สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบวางแผนการเดินทางบนโทรศัพท์มือถือ

MOBILE BANGKOK NAVIGATION SYSTEM

T104283

โดย

นายรชต	เจนจิต	48010719
นายวสิพล	ว่องเจริญลาภ	48010795
นายวัชระ	เลิศวณิชกุล	48010799
นายวิรัชพงษ์	จันทพันธ์	48010832

เลขหมู่.....
เลขทะเบียน.....104283
วัน,เดือน,ปี.....30 มี.ค. 2552

.b.....
.i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2551

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MOBILE BANGKOK NAVIGATION SYSTEM

BY

RACHATA	JANJIT	48010719
WASIPOL	WONGCHAREONLAP	48010795
VACHARA	LERSWANICHKUL	48010799
WIRATCHAPONG	CHANTAPAN	48010832

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENT FOR THE DEGREE OF

BACHELOR IN DEPARTMENT OF INFORMATION ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2008

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	ระบบวางแผนการเดินทางบนโทรศัพท์มือถือ		
ชื่อนักศึกษา	นายรชต	เจนจิต	รหัสประจำตัว 48010719
	นายวสิพล	ว่องเจริญลาภ	รหัสประจำตัว 48010795
	นายวัชระ	เลิศวณิชกุล	รหัสประจำตัว 48010799
	นายวิรัชพงษ์	จันทพันธ์	รหัสประจำตัว 48010832
อาจารย์ที่ปรึกษา	ผศ.มยุรี เลิศเวชกุล		
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต		
	สาขาวิศวกรรมสารสนเทศ		
ภาควิชา	วิศวกรรมสารสนเทศ		
ปีการศึกษา	2551		

ปริญญานิพนธ์นี้ได้รับความเห็นชอบจากอาจารย์ที่ปรึกษาเป็นที่เรียบร้อยแล้ว

.....

(ผศ. มยุรี เลิศเวชกุล)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	ระบบวางแผนการเดินทางบนโทรศัพท์มือถือ		
ชื่อนักศึกษา	นายรชต	เจนจิต	รหัสประจำตัว 48010719
	นายวสิพล	ว่องเจริญลาภ	รหัสประจำตัว 48010795
	นายวัชระ	เลิศวณิชกุล	รหัสประจำตัว 48010799
	นายวิรัชพงษ์	จันทพันธ์	รหัสประจำตัว 48010832
อาจารย์ที่ปรึกษา	ผศ.มยุรี เลิศเวชกุล		
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต		
	สาขาวิศวกรรมสารสนเทศ		
ภาควิชา	วิศวกรรมสารสนเทศ		
ปีการศึกษา	2551		

บทคัดย่อ

โครงงานนี้เป็นระบบค้นหาเส้นทางอัจฉริยะภายเขตในกรุงเทพมหานคร เป็นแอพพลิเคชั่นที่พัฒนาเพื่อใช้งานบน Window Mobile ร่วมกับระบบ GPS โดยอ้างอิงจากสถานที่สำคัญเพื่อกำหนดเป็นสถานที่ต้นทางและสถานที่ปลายทาง และใช้อัลกอริทึมในการหาเส้นทางที่สั้นที่สุดหรือการหาเส้นทางที่คาดว่าจะใช้เวลาในการเดินทางน้อยที่สุด โดยคำนึงถึงตัวแปรระยะทางและสภาพการจราจร ณ เวลานั้น เพื่อประมวลผลหาเส้นทางที่เหมาะสมที่สุด และนำเส้นทางที่ได้มาแสดงผลโดยใช้แผนที่พร้อมทั้งวาดเส้นแสดงเส้นทางอย่างชัดเจน เพื่อให้ผู้ใช้สามารถเลือกเส้นทางที่จะเดินทางได้อย่างรวดเร็ว เพื่อเป็นการลดระยะเวลาในการเดินทาง และประหยัดพลังงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Mobile Bangkok Navigation System		
Student	Rachata	Jenjit	48010719
	Wasipol	Wongchareonlap	48010795
	Vachara	Lerswanichkul	48010799
	Wiratchapong	Chantapan	48010832
Advisor	Asst. Prof. Mayuree Lertwatechakul		
Graduate Level	Bachelor Degree of Information Engineering		
Department	Information Engineering		
Academic Year	2008		

ABSTRACT

This project proposes Mobile Bangkok Navigation System for Bangkok area. The application is developed to be used on Window Mobile cooperating with GPS system. The beginning and destination points could be marked the interesting place to calculate for the shortest distance route or shortest time route depending on distance and traffic condition by the time. The result is shown as a high-light line onto the Bangkok map. The advantage of the system is the user can choose the appropriate route as to save time and energy.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้จะสำเร็จลุล่วงด้วยดีมิได้ หากไม่มีผู้คอยสนับสนุนอันได้แก่ ผศ.มยุรี เลิศเวชกุล ซึ่งคอยให้คำแนะนำและสิ่งต่าง ๆ ที่เป็นประโยชน์ ขอบคุณพี่ ๆ ทุกคนที่ให้คำปรึกษาที่เป็นประโยชน์ รวมถึงเพื่อน ๆ ทุกคนที่คอยเป็นกำลังใจและสนับสนุนเป็นอย่างดี นอกจากนี้ต้องขอบพระคุณบิดามารดาที่คอยสนับสนุนและเป็นกำลังใจในทุก ๆ ด้านและคอยเป็นกำลังใจในทุก ๆ ด้านและที่ขาดมิได้ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังที่ให้การสนับสนุนให้โอกาสในการแสดงออกถึงความรู้และความสนใจที่ได้ศึกษาและค้นคว้ามาตลอดระยะเวลา 4 ปี ขอให้คุณประโยชน์อันใดที่จะสามารถพึงเกิดขึ้นได้ของปริญญานิพนธ์ฉบับนี้ ได้ส่งผลกลับไปถึงผู้มีพระคุณทุก ๆ ท่าน

ผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญรูปภาพ	ฉ
สารบัญตาราง	ช
บทที่ 1 บทนำ	
1.1 บทนำ	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตของโครงงาน	2
1.4 สถาปัตยกรรมระบบ	3
1.4.1 การทำงานในส่วนผู้ใช้ระบบ	3
1.4.2 การทำงานในส่วนเซิร์ฟเวอร์	3
1.5 ขั้นตอนการดำเนินโครงงาน	5
บทที่ 2 ทฤษฎีและหลักการ	
2.1 ความรู้ทั่วไปเกี่ยวกับระบบพิกัดจุด	6
2.1.1 ระบบพิกัดจุด (Coordinate System)	
2.2 เทคนิคการหาเส้นทางที่สั้นที่สุด(Shortest-Route Problem)	8
2.2.1 การแทนรูปแบบโครงสร้างกราฟแบบมีทิศทาง	10
2.2.2 ปัญหาทางเดินที่สั้นที่สุดแบบทางเดียว	10
2.3 อัลกอริทึมแบบ Dijkstra's algorithms	15
2.4 ความรู้ทั่วไปเกี่ยวกับฐานข้อมูลใน ProgreSQLServer	18
2.5 ความรู้ทั่วไปเกี่ยวกับ PostGIS	19
2.6 ความรู้ทั่วไปเกี่ยวกับ GIS	19
2.7 เทคโนโลยี Google Maps API	20
2.8 GLOBAL POSITIONING SYSTEM (GPS)	21
2.9 Pocket PC	23
2.10 .NET Framework	23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

เรื่อง	หน้า
2.11 PHP Web Service	23
2.12 Opera Mobile	24
บทที่ 3 การออกแบบโครงงาน	
3.1 โครงสร้างตารางฐานข้อมูล (Data Dictionary)	26
3.1.1 ฐานข้อมูลของระบบค้นหาสถานที่และระบบค้นหาเส้นทาง	26
3.2 การออกแบบระบบ	32
3.2.1 Use Case Diagram	32
3.2.2 Activity Diagram	33
3.2.3 System Flowchart	38
บทที่ 4 ผลการทดลอง	
4.1 ขั้นตอนการใช้งานโปรแกรม	
4.1.1 ฟังก์ชันการใช้งาน PLACE	41
4.1.2 ฟังก์ชันการใช้งาน ROUTE	44
4.1.3 ฟังก์ชันการใช้งาน TRAFFIC	48
4.1.4 ฟังก์ชันการใช้งาน INFO	53
4.1.5 ฟังก์ชันการใช้งาน GPS	54
บทที่ 5 สรุปผลการทดลอง	
5.1 สรุปการพัฒนาโครงงาน	56
5.2 ปัญหาที่พบในการทำโครงงาน	56
5.3 ข้อจำกัดของโครงงาน	57
5.4 แนวทางในการพัฒนาต่อ	57
บรรณานุกรม	58

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

รูป **หน้า**

รูปที่ 1.1 สถาปัตยกรรมของระบบ 3
รุปที่ 2.1 การแบ่งกริดโซนระบบพิกัดกริด UTM 8
รูปที่ 2.2 ความสัมพันธ์ของอาร์ค 8
รูปที่ 2.3 กราฟแบบมีทิศทางที่มี 4 เวอร์ทีซ 9
รูปที่ 2.4 กราฟที่กำหนดความสัมพันธ์ของอาร์ค 9
รูปที่ 2.5 เมตริกซ์ประชิดของรูปที่ 5 10
รูปที่ 2.6 กราฟที่มีทิศทางที่มีค่าบนอาร์ค 12
รูปที่ 2.7 ข้อสมมติฐานของการหาระยะทางที่สั้นที่สุดไปยัง w 13
รูปที่ 2.8 การเดินทางที่สั้นที่สุดที่เป็นไปได้ 14
รูปที่ 2.9 การสร้างกราฟจากโหนด 16
รูปที่ 2.10 วิธีการคำนวณอัลกอริทึม Dijkstra 17
รูปที่ 2.11 เส้นทางจากถนนหมายเลข 0 ไปยังทุก ๆ ถนน 18
รูปที่ 3.1 Use case diagram ของระบบ 32
รูปที่ 3.2 Activity diagram ของ การค้นหาสถานที่ 33
รูปที่ 3.3 Activity diagram ของ การคำนวณระยะทางที่สั้นที่สุด 34
รูปที่ 3.4 Activity diagram ของ การคำนวณเส้นทางที่ใช้เวลาน้อยที่สุด 35
รูปที่ 3.5 Activity diagram ของการดูแผนที่ 36
รูปที่ 3.6 Activity diagram ของการแสดงสภาพจราจร 37
รูปที่ 3.7 ระบบการไหลของข้อมูลการทำงานการหาสถานที่ 38
รูปที่ 3.8 ระบบการไหลของข้อมูลการทำงานของระบบคำนวณเส้นทางที่สั้นที่สุด 39
รูปที่ 3.9 ระบบการไหลของข้อมูลการทำงานของระบบคำนวณเส้นทางที่ใช้เวลาน้อยที่สุด 40
รูปที่ 4.1 – 4.6 ฟังก์ชันการใช้งาน PLACE 41
รูปที่ 4.7 – 4.14 ฟังก์ชันการใช้งาน ROUTE 44
รูปที่ 4.15 – 4.23 ฟังก์ชันการใช้งาน TRAFFIC 48
รูปที่ 4.24 – 4.25 ฟังก์ชันการใช้งาน INFO 53
รูปที่ 4.26 – 4.29 ฟังก์ชันการใช้งาน GPS 54

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตาราง **หน้า**

ตารางที่ 2.1 ข้อมูลของตาราง Dijkstra's algorithm 12

ตารางที่ 3.1 ข้อมูลของตาราง mainroad 26

ตารางที่ 3.2 ข้อมูลของตาราง educated 27

ตารางที่ 3.3 ข้อมูลของตาราง hospital 27

ตารางที่ 3.4 ข้อมูลของตาราง hotel 27

ตารางที่ 3.5 ข้อมูลของตาราง police 28

ตารางที่ 3.6 ข้อมูลของตาราง temple 28

ตารางที่ 3.7 ข้อมูลของตาราง govern 28

ตารางที่ 3.8 ข้อมูลตาราง mainroad_in 29

ตารางที่ 3.9 ข้อมูลตาราง mainroad_out 30

ตารางที่ 3.10 ข้อมูลตาราง traffic_dens 31

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 บทนำ

ปัญหาการคมนาคมทางบกนับเป็นปัญหาสำคัญของประชากรที่อาศัยอยู่ในกรุงเทพ ฯ มาตลอด สืบเนื่องมาจากการที่เป็นเมืองหลวงมีประชากรอาศัยอยู่อย่างหนาแน่น อีกทั้งยังเป็นศูนย์กลางทางด้านเศรษฐกิจของประเทศจึงทำให้การเดินทางรีบเร่งตลอดเวลา ที่ผ่านมารัฐบาลเองก็พยายามช่วยแก้ไขปัญหานี้ โดยการสนับสนุนระบบขนส่งมวลชน อาทิเช่น รถเมล์ รถโดยสาร รถไฟฟ้าบนดินและใต้ดิน รวมถึงการรณรงค์การเดินทางเส้นเดียวกันให้ไปด้วยกันเป็นต้น แต่ถึงอย่างไรความจำเป็นในการใช้รถยนต์ส่วนบุคคลก็ยังคงมีจำนวนมากอยู่ดี เนื่อง จากปัจจัยต่าง ๆ ที่ไม่ว่าจะเป็นความต้องการความสะดวกสบายในการเดินทางที่ไม่ต้องทนกับอากาศร้อน ความแออัดที่ต้องเผชิญในรถบริการขนส่งมวลชน สามารถเดินทางตามเส้นทางที่ตนเองต้องการได้ การแสดงฐานะหน้าที่การงานของตนและโปรโมชั่นของบริษัทผู้ผลิตรถยนต์ต่าง ๆ ที่สามารถนำรถยนต์ออกมาใช้งานได้ก่อนเวลาที่จะชำระเงินจนครบ ซึ่งทำให้เกิดปัญหาสภาพการจราจรไม่คล่องตัว ซึ่งทางผู้จัดทำได้เล็งเห็นถึงปัญหาและพยายามช่วยลดปัญหาที่เกิดขึ้น โดยการพัฒนาระบบวางแผนการเดินทางอัจฉริยะบนโทรศัพท์มือถือ โดยใช้ผ่าน Mobile Application ที่สะดวกและใช้งานง่ายเพื่อช่วยในการวางแผนการเดินทางแก่ผู้ใช้ยานพาหนะบนท้องถนน

1.2 วัตถุประสงค์

1) เพื่อค้นหาสถานที่ในเขตกรุงเทพมหานคร
2) เพื่อแนะนำเส้นทางการเดินทางในเขตกรุงเทพมหานครที่ใช้ระยะทางสั้นที่สุดหรือเส้นทางที่คาดว่าจะใช้เวลาน้อยที่สุด โดยพิจารณาการเดินทางระหว่างจุดเริ่มต้น และจุดปลายทาง
3) มีการนำเทคนิคการหาระยะทางสั้นที่สุด (Shortest path first) มาประยุกต์กับโครงข่ายของเส้นทางการจราจรในกรุงเทพมหานคร
4) เพื่อให้ผู้ใช้รถยนต์บนท้องถนนได้ใช้เส้นทางบนท้องถนนและทรัพยากรน้ำมันได้ อย่างมีประสิทธิภาพ
5) เพื่อลดปัญหามลพิษทางอากาศและปัญหาต่าง ๆ ที่ส่งผลกระทบกับสภาพแวดล้อมและประชากรที่อาศัยในกรุงเทพมหานคร
6) สามารถอำนวยความสะดวก ในการใช้งานด้วยอุปกรณ์สื่อสารขนาดพกพาได้อย่างมีประ-สิทธิภาพ
7) สามารถนำไปใช้ในการพัฒนาระบบสอบถามเส้นทางในอนาคตทั้งในภาครัฐและเอกชน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของโครงงาน

ระบบสามารถค้นหาสถานที่สำคัญในกรุงเทพมหานคร ได้ทำการกำหนดขอบเขตของปัญหาไว้ว่าจะ พิจารณาสถานที่ที่มีอยู่ในฐานข้อมูลและในแผนที่ Google Maps API เท่านั้น

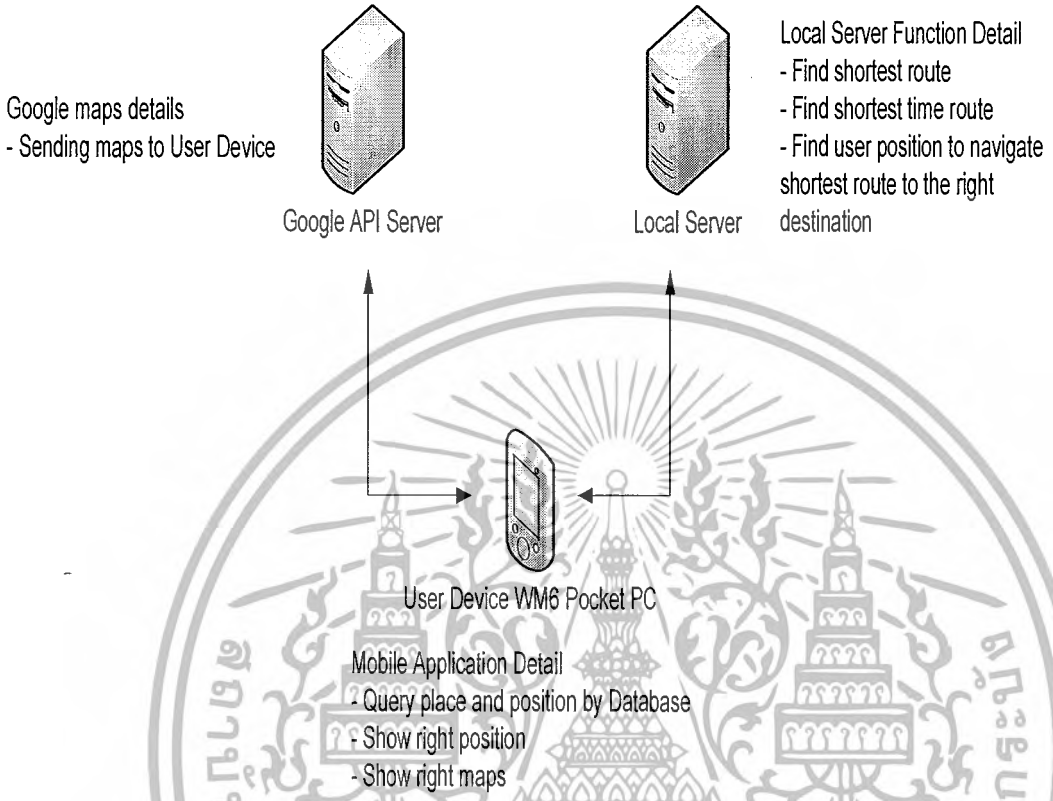
ระบบสามารถแนะนำเส้นทางที่สั้นที่สุดใช้ในการเดินทางระหว่างจุดสถานที่เริ่มต้นและจุดสถานที่ปลายทาง โดยในที่นี้เราจะพิจารณาเพียงระยะทางที่สั้นที่สุดในการเดินทาง ซึ่งได้กำหนดขอบเขต ของปัญหาไว้ดังนี้

1) การเดินทางที่กล่าวถึงในที่นี้จะพิจารณาเพียงเส้นทางการจราจรโดยการใช้รถยนต์เท่านั้น
2) สถานที่อ้างอิงจะพิจารณาระยะห่างระหว่างสถานที่และถนนใหญ่เท่านั้น ไม่ได้พิจารณาระยะทางที่เดินทางเข้าไปยังในซอยที่สถานที่อ้างอิงตั้งอยู่จริง
3) เส้นทางที่ใช้ในการพิจารณา จะพิจารณาจากเส้นทางจริงที่มีอยู่ในฐานข้อมูล
4) พิกัด GPS บนเครื่องถูกอ่านค่ามาจากไฟล์ด NMEA เนื่องจากอุปกรณ์ไม่ได้ถูกติดตั้ง GPS มาด้วย

