกว่าบนพีซี จึงใช้รูปแบบการส่งข้อมูลเป็น Query string ซึ่งมีรูปแบบดังนี้ name1=value1&name2=value2&...

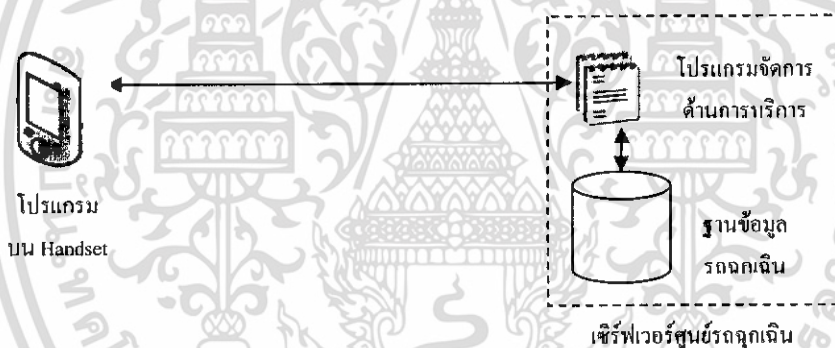
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง แสดง โปรโตคอลสำหรับการสื่อสารระหว่างเซิร์ฟเวอร์และโปรแกรมบนรถฉุกเฉิน

ฟังก์ชัน	ข้อมูล	ผลลัพธ์	กระบวนการ
onaccept	car_id	Result	สำหรับการตอบรับการเรียก
onprocess	car_id	Result	สำหรับบอกว่ากำลังให้ความช่วยเหลือ
onready	car_id + serv_id	Result	สำหรับบอกว่าเสร็จสิ้นการให้ความช่วยเหลือ
online	car_id + x_pos + y_pos	serv_id	อัปเดตตำแหน่งของรถฉุกเฉินพร้อมทั้งตรวจสอบว่ามีการเรียกขอความช่วยเหลือหรือไม่

ตารางที่ 5-4 แสดงการติดต่อระหว่างเซิร์ฟเวอร์และ โปรแกรมบนรถฉุกเฉิน

5.4 การออกแบบและพัฒนาโปรแกรมเรียกใช้บริการรถฉุกเฉินบน Handset



รูปที่ 5-4 แสดงโครงสร้างในส่วนของโทรศัพท์มือถือ

จากรูปข้างต้น จะเห็นได้ว่าการพัฒนาโปรแกรมเรียกใช้บริการรถฉุกเฉินบน Handset ให้ทำงานได้นั้น จะประกอบไปด้วยการสร้างโปรแกรมทั้งฝั่ง Handset (เป็นไคลเอนต์) และโปรแกรมทางฝั่งเซิร์ฟเวอร์ ซึ่งทั้งหมดนี้จะประกอบไปด้วยโปรแกรมที่เป็นส่วนสำคัญ นั่นคือ โปรแกรมที่อยู่บน Handset และ โปรแกรมจัดการด้านการบริการ

โปรแกรมบน Handset

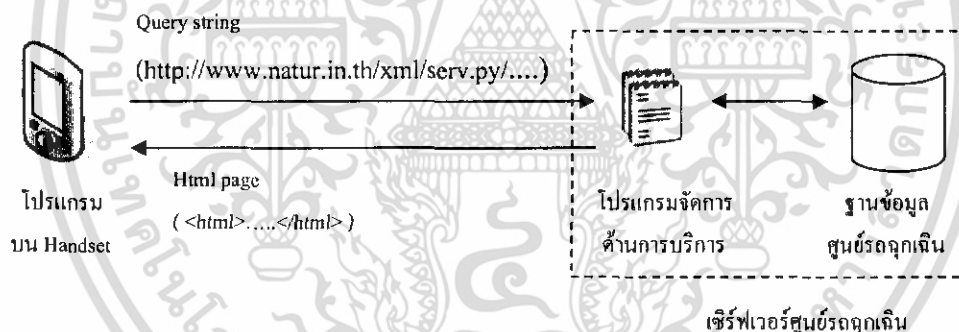
การทำงานในส่วนนี้ จะเป็นส่วนหลักในการทำหน้าที่ติดต่อกับผู้ใช้ อีกทั้งยังเป็นส่วนสำคัญในการเชื่อมต่อกับสัญญาณ GPS และสถานีฐาน เพื่อหาตำแหน่งปัจจุบันของผู้ใช้ การทำงานของโปรแกรมบน Handset นี้ จำเป็นต้องมีโปรแกรมจัดการทางด้านฝั่งเซิร์ฟเวอร์ ทำงานควบคู่ไปด้วยกันกล่าวคือ โปรแกรมบน Handset นั้นไม่มีความสามารถ ในการเข้าถึงเซิร์ฟเวอร์ศูนย์โดยตรง จำเป็นต้องผ่านโปรแกรมจัดการเสมอเปรียบเสมือนกับว่า โปรแกรมบน Handset เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมที่ใช้สำหรับ “ดู” (Viewer) ข้อมูลต่างๆที่ถูกประมวลผลเรียบร้อยแล้ว โดยการประมวลผลจากโปรแกรมจัดการฝั่งเซิร์ฟเวอร์ ซึ่งการออกแบบให้มีการทำงานเช่นนี้นั้น เพื่อประโยชน์หลายๆด้าน เช่น

- ลดภาระการทำงานของ Handset ที่มีข้อจำกัดด้านต่างๆสูง
- สามารถปรับเปลี่ยน, เปลี่ยนแปลงการแสดงผลข้อมูลต่างๆบน Handset ได้ง่าย เพียงเปลี่ยนการทำงานที่ส่วนโปรแกรมจัดการฝั่งเซิร์ฟเวอร์
- ลดจำนวนครั้ง ที่ต้องติดต่อผ่านเครือข่าย โดยติดต่อ ส่งคำสั่งไปยังโปรแกรมจัดการฝั่งเซิร์ฟเวอร์แค่ครั้งเดียว
- สามารถเปลี่ยนจากตัว Handset เป็น โปรแกรมอื่นๆได้โดยง่าย โดยไม่กระทบกับการทำงานของส่วนอื่นๆ

โดยการติดต่อระหว่างตัว Handset กับโปรแกรมจัดการฝั่งเซิร์ฟเวอร์นั้น จะเป็นการเชื่อมต่อกับโปรโตคอล HTTP, มีการส่งคำสั่งผ่านทาง Query String และมีการใช้ภาษา HTML ในบางส่วน จึงเป็นเรื่องที่ง่าย หากต้องการเปลี่ยนจากโปรแกรมบน Handset ไปเป็นโปรแกรมบนเว็บไซต์ หรือ บนอุปกรณ์อื่นๆ ที่มีความสามารถในการท่องเว็บในอินเทอร์เน็ต



รูปที่ 5-5 แสดงลักษณะของข้อมูลที่ทำการรับ-ส่งระหว่าง Handset กับ เซิร์ฟเวอร์

โดยตัวอย่างของ Query String ที่ถูกส่งโดยตัว Handset มายังโปรแกรมจัดการด้านการบริการนั้น มีลักษณะดังต่อไปนี้

http://www.natur.in.th/xml/serv.py/requestfrommobile?type=00000001&x_pos=2559223&y_pos=187859

URL ที่เรียกใช้งาน เรียกจาก handset ชนิดรถฉุกเฉิน ตำแหน่งของผู้ใช้พิกัด X และ Y

รูปที่ 5-6 แสดงตัวอย่าง Query string

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความหมายของแต่ละส่วนของ Query String ที่ถูกส่งไปโดยตัว Handset

URL: คือตำแหน่งที่อยู่ของตัวโปรแกรมจัดการด้านการบริการ จากในตัวอย่างคือ

`http://www.natur.in.th/xml/serv.py/requestfrommobile?`

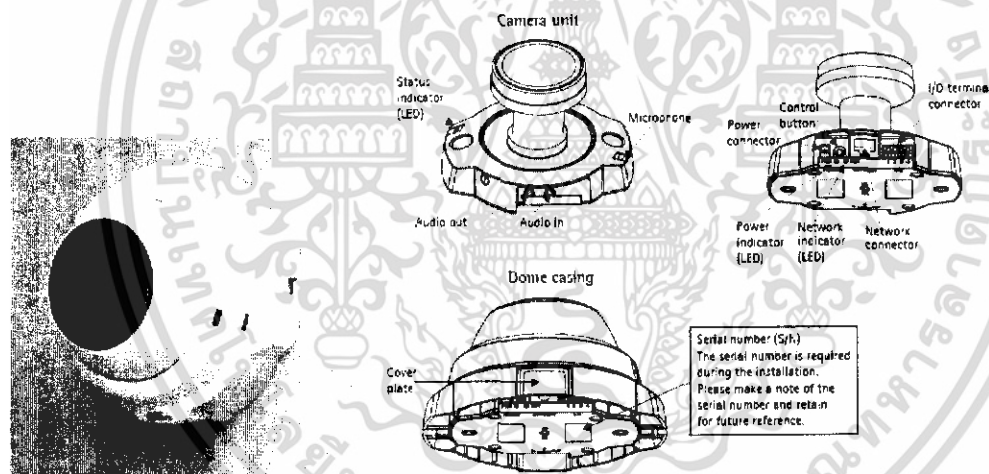
Requestfrommobile : เป็นตัวฟังก์ชันการทำงานของ Handset บนฝั่ง server

type : เป็นการระบุว่าจากการเรียกใช้บริการรถฉุกเฉินนั้น ผู้เรียกนั้นเรียกรถฉุกเฉินชนิดใด ซึ่งคำสั่งจะมีทั้งหมดดังนี้

- type=00000001 : จะเป็นการเรียกใช้บริการรถฉุกเฉินชนิดที่เป็นรถพยาบาล
- type=00000002 : จะเป็นการเรียกใช้บริการรถฉุกเฉินชนิดที่เป็นรถตำรวจ
- type=00000003 : จะเป็นการเรียกใช้บริการรถฉุกเฉินชนิดที่เป็นรถดับเพลิง

x_pos และ y_pos : เป็นการระบุตำแหน่งในขณะนั้นของผู้เรียกใช้บริการว่า อยู่ที่ตำแหน่งใด แล้วส่งตำแหน่งนั้นไปด้วย

5.5 ส่วนของกล้อง IP



รูปที่ 5-7 แสดงรูปโครงสร้างของกล้อง

ส่วนการทำงานของกล้องกล้องนี้จะถูกติดตั้งไว้กับรถทุกคันเพื่อญาติหรือเพื่อนสามารถใช้รหัสการเรียกขอความช่วยเหลือฉุกเฉินเข้าไปติดตามสามารถดูภาพจากกล้องที่ติดตั้งบนรถได้ โดยเฉพาะรถพยาบาล

ในงานเราใช้ AXIS 212 PTZ Network Camera ซึ่งเป็นกล้องเน็ตเวิร์ค กล้องนี้จะทำตัวเป็นเว็บเซิร์ฟเวอร์คอยให้บริการทั้งภาพและเสียง โดยการติดต่อและการเรียกใช้บริการทำผ่าน HTTP Protocol และมีการเตรียมฟังก์ชันต่างๆ ไว้เช่น ย่อภาพ ขยายภาพ เลื่อนซ้าย เลื่อนขวา เลื่อนขึ้น และเลื่อนลง เป็นต้น โดยฟังก์ชันดังกล่าวเป็น CGI Program ซึ่งสามารถเรียกใช้ผ่าน HTTP Protocol

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างฟังก์ชันบางฟังก์ชันที่เราเลือกใช้มีดังนี้คือ

- เลื่อนขึ้น

`http://<ServerName>/axis-cgi/com/ptz.cgi?camera=1&move=up`

- เลื่อนลง

`http://<ServerName>/axis-cgi/com/ptz.cgi?camera=1&move=down`

- เลื่อนซ้าย

`http://<ServerName>/axis-cgi/com/ptz.cgi?camera=1&move=left`

- เลื่อนขวา

`http://<ServerName>/axis-cgi/com/ptz.cgi?camera=1&move=right`

- ย่อภาพ

`http://<ServerName>/axis-cgi/com/ptz.cgi?camera=1&rzoom=-1000`

- ขยายภาพ

`http://<ServerName>/axis-cgi/com/ptz.cgi?camera=1&rzoom=1000`

- มุมกล้องภาพรวม

`http://<ServerName>/axis-cgi/com/ptz.cgi?camera=1&zoom=0`

- ขอบภาพวิดีโอ








`http://<ServerName>/axis-cgi/mjpg/video.cgi?camera=1&resolution=1280x1024
&compression=30&rotation=0&color=1&squarepixel=0&overlayimage=0&overlaypos=
0x0&date=1&clock=1&textstring=kmitl&textpos=top&fps=24`

ส่วนที่ไปเอาฟังก์ชันบางฟังก์ชันมาสร้างเป็นหน้าจอที่ใช้ในนี้ เป็นดังรูปที่ 5-7



รูปที่ 5-8 แสดงหน้าจอที่แสดงภาพที่ได้มาจากกล้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปุ่ม	การทำงาน	Query String
	เลื่อนขึ้น	http://<ServerName>/axis-cgi/com/ptz.cgi?camera=1&move=up
	เลื่อนลง	http://<ServerName>/axis-cgi/com/ptz.cgi?camera=1&move=down
	เลื่อนซ้าย	http://<ServerName>/axis-cgi/com/ptz.cgi?camera=1&move=left
	เลื่อนขวา	http://<ServerName>/axis-cgi/com/ptz.cgi?camera=1&move=right
	ขยายภาพ	http://<ServerName>/axis-cgi/com/ptz.cgi?camera=1&rzoom=1000
	ย่อภาพ	http://<ServerName>/axis-cgi/com/ptz.cgi?camera=1&rzoom=-1000
	ดูภาพรวม	http://<ServerName>/axis-cgi/com/ptz.cgi?camera=1&zoom=0

ตารางที่ 5-5 แสดง Query String ที่ส่งไปเมื่อกดปุ่ม



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

ผลการทดลอง

สำหรับในการทดลองนี้แบ่งการทดลองออกเป็น 3 ส่วนสำคัญๆ คือ ส่วนของการทำงานหลักๆ ของ Web , Program ที่ทำงานอยู่บนรถฉุกเฉิน และอีกส่วนคือของการทำงานของโทรศัพท์เคลื่อนที่ ในส่วนที่ 1 คือผลการทดลองการทำงานหลักๆ ของ Web Application นั้นจะทำการทดลองการใช้งาน Web Application โดยการเรียกหน้าเว็บผ่านทาง URL :: <http://www.natur.in.th> โดยผู้ที่เข้ามาใช้งานจะต้องเปิดดูผ่านเว็บเบราว์เซอร์ Internet Explorer เมื่อเข้าสู่หน้าแรกจะเป็นหน้าที่แสดงการใช้งานบริการต่างๆ ของแผนที่ เช่น การทดลองย่อ-ขยายเลื่อนแผนที่ไปทางซ้าย-ขวา การเลือกชั้นข้อมูลของแผนที่ที่จะแสดงโดยจะใช้เทคโนโลยีของ Ajax เข้ามาช่วยในการดึงข้อมูลจากฐานข้อมูล ซึ่งในขณะเดียวกันหน้าเว็บเพจจะไม่มีการรีเฟรช การค้นหาสถานที่ต่างๆ การแสดงตำแหน่งที่ผู้ต้องการความช่วยเหลือจะเรียก การเลือกชนิดของการร้องขอความช่วยเหลือ ซึ่งมีอยู่ 3 ชนิด คือ รถพยาบาล รถตำรวจ และรถดับเพลิง และการแสดงภาพผู้ป่วยหรือผู้ประสบอุบัติเหตุในขณะนั้นผ่านกล้องเป็นแบบ Real-time ในส่วนที่ 2 คือผลการทดลองการทำงานของ Program ที่ทำงานอยู่บนรถฉุกเฉินที่จะแสดงการทดลองในการแสดงตำแหน่งของรถและรับผู้ป่วยหรือผู้ประสบอุบัติเหตุ และในส่วนที่ 3 คือการทำงานของโทรศัพท์เคลื่อนที่นั้นจะทำการทดลองสร้าง Application โดยใช้เทคโนโลยีของ BREW ซึ่งเป็นการทดลองใช้งาน Program Application เพื่อใช้ในการเรียกรถฉุกเฉิน ที่เกิดจากการทำงานในด้านต่างๆ เช่น ด้านเมนูคอนโทรล ด้านการติดต่อกับเว็บเซิร์ฟเวอร์ เป็นต้น และการทดลองนี้จะทำการทดลองรันอยู่บน Emulator ซึ่งเป็นการจำลองการทำงานของโทรศัพท์เคลื่อนที่บนเครื่องคอมพิวเตอร์ส่วนบุคคล และสุดท้ายคือการวิจารณ์ผลการทดลอง

