

**สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง**

**ระบบแนะนำเส้นทางในเขตกรุงเทพมหานคร  
ROUTE NAVIGATION SERVICE SYSTEM  
FOR BANGKOK METROPOLITAN AREA**



**ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต**

**ภาควิชาวิศวกรรมสารสนเทศ**

**คณะวิศวกรรมศาสตร์**

**สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง**

**ปีการศึกษา 2550**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ROUTE NAVIGATION SERVICE SYSTEM  
FOR BANGKOK METROPOLITAN AREA**



**A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF  
THE REQUIREMENT FOR THE DEGREE OF  
BACHELOR IN DEPARTMENT OF INFORMATION ENGINEERING  
FACULTY OF ENGINEER  
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

**2007**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	ระบบแนะนำเส้นทางในเขตกรุงเทพมหานคร
นักศึกษา	นางสาวสติกัทยา ศรีภุชณ์ รหัสนักศึกษา 47010821 นายอำนาจ เสงประสิทธิ์วงศ์ รหัสนักศึกษา 47010990
อาจารย์ที่ปรึกษา	ผศ. มยุรี เลิศเวชกุล
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2550

ปริญญานิพนธ์นี้ได้รับความเห็นชอบจากอาจารย์ที่ปรึกษาเป็นที่เรียบร้อยแล้ว



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	ระบบแนะนำเส้นทางในเขตกรุงเทพมหานคร		
ชื่อนักศึกษา	นางสาวสลิภัทรา ศรีภุณณ์		47010821
	นายอำนาจ เสงประสิทธิ์วงศ์		47010990
อาจารย์ที่ปรึกษา	ผศ. มยุรี	เลิศเวชกุล	
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต		
ภาควิชา	วิศวกรรมสารสนเทศ		
ปีการศึกษา	2550		

### บทคัดย่อ

การคมนาคมภายในกรุงเทพมหานครที่พบในปัจจุบันมีการคมนาคมทั้งทางบก ทางน้ำ และทางอากาศ พบว่าประชากรส่วนใหญ่ใช้บริการการคมนาคมทางบกเป็นจำนวนมากไม่ว่าจะเป็นรถบริการขนส่งมวลชน หรือรถยนต์ส่วนบุคคล ปัจจุบันจะเห็นได้ว่าปริมาณของรถยนต์ส่วนบุคคลมีปริมาณมากขึ้นเนื่องจากปัจจัยในการเดินทางในช่วงโมงเร่งด่วนและความสะดวกสบายที่รถบริการขนส่งมวลชนไม่สามารถตอบสนองได้ตามต้องการ ด้วยเหตุนี้จึงทำให้ปริมาณรถยนต์มากขึ้นส่งผลให้สภาพการจราจรติดขัดมากยิ่งขึ้นทำให้ถึงที่หมายล่าช้า ดังนั้นในการศึกษาโครงการนี้มีวัตถุประสงค์คือสร้างระบบแนะนำเส้นทางให้แก่ผู้ใช้งานพาหนะบนท้องถนน เพื่อที่จะสามารถวางแผนการเดินทางล่วงหน้าได้ โดยฐานข้อมูลที่ใช้นำมาวิเคราะห์ได้รับความเชื่อถือมาจากห้องปฏิบัติการเทคโนโลยีเครือข่าย (Network Technology Laboratory) ภายใต้ศูนย์วิจัยอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งประเทศไทย (NECTEC) ในระบบแนะนำเส้นทางนี้ ผู้ใช้ระบบสามารถทำการค้นหาสถานที่สำคัญๆ ในเขตกรุงเทพมหานคร หรือทำการค้นหาเส้นทางที่สั้นที่สุดในการเดินทางได้โดยการระบุสถานที่ต้นทางและสถานที่ปลายทางระบบจะนำข้อมูลของสถานที่ต้นทางและปลายทางมาทำการวิเคราะห์ โดยใช้เทคนิคการหาเส้นทางที่สั้นที่สุด (Shortest path first) เข้ามาคำนวณหาเส้นทางแล้วแสดงผลผ่านแผนที่ Google Maps API ซึ่งเป็นระบบการให้บริการของ Google โดยจะแสดงผลบนหน้าเว็บเพจ นอกจากนี้ระบบยังถูกออกแบบให้รองรับข้อมูลสภาพจราจร ซึ่งในอนาคตหากสามารถเชื่อมโยงให้เข้ากับระบบตรวจสอบสภาพจราจรของกรมทางหลวง ก็จะสามารถแสดงข้อมูลสภาพการจราจรเพื่อช่วยในเรื่องการตัดสินใจเลือกใช้เส้นทางการเดินทางอีกด้วย

<b>Thesis Title</b>	Route Navigation Service System for Bangkok Metropolitan Area		
<b>Student</b>	Miss Sasipattra	Srikrit	ID.47010821
	Mr. Amnart	Hangprasitwong	ID.47010990
<b>Advisor</b>	Asst.Prof.Mayuree Lertwatechakul		
<b>Graduate Level</b>	Bachelor Degree of Information Engineering		
<b>Department</b>	Information Engineering		
<b>Academic year</b>	2007		

## ABSTRACT

Presently, there are many communication in Bangkok all over land, water and airplane. But most favorite communication in Bangkok is land transportation with both personal car or public transportation. The number of car is increased because it is more convenient than public transportation. For this reason, the traffic jams will be happened in rush hour every day. So we cannot estimate time or route before journey and usually reach to destination too late. In this thesis describes the designing of Route Navigation Service System for Bangkok Metropolitan Area by using Bangkok road map database from Network Technology Laboratory (NTL) under NECTEC. In this system, user can search important places on Bangkok, search shortest route from specific source place to the destination. The system analyze data by using Shortest Path First algorithm to compute route. This route is displayed by Web application by Google Maps API that is service provided by Google. Furthermore, this system is designed to support traffic information. In future, if someone can link the system to Real-time traffic monitoring system of Highways Department, users could view traffic information through Webpage and determine route.

## กิตติกรรมประกาศ

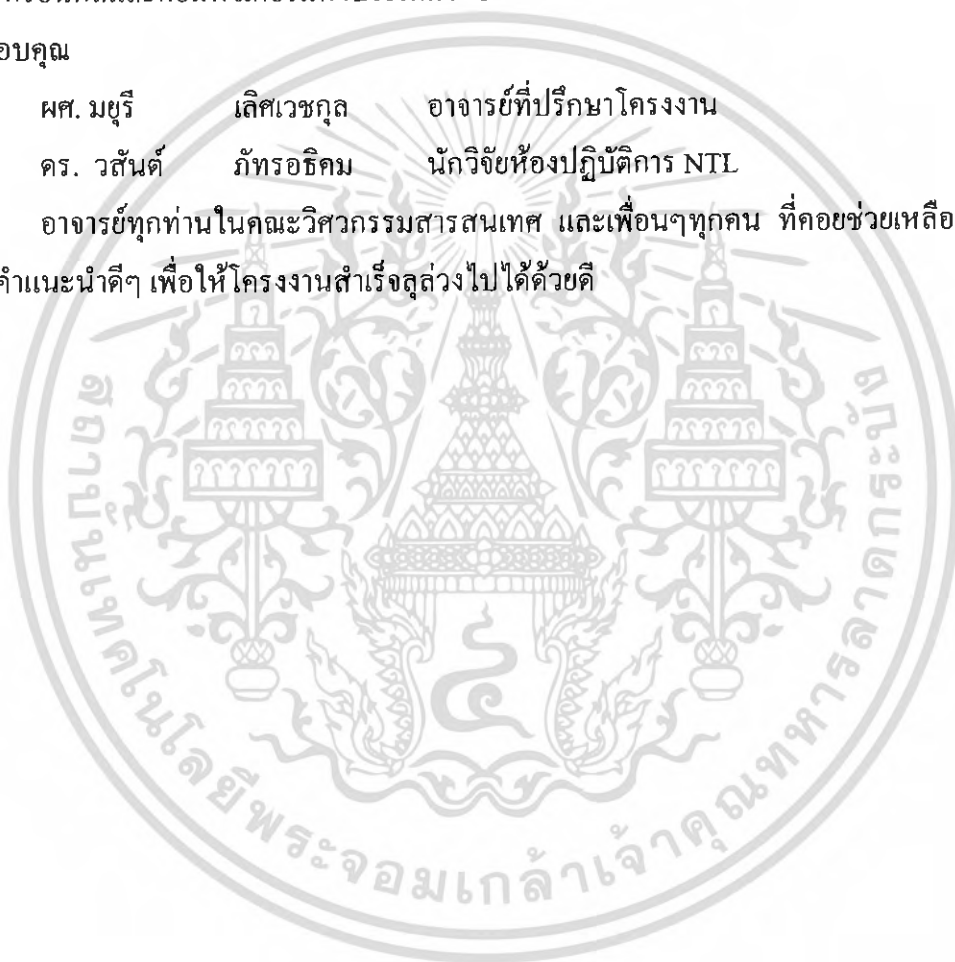
โครงการนี้เป็นโครงการร่วมกันระหว่างคณะวิศวกรรมสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง และห้องปฏิบัติการเทคโนโลยีเครือข่าย (NTL) ภายใต้ศูนย์วิจัยอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งประเทศไทย

ขอขอบคุณ

ผศ. มยุรี เลิศเวชกุล อาจารย์ที่ปรึกษาโครงการ

ดร. วสันต์ ภัทรธิตคม นักวิจัยห้องปฏิบัติการ NTL

อาจารย์ทุกท่านในคณะวิศวกรรมสารสนเทศ และเพื่อนๆทุกคน ที่คอยช่วยเหลือให้คำติชมหรือคำแนะนำดีๆ เพื่อให้โครงการสำเร็จลุล่วงไปได้ด้วยดี



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
<b>บทที่ 1 บทนำ</b>	<b>1</b>
1.1 บทนำ	1
1.2 วัตถุประสงค์	2
1.3 ขอบเขตของโครงการ	2
1.4 สถาปัตยกรรมของระบบ	3
1.5 ขั้นตอนการดำเนินโครงการ	5
<b>บทที่ 2 ทฤษฎีที่เกี่ยวข้อง</b>	<b>6</b>
2.1 ความรู้ทั่วไปเกี่ยวกับระบบพิกัดจุด	6
2.2 เทคนิคการหาเส้นทางที่สั้นที่สุด ( Shortest-Route Problem)	8
2.3 อัลกอริทึมแบบ Dijkstra's algorithms	15
2.4 ความรู้ทั่วไปเกี่ยวกับฐานข้อมูลใน PostgreSQL	18
2.5 ความรู้ทั่วไปเกี่ยวกับ PostGIS	19
2.6 เทคโนโลยี Google Maps API	20
2.7 Ajax Technology	20
<b>บทที่ 3 การออกแบบโครงการ</b>	<b>22</b>
3.1 โครงสร้างตารางฐานข้อมูล (Data Dictionary)	22
3.2 การออกแบบระบบ	30
3.2.1 Use case Diagram	30
3.2.2 Activity Diagram	31
3.2.3 System flowchart	35
<b>บทที่ 4 ผลการทดลอง</b>	<b>40</b>
4.1 ระบบค้นหาสถานที่สำคัญในกรุงเทพมหานคร	40
4.2 ระบบแนะนำเส้นทางที่ใช้ระยะทางสั้นที่สุด	43
4.3 ระบบแสดงสภาพจราจร	47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

หน้า

บทที่ 5	สรุปการพัฒนาโครงการ	50
5.1	บทสรุป	50
5.2	ปัญหาที่ยังไม่สามารถแก้ไขได้	51
5.3	แนวทางในการพัฒนาต่อ	51
บรรณานุกรม		
ภาคผนวก		



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป

หน้า

รูปที่ 1 สถาปัตยกรรมของระบบ	3
รูปที่ 2 แสดงการแบ่งกริดโซนระบบพิกัดกริด UTM	8
รูปที่ 3 แสดงความสัมพันธ์ของอาร์ค	8
รูปที่ 4 แสดงให้เห็นกราฟแบบมีทิศทางที่มี 4 เวอร์ทีซ	9
รูปที่ 5 แสดงกราฟที่กำหนดความสัมพันธ์ของอาร์ค	9
รูปที่ 6 แสดงเมตริกซ์ประชิดของรูปที่ 5	10
รูปที่ 7 แสดงกราฟที่มีทิศทางที่มีค่าบนอาร์ค	12
รูปที่ 8 ข้อสมมติฐานของการหาระยะทางที่สั้นที่สุดไปยัง w	13
รูปที่ 9 แสดงการเดินทางที่สั้นที่สุดที่เป็นไปได้	14
รูปที่ 10 แสดงการสร้างกราฟจากโหนด	16
รูปที่ 11 แสดงวิธีการคำนวณอัลกอริทึม Dijkstra	17
รูปที่ 12 แสดงเส้นทางจากถนนหมายเลข 0 ไปยังทุกๆถนน	18
รูปที่ 13 ตัวอย่างตาราง mainroad	23
รูปที่ 14 ตัวอย่างตาราง educated	23
รูปที่ 15 ตัวอย่างตาราง hospital	24
รูปที่ 16 ตัวอย่างตาราง hotel	25
รูปที่ 17 ตัวอย่างตาราง police	25
รูปที่ 18 ตัวอย่างตาราง temple	26
รูปที่ 19 ตัวอย่างตาราง govern	27
รูปที่ 20 ตัวอย่างตาราง mainroad_in	28
รูปที่ 21 ตัวอย่างตาราง mainroad_out	29
รูปที่ 22 แสดง Use case diagram ของระบบ	30
รูปที่ 23 แสดง Activity diagram ของการค้นหาสถานที่	31
รูปที่ 24 แสดง Activity diagram ของการคำนวณเส้นทางที่ใช้ระยะทางสั้นที่สุด	32
รูปที่ 25 แสดง Activity diagram ของการดูแผนที่	33
รูปที่ 26 แสดง Activity diagram ของการแสดงผลภาพจราจร	34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

หน้า

รูปที่ 27 แสดง System flowchart ของการทำงานของหน้าเว็บเพจ	35
รูปที่ 28 แสดง System flowchart ของการทำงานของระบบค้นหาสถานที่	36
รูปที่ 29 แสดง System flowchart ของการทำงานของระบบคำนวณเส้นทางที่สั้นที่สุด โดยใช้อัลกอริทึม Dijkstra	37
รูปที่ 30 แสดง System flowchart ของการทำงานของระบบแสดงสภาพจราจร	39
รูปที่ 31 แสดงการเลือกประเภทของสถานที่	41
รูปที่ 32 แสดงการเลือกสถานที่ที่ต้องการค้นหา	41
รูปที่ 33 แสดงการคลิกปุ่มค้นหาสถานที่	42
รูปที่ 34 แสดงตำแหน่งและรายละเอียดของข้อมูลรวมทั้งเก็บบันทึกข้อมูลการค้นหาสถานที่	42
รูปที่ 35 แสดงการเลือกประเภทของสถานที่ต้นทาง	43
รูปที่ 36 แสดงการเลือกสถานที่ต้นทาง	44
รูปที่ 37 แสดงการเลือกประเภทของสถานที่ปลายทาง	44
รูปที่ 38 แสดงการเลือกสถานที่ปลายทาง	45
รูปที่ 39 แสดงการคลิกปุ่ม ค้นหาเส้นทาง	45
รูปที่ 40 แสดงเส้นทางบน Google Maps API และแสดงข้อมูลเส้นทางบนแถบด้านข้าง	46
รูปที่ 41 แสดงการขยายดูสถานที่ต้นทางและปลายทาง	46
รูปที่ 42 แสดงการคลิกปุ่มสภาพการจราจรและเลือกเขตที่ต้องการดูสภาพการจราจร	47
รูปที่ 43 แสดงการคลิกปุ่ม แสดงสภาพการจราจร	48
รูปที่ 44 แสดงข้อมูลสภาพการจราจรบนแผนที่ Google Maps API	48
รูปที่ 45 แสดงข้อมูลสภาพการจราจรของแต่ละช่วงถนนพร้อมคำแนะนำ	49
รูปที่ 46 แสดงการดูแผนที่ Google Maps API	49

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญตาราง

	หน้า
ตารางที่ 1 แสดงการคำนวณของ Dijkstra ของรูปที่ 7	12
ตารางที่ 2 แสดงข้อมูลของตาราง mainroad	22
ตารางที่ 3 แสดงข้อมูลของตาราง educated	23
ตารางที่ 4 แสดงข้อมูลของตาราง hospital	24
ตารางที่ 5 แสดงข้อมูลของตาราง hotel	24
ตารางที่ 6 แสดงข้อมูลของตาราง police	25
ตารางที่ 7 แสดงข้อมูลของตาราง temple	26
ตารางที่ 8 แสดงข้อมูลของตาราง govern	27
ตารางที่ 9 แสดงข้อมูลของตาราง mainroad_in	28
ตารางที่ 10 แสดงข้อมูลของตาราง mainroad_out	29

# บทที่ 1

## บทนำ

### 1.1 บทนำ

ในปัจจุบันการเดินทางโดยเฉพาะอย่างยิ่งการคมนาคมทางบก มีบทบาทสำคัญกับชีวิตของประชากรในเขตกรุงเทพมหานคร ไม่ว่าจะเป็นการใช้บริการรถบริการขนส่งมวลชนหรือรถเมย์ การใช้บริการรถแท็กซี่หรือรถยนต์ส่วนตัวในการเดินทาง เพราะประชากรที่อาศัยในกรุงเทพมหานครมีความจำเป็นที่จะต้องเดินทางอย่างเร่งรีบให้ทันกับเวลา ไม่ว่าจะเป็นการเร่งรีบเดินทางเพื่อให้ทันเวลาเข้าทำงาน การเดินทางไปทำธุระสำคัญต่างๆ เพื่อที่จะได้มีเวลาส่วนตัวเพิ่มมากขึ้นด้วยสาเหตุนี้ทำให้เกิดปัญหาการจราจรบนท้องถนน และที่สำคัญประชากรส่วนใหญ่ต้องเร่งรีบในช่วงเวลาที่ใกล้เคียงกัน ไม่ว่าจะเป็นเวลาช่วงเช้าในการเข้าทำงาน หรือเวลาช่วงเย็นในการกลับบ้าน ทำให้รถยนต์ที่เดินทางบนท้องถนนมีจำนวนมากขึ้น โดยเฉพาะรถยนต์ส่วนบุคคลที่ปัจจุบันยังมีปริมาณมากขึ้นเนื่องจากปัจจัยต่างๆ ไม่ว่าจะเป็นความต้องการความสะดวกสบายในการเดินทางที่ไม่ต้องทนกับอากาศร้อน ความแออัดที่ต้องเผชิญในรถบริการขนส่งมวลชน สามารถเดินทางตามเส้นทางที่ตนเองต้องการ การแสดงฐานะหน้าที่การงานของตน และโปรโมชันของบริษัทผลิตภัณฑ์ต่างๆ ที่สามารถนำรถยนต์ออกมาใช้งานได้ก่อนที่จะชำระเงินจนครบ ซึ่งทำให้เกิดปัญหาสภาพการจราจรไม่คล่องตัว และด้วยพฤติกรรมของประชากรในกรุงเทพมหานครที่จะเดินทางในเส้นทางที่ตนเองใช้บริการอยู่เป็นประจำ เพราะกลัวว่าเส้นทางอื่นจะไม่สามารถเดินทางไปยังจุดหมายปลายทางของตนเองได้ ซึ่งทำให้ผลของพฤติกรรมเหล่านี้ก่อให้เกิดปัญหาสภาพการจราจรติดขัดในบางช่วง โดยบางช่วงของเส้นทางจราจรที่ไม่เป็นที่รู้จักเกิดสภาพคล่องเนื่องด้วยไม่มีการใช้บริการในเส้นทางนี้ ทำให้ประเทศชาติต้องสูญเสียพลังงานที่สำคัญอย่างน้ำมันไปเป็นจำนวนมากอย่างสูญเปล่า โดยปัจจุบันประเทศของเรขาดดุลการค้ากับประเทศต่างชาติเป็นจำนวนเงินมหาศาล อีกทั้งก่อให้เกิดปัญหาสภาพแวดล้อมไม่ว่าจะเป็น ปัญหามลพิษทางอากาศเนื่องจากการเผาไหม้เชื้อเพลิงบนท้องถนนอย่างสูญเปล่า ปัญหามลภาวะทางเสียงที่เกิดจากรถยนต์ และยานพาหนะบนท้องถนนหรือปัญหาสุขภาพของคนที่ใช้รถใช้ถนน

ในปัจจุบันปัญหาเหล่านี้ได้มีบุคคล องค์กร รัฐบาลได้สังเกตเห็นปัญหาต่างๆ ที่ได้กล่าวมาแล้วในขั้นต้น จึงพยายามหาวิธีทางในการแก้ไขปัญหา ไม่ว่าจะเป็นการรณรงค์การใช้พลังงานทดแทน เช่น การใช้แก๊สโซฮอล์ การใช้พลังงานทดแทนจากก๊าซธรรมชาติ (NGV) แทนการใช้ น้ำมัน การรณรงค์เดินทางเส้นทางเดียวกันให้ไปด้วยกัน การเพิ่มสายเส้นทางเดินทางของรถบริการขนส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มวลชน การก่อสร้างเพื่อเพิ่มเส้นทางการเดินทางของรถไฟฟ้ามหานครและรถไฟฟ้า ซึ่งทางผู้จัดทำได้สังเกตเห็นและพยายามช่วยลดปัญหาที่เกิดขึ้น โดยการพัฒนาระบบแนะนำเส้นทางในกรุงเทพมหานครในรูปแบบของหน้าเว็บเพจ ที่เหมาะสมและใช้งานง่ายเพื่อช่วยในการวางแผนการเดินทางแก่ผู้ใช้งานพาหนะบนท้องถนน

## 1.2 วัตถุประสงค์

- 1) เพื่อค้นหาสถานที่ในเขตกรุงเทพมหานคร
- 2) เพื่อแนะนำเส้นทางการเดินทางในเขตกรุงเทพมหานครที่ใช้ระยะทางสั้นที่สุดโดยพิจารณาการเดินทางระหว่างจุดเริ่มต้น และจุดปลายทาง
- 3) มีการนำเทคนิคการหาระยะทางสั้นที่สุด (Shortest path first) มาประยุกต์ใช้กับโครงข่ายของเส้นทางการจราจรในกรุงเทพมหานคร
- 4) แสดงสภาพจราจรทั้งขาเข้าขาออกบนพื้นที่ตัวอย่าง
- 5) เพื่อให้ผู้ใช้รถยนต์บนท้องถนนได้ใช้บริการเส้นทางบนท้องถนนและทรัพยากรน้ำมันได้อย่างมีประสิทธิภาพ
- 6) เพื่อลดปัญหามลพิษทางอากาศและปัญหาต่างๆที่มีผลกระทบต่อสภาพแวดล้อมและประชากรที่อาศัยอยู่ในกรุงเทพมหานคร
- 7) สามารถนำไปใช้ในการพัฒนาระบบสอบถามเส้นทางในอนาคตทั้งในภาครัฐและภาคเอกชน

## 1.3 ขอบเขตของโครงการ

ในระบบค้นหาสถานที่สำคัญในกรุงเทพมหานคร ได้ทำการกำหนดขอบเขตของปัญหาไว้ว่าจะพิจารณาสถานที่ที่มีอยู่ในฐานข้อมูลและในแผนที่ Google Maps API เท่านั้น

ในระบบแนะนำเส้นทางที่สั้นที่สุดที่ใช้ในการเดินทางระหว่างจุดสถานที่เริ่มต้น และจุดสถานที่ปลายทาง โดยในที่นี้เราจะพิจารณาเพียงระยะทางที่สั้นที่สุดในการเดินทาง ซึ่งได้กำหนดขอบเขตของปัญหาไว้ดังนี้

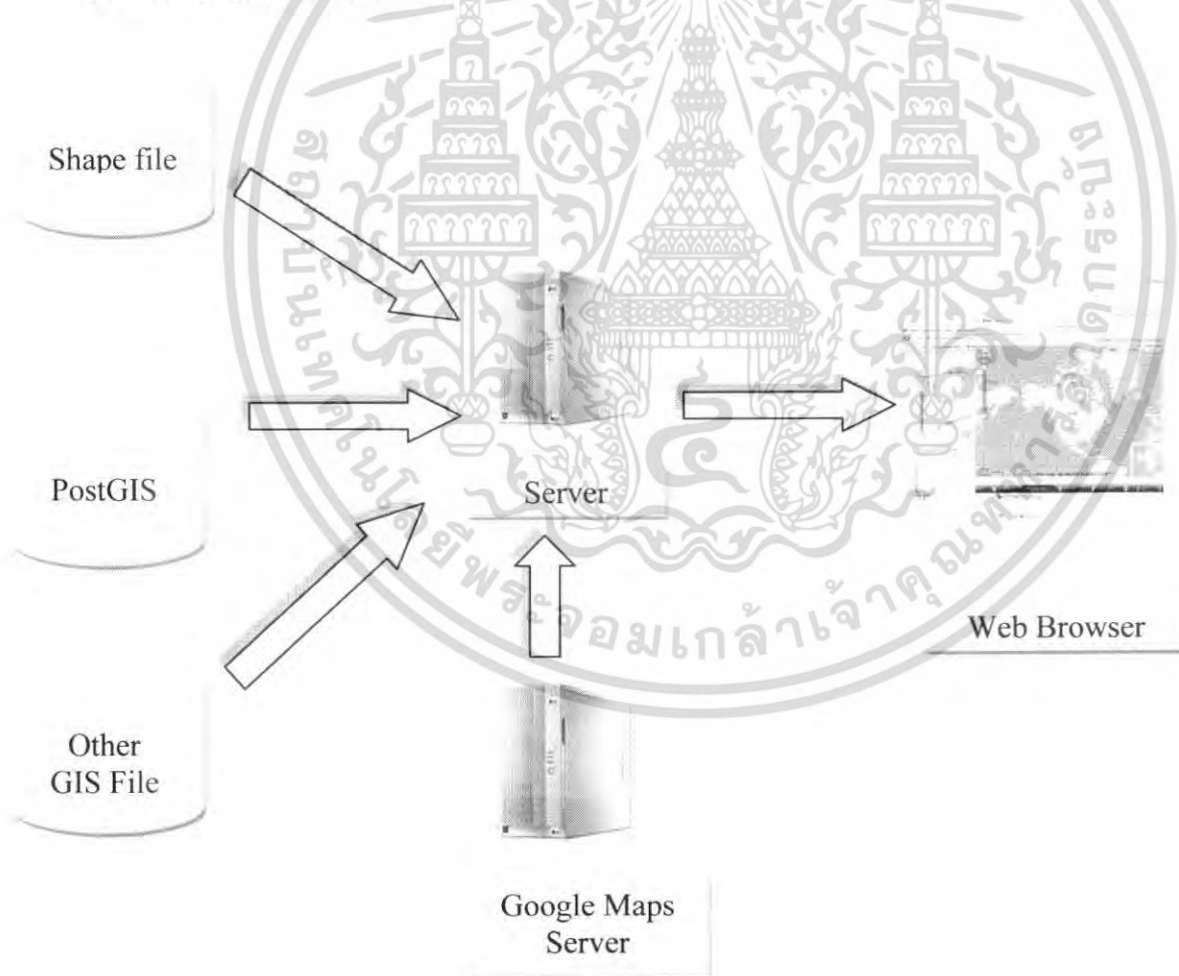
- 1) การเดินทางที่กล่าวถึงในที่นี้จะพิจารณาเพียงเส้นทางการจราจรโดยการใช้รถยนต์เท่านั้น
- 2) สถานที่อ้างอิงจะพิจารณาระยะห่างระหว่างสถานที่และถนนใหญ่เท่านั้น ไม่ได้พิจารณาระยะทางที่เดินทางเข้าไปยังในซอยที่สถานที่อ้างอิงตั้งอยู่จริง
- 3) ทิศทางการเดินทางและเงื่อนไขสภาพการจราจรยังไม่ได้นำมาพิจารณา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) เส้นทางที่ใช้ในการพิจารณา จะพิจารณาจากเส้นทางจริงที่มีอยู่ในฐานข้อมูล ส่วนในระบบแสดงสภาพจราจรนั้น จะแสดงสภาพจราจรบนเขตตัวอย่าง 7 เขต ได้แก่ เขตคลองเตย, เขตดินแดง, เขตปทุมวัน, เขตพญาไท, เขตราชเทวี, เขตวัฒนา, เขตห้วยขวาง โดยสมมติข้อมูลสภาพการจราจรทั้งขาเข้าและขาออกเพื่อนำมาแสดงบนแผนที่ Google Maps API โดยกำหนดให้