ส่วนในระบบแสดงสภาพการจราจรจะแสดงข้อมูลสภาพการจราจร โดยผ่านข้อมูลจากป้ายจราจรอัจฉริยะ ซึ่งปัจจุบันมีป้ายจราจรอัจฉริยะทั้งหมด 40 ป้าย ทั่วกรุงเทพมหานครเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 สถาปัตยกรรมของระบบ



รูปที่ 1.1 สถาปัตยกรรมของระบบ

1.4.1 การทำงานในส่วนผู้ใช้ระบบ

ฝั่งผู้ใช้ระบบจะใช้โปรแกรม Mobile Application ร่วมกับการทำงานของโปรแกรมเว็บบราวเซอร์ ซึ่งจะร้องขอข้อมูลจาก server ผ่านทางโปรโตคอล HTTP โดยผ่าน Opera Web browser เพราะ Application ที่สร้างขึ้นนี้ต้องการการสนับสนุนด้วยเทคโนโลยี AJAX และ Java Script จึงทำให้ไม่สามารถใช้เว็บบราวเซอร์มาตรฐานทั่วไปได้

1.4.2 การทำงานในส่วนเซิร์ฟเวอร์ (Server)

สามารถแบ่งการทำงานได้ดังนี้

1.4.2.1 ระบบค้นหาสถานที่สำคัญในกรุงเทพมหานคร

1) ผู้ใช้ระบบป้อนสถานที่ที่ต้องการค้นหา
2) ระบบนำค่าที่ผู้ใช้ป้อนไปค้นหาพิกัดจุดของสถานที่ในฐานข้อมูล
3) เมื่อได้พิกัดจุดแล้ว นำไปแสดงจุดในแผนที่ Google Maps API
4) แสดงผลที่ได้ผ่านหน้า Application

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4.2.2 ระบบแนะนำเส้นทางที่มีระยะทางสั้นที่สุด

1) ก่อนเริ่มการทำงาน ระบบจะทำการเรียกข้อมูลจากฐานข้อมูล PostgreSQL โดยจะนำข้อมูลส่วนของค่า Cost ของถนนสายต่าง ๆ ถนนถัดไปที่ถนนเส้นกำลังพิจารณาอยู่และระยะทางของถนนแต่ละเส้น
2) เมื่อผู้ใช้บริการกรอกข้อมูลสถานที่จุดเริ่มต้นและสถานที่ปลายทางเข้ามายังระบบ ระบบจะทำการประมวลผลเพื่อหาระยะทางที่สั้นที่สุดโดยพิจารณาจากระยะทางโดยใช้อัลกอริทึม Dijkstra
3) เมื่อโปรแกรมทำการประมวลผลเสร็จเรียบร้อยแล้ว แสดงเส้นทางบนแผนที่ Google Maps API (Static Maps)

1.4.2.3 ระบบแนะนำเส้นทางที่คาดว่าจะใช้เวลาน้อยที่สุด

1) ก่อนเริ่มการทำงาน ระบบจะทำการเรียกข้อมูลจากฐานข้อมูล PostgreSQL โดยจะนำข้อมูลส่วนของค่า Cost ของถนนสายต่าง ๆ ถนนถัดไปที่ถนนเส้นกำลังพิจารณาอยู่และระยะทางของถนนแต่ละเส้นมาทำการคูณกับ traffic ratio แล้วนำค่าที่ได้เก็บในฐานข้อมูลใหม่ เพื่อใช้เป็นค่า cost ในการคิด
2) เมื่อผู้ใช้บริการกรอกข้อมูลสถานที่จุดเริ่มต้นและสถานที่ปลายทางเข้ามายังระบบ ระบบจะทำการประมวลผลเพื่อหาเส้นทางที่คาดว่าจะใช้เวลาน้อยที่สุด โดยพิจารณาจากระยะทาง และ สภาพการจราจร
3) เมื่อโปรแกรมทำการประมวลผลเสร็จเรียบร้อยแล้ว แสดงเส้นทางบนแผนที่ Google Maps API (Static Maps)

1.4.2.4 ระบบแสดงสภาพจราจร

1) ผู้ใช้คลิกปุ่ม INFO และเลือกที่ตั้งป้ายจราจรอัจฉริยะเพื่อดูสภาพการจราจร
2) ระบบจะแสดงสภาพจราจรบนป้ายอัจฉริยะผ่าน Application
3) GPS
   1) ใช้ GPS รับค่าที่อยู่ปัจจุบันของผู้ใช้จากดาวเทียม
   2) ระบบจะนำค่าจาก GPS ไปประมวลผลเทียบกับแผนที่ Google Maps
   3) แสดงผลผ่านหน้า Application

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 ขั้นตอนการดำเนินโครงงาน

ขั้นตอนที่ 1 ศึกษาปัญหาระบบงาน กำหนดขอบเขตและจุดประสงค์ต่าง ๆ

ขั้นตอนที่ 2 ศึกษาทฤษฎีที่เกี่ยวข้อง รวมทั้งศึกษาค้นคว้าจากงานวิจัย, ปัญหาพิเศษ และบทความทางวิชาการที่มีเนื้อหาเกี่ยวข้องกัน

ขั้นตอนที่ 3 ศึกษาข้อมูลจากฐานข้อมูล PostgreSQL ที่ได้จากห้องปฏิบัติการระบบเครือข่าย (NTL) ภายใต้ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอม - พิวเตอร์แห่งประเทศไทย (NECTEC)

ขั้นตอนที่ 4 ออกแบบระบบงาน

ขั้นตอนที่ 5 ออกแบบอัลกอริทึม shortest path first โดยใช้อัลกอริทึมแบบ Dijkstra

ขั้นตอนที่ 6 เพิ่มข้อมูลลงฐานข้อมูล

ขั้นตอนที่ 7 ออกแบบหน้า Application ร่วมกับเทคโนโลยี Google Maps API

ขั้นตอนที่ 8 พัฒนาระบบ

ขั้นตอนที่ 9 ตรวจสอบระบบ

ขั้นตอนที่ 10 ทดสอบระบบ

ขั้นตอนที่ 11 จัดทำเอกสาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 ความรู้ทั่วไปเกี่ยวกับระบบพิกัดจุด

2.1.1 ระบบพิกัดจุด (Coordinate System)

เป็นระบบที่สร้างขึ้นสำหรับใช้อ้างอิงในการกำหนดตำแหน่ง หรือบอกตำแหน่งพื้นโลกจากแผนที่มีลักษณะเป็นตารางโครงข่ายที่เกิดจากตัดกันของเส้นตรงสองชุด ที่ถูกกำหนดให้วางตัวแนวเหนือ-ใต้ และแนวตะวันออก-ตะวันตก ตามแนวของจุดศูนย์กำเนิด (Origin) ที่กำหนดขึ้นค่าพิกัดที่ใช้อ้างอิงในการบอกตำแหน่งต่าง ๆ จะใช้ค่าของหน่วยที่นับออกจากจุดศูนย์กำเนิดเป็น ระยะเชิงมุม (Degree) หรือเป็นระยะทาง (Distance) ไปทางเหนือหรือใต้และตะวันออกหรือตะวันตกตามตำแหน่งของตำบลที่ต้องการหาค่าพิกัดที่กำหนดตำแหน่งต่างๆ จะถูกเรียกอ้างอิงเป็น ตัวเลขในแนวตั้งและแนวนอนตามหน่วยวัดระยะ ใช้วัดสำหรับระบบพิกัดที่ใช้อ้างอิงกำหนดตำแหน่งบนแผนที่ที่นิยมใช้กับแผนที่ในปัจจุบัน มีอยู่ด้วยกัน 2 ระบบ คือ

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

2.1.1.1 ระบบพิกัดภูมิศาสตร์ (Geographic Coordinate System)

เป็นระบบพิกัดที่กำหนดตำแหน่งต่าง ๆ บนพื้นโลก ด้วยวิธีการอ้างอิงบอกตำแหน่ง เป็นค่าระยะเชิงมุมของละติจูด (Latitude) และลองติจูด (Longitude) ตามระยะเชิงมุมที่ห่างจากศูนย์กำเนิด (Origin) ของละติจูดและลองติจูด ที่กำหนดขึ้นสำหรับศูนย์กำเนิดของละติจูด (Origin of Latitude) นั้นกำหนดขึ้นจากแนวระดับที่ตัดผ่านศูนย์กลางของโลกและตั้งฉากกับแกนหมุน เรียกแนวระนาบศูนย์กำเนิดนั้นว่าเส้นศูนย์สูตร (Equator) ซึ่งแบ่งโลกออกเป็นซีกโลกเหนือและซีกโลกใต้ ฉะนั้นค่าระยะเชิงมุมของละติจูดจะเป็นค่าเชิงมุมที่เกิดจากมุมที่ศูนย์กลางของโลกกับแนวระดับฐาน กำ - เนิดมุมที่เส้นศูนย์สูตรที่วัดค่าของมุมออกไปทั้งซีกโลกเหนือและซีกโลกใต้ ค่าของมุมจะสิ้นสุดที่ขั้วโลกเหนือและขั้วโลกใต้ มีค่าเชิงมุม 90 องศาพอดี ดังนั้นการใช้ค่าระยะเชิงมุมของละติจูดอ้างอิงบอกตำแหน่งต่าง ๆ นอกจากจะกำหนดเรียกค่าวัดเป็นองศาลิปดาและฟิลิปดา แล้วจะบอกซีกโลกเหนือหรือใต้กำกับด้วยเสมอ

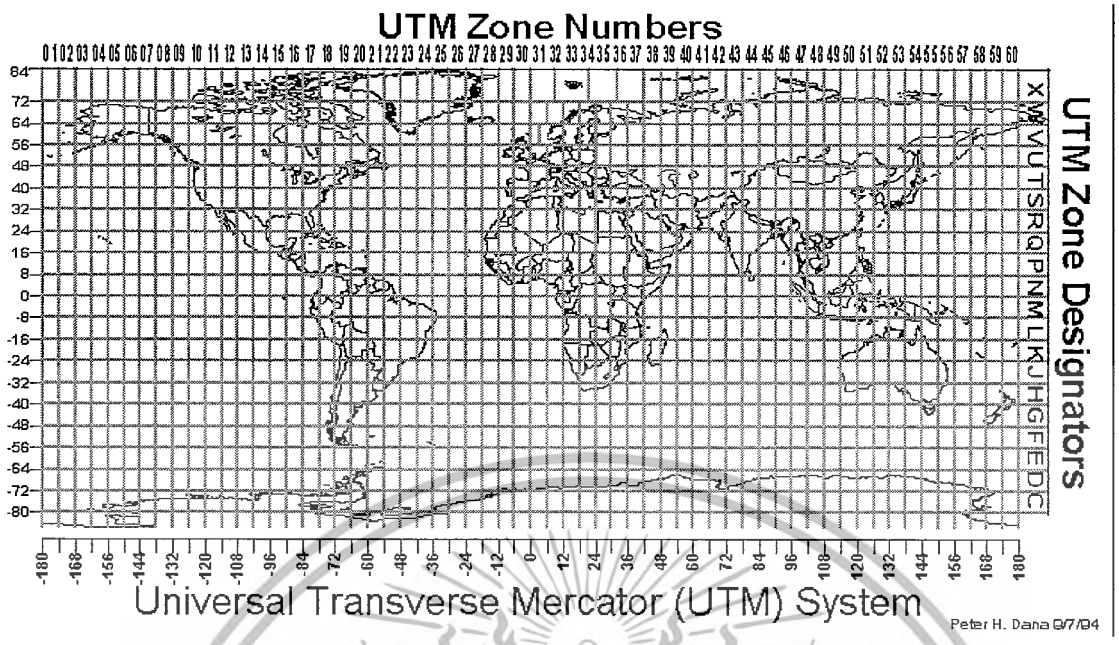
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.1.2 ระบบพิกัดกริดแบบ UTM (Universal Transverse Mercator co-ordinate System)

พิกัดกริด UTM (Universal Transverse Marcator) เป็นระบบตารางกริดที่ใช้ช่วยในการกำหนดตำแหน่งและใช้อ้างอิงในการบอกตำแหน่งที่นิยมใช้กับแผนที่ในกิจการทหารของประเทศต่าง ๆ เกือบทั่วโลกในปัจจุบันเพราะเป็นระบบตารางกริดที่มีขนาดรูปร่างเท่ากันทุกตารางและมีวิธีการกำหนดบอกค่าพิกัดที่ง่ายและถูกต้องเป็นระบบกริดที่นำเอาเส้นโครงแผนที่แบบ Universal Transverse Marcator Projection ของ Gauss – Krueger มาใช้ดัดแปลงการถ่ายทอดรายละเอียดของพื้นผิวโลกให้รูปทรงกระบอก Mercator Projection อยู่ในตำแหน่ง Mercator Projection (แกนของรูปทรงกระบอกจะทับกับแนวเส้นอิเควเตอร์ และตั้งฉากกับแนวแกนของขั้วโลก) ประเทศไทยเราได้นำเอาเส้นโครงแผนที่แบบ UTM นี้มาใช้กับการทำแผนที่ เป็นชุด L 7017 ที่ใช้ในปัจจุบันแผนที่ระบบพิกัดกริดที่ใช้เส้นโครงแผนที่แบบ UTM เป็นระบบเส้นโครงชนิดหนึ่งที่ใช้ผิวรูปทรงกระบอกเป็นผิวแสดงเส้นเมริเดียน (หรือเส้นลองติจูด) และเส้นละติจูดของโลก โดยใช้ทรงกระบอกตัดโลกระหว่างละติจูด 84 องศาเหนือและ 80 องศาใต้ในลักษณะแกนรูปทรงกระบอก แล้วทำมุมกับแกนโลก 90 องศารอบโลก แบ่งออกเป็น 60 โซน ๆ ละ 6 องศา

โซนที่ 1 อยู่ระหว่าง 180 องศา กับ 174 องศาตะวันตก และมีลองติจูด 177 องศาตะวันตกเป็นเมริเดียนย่านกลาง (Central Meridian) มีเลขกำกับแต่ละโซนจาก 1 ถึง 60 โดยนับจากซ้ายไปทางขวาระหว่างละติจูด 84 องศาเหนือ 80 องศาใต้ แบ่งออกเป็น 2 ช่อง ช่องละ 8 องศา ยกเว้นช่องสุดท้ายเป็น 12 องศา โดยเริ่มนับตั้งแต่ละติจูด 80 องศาใต้ขึ้นไปทางเหนือ ให้ช่องแรกเป็นอักษร C และช่องสุดท้ายเป็นอักษร X (ยกเว้น I และ O) จากการแบ่งตามที่กล่าวแล้วจะเห็นพื้นที่ในเขตลองติจูด 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 ลิปดาเหนือ ถึง 20 องศา 30 ลิปดาเหนือ และลองติจูดประมาณ 97 องศา 30 ลิปดาตะวันออก ถึง 105 องศา 30 ลิปดาตะวันออก ดังนั้นประเทศไทยจึงตกอยู่ใน GZD 47N 47P 47Q 48N 48P และ 48Q [2]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 การแบ่งกริดโซนระบบพิกัดกริด UTM

2.2 เทคนิคการหาเส้นทางที่สั้นที่สุด (Shortest-Route Problem)

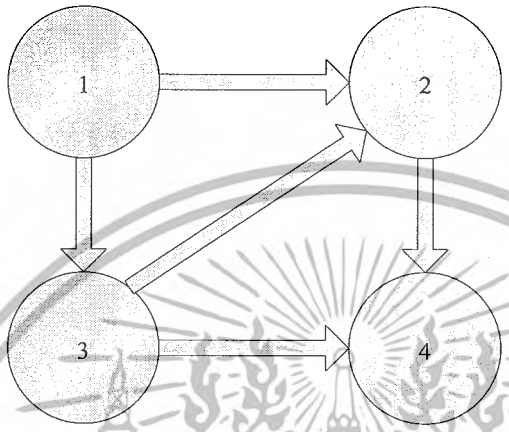
กราฟแบบมีทิศทาง G ใด ๆ จะประกอบไปด้วยเซตของเวอร์ทีซ (vertices) V และเซตของความสัมพันธ์ระหว่างเวอร์ทีซหรือที่เรียกว่าอาร์ค (arcs) E เวอร์ทีซอาจเรียกได้อีกอย่างว่าโหนด (node) และอาร์คเรียกได้อีกอย่างว่า directed lines อาร์คจะเป็นคู่ลำดับของเวอร์ทีซ (v,w) โดย v เรียกว่า Tail และ w เรียกว่า Head ของอาร์ค (v,w) มักแทนอยู่ในรูปของ $v \rightarrow w$ และเขียนได้ดังนี้

รูปที่ 2.2 ความสัมพันธ์ของอาร์ค

สังเกตว่า หัวลูกศรจะอยู่ที่เวอร์ทิซีซที่เรียกว่า Head และหางลูกศรจะอยู่ที่เวอร์ทีซที่เรียกว่า Tail เราจะกล่าวว่าอาร์ค $v \rightarrow w$ เป็นความสัมพันธ์จาก v ไปยัง w เป็นการประชิด (adjacent) ไปยัง v เวอร์ทีซของกราฟแบบมีทิศทางนี้สามารถใช้แทนวัตถุใด ๆ และอาร์คใช้แทนความสัมพันธ์ระหว่างวัตถุ เช่น เวอร์ทีซอาจแทนชื่อเมืองต่าง ๆ ในขณะที่อาร์คอาจแทนเส้นทางการบินระหว่างเมืองหนึ่งสู่อีกเมืองหนึ่ง

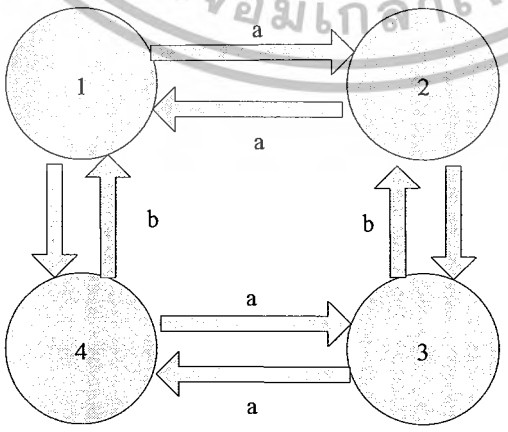
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เส้นทางของกราฟแบบมีทิศทาง หมายถึง ลำดับของเวอร์ทีซ v_1 ,v_2 ,v_3 ,..... ,v_n เช่น $v_1 \rightarrow v_2$, $v_2 \rightarrow v_3$, ..., $v_{n-1} \rightarrow v_n$ โดยที่เส้นทางนี้เริ่มจากวเอร์ทีซ v_1 ไปยังเวอร์ทีซ v_n ผ่านเวอร์ทีซ v_2 , v_3 , ... , v_{n-1} และสิ้นสุดที่ v_n Length หรือความยาวของเส้นทาง คือ จำนวนของอาร์คบนเส้นทางในกรณีนี้คือ n-1 ดังตัวอย่างเช่น



รูปที่ 2.3 กราฟแบบมีทิศทางที่มี 4 เวอร์ทีซ

จากรูป ลำดับของเวอร์ทีซ 1,2,4 คือเส้นทางที่มีความยาวเท่ากับ 2 จากเวอร์ทีซ 1 ไปยังเวอร์ทีซ 4 และเส้นทาง 3,2,4,3 จะเป็นวนรอบที่มีความยาวเท่ากับ 3 ในการประยุกต์ใช้งานหลายๆอย่าง เรามักจะกำหนดเครื่องหมายไว้บนเวอร์ทีซและอาร์คโดยที่ เครื่องหมายนี้อาจเป็นชื่อราคาหรือมูลค่าของข้อมูลต่าง ๆ ที่กำหนด ดังเช่นตัวอย่างต่อไปนี้ซึ่งแสดงให้เห็นกราฟที่กำหนดเครื่องหมายไว้ในแต่ละอาร์ค ซึ่งกำหนดโดยตัวอักษรที่แสดงถึงการเปลี่ยนแปลงจากเวอร์-ทีซหนึ่งไปยังอีกเวอร์ทีซหนึ่ง



รูปที่ 2.4 กราฟที่กำหนดความสัมพันธ์ของอาร์ค

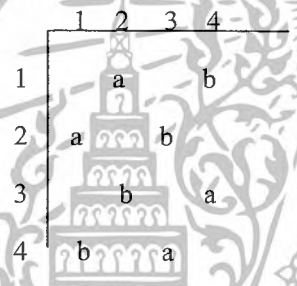
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 การแทนรูปแบโครงสร้างกราฟแบบมีทิศทาง