6.1 ส่วนที่ 1 การทำงานหลักๆ ของ Web Application

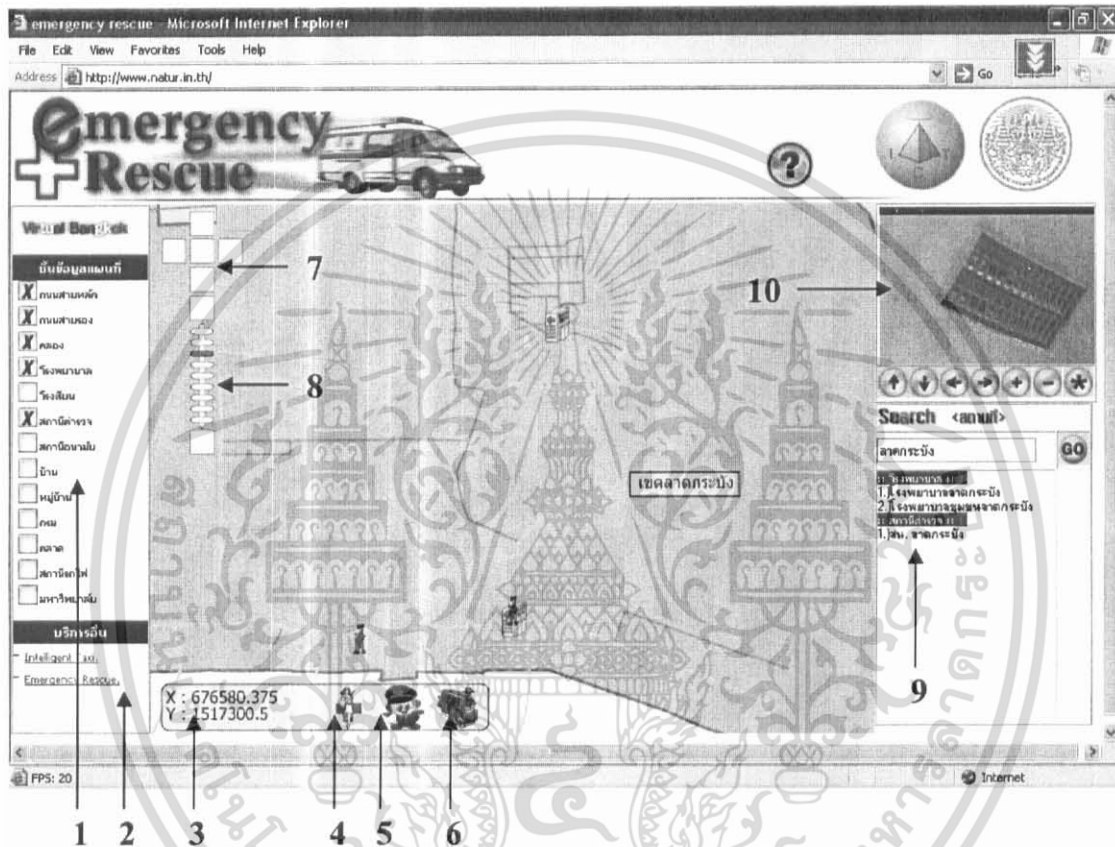
- แสดงหน้าเว็บเพจโดยรวม

โดยในส่วนของระบบ เรียกรถฉุกเฉินผ่าน Web site นี้จะมีส่วนประกอบหลักๆ อยู่ 9 ส่วน ซึ่งจะอธิบายในแต่ละส่วนได้ดังนี้

1. ส่วนแผนที่แสดงเลเยอร์ชั้นต่างๆ ที่ผู้ใช้ต้องการจะให้แสดงบนแผนที่
2. ส่วนของการบริการ อื่นๆ เช่น Intelligent Taxi
3. ส่วนการแสดงผลตำแหน่งของผู้เรียกที่ต้องการเรียกในขณะนั้น
4. เป็นการร้องขอการเรียกรถพยาบาล
5. เป็นการร้องขอการเรียกรถตำรวจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

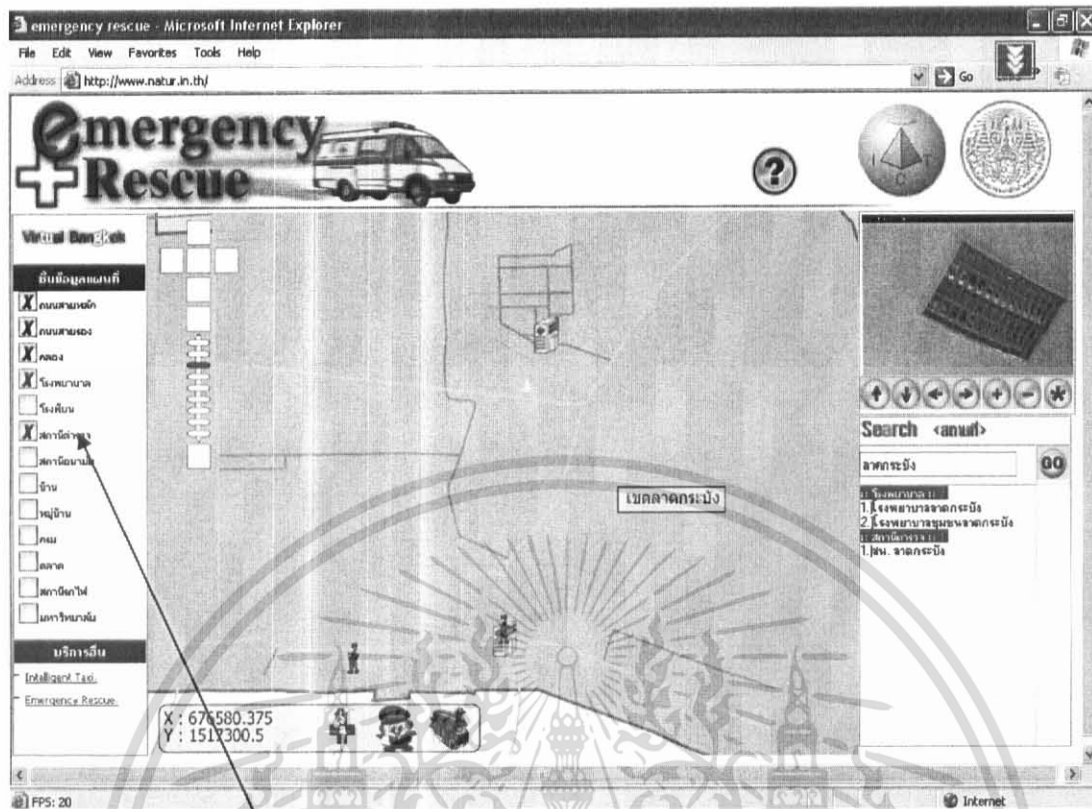
6. เป็นการร้องขอการเรียกรดับเพลิง
7. เป็นแผงควบคุมการเลื่อนแผนที่ ซึ่งสามารถเลื่อนไปทางซ้าย ขวา ขึ้นบน ลงล่าง
8. เป็นแผงแสดงการควบคุมการขยายและย่อขนาดของแผนที่
9. เป็นส่วนของการ ค้นหาสถานที่ต่างๆ ที่อยู่บนแผนที่
10. เป็นส่วนของการแสดงผลของกล้องที่เอาไว้แสดงผู้ประสบอุบัติเหตุในขณะที่อยู่บนรถ



รูปที่ 6-1 แสดงหน้าเว็บเพจของ Web Emergency Rescue

- การทดลองเทคโนโลยี Ajax มาใช้ในการดึงชั้นข้อมูลของแผนที่ โดยปกติแล้วเมื่อมีการ Update เกิดขึ้นบนเว็บเบราว์เซอร์นั้น เว็บเบราว์เซอร์ก็จะทำการติดต่อไปที่ Server ใหม่เพื่อทำการแสดงหน้าเว็บเพจขึ้นมาใหม่ แต่เมื่อนำเทคโนโลยี Ajax เข้ามาใช้แล้วทำให้เว็บเพจไม่ต้องมีการรีเฟรชหน้าเว็บเพจนั้นใหม่ ฉะนั้นเมื่อ client มี request แทนที่จะส่ง HTTP request ไปยัง server โดยตรง client จะส่ง JavaScript call ไปยัง Ajax engine เพื่อโหลดข้อมูลที่ user ต้องการ ผลการทดลองการใช้เทคโนโลยี Ajax ดังแสดงในรูปด้านล่างนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



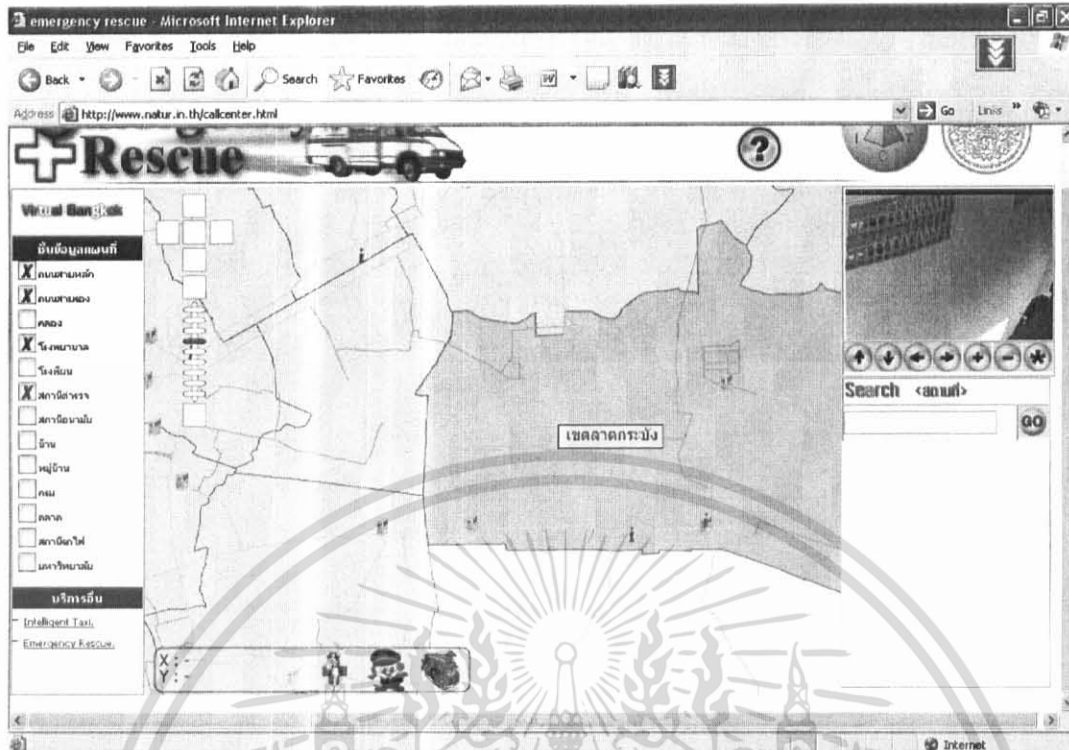
ส่วนแผนที่แสดงชั้นข้อมูลของแผนที่ที่ใช้ Ajax ในการดึงข้อมูล

รูปที่ 6-2 แสดงการนำเทคโนโลยี Ajax มาใช้กับเว็บ

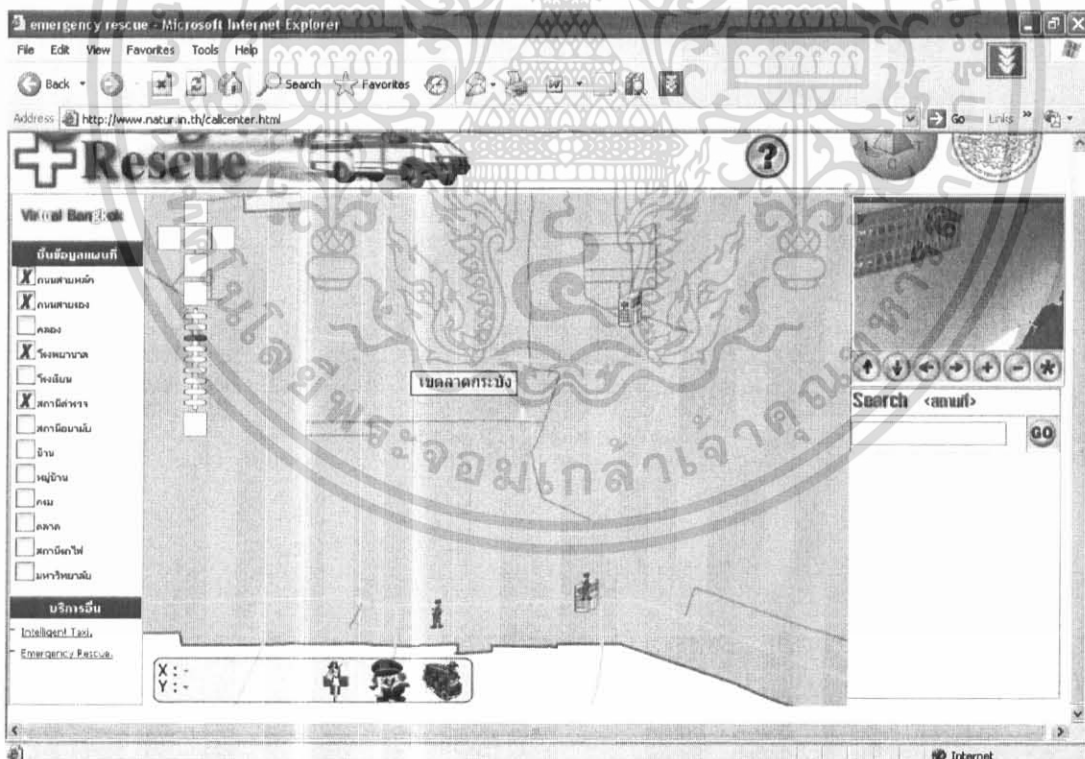
- การทดลองย่อ-ขยาย แผนที่โดยใช้ SVG

ในส่วนของแผนที่นี้เราสามารถที่จะย่อหรือขยายแผนที่ เพื่อดูรายละเอียดในจุดต่างๆได้ ซึ่งได้ใช้ SVG ในการแสดงแผนที่ ซึ่งข้อดีของการใช้ SVG คือในการย่อหรือขยายแผนที่จะทำให้การแสดงผลนั้นสวยงามและไม่เกิดการแตกของสีเนื่องจากการแสดงแบบ Graphic โดยการทดลอง ใช้ SVG ด้วยการกดปุ่ม Zoom in (+) เพื่อทำการขยายขนาดของแผนที่ และ Zoom Out (-) เพื่อทำการย่อขนาดของแผนที่บนแผนที่ ดังแสดงในรูปที่ 6-3, ในรูปที่ 6-4, รูปที่ 6-5 และ ในรูปที่ 6-6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