ความเร็วช่วง 30 km/hr ถึง 60 km/hr	มีสภาพการจราจรเป็น	สีแดง
ความเร็วช่วง 61 km/hr ถึง 90 km/hr	มีสภาพการจราจรเป็น	สีเหลือง
ความเร็วช่วง 90 km/hr ขึ้นไป	มีสภาพการจราจรเป็น	สีเขียว

#### 1.4 สถาปัตยกรรมของระบบ



รูปที่ 1 สถาปัตยกรรมของระบบ

เอกสารนี้เป็นเอกสารที่ซ่อนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่สมควรให้นำไปใช้ประโยชน์ด้านการค้า  
จากแผนภาพระบบจะมีขั้นตอนการทำงาน โดยแบ่งออกเป็น 2 ส่วนใหญ่ๆ ดังนี้  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 1.4.1 การทำงานในส่วนทางฝั่งผู้ใช้ระบบ

ฝั่งผู้ใช้ระบบ จะใช้โปรแกรมเว็บเบราว์เซอร์ (Web Browser) ซึ่งจะร้องขอข้อมูลจากเซิร์ฟเวอร์ด้วยโปรโตคอล HTTP โดยสามารถใช้เว็บเบราว์เซอร์ มาตรฐานทั่วไป โดยไม่ต้องติดตั้งซอฟต์แวร์ใดๆเพิ่มเติม เพื่อแสดงข้อมูลเส้นทางการจราจรที่สั้นที่สุดที่ตัวโปรแกรมแนะนำ แล้วทำการเชื่อมต่อกับเว็บเซิร์ฟเวอร์เพื่อร้องขอข้อมูลเส้นทางการจราจรด้วยโปรโตคอล HTTP อีกที

#### 1.4.2 การทำงานในส่วนทางฝั่งเซิร์ฟเวอร์ (Server)

สามารถแบ่งการทำงานได้ดังนี้

##### 1. ระบบค้นหาสถานที่สำคัญในกรุงเทพมหานคร

- 1) ผู้ใช้ระบบป้อนสถานที่ที่ต้องการค้นหา
- 2) ระบบนำค่าที่ผู้ใช้ป้อนไปค้นหาพิกัดจุดของสถานที่ในฐานข้อมูล
- 3) เมื่อได้พิกัดจุดแล้ว นำไปแสดงจุดในแผนที่ Google Maps API
- 4) แสดงผลที่ได้ผ่านหน้าเว็บเพจ

##### 2. ระบบแนะนำเส้นทางที่มีระยะทางสั้นที่สุด

- 1) ก่อนเริ่มการทำงาน ระบบจะทำการเรียกข้อมูลจากฐานข้อมูล PostgreSQL เพื่อทำการประมวลผลสร้างกราฟจากข้อมูลที่มีอยู่ โดยจะนำข้อมูลส่วนของหมายเลขของถนนสายต่างๆ ถนนถัดไปที่ถนนเส้นกำลังพิจารณาอยู่ และระยะทางของถนนแต่ละเส้น
- 2) เมื่อผู้ใช้บริการกรอกข้อมูลสถานที่จุดเริ่มต้นและสถานที่ปลายทางเข้ามาในระบบ ระบบจะทำการประมวลผลเพื่อหาระยะทางที่สั้นที่สุดโดยที่พิจารณาจากระยะทาง โดยใช้ อัลกอริทึม Dijkstra
- 3) เมื่อโปรแกรมทำการประมวลผลเสร็จเรียบร้อยแล้ว แสดงเส้นทางบนแผนที่ Google Maps API

##### 3. ระบบแสดงสภาพจราจร

- 1) ผู้ใช้คลิกปุ่ม สภาพการจราจร และเลือกเขตที่ต้องการดูสภาพการจราจร
- 2) ระบบจะแสดงสภาพจราจรบนเขตที่เลือก ทั้งขาเข้าและขาออก บนแผนที่ Google Maps API

## 1.5 ขั้นตอนการดำเนินโครงการ

- |            |    |  |
|------------|----|--|
| ขั้นตอนที่ | 1  | ศึกษาปัญหาระบบงาน กำหนดขอบเขตและจุดประสงค์ต่างๆ  |
| ขั้นตอนที่ | 2  | ศึกษาทฤษฎีที่เกี่ยวข้อง รวมทั้งศึกษาค้นคว้าจากงานวิจัย, ปัญหาพิเศษ และบทความทางวิชาการที่มีเนื้อหาเกี่ยวข้องกัน                                      |
| ขั้นตอนที่ | 3  | ศึกษาข้อมูลจากฐานข้อมูล PostgreSQL ที่ได้จากห้องปฏิบัติการระบบเครือข่าย (NLT) ภายใต้ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งประเทศไทย (NECTEC) |
| ขั้นตอนที่ | 4  | ออกแบบระบบงาน  |
| ขั้นตอนที่ | 5  | ออกแบบอัลกอริทึม shortest path first โดยใช้อัลกอริทึมแบบ Dijkstra  |
| ขั้นตอนที่ | 6  | เพิ่มข้อมูลลงฐานข้อมูล   |
| ขั้นตอนที่ | 7  | ออกแบบหน้าเว็บเพื่อร่วมกับเทคโนโลยี Google Maps API  |
| ขั้นตอนที่ | 8  | พัฒนาระบบ  |
| ขั้นตอนที่ | 9  | ตรวจสอบระบบงาน   |
| ขั้นตอนที่ | 10 | ทดสอบระบบงาน   |
| ขั้นตอนที่ | 11 | จัดทำเอกสาร  |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

### ทฤษฎีที่เกี่ยวข้อง

#### 2.1 ความรู้ทั่วไปเกี่ยวกับระบบพิกัดจุด

##### 2.2.1 ระบบพิกัดจุด (Coordinate System)

เป็นระบบที่สร้างขึ้นสำหรับใช้อ้างอิงในการกำหนดตำแหน่ง หรือบอกตำแหน่งพื้นโลกจากแผนที่ที่มีลักษณะเป็นตารางโครงข่ายที่เกิดจากตัดกันของเส้น ตรงสองชุดที่ถูกระบุให้วางตัวในแนวเหนือ-ใต้ และแนวตะวันออก-ตะวันตก ตามแนวของจุดศูนย์กลางกำเนิด (Origin) ที่กำหนดขึ้น ค่าพิกัดที่ใช้อ้างอิงในการบอกตำแหน่งต่างๆ จะใช้ค่าของหน่วยที่นับออกจากจุดศูนย์กลางกำเนิดเป็นระยะเชิงมุม (Degree) หรือเป็นระยะทาง (Distance) ไปทางเหนือหรือใต้และตะวันออกหรือตะวันตกตามตำแหน่งของตำบลที่ต้องการหาค่าพิกัดที่กำหนดตำแหน่งต่างๆ จะถูกเรียกอ้างอิงเป็นตัวเลขในแนวตั้งและแนวนอนตามหน่วยวัดระยะ ใช้วัดสำหรับระบบพิกัดที่ใช้อ้างอิงกำหนดตำแหน่งบนแผนที่ ที่นิยมใช้กับแผนที่ในปัจจุบัน มีอยู่ด้วยกัน 2 ระบบ คือ

- 1) ระบบพิกัดภูมิศาสตร์ (Geographic Coordinate System)
- 2) ระบบพิกัดกริดแบบ UTM (Universal Transverse Mercator co-ordinate System)

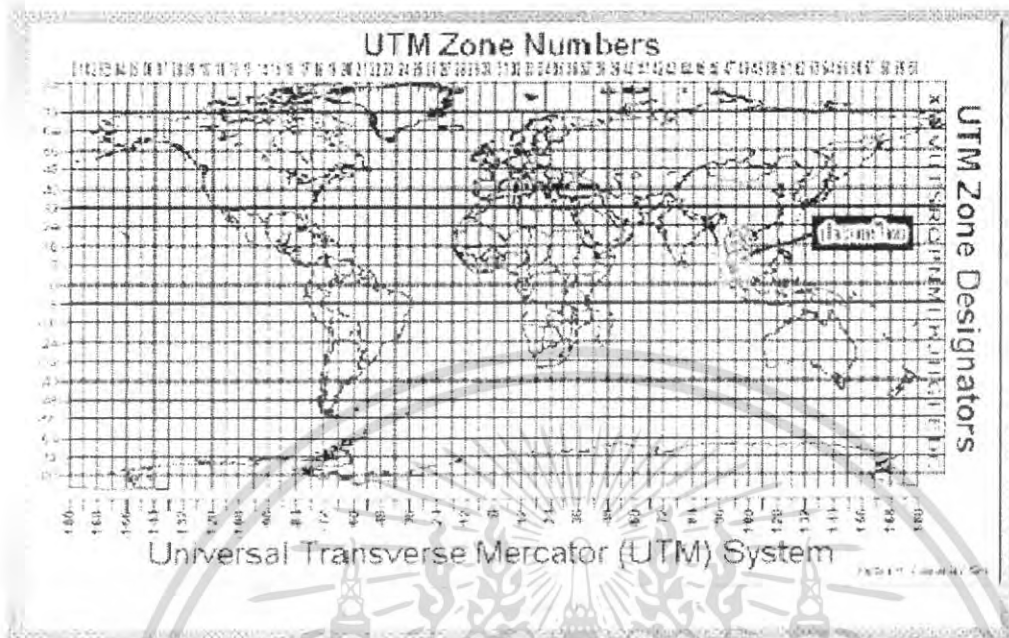
##### 1) ระบบพิกัดภูมิศาสตร์ (Geographic Coordinate System)

เป็นระบบพิกัดที่กำหนดตำแหน่งต่างๆบนพื้นโลก ด้วยวิธีการอ้างอิงบอกตำแหน่งเป็นค่าระยะเชิงมุมของละติจูด (Latitude) และ ลองจิจูด (Longitude) ตามระยะเชิงมุมที่ห่างจากศูนย์กลางกำเนิด (Origin) ของละติจูดและลองจิจูด ที่กำหนดขึ้นสำหรับศูนย์กลางกำเนิดของละติจูด (Origin of Latitude) นั้นกำหนดขึ้นจากแนวระดับที่ตัดผ่านศูนย์กลางของโลกและตั้งฉากกับแกนหมุน เรียกแนวระนาบศูนย์กลางกำเนิดนั้นว่า เส้นศูนย์สูตร (Equator) ซึ่งแบ่งโลกออกเป็นซีกโลกเหนือและซีกโลกใต้ ฉะนั้นค่าระยะเชิงมุมของละติจูด จะเป็นค่าเชิงมุมที่เกิดจากมุมที่ศูนย์กลางของโลก กับแนวระดับฐานกำเนิดมุมที่เส้นศูนย์สูตรที่วัดค่าของมุมออกไปทั้งซีกโลกเหนือและซีกโลกใต้ ค่าของมุมจะสิ้นสุดที่ขั้วโลกเหนือและขั้วโลกใต้ มีค่าเชิงมุม 90 องศาพอดี ดังนั้นการใช้ค่าระยะเชิงมุมของละติจูดอ้างอิงบอกตำแหน่งต่างๆ นอกจากจะกำหนดเรียกค่าวัดเป็น องศาลิปดา และฟิลิปดา แล้วจะบอกซีกโลกเหนือหรือใต้กำกับด้วยเสมอ

## 2) ระบบพิกัดกริดแบบ UTM (Universal Transverse Mercator co-ordinate System)

พิกัดกริด UTM (Universal Transverse Mercator) เป็นระบบตารางกริดที่ใช้ช่วยในการกำหนดตำแหน่งและใช้อ้างอิงในการบอกตำแหน่งที่นิยมใช้กับแผนที่ในกิจการทหารของประเทศต่าง ๆ เกือบทั่วโลกในปัจจุบัน เพราะเป็นระบบตารางกริดที่มีขนาดรูปร่างเท่ากันทุกตารางและมีวิธีการกำหนดบอกค่าพิกัดที่ง่ายและถูกต้องเป็นระบบกริดที่นำเอาเส้นโครงแผนที่แบบ Universal Transverse Mercator Projection ของ Gauss -Krueger มาใช้ดัดแปลงการถ่ายทอดรายละเอียดของพื้นผิวโลกให้รูปทรงระบอบ Mercator Projection อยู่ในตำแหน่ง Mercator Projection (แกนของรูปทรงระบอบจะทับกับแนวเส้นอิกเวเตอร์ และตั้งฉากกับแนวแกนของขั้วโลก) ประเทศไทยเราได้นำเอาเส้นโครงแผนที่แบบ UTM นี้มาใช้ในการทำแผนที่ เป็นชุด L 7017 ที่ใช้ในปัจจุบันแผนที่ระบบพิกัดกริดที่ใช้เส้นโครงแผนที่แบบ UTM เป็นระบบเส้นโครงชนิดหนึ่งที่ใช้ผิวรูปทรงระบอบเป็นผิวแสดงเส้นเมริเดียน (หรือเส้นลองจิจูด) และเส้นละติจูดของโลก โดยใช้ทรงระบอบตัดโลกระหว่างละติจูด 84 องศาเหนือ และ 80 องศาใต้ในลักษณะแกนรูปทรงระบอบแล้วทำมุมกับแกนโลก 90 องศารอบโลก แบ่งออกเป็น 60 โซนๆ ละ 6 องศา

โซนที่ 1 อยู่ระหว่าง 180 องศา กับ 174 องศาตะวันตก และมีละติจูด 177 องศาตะวันตก เป็นเมริเดียนย่านกลาง (Central Meridian) มีเลขกำกับแต่ละโซนจาก 1 ถึง 60 โดย นับจากซ้ายไปทางขวาระหว่างละติจูด 84 องศาเหนือ 80 องศาใต้ แบ่งออกเป็น 2 ช่อง ช่องละ 8 องศา ยกเว้นช่องสุดท้าย เป็น 12 องศา โดยเริ่มนับตั้งแต่ละติจูด 80 องศาใต้ ขึ้นไป ทางเหนือ ให้ช่องแรกเป็นอักษร C และช่องสุดท้ายเป็นอักษร X (ยกเว้น I และ O) จากการแบ่งตามที่กล่าวแล้วจะเห็นพื้นที่ในเขตละติจูด 180 องศาตะวันตก ถึง 180 องศาตะวันออก และละติจูด 80 องศาใต้ ถึง 84 องศาเหนือ จะถูกแบ่งออกเป็นรูปสี่เหลี่ยมผืนผ้า 1,200 รูป แต่ละรูปมีขนาดกว้างยาว 6 องศา x 8 องศา จำนวน 1,140 รูป และกว้างยาว 6 องศา x 12 องศา จำนวน 60 รูป รูปสี่เหลี่ยมนี้เรียกว่า Grid Zone Designation (GZD) การเรียกชื่อ Grid Zone Designation ประเทศไทยมีพื้นที่อยู่ ระหว่างละติจูด 5 องศา 30 ลิปดา เหนือ ถึง 20 องศา 30 ลิปดา เหนือ และละติจูดประมาณ 97 องศา 30 ลิปดา ตะวันออก ถึง 105 องศา 30 ลิปดา ตะวันออก ดังนั้น ประเทศไทยจึงตั้งอยู่ใน GZD 47N 47P 47Q 48N 48P และ 48 Q [1]



รูปที่ 2 แสดงการแบ่งกริด โซนระบบพิกัดกริด UTM

## 2.2 เทคนิคการหาเส้นทางที่สั้นที่สุด ( Shortest-Route Problem)

กราฟแบบมีทิศทาง  $G$  ใดๆจะประกอบไปด้วยเซตของเวอร์ทีซ (vertices)  $v$  และเซตของความสัมพันธ์ระหว่างเวอร์ทีซหรือที่เรียกว่าอาร์ค (arcs)  $E$  เวอร์ทีซอาจเรียกได้อีกอย่างว่า โหนด (node) และอาร์คเรียกได้อีกอย่างว่า directed lines อาร์คจะเป็นคู่ลำดับของเวอร์ทีซ  $(v, w)$  โดย  $v$  เรียกว่า Tail และ  $w$  เรียกว่า Head ของอาร์ค  $(v,w)$  มักแทนอยู่ในรูปของ  $v \rightarrow w$  และเขียนได้ดังนี้

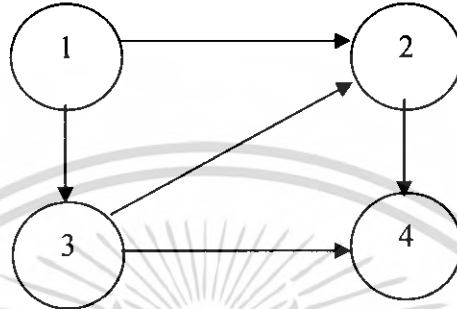


รูปที่ 3 แสดงความสัมพันธ์ของอาร์ค

สังเกตว่า หัวลูกศรจะอยู่ที่เวอร์ทีซที่เรียกว่า Head และหางลูกศรจะอยู่ที่เวอร์ทีซที่เรียกว่า Tail เราจะกล่าวว่อาร์ค  $v \rightarrow w$  เป็นความสัมพันธ์จาก  $v$  ไปยัง  $w$  เป็นการประชิด ( adjacent ) ไปยัง  $v$  เวอร์ทีซของกราฟแบบมีทิศทางนี้สามารถใช้แทนวัตถุใดๆ และอาร์คใช้แทนความสัมพันธ์ระหว่างวัตถุ เช่น เวอร์ทีซอาจแทนชื่อเมืองต่างๆ ในขณะที่อาร์คอาจแทนเส้นทางการบินระหว่างเมืองหนึ่งสู่อีกเมืองหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

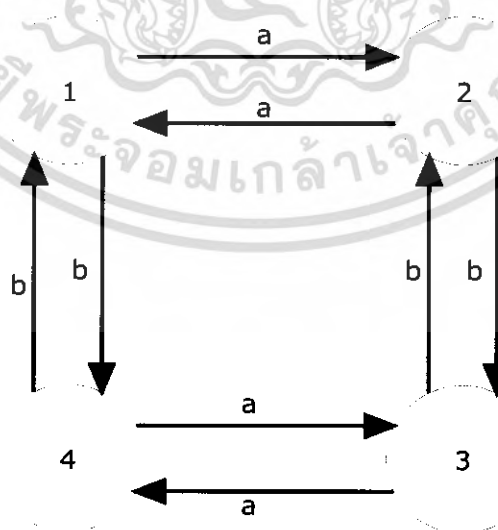
เส้นทางของกราฟแบบมีทิศทาง หมายถึง ลำดับของเวอร์ทีซ  $v_1, v_2, \dots, v_n$  เช่น  $v_1 \rightarrow v_2, v_2 \rightarrow v_3, \dots, v_{n-1} \rightarrow v_n$  โดยที่เส้นทางนี้เริ่มจากเวอร์ทีซ  $v_1$  ไปยังเวอร์ทีซ  $v_n$  ผ่านเวอร์ทีซ  $v_2, v_3, \dots, v_{n-1}$  และสิ้นสุดที่  $v_n$  Length หรือความยาวของเส้นทางคือจำนวนของอาร์คบนเส้นทาง ในกรณีนี้คือ  $n-1$  ดังตัวอย่างเช่น



รูปที่ 4 แสดงให้เห็นกราฟแบบมีทิศทางที่มี 4 เวอร์ทีซ

จากรูป ลำดับของเวอร์ทีซ 1,2,4 คือเส้นทางที่มีความยาวเท่ากับ 2 จากเวอร์ทีซ 1 ไปยังเวอร์ทีซ 4 และ เส้นทาง 3,2,4,3 จะเป็นวนรอบที่มีความยาวเท่ากับ 3

ในการประยุกต์ใช้งานหลายๆอย่าง เรามักจะกำหนดเครื่องหมายไว้ในแต่ละอาร์ค โดยที่เครื่องหมายนี้อาจเป็นชื่อราคาหรือมูลค่าของข้อมูลต่างๆที่กำหนด ดังเช่นตัวอย่างต่อไปนี้ซึ่งแสดงให้เห็นกราฟที่กำหนดเครื่องหมายไว้ในแต่ละอาร์ค ซึ่งกำหนดโดยตัวอักษรที่แสดงถึงการเปลี่ยนแปลงจากเวอร์ทีซหนึ่งไปยังอีกเวอร์ทีซหนึ่ง



รูปที่ 5 แสดงกราฟที่กำหนดความสัมพันธ์ของอาร์ค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.1.1 การแทนรูปแบบโครงสร้างกราฟแบบมีทิศทาง

เราสามารถใช้โครงสร้างข้อมูลหลายชนิดมาใช้แทนกราฟแบบมีทิศทางนี้ แต่โครงสร้างข้อมูลที่เหมาะสมนี้ จะขึ้นอยู่กับกรกระทำที่จะนำมาประยุกต์ใช้กับเวอร์ทีซและอาร์คเหล่านั้นวิธีการที่นิยมสำหรับกราฟแบบมีทิศทาง  $G=(V,E)$  วิธีหนึ่งก็คือ ใช้ เมตริกซ์การประชิด (adjacency matrix )

สมมติว่า  $v=\{1,2,\dots,n\}$  เมตริกซ์การประชิด สำหรับ  $G$  คือ  $n \times n$  เมตริกซ์  $A$  ของบูลีน (Booleans) โดยที่  $A[i, j]$  เป็นจริงถ้าหากมีอาร์คจากเวอร์ทีซ  $i$  ไปยัง  $j$  โดยทั่วไปเรามักจะแทนเมตริกซ์การประชิดด้วย 1 สำหรับกรณีที่เป็นจริง และ 0 ในกรณีที่เป็นเท็จ

จากตัวอย่างแสดงให้เห็นเมตริกซ์การประชิดของกราฟแบบมีทิศทางในที่นี้ เครื่องหมายกำกับจะเป็นตัวอักษร และช่องว่างจะหมายถึงการที่ไม่มีอาร์คอยู่

	1	2	3	4
1		a		b
2	a		b	
3		b		a
4	b		a	

รูปที่ 6 แสดงเมตริกซ์ประชิดของรูปที่ 5

### 2.1.2 ปัญหาทางเดินที่สั้นที่สุดแบบทางเดียว

ปัญหาการหาเส้นทางเดินในกราฟแบบมีทิศทาง สมมติว่ามีกราฟแบบมีทิศทาง  $G=(V,E)$  อันหนึ่ง ซึ่งแต่ละอาร์คไม่มีค่าลบ ในเวอร์ทีซใดเวอร์ทีซหนึ่ง จะถูกกำหนดให้เป็นจุดเริ่มต้น (source) ปัญหาก็คือการพิจารณาค่าคอส (cost) ซึ่งอาจจะเป็น ค่าใช้จ่าย ระยะทาง หรือเวลา ฯลฯ ของระยะทางที่น้อยที่สุด จากจุดเริ่มต้นไปยังเวอร์ทีซอื่นๆ ทุกๆ เวอร์ทีซใน  $V$  โดยที่ความยาวของเส้นทาง (length of path ) ก็คือผลรวมของค่าคอส (cost) บนเส้นทางนั้น

$G$  อาจจะแทนแผนที่การเดินทาง ซึ่งในเวอร์ทีซต่างๆ จะแทนสถานที่สำคัญต่างๆ บนสายการเดินทาง และแต่ละอาร์ค  $v \rightarrow w$  คือ เส้นทางการเดินทางระหว่างสถานที่  $v$  ไปยังสถานที่  $w$  เครื่องหมายบนอาร์ค  $v \rightarrow w$  ก็คือ เวลาหรือระยะทางในการเดินทางจาก  $v$  ไปยัง  $w$  การแก้ปัญหานี้ก็คือ การหาเวลาหรือระยะทางในการเดินทางที่น้อยที่สุดจากสถานที่ที่กำหนดไปยังทุกสถานที่ในแผนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

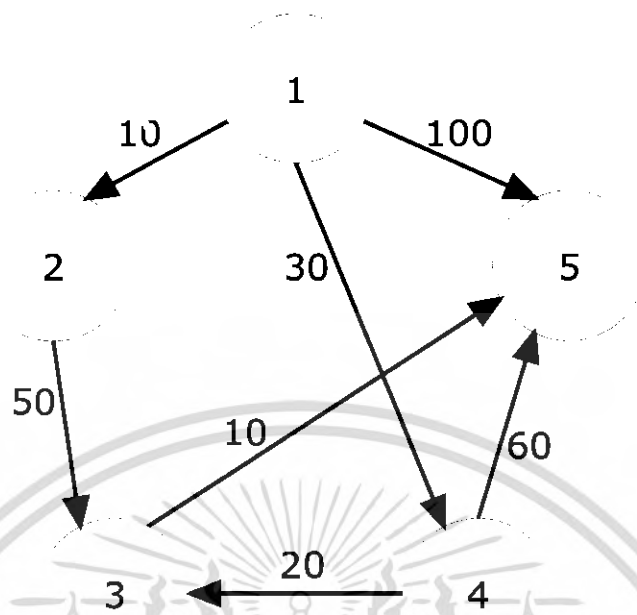
ในการแก้ปัญหาให้นำมาประยุกต์ใช้กับอัลกอริทึมที่เรียกว่า “ Dijkstra ‘s algorithm ” อัลกอริทึมนี้ทำงานโดยการพยายามรักษาเซต S ของเวอร์ทีซต่างๆ ซึ่งเราทราบระยะทางสั้นที่สุดจากจุดเริ่มต้นไปยังเวอร์ทีซเหล่านั้น สมมติในตอนเริ่มต้นเซต S มีเพียงเวอร์ทีซของจุดเริ่มต้นเท่านั้น ต่อมาในแต่ละขั้นตอน เราก็จะบวกเวอร์ทีซ v ซึ่งมีระยะห่างจากจุดเริ่มต้นน้อยที่สุดเท่าที่จะน้อยได้ลงไป และในแต่ละขั้นของอัลกอริทึม เราจะใช้อาร์เรย์ D ในการเก็บค่าระยะทางที่สั้นที่สุดที่หาได้

```

S :={1};
For I :=2 to n do
    D[i] :=C[1,i]; {initialize D}
For I :=1 to n-1 do
Begin
    Choose a vertex w in V-s such that
        D[w] is a minimum;
    Add w to s
    For each vertex v in V-S do
        D[v] :=min(D[v],D[w]+C[w,v])
End

```

จากอัลกอริทึมนี้ สมมติว่ามีกราฟ  $G = (V,E)$  ซึ่ง  $V = \{1,2,\dots,n\}$  และเวอร์ทีซ 1 คือจุดเริ่มต้น C คือ อาร์เรย์ของระยะทาง (cost) โดยที่  $C[i, j]$  คือ ระยะทาง (cost) ของการเดินทางจากเวอร์ทีซ i ไปยัง j บนอาร์ค  $i \rightarrow j$  ถ้าไม่มีอาร์คจากเวอร์ทีซ i ไปยังเวอร์ทีซ j ดังนั้นเราจะอนุมานค่าของ  $C[i, j]$  เป็น  $\infty$  และในทุกๆขั้นตอน  $D[i]$  จะเก็บความยาวเส้นทางที่น้อยที่สุดจากจุดเริ่มต้นไปยังเวอร์ทีซ i ใดๆแสดงตัวอย่างดังรูปต่อไปนี้