เราสามารถใช้โครงสร้างข้อมูลหลายชนิดมาใช้แทนกราฟแบบมีทิศทางนี้แต่โครงสร้างข้อมูลที่เหมาะสมนี้จะขึ้นอยู่กับการกระทำที่จะนำมาประยุกต์ใช้กับเวอร์ทีซและอาร์คเหล่านั้นวิธีการที่นิยมสำหรับกราฟแบบมีทิศทาง G=(V,E) วิธีหนึ่งก็คือ ใช้เมตริกซ์การ ประชิด (adjacency matrix)

สมมุติว่า v={1,2,...,n} เมตริกซ์การประชิดสำหรับ G คือ nxn เมตริกซ์ A ของบูลีน (Booleans) โดยที่ A[i,j] เป็นจริงถ้าหากมีอาร์คจากเวอร์ทีซ i ไปยัง j โดยทั่วไปเรามักจะแทนเมตริกซ์การประชิดด้วย 1 สำหรับกรณีที่เป็นจริง และ 0 ในกรณีที่เป็นเท็จ

จากตัวอย่างแสดงให้เห็นเมตริกซ์การประชิดของกราฟแบบมีทิศทางในที่นี้ เครื่องหมายกำกับจะเป็นตัวอักษร และช่องว่างจะหมายถึงการที่ไม่มีอาร์คอยู่



รูปที่ 2.5 เมตริกซ์ประชิดของรูปที่ 5

2.2.2 ปัญหาทางเดินที่สั้นที่สุดแบบทางเดียว

ปัญหาการหาเส้นทางเดินในกราฟแบบมีทิศทาง สมมติว่ามีกราฟแบบมีทิศทาง G=(V,E) อันหนึ่ง ซึ่งแต่ละอาร์คไม่มีค่าลบ ในเวอร์ทีซใดเวอร์ทีซหนึ่ง จะถูกกำหนดให้เป็นจุดเริ่มต้น (source) ปัญหาก็คือการพิจารณาค่าคอส (cost) ซึ่งอาจจะเป็น ค่าใช้ค่าย ระยะทาง หรือเวลา ฯลฯ ของระยะทางที่น้อยที่สุด จากจุดเริ่มต้นไปยังเวอร์ทีซอื่น ๆ ทุก ๆ เวอร์ทีซใน v โดยที่ความยาวของเส้นทาง (length of path) ก็คือผลรวมของค่าคอส (cost) บนเส้นทางนั้น

G อาจจะแทนแผนที่การเดินรถ ซึ่งในเวอร์ทีซต่าง ๆ จะแทนสถานที่สำคัญต่าง ๆ บนสายการเดินรถและแต่ละอาร์ค $v \rightarrow w$ คือ เส้นทางการเดินรถระหว่างสถานที่ v ไปยังสถานที่ w เครื่องหมายบนอาร์ค $v \rightarrow w$ ก็คือ เวลาหรือระยะทางในการเดินทางจาก v ไปยัง w การแก้ปัญหาในเรื่องนี้ ก็คือการหาเวลาหรือระยะทางในการเดินทางที่น้อยที่สุดจากสถานที่ที่กำหนดไปยังทุกสถานที่ในแผนที่

ในการแก้ปัญหานี้นำมาประยุกต์ใช้กับอัลกอริทึมที่เรียกว่า "Dijkstra's algorithm" อัลกอริทึมนี้ทำงานโดยการพยายามรักษาเซต S ของเวอร์ทีซต่าง ๆ ซึ่งเราทราบระยะทางสั้น