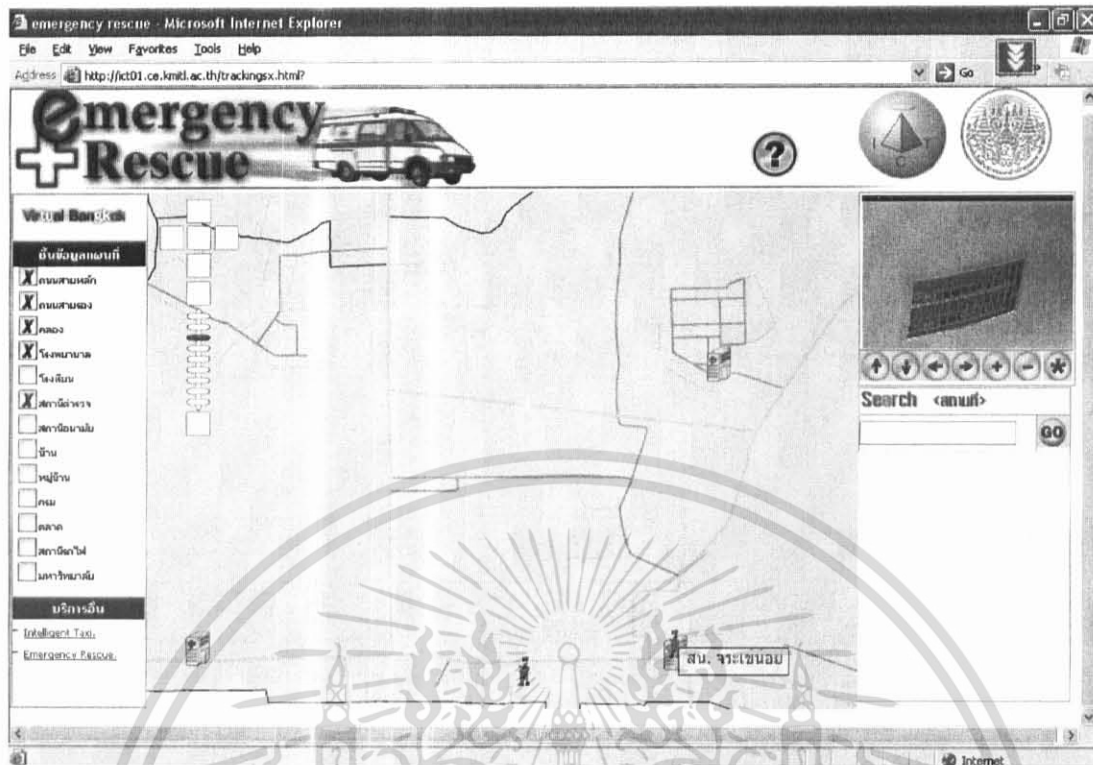


รูปที่ 6-3 แสดงแผนที่ก่อนการกดปุ่ม Zoom in (+) เพื่อทำการขยายขนาดของแผนที่

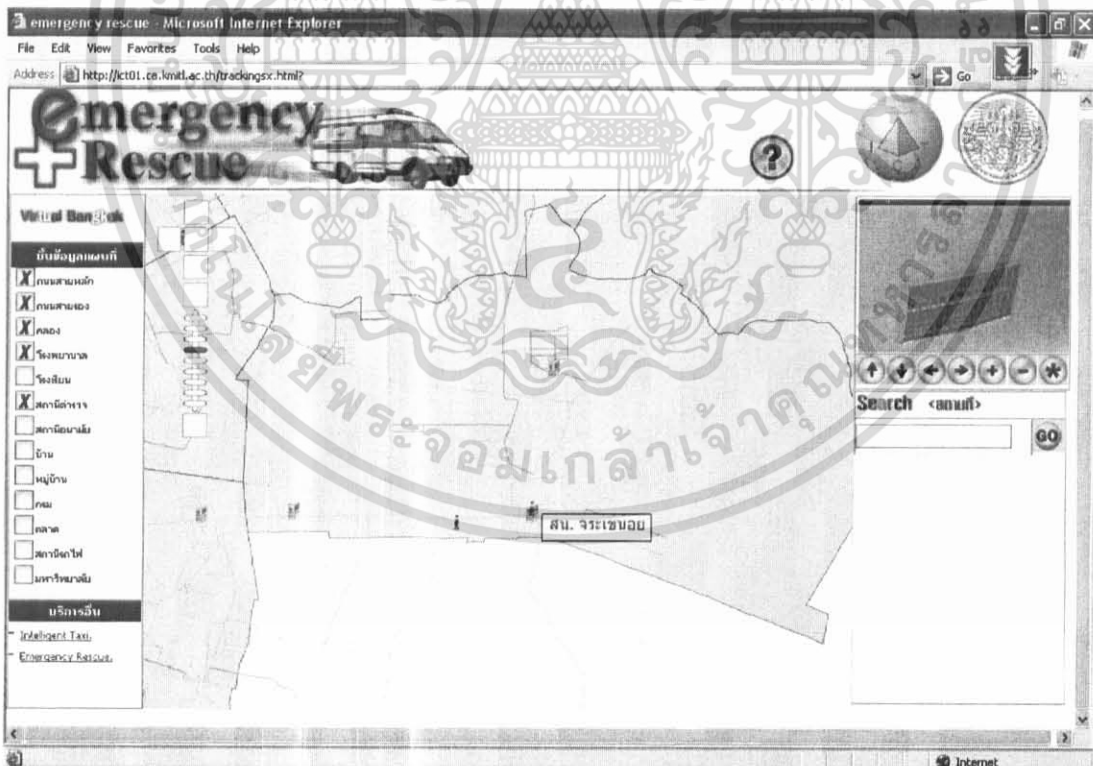


รูปที่ 6-4 แสดงแผนที่หลังการกดปุ่ม Zoom in (+) เพื่อทำการขยายขนาดของแผนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



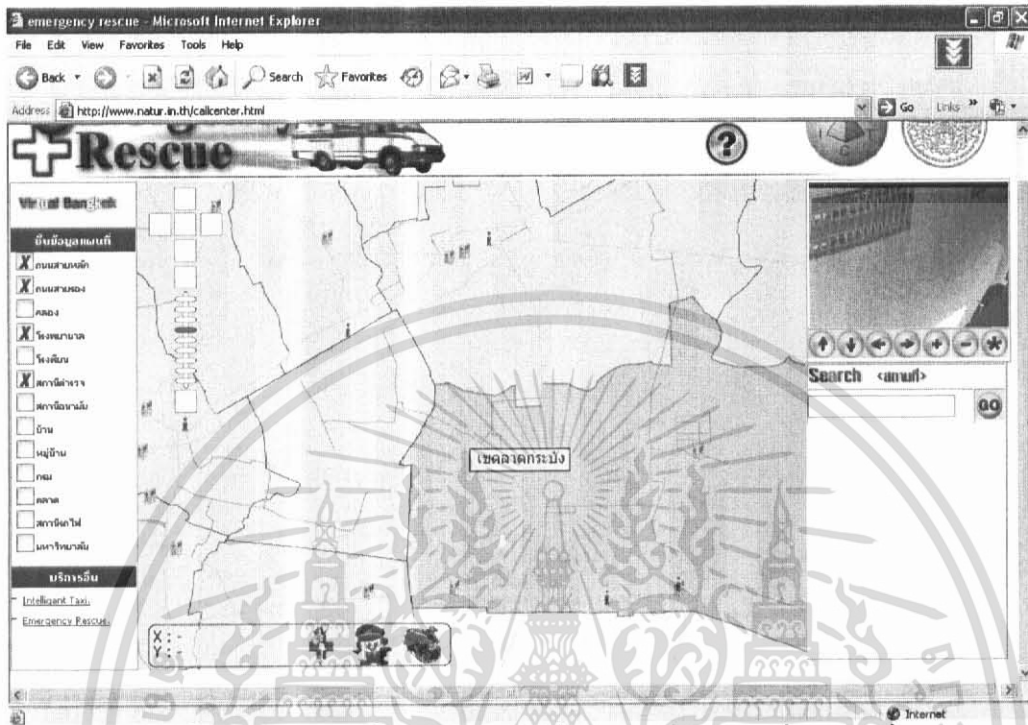
รูปที่ 6-5 แสดงแผนที่ก่อนการกดปุ่ม Zoom out (-) เพื่อทำการย่อขนาดของแผนที่



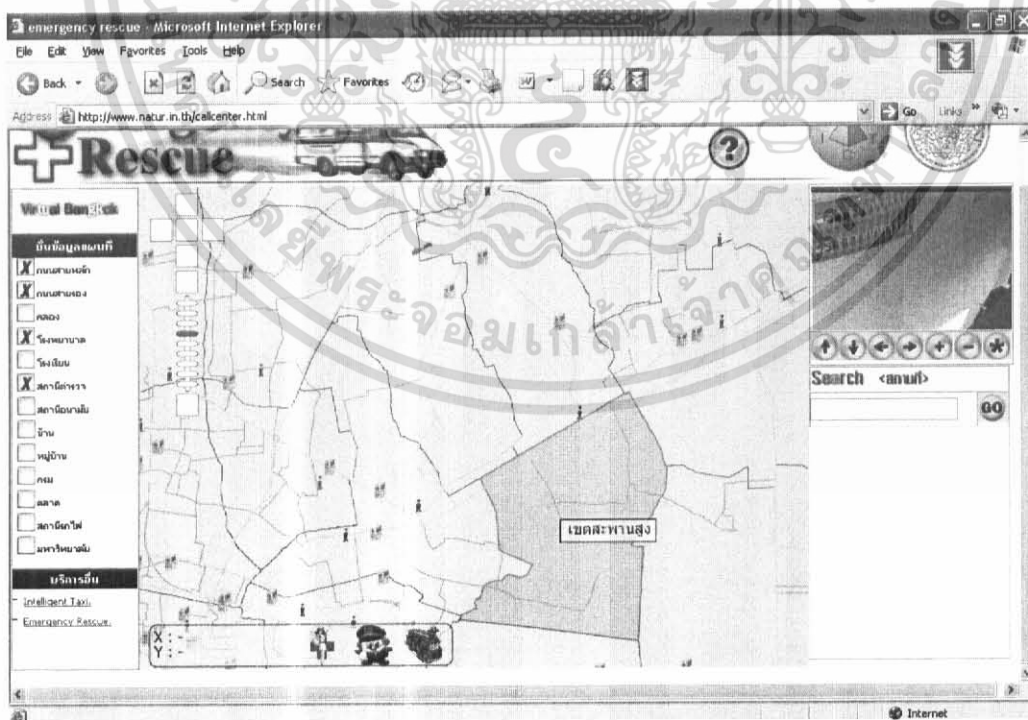
รูปที่ 6-6 แสดงแผนที่หลังการกดปุ่ม Zoom out (-) เพื่อทำการย่อขนาดของแผนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การเลื่อนแผนที่
การเลื่อนแผนที่นั้น โดยจะสามารถเลื่อนได้ทั้ง 4 ทิศทาง คือ เลื่อนขึ้น เลื่อนลง เลื่อนไปทางซ้าย และ เลื่อนไปทางขวา ดังแสดงในรูปที่ 6-5 และรูปที่ 6-6



รูปที่ 6-7 แสดงแผนที่ก่อนการกดปุ่มเลื่อนซ้าย เพื่อทำการเลื่อนแผนที่



รูปที่ 6-8 แสดงแผนที่หลังการกดปุ่มเลื่อนซ้าย เพื่อทำการเลื่อนแผนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การทดลองค้นหาสถานที่บนแผนที่



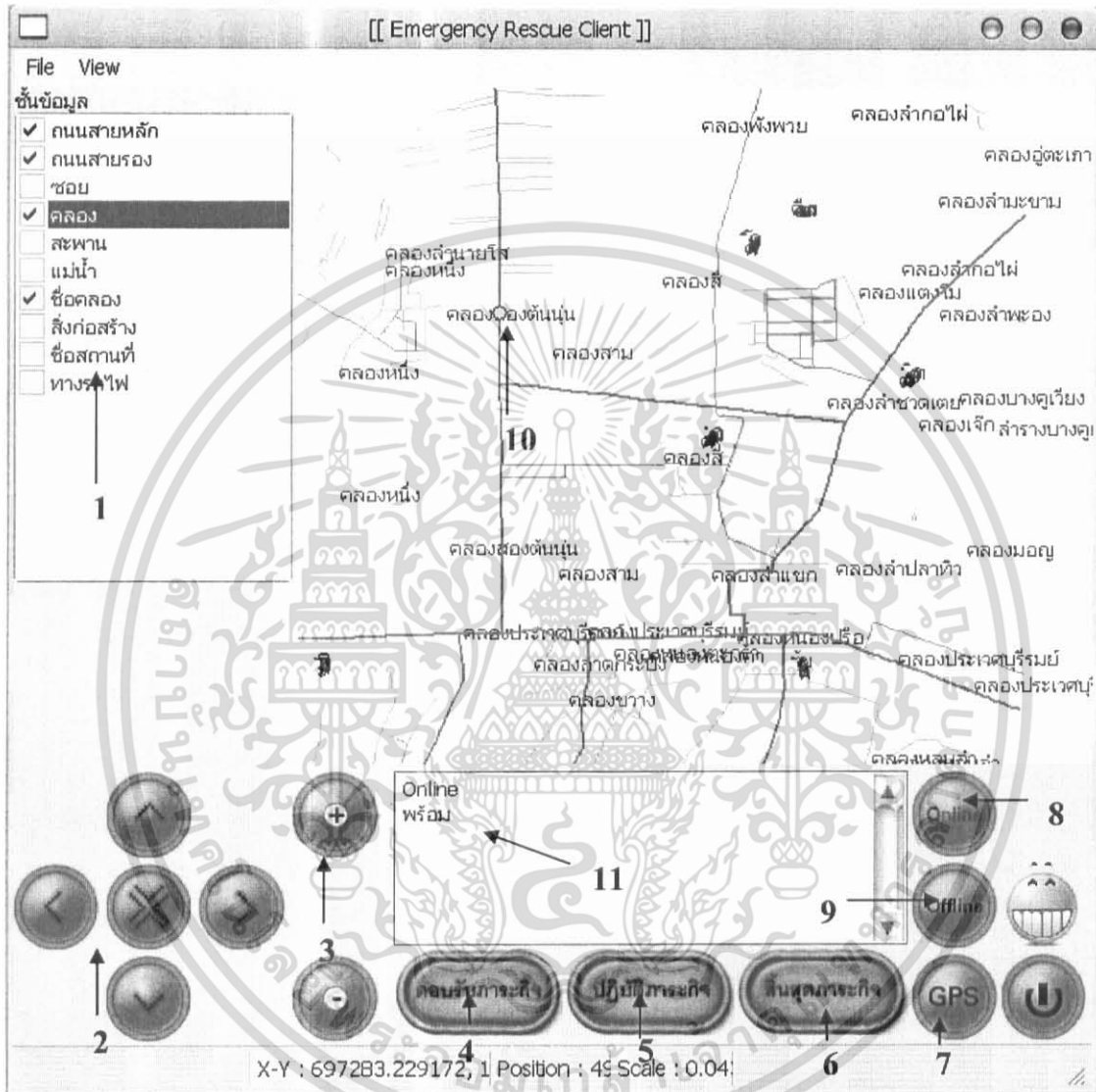
รูปที่ 6-9 แสดงการทดลองค้นหาสถานที่บนแผนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2 ส่วนที่ 2 การทำงานของ Program ที่อยู่บนรถฉุกเฉิน