รูปที่ 7 แสดงกราฟที่มีทิศทางที่มีค่าน้ำหนัก

สมมติว่าใช้ Dijkstra's algorithm กับรูปตัวอย่าง ในคอนเริ่มต้น  $S = \{1\}$ ,  $D[2] = 10$ ,  $D[3] = \infty$ ,  $D[4] = 30$  และ  $D[5] = 100$  ในรอบแรกของ for-loop ในบรรทัดที่ (4) ถึง (8) จะได้ว่า  $w=2$  คือ เวก์ที่ซที่ถูกเลือกซึ่งมีค่า  $D$  ต่ำที่สุด ดังนั้นเราจะเซตให้  $D[3] = \min(\infty, 10+50) = 60$ ,  $D[4]$  และ  $D[5]$  ไม่มีการเปลี่ยนแปลง เนื่องจากการเข้าหาเวก์ที่ซ 4 และ 5 โดยตรงจากเวก์ที่ซ 1 จะสั้นกว่า การเข้าหาผ่านทางเวก์ที่ซ 2 ถ้าดับของค่า  $D$  หลังจากการวนรอบของ for-loop จะได้ดังตาราง ต่อไปนี้

Loop	S	w	D[2]	D[3]	D[4]	D[5]
0	{1}	-	10{1->2}	$\infty$ {-}	30{1->4}	100{1->5}
1	{1}, {1->2}	2	10{1->2}	60{1->2->3}	30{1->4}	100{1->5}
2	{1}, {1->2}, {1->4}	4	10{1->2}	50{1->4->3}	30{1->4}	90{1->4->5}
3	{1}, {1->2}, {1->4},	3	10{1->2}	50{1->4->3}	30{1->4}	60{1->4->3->5}
4	{1->4->3}	5	10{1->2}	50{1->4->3}	30{1->4}	60{1->4->3->5}
	{1}, {1->2}, {1->4},					
	{1->4->3}, {1->4->3->5}					

ตารางที่ 1 แสดงการคำนวณของ Dijkstra ของรูปที่ 7

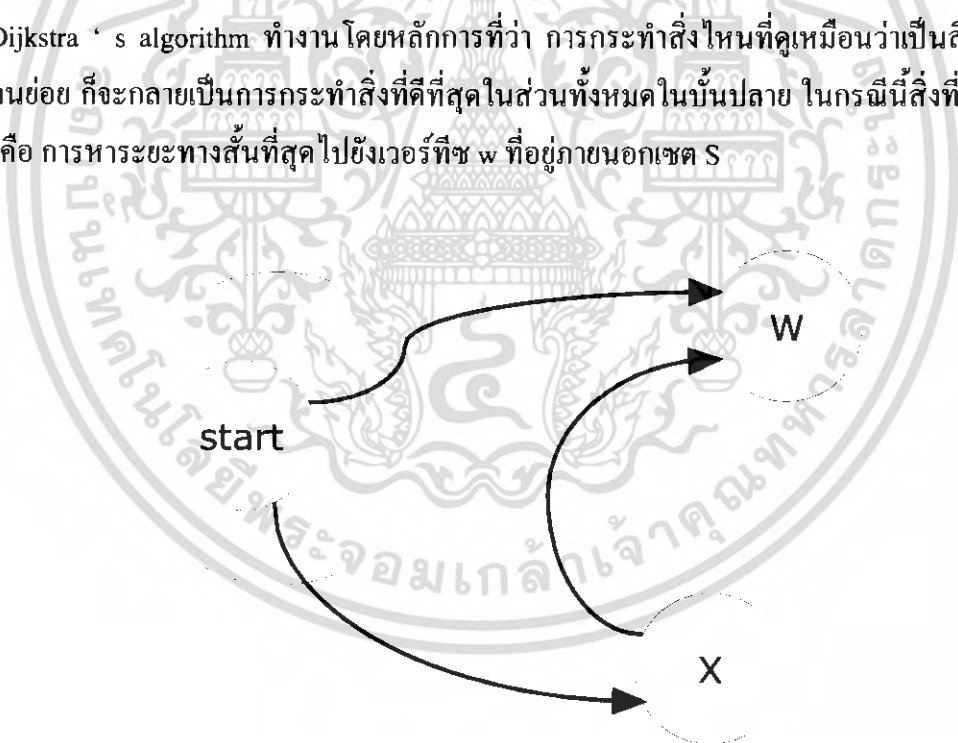
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในตอนเริ่มต้นจะเห็นได้ว่า  $D[3] = 60$  อันเนื่องจากในรอบที่ 1 ที่เริ่มวนรอบนั้น ค่าของเซต  $S$  จะมีเวอร์ทีซที่ถูกนำเข้ามาพิจารณาเพียง 2 เวอร์ทีซคือ  $S = \{1, 2\}$  ดังนั้นระยะทางที่สั้นที่สุดจากเวอร์ทีซ 1 ไปยัง 3 ที่สั้นที่สุดเป็นไปได้โดยผ่านทางเวอร์ทีซ 2 ก็คือ 6 นั่นคือ  $D[3] = 60$  และ  $D[5]$  ก็ยังคงเป็น 100 อยู่ เพราะไม่อาจหาทางที่สั้นกว่านี้ได้ถ้าให้ผ่านเวอร์ทีซ 2 เพียงตัวเดียว

ต่อมา จะเริ่มผนวกเอาเวอร์ทีซที่สั้นที่สุดจากเวอร์ทีซเริ่มต้นเข้ามาอีก ทำให้  $S = \{1, 2, 4\}$  อันเนื่องมาจาก เวอร์ทีซ 4 เป็นเวอร์ทีซที่สั้นที่สุดจากกลุ่มเวอร์ทีซ  $\{1, 2\}$  เพราะว่าจากเวอร์ทีซ 2 ไป 3 มีค่าเท่ากับ 50 ส่วนจากเวอร์ทีซ 1 ไป 5 มีค่าเท่ากับ 100 ดังนั้นเวอร์ทีซที่สั้นที่สุดจากกลุ่ม  $S = \{1, 2\}$  ก็คือ 4

จากการวนซ้ำครั้งที่ 2 เมื่อ  $S = \{1, 2, 4\}$  จะเห็นว่าค่า  $D[5]$  สามารถลดลงได้เป็น 90 อันเนื่องมาจากค่าที่สั้นที่สุดจากเวอร์ทีซ 1 ไปยังเวอร์ทีซ 5 โดยสามารถผ่านทางเวอร์ทีซ 1, 2 และ 4 นั่นก็คือ  $1 \rightarrow 4 \rightarrow 5$  ทำให้ได้ระยะทางรวมเป็น 90 เหตุการณ์ จะเป็นเช่นนี้เรื่อยๆ จนกว่าจะหมดเวอร์ทีซที่จะนำมารวมในเซต  $S$

Dijkstra's algorithm ทำงานโดยหลักการที่ว่า การกระทำสิ่งไหนที่ดูเหมือนว่าเป็นสิ่งดีที่สุดในส่วนย่อย ก็จะกลายเป็นการกระทำสิ่งที่ดีที่สุดในส่วนทั้งหมดในบั้นปลาย ในกรณีนี้สิ่งที่ดีที่สุดที่จะทำคือ การหาระยะทางสั้นที่สุดไปยังเวอร์ทีซ  $w$  ที่อยู่นอกเซต  $S$

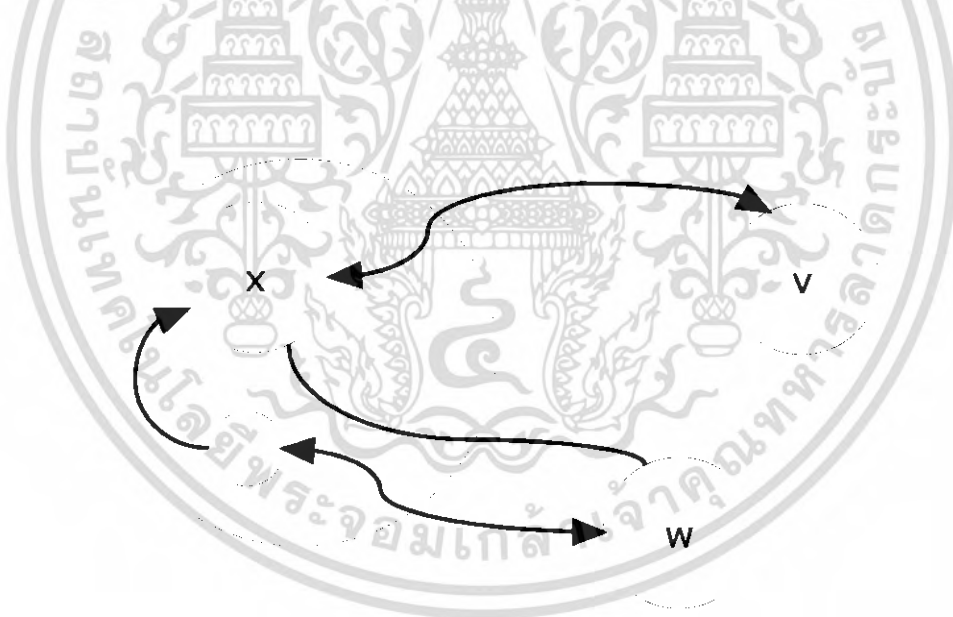


รูปที่ 8 ข้อสมมติฐานของการหาระยะทางที่สั้นที่สุดไปยัง  $w$

จากรูปจะเห็นได้ว่า เราสามารถเดินทางไปยังเวอร์ทีซ  $w$  ได้ 2 ทาง ทางแรกเราจะไปโดยวิ่งจากจุดเริ่มต้น ไปยังเวอร์ทีซ  $w$  เลย และทางที่สองเราจะเดินทางไปยังเวอร์ทีซ  $x$  ก่อน จากนั้นอาจจะเดินทางเข้าหรือออกในเซต  $S$  หลายครั้งก่อนที่จะไปถึง  $w$

จะเห็นว่า เป็นการยากที่จะหาระยะทางที่สั้นที่สุดโดยวิ่งจากจุดเริ่มต้นไปยังเวอร์ทีซ  $x$  ก่อน (ซึ่งอยู่นอกเซต  $S$  เหมือนกันกับ  $w$ ) แต่ถ้าเส้นทางนี้สั้นกว่าเดิน ไปยังเวอร์ทีซ  $w$  เราจะเลือก  $x$  แทน เพราะว่า  $D[x]$  น้อยกว่า  $D[w]$

เพื่อเป็นการพิสูจน์ว่าอัลกอริทึมทำงานได้จริง สมมติว่า  $D[v]$  เป็นระยะทางสั้นที่สุดไปยังเวอร์ทีซ  $v$  ที่ช่วงเวลาใดๆ ปัญหาของเรื่องนี้ก็คือ เมื่อเราเพิ่มเวอร์ทีซ  $w$  เข้าไปยังเซต  $S$  ที่บรรทัด (6), (7) และ (8) จะทำการปรับค่า  $D$  เพื่อจัดบันทึกความเป็นไปได้ที่ว่า มันอาจจะมีเส้นทางที่สั้นกว่าที่เดินทางไปยัง  $v$  โดยผ่าน  $w$  ถ้าเส้นทางนั้นเดินทางจาก old  $S$  ไปยัง  $w$  และพุ่งตรงไปยัง  $v$  ดังนั้น ค่าคอส (cost) ของมัน (คือค่า  $D[w] + C[w, v]$ ) จะถูกนำมาเปรียบเทียบกับ  $D[v]$  ที่บรรทัด (8) และค่า  $D[v]$  นี้จะถูกลดลงถ้าค่าใหม่นี้น้อยกว่าค่าเดิม เส้นทางสั้นที่สุดนี้อาจเป็นไปได้อีกทางหนึ่งได้แสดงดังรูป



รูปที่ 9 แสดงการเดินทางที่สั้นที่สุดที่เป็นไปได้

เส้นทางที่ว่านี้จะเดินทางผ่านไปยัง  $w$  และย้อนมายัง old  $S$  เข้าไปยังสมาชิกบางตัวของ old  $S$  คือเวอร์ทีซ  $x$  หลังจากนั้นก็ไปยังเวอร์ทีซ  $v$

แต่จริงๆ แล้วเส้นทางนี้จะไม่สามารถใช้ได้ เนื่องจาก  $x$  ถูกนำเข้าสู่เซต  $S$  ก่อน  $w$  และเส้นทางที่สั้นที่สุดจากจุดเริ่มต้นมายัง  $x$  จะผ่านแค่ old  $S$  เท่านั้น ดังนั้นเส้นทางไปยัง  $x$  ผ่านทาง  $w$  ตามที่เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แสดงในรูป จึงไม่อาจสั้นกว่าเส้นทางจาก  $x$  ผ่านในเซต  $S$  โดยตรง ผลก็คือความยาวเส้นทางในรูป จากจุดเริ่มต้นไปยัง  $w$ ,  $x$  และ  $v$  ไม่อาจสั้นกว่าค่าเดิมของ  $D[v]$  เนื่องจาก  $D[v]$  ไม่อาจมีค่ามากกว่า ความยาวของทางเดินที่สั้นที่สุด ที่เดินทางไปยัง  $x$  ผ่านเซต  $S$  และตรงไปยัง  $v$  ดังนั้น  $D[v]$  ไม่อาจ ถูกลดลงได้

### 2.3 อัลกอริทึมแบบ Dijkstra's algorithms

โครงสร้างในการทำงานของระบบแนะนำเส้นทางจราจรนี้ ได้มีส่วนที่ช่วยเหลือผู้ใช้งานในการหาเส้นทางในการเดินทางโดยใช้รถยนต์ส่วนตัวที่สั้นที่สุดที่ใช้ในการเดินทาง โดยใช้อัลกอริทึม Dijkstra ซึ่งเป็นอัลกอริทึม ที่ใช้ในการค้นหาเส้นทางที่สั้นที่สุดโดยนำค่าของระยะทางของแต่ละช่วงถนนสายหลักมาคิด ในการทำงานของส่วนระบบการค้นหาเส้นทางนี้ ได้นำข้อมูลบางส่วน บางคอลัมน์จากฐานข้อมูลแต่ละช่วงถนนสายหลัก โดยข้อมูลส่วนคอลัมน์ของช่วงถนนสายหลักที่นำมาใช้งานจะประกอบไปด้วย

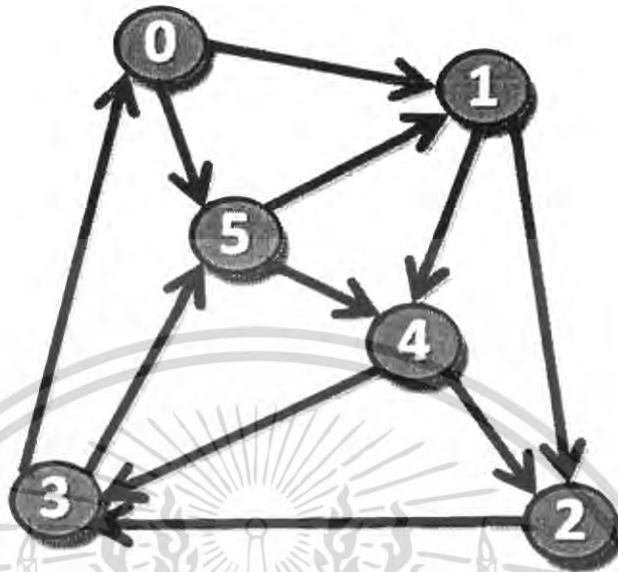
- หมายเลขของถนนสายหลัก
- ชื่อของถนนสายหลักแต่ละเส้น
- โหนดเริ่มต้นและโหนดสุดท้ายของถนนสายหลักที่เป็นจุดเชื่อมต่อของถนนแต่ละเส้น
- จุดละติจูด ลองจิจูด ของจุดเริ่มต้นและจุดสุดท้ายของถนนสายหลักแต่ละเส้น
- ระยะทางของถนนสายหลักในแต่ละช่วง

ข้อมูลที่สำคัญอีกส่วนหนึ่งในการจะนำมาใช้วาดเส้นบนแผนที่ คือ ข้อมูลที่เก็บในรูปแบบของ บิตข้อมูลที่ได้รับมาจากเครื่องจีพีเอส (GPS) ซึ่งส่วนนี้จะนำมาใช้เพียงในการวาดรูปเท่านั้น ไม่ได้ใช้ในการคำนวณเส้นทางที่สั้นที่สุด

เมื่อเรานำข้อมูลของแต่ละถนนสายหลักมาแล้ว จะนำข้อมูลต่างๆมาทำการสร้างเป็นกราฟ โดยเราจะทำการดึงข้อมูลของแต่ละเส้นมาเช็คว่า มีถนนสายไหนบ้างที่เชื่อมต่อกับถนนที่เรากำลังสนใจอยู่ โดยการตรวจสอบจุดโหนดที่มีการเชื่อมต่อที่ได้อธิบายไว้ข้างต้น ถ้าพบว่ามีถนนที่มีการเชื่อมต่อเกิดขึ้น ตัวโปรแกรมจะทำการสร้างโหนดขึ้นมา โดยโหนดที่เราพิจารณาอยู่นั้นจะกำหนดให้เป็นโหนดแม่หรือโหนดเริ่มต้น และถนนที่มีการเชื่อมต่อกับโหนดแม่จะให้เป็นโหนดลูกหรือโหนดปลายทาง และทำการสร้างความสัมพันธ์ของทั้งสองโหนดที่เรียกว่าค่านหรืออาร์ค ซึ่งเป็นค่าของระยะทาง โดยค่านี่จะเป็นค่านำหนักที่เราจะใช้ในการตัดสินใจเลือกเส้นทางที่สั้นที่สุด โดยเราจะทำเช่นนี้ไปเรื่อยๆจนครบทุกถนน ซึ่งแสดงดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

0-1	.41
1-2	.51
2-3	.50
4-3	.36
3-5	.38
3-0	.45
0-5	.29
5-4	.21
1-4	.32
4-2	.32
5-1	.29

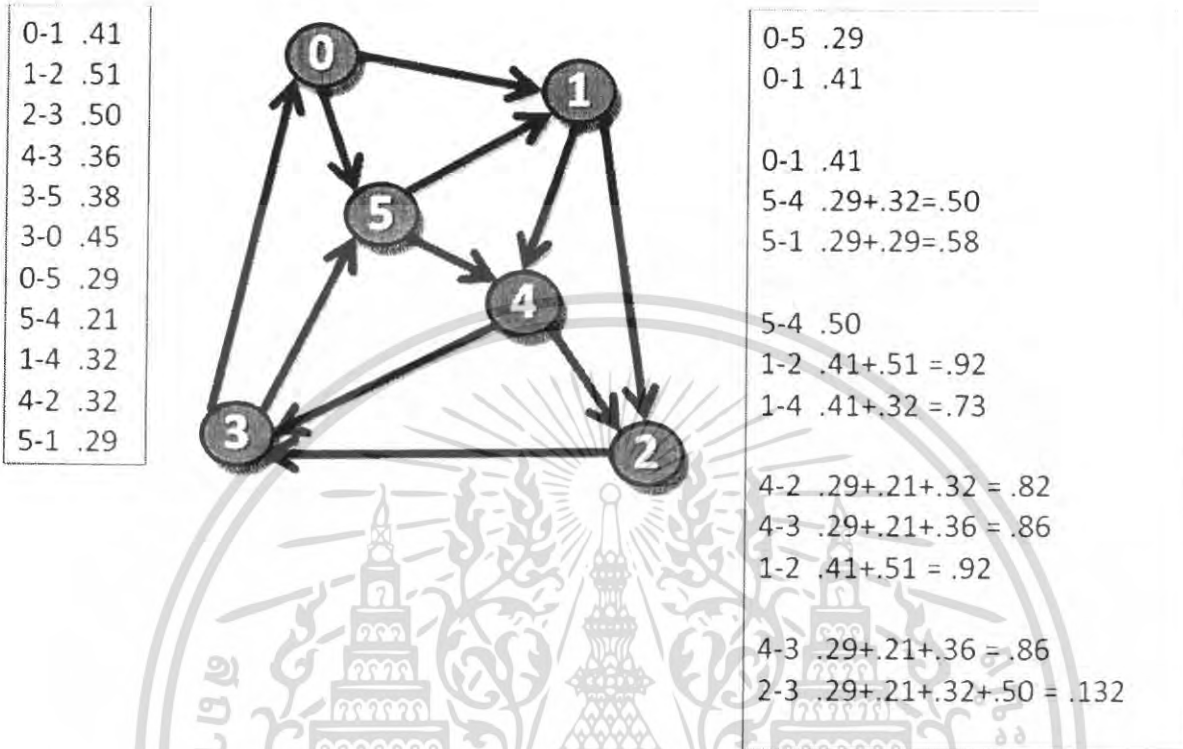


รูปที่ 10 แสดงการสร้างกราฟจากโหนด

จากรูปจะเห็นได้ว่า ถนนหมายเลขที่ 1 เชื่อมต่อกับถนนหลายเลข 2 โดยมีระยะทางจากถนนหมายเลข 1 ไปยังถนนหมายเลข 2 มีระยะทาง 510 เมตร ถนนหมายเลขที่ 2 เชื่อมต่อกับถนนหมายเลข 3 โดยมีระยะทางเชื่อมต่อกันเป็นระยะทาง 500 เมตร เป็นต้น

ในการสร้างกราฟขึ้นมา เราได้นำเทคนิคการใช้ อาร์เรย์ลิงค์ลิสต์ เข้ามาช่วย โดยในที่นี้เราจะนำเอา ลิงค์ลิสต์ มาประยุกต์การใช้งานกับอาร์เรย์เพราะ ลิงค์ลิสต์ มีคุณสมบัติในการจัดเก็บชุดข้อมูลเชื่อมโยงต่อเนื่องกันไปตามลำดับ ซึ่งอาจอยู่ในลักษณะแบบเชิงเส้นตรง หรือ ไม่เป็นเส้นตรงก็ได้ ซึ่งในลิสต์จะประกอบไปด้วยข้อมูลที่เรียกว่า โหนด ในหนึ่งโหนดจะประกอบด้วยส่วนของข้อมูลที่ต้องการจัดเก็บ เรียกว่า ส่วนข้อมูล และส่วนที่เป็น พอยเตอร์ที่ชี้ไปยังโหนดถัดไป หรือชี้ไปยังโหนดอื่นๆที่อยู่ในลิสต์ ซึ่งเราได้ใช้ลิงค์ลิสต์หลายๆตัวนำมาจัดเป็นอาร์เรย์ โดยหนึ่งลิงค์ลิสต์จะประกอบด้วยโหนดแม่ ซึ่งโหนดแม่ก็คือถนนเริ่มต้น โหนดลูกก็คือโหนดที่เป็นถนนที่มีการเชื่อมต่อกับถนนที่เราพิจารณา เหตุผลที่นำอาร์เรย์ลิงค์ลิสต์มาใช้งานอันเนื่องมาจากต้องการเพิ่มประสิทธิภาพในการทำงานของโปรแกรมซึ่งถ้าใช้ อาร์เรย์จำเป็นต้องมีการจองหน่วยความจำไว้ล่วงหน้า ซึ่งถ้ามากเกินไปก็อาจก่อให้เกิดการล่าช้าต่อการเรียกข้อมูล การค้นหาข้อมูลมาใช้งาน

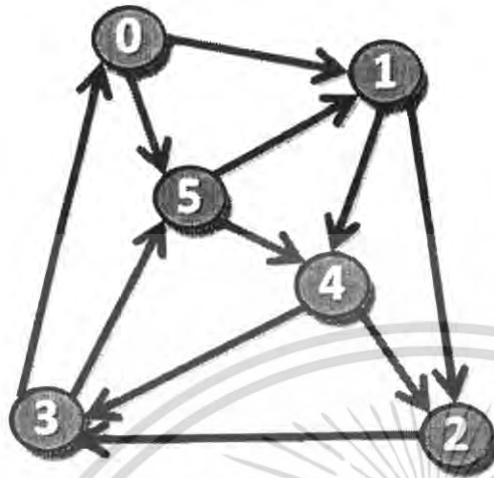
เมื่อเราทำการสร้างอาร์เรย์ลิงค์ลิสต์ของทุกถนนเสร็จแล้ว ก็จะทำการให้ผู้ใช้กรอกข้อมูลสถานที่ที่ต้องการ ซึ่งเมื่อกรอกข้อมูลเสร็จเรียบร้อยแล้ว โปรแกรมจะทำการตรวจสอบว่ามีชื่อสถานที่ที่มีอยู่ในฐานข้อมูลหรือไม่ ถ้ามีจะทำการค้นหาถนนสายหลักที่ใกล้สถานะนั้นมากที่สุด เป็นการเลื่อนจุดลงถนนที่ใกล้ที่สุดเพื่อทำการค้นหาเส้นทางโดยใช้อัลกอริทึมแบบ Dijkstra ซึ่งการทำงานจะมีลักษณะดังรูปวงไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 11 แสดงวิธีการคำนวณอัลกอริทึม Dijkstra

สมมติเราให้โหนดที่เป็นถนนสายหลักคือถนนหมายเลข 0 ได้มาจากการเลื่อนจุดจากสถานที่ลงถนนแล้ว จากนั้นโปรแกรมจะทำการคำนวณหาว่า เส้นทางจากถนนหลายเลข 0 ไปยังทุกๆ ถนนที่ใช้ค่าใช้จ่ายหรือระยะทางสั้นที่สุด คือถนนสายไหนดังรูปข้างต้น จะแสดงได้ดังรูป

0-1 .41  
 1-2 .51  
 2-3 .50  
 4-3 .36  
 3-5 .38  
 3-0 .45  
 0-5 .29  
 5-4 .21  
 1-4 .32  
 4-2 .32  
 5-1 .29



0-5 .29  
 0-1 .41

0-1 .41  
 5-4 .29+.32=.50  
 5-1 .29+.29=.58

5-4 .50  
 1-2 .41+.51 =.92  
 1-4 .41+.32 =.73

4-2 .29+.21+.32 = .82  
 4-3 .29+.21+.36 = .86  
 1-2 .41+.51 = .92  
 4-3 .29+.21+.36 = .86  
 2-3 .29+.21+.32+.50 = .132

รูปที่ 12 แสดงเส้นทางจากถนนหมายเลข 0 ไปยังทุกๆถนน

จะเห็นได้ว่าค่าการเริ่มต้นไม่ย่อยลงไปเรื่อยๆ โดยโหนดแม่ของมันคือ โหนด 0 level ถัดมาก็คือโหนดของถนนหมายเลข 5 และหมายเลข 1 และลงมาเรื่อยๆตามรูป และต่อมาเราจะทำการป้อนถนนปลายทางที่ได้จากการสแนปลงถนนสายหลัก สมมุติว่าได้ถนนหมายเลข 3 ซึ่งถนนหมายเลข 3 มีถนนหมายเลข 4 เป็นโหนดแม่ซึ่งถนนหมายเลข 4 ก็มีถนนหมายเลข 5 เป็นโหนดแม่ และถนนหมายเลข 5 ก็มีถนนหมายเลข 0 ซึ่งเป็นถนนในระดับสูงสุดซึ่งก็คือถนนต้นทางเช่นกัน ฉะนั้นเราจะได้ว่า เราต้องเดินทางจากถนน หมายเลข

0->5->4->3 ซึ่งข้อมูลของชื่อ ระยะทาง หรือข้อมูลบางอย่างที่เราต้องการแสดงผลในฐานข้อมูล ก็สามารถเรียกโดยการค้นหาข้อมูลจากหมายเลขถนนมาได้ไม่ยากนัก

## 2.4 ความรู้ทั่วไปเกี่ยวกับฐานข้อมูลใน PostgreSQL

### 2.4.1 เกี่ยวกับ PostgreSQL

เป็น Object-Relational DBMS โดยสามารถใช้รูปแบบของภาษา SQL ได้เกือบทั้งหมด และสามารถใส่ subselects , transactions , user-defined types และ functions ได้ อีกทั้งเป็นฐานข้อมูลซึ่งให้ซอร์สโค้ด (source code) ฟรี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.2.2 ลักษณะโครงสร้าง