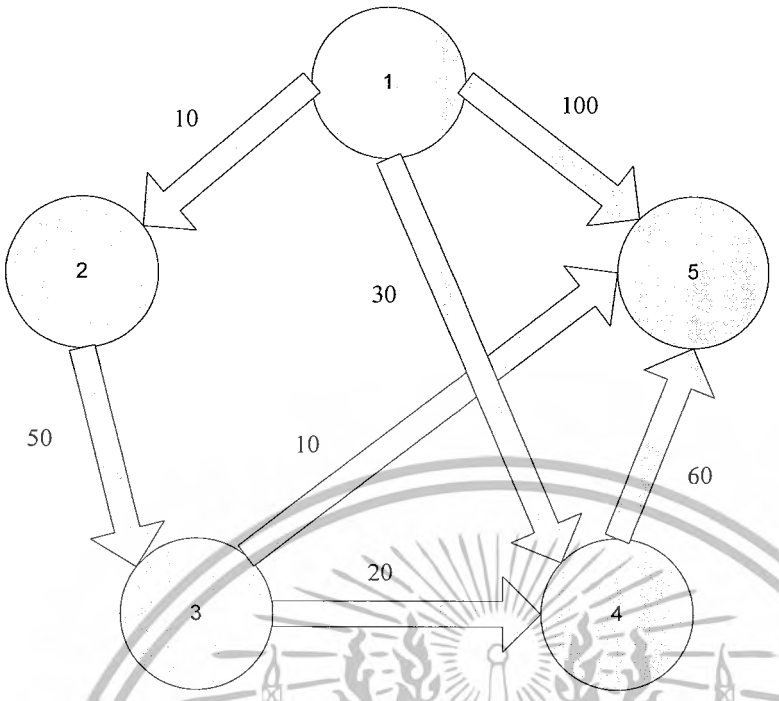
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่สุดจากจุดเริ่มต้นไปยังเวอร์ทีซเหล่านั้น สมมติในตอนเริ่มต้นเซต S มีเพียงเวอร์ทีซของจุดเริ่มต้นเท่านั้น ต่อมาในแต่ละขั้นตอน เราก็จะบวกเวอร์ทีซ v ซึ่งมีระยะห่างจากจุดเริ่มต้นน้อยที่สุดเท่าที่จะน้อยได้ลงไป และในแต่ละขั้นของอัลกอริทึม เราจะใช้อาร์เรย์ D ในการเก็บค่าระยะทางที่สั้นที่สุดที่หาได้

```
S :={1};
        For I := 2 to n do
                D[i] :=C[l,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 สมมติว่ากราฟ G=(V,E) ซึ่ง V = {1,2,...,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 ∞ และในทุก ๆ ขั้นตอน D[i] จะเก็บความยาวเส้นทางที่น้อยที่สุดจาก จุดเริ่มต้นไปยังเวอร์ทีซ i ใด ๆ แคสงตัวอย่างดังรูปต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 กราฟที่มีทิศทางที่มีค่าบนอาร์ค

สมมติว่าใช้ Dijkstra's algorithm กับรูปตัวอย่าง ในตอนเริ่มต้น S = {1}, D[2] = 10 ,D[3] = ∞ , D[4] = 30 และ D[5] = 100 ในรอบแรกของ for-loop ในบรรทัดที่ (4) ถึง (8) จะได้ว่า w = 2 คือ เวอร์ทีซที่ถูกเลือกซึ่งมีค่า D ต่ำที่สุด ดังนั้นเราจะเซตให้ D[3] = min(∞ ,10+50) = 60 , D[4] และ D[5] ไม่มีการเปลี่ยนแปลง เนื่องจากการเข้าหาเวอร์ทีซ 4 และ 5 โดยตรงจากเวอร์ทีซ 1 จะสั้นกว่าการเข้าหาผ่านเวอร์ทีซ 2 ลำดับของค่า D หลังจากการวนรอบของ for-loop จะได้ดังตารางต่อไปนี้

ตารางที่ 2.1 แสดงข้อมูลของตาราง Dijkstra's algorithm

Loop	S	W	D[2]	D[3]	D[4]	D[5]
0	{1}	-	**10{1->2}**	∞ {-}	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}, {1->4->3}	3	10{1->2}	50{1->4->3}	30{1->4}	**60{1->4->3->5}**
4	{1},{1->2},{1->4}, {1->4->3},{1->4->3->5}	5	10{1->2}	50{1->4->3}	30{1->4}	60{1->4->3->5}

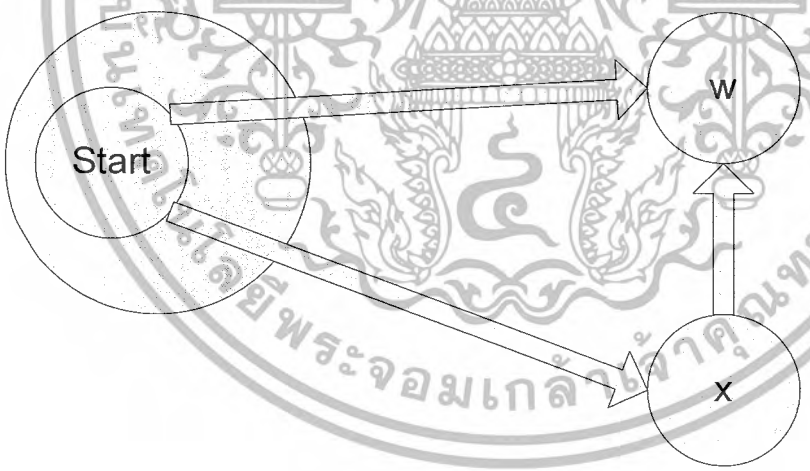
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

ต่อมา จะเริ่มผนวกเอาเวอร์ทีซที่สั้นที่สุดจากเวอร์ทีซเริ่มค้นเข้ามาอีก ทำให้ S = {1,2,4} อันเนื่องมาจาก เวอร์ทีซ 4 เป็นเวอร์ทีซที่สั้นที่สุดจากลุ่มเวอร์ทีซ {1,2} เพราะว่าจากเวอร์ทีซ 2 ไป 3 มีค่าเท่ากับ 50 ส่วนจากเวอร์ทีซ 1 ไป 5 มีค่าเท่ากับ 100 ดังนั้นเวอร์ทีซที่สั้นที่สุดจากลุ่ม S = {1,2} ก็คือ 4

จากการรวนซ้ำครั้งที่ 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 จะเป็นเช่นนี้เรื่อย ๆ จนกว่าจะหมดเวอร์ทีซที่จะนำมารวมในเซต S

Dijkstra's algorithm ทำงานโดยหลักการที่ว่า การกระทำสิ่งไหนที่ดูเหมือนว่าเป็นสิ่งที่ดีที่สุดในส่วนย่อย ก็จะกลายเป็นการกระทำสิ่งที่ดีที่สุดในส่วนทั้งหมดในขั้นปลาย ในกรณีนี้สิ่งที่ดีที่สุดที่จะทำคือการหาระยะทางสั้นที่สุดไปยังเวอร์ทีซ w ที่อยู่ภายนอกเซต S



รูปที่ 2.7 ข้อสมมติฐานของการหาระยะทางที่สั้นที่สุดไปยัง w

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

เพื่อเป็นการพิสูจน์ว่าอัลกอริทึมทำงานได้จริง สมมติว่า D[v] เป็นระยะทางสั้นที่สุดไปยังเวอร์ทีซ v ที่ขณะเวลาใด ๆ ปัญหาของเรื่องนี้ก็คือ เมื่อเราเพิ่มเวอร์ทีซ w เข้าไปยังเซต S ที่บรรทัด (6) , (7) , (8) จะทำการปรับค่า D เพื่อจดบันทึกความเป็นไปได้ที่ว่า 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] นี้จะถูกลดลงถ้าค่าใหม่นี้น้อยกว่าเดิม เส้นทางที่สั้นที่สุดนี้อาจจะเป็นไปได้อีกทางหนึ่งได้แสดงดังรูป

รูปที่ 2.8 การเดินทางที่สั้นที่สุดที่เป็นไปได้

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

แต่จริง ๆ แล้วเส้นทางนี้จะไม่สามารถใช้ได้ เนื่องจาก x ถูกนำเข้าสู่เซต S ก่อน w และเส้นทางที่สั้นที่สุดจากจุดเริ่มต้นมายัง x จะผ่านแต่ old S เท่านั้น ดังนั้นเส้นทางไปยัง x ผ่านทาง w ตามที่แสดงในรูป จึงไม่อาจสั้นกว่าเส้นทางจาก x ผ่านในเซต S โดยตรง ผลก็คือความยาวเส้นทางในรูปจากจุดเริ่มต้นไปยัง w, x และ v ไม่อาจสั้นกว่าค่าเดิมของ D[v] เนื่องจาก D[v] ไม่อาจมีค่ามากกว่าความยาวของทางเดินที่สั้นที่สุด ที่เดินทางไปยัง x ผ่านเซต S และตรงไปยัง v ดังนั้น D[v] ไม่อาจถูกลดลงได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 อัลกอริทึมแบบ Dijkstra's algorithms

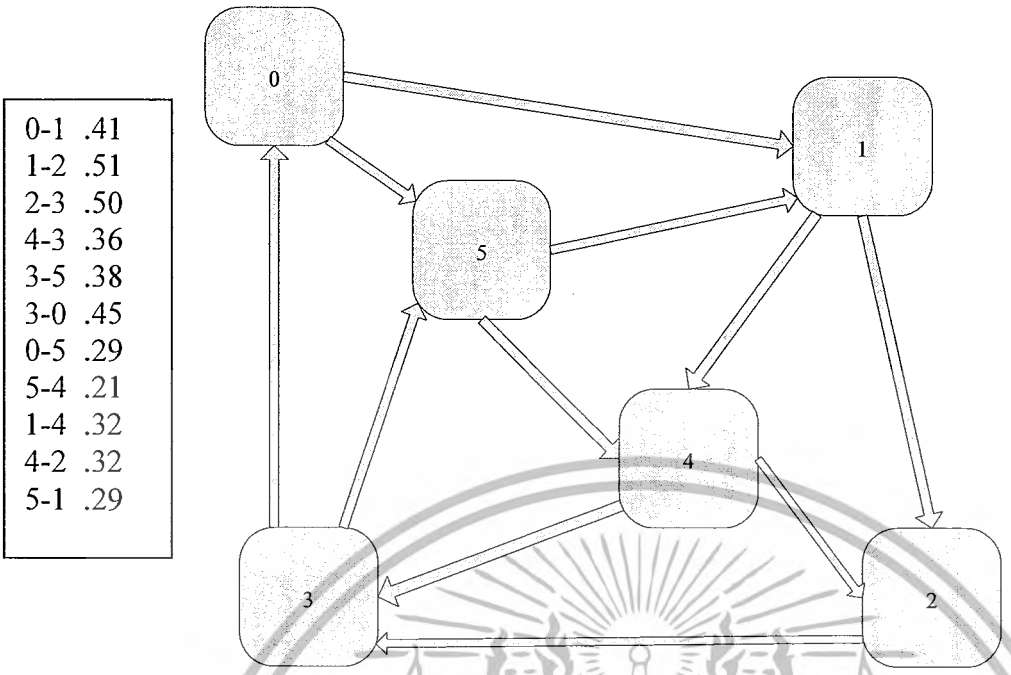
โครงสร้างในการทำงานของระบบแนะนำเส้นทางการจราจรนี้ได้มีส่วนที่ช่วยเหลือผู้ใช้-งานในการหาเส้นทางในการเดินทางโดยใช้รถยนต์ส่วนตัวที่สั้นที่สุดที่ใช้ในการเดินทาง โดยใช้อัลกอริทึม Dijkstra ซึ่งเป็นอัลกอริทึม ที่ใช้ในการค้นหาเส้นทางที่สั้นที่สุด โดยนำค่าของระยะทางของแต่ละช่วงถนนสายหลักมาคิด ในการทำงานของส่วนระบบการค้นหาเส้นทางนี้ได้นำข้อมูลบางส่วนบางคอลัมน์จากฐานข้อมูลแต่ละช่วงถนนสายหลัก โดยข้อมูลส่วนคอลัมน์ของช่วงถนนสายหลักที่นำมาใช้งานจะประกอบไปด้วย

- หมายเลขของถนนสายหลัก
- ชื่อของถนนสายหลักแต่ละเส้น
- โหนดเริ่มต้นและโหนดสุดท้ายของถนนสายหลักที่เป็นจุดเชื่อมต่อของถนนแต่ละเส้น
- ละติจูด ลองติจูด ของจุดเริ่มต้นและจุดสุดท้ายของถนนสายหลักแต่ละเส้น
- ระยะทางของถนนสายหลักในแต่ละช่วง

ข้อมูลที่สำคัญอีกส่วนหนึ่งในการจะนำมาใช้วาดเส้นบนแผนที่ คือ ข้อมูลที่เก็บในรูปแบบของบิตข้อมูลที่ได้รับมาจากเครื่องจีพีเอส (GPS) ซึ่งส่วนนี้จะนำมาใช้เพียงในการวาดรูปเท่านั้น ไม่ได้ใช้ในการคำนวณเส้นทางที่สั้นที่สุด

เมื่อเรานำข้อมูลของแต่ละถนนสายหลักมาแล้ว จะนำข้อมูลต่าง ๆ มาทำการสร้างเป็นกราฟ โดยเราจะทำการดึงข้อมูลของถนนแต่ละเส้นมาเช็คว่า มีถนนสายไหนบ้างที่เชื่อมต่อกับถนนที่เรากำลังสนใจอยู่ โดยการตรวจสอบจุดโหนดที่มีการเชื่อมต่อที่ได้อธิบายไว้ขั้นต้น ถ้าพบว่ามีถนนที่มีการเชื่อมต่อเกิดขึ้น ตัวโปรแกรมจะทำการสร้างโหนดขึ้นมา โดยโหนดที่เราพิจารณาอยู่นั้นจะกำหนดให้เป็นโหนดแม่หรือโหนดเริ่มต้น และถนนที่มีการเชื่อมต่อกับโหนดแม่จะให้เป็นโหนดลูกหรือโหนดปลายทาง และทำการสร้างความสัมพันธ์ของทั้งสองโหนดที่เรียกว่าด้านหรืออาร์ค ซึ่งเป็นค่าของระยะทางโดยค่านี้จะเป็นค่าน้ำหนักที่เราจะใช้ในการตัดสินใจเลือกเส้นทางที่สั้นที่สุด โดยเราจะทำเช่นนี้ไปเรื่อย ๆ จนครบทุกถนน ซึ่งแสดงดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 การสร้างกราฟจากโหนด

จากรูปจะเห็นได้ว่า ถนนหมายเลขที่ 1 เชื่อมต่อกับหมายเลข 2 โดยมีระยะทางจากถนนหมายเลข 1 ไปยังถนนหมายเลข 2 มีระยะทาง 510 เมตร ถนนหมายเลข 2 เชื่อมต่อกับถนนหมายเลข 3 โดยระยะทางเชื่อมต่อกันเป็นระยะทาง 500 เมตร

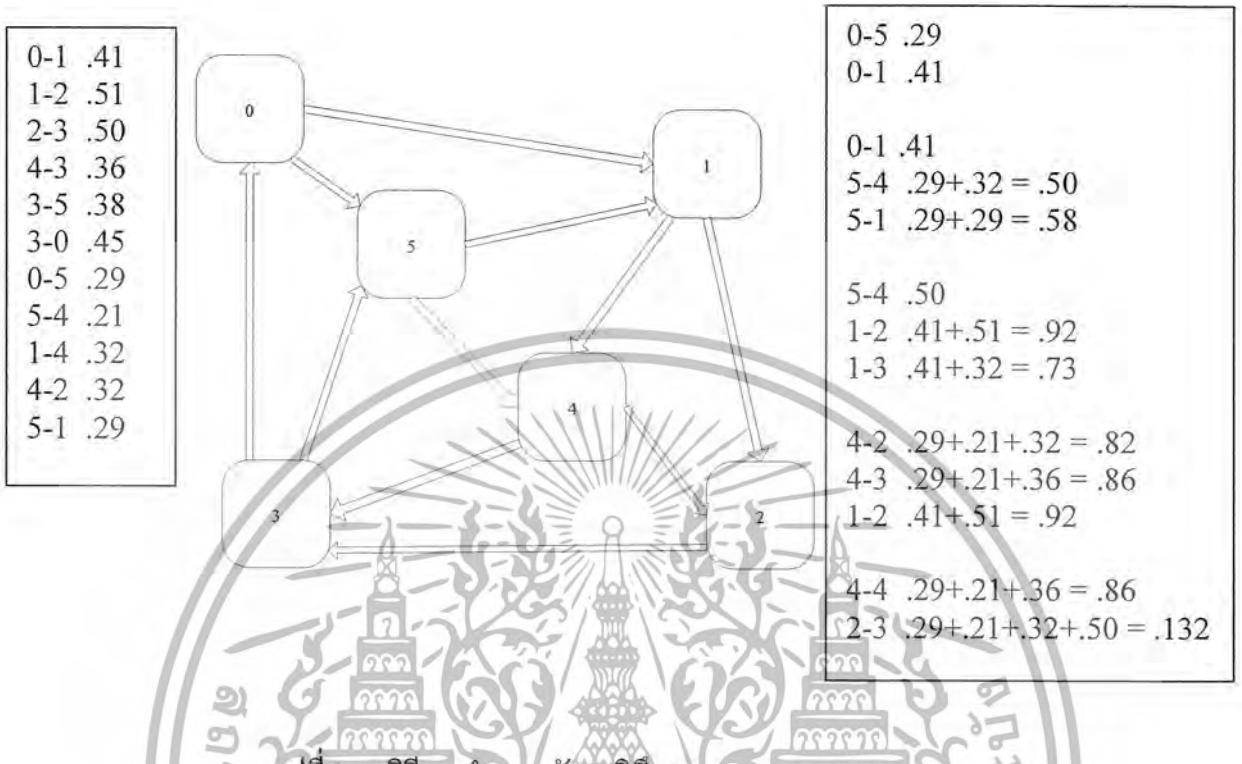
ในการสร้างกราฟขึ้นมานี้ เราได้นำเทคนิคการใช้อาร์เรย์ลิงค์ลิสต์เข้ามาช่วย โดยในที่นี้เราจะนำเอาลิงค์ลิสต์มาประยุกต์การใช้งานกับอาร์เรย์ เพราะลิงค์ลิสต์มีคุณสมบัติในการจัด เก็บชุดข้อมูลเชื่อมโยงต่อเนื่องกันไปตามลำดับ ซึ่งอาจอยู่ในลักษณะแบบเชิงเส้นตรงหรือไม่เป็นเส้นตรงก็ได้ ซึ่งในลิสต์จะประกอบไปด้วยข้อมูลที่เรียกว่าโหนด ในหนึ่งโหนดจะประกอบด้วยส่วนของข้อมูลที่ต้องการจัดเก็บเรียกว่าส่วนข้อมูล และส่วนที่เป็นพอยเตอร์ที่ชี้ไปยังโหนดถัดไปหรือชี้ไปยังโหนดอื่น ๆ ที่อยู่ในลิสต์ ซึ่งเราได้ใช้ลิงค์ลิสต์หลาย ๆ ตัวนำมาจัดเป็นอาร์เรย์ โดยหนึ่งลิงค์ลิสต์จะประกอบด้วยโหนดแม่ ซึ่งโหนดแม่ก็คือถนนเริ่มต้น โหนดลูกก็คือโหนด ที่เป็นถนนที่มีการเชื่อมต่อกับถนนที่เราพิจารณา เหตุผลที่นำอาร์เรย์ลิงค์ลิสต์มาใช้งานอันเนื่องมาจากต้องการเพิ่มประสิทธิภาพในการทำงานของโปรแกรม ซึ่งถ้าใช้อาร์เรย์จำเป็นต้องมีการจอง หน่วยความจำไว้ล่วงหน้า ซึ่งถ้ามากเกินไปก็อาจก่อให้เกิดการล่าช้าต่อการค้นหา ข้อมูลมาใช้งาน

เมื่อเราทำการสร้างอาร์เรย์ลิงค์ลิสต์ของทุกถนนเสร็จแล้ว ก็จะทำการให้ผู้ใช้กรอกข้อมูลสถานที่ที่ต้องการ ซึ่งเมื่อกรอกข้อมูลเสร็จเรียบร้อยแล้วโปรแกรมจะทำการตรวจสอบว่ามีชื่อสถานที่ที่ว่ามีอยู่ในฐานข้อมูลหรือไม่ ถ้ามีจะทำการค้นหาถนนสายหลักที่ใกล้สถานที่นั้นมากที่สุด เป็นการเลื่อนจุดลงถนนที่ใกล้ที่สุดเพื่อทำการค้นหาเส้นทางโดยใช้อัลกอริทึมแบบ Dijkstra ซึ่งการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ทำงานจะมีลักษณะดังรูป