6.2.1 Program ที่อยู่บนรถฉุกเฉิน

โดยในส่วนของระบบที่เป็น Application Program ที่ต้องติดตั้งและมืออยู่บนรถฉุกเฉินนี้จะมีส่วนประกอบหลักๆ อยู่ 11 ส่วนซึ่งจะอธิบายในแต่ละส่วนได้ดังนี้



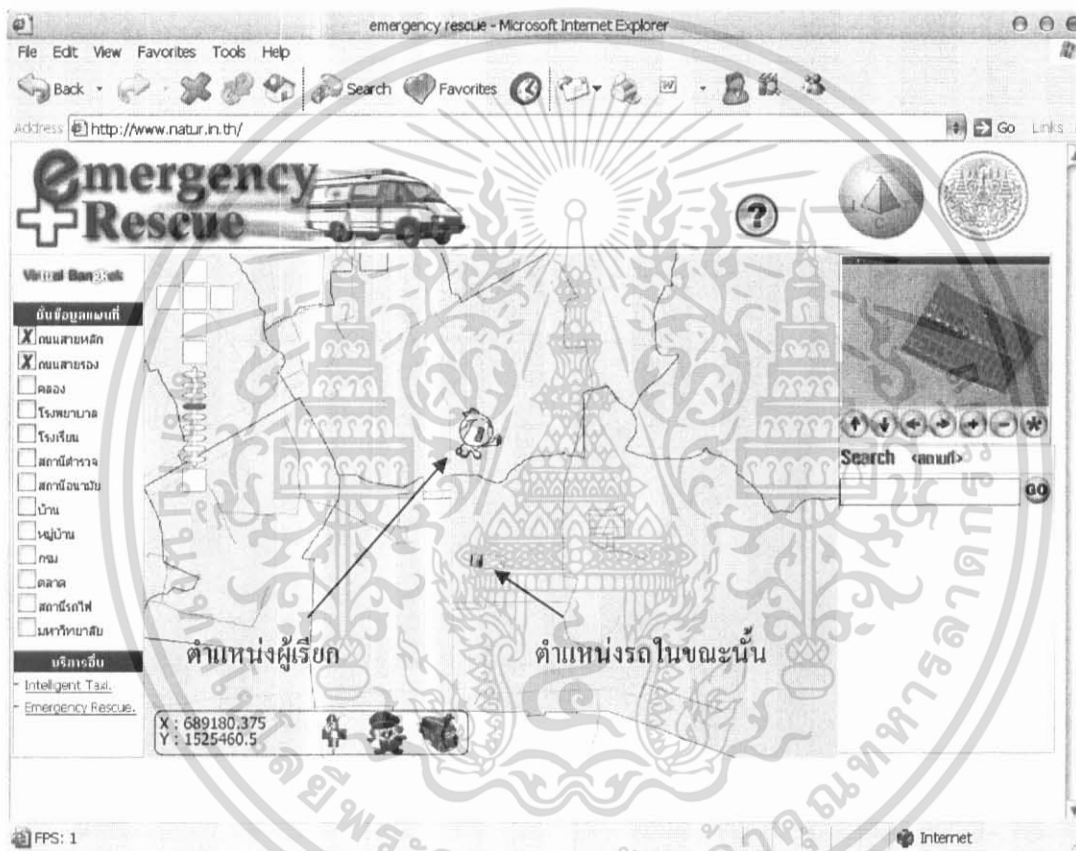
รูปที่ 6-10 แสดงโปรแกรมที่อยู่ในส่วนของรถฉุกเฉิน

คำอธิบายในแต่ละส่วน

1. ส่วนแผนที่แสดงเลเยอร์ชั้นต่างๆ ที่ผู้ใช้งานต้องการจะให้เห็นแสดงบนแผนที่
2. เป็นแผงควบคุมการเลื่อนแผนที่ ซึ่งสามารถเลื่อนไปทางซ้าย ขวา ขึ้นบน ลงล่าง
3. เป็นแผงแสดงการควบคุมการขยายและย่อขนาดของแผนที่
4. เป็นปุ่มการตอบรับการบริการ
5. เป็นปุ่มที่ต้องกดเมื่อเริ่มการรับผู้เรียกใช้บริการ
6. เป็นปุ่มที่กดเมื่อบริการผู้ใช้เสร็จสิ้น

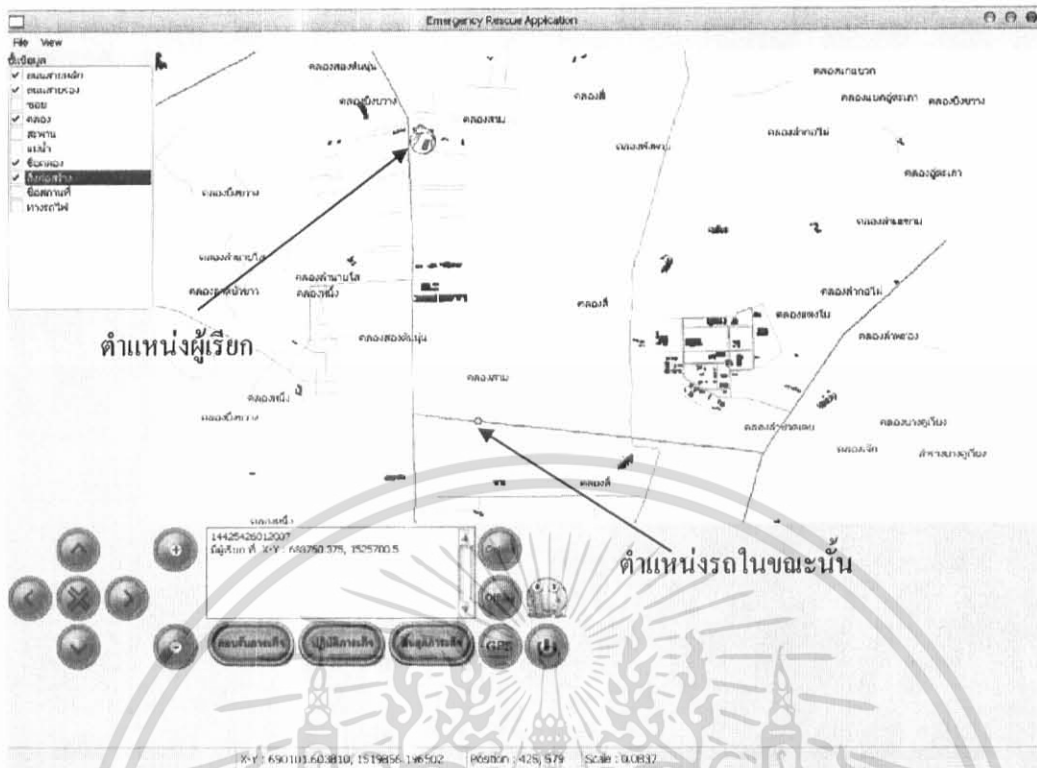
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. เป็นตัวที่ setup GPS ที่ติดตั้งอยู่บนรถฉุกเฉิน
 8. เป็นปุ่มที่ใช้ในการเข้าสู่ระบบเมื่อรถฉุกเฉินคันนั้นพร้อมให้บริการ
 9. เป็นปุ่มที่ใช้ในการออกจากระบบเมื่อรถฉุกเฉินคันนั้นเลิกให้บริการ
 10. แสดงตำแหน่งรถในขณะนั้น
 11. แสดงตำแหน่งพิกัด X Y
- การแสดงตำแหน่งของรถคันนั้นและแสดงตำแหน่งผู้เรียกเมื่อมีผู้เรียกใช้บริการฉุกเฉินเข้ามา



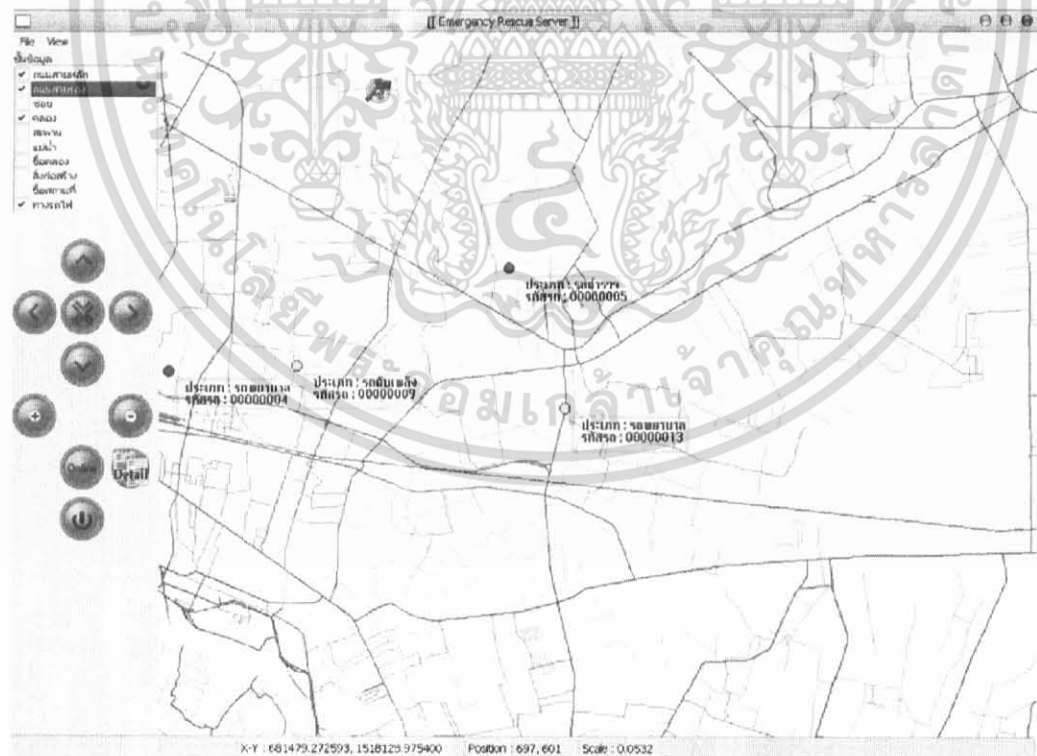
รูปที่ 6-11 แสดงผลบนเว็บเพจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6-12 แสดงอยู่บน Program ที่อยู่บนรถฉุกเฉิน

6.2.2 Program ที่ใช้ติดตามรถทุกคัน



รูปที่ 6-13 แสดงอยู่บน Program ที่ใช้ติดตามรถฉุกเฉินทุกคัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3 ส่วนที่ 3 การทำงานของโทรศัพท์เคลื่อนที่

ผลการทดลองเขียนพัฒนา Application Emergency Rescue บนโทรศัพท์เคลื่อนที่ เป็นผลการทดลองแสดงการใช้งาน Application บางส่วนของโครงการนี้ การทดลองในส่วนนี้นั้นจะแสดงเพียง รูปภาพการเรียกรถพยาบาลฉุกเฉินเท่านั้น เนื่องจากครั้งก่อนได้แสดงผลการทดลองเกี่ยวกับระบบที่ใช้เรียกรถฉุกเฉินนี้ทางโทรศัพท์มือถือทั้งหมดไปแล้ว โดยถ้าเป็นการเรียกรถตำรวจ หรือ เรียกรถดับเพลิง ก็จะมีลักษณะการเรียกรูปแบบเดียวกัน ต่อไปจะเป็นการแสดงผลการทดลองพร้อมกับ ขั้นตอนการเรียกรถพยาบาลฉุกเฉิน

- 1.1. แสดงไอคอน Program Application on Mobile ทั้งหมดที่มีอยู่ในโทรศัพท์มือถือในขณะนั้นซึ่ง ก็จะมี Application ที่ชื่อว่า Emergency Rescue ที่เป็น Application ที่พัฒนาโดยโครงการนี้



รูปที่ 6-14 แสดงไอคอน Program Application on Mobile ของ Emergency Rescue

รูปนี้แสดงผลการทดลองในส่วนของ Program Application on Mobile ซึ่งการแสดงผลของรูปนี้เป็นการแสดงรูปไอคอนของ Application Program ต่างๆที่มีอยู่ภายในโทรศัพท์มือถือ โดยในรูปตัวอย่างนี้จะมีอยู่เพียง 3 Application Program ที่แสดงอยู่บนโทรศัพท์มือถือ และหนึ่งใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นั่นก็มี Application Program ที่ชื่อว่า “Emergency Rescue” ซึ่งเป็น Application Program ของโครงการนี้ จากนั้นก็ทำการ Select เพื่อเลือกเข้าไปใน Application นี้โดยการกดปุ่ม ตามที่ลูกศรชี้

1.2. แสดงหน้าจอหน้าแรกของการทำงานเมนูหลัก ของระบบการเรียกรถฉุกเฉิน



รูปที่ 6-15 แสดงหน้าจอหน้าแรกของการทำงานเมนูหลัก

รูปนี้คือการแสดงผลต่อจากที่เมื่อเราได้กดปุ่ม Select เพื่อเลือกเข้ามาที่ Application Program ของโครงการนี้ที่ชื่อว่า “Emergency Rescue” แล้ว ซึ่งหน้าจอของโทรศัพท์มือถือจะทำการแสดงหน้าเมนูหลักของ Application นี้ โดยจะมีอยู่ 3 เมนู ดังนี้

1. เมนูเรียกรถพยาบาล คือ เมนูที่ใช้ในการเรียกรถพยาบาลฉุกเฉิน
2. เมนูเรียกรถตำรวจ คือ เมนูที่ใช้ในการเรียกรถตำรวจฉุกเฉิน
3. เมนูเรียกรถดับเพลิง คือ เมนูที่ใช้ในการเรียกรถดับเพลิงฉุกเฉิน

จากนั้นให้ทำการกดปุ่มเลื่อนไปที่เมนู เรียกรถพยาบาล และกดปุ่ม Select เพื่อทำการเลือกและเข้าไปในเมนูของการเรียกรถพยาบาล ดังแสดงต่อด้านล่างนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3. แสดงหน้าเมนูของการเรียกรถพยาบาลแบบฉุกเฉิน



รูปที่ 6-16 แสดงหน้าเมนูของการเรียกรถพยาบาลแบบฉุกเฉิน

เป็นเมนูที่อยู่ภายใต้เมนูเรียกรถพยาบาล ซึ่งเมื่อเลือกเมนูเรียกรถพยาบาลในหน้าจอเมนูหลักของ Application Program แล้ว และภายใต้เมนูนี้จะมีอยู่ 2 เมนู ดังนี้

1. เมนูเรียก (ด่วน!) คือ เมนูที่ใช้ในการเรียกรถพยาบาลฉุกเฉิน
2. เมื่อกดยกเลิกการเรียก คือ เมนูที่ใช้ในการยกเลิกการเรียกรถพยาบาลฉุกเฉิน ซึ่งภายในเมื่อกดยกเลิกการเรียกนี้จะมีเมนูย่อยขึ้นมาแสดงอีกที เพื่อถามความแน่ใจของผู้ใช้ว่าต้องการจะยกเลิกการเรียกรถพยาบาลฉุกเฉินจริงๆหรือไม่ ซึ่งจะแสดงในรูปต่อไป

จากนั้นให้เลือกไปที่ เมนูเรียกด่วน เมื่อเลือกไปที่เมนูเรียกด่วนแล้ว Program ก็จะทำการแสดงสถานะ การทำงานในขณะนั้นว่า กำลังติดต่อ และเมื่อติดต่อสำเร็จแล้วก็จะแสดง ผลการติดต่อนั้นๆ ว่าเรียกรถพยาบาลแล้ว และก็ให้กลับไปยังหน้าเมนูหลัก เท่านั้นที่เสร็จสิ้นขั้นตอนการเรียกรถพยาบาลฉุกเฉินดัง รูปด้านล่างนี้



