

**สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง**

**ระบบค้นหาเส้นทางอ้างอิงจากสภาพการจราจร**

**A NAVIGATION SYSTEM BASED ON TRAFFIC CONDITIONS**



T119535



พิชญ์ เกรียงไกรวสิน

PITCH KREANGKRIWASIN

ศิริพร ตันตินราโรจน์

SIRIPORN TUNTINARAROAT

เลขหมู่.....  
เลขทะเบียน.....**119535**  
วัน,เดือน,ปี.....**- 8 S.A. 2554**

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

**ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต**

**สาขาวิชาวิศวกรรมสารสนเทศ**

**คณะวิศวกรรมศาสตร์**

**สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง**

**ปีการศึกษา 2553**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# A NAVIGATION SYSTEM BASED ON TRAFFIC CONDITIONS



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

**ACADEMIC YEAR 2010**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์

ระบบค้นหาเส้นทางอ้างอิงจากสภาพการจราจร

A Navigation System based on Traffic Conditions

รายชื่อนักศึกษา

นายพิชญ์ เกรียงไกรวศิน

รหัสนักศึกษา 50011090

นางสาวศิริพร ตันตินราโรจน์

รหัสนักศึกษา 50011564

ปริญญา

วิศวกรรมศาสตรบัณฑิต

สาขาวิชา

วิศวกรรมสารสนเทศ

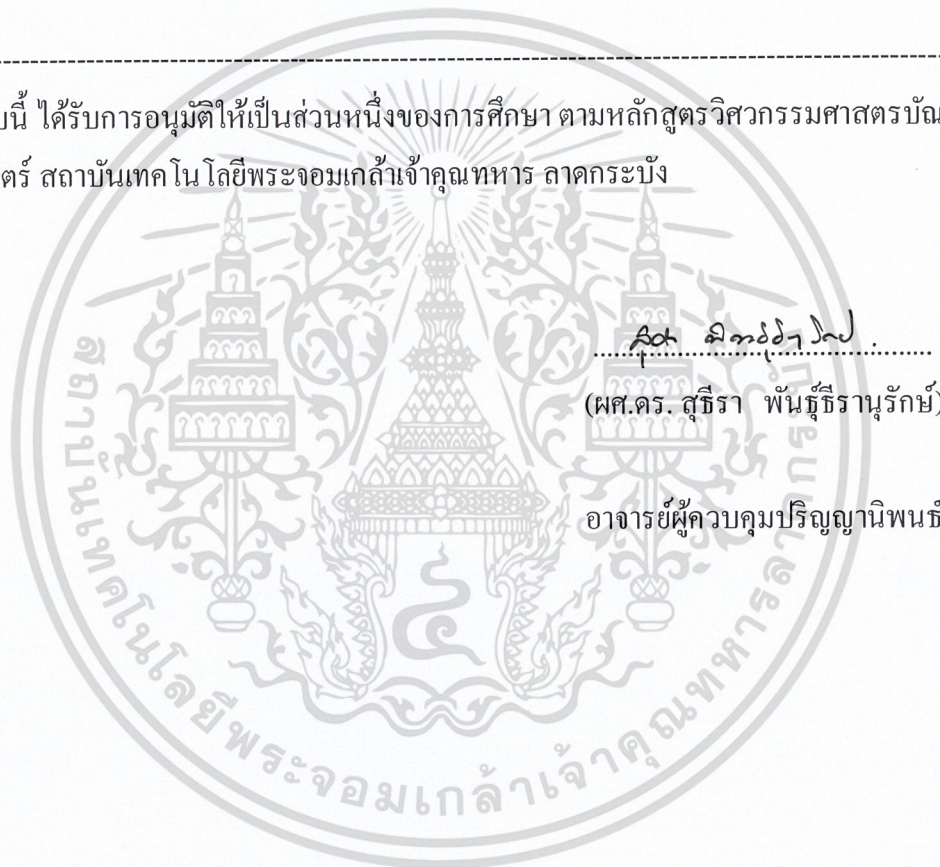
พ.ศ.

2553

อาจารย์ที่ปรึกษาปริญญานิพนธ์

ผศ.ดร. สุธีรา พันธุ์ธีรานุรักษ์

ปริญญานิพนธ์ฉบับนี้ ได้รับการอนุมัติให้เป็นส่วนหนึ่งของการศึกษา ตามหลักสูตรวิศวกรรมศาสตรบัณฑิต  
คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง



.....  
(ผศ.ดร. สุธีรา พันธุ์ธีรานุรักษ์)

อาจารย์ผู้ควบคุมปริญญานิพนธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	ระบบค้นหาเส้นทางอ้างอิงจากสภาพการจราจร (A Navigation System based on Traffic Conditions)	
รายชื่อนักศึกษา	นายพิชญ์ เกรียงไกรวศิน	รหัสนักศึกษา50011090
	นางสาวศิริพร ตันดินราโรจน์	รหัสนักศึกษา50011564
ปริญญา	วิศวกรรมศาสตรบัณฑิต	
สาขาวิชา	วิศวกรรมสารสนเทศ	
พ.ศ.	2553	
อาจารย์ที่ปรึกษาปริญญานิพนธ์	ผศ.ดร.สุธีรา พันธุ์ธีรานุรักษ์	

## บทคัดย่อ

โครงการระบบค้นหาเส้นทางอ้างอิงจากสภาพการจราจรนี้มีแนวคิดที่จะนำระบบการค้นหาเส้นทางและระบบการแสดงผลสภาพการจราจร นำมาประยุกต์ร่วมกันเพื่อช่วยสนับสนุนการตัดสินใจของผู้ขับขี่ในการเลือกเส้นทางที่ต้องการสัญจร ทำให้ผู้ขับขี่เดินทางถึงที่หมายรวดเร็วขึ้น โครงการนี้จัดทำขึ้นในรูปแบบของเว็บเซอร์วิสมีการจำลองการแสดงผลข้อมูลออกมาในรูปแบบของเว็บแอปพลิเคชัน โดยมีวัตถุประสงค์เพื่อต้องการสร้างเว็บเซอร์วิสที่สามารถแสดงผลลัพธ์ของการค้นหาเส้นทางตามชนิดของการค้นหาเส้นทางที่ผู้ใช้ต้องการ คือ ตามสภาพการจราจรและตามระยะทาง โดยผู้ใช้งานกำหนดจุดต้นทาง จุดปลายทางและชนิดการค้นหาเส้นทาง จากนั้นเว็บเซอร์วิสจะส่งผลลัพธ์ของการค้นหาเส้นทาง มาในรูปแบบของเอกซ์เอ็มแอล จึงได้ทำในส่วนของ การจำลองเว็บแอปพลิเคชันเพื่อทำการทดลองเรียกใช้เว็บเซอร์วิสที่ได้สร้างขึ้นและพบว่า ระบบค้นหาเส้นทางอ้างอิงจากสภาพการจราจรนี้สามารถนำไปช่วยสนับสนุนการตัดสินใจของผู้ขับขี่ในการเลือกเส้นทางที่ต้องการสัญจรได้

<b>Thesis Title</b>	A Navigation System based on Traffic Conditions	
<b>Student</b>	Mr. Pitch Kreangkriwasin	Student ID. 50011090
	Ms. Siriporn Tuntinararaot	Student ID.50011564
<b>Degree</b>	Bachelor of Engineering	
<b>Program</b>	Information Engineering	
<b>Year</b>	2010	
<b>Thesis Advisor</b>	Asst. Prof. Dr. Sutheera Puntheeranurak	

## ABSTRACT

At the present, there is a navigation system that can be used to help people to go to their destination easily. But the traffic in Thailand is not so good. Sometimes, the route to go to their destination is very short but they take a very long time on the road. Then the navigation system based on traffic conditions is developed to help them to choose a suitable route. The objective is to provide the web services which can support two kind of searching route, based on distance and based on traffic. The input of web service is composed of the initial place, destination place and type of routing. Web service will automatically send the results of searching path in XML format. To show that our web service can work as well, we create a web application to call our web service.

## กิตติกรรมประกาศ

โครงการนี้ได้ดำเนินการจนสำเร็จลุล่วงไปได้ด้วยดีเนื่องจากได้รับคำแนะนำและความช่วยเหลือในด้านต่างๆเป็นอย่างดีตลอดระยะเวลาในการทำงานซึ่งคณะผู้จัดทำต้องขอกราบขอบพระคุณผศ.ดร.สุธีราพันธุ์ธีรานุรักษ์อาจารย์ที่ปรึกษาโครงการที่คอยเอาใจใส่ให้คำแนะนำและช่วยตรวจสอบแก้ไขข้อบกพร่องต่างๆมาโดยตลอด

ขอขอบพระคุณคณาจารย์ภาควิชาวิศวกรรมสารสนเทศทุกท่านที่มีส่วนช่วยในการขจัดเกลาให้โครงการนี้สำเร็จลุล่วงไปได้อย่างสมบูรณ์

ขอขอบพระคุณ ดร.วสันต์ภักธรุธิคมหัวหน้าโครงการประเมินและรายงานสภาพอาจารย์ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ สำหรับฐานข้อมูลที่ใช้ในการทำโครงการนี้

ขอขอบคุณเพื่อนๆทุกคนที่คอยให้กำลังใจและช่วยเหลือซึ่งกันและกันมาโดยตลอดทำให้สามารถผ่านพ้นอุปสรรคต่างๆไปได้ด้วยดี

ท้ายที่สุดนี้คณะผู้จัดทำต้องขอกราบขอบพระคุณบิดามารดาและทุกคนในครอบครัวที่ให้การเลี้ยงดูอบรมสั่งสอนจนทำให้มีวันนี้รวมทั้งได้ให้การสนับสนุนและคอยให้กำลังใจเสมอมา และขอขอบคุณทุกๆท่านที่มีส่วนร่วมในความสำเร็จของโครงการชิ้นนี้ที่ไม่สามารถกล่าวไว้ ณ. ที่นี้ได้หมดคุณประโยชน์อันใดที่เกิดจากโครงการนี้เป็นผลมาจากความกรุณาของทุกท่านที่กล่าวมาข้างต้นคณะผู้จัดทำซาบซึ้งเป็นอย่างยิ่งจึงใคร่ขอขอบพระคุณไว้ ณ. ที่นี้ด้วย

นายพิชญ์ เกรียงไกรวสิน

นางสาวศิริพร ตันตินราโรจน์

# สารบัญ

	หน้า
บทคัดย่อภาษาไทย .....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VII
สารบัญรูป .....	VIII
บทที่ 1 บทนำ.....	1
1.1 แนวคิดและที่มาของปัญหา.....	1
1.2 จุดประสงค์ .....	1
1.3 ขอบเขตของโครงการ .....	2
1.4 ผลที่คาดว่าจะได้รับ .....	2
1.5 อุปกรณ์ที่ใช้ในการพัฒนาโครงการ .....	3
บทที่ 2 ทฤษฎีที่ใช้ในโครงการ .....	4
2.1 แผนภาพกระแสข้อมูล (Data Flow Diagrams: DFD) .....	4
2.1.1 สัญลักษณ์ที่ใช้ในแผนภาพกระแสข้อมูล .....	4
2.1.2 ระดับแผนภาพกระแสข้อมูลของระบบ.....	6
2.2 แบบจำลองฐานข้อมูลของไนแอม (Nijssen's Information Analysis Methodology: NIAM) .....	6
2.2.1 สัญลักษณ์ .....	6
2.2.2 ชนิดของความสัมพันธ์.....	7
2.3 ขั้นตอนการหาเส้นทางที่สั้นที่สุด (Shortest Path Algorithm) .....	9
2.3.1 เบลแมน ฟอร์ด อัลกอริทึม (Bellman Ford Algorithm).....	10
2.4 เว็บเซอร์วิส (Web Services) .....	11
2.4.1 หลักการพื้นฐานของเทคโนโลยีต่าง ๆ ที่พัฒนามาสู่เว็บเซอร์วิส .....	12
2.4.2 ลักษณะของเว็บเซอร์วิส .....	12
2.4.3 ภาษาที่ใช้ในเว็บเซอร์วิส .....	13

## สารบัญ (ต่อ)

	หน้า
2.4.4 ชนิดของเว็บเซอร์วิส.....	17
2.5 ภาษาจาวา .....	20
2.6 โปสเกรสเกรสคิวเอล (PostgreSQL).....	21
2.6.1 สถาปัตยกรรมการทำงานของโปสเกรสเอสคิวเอล.....	21
2.7 โปสจีเอส(PostGIS) .....	24
2.7.1 ชนิดข้อมูลของโปสจีเอสที่นำมาใช้งาน.....	24
2.7.2 ฟังก์ชันของโปสจีเอสที่นำมาใช้งาน .....	25
2.8 เจเอสพี (JSP) .....	25
2.8.1 ข้อดีของเจเอสพี .....	26
2.9 จาวาสคริป (JavaScript) .....	27
2.10 ทราฟฟีเอพีไอ (TraffyAPI).....	28
2.10.1 เอพีไอของทราฟฟีที่นำมาใช้งาน .....	29
บทที่ 3 การออกแบบและพัฒนาระบบ.....	30
3.1 แผนภาพกระแสข้อมูล (Data Flow Diagram: DFD) .....	30
3.2 แบบจำลองฐานข้อมูลในแอมของระบบค้นหาเส้นทางอ้างอิงจากสภาพจราจร .	31
3.3 ตารางฐานข้อมูล.....	32
3.4 การออกแบบเว็บเซอร์วิส.....	34
3.5 การออกแบบวิธีการค้นหาเส้นทางอ้างอิงสภาพจราจร.....	36
บทที่ 4 ผลการดำเนินงาน.....	43
4.1 ขั้นตอนวิธีการสร้างระบบ .....	43
4.2 ขั้นตอนการเรียกใช้งานเราต์ติ้งเอพีไอ (Routing API) .....	47
4.3 ขั้นตอนการทดสอบการค้นหาเส้นทาง .....	49
4.3.1 การค้นหาเส้นทางตามระยะทาง.....	50
4.3.2 การค้นหาเส้นทางตามสภาพจราจร .....	53
บทที่ 5 บทวิจารณ์และสรุป .....	55
5.1 ผลที่ได้รับ.....	55
5.2 ปัญหาที่พบ .....	55
5.3 แนวทางการแก้ปัญหา .....	55

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา V และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
5.2 แนวทางการพัฒนาต่อ.....	55
ภาคผนวก .....	56
ภาคผนวก ก. คู่มือการติดตั้งและตั้งค่าโปรแกรมโพสเกรสเกรสคิวแอล .....	57
ภาคผนวก ข. การติดตั้งโปรแกรมโพสจีส .....	62
ภาคผนวก ค. คู่มือการติดตั้งโปรแกรมทอมแคท 6.0.....	66
บรรณานุกรม .....	71



# สารบัญตาราง

ตารางที่	หน้า
2.1 ตารางแสดงการวิเคราะห์ความแตกต่างของอัลกอริทึม .....	9
2.2 ตารางแสดงฟังก์ชันของ โปสิทีฟที่นำมาใช้งาน .....	25
2.3 ตารางแสดงฟังก์ชันของกราฟฟี่ที่นำมาใช้งาน .....	29
3.1 ตารางแสดงรายละเอียดถนน .....	32
3.2 ตารางแสดงรายละเอียดของช่วงถนน .....	32
3.3 ตารางแสดงละเอียดของพิกัด .....	33
3.4 ตารางแสดงข้อมูลระหว่างพิกัดกับช่วงถนน .....	33
3.5 ตารางแสดงละเอียดของลิงค์ .....	33
3.6 ตารางแสดงละเอียดของผลลัพธ์เส้นทาง .....	33



# สารบัญรูป

รูปที่	หน้า
2.1 สัญลักษณ์ที่ใช้ในแผนภาพกระแสข้อมูลแบบจินีแอนด์ชาร์ตัน .....	5
2.2 แสดงสัญลักษณ์ต่างๆ ที่ใช้ในไบนารี .....	7
2.3 แสดงความสัมพันธ์แบบหนึ่งต่อหนึ่งหน่วย .....	7
2.4 แสดงความสัมพันธ์แบบหนึ่งหน่วยต่อหลายหน่วย .....	8
2.5 แสดงความสัมพันธ์แบบหลายหน่วยต่อหลายหน่วย.....	8
2.6 แสดงความสัมพันธ์แบบคอนสแตนต์.....	8
2.7 แสดงวิธีการเข้าถึงข้อมูลโดยวิธีดีไอเอ็ม .....	15
2.8 แสดงวิธีการเข้าถึงข้อมูลโดยวิธีเอสเออีก.....	16
2.9 แสดงการทำงานของโซป เว็บเซอร์วิส .....	17
2.10 แสดงการทำงานของเรส เว็บเซอร์วิส.....	19
2.11 แสดงไคลเอนต์โพรเซสเซอร์บริการโพสมาสเตอร์.....	22
2.12 แสดงโพสมาสเตอร์สร้างโพสเกรส โพรเซส .....	23
2.13 แสดง โพสเกรส โพรเซสให้บริการแก่ ไคลเอนต์โพรเซส.....	23
2.14 แสดงการขอบริการเซิร์ฟเวอร์ผ่านอินเตอร์เฟซเจดีบีซี, โอดีบีซี.....	23
2.15 แสดงการย้ายไฟล์เจเอสพีจากระบบยูนิกซ์ไประบบวิน โดว์.....	26
2.16 แสดงหน้าหลักของเว็บ ไซตฺ์ทราฟฟี่เอพีไอ .....	28
3.1 แผนภาพกระแสข้อมูลแสดงภาพรวมของระบบ.....	30
3.2 แผนภาพกระแสข้อมูลระดับที่ 1 ของระบบ.....	31
3.3 ผังไบนารีแสดงความสัมพันธ์ของข้อมูลระบบค้นหาเส้นทางอ้างอิงจากสภาพการจราจร ...	31
3.4 บล็อกไดอะแกรมการทำงานของระบบค้นหาเส้นทาง.....	34
3.5 บล็อกไดอะแกรมการทำงานของเรตติ้งเซิร์ฟวิส.....	35
3.6 บล็อกไดอะแกรมการทำงานของทราฟฟี่คเซิร์ฟวิส.....	36
3.7 โฟว์ชาร์ทแสดงภาพรวมการทำงานของเรตติ้งเว็บเซอร์วิส.....	37
3.8 โฟว์ชาร์ทแสดงการทำงานของโมดูลเบลแมน ฟอร์ด อัลกอริทึม .....	38
3.9 ส่วนของพารามิเตอร์ในการเรียกใช้ ทราฟฟี่เอพีไอ getTrafficCongestion .....	39

## สารบัญรูป (ต่อ)

รูปที่	หน้า
3.10 ส่วนของพารามิเตอร์ที่ส่งออกจากการเรียกใช้ทราฟฟิเคอพีไอ getTrafficCongestion.....	40
3.11 แสดงส่วนของการคำนวณหาเส้นทางในส่วนการตั้งค่าเริ่มต้นให้การค้นหาเส้นทาง .....	41
3.12 แสดงส่วนของการคำนวณหาเส้นทางในส่วนการค้นหาเส้นทางแบบหลายเส้นทาง .....	42
4.1 แสดงส่วนต่างๆของโปรแกรมค้นหาเส้นทางอ้างอิงสภาพจราจร.....	43
4.2 แสดงส่วนของจาวาโปรแกรมทั่วไป.....	44
4.3 แสดงส่วนของจาวาโปรแกรมเอนเตอร์ไพรส์แอปพลิเคชัน .....	45
4.4 แสดงส่วนของจาวาโปรแกรมการจัดการข้อมูล.....	45
4.5 แสดงส่วนของจาวาโปรแกรมเว็บเซอร์วิส .....	46
4.6 แสดงส่วนของจาวาโปรแกรมเซิร์ฟเวอร์.....	46
4.7 แสดงรายละเอียดของเอพีไอ .....	47
4.8 หน้าแรกของเว็บไซต์ .....	49
4.9 แสดงการค้นหาเส้นทางตามสภาพจราจร .....	50
4.10 ผู้ใช้เลือกสถานที่ วัดอภัยทวารามและรางน้ำ.....	51
4.11 ตัวอย่างการเลือกจำนวนเส้นทางที่ต้องการค้นหา.....	51
4.12 เมื่อเลือกเส้นทางที่ต้องการค้นหาจำนวน 5 เส้นทาง.....	52
4.13 การเลือกให้ระบบแสดงผลเส้นทางที่ 2 .....	52
4.14 เลือกเป็นค้นหาเส้นทางตามสภาพจราจร .....	53
4.15 แสดงผลการค้นหาเส้นทาง .....	53
4.16 เปรียบเทียบการค้นหาเส้นทาง .....	54

## สารบัญรูป (ต่อ)

รูปที่	หน้า
ก.1 การติดตั้งโปรแกรมโพสเกรสเกรสคิวแอลขั้นตอนที่ 1 .....	58
ก.2 การติดตั้งโปรแกรมโพสเกรสเกรสคิวแอลขั้นตอนที่ 2 .....	58
ก.3 การติดตั้งโปรแกรมโพสเกรสเกรสคิวแอลขั้นตอนที่ 3 .....	59
ก.4 การติดตั้งโปรแกรมโพสเกรสเกรสคิวแอลขั้นตอนที่ 4 .....	59
ก.5 การติดตั้งโปรแกรมโพสเกรสเกรสคิวแอลขั้นตอนที่ 5 .....	60
ก.6 การติดตั้งโปรแกรมโพสเกรสเกรสคิวแอลขั้นตอนที่ 6 .....	60
ก.7 การติดตั้งโปรแกรมโพสเกรสเกรสคิวแอลขั้นตอนที่ 7 .....	61
ก.8 การติดตั้งโปรแกรมโพสเกรสเกรสคิวแอลขั้นตอนที่ 8 .....	61
ข.1 การติดตั้งโปรแกรมโพสจีสขั้นตอนที่ 1 .....	63
ข.2 การติดตั้งโปรแกรมโพสจีสขั้นตอนที่ 2 .....	63
ข.3 การติดตั้งโปรแกรมโพสจีสขั้นตอนที่ 3 .....	64
ข.4 การติดตั้งโปรแกรมโพสจีสขั้นตอนที่ 4 .....	64
ข.5 การติดตั้งโปรแกรมโพสจีสขั้นตอนที่ 6 .....	65
ค.1 การติดตั้งโปรแกรมทอมแคท 6.0 ขั้นตอนที่ 1 .....	67
ค.2 การติดตั้งโปรแกรมทอมแคท 6.0 ขั้นตอนที่ 2 .....	67
ค.3 การติดตั้งโปรแกรมทอมแคท 6.0 ขั้นตอนที่ 3 .....	68
ค.4 การติดตั้งโปรแกรมทอมแคท 6.0 ขั้นตอนที่ 4 .....	68
ค.5 การติดตั้งโปรแกรมทอมแคท 6.0 ขั้นตอนที่ 5 .....	69
ค.6 การติดตั้งโปรแกรมทอมแคท 6.0 ขั้นตอนที่ 6 .....	69
ค.7 การติดตั้งโปรแกรมทอมแคท 6.0 ขั้นตอนที่ 7 .....	70

# บทที่ 1

## บทนำ

### 1.1 แนวคิดและที่มาของปัญหา

จากปัญหาการจราจรในเขตกรุงเทพมหานคร ที่อัตราการเพิ่มขึ้นของปริมาณรถยนต์ส่วนบุคคลมีแนวโน้มเพิ่มขึ้น ผู้ขับขี้อาจมีความชำนาญในเส้นทางที่จำกัดเฉพาะที่ตนใช้อยู่เป็นประจำ จำนวนช่องทางการจราจรและทิศทางการจราจรมีการเปลี่ยนแปลงอยู่บ่อยครั้ง การที่ไม่สามารถทราบเส้นทางที่กำลังเดินทางอยู่นั้นมีสภาพการจราจรเป็นอย่างไร อาจทำให้เกิดการตัดสินใจที่ผิดพลาด ทำให้ไปถึงจุดหมายล่าช้าได้ นอกจากนี้เทคโนโลยีจีพีเอส (GPS) ปัจจุบัน สามารถให้บริการการค้นหาเส้นทางจราจรโดยหลีกเลี่ยงเส้นทางที่เป็นปัญหาทำให้การไปถึงจุดหมายล่าช้า เช่น รถติด อุบัติเหตุ และอื่น ๆ ได้ แต่ก็ยังมีข้อจำกัดในการใช้งาน เนื่องจากผู้ใช้ต้องทำการซื้ออุปกรณ์ซึ่งมีราคาค่อนข้างสูง และในประเทศไทยยังไม่สามารถใช้งานเทคโนโลยีนี้ได้

ด้วยเหตุผลดังกล่าว ผู้จัดทำโครงการจึงได้เกิดแนวคิดในการพัฒนาระบบการค้นหาเส้นทางภายในกรุงเทพมหานครเพื่ออำนวยความสะดวกในการค้นหาเส้นทางที่มีสภาพการจราจรที่ดีที่สุดในช่วงเวลานั้น ๆ ทำให้ไปถึงจุดหมายโดยใช้เวลาน้อยที่สุด โดยผู้ใช้สามารถเข้าถึงโปรแกรมผ่านทางระบบอินเทอร์เน็ต ทำให้บุคคลทั่วไปสามารถตรวจสอบเส้นทางผ่านทางอุปกรณ์ทุกชนิดที่สามารถใช้อินเทอร์เน็ตได้ และสามารถวางแผนการเดินทางได้โดยใช้รถส่วนตัวหรือใช้รถโดยสาร (Taxi) ได้

### 1.2 จุดประสงค์

- 1.2.1 เพื่อสร้างระบบที่อำนวยความสะดวกในการค้นหาเส้นทางที่ใช้เวลาน้อยที่สุดในการเดินทางโดยอิงกับสภาพการจราจรขณะนั้นหรือค้นหาเส้นทางที่มีระยะทางสั้นที่สุด
- 1.2.2 ศึกษาการออกแบบระบบ และการนำอัลกอริทึมมาประยุกต์ในการค้นหาเส้นทางที่ใช้เวลาน้อยที่สุดในการเดินทางโดยอิงกับสภาพการจราจรขณะนั้น
- 1.2.3 ศึกษาการออกแบบ และการใช้งานระบบฐานข้อมูล
- 1.2.4 เพื่อศึกษาเกี่ยวกับการทำงานของเว็บเซอร์วิส (Web Service)

### 1.3 ขอบเขตของโครงการ

เป็นระบบที่ช่วยในการตัดสินใจเลือกเส้นทางการจราจรภายในกรุงเทพมหานคร โดยหลีกเลี่ยงเส้นทางที่อาจเป็นปัญหาทำให้การไปถึงจุดหมายล่าช้า เช่น รถติด อุบัติเหตุ หรือเหตุการณ์อื่น ๆ ที่ทำให้การจราจรติดขัด การใช้งานในระบบนี้จะอยู่ในรูปแบบของเราที่ดึงเว็บเซอร์วิส (Routing Web service)

ระบบทั้งหมดจะแบ่งเป็นสองส่วนคือเว็บแอปพลิเคชันซึ่งเป็นส่วนแสดงผลที่ติดต่อกับผู้ใช้ โดยรับค่าของจุดหมายต้นทางกับปลายทางและรูปแบบการค้นหาจากผู้ใช้ และเราที่ดึงเว็บเซอร์วิส ซึ่งเป็นระบบภายในที่ใช้ในการค้นหาเส้นทางหลังจากรับค่าจุดหมายต้นทางกับปลายทางจากส่วนเว็บแอปพลิเคชันมา โดยโปรแกรมที่เขียนขึ้นเพื่อใช้ในการค้นหาเส้นทางนั้น จะใช้ภาษาจาวาในการพัฒนา ทั้งนี้โปรแกรมที่เขียนขึ้นจะมีความสามารถในการทำงานดังต่อไปนี้

- 1.3.1 ผู้ใช้กำหนดเส้นทางจากจุดต้นทางถึงจุดปลายทางได้
- 1.3.2 ผู้ใช้สามารถเลือกรูปแบบการเดินทางตามระยะทาง
- 1.3.3 ผู้ใช้สามารถเลือกรูปแบบการเดินทางตามสภาพการจราจร
- 1.3.4 ระบบจะแสดงเส้นทางที่เป็นไปได้ตามรูปแบบของการเดินทางที่ผู้ใช้ได้เลือกไว้
- 1.3.5 ผู้ใช้สามารถเลือกเส้นทางการเดินทางตามที่ผู้ใช้ต้องการจากผลลัพธ์ที่ระบบทำการค้นหาให้

### 1.4 ผลที่คาดว่าจะได้รับ

- 1.4.1 ได้รับความรู้ ความสามารถในการเขียนโปรแกรมภาษาจาวา
- 1.4.2 ได้รับความรู้ในการออกแบบระบบฐานข้อมูล
- 1.4.3 ได้รับความรู้เกี่ยวกับเว็บเซอร์วิส
- 1.4.4 ได้รับความรู้ในการนำอัลกอริทึมมาประยุกต์ใช้ให้ตรงกับความต้องการในการออกแบบระบบ
- 1.4.5 เป็นประโยชน์ต่อผู้ใช้งานระบบที่ต้องการค้นหาเส้นทางในการเดินทางที่ใช้เวลาเดินทางน้อยที่สุด หลีกเลี่ยงสภาพจราจรที่ติดขัดหรืออุบัติเหตุ ในช่วงเวลานั้นๆ หรือการเดินทางในเส้นทางที่ไม่คุ้นเคย

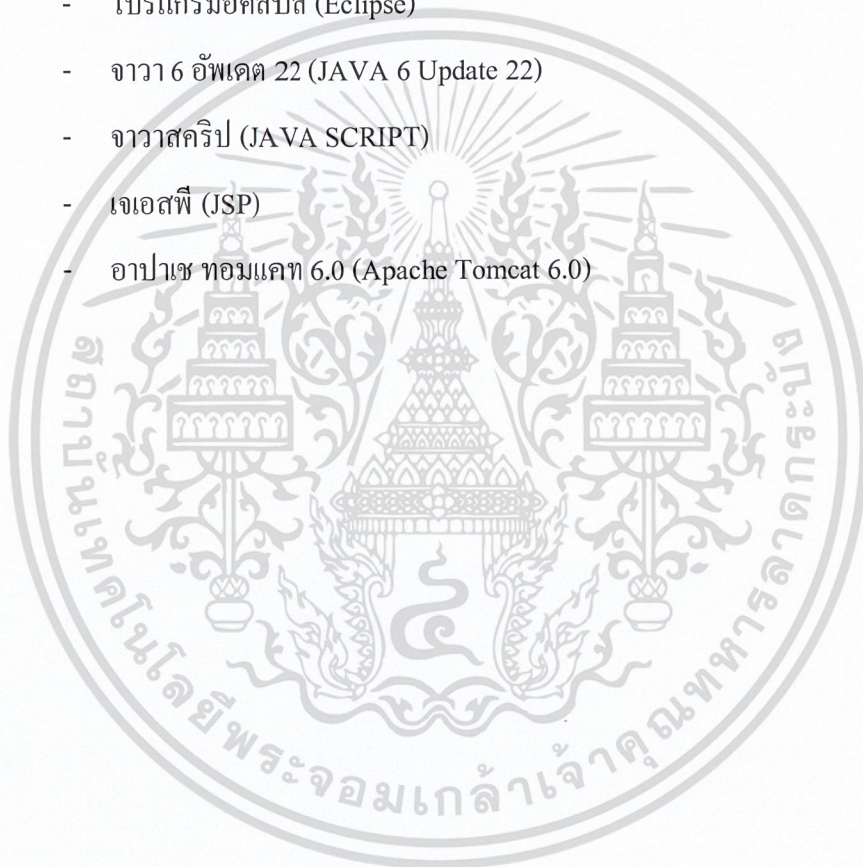
## 1.5 อุปกรณ์ที่ใช้ในการพัฒนาโครงการ

### 1.5.1 ฮาร์ดแวร์

- เครื่องคอมพิวเตอร์สำหรับพัฒนาโปรแกรมที่มีการต่อเชื่อมกับเน็ตเวิร์คและเป็นเซิร์ฟเวอร์ระบบฐานข้อมูล จำนวน 1 เครื่อง