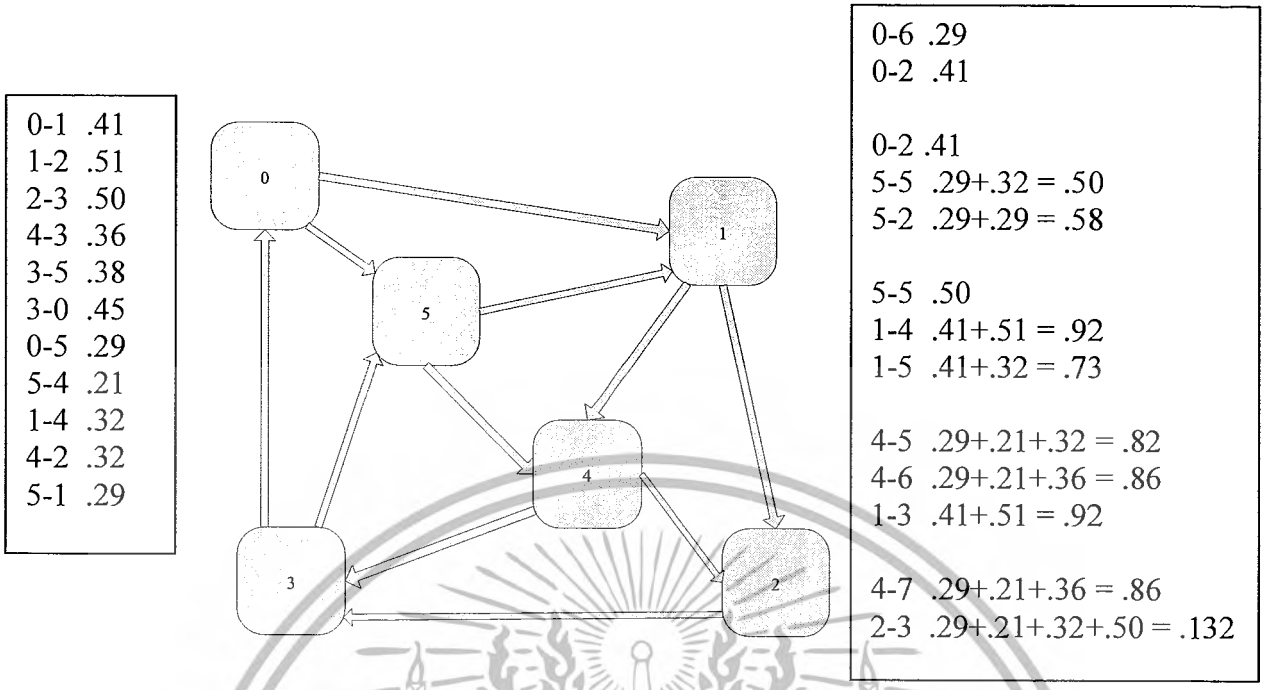


รูปที่ 2.10 วิธีการคำนวณอัลกอริทึม Dijkstra

สมมุติเราให้โหนดที่เป็นถนนสายหลักคือถนนหมายเลข 0 ได้มาจากการเลื่อนจุดจากสถานที่ลงถนนแล้ว จากนั้นโปรแกรมจะทำการคำนวณหาว่า เส้นทางจากถนนหมายเลข 0 ไปยังทุก ๆ ถนนที่ใช้ค่าใช้จ่ายหรือระยะทางสั้นที่สุด คือถนนสายไหนดังรูปข้างต้น จะแสดงได้ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

104283



รูป 2.11 เส้นทางจากถนนหมายเลข 0 ไปยังทุก ๆ ถนน

จะเห็นได้ว่าคล้ายการสร้างต้นไม้ย่อยลงไปเรื่อย ๆ โดยโหนดแม่ของมันคือโหนด 0 level ถัดมาก็คือโหนดของถนนหมายเลข 5 และหมายเลข 1 และลงมาเรื่อย ๆ ตามรูป และต่อมาเราจะทำการป้อนถนนปลายทางที่ได้จากการแปลงถนนสายหลัก สมมุติว่าได้ถนนหมายเลข 3 ซึ่งถนนหมายเลข 3 มีถนนหมายเลข 4 เป็นโหนดแม่ซึ่งถนนหมายเลข 4 ก็มีถนนหมายเลข 5 เป็นโหนดแม่และถนนหมายเลข 5 ก็มีถนนหมายเลข 0 ซึ่งเป็นถนนในระดับสูงสุดซึ่งก็คือถนนต้นทางเช่นกัน ฉะนั้นเราจะได้ว่า เราต้องเดินทางจากถนนหมายเลข 0->5->4->3 ซึ่งข้อมูลของชื่อระยะทาง หรือข้อมูลบางอย่างที่เราต้องการแสดงผลในฐานข้อมูล ก็สามารถเรียกโดยการค้นหาข้อมูลจากหมายเลขถนนมาได้ไม่ยากนัก [1]

2.4 ความรู้ทั่วไปเกี่ยวกับฐานข้อมูลใน PostgreSQL

2.4.1 เกี่ยวกับ PostgreSQL

เป็นระบบจัดการฐานข้อมูลแบบ object-Relational โดยสามารถใช้รูปแบบของภาษา SQL ได้เกือบทั้งหมด และสามารถใช้ subselects, transactions, user-defined types และ functions ได้ อีกทั้งเป็นระบบจัดการฐานข้อมูลเปิดเผยซอร์สโค้ด (source code)

2.4.2 ลักษณะโครงสร้าง

ระบบที่ใช้ PostgreSQL จะติดตั้ง PostgreSQL ไว้ที่เครื่องเซิร์ฟเวอร์ ซึ่งเป็นที่เก็บฐานข้อ - มูลด้วยและยังสามารถติดตั้ง PostgreSQL ได้มากกว่า 1 ชุดในเซิร์ฟเวอร์เครื่องเดียว

เอกสารนี้มีลิขสิทธิ์ ห้ามนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผู้ดูแลระบบ PostgreSQL จะใช้ชื่อว่า postgres ซึ่งเป็นผู้ดูแลทั้งตัวโปรแกรมและ database ซึ่งสามารถทำงานกับบางคำสั่งเฉพาะ เพื่อจัดการฐานข้อมูลและผู้ใช้บริการ (user) ซึ่งผู้ดูแลระบบ database (postgres) จะคล้ายการทำงานของ superuser ในระบบยูนิกซ์ (UNIX) หน้าที่ของ postgres สามารถสร้างบัญชีผู้ใช้ และกำหนดสิทธิและระดับการใช้งานต่าง ๆ ได้ [3]

2.5 ความรู้ทั่วไปกี่ยวกับ PostGIS

2.5.1 เกี่ยวกับ PostGIS

PostGIS เป็นสิ่งที่ถูกพัฒนาขึ้นโดยบริษัท Refractions Research ซึ่งเป็นโครงการวิจัยทางเทคโนโลยีด้านฐานข้อมูลเชิงพื้นที่ Refractions คือ ระบบสารสนเทศภูมิศาสตร์ (GIS) และการให้คำปรึกษาฐานข้อมูลของบริษัทใน Victoria, British Columbia และ Canada ซึ่งฐานข้อมูลนี้จะมีความพิเศษในการรวบรวมข้อมูล และพัฒนาซอฟท์แวร์ของลูกค้าโดยจะมีแผนการสนับสนุน และพัฒนา PostGIS เพื่อให้รองรับลำดับของความสำคัญในบทบาทหน้าที่ของระบบสารสนเทศภูมิ - ศาสตร์ซึ่งจะประกอบไปด้วย OpenGIS เต็มรูปแบบ ซึ่งสิ่งที่สร้างขึ้นนี้จะเป็นลักษณะภูมิประเทศใน ชั้นสูง (บริเวณที่ปกคลุม, พื้นผิว , เครือข่าย) ที่หน้าจอของผู้ใช้จะมีเครื่องมือที่ใช้ติดต่อสื่อสารสำหรับแสดงผลและแก้ไขข้อมูลสารสนเทศภูมิศาสตร์ และเครื่องมือในการเข้าถึงเว็บขั้นพื้นฐาน

PostGIS เป็นการเพิ่มเติมในส่วนของฐานข้อมูลเชิงวัตถุสัมพันธ์ (object-relational database system) ของ PostgreSQL ให้มีการรองรับวัตถุทางระบบสารสนเทศภูมิศาสตร์ (GIS object) เข้ามาเก็บไว้ในฐานข้อมูล มีการสนับสนุน Gist index และ R-tree index ซึ่งเป็นวิธีการค้นข้อมูลแบบตัวชี้ (Indexing) ที่ใช้ในฐานข้อมูลเชิงพื้นที่สำหรับฐานข้อมูลที่มีขนาดใหญ่ ซึ่ง PostGIS เองมีการกำหนดการใช้งานโดย OpenGIS ที่เป็นลักษณะพื้นฐานของ SQL(SFSQL)

ดังที่กล่าวมาแล้วว่า PostGIS มีการกำหนดการใช้งานที่เป็นลักษณะพื้นฐานโดย OpenGIS Consortium (OGC) ซึ่งเป็นสถาบันที่ศึกษาเพื่อการสร้างอินเตอร์เฟสที่ทำโปรแกรมประยุกต์ซอฟแวร์ ให้มีการใช้งานได้กับระบบสารสนเทศภูมิศาสตร์ (GIS)

OpenGIS เป็นสิ่งกำหนดความชัดเจนในการเข้าถึงข้อมูลทางด้านธรณีที่แตกต่างกัน และประมวลผลทางธรณีของแหล่งทรัพยากรในสภาพแวดล้อมที่เป็นเครือข่าย (Network) [4]

2.6 ความรู้ทั่วไปเกี่ยวกับ GIS

ระบบสารสนเทศภูมิศาสตร์ หรือ Geographic Information System : GIS คือกระบวนการทำงานเกี่ยวกับข้อมูลในเชิงพื้นที่ด้วยระบบคอมพิวเตอร์ ที่ใช้กำหนดข้อมูลและสารสนเทศ ที่มีความสัมพันธ์กับตำแหน่งในเชิงพื้นที่ เช่น ที่อยู่ บ้านเลขที่ สัมพันธ์กับตำแหน่งในแผนที่ ตำแหน่งเส้นรุ้ง เส้นแวง ข้อมูลและแผนที่ใน GIS เป็นระบบข้อมูลสารสนเทศที่อยู่ในรูปของตารางข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และฐานข้อมูลที่มีส่วนสัมพันธ์กับข้อมูลเชิงพื้นที่ (Spatial Data) ซึ่งรูปแบบและความสัมพันธ์ของข้อมูลเชิงพื้นที่ทั้งหลาย จะสามารถนำมาวิเคราะห์ด้วย GIS และทำให้สื่อความหมายในเรื่องการเปลี่ยนแปลงที่สัมพันธ์กับเวลาได้ เช่น การแพร่ขยายของโรคระบาด การเคลื่อนย้าย ถิ่นฐาน การบุกรุกทำลาย การเปลี่ยนแปลงของการใช้พื้นที่ ฯลฯ ข้อมูลเหล่านี้ เมื่อปรากฏบนแผนที่ทำให้สามารถแปลและสื่อความหมายใช้งานได้ง่าย

GIS เป็นระบบข้อมูลข่าวสารที่เก็บไว้ในคอมพิวเตอร์ แต่สามารถแปลความหมายเชื่อมโยงกับสภาพภูมิศาสตร์อื่นๆ สภาพท้องที่ สภาพการทำงานของระบบสัมพันธ์กับสัดส่วนระยะทางและพื้นที่จริงบนแผนที่ ข้อแตกต่างระหว่าง GIS กับ MIS นั้นสามารถพิจารณาได้จากลักษณะของข้อมูล คือ ข้อมูลที่จัดเก็บใน GIS มีลักษณะเป็นข้อมูลเชิงพื้นที่ (Spatial Data) ที่แสดงในรูปของภาพ (graphic) แผนที่ (map) ที่เชื่อมโยงกับข้อมูลเชิงบรรยาย (Attribute Data) หรือฐานข้อมูล การเชื่อมโยงข้อมูลทั้งสองประเภทเข้าด้วยกัน จะทำให้ผู้ใช้สามารถที่จะแสดงข้อมูลทั้งสองประเภทได้พร้อมๆ กัน เช่น สามารถจะค้นหาตำแหน่งของจุดตรวจวัดควันดำ - ควันขาวได้โดยการระบุชื่อจุดตรวจ หรือในทางตรงกันข้ามสามารถที่จะสอบถามรายละเอียดของ จุดตรวจจากตำแหน่งที่เลือกขึ้นมา ซึ่งจะต่างจาก MIS ที่แสดงภาพเพียงอย่างเดียว โดยจะขาดการเชื่อมโยงกับฐานข้อมูลที่เชื่อมโยงกับรูปภาพนั้น เช่นใน CAD (Computer Aid Design) จะแสดงเป็นภาพเพียงอย่างเดียว แต่แผนที่ใน GIS จะมีความสัมพันธ์กับตำแหน่งในเชิงพื้นที่ทางภูมิศาสตร์ คือเป็นค่าพิกัดที่แน่นอน ข้อมูลใน GIS ทั้งข้อมูลเชิงพื้นที่และข้อมูลเชิงบรรยายสามารถอ้างอิงถึงตำแหน่งที่มีอยู่จริงบนพื้นโลกได้โดยอาศัยระบบพิกัดทางภูมิศาสตร์ (Geocode) ซึ่งจะสามารถอ้างอิงได้ทั้งทางตรงและทางอ้อม ข้อมูลใน GIS ที่อ้างอิงกับพื้นผิวโลกโดยตรง หมายถึง ข้อมูลที่มีค่าพิกัดหรือมีตำแหน่งจริงบนพื้นโลกหรือในแผนที่ เช่น ตำแหน่งอาคาร ถนน ฯลฯ สำหรับข้อมูล GIS ที่จะอ้างอิงกับข้อมูลบนพื้นโลกได้โดยทางอ้อมได้แก่ ข้อมูลของบ้าน (รวมถึงบ้านเลขที่ ซอย เขต แขวง จังหวัด และรหัสไปรษณีย์) โดยจากข้อมูลที่อยู่ เราสามารถทราบได้ว่าบ้านหลังนี้มีตำแหน่งอยู่ ณ ที่ใดบนพื้นโลกเนื่องจากบ้านทุกหลังจะมีที่อยู่ไม่ซ้ำกัน [5]

2.7 เทคโนโลยี Google Maps API

Google Maps API ช่วยให้เราสามารถพัฒนาโปรแกรมเพื่อแทรก Google Maps API เข้าไปเป็นองค์ประกอบส่วนหนึ่งในเว็บเพ็จที่ต้องการได้โดยเขียนเป็นรหัส Html และ Javascript ในรูปแบบที่ไม่สลับซับซ้อนนักสำหรับงานแผนที่ง่าย ๆ Google Maps API มีขีดความสามารถกว้างขวางเน้นในด้านการนำเสนอข้อมูลแผนที่ในลักษณะหมุดปัก (Push pin / Place market) ซึ่งสามารถกำหนดให้แสดงข้อมูลประกอบแผนที่เมื่อผู้ใช้คลิกที่ตัว push pin / place market นั้น ๆ หรือองค์แผนที่แบบเส้น (Polyline) พื้นที่ (Polygon) และภาพ (Ground overlay) บริการด้านแผนที่ของ Google นี้เริ่มต้นตั้งแต่กลางปีคศ. 2005 เป็นบริการฟรี จัดให้แก่ผู้ใช้ทั่วโลก ใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การจัดบริการแผนที่นี้ส่วนประกอบพื้นฐานสำคัญที่ดึงดูดใจให้มีผู้ใช้งานแผนที่ของกูเกิ้ล (Google) เป็นอย่างมากคือแผนที่ และภาพถ่ายดาวเทียมคุณภาพดีซึ่งครอบคลุมทั่วพื้นผิวโลกในมาตราส่วนต่าง ๆ ตามความเหมาะสม ทำให้การพัฒนาต่อยอดจากสิ่งที่ Google จัดไว้ให้แล้วเป็นงานที่น่าสน - ใจ ไม่ต้องลงทุนจัดหาทรัพยากรที่หายากและราคาแพงเองมาใช้ในโครงการอย่างที่เคยเป็นในอดีต