รูปที่ 6-17 แสดงหน้าเมนูของผลการเรียกรถพยาบาลแบบฉุกเฉินว่าสำเร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7

บทวิจารณ์และสรุป

7.1 บทสรุป

ในปัจจุบันเทคโนโลยีทันสมัยได้เข้าไปอยู่ในชีวิตของคนมากขึ้น ระบบแผนที่ที่สามารถเรียกใช้งานได้ทุกที่ทุกเวลาผ่านทาง Browser ที่มีอยู่ในคอมพิวเตอร์ทุกเครื่องและ โทรศัพท์มือถือที่มีระบบ CDMA ที่อยู่ติดตัวตลอดเวลา ย่อมทำให้การใช้ชีวิตสะดวกยิ่งขึ้น โครงการนี้พัฒนาขึ้นเป็นส่วนที่ช่วยในการสร้างและให้บริการดึงข้อมูลแผนที่ ซึ่งสามารถทำงานได้ตามความต้องการ นั่นคือ สามารถดึงภาพแผนที่ ณ ตำแหน่ง Latitude, Longitude ใดๆ ได้, สามารถดึงตำแหน่งของสถานที่ต่างๆ ได้, สามารถคำนวณระยะทางได้ และสามารถหาพื้นที่ได้ ซึ่งเทคโนโลยีที่นำมาใช้ในการพัฒนาระบบที่เกี่ยวข้องกับโครงการนี้ซึ่งประกอบไปด้วย XML, GML, SVG, AJAX ทำให้การแสดงผลภาพบนเว็บเบราว์เซอร์นั้น มีประสิทธิภาพในการแสดงผลออกมาได้อย่างละเอียดและสวยงาม และยังมีความสามารถในการย่อขยายแผนที่โดยที่คุณภาพของการแสดงผลแผนที่นั้นยังคงเดิม นอกจากนี้ นั้นเอกสาร SVG ยังมีรูปแบบพื้นฐานเป็น XML ด้วยทำให้ช่วยในการจัดการกับแผนที่ได้ง่ายขึ้นอีกด้วย ส่วนเทคโนโลยีฐานข้อมูลเชิงพื้นที่ (Spatial Database) นั้นทำให้ระบบมีการจัดเก็บข้อมูลแผนที่ที่ถูกต้อง สามารถนำมาใช้งานและแสดงผลได้หลากหลายรูปแบบ และเทคโนโลยีของกล้อง IP ทำให้มีการแสดงภาพของผู้ป่วยในขณะนั้นไปให้หมอเพื่อทำการวินิจฉัยล่วงหน้า และ BREW Platform นั้นทำให้มีการพัฒนา Application บน โทรศัพท์มือถือได้เป็นอย่างดี ดังนั้นจึงเป็นวิธีการที่เหมาะสมที่จะนำไปใช้ในการพัฒนาระบบให้บริการเพื่อความช่วยเหลือประชาชนผู้เดือดร้อนต่อไปในอนาคต

7.2 ประโยชน์ที่ได้จากโครงการ

การทำโครงการชิ้นนี้ ทำให้เราได้รับความรู้และประสบการณ์มากมาย สิ่งเหล่านี้ประกอบด้วยการเขียนโปรแกรมด้วยภาษา C, การเขียนโปรแกรมด้วยภาษา Python, ได้รู้จักการเก็บข้อมูลในฐานข้อมูลเชิงพื้นที่ (Spatial Database), ความรู้ความเข้าใจเกี่ยวกับ XML และการพัฒนา และความรู้เกี่ยวกับการออกแบบและใช้งาน Database นอกจากนี้ความรู้ต่างๆ เหล่านี้แล้วเรายังได้รับประสบการณ์ต่างๆ เช่น การกำหนดระยะเวลาและวางแผนการทำงาน, การหาข้อมูลใน Internet และสิ่งที่สำคัญที่สุดในการทำโครงการชิ้นนี้ก็คือ การทำงานร่วมกัน อีกทั้งสภาวะแวดล้อมที่อยู่ภายในห้องวิจัยแทบจะตลอดเวลา สิ่งเหล่านี้ล้วนช่วยเสริมสร้างความแข็งแกร่งของเราเป็นอย่างมาก จึงนับได้ว่า ประโยชน์ที่ได้รับจากการทำโครงการชิ้นนี้มีคุณค่ามาก.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.3 ปัญหาที่พบในการพัฒนา

1. ข้อจำกัดของ Java Script ที่ไม่สามารถดักจับได้ จึงทำให้ยากต่อการพัฒนาโปรแกรม
2. ความสามารถทางด้านการจัดการข้อมูลแผนที่ของ Java Script ยังไม่ดีเท่าที่ควร
3. การ passing เอกสาร XML ของแต่ละ browser ไม่เหมือนกัน
4. ตำแหน่งแผนที่บนหน้าเว็บเพื่อมีความคลาดเคลื่อนบ้าง
5. ความแม่นยำของ GPS อาจมีความคลาดเคลื่อนของตำแหน่ง
6. เรื่อง Dynamic IP ที่ต้องนำมาใช้งานกับตัวกล้อง ซึ่งต้องใช้ IP ที่เป็น IP จริง
7. ในส่วนของ BREW Emulator นั้นบางรุ่นยังไม่สามารถรองรับ Application ที่พัฒนาขึ้นมาบาง Application ได้
8. Application ที่พัฒนาขึ้นมาบน Emulator เมื่อจะนำเอา Application ที่พัฒนานั้นลงสู่ Handset จริงๆแล้วต้องทำการ compile ด้วย ARM ทำให้เกิดข้อผิดพลาดบ้าง
9. เทคโนโลยี BREW ยังถือได้ว่าเป็นเทคโนโลยีใหม่สำหรับประเทศไทย จึงทำให้ข้อมูลที่จะใช้ประกอบในการพัฒนานั้นหาได้ยาก และมีข้อมูลในการอ้างอิงน้อย

7.4 แนวทางการแก้ไข

1. สร้างฟังก์ชันของ Java Script ขึ้นมาใช้งานเอง
2. มีการติดต่อขอแผนที่ความละเอียดสูง จากกรมผังเมือง สำนักงานกรุงเทพมหานคร
3. ต้องมีการเพิ่มค่าตำแหน่งของแผนที่ก่อนทำการแสดงผลบนหน้าเว็บเพจ
4. ใช้ Differential GPS คือการเอา GPS ติ่งที่มีความแม่นยำสูง มาระบุตำแหน่งตัวเอง แล้วให้ GPS อีกตัวปรับแก้ Error
5. เลือกรุ่นของ Emulator ที่รองรับ Application ได้มากที่สุดเท่าที่จะเป็นไปได้
6. ถามข้อสงสัยต่างๆจาก web board ของกลุ่มผู้ที่พัฒนา Application ด้วย BREW Platform

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

การติดตั้งโปรแกรมและการใช้โปรแกรม

ในคู่มือการติดตั้งโปรแกรมและการใช้โปรแกรมของระบบเรียกรถฉุกเฉินนี้ได้แบ่งออกเป็นส่วนหลักๆ ได้ 4 ส่วนดังนี้

1. ส่วนของการพัฒนาโปรแกรมบนโทรศัพท์มือถือ
2. ส่วนของการพัฒนาโปรแกรมบนรถฉุกเฉิน
3. ส่วนของการพัฒนาโปรแกรมที่ใช้ติดตามรถฉุกเฉินทุกคัน
4. ส่วนของการพัฒนาโปรแกรมบน Web Server

ส่วนของการพัฒนาโปรแกรมบนโทรศัพท์มือถือ

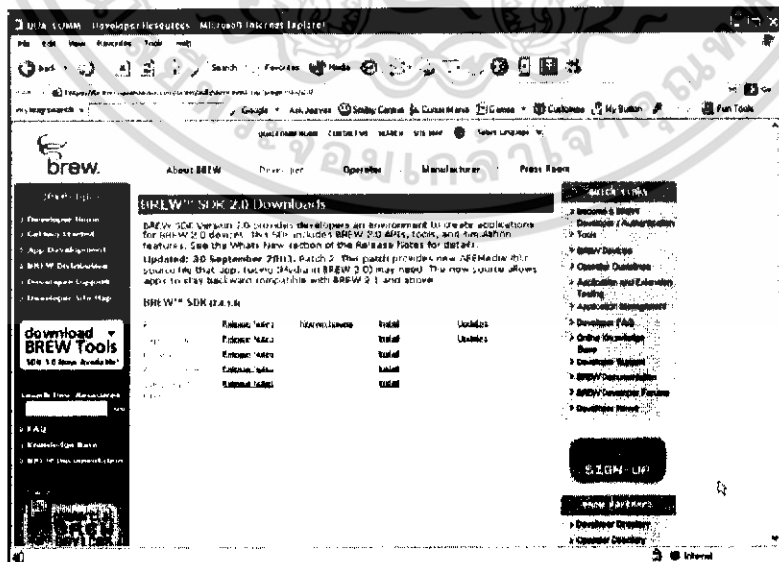
ขั้นตอนการติดตั้งโปรแกรมที่ใช้ในการพัฒนา Application บน โทรศัพท์มือถือ

- ติดตั้ง BREW SDK เพื่อใช้ BREW Emulator ในการทดสอบ content BREW System Requirements = Microsoft Windows XP/2000, Disk Space 50MB

หมายเหตุถ้าต้องการพัฒนา BREW Application ต้องติดตั้ง Microsoft Visual Studio 6.0 ก่อน ในเอกสารชุดนี้ไม่กล่าวถึง

ขั้นตอนโดยละเอียด

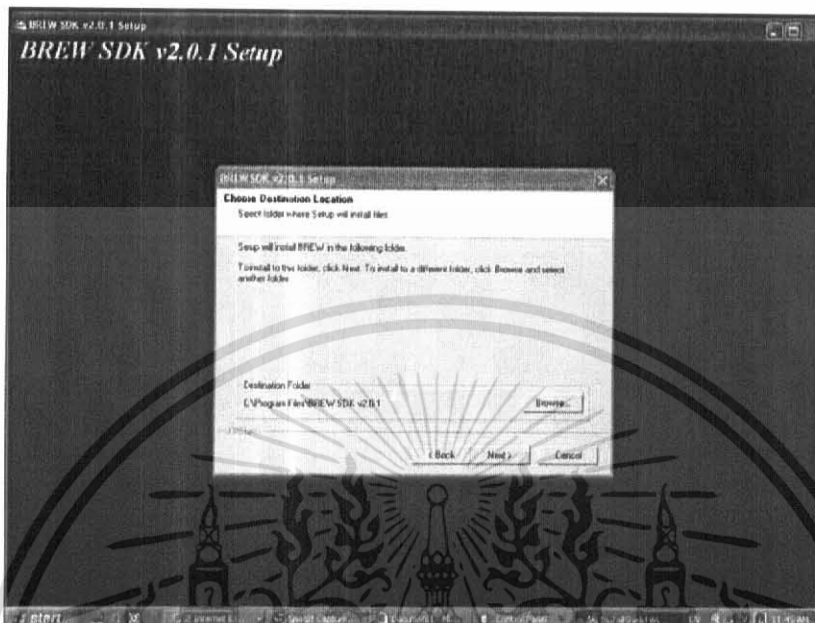
1. Download และติดตั้ง BREW SDK version 2.01 ซึ่งโปรแกรมจะมี BREW Emulator อยู่ในโฟลเดอร์ BREW SDK 2.01 downloads ได้จาก <https://brewx.qualcomm.com/brew/sdk/download.jsp>



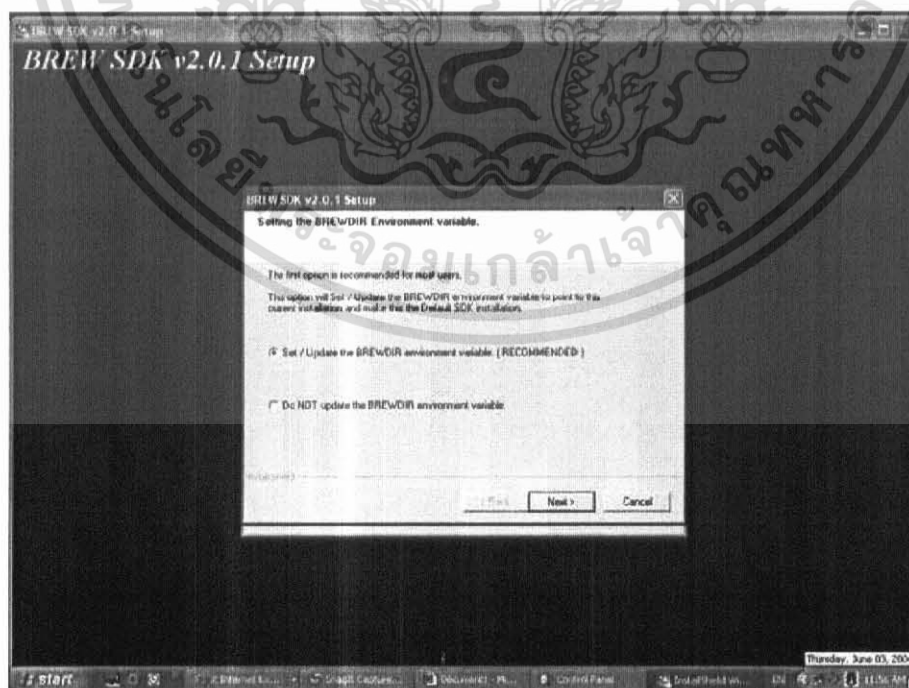
1.1 เลือก English version ตามด้วยการกด Install

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 เลือก Path ในการติดตั้งไปที่ C:\Program File\BREW SDK v2.0.1 แล้วกด next จากนั้น Confirm path ในการติดตั้งอีกครั้ง ด้วยการกด Yes แล้วก็กด next ระบุ Folder ของ Program (ใช้ Default) แล้ว กด Next



1.5 ระบบจะทำการติดตั้ง BREW SDK รอจน ระบบติดตั้งเสร็จ เมื่อระบบติดตั้งเสร็จให้ทำการเลือก Set/Update the BREWDIR environment variable (Recommended) แล้วกด Next เมื่อระบบติดตั้งเสร็จก็ให้กด Finish แล้วก็ทำการรีสตาร์ทเครื่องใหม่เป็นอันเสร็จสิ้นขั้นตอนการติดตั้ง BREW SDK



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ขั้นตอน การ เรียก BREW Application มาทดสอบ (ในที่หมายถึง BREW Emulator)
 ในเอกสารนี้จะทำการเรียก Emergency Rescue ซึ่งเป็น BREW Emulator ที่ได้ทำการ
 Compile และ Convert content ให้เป็นไฟล์ .dll มาเรียกให้ทำงานบน BREW Emulator

2.1 ทำการ folder file ของ BREW Application(Emergency Rescue) ไปเก็บ ที่ C:\Program
 Files\BREW SDK v2.0.1\Examples (หรือ ที่ใด ๆ ก็ได้ตามความเหมาะสม) ภายในจะประกอบด้วย
 file .mif 1 file และ 1 folder ดังนี้

a.) emergency.mif

b.) emergency\emergency.bar และ emergency\emergency.dll

2.2 เปิด BREW Emulator ตามขั้นตอนที่ 2.2

2.3 หลังจาก Emulator Program เปิดขึ้นให้ไปที่ File แล้วเลือก Change Applet Dir...