ระบบที่ใช้ PostgreSQL จะติดตั้ง PostgreSQL ไว้ที่เครื่อง Server ซึ่งเป็นที่เก็บ database ด้วย และยังสามารถ 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* ในระบบยูนิกซ์ (UNIX) หน้าที่ของ *postgres* สามารถ สร้างชื่อ user และกำหนดสิทธิและระดับการใช้งานต่างๆ ได้ [2]

## 2.5 ความรู้ทั่วไปเกี่ยวกับ PostGIS

### 2.5.1 เกี่ยวกับ PostGIS

PostGIS เป็นสิ่งที่ถูกพัฒนาขึ้นโดยบริษัท Refractions Research ซึ่งเป็นโครงการวิจัยทางเทคโนโลยีด้านฐานข้อมูลเชิงพื้นที่ Refractions คือ ระบบสารสนเทศภูมิศาสตร์ (GIS) และ การให้คำปรึกษาฐานข้อมูลของบริษัทใน Victoria, British Columbia และ Canada ซึ่งฐานข้อมูลนี้จะมี ความพิเศษในการรวบรวมข้อมูล และพัฒนาซอฟต์แวร์ของลูกค้าโดยจะมีแผนการสนับสนุน และ พัฒนา PostGIS เพื่อให้รองรับลำดับของความสำคัญในบทบาทหน้าที่ของระบบสารสนเทศ ภูมิศาสตร์ ซึ่งจะประกอบไปด้วย OpenGIS เดิมรูปแบบ ซึ่งสิ่งที่สร้างขึ้นนี้เป็นลักษณะภูมิ ประเทศในชั้นสูง (บริเวณที่ปกคลุม , พื้นผิว , เครื่องช่วย) ที่หน้าจอของผู้ใช้จะมีเครื่องมือที่ใช้ ติดต่อสื่อสารสำหรับแสดงผลและแก้ไขข้อมูลทางสารสนเทศภูมิศาสตร์ และเครื่องมือในการเข้าถึง เว็บชั้นพื้นฐาน

PostGIS เป็นการเพิ่มเติมในส่วนของฐานข้อมูลเชิงวัตถุสัมพันธ์ (Object-relational database system) ของ PostgreSQL ให้มีการรองรับวัตถุทางระบบสารสนเทศภูมิศาสตร์ (GIS object) เข้ามา เก็บไว้ในฐานข้อมูล มีการสนับสนุน Gist index และ R-tree index ซึ่งเป็นวิธีการค้นข้อมูลแบบตัวชี้ (Indexing) ที่ใช้ในฐานข้อมูลเชิงพื้นที่สำหรับฐานข้อมูลที่มีขนาดใหญ่ ซึ่ง PostGIS เองมีการ กำหนดการใช้งานโดย OpenGIS ที่เป็นลักษณะพื้นฐานของ SQL (SFSQL)

ดังที่กล่าวมาแล้วว่า PostGIS มีการกำหนดการใช้งานที่เป็นลักษณะพื้นฐานโดย OpenGIS Consortium (OGC) ซึ่งเป็นสถาบันที่ศึกษาเพื่อการสร้างอินเทอร์เน็ตที่ทำโปรแกรมประยุกต์ ซอฟต์แวร์ ให้มีการใช้งานได้กับระบบสารสนเทศภูมิศาสตร์ (GIS)

OpenGIS เป็นสิ่งกำหนดความชัดเจนในการเข้าถึงข้อมูลทางด้านธรณีที่แตกต่างกัน และ ประมวลผลทางธรณีของแหล่งทรัพยากรในสภาพแวดล้อมที่เป็นเครือข่าย (Network) [3]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.6 เทคโนโลยี Google Maps API

Google Maps API ช่วยให้เราสามารถพัฒนาโปรแกรมเพื่อแทรก Google Maps API เข้าไปเป็นองค์ประกอบส่วนหนึ่งในเว็บเพจที่ต้องการได้โดยเขียนเป็นรหัส HTML และ Javascript ในรูปแบบที่ไม่สลับซับซ้อนนักสำหรับงานแผนที่ต่างๆ Google Maps API มีขีดความสามารถกว้างขวางเน้นในด้านการนำเสนอข้อมูลแผนที่ในลักษณะหมุดปัก (Push pin / Place marker) ซึ่งสามารถกำหนดให้แสดงข้อมูลประกอบแผนที่เมื่อผู้ใช้คลิกที่ตัว push pin /marker นั้นๆ หรือองค์แผนที่แบบเส้น (Polyline) พื้นที่ (Polygon) และภาพ (Ground overlay) บริการด้านแผนที่ของ Google นี้เริ่มต้นตั้งแต่กลางปีค.ศ. 2005 เป็นบริการฟรี จัดให้แก่ผู้ใช้ทั่วโลกโดยคาดหวังที่จะใช้การโฆษณาบนแผนที่เป็นรายได้กลับคืนแต่ในระยะแรกจะยังไม่มีโฆษณา ดังกล่าว ในการจัดบริการแผนที่นี้ ส่วนประกอบพื้นฐานสำคัญที่ดึงดูดใจให้มีผู้ใช้งานแผนที่ของกูเกิ้ล (Google) เป็นอย่างมากคือแผนที่และภาพถ่ายดาวเทียมคุณภาพดีซึ่งครอบคลุมทั่วพื้นผิวโลกในมาตราส่วนต่างๆ ตามความเหมาะสม ทำให้การพัฒนาต่อยอดจากสิ่งที่ Google จัดไว้ให้แล้วเป็นงานที่น่าสนใจ ไม่ต้องลงทุนจัดหาทรัพยากรที่หายากและราคาแพงเองมาใช้ในโครงการอย่างที่เคยเป็นในอดีต

เนื่องจากจัดทำ Google Maps API เป็นโปรแกรมรหัสเปิด (Open source program) ในภาษาจาวาสคริปต์ จึงทำให้ผู้ใช้ที่เป็นนักพัฒนาโปรแกรมสามารถเข้าไปดูรายละเอียดของรหัสโปรแกรมได้สะดวก รวมทั้งสามารถปรับเปลี่ยนแก้ไขโปรแกรมได้ ทำให้ Google Maps API มีผู้ใช้งานอย่างกว้างขวาง เหตุผลสำคัญอีก 2 อย่างที่ส่งเสริมให้มีผู้ใช้งานมากคือแผนที่และภาพถ่ายดาวเทียมคุณภาพดีที่ใช้สนับสนุนการทำแผนที่ที่มีให้ครอบคลุมพื้นที่ต่างๆ อย่างกว้างขวาง และชื่อเสียงของโปรแกรม Google Earth เสริมด้วยบริการ Google Local ที่มีมาก่อน [4]

## 2.7 AJAX Technology

AJAX โดยตัวมันเองแล้วไม่ได้เป็นเทคโนโลยีหรือภาษาโปรแกรมชนิดใหม่ แต่เป็นการรวมกลุ่มของเทคโนโลยีที่มีใช้กันอยู่แล้วดังที่กล่าวข้างต้น โดยวิวัฒนาการของ AJAX เริ่มต้นเมื่อปี ค.ศ. 2002 ไมโครซอฟท์ได้นำเสนอ XMLHttpRequest ขึ้นมาเพื่อเป็นทางเลือกในการเขียนโปรแกรมบนเว็บเพจ เพื่อใช้ติดต่อกับ เว็บเซิร์ฟเวอร์ ในการร้องขอข้อมูล ซึ่งในขณะนั้นมีแต่เพียง Internet Explorer เท่านั้นที่สามารถใช้งานได้ ต่อมาเว็บเบราว์เซอร์อื่นๆ เช่นจาก มอซิลลา (mozilla) ได้นำแนวคิดของ XMLHttpRequest ไปใส่ในเบราว์เซอร์ของตนด้วย จึงเริ่มทำให้มีการใช้อย่างกว้างขวางขึ้น จนปัจจุบันได้กลายเป็นมาตรฐานที่ทุกเว็บเบราว์เซอร์ต้องมี ต่อมาเว็บไซต์ชื่อดังอย่าง Google ได้ออกสินค้าใหม่ชื่อ Gmail ซึ่งใช้ XMLHttpRequest เป็นหัวใจหลักในการดึงข้อมูลจากเว็บเซิร์ฟเวอร์ ประกอบกับคุณ Jesse James Garrett ได้บัญญัติศัพท์คำว่า AJAX ขึ้นมาเมื่อปี ค.ศ.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2005 จึงทำให้แนวคิดและเทคนิคการพัฒนาเว็บแอปพลิเคชันด้วย AJAX เริ่มรู้จักเป็นที่กว้างขวางขึ้น จนปัจจุบันถือว่าเป็นหนึ่งในหัวใจหลักของแนวคิดเรื่อง Web 2.0 ไปด้วย Asynchronous JavaScript And XML หรือ AJAX เป็นเทคนิควิธีการพัฒนาเว็บแอปพลิเคชัน ให้มีความสามารถโต้ตอบกับผู้ใช้งานได้ดีขึ้น ปกติแล้วหน้าเว็บเพจที่มีการเปลี่ยนแปลงข้อมูลอยู่เสมอๆ จะถูกสร้างด้วยโปรแกรมที่ฝั่งเซิร์ฟเวอร์ที่เรียกว่า CGI หรือ Server Side Script ซึ่งเมื่อต้องการข้อมูลที่เปลี่ยนแปลงไป ผู้ใช้ต้องทำการร้องขอหน้าเว็บเพจนั้นใหม่ หรือที่เรียกว่า รีโหลด (reload) แต่หากเว็บเพจนั้นใช้เทคนิค AJAX แล้วเว็บเพจนั้นไม่จำเป็นต้องรีโหลดหน้าใหม่ทั้งหน้า แต่จะทำการดึงข้อมูลเฉพาะส่วนที่เปลี่ยนแปลงเพื่อการแสดงผลเท่านั้น

AJAX เป็นเทคนิควิธีการที่ใช้ความสามารถของเทคโนโลยีหลายๆ อย่างมารวมกันอันได้แก่

- Document Object Model และ จาวาสคริปต์ใช้ในการจัดการกับข้อมูลและประมวลผลข้อมูล
- XHTML และ CSS ใช้ในการแสดงผลลัพธ์ข้อมูลให้ผู้ใช้ดู
- XML และ XSLT ใช้เป็น โครงสร้างของข้อมูล และเป็นสื่อในการรับส่งข้อมูล
- XMLHttpRequest ใช้ในการร้องขอข้อมูลจาก เว็บเซิร์ฟเวอร์ และ
- JavaScript จะผูกทุกสิ่งเข้าด้วยกัน

[5]

### บทที่ 3

#### การออกแบบโครงการ

#### 3.1 โครงสร้างตารางฐานข้อมูล (Data Dictionary)

ฐานข้อมูลที่ได้จากห้องปฏิบัติการเทคโนโลยีเครือข่าย (NTL) ภายใต้ศูนย์วิจัยอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งประเทศไทย มีดังนี้

##### 3.1.1 ฐานข้อมูลของระบบค้นหาสถานที่และระบบค้นหาเส้นทาง

ชื่อตาราง                      mainroad

Field	Constraint	Type	Description
Gid	Primar key	Serial	ลำดับของข้อมูล
Fnode_	Not null	Bigint	From node
Tnode_	Not null	Bigint	To node
Length	Not null	double	ระยะทางระหว่าง node
Mainroad_	Not null	Bigint	ลำดับของถนนหลัก
Mainroad_i	Not null	Bigint	หมายเลขของถนนหลัก
Rdlnnamt	CHECK (ndims(the_geom) = 2)	Bigint	ชื่อถนน
Startx_coo	Not null	Double	จุดเริ่มต้นพิกัด x
Starty_coo	Not null	Double	จุดเริ่มต้นพิกัด y
Endx_coord	Not null	Double	จุดสิ้นสุดพิกัด x
Endy_coord	Not null	Double	จุดสิ้นสุดพิกัด y
The_geom	CHECK (geometrytype(the_geom)	Geometry	พิกัด

ตารางที่ 2 แสดงข้อมูลของตาราง mainroad

gid	name	name	length	mainroad_	mainroad_	rdname	startx_coo	starty_coo	endx_coo	endy_coo	the_geom
[PK] serial	Copy selected lines to clipboard		double precis	bigint	bigint	character var	double precis	double precis	double precis	double precis	geometry
1	20564	20565	53.53	1	12543	ถนนพหลโยธิน 2	1520263.25	1520102.375	666373.5	1520160	0105000000010
2	20583	20584	523.061	2	77454	ทางหลวงชนบท 652260.125	1515506	652353.9375	1516313.875	0105000000010	
3	20585	20586	382.506	3	77455	ทางหลวงชนบท 652419.1875	1515925.375	652354.0625	1516300.675	0105000000010	
4	20587	20588	347.603	4	77456	ทางหลวงชนบท 652399.8125	1515468	652366.125	1515560	0105000000010	
5	20586	20585	452.906	5	77457	ทางหลวงชนบท 652401.125	1515475.875	652419.1875	1515925.375	0105000000010	
6	20593	20589	325.623	6	77458	ทางหลวงชนบท 652206	1515793.75	652205.5625	1515943.125	0105000000010	
7	20596	20590	583.737	7	77459	ทางหลวงชนบท 652457.5	1515865.25	652490.0625	1516913.125	0105000000010	
8	20597	20593	1003.745	8	77464	ทางหลวงชนบท 652693.9375	1523243.375	652694.5	1524242.75	0105000000010	
9	20599	20600	76.113	9	77465	ถนนประชาปถ 661853	1517672.375	661804.75	1517725.75	0105000000010	
10	20601	20599	135.267	10	77466	ถนนประชาปถ 661701.125	1517665	661804.75	1517725.75	0105000000010	

รูปที่ 13 ตัวอย่างตาราง mainroad

ชื่อตาราง educated

Field	Constraint	Type	Description
Gid	Primary key	Serial	ลำดับของข้อมูล
Namt	CHECK (ndims(the_geom) = 2)	Character	ชื่อสถานศึกษา
Brancht	Not null	Character	สถานที่ตั้ง
Group	Not null	Character	สถานศึกษา
The_geom	CHECK (geometrytype(the_geom)	Geometry	พิกัดของสถานศึกษา

ตารางที่ 3 แสดงข้อมูลของตาราง educated

gid	namt	brancht	group	the_geom
[PK] serial	character var	character var	character var	geometry
1	โรงเรียนบ้านหนองถนนห้วยราชบุรี	สถานศึกษา	สถานศึกษา	0101000020877
2	โรงเรียนนวมินทร์ ถนนเฉลิมพระเกียรติ	สถานศึกษา	สถานศึกษา	0101000020877
3	สถาบันคริสตศาสน	สถานศึกษา	สถานศึกษา	0101000020877
4	โรงเรียนชุมชนองรี ถนนเข้าวัดแป้น	สถานศึกษา	สถานศึกษา	0101000020877
5	โรงเรียนวัดแป้น ถนนเข้าวัดแป้น	สถานศึกษา	สถานศึกษา	0101000020877
6	โรงเรียนวัดพระยา ถนนพระยาสุเรน	สถานศึกษา	สถานศึกษา	0101000020877
7	โรงเรียนบริหารธุรกิจ ซอยหมู่บ้านมโนะ	สถานศึกษา	สถานศึกษา	0101000020877
8	โรงเรียนกลางคด ซอยโรงเรียนกลาง	สถานศึกษา	สถานศึกษา	0101000020877
9	โรงเรียนวัดคูบอน ถนนเลียบคลองส	สถานศึกษา	สถานศึกษา	0101000020877
10	โรงเรียนจินดาป่า ถนนสายที่ 4	สถานศึกษา	สถานศึกษา	0101000020877
11	โรงเรียนวัดลาดป ถนนลาดปลาเค้า	สถานศึกษา	สถานศึกษา	0101000020877
12	โรงเรียนสายอักษ ซอยรามอินทรา 6	สถานศึกษา	สถานศึกษา	0101000020877
13	โรงเรียนลาดปลา ซอยลาดปลาเค้า	สถานศึกษา	สถานศึกษา	0101000020877
14	โรงเรียนประชาชา ถนนพระยาสุเรน	สถานศึกษา	สถานศึกษา	0101000020877
15	โรงเรียนนาภิยะห์ ถนนพระยาสุเรน	สถานศึกษา	สถานศึกษา	0101000020877
16	โรงเรียนสมิทธิสนธิ์ ซอยลาดปลาเค้า	สถานศึกษา	สถานศึกษา	0101000020877

รูปที่ 14 ตัวอย่างตาราง educated

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อตาราง hospital

Field	Constraint	Type	Description
Gid	Primary key	Serial	ลำดับของข้อมูล
Namt	CHECK (ndims(the_geom) = 2)	Character	ชื่อโรงพยาบาล
Brancht	Not null	Character	สถานที่ตั้ง
Group	Not null	Character	โรงพยาบาล
The_geom	CHECK (geometrytype(the_geom)	Geometry	พิกัดของโรงพยาบาล

ตารางที่ 4 แสดงข้อมูลของตาราง hospital

gid [PK] serial	namt character var	brancht character var	group character var	the_geom geometry
1	โรงพยาบาลสินเหล็กนครราชสีมา	โรงพยาบาล	โรงพยาบาล	0101000000C96
2	โรงพยาบาลเสนา ถนนเสนาเดิม	โรงพยาบาล	โรงพยาบาล	0101000000490
3	โรงพยาบาลศรีสข ถนนสุขาภิบาล 1	โรงพยาบาล	โรงพยาบาล	010100000075C
4	โรงพยาบาลพระ ร้อยเมกไม้	โรงพยาบาล	โรงพยาบาล	0101000000000
5	โรงพยาบาลนวมินทร์ ร้อยเมกบุรีรัมย์	โรงพยาบาล	โรงพยาบาล	01010000000EE
6	โรงพยาบาลบางโพ ร้อยประชาราชว	โรงพยาบาล	โรงพยาบาล	010100000005E1
7	โรงพยาบาลสยาม ถนนโชคชัย 4 (ส	โรงพยาบาล	โรงพยาบาล	0101000000F3E
8	โรงพยาบาลเปาโล ร้อยสุทธานารวิน	โรงพยาบาล	โรงพยาบาล	0101000000057
9	โรงพยาบาลสุทธานารวิน ร้อยอ	โรงพยาบาล	โรงพยาบาล	0101000000F5F
10	โรงพยาบาลรังษี ถนนสีรินธร(ราชวิ	โรงพยาบาล	โรงพยาบาล	0101000000F39

รูปที่ 15 ตัวอย่างตาราง hospital

ชื่อตาราง hotel

Field	Constraint	Type	Description
Gid	Primary key	Serial	ลำดับของข้อมูล
Namt	CHECK (ndims(the_geom) = 2)	Character	ชื่อโรงแรม
Brancht	Not null	Character	สถานที่ตั้ง
Group	Not null	Character	โรงแรม
The_geom	CHECK (geometrytype(the_geom)	Geometry	พิกัดของโรงแรม

ตารางที่ 5 แสดงข้อมูลของตาราง hotel

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

gid [PK] serial	namt character var	brancht character var	group character var	the_geom geometry
1	โรงแรมเจ้าพระยา ถนนรัชดาภิเษก		โรงแรม	0101000020877
2	โรงแรมยูเรเชีย ซอยพัฒน์นาเวศน์		โรงแรม	0101000020877
3	โรงแรมมังกรอิม ซอยวิภาวดีรังสิต		โรงแรม	0101000020877
4	โรงแรมแอมบาสเซอ ซอยประดิพัทธ์ 1		โรงแรม	0101000020877
5	โรงแรมกานต์มณี ซอยประดิพัทธ์ 1		โรงแรม	0101000020877
6	โรงแรมประจักษ์พิหาร ซอยประดิพัทธ์ 1		โรงแรม	0101000020877
7	โรงแรมลิเบอร์ตี้ ซอยประดิพัทธ์ 1		โรงแรม	0101000020877
8	โรงแรมมิโด้ ซอยประดิพัทธ์ 1		โรงแรม	0101000020877
9	โรงแรมไพเราะวิดี ซอยอินทามระ 4		โรงแรม	0101000020877
10	สวนพลาเวซ ซอยประดิพัทธ์ 1		โรงแรม	0101000020877
11	โรงแรมในรัชดาภิเษก ซอยอินทามระ 1/1		โรงแรม	0101000020877

รูปที่ 16 ตัวอย่างตาราง hotel

ชื่อตาราง

police

Field	Constraint	Type	Description
Gid	Primary key	Serial	ลำดับของข้อมูล
Namt	CHECK (ndims(the_geom) = 2)	Character	ชื่อสถานีตำรวจ/สถานีตำรวจ ดับเพลิง
Brancht	Not null	Character	สถานที่ตั้ง
Group	Not null	Character	สถานีตำรวจ/ สถานีตำรวจดับเพลิง
The_geom	CHECK (geometrytype(the_geom))	Geometry	พิกัดของสถานีตำรวจ / สถานี ตำรวจดับเพลิง

ตารางที่ 6 แสดงข้อมูลของตาราง police

gid [PK] serial	namt character var	brancht character var	group character var	the_geom geometry
1	สถานีตำรวจนครฯ ซอยรามอินทรา		สถานีตำรวจ	0101000020877
2	สถานีตำรวจนครฯ ซอยแจนเกษ 1		สถานีตำรวจ	0101000020877
3	สถานีตำรวจนครฯ ซอยกรุงเทพา-นา		สถานีตำรวจ	0101000020877
4	กองกำกับการสอ ซอยกรุงเทพา-นา		สถานีตำรวจ	0101000020877
5	สถานีตำรวจนครฯ ซอยประชาเกษ		สถานีตำรวจ	0101000020877
6	สถานีตำรวจดับเพลิง ซอยประชาเกษ		สถานีตำรวจดับเพลิง	0101000020877
7	สถานีตำรวจนครฯ ซอยหมู่บ้านแรก		สถานีตำรวจ	0101000020877
8	สถานีตำรวจนครฯ ซอยสุทธิพงษ์ 1		สถานีตำรวจ	0101000020877
9	สถานีตำรวจดับเพลิง ถนนสุทธิสารวินิจฉัย		สถานีตำรวจดับเพลิง	0101000020877
10	สถานีตำรวจนครฯ ซอยพหลโยธิน 1		สถานีตำรวจ	0101000020877
11	สถานีตำรวจดับเพลิง ถนนเศรษฐศิริ		สถานีตำรวจดับเพลิง	0101000020877

รูปที่ 17 ตัวอย่างตาราง police

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ชื่อตาราง temple

Field	Constraint	Type	Description
Gid	Primary key	Serial	ลำดับของข้อมูล
Namt	CHECK (ndims(the_geom) = 2)	Character	ชื่อวัด/ มัสยิด/ สุเหร่า/ โบสถ์ คริสต์
Brancht	Not null	Character	สถานที่ตั้ง
Group	Not null	Character	วัด/ มัสยิด/ สุเหร่า/ โบสถ์คริสต์
The_geom	CHECK (geometrytype(the_geom)	Geometry	พิกัดของวัด/ มัสยิด/ สุเหร่า/ โบสถ์คริสต์

ตารางที่ 7 แสดงข้อมูลของตาราง temple

gid	namt	brancht	group	the_geom
[PK] serial	character var	character var	character var	geometry
1	วัดเป็นทองโกลน	ถนนเข้าวัดเป็นท	วัด	01010000000000
2	วัดพระยาสุเรนทร์	ถนนพระยาสุเรน	วัด	01010000000000
3	วัดสันมาชัญญาว	ถนนพระยาสุเรน	วัด	01010000000000
4	วัดคูบอง	ซอยวัดคูบอง	วัด	01010000000000
5	วัดลาดปลาเค้า	ถนนลาดปลาเค้า	วัด	010100000066C
6	มัสยิดขามมีลีนาห์	ถนนพระยาสุเรน	มัสยิด/สุเหร่า	01010000000000
7	มัสยิดขามมีลีนาห์	ซอยเจริญพัฒนา	มัสยิด/สุเหร่า	01010000000000
8	มัสยิดอัล-เอารกี	ซอยรามอินทรา	มัสยิด/สุเหร่า	010100000000AS
9	วัดนวมินทร์(บาง)	ถนนนวมินทร์	วัด	01010000001A4
10	วัดบุญศรีมงคล	ซอยหมู่บ้านเอื้อ	วัด	01010000000076
11	คริสต์จักรกรุงเทพ	ซอยเสนานิเทศน์	โบสถ์คริสต์	01010000000B4F
12	วัดสาครสุนัประช	ซอยหมู่บ้านเขนเค	วัด	010100000000773
13	วัดสิริกมลवासเร	ถนนสุขาภิบาล 1	วัด	01010000000265
14	มัสยิดจุฬาลักษณ์	ซอยเจริญพัฒนา	มัสยิด/สุเหร่า	01010000000000
15	วัดคณอศุ์ปิรุโย	ซอยสำนักสงฆ์	วัด	0101000000000E
16	มัสยิดมีเตาผู้ลย	ซอยสุขา	มัสยิด/สุเหร่า	0101000000007F
17	วัดแสนสุข	ซอยวัดแสนสุข	วัด	0101000000091E

รูปที่ 18 ตัวอย่างตาราง temple

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ชื่อตาราง govern

Field	Constraint	Type	Description
Gid	Primary key	Serial	ลำดับของข้อมูล
Namt	CHECK (ndims(the_geom) = 2)	Character	ชื่อสถานที่ราชการ
Brancht	Not null	Character	สถานที่ตั้ง
Group	Not null	Character	ประเภทสถานที่ราชการ
The_geom	CHECK (geometrytype(the_geom))	Geometry	พิกัดของสถานที่ราชการ

ตารางที่ 8 แสดงข้อมูลของตาราง govern

gid	namt	brancht	group	the_geom
[PK] serial	character varyin	character var	character var	geometry
1	คำขวัญเกลือโลมาเรท	ซอยหมู่บ้านเพ็ญศรี	หน่วยงานราชการ	0101000020877
2	สำนักงานหมวดการท	ถนนประชาราษฎร์	หน่วยงานราชการ	0101000020877
3	ที่ว่าการหมวดควนคง	ซอยแสงเลข 1	หน่วยงานราชการ	0101000020877
4	องค์การขนส่งมวลชน	ซอยศรีนครินทร์	หน่วยงานรัฐวิสาห	0101000020877
5	เรือนำพิเศษมินบุรี	ซอยแสงสุข 1	หน่วยงานราชการ	0101000020877
6	พื้นที่เขตกานวิชัยมนตรี	ซอยแสงผล(ผู้ใหญ่)	หน่วยงานราชการ	0101000020877
7	สำนักงานเกษตรนิคม	ซอยแสงสวัสดิ์ผู้ใหญ่	หน่วยงานราชการ	0101000020877
8	สำนักงานสรรพากรเขต	ซอยสหกรณ์	หน่วยงานราชการ	0101000020877
9	สำนักงานภาษีสรรพากร	ถนนวิบูลย์ราษฎร์	หน่วยงานราชการ	0101000020877
10	ศูนย์เยาวชนมินบุรี	ถนนวิบูลย์ราษฎร์	หน่วยงานราชการ	0101000020877
11	ที่ว่าการนายตำรวจภาค	ถนนประชาราษฎร์	หน่วยงานราชการ	0101000020877
12	องค์การขนส่งมวลชน	ซอยหมู่บ้านเป็ญศรี	หน่วยงานรัฐวิสาห	0101000020877
13	ที่ว่าการหน่วยบางศรี	ถนนรถไฟ	หน่วยงานราชการ	0101000020877
14	ที่ว่าการพนักงานตรวจ	ทางรถไฟ	หน่วยงานรัฐวิสาห	0101000020877
15	ศูนย์รวมรถสามล้อ	ซอยอ้อและเอ็ด	หน่วยงานราชการ	0101000020877
16	กองกำลังกองปราบปร	ซอยโชคชัย 4	หน่วยงานราชการ	0101000020877

รูปที่ 19 ตัวอย่างตาราง govern

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.2 ฐานข้อมูลของระบบสภาพจราจร

ในระบบสภาพจราจรได้เพิ่มข้อมูลสภาพจราจรขาเข้าและขาออกของทั้ง 7 เขตตัวอย่าง  
ชื่อตาราง mainroad\_in