เนื่องจากจัดทำ Google Maps API เป็นโปรแกรมเผยรหัสเปิด (Open source program) ในภาษาจาวาสคริปต์ จึงทำให้ผู้ที่ใช้ที่เป็นนักพัฒนาโปรแกรมสามารถเข้าไปดูรายละเอียดของรหัสโปรแกรมได้สะดวก รวมทั้งสามารถปรับเปลี่ยนแก้ไขโปรแกรมได้ ทำให้ Google Maps API มีผู้ใช้กันอย่างกว้างขวาง เหตุผลสำคัญอีก 2 อย่างที่ส่งเสริมให้มีผู้ใช้มากคือแผนที่และภาพถ่ายดาวเทียมคุณภาพดีที่ใช้สนับสนุนการทำแผนที่มีให้ครอบคลุมพื้นที่ต่าง ๆ อย่างกว้างขวาง และชื่อเสียงของโปรแกรม Google Earth เสริมด้วยบริการ Google Local ที่มีมาก่อน [6]

2.8 GLOBAL POSITIONING SYSTEM (GPS)

GPS (Global Positioning System) เป็นระบบเดียวในปัจจุบัน ที่สามารถ แสดงตำแหน่งที่อยู่ ที่แน่นอนว่าอยู่ ณ. ตำแหน่งใดบนพื้นโลกได้ทุกเวลาทุกสภาพอากาศ ระบบนี้มีดาวเทียม 24 ดวง หมุนอยู่รอบโลกอยู่สูงขึ้นไป 11,000 nautical miles หรือประมาณ 20,200 km. จากพื้นโลก ดาวเทียมหมุนรอบโลก แบ่งเป็น 6 ระนาบ ระนาบละ 4 ดวง โดยทำมุมเอียง 55 องศา ดาวเทียมทั้งหมดจะได้รับการควบคุมดูแลจากสถานีภาคพื้นดินทั่วโลกตลอดเวลา

GPS ELEMENTS เราสามารถแบ่งระบบ GPS ออกเป็น ส่วน ๆ ดังนี้

- SPACE SEGMENT
- USER SEGMENT
- CONTROL SEGMENT

SPACE SEGMENT ส่วนของอวกาศประกอบไปด้วยเครือข่ายของดาวเทียม ระบบ GPS ทั้งระบบประกอบด้วย ดาวเทียม 24 ดวง โคจรรอบโลกที่ระยะ 11,000 ไมล์อากาศจากพื้นโลกใช้เวลา 12 ช.ม ในการโคจรรอบโลกหนึ่งรอบ ดาวเทียมโคจรรอบโลก แบ่งเป็น 6 ระนาบและทำมุมเอียง 55 องศา การวางวงโคจรเช่นนี้ทำให้เราสามารถรับสัญญาณจากดาวเทียมได้คราวละถึง 6 ดวง ดาวเทียมติดตั้งนาฬิกาที่เที่ยงตรงมากๆ ถึง 3 nanoseconds (ความเที่ยงตรง 0.000000003 ของวินาที หรือ 3e-9) ความเที่ยงตรงมีความสำคัญมาก สำหรับเครื่องรับ เพราะเครื่องรับจำเป็นต้องทราบเวลาที่เที่ยงตรงแน่นอนว่าระยะเวลาเท่าไรที่สัญญาณคลื่นจากดาวเทียมเดินทางถึงเครื่องรับดาวเทียมแต่ละดวงมีเชื้อเพลิง และเครื่องยนต์ขนาดเล็กซึ่งสามารถที่จะปรับแต่งดาวเทียมให้อยู่ในตำแหน่งที่ถูกต้อง ในวงโคจร ถ้าดาวเทียมเกิดเคลื่อนออกจากตำแหน่งที่กำหนดดาวเทียม แต่ละดวงมีนาฬิกา atomic clocks 4 อัน นาฬิกานี้มีความเที่ยงตรงถึงหนึ่งในหนึ่งพันล้านของวินาที หรือ nanosecond .

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดาวเทียมแต่ละดวงจะส่งคลื่นสัญญาณออกมาสองคลื่นสัญญาณหนึ่งคลื่นสำหรับการทหาร และอีกคลื่นหนึ่งสำหรับพลเรือน

USER SEGMENT เมื่อนักบินทำการบินเครื่องรับ GPS จะคำนวนตำแหน่งปัจจุบันอยู่ตลอดเวลาและแสดงตำแหน่งและทิศทางที่ถูกต้อง ระบบ GPS จะฟังสัญญาณจากดาวเทียมและวัดระยะเวลาจากเครื่องส่งสัญญาณจากดาวเทียมกับเครื่องรับสัญญาณ (รู้เวลาก็สามารถรู้ระยะทาง) และ โดยวิธีการของสามเหลี่ยม หรือ ตรีโกณระหว่างดาวเทียมหลายดวงที่ได้รับระบบเครื่องรับของดาวเทียมจะคำนวณตำแหน่งของเครื่องรับ เครื่องรับเองก็ต้องได้รับสัญญาณจากดาวเทียมอย่างน้อยสี่ดวง (ก็คือ รู้ระยะทางจากเครื่องรับถึงดาวเทียมสี่ดวง) ถึงจะคำนวนตำแหน่งลักษณะของ 3 มิติได้ (เครื่องรับสามารถคำนวนได้ถึงแม้จะได้รับสัญญาณจากดาวเทียมเพียงสามดวงแต่คำนวนได้เพียงสองมิตินอกจากจะรู้ความสูง) ไม่เพียงแต่รู้ตำแหน่งของเส้นรุ้งและเส้นแวงเท่านั้น ยังรู้ระยะความสูงด้วยมันมีหลายรูปแบบที่แสดงบนหน้าจอ ซึ่งแล้วแต่บริษัทผู้ผลิตไม่ต้องปรับหรือขูนหาคลื่นเพราะว่าความถี่ของดาวเทียมนั้นเครื่องรับได้ทราบแล้ว

CONTROL SEGMENT ส่วน ควบคุมดาวเทียมประกอบด้วย

- Master Control Station : สถานีควบคุมแม่ข่ายมีอยู่ 1 สถานี ทำหน้าที่รับผิดชอบในการจัดการทั่วไปและบริการสถานีลูกข่ายเป็นศูนย์กลางที่ให้การสนันสนุนการทำงานเครื่อง แม่ข่ายจะคำนวณตำแหน่งและนาฬิกาเพื่อดูความคลาดเคลื่อนของดาวเทียมแต่ละดวงจากสถานีลูกข่ายภาคพื้นและสั่งคำสั่งแก้ไขกลับไปยังสถานีลูกข่ายพื่อส่งไปยังดาวเทียมดวงนั้น ๆ
- Monitor Stations : สถานีควบคุมลูกข่ายมีอยู่ 4 สถานีจะทำการตรวจสอบความสูงตำแหน่ง ความเร็ว และวงจรทั่วไปของดาวเทียม สถานีควบคุมนี้ตรวจสอบดาวเทียมได้ครั้งละ 11 ดวง การตรวจสอบนี้แต่ละสถานีกระทำวันละ 2 ครั้งเมื่อดาวเทียมโคจรรอบโลก

OPERATION หลักพื้นฐานของระบบ GPS คือ วัดระยะทางระหว่างเครื่องรับในอากาศยาน กับดาวเทียมต่าง ๆ ในระบบดาวเทียมเองก็บอกแล้วว่าอยู่ที่ใดตำแหน่งใดในวงโคจรรอบโลก เครื่องรับรู้ระยะทางแน่ ๆ จากดาวเทียม และรู้ระยะทางระหว่างดาวเทียม เครื่องรับ GPS มีระบบคอมพิวเตอร์ ที่สามารถใช้วิธีทางคณิตศาสตร์สามารถที่จะคำนวณหาตำแหน่งของเครื่องรับแน่นอน [7]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9 Pocket Pc

พ็อคเกตพีซี (PocketPC) คือคอมพิวเตอร์ขนาดมือถือที่ใช้งานระบบปฏิบัติการวินโดวส์โมเบิล มันมีความสามารถของพีซีเดสก์ท็อปสมัยใหม่หลายอย่างในปัจจุบันฟรีแวร์หลายตัวและรวมทั้งก็ลักษณะเฉพาะโทรศัพท์เคลื่อนที่. พ็อคเกตพีซีสามารถใช้เป็นจีพีเอส (GPS) หรือตัวอ่านบาร์โค้ดและกล้องถ่ายรูปก็ได้ยกตัวอย่าง เช่น เครื่องมือใด ๆ ที่จำเป็นต้องถูกแยกประเภทในฐานะพีซีกระเป๋าก็ยังต้องทำดำเนินการด้วยระบบปฏิบัติการวินโดวส์ CE/Mobiles 2002, 2003, 5.0 (ฉบับพ็อคเกตพีซี) ซีพียูที่ร่วมกันได้คืออินเทล XScale [8]

2.10 .NET Framework

.NET Framework คือกรอบหรือโครงร่างการพัฒนาโปรแกรมคอมพิวเตอร์ซึ่ง Microsoft ได้เริ่มทำการพัฒนามาตั้งแต่เดือนกุมภาพันธ์ ปีค.ศ 2002 และได้รับการยอมรับของเหล่าโปรแกรม-เมอร์ทั้งยังมีการพัฒนาอย่างต่อเนื่อง โดยเทคโนโลยีนี้จะเน้นการทำงานของโปรแกรมต่างๆในรูปแบบของ Web service ซึ่งเป็นส่วนที่ทำให้แอพพริเคชั่นต่าง ๆ ของผู้ใช้สามารถติดต่อกันผ่านทางเครือข่ายอินเตอร์เน็ต ได้ง่ายขึ้นโดยที่โปรแกรมที่ทำการเชื่อมต่อกันนั้นไม่จำเป็นต้องอยู่บนระบบปฏิบัติการเดียวกัน เพราะฉะนั้น .NET framework นั้นไม่ใช่ภาษาที่ใช้เขียนโปรแกรมแต่เป็นชื่อเรียกโดยรวมของการพัฒนาโปรแกรมแบบใหม่ที่ Microsoft พัฒนาขึ้น และได้ทำการพัฒนา .NET Framework ที่มีขนาดเล็กลงเพื่อให้สามารถใช้พัฒนาโปรแกรมบนเครื่อง Pocket PC ได้ .NET Compact Framework คือ เป็น .NET Framework อีกเวอร์ชั่นที่มีขนาดเล็กทำงานได้รวดเร็วและตัดคุณสมบัติการทำงานบางส่วนของ .NET Framework ออกไปสำหรับการพัฒนาโปรแกรมสำหรับอุปกรณ์ต่าง ๆ โดยเฉพาะ โดยเราสามารถพัฒนาโปรแกรมสำหรับ PocketPC และ Windows CE ได้โดยใช้ภาษา C#.NET หรือ Visual Basic.NET ได้ [9]

2.11 PHP Web Service

Web Service คือ เว็บแอพพริเคชั่นยุคใหม่ที่ประกอบด้วยส่วนย่อย ๆ มีความสมบูรณ์ในตัวเองสามารถติดตั้งค้นหา เริ่มทำงานได้ผ่านเว็บ Web Service สามารถทำอะไรก็ได้ตั้งแต่งานง่ายๆ เช่นดึงข้อมูลจนถึงกระบวนการทางธุรกิจที่ซับซ้อนเมื่อ Web Service ตัวใดตัวหนึ่งเริ่มทำงาน Web Service ตัวอื่นก็สามารถรับรู้และเริ่มทำงานได้อีกด้วย

พื้นฐานของ Web Service ก็คือ XML กับ HTTP ซึ่งจะพบว่า HTTP ก็เป็นที่รู้จักกันดี และไปได้ทั่วทุกแห่งที่มีอินเตอร์เน็ต ส่วน XML คือภาษาสากลที่สามารถปรับแต่งได้ตามใจชอบเพื่อให้เกิดกิจกรรมระหว่างผู้ใช้บริการและบริการ หรือระหว่างส่วนประกอบต่าง ๆ เบื้องหลัง Web

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

server ก็คือ ข้อความ XML จะถูกแปลงให้การขอบริการจากเซิร์ฟเวอร์ของฐานข้อมูล และผลที่ได้ก็จะแปลงกลับมาในรูป XML

ส่วนประกอบของ Web Service ประกอบด้วย

แอพพลิเคชั่น สิ่งที่ขาดไม่ได้ก็คือแอพพลิเคชั่นที่ให้บริการเว็บเซอร์วิสนั่นเอง แอพพลิเคชั่นที่จะให้บริการเว็บเซอร์วิส ควรจะอยู่บนเครื่องเว็บเซิฟเวอร์ที่เปิดให้บริการตลอดเวลา สามารถติดต่อด้วยโพรโตคอล HTTP ได้ และพัฒนาด้วยภาษาที่มีความสามารถจัดการกับ SOAP โดยอาจเป็นโมดูลเสริม หรือมีคลาสให้เรียกใช้งานก็ได้ไม่อย่างนั้นอาจจะต้องพัฒนาโปรแกรม ให้สามารถรับ/ส่งข้อมูลในรูปแบบ SOAP