### 1.5.2 ซอฟต์แวร์

- ระบบจัดการฐานข้อมูลโพสเกรสเควสคิวแอล 8.4 (PostgreSQL 8.4)
- โปสจีเอส 1.5.1 (PostGIS 1.5.1)
- โปรแกรมอีคลิปส์ (Eclipse)
- จาวา 6 อัปเดต 22 (JAVA 6 Update 22)
- จาวาสคริป (JAVA SCRIPT)
- เจเอสพี (JSP)
- อาปาเช ทอมแคท 6.0 (Apache Tomcat 6.0)



## บทที่ 2

# ทฤษฎีพื้นฐานที่ใช้

### 2.1 แผนภาพกระแสข้อมูล (Data Flow Diagrams : DFD)

แผนภาพกระแสข้อมูล (Data Flow Diagram : DFD) เป็นแบบจำลองกระบวนการ ที่นำมาใช้ งานกับการวิเคราะห์และออกแบบระบบเชิงโครงสร้าง เพื่อเป็นเครื่องมือแสดงถึงขั้นตอนการทำงานของระบบ ทิศทางการส่งผ่านข้อมูลและแสดงความสัมพันธ์ระหว่างกระบวนการกับข้อมูลที่เกี่ยวข้อง

#### 2.1.1 สัญลักษณ์ที่ใช้ในแผนภาพกระแสข้อมูล

สัญลักษณ์สร้างแผนภาพกระแสข้อมูล (DFD Symbols: DFDs) ที่นิยมใช้เป็นของจีนีแอนด์ซาร์สัน (Gene and Sarson) ประกอบด้วยสัญลักษณ์ 4 ตัวดังแสดงในรูปที่ 2.1

##### 1) สัญลักษณ์การประมวลผล หรือ โพรเซส (Process)

จะแทนด้วยภาพสี่เหลี่ยมกรอบมนหรือภาพวงกลมระบุชื่อของกระบวนการอยู่ในรูปของคำกริยาตามด้วยคำนามเอกพจน์ เช่น คำนวณ, แก้ไข, พิมพ์ เป็นต้น ซึ่งแสดงถึงวิธีการรับข้อมูลและทำให้เกิดผลลัพธ์การประมวลผลจะเปลี่ยนข้อมูลขาเข้าเป็นผลลัพธ์ นั้นหมายความว่า จะต้องมีการกระทำบางอย่างต่อข้อมูลทำให้เกิดผลลัพธ์ขึ้นมาโดยปกติแล้วข้อมูลที่นำเข้าสู่โพรเซสจะแตกต่างจากข้อมูลเมื่อออกจากโพรเซส

โพรเซสเป็นตัวอย่างหนึ่งของ “กล่องดำ” หมายถึงว่าเราทราบว่ามีข้อมูลเป็นอะไร ผลลัพธ์อะไรที่เราต้องการและหน้าที่โดยทั่ว ๆ ไปของโพรเซสแต่จะไม่ทราบว่าโพรเซสนั้นทำงานอย่างไร หลักการของกล่องดำมีประโยชน์ในการเขียนแผนภาพแสดงการเปลี่ยนแปลงของข้อมูลโดยที่ยังไม่ต้องทราบในรายละเอียดว่าโพรเซสนั้นมีรายละเอียดอะไรบ้างซึ่งสามารถหา รายละเอียดเหล่านั้นได้ในภายหลัง

##### 2) สัญลักษณ์กระแสข้อมูล หรือดาต้าโฟว์ (Data Flow)

แทนสัญลักษณ์ด้วยเส้นที่มีลูกศรทิศทางเดียวหรือสองทิศทางระบุชื่อของข้อมูลด้วยคำนามเอกพจน์และคำคุณศัพท์ที่อยู่ด้านบนด้านล่าง หรือทั้งสองด้านของเส้น แสดงถึงทิศทางการส่งผ่านข้อมูลจากแหล่งหนึ่งไปยังอีกแหล่งหนึ่งเท่านั้น

ข้อมูลจะไหลระหว่างโพรเซสต่าง ๆ และอาจจะเคลื่อนที่มาจากสิ่งที่ยอยู่นอกระบบก็ได้ ข้อมูลที่เคลื่อนที่อาจจะเพียงข้อมูลเดียว ๆ เช่น เลขที่สินค้าหรือกลุ่มของข้อมูล เช่น ข้อมูลพนักงาน ข้อมูลลูกค้า เป็นต้น กลุ่มของข้อมูลควรจะเป็นเรื่องเดียวกัน หรือสัมพันธ์กัน

ข้อมูลแต่ละอันหรือกลุ่มข้อมูล ควรจะมีชื่อของตัวเองที่ไม่เหมือนกัน ควรหลีกเลี่ยงใช้ชื่อที่กว้างเกินไป เช่น “ข้อผิดพลาด” เพราะว่ามีในระบบหนึ่งๆ อาจจะมี “ข้อผิดพลาด” เกิดขึ้นหลาย ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แห่ง ควรใช้ชื่อเฉพาะเจาะจงมากกว่านี้ เช่น “เลขที่ลูกค้าไม่ถูกต้อง” หรือ “ไม่มีสินค้าในคลัง” เป็นต้น ในระบบใหญ่ ๆ ต้องแยกรายละเอียดเหล่านี้ออกให้ชัดเจน

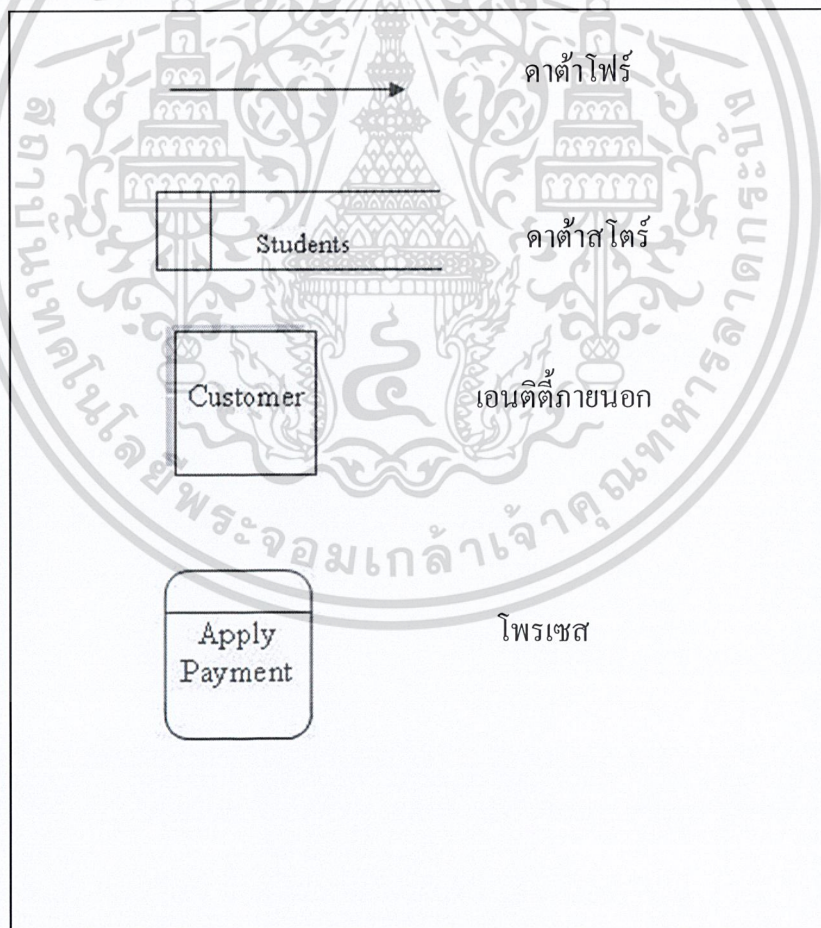
### 3) สัญลักษณ์แหล่งเก็บข้อมูล หรือดาต้าสโตร์ (Data Store)

แทนด้วยรูปสี่เหลี่ยมผืนผ้าปลายเปิดทั้งสองด้านหรือเปิดที่ปลายด้านขวาโดยระบุชื่อของเพิ่มข้อมูลที่ใส่เก็บในรูปคำนามพหูพจน์เพื่อแสดงถึงแหล่งที่เก็บข้อมูล

ข้อมูลจะถูกเก็บในไฟล์และถูกเรียกใช้เมื่อต้องการ ถ้าหัวลูกศรวิ่งเข้าสู่ไฟล์แสดงว่า มีการเขียนข้อมูลหรือการแก้ไขข้อมูลในไฟล์ ถ้าลูกศรวิ่งออกจากไฟล์แสดงว่ามีการอ่านข้อมูล

### 4) สัญลักษณ์สิ่งที่ยอยู่นอกระบบ หรือเอนติตี้ภายนอก (External Entity)

แทนด้วยเครื่องหมายสี่เหลี่ยมมีเงาแสดงให้เห็นเป็นภาพสามมิติโดยจะระบุชื่อสิ่งที่เกี่ยวข้องไว้ในช่องสี่เหลี่ยม โดยเอนติตี้ภายนอกอาจเป็น คน หน่วยงาน องค์กรภายนอกหรือระบบสารสนเทศที่เกี่ยวข้องในการส่งข้อมูลเข้าหรือรับข้อมูลจากระบบ เอนติตี้ภายนอกเรียกอีกอย่างว่าเทอร์มินเตอร์ (Terminator)



รูปที่ 2.1 สัญลักษณ์ที่ใช้ในแผนภาพกระแสข้อมูลแบบจินีแอนด์ชาร์สัน

## 2.1.2 ระดับแผนภาพกระแสข้อมูลของระบบ

แผนภาพกระแสข้อมูลของระบบสามารถแบ่งออกเป็นระดับได้ดังนี้

- **แผนภาพบริบท (Context Diagram)**

แผนภาพบริบท เป็นแผนภาพกระแสข้อมูล ระดับสูงสุดของระบบ ซึ่งในระดับนี้จะบอกว่าระบบที่น่าสนใจมีอินพุตเป็นอะไร ผลลัพธ์ที่ได้จากระบบคืออะไร และข้อมูลที่เกี่ยวข้องกับระบบภายนอก โดยจะยังไม่แสดงแหล่งเก็บข้อมูลออกมา ซึ่งแหล่งเก็บข้อมูลจะถูกแสดงในระดับถัดไป

- **แผนภาพกระแสข้อมูลระดับรอง (Lower Level Data Flow Diagram)**

แผนภาพกระแสข้อมูล จะมีการเขียนแตกออกเป็นระดับ โดยเริ่มจาก ระดับ 0 ซึ่งเป็นการเขียนภาพรวมทั้งหมดของระบบ จากนั้นจะแตกรายละเอียดลงไปในแต่ละส่วน มาเป็นแผนภาพกระแสข้อมูลระดับรอง ซึ่งจำเป็นต้องกำหนดระดับชั้นและความสมดุลอย่างถูกต้องชัดเจน ซึ่งระดับชั้นหมายถึง หมายเลขที่ระบุไว้ของโพเรซตามขั้นตอนของการดำเนินงาน เรียงตามลำดับของชุดกระบวนการนั้น ส่วนความสมดุล หมายถึง ความถูกต้องตรงกันทั้งหมดของลำดับในแผนผัง รวมถึงกระแสข้อมูลทั้งนำเข้าและส่งออก

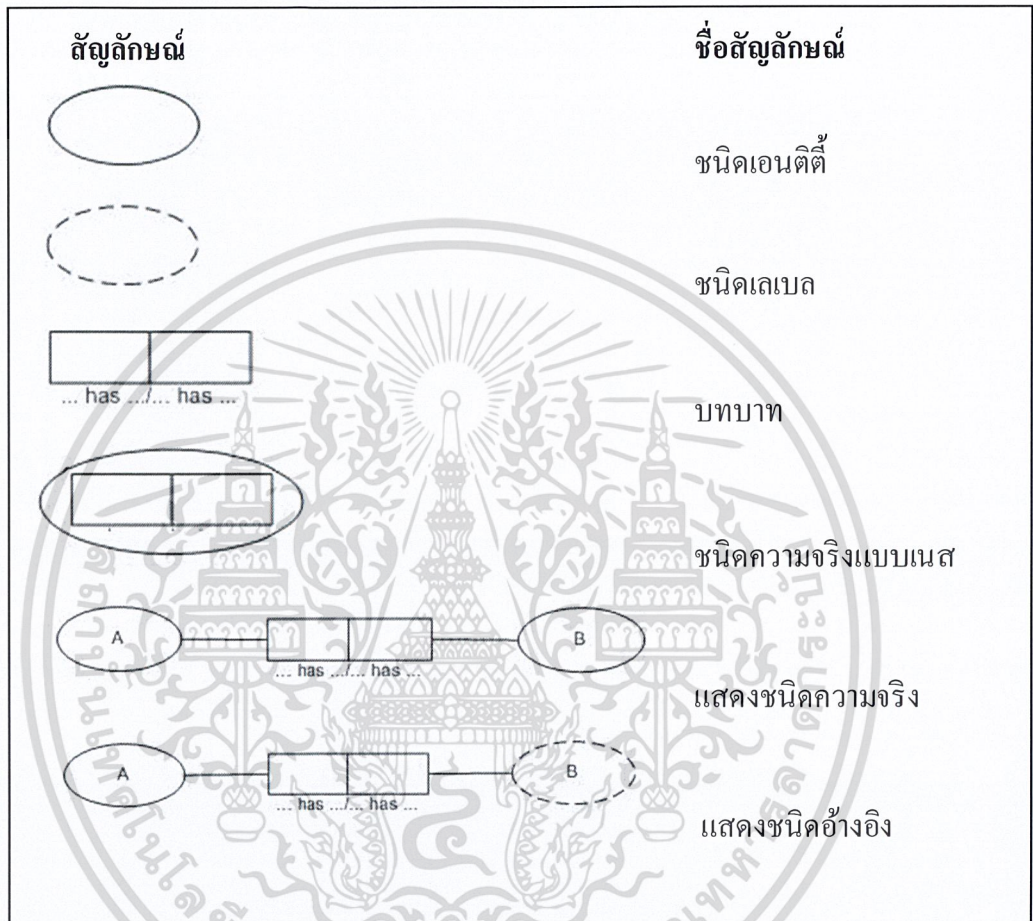
## 2.2 แบบจำลองฐานข้อมูลของไนแอม (Nijssen's Information Analysis Methodology: NIAM)

### 2.2.1 สัญลักษณ์

ไนแอม (NIAM) เป็นวิธีการออกแบบฐานข้อมูล โดยการแสดงความหมาย ความสัมพันธ์ และข้อจำกัดต่างๆ ของข้อมูลด้วยแบบจำลองข้อมูล ที่ประกอบไปด้วยสัญลักษณ์ต่างๆ ดังแสดงได้ในรูปที่ 2.2 โดยมีรายละเอียดดังนี้

- ชนิดเอนติตี้ (Entity Type) หมายถึง เซตของสิ่งที่น่าสนใจ ทั้งที่อยู่ในรูปของนามธรรมหรือรูปธรรม ซึ่งอาจเป็นสิ่งที่จับต้องได้หรือจับต้องไม่ได้ โดยใช้ร่วมกับสัญลักษณ์อื่น
- ชนิดเลเบล (Label Type ,Value Type) หมายถึง เซตของสิ่งที่ใช้บ่งบอกความแตกต่างหรือชื่อของแต่ละเอนติตี้ (Entity) ที่กำหนด
- บทบาท (Role) หมายถึง ความสัมพันธ์ที่เกี่ยวข้องกับเอนติตี้ที่เชื่อมต่ออยู่
- ชนิดความจริงแบบเนสต์ (Nested Fact Type) หมายถึง ชนิดเอนติตี้ที่แสดงความสัมพันธ์ในการกำหนดกลุ่มของความจริงที่มีตั้งแต่ 2 บทบาทขึ้นไป

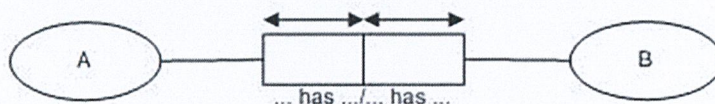
- ชนิดความจริง (Fact Type) หมายถึง เซตของความสัมพันธ์ระหว่างสมาชิกของชนิดเอนทิตีตั้งแต่ 2 เอนทิตีขึ้นไป
- ชนิดอ้างอิง (Reference Type) หมายถึง เซตของความสัมพันธ์ระหว่างสมาชิกของเอนทิตีกับสมาชิกของชนิดของเลเบลที่มีอยู่



รูปที่ 2.2 แสดงสัญลักษณ์ต่างๆ ที่ใช้ในโอเอม

### 2.2.2 ชนิดของความสัมพันธ์

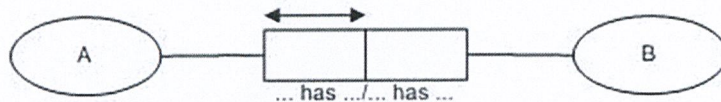
- ความสัมพันธ์อ้างอิงแบบหนึ่งต่อหนึ่งหน่วย (One To One)



รูปที่ 2.3 แสดงความสัมพันธ์แบบหนึ่งต่อหนึ่งหน่วย

สมาชิกของ A จะมีความสัมพันธ์เกี่ยวข้องกับสมาชิกของ B ได้เพียง 1 ค่าเท่านั้น และสมาชิกของ B จะมีความสัมพันธ์เกี่ยวข้องกับสมาชิก A ได้เพียง 1 ค่าเท่านั้นเช่นกัน

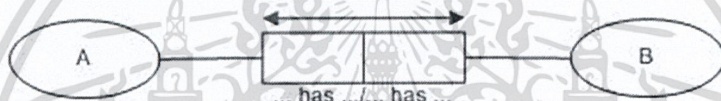
- ความสัมพันธ์แบบหนึ่งหน่วยต่อหลายหน่วย (One To Many)



รูปที่ 2.4 แสดงความสัมพันธ์แบบหนึ่งหน่วยต่อหลายหน่วย

สมาชิกของ A จะมีความสัมพันธ์เกี่ยวข้องกับสมาชิกของ B ได้เพียง 1 ค่าเท่านั้น แต่สมาชิกของ B จะมีความสัมพันธ์เกี่ยวข้องกับสมาชิกของ A ได้มากกว่า 1 ค่า

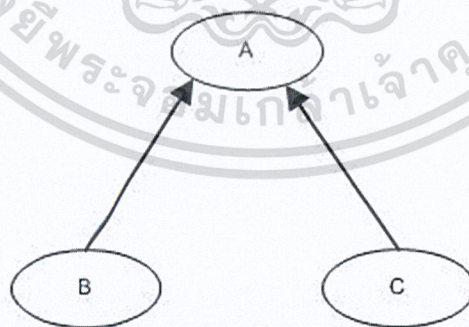
- ความสัมพันธ์แบบหลายหน่วยต่อหลายหน่วย (Many To Many)



รูปที่ 2.5 แสดงความสัมพันธ์แบบหลายหน่วยต่อหลายหน่วย

สมาชิกของ A จะมีความสัมพันธ์เกี่ยวข้องกับสมาชิก B ได้มากกว่า 1 ค่า และสมาชิกของ B จะมีความสัมพันธ์เกี่ยวข้องกับสมาชิก A ได้มากกว่า 1 ค่าเช่นกัน

- ความสัมพันธ์แบบซับไทป์คอนสเตรนต์ (Subtype Constraints)



รูปที่ 2.6 แสดงความสัมพันธ์แบบคอนสเตรนต์

ข้อมูลสมาชิกของ A จะถูกเรียกว่าซูเปอร์ไทป์ (Super type) โดยข้อมูลสมาชิก A นั้นสามารถแบ่งออกได้เป็น 2 กลุ่ม คือ B และ C ซึ่งจะถูกรเรียกว่า ซับไทป์ (Subtype) เช่น เอนติตี้สามารถแบ่งออกเป็นซับไทป์ อาจารย์และนักศึกษาได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการเชิงงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

## 2.3 ขั้นตอนการหาเส้นทางที่สั้นที่สุด (Shortest Path Algorithm)

ในการหาเส้นทางที่สั้นที่สุดมีอัลกอริทึมหลายแบบที่แตกต่างกันที่ใช้หาเส้นทางที่สั้นที่สุดระหว่าง 2 จุดยอดในกราฟมีน้ำหนัก จากการศึกษาทั้ง 3 อัลกอริทึมทำให้เราสามารถทำการเปรียบเทียบข้อดีและข้อเสียของแต่ละอัลกอริทึม เพื่อที่จะสามารถเลือกอัลกอริทึมที่เหมาะสมกับระบบงานของเราที่ทำการพัฒนาขึ้น โดยทำการแสดงดังในตารางที่ 2.1

ตารางที่ 2.1 แสดงการวิเคราะห์ความแตกต่างของอัลกอริทึม

Bellman Ford Algorithm	Dijkstra's algorithm	Floyd's algorithm
1. ใช้หน่วยความจำน้อยกว่า เพราะสถานะในเส้นทางค้นหา ปัจจุบันเท่านั้นที่ถูกเก็บ(ในขณะที่ใดๆ จะเก็บเส้นทางเดียว พอจะไปเส้นทางอื่นเส้นทางที่ผ่านมาก็ไม่จำเป็นต้องเก็บ)	1. ใช้หน่วยความจำมาก เพราะต้องเก็บสถานะไว้ทุกตัวเพื่อหาเส้นทางจากสถานะเริ่มต้นไปหาคำตอบ	1. ใช้หน่วยความจำน้อยกว่า เพราะสถานะในเส้นทางค้นหา ปัจจุบันเท่านั้นที่ถูกเก็บ (ในขณะที่ใดๆ จะเก็บเส้นทางเดียว พอจะไปเส้นทางอื่นเส้นทางที่ผ่านมาก็ไม่จำเป็นต้องเก็บ)
2. . จะไม่ติดเส้นทางที่ลึกลับมาก ๆ โดยไม่พบคำตอบ	2. จะไม่ติดเส้นทางที่ลึกลับมาก ๆ โดยไม่พบคำตอบ	2. จะไม่ติดเส้นทางที่ลึกลับมาก ๆ โดยไม่พบคำตอบ
3. ถ้าคำตอบอยู่ในระดับ n+1 สถานะทุกตัวที่ระดับ 1 ถึงระดับ n จะต้องถูกกระจายจนหมด ทำให้มีสถานะที่ไม่จำเป็นในเส้นทางที่จะไปสู่คำตอบถูกกระจายออกด้วย	3. ถ้าคำตอบอยู่ในระดับ n+1 สถานะทุกตัวที่ระดับ 1 ถึงระดับ n จะต้องถูกกระจายจนหมด ทำให้มีสถานะที่ไม่จำเป็นในเส้นทางที่จะไปสู่คำตอบถูกกระจายออกด้วย	3. ถ้าคำตอบอยู่ในระดับ n+1 สถานะทุกตัวที่ระดับ 1 ถึงระดับ n จะต้องถูกกระจายจนหมด ทำให้มีสถานะที่ไม่จำเป็นในเส้นทางที่จะไปสู่คำตอบถูกกระจายออกด้วย
4. ถ้ามีคำตอบจะรับประกันได้ว่า จะพบคำตอบแน่ ๆ และจะได้เส้นทางสั้นที่สุดด้วย	4. ถ้ามีคำตอบจะรับประกันได้ว่า จะพบคำตอบแน่ ๆ และจะได้เส้นทางสั้นที่สุดด้วย	4. ถ้ามีคำตอบจะรับประกันได้ว่า จะพบคำตอบแน่ ๆ และจะได้เส้นทางสั้นที่สุดด้วย
5. Worst case performance $O( V  E )$	5. Worst case performance $O( E  +  V  \log  V )$	5. Worst case performance $O( V ^3)$

จากการเปรียบเทียบอัลกอริทึมทั้งสามอัลกอริทึมในตารางที่ 2.1 ได้ทำการเลือกทฤษฎีของเบลแมน ฟอร์ด อัลกอริทึมเข้ามาประยุกต์ในการคำนวณหาเส้นทางในการคำนวณเส้นทางในการเดินทางการคำนวณนั้นจำเป็นต้องทราบจุดต้นทางและปลายทาง เพื่อใช้เป็นตำแหน่งในการคำนวณ และทราบชนิดของการค้นหาเพื่อใช้ในการกำหนดตัวแปร โดยจะใช้ระยะทางเป็นตัวแปร ถ้าชนิดการค้นหาเป็นแบบตามระยะทาง (Base on distance) และใช้ระยะทางคูณกับอัตราความติดขัด ถ้าชนิดการค้นหาเป็นตามสภาพจราจร (Base on traffic) โดยผลจากการคำนวณนั้นจะแสดงผลของการคำนวณออกเป็น หลายคำตอบ ผลจากคำตอบ จะทำการแสดงให้ผู้ใช้เลือกโดยผลที่ได้จากการคำนวณนั้นสามารถแสดงเส้นทางระยะทาง ความเร็ว และเวลา ของเส้นทางที่เราได้ทำการเลือก

### 2.3.1 เบลแมน ฟอร์ด อัลกอริทึม (Bellman Ford Algorithm)

การทำงานของเบลแมน ฟอร์ด อัลกอริทึมในการคำนวณหาเส้นทาง จะนับน้ำหนักที่เกิดขึ้นจากตัวมันเอง บวกค่าผลรวมของน้ำหนัก ที่ได้รับมาจากโหนดเพื่อนบ้านทุก ๆ โหนด แสดงขั้นตอนการทำงาน ได้ดังนี้