Field	Constraint	Type	Description
Gid	Primar key	Serial	ลำดับของข้อมูล
Fnode_	Not null	Bigint	Form node
Tnode_	Not null	Bigint	To node
Length	Not null	double	ระยะทางระหว่าง node
Mainroad_	Not null	Bigint	ลำดับของถนนหลัก
Mainroad_i	Not null	Bigint	หมายเลขของถนนหลัก
Rdlnnamt	CHECK (ndims(the_geom) = 2)	Bigint	ชื่อถนน
Startx_coo	Not null	Double	จุดเริ่มต้นพิกัด x
Starty_coo	Not null	Double	จุดเริ่มต้นพิกัด y
Endx_coord	Not null	Double	จุดสิ้นสุดพิกัด x
Endy_coord	Not null	Double	จุดสิ้นสุดพิกัด y
The_geom	CHECK (geometrytype(the_geom))	Geometry	พิกัด
Velocity	Not null	Bigint	อัตราเร็วของรถ
Khet	Not null	Text	เขต

ตารางที่ 9 แสดงข้อมูลตาราง mainroad\_in

gid	fnode_	tnode_	length	mainroad_	mainroad_i	rdlnnamt	startx_coo	starty_coo	endx_coord	endy_coord	velocity	khet
integer	bigint	bigint	double precis	bigint	bigint	character var	double precis	double precis	double precis	double precis	bigint	text
606	21348	21286	480.623	605	78492	ถนนสุขุมวิท	670025.5625	1524721.375	670502.875	1524729	87	เขตวัฒนา
606	21341	21348	301.636	606	78494	ถนนสุขุมวิท	669724.6875	1524701	670025.5625	1524721.375	34	เขตวัฒนา
609	21352	20951	696.878	609	78497	ถนนพหลโยธิน	665604.8125	152423.125	665857.1875	1523072.625	76	เขตจตุจักร
618	20985	20354	682.689	618	78510	ถนนรัชดาภิเษก	669367.375	1520209.25	669507.625	1520897.375	28	เขตจตุจักร
636	20938	21345	680.493	636	78549	ถนนพหลโยธิน	666952.25	1520305.375	667091	1520971.25	56	เขตจตุจักร
637	21945	21094	774.921	637	78550	ถนนพหลโยธิน	667091	1520971.25	667107.0625	1521745	123	เขตจตุจักร
638	21097	21098	838.176	638	78551	ถนนพหลโยธิน	667693.5625	1521855.875	668440	1521475	76	เขตจตุจักร
639	20595	20586	765.434	639	78552	ถนนพหลโยธิน	666781.6875	1521098.125	666532.4375	1520997.125	33	เขตจตุจักร
640	21105	21358	612	640	78553	ถนนพหลโยธิน	669818.8125	1523104.875	669852.1875	1523464.5	67	เขตจตุจักร
667	21364	21302	2396.276	667	78588	ถนนพหลโยธิน (สุข)	671665.875	1517023.875	672228.5	1519399.875	6	เขตจตุจักร
678	21096	21384	491.956	678	78498	ถนนพหลโยธิน	666221.1875	1521179.375	666454.75	1521612.25	34	เขตจตุจักร
705	21409	21276	934.66	705	78665	ถนนพหลโยธิน	667700.0625	1524714.5	667997	1525600.5	34	เขตจตุจักร

รูปที่ 20 ตัวอย่างตาราง mainroad\_in

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ชื่อตาราง mainroad\_out

Field	Constraint	Type	Description
Gid	Primar key	Serial	ลำดับของข้อมูล
Fnode_	Not null	Bigint	Form node
Tnode_	Not null	Bigint	To node
Length	Not null	double	ระยะทางระหว่าง node
Mainroad_	Not null	Bigint	ลำดับของถนนหลัก
Mainroad_i	Not null	Bigint	หมายเลขของถนนหลัก
Rdlnnamt	CHECK (ndims(the_geom) = 2)	Bigint	ชื่อถนน
Startx_coo	Not null	Double	จุดเริ่มต้นพิกัด x
Starty_coo	Not null	Double	จุดเริ่มต้นพิกัด y
Endx_coord	Not null	Double	จุดสิ้นสุดพิกัด x
Endy_coord	Not null	Double	จุดสิ้นสุดพิกัด y
The_geom	CHECK (geometrytype(the_geom))	Geometry	พิกัด
Velocity	Not null	Bigint	อัตราเร็วของรถ
Khet	Not null	Text	เขต

ตารางที่ 10 แสดงข้อมูลตาราง mainroad\_out

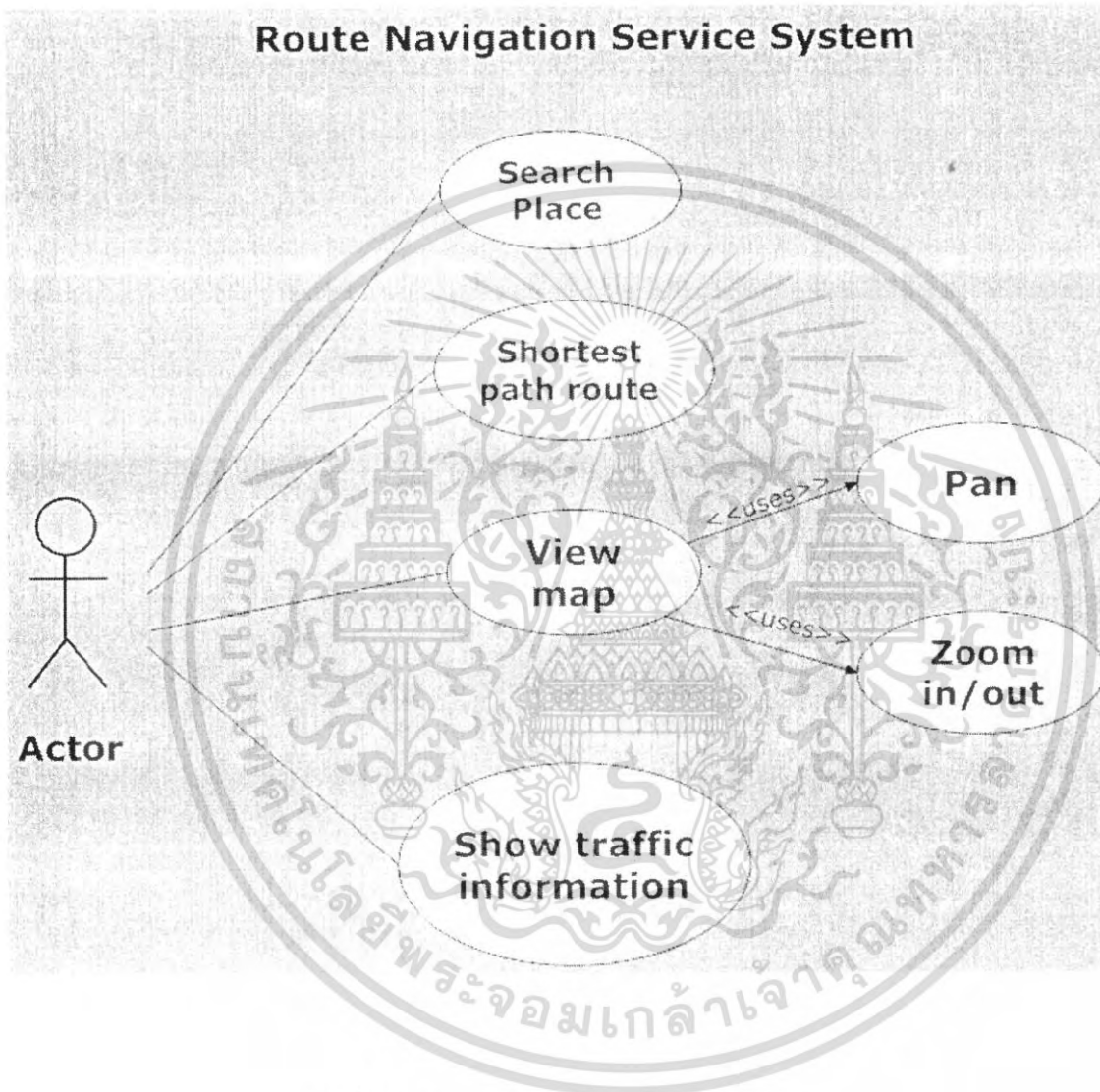
gid	fnode_	tnode_	length	mainroad_	mainroad_i	rdlnnamt	startx_coo	starty_coo	endx_coord	endy_coord	velocity	khet
integer	bigint	bigint	double precis	bigint	bigint	character var	double precis	double precis	double precis	double precis	bigint	text
504	20923	20952	315.008	304	78048	ถนนพหลโยธิน	664748.8125	1521082.75	665045	1520979.75	2	เขตจตุจักร
1539	21092	21703	1060.008	1539	78604	ถนนสายเมือง	664614.875	1519020.25	664755.375	1520070.75	15	เขตปทุมวัน
1540	21703	21375	201.321	1540	78604	ถนนสายเมือง	664589.125	1518820.625	664614.875	1519020.25	40	เขตปทุมวัน
1011	21375	21078	289.457	1011	79160	ถนนพหลโยธิน 4	664589.125	1518820.625	664858.3125	1518714.25	60	เขตปทุมวัน
295	21078	21079	519.166	295	78037	ถนนพหลโยธิน 4	664858.3125	1518714.25	665339.6875	1518520.125	110	เขตปทุมวัน
296	21079	21071	314.563	296	78038	ถนนพหลโยธิน 4	665339.6875	1518520.125	665631.4375	1518402.25	160	เขตปทุมวัน
288	21071	20835	568.939	288	78029	ถนนพหลโยธิน 4	665631.4375	1518402.25	666156.0625	1518182.125	60	เขตปทุมวัน
127	20835	20836	357.626	127	77659	ถนนพหลโยธิน 4	666156.0625	1518182.125	666485.9375	1518044	68	เขตปทุมวัน
562	20836	21072	343.414	562	78438	ถนนพหลโยธิน 4	666485.9375	1518044	666802	1517909.75	90	เขตปทุมวัน
289	21072	21016	437.925	289	78030	ถนนพหลโยธิน 4	666802	1517909.75	667205.6875	1517740.125	150	เขตปทุมวัน
298	20762	21064	1036.081	298	78042	ถนนพหลโยธิน	667406.3125	1518553	667651.8125	1519575.625	80	เขตปทุมวัน

รูปที่ 21 ตัวอย่างตาราง mainroad\_out

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3.2 การออกแบบระบบ

### 3.2.1 Use case diagram

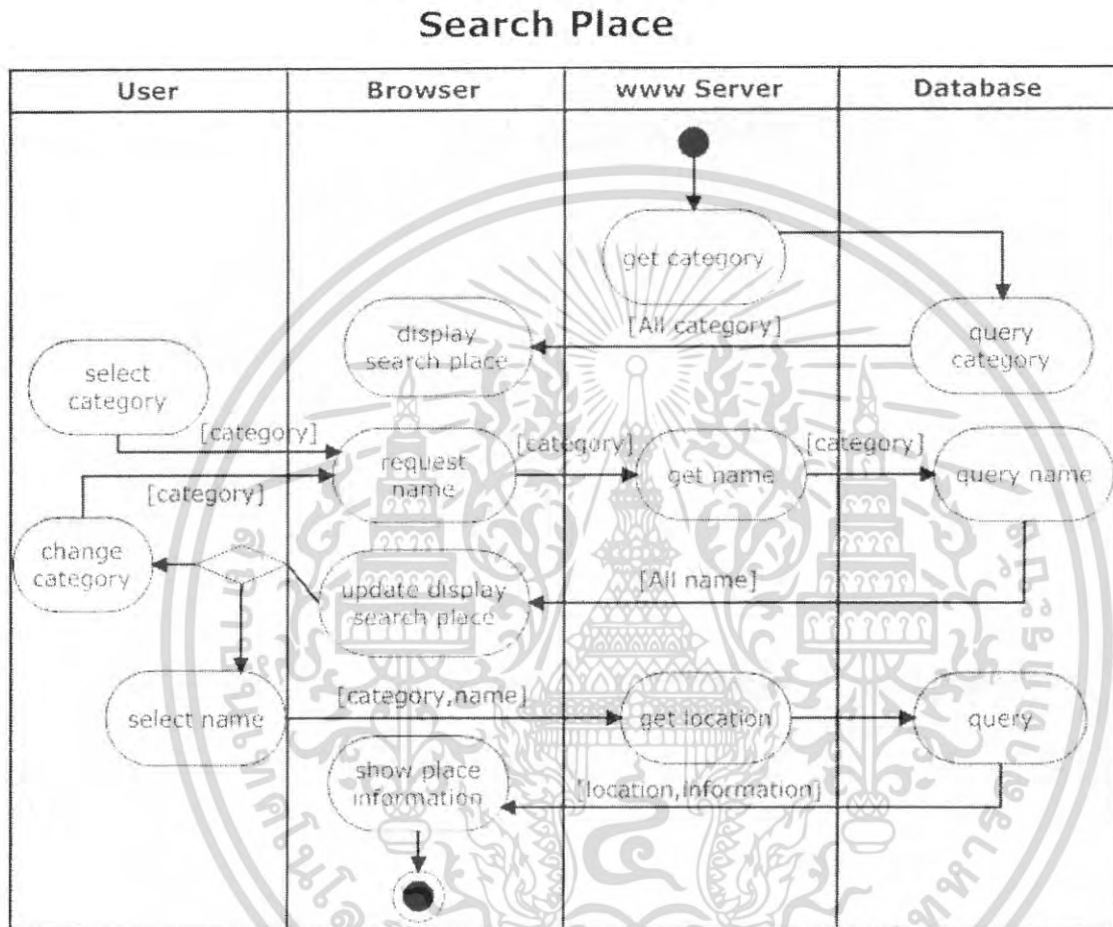


รูปที่ 22 แสดง Use case diagram ของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.2 Activity diagram

#### 1) ค้นหาสถานที่ (Search Place)

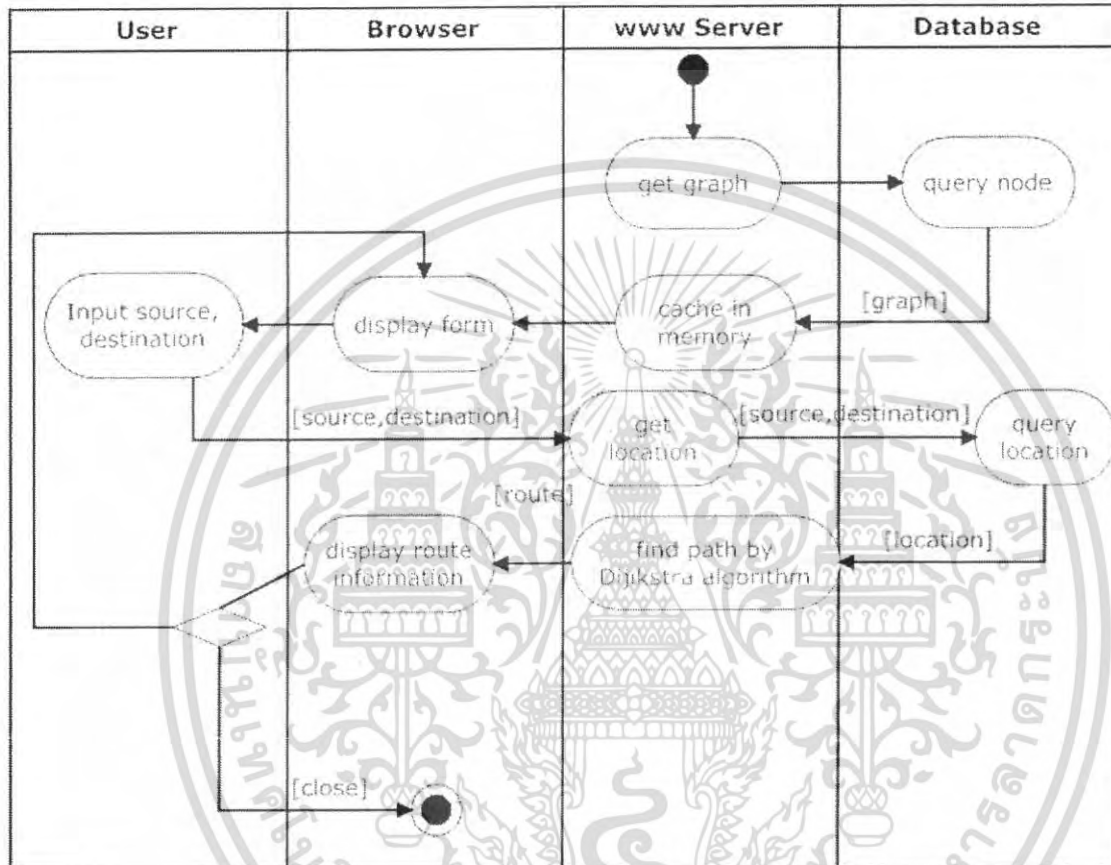


รูปที่ 23 แสดง Activity diagram ของ การค้นหาสถานที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) คำนวณเส้นทางที่ใช้ระยะทางสั้นที่สุด (Shortest path route)

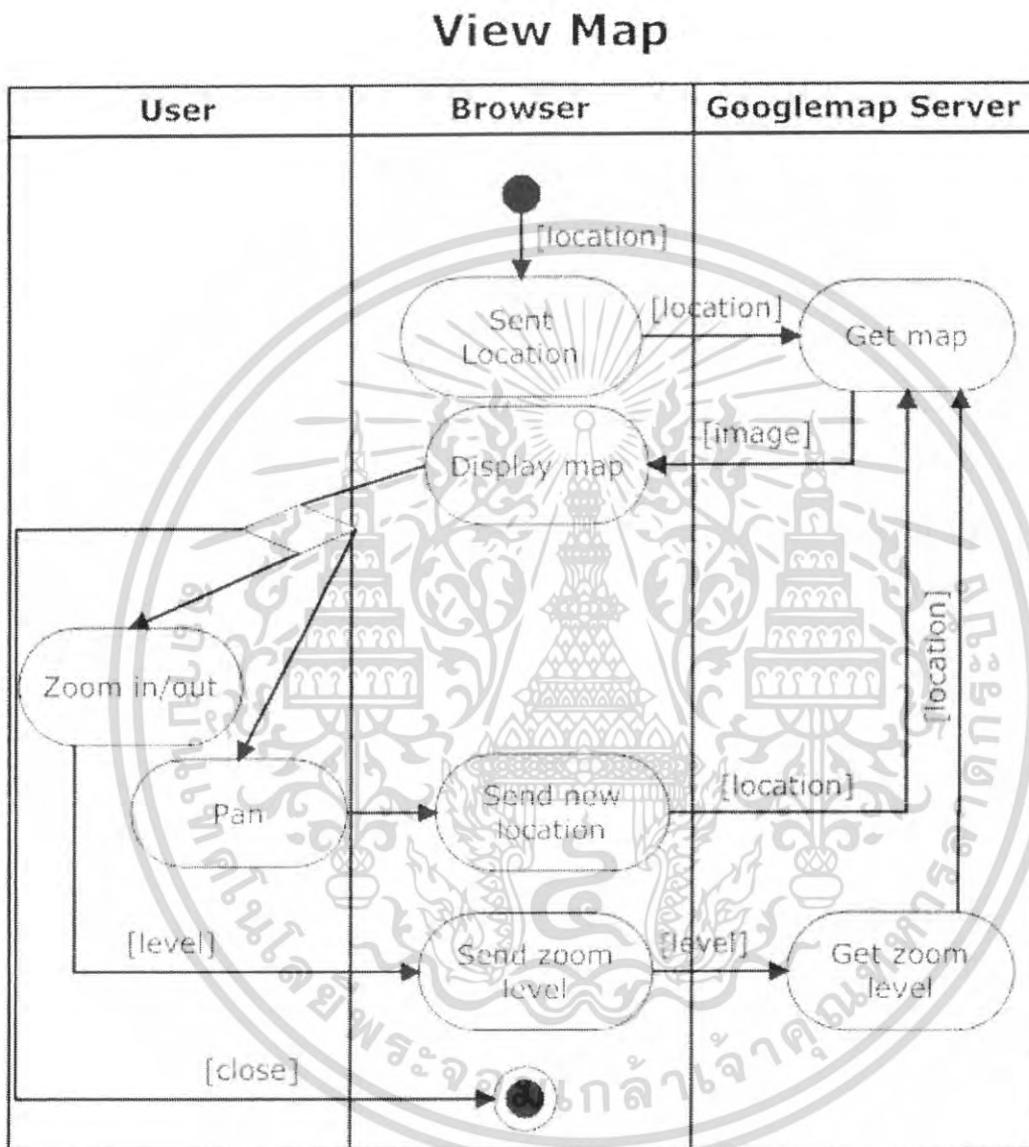
### Shortest path Route



รูปที่ 24 แสดง Activity diagram ของ การคำนวณเส้นทางที่ใช้ระยะทางสั้นที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3) แสดงแผนที่ (View map)

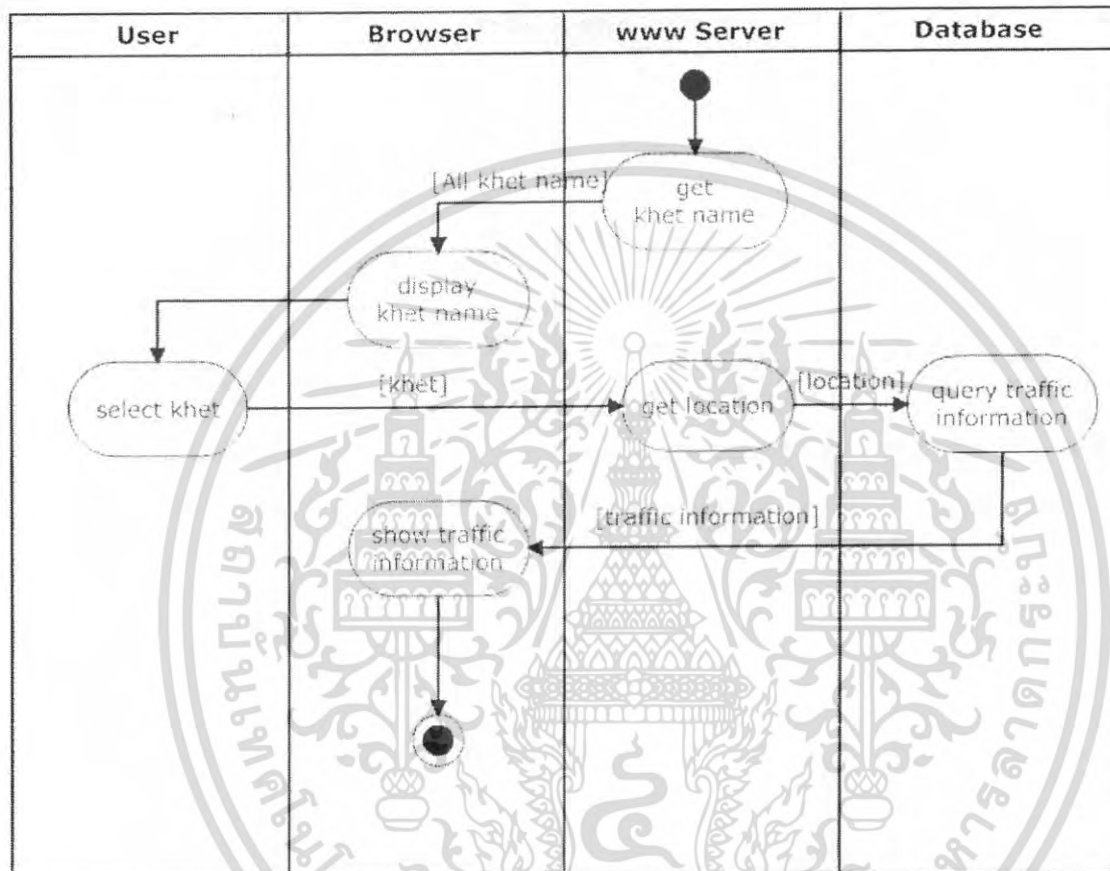


รูปที่ 25 แสดง Activity diagram ของ การดูแผนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4) แสดงสภาพจราจร (Traffic information system)

## Traffic Information System

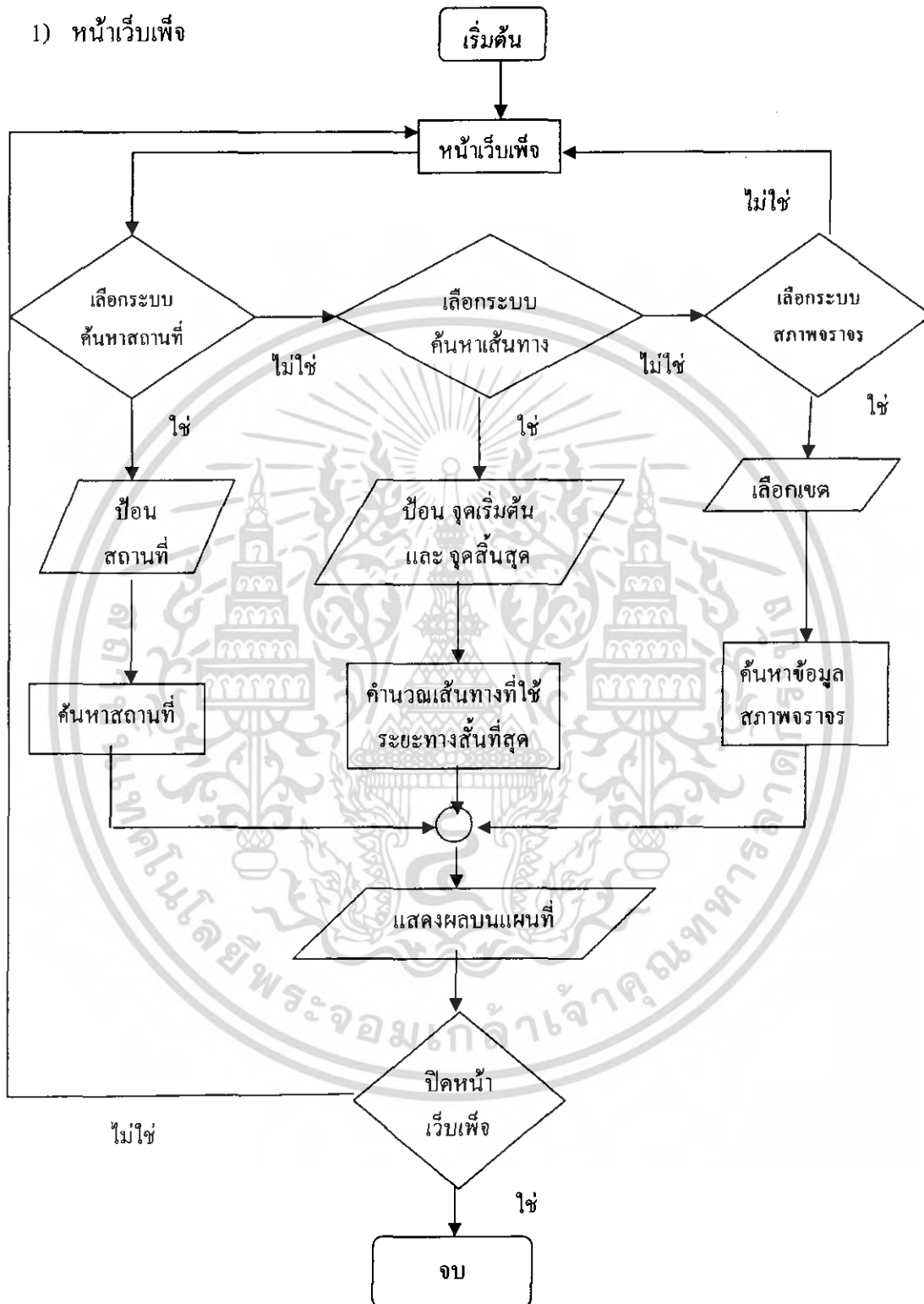


รูปที่ 26 แสดง Activity diagram ของการแสดงผลจราจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.3 System flowchart