SOAP (Simple Object Access Protocol) คือโปรโตคอลหรือระเบียบวิธีในการสื่อสารระหว่างเว็บเซอร์วิส โดยใช้ข้อมูลที่กำหนดรูปแบบด้วยภาษา XML ทำให้เว็บเซอร์วิสสามารถสื่อสารกันได้แม้ว่าจะอยู่บนเครื่องคอมพิวเตอร์คนละเพลตฟอร์ม หรือพัฒนาด้วยภาษาโปรแกรมที่ต่างกันก็ตาม

เมื่อผู้พัฒนาแอพพลิเคชั่นต้องการใช้งานเว็บเซอร์วิสก็เพียงแค่เขียนโปรแกรมเพื่อติดต่อกับโมดูล SOAP ในภาษาที่ตนใช้ จากนั้น SOAP ก็จะสร้าง SOAP message เพื่อติดต่อกับแอพพลิเคชั่นปลายทางให้โดยอัตโนมัติ

Web Services Description Language (WSDL) คือ เอกสาร XML ที่อธิบายรายละเอียดในการติดต่อกับเว็บเซอร์วิสเพื่อให้แอพพลิเคชั่นที่ต้องการเรียกใช้เว็บเซอร์วิสรู้ว่าเซอร์วิสนั้นให้บริการอะไรบ้าง และจะติดต่อได้อย่างไร

UDDI (Universal Description, Discovery and Integration) เป็นไดเรกทอรีที่เก็บรวบรวม Web Service ที่มีการลงทะเบียนไว้ ซึ่งอาจรวมไปถึงบริการอื่นๆที่เป็นอิเลคทรอนิกส์ และไม่เป็นอิเลคทรอนิกส์ด้วย UDDI จะเก็บรวบรวมข้อมูลของ Web Service ต่างๆไว้ในรูปแบบ WSDL

หน้าที่ของ UDDI จะคล้ายกับเว็บไดเรกทอรี กล่าวคือ UDDI ช่วยให้ผู้พัฒนา Web Service ได้ประกาศหรือประชาสัมพันธ์บริการของตนเองสู่สาธารณะ และช่วยให้ผู้ใช้ Web Service ค้นพบ Web Service ที่ต้องการใช้งาน [10]

2.12 Opera Mobile

โอเปร่า (Opera) คือซื่อซอฟต์แวร์ที่รวมเว็บเบราว์เซอร์และโปรแกรมสำหรับใช้งานอินเทอร์เน็ตอื่น ๆ พัฒนาโดยบริษัทโอเปร่าประเทศนอร์เวย์ในปัจจุบันโอเปร่าเป็นผู้นำในตลาดเว็บเบราว์เซอร์สำหรับโทรศัพท์มือถือและพีดีเอ นอกจากนี้ยังถูกนำไปใช้ในระบบโทรทัศน์ที่มีการโต้ตอบระหว่างผู้ชม (interactive television, iTV) ในบางประเทศ เมื่อไม่นานนี้บริษัทโอเปร่าได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ร่วมมือกับบริษัทอะโดบีซิสเต็มส์ ผู้ผลิตซอฟต์แวร์ด้านการสร้างสรรค์สื่อเพื่อรวมโอเปร่ากับโปรแกรมในชุดอะโดบีครีเอทีฟสวีท [11]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบโครงงาน

3.1 โครงสร้างตารางฐานข้อมูล (Data Dictionary)

ฐานข้อมูลที่ได้จากห้องปฏิบัติการเทคโนโลยีเครือข่าย (NTL) ภายใต้ศูนย์วิจัยอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งประเทศ มีดังนี้

3.1.1 ฐานข้อมูลของระบบค้นหาสถานที่และระบบค้นหาเส้นทาง

ตารางที่ 3.1 แสดงข้อมูลของตาราง mainroad

Field	Constraint	Type	Description
Gid	Primary 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	Bigint	ชื่อถนน
Startx_coo	Not null	Double	จุดเริ่มต้นพิกัด x
Starty_coo	Not null	Double	จุดเริ่มต้นพิกัด y
Endx_coord	Not null	Double	จุดเริ่มต้นพิกัด x
Endy_coord	Not null	Double	จุดเริ่มต้นพิกัด y
The_geom	CHECK(geometrytype(the_geom))	Geometry	พิกัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.2 แสดงข้อมูลของตาราง 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 3.3 แสดงข้อมูลของตาราง 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	สถานศึกษา
The_geom	CHECK(geometrytype(the_geom))	Geometry	พิกัดของโรงพยาบาล

ตารางที่ 3.4 แสดงข้อมูลของตาราง 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	สถานศึกษา
The_geom	CHECK(geometrytype(the_geom))	Geometry	พิกัดของโรงแรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.5 แสดงข้อมูลของตาราง police

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

ตารางที่ 3.6 แสดงข้อมูลของตาราง temple

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

ตารางที่ 3.7 แสดงข้อมูลของตาราง 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	สถานศึกษา
The_geom	CHECK(geometrytype(the_geom))	Geometry	พิกัดของสถานที่ราชการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.8 แสดงข้อมูลของตาราง mainroad_in

Field	Constraint	Type	Description
Gid	Primary 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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.9 แสดงข้อมูลของตาราง mainroad_out

Field	Constraint	Type	Description
Gid	Primary 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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.10 แสดงข้อมูลของตาราง traffic_dens

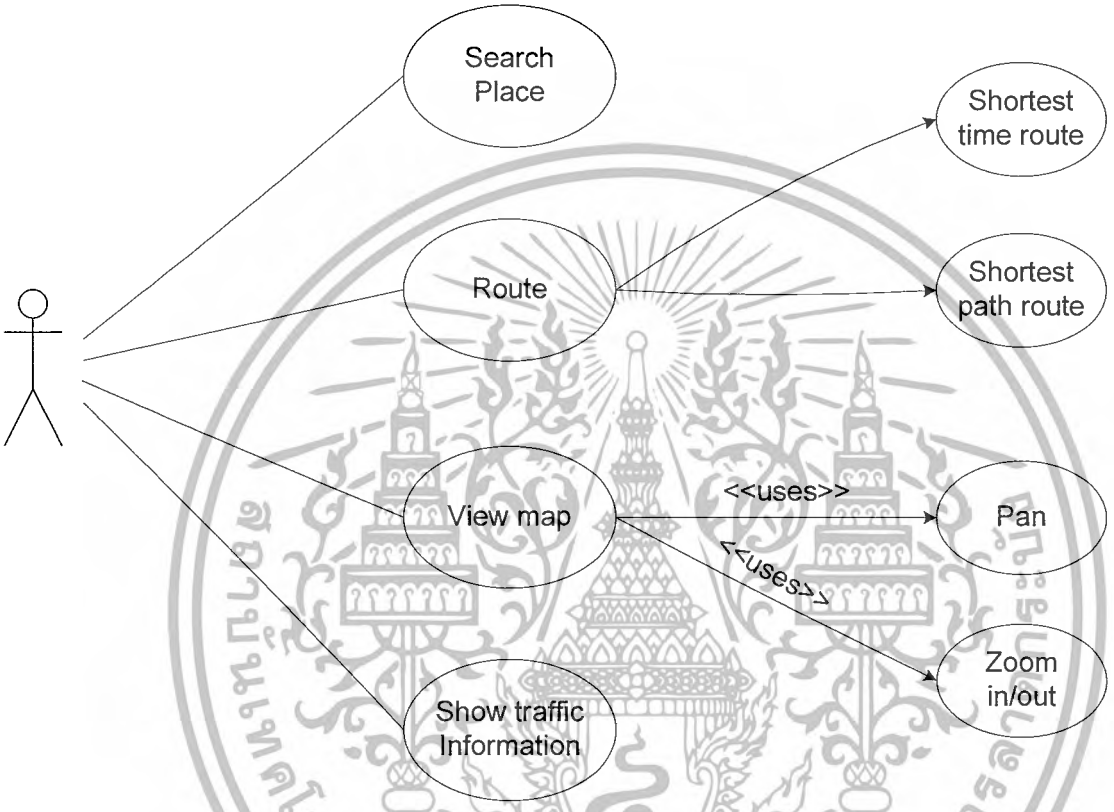
Field	Constraint	Type	Description
Gid	Primary key	Serial	ลำดับของข้อมูล
Fnode_	Not null	Bigint	Form node
Tnode_	Not null	Bigint	To node
Length	Not null	Double	ระยะทางระหว่าง node
Mainroad_	Not null	Bigint	ลำดับของถนนหลัก
Mainroad_i	Not null	Bigint	หมายเลขของถนนหลัก
Rdlnnamt	CHECK	Bigint	ชื่อถนน
Startx_coo	Not null	Double	จุดเริ่มต้นพิกัด x
Starty_coo	Not null	Double	จุดเริ่มต้นพิกัด y
Endx_coord	Not null	Double	จุดเริ่มต้นพิกัด x
Endy_coord	Not null	Double	จุดเริ่มต้นพิกัด y
The_geom	CHECK(geometrytype(the_geom))	Geometry	พิกัด
Density_ratio	Not null	Double	อัตราความหนาแน่นของสภาพการจราจร
Density_cost	Not null	Double	ค่าความหนาแน่นของสภาพการจราจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การออกแบบระบบ

3.2.1 Use Case Diagram

Bangkok Intelligent Navigation System



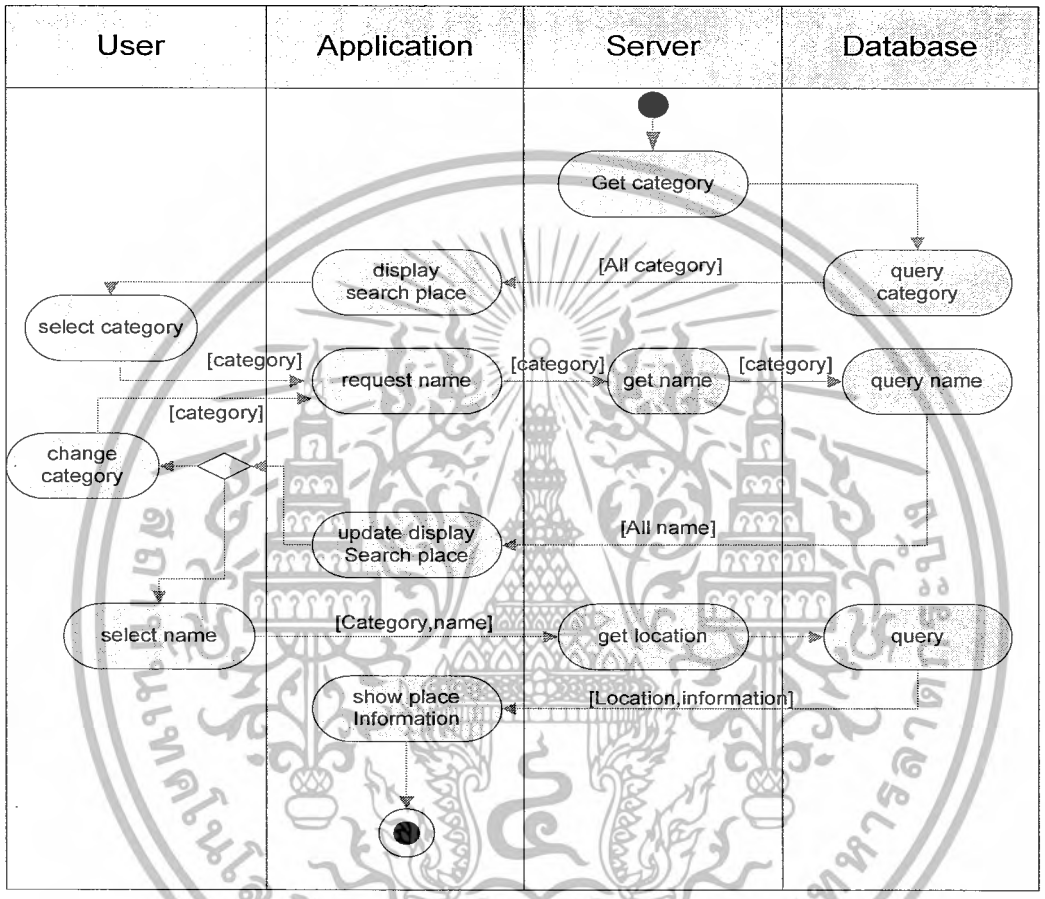
รูปที่ 3.1 Use case diagram ของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 Activity diagram

1) ค้นหาสถานที่ (Search Place)

Search Place

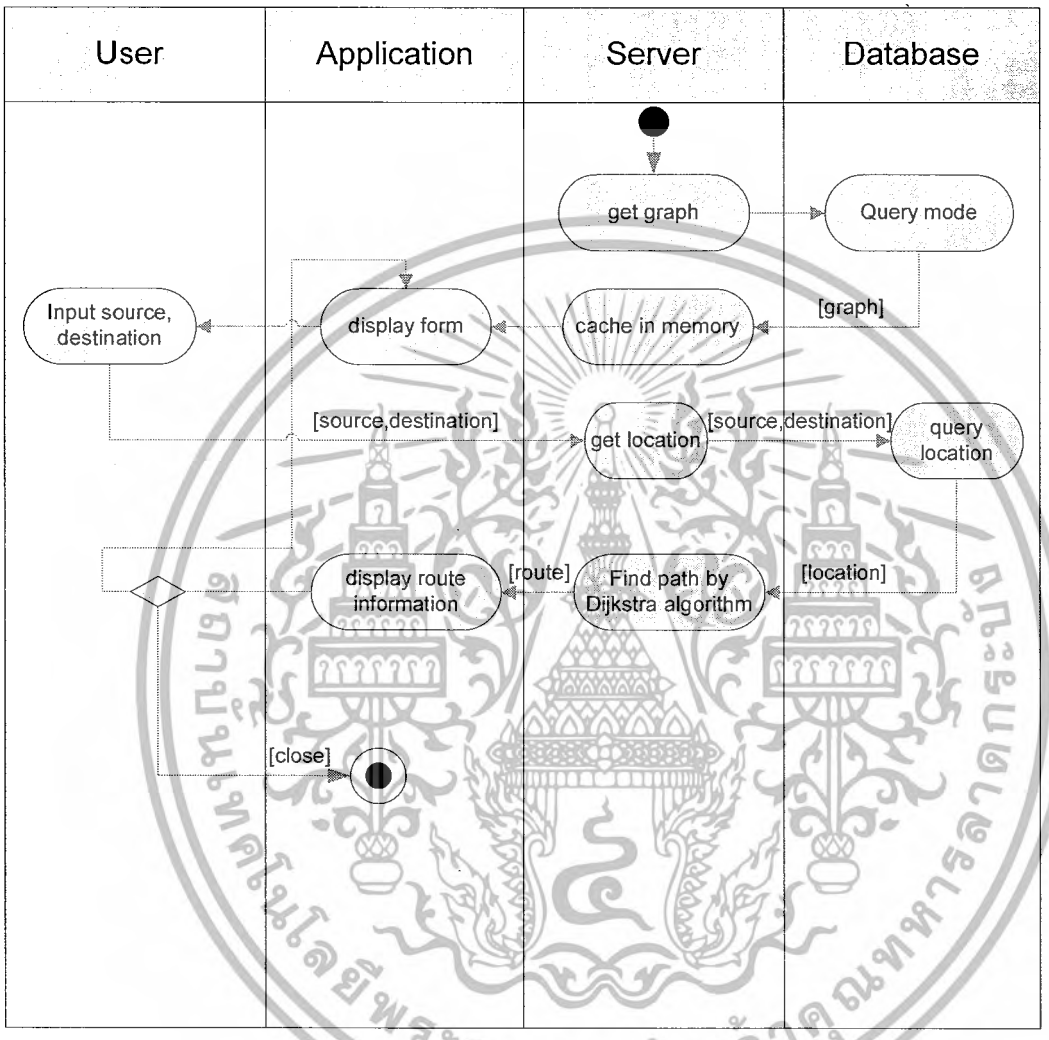


รูปที่ 3.2 Activity diagram การค้นหาสถานที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) คำนวณเส้นทางที่ใช้ระยะทางสั้นที่สุด (Shortest path route)

Shortest path Route

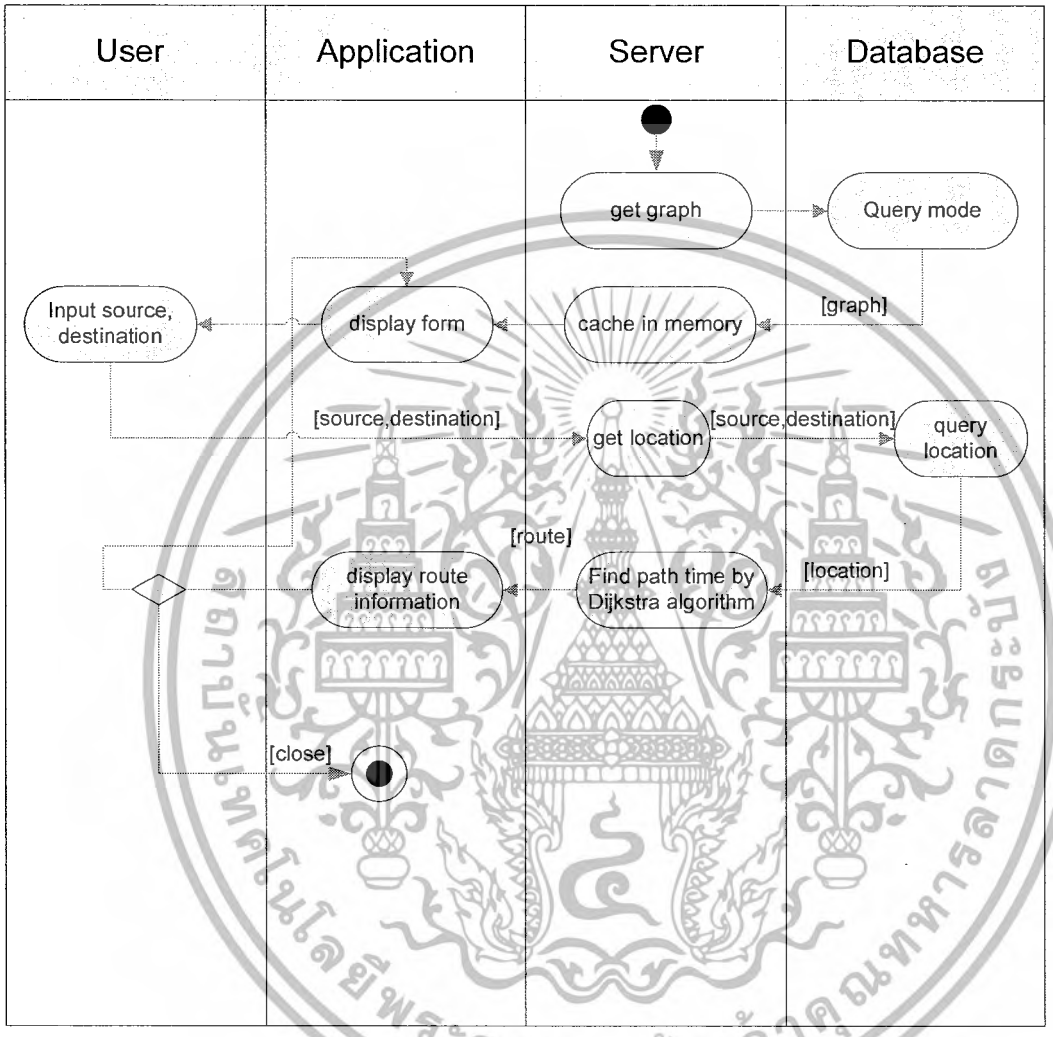


รูปที่ 3.3 Activity diagram การคำนวณระยะทางที่สั้นที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) คำนวณเส้นทางที่ใช้เวลาน้อยที่สุด (Shortest time Route)

Shortest time Route

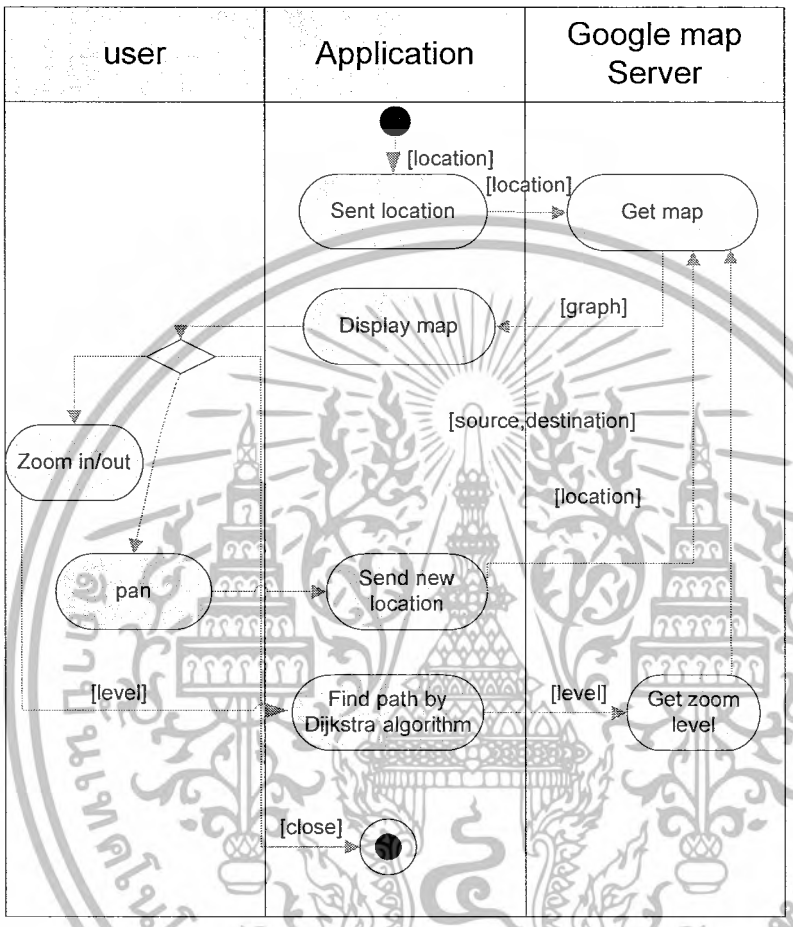


รูปที่ 3.4 Activity diagram การคำนวณเส้นทางที่ใช้เวลาน้อยที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) แสดงแผนที่ (View map)

Veiw Map

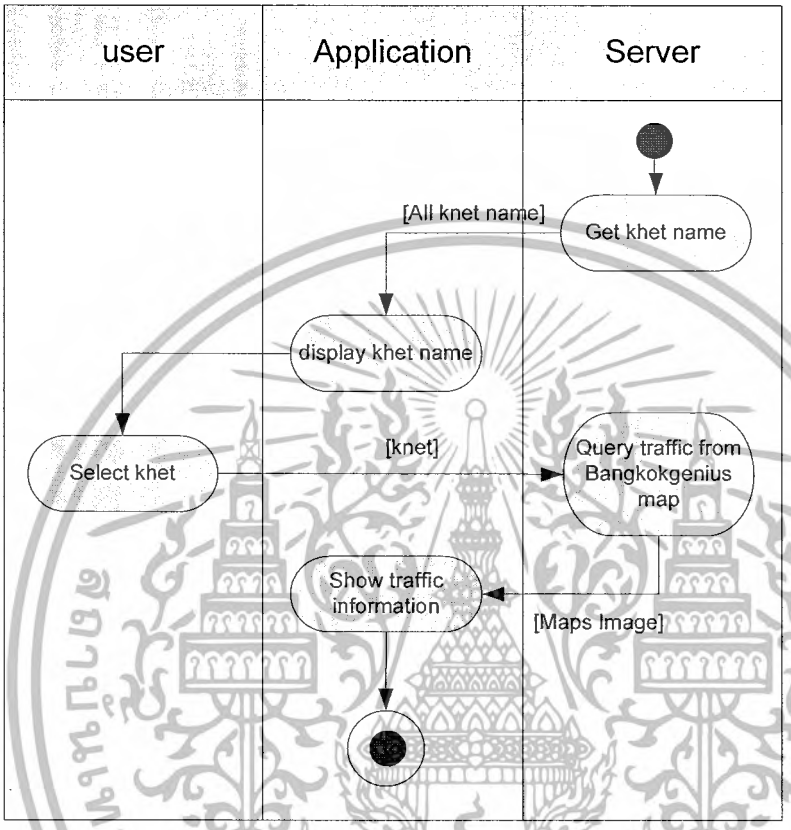


รูปที่ 3.5 Activity diagram ของการดูแผนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) แสดงสภาพจราจร (Traffic information system)

Traffic information system

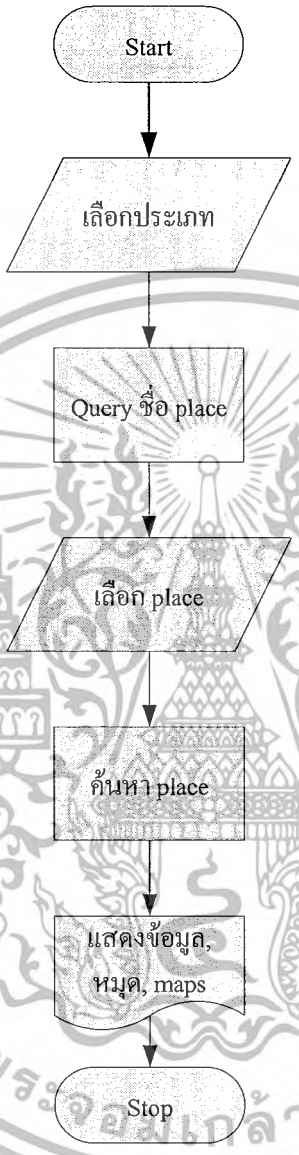


รูปที่ 3.6 Activity diagram ของการแสดงสภาพจราจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3 System Flowchart

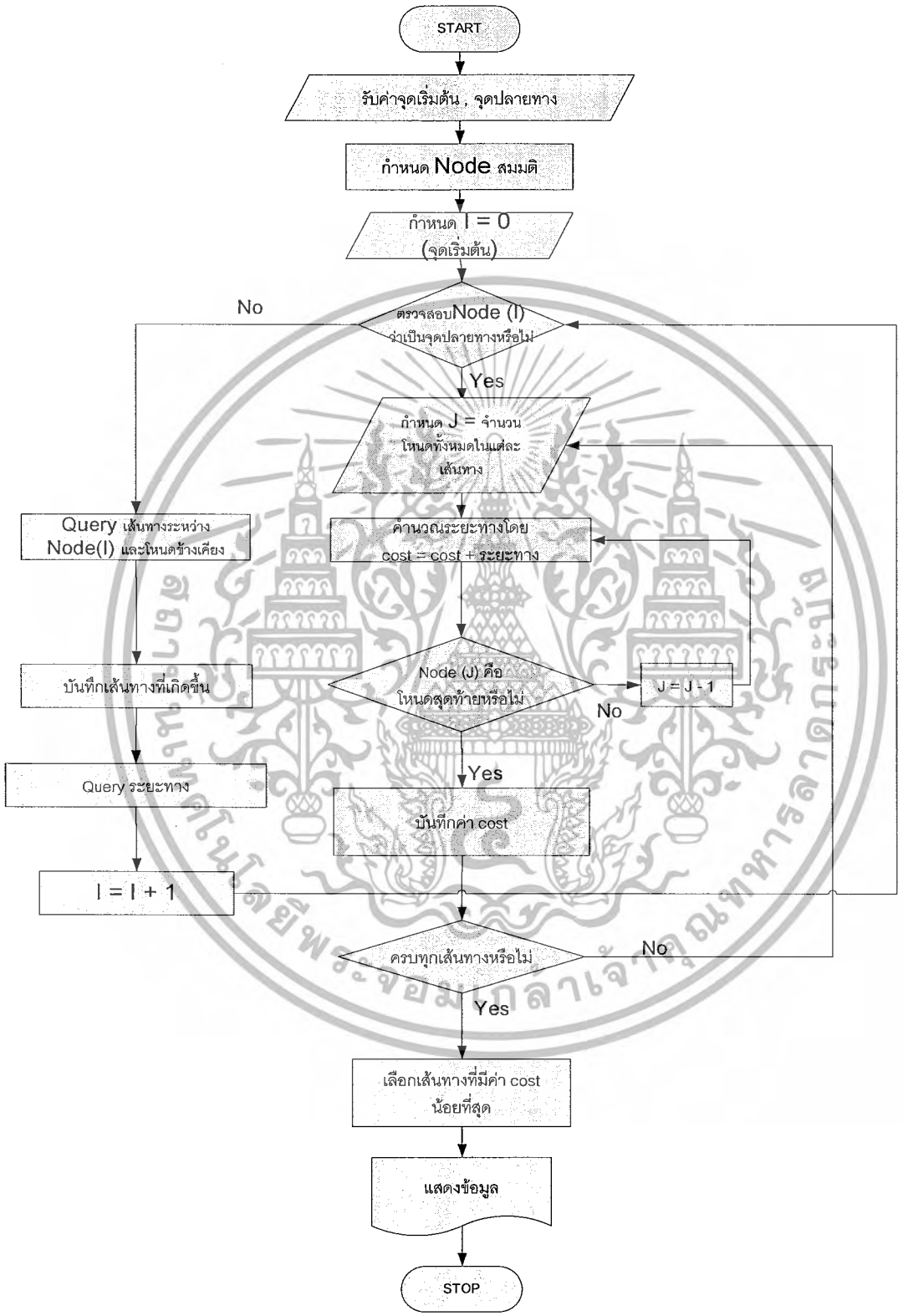
1) ระบบค้นหาสถานที่สำคัญในกรุงเทพมหานคร



รูปที่ 3.7 ระบบการไหลของข้อมูลการทำงานการหาสถานที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

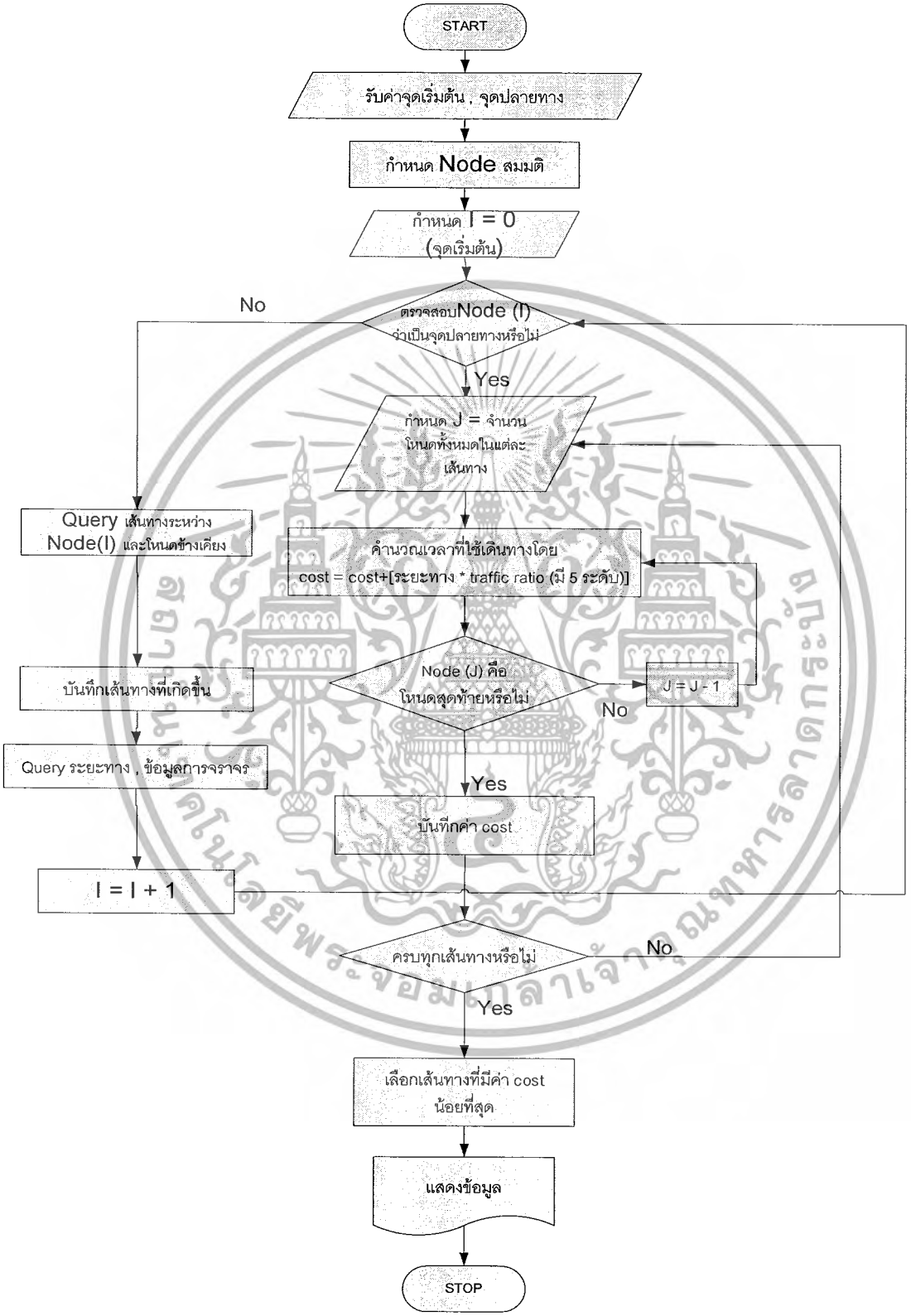
2) ระบบคำนวณหาเส้นทางที่ใช้ระยะทางสั้นที่สุดโดยใช้อัลกอริทึม Dijkstra



รูปที่ 3.8 ระบบการไหลของข้อมูลการทำงานของระบบคำนวณเส้นทางที่สั้นที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ระบบคำนวณหาเส้นทางที่ใช้ระยะเวลาน้อยที่สุดโดยใช้อัลกอริทึม Dijkstra



รูปที่ 3.9 ระบบการไหลของข้อมูลการทำงานของระบบคำนวณเส้นทางที่ใช้เวลาน้อยที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

4.1 ขั้นตอนการใช้งานโปรแกรม

4.1.1 ฟังก์ชันการใช้งาน PLACE

รูปที่ 4.1 ฟังก์ชันการใช้งาน PLACE ใช้เพื่อค้นหาแผนที่ ของสถานที่ที่ต้องการทราบข้อมูล

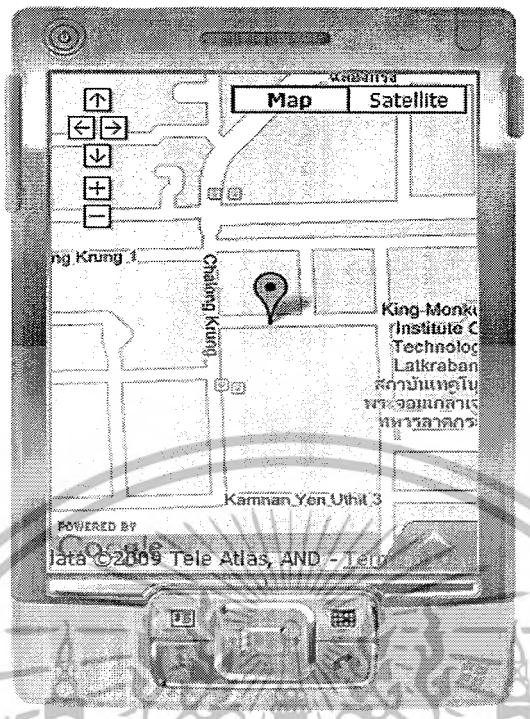
รูปที่ 4.2 เลือกประเภทของสถานที่ที่ต้องการทราบข้อมูล จากนั้นกดปุ่ม “ต่อไป >”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

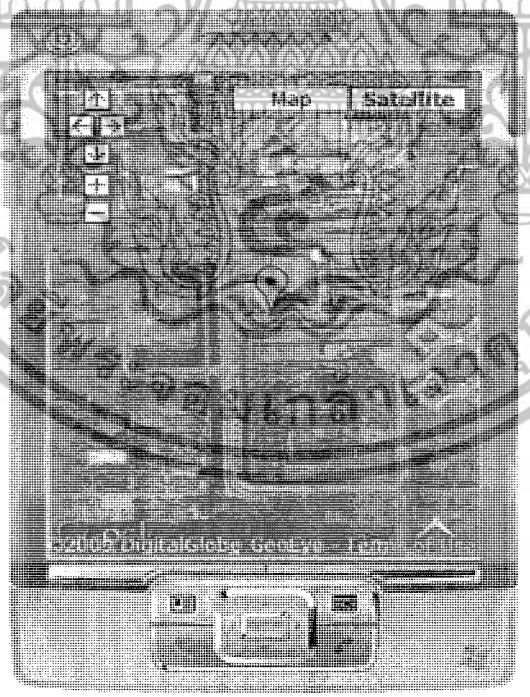
รูปที่ 4.3 เลือกชื่อสถานที่ที่ต้องการทราบข้อมูล จากนั้นกดปุ่ม “ต่อไป >”