#### ขั้นตอนการทำงานของเบลแมน ฟอร์ด อัลกอริทึม (Pseudocode)

```
procedure BellmanFord(list vertices, list edges, vertex source)
```

```
// This implementation takes in a graph, represented as lists of vertices
```

```
// and edges, and modifies the vertices so that their distance and
```

```
// predecessor attributes store the shortest paths.
```

```
// Step 1: initialize graph
```

```
for each vertex v in vertices:
```

```
if v is source then v.distance := 0
```

```
else v.distance := infinity
```

```
v.predecessor := null
```

```
// Step 2: relax edges repeatedly
```

```
for i from 1 to size(vertices)-1:
```

```
for each edge uv in edges: // uv is the edge from u to v
```

```
u := uv.source
```

```
v := uv.destination
```

```
if u.distance + uv.weight < v.distance:
```

```
v.distance := u.distance + uv.weight
```

```
v.predecessor := u
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

// Step 3: check for negative-weight cycles

for each edge  $uv$  in edges:

$u := uv.source$

$v := uv.destination$

if  $u.distance + uv.weight < v.distance$ :

error "Graph contains a negative-weight cycle"

## 2.4 เว็บเซอร์วิส (Web Services)

การพัฒนาทางเทคโนโลยีใดๆ ล้วนแล้วแต่มีการพัฒนาเป็นยุคสมัย แต่ละยุคสมัยก็จะมี การใช้เทคโนโลยีที่มีจุดสำคัญของการพัฒนาที่เด่นชัด การพัฒนาเทคโนโลยีประยุกต์ใช้งานบน เว็บก็เช่นเดียวกัน เริ่มจากยุคแรกที่เป็นแบบสแตติกเว็บเพจ (Static Web Page) ที่เป็นเพียงแค่ไฟล์ เอกซ์เอ็มแอลไว้ให้ผู้ดูข้อมูลเฉย ๆ จนสู่ยุคที่ 2 เป็นแบบไดนามิกเว็บเพจ (Dynamic Web Page) ที่สามารถแปรเปลี่ยนตามคำร้องขอของผู้ใช้งานได้ ซึ่งเป็นการเพิ่มประสิทธิภาพของเว็บเพจ จนกระทั่งในปัจจุบันมีการพัฒนามาสู่ยุคที่ 3 คือแบบบริการผ่านเว็บเซอร์วิส โดยจุดประสงค์เพื่อ รองรับความต้องการการใช้งานที่สูงขึ้น และเป็นมาตรฐานในการทำงานร่วมกัน

เว็บเซอร์วิส คือ ซอฟต์แวร์คอมโพเนนต์ (Software Component) ที่ให้ผู้บริการนำมาสร้าง เป็นแอปพลิเคชันสำหรับให้บริการการทำงานหนึ่ง ๆ ให้กับผู้ใช้บริการทางอินเทอร์เน็ต หรือผู้ขอ บริการสามารถที่จะขอบริการจากหลาย ๆ ที่มาประกอบกันได้ โดยที่แต่ละระบบนั้นมีความเป็น อิสระจากกัน (Loosely Coupled) ซึ่งปัจจัยพื้นฐานของเว็บเซอร์วิสที่ควรมี ซึ่งมีความเกี่ยวข้องกัน ในทางเทคนิคและทางธุรกิจ คือ

- การรวมซอฟต์แวร์ต่างระบบกันจะต้องอนุญาตให้แต่ละระบบมีความอิสระต่อกัน
- อินเทอร์เน็ตทางด้านการบริการของซอฟต์แวร์ควรจะเผยแพร่สู่สาธารณชนและสามารถเข้าถึงได้ง่าย
- ข้อความที่ใช้ติดต่อกันของการทำงานแบบแอปพลิเคชันกับแอปพลิเคชันต้อง สอดคล้องกับมาตรฐานเปิดบนอินเทอร์เน็ต
- แอปพลิเคชันสามารถสร้างได้จากการใช้ซอฟต์แวร์คอมโพเนนต์ทั้งจากภายใน และภายนอกองค์กร โดยสร้างตามการดำเนินธุรกิจหลักขององค์กร
- มีแหล่งซอฟต์แวร์คอมโพเนนต์ที่ทำได้ง่าย ซึ่งช่วยเพิ่มความยืดหยุ่นในการสร้าง แอปพลิเคชันที่มีคุณลักษณะเฉพาะตามกระบวนการการดำเนินการทางธุรกิจ

- การนำซอฟต์แวร์จากภายนอกมาใช้ใหม่ ช่วยให้เกิดการลดต้นทุนและช่วยปรับปรุงบริการให้แก่ลูกค้า
- ซอฟต์แวร์สามารถขายเป็นบริการได้

#### 2.4.1 หลักการพื้นฐานของเทคโนโลยีต่าง ๆ ที่พัฒนามาสู่เว็บเซอร์วิส

- การพัฒนาโปรแกรมแบบซอฟต์แวร์คอมโพเนนท์ ตามแนวคิดของการออกแบบและพัฒนาโปรแกรมเชิงวัตถุ (Object-Oriented Concept)
- การออกแบบระบบแบบกระจายจากศูนย์กลาง (Distributed Computing) ซึ่งเป็นเป้าหมายสำคัญของการพัฒนาระบบตามสถาปัตยกรรมแบบไคลเอนต์เซิร์ฟเวอร์ (Client-Server)
- การทำ อีดีไอ (Electronic Data Interchange : EDI) ซึ่งสร้างขึ้นโดยกำหนดรูปแบบและมาตรฐานของข้อมูลสำหรับการทำธุรกิจ
- การบูรณาการของซอฟต์แวร์ต่างระบบอีเอไอ (Enterprise Application Integration : EAI) ที่อยู่บนพื้นฐานของความต้องการใช้ข้อมูลร่วมกันทั้งการแลกเปลี่ยนข้อมูลระหว่างแอปพลิเคชันให้สามารถทำงานได้อย่างถูกต้องเหมาะสม
- รูปแบบการให้บริการซอฟต์แวร์แบบเอสพี (Application Service Provider : ASP)
- แนวคิดการทำเหมืองข้อมูล (Data Mining) ซึ่งต้องการนำข้อมูลที่ถูกจัดเก็บไว้ในรูปแบบที่แตกต่างกันตามแหล่งต่าง ๆ มาใช้งานร่วมกัน โดยที่เว็บเซอร์วิสได้นำหลักการต่าง ๆ เหล่านี้มาดำเนินการปรับปรุง แก้ไขข้อจำกัดต่าง ๆ โดยการคิดค้นเทคโนโลยีที่เป็นมาตรฐานเปิด (Opened Standard) และไม่มีใครเป็นเจ้าของ (Neural Vendor)

#### 2.4.2 ลักษณะของเว็บเซอร์วิส

การพัฒนาเว็บเซอร์วิสใช้สถาปัตยกรรมบริการในลักษณะที่เรียกว่า สถาปัตยกรรมของแนวคิดทางด้านบริการ หรือ เอสโอเอ (Service-Oriented Architecture : SOA) เป็นแนวคิดเบื้องต้น แอปพลิเคชันส่วนใหญ่ในโลกของธุรกิจที่ใช้งานในปัจจุบัน เป็นแอปพลิเคชันและระบบย่อยที่ถูกสร้างขึ้น มีการทำงานที่ต้องสัมพันธ์กันอย่างไม่เป็นอิสระจากกัน การเปลี่ยนแปลงการทำงานในระบบย่อยหรือแอปพลิเคชันหนึ่ง อาจจะมีผลกระทบกับอีกแอปพลิเคชันหนึ่ง หรือบางครั้งอาจจะกระทบทั้งระบบ ทำให้การบำรุงรักษานั้นทำได้ยาก และมีต้นทุนที่สูงขึ้น รวมทั้งยังเป็นข้อจำกัดในการเชื่อมต่อกับระบบของคู่ค้าอื่น ๆ เอสโอเอไม่ใช่แนวคิดใหม่ แต่ได้เกิดขึ้นมานานแล้ว ซึ่งอยู่ในส่วนหนึ่งของแนวคิดการออกแบบระบบแบบกระจายจากศูนย์กลาง

(Distributed Computing Concepts) เอสโอเอนั้นเป็นแนวคิดที่ได้รับการยอมรับและประสบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ผลสำเร็จในระดับหนึ่ง การพัฒนาเอสโอเอนั้นทำกันบนมาตรฐานเปิด ซึ่งได้รับการรับรองจากผู้ผลิตซอฟต์แวร์ชั้นนำและยังได้รับความร่วมมือกันรับรองมาตรฐานของเทคโนโลยีต่าง ๆ ที่ประกอบกันเป็นเว็บเซอร์วิส เช่นเอสโอเอพี (SOAP) , ดับเบิลยูเอสดีเอล (WSDL) , ยูดีดีไอ (UDDI) เป็นต้น

เอสโอเอมีส่วนประกอบหลัก 3 ส่วนคือ

1. ผู้ให้บริการ (Service Provider)
2. ผู้ขอใช้บริการ (Service Requester)
3. ตัวแทนของผู้ให้บริการ (Service Broker)

โดยส่วนประกอบหลักทั้ง 3 ส่วนนี้ สามารถติดต่อถึงกันได้โดยใช้ฟังก์ชันพื้นฐาน (Primary Function) คือการประกาศ (Publish) การค้นหา (Find) และการเรียกใช้ (Bind) ซึ่งฟังก์ชันทั้ง 3 มีการทำงานดังนี้

1. ผู้ให้บริการทำการประกาศบริการที่ตนเองให้บริการ ไปยังตัวแทนของผู้ให้บริการ ซึ่งตัวแทนของผู้ให้บริการจะทำการเก็บบันทึกไว้ใน ไคลเอนท์ของเซอร์วิส
2. ผู้ขอใช้บริการจะทำการค้นหาเซอร์วิสจากตัวแทนของผู้ให้บริการ
3. เมื่อพบเซอร์วิสที่ต้องการแล้ว ผู้ให้บริการและผู้ขอใช้บริการจะทำการติดต่อกัน โดยผู้ขอใช้บริการจะทำการเรียกใช้เซอร์วิส ไปยังผู้ให้บริการนั้น

จากแนวคิดของเอสโอเอถูกนำมาใช้เป็นหลักการพื้นฐานของการพัฒนาเทคโนโลยีด้านเว็บเซอร์วิส ซึ่งเทคโนโลยีที่ถูกพัฒนาขึ้นมานี้เป็นมาตรฐานเปิดบนอินเทอร์เน็ตที่เกิดจากการทำงานร่วมกันของนักวิจัยและที่ปรึกษาจากบริษัทซอฟต์แวร์ต่างๆ

## 2.4.3 ภาษาที่ใช้ในเว็บเซอร์วิส

### 2.4.3.1 เอกซ์เอ็มแอล(XML)

เอกซ์เอ็มแอล (Extensive Markup Language : XML) เป็นภาษาที่ใช้กำหนดรูปแบบของคำสั่งภาษาเอกซ์เอ็มแอลหรือที่เรียกว่าเมตา ดาต้า (Meta Data) ซึ่งจะใช้สำหรับกำหนดรูปแบบของคำสั่งมาร์กอัพต่างๆแต่มีข้อแตกต่างกับเอกซ์เอ็มแอลที่เป็นภาษามาร์กอัพ (Markup Language) ซึ่งเอกซ์เอ็มแอลได้รับการพัฒนามาจากเอสจีเอ็มแอล (Standard Generalized Markup Language : SGML) ที่เป็นข้อกำหนดในการสร้างหรือจัดทำเอกสารในรูปแบบอิเล็กทรอนิกส์ที่กำหนดโดยดับเบิลยูทีซี (World Wide Web Consortium : W3C) ซึ่งเป็นภาษาที่นิยมใช้และได้รับการพัฒนาให้มีประสิทธิภาพสูงสุดในการทำงานบนเว็บ

### ส่วนประกอบพื้นฐานของเอกซ์เอ็มแอล

โดยเอกซ์เอ็มแอลจะประกอบด้วย 3 ส่วนพื้นฐานด้วยกันคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. เอกสารข้อมูล (data document)
2. เอกสารนิยามความหมาย (definition document)
3. นิยามภาษา (definition language)

หน้าที่ของเอกซ์เอ็มแอลและความแตกต่างในเรื่องการใช้งานระหว่างเอกซ์เอ็มแอลกับเอกซ์เอ็มแอลในปัจจุบันนี้ได้มีการพัฒนาภาษามาร์กอัปตามข้อกำหนดของเอกซ์เอ็มแอลแล้ว เช่น เอกซ์เอ็มแอลสำหรับควบคุมข้อมูลมัลติมีเดียเอกซ์เอ็มแอลเป็นส่วนหนึ่งของเอกซ์เอ็มแอลแต่เอกซ์เอ็มแอลจะให้รายละเอียดเกี่ยวกับข้อมูลต่าง ๆ เช่น ชื่อเมือง อุณหภูมิ ความกดอากาศ เป็นต้น ภาษาเอกซ์เอ็มแอล เป็นการกำหนดแท็กต่าง ๆ ที่จะให้ข้อมูลแสดงผลในรูปแบบใด ซึ่งข้อมูลสามารถแสดงผลได้หลายรูปแบบไม่ว่าจะเป็นตาราง หรือ แท็กธรรมดาขึ้นอยู่กับข้อกำหนดของเอกซ์เอ็มแอล และเอกซ์เอ็มแอลยังสามารถให้รายละเอียดของเนื้อหาเอกสารเรียกว่า ดีทีดี (Document Type Definition : DTD) ที่จะแสดงหรือซ่อนส่วนใดของเอกสาร ดังนั้นเอกซ์เอ็มแอลเกิดประโยชน์เต็มที่เมื่อนำมาใช้งานร่วมกับเอกซ์เอ็มแอล เนื่องจากเอกซ์เอ็มแอลมีความพร้อมในแง่ของรายละเอียดการนำข้อมูลตลอดจน โครงสร้างข้อมูลมาแสดงในรูปแบบข้อความผ่านทางเอกซ์ทีทีพี และมีความสามารถในการจัดข้อมูลซึ่งการเขียนเว็บเพจโดยใช้ภาษาเอกซ์เอ็มแอลผู้พัฒนาสามารถกำหนดได้ว่าส่วนไหนจะเป็นตัวหนา ตัวเอียง หรือ ตัวอักษรเป็นแบบไหน ส่วนเอกซ์เอ็มแอลนั้นจะเป็นการเตรียมส่วนของข้อมูลที่จะนำไปใส่ในช่องที่กำหนดตามการเขียนของเอกซ์เอ็มแอล เช่น ข้อมูลราคา หรือ ราคาที่ตั้งสำหรับการจัดรายการส่งเสริมการขาย อัตราภาษี ค่าขนส่ง เป็นต้น

### คุณลักษณะต่างๆของเอกซ์เอ็มแอล

เอกซ์เอ็มแอลสามารถจัดการ ได้หลายรูปแบบทั้ง องค์ประกอบ โครงสร้างเอกสาร ลักษณะประเภทแอททริบิว และอีลิเมนต์ (Element) โดยเป็นภาษาที่ถูกรอกออกมาเฉพาะสำหรับการพัฒนาโปรแกรมเว็บ เพื่อการจัดส่งข้อมูลสารสนเทศ ตลอดจนถูกนำมาใช้สร้างภาษามาร์กอัปซึ่งตรงกันข้ามกับเอสจีเอ็มแอลที่มีความซับซ้อนมากกว่า ในทางปฏิบัติเอกสารเอกซ์เอ็มแอลมีพื้นฐานเพื่อให้การสร้างเอกสารมีรูปแบบที่ถูกต้องในการใช้งานจริง โดยปกติแล้วเอกซ์เอ็มแอลที่มีลักษณะแบบเวกเตอร์และการสื่อสารระหว่างโปรแกรมต่าง ๆ นอกจากนี้เอกซ์เอ็มแอลยังสามารถช่วยในการประมวลผลข้อมูลแล้วส่งผ่านไปโปรแกรมประยุกต์ไปยังแหล่งเก็บข้อมูล เอกซ์เอ็มแอลเป็นเอกสารที่เขียนด้วยข้อความปกติธรรมดาสามารถสร้างเอกสารหรือแก้ไขไฟล์ เอกซ์เอ็มแอลได้อย่างง่ายดายด้วยโปรแกรมแก้ไขข้อความ (Text Editor) หากต้องการใช้โปรแกรมที่มีความสามารถพิเศษมากกว่านี้ก็ต้องใช้โปรแกรมแก้ไขข้อความที่อยู่ในชุดโปรแกรม ไมโครซอฟท์ วิวอลสตูดิโอ เช่น ไมโครซอฟท์ วิวอล ซี++ (C++) , ไมโครซอฟท์ วิวอล เบสิก (Microsoft Visual Basic) และ ไมโครซอฟท์ วิวอล ฟอกซ์โปร (Microsoft Visual Foxpro) เป็นต้น เอกซ์เอ็มแอลใช้แท็กเริ่มต้น [Tag] และแท็กปิด [/Tag] เสมอเช่นเดียวกับภาษามาร์กอัป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

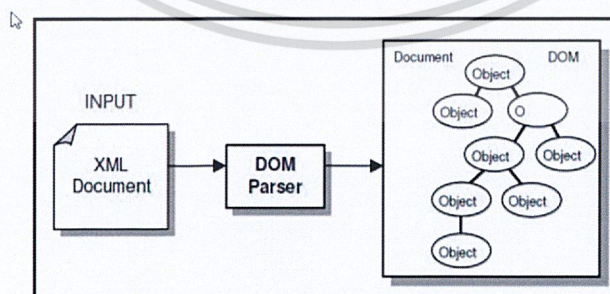
เรียกว่าอิลิเมนต์ เป็นการแบ่งแยกระหว่างข้อมูลและคำสั่งเพื่อระบุว่าข้อมูลที่อยู่ระหว่างแท็กดังกล่าวคือข้อมูลอะไร ส่วนประกอบในเอกสารเอกซ์เอ็มแอลมีอยู่ 2 ส่วนหลักคือโปรล็อกอิลิเมนต์ (Prolog Element) และด็อกคิวเมนต์อิลิเมนต์ (Document Element หรือ Root Element) ในส่วนของเอกสารเอกซ์เอ็มแอลคืออิลิเมนต์เดี่ยว ซึ่งสามารถบรรจุอิลิเมนต์เพิ่มเติมในเอกสารเอกซ์เอ็มแอลได้โดยในเอกสารเอกซ์เอ็มแอลนั้นอิลิเมนต์จะแสดงลักษณะโครงสร้างของเอกสารและจะแสดงส่วนประกอบเนื้อหาของเอกสารอยู่ภายในสัญลักษณ์อิลิเมนต์ประกอบด้วย แท็กเริ่มต้น (start-tags) เนื้อหาภายในอิลิเมนต์ และแท็กสิ้นสุด (end-tags) ส่วนเนื้อหาภายในอิลิเมนต์สามารถเป็นได้ทั้งข้อมูล หรืออิลิเมนต์อื่น ๆ ที่ซ่อนอยู่ภายในหรือทั้งสองแบบ

### วิธีการเข้าถึงข้อมูลในเอกซ์เอ็มแอล

เพื่อให้สามารถเข้าถึงข้อมูลและดึงข้อมูลนำมาใช้จากเอกสารเอกซ์เอ็มแอลได้ วิธีการเข้าถึงข้อมูลในเอกสารเอกซ์เอ็มแอลโดยใช้ตัววิเคราะห์โครงสร้างของเอกสาร (Parser) ซึ่งปัจจุบันมีวิธีการที่นิยมใช้อยู่ 2 วิธีคือ

#### 1. ดีโอเอ็ม (Document Object Model : DOM)

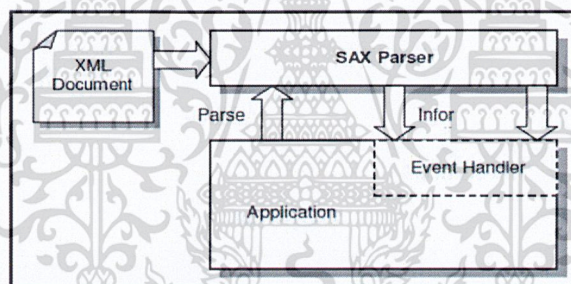
เป็นการทำงานแบบทรีเบสเพอร์เซอร์ (Tree- Based parser) และใช้การเข้าถึงข้อมูลด้วยวิธีการรันคอม แอคแซส (Random-Access) คือจะประมวลโครงสร้างของเอกสารเอกซ์เอ็มแอลให้เป็นโครงสร้างต้นไม้ (tree structure) แบบจำลองวัตถุแบบลำดับชั้น (hierarchical object model) ซึ่งในโครงสร้างต้นไม้จะประกอบไปด้วย โหนดอิลิเมนต์ (element node) โดยภายในแต่ละโหนดอิลิเมนต์ประกอบไปด้วยข้อมูลที่เกี่ยวข้องกับอิลิเมนต์นั้นตามโครงสร้างและข้อมูลในเอกสารเอกซ์เอ็มแอลซึ่งจะประกอบไปด้วยข้อมูลทั้งหมดของเอกสารเอกซ์เอ็มแอล เพื่อให้แอปพลิเคชันสามารถเข้าหาจุดต่าง ๆ ของโครงสร้างต้นไม้ได้โดยที่ดีโอเอ็มจะโหลดเอกซ์เอ็มแอลทั้งหมดคือข้อมูลในรูทโหนดทั้งหมด เข้ามาเป็นทรีในหน่วยความจำหลักของคอมพิวเตอร์ก่อนจึงทำงานได้ดังที่แสดงในรูปที่ 2.7



รูปที่ 2.7 แสดงวิธีการเข้าถึงข้อมูลโดยวิธี DOM Parsing

## 2. เอสเอเอกซ์ (Simple API for XML : SAX)

เป็นการทำงานแบบอีเวนท์ เบสเอพีไอ (Event-Based API) และใช้การเข้าถึงข้อมูลด้วยวิธีการซีควนเชียล แอคเซส (Sequential-Access) คือจะเข้าถึงข้อมูลโดยใช้เหตุการณ์เป็นตัวกำหนดด้วยการอ่านเอกสารเอกซ์เอ็มแอลและตอบสนองต่อมาร์คอัพที่อ่านพบ โดยถือว่าเป็นเสมือนเหตุการณ์ต่าง ๆ ซึ่งรายงานข้อมูลตั้งแต่จุดเริ่มต้นจนถึงจุดสิ้นสุดของอีลิเมนต์ต่าง ๆ ไปให้แอปพลิเคชัน โดยไม่ต้องมีการสร้างโครงสร้างต้นไม้ขึ้นมา ส่วนการเข้าถึงข้อมูลจะอาศัยกระบวนการสแกน โหนด ซึ่งจะทำการสแกนไปที่ละเส้น ทีละแท็ก เริ่มต้นจากรูทโหนด มาถึงส่วนที่เราต้องการติดต่อ จากนั้นก็ใช้งานฟังก์ชันของเอสเอเอกซ์ (SAX) ต่อไปส่วนใหญ่วิธีนี้มักใช้ในงานที่เกี่ยวข้องกับการรับ-ส่งข้อมูลเอกสารเอกซ์เอ็มแอลผ่านระบบเครือข่ายเนื่องจากวิธีนี้ให้ความรวดเร็วและสิ้นเปลืองหน่วยความจำน้อย แต่ในทางโปรแกรมมิ่งมักไม่ใช่เนื่องจากวิธีนี้เป็นวิธีที่ยากต่อการใช้งาน เพราะการเข้าถึงข้อมูลเป็นไปตามลำดับเหตุการณ์ซึ่งไม่สอดคล้องกับวิธีการทางโปรแกรมในการเข้าถึงข้อมูลดังแสดงในรูปที่ 2.8



รูปที่ 2.8 แสดงวิธีการเข้าถึงข้อมูลโดยวิธีเอสเอเอกซ์ เพอร์ซิง (SAX Parsing)

### 2.4.3.2 ดับเบิลยูเอสดีแอล (WSDL)

ดับเบิลยูเอสดีแอล (Web Services Description Language : WSDL) คือข้อมูลรูปแบบ เอกซ์เอ็มแอลที่ใช้สำหรับอธิบายเซอร์วิสบนเครือข่ายที่ให้บริการ เราสามารถใช้ดับเบิลยูเอสดี แอลเพื่อสร้างไฟล์ที่ระบุถึงเซอร์วิสที่เซิร์ฟเวอร์ให้บริการและกลุ่มปฏิบัติการภายในบริการแต่ละ เซิร์ฟเวอร์สนับสนุนสำหรับปฏิบัติการแต่ละตัวไฟล์ ดับเบิลยูเอสดีแอลยังทำหน้าที่อธิบาย รูปแบบข้อมูลที่ไคลเอนต์ต้องปฏิบัติตามในการร้องขอปฏิบัติการ คิดค้นขึ้น โดย 2 บริษัทยักษ์ ใหญ่ คือ ไอบีเอ็มและไมโครซอฟต์

เนื่องจากไฟล์ดับเบิลยูเอสดีแอลทำหน้าที่กำหนดความต้องการทั้งเซิร์ฟเวอร์และไคล เอนต์ ไฟล์นี้จึงเสมือนเป็นสัญญาสำหรับเครื่องทั้งสองฝั่ง เซิร์ฟเวอร์ตกลงที่จะให้บริการที่กำหนด

ไว้ เมื่อไคลเอนต์ส่งคำร้องขอเอสไอเอพีในรูปแบบที่เหมาะสมเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ในทางปฏิบัติหากเราต้องการสร้างเว็บเซอร์วิสขึ้นมาเป็นของตนเอง ก็สามารถสร้างเอกสารดับเบิลยูเอสดีแอลได้โดยอัตโนมัติ จึงไม่ต้องไปกังวลในรายละเอียดในข้อกำหนดในดับเบิลยูเอสดีแอลมากนัก (สามารถหาอ่านเพิ่มเติมได้ที่ <http://www.w3.org/TR/wsdl>)

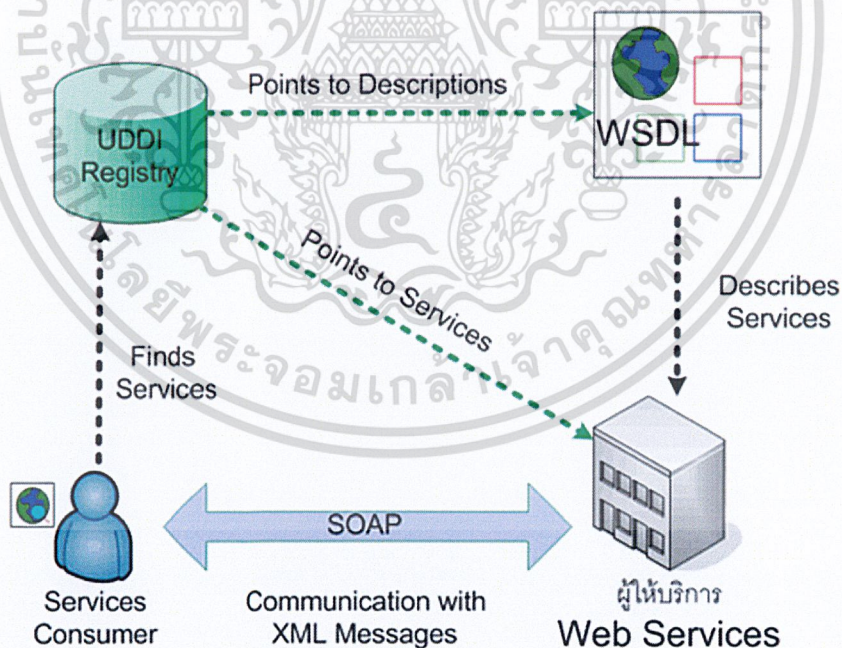
### 2.4.4 ชนิดของเว็บเซอร์วิส

2.4.4.1 โซป (Simple Object Access Protocol : SOAP) เป็นเอกซ์เอ็มแอล เบส (XML-based) โพรโทคอล

น้ำหนักเบา (light weight protocol) สำหรับการแลกเปลี่ยนข้อมูลในสภาวะแวดล้อมแบบกระจายศูนย์ (decentralized, distributed environment) โซปได้กำหนดเมตาดาทาโพรโทคอลระหว่างผู้ขอบริการ (requester) กับผู้ให้บริการ (provider)

ประกอบด้วย 3 ส่วนคือ

1. เครื่องห่อหุ้ม (envelope) ซึ่งกำหนดเค้าโครงสำหรับอธิบายข้อความ และมีกระบวนการอย่างไรบ้าง
2. กฎการเข้ารหัสสำหรับแสดงตัวอย่างของการกำหนดแอปพลิเคชัน
3. รูปแบบสำหรับการแทนฟังก์ชันและตอบสนอง



รูปที่ 2.9 แสดงการทำงานของโซป เว็บเซอร์วิส

อธิบายรูปที่ 2.9 โซปมีขนาดเล็ก สามารถยืดขยายได้ เป็นโพรโทคอลพื้นฐานของเอกซ์เอ็มแอลสำหรับการแลกเปลี่ยนข้อมูลขึ้นแรกโซปจะกำหนดโครงสร้างและรูปแบบของข้อมูล เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และยังกำหนดเขตสำหรับกฎการเข้ารหัสข้อมูลที่เรียงต่อกัน และมีรูปแบบสำหรับการเรียก ฟังก์ชัน รูปแบบของโซปได้เพิ่มความสามารถในการติดต่อระหว่างโปรโตคอลอย่างแพร่หลาย เช่น เอชทีทีพี

### ข้อดีของโซป

โซปมีความสามารถให้เราเรียกใช้คอมโพเนนท์หรือเว็บเซอร์วิสข้ามเครื่อง ข้ามแพลตฟอร์ม ข้ามภาษาได้อย่างง่ายดาย โดยอาศัยโปรโตคอลที่มีอยู่เดิมในอินเทอร์เน็ตอย่างเฮชทีทีพี (HTTP) และรูปแบบข้อความที่สื่อสารกันด้วยภาษาเอกซ์เอ็มแอลซึ่งมีลักษณะเป็นข้อความธรรมดาๆ ปิดล้อมด้วยแท็ก ทำให้เข้าใจได้ในทุกแพลตฟอร์ม ขอเพียงมีเอกซ์เอ็มแอล เพอร์เซอร์ (XML parser) มาแปลข้อความนั้น ก็เข้าใจได้แล้ว

นอกจากนี้โซปยังมีข้อดีอีกอย่างที่เหนือกว่าโปรโตคอลแบบเดิม เช่น ดีซีไอเอ็ม (DCOM), อาร์เอ็มไอ (RMI) หรือ ไอไอโอพี (IIOPI) คือ ข้อความโซปสามารถผ่านระบบไฟร์วอลล์ (firewall) ป้องกันการบุกรุก ทั้งนี้เนื่องจากข้อความโซปทำงานอยู่กับโปรโตคอลเอชทีทีพี ซึ่งโดยปกติของไฟร์วอลล์จะเปิดให้การสื่อสารด้วยโปรโตคอลเอชทีทีพีผ่านได้อย่างสะดวก ดังนั้นข้อความโซปจึงผ่านได้เช่นกัน โดยไม่มีปัญหาอะไร ในขณะที่โปรโตคอลแบบเดิม ๆ เหล่านั้นไฟร์วอลล์มักจะไม่นิยมให้ผ่านง่ายๆ ซึ่งจุดนี้คือข้อเสียที่สำคัญที่สุดและทำให้โซปได้เปรียบขึ้นมาทันที