2.4 ทำการ ไปยัง Folder ที่เก็บ BREW Application ตามที่เราทำไว้ในข้อ 3.1 (ในที่นี้คือ
 C:\Program Files\BREW SDK v2.0.1\Examples) แล้ว กด OK

2.5 BREW Emulator จะทำการเปิด BREW Application Emergency Rescue ขึ้นมา ให้
 พร้อมใช้งานหลังจากนี้เราก็สามารถใช้ Emulator เข้าไปทดสอบ Application และความถูกต้องของ
 contents ภายในได้



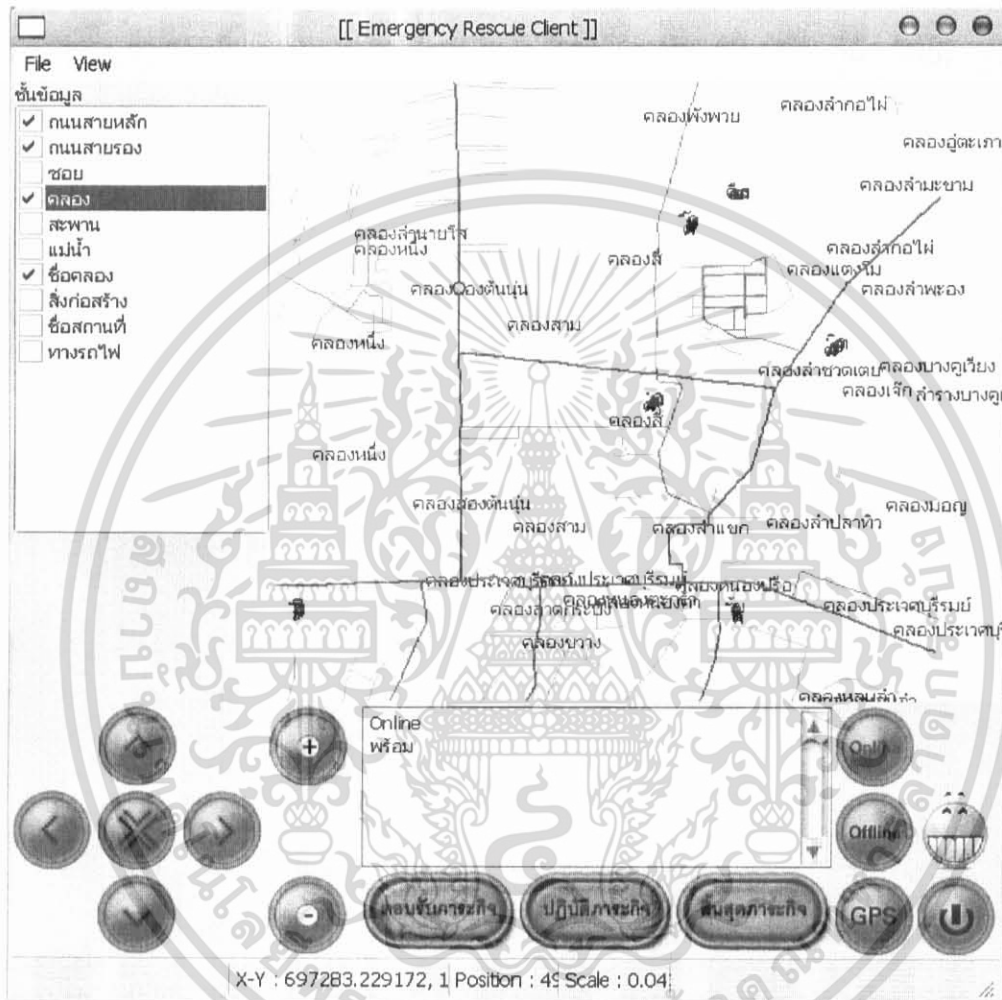
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนของการพัฒนาโปรแกรมบรรดุกเงิน

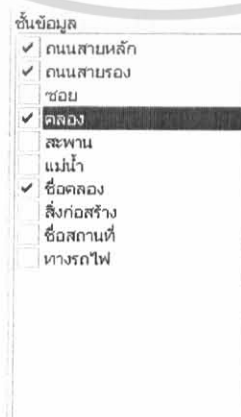
ขั้นตอนการติดตั้งโปรแกรมที่ใช้ในการแสดงผลบรรดุกเงิน

โปรแกรมที่ใช้ในการแสดงผลบรรดุกเงิน โดยมีขั้นตอนทำงานอย่างง่ายมีดังนี้

1. คลิกที่ไฟล์ client.exe เพื่อเปิดโปรแกรม จากนั้น โปรแกรมเริ่มต้นจะเป็นดังรูปด้านล่าง






2. เปิดชั้นข้อมูลที่ต้องการ (โปรแกรมจะโหลดข้อมูล เมื่อเปิดชั้นข้อมูลเป็นครั้งแรก)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความหมายและหน้าที่ของปุ่มต่างๆของโปรแกรมไคลเอนต์

ส่วนควบคุมการให้บริการ

ปุ่ม	ความหมาย
	ตอบรับการเรียกขอความช่วยเหลือ
	ขณะที่กำลังให้ความช่วยเหลือ
	เมื่อสิ้นสุดการให้ความช่วยเหลือ

ส่วนควบคุมแผนที่

ปุ่ม	ความหมาย
	เลื่อนขึ้น
	เลื่อนลง
	เลื่อนซ้าย
	เลื่อนขวา
	ปรับแผนที่ให้เต็มหน้าจอ
	ขยายแผนที่
	ย่อแผนที่

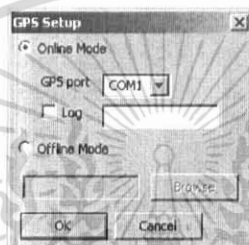
ส่วนควบคุมเครื่องมืออื่นๆ

ปุ่ม	ความหมาย
	เริ่มส่งตำแหน่งรถถูกเงินไปเซิร์ฟเวอร์
	หยุดส่งตำแหน่งรถถูกเงินไปยังเซิร์ฟเวอร์
	กำหนดค่าเกี่ยวกับการติดต่อ GPS
	แสดงว่ายังไม่มีการเรียกขอความช่วยเหลือ
	แสดงว่ามีการเรียกขอความช่วยเหลือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนการใช้งานโปรแกรม

1. ล็อกอินเข้าสู่ระบบ (โดยจะต้องเชื่อมต่อกับอินเทอร์เน็ตไร้สายโดยใช้โปรแกรม Airplus ที่มากับ CDMA modem)
2. กรอกข้อมูลสำหรับการล็อกอิน
3. เมื่อเข้าสู่ระบบเรียบร้อยแล้ว จากนั้นทำการตั้งค่าการเชื่อมต่อกับ GPS โดยเลือก COM port ที่เหมาะสม ถ้าต้องการเก็บ logging ข้อมูล GPS ด้วยให้เลือกช่อง log และระบุชื่อไฟล์ที่ต้องการเก็บ logging (ส่วน Offline mode เป็นโหมดสำหรับการทดลองโดยสามารถเปิดไฟล์ที่บันทึก logging เพื่อให้โปรแกรมอ่านข้อมูลตำแหน่งแทนข้อมูลจาก GPS)



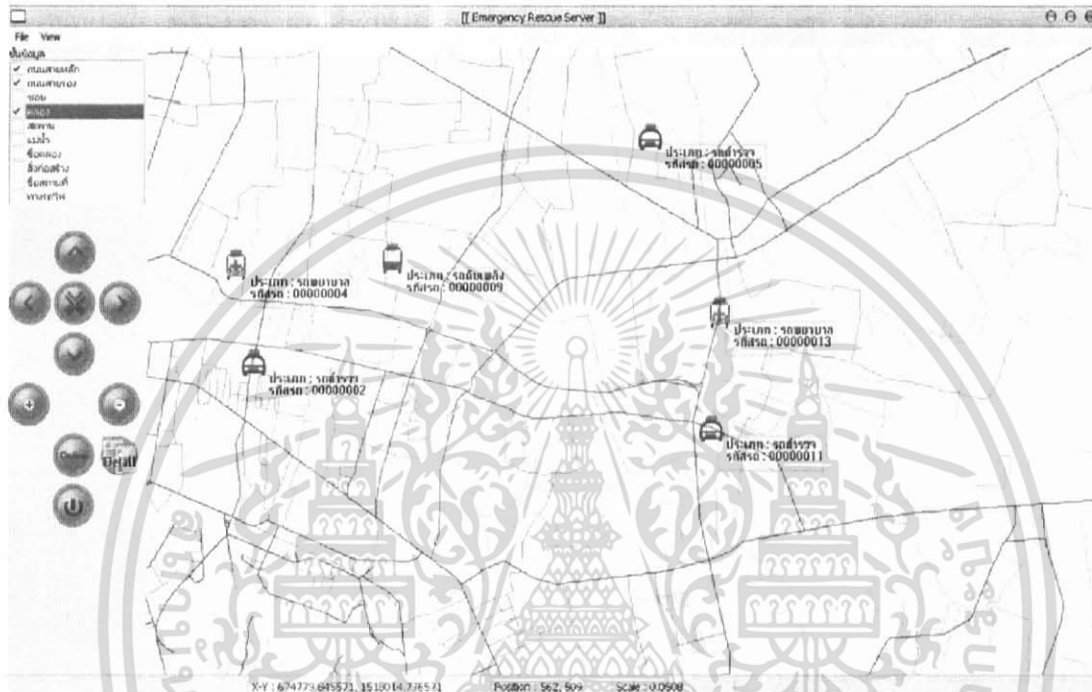
4. เมื่อโปรแกรมทำการล็อกอินเข้าสู่ระบบเรียบร้อยแล้ว โปรแกรมก็พร้อมสำหรับให้บริการแก่ผู้ประสบความสำเร็จอื่น
5. ในกรณีที่มีการขอบริการมาจากเซิร์ฟเวอร์ จะมีเสียงเตือน และแสดงรายละเอียดของบริการให้ทราบ
6. รถฉุกเฉินสามารถตอบรับผู้ประสบความสำเร็จอื่นด้วย 
7. เมื่อรถฉุกเฉินตอบรับบริการผู้ประสบความสำเร็จอื่นสีของรถฉุกเฉินจะเป็นสีเหลือง
8. เมื่อรถฉุกเฉินรับผู้ป่วยขึ้นรถโดยคนป้อน  รถฉุกเฉินเปลี่ยนเป็น สีแดง
9. และเมื่อรถฉุกเฉินส่งผู้ป่วยถึงโรงพยาบาลเรียบร้อยแล้ว โดยคนป้อน  รถจะกลับกลายเป็นสภาพพร้อมให้บริการอีกครั้งคือสีเขียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนของการพัฒนาโปรแกรมที่ใช้ติดตามรถฉุกเฉินทุกคัน

ขั้นตอนการติดตั้งโปรแกรมที่ใช้ติดตามรถฉุกเฉินทุกคัน

โดยในส่วนของระบบ ที่ใช้ติดตามรถฉุกเฉินทุกคัน จะมีการติดตั้งเหมือนกันกับโปรแกรมที่ติดตั้งอยู่บนรถฉุกเฉิน แต่จะมีส่วนควบคุมบางส่วนที่ไม่มีเหมือนกับในส่วนของโปรแกรมในส่วนที่อยู่บนรถฉุกเฉิน และจะมีหน้าตาคล้ายๆกัน ดังรูปด้านล่างนี้



ขั้นตอนการใช้งานโปรแกรม

หลังจากที่ทำตามขั้นตอนข้างต้นแล้ว เมื่อจะทำการแสดงผลรถฉุกเฉินทุกคันที่อยู่ในระบบ ก็คลิกเลือกไปที่  จากนั้นเมื่อคลิกเลือกแล้ว โปรแกรมก็จะแสดงรถฉุกเฉินทุกคันพร้อมกับรายละเอียดของรถ เช่น ประเภทของรถ และ รหัสของรถฉุกเฉิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข. รายละเอียดไฟล์โปรแกรมและโมดูลทั้งหมด

ในคู่มือสำหรับโปรแกรมของโครงการนี้ได้แบ่งออกเป็นส่วนหลักๆ ได้ 6 ส่วนดังนี้

1. ส่วนของฐานข้อมูลแผนที่
2. ส่วนของเว็บเซิร์ฟเวอร์
3. ส่วนที่เป็นหน้าเว็บ
4. โปรแกรมสำหรับรถฉุกเฉิน
5. โปรแกรมที่ใช้ในการติดตามรถฉุกเฉินทุกคันที่อยู่ในระบบ
6. โปรแกรมบนโทรศัพท์เคลื่อนที่

1. ส่วนของฐานข้อมูลแผนที่

1.1 ข้อมูลแผนที่เป็นพื้นที่เฉพาะกรุงเทพ รูปแบบ UTM มาตรฐาน WGS-84 เราได้ใช้ PostgreSQL ในการจัดเก็บข้อมูล โดยมีชั้นข้อมูลดังนี้

1. ชั้นข้อมูลถนนสายหลัก
2. ชั้นข้อมูลถนนสายรอง
3. ชั้นข้อมูลท่าคลอง
4. ชั้นข้อมูลโรงพยาบาล
5. ชั้นข้อมูลโรงเรียน
6. ชั้นข้อมูลสถานีตำรวจนครบาล
7. ชั้นข้อมูลสถานีอนามัย
8. ชั้นข้อมูลบ้าน
9. ชั้นข้อมูลหมู่บ้าน
10. ชั้นข้อมูลมหาวิทยาลัย
11. ชั้นข้อมูลสถานีรถไฟ
12. ชั้นข้อมูลสถานีดับเพลิง
13. ชั้นข้อมูลตลาด

1.2 ข้อมูลของรถฉุกเฉินก็จะเก็บตำแหน่งของรถและรายละเอียดต่างของรถด้วย

1.3 ข้อมูลของการเรียกขอความช่วยเหลือจะมีตำแหน่งและประเภทของการเรียกขอความช่วยเหลือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ส่วนของเว็บเซิร์ฟเวอร์

2.1 ส่วนที่ให้บริการข้อมูลแผนที่ ส่วนนี้จะสร้างข้อมูลแผนที่ในรูปแบบของเอกสาร XML การเรียกใช้งานก็จะทำการเรียกใช้ผ่าน HTTP Protocol ในลักษณะของ Query String