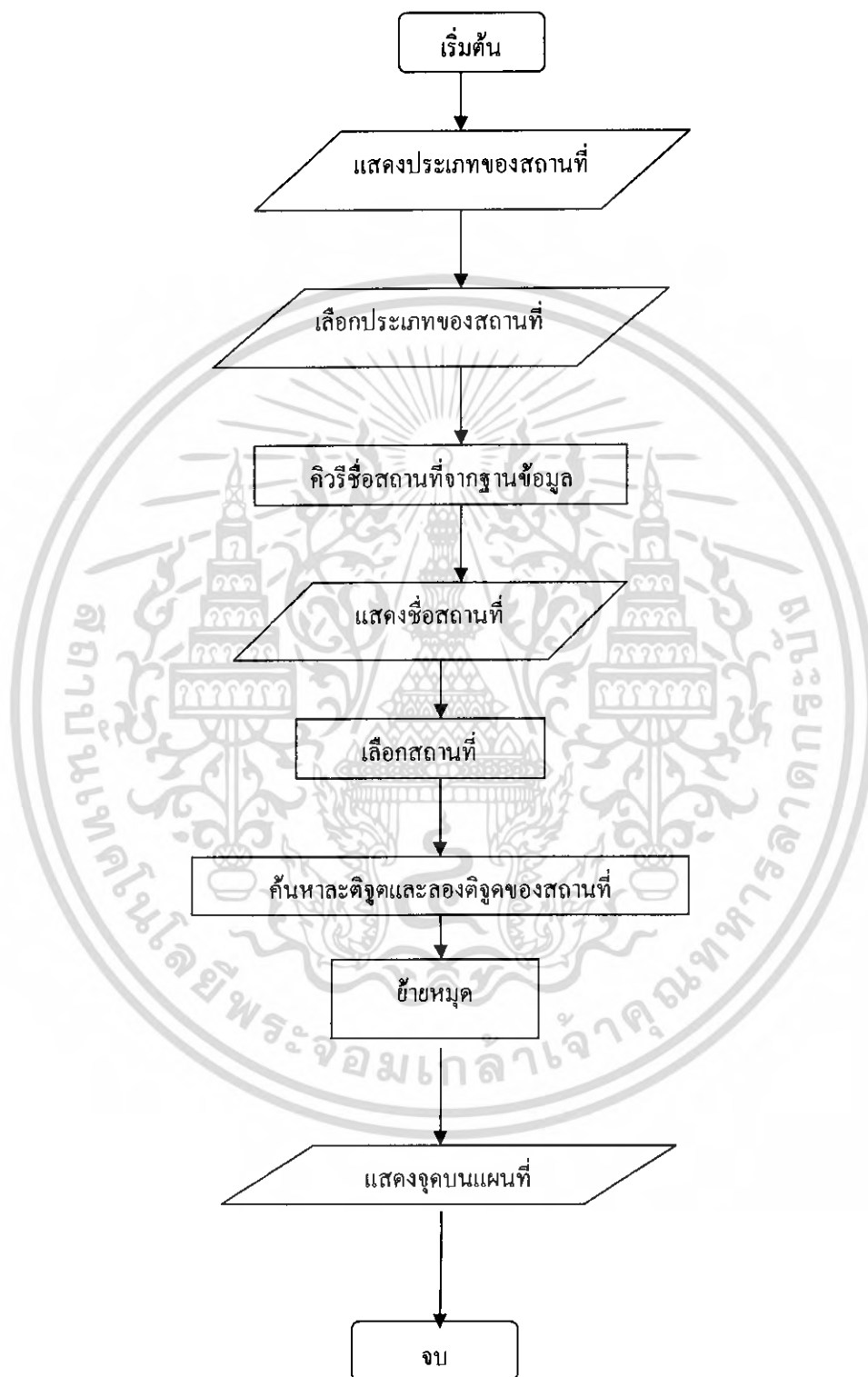
#### 1) หน้าเว็บเพจ



รูปที่ 27 แสดง System flowchart ของการทำงานของหน้าเว็บเพจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

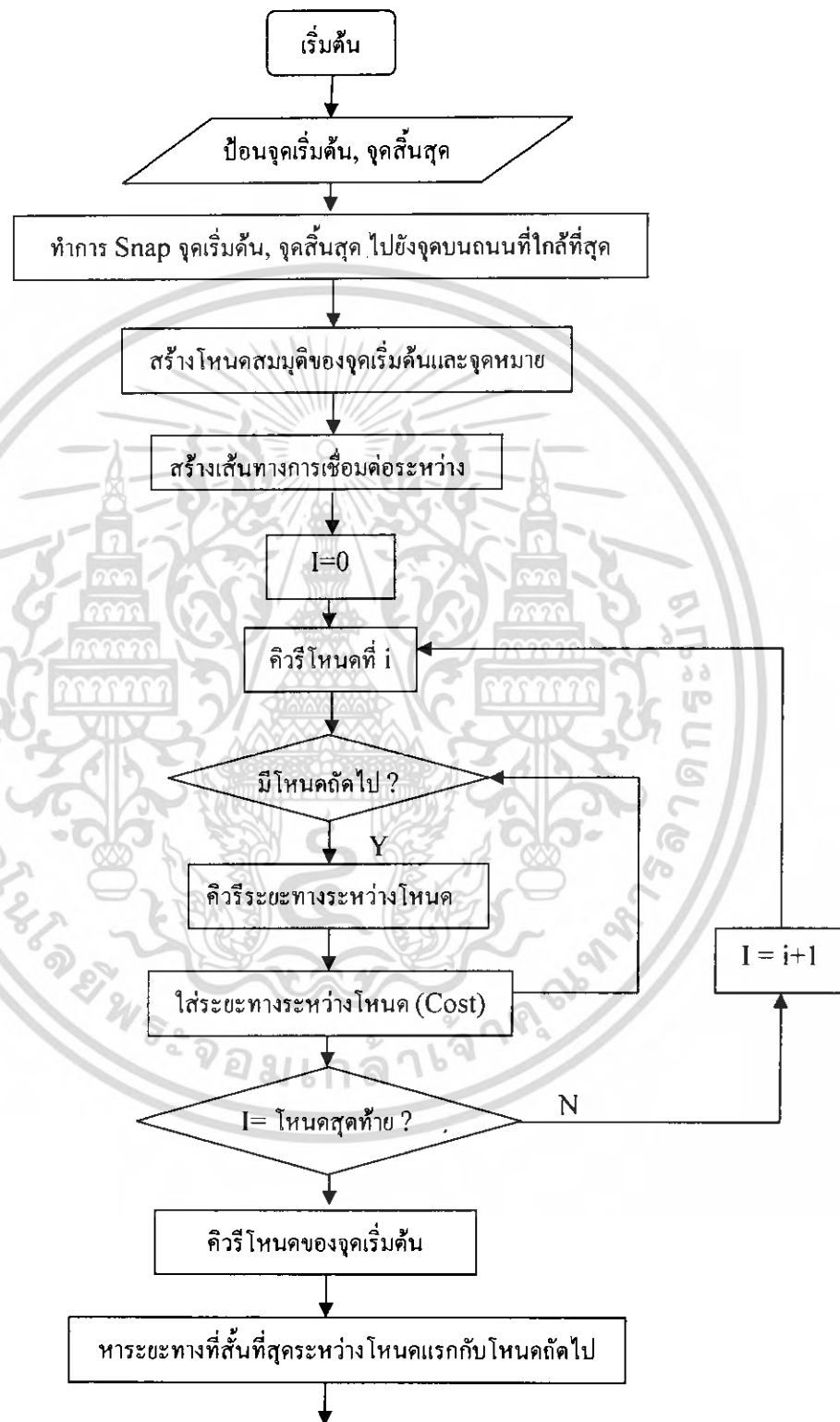
2) ระบบค้นหาสถานที่สำคัญในกรุงเทพมหานคร



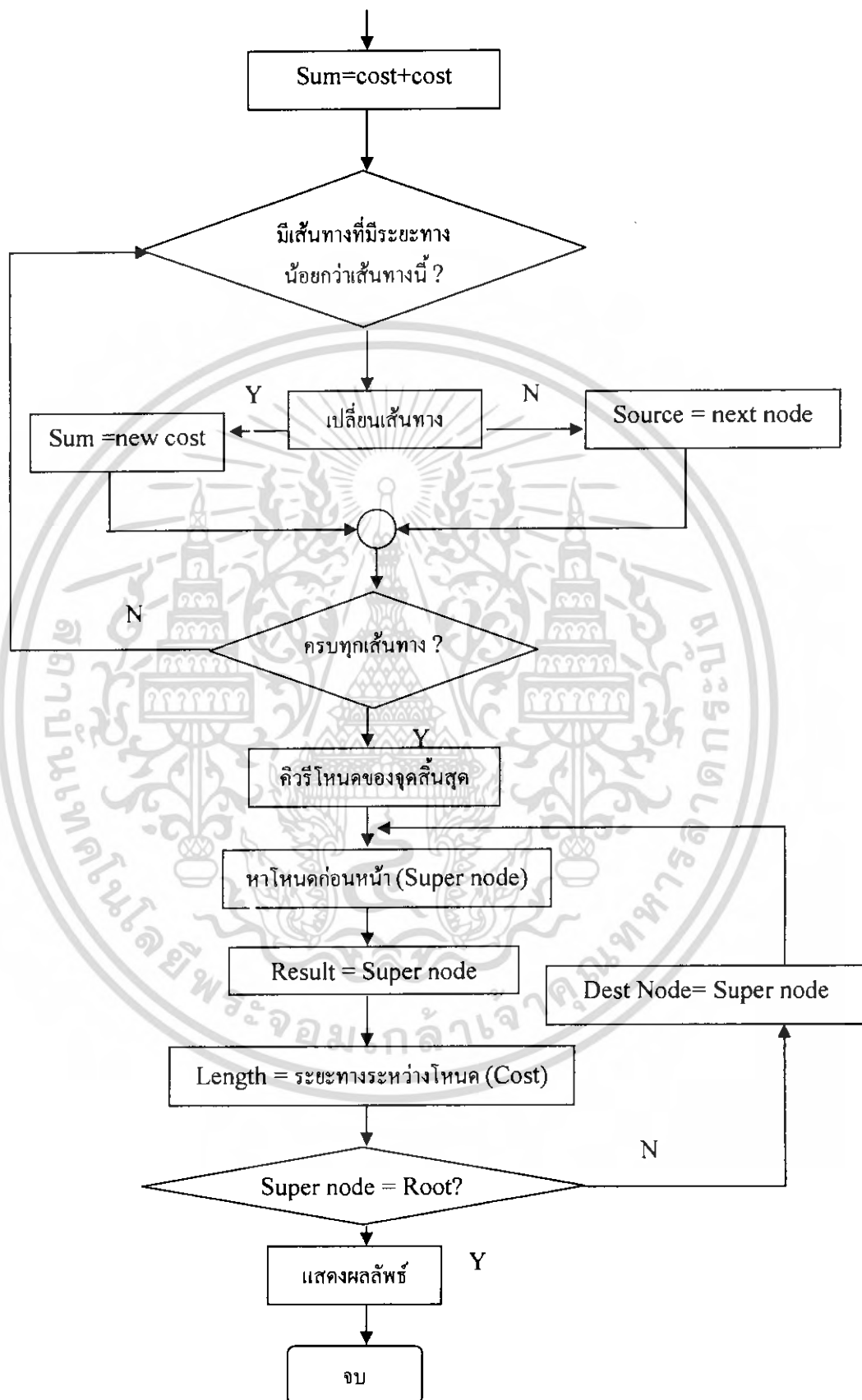
รูปที่ 28 แสดง System flowchart ของการทำงานของระบบค้นหาสถานที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ระบบคำนวณหาเส้นทางที่ใช้ระยะทางสั้นที่สุด โดยใช้อัลกอริทึม Dijkstra

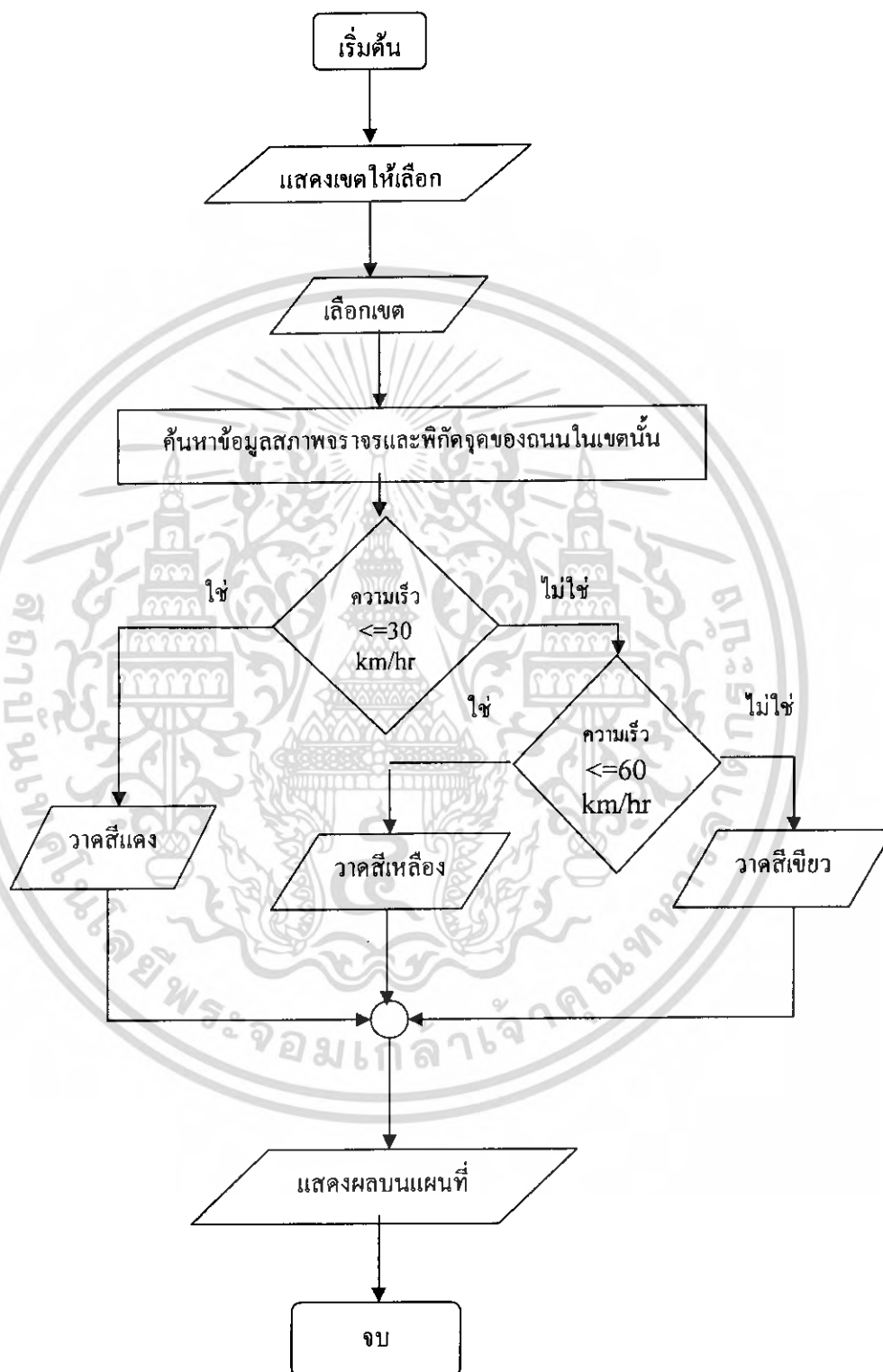


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 29 แสดง System flowchart ของการทำงานของระบบคำนวณเส้นทางที่สั้นที่สุด เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4) ระบบแสดงสภาพการจราจร



รูปที่ 30 แสดง System flowchart ของการทำงานของระบบแสดงสภาพการจราจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### ผลการทดลอง

#### 4.1 ระบบค้นหาสถานที่สำคัญในกรุงเทพมหานคร

##### 4.1.1 ปัญหาและวิธีการแก้ไข

ปัญหาที่เกิดขึ้นจากโครงการ 1 คือ ไม่สามารถใช้ฐานข้อมูลที่เป็นระบบพิกัดกริด (UTM) ในการค้นหาสถานที่ได้ เพราะจะแสดงตำแหน่งไม่ตรงกับแผนที่ Google Maps API จึงแก้ปัญหาโดยใช้ฐานข้อมูลที่เป็นระบบละติจูด, ลองจิจูด

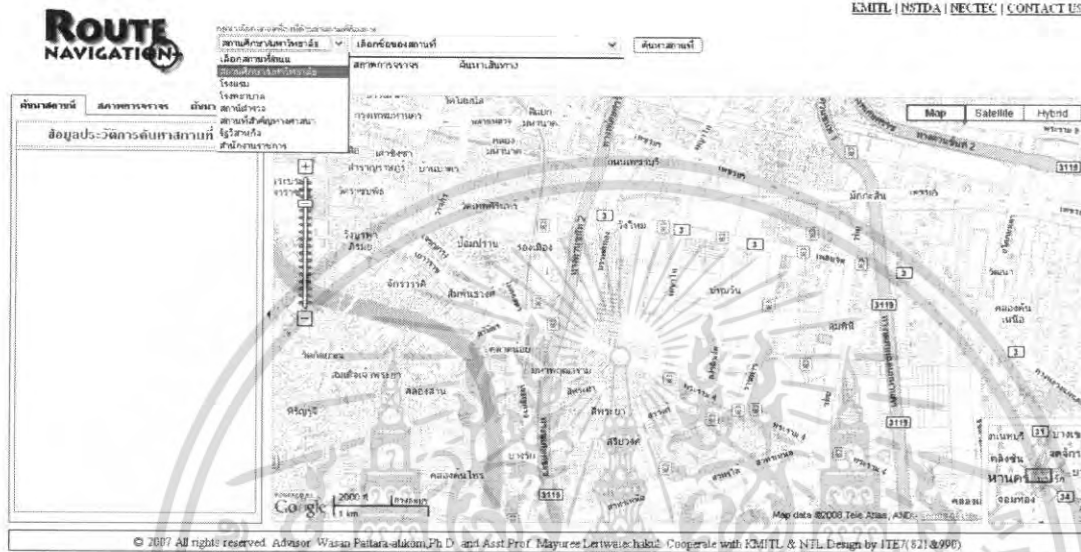
เนื่องจากพิกัดจุดในฐานข้อมูลระบบละติจูด, ลองจิจูด จะมีค่าผิดพลาด (Error) คงที่เมื่อเทียบกับพิกัดจุดบนแผนที่ Google Maps API ในโครงการ 2 จึงทำการปรับการแสดงผลให้สามารถทำการแสดงตำแหน่งของสถานที่บนแผนที่ Google Maps API ได้โดยใช้ฐานข้อมูลระบบพิกัดกริด (UTM)

##### 4.1.2 ผลการทดลอง

- 1) สถานที่ที่ต้องการค้นหาต้องมีอยู่ในฐานข้อมูล
- 2) สามารถค้นหาสถานที่ในกรุงเทพมหานครได้ โดยแสดงผลผ่าน Google Maps API บนเว็บไซต์
- 3) แสดงข้อมูลสถานที่บนแผนที่ และสามารถเก็บบันทึกข้อมูลที่เคยค้นหาแล้วได้
- 4) สามารถค้นหาข้อมูลเพิ่มเติมได้โดยจะทำการลิงค์ไปที่ [www.google.co.th](http://www.google.co.th)

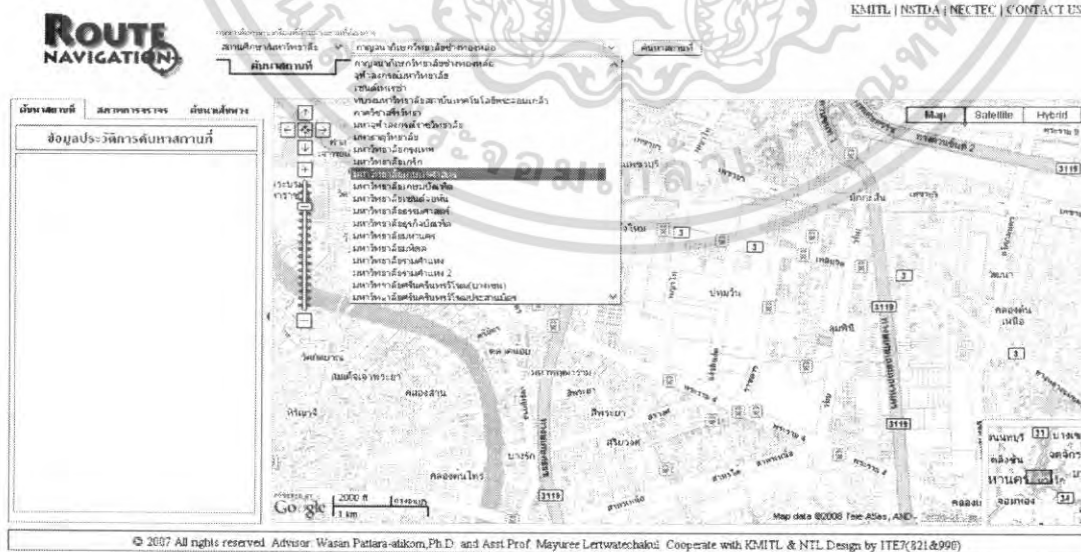
### 4.1.3 การแสดงผลของระบบ

#### 1) คลิกปุ่ม ค้นหาสถานที่ และเลือกประเภทของสถานที่



รูปที่ 31 แสดงการเลือกประเภทของสถานที่

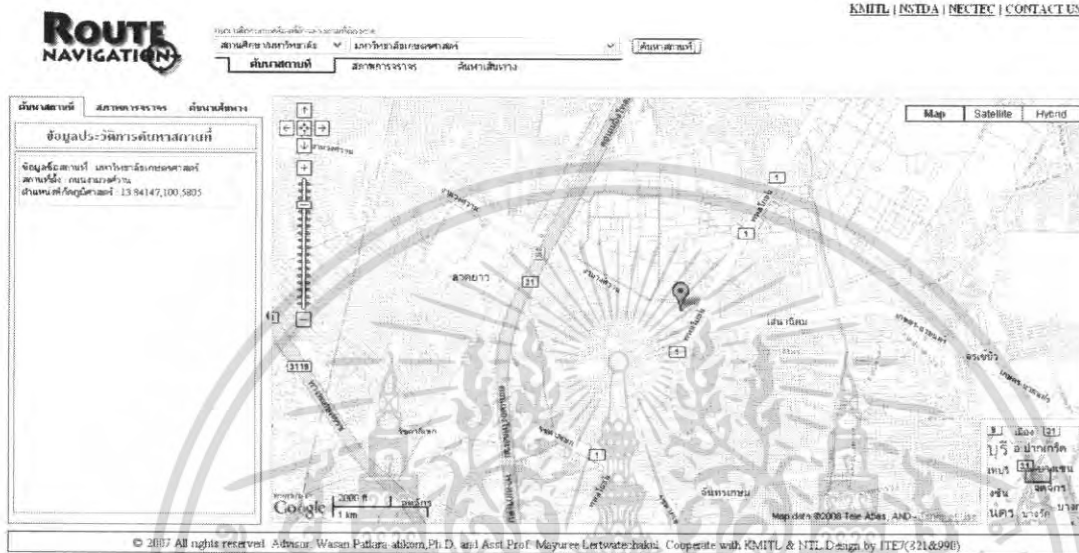
#### 2) เลือกสถานที่ที่ต้องการค้นหา



รูปที่ 32 แสดงการเลือกสถานที่ที่ต้องการค้นหา

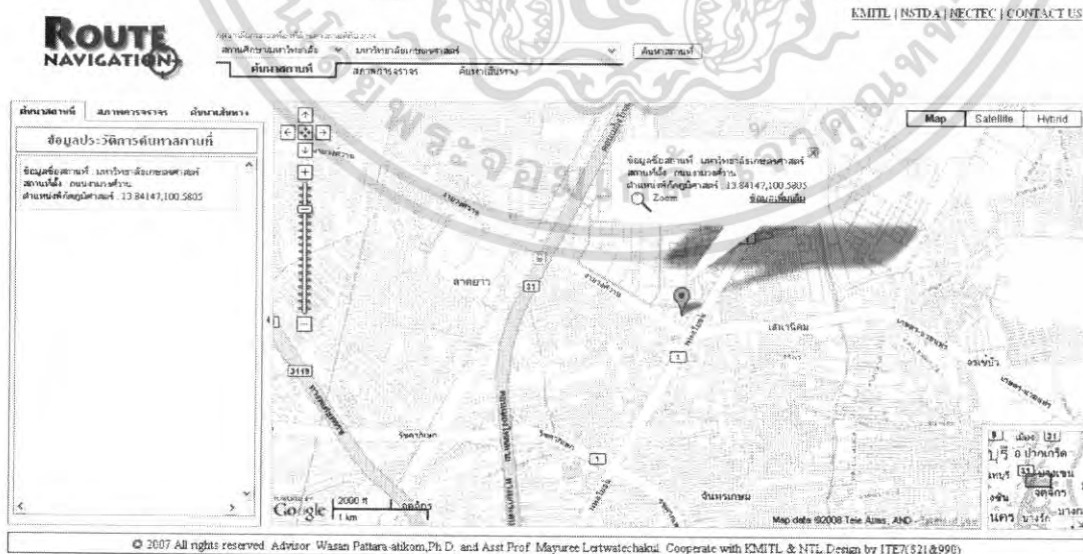
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3) คลิกปุ่มค้นหาสถานที่



รูปที่ 33 แสดงการคลิกปุ่มค้นหาสถานที่

### 4) แสดงตำแหน่งและรายละเอียดของข้อมูล รวมทั้งเก็บบันทึกข้อมูลการค้นหาสถานที่



รูปที่ 34 แสดงตำแหน่งและรายละเอียดของข้อมูล รวมทั้งเก็บบันทึกข้อมูลการค้นหาสถานที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.2 ระบบแนะนำเส้นทางที่ใช้ระยะทางสั้นที่สุด

### 4.2.1 ปัญหาและวิธีการแก้ไข

ปัญหาที่เกิดขึ้นจากโครงการ 1 คือ สามารถแสดงข้อมูลเส้นทางที่สั้นที่สุดอยู่ในรูปแบบของตัวอักษรเท่านั้น ไม่สามารถแสดงผลผ่านแผนที่ Google Maps API ได้ ในโครงการ 2 จึงทำการพัฒนา โดยใช้ข้อมูลจากฐานข้อมูลพิกัดกริด (UTM) คำนวณหาระยะทางสั้นที่สุด และแสดงผลบนแผนที่ Google Maps API ได้พร้อมทั้งแสดงข้อมูลเส้นทางบนแถบด้านข้างของหน้าเว็บเพจ

### 4.2.2 ผลการทดลอง

- 1) สามารถแนะนำเส้นทางที่ใช้ระยะทางสั้นที่สุด โดยกำหนดจุดเริ่มต้นและจุดสิ้นสุดของการเดินทางได้
- 2) แสดงเส้นทางผ่าน Google Maps API
- 3) แสดงข้อมูลเส้นทางบนแถบด้านข้าง
- 4) การเดินทางที่กล่าวถึงในที่นี้จะพิจารณาเพียงเส้นทางจราจร โดยการจราจรที่หนาแน่น
- 5) เส้นทางที่แนะนำยังไม่ได้พิจารณาเรื่อง สภาพจราจร, ทิศทางการจราจร