### ข้อเสียของโซป

ยังมีความจริงที่ปฏิเสธไม่ได้ว่าโซปมีข้อด้อยอยู่บางประการ ประการแรก เนื่องจากลักษณะของข้อความโซปเป็นเอกสารเอกซ์เอ็มแอลที่ให้เสียเวลาให้การแปลกลับมาเป็นรูปแบบโปรแกรมที่เข้าใจ และประการที่สองในกรณีที่โซปทำงานอยู่กับโปรโตคอลเอชทีทีพีซึ่งโดยปกติโปรโตคอลชนิดนี้มีสมรรถนะให้การรับและส่งข้อมูลต่ำ โดยเฉพาะเมื่อเปรียบเทียบกับโปรโตคอลอื่นๆ อย่างไอไอโอพี

ด้วยเหตุนี้โซปจึงมีอัตราการรับและส่งข้อมูลที่ต่ำด้วย และเหตุผลทั้งสองประการนี้ ทำให้อัตราการรับส่งข้อมูลต่ำว่าโปรโตคอล ดีซีไอเอ็ม, อาร์เอ็มไอ หรือ ไอไอโอพี

#### 2.4.4.2 ยูดีดีไอ (UDDI)

ยูดีดีไอ (Universal Description, Discovery and Integration : UDDI) เป็นโปรโตคอลที่ถ่ายคอมพิวเตอร์ 4 ค่ายยักษ์ คือ ไอบีเอ็ม (IBM) , ซันไมโครซิสเต็ม (Sun Microsystem) , ฮิวเลตต์แพคการ์ด (Hewlett-Packard) และ อะริบา (Ariba) พยายามผลักดันให้เป็นมาตรฐานอินเทอร์เน็ตสากล ยูดีดีไอถูกสร้างขึ้นมาเพื่อเป็นมาตรฐานในการค้นหาบริการเว็บเซอร์วิส สำหรับคู่ค้าทางธุรกิจ (Business partner) โดยยูดีดีไอเปรียบเสมือนฐานข้อมูลขนาดใหญ่ซึ่งมีข้อมูลของเว็บเซอร์วิสที่ให้บริการทางธุรกิจ

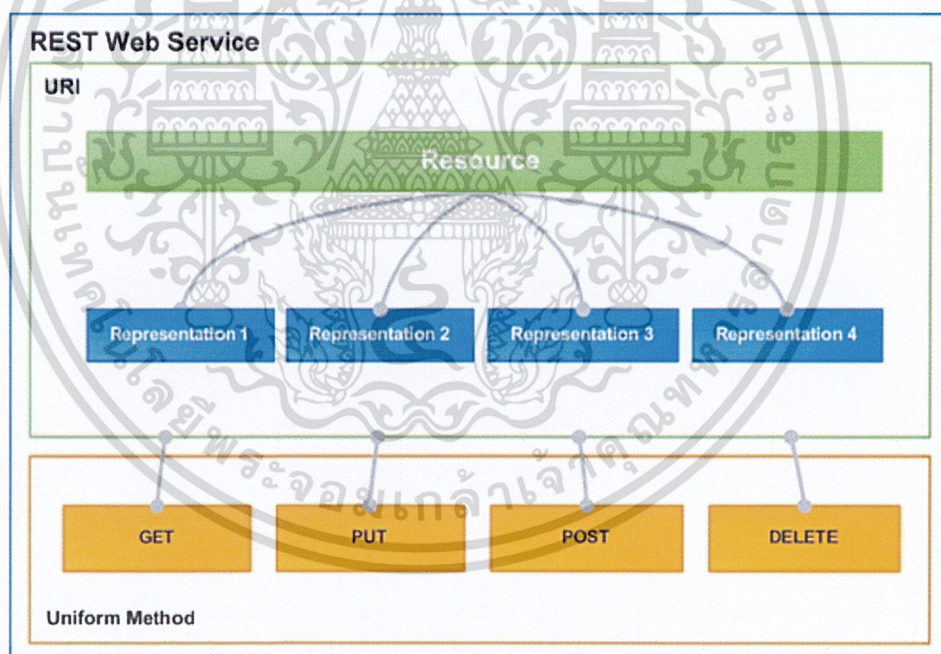
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบการทำงานของยูดีดีไอจะเปลี่ยนแปลงวิธีการค้นหาข้อมูลบนอินเทอร์เน็ตที่เราให้อยู่ในปัจจุบัน จากการพิมพ์ชื่อ โดเมนเนมในช่องแอดเดรสบาร์ไปเป็นการใช้คำสามัญหรือชื่อเฉพาะในการค้นหา และเข้าสู่เว็บไซต์แทนยูดีดีไอจะทำหน้าที่เหมือนสมุดหน้าเหลือง ที่จำแนกผู้จดทะเบียนตามรูปแบบของธุรกิจ แยกประเภทตามกิจการ

โดยสรุปยูดีดีไอเหมือน โครงสร้างพื้นฐานที่มารองรับการค้นหาข้อมูลแบบใหม่ โดยใช้ตัวเลขหรือคำสามัญแทน โดเมนเนม ยูดีดีไอจะมาเป็นอีกมาตรฐานหนึ่งสำหรับการติดต่อกับไอพีแอดเดรสเหมือนที่เคยมีโดเมนเนมหรือยูอาเอลในการทำให้ผู้เข้าสู่เว็บไซต์รู้จักและจดจำได้โดยง่าย โดยที่เว็บไซต์สำหรับค้นหาเว็บเซอร์วิสมีอยู่หลายที่อย่างเช่น

- <http://uddi.microsoft.com/search.aspx>
- <http://www-3.ibm.com/services/uddi/testregistry/find>
- <http://uddi.org>

#### 2.4.4.3 เรส (REST)



รูปที่ 2.10 แสดงการทำงานของเรส เว็บเซอร์วิส

เรส (REST, Representational State Transfer) เป็นอีกทางเลือกหนึ่งที่น่าเว็บเทคโนโลยีพื้นฐาน ได้แก่ยูอาร์แอลเอชทีทีพี (รวมทั้งพุท, เก็ท, โปส และ คีลีท โอเปอร์เรชั่น) และเอกซ์เอ็มแอลมาใช้ประโยชน์อย่างเต็มที่ โดยเฉพาะอย่างยิ่งการใช้ประโยชน์จากยูอาร์แอลหรือยูอาร์ไอที่ถูกนำมาใช้แสดงเมธอดและพารามิเตอร์ โดยเรสเป็นการใช้เอชทีทีพีในการรับส่งค่า และช่วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้เราสามารถเรียกยูอาร์แอลในรูปแบบที่เฉพาะเจาะจงได้ เราไม่สามารถจัดได้ว่าเป็นมาตรฐาน หากแต่เป็นรูปแบบที่นักพัฒนาสามารถยึดเป็นแนวทางในการพัฒนาได้ การทำงานของเรส เว็บ เซอร์วิส แสดงดังรูปที่ 2.10

## 2.5 ภาษาจาวา

นั้นถูกออกแบบขึ้นมาให้มีลักษณะดังนี้

1. เป็นภาษาที่ง่ายในการเรียนรู้และใช้งาน ซึ่งอาจจะสามารถตีความหมายได้ดังนี้

- ภาษาจาวา ได้นำไวยากรณ์ภาษาส่วนใหญ่มาจากภาษาซี และ ซี++
- ภาษาจาวา มีจำนวนของภาษาไม่มากและไม่ซับซ้อน โดยตัดกลไกของภาษาซี และ ภาษาซี++ ที่ทำให้ยุ่งยากออกไปบางส่วน แต่ในขณะเดียวกันก็เพิ่มความสามารถให้กับคอมไพเลอร์ของภาษาจาวา ทำให้ไม่มีพรี โพรเซสเซอร์ คอมมาน (preprocessor commands) และในเมื่อภาษาจาวาเป็นภาษาเชิงวัตถุแล้ว โครงสร้างต่าง ๆ เช่น สตรัคเจอร์ (structures) หรือยูเนียน (unions) จึงไม่มีความจำเป็น จะพบว่าภาษาจาวานั้นรอบคอบในเรื่องของการทำให้เป็นภาษาเชิงวัตถุ เป็นอย่างมาก จากการที่ได้ตัดตัวส่วนของเฟรนด์เมทอด (friend method) ออกไป เพราะอาจจะทำลายแบบแผนของการออกแบบภาษาเชิงวัตถุที่ดี
- ภาษาจาวา นั้นมีส่วนของการนำกลับมาทำซ้ำทำให้งานบางงานไม่จำเป็นที่จะต้องเชี่ยวชาญ ในเรื่องนั้นนั่นก็สามารถทำออกมาได้ เนื่องจากสามารถนำ โปรแกรมที่มีอยู่แล้วกลับมาใช้ได้ใหม่นั้นเอง
- ในภาษาจาวา มีกฎเกณฑ์ที่เกี่ยวข้องกับอโตเมติก ไทป์ คอรัเซียน (automatic type coercion) ซึ่งเป็นการเปลี่ยนแปลงค่าระหว่างชนิดของตัวแปรที่ต่างกันช่วยให้ลดความยุ่งยากของการเปลี่ยนระหว่างชนิดตัวแปรลงได้

2. โปรแกรมที่สร้างขึ้นโดยภาษาจาวาจะไม่มีการผลิตเล็ก ๆ น้อย ๆ ที่ไม่เกี่ยวกับ ธรรมชาติของโปรแกรม ซึ่งภาษาจาวานั้นถูกทำให้มีความอดทน (robust) ด้วยวิธีการ ดังนี้

- ภาษาจาวาเป็นประเภทสตรองลิ ไทป์ (strongly type) หมายถึงภาษาที่เน้นความถูกต้องของชนิดข้อมูลที่ใช้ในโปรแกรม คอมไพเลอร์ของภาษาประเภทนี้จะทำการตรวจสอบว่าโปรแกรมการจัดการกับชนิดข้อมูลของตัวแปรถูกต้องหรือไม่

3. โปรแกรมภาษาจาวา โดยมากแล้วจะถูกส่งผ่านระบบเครือข่ายไปทำงานบนเครื่องคอมพิวเตอร์เครื่องอื่น ทำให้ต้องมีการรับประกันความปลอดภัยเพื่อไม่ให้เกิดอันตรายกับเครื่องที่ใช้งานได้ ซึ่งทำให้จาวาเป็นภาษาที่มีคุณสมบัติเรื่องความปลอดภัยสูงมาก
4. จุดมุ่งหมายสำคัญของการออกแบบภาษาจาวาคือ โปรแกรมต้องสามารถทำงานบนเครื่องที่มีแพลตฟอร์มที่ต่างกันได้

## 2.6 โปสเกรสเอสคิวแอล (PostgreSQL)

โปสเกรสเอสคิวแอล คือระบบจัดการฐานข้อมูลเชิงวัตถุสัมพันธ์ (Object-Relational Database Management System หรือ ORDBMS) ซึ่งปรับปรุงมาจากต้นแบบระบบฐานข้อมูล POSTGRES 4.2 ของมหาวิทยาลัยแคลิฟอร์เนีย

โปสเกรสเอสคิวแอลเป็น โปรแกรม โอเพ่นซอร์สที่สามารถนำไปใช้งานได้โดยไม่มีค่าใช้จ่าย และสนับสนุนมาตรฐานเอสคิวแอล : 2003 และมีความสามารถต่างๆ เพิ่มขึ้นมากมา

โปสเกรสได้นำเอาแนวคิดของ ข้อมูลเชิงวัตถุสัมพันธ์ มาใช้ ซึ่งปัจจุบันก็เป็นต้นแบบของระบบฐานข้อมูลนำมาทำการค้ามากมาย และเป็นธรรมดาที่ฐานข้อมูลเชิงวัตถุสัมพันธ์ (Relational Database Management Systems : RDBMS) จะมีการสนับสนุนรูปแบบข้อมูลที่ประกอบไปด้วยกลุ่มของข้อมูลที่มีความสัมพันธ์กัน แต่โปสเกรสเอสคิวแอลได้มีการเพิ่มเติมสิ่งที่น่าสนใจเพิ่มขึ้นทำให้ผู้ใช้งานสามารถจัดการการขยายระบบของตนเองได้ง่ายขึ้นดังนี้ คือ สืบทอด ชนิดข้อมูล และ ฟังก์ชัน

นอกจากนี้โปสเกรสเอสคิวแอลยังได้มีการเพิ่มเติมความสามารถให้ระบบมีประสิทธิภาพ และมีความยืดหยุ่นเพิ่มขึ้นหลายอย่าง เช่น ข้อจำกัด ทริกเกอร์ กฎ และ ความสมบูรณ์ของธุรกรรม

ความสามารถเหล่านี้นำไปให้โปสเกรสเอสคิวแอลถือว่าเป็นฐานข้อมูลประเภท ข้อมูลเชิงวัตถุสัมพันธ์

### 2.6.1 สถาปัตยกรรมการทำงานของโปสเกรสเอสคิวแอล

สถาปัตยกรรมมาตรฐานสำหรับระบบที่ทำงานบนระบบเครือข่าย คือ สถาปัตยกรรมที่เรียกว่า ไคลเอนต์-เซิร์ฟเวอร์ซึ่งในสถาปัตยกรรมนี้ประกอบด้วย

- เครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นเซิร์ฟเวอร์ซึ่งจะมีโปรแกรมทำงานอยู่ ซึ่งจากนี้ไปจะขอเรียกว่า เซิร์ฟเวอร์โปรแกรม

- เครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นไคลเอนต์ซึ่งจะมี โพรเซสทำงานอยู่ ซึ่งจากนี้ไปจะขอเรียกว่า 'ไคลเอนต์ โพรเซส'

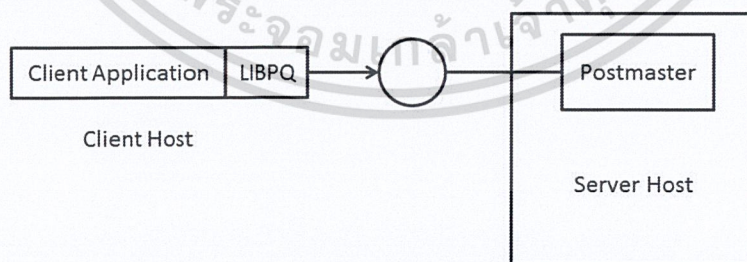
เซิร์ฟเวอร์โพรเซสจะทำหน้าที่รองรับการติดต่อจาก ไคลเอนต์โพรเซสและทำงานให้บริการแก่ ไคลเอนต์โพรเซสโดยทั่วไปเซิร์ฟเวอร์โพรเซสยังแบ่งตามลักษณะช่วงเวลาที่ไคลเอนต์โพรเซสติดต่อกับเซิร์ฟเวอร์โพรเซสซึ่งสามารถแบ่งออกเป็น 2 ประเภท คือ

- การติดต่อแบบเซิร์ฟเวอร์ซ้ำ (Iterative Server) ใช้สำหรับที่สามารถประมาณช่วงเวลาได้ ตัวอย่างเช่น เอชทีทีพีโพรเซส (http process) ในเว็บเซิร์ฟเวอร์ซึ่งให้บริการช่วงสั้นๆ ในการส่งข้อมูล เว็บเพจให้แก่ เว็บเบราว์เซอร์
- การติดต่อแบบเรียกเซิร์ฟเวอร์พร้อมกัน (Concurrent Server) ใช้กับระบบที่ไม่สามารถประมาณช่วงเวลานั้นได้ ไคลเอนต์อาจติดต่อเป็นช่วงเวลาสั้น ๆ หรือนานมากในการขอบริการ ซึ่งระบบจัดการฐานข้อมูลทำงานในลักษณะนี้

โพสเกรสเอสคิวแอล ทำงานภายใต้โมเดลของสถาปัตยกรรมแบบ เซิร์ฟเวอร์โพรเซส โดย เซิร์ฟเวอร์โพรเซส ทำงานในลักษณะ เซิร์ฟเวอร์พร้อมกันซึ่งมีขั้นตอนดังต่อไปนี้

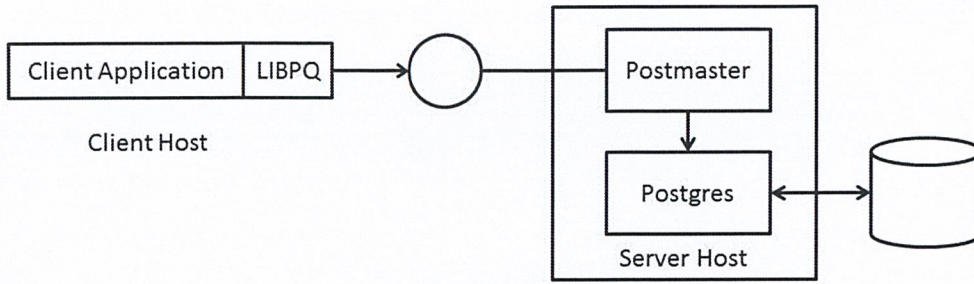
เซิร์ฟเวอร์โพรเซสที่ชื่อว่าโพสมาสเตอร์ถูกเรียกให้เริ่มต้นทำงานบนคอมพิวเตอร์ที่ทำหน้าที่เป็น เซิร์ฟเวอร์ทำการเปิดช่องการติดต่อและแจ้งกับระบบปฏิบัติการที่จะรับคำขอบริการจากไคลเอนต์ที่ พอร์ตที่กำหนด พอร์ตเริ่มต้นไว้ที่หมายเลข 5432

โพสมาสเตอร์จะเข้าสู่สภาวะหลับเพื่อรอการเรียกจาก ไคลเอนต์โพรเซสโดยที่ ไคลเอนต์โพรเซสจากเครื่องคอมพิวเตอร์ถูกข่าย ติดต่อขอรับบริการจากไลบรารีแอลไอบีพีคิว (LIBPQ) มาที่ พอร์ตที่กำหนดไว้ แสดงในรูปที่ 2.11



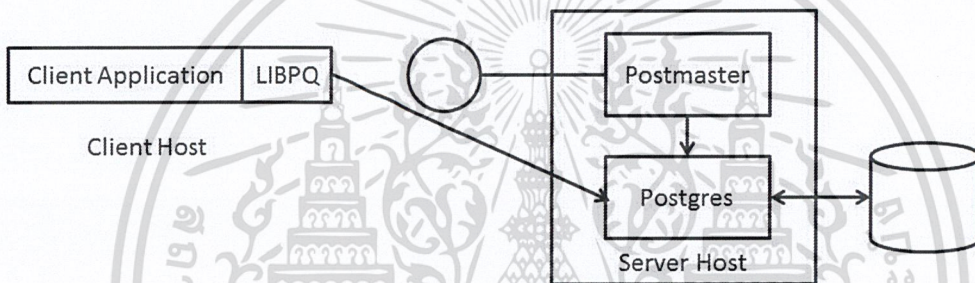
รูปที่ 2.11 แสดงไคลเอนต์โพรเซสขอรับบริการโพสมาสเตอร์

เมื่อโพสมาสเตอร์ได้รับการติดต่อจากไคลเอนต์โพรเซสแล้วจะทำการสร้าง โพรเซสลูก คือ โพสเกรสเพื่อให้บริการแทนตัวเองดังแสดงดังรูปที่ 2.12



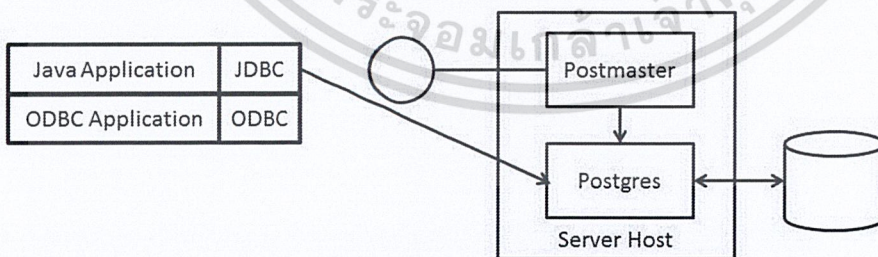
รูปที่ 2.12 แสดงโพสมาสเตอร์สร้างโพสเกรสโพรเซส

หลังจากนั้น โพสมาสเตอร์จะรอรับการขอบริการจากไคลเอนต์โพรเซสอื่นต่อไป ในขณะที่ โพสเกรสโพรเซสเมื่อให้บริการจนจบ ก็จะปิดตัวเองดังแสดงดังรูปที่ 2.13



รูปที่ 2.13 แสดงโพสเกรสโพรเซสให้บริการแก่ไคลเอนต์โพรเซส

โพสเกรสเอสคิวแอลทำงานในลักษณะฟรี ยูเซอร์ โพรเซส (per – user process) โดยทั้ง โพสมาสเตอร์และ โพสเกรส โพรเซสทำงานด้วย ยูเซอร์ ใด่ของ โพสเกรส ซูเปอร์ ยูเซอร์



รูปที่ 2.14 แสดงการขอบริการเซิร์ฟเวอร์ผ่านอินเตอร์เฟสเจดีบีซี, โอดีบีซี

แอลไอบีพีคิว เป็นไลบรารีมาตรฐานสำหรับ ไคลเอนต์โปรแกรมที่เขียนด้วย ซี โปรแกรมในการติดต่อกับ โพสเกรสเอสคิวแอล เซิร์ฟเวอร์ อย่างไรก็ตาม เราสามารถที่จะติดตั้งซอฟต์แวร์มาตรฐานอื่น เช่น โอดีบีซี (ODBC) หรือ เจดีบีซี (JDBC) ดังแสดงในรูปที่ 2.14 เพื่อให้เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่ขึ้นด้านการค้า

โคลเอนต์โปรแกรมต่าง ๆ ที่ใช้อินเตอร์เฟซมาตรฐานเหล่านี้สามารถติดต่อกับโพสต์เกรสเอสคิวแอลได้เช่นกัน

## 2.7 โพลจิส (PostGIS)

โพลจิสเป็นสิ่งที่ถูกพัฒนาขึ้นมาโดยบริษัทรีแฟกชัน รีเซิร์ช(Refractions Research) ซึ่งเป็นโครงการวิจัยทางเทคโนโลยีด้านฐานข้อมูลเชิงพื้นที่ที่รีแฟกชัน คือระบบสารสนเทศภูมิศาสตร์ (GIS) ซึ่งฐานข้อมูลนี้จะมีคุณสมบัติในการรวบรวมข้อมูล และพัฒนาซอฟต์แวร์ของลูกค้าโดยจะมีแผนการสนับสนุน และพัฒนาโพลจิสเพื่อให้รองรับลำดับของความสำคัญในบทบาทหน้าที่ของระบบสารสนเทศภูมิศาสตร์ ซึ่งจะประกอบไปด้วยโอเพ่นจิสเต็มรูปแบบ ซึ่งสิ่งที่สร้างขึ้นนี้จะ เป็นลักษณะภูมิประเทศในชั้นสูง เช่น บริเวณที่ปกคลุม , พื้นผิว , เครื่องข่าย ที่หน้าจอของผู้ใช้จะมี เครื่องมือที่ใช้ติดต่อสื่อสารสำหรับแสดงผลและแก้ไขข้อมูลทางสารสนเทศภูมิศาสตร์ และ เครื่องมือในการเข้าถึงเว็บชั้นพื้นฐาน

โพลจิสเป็นการเพิ่มเติมในส่วนของฐานข้อมูลเชิงวัตถุสัมพันธ์ (Object-relational database system) ของโพสต์เกรสเอสคิวแอลให้มีการรองรับวัตถุทางระบบสารสนเทศภูมิศาสตร์ (GIS object) เข้ามาเก็บไว้ในฐานข้อมูล มีการสนับสนุนจิส อินเด็กซ์ (Gist index) และ อาร์-ทรี อินเด็กซ์ (R-tree index) ซึ่งเป็นวิธีการค้นข้อมูลแบบตัวชี้ (Indexing) ที่ใช้ในฐานข้อมูลเชิงพื้นที่ สำหรับฐานข้อมูลที่มีขนาดใหญ่ ซึ่งโพลจิสมีการกำหนดการใช้งาน โดย โอเพ่นจิสที่เป็นลักษณะ พื้นฐานของ เอสเอฟเอสคิวแอล (SFSQL)

ดังที่กล่าวมาแล้วว่า โพลจิสมีการกำหนดการใช้งานที่เป็นลักษณะพื้นฐานโดย โอจีซี (OpenGIS Consortium: OGC) ซึ่งเป็นสถาบันที่ศึกษาเพื่อการสร้างอินเทอร์เน็ตเฟชที่ทำโปรแกรมประยุกต์ซอฟต์แวร์ ให้มีการใช้งานได้กับระบบสารสนเทศภูมิศาสตร์ (GIS)

โอเพ่นจิสเป็นสิ่งที่กำหนดความชัดเจนในการเข้าถึงข้อมูลทางด้านธรณีที่แตกต่างกัน และประมวลผลทางธรณีของแหล่งทรัพยากรในสภาพแวดล้อมที่เป็นเครือข่าย

### 2.7.1 ชนิดข้อมูลของโพลจิสที่นำมาใช้งาน

ชนิดเรขาคณิต (geometry type) เป็นลักษณะพื้นฐานที่กำหนดโดยโอจีซี โดยจะเก็บ ข้อมูลอยู่ในรูปแบบไบนารี (Well - Know Binary :WKB) โดยส่วนของชนิดที่นำมาใช้งานคือจุด (POINT) , เส้น (LINESTRING)

จุด (POINT) จะมีการเก็บข้อมูลในลักษณะหลายมิติ แต่ที่นำมาใช้งานคือการเก็บข้อมูล ใน 2 มิติ นำมาใช้งานในส่วนของการเก็บพิกัดข้อมูล มีรูปแบบดังนี้

POINT(0 0)

เส้น (LINESTRING) เป็นการเก็บข้อมูลของจุด (POINT) หลายๆจุด (POINT) นำมาใช้  
งานในส่วนของการเก็บเส้นทาง ถนน มีรูปแบบดังนี้

LINESTRING(0 0,1 1,1 2)

## 2.7.2 ฟังก์ชันของโพลีจิสที่นำมาใช้งาน

ตารางที่ 2.2 แสดงฟังก์ชันของโพลีจิสที่นำมาใช้งาน

ชื่อฟังก์ชัน	ลักษณะฟังก์ชัน
ST_AsText(geometry) : String	สกัดข้อมูลจากข้อมูลไบนารีให้อยู่ในรูปแบบข้อมูลข้อความ
ST_GeomFromText(String) :Geometry	สกัดข้อมูลจากข้อมูลข้อความให้อยู่ในรูปแบบข้อมูลไบนารี
ST_Length_Spheroid(geometry a_linestring, spheroid a_spheroid)	คำนวณความยาวของข้อมูลชนิดเส้น
ST_NumPoints(the_geom)	นับจำนวนข้อมูลชนิดจุด
ST_X(geometry a_point)	คืนค่า พิกัด ตัวแรก ของข้อมูลชนิดจุด
ST_Y(geometry a_point)	คืนค่าข้อมูลตัวที่สองของข้อมูลชนิดจุด
ST_Distance(geometry g1, geometry g2)	คำนวณระยะห่างระหว่างข้อมูลชนิดเรขาคณิตสองตัว

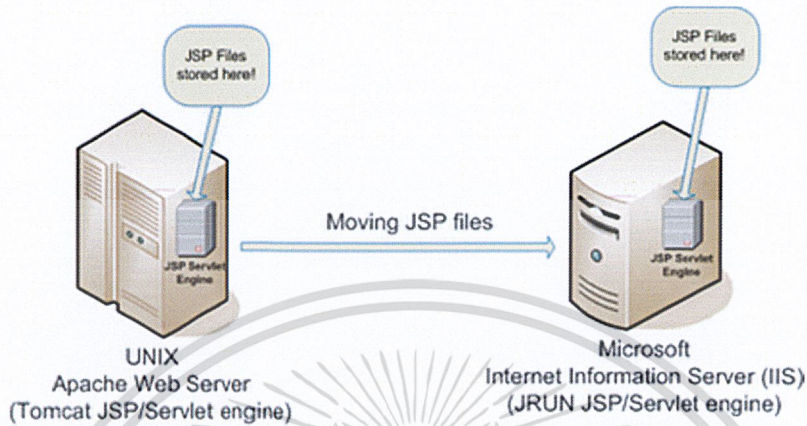
## 2.8 เจเอสพี (JSP)

เจเอสพี (Java Server Pages: JSP) เป็นเทคโนโลยี จาวา ที่เปิดช่องทางให้ผู้พัฒนาซอฟต์แวร์สร้างเอชทีเอ็มแอล, เอกซ์เอ็มแอล หรือไฟล์เอกสารในประเภทนี้ ตามความต้องการของเครื่องลูกข่ายร้องขอ ซึ่งเทคโนโลยีนี้ เปิดให้ใช้ ภาษาจาวา ในการสร้างและกรทำการใดๆ เพื่อให้หน้าเพจธรรมดา กลายเป็น หน้าเพจที่สามารถตอบสนองได้ โดยเจเอสพีจะถูก โปรแกรมเว็บเซิร์ฟเวอร์ที่สนับสนุนเจเอสพี เช่น อะแพชชีทอมแคต (apache Tomcat) แปลให้เป็น จาวา คลาส ที่เรียกว่า เซิร์ฟเล็ต (Servlet) ซึ่งพร้อมที่จะประมวลผลด้วย จาวา และแสดงผลออกเป็น เอชทีเอ็มแอล

เจเอสพีเป็นสคริปต์ อีกภาษาหนึ่ง ซึ่งเป็นทางเลือกที่น่าสนใจสำหรับการพัฒนาเว็บแอปพลิเคชัน จุดเด่นที่สำคัญของเจเอสพีอยู่ที่การใช้ภาษาจาวาซึ่งเป็นเชิงโอโอพี (Object-Oriented

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Programming : OOP) ที่ช่วยสามารถพัฒนาแอปพลิเคชันขนาดใหญ่และซับซ้อนได้อย่างสะดวกและรวดเร็ว การย้ายไฟล์เจเอสพีจากระบบยูนิกซ์ไประบบวินโดวส์แสดงดังรูปที่ 2.15



รูปที่ 2.15 แสดงการย้ายไฟล์เจเอสพีจากระบบยูนิกซ์ไประบบวินโดวส์

### 2.8.1 ข้อดีของเจเอสพี

การพัฒนาเว็บแอปพลิเคชันด้วยเจเอสพี มีข้อดีต่างๆ มากมาย ดังนี้

- ทำงานโดยไม่ยึดติดแพลตฟอร์มใด ๆ เจเอสพีได้สืบทอดคุณสมบัติของจาวาอย่างเต็มที่ โดยสามารถทำงานได้ในทุกแพลตฟอร์ม ไม่ว่าจะเป็นวินโดวส์, ลินุกซ์, แมคโอเอส
- ใช้งานจาวา เอพีไอ ได้หลากหลาย ซึ่งจาวาเอพีไอคือกลุ่มของคลาสที่ช่วยอำนวยความสะดวกในการใช้งาน ไม่ว่าจะเป็น การติดต่อกับฐานข้อมูล, การรับส่งอีเมล เป็นต้น
- นำคอมโพเนนต์กลับมาใช้ได้ อีก ไม่ต้องเสียเวลาสร้างใหม่ ดังนั้นเราจึงไม่ต้องเสียเวลาเขียนสคริปต์ใหม่เพื่อทำงานครั้งต่อไป จึงช่วยให้นักพัฒนาสามารถสร้างสรรค์ผลงานได้เร็วขึ้น
- มีความยืดหยุ่นในการใช้งาน คือเราสามารถกำหนดแท็กส์ใหม่ขึ้นมาใช้งานได้ รวมทั้งยังสามารถนำไปใช้งานร่วมกับเอ็็กเอ็มแอลได้เป็นอย่างดี
- ความปลอดภัยสูง เจเอสพีมีระบบจัดการข้อผิดพลาดต่าง ๆ ไม่ว่าจะเป็นข้อผิดพลาดที่เกิดขึ้นระหว่างการเขียนโปรแกรมหรือข้อผิดพลาดที่เกิดขึ้นระหว่างรันทามลั้วสามารถตรวจสอบข้อผิดพลาดได้ทันทีและตรงจุด

## 2.9 จาวาสคริปต์ (JavaScript)

จาวาสคริปต์ (JavaScript) เป็นภาษาโปรแกรม ประเภทหนึ่ง ที่เรียกกันว่า สคริปต์ (script) ซึ่งมีวิธีการทำงานในลักษณะแปลความและดำเนินงานไปที่ละคำสั่ง เพื่อที่จะช่วยให้เว็บเพจสามารถแสดงเนื้อหา ที่มีการเปลี่ยนแปลง ไปได้ ตามเงื่อนไขหรือสภาพแวดล้อมต่าง ๆ กัน หรือสามารถโต้ตอบกับผู้ชมได้มากขึ้น ทั้งนี้เพราะภาษาเอชทีเอ็มแอล (HTML) แต่เดิมนั้น เหมาะสำหรับใช้แสดงเอกสาร ที่มีเนื้อหาคงที่แน่นอน และไม่มีลูกเล่นอะไรมากมายนัก

เนื่องจากจาวาสคริปต์ ช่วยให้ผู้พัฒนา สามารถสร้างเว็บเพจได้ตรงกับความต้องการ และมีความน่าสนใจมากขึ้น ประกอบกับเป็นภาษาเปิด ที่ใครก็สามารถนำไปใช้ได้ ดังนั้นจึงได้รับความนิยมเป็นอย่างสูง มีการใช้งานอย่างกว้างขวางรวมทั้งได้ถูกกำหนดให้เป็นมาตรฐานโดยอีซีเอ็มเอ (ECMA) ซึ่งเราจะพบว่าปัจจุบัน จะหาเว็บเพจที่ไม่ใช้จาวาสคริปต์เลยนั้น ได้ยากเต็มที

การทำงานของจาวาสคริปต์ (JavaScript) จะต้องมีการแปลความคำสั่ง ซึ่งขั้นตอนนี้จะถูกจัดการโดยบราวเซอร์ ดังนั้นจาวาสคริปต์ (JavaScript) จึงสามารถทำงานได้ เฉพาะบนบราวเซอร์ที่สนับสนุน ซึ่งปัจจุบันบราวเซอร์เกือบทั้งหมดก็สนับสนุนจาวาสคริปต์ (JavaScript) แล้ว อย่างไรก็ตาม สิ่งที่ต้องระวังคือจาวาสคริปต์ (JavaScript) มีการพัฒนาเป็นเวอร์ชันใหม่ๆ ออกมาด้วย (ปัจจุบันคือรุ่น 1.5) ดังนั้น ถ้านำโค้ดของเวอร์ชันใหม่ ไปรันบนบราวเซอร์รุ่นเก่าที่ยังไม่สนับสนุน ก็อาจจะทำให้เกิดปัญหาได้

การทำงานของจาวาสคริปต์ เกิดขึ้นบนบราวเซอร์ดังนั้น ไม่ว่าคุณจะใช้เซิร์ฟเวอร์อะไร หรือที่ไหน ก็ยังคงสามารถใช้จาวาสคริปต์ ในเว็บเพจได้ ต่างกับภาษาสคริปต์ อื่น เช่น พีเอชพี (PHP) หรือเอเอสพี (ASP) ซึ่งต้องแปลความและทำงานที่ตัวเครื่องเซิร์ฟเวอร์ ดังนั้นจึงต้องใช้บนเซิร์ฟเวอร์ ที่สนับสนุนภาษาเหล่านี้เท่านั้น อย่างไรก็ตาม จากลักษณะดังกล่าวก็ทำให้จาวาสคริปต์ มีข้อจำกัด คือไม่สามารถรับและส่งข้อมูลต่างๆ กับเซิร์ฟเวอร์โดยตรง เช่น การอ่านไฟล์จากเซิร์ฟเวอร์ เพื่อนำมาแสดงบนเว็บเพจ หรือรับข้อมูลจากผู้ชม เพื่อนำไปเก็บบนเซิร์ฟเวอร์ เป็นต้น ดังนั้นงานลักษณะนี้ จึงยังคงต้องอาศัยภาษาเซิร์ฟเวอร์-ไซด์ สคริปต์ (server-side script) อยู่

การทำงานของจาวาสคริปต์ (JavaScript) จะมีประสิทธิภาพมาก ถ้ามั่นสามารถดัดแปลงคุณสมบัติ ขององค์ประกอบต่างๆ บนเว็บเพจ และสามารถรับรู้เหตุการณ์ ที่ผู้ชมเว็บเพจโต้ตอบกับองค์ประกอบเหล่านั้นได้ ดังนั้นจากภาษาเอชทีเอ็มแอลเดิม ที่มีลักษณะคงที่ (static) ในเอชทีเอ็มแอลเวอร์ชันใหม่ ๆ จึงได้มีการพัฒนาให้มีคุณสมบัติบางอย่างเพิ่มขึ้น และมีลักษณะเป็นออบเจ็กต์ (object) มากขึ้น การทำงานร่วมกันระหว่างคุณสมบัติใหม่ของเอชทีเอ็มแอลร่วมกับจาวาสคริปต์นี้เอง ทำให้เกิดเป็นสิ่งที่เรียกว่าไดนามิก เอชทีเอ็มแอล (Dynamic HTML) คือภาษาเอชทีเอ็มแอลที่สามารถใช้สร้างเว็บเพจที่มีลักษณะพลวัตได้นั่นเอง