มีไฟล์ อยู่ 1 ไฟล์ คือ mapgml.py มีฟังก์ชันดังนี้

1. ฟังก์ชัน tagAmphoe50 มีพารามิเตอร์หนึ่งตัว ชื่อว่า layer เป็นชั้นข้อมูลที่เป็นเขตของกรุงเทพทั้ง 50 เขต ฟังก์ชันนี้จะรีเทิร์นเป็น string ที่เป็นแท็ก XML
2. ฟังก์ชัน amphoe50 ไม่มีพารามิเตอร์ฟังก์ชันนี้จะถูกเรียกใช้จากหน้าเว็บไซต์และการทำงานคือจะทำการเปิดชั้นข้อมูลเขตแล้วเอาชั้นข้อมูลที่ได้เป็นพารามิเตอร์เพื่อเรียกใช้ฟังก์ชัน tagAmphoe50 หลังจากเรียกแล้วจะได้แท็ก XML และเอาแท็กนี้เป็นผลลัพธ์ของฟังก์ชัน amphoe50
3. ฟังก์ชัน tagRiver1 มีพารามิเตอร์หนึ่งตัวชื่อว่า layer เป็นชั้นข้อมูลลำคลองทั่วเขตกรุงเทพ ฟังก์ชันนี้จะให้ผลลัพธ์เป็น String ที่เป็นแท็ก XML
4. ฟังก์ชัน river1 ไม่มีพารามิเตอร์ฟังก์ชันนี้จะถูกเรียกใช้จากหน้าเว็บไซต์และการทำงานคือจะทำการเปิดชั้นข้อมูลลำคลองในกรุงเทพแล้วเอาชั้นข้อมูลที่ได้เป็นพารามิเตอร์เพื่อเรียกใช้ฟังก์ชัน tagRiver1 หลังจากเรียกแล้วจะได้แท็ก XML และเอาแท็กนี้เป็นผลลัพธ์ของฟังก์ชัน river1
5. ฟังก์ชัน tagMainRoad มีพารามิเตอร์หนึ่งตัวชื่อว่า layer เป็นชั้นข้อมูลถนนสายหลักทั่วเขตกรุงเทพ ฟังก์ชันนี้จะให้ผลลัพธ์เป็น String ที่เป็นแท็ก XML
6. ฟังก์ชัน mainroad ไม่มีพารามิเตอร์ฟังก์ชันนี้จะถูกเรียกใช้จากหน้าเว็บไซต์และการทำงานคือจะทำการเปิดชั้นข้อมูลถนนสายหลักในกรุงเทพแล้วเอาชั้นข้อมูลที่ได้เป็นพารามิเตอร์เพื่อเรียกใช้ฟังก์ชัน tagMainRoad หลังจากเรียกแล้วจะได้แท็ก XML และเอาแท็กนี้เป็นผลลัพธ์ของฟังก์ชัน mainroad
7. ฟังก์ชัน tagPlace มีพารามิเตอร์หนึ่งตัวชื่อว่า layer เป็นชั้นข้อมูลสถานที่ต่างๆ เช่น โรงพยาบาล โรงเรียน สถานีตำรวจเป็นต้น ฟังก์ชันนี้จะให้ผลลัพธ์เป็น String ที่เป็นแท็ก XML
8. ฟังก์ชัน place มีพารามิเตอร์หนึ่งตัวชื่อว่า typePlace ประเภท String ค่าที่เป็นไปได้มีดังนี้
 - hospital
 - school
 - policestation
 - hospitalstation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- house
- ghouse
- subdepartment
- market
- trainstation
- office
- univer
- grm
- firestation

ฟังก์ชันนี้จะถูกเรียกใช้จากหน้าเว็บไซต์และการทำงานคือจะทำการเปิดชั้นข้อมูลสถานที่ต่างๆ ในกรุงเทพตามพารามิเตอร์ที่ส่งมาแล้วเอาชั้นข้อมูลที่ได้เป็นพารามิเตอร์เพื่อเรียกใช้ฟังก์ชัน tagPlace หลังจากเรียกแล้วจะได้แท็ก XML และเอาแท็กนี้เป็นผลลัพธ์ของฟังก์ชัน place

9. ฟังก์ชัน getMainRoad1234 มีพารามิเตอร์หนึ่งตัวชื่อว่า layer เป็นชั้นข้อมูลถนนสายรองทั่วเขตกรุงเทพ ฟังก์ชันนี้จะให้ผลลัพธ์เป็น String ที่เป็นแท็ก XML
10. ฟังก์ชัน subroad ไม่มีพารามิเตอร์ฟังก์ชันนี้จะถูกเรียกใช้จากหน้าเว็บไซต์และการทำงานคือจะทำการเปิดชั้นข้อมูลถนนสายรองในกรุงเทพแล้วเอาชั้นข้อมูลที่ได้เป็นพารามิเตอร์เพื่อเรียกใช้ฟังก์ชัน getMainRoad1234 หลังจากเรียกแล้วจะได้แท็ก XML และเอาแท็กนี้เป็นผลลัพธ์ของฟังก์ชัน subroad

2.2 ส่วนที่ให้บริการการเรียกขอความช่วยเหลือ มีหนึ่งไฟล์คือ serv.py

1. ฟังก์ชัน requests มีพารามิเตอร์ 3 ตัวดังนี้
 - type คือ ประเภทของรถฉุกเฉิน
 - x_pos คือ ตำแหน่ง X ของการเรียกขอความช่วยเหลือ
 - y_pos คือ ตำแหน่ง Y ของการเรียกขอความช่วยเหลือ
 ฟังก์ชันนี้จะทำการจัดการรถฉุกเฉินคันที่ใกล้กับจุดเรียกขอความช่วยเหลือจากหน้าเว็บไซต์
2. ฟังก์ชัน requestsfrommobile มีพารามิเตอร์ 3 ตัวดังนี้
 - type คือ ประเภทของรถฉุกเฉิน
 - x_pos คือ ตำแหน่ง X ของการเรียกขอความช่วยเหลือ
 - y_pos คือ ตำแหน่ง Y ของการเรียกขอความช่วยเหลือ
 จัดการรถฉุกเฉินคันที่ใกล้กับจุดเรียกขอความช่วยเหลือจากโทรศัพท์เคลื่อนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ฟังก์ชัน `onaccept` มีพารามิเตอร์ 1 ตัวคือ `car_id` รหัสของรถถูกเงิน ฟังก์ชันนี้จะทำการเปลี่ยนสถานะของรถถูกเงินเมื่อตอบรับการเรียกขอความช่วยเหลือ
4. ฟังก์ชัน `onprocess` มีพารามิเตอร์ 1 ตัวคือ `car_id` รหัสของรถถูกเงิน ฟังก์ชันนี้จะทำการ เปลี่ยนสถานะของรถถูกเงินเมื่อกำลังให้ความช่วยเหลือหรือกำลังปฏิบัติการกิจของเจ้าหน้าที่ประจำหน่วยถูกเงินนั้น
5. ฟังก์ชัน `onready` มีพารามิเตอร์ 1 ตัวคือ `car_id` รหัสของรถถูกเงิน ฟังก์ชันนี้จะทำการ เปลี่ยนสถานะของรถถูกเงินที่ปฏิบัติการกิจเสร็จแล้ว

6. ฟังก์ชัน `online` มีพารามิเตอร์ 3 ตัวดังนี้
 - `car_id` คือ รหัสของรถถูกเงิน
 - `x_pos` คือ ตำแหน่ง X ของรถถูกเงิน
 - `y_pos` คือ ตำแหน่ง Y ของรถถูกเงิน
 สำหรับอัปเดตตำแหน่งของรถถูกเงินแต่ละคันเพื่อการติดตามของศูนย์และญาติผู้เข้ามา ติดตาม โดยรถถูกเงินแต่ละคันจะส่งตำแหน่งของตนเองมาเป็นระยะๆ เพื่ออัปเดตตำแหน่งตัวเอง
7. ฟังก์ชัน `ontrack` มีพารามิเตอร์ ชื่อ `serv_id` รหัสการขอความช่วยเหลือ ฟังก์ชันนี้จะใช้ในการขอตำแหน่งของรถถูกเงินเพื่อใช้ในการติดตามรถถูกเงินคันนั้นๆ ไป ผลลัพธ์ของฟังก์ชันคือตำแหน่งและสถานะของรถถูกเงินที่ประจำรหัสการเรียกขอความช่วยเหลือ
8. ฟังก์ชัน `carserver` ไม่มีพารามิเตอร์ จะใช้ในการขอเอาตำแหน่งและสถานะของรถถูกเงินทุกคันในระบบเพื่อเอาตำแหน่งนั้นมาแสดงผลบนแผนที่

3. ส่วนที่เป็นหน้าเว็บ

3.1 สำหรับส่วนของเว็บไซต์รถถูกเงิน มีไฟล์ภาษาจาวาสคริปต์ 3 ไฟล์ดังนี้คือ

1. `bkkbtn.js` จัดการเกี่ยวกับปุ่มต่างๆ ในหน้าเว็บไซต์
2. `loaddata.js` จัดการ โหลดข้อมูลแผนที่จากเว็บเซิร์ฟเวอร์ส่วนที่ให้บริการข้อมูลแผนที่พร้อมทั้งสร้างแท็ก SVG เพื่อแสดงแผนที่บนหน้าเว็บไซด์
3. `search.js` การค้นหาสถานที่ตามชื่อสถานที่ที่ผู้ใช้ต้องการหา

และมีฟังก์ชันต่างๆ ดังนี้

1. ฟังก์ชันในไฟล์ `bkkbtn.js` มี 4 ฟังก์ชัน คือ

- ฟังก์ชัน `overToggle(btnid)` มีพารามิเตอร์ชื่อว่า `btnid` ประเภท `String` ก็คือ ID ของปุ่มในหน้าเว็บเพจฟังก์ชันนี้จะทำการเปลี่ยนรูปของปุ่มเมื่อมีเมาส์โอเวอร์ที่ปุ่มนั้น
- ฟังก์ชัน `outToggle(btnid)` มีพารามิเตอร์ชื่อว่า `btnid` ประเภท `String` ก็คือ ID ของปุ่มในหน้าเว็บเพจฟังก์ชันนี้จะทำการเปลี่ยนรูปของปุ่มเมื่อเมาส์เคลื่อนที่ออกจากปุ่มนั้นแล้ว
- ฟังก์ชัน `overSearch(btnid)` มีพารามิเตอร์ชื่อว่า `btnid` ประเภท `String` ก็คือ ID ของปุ่มในหน้าเว็บเพจฟังก์ชันนี้จะเปลี่ยนรูปภาพของปุ่มเมื่อมีการเคลื่อนที่เมาส์อยู่บนปุ่มนั้น
- ฟังก์ชัน `outSearch(btnid)` มีพารามิเตอร์ชื่อว่า `btnid` ประเภท `String` ก็คือ ID ของปุ่มในหน้าเว็บเพจฟังก์ชันนี้จะเปลี่ยนรูปภาพของปุ่มนั้นเมื่อมีการเคลื่อนที่เมาส์ออกจากปุ่ม

2. ฟังก์ชันในไฟล์ `loaddata.js`

ฟังก์ชัน `GetXmlHttpRequest()` ไม่มีพารามิเตอร์ฟังก์ชันนี้จะใช้สร้าง `XMLHttpRequest` Object
ฟังก์ชัน `river1()` ไม่มีพารามิเตอร์ฟังก์ชันนี้จะทำการโหลดข้อมูลแผนที่ชั้นข้อมูลลำคลองจากเว็บเซิร์ฟเวอร์

ฟังก์ชัน `stateChangedRiver1()` ฟังก์ชันนี้จะสร้างแท็ก `SVG` แสดงผลภาพแผนที่โดยเอาข้อมูลแผนที่จากฟังก์ชัน `river1()` ที่โหลดมาแล้วนั้น

ฟังก์ชัน `mainroad1234()` ฟังก์ชันนี้จะโหลดข้อมูลแผนที่ชั้นข้อมูลถนนสายรองจากเว็บเซิร์ฟเวอร์

ฟังก์ชัน `stateChangedSubRoad()` ฟังก์ชันนี้จะสร้างแท็ก `SVG` แสดงผลภาพแผนที่โดยเอาข้อมูลแผนที่จากฟังก์ชัน `mainroad1234()` ที่โหลดมาแล้วนั้น

ฟังก์ชัน `mainRoad()` ฟังก์ชันนี้จะโหลดข้อมูลแผนที่ชั้นข้อมูลถนนสายหลักจากเว็บเซิร์ฟเวอร์

ฟังก์ชัน `stateChangedMainRoad()` ฟังก์ชันนี้จะสร้างแท็ก `SVG` แสดงผลภาพแผนที่โดยเอาข้อมูลแผนที่จากฟังก์ชัน `mainRoad()` ที่โหลดมาแล้วนั้น

ฟังก์ชัน `loadamphoe()` ฟังก์ชันนี้จะโหลดข้อมูลแผนที่ชั้นข้อมูลเขตของกรุงเทพฯ จากเว็บเซิร์ฟเวอร์

ฟังก์ชัน `stateChangedAmphoe50()` ฟังก์ชันนี้จะสร้างแท็ก `SVG` แสดงผลภาพแผนที่โดยเอาข้อมูลแผนที่จากฟังก์ชัน `loadamphoe()` ที่โหลดมาแล้วนั้น

ฟังก์ชัน `hospital()` ฟังก์ชันนี้จะโหลดข้อมูลแผนที่ชั้นข้อมูลที่ตั้งโรงพยาบาลในกรุงเทพฯ จากเว็บเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน `stateChangedHospital ()` ฟังก์ชันนี้จะสร้างแท็ก SVG แสดงผลภาพแผนที่โดยเอาข้อมูลแผนที่จากฟังก์ชัน `hospital ()` ที่โหลดมาแล้วนั้น

ฟังก์ชัน `school ()` ฟังก์ชันนี้จะโหลดข้อมูลแผนที่ชั้นข้อมูลที่ตั้งโรงเรียนในกรุงเทพจากเว็บเซิร์ฟเวอร์

ฟังก์ชัน `stateChangedSchool ()` ฟังก์ชันนี้จะสร้างแท็ก SVG แสดงผลภาพแผนที่โดยเอาข้อมูลแผนที่จากฟังก์ชัน `school ()` ที่โหลดมาแล้วนั้น

ฟังก์ชัน `police ()` ฟังก์ชันนี้จะโหลดข้อมูลแผนที่ชั้นข้อมูลที่ตั้งสถานีตำรวจนครบาลในกรุงเทพจากเว็บเซิร์ฟเวอร์

ฟังก์ชัน `stateChangedPolice ()` ฟังก์ชันนี้จะสร้างแท็ก SVG แสดงผลภาพแผนที่โดยเอาข้อมูลแผนที่จากฟังก์ชัน `police ()` ที่โหลดมาแล้วนั้น

ฟังก์ชัน `stHospital ()` ฟังก์ชันนี้จะโหลดข้อมูลแผนที่ชั้นข้อมูลที่ตั้งสถานีอนามัยในเขตกรุงเทพจากเว็บเซิร์ฟเวอร์

ฟังก์ชัน `stateChangedStHospital ()` ฟังก์ชันนี้จะสร้างแท็ก SVG แสดงผลภาพแผนที่โดยเอาข้อมูลแผนที่จากฟังก์ชัน `stHospital ()` ที่โหลดมาแล้วนั้น

ฟังก์ชัน `lhouse ()` ฟังก์ชันนี้จะโหลดข้อมูลแผนที่ชั้นข้อมูลที่ตั้งบ้านในเขตกรุงเทพจากเว็บเซิร์ฟเวอร์

ฟังก์ชัน `stateChangedHouse ()` ฟังก์ชันนี้จะสร้างแท็ก SVG แสดงผลภาพแผนที่โดยเอาข้อมูลแผนที่จากฟังก์ชัน `lhouse ()` ที่โหลดมาแล้วนั้น

ฟังก์ชัน `ghouse ()` ฟังก์ชันนี้จะโหลดข้อมูลแผนที่ชั้นข้อมูลที่ตั้งหมู่บ้านในเขตกรุงเทพจากเว็บเซิร์ฟเวอร์

ฟังก์ชัน `stateChangedGHouse ()` ฟังก์ชันนี้จะสร้างแท็ก SVG แสดงผลภาพแผนที่โดยเอาข้อมูลแผนที่จากฟังก์ชัน `ghouse ()` ที่โหลดมาแล้วนั้น

ฟังก์ชัน `complex ()` ฟังก์ชันนี้จะโหลดข้อมูลแผนที่ชั้นข้อมูลที่ตั้งกรมต่างๆ ในเขตกรุงเทพจากเว็บเซิร์ฟเวอร์

ฟังก์ชัน `stateChangedComplex ()` ฟังก์ชันนี้จะสร้างแท็ก SVG แสดงผลภาพแผนที่โดยเอาข้อมูลแผนที่จากฟังก์ชัน `complex ()` ที่โหลดมาแล้วนั้น