### 4.2.3 การแสดงผลของระบบ

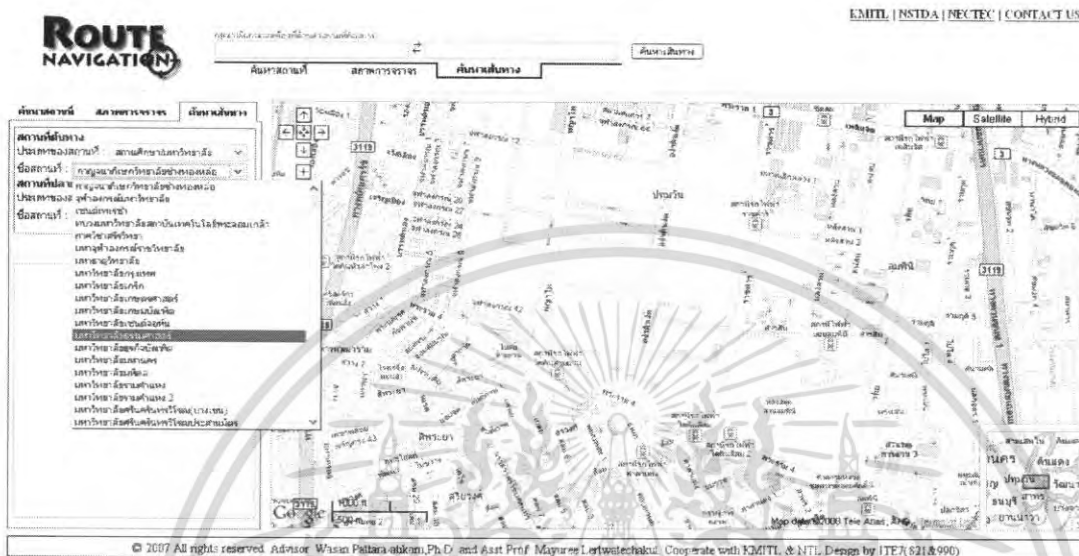
- 1) คลิกปุ่ม ค้นหาเส้นทาง และเลือกประเภทของสถานที่ต้นทาง



รูปที่ 35 แสดงการเลือกประเภทของสถานที่ต้นทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) เลือกสถานที่ต้นทาง



รูปที่ 36 แสดงการเลือกสถานที่ต้นทาง

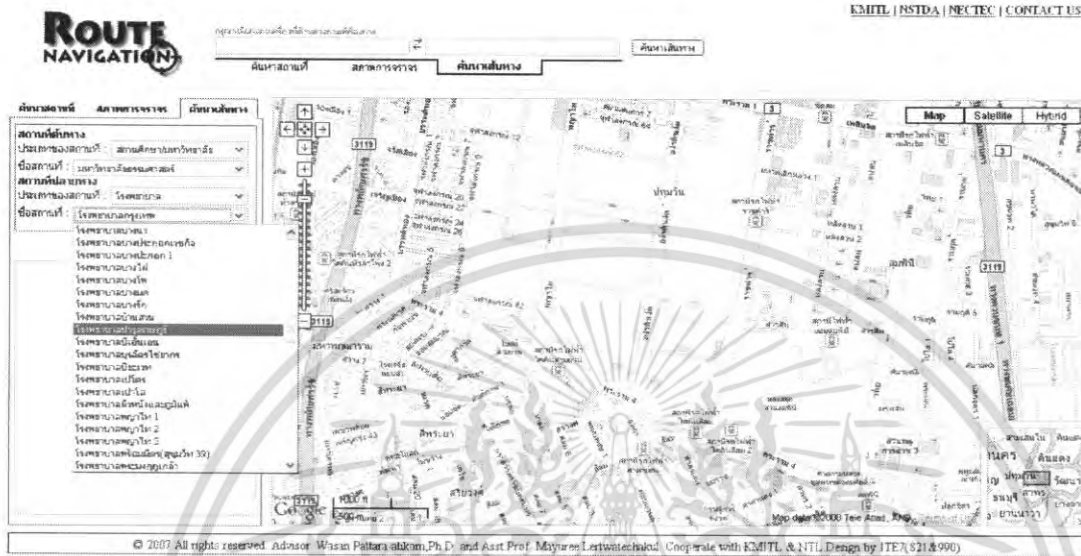
3) เลือกประเภทของสถานที่ปลายทาง



รูปที่ 37 แสดงการเลือกประเภทของสถานที่ปลายทาง

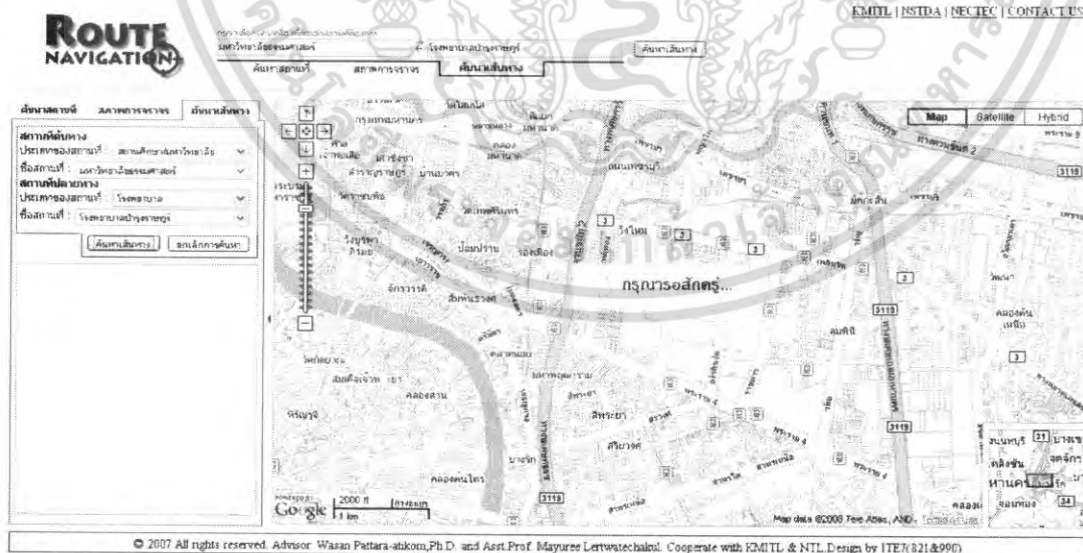
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) เลือกสถานที่ปลายทาง



รูปที่ 38 แสดงการเลือกสถานที่ปลายทาง

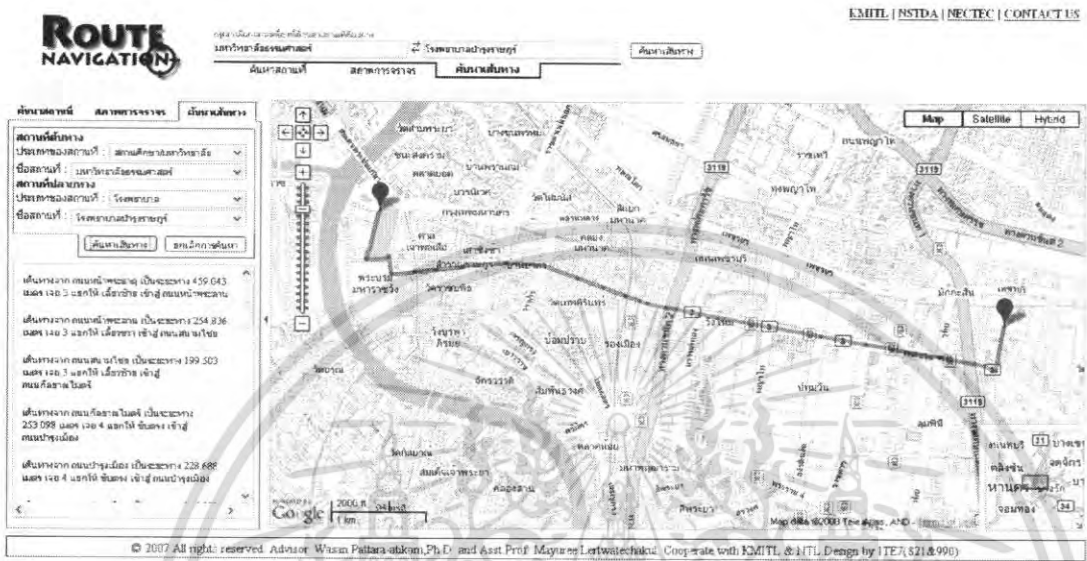
5) คลิกปุ่ม ค้นหาเส้นทาง



รูปที่ 39 แสดงการคลิกปุ่ม ค้นหาเส้นทาง

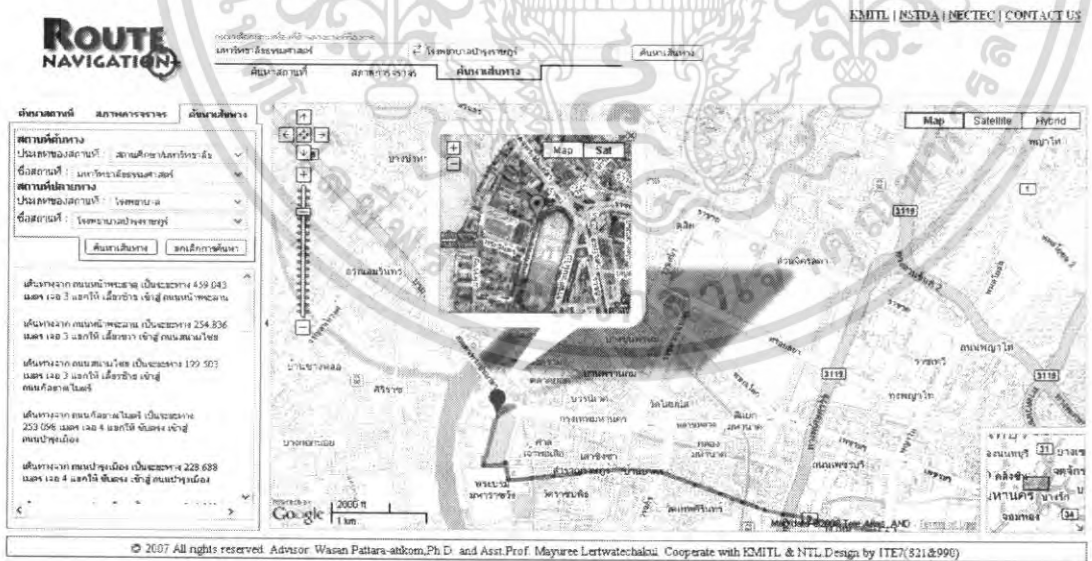
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6) แสดงเส้นทางบน Google Maps API และแสดงข้อมูลเส้นทางบนแถบด้านข้าง



รูปที่ 40 แสดงเส้นทางบน Google Maps API และแสดงข้อมูลเส้นทางบนแถบด้านข้าง

7) สามารถขยาย (zoom) ดูสถานที่ที่ต้นทางและปลายทางได้



รูปที่ 41 แสดงการขยายดูสถานที่ที่ต้นทางและปลายทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.3 ระบบแสดงสภาพจราจร

#### 4.3.1 ปัญหาและวิธีการแก้ไข

เนื่องจากไม่มีข้อมูลสภาพการจราจรจริง จึงทำการสมมติข้อมูลสภาพการจราจรในรูปแบบความเร็ว ใน 7 เขต ได้แก่ เขตปทุมวัน, เขตราชเทวี, เขตพญาไท, เขตห้วยขวาง, เขตดินแดง, เขตวัฒนา และเขตคลองเตย โดยสมมติข้อมูลสภาพการจราจรทั้งขาเข้าและขาออกเพื่อนำมาแสดงบนแผนที่ Google Maps API โดยกำหนดให้

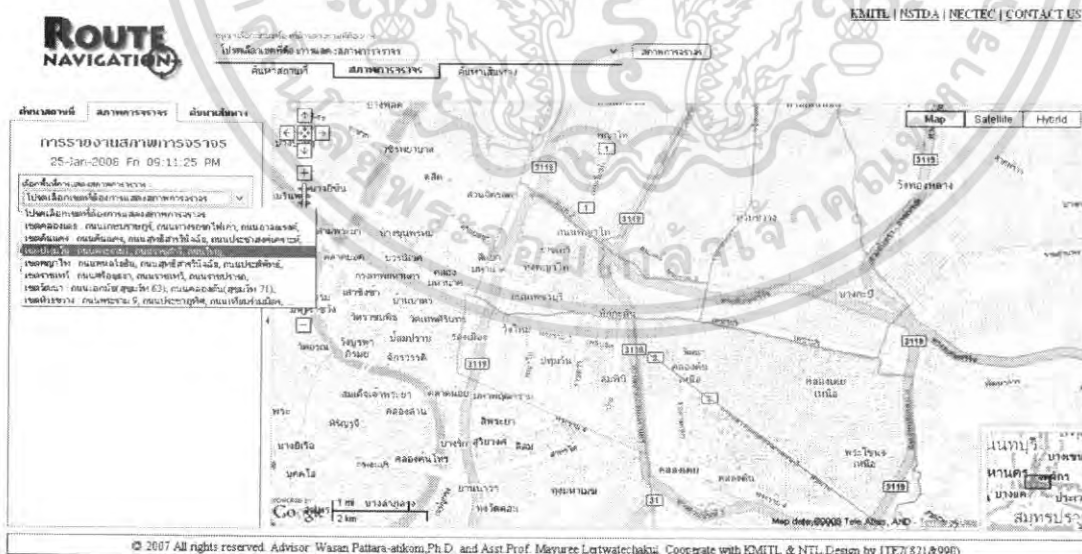
ความเร็วช่วง 30 km/hr ถึง 60 km/hr	มีสภาพการจราจรเป็น สีแดง
ความเร็วช่วง 61 km/hr ถึง 90 km/hr	มีสภาพการจราจรเป็น สีเหลือง
ความเร็วช่วง 90 km/hr ขึ้นไป	มีสภาพการจราจรเป็น สีเขียว

#### 4.3.2 ผลการทดลอง

- 1) สามารถแสดงสภาพจราจรขาเข้าและขาออก ทั้ง 7 เขตบนแผนที่ Google Maps API ได้
- 2) สามารถแสดงข้อมูลสภาพจราจรขาเข้าและขาออกบนแถบด้านข้างได้
- 3) สามารถแสดงคำแนะนำประกอบการพิจารณาได้

#### 4.3.3 การแสดงผลของระบบ

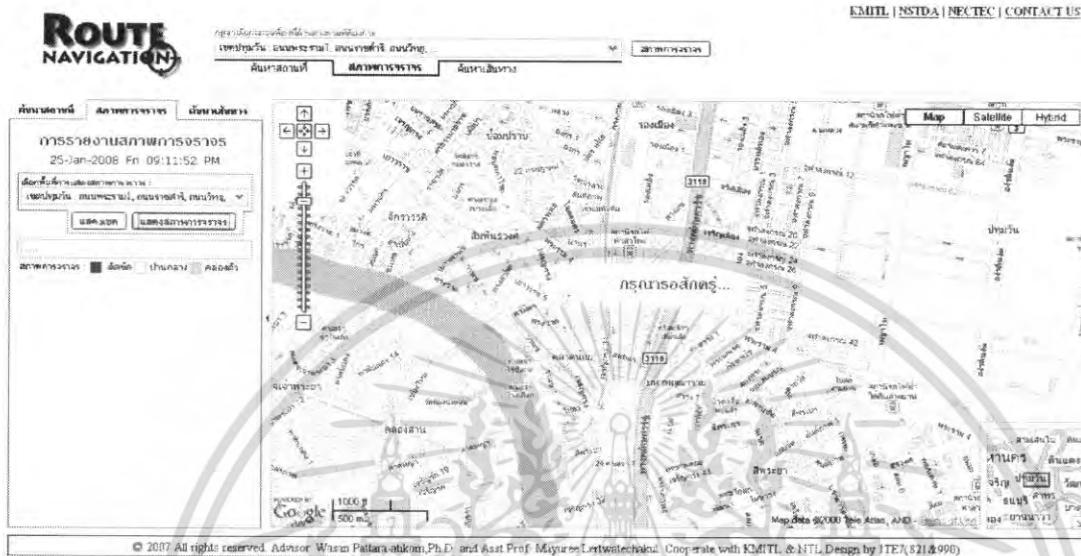
- 1) คลิกปุ่ม สภาพการจราจรและเลือกเขตที่ต้องการดูสภาพการจราจร



รูปที่ 42 แสดงการคลิกปุ่ม สภาพการจราจรและเลือกเขตที่ต้องการดูสภาพการจราจร

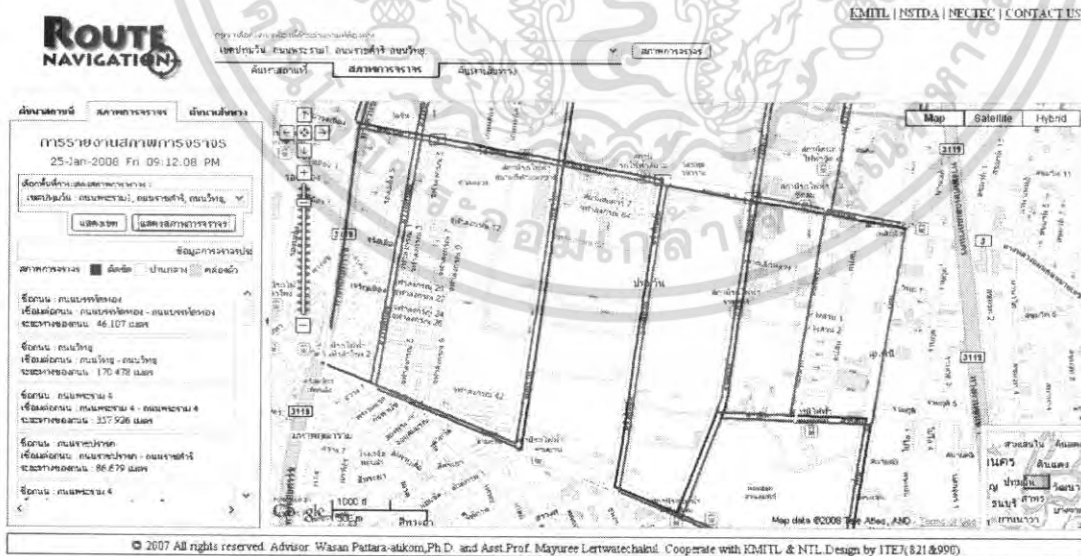
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) คลิกปุ่ม แสดงสภาพจราจร



รูปที่ 43 แสดงการคลิกปุ่ม แสดงสภาพการจราจร

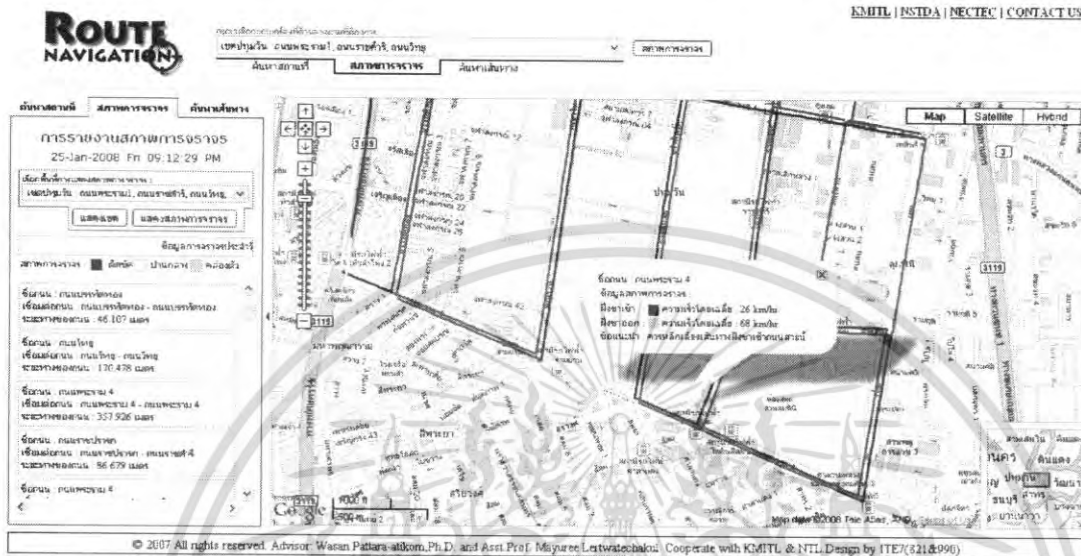
3) แสดงข้อมูลสภาพการจราจรบนแผนที่ Google Maps API



รูปที่ 44 แสดงข้อมูลสภาพการจราจรบนแผนที่ Google Maps API

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4) แสดงข้อมูลสภาวะจราจรของแต่ละช่วงถนนพร้อมคำแนะนำ



รูปที่ 45 แสดงข้อมูลสภาวะจราจรของแต่ละช่วงถนนพร้อมคำแนะนำ

นอกจากนี้ผู้ใช้งานสามารถดูแผนที่ Google Maps API ได้อีก



รูปที่ 46 แสดงการดูแผนที่ Google Maps API

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### สรุปการพัฒนาโครงการ

#### 5.1 บทสรุป

ระบบค้นหาสถานที่สำคัญในกรุงเทพมหานครนั้นสามารถใช้งานได้จริง จากฐานข้อมูลระบบพิกัดกริด (UTM) โดยแสดงผลบนแผนที่ Google Maps API นอกจากนี้ยังสามารถแสดงรายละเอียดของข้อมูลที่ค้นหารวมทั้งเก็บบันทึกข้อมูลที่เคยทำการค้นหาได้ และในอนาคตหากมีข้อมูลเพิ่มมากขึ้นก็สามารถที่จะนำมาประยุกต์ใช้กับระบบนี้ได้

ส่วนในระบบแนะนำเส้นทางที่ใช้ระยะทางสั้นที่สุดนั้น เมื่อลองพิจารณาฐานข้อมูลพิกัดกริด (UTM) สามารถนำไปคำนวณระยะทางที่สั้นที่สุดได้ โดยแสดงผลบนแผนที่ Google Maps API และสามารถแสดงข้อมูลเส้นทางได้ โดยการพัฒนาเรื่องนี้ได้ใช้หลักการ Shortest path first แบบ Dijkstra ' s algorithm มาใช้ในการคำนวณเพื่อหาระยะทางสั้นที่สุด โดยพิจารณาจากจุดเริ่มต้นและจุดสิ้นสุด แต่ไม่ได้คำนึงถึงสภาพจราจร หรือทิศทางการจราจร

ส่วนในระบบแสดงสภาพการจราจรนั้นได้ทำการสมมติข้อมูลสภาพการจราจรในรูปแบบของความเร็วทั้งขาเข้าและขาออก ของ 7 เขต ได้แก่ เขตปทุมวัน, เขตราชเทวี, เขตพญาไท, เขตห้วยขวาง, เขตดินแดง, เขตวัฒนา และเขตคลองเตย โดยสมมติข้อมูลสภาพการจราจรทั้งขาเข้าและขาออกเพื่อนำมาแสดงบนแผนที่ Google Maps API โดยกำหนดให้

ความเร็วช่วง 30 km/hr ถึง 60 km/hr มีสภาพการจราจรเป็น สีแดง

ความเร็วช่วง 61 km/hr ถึง 90 km/hr มีสภาพการจราจรเป็น สีเหลือง

ความเร็วช่วง 90 km/hr ขึ้นไป มีสภาพการจราจรเป็น สีเขียว

นอกจากนี้ยังสามารถแสดงข้อมูลสภาพจราจรในแต่ละช่วงถนนบนแถบด้านข้างเป็นคำแนะนำเพื่อประกอบการตัดสินใจของผู้ใช้ระบบ

ส่วนในการออกแบบหน้าเว็บเพจได้ใช้เทคโนโลยี AJAX เพื่อส่งค่าเฉพาะที่มีการเปลี่ยนแปลงไปเท่านั้น ทำให้ไม่ต้องมีการ refresh ทั้งหน้าเว็บเพจ จึงทำให้การแสดงผลใช้เวลาเร็วขึ้น นอกจากนี้ยังใช้เทคนิค div แทน tag table ของ XHTML ทำให้การแสดงผลยิ่งเร็วขึ้น

## 5.2 ปัญหาที่ยังไม่สามารถแก้ไขได้

- 1) ในระบบค้นหาสถานที่ สามารถค้นหาสถานที่ที่มีอยู่บนฐานข้อมูลเท่านั้นไม่สามารถค้นหาสถานที่อื่นๆได้
- 2) ในระบบค้นหาเส้นทาง โดยคำนวณเส้นทางที่ใช้ระยะทางสั้นที่สุด ไม่ได้พิจารณาข้อมูลทิศทางการจราจร เพราะไม่มีข้อมูลทิศทางการจราจร
- 3) ในระบบแสดงสภาพจราจร สามารถแสดงสภาพจราจรทั้งขาเข้าและขาออกได้เพียงในเขต 7 ตัวอย่างเท่านั้น เพราะไม่มีข้อมูลสภาพการจราจรทั้งหมด

## 5.3 แนวทางในการพัฒนาต่อ

- 1) เพิ่มประสิทธิภาพของระบบแนะนำเส้นทางที่ใช้ระยะทางสั้นที่สุด โดยพิจารณาทิศทางการจราจร
- 2) คำนวณหาเส้นทางที่ดีที่สุดในการเดินทางโดยคำนึงถึงระยะทางในการเดินทางและสภาพการจราจร
- 3) เพิ่มข้อมูลสภาพการจราจรเพื่อที่จะสามารถแสดงสภาพการจราจรได้ทุกเขต

## บรรณานุกรม

Michael Purvis, Jeffrey Sambells and Cameron Turner. 2005. Beginning Google Maps Applications with PHP and Ajax. New york. Apress.

ดวงกมล กติบจินดา, กิตติเชษฐ ยิ่งกิจไพบูลย์. 2550. การพัฒนาเว็บด้วย Javascript CSS และ DOM. พิมพ์ครั้งที่ 2 . กรุงเทพฯ. บริษัท เคทีพี คอมพ์แอนด์คอนซัลท์ จำกัด.