นอกจากนี้ อีกองค์ประกอบหนึ่งที่เกี่ยวข้อง ก็คือชีเอสเอส (Cascading Style Sheet : CSS) ซึ่งเป็นภาษาที่ช่วยให้เราควบคุมรูปแบบขององค์ประกอบต่างๆบนเว็บเพจได้อย่างมีประสิทธิภาพมากกว่าคำสั่งหรือแท็กปกติของเอชทีเอ็มแอลเนื่องจากจาวาสคริปต์ สามารถดัดแปลงคุณสมบัติของชีเอสเอสได้เช่นกัน ดังนั้นมันจึงช่วยให้เราควบคุมเว็บเพจ ได้อย่างมีประสิทธิภาพมาก

## 2.10 ทราฟฟีเอพีไอ (TraffyAPI)

ทราฟฟีคือ ระบบรายงานสภาพจราจร ที่มีจุดประสงค์หลักคือแสดงข้อมูลสภาพจราจร เพื่อให้ผู้โดยสารสามารถที่จะวางแผนการเดินทางล่วงหน้าได้ และรับทราบข้อมูลสภาพจราจร ระบบทราฟฟีได้ข้อมูลสภาพจราจรจากแหล่งข้อมูลหลายแห่ง เช่น ป้ายจราจรอัจฉริยะของกรุงเทพมหานคร, ข้อมูลจากกล้องวงจรปิด (CCTV), ข้อมูลจากตำรวจทางหลวง, ข้อมูลจากตำรวจจราจร และอาสาสมัครบนท้องถนนที่ต้องแบ่งปันข้อมูล ทำให้การประมวลผลและรายงานเป็นไปอย่างถูกต้องและแม่นยำ ปัจจุบันทราฟฟีมีการเปิดให้เข้าใช้เอพีไอต่าง ๆ เพื่อให้นักพัฒนานำไปต่อยอดเป็นแอปพลิเคชันใหม่ ๆ

ระบบทราฟฟีถูกพัฒนาขึ้น โดยโครงการรายงานประเมินสภาพจราจร ห้องปฏิบัติการเทคโนโลยีเครือข่าย โปรแกรมขนส่งและจราจรอัจฉริยะ ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ (สวทช) ซึ่งเป็นหน่วยงานของรัฐ เพื่อเผยแพร่ให้คนทั่วไปเข้าใช้ได้ฟรี ระบบทราฟฟีช่วยสร้างเครือข่ายกลุ่มคนที่สนใจร่วมพัฒนาร่วมแบ่งปันข้อมูลจราจรซึ่งจะเป็นประโยชน์ ในการบรรเทาปัญหาจราจรที่เป็นอยู่ในปัจจุบัน ระบบทราฟฟีอยู่ในช่วงวิจัยและพัฒนาให้มีฟังก์ชันการทำงานที่มากขึ้นพร้อม ๆ กับเผยแพร่ให้คนทั่วไปได้เข้าไปใช้ประโยชน์ หน้าหลักของเว็บไซต์ทราฟฟีเอพีไอ แสดงดังรูปที่ 2.16



รูปที่ 2.16 หน้าหลักของเว็บไซต์ทราฟฟีเอพีไอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.10.1 เอพีไอของกราฟฟี่

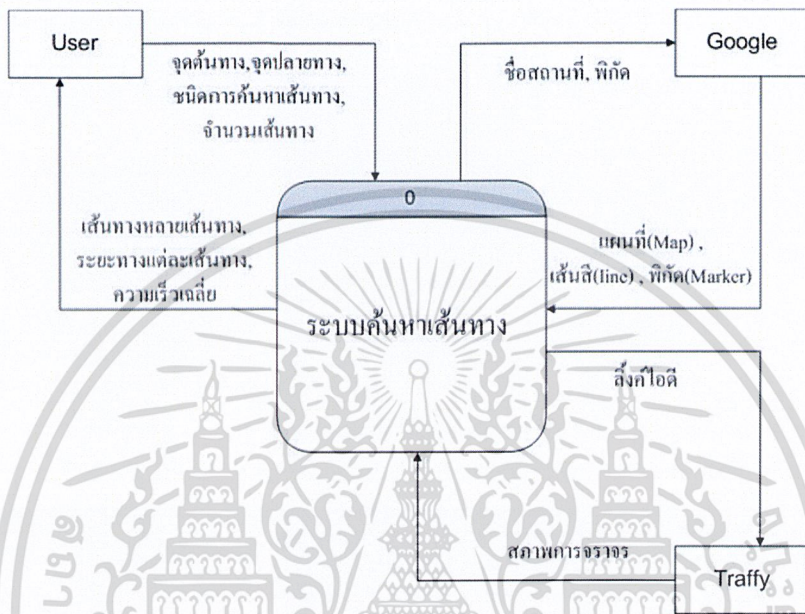
ตารางที่ 2.3 แสดงฟังก์ชันของกราฟฟี่ที่นำมาใช้งาน

ชื่อเอพีไอ	ลักษณะเอพีไอ
getCCTV	ให้ข้อมูลภาพจาก กล้องซีซีทีวี (CCTV) มีข้อมูลตำแหน่ง รูปภาพ และวัน เดือน ปี ที่บันทึกภาพ
getCCTVimg	เอพีไอสำหรับการดึงภาพ ซีซีทีวี ซึ่งสามารถเลือกภาพซีซีทีวี ย้อนหลังได้ 5 ภาพ
getVMS	แผนที่อย่างง่ายบอกสภาพการจราจร
getvmsimg	เอพีไอสำหรับการดึงภาพป้ายจราจรอัจฉริยะ (VMS) ที่ต้องการทั่วกรุงเทพฯ 40 ป้าย
getIncident	ให้ข้อมูลข่าวเกี่ยวกับการจราจร เช่น อุบัติเหตุ ระเบิด ฝนตก น้ำท่วม จากหลายๆ แหล่งข้อมูล เช่น บกจร. ทวิตเตอร์ สามารถเรียกดูข้อมูลย้อนหลังได้ 1 วัน นับจากเวลาปัจจุบัน ระบบจะทำการอัปเดต ทุกๆ 5 นาที
getTrafficCongestion	เป็นคำสั่งสำหรับการแสดงข้อมูลจราจร โดยจะประกอบด้วย
getRainForecast	บริการเอพีไอ สำหรับพยากรณ์ฝนล่วงหน้า ในพื้นที่กรุงเทพฯและปริมณฑล ใน 10, 30, และ 60 นาทีล่วงหน้า ว่าจะมีฝนตกหรือไม่ และมีปริมาณน้ำฝนเท่าใด สามารถ สอบถามในรูปแบบละติจูด ลองจิจูด, เขตพื้นที่, ถนน แล้วรายงานผลในรูปแบบเอกซ์เอ็มแอล
getLinkInfo	ให้ข้อมูลรายละเอียดเกี่ยวกับถนน
getLinkALLInfo	ให้ ข้อมูลรายละเอียดของถนนทั้งหมด
postGPSData	เป็นคำสั่งสำหรับให้ส่งข้อมูลจีพีเอสเข้าฐานข้อมูลกราฟฟี่

### บทที่ 3

## การออกแบบและพัฒนาระบบ

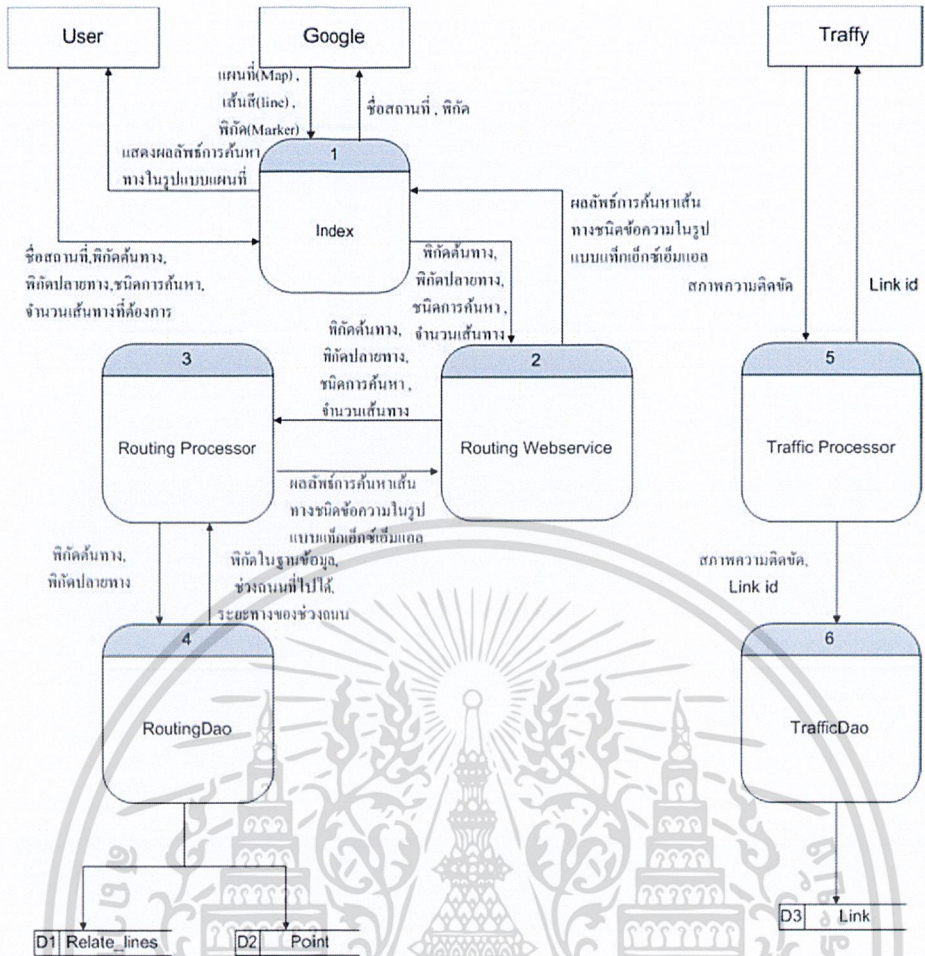
### 3.1 แผนภาพกระแสข้อมูลของระบบ



รูปที่ 3.1 แผนภาพกระแสข้อมูลแสดงภาพรวมของระบบ

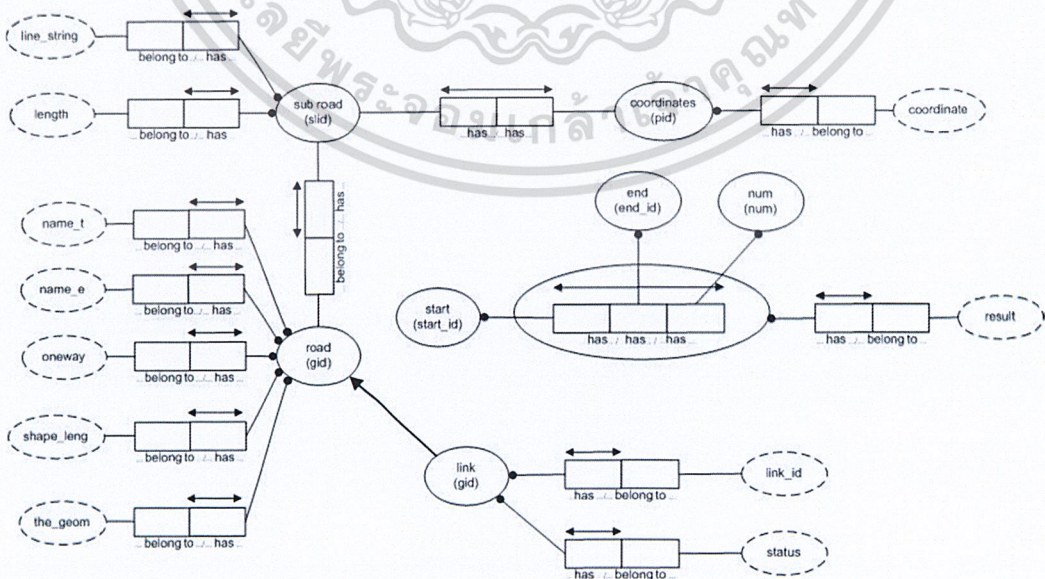
รูปที่ 3.1 แสดงแผนภาพกระแสข้อมูลในส่วนภาพรวมของระบบค้นหาเส้นทางอ้างอิงจากสภาพการจราจร โดยจะมี 2 ส่วนที่เข้ามาดำเนินกิจกรรมต่างๆภายในระบบคือผู้ใช้และ Traffy ในส่วนของผู้ใช้ จะกำหนดจุดต้นทาง, จุดปลายทาง, เลือกชนิดการค้นหาเส้นทาง และจำนวนเส้นทางที่ต้องการ ระบบจะทำการส่งข้อมูลของเส้นทางกลับไปให้ผู้ใช้ ได้เลือกเส้นทางที่ต้องการ ส่วนของระบบ Traffy ระบบจะส่งสิ่งกีดขวาง ไปที่ระบบ Traffy เพื่อเรียกสภาพการจราจร

รูปที่ 3.2 แสดงแผนภาพกระแสข้อมูลระดับที่ 1 ของระบบ แสดงกิจกรรมต่าง ๆ ของระบบ และแต่ละกิจกรรมมีใครมาเกี่ยวข้องบ้าง โดยในส่วนแรกผู้ใช้จะป้อนข้อมูลที่ใช้ในการค้นหา ได้แก่ จุดเริ่มต้น, จุดปลายทาง, ชนิดการค้นหาและจำนวนเส้นทางไว้ที่หน้าแสดงผล หน้าแสดงผลก็จะส่งข้อมูลมาที่เว็บเซอร์วิส (Routing Web service) เว็บเซอร์วิสก็จะส่งข้อมูลไปที่เราต์ติ้งโปรเซสเซอร์ (Routing Processor) เราต์ติ้งโปรเซสเซอร์ทำการค้นหาเส้นทาง ด้วยการดึงข้อมูลเส้นทางและสภาพการจราจรจากเราต์ติ้งดีเอโอ (RoutingDao) จากนั้นก็จะส่งผลลัพธ์การค้นหาเส้นทางกลับไปเว็บเซอร์วิส เว็บเซอร์วิสจะส่งข้อมูลกลับไปให้ส่วนหน้าแสดงผล



รูปที่ 3.2 แผนภาพกระแสน้ำข้อมูลระดับที่ 1 ของระบบ

### 3.2 แบบจำลองฐานข้อมูลในแอมของระบบค้นหาเส้นทางอ้างอิงจากสภาพการจราจร



รูปที่ 3.3 ฟังในแอมแสดงความสัมพันธ์ของข้อมูลระบบค้นหาเส้นทางอ้างอิงจากสภาพการจราจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3 ตารางฐานข้อมูล

ตารางที่ 3.1 ตารางแสดงรายละเอียดถนน

ชื่อ	คีย์	ชนิดข้อมูล	ความหมาย	ตัวอย่าง	หมายเหตุ
gid	PK	integer	หมายเลขของถนน	123425	
name_t:		Character varying (70)	ชื่อถนน(ไทย)	ถนนดินแดง	
name_e		Character varying (70)	ชื่อถนน(อังกฤษ)	DIN DAENG RD.	
oneway		Character varying (2)	เดินรถทางเดียว	FT	ถ้าเป็น FT คือ ทิศทางจากต้นไป ปลายทาง TF จาก ปลายทางไปต้นทาง
shape_leng		numeric	ความยาวของถนน	25.4816142	หน่วยเป็นเมตร
the_geom		geometry	โบนี่ข้อมูลรูปร่าง ของถนน	LINestring(100.545 267318618 13.7634518866782,100 .545493647122 13.7635160007176)	

ตารางที่ 3.2 ตารางแสดงรายละเอียดของช่วงถนน

ชื่อ	คีย์	ชนิดข้อมูล	ความหมาย	ตัวอย่าง	หมายเหตุ
slid	PK	integer	หมายเลขของช่วง ถนน	1	
gid	FK	integer	ถนน	161868	
linestring		geometry	ข้อมูลของช่วงถนน	LINestring(100.545 267318618 13.7634518866782,100 .545493647122 13.7635160007176)	เก็บละติจูด,ลองจิจูด ของจุดต้นทางกับจุด ปลายทาง
length		double	ความยาวช่วงถนน	25.4816142	หน่วยเป็นเมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.3 ตารางแสดงละเอียดของพิกัด

ชื่อ	คีย์	ชนิดข้อมูล	ความหมาย	ตัวอย่าง	หมายเหตุ
pid	PK	integer	หมายเลขของพิกัด	1	
coordinate		geometry	พิกัด	POINT(100.54526731 8618 13.7634518866782)	เก็บละติจูด, ลองจิจูด ของพิกัด

ตารางที่ 3.4 ตารางแสดงข้อมูลระหว่างพิกัดกับช่วงถนน

ชื่อ	คีย์	ชนิดข้อมูล	ความหมาย	ตัวอย่าง	หมายเหตุ
pid	PK	integer	หมายเลขของพิกัด	1	
slid	FK	integer	หมายเลขของช่วงถนน	1	
nid		integer	หมายเลขลำดับของพิกัด	1	แสดงว่าพิกัดนั้นเป็นพิกัดลำดับที่เท่าไร ของช่วงถนนนั้น

ตารางที่ 3.5 ตารางแสดงละเอียดของลิงค์

ชื่อ	คีย์	ชนิดข้อมูล	ความหมาย	ตัวอย่าง	หมายเหตุ
gid	PK	integer	หมายเลขของถนน	1	
link_id		integer	หมายเลขลิงค์ไฮเวย์	118	
status		numeric	ความเร็วของถนน	80.0	หน่วยเป็นกิโลเมตร ต่อ ชั่วโมง

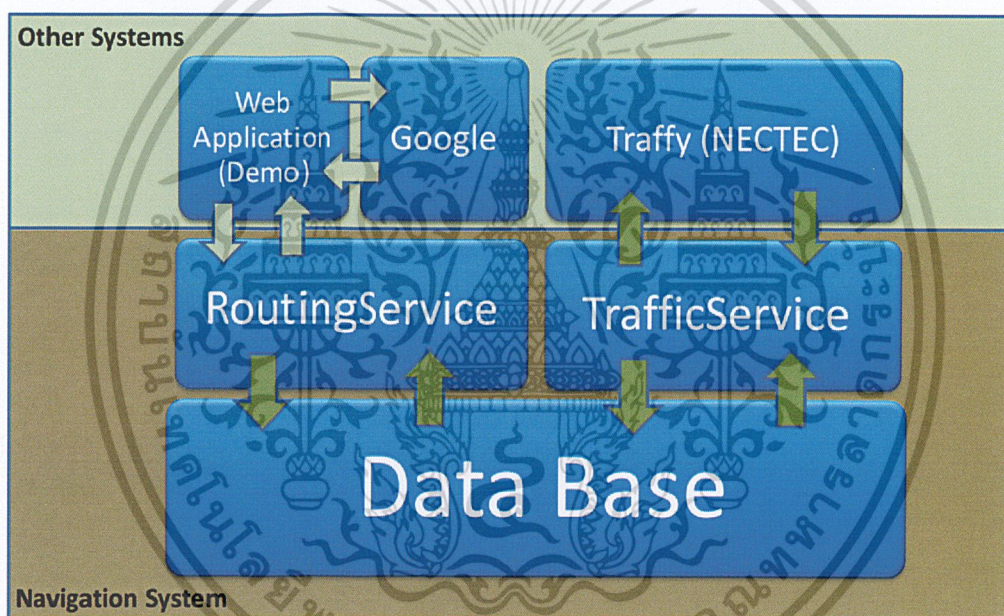
ตารางที่ 3.6 ตารางแสดงละเอียดของผลลัพธ์เส้นทาง

ชื่อ	คีย์	ชนิดข้อมูล	ความหมาย	ตัวอย่าง	หมายเหตุ
start_id	PK	integer	พิกัดต้นทาง	1356	เก็บเป็น pid
end_id		integer	พิกัดปลายทาง	852	เก็บเป็น pid
num		integer	เก็บจำนวนเส้นทาง	2	
result		text	เก็บเส้นทางที่เคยค้นหาไว้	"<Paths><Path wieght='623.1701450	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

				<pre> 648735'distance='623. 1701450648735'&gt; &lt;Vertexlat='13.76367 8083257377' lng='100.5409032876 5594' /&gt; ... /&gt;&lt;/Path&gt;&lt;/Paths&gt;" </pre>	
--	--	--	--	---	--

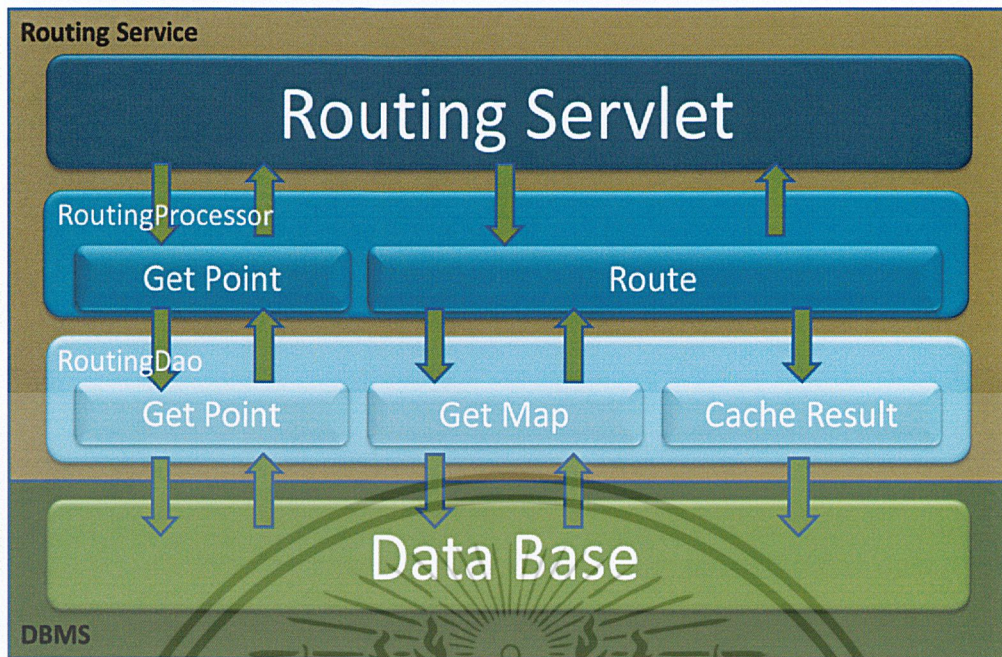
### 3.4 การออกแบบเว็บเซอร์วิส



รูปที่ 3.4 บล็อกไดอะแกรมการทำงานของระบบค้นหาเส้นทาง

รูปที่ 3.4 แสดงการทำงานของระบบค้นหาเส้นทาง เมื่อผู้ใช้ต้องการค้นหาเส้นทางโดยทำการป้อนชื่อสถานที่ หรือทำการปักหมุดสถานที่ ผ่านทางเว็บแอปพลิเคชัน เว็บแอปพลิเคชันจะไปติดต่อกับระบบกูเกิ้ล เพื่อหาพิกัดของสถานที่ หรือพิกัดที่ผู้ใช้ปักหมุด เมื่อผู้ใช้กดปุ่มค้นหาจะส่งการร้องขอไปที่เราท์ติ้งเซอร์วิส (Routing service) เราท์ติ้งเซอร์วิสจะทำการติดต่อฐานข้อมูลที่ได้รับมาจากเนคเทค และทำการค้นหาเส้นทางส่งกลับไปให้เว็บแอปพลิเคชัน ในระหว่างนั้นกราฟฟิคเซอร์วิส (Traffic service) จะทำงาน โดยไปติดต่อกับฐานข้อมูลเพื่อดึงข้อมูลลิงค์จากฐานข้อมูลและระบบกราฟฟี่ (Traffy) เพื่อร้องขอสภาพการจราจร โดยทำงานขนานกันไปกับเราท์ติ้งเซอร์วิส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

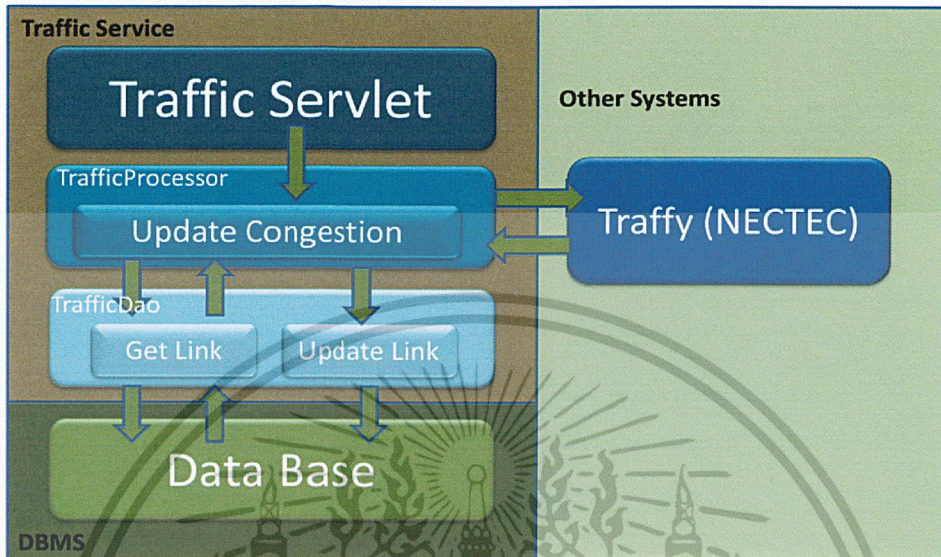


รูปที่ 3.5 บล็อก ไดอะแกรมการทำงานของเราต์ติ้งเซอร์วิส

รูปที่ 3.5 แสดงการทำงานของเราต์ติ้งเซอร์วิส (Routing Service) เมื่อมีการเรียกใช้ เราต์ติ้งเซอร์วิส ด้วยการร้องขอมาที่เราต์ติ้งเซิร์ฟเลต (Routing Servlet) ผ่านยูอาร์แอล เราต์ติ้งเซิร์ฟเลตจะเรียกใช้เราต์ติ้งโพรเซสเซอร์ (Routing processor) เพื่อหาพิกัดจากฐานข้อมูลด้วยการ ส่งพิกัดจากพารามิเตอร์ที่รับเข้ามาไปที่เราต์ติ้งโพรเซสเซอร์ จากนั้นเราต์ติ้งโพรเซสเซอร์ จะทำการเรียกใช้เราต์ติ้งดีเอโอ (RoutingDao) เพื่อค้นหาพิกัดที่ใกล้ที่สุดในฐานข้อมูลโดยเปรียบเทียบพิกัดที่รับเข้ามากับพิกัดในฐานข้อมูล เมื่อได้พิกัดจากฐานแล้ว เราต์ติ้งเซิร์ฟเลตจะทำการเรียกใช้ เราต์ติ้งโพรเซสเซอร์เพื่อทำการค้นหาเส้นทาง เราต์ติ้งโพรเซสเซอร์จะเรียกใช้เราต์ติ้งดีเอโอเพื่อดึงแผนที่จากฐานข้อมูล มาทำการค้นหาเส้นทาง โดยนำอัลกอริทึมเบลแมน ฟรูด มาประยุกต์ใช้ในการค้นหาเส้นทาง แล้วส่งเส้นทางที่ค้นหาได้กลับไปให้เราต์ติ้งเซิร์ฟเลตถ้าชนิดการค้นหาเส้นทางเป็นตามระยะทาง (Based on Distance) จะมีการเก็บเส้นทางที่ค้นหาได้ลงในฐานข้อมูล เพื่อให้ระบบทำงานได้เร็วขึ้นในครั้งต่อไป โดยถ้ามีร้องขอมาที่พิกัดเดิม และจำนวนเส้นทางเท่าเดิมระบบจะนำผลลัพธ์ที่เก็บไว้ส่งกลับไปให้

รูปที่ 3.6 แสดงการทำงานของทราฟฟิกเซอร์วิส (Traffic Service) ทราฟฟิกเซอร์วิสคือระบบภายในเว็บเซอร์วิสที่มีหน้าที่อัปเดตสภาพการจราจร จะทำงานเมื่อระบบต้องการอัปเดตสภาพการจราจร โดยการส่งการร้องขอไปที่ทราฟฟิกเซิร์ฟเลต (Traffic Servlet) ทราฟฟิกเซิร์ฟเลตจะเรียกใช้ทราฟฟิกโพรเซสเซอร์ (Traffic Processor) เพื่ออัปเดตสภาพการจราจร โดยลักษณะการทำงานของทราฟฟิกโพรเซสเซอร์ จะเรียกใช้ทราฟฟิกดีเอโอ (Traffic Dao) เพื่อดึงข้อมูลลิงค์จาก

ฐานข้อมูล ส่งไปที่ระบบทราฟฟี่ โดยเรียกใช้ทราฟฟี่เอพีไอ (TraffyAPI) คือgetTrafficCongestion แล้วประมวลผลไปอัปเดตสภาพการจราจร โดยการทำงานทั้งหมดนี้เป็นการทำงานของเทรด (Thread) ที่จะทำงานตลอด ในขณะที่เซิร์ฟเวอร์ยังทำงานอยู่



รูปที่3.6 บล็อก โค้ดโปรแกรมการทำงานของทราฟฟี่เซิร์ฟเวอร์

### 3.5 การออกแบบวิธีการค้นหาเส้นทางอ้างอิงสภาพจราจร

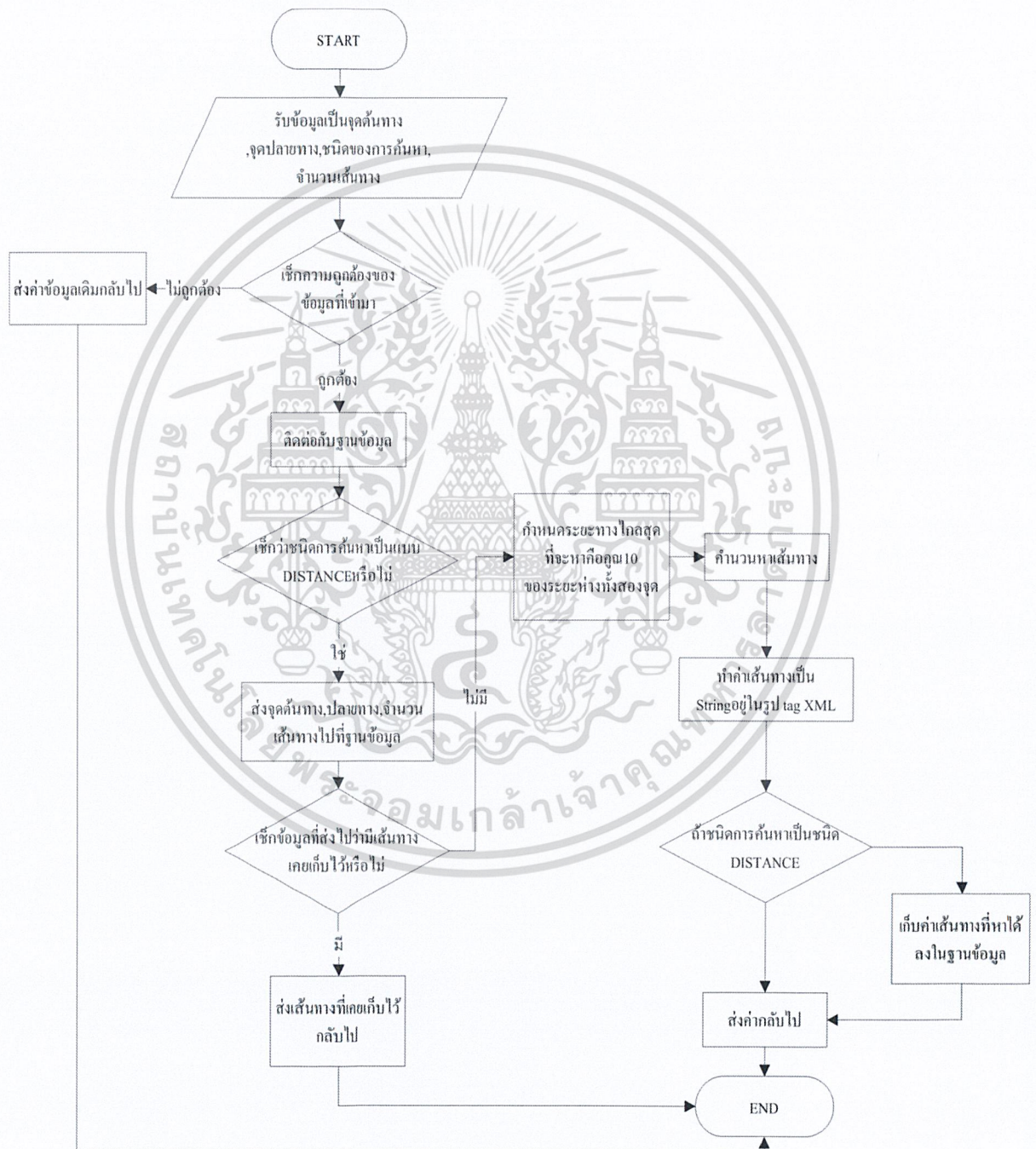
ในโครงการนี้จะใช้วิธีการของขั้นตอนการหาเส้นทางที่สั้นที่สุด (Shortest path Algorithm) โดยวิธีเบลแมน ฟอร์ด อัลกอริทึม (Bellman Ford Algorithm) เพื่อการวิเคราะห์หาเส้นทางที่สั้นที่สุด ตามน้ำหนักของเส้นทางให้ออกมาหลายเส้นทาง โดยจะมีการเปลี่ยนค่าของน้ำหนักของเส้นทางตามชนิดของการค้นหาเส้นทาง ชนิดของการค้นหาเส้นทางมี 2 ชนิด คือเลือกตามระยะทาง (Base on distance) และเลือกตามสภาพจราจร (Base on traffic) การออกแบบส่วนของโปรแกรมจำลองการค้นหาเส้นทางอ้างอิงสภาพจราจรจะประกอบไปด้วย 2 ส่วนหลัก ๆ คือ ส่วนของเว็บเซิร์ฟเวอร์การค้นหาเส้นทาง และส่วนของเว็บแอปพลิเคชันที่ทำขึ้นมาเพื่อแสดงผลลัพธ์จากเว็บเซิร์ฟเวอร์การค้นหาเส้นทาง

ในส่วนของการทำงานของเราที่ตั้งเว็บเซิร์ฟเวอร์สามารถแสดงได้ดังรูปที่ 3.7 ถ้ามีการร้องขอ (request) จากผู้ใช้ จะมีการรับการร้องขอและพารามิเตอร์ แล้วทำการตรวจสอบความถูกต้องของข้อมูลที่เข้ามาแล้วจะทำการติดต่อไปยังฐานข้อมูล ทำการตรวจสอบชนิดการค้นหาเส้นทางคือเลือกตามระยะทาง หรือเลือกตามสภาพจราจร แล้วทำการส่งข้อมูลจุดต้นทาง ปลายทาง และจำนวนเส้นทางไปยังฐานข้อมูลเพื่อค้นหาพิกัดในฐานข้อมูล แล้วทำการตรวจสอบจากข้อมูลที่ส่งไปว่ามีเส้นทางเก็บไว้หรือไม่ ถ้ามีจะทำการส่งค่าเส้นทางที่เคยคำนวณเก็บไว้แล้วให้ ถ้าไม่มีจะ

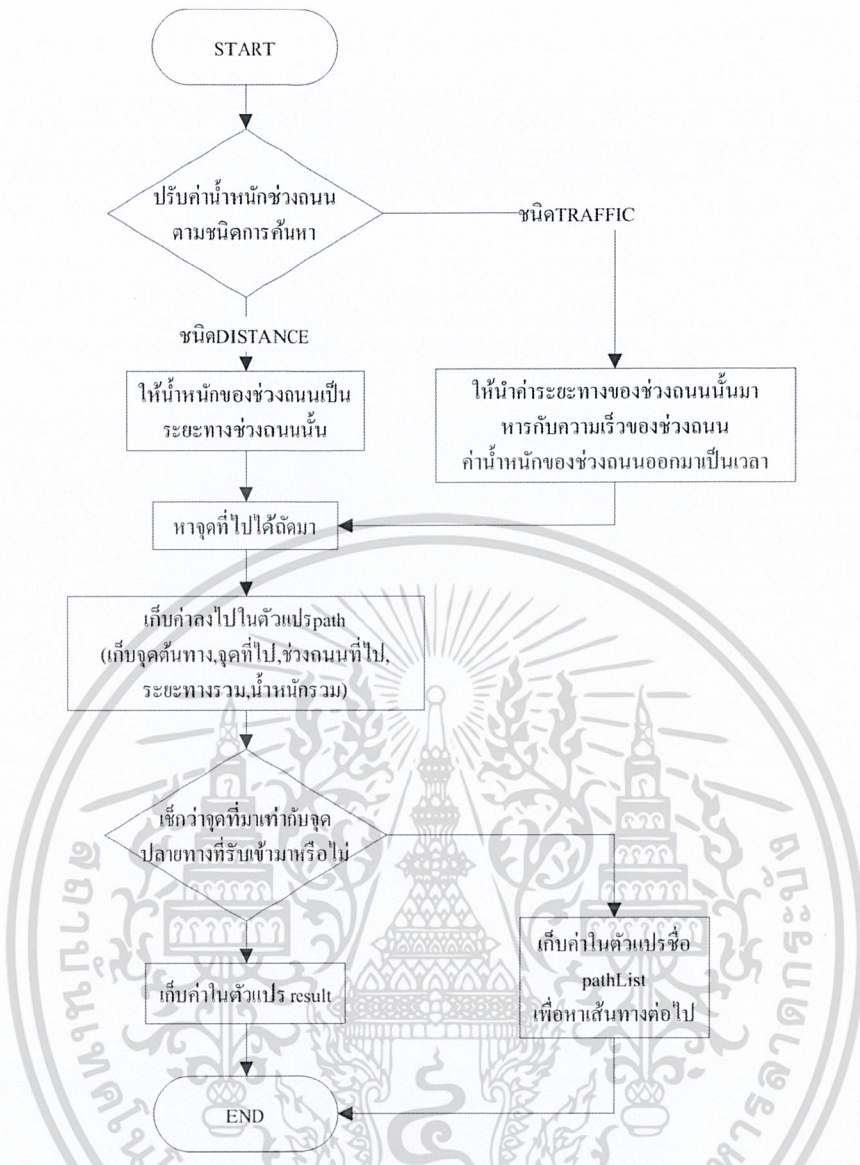
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการกำหนดระยะทางที่จะทำการค้นหาคือ ระยะห่างระหว่างจุดต้นทาง ปลายทาง คูณ 10 แล้วทำการค้นหาเส้นทางออกมา จากนั้นทำเส้นทางที่ค้นหาแล้วทั้งหมดให้เป็นข้อความ (String) อยู่ในรูปแท็กเอกซ์เอ็มแอล (tag XML) แล้วทำการตรวจสอบว่าถ้าการค้นหาเส้นทางเป็นชนิดเลือกตามระยะทาง จะทำการเก็บค่าเส้นทางที่ค้นหาได้ลงในฐานข้อมูลแล้วส่งค่ากลับไป ถ้าเป็นชนิดเลือกตามสภาพจราจร จะทำการส่งข้อมูลกลับไปเลย



รูปที่ 3.7 โฟว์ชาร์ตแสดงภาพรวมการทำงานของเร้าที่ตั้งเว็บเซอร์วิส



รูปที่ 3.8 โฟลว์ชาร์ตแสดงการทำงานของโมดูลเบลแมน ฟอร์ด อัลกอริทึม

รูปที่ 3.8 จะอธิบายการทำงานของโมดูลเบลแมน ฟอร์ด อัลกอริทึม ที่ทำการค้นหาเส้นทาง โดยมีการปรับน้ำหนักของช่วงถนนตามชนิดการค้นหาเส้นทาง ถ้าชนิดการค้นหาเส้นทางเป็นตามระยะทาง น้ำหนักของช่วงถนนจะเป็นระยะทาง ถ้าชนิดการค้นหาเส้นทางเป็นตามสภาพการจราจร น้ำหนักของช่วงถนนจะเป็นเวลา โดยนำค่าความเร็วมาทำการกับระยะทางของช่วงถนนนั้น แล้วทำการหาจุดที่ไปได้ต่อไป จากนั้นเก็บค่าลงในตัวแปร path แล้วทำการตรวจสอบว่าจุดที่ไปได้ใช่จุดปลายทางที่รับเข้ามาหรือไม่ ถ้าใช่จะทำการเก็บค่าลงในตัวแปร result ถ้าไม่ใช่จะทำการเก็บลงในตัวแปร pathList เพื่อนำไปวนลูปค้นหาเส้นทางในส่วนต่อไป

การเรียกใช้กราฟฟี่เอพีไอที่ <http://info.traffy.in.th/api/> จะทำการเรียกใช้ในส่วนของเอพีไอ `getTrafficCongestion` เป็นคำสั่งสำหรับการแสดงข้อมูลจราจรแสดงดังรูปที่ 3.9 โดยมีพารามิเตอร์ในการเรียกใช้ คือ

- `format` = รูปแบบของการแสดงผล (XML,CSV)
- `api` = ประเภทของเอพีไอที่ต้องการ (`getCL`)
- `key` = รหัสที่ได้รับจากการลงทะเบียน
- `appid` = ไอดี ที่ได้รับจากการลงทะเบียน
- `linkid` = สำหรับเรียกข้อมูลเฉพาะบาง link โดยให้ใส่เป็นรหัส link
- `q` = สำหรับเรียกข้อมูลทุกลิงค์
- `source` = สำหรับเรียกข้อมูลจราจรที่ต้องการเช่น BMA (default), police

Parameters :

Required \*

\*`format` = รูปแบบของการแสดงผล (XML,CSV)

Example: <http://api.traffy.in.th/apis/apitraffy.php?format=XML&api=getCL&q=all>

\*`api` = ประเภทของ api ที่ต้องการ (`getCL`)

Example: <http://api.traffy.in.th/apis/apitraffy.php?format=XML&api=getCL&q=all>

\* `key` = รหัสที่ได้รับจากการลงทะเบียน (ลงทะเบียน)

Example: [http://api.traffy.in.th/apis/apitraffy.php?format=XML&api=getCL&key=\(คีย์ที่ได้รับจากการลงทะเบียน\)&q=all](http://api.traffy.in.th/apis/apitraffy.php?format=XML&api=getCL&key=(คีย์ที่ได้รับจากการลงทะเบียน)&q=all)

\*`appid` = id ที่ได้รับจากการลงทะเบียน (ลงทะเบียน)

Example: [http://api.traffy.in.th/apis/apitraffy.php?format=XML&api=getCL&key=\(คีย์ที่ได้รับจากการลงทะเบียน\)&appid=\(id ที่ได้รับจากการลงทะเบียน\)&q=all](http://api.traffy.in.th/apis/apitraffy.php?format=XML&api=getCL&key=(คีย์ที่ได้รับจากการลงทะเบียน)&appid=(id ที่ได้รับจากการลงทะเบียน)&q=all)

`linkid` = สำหรับเรียกข้อมูลเฉพาะบาง link โดยให้ใส่เป็นรหัส link

Example: [http://api.traffy.in.th/apis/apitraffy.php?format=XML&api=getCL&key=\(คีย์ที่ได้รับจากการลงทะเบียน\)&appid=\(id ที่ได้รับจากการลงทะเบียน\)&linkid=101,102](http://api.traffy.in.th/apis/apitraffy.php?format=XML&api=getCL&key=(คีย์ที่ได้รับจากการลงทะเบียน)&appid=(id ที่ได้รับจากการลงทะเบียน)&linkid=101,102)

`q`= สำหรับเรียกข้อมูลทุก link (all)

Example: [http://api.traffy.in.th/apis/apitraffy.php?format=XML&api=getCL&key=\(คีย์ที่ได้รับจากการลงทะเบียน\)&appid=\(id ที่ได้รับจากการลงทะเบียน\)&q=all](http://api.traffy.in.th/apis/apitraffy.php?format=XML&api=getCL&key=(คีย์ที่ได้รับจากการลงทะเบียน)&appid=(id ที่ได้รับจากการลงทะเบียน)&q=all)

`source` = สำหรับเรียกข้อมูลจราจรที่ต้องการเช่น BMA (default), police

Example: [http://api.traffy.in.th/apis/apitraffy.php?format=XML&api=getCL&key=\(คีย์ที่ได้รับจากการลงทะเบียน\)&source=BMA](http://api.traffy.in.th/apis/apitraffy.php?format=XML&api=getCL&key=(คีย์ที่ได้รับจากการลงทะเบียน)&source=BMA)

### รูปที่ 3.9 ส่วนของพารามิเตอร์ในการเรียกใช้ กราฟฟี่เอพีไอ `getTrafficCongestion`

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่หรือใช้เพื่อการพาณิชย์

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พารามิเตอร์ที่ส่งออก (Output Parameter) แสดงดังรูปที่ 3.10 จะประกอบไปด้วย

- linkid = หมายเลขประจำ link ของข้อมูลสามารถรายละเอียดได้จาก GetLinkAllInfo
- Status = แสดงระดับความติดขัด โดยจะแบ่งเป็น 3 ระดับติดขัด (H) , หนาแน่น (M) และ คล่องตัว (L)
- lastupdate = เวลาที่ข้อมูลถูกรายงาน โดยมีรูปแบบเป็น [yyyy-mm-dd hh:mm:ss]

#### Output Parameter

linkid = หมายเลขประจำ link ของข้อมูลสามารถรายละเอียดได้จาก GetLinkAllInfo

Status = แสดงระดับความติดขัดโดยจะแบ่งเป็น 3 ระดับคือ H(ติดขัด) , M (หนาแน่น) และ L (คล่องตัว)

lastupdate = เวลาที่ข้อมูลถูกรายงาน โดยมี format เป็น [yyyy-mm-dd hh:mm:ss]

รูปที่ 3.10 ส่วนของพารามิเตอร์ที่ส่งออกจากการเรียกใช้กราฟพีเอชไอ getTrafficCongestion

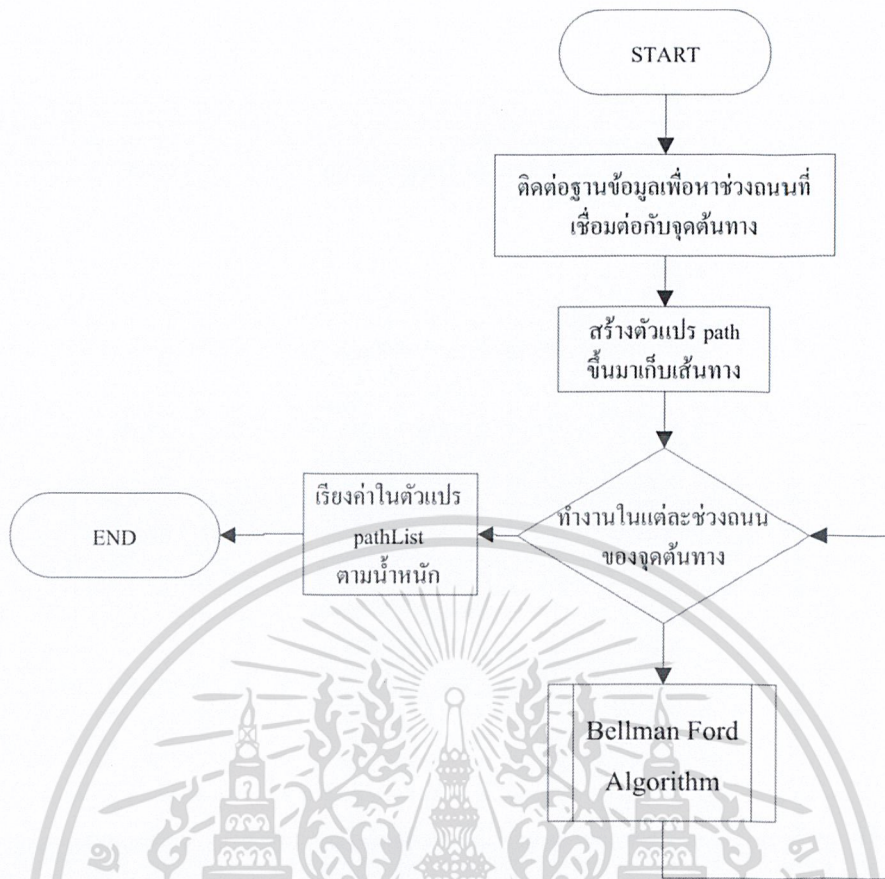
ตัวอย่างการแสดงผลการเรียกพีเอชไอ