ฟังก์ชัน `market ()` ฟังก์ชันนี้จะโหลดข้อมูลแผนที่ชั้นข้อมูลที่ตั้งตลาดต่างๆ ในเขตกรุงเทพจากเว็บเซิร์ฟเวอร์

ฟังก์ชัน `stateChangedMarket ()` ฟังก์ชันนี้จะสร้างแท็ก SVG แสดงผลภาพแผนที่โดยเอาข้อมูลแผนที่จากฟังก์ชัน `market ()` ที่โหลดมาแล้วนั้น

ฟังก์ชัน `train ()` ฟังก์ชันนี้จะโหลดข้อมูลแผนที่ชั้นข้อมูลที่ตั้งสถานีรถไฟในเขตกรุงเทพจากเว็บเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน `stateChangedTrain()` ฟังก์ชันนี้จะสร้างแท็ก SVG แสดงผลภาพแผนที่โดยเอาข้อมูลแผนที่จากฟังก์ชัน `train()` ที่โหลดมาแล้วนั้น

ฟังก์ชัน `univer()` ฟังก์ชันนี้จะโหลดข้อมูลแผนที่ชั้นข้อมูลที่ตั้งมหาวิทยาลัยในเขตกรุงเทพจากเว็บเซิร์ฟเวอร์

ฟังก์ชัน `stateChangedUniver()` ฟังก์ชันนี้จะสร้างแท็ก SVG แสดงผลภาพแผนที่โดยเอาข้อมูลแผนที่จากฟังก์ชัน `univer()` ที่โหลดมาแล้วนั้น

ฟังก์ชัน `firestation()` ฟังก์ชันนี้จะโหลดข้อมูลแผนที่ชั้นข้อมูลที่ตั้งสถานีดับเพลิงในเขตกรุงเทพจากเว็บเซิร์ฟเวอร์

ฟังก์ชัน `stateChangedFire()` ฟังก์ชันนี้จะสร้างแท็ก SVG แสดงผลภาพแผนที่โดยเอาข้อมูลแผนที่จากฟังก์ชัน `firestation()` ที่โหลดมาแล้วนั้น

3. ฟังก์ชันในไฟล์ `insearch.js`

ฟังก์ชัน `insearch(xmlDocument, namelayer)` มีพารามิเตอร์ 2 ตัวคือ

- `xmlDocument` ประเภท `xmlDocument Object` ชุดข้อมูลที่ต้องการจะค้นหา

- `namelayer` ประเภท `String` ชื่อของชั้นข้อมูล

ฟังก์ชันนี้จะทำการเอาค่าที่ผู้ใช้ต้องการค้นหาที่ได้ป้อนเข้ามาไป

เปรียบเทียบกับในเอกสาร XML `xmlDocument Object` ที่เป็นพารามิเตอร์

ฟังก์ชัน `onmarking(x_utm, y_utm, sl1, sl2, sl3, sl4)`

- `x_utm` ชนิดข้อมูล จำนวนจริงเป็นตำแหน่งแกน X ที่ต้องการทำเครื่องหมาย

- `y_utm` ชนิดข้อมูล จำนวนจริงเป็นตำแหน่งแกน Y ที่ต้องการทำเครื่องหมาย

- `sl1, sl2, sl3` และ `sl4` ชนิดข้อมูล `String` เป็นข้อมูลทั่วไปที่ต้องการแสดงบนแผนที่

ฟังก์ชัน `imapsearch()` ฟังก์ชันนี้จะตรวจสอบว่ามีการเปิดชั้นข้อมูลของผู้ใช้ใหม่และมีการป้อนค่าที่ต้องการหาในช่องใหม่ถ้าไม่ทำอย่างใดอย่างหนึ่ง โปรแกรมจะไม่ทำการค้นหาให้

4. ไฟล์ SVG ที่แสดงแผนที่ในไฟล์นี้ก็จะฟังก์ชันที่ใช้ควบคุมแผนที่เช่น ย่อขยาย และเลื่อน แผนที่

- ฟังก์ชัน `onRequest(type)` ฟังก์ชันนี้จะส่งประเภทของรถฉุกเฉินที่จะขอความช่วยเหลือพร้อมกับตำแหน่งของการเรียกนั้นไปยังเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พารามิเตอร์มีหนึ่งตัวคือ type ชนิด String เพื่อบอกว่าเป็นรถฉุกเฉิน
ประเภทไหนซึ่งฟังก์ชัน onRequest นี้จะอยู่ในไฟล์ mapscallcenter.svg

3.2 สำหรับให้ญาติหรือเพื่อนของผู้เรียกขอความช่วยเหลือใช้รหัสการเรียกขอความช่วยเหลือที่ส่งอีเมลไปให้กับญาติอีกคนเข้าไปติดตามสถานการณ์ก็จะอีกหน้า
เว็บไซต์หนึ่ง สามารถแยกย่อยได้อีก ดังนี้

- ฟังก์ชัน Javascript สำหรับโหลดข้อมูลแผนที่จากเว็บเซิร์ฟเวอร์ส่วนที่ให้บริการข้อมูลแผนที่พร้อมทั้งสร้างแท็ก SVG เพื่อแสดงแผนที่และจะมีฟังก์ชันสำหรับแต่ละชั้นข้อมูล ดังนั้นจะมีฟังก์ชันการจัดการแผนที่อันเดียวกันกับเว็บไซต์ส่วนที่ใช้เรียกรถฉุกเฉิน
- ฟังก์ชัน onTrack () ซึ่งเป็น Javascript สำหรับไปขอเอาข้อมูลตำแหน่งของรถฉุกเฉินทุกๆ หนึ่งวินาทีเพื่อติดตามรถฉุกเฉินคันที่ให้ความช่วยเหลือกับรหัสการเรียกขอความช่วยเหลือนั้นๆ

4. โปรแกรมสำหรับรถฉุกเฉินและโปรแกรมที่ใช้ในการติดตามรถฉุกเฉินทุกคันที่อยู่ในระบบ

โปรแกรมใน 2 ส่วนนี้จะมีคลาสหลักๆ ดังนี้ คือ

1. LayerPanel คลาสนี้จะสร้าง และ ลบ ชั้นข้อมูลแผนที่ซึ่งจะมี Member function ดังนี้

- CreateLayer(self, id, layername, LineStyle=None, PolygonStyle=None, TextStyle=None, fillColor='red', lineColor='black')
- CreateTrack(self, trackname, LineStyle=None)
- CreateRequest(self, trackname, LineStyle=None,)
- GetLayerByID(self, id)
- GetLayerByName(self, layername)
- RemoveLayerByName(self, layername)
- EvtCheckListBox(self, event)

2. ContrlPanel คลาสนี้จะใช้ควบคุมแผนที่เช่นย่อขยาย และ เลื่อนแผนที่ ซึ่งจะมี Member function ดังนี้คือ

- ZoomInOnOver(self, event)
- ZoomInOnOut(self, event)
- ZoomOutOnOver(self, event)
- ZoomOutOnOut(self, event)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- PlanOnOver(self, event)
- PlanOnOut(self, event)
- UpBtnOnOver(self, event)
- UpBtnOnOut(self, event)
- DownBtnOnOver(self, event)
- DownBtnOnOut(self, event)
- RightBtnOnOver(self, event)
- RightBtnOnOut(self, event)
- LeftBtnOnOver(self, event)
- LeftBtnOnOut(self, event)
- zToFitBtnOnOver(self, event)
- zToFitBtnOnOut(self, event)
- BuildToolbar(self)
- OnZoomIn(self, event)
- OnZoomOut(self, event)
- OnZoomFit(self, event)
- OnMoveLeft(self, event)
- OnMoveRight(self, event)
- OnMoveUp(self, event)
- OnMoveDown(self, event)

3. MapPanel เป็นคลาสจัดการ (Handle Class) จะมีการประกาศ Object LayerPanel กับ Object ControlPanel ขึ้นมาจัดการและมี Member function ดังนี้คือ

- OnMove(self, event)
- OnWheel(self, event)
- SetSpecialMode(self, mode)
- Update1(self) หมายถึง จะมีเฉพาะในส่วนของการติดตามรถทุกคัน
- Update2(self) หมายถึง จะมีเฉพาะในส่วนของการติดตามรถทุกคัน
- OnAccept(self, event) หมายถึง จะมีเฉพาะในส่วนของการรถฉุกเฉิน
- OnProcess(self, event) หมายถึง จะมีเฉพาะในส่วนของการรถฉุกเฉิน
- OnComplete(self, event) หมายถึง จะมีเฉพาะในส่วนของการรถฉุกเฉิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. DrawFrame เป็นคลาสที่คอยจัดการพิกัดอีเวนต์ต่างๆ ของโปรแกรม ซึ่งมี Member function ดังนี้คือ

- onSave(self, event)
- OnQuit(self, event)
- OnCloseWindow(self, event)
- ZoomToFit(self, event)
- SetupGPS(self, event)
- OnLine(self, event)
- OffLine(self, event)
- OnStartStop(self, event)

5. โปรแกรมบนโทรศัพท์เคลื่อนที่

โปรแกรมในส่วนของโทรศัพท์เคลื่อนที่นี้ประกอบไปด้วยไฟล์ที่ได้พัฒนาหลักๆ ที่เกี่ยวข้องมีอยู่ทั้งหมด 3 ไฟล์ ดังนี้ emergency.c, emergency.h, emergency_res.h

■ emergency.c

ไฟล์นี้เป็นไฟล์ที่ทำหน้าที่เป็นหลักของการทำงานทั้งหมดของโปรแกรมในส่วนของโทรศัพท์เคลื่อนที่ ภายในไฟล์นี้มีการแบ่งเป็นโปรแกรมย่อยๆ จำนวน 23 โปรแกรมย่อย

1. ฟังก์ชัน emergency_HandleEvent
2. ฟังก์ชัน emergency_FreeAppData
3. ฟังก์ชัน emergency_InitAppData
4. ฟังก์ชัน emergency_CleanUp
5. ฟังก์ชัน showMainMenu
6. ฟังก์ชัน showAmbulanceMenu
7. ฟังก์ชัน showPoliceMenu
8. ฟังก์ชัน showFirerMenu
9. ฟังก์ชัน drawStartAmbulanceLogo
10. ฟังก์ชัน drawStartPoliceLogo
11. ฟังก์ชัน drawStartFirerLogo
12. ฟังก์ชัน cancelCall
13. ฟังก์ชัน callQuickChkType
14. ฟังก์ชัน GetGPSInfo
15. ฟังก์ชัน emergency_GPSCB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

16. ฟังก์ชัน GPSCleanUp
17. ฟังก์ชัน showProgressSuccess
18. ฟังก์ชัน OpenConnectionCB
19. ฟังก์ชัน OpenConnectionToWebsite
20. ฟังก์ชัน DisplayMessage
21. ฟังก์ชัน WebCleanup
22. ฟังก์ชัน applenPath
23. ฟังก์ชัน StartProgressDisplay

- emergency.h

ไฟล์นี้เปรียบเสมือนว่าเป็นไฟล์ Header ของไฟล์หลัก ซึ่งจะทำหน้าที่เป็นไฟล์ที่ใช้ในการ include Module ต่างๆ ที่มีอยู่แล้ว นำมาใช้ในโปรแกรมหลัก นิยามตัวแปรที่เป็นแบบ Global และ นิยามฟังก์ชันที่จะถูกนำไปใช้ในไฟล์ที่เป็นโปรแกรมหลัก รวมทั้งการสร้างตัวแปรที่เป็นโครงสร้างแบบ Struct หรือแบบ enum เป็นต้น

- emergency_res.h

ไฟล์นี้เปรียบเสมือนว่าเป็นไฟล์ Header ของไฟล์หลักเช่นกัน ไฟล์นี้ทำหน้าที่เป็นไฟล์ที่ใช้ในการกำหนดคำว่า resource ที่เราจะนำมาใช้ในโปรแกรมหลัก

บรรณานุกรม

เอกสารอ้างอิงที่เป็น Web-site

- [1] AN INTRODUCTION TO C.
Available : <http://www.cs.ntu.edu.au/sit/resources/cprogram/default.htm>
- [2] Extensible Markup Language (XML).
Available : <http://www.w3.org/XML/>
- [3] THAIXML.com : XML From The Inside Out – XML Development, XML Resources.
Available : <http://www.thaixml.com/>
- [4] W3Schools Online Web Tutorials.
Available : <http://www.w3schools.com/>
- [5] XML Tutorials – Free Training.
Available : <http://www.exforsys.com/content/category/17/283/373/>
- [6] How to Perform Geo-Location on BREW – based Phones -
Available : <http://www.devx.com/wirless/Article/11195>
- [7] QUALCOMM BREW Home.
Available : <http://brew.qualcomm.com/brew/en/>
- [8] XML Tutorials – Free Training.
Available : <http://www.exforsys.com/content/category/17/283/373/>
- [9] Philippe Le Hegaret; “Scalable Vector Graphic (SVG) 1.1
Available : <http://www.w3.org/DOM/>
- [10] GML – a Markup Language for Geographiy.
Available : <http://www.opengis.net/gml/>
- [11] Simon Cox; “Geography Markup Language(GML)
Available : <http://www.opengis.net/gml/01-029/GML2.pdf>
- [12] Scalable Vector Graphics (SVG) 1.1 Specification.
Available : <http://www.w3.org/TR/SVG/index.html#minitoc>
- [13] PostgreSQL; “The world ‘s most advanced open source database
Available : <http://www.postgresql.org/>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิงที่เป็นวิทยานิพนธ์

- [14] ฐานันดร พงศ์ภัทรวัฒน์ และ ประภัสร์ รุ่งเรืองอนันต์ “ระบบให้บริการระบุตำแหน่ง โดยใช้ ภาษาเอ็กซ์เอ็มแอลและเว็บเซอร์วิส” วิทยานิพนธ์วิศวกรรมศาสตรบัณฑิต ภาควิชา วิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง 2547
- [15] เกียรติศักดิ์ คิวขุนทด และ นรินทร์ เรืองศรี “ระบบให้บริการแผนที่บนเว็บ” วิทยานิพนธ์ วิศวกรรมศาสตรบัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบัน เทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง 2545
- [16] พงศ์รับ สายสุวรรณ และ พงเทพ ราชวงศ์ “ระบบให้บริการแผนที่” วิทยานิพนธ์ วิศวกรรมศาสตรบัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบัน เทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง 2548
- [17] วิศิษฐ์ นวอิทธิพร , สมพล แซ่ปึง และ สิริวรรณ พรกิตติวัฒนากุล “เว็บเซอร์วิสเชิงพื้นที่ สำหรับระบบสารสนเทศเชิงภูมิศาสตร์” วิทยานิพนธ์วิศวกรรมศาสตรบัณฑิต ภาควิชา วิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง 2547

เอกสารอ้างอิงที่เป็นหนังสือ

- [18] ขยัน จันทรสถาพร 2544 **เรียนลัด XML ฉบับรู้เต็มร้อย!** เอ.อาร์. อินฟอร์เมชันแอนพับลิเคชัน จำกัด กรุงเทพฯ.
- [19] ทวีชัย หงส์สุมาลย์ และ สงวนชัย สุวรรณชีวะศิริ 2545 **ใส่ลูกเล่นให้เว็บไซต์ด้วย Java Script** สำนักพิมพ์ อินโฟเพรส นนทบุรี.
- [20] Jonathan Lipari. **Python HOW TO PROGRAM Introducing XML**, Prentice-Hall, Inc. Upper Saddle River, New Jersey 07458.
- [21] Bill Kropla. **Beginning MapServer Open source GIS Development**, APRESS, United State of America.
- [22] Nicholas C. Zakas. **Professional JavaScript for Web Development**, Wiley Publishing, Inc. Canada.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้