[1] <http://www.dgr.go.th/water2006/technique37.html>

[2] <http://iwis.pcd.go.th/IWIS/document/other/postgres.htm>

[3] <http://angsila.compsci.buu.ac.th/~sc443183/Linux/Report.doc>

[4] <http://lecturer.eng.chula.ac.th/fsvskk/gglmap2/get-start-gmap-apiv2.html>

[5]วิชา ศิริธรรมจักร. 2549. Web programming ด้วย Ajax และ PHP. พิมพ์ครั้งที่ 2. กรุงเทพฯ. บริษัท เคทีพี คอมพ์แอนด์คอนซัลท์ จำกัด.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## คู่มือการติดตั้ง

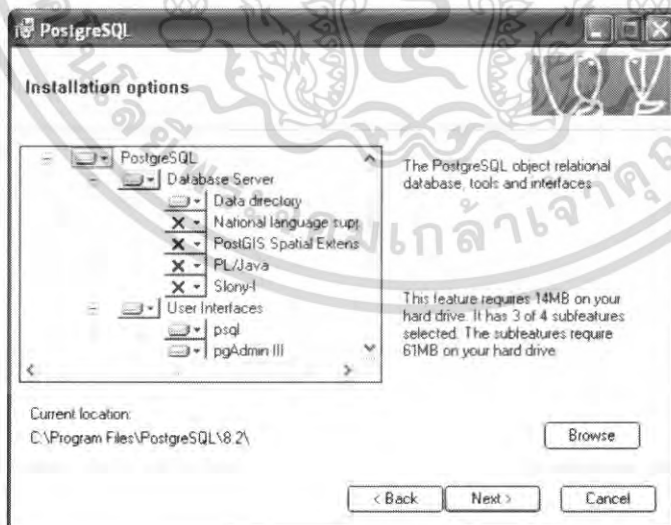
### 1. Postgresql

PostgreSQL 8.2.0-1 (ไม่จำเป็นต้องติดตั้งหากได้ติดตั้งเวอร์ชันเดียวกันแล้ว)

ไฟล์ \software\exe\PostgreSQL\PostgreSQL\postgresql-8.2.msi

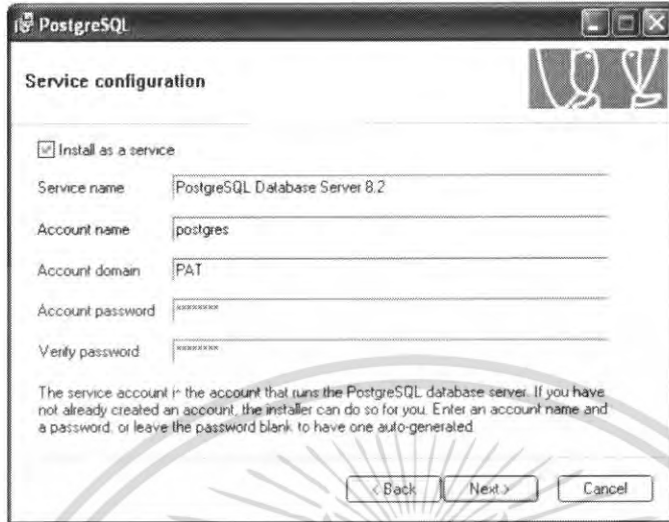


เลือกภาษาที่ใช้ติดตั้ง English

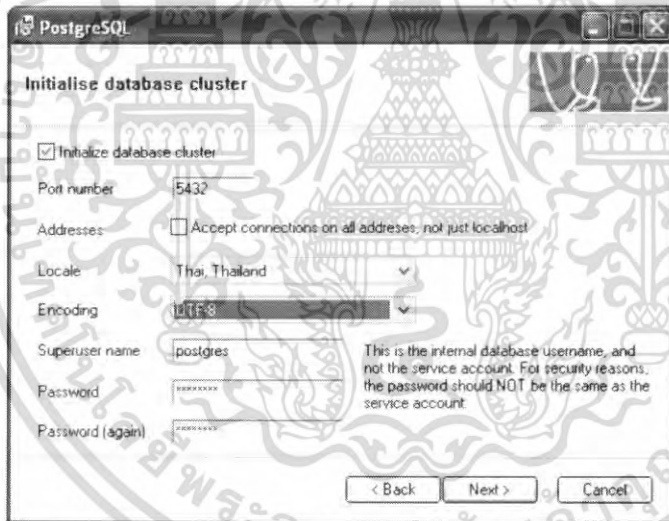


เลือก Next (ไม่มีการแก้ไขค่าตั้งต้น)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



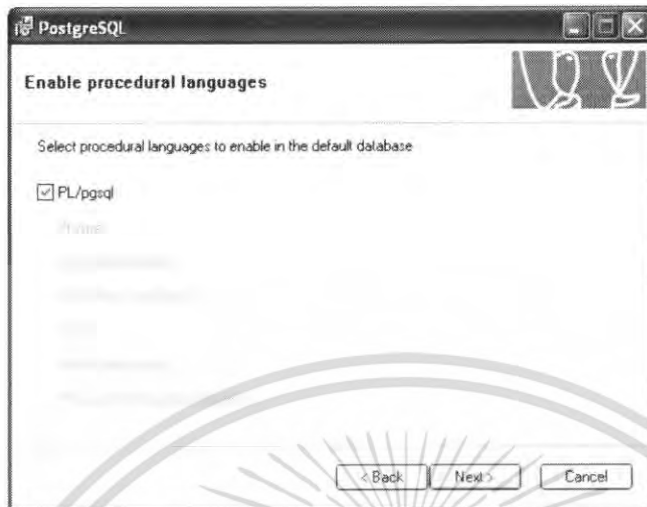
ตั้ง password สำหรับ postgres (ตั้ง "postgres")



ตั้ง Locale = Thailand, ตั้ง Encoding = UTF-8

ตั้ง username และ password สำหรับ superuser ชื่อ postgres (ตั้ง "postgres")

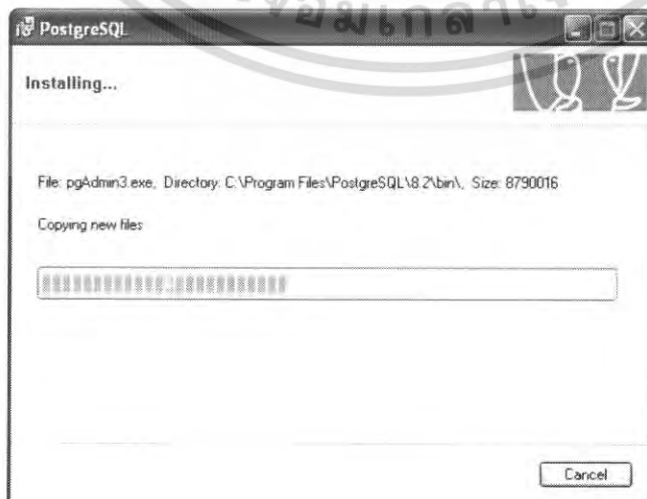
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เลือก PL/pgsql เลือก next



เลือก next (ไม่มีการแก้ไขค่า default)

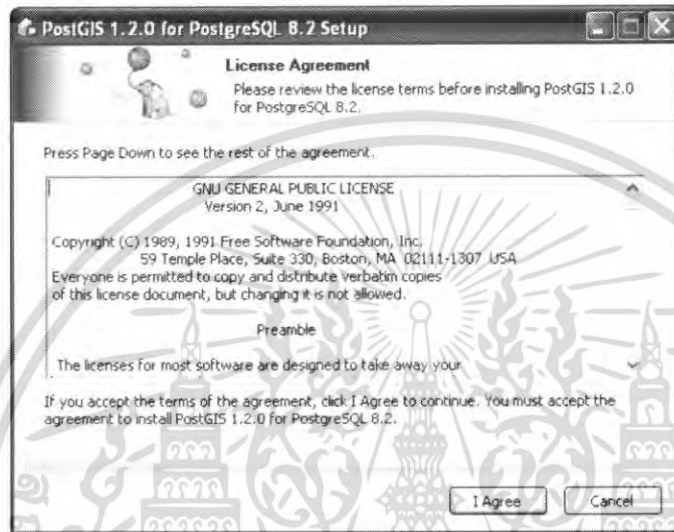


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับธุรกิจส่วนตัวเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2. Postgis

PostGIS 1.2.0-1 (ไม่จำเป็นต้องติดตั้งหากได้ติดตั้งเวอร์ชันเดียวกันแล้ว)

ไฟล์ \software\exe\PostgreSQL\PostGISpostgis-pg82-setup-1.2.0-1.exe

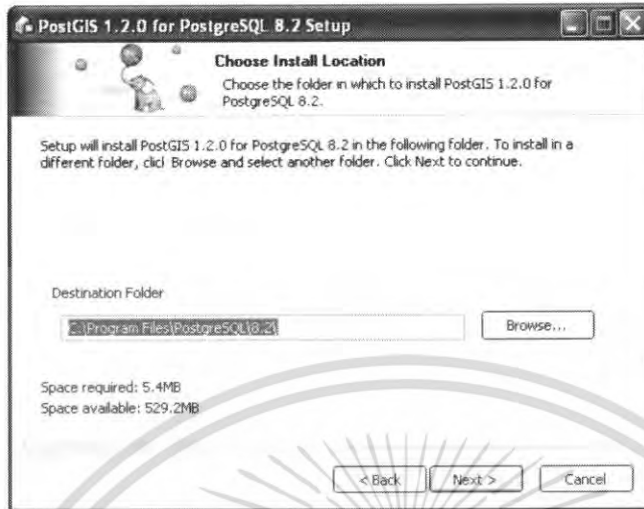


License Agreement เลือก next



เลือก Next (ไม่มีการแก้ไขค่า default)

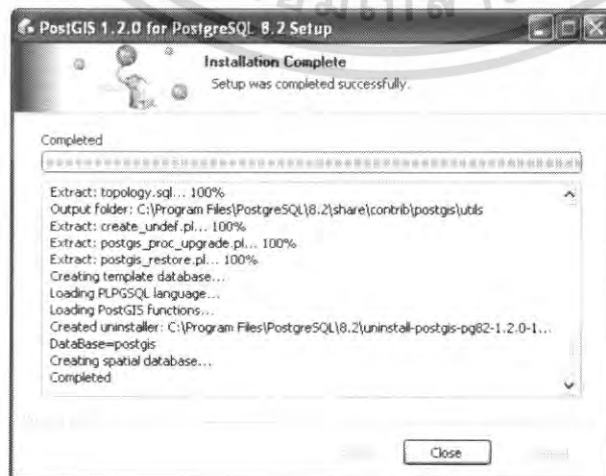
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เลือก Next (ไม่มีกรแก้ไขค่า default)



ตั้งค่าเพื่อการเชื่อมต่อ เลือก next



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับเสร็จสิ้นการติดตั้งโปรแกรมเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

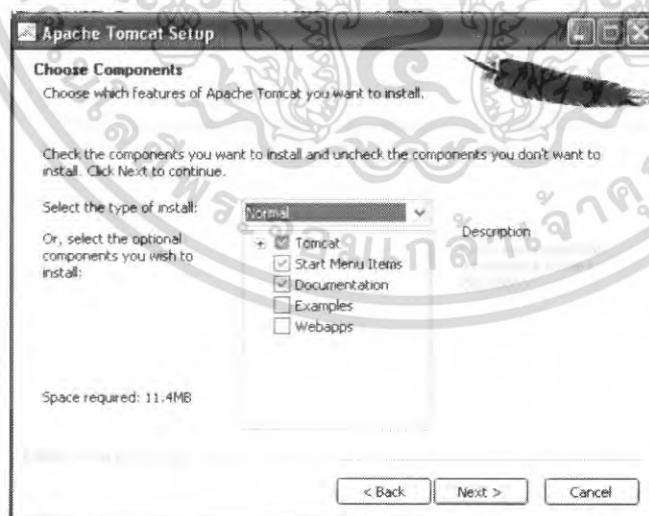
### 3. Apache Tomcat

Apache Tomcat 5.5.17 (ถ้าติดตั้ง Web server อื่นแล้วสามารถใช้แทนได้ทันที ไม่มีความจำเป็นต้องติดตั้ง)

ไฟล์ \software\exe\ApacheTomcat\apache-tomcat-5.5.17.exe

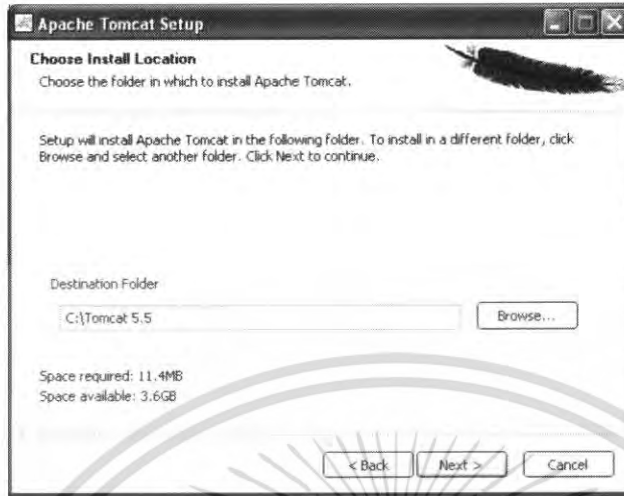


หน้าจอเริ่มแรก



เลือก Normal โหมด

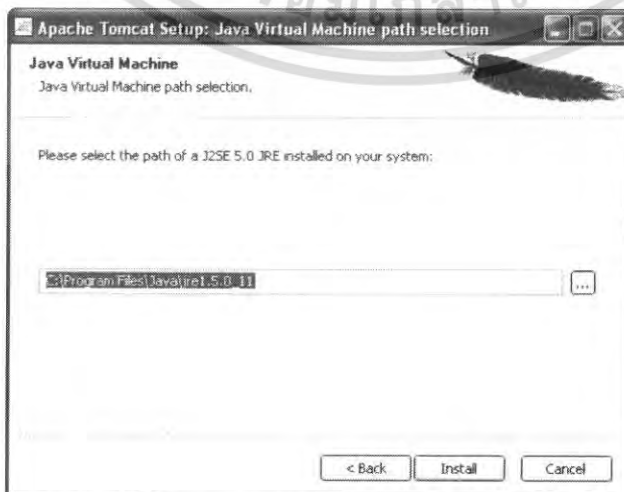
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



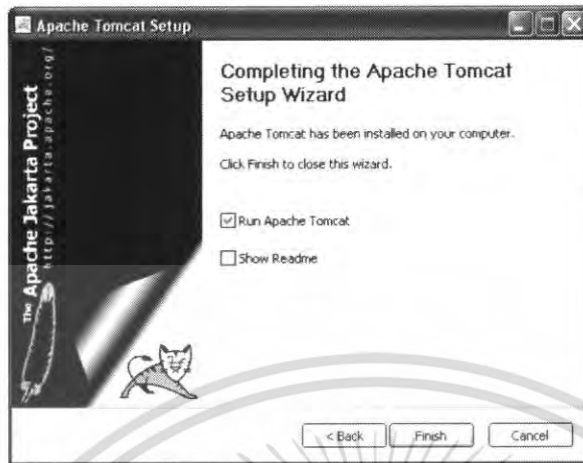
เลือก directory C:\Tomcat 5.5



เลือก Port 8080 และตั้งค่า Password



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรณีฉุกเฉินเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
BrowSe หากสถานที่ที่ติดตั้ง JRE ไว้แล้ว  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

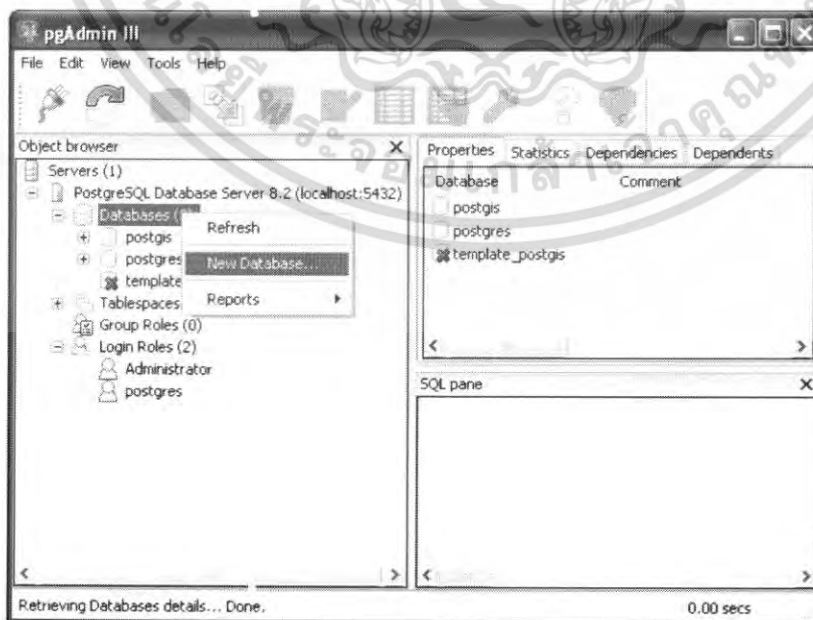


สิ้นสุดการติดตั้งโปรแกรม

#### 4. การติดตั้งฐานข้อมูลตัวอย่าง



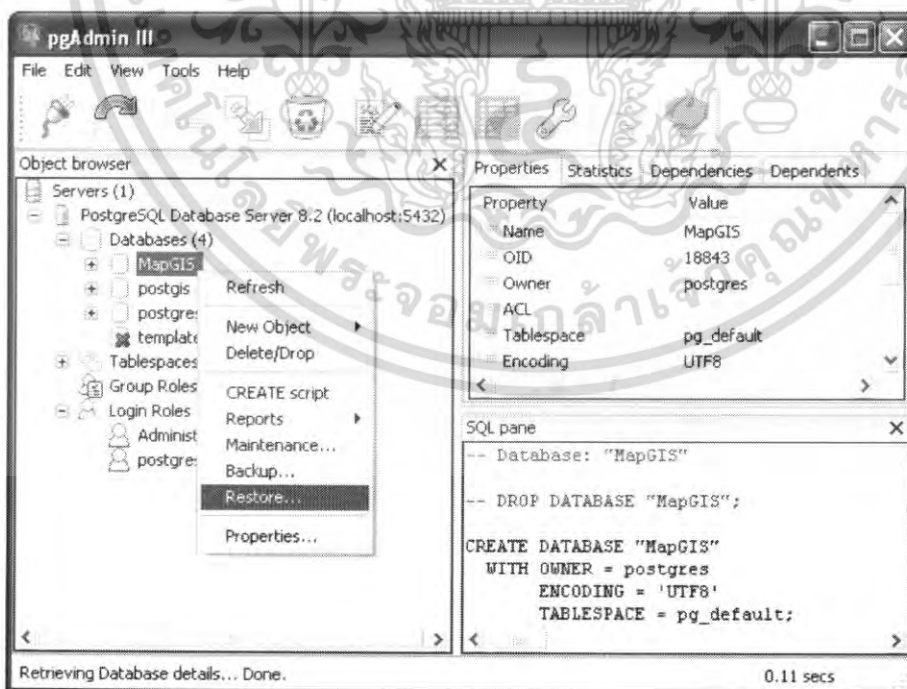
เปิดโปรแกรม pgAdmin III บน Start > All Programs > PostgreSQL 8.2 > pgAdmin III



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
**สร้างฐานข้อมูลใหม่**  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

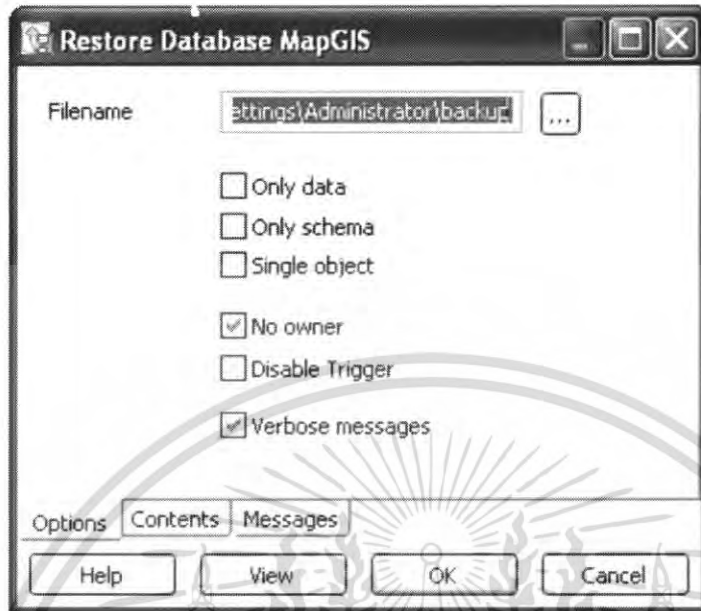


ตั้งชื่อฐานข้อมูลใหม่ว่า “MapGIS” และเลือก Encoding = UTF-8

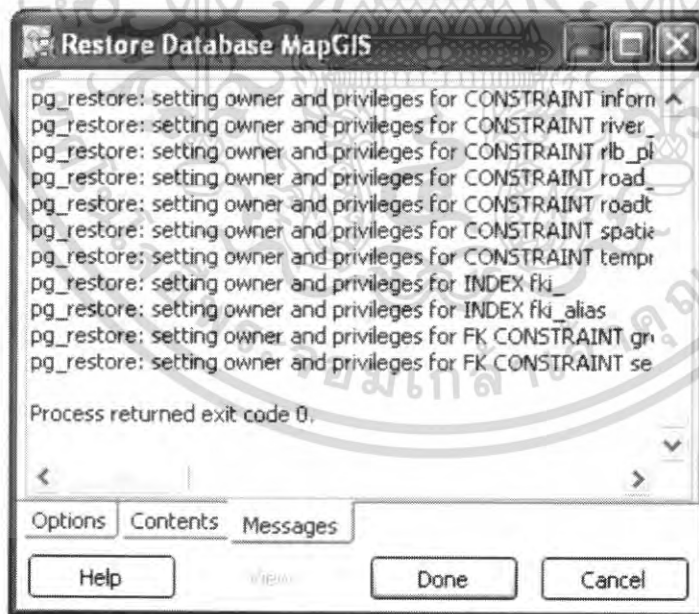


คลิกขวาที่ฐานข้อมูลอันใหม่แล้วเลือก Restore

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เลือกไฟล์ /DatabaseTemplate/backup ทำเครื่องหมายที่หน้า Noowner, Disable Trigger และ Vebose messages



รอนจนเสร็จสิ้นการทำงาน

(หากทำการ restore ฐานข้อมูลข้ามเครื่องอาจเกิด error ทางเทคนิค ขึ้นได้

แต่ไม่มีผลกระทบต่อ schema และ data)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 5. XAMPP server

xampp-win32-1.6.4-installer (ไม่จำเป็นต้องติดตั้งหากได้ติดตั้งเวอร์ชันเดียวกันแล้ว)

ไฟล์ \software\exe\xampp-win32-1.6.4-installer.exe

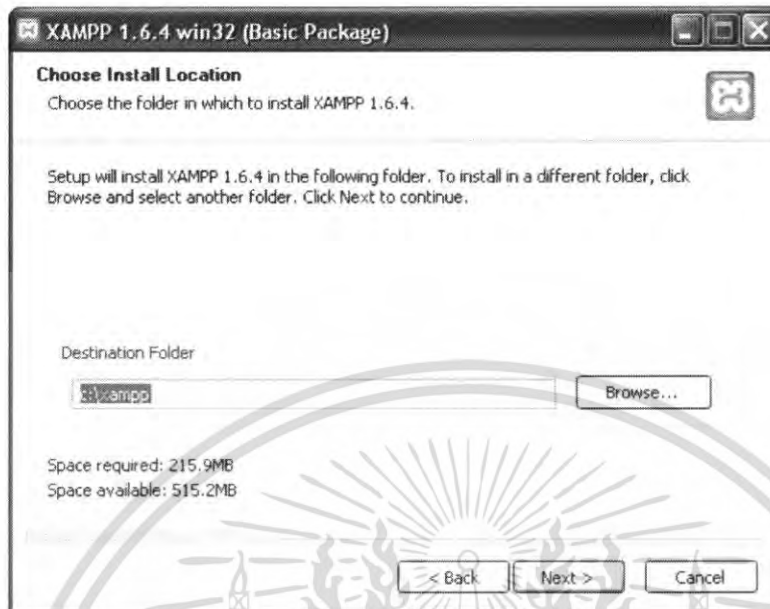


เลือกภาษาที่ต้องการลง

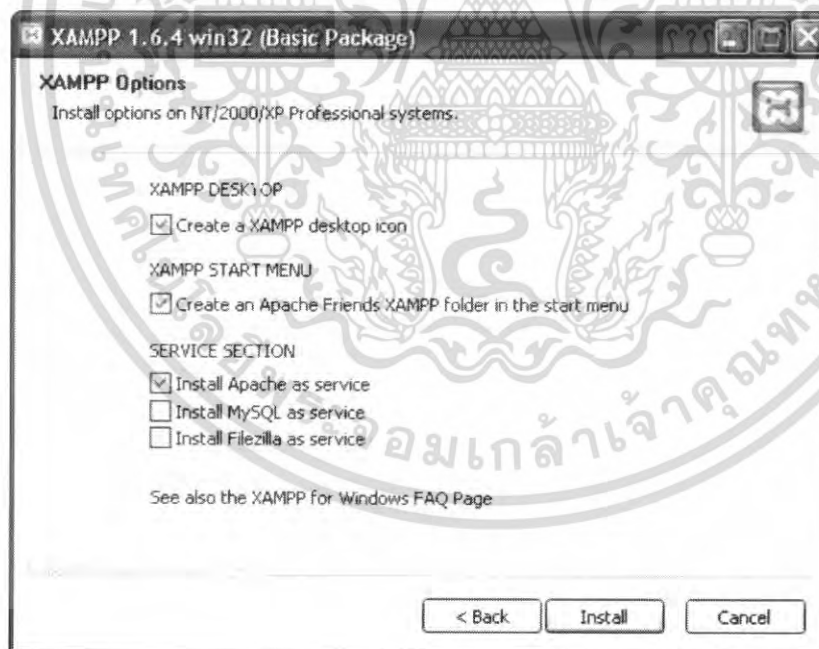


รายละเอียดเกี่ยวกับ โปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เลือก Destination Folder ที่ทำการลงโปรแกรม

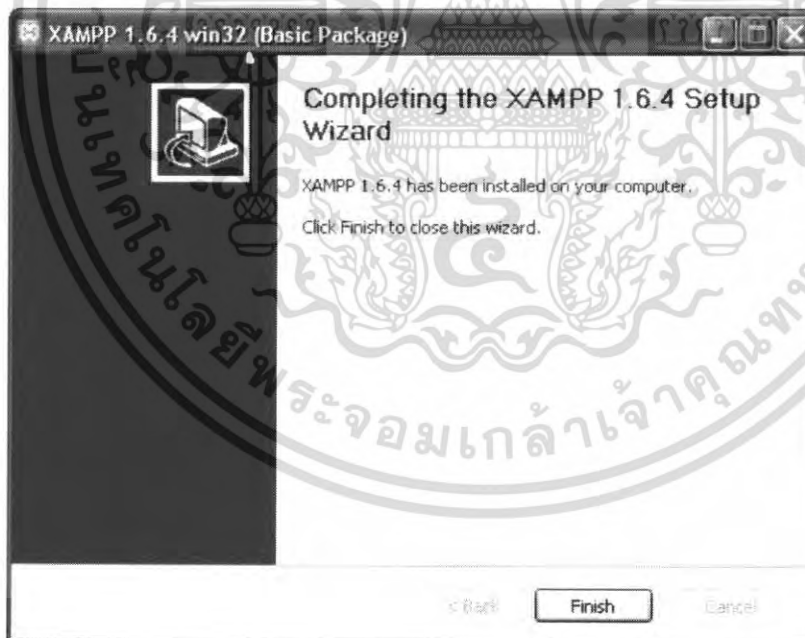


เลือก SERVICE SECTION ที่ต้องการลง ในที่นี้ต้องการ Apache service

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

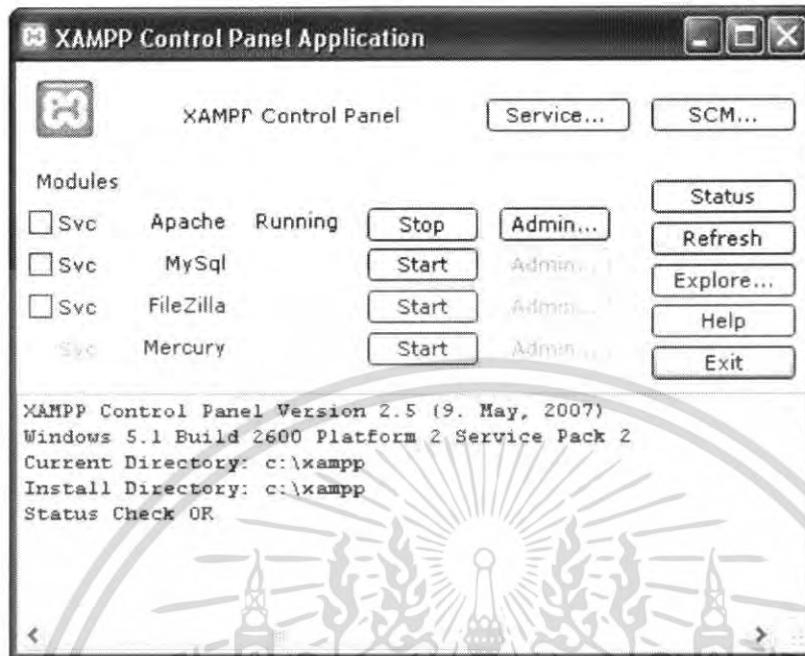


รอการติดตั้ง XAMPP-1.6.4 win32 ลงใน Destination Folder



เมื่อลงโปรแกรมเสร็จแล้ว จะพบหน้าต่างแสดงดังรูปด้านบน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



หน้าต่าง Control Panel แสดงการทำงานของโปรแกรม XAMPP



ทำการทดสอบการใช้งานจาก Firefox browser โดยพิมพ์ <http://localhost/> จะปรากฏดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้