```
<trafficmessage>
  <linkid>480</linkid>
  <datetime>2011-01-17 08:37:53</datetime>
  <status>L</status>
</trafficmessage>
<trafficmessage>
  <linkid>1059</linkid>
  <datetime>2011-01-17 08:37:53</datetime>
  <status>L</status>
</trafficmessage>
```

หลังจากที่ได้ค่าสถิติของช่วงถนนมาแล้ว จะนำมากำหนดความเร็วของช่วงถนน คือ ติดขัด เท่ากับ 20 กิโลเมตร/ชั่วโมง, หนาแน่น เท่ากับ 40 กิโลเมตร/ชั่วโมง และ คล่องตัว เท่ากับ

80 กิโลเมตร/ชั่วโมง

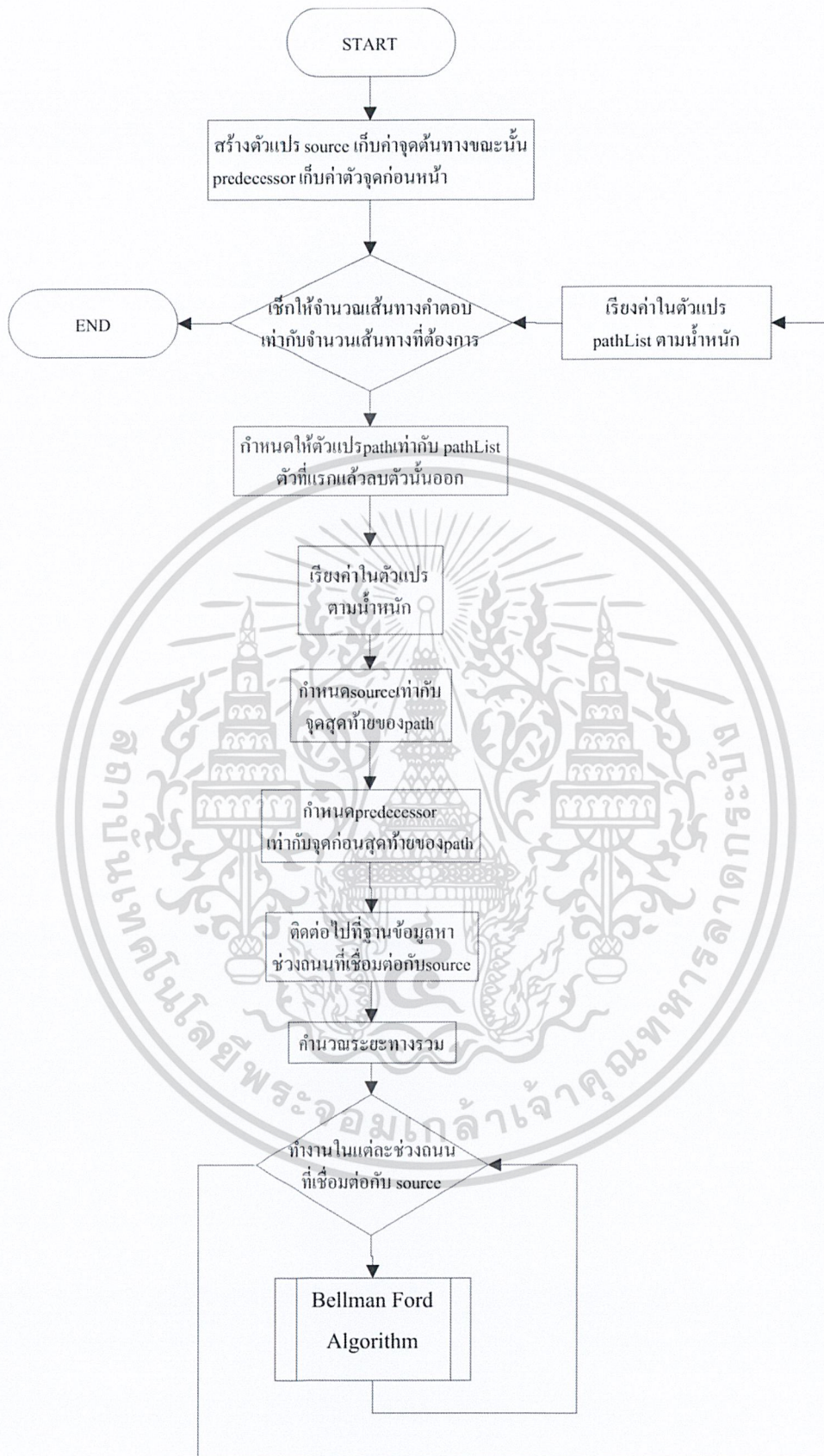
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่3.11 แสดงส่วนของการคำนวณหาเส้นทางในส่วนการตั้งค่าเริ่มต้นให้การค้นหาเส้นทาง

ในส่วนของรูปที่ 3.11 จะเป็นส่วนของการคำนวณหาเส้นทางในส่วนของการตั้งค่าเริ่มต้น โดยจะมีการติดต่อกับฐานข้อมูลเพื่อหาช่วงถนนที่เชื่อมต่อกับจุดต้นทางที่รับเข้ามาแล้วสร้างตัวแปร path ขึ้นมาเก็บเส้นทาง แล้วทำการวนลูปทำงานในทุกช่วงถนนที่เชื่อมต่อกับจุดต้นทาง โดยทำการเรียกใช้โมดูลเบลแมน ฟอร์ด อัลกอริทึมจากนั้นนำค่าในตัวแปร pathList มาเรียงค่าตามน้ำหนักจากค่าน้อยไปหาค่ามาก

ส่วนของรูปที่3.12 จะเป็นส่วนของการค้นหาเส้นทางแบบหลายเส้นทาง (Multipath) ทำหน้าที่ในการค้นหาเส้นทางออกมาเป็นหลายๆเส้นทางจนเท่ากับจำนวนเส้นทางที่ต้องการ โดยเริ่มค้นหาจากตัวแปร pathList ตัวแรก มีการสร้างตัวแปร source เพื่อเก็บค่าจุดต้นทางขณะนั้น และตัวแปร predecessor เพื่อเก็บค่าจุดที่เคยไปมาก่อนหน้า โดยจะติดต่อไปที่ฐานข้อมูลเพื่อหาช่วงถนนที่เชื่อมต่อกับจุดต้นทางขณะนั้น แล้วทำการวนลูปทำงานทุกช่วงถนนที่เชื่อมต่อกับจุดต้นทางขณะนั้น โดยทำการเรียกใช้โมดูลเบลแมน ฟอร์ด อัลกอริทึม จากนั้นเรียงค่าในตัวแปร pathList เพื่อไว้ทำการค้นหาเส้นทางต่อไปจนได้จำนวนเส้นทางเท่ากับจำนวนเส้นทางที่ต้องการ

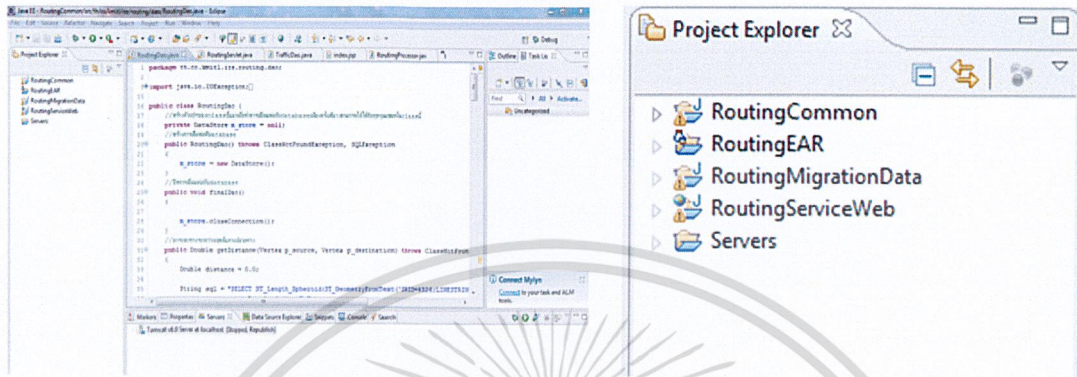


รูปที่3.12 แสดงส่วนของการคำนวณหาเส้นทางในส่วนการค้นหาเส้นทางแบบหลายเส้นทาง

## บทที่ 4

### ผลการดำเนินงาน

#### 4.1 ขั้นตอนวิธีการสร้างระบบ



รูปที่ 4.1 แสดงส่วนต่างๆของโปรแกรมค้นหาเส้นทางอ้างอิงสภาพจราจร

ในส่วนของรูปที่ 4.1 จะแสดงส่วนต่างๆของโปรแกรมค้นหาเส้นทางอ้างอิงสภาพจราจร โดยจะประกอบไปด้วย 4 จาวาโปรเจกต์ คือ จาวาโปรเจกต์ทั่วไป (RoutingCommon) เป็นจาวาโปรเจกต์ที่รวบรวมคลาสที่ถูกเรียกใช้บ่อย, จาวาโปรเจกต์เอ็นเตอร์ไพรส์แอปพลิเคชัน (RoutingEAR) เป็นจาวาโปรเจกต์ที่ใช้เชื่อมโยงโปรเจกต์อื่นให้เชื่อมโยงถึงกัน, จาวาโปรเจกต์การจัดการข้อมูล (RoutingMigrationData) เป็นจาวาโปรเจกต์ที่จัดเก็บคลาสที่ทำการจัดการกับข้อมูลในฐานข้อมูลตามที่ออกแบบไว้, จาวาโปรเจกต์เว็บเซอร์วิส (RoutingServiceWeb) เป็นจาวาโปรเจกต์ที่จัดการเกี่ยวกับเว็บเซอร์วิส และจาวาโปรเจกต์เซิร์ฟเวอร์ เป็นจาวาโปรเจกต์ที่จัดการเกี่ยวกับเซิร์ฟเวอร์

ในส่วนรูปที่ 4.2 จะแสดงส่วนประกอบของจาวาโปรเจกต์ทั่วไปคือแพ็คเกจ

th.co.kmitl.routing.conts ประกอบด้วยคลาส

- Encoding.java
- RoutingType.java

แพ็คเกจ th.co.kmitl.routing.dao ประกอบด้วยคลาส ที่จัดการเกี่ยวกับการเชื่อมต่อฐานข้อมูลของแต่ละคลาส คือ

- DataStore.java
- RoutingDao.java
- TrafficDao.java

แพ็คเกจ th.co.kmitl.routing.dto เป็นคลาสที่จัดการเกี่ยวกับชนิดของตัวแปรของระบบ ประกอบด้วย

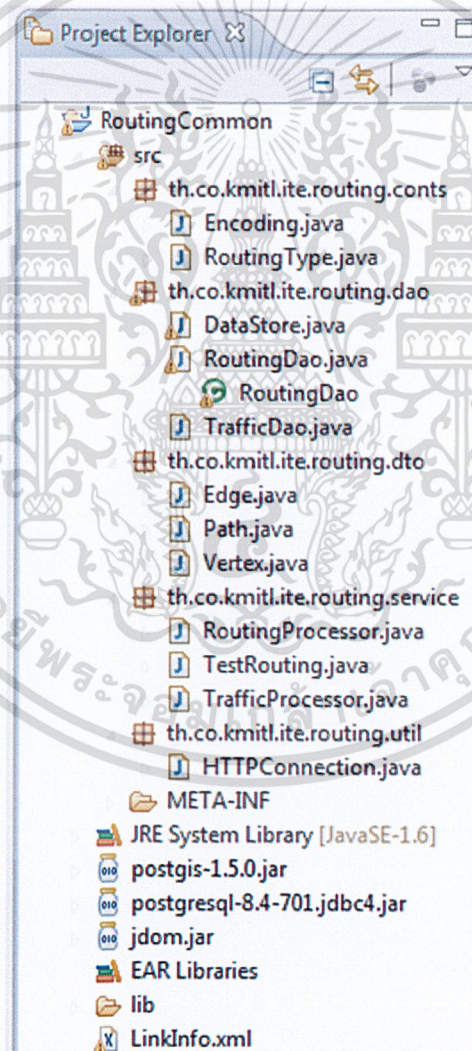
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

- Edge.java
- Path.java
- Vertex.java

แพ็คเกจ th.co.kmitl.routing.service เป็นคลาสที่จัดการเกี่ยวกับการค้นหาเส้นทาง และอ็อปเททสภาพความติดขัด ประกอบไปด้วยคลาส

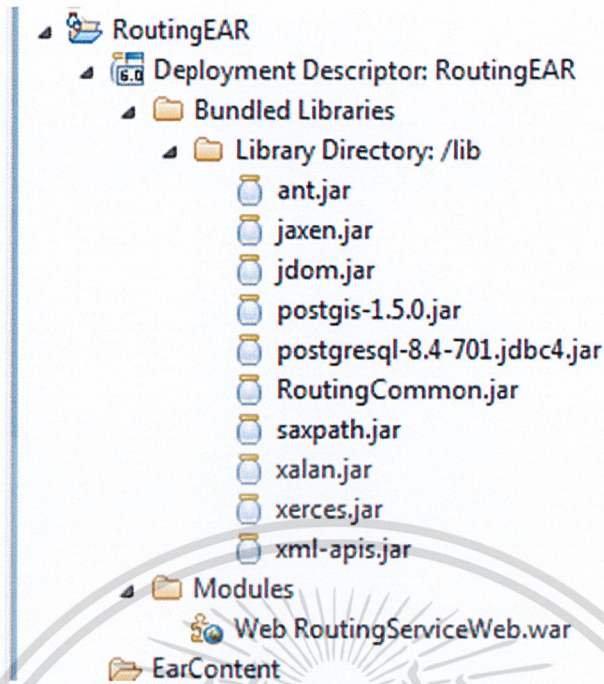
- RoutingProcessor.java
- TrafficProcessor.java

แพ็คเกจ th.co.kmitl.routing.util ประกอบไปด้วยคลาส HTTPConnection.java เป็นคลาสที่จัดการเกี่ยวกับวิธีเรียกใช้เว็บเซอร์วิส (GET,POST)



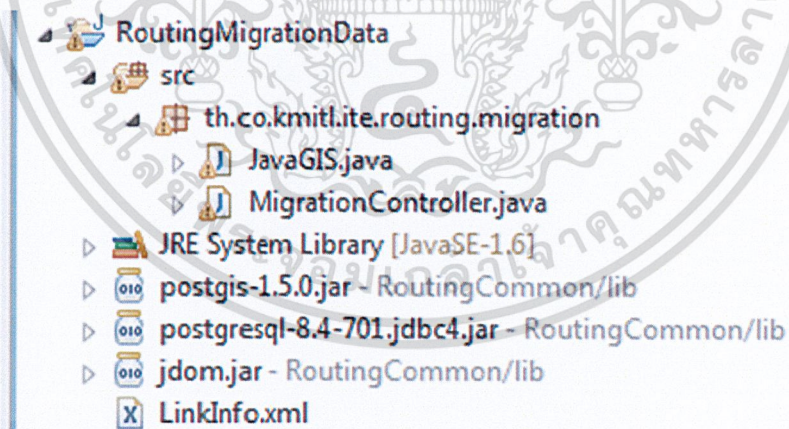
รูปที่ 4.2 แสดงส่วนของจาวาโปรเจกต์ทั่วไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



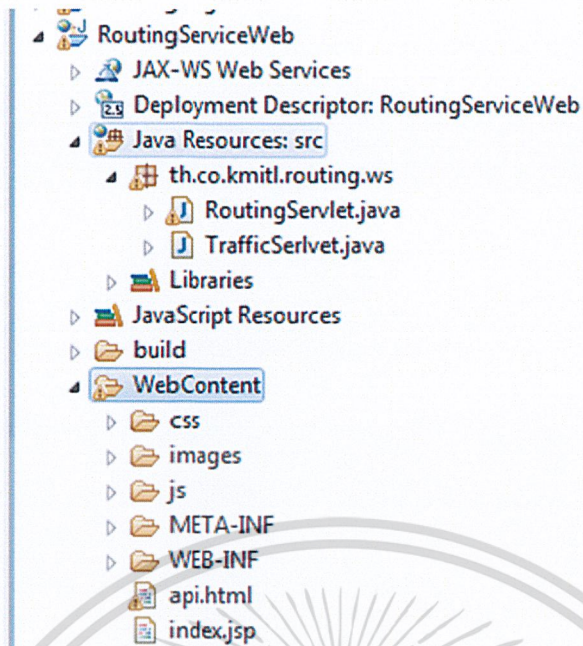
รูปที่ 4.3 แสดงส่วนของจาวาโปรเจกต์อินเทอร์เน็ตพอร์ทัลแอปพลิเคชัน

ส่วนของรูปที่ 4.3 จะแสดงส่วนประกอบของจาวาโปรเจกต์อินเทอร์เน็ตพอร์ทัลแอปพลิเคชันที่ใช้เชื่อมโยงโปรเจกต์อื่นให้เชื่อมโยงถึงกัน



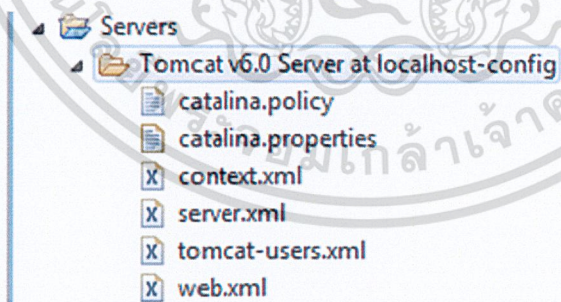
รูปที่ 4.4 แสดงส่วนของจาวาโปรเจกต์การจัดการข้อมูล

ส่วนรูปที่ 4.4 แสดงส่วนประกอบของจาวาโปรเจกต์การจัดการข้อมูลมีคลาส MigrationController.java ที่ทำเปลี่ยนข้อมูลตัวอย่างเป็นข้อมูลของระบบ



รูปที่ 4.5 แสดงส่วนของจาวาโปรเจกต์เว็บเซอร์วิส

ส่วนของรูปที่ 4.5 แสดงส่วนประกอบต่างๆของจาวาโปรเจกต์เว็บเซอร์วิสมีแพ็คเกจ th.co.kmitl.routing.ws ที่ประกอบไปด้วยคลาส RoutingServlet.java เป็นเซอร์วิสที่รับการร้องขอจากผู้ใช้งานโปรโตคอลเอชทีทีพีจัดการเกี่ยวกับการค้นหาเส้นทางและ TrafficServlet.java เป็นเซอร์วิสที่รับการร้องขอจากผู้ใช้งานโปรโตคอลเอชทีทีพีจัดการเกี่ยวกับสภาพความติดขัด



รูปที่ 4.6 แสดงส่วนของจาวาโปรเจกต์เซิร์ฟเวอร์

ส่วนของรูปที่ 4.6 จะแสดงส่วนประกอบของจาวาโปรเจกต์เซิร์ฟเวอร์คือโปรเจกต์ของเซิร์ฟเวอร์ทอมแคทเวอร์ชัน 6.0

## 4.2 ขั้นตอนการเรียกใช้งานเราท์ติ้งเอพีไอ (Routing API)

**Navigation Systems**  
based on Traffic Conditions

### Routing API

**RoutingServiceWeb :**  
web service สำหรับทำการค้นหาเส้นทางแบบหลายเส้นทางตามชนิดการค้นหา

**Route :**  
เป็นคำสั่งสำหรับแสดงเส้นทางจากจุดค้นหาไปยังปลายทาง

**URL :**  
http://localhost:8080/RoutingServiceWeb/route?src=...&dest=...&type=...&num=...

**HTTP Method :**  
get , post

**Formats :**  
XML

**Parameters :**

- \*src = ละติจูดของจุดเริ่มต้นทาง
- \*dest = ละติจูดของจุดปลายทาง
- \*type = ชนิดการค้นหาเส้นทาง มี 2 ชนิด คือ DISTANCE และ TRAFFIC
- \*num = จำนวนเส้นทางที่ต้องการ ตั้งแต่ 1 ถึง 5

**Example :**