รูปที่ 4.4 กดปุ่ม “Launch MAPS” เพื่อแสดงแผนที่จาก Google Maps

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



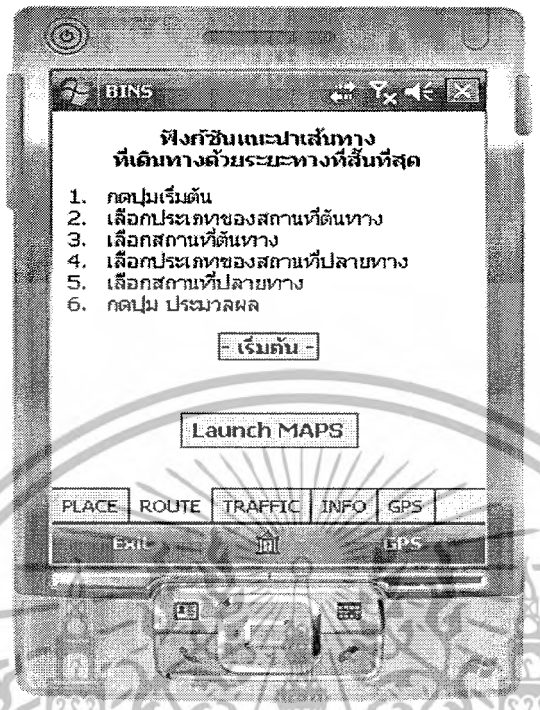
รูปที่ 4.5 เมื่อกดปุ่ม “ค้นหาสถานที่” แผนที่จะแสดงตำแน่งของสถานที่ที่ต้องการทราบข้อมูล



รูปที่ 4.6 สามารถเปลี่ยนแผนที่ให้เป็นรูปแบบแผนที่ภาพถ่ายดาวเทียมได้ทันทีด้วยปุ่ม “Satellite”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

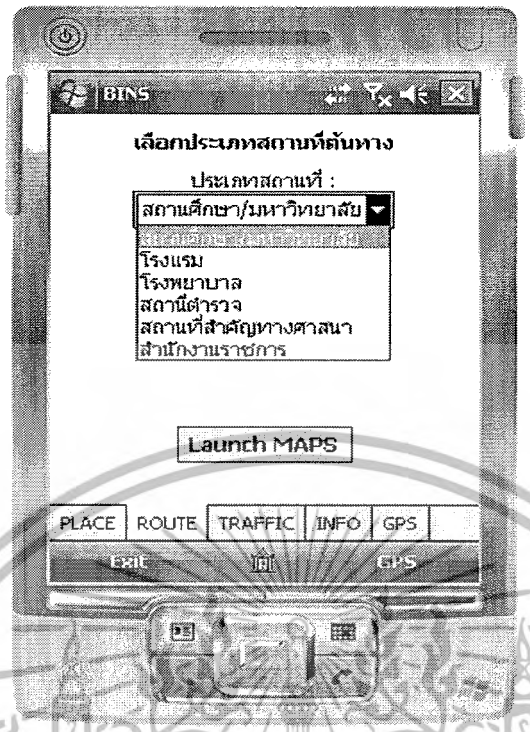
4.1.2 ฟังก์ชันการใช้งาน ROUTE



รูปที่ 4.7 ฟังก์ชันการใช้งาน ROUTE

รูปที่ 4.8 ใช้เพื่อค้นหาเส้นทางที่ใช้ระยะทางในการเดินทางสั้นที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 เลือกประเภทของสถานที่ต้นทาง จากนั้นกดปุ่ม “ต่อไป >”

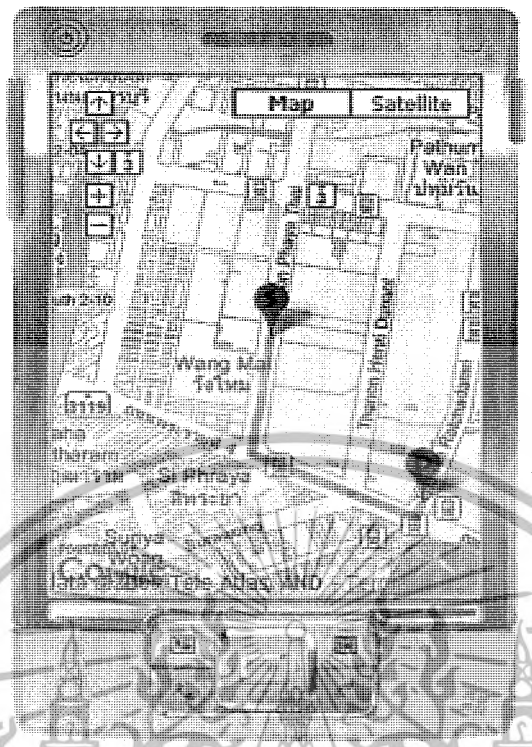
รูปที่ 4.10 จากนั้นเลือกชื่อของสถานที่ต้นทาง จากนั้นกดปุ่ม “ต่อไป >”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

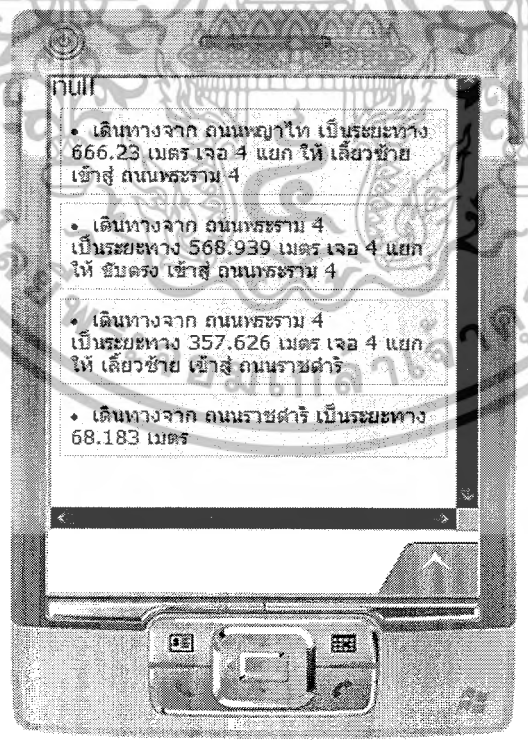
รูปที่ 4.11 เลือกประเภทของสถานที่ปลายทาง จากนั้นกดปุ่ม “ต่อไป >”

รูปที่ 4.12 เลือกชื่อสถานที่ปลายทาง จากนั้นกดปุ่ม “ต่อไป >” จากนั้นกดปุ่ม “Launch MAPS”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



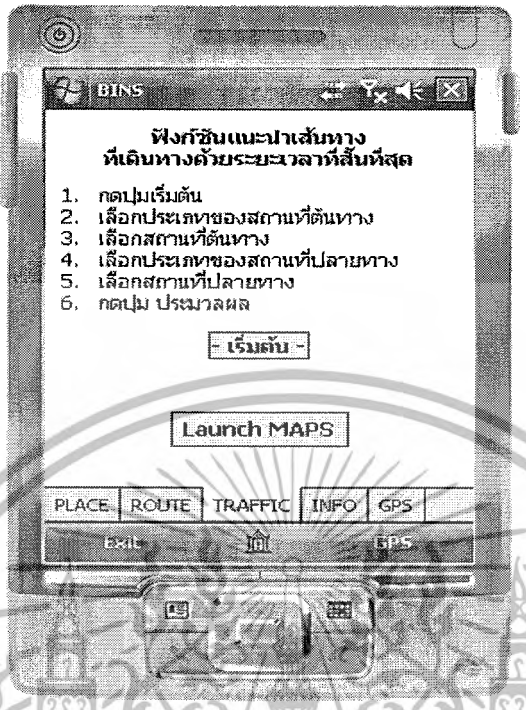
รูปที่ 4.13 เมื่อกดปุ่ม "ค้นหาเส้นทาง" แผนที่จะแสดงเส้นทางที่สั้นที่สุดจากต้นทาง – ปลายทาง



รูปที่ 4.14 โดยด้านล่างจะมีช่อง Direction list เพื่อแสดงข้อความการนำทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

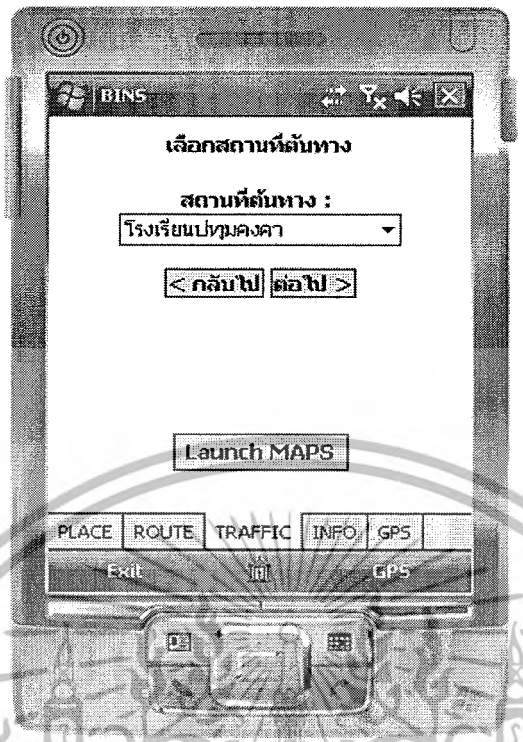
4.1.3 ฟังก์ชันการใช้งาน **TRAFFIC**



รูปที่ 4.15 ฟังก์ชันการใช้งาน TRAFFIC

รูปที่ 4.16 เลือกประเภทของสถานที่ต้นทาง จากนั้นกดปุ่ม “ต่อไป >”

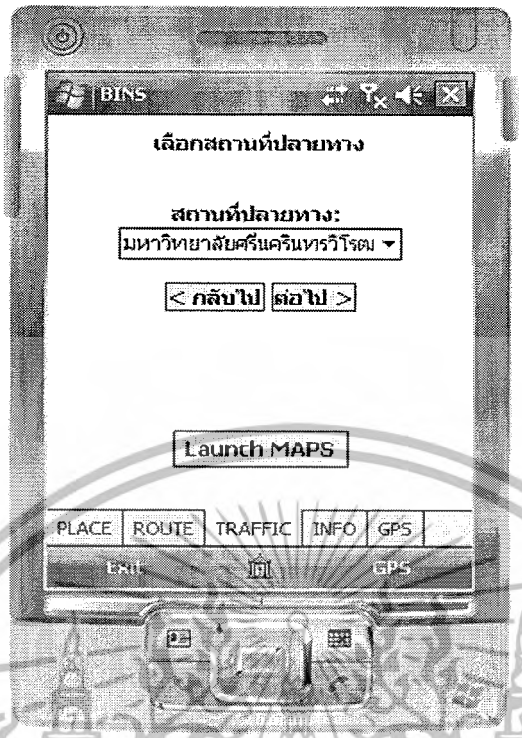
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.17 จากนั้นเลือกชื่อของสถานที่ต้นทาง จากนั้นกดปุ่ม “ต่อไป >”

รูปที่ 4.18 เลือกประเภทของสถานที่ปลายทาง จากนั้นกดปุ่ม “ต่อไป >”

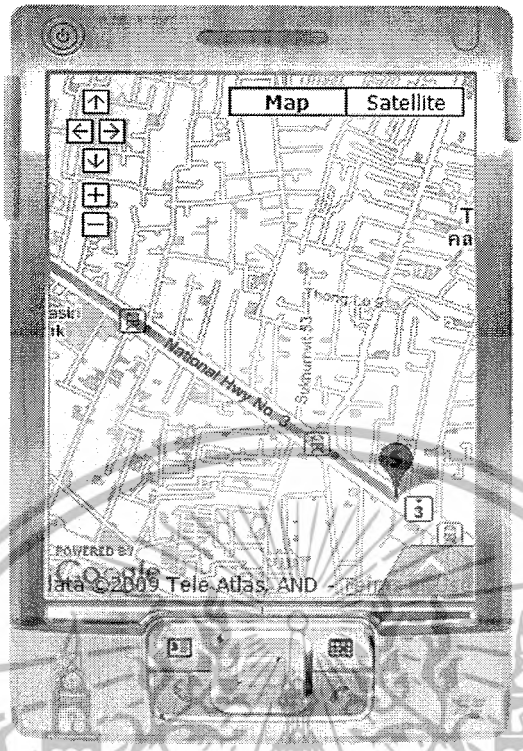
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



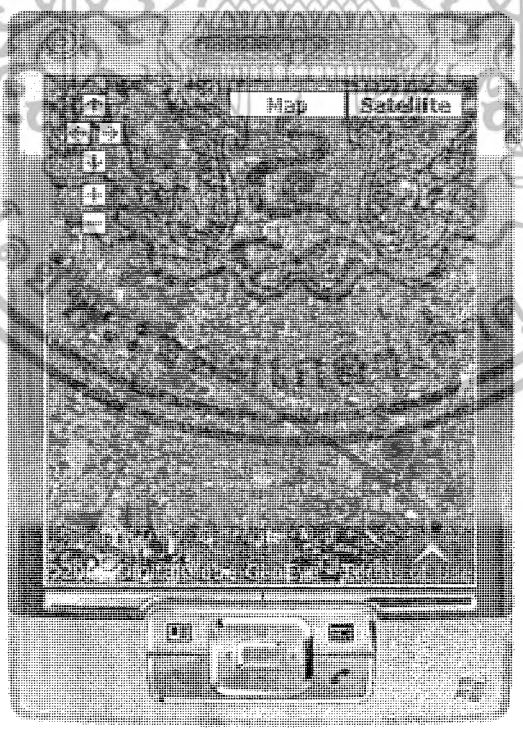
รูปที่ 4.19 เลือกชื่อสถานที่ปลายทาง จากนั้นกดปุ่ม “ต่อไป >”

รูปที่ 4.20 กดปุ่ม “Launch MAPS” เพื่อแสดงแผนที่บน Google Maps

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

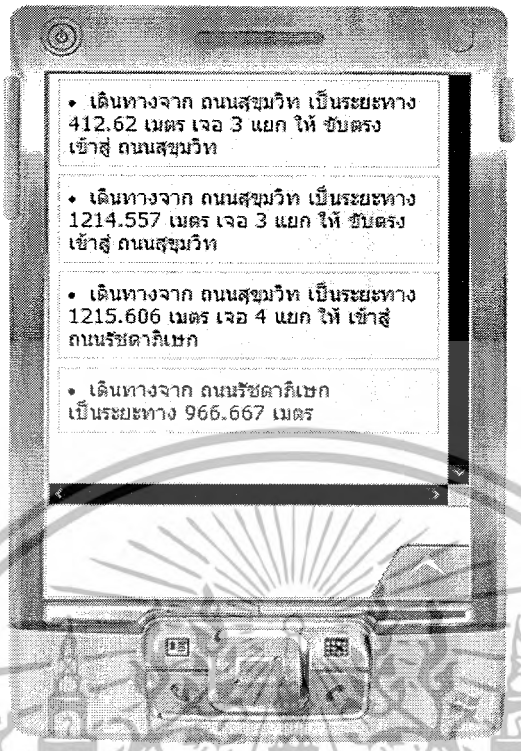


รูปที่ 4.21 กดปุ่ม "ค้นหาเส้นทาง" แผนที่จะแสดงเส้นทางที่ใช้เวลาสั้นสุดจากต้นทาง – ปลายทาง



รูปที่ 4.22 สามารถเปลี่ยนรูปแบบภาพเป็นภาพถ่ายดาวเทียมได้ด้วยปุ่ม "Satellite"

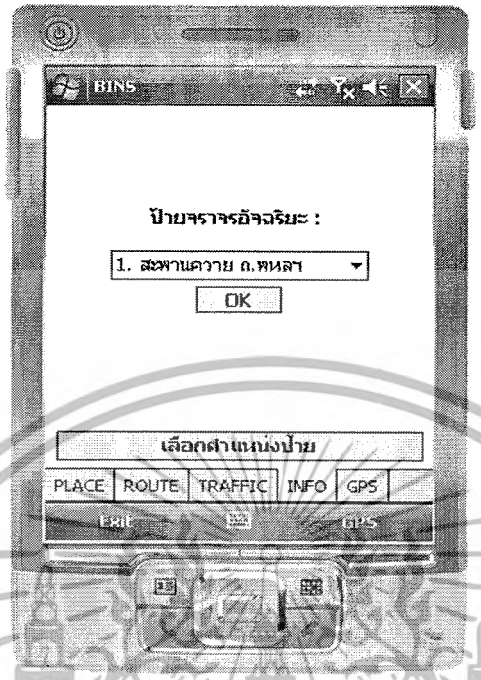
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



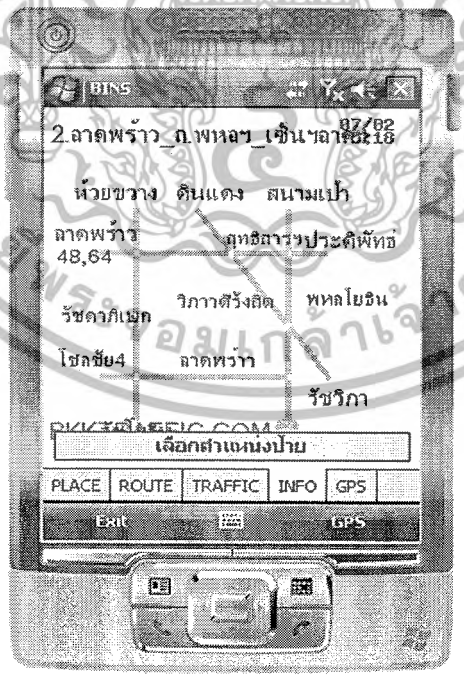
รูปที่ 4.23 โดยด้านล่างจะมีช่อง Direction list เพื่อแสดงข้อความการนำทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.4 ฟังก์ชันการใช้งาน INFO



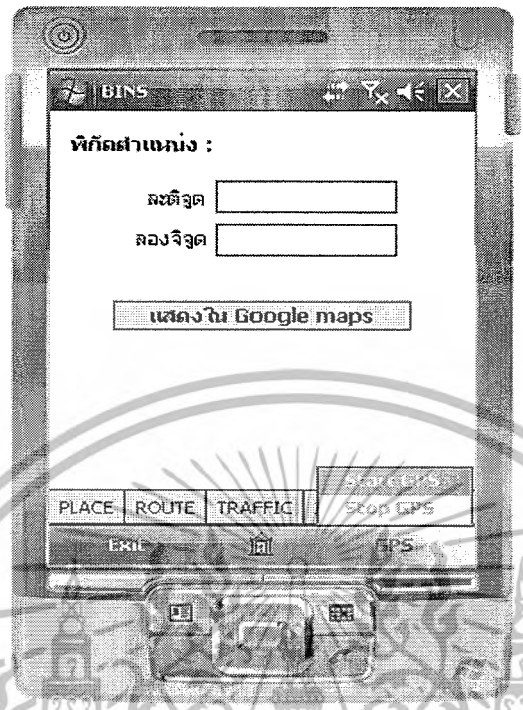
รูปที่ 4.24 เลือกตำแหน่งที่ติดตั้งป้ายจราจรอัจฉริยะ จากนั้นกดปุ่ม “OK”



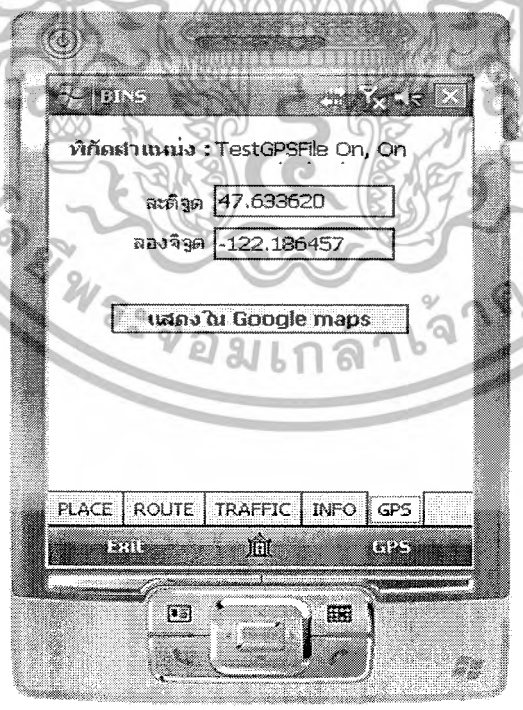
รูปที่ 4.25 ระบบจะแสดงผลสภาพการจราจร จากป้ายจราจรอัจฉริยะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.5 ฟังก์ชันการใช้งาน GPS

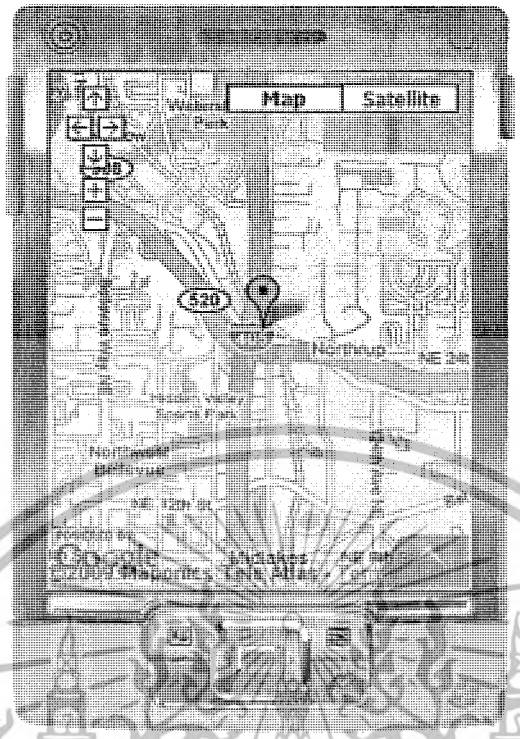


รูปที่ 4.26 กดปุ่มเปิดการใช้งาน GPS

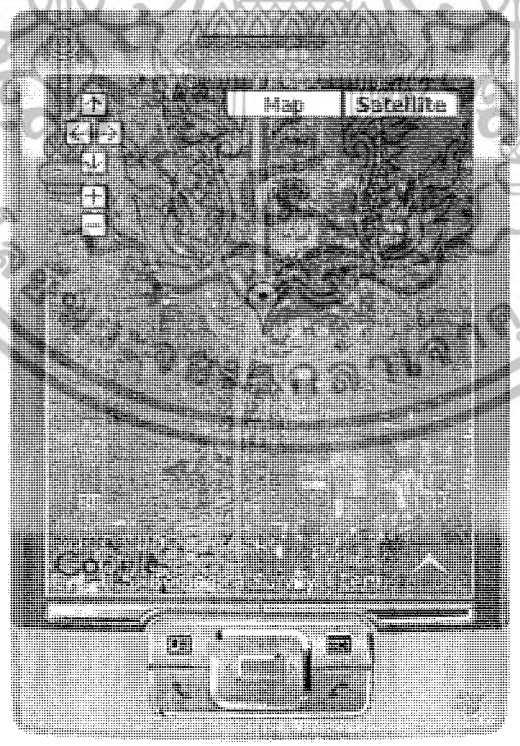


รูปที่ 4.27 กดปุ่ม "แสดงใน Google maps" เพื่อแสดงสถานที่ที่สัมพันธ์กับพิกัด GPS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.28 Google maps จะแสดงแผนที่ ที่ตั้งของพิกัด GPS จริง



รูปที่ 4.29 สามารถเปลี่ยนรูปแบบภาพเป็นภาพถ่ายดาวเทียวโดยกดปุ่ม “Satellite”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการทดลอง

5.1 สรุปการพัฒนาโครงงาน

ระบบสามารถคำนวณหาเส้นทางที่เหมาะสมระหว่างสถานที่ 2 สถานที่ และคำนวณเวลาที่คาดว่าจะใช้ในการเดินทาง ซึ่งอยู่ในพื้นที่ของกรุงเทพมหานครและสามารถแสดงสภาพการจราจรซึ่งมาจากป้ายจราจรอัจฉริยะที่ติดตั้งอยู่ 40 จุด ทั่วกรุงเทพมหานคร มาแสดงบนแอพพริเคชั่นโดยมีตัวแปรที่นำมาคำนวณคือ ระยะทางซึ่งวัดจากแผนที่จริงและสภาพการจราจรที่ได้มาจากการสุ่มค่าลงในฐานข้อมูล และใช้แผนที่จริงในการแสดงผลได้อย่างชัดเจน อีกทั้งระบบยังรองรับ GPS ทำให้ผู้ใช้สามารถระบุตำแหน่งที่อยู่ปัจจุบันได้ โดยระบบนี้เป็นแอพพลิเคชันบนโทรศัพท์มือถือ (Mobile Application) ที่ใช้ระบบปฏิบัติการวินโดว์โมบาย 6.0

5.2 ปัญหาที่พบในการทำโครงงาน

- ซอฟท์แวร์ที่สนับสนุนการใช้งานฐานข้อมูลทางด้าน GIS มีน้อยและมีราคาแพง ถึงจะมีแบบเปิดเผยรหัสแต่ใช้งานยาก
- เส้นทางในกรุงเทพฯ มีจำนวนมากและซับซ้อนทำให้ไม่สามารถรวบรวมมาได้ครบถ้วน และต้องใช้เวลาอย่างมากในการเพิ่มข้อมูลลงในฐานข้อมูล ซึ่งทำให้เกิดความผิดพลาดบ่อยครั้ง
- เว็บเบราเซอร์ที่สนับสนุนการใช้งาน AJAX บน PocketPC มีน้อย และเครื่องมือสนับสนุนเว็บเบราเซอร์ที่มากับชุดภาษา C# ไม่สนับสนุนการใช้งาน AJAX เช่นเดียวกัน
- ในการคำนวณหาเส้นทางมีเงื่อนไขที่ซับซ้อนและหลากหลาย เช่น การที่สถานที่บางสถานที่ตั้งอยู่ติดถนนหลายถนนทำให้มีแยกที่ติดกับสถานที่จำนวนมาก หรือการต้องออกแบบโปรแกรมให้ป้องกันการเกิดลูป (loop) และการวกกลับของเส้นทาง ทำให้ยากต่อการพัฒนาให้ครอบคลุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 ข้อจำกัดของโครงงาน

- ไม่มีข้อมูลของเส้นทางในซอยทางลัด และทางด่วน
- ข้อมูลเส้นทางไม่ครอบคลุมทั่วกรุงเทพฯ
- เนื่องจากมีการคำนวณข้อมูลปริมาณมากทำให้ต้องใช้เวลานานในการประมวลผล
- ระบบไม่มีการคำนึงถึงทิศทางการเดินรถจริง คือ ถนนที่มีการเดินรถทางเดียว การห้ามเลี้ยว การห้ามกลับรถ เป็นต้น
- สมรรถนะของเครื่อง PocketPCไม่สูง ทำให้การตอบสนองต่อการใช้งานไม่รวดเร็ว
- ระบบฐานข้อมูลบนเครื่อง PocketPC มีความซับซ้อนและใช้งานได้ยาก

5.4 แนวทางในการพัฒนาต่อ

- เพิ่มฐานข้อมูลให้สมบูรณ์ครอบคลุมทุกพื้นที่ รวมไปถึงเส้นทางที่เป็นซอยทางลัด และทางด่วน
- พัฒนาโปรแกรมให้มีการคำนวณที่น้อยลงเพื่อลดเวลาในการประมวลผล
- พัฒนาให้ข้อมูลสภาพจราจรของถนนต่าง ๆ แบบเรียลไทม์ (Real time) และครอบคลุมพื้นที่ในเขตที่ยังไม่มีฐานข้อมูล
- พัฒนาให้ระบบคำนึงถึงทิศทางการเดินรถจริง คือ ถนนที่มีการเดินรถทางเดียว การห้ามเลี้ยว การห้ามกลับรถ เป็นต้น
- พัฒนาแอพพริเคชั่นให้สามารถใช้พิกัดจาก GPS เป็นต้นทางในการคำนวณ
- พัฒนาแอพพริเคชั่นให้แสดงตำแหน่งปัจจุบันแบบเปลี่ยนแปลงทันที (Real time) โดยใช้ GPS ในการระบุตำแหน่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] นางสาวสสิภัทรา ศรีกฤษณ์, นายอำนาจ เฮงประสิทธิ์วงศ์. 2551. ระบบแนะนำเส้นทางในเขตกรุงเทพมหานคร (Bangkok Navigation Service System For Bangkok Metropolitan Area). กรุงเทพ. สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

[2] http://www.dgr.go.th/water2006/technique37.html

[3] http://iwis.pcd.go.th/IWIS/document/other/postgres.htm

[4] http://angsila.compsci.buu.ac.th/sc443183/Linux/Report.doc

[5] [http://www.gisthai.org/about-gis/gis.html

[6] วิชา ศิริธรรมจักร. 2549. Web Programming ด้วย Ajax และ PHP. พิมพ์ครั้งที่ 2. กรุงเทพ. บริษัทเคทีพี คอมพ์แอนด์คอนซัลท์ จำกัด.

[7] http://www.thaitechnics.com/nav/gps_t.html

[8]http://th.wikipedia.org/wiki/%E0%B8%9E%E0%B9%87%E0%B8%AD%E0%B8%84%E0%B9%80%E0%B8%81%E0%B8%95%E0%B8%9E%E0%B8%B5%E0%B8%8B%E0%B8%B5

[9] http://coresharp.net/blogs/article/archive/2008/01/20/net-compact-framework-pocket-pc-1.aspx

[10] http://truehits.net/faq/webmaster/webservice/#desc_soap

[11]http://th.wikipedia.org/wiki/%E0%B9%82%E0%B8%AD%E0%B9%80%E0%B8%9B%E0A3*E0*B9%8C)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้