```
http://localhost:8080/RoutingServiceWeb/route?src=13.761806832561135,100.53652373337403
&dest=13.766310593859009,100.53523627304688&type=DISTANCE&num=3
```

```
http://localhost:8080/RoutingServiceWeb/route?src=13.761806832561135,100.53652373337403
&dest=13.766310593859009,100.53523627304688&type=TRAFFIC&num=1
```

**Output Parameter:**

Paths = แสดงเส้นทางทั้งหมดได้

Path = แสดงรายละเอียดของเส้นทางทั้งหมดและเส้น

weight = แสดงน้ำหนักของแต่ละ Path ที่ชนิดการค้นหาเป็น DISTANCE ค่า weight จะเป็นระยะทาง

ถ้าชนิดการค้นหาเป็น TRAFFIC ค่า weight จะเป็นเวลาที่ใช้ในการเดินทาง

distance = แสดงระยะทางในแต่ละ Path

vertex = แสดงพิกัดในแต่ละเส้นทางทั้งหมด

lat = แสดงละติจูดในแต่ละ Vertex

lng = แสดงลองจิจูดในแต่ละ Vertex

**Output Response :**

```
DISTANCE : < Paths >
  < Path weight="902.6497445557272" distance="902.6497445557272" >
    < Vertex lat="13.761851451752186" lng="100.53649349125233" />
    < Vertex lat="13.766281685614244" lng="100.53543293255238" />
  < /Path >
  < Path weight="1085.9879266466364" distance="1085.9879266466364" >
    < Vertex lat="13.761851451752186" lng="100.53649349125233" />
    < Vertex lat="13.766281685614244" lng="100.53543293255238" />
  < /Path >
< /Paths >
```

```
TRAFFIC : < Paths >
  < Path weight="0.011283121806946593" distance="902.6497445557272" >
    < Vertex lat="13.761851451752186" lng="100.53649349125233" />
    < Vertex lat="13.764519255730296" lng="100.53774492527774" />
  < /Path >
  < Path weight="0.013574849083082957" distance="1085.9879266466364" >
    < Vertex lat="13.761851451752186" lng="100.53649349125233" />
    < Vertex lat="13.766281685614244" lng="100.53543293255238" />
  < /Path >
< /Paths >
```

รูปที่ 4.7 แสดงรายละเอียดของเอพีไอ

รายละเอียดของเอพีไอในรูปที่ 4.7 ในกรอบ A จะแสดงพารามิเตอร์ที่ใช้ในการเรียกเอพีไอ มีรายละเอียดดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Src คือละติจูดและลองจิจูดของจุดต้นทาง  
Dest คือละติจูดและลองจิจูดของจุดปลายทาง  
Type คือชนิดการค้นหาเส้นทาง  
Num คือจำนวนเส้นทางที่ให้ระบบแสดงผล

ในกรอบ B ของรูปที่ 4.7 จะแสดงเอาต์พุตพารามิเตอร์ มีรายละเอียดดังนี้

Paths คือเส้นทางที่หามาได้

Path คือรายละเอียดของเส้นทางที่หามาได้แต่ละเส้น

Weight คือแสดงน้ำหนักของแต่ละ Path ถ้าชนิดการค้นหาเป็น DISTANCE ค่า weight จะเป็นระยะทาง ถ้าชนิดการค้นหาเป็น TRAFFIC ค่า weight จะเป็นเวลาที่ใช้ในการเดินทาง

Distance คือระยะทางในแต่ละ Path มีหน่วยเป็นเมตร

Vertex คือพิกัดในแต่ละเส้นทางที่หามาได้

Lat คือละติจูดในแต่ละ Vertex

Lng คือลองจิจูดในแต่ละ Vertex

ตัวอย่างการเรียกใช้พีไอ

```
http://localhost:8080/RoutingServiceWeb/route?src=13.761808832561135  
100.53652373337403&dest=13.766310593859009  
100.53523627304688&type=DISTANCE&num=2
```

src มีค่า 13.761808832561135 100.53652373337403

dest มีค่า 13.766310593859009 100.53523627304688

type คือ DISTANCE

num คือ 2

ตัวอย่างการแสดงผลการเรียกพีไอ

```
<Paths>  
  <Path weight='902.6497445557272' distance='902.6497445557272' >  
    <Vertex lat='13.761851451752186' lng='100.53649349125233' />  
    ...  
    <Vertex lat='13.766281665614244' lng='100.53543293255238' />  
  </Path>  
  <Path weight='1085.9879266466364' distance='1085.9879266466364' >
```

```

<Vertex lat='13.761851451752186' lng='100.53649349125233' />
...
<Vertex lat='13.766281665614244' lng='100.53543293255238' />

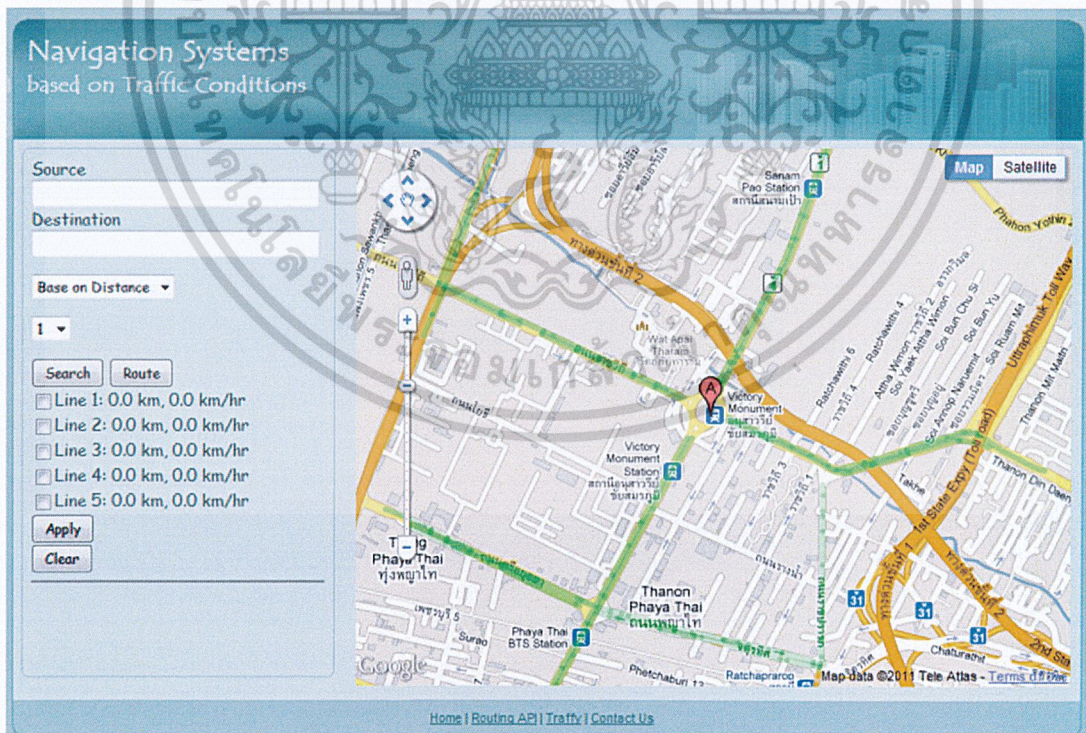
</Path>
</Paths>

```

ในผลลัพธ์นี้จะแสดงเส้นทางจำนวน 2 เส้นทาง โดยเส้นทางแรกนั้นจะมีค่า Weight = 902.6497445557272 มีค่า Distance = 902.6497445557272 ในเส้นทางที่สองจะมีค่า Weight = 1085.9879266466364 มีค่า Distance = 1085.9879266466364 และภายในก็จะมี Vertex หลาย ๆ Vertex ซึ่งในแต่ละ Vertex ก็จะเก็บค่าของ lat และ lng ของ Vertex นั้น ๆ

### 4.3 ขั้นตอนการทดสอบการค้นหาเส้นทาง

ในขั้นตอนนี้ผู้จัดทำได้ทำการสร้างเว็บแอปพลิเคชันขึ้นมาเพื่อทำการทดสอบการใช้งานเราท์ติงเอพีไอ



รูปที่ 4.8 หน้าแรกของเว็บไซต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

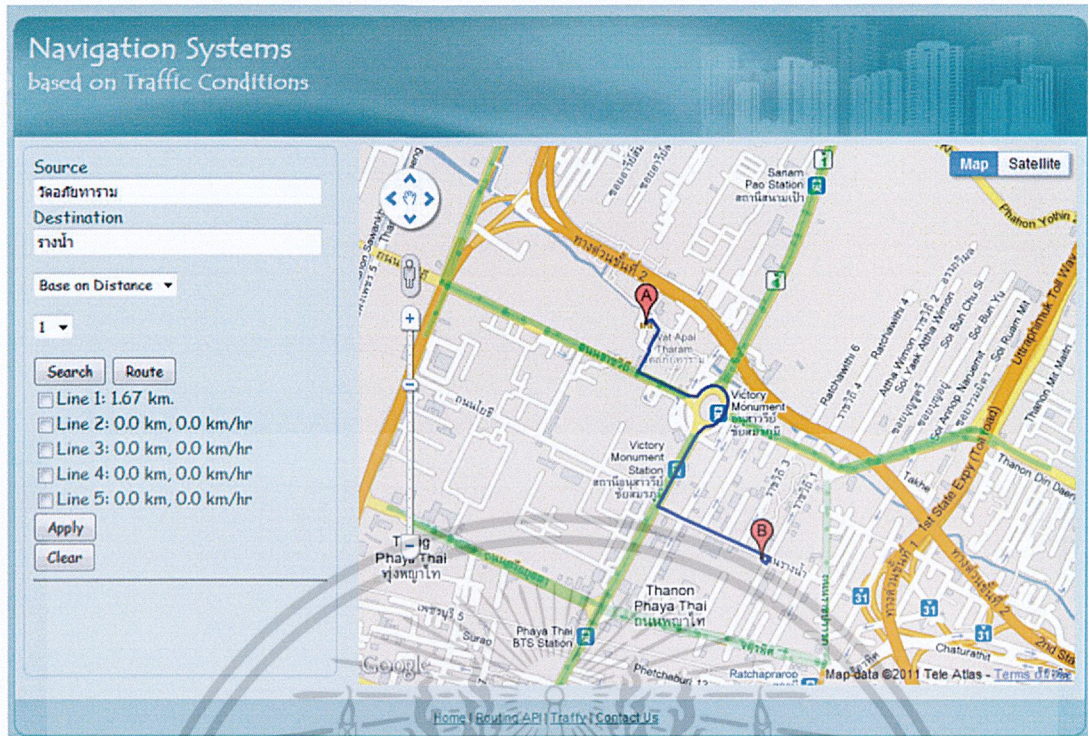
#### 4.3.1 การค้นหาเส้นทางตามระยะทาง

เมื่อเข้ามายังหน้าแรกของเว็บไซต์ ดังรูปที่ 4.8 รายละเอียดของหน้าเว็บไซต์จะประกอบไปด้วยช่อง Source สำหรับใส่จุดต้นทาง, ช่อง Destination สำหรับใส่จุดปลายทาง, ชนิดการค้นหาเส้นทาง, จำนวนเส้นทางที่ต้องการค้นหา (สูงสุดห้าเส้นทาง), ปุ่ม Search ใช้ในการค้นหาสถานที่จากช่อง Source และ Destination, ปุ่ม Route ใช้ในการค้นหาเส้นทาง หลังจากค้นหาสถานที่บนแผนที่แล้ว, ช่องสำหรับบอกระยะทาง อัตราเร็ว ของแต่ละเส้นทาง, ปุ่ม Apply ไว้สำหรับเลือกเส้นทาง, ปุ่ม Clear ใช้สำหรับเริ่มต้นการค้นหาใหม่อีกครั้ง

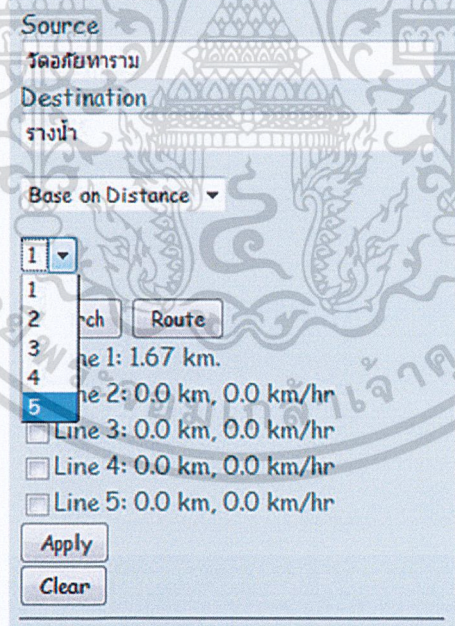


รูปที่ 4.9 แสดงการค้นหาเส้นทางตามสภาพจราจร

เมื่อผู้ใช้งานระบุจุดต้นทาง จุดปลายทาง, รูปแบบการค้นหา, ชนิดการค้นหาเส้นทางและจำนวนเส้นทางแล้ว ต่อจากนั้นผู้ใช้งานที่ปุ่ม Search เพื่อให้ระบบค้นหาสถานที่จากข้อมูลที่ใช้ได้ไว้ ระบบก็จะแสดงจุดต้นทางและจุดปลายทางบนแผนที่ หรือถ้าผู้ใช้ไม่ทราบชื่อของสถานที่ผู้ใช้สามารถใช้การปักหมุดในแผนที่เพื่อกำหนดจุดต้นทางและจุดปลายทางได้ ดังรูป 4.9 หลังจากนั้นกดที่ปุ่ม Route เพื่อให้ระบบทำการค้นหาเส้นทางและแสดงเส้นทางตามที่คุณได้เลือกไว้ ดังรูป 4.10 ที่กำหนดจุดต้นทางที่วัดเกษิหาราม, กำหนดจุดปลายทางที่ถนนรางน้ำ, เลือกชนิดการค้นหาเส้นทางแบบตามระยะทาง, จำนวนเส้นทางที่ต้องการแสดงผลคือ 1 เส้นทาง ผลลัพธ์การค้นหาเส้นทางที่ได้จะมีระยะทาง 1.67 กิโลเมตร



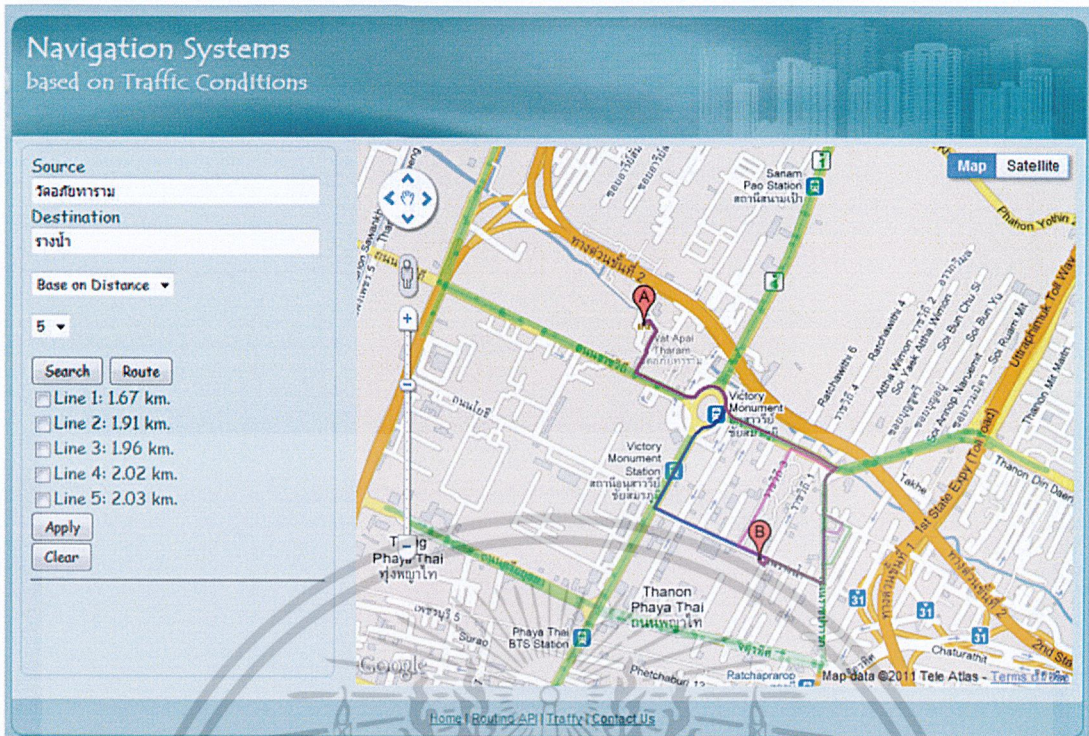
รูปที่ 4.10 ผู้ใช้เลือกสถานที่ วัดอภัยทวารามและรางน้ำ



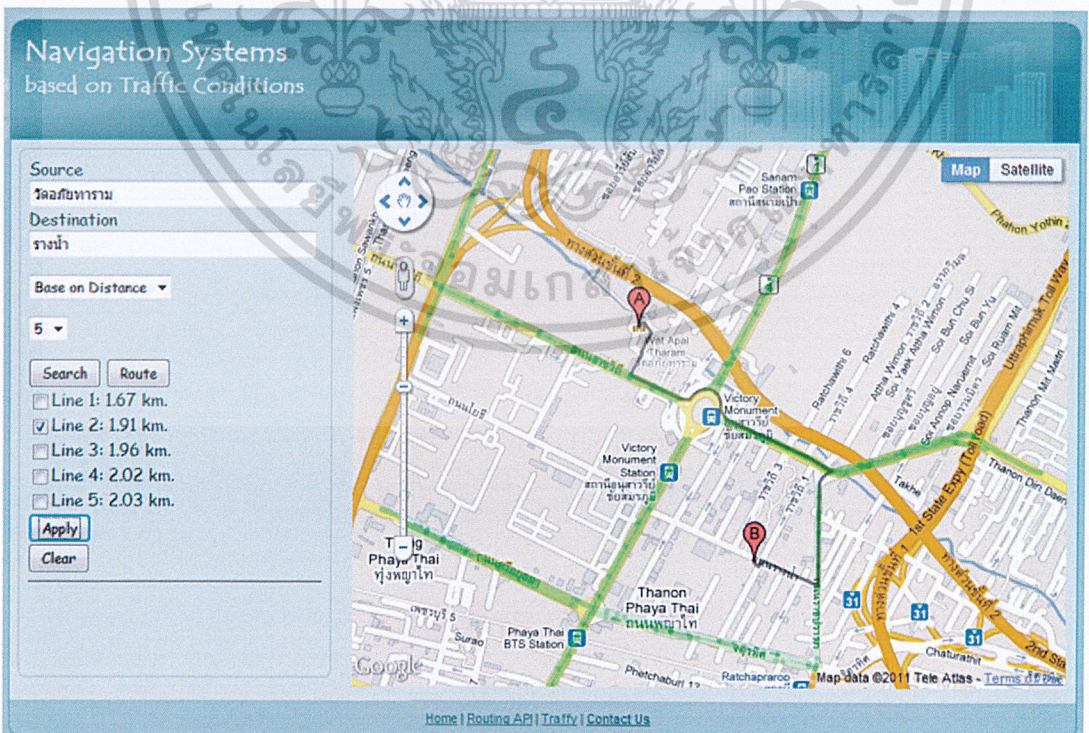
รูปที่ 4.11 ตัวอย่างการเลือกจำนวนเส้นทางที่ต้องการค้นหา

จากผลลัพธ์ที่ได้รูปที่ 4.10 หากผู้ใช้ต้องการเส้นทางการเดินทางมากกว่า 1 เส้นทาง ผู้ใช้สามารถเลือกจำนวนเส้นทางได้จากเมนูจำนวนเส้นทาง โดยเลือกจำนวนเส้นทางได้สูงสุด 5 เส้นทางดังรูปที่ 4.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 เมื่อเลือกเส้นทางที่ต้องการค้นหาจำนวน 5 เส้นทาง เมื่อเลือกจำนวนเส้นทางที่ต้องการ ระบบก็จะแสดงเส้นทางออกมาให้ตามจำนวนเส้นทางที่ผู้ใช้ได้เลือกไว้ โดยแต่ละเส้นทางจะบอกระยะทางของแต่ละเส้นทาง เช่น เส้นทางที่ 1 มีระยะทาง 1.67 กิโลเมตร เส้นทางที่ 2 มีระยะทาง 1.91 กิโลเมตร ดังรูป 4.12

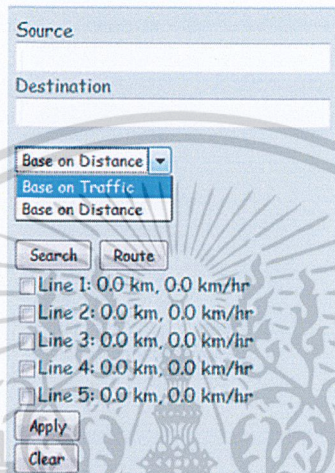


รูปที่ 4.13 การเลือกให้ระบบแสดงผลเส้นทางที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

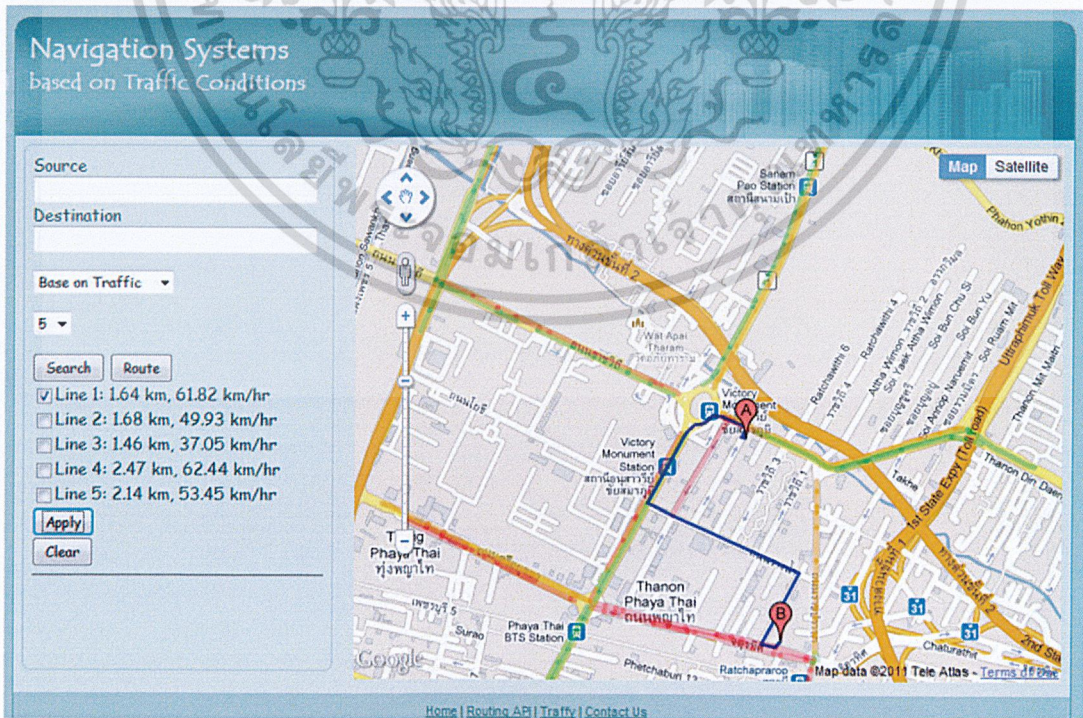
ผู้ใช้สามารถเลือกให้แสดงผลเฉพาะเส้นทางที่ผู้ใช้ต้องการได้ โดยในส่วนนี้ ถ้าเลือกชนิดการค้นหาเป็นแบบตามระยะทาง ระบบจะแสดงระยะทางของแต่ละเส้นทางให้ผู้ใช้ได้ทราบ เพื่อช่วยในการตัดสินใจให้ผู้ใช้ ในการเลือกเส้นทาง โดยในการเลือกเส้นทางให้ ให้ตั้งเครื่องหมายถูกหน้าเส้นทางที่ต้องการให้แสดงผล จากนั้นกดที่ปุ่ม Apply ระบบก็จะแสดงเส้นทางเฉพาะเส้นทางที่ผู้ใช้ได้เลือกไว้ ดังรูป 4.13

#### 4.3.2 การค้นหาเส้นทางตามสภาพจราจร



รูปที่ 4.14 เลือกเป็นค้นหาเส้นทางตามสภาพจราจร

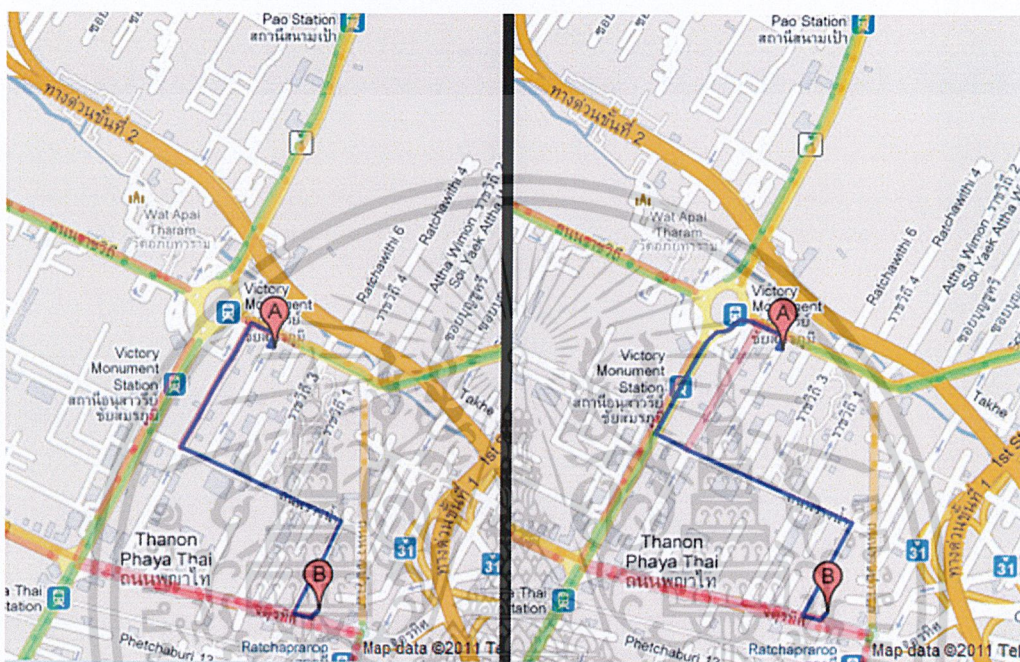
ผู้ใช้สามารถเลือกการค้นหาเส้นทางตามสภาพจราจรได้ โดยทำการเลือกชนิดการค้นหาเส้นทางเป็นแบบ Base on Traffic ดังรูปที่ 4.14



รูปที่ 4.15 แสดงผลการค้นหาเส้นทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากในรูปที่ 4.15 นี้ เส้นสีต่าง ๆ จะแสดงถึงสภาพการจราจรของถนนเส้นนั้น โดยสีแดง จะแสดงถึงเส้นทางที่มีการจราจรติดขัดมาก เส้นสีส้มจะแสดงสภาพการจราจรติดขัดที่รองลงมา และสีเขียวจะแสดงสภาพการจราจรที่คล่องตัว ในรูปนี้แสดงการค้นหาเส้นทางตามสภาพ การจราจร ในแต่ละเส้นทางจะบอกระยะทางและอัตราเร็วของเส้นทางนั้น ๆ



รูปที่ 4.16 เปรียบเทียบการค้นหาเส้นทาง

ในรูปที่ 4.16 จะเป็นการเปรียบเทียบผลการค้นหาเส้นทาง โดยทางด้านซ้ายจะเป็นการค้นหาเส้นทางตามระยะทางและด้านขวาจะเป็นการค้นหาเส้นทางตามสภาพจราจร จะเห็นได้ว่าการค้นหาเส้นทางตามสภาพจราจรนั้นจะหลีกเลี่ยงเส้นทางที่มีมีการจราจรติดขัดไปยังเส้นทางอื่น

## บทที่ 5

# บทวิจารณ์และสรุป

### 5.1 ผลที่ได้รับ

- ระบบการค้นหาเส้นทางอ้างอิงสภาพจราจรมีประสิทธิภาพ และง่ายต่อการใช้งาน
- สามารถเพิ่มทางเลือกให้กับระบบการค้นหาเส้นทาง
- คณะผู้จัดทำได้รับความรู้เกี่ยวกับการนำวิธีการหาเส้นทางที่สั้นที่สุดมาประยุกต์ใช้งานและการพัฒนาเว็บเซอร์วิสและเว็บแอปพลิเคชัน

### 5.2 ปัญหาที่พบ

- การติดต่อไปเรียกใช้ ทราฟฟี่เอพีไอยังมีปัญหา เนื่องจากความไม่ชำนาญของคณะผู้จัดทำ ในเรื่องของภาษาที่ใช้ในการติดต่อกับระบบทราฟฟี่ทำให้เกิดปัญหาในการทดลองเรียกผลลัพธ์จากระบบทราฟฟี่เพื่อนำมาประยุกต์ใช้กับ โครจงานนี้
- ความเร็วในการค้นหาเส้นทางยังไม่เป็นที่น่าพอใจ
- ฐานข้อมูลเส้นทางที่ใช้ในการทดลองในโปรเจกต์นี้มีค่าน้อยไปทำให้การค้นหาเส้นทางบางจุดเกิด ลูปขึ้น ทำให้ทำการค้นหาเส้นทางไม่ได้

### 5.3 แนวทางในการแก้ไขปัญหา

- ศึกษาภาษาที่ใช้ให้มีความเข้าใจมากขึ้น และได้ทำการติดต่อไปยังส่วนที่ดูแลระบบของทราฟฟี่เอพีไอเพื่อขอคำปรึกษาในส่วนนี้
- เพิ่มตารางในฐานข้อมูลเพื่อเก็บข้อมูลผลลัพธ์ของการค้นหาเส้นทางชนิดตามระยะทาง ที่เคยค้นหาแล้ว ทำให้ความเร็วในการค้นหาจุดที่เคยค้นหาแล้วเร็วขึ้นมาก

### 5.4 แนวทางในการพัฒนาต่อ

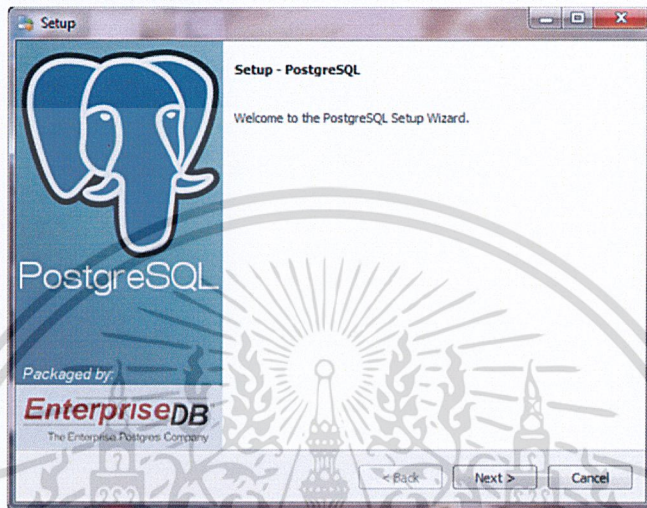
- พัฒนาระบบค้นหาเส้นทางอ้างอิงสภาพการจราจรของโทรศัพท์เคลื่อนที่
- พัฒนาในส่วนของอัลกอริทึมให้มีความเร็วและความถูกต้องเพิ่มขึ้น





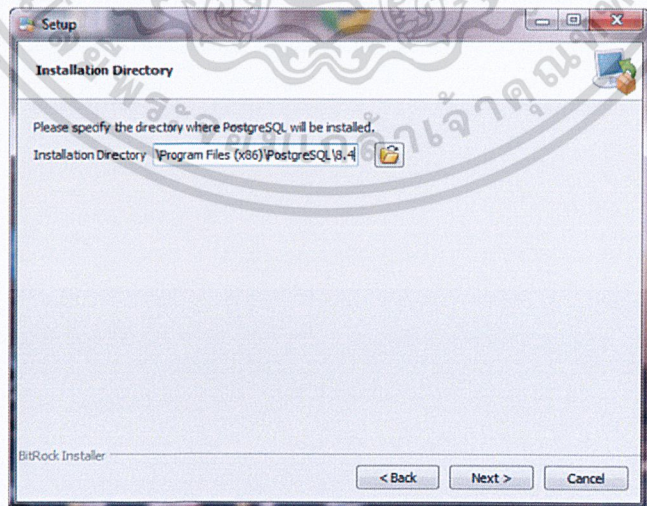
## การติดตั้งโพสเกรสเควสคิวแอล(PostgreSQL)

ดาวน์โหลดตัวติดตั้งโพสเกรสเควสคิวแอลจาก <http://www.postgresql.org/download/> ชื่อไฟล์ postgresql-8.4.4-1-windows.exe และเปิดตัวติดตั้งโปรแกรมดังรูป ก.1 และคลิก Next



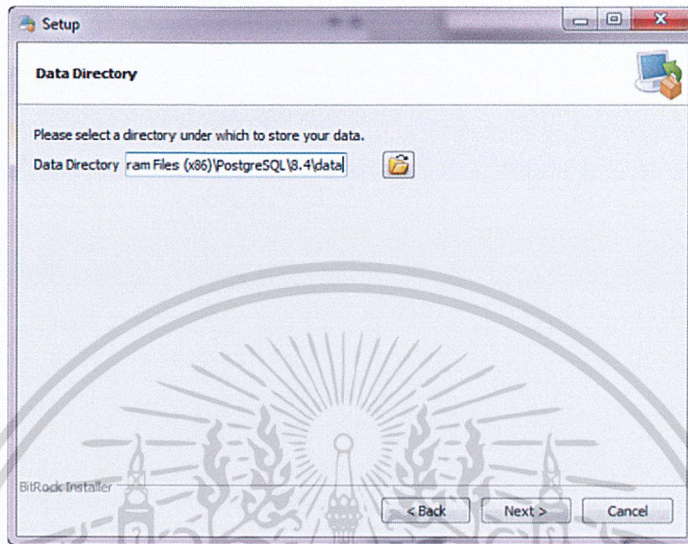
รูปที่ ก.1 การติดตั้งโปรแกรมโพสเกรสเควสคิวแอลขั้นตอนที่ 1

ในรูป ก.2 เป็นการกำหนดไดเรกทอรีในการติดตั้งโปรแกรม โดยกำหนดที่ C:\Program Files (x86)\PostgreSQL\8.4 จากนั้นคลิก Next



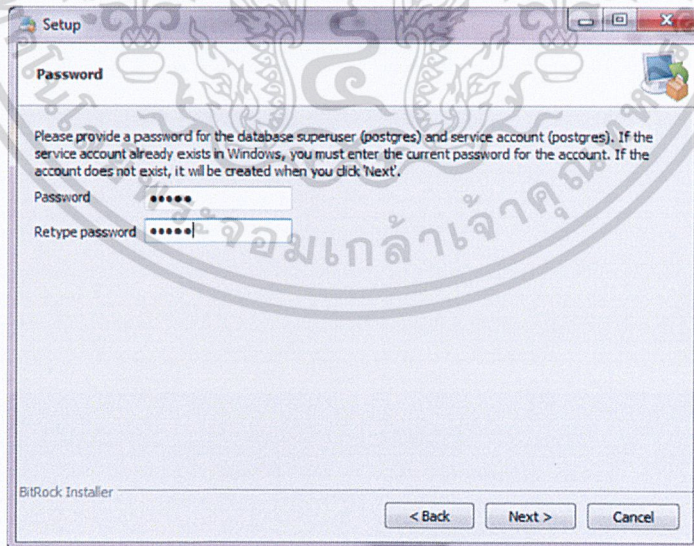
รูปที่ ก.2 การติดตั้งโปรแกรมโพสเกรสเควสคิวแอลขั้นตอนที่ 2

รูปที่ ก.3 เป็นการกำหนดไดเรกทอรีสำหรับเก็บข้อมูลของฐานข้อมูล กำหนดที่ C:\Program Files (x86)\PostgreSQL\8.4\data จากนั้นคลิก Next



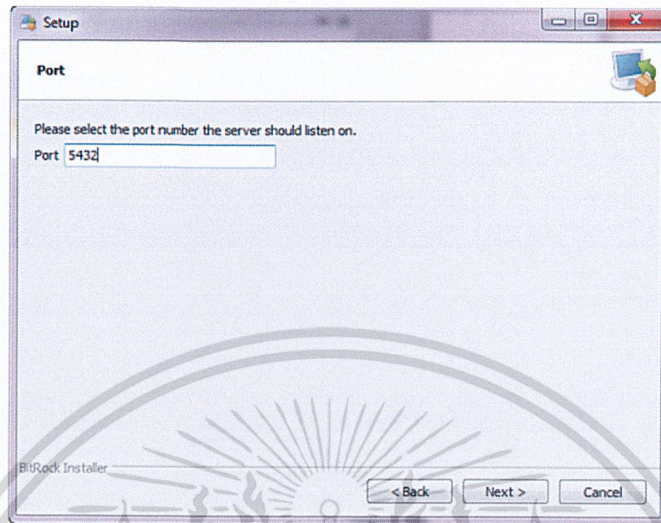
รูปที่ ก.3 การติดตั้งโปรแกรมโพสเกรสเกรสคิวแอลขั้นตอนที่ 3

รูปที่ ก.4 เป็นการตั้งรหัสผ่านสำหรับผู้ไ้ postgresQL แล้ว คลิก Next



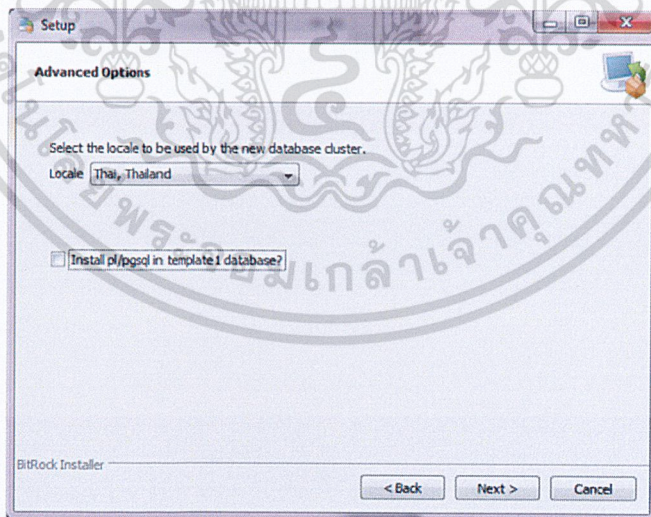
รูปที่ ก.4 การติดตั้งโปรแกรมโพสเกรสเกรสคิวแอลขั้นตอนที่ 4

รูปที่ ก.5 ระบุพอร์ตสำหรับฐานข้อมูล โดยกำหนดพอร์ตหมายเลข5432 จากนั้นคลิก Next



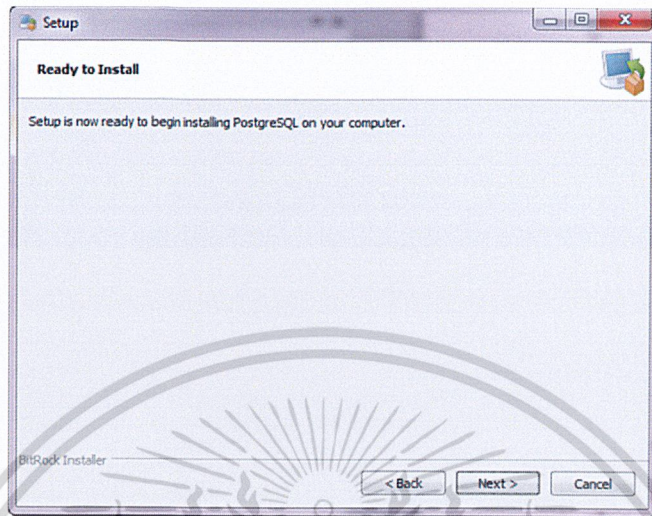
รูปที่ ก.5 การติดตั้งโปรแกรมโพสเกรสเกรสคิวแอลขั้นตอนที่ 5

รูปที่ ก.6 เลือกภาษาและประเทศของผู้ใช้ โดยเลือก Thai, Thailand และ ไม่ต้องเลือกให้ติดตั้งเทมเพลตของดาต้าเบส จากนั้นคลิก Next



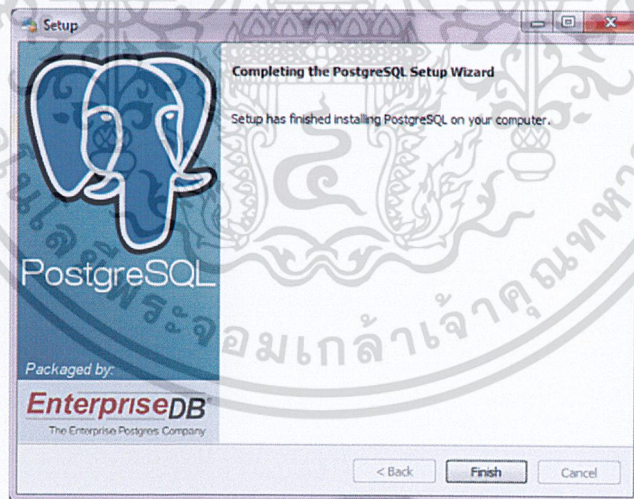
รูปที่ ก.6 การติดตั้งโปรแกรมโพสเกรสเกรสคิวแอลขั้นตอนที่ 6

คลิก Next เพื่อเริ่มการติดตั้ง ดังรูปที่ ก.7



รูปที่ ก.7 การติดตั้งโปรแกรมโพสเกรสเกรสคิวแอลขั้นตอนที่ 7

รูปที่ ก.8 การติดตั้งเสร็จเรียบร้อยคลิก Finish



รูปที่ ก.8 การติดตั้งโปรแกรมโพสเกรสเกรสคิวแอลขั้นตอนที่ 8

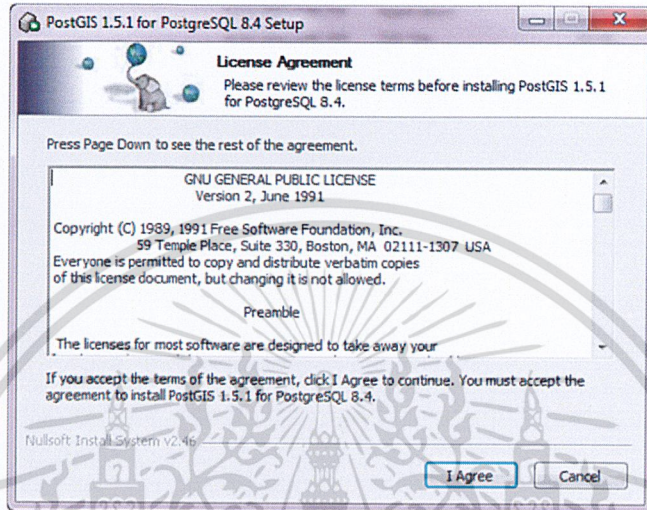


ภาคผนวก ข.

การติดตั้งโปรแกรมโพสจิส

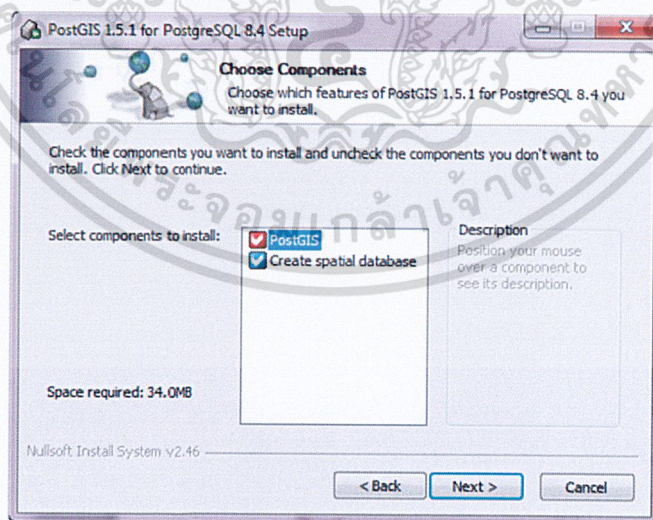
## การติดตั้งโพสจีเอส(PostGis)

ดาวน์โหลดโพสจีเอสจาก <http://postgis.refrains.net/download/windows/#windbinaries> เลือกที่ไฟล์ PostGIS 1.5.2 release for PostgreSQL 8.4 และเปิดตัวติดตั้งโปรแกรม คลิก I Agree ดังรูปที่ ข.1



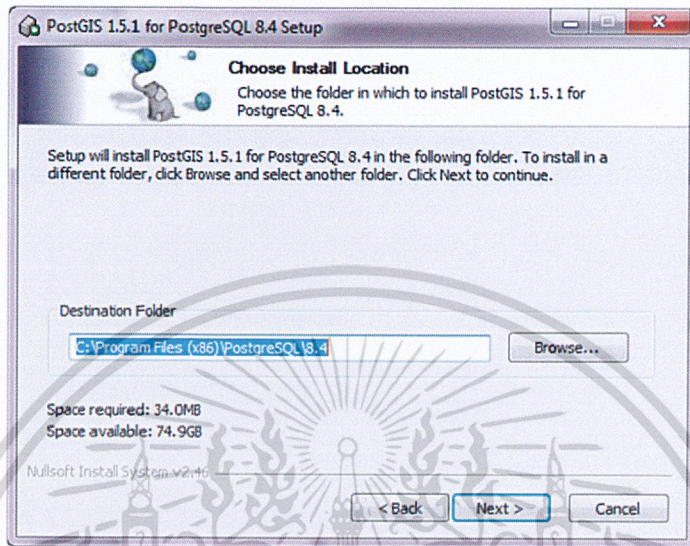
รูปที่ ข.1 การติดตั้งโปรแกรมโพสจีเอสขั้นตอนที่ 1

รูปที่ ข.2 เลือกคอมโพเนนท์ในการติดตั้ง จากนั้นคลิก Next



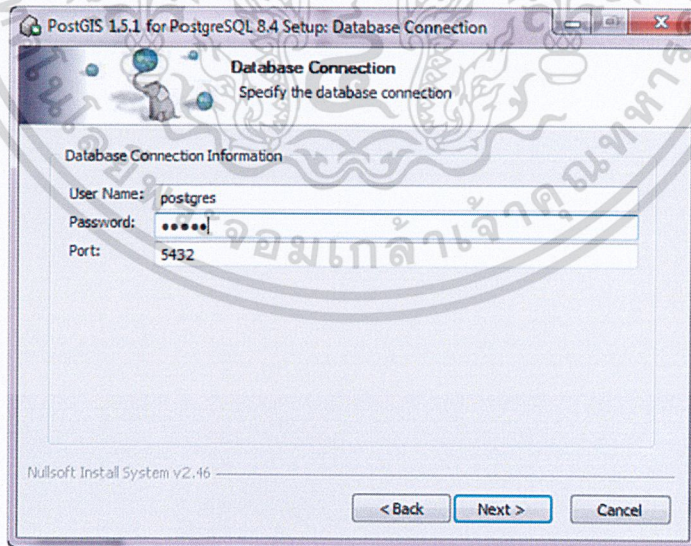
รูปที่ ข.2 การติดตั้งโปรแกรมโพสจีเอสขั้นตอนที่ 2

รูปที่ ข.3 เป็นการกำหนดไดเรกทอรีสำหรับติดตั้งโดยเลือกที่ C:\Program Files (x86)\PostgreSQL\8.4 จากนั้นคลิก Next



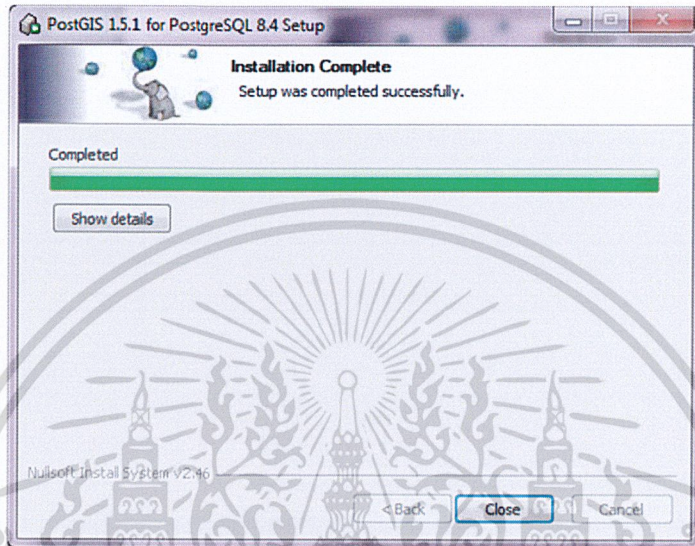
รูปที่ ข.3 การติดตั้งโปรแกรมโพสจีสขั้นตอนที่ 3

รูปที่ ข.4 ระบุชื่อ รหัสผ่าน และพอร์ต ที่ใช้ในโพสเกรสเกรสคิวเอดแล้วคลิก Next



รูปที่ ข.4 การติดตั้งโปรแกรมโพสจีสขั้นตอนที่ 4

รูปที่ ข.5 การติดตั้งเสร็จเรียบร้อย คลิก Close



รูปที่ ข.5 การติดตั้งโปรแกรมโพสจิสขั้นตอนที่ 6



ภาคผนวก ค.  
คู่มือการติดตั้งโปรแกรมทอมแคท 6.0

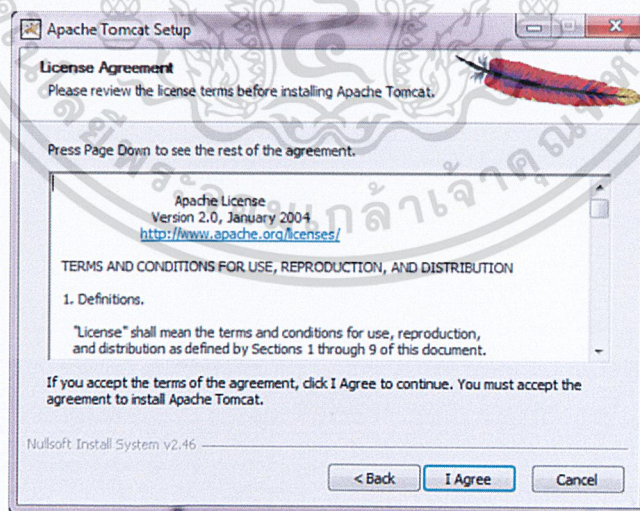
## การติดตั้งทอมแคท 6.0 (Tomcat 6.0)

ดาวน์โหลดโปรแกรมติดตั้งจาก <http://tomcat.apache.org/download-60.cgi> ชื่อไฟล์ Tomcat 6.0 และเปิดโปรแกรมติดตั้ง ดังรูปที่ ค.1 แล้วคลิก Next



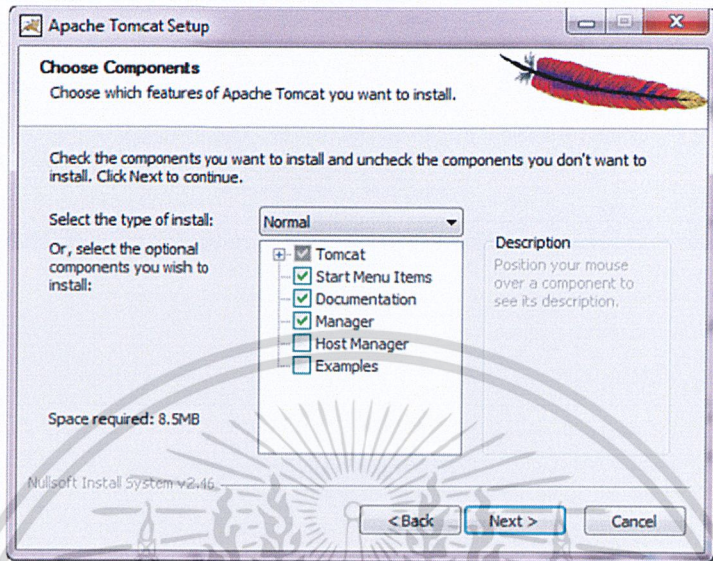
รูปที่ ค.1 การติดตั้งโปรแกรมทอมแคท 6.0 ขั้นตอนที่ 1

คลิก I Agree เพื่อยอมรับข้อตกลงการใช้โปรแกรม ดังรูปที่ ค.2



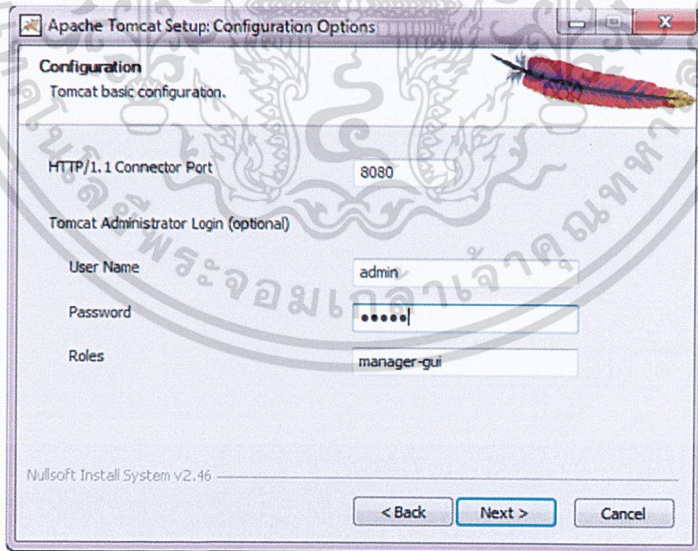
รูปที่ ค.2 การติดตั้งโปรแกรมทอมแคท 6.0 ขั้นตอนที่ 2

รูปที่ ค.3 เป็นการเลือกคอมโพเนนท์ที่ต้องการติดตั้ง แล้วคลิก Next



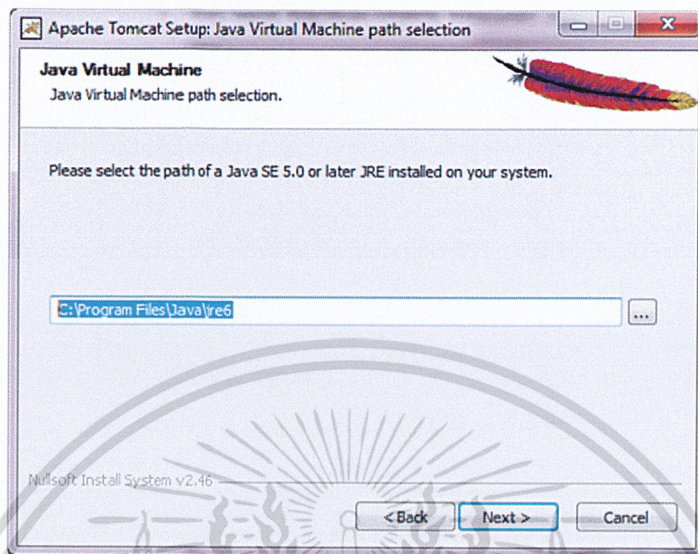
รูปที่ ค.3 การติดตั้งโปรแกรมทอมแคท 6.0 ขั้นตอนที่ 3

รูปที่ ค.4 ตั้งค่าพอร์ทหมายเลข 8080 ชื่อผู้ใช้และรหัสผ่าน จากนั้นคลิก Next



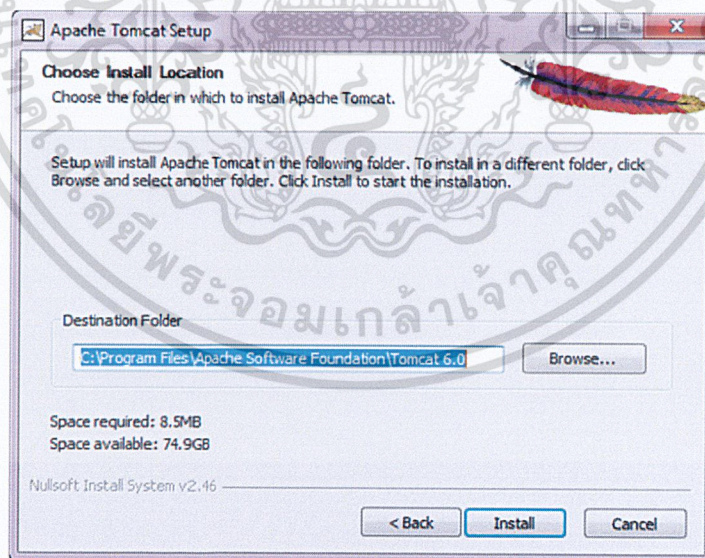
รูปที่ ค.4 การติดตั้งโปรแกรมทอมแคท 6.0 ขั้นตอนที่ 4

รูปที่ ค.5 เลือกไดเรกทอรีที่ Java Runtime ติดตั้งอยู่ ในที่นี้คือ C:\Program Files\Java\jre6 จากนั้นคลิก Next



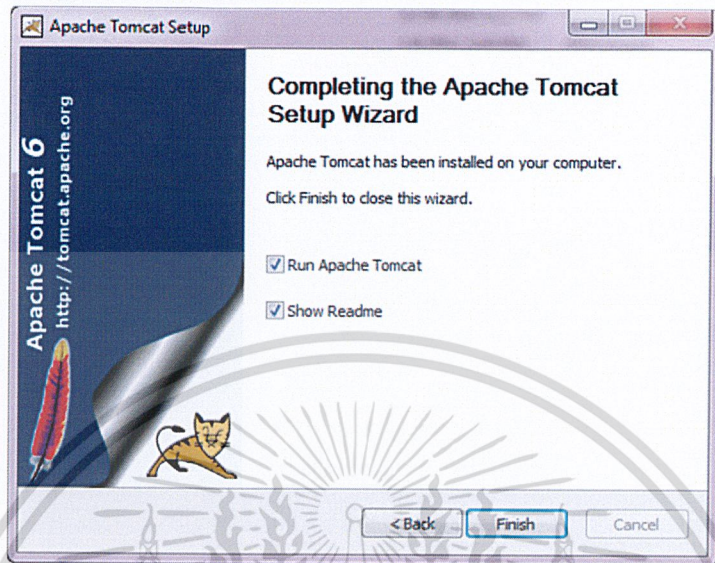
รูปที่ ค.5 การติดตั้งโปรแกรมทอมแคท 6.0 ขั้นตอนที่ 5

รูปที่ ค.6 เป็นการกำหนดไดเรกทอรีสำหรับลงทอมแคท แล้วคลิก Install



รูปที่ ค.6 การติดตั้งโปรแกรมทอมแคท 6.0 ขั้นตอนที่ 6

คลิก Finish เพื่งจบการติดตั้ง ดังรูปที่ ค.7



รูปที่ ค.7 การติดตั้งโปรแกรมทอมแคท 6.0 ขั้นตอนที่ 7

## บรรณานุกรม

- [1] REST vs. SOAP Web Services. [ออนไลน์]. เข้าถึงได้จาก :  
<http://www.blognone.com/node/5968> (วันที่ค้นข้อมูล: 1 กันยายน 2553).
- [2] Web Service Testing [ออนไลน์]. เข้าถึงได้จาก :  
<http://wiki.nectec.or.th/setec/Knowledge/WebServicesTesting#REST> (วันที่ค้นข้อมูล: 1 กันยายน 2553).
- [3] Shortest Path Problem [ออนไลน์]. เข้าถึงได้จาก :  
[http://en.wikipedia.org/wiki/Shortest\\_path\\_problem](http://en.wikipedia.org/wiki/Shortest_path_problem) (วันที่ค้นข้อมูล: 1 กันยายน 2553).
- [4] PostgreSQL [ออนไลน์]. เข้าถึงได้จาก :<http://www.postgresql.org/>(วันที่ค้นข้อมูล : 1 กันยายน 2553)
- [5] “การนำเสนอผลการศึกษามาตรฐานระบบภูมิสารสนเทศตามมาตรฐาน ISO/TC21”  
[ออนไลน์]. เข้าถึงได้จาก :  
<http://thaisdi.gistda.or.th/document/54/Part%205%20-%20ISO%2019125-2.pdf> (วันที่ค้นข้อมูล: 11 กุมภาพันธ์ 2554).
- [6] JSP คืออะไร? [ออนไลน์]. เข้าถึงได้จาก :  
<http://thaiprogrammer.blogspot.com/2008/09/jsp.html> (วันที่ค้นข้อมูล: 11 กุมภาพันธ์ 2554)